FLEXERA SOFTWARE™

AdminStudio®

AdminStudio 10.0 SP2

User Guide

Legal Information

Book Name: AdminStudio 10.0 SP2 User Guide
Part Number: ADS-1000-UGSP2
Product Release Date: 15 September 2011

Copyright Notice

Copyright © 2001–2011 Flexera Software, Inc. and/or InstallShield Co. Inc. All Rights Reserved.

This product contains proprietary and confidential technology, information and creative works owned by Flexera Software, Inc. and/or InstallShield Co. Inc. and their respective licensors, if any. Any use, copying, publication, distribution, display, modification, or transmission of such technology in whole or in part in any form or by any means without the prior express written permission of Flexera Software, Inc. and/or InstallShield Co. Inc. is strictly prohibited. Except where expressly provided by Flexera Software, Inc. and/or InstallShield Co. Inc. in writing, possession of this technology shall not be construed to confer any license or rights under any Flexera Software, Inc. and/or InstallShield Co. Inc. intellectual property rights, whether by estoppel, implication, or otherwise.

All copies of the technology and related information, if allowed by Flexera Software, Inc. and/or InstallShield Co. Inc., must display this notice of copyright and ownership in full.

Trademarks

Flexera Software, AdminStudio, FlexEnabled, FlexLM, FlexNet, FlexNet Connect, FlexNet Manager, FlexNet Publisher, InstallAnywhere, InstallShield, InstallShield Professional, and PackageExpert are registered trademarks or trademarks of Flexera Software, Inc. and/or InstallShield Co. Inc. in the United States of America and/or other countries. All other brand and product names mentioned herein are the trademarks and registered trademarks of their respective owners.

Restricted Rights Legend

The software and documentation are “commercial items,” as that term is defined at 48 C.F.R. §2.101, consisting of “commercial computer software” and “commercial computer software documentation,” as such terms are used in 48 C.F.R. §12.212 or 48 C.F.R. §227.2702, as applicable. Consistent with 48 C.F.R. §12.212 or 48 C.F.R. §227.2702-1 through 227.7202-4, as applicable, the commercial computer software and commercial computer software documentation are being licensed to U.S. government end users (A) only as commercial items and (B) with only those rights as are granted to all other end users pursuant to the terms and conditions set forth in the Flexera Software, Inc. standard commercial agreement for this software. Unpublished rights reserved under the copyright laws of the United States of America.

Contents

Part 1: Getting Started 57

1 AdminStudio 10.0 SP2 Help Library 59

What's New in AdminStudio 10.0 SP2. 63

AdminStudio Editions and Components 63

Activating AdminStudio. 68

Evaluating AdminStudio 69

Upgrading Your Product Edition. 72

Using Help 73

Contacting Us. 75

2 Getting Started with AdminStudio. 77

Getting Started Tab 78

Test for Application Compatibility Tab. 79

Perform Portfolio Application Compatibility Reporting (Migration Project Scoping). 79

Pinpoint Specific Compatibility Issues 80

Fix Compatibility Issues 80

Migrate to Application Virtualization Tab. 81

Identify Packages to Virtualize 81

Convert to Virtual Formats 82

Test and Distribute Converted Packages. 82

Migrate to Windows Installer Tab. 83

Repackage Legacy Package 83

Import Into Application Catalog 84

Test and Distribute Repackaged Applications 84

Set Up Infrastructure Tab 85

Create/Connect to an Application Catalog 85
Configure Virtual Machines 86
Set E-Mail Notification Settings 86
Configure Compatibility Solver 87

3 Using the AdminStudio Interface 89

Configuring the AdminStudio Interface 90
Launching AdminStudio Applications 91
Specifying the Shared AdminStudio Application Catalog 92
Specifying the AdminStudio Shared Location 92
Setting E-Mail Notification Settings 93
Setting the Workflow Task Help Page Location 94
Configuring How Often AdminStudio Checks for Updates 95
Configuring AdminStudio to Stay on Top 95
Generating a Debug Log for AdminStudio 95
Specifying the AdminStudio Enterprise Server URL 96
Specifying the Configuration Manager Web Console URL 96

Working with Tools 97
Adding New Tools to the Tools Gallery 97
Editing Properties for an Existing Tool 98
Adding Command-Line Configurations for an Existing Tool 98
Modifying Command-Line Configurations for an Existing Tool 99
Deleting Command-Line Configurations from an Existing Tool 99
Associating Tools with Tasks 100
Running Associated Tools in Projects 100
Deleting Existing Tools 101
Limiting Tool Accessibility 101

Workflows and Projects 101
Creating and Editing Workflows 102
Creating New Workflows 103
Renaming Workflows 103
Filtering Workflows 103
Deleting Workflows 103
Creating New Tasks 104
Modifying Task Properties 104
Creating Notes for a Task 104
Renaming Tasks 105
Reordering Tasks 105
Associating Help Files with Tasks 105
Deleting Tasks 106
Adding New Tools from the Process Template Editor 106
Creating and Using Projects 106
Creating Projects with the New Workflow Project Wizard 107
Filtering Projects 107

Executing Projects . . . 107
Running Associated Tools in Projects . . . 108
Deleting Projects . . . 108
Saving Workflow and Project Changes . . . 108
Workflow Project Example: Using the New Workflow Project Wizard . . . 109
Workflows, Projects, and Permissions . . . 111
Frequently Asked Questions . . . 112
AdminStudio Interface Reference . . . 113
AdminStudio Start Page . . . 114
Tools Tab . . . 115
Process Assistants Tab . . . 116
Report Center Tab . . . 117
Enterprise Server Tab . . . 117
Workflow Manager Tab . . . 118
Process Template Editor . . . 118
Tasks . . . 118
AdminStudio Menus and Toolbar . . . 119
Dialog Boxes . . . 122
About AdminStudio Dialog Box . . . 122
Add New Tool Dialog Box . . . 123
Command Line Properties Dialog Box . . . 124
Options Dialog Box . . . 124
Application Catalog Tab . . . 125
Locations Tab . . . 126
Updates Tab . . . 126
Quality Tab: Customer Experience Improvement Program . . . 126
Virtual Packages Tab . . . 127
Notification Settings Tab . . . 129
Tool Properties Dialog Box . . . 130
Properties Tab . . . 130
Configuration Tab . . . 132
Wizards . . . 133
Add Tool Wizard . . . 133
Welcome Panel . . . 133
Tool Properties Panel . . . 134
Command-Line Configurations Panel . . . 134
New Workflow Project Wizard . . . 135
Welcome Panel . . . 135
Workflow Selection Panel . . . 135
Source Package Panel . . . 135
Target Directory and File Name Panel . . . 136
Log Files . . . 136

Part 2: Managing Users, Roles, Permissions, Application Catalogs. . 137

4 Managing Users, Directory Services, and User Logins. 139

Managing Users and Groups 139
Creating a New User Account 140
Importing Directory Services Users and Groups 140
Disabling a User or Group Account 141
Deleting a User Account 142
Managing Directory Services Configurations 143
Managing Directory Services Connections 143
Adding a Directory Service Connection 143
Editing an Existing Directory Service Connection 146
Deleting a Directory Service Connection 147
Managing Directory Services Attributes 147
Managing User Logins. 148
Login Methods 149
IIS Authentication Methods Dialog Box 150
Using AdminStudio Enterprise Server Account Login 151
Using Domain Account Login. 151
Using Single Sign-On Login 152
Using Guest Account Login 153
Setting Up a Guest Account 153
Logging in as a Guest. 155
Setting the Session Timeout Value 156
Users/Groups and Directory Services Reference 156
User Administration Page 157
User Details View 158
Directory Services User/Group Add View 159
Directory Services Page 160
Directory Services List Page 160
Add Directory Service Connection View. 161
Directory Services Attributes Administration Page. 163
Add Directory Service Attributes View 164

5 Managing Roles and Permissions. 165

AdminStudio and Workflow Manager Roles and Permissions. 165
Role Permission Lists 166
AdminStudio Client Tools Permissions 166
AdminStudio Enterprise Server Tools Permissions 169
Workflow Manager Permissions 171
System Roles 174
Superuser Role: AMSSuper 174
Default System Roles 175

Default System Users 176
Creating a New Role 177
Editing an Existing User Role 177
Copying an Existing Role 178
Deleting a Role 178
Roles Reference 179
Role Administration Page 179
Copy Role View 180
Role Details View 181

6 Managing Application Catalog Databases 183
Application Manager and ConflictSolver 184
Functionality Comparison 185
Accessing the Tools 186
Displaying ConflictSolver Options in Application Manager 187
Creating and Connecting to Application Catalogs 187
Managing Application Catalogs 188
About AdminStudio Application Catalogs 188
Application Manager Organization and Structure 189
Creating a New Application in Application Manager 190
Overview of Application Catalogs 191
Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog 192
Using the AdminStudio Quick Start Evaluation Experience 193
Connecting to an Existing Application Catalog 193
Connecting AdminStudio Client Tools to a Standalone Application Catalog 194
Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog 195
Creating New Application Catalogs 196
Creating New Application Catalogs Using the AdminStudio Interface 196
Creating New Application Catalogs Using Scripts 197
Specifying a Default AdminStudio Application Catalog 199
Using the AdminStudio Interface 200
Editing the Shared AdminStudio.ini File 201
Searching an Application Catalog 202
Disconnecting from an Application Catalog 203
Upgrading an Existing Application Catalog 204
Organizing Your Application Catalog Using Groups 204
Adding Groups 205
Organizing Applications in Application Manager 205
Deleting Application Manager Groups 206
Editing Group Properties 206
Copying Packages to Multiple Groups 206
Moving Applications, OS Snapshots, and Groups 207
Deleting Packages 207

Importing Data 207
Importing a Single Windows Installer or Virtual Package 211
Importing a Single Package 211
About Windows Installer Packages (.msi) 213
Performing an Ad-Hoc Import of Transform Files or Patch Files 214
About Transforms (.mst) 215
About Patches (.msp) 215
Importing a Directory of Windows Installer and/or App-V Packages. 216
Importing Windows Installer and/or App-V Packages From Microsoft Configuration Manager 217
Importing Virtual Packages 218
Importing a Virtual Package Without Its Source Windows Installer Package 219
Importing a Virtual Package During the Import of its Source Windows Installer Package 219
Importing a Virtual Package After the Import of its Source Windows Installer Package 221
Manually Associating a Virtual Package with a Windows Installer Package 223
Deleting a Virtual Package Association 223
Importing Merge Modules 224
About Merge Modules (.msm) 224
Importing OS Snapshots 225
About OS Snapshots (.osc) 226
Importing Marimba NCP Files 226
About Marimba Native Channel Packager Files (.ncp) 227
Importing Other Setup Types 228
About Other Setup Types 229
Using Duplicate Package Identifiers 229
Viewing Application Catalog Enhanced Reporting 233
Viewing the Application Readiness Dashboard 233
Application Deployment Type Breakdown Chart 235
Windows 7 Application Compatibility Chart 235
App-V Virtualization Readiness Chart 237
Windows Installer Best Practices Chart 239
Virtualization Best Practices Chart 243
Viewing the Application Compatibility Dashboard 246
Connecting to the Compatibility Solver Database 249
Viewing the Virtualization Readiness Dashboard. 249
Virtualization Readiness Status in Application Catalog vs. Automated Application Converter 254
Manually Run Virtualization Readiness Tests 256
Viewing Other Summary Reports 256
Viewing the Shared Extensions Report 256
Viewing the Error Category Breakdown Report 257
Viewing the App-V Dependencies/Dependents Reports 259
Exporting a Report in PDF, Excel, or Word Format 262
Creating Customizable Reports 263
Viewing Package Data 264
Viewing Windows Installer Package Data 264

Viewing and Editing Package Extended Attributes . . . 265
Using Extended Attributes . . . 265
Extended Attribute Description File . . . 266
Integrating Package Extended Attribute Data with an Application Request . . . 268
Viewing Windows Installer Package Files/Components . . . 269
Viewing Windows Installer Package INI File Changes . . . 269
Viewing Windows Installer Package Registry Information . . . 270
Viewing Windows Installer Package Shortcuts . . . 270
Viewing Windows Installer Package Virtualization Suitability Issues . . . 271
Viewing Windows Installer Package Application Compatibility Issues . . . 273
Viewing Windows Installer Package Merge Modules . . . 274
Viewing Package Catalog History . . . 275
Viewing Package Tables . . . 276
Viewing App-V Package Data . . . 276
Viewing App-V Package History . . . 277
Viewing App-V Package Dependencies . . . 278
Viewing App-V Package Files and Directories . . . 279
Viewing App-V Package Registry . . . 280
Viewing App-V Package Shortcuts . . . 281
Viewing App-V Package File Type Associations . . . 282
Viewing App-V Package Environment Variables . . . 283
Using the Software Repository . . . 284
Enabling the Software Repository in Application Catalogs . . . 285
Enabling the Software Repository in a New Application Catalog . . . 286
Enabling the Software Repository in an Existing Application Catalog . . . 286
Methods to Import Packages into the Software Repository . . . 287
Importing a Package into the Software Repository Using the Import Wizard . . . 287
Adding a Package to the Software Repository by Overwriting Original Package . . . 288
Software Repository Auto Import Method . . . 288
Identifying Software Repository Packages in Application Manager . . . 289
Using Version Management Features . . . 290
Checking-Out and Checking-In Packages . . . 291
Cancel Check Out . . . 291
Getting a Copy of the Latest Version of a Package . . . 292
Viewing Package Version History . . . 292
Software Repository Integration into Other AdminStudio Tools . . . 293
Sharing Application Catalog Data . . . 295
Merging Application Catalogs Using the Merge Wizard . . . 299
Automatically Importing Packages . . . 300
Automatically Importing Packages in a Remote Application Catalog . . . 301
Linking to a Remote Application Catalog . . . 302
Editing a Link to a Remote Application Catalog . . . 304
Manually Updating a Linked Package . . . 305
Deleting a Link to a Remote Application Catalog . . . 306

Viewing Remote Package Links on the Application Manager Product View *306*
Automatically Importing Packages from a Network Directory 307
Linking to a Network Directory *307*
Editing a Link to a Network Directory *309*
Forcing an Import of Packages in a Network Directory *310*
Deleting a Link to a Network Directory *310*
Package Auto Import and Duplicate Package Names 310
Replicating Application Catalogs 311
About Application Catalog Replication 312
Benefits of Using Application Catalog Replication *312*
Creating and Managing Publisher and Subscriber Application Catalogs *313*
Configuring the AdminStudio Shared Location for Replication *315*
Initial Configuration Checklist *315*
Publication/Subscription Lifecycle 316
Specialized User Scenarios 318
Creating and Managing Publications 319
Managing Publications *320*
Creating a New Publication *321*
Publishing a Publication *323*
Setting a Publication Schedule *324*
Editing a Publication *324*
Editing a Publication Access List *325*
Deleting a Publication *326*
Creating and Managing Subscriptions 326
Managing Subscriptions *327*
Creating a New Subscription *328*
Enabling/Disabling a Subscription *329*
Manually Updating a Subscription *330*
Deleting a Subscription *331*
Using Microsoft SQL Server to Perform Merge Replication of Application Catalogs 332
Taking OS Snapshots 338
OS Snapshot Best Practices 338
Configuring OS Snapshot Analysis Options 339
Capturing an OS Snapshot 340
Reference 340
Application Manager Interface 340
Menus and Toolbar 341
Product View Icons 344
Merge Modules View Icons 347
Context Menus 348
Output Window 352
User Permissions in Application Manager 353
Application Manager Views 354
Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards 354

Application Readiness Dashboard 355
Application Compatibility Dashboard 367
Virtualization Readiness Dashboard 371
Error Category Breakdown 372
Shared Extensions Report 374
App-V Dependencies Report 375
App-V Dependents Report 376
App-V Recommended Dependencies Report 376
Customizable Reports 377
Application Catalog View (Professional Edition) 377
Group View 378
Product View 378
Extended Attributes View 381
Files/Components View 381
INI File Changes View 383
Registry View 383
Shortcuts View 384
Virtualization Suitability View 384
Application Compatibility View 386
Merge Modules View 388
Catalog History View 388
Tables View 389
App-V Package View 389
App-V History View 391
App-V Dependencies View 392
App-V Files/Directories View 392
App-V Registry View 393
App-V Shortcuts View 394
App-V File Type Associations View 396
App-V Environment Variables View 397
App-V Tables View 398
Citrix / ThinApp Package View 399
OS Snapshot View 400
Extended Attributes View for OS Snapshots 400
Files View for OS Snapshots 400
INI File Changes View for OS Snapshots 401
Registry View for OS Snapshots 401
Shortcuts View for OS Snapshots 402
Tables View for OS Snapshots 402
Other Setup Types View 402
Extended Attributes View for Other Setup Types 403
Marimba NCP Files Views 403
Extended Attributes View for Marimba NCP Files 404
Files View for Marimba NCP Files 404

INI File Changes View for Marimba NCP Files 405
Registry View for Marimba NCP Files 405
Shortcuts View for Marimba NCP Files 406
Tables View for Marimba NCP Files 406
Patches Tab Views 407
Patches Group View 407
New Patches Group View 408
Group View of a Selected Group 409
Patch View 409
Merge Module Views 411
All Merge Modules View 411
Merge Module View 411
Merge Module | Components View 412
Merge Module | Dependency View 412
Merge Module | Exclusion View 412
Merge Module | Files View 413
Merge Module | Products View 413
Dialog Boxes 413
Application Catalog Properties Dialog Box 414
Associate with Workflow Manager Application Dialog Box 415
Change Enterprise Server Password Dialog Box 415
Command-Line Parameters Dialog Box 416
Connect Application Catalog Dialog Box 416
Enterprise Server Tab 417
Standalone Tab 418
Recent Tab 419
Default Application Catalog Dialog Box 420
Extended Attribute Property Dialog Box 421
Find Dialog Box 421
Login Required Dialog Box 423
Group Properties Dialog Box 423
Package Auto Import Dialog Box 424
Package Auto Import Properties (Remote Application Catalog) Dialog Box 425
Package Auto Import Properties (Network Directory) Dialog Box 427
Publication Manager Dialog Box 428
Publication Processing Dialog Box 429
Publication Properties Dialog Box 429
Select Application Catalog Dialog Box 430
Select AdminStudio Enterprise Server URL Dialog Box 430
Subscription Manager Dialog Box 430
Task Scheduling Dialog Box 432
Test Connection Dialog Box 433
Virtual Package Association Dialog Box 433
Wizards 434

Application Catalog Wizard 434
Welcome Panel 434
Specify Database Information Panel 435
Select Software Repository Location Panel 436
Creating Application Catalog Panel 437
Import Wizard 437
Welcome Panel 438
Select Package Source Panel 439
Connect to a Microsoft Configuration Manager Server 440
Select Packages Panel 441
File Selection Panel 442
Folder Selection Panel 443
Associate Virtual to MSI Packages Panel 445
MST Source Information Panel 446
MSP Source Information Panel 446
MSM Source Information Panel 447
Virtual Package Import Panel 448
Target Package Information Panel 448
Target Package Information Panel (Software Repository) 449
Target Package Information Panel (Transforms/Patches) 449
Import Options Panel 450
Additional non-MSI Import Options Panel 450
Destination Group Panel 450
Summary Panel 450
Package Auto Import Wizard 451
Welcome Panel 451
Package Auto Import Type Selection Panel 452
Remote Link Application Catalog Panel 452
Folder Panel 453
Groups Panel 453
Destination Group Panel 453
Schedule Panel 454
Summary Panel 454
Merge Wizard 454
Welcome Panel 455
Source Application Catalog Panel 455
Groups Panel 455
Progress Panel 456
OS Snapshot Wizard 456
Welcome Panel 456
Project Information Panel 456
Analyzing Panel 457
OS Snapshot Summary Panel 457
Analysis Options Dialog Box 457

ISSnapshot.ini File 457
Publication Wizard 458
Welcome Panel 458
Publication Details Panel 459
Publication Data Options Panel 459
Publication Access List Panel 460
Publication Packages Panel 460
Publication Summary Panel 461
Subscription Wizard 461
Welcome Panel 462
Subscription Details Panel 462
Subscription Selection Panel 463
Subscription Summary Panel 463
Upgrade Wizard 463
Database Server Permissions 464
Application Manager Command-Line Functionality 466
Specifying User Interface Mode via Command Line 469
Using a Configuration File 469
Application Manager Configuration File 469
Using a Configuration File with Command-Line Options 479
Importing 479
Applying Transforms and Patches During Command-Line Import 479
Importing Multiple Windows Installer Packages Simultaneously 480
Importing Multiple Merge Modules Simultaneously 480
Simultaneously Importing Windows Installer Packages and Merge Modules 480
Using the Command Line to Import All Packages in a Directory 480
Running Import Silently 481
Creating a Log File During Command-Line Import 481
Connecting to Standalone Application Catalogs 482
Connecting to a Specific Standalone Application Catalog Using Command-Line Options 482
Creating Shortcuts to Specific Standalone Application Catalogs 482
Application Catalog Replication Command Line Functionality 484
Virtualization Suitability Issues 485

Part 3: Repackaging and Customizing Installations 487

7 Repackaging Legacy Installations Using the Repackaging Wizard 489

About Repackaging 490
Purpose of Repackaging Applications 490
Supported Legacy Installation Types 491
Repackaging 64-Bit Applications 492
Repackaging Options Comparison 493
Repackaging Wizard Best Practices 494

About Repackaging on Clean Systems496
Alternate-Language Repackaging on Clean Machines .496
Including the InstallScript Engine With a Windows Installer Package .496
Repackaging Methods .497
Installation Monitoring Method .497
Snapshot Method .498
InstallShield Professional Logging Method .498
Configuring Repackager to Ensure Optimal Installation Capture .500
Launching Repackager Remotely .501
Installing Repackager on a Clean Machine .503
Repackaging Legacy Installations Using the Repackaging Wizard .504
Repackaging Using the Installation Monitoring Method .505
Step 1: Selecting the Repackaging Method .505
Step 2: Excluding Processes (Optional) .507
Step 3: Collecting Product Information .508
Step 4: Adding Additional Setup Programs (Optional) .509
Step 5: Set Target Project Information .510
Step 6: Set Capture Settings (Optional) .511
Step 7: Beginning the Repackaging Process .513
Repackaging Using the Snapshot Method .515
Performing Multiple Step Snapshot Repackaging .516
Step 1: Selecting the Repackaging Method .516
Step 2: Initial Analysis .518
Step 3: Install Setup and Make Manual System Changes .519
Step 4: Entering Product Information .519
Step 5: Set Target Project Information .520
Step 6: Set Capture Settings (Optional) .521
Step 7: Beginning the Repackaging Process .523
Performing Single Step Snapshot Repackaging .525
Step 1: Selecting the Repackaging Method .525
Step 2: Collecting Product Information .527
Step 3: Set Target Project Information .529
Step 4: Set Capture Settings (Optional) .529
Step 5: Beginning the Repackaging Process .531
Repackaging Using the InstallShield Professional Logging Method .534
Repackaging an InstallScript MSI Setup to a Basic MSI Setup .538
Running the Repackaging Wizard from the Command Line .539
Repackaging a Windows Installer (.msi) Package .540
Repackaging Wizard Reference .545
Repackaging Wizard .546
Welcome Panel .546
Method Selection Panel .547
Snapshot Method Panel .549
Collect Product Information Panel .550

InstallScript MSI Identified Panel . . . 552
InstallShield Professional Setup Panel . . . 554
Set Target Project Information and Capture Settings Panel . . . 554
InstallScript MSI Conversion Output Panel . . . 555
Repackaging Panel . . . 556
Summary Panel . . . 558
Additional Repackaging Wizard Dialog Boxes . . . 559
Additional Setup Programs Dialog Box . . . 559
Setup Information Dialog Box . . . 560
Excluded Processes Dialog Box . . . 560
Analysis Options Dialog Box . . . 561
Repackaging Wizard Command-Line Options . . . 562
Reboot Handling in the Repackaging Wizard . . . 565

8 Converting Legacy Installations Using the Repackager Interface . . . 567

About the Repackager Interface . . . 569
Launching the Repackager Interface . . . 569
Setting Repackager Options . . . 569
Selecting Data Display Colors . . . 570
Specifying Additional Merge Module Directories . . . 570
Controlling the Display of ICE Validation Warnings . . . 571

Creating Repackager Projects . . . 571
Converting Legacy Installations Using the Repackager Interface . . . 572
Converting Repackager 3.x Output Files . . . 573
Converting a Microsoft SMS Project to a Repackager Project . . . 573
Converting Novell ZENworks Projects . . . 573
Converting a Novell ZENworks Project Using the Repackager Interface . . . 574
Converting Multiple Novell ZENworks Projects Using the Command Line . . . 575
Converting WinINSTALL Projects . . . 577
Converting Wise Installation Projects . . . 577
Converting InstallShield Professional Log Files . . . 577

Working With Repackager Projects . . . 578
Building an InstallShield Editor Project . . . 578
Building a Windows Installer Package . . . 581
About the Context.msi File . . . 586
Configuring Advanced Conversion Options . . . 587
Automatically Generating a Virtual Application During Repackager Project Build . . . 589
Viewing Repackager Project Properties . . . 592
Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project . . . 593
Creating a Setup Capture Report for a Project . . . 594
Saving Repackager Projects . . . 595
Opening InstallShield Editor from Repackager . . . 596

Isolating Windows Installer Packages . . . 596

About Application Isolation 597
Isolating Windows Installer Packages Using Application Isolation Wizard 597
About Assemblies 598
About Manifests 598
About Digital Certificates 599
Setting Isolation Options 600
Specifying Manifest Options 601
Selecting the Assembly Type *601*
Specifying the Assembly Naming Conventions *602*
Setting Digital Signature Options for Shared Assemblies 602
Building an Isolated Windows Installer Package 603
Configuring Exclusions 603
Configuring Exclusions Using Repackager 604
Excluding Files 604
Excluding All Files in a Directory 605
Excluding Directories and Subdirectories 605
Excluding Registry Keys 605
Excluding Registry Values 606
Excluding .ini Files 606
Excluding .ini File Sections 606
Excluding Shortcuts 607
Excluding All Shortcuts in a Directory 607
Excluding Shortcuts from Subdirectories 607
Specifying the External Configuration File 608
Modifying External Configuration Files 609
Configuring Exclusions Using the Exclusions Editor 609
Exclusions and Repackager 610
Exclusions and the OS Snapshot Wizard 610
Launching Exclusions Editor 610
Excluding Files 612
Excluding Files with Specific Extensions 612
Excluding Directories 613
Editing Existing File Exclusions 613
Removing File Exclusions 614
Excluding .ini Files 614
Excluding Sections from .ini Files 615
Editing Existing .ini File Exclusions 616
Removing .ini File Exclusions 616
Excluding Registry Data 617
Editing Existing Registry Exclusions 617
Removing Registry Exclusions 618
Repackaging and Anti-Virus Software 618
Scanning InstallShield Professional Setups for Additional Data 619
Creating an InstallShield Editor Template to Use Within Repackager 621

Repackager Interface Reference 626
Repackager Interface 627
Repackager Start Page 628
Menus and Toolbar 630
Dialog Boxes 632
About Repackager Dialog Box *633*
Create Report Dialog Box *633*
Isolation Options Dialog Box *634*
Options Dialog Box *637*
Password Required Dialog Box *638*
Project Properties Dialog Box *639*
WinINSTALL Conversion Dialog Box *641*
Repackager Views 642
Captured Installation View *642*
Files and Folders View *645*
Registry Entries View *646*
Shortcuts View *648*
INI Files View *649*
Deleted Files View *650*
Deleted Registry Entries View *651*
Repackaged Output View *652*
Package Information View *657*
Advanced Settings View *658*
SmartScan Wizard 660
Welcome to the SmartScan Wizard Panel 661
Original InstallShield Professional Setup Panel 662
Scanning Panel 663
Setup Feature Tree Panel 663
Scanning Media Panel 664
Setup Intent Wizard 665
Welcome Panel 665
Scanning Project Panel 666
Results Panel 666
VMware Repackaging Wizard 667
Welcome Panel 667
VMware Virtual Machines Panel 667
Exclusions Editor Interface 668
Menus 668
Files Tab 669
.ini Files Tab 670
Registry Tab 671
File Exclusion Information Dialog Box 672
INI File Exclusion Information Dialog Box 673
Choose Registry Key Dialog Box 673

Edit Registry Key Dialog Box 674
About Exclusions Editor Dialog Box 674
Options.ini File 674
Files Associated with Repackager 681
Repack.ini File 684
Using InstallShield to Chain Multiple Windows Installer Packages Together 685
Troubleshooting 685
Troubleshooting Guidelines for WinINSTALL Conversion 685
Troubleshooting Guidelines for SMS Conversion 686
Resolving an “Error Building Table File” Error 686

9 Performing Automated Repackaging and Virtualization Using the Automated Application Converter 689

Getting Started With Application Virtualization 691
About Application Virtualization 693
About Microsoft Application Virtualization 695
About Microsoft Application Virtualization (App-V) 696
Creating 64-Bit App-V Packages 699
Components of an App-V Application 700
How Windows Services Are Integrated into an App-V Application 703
About ThinApp Virtual Packages 704
About ThinApp Applications 704
ThinApp Virtual Operating System *705*
Components of a ThinApp Application *705*
Benefits of Deploying ThinApp Applications 707
Prerequisites for Building a ThinApp Application 707
About Citrix Virtual Packages 707
About Citrix XenApp and Citrix Profiles 708
About Citrix XenApp *708*
About Citrix Profiles (.profile) *709*
Benefits of Deploying Citrix Profiles 711
About the Automated Application Converter 713
Benefits of Using the Automated Application Converter 714
Automated Application Converter Workflow Diagram 715
Supported Operating Systems 716
Supported Virtual Machines 716
Launching the Automated Application Converter 717
Getting Started With the Automated Application Converter 718
Opening a Project 719
Using the Application Conversion Project Wizard 721
About Automated Application Converter Project Files 734
Using Automated Application Converter in Evaluation Mode 737
Managing Virtual Machines 738

Virtual Machine System Requirements 738
Preparing Your Virtual Machines for Use With the Automated Application Converter 740
VMware VIX API Requirement 743
Adding Virtual Machines Using the Virtual Machine Import Wizard 743
Editing Virtual Machine Properties on the Machines Tab 748
Connecting to Active Virtual Machines. 751
Managing Packages to Convert 751
Selecting Packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server 752
Selecting Packages from a Local Machine or Network 756
Editing Package Properties on the Packages Tab 761
About Repackaging Windows Installer Packages 764
Using the Application Conversion Wizard to Perform Automated Package Conversion 765
Performing a Conversion Using the Application Conversion Wizard 765
Viewing Conversion Results 768
Launching Packages for Testing 769
Publishing Converted Packages. 771
Setting Automated Application Converter Options. 774
Capturing Virtualization Context. 775
Reference 775
Automated Application Converter User Interface. 777
Packages Tab 778
Adding Packages to the List 779
Viewing Package Information on the Packages Tab 779
Packages Tab Properties 780
Icons Used on the Packages Tab. 787
Context Menu Commands on Packages Tab 789
Machines Tab 790
Adding Virtual Machines to the List 791
Viewing Virtual Machine Information on the Machines Tab 792
Machines Tab Properties 793
Context Menu Commands on Machines Tab 795
Results Tab 796
Results Tab Properties 797
Icons Used on the Results Tab. 798
Context Menu Commands on Results Tab 799
Menus & Toolbar Buttons 800
Output Window 802
Column Selector and Properties Windows 803
AdminStudio Automated Application Converter Log Report 804
Using List Features 808
Sorting Lists 808
Changing Which List Columns Are Displayed 809
Changing Column Order 810

Resizing List Columns 810
Grouping Lists 811
Wizards 813
Application Conversion Project Wizard 813
Open Project Panel 816
Application Conversion Project Wizard Welcome 817
Select Package Source 818
Connect to an AdminStudio Application Catalog 819
Connect to a Microsoft Configuration Manager Server 820
Select Packages 822
Selected Package List 824
Select Virtual Machine Source 829
Select Virtual Machines from a Microsoft Hyper-V Server 830
Select Virtual Machines from a VMware ESX or ESXi Server 831
Select Virtual Machines 832
User Credentials 834
Initial Configuration Complete 836
Select Output Formats 837
Automated Repackaging on Virtual Machines 838
Application Conversion Project Wizard Complete Panel 839
Package Import Wizard 839
Package Import Wizard Welcome 840
Package Import Wizard Complete 841
Virtual Machine Import Wizard 841
Virtual Machine Import Wizard Welcome 842
Virtual Machine Import Wizard Complete 843
Application Conversion Wizard 843
Application Conversion Wizard Welcome 844
Application Conversion Wizard Complete 845
Package Publish Wizard 845
Package Publish Wizard Welcome 846
Select Publish Target 847
Connect to an AdminStudio Application Catalog 848
Connect to a Microsoft Configuration Manager Server 849
Select Destination Folder 851
Select Group 852
Select Package Types to Publish 853
Package Publish Wizard Complete 854
Dialog Boxes 854
Browse for Folder Dialog Box 855
Guest Agent 856
Open Dialog Box 856
MST Dialog Box 857
Project Options Dialog Box 857

Select Package Installation File Dialog Box . . . 861
Select Transform Dialog Box . . . 862
Select Virtual Machine Dialog Box . . . 863
Select Virtual Machine Image File Dialog Box . . . 864
Command Line Support . . . 865
Specifying Global Default Virtual Conversion Settings . . . 868
Troubleshooting . . . 869
First Things to Check . . . 869
Problems and Solutions . . . 872
Best Practices for Optimal Performance . . . 879
How to Test a Virtual Machine . . . 880
Automated Application Converter Error Messages . . . 881
Error -4308: VM failed to start up . . . 882
Error -4309: VM failed to shut down . . . 882
Error -4310: Failed to connect to VM . . . 883
Error -4312: Failed to prepare Repackager . . . 884
Error -4313: Failed to access the package . . . 885
Error -4314: Failed to copy repackaged output from virtual machine . . . 886
Error -4315: Failed to send command to VM . . . 886
Error -4316: Failed getting response from VM . . . 887
Error -4317: Failed running pre-snapshot . . . 887
Error -4318: Failed running post-snapshot . . . 888
Error -4319: Failed running package installation . . . 888
Error -4320: Failed creating folder on VM . . . 889
Error -4333: Preparing command-line... . . . 890
Error -4380: Failed to prepare AppV . . . 891
Error -4388: Failed preparing for pre-snapshot . . . 892
Error -4389: Failed connecting to server . . . 893
Error -4390: Failed connecting to image . . . 893
Error -4391: Failed to reboot . . . 894
Error -4395: Failed to create VM directory . . . 894
Error -4409: Failed to delete package cache folder . . . 895
Virtualization Conversion Error Messages . . . 895
Error -9000: Unknown Exception . . . 895
Error -9001: Unknown COM . . . 896
Error -9002: Error Opening Package . . . 896
Error -9003: Error Saving Package . . . 896
Error -9004: Process Cancelled By User . . . 897
Error -9005: Error Creating Temporary Folder . . . 897
Error -9006: Error Decompressing Package . . . 898
Error -9007: File With Extension Not Found . . . 898
Error -9008: Error Extracting Icon . . . 899
Error -9009: Unknown Provider . . . 899
Error -9010: Invalid Target File Name . . . 899

Error -9011: Error Reading MSI Table 900
Error -9012: Unexpected Error in Method 900
Error -9013: Type Library Not Found 901
Error -9014: ShellExecute Failed 901
Error -9015: Unable to Determine Full Path for Driver 902
Warning -9016: Contents of Table Ignored 902
Warning -9017: .NET 1.x Assembly Not Supported 903
Warning -9018: Custom Actions Ignored 903
Warning -9019: Conditionalized Components 904
Error -9020: Directory With Null Parent 905
Error -9021: Unable to Extract COM Data 905
Error -9022: Complus Table 906
Error -9024: FileSFPCatalog 906
Warning -9026: LaunchCondition Table 907
Warning -9027: LockPermissions Table 907
Error -9028: MoveFile Table 908
Error -9029: MsiDriverPackages Table 909
Warning -9030: ODBCTranslator Table 909
Warning -9031: RemoveFile Table 910
Warning -9032: RemoveIniFile Table 910
Warning -9033: RemoveRegistry Table 911
Error -9036: ISCEInstall Table 911
Error -9037: ISComPlusApplication Table 912
Error -9038: ISPalmApp Table 912
Error -9039: ISSQLScriptFile Table 913
Error -9040: ISVRoot Table 913
Error -9041: ISXmlFile Table 914
Error -9051: Package Decompression Canceled 914
Error -9100: CreateInstance of Package Object Failed 914
Error -9101: Create Operation of Package Object Failed 915
Error -9102: Failed to Write Header Information 915
Error -9103: Citrix Finalization Failed 916
Error -9104: Citrix Save Failed 916
Error -9105: Error Initializing Citrix Writer 916
Error -9106: Error Initializing Citrix Package 917
Error -9107: Error Writing Citrix File Entries 917
Error -9108: Error Determining Source File Path 918
Error -9109: Error Writing Citrix Folder Entries 918
Error -9110: Error Writing Citrix Registry Entries 918
Error -9113: Error Writing Citrix INI File Entries 919
Error -9114: Error Writing Citrix Shortcuts 919
Error -9115: Error Saving Citrix Profile 920
Error -9116: Error Creating Empty Citrix Profile 920
Error -9117: Error Creating Intermediate Folder 920

Error -9118: Error Initializing Citrix Profile. 921
Error -9119: Error Creating Default Target in Citrix Profile 921
Error -9120: Error Deleting File From Profile 921
Error -9121: Failed to Copy File into Citrix Profile 922
Error -9122: Target Does Not Exist in Citrix Profile 922
Error -9124: No Shortcuts Created for this Profile 923
Error -9125: Error Writing Citrix File Type Associations 923
Error -9126: Failed to Sign Profile Using Certificate. 924
Error -9127: Could Not Create Script Execution 924
Warning -9128: Duplicate Shortcut 924
Warning -9129: Duplicate Shortcut Names 925
Warning -9130: Duplicate Shortcut Targets 925
Warning -9131: Unable to Resolve Installer Variable 926
Warning -9132: 16 Color Shortcut Icon Not Found 926
Warning -9133: Shortcut Icon Not Found 926
Warning -9134: Failure to Extract Icon from Executable 927
Error -9135: Shortcut Target is 16-Bit. 927
Warning -9136: Some Files May Not Be Decompressed 928
Warning -9137: Destination Directory Cannot Be Found. 928
Warning -9138: Ignoring a DuplicateFile Table Entry 929
Error -9139: 64-Bit Executables (XenApp) 929
Error -9200: ThinApp Must Be Installed 930
Warning -9201: Extension for Shortcut Files Must Be “.exe” 930
Error -9202: No Applications Were Created 931
Error -9203: ThinApp Tool is Missing 931
Error -9204: Duplicate Shortcut 931
Error -9205: Identically-Named Shortcut Already Exists, But With Different Parameters 932
Error -9206: Identically-Named Shortcut Already Exists, But With a Different Target 932
Error -9207: Error During Build Process (vregtool.exe) 933
Error -9208: Error Occurred During Build Process (vftool.exe) 933
Error -9209: Error Occurred During ThinApp Build Process (tlink.exe) 934
Error -9210: 64-Bit Executables (ThinApp). 934
Error -9300: Unhandled Exception During AdviseFile Operation 934
Error -9301: Unhandled Exception During AdviseRegistry Operation. 935
Error -9302: Unhandled Exception During Command Action 935
Error -9303: Unhandled Exception During Alter File Action. 935
Error -9304: Unhandled Exception During Alter Registry Action 936
Error -9305: Unhandled Exception During Create Action 936
Error -9306: Unhandled Exception During Execution of Rules Engine 937
Error -9401: Error Initializing App-V Writer 937
Error -9402: Error Initializing App-V Package. 937
Error -9403: Error Writing App-V File Entries 938
Error -9404: Error Writing App-V Folder Entries. 938
Error -9405: Error Writing App-V Registry Entries 938

Error -9406: Error Writing App-V INI File Entries 939
Error -9407: Error Writing App-V Shortcuts 939
Error -9408: Error Writing App-V File Type Data 939
Error -9409: Error Saving App-V Data 940
Error -9410: Error Determining Source File Path 940
Error -9411: OSD File Template Could Not Be Extracted 940
Error -9412: OSD File Could Not Be Saved 941
Error -9413: App-V OSD Save 941
Warning -9414: Local App-V Application Specified as a Dependency of the Primary Application 941
Error -9415: Dependency Application Was Not Found 942
Warning -9416: Invalid Primary Application Directory 942
Error -9417: Dependency Application's OSD File Contains an Invalid HREF Value 943
Error -9418: Error While Privatizing Side-By-Side Assemblies 943
Error -9419: Error Inserting Watermark 943
Error -9420: Error During App-V Package Upgrade 944
Warning -9421: 16-Bit Application 944
Error -9422: Package Cannot Be Opened 945
Warning -9423: No Shortcuts Detected 946
Warning -9500: Shortcut Missing 946
Error -10000: Process Cancelled By User 947
Warning -10001: Suite File Missing 947
Warning -10002: Suite File is Duplicate 947
Warning -10003: Application File Missing 948
Warning -10004: INI File Missing 948
Fix 11000: Excluding TCPIP Registry Entries 948
Fatal Error 11001: Fail on VMware 949
Warning 11003: Control Panel Applet - Citrix 949
Fix 11004: Control Panel Applet - ThinApp 949
Fatal Error 11005: QuickTime 7.4.1 Causes Fatal Error 950
Fix 11006: Adobe Distiller Exclude AdobePDFSettings 950
Fix 11007: Exclude URL Shortcut 950
Steps to Take Before Calling Technical Support 951
Application Features Requiring Pre- or Post-Conversion Actions 951

10 Using the Virtual Package Editor 955

About Virtualization 956

About the Virtual Package Editor 957

Components of an App-V Package 958

Getting Started with the Virtual Package Editor 958

Starting the Virtual Package Editor 958
Creating a New Virtual Package 959
Opening an Existing Virtual Package 959
Saving a Virtual Package 960
Closing a Virtual Package 963

Working with the Virtual Package Editor Interface 963
Configuring the Value of a Setting for More Than One Item at a Time 963
Showing or Hiding the Start Page in the Virtual Package Editor 964
Rearranging the Start Page and Virtual Package Tabs 965
Showing or Hiding the Settings and Output Windows 965
Moving the Settings, Output, and Script Windows 965
Showing or Hiding Toolbars 966
Adding Buttons and Menus to a Toolbar 966
Removing Buttons and Menus from a Toolbar 966
Creating a Custom Toolbar 967
Editing Virtual Packages 967
Specifying Virtual Package Information 968
Viewing History for a Virtual Package 968
Configuring General Information for a Virtual Package 968
Specifying a Virtual Package's Dependencies 969
Adding a Dependency to a Virtual Package 969
Configuring a Dependency in a Virtual Package 969
Associating a Package's Targets with a Dependency in a Virtual Package 970
Specifying Whether a Dependency is Mandatory for a Target in a Virtual Package 970
Removing a Target from a Dependency in a Virtual Package 971
Removing a Dependency from a Virtual Package 971
Organizing Virtual Application Data 971
Including Files and Folders 971
Adding a Predefined Folder to the VFS Folder in an App-V Package 972
Adding a Folder to an App-V Package 972
Adding a File to an App-V Package 972
Configuring a File or Folder in an App-V Package 973
Setting the VFS Path for the Contents of a Predefined Folder in an App-V Package 973
Moving a File or Folder in an App-V Package 974
Extracting Files and Folders from the App-V Package 974
Removing a File or Folder in an App-V Package 975
Editing the Virtual Registry 975
Adding a Registry Key to a Virtual Package 976
Configuring a Registry Key in a Virtual Package 976
Configuring the App-V Override Setting for All of the Subkeys Under One or More Keys 977
Adding a Registry Value to a Registry Key in a Virtual Package 978
Configuring a Registry Value and Its Value Data in a Virtual Package 978
Removing a Registry Value from a Registry Key in a Virtual Package 979
Removing a Registry Key from a Virtual Package 979
Defining Targets in a Virtual Application 979
Adding a Target to a Virtual Package 979
Configuring a Target in a Virtual Package 980
Removing a Target from a Virtual Package 980
Creating Shortcuts to the Virtual Application on the Client System 980

Adding a Shortcut for a Virtual Package *981*
Configuring a Shortcut in a Virtual Package *981*
Removing a Shortcut from a Virtual Package *982*
Using Environment Variables in a Virtual Environment 982
Setting an Environment Variable in a Virtual Package *982*
Configuring an Environment Variable in a Virtual Package *983*
Removing an Environment Variable from a Virtual Package *983*
Configuring File Extension Associations for the Virtual Application 983
Adding a File Extension to a Target in a Virtual Package *984*
Configuring a File Extension in a Virtual Package *984*
Adding a Verb to a File Extension in a Virtual Package *985*
Configuring a Verb for a File Extension in a Virtual Package *985*
Removing a Verb from a File Extension in a Virtual Package *986*
Removing a File Extension from a Virtual Package *986*
Creating Scripts that Run Before or After the App-V Application Is Streamed or Launched 986
Adding a Script to a Target in a Virtual Package *987*
Configuring a Script in a Virtual Package *987*
Causing the App-V Application to Close After a Script Failure *988*
Removing a Script from a Virtual Package *989*
Configuring Virtual Services 990
Adding a Virtual Service to a Virtual Package 990
Configuring a Virtual Service in a Virtual Package 990
Removing a Virtual Service from a Virtual Package 991
Testing and Troubleshooting Virtual Packages 991
Using the App-V Application Launcher to Test the Virtual Package 991
Using Debug Tools with a Virtual Package 993
Using the Virtual Package Editor to Resolve Application Conflict Evaluators (ACEs) in App-V Packages 994
Virtual Package Editor Reference 998
Virtual Package Editor Start Page 999
Virtual Package Editor Menu, Toolbar, and Window Reference 999
Menus in the Virtual Package Editor 999
File Menu in the Virtual Package Editor *1000*
Edit Menu in the Virtual Package Editor *1001*
View Menu in the Virtual Package Editor *1001*
Window Menu in the Virtual Package Editor *1002*
Help Menu in the Virtual Package Editor *1002*
Standard Toolbar in the Virtual Package Editor 1002
Script Window 1003
Settings Window 1004
Output Window 1004
Virtual Package Editor Dialog Box Reference 1004
Browse for Folder Dialog Box 1004
Edit Value Dialog Box 1005
Save As Dialog Box 1005

Select a File Dialog Box 1006
Select a Folder Dialog Box 1006
Select Files to Add to the Virtual Package Dialog Box 1006
Virtual Package Editor View Reference 1006
Package Information View 1007
General Information View *1007*
Dependencies View *1010*
Application Data View 1012
Files and Folders View *1012*
Registry View *1015*
Shortcuts View *1017*
System Configuration View 1026
Virtual Services View *1026*

11 Creating Customized Virtual Applications 1031

About Virtualization 1031
About the AdminStudio Virtualization Interface 1033
About the Virtualization Assistant Tabs 1034
Using the More Options, Other Places, and Help Links Sections in a Virtualization Assistant 1035
Navigating in a Virtualization Assistant 1035
Opening the Installation Designer 1035
Showing or Hiding the Virtualization Assistants 1036
Creating Microsoft App-V Applications 1036
Overview of Microsoft Application Virtualization and the Microsoft App-V Assistant 1037
About Microsoft Application Virtualization (App-V) and the App-V Assistant 1038
Components of an App-V Package 1042
About the Microsoft App-V Assistant 1043
Process for Authoring an App-V Application Using the Microsoft App-V Assistant *1043*
Supported InstallShield Project Types *1046*
How Transforms are Included in an App-V Application *1046*
How Windows Services Are Integrated into an App-V Application *1047*
Quick Start for Microsoft App-V Sequencer Users *1047*
Using the Microsoft App-V Assistant to Create an App-V Application 1050
Specifying Package Information and Deployment Options 1051
Specifying Package Information *1051*
Specifying Operating System Requirements *1052*
Specifying Upgrade Package Information *1052*
Specifying the Deployment Server *1053*
Including Diagnostic Tools With an App-V Application *1054*
Managing Files in an App-V Application 1055
Adding, Deleting, and Moving Files and Folders in an App-V Application *1055*
Controlling the Display of Predefined Folders *1059*
Specifying the Primary Application Directory *1060*
Setting Isolation Options for Folders and Files 1062

Overview of App-V Isolation Options *1063*
Setting Isolation Options for Folders and Files *1064*
Inheritance of Isolation Options from Folders to Files *1065*
Modifying Shortcuts to the App-V Application's Executable Files 1065
App-V Applications and the Virtual Environment *1066*
App-V Shortcut Requirements *1066*
Creating a New App-V Application *1066*
Including an Existing App-V Application *1067*
Excluding or Deleting an Existing App-V Application *1067*
Excluding vs. Deleting App-V Application Shortcuts *1069*
Renaming an App-V Application *1069*
Modifying App-V Application Registry Settings 1070
About the Windows Registry *1070*
Adding or Deleting Registry Keys and Values *1071*
Setting App-V Application Registry Isolation Options 1071
Inheritance of Isolation Options in the Registry *1072*
Performing Dynamic Suite Composition 1072
Modifying Build Options 1074
Selecting Releases to Build *1074*
Enabling App-V Application Building When in Direct Edit Mode *1075*
Specifying Whether to Compress the Data Files in an App-V Package *1075*
Including Additional Windows Installer Packages in an App-V Application *1075*
Building a Windows Installer Package to Assist in the Distribution of an App-V Application *1076*
Specifying Package Feature Block Optimizations *1076*
Building an App-V Application 1078
Testing an App-V Application Using the App-V Application Launcher 1080
Microsoft App-V Assistant Reference 1081
Pages 1082
Microsoft App-V Assistant Home Page *1082*
Package Information Page *1083*
Files Page *1087*
Applications Page *1090*
Registry Page *1091*
Dynamic Suite Composition Page *1093*
Build Options Page *1096*
Dialog Boxes 1099
Advanced Settings Dialog Box *1099*
App-V Diagnostic Tools Dialog Box *1101*
App-V Package Upgrade Settings Dialog Box *1101*
File Isolation Options Dialog Box *1102*
Folder Isolation Options Dialog Box *1102*
Launch App-V Application Dialog Box *1103*
Package Optimizations Dialog Box *1104*
Primary Application Directory Dialog Box *1105*

Registry Isolation Options Dialog Box *1107*
Building App-V Applications Using the Command Line 1108
App-V Application Conversion Error and Warning Messages 1108
Application Features Requiring Pre- or Post-Conversion Actions 1108
Creating ThinApp Applications 1108
Overview of the ThinApp Assistant 1109
About ThinApp Applications 1109
The ThinApp Virtual Operating System *1109*
Benefits of Deploying ThinApp Applications *1110*
About the ThinApp Assistant 1110
Process for Authoring a ThinApp Application Using the ThinApp Assistant *1110*
Components of a ThinApp Application *1113*
Supported InstallShield Project Types *1115*
How Transforms are Included in a ThinApp Application *1115*
About Sandboxes *1116*
Using the ThinApp Assistant to Create a ThinApp Application 1116
Specifying ThinApp General Settings 1117
Specifying Sandbox Information *1117*
Specifying Control Access via Active Directory *1118*
Prerequisites for Building a ThinApp Application *1119*
Including Diagnostic Tools With a ThinApp Application *1119*
Managing Files and Folders in a ThinApp Application 1120
Adding, Deleting, and Moving Files and Folders in a ThinApp Application *1121*
Controlling the Display of Predefined Folders *1125*
Setting ThinApp Isolation Options 1126
Overview of ThinApp Isolation Options *1126*
Setting Isolation Options for Folders *1129*
Inheritance of Isolation Options from Folders to Files *1129*
Modifying Shortcuts to the ThinApp Application's Executable Files 1130
ThinApp Applications and the Virtual Environment *1131*
Compressing a ThinApp Application *1131*
ThinApp Shortcut Requirements *1132*
Creating a New ThinApp Application *1132*
Including an Existing ThinApp Application *1132*
Excluding or Deleting an Existing ThinApp Application *1133*
Excluding vs. Deleting ThinApp Application Shortcuts *1134*
Renaming a ThinApp Application *1135*
Modifying ThinApp Application Registry Settings 1135
About the Windows Registry *1135*
Adding or Deleting Registry Keys and Values *1136*
Setting ThinApp Isolation Options on Registry Keys *1136*
Inheritance of ThinApp Isolation Options in the Registry *1137*
Modifying Build Options 1137
Selecting Releases to Build *1138*

Enabling ThinApp Application Building When in Direct Edit Mode 1139
Including Additional Windows Installer Packages in a ThinApp Application 1139
Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application 1140
Setting ThinApp Log Monitor Tracing Options 1140
Setting AppLink Options 1141
Setting AppSync Options 1144
Building a ThinApp Application 1147
ThinApp Assistant Reference 1149
Pages 1149
ThinApp Assistant Home Page 1149
General Settings Page 1151
Files & Folders Page 1153
Applications Page 1156
Registry Page 1157
Build Options Page 1159
Dialog Boxes 1165
ThinApp Diagnostic Tools Dialog Box 1165
Folder Isolation Options Dialog Box 1166
Registry Isolation Options Dialog Box 1168
AppLink Settings Dialog Box 1170
Add AppLink Reference Dialog Box 1173
AppSync Settings Dialog Box 1176
Building ThinApp Applications Using the Command Line 1180
ThinApp Application Conversion Error and Warning Messages 1180
Application Features Requiring Pre- or Post-Conversion Actions 1180
ThinApp Not Found 1180
ThinApp Application Configuration File: package.ini 1180
[BuildOptions] 1181
[Compression] 1186
[Isolation] 1186
[MainApp.exe] 1187
[Test.exe] 1190
Creating Citrix Profiles 1190
Overview of the Citrix Assistant 1190
About Citrix XenApp 1192
About the Citrix Assistant 1193
About Citrix Profiles 1194
Benefits of Deploying Citrix Profiles 1195
Supported InstallShield Project Types 1197
How Transforms are Included in a Citrix Profile 1197
Using the Citrix Assistant to Create a Citrix Profile 1198
Specifying Citrix Profile Information 1198
Specifying the Profile Name, Description, and Version 1199
Specifying Whether Users Should Be Able to Update Applications 1199

Including Diagnostic Tools With a Citrix Profile *1200*
Specifying Operating System and Language Requirements 1201
Setting Operating System Requirements and Service Pack Levels *1201*
Setting Language Requirements *1202*
How Requirements are Applied at Runtime *1202*
Adding Pre-Launch and Post-Exit Scripts *1203*
Managing Files and Folders in a Citrix Profile 1206
Managing Files and Folders in a Citrix Profile *1206*
Controlling the Display of Predefined Folders *1210*
Setting Isolation Options 1211
Overview of Citrix Isolation Options *1212*
Setting Isolation Options for Folders and Files *1214*
Inheritance of Isolation Options from Folders to Files *1214*
Modifying Profile Shortcut Settings 1214
Shortcuts and the Isolation Environment *1215*
Shortcut Requirements *1217*
Creating a New Profile Shortcut *1217*
Including an Existing Profile Shortcut *1218*
Excluding vs. Deleting a Profile Shortcut *1218*
Conditions When a Shortcut Should be Excluded or Deleted *1219*
Renaming a Shortcut *1220*
Modifying Profile Registry Settings 1220
About the Windows Registry *1220*
Adding or Deleting Registry Keys and Values *1221*
Setting Registry Isolation Options *1221*
Inheritance of Isolation Options in the Registry *1222*
Modifying Build Settings 1222
Selecting Releases to Build *1223*
Digitally Signing a Citrix Profile *1224*
Including Additional Windows Installer Packages in a Citrix Profile *1224*
Enabling Citrix Profile Building When in Direct Edit Mode *1225*
Building a Citrix Profile 1225
Citrix Assistant Reference 1227
Pages 1227
Home Page *1227*
Profile Information Page *1229*
Profile Requirements Page *1231*
Profile Files Page *1234*
Profile Shortcuts Page *1237*
Profile Registry Page *1239*
Build Settings Page *1240*
Dialog Boxes 1243
Script Execution Dialog Box *1243*
Diagnostic Tools Dialog Box *1244*

File Isolation Options Dialog Box *1245*
Folder Isolation Options Dialog Box *1247*
Registry Isolation Options Dialog Box *1248*
Service Packs Requirement Dialog Box *1250*
Building Citrix Profiles Using the Command Line 1251
Citrix Profile Conversion Error and Warning Messages 1251
Application Features Requiring Pre- or Post-Conversion Actions 1252

12 Customizing and Authoring Installations Using InstallShield 1253

AdminStudio-Specific Functionality in InstallShield Editor 1254
InstallShield Editor Integration with Application Manager and the Software Repository 1256
Getting a Copy of a Software Repository Package 1257
Opening and Editing a Software Repository Package 1258
Adding a Package to the Software Repository via the InstallShield Editor Build Process 1260
Using the Application Manager Context Menu 1260
Application Manager Tab on InstallShield Editor Options Dialog Box 1261
Microsoft App-V, VMware ThinApp, and Citrix XenApp Virtualization Support 1261
Differences Between InstallShield Editor and InstallShield Professional Edition 1262
InstallShield Editor Help Library 1262

13 Customizing Installations Using Tuner 1263

Working with Transforms 1264
Creating New Transform Files 1265
Opening Existing Transforms 1266
Opening Recently Accessed Transforms 1267
Creating Generic Transforms 1267
Using Response Transforms 1268
Viewing Transform Properties 1268
Validation 1268
Prevalidating Windows Installer Packages 1269
Handling Invalid Windows Installer Packages 1270
Postvalidating Transforms 1271
Evaluation Files and Internal Consistency Evaluators 1272
Setup Organization 1272
Changing a Feature's Visibility 1273
Setting the Initial State of a Feature 1273
Editing a Feature's Description 1274
Setting the Default Destination 1274
Setting the Default Organization 1274
Changing the Destination Variable 1275
Preventing Features from Displaying During Custom Installation 1275
Setting Feature Properties 1275
Using Feature Advertisement 1276

Configuring Package Content 1277
Files and Folders 1277
Adding Files 1277
Displaying Files from the Base Windows Installer Package 1278
Preventing Installation of Files from the MSI 1278
Removing Added Files 1279
Storing Added Files 1279
Registry Entries 1279
Creating a Registry Key 1280
Creating a Registry Value 1281
Importing REG Files 1281
Removing Registry Information 1282
Shortcuts 1282
Creating Shortcuts 1283
Changing a Shortcut's Icon 1283
Change a Shortcut's Location 1284
Changing a Shortcut's Target 1284
Creating a Hot Key 1284
Removing Shortcuts 1285
Determining the Path of Changed Shortcuts 1285
INI Files 1285
Adding INI Files 1286
Importing Existing INI Files 1286
Adding Sections to INI Files 1287
Adding New Keys to INI File Sections 1287
Modifying INI File Keys, Values, and Actions 1287
Removing INI Files 1288
Removing Sections from INI Files 1289
Removing INI File Section Keys 1289
ODBC Resources 1289
Adding New Data Sources 1290
Adding New ODBC Data Source Attributes 1290
Adding New ODBC Driver Attributes 1290
Editing ODBC Data Source Attributes 1291
Editing ODBC Driver Attributes 1291
Removing Existing ODBC Data Sources 1291
Removing ODBC Driver Attributes 1291
Removing ODBC Data Source Attributes 1292
NT Services 1292
Setting NT Service Arguments 1292
Setting NT Service Dependencies 1293
Setting the NT Service Description 1293
Setting the NT Service Display Name 1293
Setting the NT Service Error Control Level 1294

Setting the NT Service Load Order Group 1294
Setting the NT Service Overall Install Result 1294
Setting the NT Service Start Type 1295
Setting NT Service Start Name and Password 1295
Setting the NT Service Type 1295
Working with Dialogs 1296
Hiding Dialogs During UI Sequences 1296
Restoring Dialog Sequences 1297
Suppressing the License Agreement Dialog Box 1297
Disabling Custom Setups 1297
Editing Dialog Properties 1298
Dialogs View vs. Command-Line Options 1299
Dialog Suppression Issues 1299
Configuring Additional Server Locations 1299
Adding Additional Server Locations 1300
Modifying Server Locations 1300
Removing Server Locations 1300
Reordering Server Locations 1301
Changing Add/Remove Program Settings 1301
Changing Add/Remove Programs Properties 1301
Disabling the Modify, Remove, or Repair Buttons 1302
Customizing Setup Properties 1302
Adding Custom Setup Properties 1302
Adding and Editing Comments 1303
Removing Custom Setup Properties 1303
Modifying Setup Properties 1303
Preparing Packages for Distribution 1304
Copying the Installation to a Network Location 1304
Copying the Installation to an FTP Server 1305
Creating a Package Definition File (PDF) 1305
Creating an SMS File 1306
Instructing SMS to Create a Management Information Format File at Deployment Time 1307
Deploying Windows Installer Setup Packages with Systems Management Server 2.0 1307
Creating a Setup.exe File for the Package and Transform 1307
Additional Setup.ini Parameters 1308
Directly Editing Packages 1308
Adding a New Record Using the Direct Editor 1309
Finding and Replacing Using the Direct Editor 1309
Launching the Direct Editor from the Validation Tab 1309
Tuner Reference 1310
User Interface Reference 1310
Menus and Toolbar 1311
View Bar 1314

Checklist 1314
Customization Steps Checklist *1314*
Output Window 1315
Customize Dialog Box 1316
Properties Dialog Box 1316
Options Dialog Box 1316
Transform Summary Dialog Box 1316
Dialog Properties Dialog Box 1318
Tuner Views 1319
Tuner Start Page 1320
Welcome to Tuner View *1320*
Create a New Transform View *1320*
Open a Recent Transform View *1322*
Open an Existing Transform View *1323*
Help View 1323
Package Validation View 1323
Prevalidation View *1323*
Organization View 1324
Product Properties View *1324*
Features View *1325*
System Configuration View 1326
Files and Folders View *1327*
Registry View *1327*
Shortcuts View *1328*
Shortcuts View/Shortcut Properties *1329*
Shortcuts View/Shortcut Target *1331*
Shortcuts View/Shortcut Locations *1332*
INI Files View *1333*
ODBC Resources View *1334*
NT Services View *1335*
Application Configuration View 1337
Server Locations View *1337*
Setup Properties View *1338*
Dialogs View *1338*
Add/Remove Programs View *1338*
Package Preparation View 1339
Postvalidation View *1340*
Package View *1341*
Package View/Location View *1342*
Package View/Setup View *1342*
Package View/SMS View *1342*
Additional Tools View 1342
Direct Editor *1342*
Import INI File Wizard 1344

Welcome Panel 1344
Import INI File Panel 1344
Import Conflict Options Panel 1344
Finishing INI File Import Panel 1345
Import REG File Wizard 1345
Welcome Panel 1345
Import Registry File Panel 1345
Import Conflict Options Panel 1345
Finishing Registry Import Panel 1346
Packaging Wizard 1346
Location Panel 1346
Setup.exe Panel 1347
SMS Panel 1347
Packaging Summary Panel 1347

Part 4: Testing Windows Installer Packages 1349

14 Identifying and Resolving Application Conflicts Using ConflictSolver 1351

About ConflictSolver 1352
Application Manager and ConflictSolver 1352
Functionality Comparison 1353
Accessing the Tools 1354
Displaying ConflictSolver Options in Application Manager 1354
Creating and Connecting to Application Catalogs 1354
Types of ACE Rules: Best Practice and Conflict 1355
Connecting to Existing Application Catalogs 1355
Best Practices for Conflict Detection 1356
Plan Your Import 1356
Plan Your Conflict Detection 1358
Perform Conflict Detection 1358
Scanning for Dependencies 1360
Validating Packages 1361
Performing Validation 1362
Validating Before Import 1362
Validating During Import 1363
Validating After Import 1364
Validating an Imported Package 1364
Viewing ICE Error Information 1365
Suppressing an ICE Error 1366
Deleting an ICE Error 1367
Changing the Default Validation File 1368
Excluding Specific ICEs from Execution During Validation 1368
Handling Invalid Windows Installer Packages 1369

About ICE43, ICE50, and ICE57 Validation Rules for Shortcuts 1369
Identifying and Resolving Conflicts for Windows Installer Packages 1370
Using Multiple Source Packages in an Analysis 1371
Setting Options 1371
Changing Default Conflict Types Checked 1372
Changing Resolution Options 1373
Checking for Conflicts Across All Source and Target Packages 1374
Checking for Conflicts 1374
Checking for Conflicts Using the Conflict Wizard 1374
Checking for Conflicts During Import 1377
Viewing Identified Conflicts in the Conflicts View 1377
Running Conflict Identification Again 1378
Deleting Persisted Conflict Data 1379
Conflict Persistence 1379
Resolving Conflicts 1379
Conflict Resolution Process 1380
Conflict Resolution and the Software Repository 1381
Automatically Resolving Conflicts 1382
Manually Resolving Conflicts 1383
Resolving Conflicts Directly in Windows Installer Packages 1384
Using Transforms for Conflict Resolution 1384
Reimporting Packages after Successful Conflict Resolution 1385
Testing Microsoft App-V Packages 1385
Performing Best Practice Analysis of App-V Packages 1385
Performing Conflict Analysis of App-V Packages 1388
Generating Reports 1390
Generating Package Reports 1391
Generating File and Registry Reports 1392
Generating Reports in Crystal Reports Format 1393
Creating Custom Reports 1393
Step 1: Creating an ODBC File DSN 1394
Step 2: Creating the Base Report 1395
Step 3: Configuring Basic Settings 1395
Step 4: Creating Page Column Headings 1396
Step 5: Adding Database Fields 1396
Step 6: Adding Parameters 1397
Step 7: Creating Conditions for Filtering 1398
Step 8: Modifying the Footer 1398
Step 9: Using the Report in ConflictSolver 1398
Application Conflict Evaluators (ACEs) 1399
ACE Index 1400
ACE02 1406
ACE03 1406
ACE04 1408

ACE05 1408
ACE06 1409
ACE07 1410
ACE08 1412
ACE09 1413
ACE10 1414
ACE12 1416
ACE13 1418
ACE14 1419
ACE15 1420
ACE16 1421
ACE17 1422
ACE18 1423
ACE19 1424
ACE20 1424
ACE21 1425
ACE22 1426
ACE23 1428
ACE24 1429
ACE25 1431
ACE26 1431
ACE27 1432
ACE28 1433
ACE29 1434
ACE30 1434
ACE31 1435
ACE32 1436
ACE33 1437
ACE34 1437
ACE35 1438
ACE36 1439
ACE200 1440
ACE201 1441
ACE202 1442
ACE203 1444
ACE204 1446
ACE205 1447
ACE206 1448
ACE207 1448
ACE208 1450
ACE209 1451
ACE210 1452
ACE211 1453
ACE212 1454

ACE213 1455
ACE214 1456
ACE215 1457
ACE216 1458
WTS01 1459
WTS02 1460
WTS03 1461
WTS04 1462
WTS05 1463
WTS06 1463
WTS07 1464
WTS08 1465
WTS09 1466
User-Defined ACEs 1467
Types of User-Defined ACEs 1467
Creating User-Defined ACEs 1468
Creating a Custom/Source Only Packages ACE 1469
Creating a Custom/Source and Target Packages ACE 1471
Creating a User Provided DLL-Based ACE 1474
Editing User-Defined ACEs 1477
Deleting User-Defined ACEs 1477
Viewing ACE Metrics 1477
Location of ACE Files 1479
Conflict Application Resolution Definitions (CARDs) 1479
CARD Index 1480
CARD02 1481
CARD04 1482
CARD05 1482
CARD06 1482
CARD07 1483
CARD15 1483
CARD18 1483
CARD19 1484
CARD20 1484
ConflictSolver Reference 1484
ConflictSolver Interface 1485
Menus and Toolbar 1485
Product View Icons 1488
Context Menus 1492
Output Window 1494
Views 1497
Product View 1497
Conflicts View 1499
Validation View 1503

Test Results View 1504
Patch Impact View (Products Tab) 1504
Dependencies View 1505
Associated Patches View 1506
Tables View 1507
OS Snapshot View 1507
Tables View for OS Snapshots 1507
Other Setup Types View 1507
Tables View for Other Setup Types 1508
NCP Views 1508
Conflicts View for Marimba NCP Files 1509
Tables View for Marimba NCP Files 1509
App-V Package View 1509
App-V Conflict View 1510
App-V Dependencies View 1511
App-V Tables View 1513
Patches Tab Views 1513
Patches Group View 1514
New Patches Group View 1514
Group View of a Selected Group 1515
Patch View 1516
Impact Analysis View (Patches Tab) 1517
Dialog Boxes 1518
About ConflictSolver Dialog Box 1519
ACE Rule Properties Dialog Box 1519
General Information Tab 1519
Additional Information Tab 1520
Custom Options Tab 1521
Where Clause Tab 1522
DLL Information Tab 1523
Add Ignore Table Dialog Box 1523
CARD Resolution Options Dialog Box 1523
Conflict Information Dialog Box 1523
Conflict Resolution Dialog Box 1524
Customize Dialog Box 1526
Properties Dialog Box 1526
Expression Builder Dialog Box 1527
Options Dialog Box 1528
Rules Tab 1529
Duplicate Package Tab 1530
Extended Attributes Tab 1533
General Tab 1533
Import Tab 1534
Resolution Tab 1535

Validate Tab *1536*
History Tab *1538*
Patch Impact Information Dialog Box 1538
Resolution Details Dialog Box 1539
Resolution Operations Dialog Box 1539
Resolution Options Dialog Box 1540
Rules Viewer Dialog Box 1540
Wizards 1540
Conflict Wizard 1540
Welcome Panel *1541*
Source Type Panel *1542*
MSI Source Information Panel *1542*
Source Package Panel *1542*
Choose Action Panel *1543*
Best Practice Rules Panel *1543*
Conflict Rules Panel *1544*
Target Information Panel *1545*
Summary Panel *1545*
Rules Wizard 1545
Welcome Panel *1545*
General Information Panel *1546*
Additional Information *1546*
Custom Options Panel *1548*
Token Grammar *1549*
Where Clause Panel *1549*
DLL-Based ACEs Panel *1550*
Summary Panel *1550*
Validation Wizard 1550
MSI Source Information Panel *1551*
Summary Panel *1551*

15 Testing and Fixing Application Compatibility Using Compatibility Solver 1553

16 Identifying and Resolving Package Errors Using PackageExpert 1555

About PackageExpert 1556
Configuring PackageExpert 1558
Setting Default Test Configuration 1558
Specifying the Resolution Method 1559
Testing Packages and Viewing Results 1560
Testing Packages 1560
Testing External Packages 1561
Testing Packages in an Application Catalog 1562
Testing Packages From ConflictSolver 1563

Viewing and Managing Test Results 1565
Viewing Test Results in PackageExpert 1566
Information Shown for an Error Message *1566*
Test Result Views *1567*
Publishing Test Results to the Application Catalog 1572
Publishing Test Results When Prompted *1573*
Toggling Between Local and Published Test Results *1573*
Viewing Test Results in ConflictSolver 1574
Deleting Test Results 1575
Customizing Test Results 1576
Changing the Severity of a Specific Error *1576*
Suppressing the Display of Specific Errors *1581*
Adding Ad-Hoc Test Results 1584
Resolving Errors 1587
Adding Manual Tests 1588
About Manual Tests 1588
Creating a Manual Test 1590
Running and Resolving a Manual Test 1591
Sample Manual Test 1593
PackageExpert Reference 1595
PackageExpert Interface 1596
PackageExpert Home Page 1596
PackageExpert Package Tree 1597
PackageExpert Toolbar Menus 1598
PackageExpert Icons 1599
Status Bar 1601
PackageExpert Views 1603
Test and Resolve View 1603
Configuration View 1606
Filter View 1607
Options View 1607
PackageExpert Dialog Boxes 1608
Connect Application Catalog Dialog Box 1608
Enterprise Server Tab *1609*
Standalone Tab *1610*
Custom Test Results Editor Dialog Box 1611
Manual Test Perform Dialog Box 1613
Manual Test Resolve Dialog Box 1614
Select a Package Dialog Box 1615
Test Result Filter Editor Dialog Box 1616
Unpublished Packages Dialog Box 1617

17 Analyzing the Impact of Installing Microsoft Operating System Patches 1619

About Microsoft Operating System Patches 1620
Using OS Security Patch Wizard to Import Microsoft OS Patches 1622
Identifying and Downloading Microsoft Operating System Patch Files 1623
Importing a Microsoft Operating System Security Patch Into the Application Catalog. 1626
About Dual-Mode OS Patches 1629
Analyzing the Impact of Installing a Microsoft Operating System Patch 1629
Performing Patch Impact Analysis 1630
Viewing Patch Impact Analysis Results 1631
Viewing Patch and Patch Impact Information in Application Manager and ConflictSolver. 1633
Generating the Patch Report 1637
Reference 1638
Patch Impact Analysis Wizard 1638
Welcome Panel 1638
OS Snapshot Panel. 1639
Source Patches Panel. 1639
Target Products Panel 1640
Summary Information Panel 1640
OS Security Patch Wizard 1640
Welcome Panel 1641
Patch File Selection Panel 1641
Patch File Branch Selection Panel 1642
Patch Bulletin Information Panel. 1643
Additional Patch Information Panel 1643
Patch Import Summary Panel. 1644
Import Process Results Panel. 1644
Patch Properties Dialog Box. 1644
General Tab 1645
Contents Tab. 1645
Products Tab. 1646

18 Isolating Applications Using Application Isolation Wizard 1647
About Application Isolation Wizard 1648
Isolating Repackaged Setups Using Repackager. 1649
Launching the Application Isolation Wizard 1649
Isolation Methods 1650
Assemblies. 1650
Manifests 1651
Digital Signatures 1652
Certificates 1652
Code Signing Technologies. 1652
Software Publishing Credentials 1653
Certificate Store 1653
Private Keys 1653

Isolating Applications 1653
Setting Assembly Naming Conventions 1654
Modifying the Default Isolation Recommendations 1655
Filtering File Listings when Manually Configuring Isolation 1656
Servicing Published Shared Assemblies 1657
Application Isolation Wizard Reference 1657
Welcome Panel 1657
Windows Installer File Selection Panel 1658
Isolation Method Panel 1658
Summary Information Panel 1658
Application Isolation Progress Panel 1659
Completing the Application Isolation Wizard Panel 1659
Advanced Options Dialog Box 1659
Manifest Options Tab 1660
Digital Signature Tab 1661
Manifest and Assembly Design Dialog Box 1662
Isolated Components Design Dialog Box 1662
Assembly Properties Dialog Box 1662
Application Manifest Properties Dialog Box 1663
Command-Line Options 1664
Configuration Files 1664
Manifest Examples 1666

19 Ensuring Package Quality Using QualityMonitor 1667
About QualityMonitor 1668
Creating New QualityMonitor Project Files 1668
Opening Existing QualityMonitor Project Files 1669
Working with Test Cases 1669
Running Individual Test Items 1670
Running Multiple Test Items 1670
Adding Test Item Comments 1671
Adding Test Case Comments 1672
Viewing Test Item Details 1672
Clearing Test Case Results 1673
Manually Setting Test Case Status 1674
Manually Setting Test Item Status 1675
Filtering Test Case Data 1675
Deployment Testing 1676
Checking Class IDs 1677
Checking File Associations 1678
Checking Help Files 1678
Checking Prog IDs 1679
Checking Services 1679

Checking Shortcuts 1680
Checking Type Libraries 1680
Checking Manifests 1681
Checking ODBC Data Sources 1682
Checking ODBC Drivers 1682
Specifying Exclusions for Deployment Testing 1683
Lockdown and Runtime Testing 1685
Performing Lockdown and Runtime Tests 1686
Performing Lockdown and Runtime Tests Under a Different User Account 1686
Running Lockdown and Runtime Tests in Restricted Environments 1687
Performing Isolation Tests 1688
Filtering Results of Lockdown and Runtime Tests 1689
Using MSI Doctor to Verify Package Deployment Status 1689
View Product, Feature, or Component Deployment Status Properties 1692
Verify Product, Feature, or Component Data 1692
Install or Configure Products or Features 1693
Reinstall Features 1694
Reinstall Components 1695
Creating Custom Test Cases 1695
Test Reports 1697
Running QualityMonitor from the Command Line 1698
QualityMonitor Reference 1699
Menus and Toolbar 1699
QualityMonitor Interface 1701
Dialog Boxes 1701
About QualityMonitor Dialog Box 1702
Add Exclusions Dialog Box 1702
Component Properties Dialog Box 1702
Feature Properties Dialog Box 1703
Install or Configure Feature Dialog Box 1704
Install or Configure Product Dialog Box 1704
Installed Data Dialog Box 1705
Open QualityMonitor Project Dialog Box 1706
Options Dialog Box 1707
General Tab 1707
Exclusions Tab 1708
Product Properties Dialog Box 1709
Re-install Product/Feature Dialog Box 1710
Runtime Test Filters Dialog Box 1710
Test Item Information Dialog Box 1711
Test Progress Dialog Box 1711
Test Result Dialog Box 1711
Views 1711

Welcome to QualityMonitor View 1712
Product Information View 1713
Test Cycle Summary View 1713
Deployment Tests View 1713
Class IDs View *1714*
File Associations View *1715*
Help Files View *1715*
Prog IDs View *1715*
Shortcuts View *1715*
Type Libraries View *1715*
Manifests View *1716*
ODBC Data Sources View *1717*
ODBC Drivers View *1717*
Services View *1717*
Lockdown and Runtime Tests View 1718
Runtime Execution Details View *1718*
Files View *1719*
Folders View *1720*
Registry Entries View *1722*
Isolation Tests View *1723*
User-Defined Tests View 1724
Test Case View *1724*
Deployment Status View 1724

Part 5: Preparing Installations for Deployment 1727

20 Preparing Packages for Distribution Using Distribution Wizard 1729

Creating Administrative Installations for Packages 1730
Distributing Packages to FTP Servers 1731
Preparing for Altiris Distribution 1732
Preparing for LANDesk Distribution 1733
Preparing for ManageSoft Distribution 1734
Using Marimba Channel Publishing for Package Deployment 1735
Distributing Packages to Network Locations 1737
Distributing to Microsoft Configuration Manager 1738
Preparing for SMS Distribution 1740
SMS File Templates 1741
Preparing for Tivoli Distribution 1742
Preparing for ZENworks Configuration Management 10 Distribution 1743
Preparing for ZENworks Desktop Application Distribution 1748
Preparing for ZENworks Server Distribution 1752

Deploying InstallScript MSI Installations 1757
Reference 1758
Distribution Wizard 1759
Welcome Panel 1759
Distribution Type Panel 1760
Administrative Install Panel 1761
Connect to a Microsoft System Center Configuration Manager Server Panel 1761
Select Destination Folder 1762
Select Group 1763
Altiris Integration Panel 1764
Altiris XML Template 1765
FTP Location Panel 1766
LANDesk Integration Panel 1766
ManageSoft Distribution Settings Panel 1767
ManageSoft Package and Environmental Settings Panel 1768
Marimba Panels 1768
Patches Panel 1769
Marimba Integration Panel 1769
Advanced Marimba Options Dialog Box 1770
Network Location Panel 1771
SMS Distribution Panel 1771
Tivoli Integration Panel 1772
Tivoli Settings Panel 1772
SPD Parameters Panel 1773
SPD File Details 1774
ZENworks Login Panel 1777
ZENworks Desktop Application Panel 1779
ZENworks Server Distribution/Object Panel 1781
ZENworks Server Distribution/Distributor Panel 1783
Package Information Panel 1785
Distribution Summary Panel 1785
Distribution Output Panel 1786
Distribution Wizard for ZENworks Configuration Management 1786
Welcome Panel 1786
Login Panel 1787
Windows Installer Package Information Panel 1787
Bundle Creation Options Panel 1793
Bundle Information Panel 1793
Summary Panel 1794
Publishing Process Panel 1795

21 Distributing Packages Using Configuration Manager Web Console 1797

About Microsoft System Center Configuration Manager and Configuration Manager Web Console 1798
Distributing Software Through Configuration Manager Server 1799

Configuration Manager Web Console Overview 1801
Getting Started 1802
Distributing Packages Using the Configuration Manager Web Console 1802
Configuring Distribution Settings 1803
Distributing a New Package 1803
Selecting a Package for Distribution 1804
Setting Package Configuration Options 1805
Setting Package Settings Options 1806
Setting Access Accounts Options 1810
Setting Distribution Points Options 1811
Setting Programs Options 1812
Setting Advertisements Options 1819
Saving the Package on the Configuration Manager Server 1822
Viewing the Status of a Distributed Package 1822
Modifying the Distribution Settings of a Package 1822
Deleting Packages 1823
Configuration Manager Web Console Reference 1825
Microsoft System Center Configuration Manager Web Console Home Page 1826
Connect to Configuration Manager Page 1826
Distribute Package Page 1827
PDF Selection Page 1827
Configuration Manager Package Configuration Page 1828
Package Settings View 1829
Access Accounts View 1832
Distribution Points View 1833
Programs View 1834
Advertisements View 1841
Package Summary Page 1844
Select a Configuration Manager Collection Dialog Box 1844
Distribution Settings Page 1844
Package Administration Page 1845
Package Status Page 1845
Confirm Delete Page 1848
Distribute New Package Page 1849

22 Analyzing Installations Prior to Deployment 1851
About the Predeployment Test Tool 1852
How the Predeployment Test Tool Works 1852
Predeployment Test Tool Components 1852
Overview of Predeployment Test Tool Results 1853
Configuring the Predeployment Test Environment 1854
Configuring the Predeployment Test Results Web Site 1854
Configuring the Predeployment Test Web Service 1856

Performing Predeployment Analysis 1858
Creating a Test-Ready Windows Installer Package 1858
Setting Predeployment Test Command Line Parameters 1860
Distributing a Test-Ready Windows Installer Package 1862
Running Diagnostic Tests When Distributing Test-Ready Packages 1862
Viewing Predeployment Test Results. 1864
Managing Job Data 1866
Predeployment Test Reference 1867
Wizards and Dialog Boxes 1868
Predeployment Test Start Page 1868
Predeployment Test Preparation Wizard 1869
Welcome Panel 1869
Windows Installer File Selection Panel 1870
Web Service URL Panel 1871
Select Application Catalog Panel 1871
Deployment & Conflict Test Selection Panel 1872
Review Panel 1873
Progress Panel 1874
Summary Panel 1874
Options Dialog Box 1874
Configure Application Catalog for Predeployment Web Service Page 1875
Technical Information 1875
Predeployment Custom Properties 1876
Predeployment Test Identifiers 1876
Predeployment Test Results XML File 1877
Changes Made During Test-Ready Package Creation 1882
Predeployment Test Results. 1883
Predeployment Test Results By Job Report 1884
Job Summary Report 1885
Test Result Summary Report 1886
Job Details Report 1888
Test Details Report 1889
File/Registry Conflicts Report 1891
Predeployment Test Results By Machine Report. 1893
Machine Result Summary Report 1893
Connection to New Catalog Page 1895
Manage Job Page. 1896
Assigning ASP.NET Permissions to Files or Folders 1897
ConflictSolver ACE Reference 1901
ACE07 1902
ACE10 1903
ACE23 1905
ACE24 1907

23 License-Enabling Packages Using FLEXwrap 1909

Enabling License Tracking of Windows Installer Packages. 1910

About FLEXnet Licensing Technology and FLEXwrap 1910

Benefits of FLEXnet Software Licensing Technology 1911

Types of Applications that Can Be FLEX-enabled 1912

About FLEXwrap License Files 1912

Selecting the Executable File to Wrap 1913

About Output Folders 1913

How FLEXwrap Assistant Works 1914

Launch Process for a Wrapped Application 1916

Using the FLEXwrap Assistant 1918

Launching the FLEXwrap Assistant 1918

License-Enabling a Windows Installer Package 1919

Selecting a Windows Installer Package to License-Enable 1919

Setting License Limits and Queuing Behavior 1920

Defining License Requirements and Access Restrictions 1921

Configuring a Connection to the FLEXwrap License Server 1922

Specify the Output Folder and Build Package 1924

Next Steps 1924

Setting Licensing Options 1924

Benefits of Using the Suite Option 1925

Benefits of Encrypting the Source Executables 1925

Setting Your License Limit 1925

Using Overdraft Support to Determine Application Usage 1926

Setting a License Expiration Date 1926

Choosing the Right Queuing Option for Your Application 1926

Grouping Options for Concurrent Use 1927

Setting Licensing Behavior During Server Down Time 1927

Setting Timeout Periods 1927

Setting Configuration Options 1928

About Hostids 1928

Changing the Host Name or Port at Install Time 1928

When to Update the Package Code 1928

FLEX-enabling a Signed Package 1929

Purpose of Packaging Comments 1929

FLEXwrap Assistant Reference 1929

Home Page 1930

Select Files Page 1931

Set Limits Page 1932

Define Access Page 1933

Configure Connection Page 1934

Finalize Package Page 1935

Permitted Hostid Dialog Box 1936

Options Dialog Box 1936

FLEXwrap Server Name Dialog Box 1937
Output Window 1937
Managing FLEXwrap License Files 1937
Overview of the FLEXwrap License Server 1938
Supported Platforms 1939
Communications Issues 1940
Installation Notes 1940
About the FLEXwrap Server Configuration Tool 1941
About the FLEXwrap LMTOOLS 1941
Configuring License Server Using a License File 1942
Configuring License Server Using Services 1942
Using the FLEXwrap Server Configuration Tool to Manage the FLEXwrap License Server 1942
Installing Your Production FLEXwrap License File 1942
Starting the License Server 1945
Configuring FLEXwrap License Server to Run as a Service 1945
Manually Starting the FLEXwrap License Server from the FLEXwrap Interface 1947
Manually Starting the FLEXwrap License Server from the Command Line 1948
Stopping the License Server 1949
Checking the Status of the FLEXwrap License Server 1949
Checking the License Server Status from the FLEXwrap Server Configuration Tool 1949
Checking the License Server Status from the LMTOOLS Interface 1950
Tracking Projects with FLEXwrap 1951
Using the FLEXwrap Server Configuration Tool to FLEX-Enable Executable (.exe) Files 1951
Overview 1952
Available Options 1954
FLEX-Enabling an Executable (.exe) File 1954
Specifying Basic Settings for the License 1955
Specifying Advanced Settings for the License and Binary 1956
Creating an Options File 1960
Saving the New License File and Wrapped Application 1961
Testing a Wrapped Application 1962
Setting Up Fail Safe Modes 1963
Setting Up Regular (License) Failsafe Mode 1963
Setting Up License Server Failsafe Mode 1964
FLEXwrap Server Configuration Tool Reference 1966
FLEXwrap Terminology 1966
About FLEXwrap License Parameters 1968
Specifying FLEXwrap License Parameters 1968
Understanding FLEXwrap License File FEATURE Lines 1973
Dialog Boxes and Wizards 1975
FLEXwrap Server Configuration Dialog Box 1976
FLEXwrap Application Information Dialog Box 1980
Wrapping Wizard 1985
Encryption Wizard 1987

Wrapping and Encryption Wizard 1988
FLEXwrap Command Dialog Box 1991
Invalid Certificate Dialog Box 1992

Part 6: Using AdminStudio Enterprise Server Tools 1993

24 Generating and Viewing Reports in Report Center 1995

Generating and Viewing AdminStudio Reports 1995
Viewing Package Reports 1996
Searching for a Package on the Search Packages Page 1996
Information Included in Package Reports 1999
Package Summary Information View 2000
Files View 2001
Registry View 2002
Shortcuts View 2003
ODBC Drivers View 2004
ODBC DS View 2005
Extended Attributes View 2006
Validation View 2008
Conflicts View 2009
History View 2010
Dependencies View 2010
Navigating Through a Package Report 2011
Archiving a Package Report 2014
Exporting a Package Report 2015
Viewing the Application Readiness Dashboard 2017
Generating a Custom SQL Query Report for AdminStudio 2019

Report Center Reference 2020
All Reports Page 2020
Report View 2021
Search Packages Page 2022
Application Catalog Reports Page 2024
Viewing the Application Readiness Dashboard 2024
Exporting a Report in PDF, Excel, or Word Format 2026
Package Report 2028
Reports Wizard 2033
Select Report Type Panel 2033
Enter SQL Query Panel 2033
General Information Panel 2034
Summary Panel 2034

25 Automating Tasks Using Job Manager 2035

About Job Manager 2036

Job Steps: Tasks You Can Automate 2037
Template Jobs vs. Custom Jobs 2038
Selecting Contextual Packages in a Custom Job 2039
Dynamic Group Content Identification 2039
Conflict Resolution and the Software Repository 2040
User Permissions and Job Manager 2041
Directory Monitoring and Job Manager 2041
Managing Jobs 2042
Creating a New Job 2042
Creating a New Job Based on a Template 2042
Creating a New Custom Job 2043
Editing an Existing Job 2048
Rescheduling a Job 2049
Setting Email Alerts 2050
Viewing Job Status 2051
Deleting a Job 2052
Managing Templates 2053
Source and Target Packages in Template Jobs vs. Custom Jobs 2053
Creating a New Template 2054
Editing an Existing Template 2056
Deleting a Template 2057
Setting Job Manager Conflict Detection Options 2057
Managing Jobs Using Job Manager Engine 2059
About Job Manager Engine 2059
Opening the Job Manager Engine User Interface 2060
Adding an Application Catalog 2060
Deleting a Connection to an Application Catalog 2060
Executing Jobs 2061
Pausing, Shutting Down, and Restarting Application Catalog Jobs 2061
Viewing Job Manager Engine Status in Job Manager 2062
Setting Configuration Options 2063
Job Manager Reference 2065
Jobs Queue Page 2066
Add Job Page 2067
Edit Job Page 2070
Delete Job/Template Page 2072
Schedule Job Page 2072
Job Alerts Page 2073
Job Status Page 2073
Job Template List Page 2074
Add Template Page 2074
Edit Template Page 2076
Conflict Detection Options Page 2077

Job Manager Engine 2078
Manage Application Catalogs View 2078
Conflict Detection Configuration View 2080
Validation Configuration View 2081
E-mail Alerts View 2081

Index 2083

Part 1
Getting Started

This part of the AdminStudio 10.0 SP2 User Guide includes the following chapters:

- AdminStudio 10.0 SP2 Help Library
- Getting Started with AdminStudio
- Using the AdminStudio Interface

1

AdminStudio 10.0 SP2 Help Library

AdminStudio enables systems administrators to rapidly prepare error-free applications to deploy into their enterprise environment through a structured process built on application management best practices.

The AdminStudio Start Page provides information on major tasks that you can accomplish using AdminStudio, quick access to AdminStudio tools, and links to help resources.

The AdminStudio user documentation contains information about the functionality and features of all of the components of AdminStudio and is presented in the following sections:

Table 1-1 • AdminStudio Help Library

	Topic	Content
Part 1: Getting Started		
	Getting Started with AdminStudio	Describes how to use the AdminStudio Start Page tabs—which provide process information on how to perform key tasks using AdminStudio tools—to quickly get started evaluating and using AdminStudio.
	Using the AdminStudio Interface	Describes the AdminStudio Interface, the central application for AdminStudio. From it, you can launch the AdminStudio tools, create process templates and projects, use AdminStudio Enterprise Server tools, and connect to and create Application Catalogs.

Table 1-1 • AdminStudio Help Library (cont.)

	Topic	Content
Part 2: Managing Users, Roles, Permissions, Application Catalogs		
	Managing Users, Directory Services, and User Logins	Explains how to create an account for each person that you want to have access to AdminStudio, and how to import users or groups of users from a directory service. Also explains how to set up the AdminStudio account, domain account, single sign-on, and guest account login methods.
	Managing Roles and Permissions	Explains how to create and edit roles to manage access to AdminStudio functionality.
	Managing Application Catalog Databases	Explains how to use **Application Manager** to import applications into the Application Catalog, organize them, and set up automatic import. It also explains how to share Application Catalog data in your enterprise.
Part 3: Repackaging and Customizing Installations		
	Repackaging Legacy Installations Using the Repackaging Wizard	Explains how to use Repackager's Repackaging Wizard to convert existing legacy installations into Windows Installer (MSI) packages.
	Converting Legacy Installations Using the Repackager Interface	Explains how to use the Repackager interface to create and modify Repackager project files, and how to build those files into InstallShield Editor projects or Windows Installer packages.
	Performing Automated Repackaging and Virtualization Using the Automated Application Converter	Explains how to use the Automated Application Converter to examine a group of setups and perform automated virtualization of those setups (including peforming automated repackaging of those setups that require it).
	Using the Virtual Package Editor	Explains how to use the Virtual Package Editor to edit App-V packages and perform tasks such as customizing your App-V applications, resolving virtualization Best Practice issues and application conflicts, and fixing run-time problems.
	Creating Customized Virtual Applications	Explains how to use the InstallShield Virtualization Assistants to create customized virtual applications in the Microsoft App-V, VMware ThinApp, and Citrix XenApp virtual application formats.

Table 1-1 • AdminStudio Help Library (cont.)

	Topic	Content
	Customizing and Authoring Installations Using InstallShield	Describes how to use **InstallShield Editor** to create setup packages that utilize Windows Installer technology, while harnessing the flexibility provided by InstallScript. Also explains how to use InstallShield to create virtual applications.
	Customizing Installations Using Tuner	Explains how to use **Tuner** to create a transform file to add to, modify, or remove information from a Windows Installer package.
Part 4: Testing Windows Installer Packages		
	Identifying and Resolving Application Conflicts Using ConflictSolver	Explains how to use **ConflictSolver** to identify conflicts between Windows Installer packages before you deploy them, and resolve the problems before they affect your end users.
	Testing and Fixing Application Compatibility Using Compatibility Solver	Explains how to test for application readiness on Microsoft Windows 7 (32-bit and 64-bit) and Windows Server 2008 R2 platforms, as well as compatibility with Internet Explorer 8.0.
	Identifying and Resolving Package Errors Using PackageExpert	Explains how to use PackageExpert to automate the operation of specific tests and resolutions against Windows Installer (.msi) software packages. PackageExpert can check the integrity of your Windows Installer packages, and identify and resolve any issues relating to Windows Vista OS deployment.
	Analyzing the Impact of Installing Microsoft Operating System Patches	Explains how to import Microsoft application patches into the Application Catalog and thoroughly test the impact they will have on your environment before they are deployed.
	Isolating Applications Using Application Isolation Wizard	Explains how to use the **Application Isolation Wizard** to solve component versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.
	Ensuring Package Quality Using QualityMonitor	Explains how to use **QualityMonitor** to run a series of built-in tests to installed Windows Installer–based products, helping to ensure they run correctly, especially in a locked-down environment.

Table 1-1 • AdminStudio Help Library (cont.)

	Topic	Content
Part 5: Preparing Installations for Deployment		
	Preparing Packages for Distribution Using Distribution Wizard	Explains how to use the **Distribution Wizard** to assist you in deploying your package (and any associated transforms) to a network location, an FTP server, as an administrative installation, or using any of the following distribution technologies: • Altiris • LANDesk • ManageSoft • Marimba • Microsoft Configuration Manager • Microsoft SMS • Tivoli • ZENworks
	Distributing Packages Using Configuration Manager Web Console	Explains how to use **Configuration Manager Web Console**, a Web-based application, to configure package distribution options and distribute setup packages without going to Configuration Manager Server.
	Analyzing Installations Prior to Deployment	Explains how to use the **Predeployment Test** tool to determine if a Windows Installer `.msi` package will succeed or fail when it is installed in production by finding issues with disk space, install conditions, and conflicts with .msi packages already installed on the system.
	License-Enabling Packages Using FLEXwrap	Explains how to use **FLEXwrap** to inject software licensing technology into existing Windows Installer (.msi) packages so that they can track usage of the executables in those packages.
Part 6: Using Enterprise Tools: Job Manager and Report Center		
	Generating and Viewing Reports in Report Center	Explains how to use the **Report Center** to report on or view all of the information regarding the applications in your Application Catalog from a single location.
	Automating Tasks Using Job Manager	Explains how to use **Job Manager** to automate time consuming application migration tasks, saving you time and enabling you to enforce standardized business practices on the packaging process.

What's New in AdminStudio 10.0 SP2

Detailed information on the new features in AdminStudio 10.0 is available in the AdminStudio Release Notes You can view the Release Notes by selecting **ReadMe** from the **Help** menu of the AdminStudio interface.

A PDF version of the Release Notes is also available in the AdminStudio section of the Flexera Software Web site:

http://www.flexerasoftware.com/products/adminstudio/resources.htm

AdminStudio Editions and Components

AdminStudio 10.0 is available in Standard, Professional, and Enterprise Editions. You have the option of purchasing an additional Virtualization Pack and Application Compatibility Pack. Also, functionality varies depending upon whether you have purchased the Per Admin pricing model or the Per Desktop pricing model.

Table 1-2 • AdminStudio Editions and Components

Edition	AdminStudio	Virtualization Pack	Application Compatibility Pack
Standard	**Tools** • Repackager • Distribution Wizard • FlexWrap • InstallShield 2011 • Tuner • Application Isolation Wizard **Functionality** • Windows Installer repackaging • Windows Installer customization • Preparation for distribution	**Tools** • Automated Application Converter (Single Application Version) • Virtual Package Editor • Microsoft App-V Assistant • ThinApp Assistant • Citrix Assistant **Functionality** • Convert to virtual applications • App-V package editing	N/A

Table 1-2 • AdminStudio Editions and Components (cont.)

Edition	AdminStudio	Virtualization Pack	Application Compatibility Pack
Professional	**Tools** Same as **Standard**, plus: • ConflictSolver • Application Manager • OS Snapshot Wizard • QualityMonitor • Predeployment Test **Functionality** Same as **Standard**, plus: • Application management • Testing, validation, and conflict testing • Standard reporting	**Tools** Same as **Standard**, plus: • Automated Application Converter (Single Application Version) with Virtualization Suitability Testing **Functionality** Same as **Standard**, plus: • Test for virtualization suitability • Store App-V data in Application Manager • App-V package testing, validation, and conflict testing • Automated conversion of legacy packages to Windows Installer	**Tools & Functionality** **Compatibility Solver** with the following functionality: • Project-level compatibility assessment and cost/effort reporting for all applications • Application testing and reporting for Windows 7 (32- and 64-bit versions) • Auto-fixing for Windows 7 and Windows Server 2008 and 2008 R2
Professional Complete	Includes Professional Edition with both the **Virtualization Pack** and the **Application Compatibility Pack**.		

Table 1-2 • AdminStudio Editions and Components (cont.)

Edition	AdminStudio	Virtualization Pack	Application Compatibility Pack
Enterprise (Per Admin)	**Tools** Same as **Professional**, plus: • PackageExpert • OS Security Patch Wizard • Job Manager • Report Center • Software Repository • Security Console **Functionality** Same as **Professional**, plus: • Application Readiness Dashboard • Process and workgroup management features	**Tools** Same as **Professional**, plus: • Automated Application Converter (Multiple Application Version) with Suitability Testing **Functionality** Same as **Professional**, plus: • Automated conversion of multiple packages to virtual applications • Automated conversion of multiple legacy packages to Windows Installer • Application Readiness Dashboard (Including Virtualization Reports)	**Tools & Functionality** All of the **Compatibility Solver** functionality available in Professional, plus: • Detailed compatibility assessment reports • Application testing and reporting for Windows 7 (32- and 64-bit versions), and Windows Server 2008 and 2008 R2
Enterprise (Per Admin) Complete	Includes Enterprise Edition (Per Admin) with both the **Virtualization Pack** and the **Application Compatibility Pack**.		

Table 1-2 • AdminStudio Editions and Components (cont.)

Edition	AdminStudio	Virtualization Pack	Application Compatibility Pack
Enterprise (Per Desktop)	**Tools & Functionality** Same as **Enterprise (Per Admin)**	**Tools & Functionality** Same as **Enterprise (Per Admin)**	**Tools & Functionality** All of the **Compatibility Solver** functionality available in **Enterprise (Per Admin)**, plus: • Testing can be done on groups of applications at a time **Web Application Compatibility Pack (Optional Add-On)** • Test compatibility of web applications with Internet Explorer 8 browser • Test integration of desktop application with Internet Explorer 8 browser
Enterprise (Per Desktop) Complete	Includes Enterprise Edition (Per Desktop) with both the **Virtualization Pack** and the **Application Compatibility Pack**. ***Note*** • *Web Application Compatibility Pack is available as an optional add-on feature.*		

Documentation Edition Notes

Documentation on features that are only available in specific Editions or add-on Packs include the following notes:

Table 1-3 • Documentation Edition Notes

Edition/Pack	Note
Professional	***Edition*** • *This feature is included with AdminStudio Professional and Enterprise Editions.*
Enterprise	***Edition*** • *This feature is included with AdminStudio Enterprise Edition.*
Virtualization Pack	***Edition*** • *This feature is included in the AdminStudio Virtualization Pack.*
Application Compatibility Pack	***Edition*** • *This feature is included in the AdminStudio Application Compatibility Pack.*

Note • *For a comprehensive list of the features available in the AdminStudio Standard, Professional, and Enterprise Editions, plus the add-on features available for Enterprise Edition, see the AdminStudio Web site.*

Activating AdminStudio

When you launch AdminStudio or one of its tools for the first time, you are notified that you are using a time-limited trial version, and you are given the opportunity to evaluate the product or to activate it by entering a valid Serial Number for an AdminStudio Edition.

Task: ***To activate AdminStudio:***

1. Install AdminStudio, as described in the *AdminStudio Client Tools Installation Guide*.
2. Launch AdminStudio or one of its tools. A dialog box opens, stating that you are using a time-limited trial version.
3. Select **Activate or Purchase AdminStudio** and click **Next**. The **AdminStudio Product Activation** dialog box opens, prompting you to enter a serial number.
4. Enter the serial number of the edition you purchased and click the **Activate** button. After a few seconds, you will receive a message that activation was successful.
5. Click **Finish**. AdminStudio will launch.

Note • *If you want to evaluate the AdminStudio Enterprise Server Web tools (Job Manager, Report Center, Security Console), you need to obtain an Enterprise Server Evaluation Serial Number from an AdminStudio Sales Representative.*

Ports Used in Activation

AdminStudio product activation uses ports 80 (HTTP), 443 (HTTPS), and 8443. If these ports are locked down or if you do not have an available internet connection, you can activate AdminStudio using one of the following alternative methods:

- **Self-hosted activation using FLEXnet License Server**—Your organization can choose to purchase and use a self-hosted FLEXnet License Server to manage their AdminStudio licenses. Using the FLEXnet License Server eliminates the need to manually activate each installation of AdminStudio over the internet.
- **Offline activation**—You can perform offline activation using email.

Silent Activation

AdminStudio supports silent activation, enabling you to use public Windows Installer properties to enter the Serial Number automatically and activate during installation. To enable silent activation, you need to set the `PRODUCTID` and `ASACTSTATUS` public Windows Installer properties. The following is an example of how to accomplish this via the command line:

```
<AdminStudio_Install_Directory>Setup.exe /v"PRODUCTID=XXXXXXX-XXX-XXXXXXXXXX ASACTSTATUS=1"
```

You can additionally make the installation completely silent by adding the `/s` and `/qn` switches as follows:

```
<AdminStudio_Install_Directory>Setup.exe /s /v"PRODUCTID=XXXXXXX-XXX-XXXXXXXXXX ASACTSTATUS=1 /qn"
```

Note • *Consult the Windows Installer Help Library for any additional Windows Installer command line options.*

Evaluating AdminStudio

You can choose to evaluate AdminStudio for 21 days. By clicking **Continue to Evaluate AdminStudio** dialog box that opens when you launch AdminStudio, you can begin evaluating the AdminStudio 10.0 SP2 Enterprise Edition client tools. To evaluate the AdminStudio Enterprise Server Web tools, you need to contact AdminStudio Sales and obtain an evaluation serial number.

Information about evaluating AdminStudio is presented in the following sections:

- Evaluating AdminStudio Enterprise Edition Client Tools
  - Evaluating AdminStudio's Microsoft App-V Support
  - Evaluating the Automated Application Converter "Multiple Application" Option
- Evaluating the AdminStudio Enterprise Server Web Tools

Evaluating AdminStudio Enterprise Edition Client Tools

To evaluate the Enterprise Edition client tools, perform the following steps.

Task: ***To evaluate the AdminStudio Enterprise Edition client tools:***

1. Install AdminStudio, as described in the *AdminStudio Client Tools Installation Guide*.
2. Launch AdminStudio. A dialog box opens, stating that you are using a time-limited trial version.
3. If you want to evaluate AdminStudio, select **Continue to Evaluate AdminStudio** and click **Next** (or just wait ten seconds). The product will launch.
   - When you run AdminStudio in this trial mode, all of its features are fully available.
   - Each time you open AdminStudio while you are in evaluation mode, this dialog box shows you how many days are left in your trial period.
4. If you have five or fewer days left in your trial period, the dialog box remains open, requiring you to click before you can proceed. Do one of the following:
   a. If your trial period is not over, you can continue to use AdminStudio by selecting the **Continue to Evaluate AdminStudio** option and clicking **Next**.
   b. If you have already purchased a serial number or want to purchase one online, select **Activate or Purchase AdminStudio** and click **Next**.

Evaluating AdminStudio's Microsoft App-V Support

While evaluating the AdminStudio Enterprise Edition client tools, you will be able to convert a Windows Installer package to an App-V application using the Automated Application Converter, Repackager, and the InstallShield App-V Assistant. However, an App-V application built using an evaluation version of AdminStudio will display the following message every time it is launched:

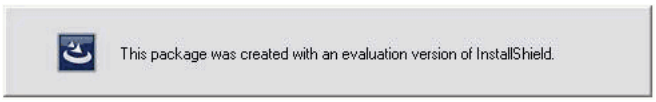


Figure 1-1: Evaluation Version Message

After purchasing the AdminStudio Virtualization Pack, you will be able to remove this message by rebuilding the App-V application.

Evaluating the Automated Application Converter "Multiple Application" Option

The Multiple Application option of Automated Application Converter is only available when you purchase the Virtualization Pack with AdminStudio Enterprise Edition.

***Note** • If you purchase the Virtualization Pack with AdminStudio Standard or Professional Editions, you will only be able to convert one package at a time, using one virtual machine.*

When using an evaluation version of AdminStudio, you will be able to use Automated Application Converter to convert a directory full of Windows Installer packages into individual virtual packages, but the conversion will be limited to three packages per run, using only one virtual machine. Therefore, only the first three packages that Automated Application Converter encounters will be converted to virtual applications.

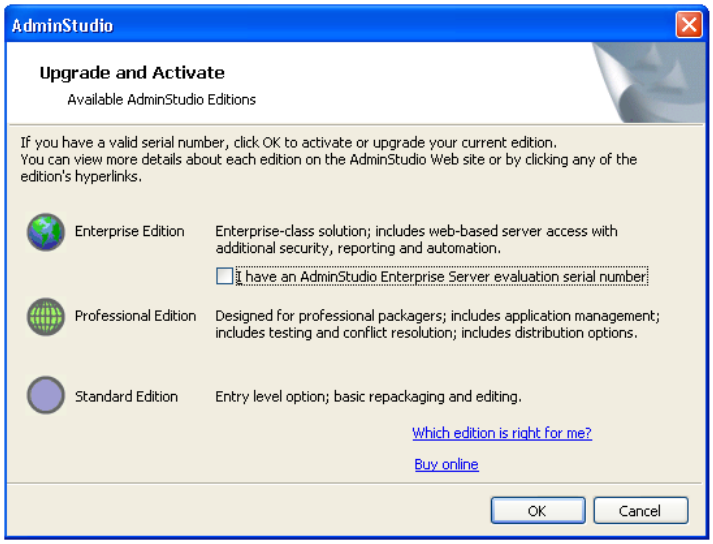
Evaluating the AdminStudio Enterprise Server Web Tools

If you want to evaluate the AdminStudio Enterprise Server Web tools (Job Manager, Report Center, Security Console), you need to obtain an AdminStudio Enterprise Server Evaluation Serial Number from an AdminStudio Sales Representative.

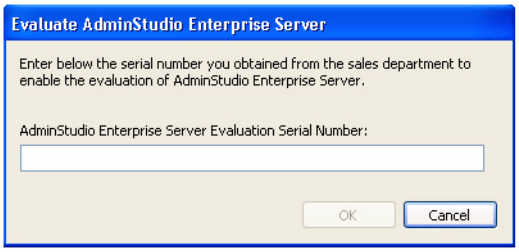
Task: ***To evaluate the AdminStudio Enterprise Server Web tools:***

1. Follow the steps in Evaluating AdminStudio Enterprise Edition Client Tools.
2. Contact an AdminStudio Sales Representative and obtain an AdminStudio Enterprise Server Evaluation Serial Number.
3. Open the AdminStudio interface and select **About AdminStudio** from the **Help** menu.

4. Click **Activate**. The **Upgrade and Activate** dialog box opens.



5. Select the **I have an AdminStudio Enterprise Server evaluation serial number** option and click **OK**. The **Evaluate Enterprise Edition** dialog box opens.



6. Enter the serial number and click **OK**. You will be prompted to install the AdminStudio Enterprise Server on the current machine.
7. Click **Yes**. The AdminStudio Enterprise Server installation begins.
8. Install AdminStudio Enterprise Server, as described in the *AdminStudio and Workflow Manager Installation Guide*, which is available on the AdminStudio Downloads site.

Upgrading Your Product Edition

An upgrade feature has been built-in to AdminStudio that allows you to activate features in a higher Edition without re-installing the application. You just need to enter a Serial Number for the upgrade that you purchased, and the features of that Edition are immediately unlocked and are available to you.

To upgrade, perform the following steps:

Task: ***To upgrade your AdminStudio Edition:***

1. Contact an AdminStudio Sales Representative and purchase a Serial Number for the desired Edition.
2. Launch AdminStudio.
3. On the **Help** menu, click **About AdminStudio**. The **About AdminStudio** dialog box opens.
4. Click the **Upgrade** button. The **Upgrade and Activate** dialog box opens.
5. Click **OK** to upgrade your edition. The **AdminStudio Product Activation** dialog box opens, prompting you to enter the serial number of the edition that you want to upgrade to.
6. Enter the serial number of the edition you purchased and click the **Activate** button. After a few seconds, you will receive a message that activation was successful. The functionality of the upgraded edition is immediately available to you.

Using Help

Help is available both from the AdminStudio interface **Help** menu and directly from certain individual interface elements.

When you have questions about this product, first consult the AdminStudio Help Library, which is the complete user's guide for using AdminStudio.

Web-Based Online Help

Web-based online help is available to you 24 hours a day, seven days a week, on our Web site at:

http://helpnet.flexerasoftware.com

Documentation as PDF

AdminStudio documentation is also available as a PDF. Visit the AdminStudio Downloads site at:

http://www.flexerasoftware.com/downloads.htm

Help Conventions

In this documentation, reader alert and style conventions are used to bring your attention to specific information or help you identify information.

- Reader Alert Conventions
- Style Conventions

Reader Alert Conventions

Reader alerts are used throughout this documentation to notify you of both supplementary and essential information. The following table explains the meaning of each alert.

Table 1-4 • Reader Alert Conventions

Image	Alert Name	Description
	Note	Notes are used to draw attention to pieces of information that should stand out.
	Important Note	Important notes are used for information that is essential for users to read.
	Caution	Cautions indicate that this information is critical to the success of the desired feature or product functionality.
	Tip	Tips are used to indicate helpful information that could assist you in better utilizing the desired function or feature.

Table 1-4 • Reader Alert Conventions (cont.)

Image	Alert Name	Description
	Best Practices	Best Practices alerts instruct you on the best way to accomplish a task.
	Edition-Specific Note	Edition-specific notes indicate that the information applies to a specific edition of a product (such as Professional or Premier edition).
	Procedure	The Procedure graphic indicates that procedural instructions follow.

Style Conventions

The following style conventions are used throughout this documentation.

Table 1-5 • Style Conventions

Style	Example	Description
User Interface Elements	On the **File** menu, click **Open**.	User interface elements appear in bold when referenced in tasks.
Variables	*fileName*	Variables appear in italics.
Code	`#define HWND_BROADCAST 0xffff`	Code snippets appear in a monospace typeface.
User-Inputted Text	Type `$D(install)`.	Text that is to be entered as a literal value is displayed in a monospace typeface, in bold, and in blue.
File Names and Directory Paths	My files are located in the `C:\MyDocuments\SampleCode` directory.	File names and directory paths are presented in a monospace typeface.
.INI File Text	Insert the line `LimitedUI=Y` into the file to display only the Welcome dialog box when the Windows Installer package is run.	Text in .INI files is presented in a monospace typeface.
Command-Line Statements	To run the installation silently, enter: `Setup.exe /s /v/qn`	Command-line statements and parameters are presented in a monospace typeface.
Environment Variables	Set the value of the `windir` environment variable.	Environment variables are presented in a monospace typeface.

Table 1-5 • Style Conventions (cont.)

Style	Example	Description
Examples	Create two groups, one called **Admin** and the other called **General**.	Examples are presented in bold.
Functions	**FeatureAddItem** adds a new feature to a script-created feature set.	Functions are presented in presented in bold.
Properties	In the **Name** property, enter a name for this custom control that is unique among all of the controls in your project.	Properties are presented in bold.
Screen Output	If you type an incorrect parameter, the message `The system cannot find the path specified.` is displayed.	Screen output (from a log file or from the console) is displayed in a monospace typeface, and in blue.
Links	Obtain the latest white papers, project samples, and more from: `http://www.mycompany.com/downloads.htm`	Links appear in blue.

Contacting Us

You may contact us from anywhere in the world by visiting our Web site at:

http://www.flexerasoftware.com

2

Getting Started with AdminStudio

The AdminStudio Start Page, which is designed to help you quickly get started evaluating and using AdminStudio, provides process information on how to perform key tasks using AdminStudio tools. Information is organized into the following tabs:

Table 2-1 • AdminStudio Start Page Organization

Icon	Start Page Tab	Description
	Getting Started	Describes the main tasks that you can use AdminStudio to accomplish, and provides links to additional information. See Getting Started Tab.
	Test for Application Compatibility	Provides a flowchart that outlines how to use Compatibility Solver to test for application readiness on Microsoft Windows 7 32-bit and 64-bit platforms, as well as compatibility with Internet Explorer 8.0. See Test for Application Compatibility Tab.
	Migrate to Application Virtualization	Provides a flowchart that outlines the steps required to migrate your application portfolio into virtual applications that are ready for deployment within the enterprise. See Migrate to Application Virtualization Tab.
	Migrate to Windows Installer	Provides a flowchart that outlines the steps required to migrate legacy setups (such as `.exe` files) to deployable Windows Installer packages (`.msi`). See Migrate to Windows Installer Tab.
	Set Up Infrastructure	Lists the infrastructure setup steps that you need to perform prior to using AdminStudio for the first time: connect to an Application Catalog, configure virtual machines, set e-mail notification settings, and configure Compatibility Solver. See Set Up Infrastructure Tab.

Getting Started Tab

The **Getting Started** tab of the AdminStudio Start Page describes the main tasks that you can use AdminStudio to accomplish, and provides links to additional information. It also provides a link to information on setting up AdminStudio infrastructure.

To quickly get started evaluating and using AdminStudio, click on the link of the task you want to accomplish:

Table 2-2 • Getting Started Tab

Link	Description
Test for Application Compatibility	Test for application readiness on Microsoft Windows 7 (32-bit and 64-bit) and Windows Server 2008 R2 platforms, as well as compatibility with Internet Explorer 8.0.
Migrate to Application Virtualization	Automatically repackage and convert Windows Installer packages, as well as setups in other formats, into virtual applications in Microsoft App-V, VMware ThinApp, and Citrix XenApp formats.
Migrate to Windows Installer	Capture, repackage, and customize installations, analyze packages for conflicts with target applications, and prepare packages for distribution.
Set Up Infrastructure	Create/connect to a Microsoft SQL Server Application Catalog database. Prepare virtual machines for use in automated repackaging and testing.

Test for Application Compatibility Tab

Edition • *AdminStudio Compatibility Solver is included in the AdminStudio Application Compatibility Pack. Support for Internet Explorer 8 is included in the Web Application Compatibility Pack.*

The flowchart on this tab outlines how to use Compatibility Solver to test for application readiness on Microsoft Windows 7 (32-bit and 64-bit) and Windows Server 2008 R2 platforms, as well as compatibility with Internet Explorer 8.0.

Note • *Prior to performing these steps, you should have already set up infrastructure as outlined in the Set Up Infrastructure Tab.*

To test for application compatibility, perform the following steps:

- Perform Portfolio Application Compatibility Reporting (Migration Project Scoping)
- Pinpoint Specific Compatibility Issues
- Fix Compatibility Issues

Perform Portfolio Application Compatibility Reporting (Migration Project Scoping)

For instructions on how to perform portfolio application compatibility reporting, see the following help topics:

Table 2-3 • Perform Portfolio Application Compatibility Reporting (Migration Project Scoping)

#	Step	Description
1	**Browse for or Drag and Drop Packages**	See Importing Packages into Compatibility Solver.
2	**Select the Platform Reports to Run**	See Selecting the Platforms for the Summary Application Compatibility Report.
3	**Load Packages to Test**	See Loading Packages in Compatibility Solver.
4	**View Summary Application Compatibility Report**	See Viewing the Summary Application Compatibility Report.

Pinpoint Specific Compatibility Issues

For instructions on how to pinpoint specific compatibility issues, see the following help topics:

Table 2-4 • Pinpoint Specific Compatibility Issues

#	Step	Description
5	**Select Packages to Test**	See Selecting Packages.
6	**Run Compatibility Assessment of Packages**	See Running Reports.
7	**View the Status and Details of Detected Compatibility Issues**	See Viewing Reports.
8	**Click on the Fixes Available Icon to View Description of Issues With Fixes**	See Viewing Descriptions of Issues and Available Fixes.

Fix Compatibility Issues

For instructions on how to fix compatibility issues, see the following help topics:

Table 2-5 • Fix Compatibility Issues

#	Step	Description
9	**Select Packages to Fix**	See Fixing Application Issues.
10	**Click Fix to Automatically Remediate the Selected Issues by Generating Transform Files**	See Fixing Application Issues.

Migrate to Application Virtualization Tab

The flowchart on this tab outlines the steps required to migrate your application portfolio into virtual applications that are ready for deployment within the enterprise.

Note • *Prior to performing these steps, you should have already set up infrastructure as outlined in the Set Up Infrastructure Tab.*

To migrate a Windows Installer or legacy application to a virtual package, perform the following steps:

- Identify Packages to Virtualize
- Convert to Virtual Formats
- Test and Distribute Converted Packages

Identify Packages to Virtualize

For instructions on how to identify the packages to virtualize, see the following help topics:

Table 2-6 • Identify Packages to Virtualize

#	Step	Description
1	**Import Packages into Application Catalog**	For instructions on how to import Windows Installer and legacy applications into the Application Catalog, see: • Importing a Single Windows Installer or Virtual Package • Importing a Directory of Windows Installer and/or App-V Packages • Importing Windows Installer and/or App-V Packages From Microsoft Configuration Manager • Importing Other Setup Types
2	**View Virtualization Suitability Report**	For instructions on how to view the virtualization readiness status of imported packages, see App-V Virtualization Readiness Chart.
3	**Identify Candidates for Virtualization**	Based upon the virtualization readiness information reported, decide which of the imported applications to select for virtualization.

Convert to Virtual Formats

For instructions on how to convert a package to a virtual format, see the following help topics:

Table 2-7 • Convert to Virtual Formats

#	Step	Description
4	**Import Candidate Packages into Automated Application Converter**	For instructions on how to import candidate packages into Automated Application Converter, see: • Selecting Packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server • Selecting Packages from a Local Machine or Network
5	**Convert to Virtual Packages**	For instructions on convert selected packages to virtual packages, see Performing a Conversion Using the Application Conversion Wizard.
6	**Test Launch Virtual Packages**	For instructions on how to test the virtual packages that you have just created, see Launching Packages for Testing.
7	**Publish Virtual Packages to Application Catalog**	For instructions on how to publish converted packages to the Application Catalog, see Publishing Converted Packages.

Test and Distribute Converted Packages

For instructions on how to test and distribute converted virtual packages, see the following help topics:

Table 2-8 • Test and Distribute Converted Packages

#	Step	Description
8	**Perform Validation Against Virtualization Best Practices**	For instructions on how to validate App-V packages against virtualization Best Practice rules, see Performing Best Practice Analysis of App-V Packages.
9	**Perform Conflict Analysis Against Other Packages**	For instructions on how to perform conflict analysis of an App-V packages other packages, see Performing Conflict Analysis of App-V Packages.
10	**Edit App-V Packages (If Necessary)**	To resolve any warnings or errors that were found during testing, you can edit the App-V package in the Virtual Package Editor, as described in Using the Virtual Package Editor.

Table 2-8 • Test and Distribute Converted Packages

#	Step	Description
11	**Distribute to Enterprise for User Acceptance Testing and Production**	For instructions on how to distribute an App-V package to your enterprise, see Preparing Packages for Distribution Using Distribution Wizard.

Migrate to Windows Installer Tab

The flowchart on this tab outlines the steps required to migrate legacy applications to Windows Installer packages that are ready for deployment within the enterprise.

Note • *Prior to performing these steps, you should have already set up infrastructure as outlined in the Set Up Infrastructure Tab.*

To migrate a legacy application to a Windows Installer package, perform the following steps:

- Repackage Legacy Package
- Import Into Application Catalog
- Test and Distribute Repackaged Applications

Repackage Legacy Package

For instructions on how to repackage a legacy package, see the following help topics:

Table 2-9 • Repackage Legacy Package

#	Step	Description
1	**Select legacy packages (.exe)**	For instructions about how to get started with repackaging, see About Repackaging.
2	**Repackage to Windows Installer Package (.msi)**	For instructions on how to repackage a Windows Installer package, see Repackaging Legacy Installations Using the Repackaging Wizard.
3	**Edit Packages in Repackager**	For instructions on how to edit a Repackager project, see Working With Repackager Projects.

Import Into Application Catalog

For instructions on how to import a package into the Application Catalog, see the following help topics:

Table 2-10 • Repackage Legacy Package

#	Step	Description
4	**Import Windows Installer Package Into Application Catalog**	For instructions on how to import a Windows Installer package into the Application Catalog, see Importing a Single Windows Installer or Virtual Package.

Test and Distribute Repackaged Applications

For instructions on how to test and distribute a repackaged application, see the following help topics:

Table 2-11 • Test and Distribute Repackaged Applications

#	Step	Description
5	**Validate Package Against MSI Best Practices (ICE Rules)**	For instructions on how to validate a package against Microsoft ICE rules, see Performing Validation.
6	**Check Package Integrity Against Best Practices (ACE Rules)**	For instructions on how to validate a package against AdminStudio Best Practice ACE rules, see Checking for Conflicts.
7	**Perform Conflict Detection and Resolution**	For instructions on how to perform conflict analysis and resolution, see Checking for Conflicts and Resolving Conflicts.
8	**Distribute Package**	For instructions on how to distribute a package, see Preparing Packages for Distribution Using Distribution Wizard.

Set Up Infrastructure Tab

To get started using AdminStudio, you need to connect to a Microsoft SQL Server Application Catalog database, and prepare virtual machines for use in automated repackaging and testing. To get started using Compatibility Solver, you need to create a new database and download the latest report upgrades.

- Create/Connect to an Application Catalog
- Configure Virtual Machines
- Set E-Mail Notification Settings
- Configure Compatibility Solver

Create/Connect to an Application Catalog

For instructions on create a new Application Catalog or connect to an existing one, see the following help topics:

Table 2-12 • Create/Connect to an Application Catalog

Step	Description
Create/Connect to an Application Catalog	For instructions on how to create or connect to an Application Catalog, see Creating and Connecting to Application Catalogs.
Enable Software Repository	For instructions on enabling the Software Repository, see Using the Software Repository.
Set Default Application Catalog	For instructions on setting the default Application Catalog, see Specifying a Default AdminStudio Application Catalog.

Configure Virtual Machines

For instructions on how to configure virtual machines for use with Automated Application Converter, see the following help topics:

Table 2-13 • Configure Virtual Machines

Step	Description
Run Virtual Machine Preparation Setup to Enable Auto Login	For instructions on how to run the virtual machine preparation setup, see Preparing Your Virtual Machines for Use With the Automated Application Converter and Running the Virtual Machine Preparation Setup.
Create a Snapshot	For instructions on how to create a snapshot on your virtual machine, see Taking a Snapshot.
Install VMware VIX	For instructions on how to install VMware VIX, see VMware VIX API Requirement.
Test Virtual Machine Setup by Converting a Simple Package	For instructions on how to test a virtual machine setup, see Using the Application Conversion Wizard to Perform Automated Package Conversion.

Set E-Mail Notification Settings

For instructions on how to set e-mail notification settings, see the following help topics:

Table 2-14 • Set E-Mail Notification Settings

Step	Description
Set SMTP Notification Settings	For instructions on setting SMTP notification settings, see Setting E-Mail Notification Settings.

Configure Compatibility Solver

For instructions on how to configure Compatibility Solver, see the following help topics:

Table 2-15 • Configure Compatibility Solver

Step	Description
Create Compatibility Solver Database	See Creating a New Database.
Load Report Upgrades	See Downloading and Importing Reports (Plugins).

3

Using the AdminStudio Interface

The AdminStudio interface is the central application for AdminStudio. From it, you can launch AdminStudio tools, launch the Process Assistant, create workflow templates and projects, and connect to and create Application Catalogs. You can launch the AdminStudio interface from the Windows Start menu.

Note • *The Process Assistant is a procedure-based alternative interface to the standard AdminStudio interface. For more information, see Using the Process Assistant.*

AdminStudio interface documentation is presented in the following sections:

Table 3-1 • AdminStudio interface Documentation

Section	Description
Configuring the AdminStudio Interface	Includes information on customizing the AdminStudio Interface, including setting shared directories.
Working with Tools	Explains how to configure AdminStudio tools and also external tools necessary to perform workflow tasks.
Workflows and Projects	Explains how to use workflow functionality to define repeatable processes to accomplish your goals.
Frequently Asked Questions	A list of questions frequently asked by AdminStudio users, with links to the appropriate help topics.
AdminStudio Interface Reference	This section contains an exhaustive description of each dialog box, Wizard, and UI element in AdminStudio.

Configuring the AdminStudio Interface

This section explains how to configure and customize the AdminStudio interface. The following topics are discussed:

Table 3-2 • Topics on AdminStudio Interface Configuration

Category	Topics
Launching Applications	• Launching AdminStudio Applications
Configuring Application Catalog Settings	• Specifying the Shared AdminStudio Application Catalog • Specifying the AdminStudio Shared Location
Setting Interface Preferences	• Setting E-Mail Notification Settings • Setting the Workflow Task Help Page Location • Configuring How Often AdminStudio Checks for Updates • Configuring AdminStudio to Stay on Top • Generating a Debug Log for AdminStudio
Configuring Interactivity Settings	• Specifying the AdminStudio Enterprise Server URL • Specifying the Configuration Manager Web Console URL

Launching AdminStudio Applications

Individual AdminStudio applications, such as ConflictSolver, InstallShield Editor, or Repackager, can be launched from the Windows Start menu or by double-clicking on the tool icon in the Tools Gallery on the Tools Tab.

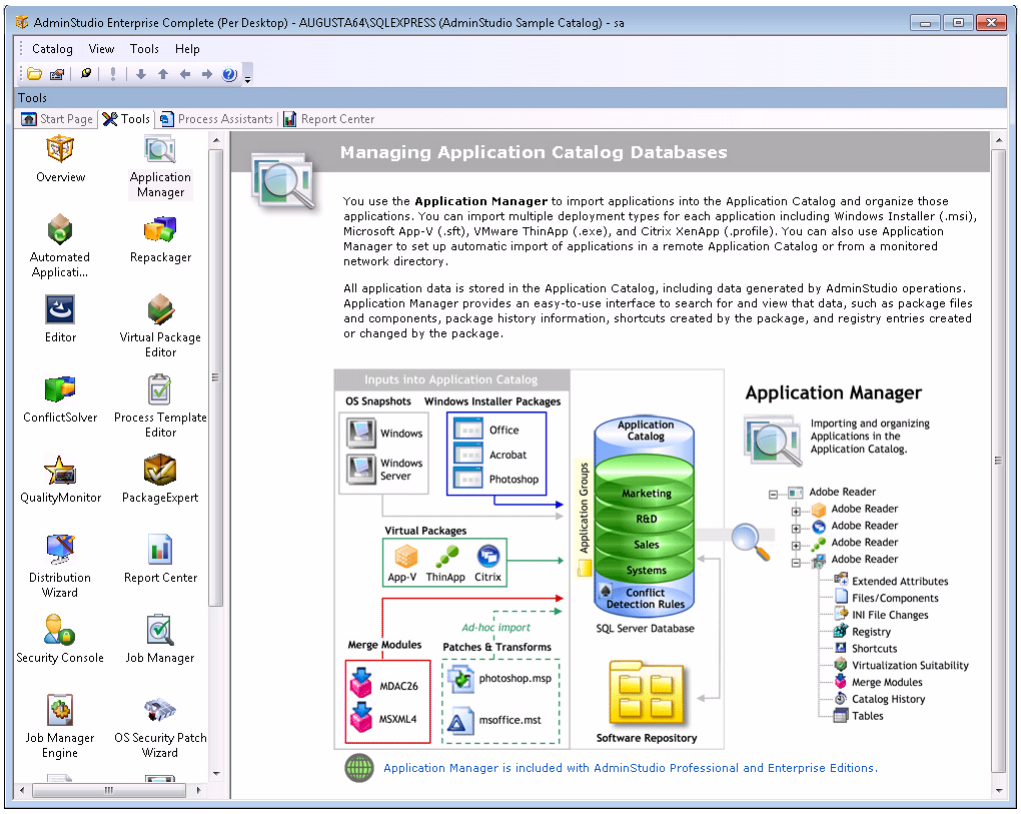


Figure 3-1: Tools Gallery on the Tools Tab

Note • *If a core AdminStudio tool is not in the Tools Gallery or available from the menu, it may mean that either you are not assigned to a Role that has permission to use that Tool, or that the Tool is not available in your Edition of AdminStudio.*

Note • *You can also launch AdminStudio tools from the Process Assistant. See Using the Process Assistant for more information.*

Specifying the Shared AdminStudio Application Catalog

This field displays the name and location of the shared AdminStudio Application Catalog. Click the Browse button to connect to another Application Catalog using the Connect to Application Catalog dialog box.

Task: ***To specify the location of the AdminStudio Shared Location:***

1. Launch the AdminStudio interface.
2. From the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. Select the **Application Catalog** tab.
4. Select the **Shared AdminStudio Application Catalog**.
5. Click **OK** to close the **Options** dialog box.

Specifying the AdminStudio Shared Location

The AdminStudio Shared Directory (also referred to as the AdminStudio Shared Location) contains shared information for repackaging and conflict identification, and other AdminStudio functions. The AdminStudio Shared Directory contains the following:

- The Shared `AdminStudio.ini` file, which specifies default Application Catalog database settings
- Application Manager duplicate package identifier options
- Repackager `isrepackager.ini` exclusion list
- OS Snapshot `issnapshot.ini` file
- ConflictSolver user-defined ACEs
- Distribution Wizard Distribution Type templates and `.ini` files

If you are working in a team environment, the AdminStudio Shared Directory should be set to a centralized network location, accessible by all AdminStudio users at your organization, rather than on your local machine. Follow the steps below to specify the location of the AdminStudio Shared Directory.

Note • *To maintain consistency when creating workflows, InstallShield recommends that you set the AdminStudio Shared Directory the same for each AdminStudio user.*

Task: ***To specify the location of the AdminStudio Shared Directory:***

1. Launch the AdminStudio Interface.
2. From the **Tools** menu, select Options. The **Options** dialog box opens.
3. In the **Options** dialog box, select the **Locations** tab.
4. Enter or browse to the directory for the **AdminStudio Shared Location**.
5. Click **OK** to close the **Options** dialog box.

Setting E-Mail Notification Settings

To enable AdminStudio to send you e-mail notifications during various processes, you need to configure your SMTP notification settings.

Note • *Currently, e-mail notifications are sent when soft time-outs are encountered while using Automated Application Converter to repackage an application on a virtual machine.*

To set your e-mail notification settings, perform the following steps:

Task: ***To set e-mail notification settings:***

1. Launch the AdminStudio interface.
2. From the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. Select the **Notification Settings** tab.
4. On the **Notification Settings** tab, enter the following information:

Option	Description
SMTP Server	Enter the address of your e-mail server, such as: `smtp.yourcompany.com`
Authentication	Specify how your e-mail is authenticated by selecting one of the following options: • **Server Authentication**—Select this option if you want to perform server authentication on your AdminStudio e-mail. • **Anonymous**—Select this option if you do not want to perform authentication on your AdminStudio e-mail.
Domain	Enter the Domain of the user account listed in the **User Name** field.

Option	Description
User Name	Enter the name of an existing user account in the Domain specified in the **Domain** field. This user must have permission to send e-mail.
Password	Enter the password of the user account defined in the **User Name** and **Domain** fields. ***Note*** • *If your network domain requires that user passwords are changed periodically, you will have to open this dialog box again to update this account's password. To avoid this, try to obtain a user account that has a password that does not expire.*
From E-Mail ID	Enter the e-mail address to serve as the identity of AdminStudio. All e-mails sent by AdminStudio will have this e-mail address in the **From** field.
To E-Mail ID(s)	Enter the e-mail address to serve as the system account for AdminStudio e-mail. All e-mails sent to AdminStudio will be sent to this address. ***Note*** • *Use a semi-colon to separate multiple e-mail addresses in the **To E-Mail ID(s)** field.*
SMTP Server Port	Enter the port of your SMTP server.
Use SSL	Select this option if you want to use SSL security for the AdminStudio e-mail account.

5. Click **OK**.

Setting the Workflow Task Help Page Location

The **Task Help Page Location** is the directory where you want to store all HTML pages that serve as workflow task instructions.

Task: ***To specify the location of task help pages:***

1. Launch the AdminStudio interface.
2. From the **Tools** menu, select **Options**.The **Options** dialog box opens.
3. Select the **Locations** tab.
4. Enter or browse to the **Task Help Page Location**, the location where workflow task help pages are stored.
5. Click **OK** to close the **Options** dialog box.

Configuring How Often AdminStudio Checks for Updates

You specify how often you want AdminStudio to check for updates on the **Updates** tab of the **Options** dialog box.

Task: ***To configure how often AdminStudio checks for updates:***

1. Open AdminStudio.
2. On the **Tools** menu, click **Options**. The **Options** dialog box opens.
3. On the **Updates** tab, select the frequency that AdminStudio will check for software updates:
   - Once every 15 days
   - Once every 30 days
   - Once every 60 days
   - Never
4. Click **OK** to close the **Options** dialog box.

Configuring AdminStudio to Stay on Top

When you launch AdminStudio tools, you can specify whether they open in front of or behind the AdminStudio interface. If you select **Always on Top** from the **View** menu, application will always open behind the AdminStudio interface.

Task: ***To configure AdminStudio to stay on top of other applications:***

1. Launch the AdminStudio interface.
2. From the **View** menu, select **Always on Top**.

Generating a Debug Log for AdminStudio

To create a debug log for AdminStudio, perform the following steps.

Task: ***To generate a debug log for AdminStudio:***

Use the following registry value to turn debugging on. Once this debugging is turned on, a log file will be created in the same location as the `.exe` file.

```
[HKLM\Software\InstallShield\AdminStudio] DebugLogLevel="3"
```

Levels 0, 1, 2, 3 are supported with 3 being the highest. Default is level 0.

Specifying the AdminStudio Enterprise Server URL

You specify the Web site URL of AdminStudio Enterprise Server on the **Application Catalog** tab of the **Options** dialog box.

Task: ***To specify the AdminStudio Enterprise Server URL:***

1. Launch the AdminStudio interface.
2. On the **Tools** menu, click **Options**. The **Options** dialog box opens.
3. Select the **Application Catalog** tab.
4. Enter the **AdminStudio Enterprise Server URL**.
5. If you have more than one AdminStudio Enterprise Servers, select the **Make this the default Shared AES URL** to make the URL in the **AdminStudio Enterprise Server URL** field the default.
6. If you want to **Allow automatic authentication** by AdminStudio Enterprise Server, select this option.
7. Click **OK** to close the **Options** dialog box.

Specifying the Configuration Manager Web Console URL

AdminStudio 10.0 SP2

Enter the URL for the Configuration Manager Web Console Web site. You specify the Web site URL of the Configuration Manager Web Console on the **Locations** tab of the **Options** dialog box.

Task: ***To specify the Configuration Manager Web Console URL:***

1. Launch the AdminStudio interface.
2. On the **Tools** menu, click **Options**. The Options Dialog Box opens.
3. Select the **Locations** tab.
4. Enter the **Configuration Manager Web Console URL**.
5. Click **OK** to close the **Options** dialog box.

Working with Tools

Topics in this section involve adding, configuring, and associating tools in the **Tools Gallery** on the Tools Tab. This section contains the following topics:

- Adding New Tools to the Tools Gallery
- Editing Properties for an Existing Tool
- Adding Command-Line Configurations for an Existing Tool
- Modifying Command-Line Configurations for an Existing Tool
- Deleting Command-Line Configurations from an Existing Tool
- Associating Tools with Tasks
- Running Associated Tools in Projects
- Deleting Existing Tools
- Limiting Tool Accessibility

Adding New Tools to the Tools Gallery

Task: ***To add a new tool to the Tools Gallery:***

1. From the **Tools Tab**, right-click in the Tools Gallery and select **Add Tool**. The **Add Tool Wizard** opens.
2. On the **Welcome Panel**, click **Next**. The **Tool Properties Panel** opens.
3. On the **Tool Properties Panel**, provide the necessary details about the tool. Click **Next**. The **Command-Line Configuration Panel** opens.
4. If there are command-line options you want associated with the tool, do so from the **Command-Line Configurations Panel**. Each tool can have multiple command-line options associated with it for different purposes.
5. Click **Finish**.

After using the **Add Tool Wizard**, the new tool appears in the Tools Gallery and is available for use in workflows and projects.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Editing Properties for an Existing Tool

To edit the properties of a Tool in the Tools Gallery, perform the following steps.

Task: ***To edit an existing tool's properties:***

1. From the **Tools Tab**, right-click the tool in the Tools Gallery and select **Properties**. The **Tool Properties** dialog box is displayed.
2. Click the **Properties** tab.
3. Modify tool properties as necessary.
4. Click **OK** to apply the changes.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Adding Command-Line Configurations for an Existing Tool

To add command-line configurations for an existing Tool, perform the following steps:

Task: ***To add a command-line configuration to an existing tool:***

1. From the **Tools Tab**, right-click the tool to which you want to add a configuration from the Tools Gallery and select Properties. The **Tool Properties** dialog box is displayed.
2. Click the **Configuration** tab.
3. Click Add. The **Command-Line Properties** dialog box is displayed.
4. Enter a description and the command-line configuration.
5. Click **OK** to close the **Command-Line Properties** dialog box.
6. Click **OK** to close the **Tool Properties** dialog box.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Modifying Command-Line Configurations for an Existing Tool

To modify command-line configurations for an existing Tool, perform the following steps:

Task: ***To edit and existing tool's command-line configurations:***

1. From the **Tools Tab**, right-click the tool in the Tools Gallery and select Properties. The **Tool Properties** dialog box is displayed.
2. Click the **Configuration** tab.
3. Select the command-line configuration you want to edit and click **Modify**. The **Command-Line Properties** dialog box appears.
4. Modify the description and/or command line.
5. Click **OK** to dismiss the **Command-Line Properties** dialog box.
6. Click **OK** in the **Tool Properties** dialog box to apply the changes.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Deleting Command-Line Configurations from an Existing Tool

To delete command-line configurations from an existing Tool, perform the following steps.

Task: ***To delete a command-line configuration from an existing tool:***

1. From the **Tools Tab**, right-click the tool from which you want to remove the configuration in the Tools Gallery and select **Properties**. The **Tool Properties** dialog box is displayed.
2. Click the **Configuration** tab.
3. Select the configuration you want to remove and click **Delete**.
4. Click OK to close the **Tool Properties** dialog box.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Associating Tools with Tasks

To associate a Tool with a Workflow task, perform the following steps.

Task: ***To associate a tool with a task:***

1. Open the **Process Template Editor**.
2. In the **Workflows** tree, expand a Workflow to display all of its tasks.
3. From the **Workflows** tree, select the task with which you want to associate the tool. The **Task Properties** are displayed.
4. From the **Tool** list, select the tool you want to associate with the task. If the necessary tool is not listed, select **<New Tool...>** from the list to add the tool.
5. In the **Tool Configuration** list, select the configuration you want to use with the tool. If the configuration is not listed, click **Configure** to create the new configuration.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Running Associated Tools in Projects

To run associated Tools in Projects, perform the following steps.

Task: ***To run a tool associated with a task:***

1. Select the **Process Assistants** tab from the Interface.
2. Expand a Project in the **Projects** tree to display all of its tasks.
3. Right-click the task with which the tool is associated and select **Run Task** from the context menu.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Deleting Existing Tools

To delete a Tool from the Tools Gallery, perform the following steps.

Task: ***To delete an existing tool from the Tools Gallery:***

1. From the **Tools Tab**, right-click the tool you want to delete from the Tools Gallery and select **Delete**.
2. Confirm the deletion.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Limiting Tool Accessibility

Edition • *AdminStudio Enterprise Server is included in AdminStudio Enterprise Edition.*

Accessibility of tools is determined by a user's assigned Roles. If you want to modify an existing Role so that users assigned to that Role can no longer access a specific tool, see Editing an Existing User Role in the Managing Roles and Permissions section.

Workflows and Projects

Workflows, which can be created and modified using the **Process Template Editor**, are the basis for all projects in AdminStudio. These workflows consist of defined tasks, with which instructions (in the form of HTML files) and tools can be associated. Users can then create projects based on these workflows, and execute them—following the specific steps defined in the workflow. This allows you to create specific, repeatable procedures to accomplish your application migration goals.

Important • *In AdminStudio 10.0, the* ***Workflow Templates*** *tab of the AdminStudio interface has been moved into its own tool named* ***Process Template Editor****, which can be launched from the* ***Tools*** *tab or the* ***Tools*** *menu. All functionality remains the same. Also, the former* ***Projects*** *tab of the AdminStudio interface has been renamed to* ***Process Assistants****.*

Tip • *If you update a workflow, all projects based on that workflow will reflect the changes made to the workflow.*

The following topics relate to workflows and projects:

- Creating and Editing Workflows
- Creating and Using Projects
- Saving Workflow and Project Changes
- Workflow Project Example: Using the New Workflow Project Wizard
- Workflows, Projects, and Permissions

Creating and Editing Workflows

Workflows serve as templates upon which projects are based. Typically, only a few individuals create workflows, while others create projects and execute the projects to accomplish the workflow goal.

The following topics relate to creating and executing workflows:

- Creating New Workflows
- Renaming Workflows
- Filtering Workflows
- Deleting Workflows
- Creating New Tasks
- Modifying Task Properties
- Creating Notes for a Task
- Renaming Tasks
- Reordering Tasks
- Associating Help Files with Tasks
- Deleting Tasks
- Adding New Tools from the Process Template Editor

Creating New Workflows

Task: *To create a new workflow:*

1. Open the **Process Template Editor** by clicking its icon on the Tools tab. You are prompted to connect to an Application Catalog.
2. Enter the Application Catalog connection information and click OK. The Process Template Editor interface opens.
3. Right-click in the Workflows tree pane and select New Workflow. A new Workflow is listed.
4. Provide a name for the workflow.

Renaming Workflows

Task: *To rename an existing workflow:*

1. Open the **Process Template Editor**.
2. Right-click the workflow you want to rename and select Rename from the context menu.
3. Provide a new name for the workflow.

Filtering Workflows

Task: *To display a specific workflow:*

1. Open the **Process Template Editor**.
2. From the drop-down menu above the Workflows tree, select the workflow you want to display.

Deleting Workflows

Task: *To delete an existing workflow:*

1. Open the **Process Template Editor**.
2. From the Workflows tree, right-click the workflow you want to delete and select Delete from the context menu.
3. Confirm the deletion by clicking Yes in the resulting dialog box.

Creating New Tasks

Task: ***To create a new task:***

1. Open the **Process Template Editor**.
2. Right-click the workflow to which you want to add the task and select New Task. Alternatively, right-click on a task and select New Task to create a subtask.

   A new task appears named NewTask*nn*, and the Task Properties view for that task is displayed.
3. Enter a name for the new task.
4. Modify properties for the task.

Modifying Task Properties

Task: ***To modify properties for an existing task:***

1. Open the **Process Template Editor**.
2. From the Workflows tree, select the task you want to modify. The Task Properties view for the selected task is displayed.
3. Change Task Properties as necessary for the task.

Creating Notes for a Task

Task: ***To create notes for a task:***

1. Select the **Process Assistants** tab in the Interface.
2. From the Projects tree, select the task to which you want to add notes. The Project Task Properties view appears for the selected task.
3. Enter notes in the Notes field.

Tip • *You can also add notes to a task in the* ***Process Template Editor****. If you do, all projects based on that workflow will use the notes you enter as the default notes for the specific task.*

Note • *There is a 255 character limit on notes.*

Renaming Tasks

Task: ***To rename an existing task:***

1. Open the **Process Template Editor**.
2. Right-click the task you want to rename and select Rename from the context menu.
3. Provide a new name for the task.

Reordering Tasks

Task: ***To change the task order:***

1. Open the **Process Template Editor**.
2. In the Workflows tree, select the task you want to move.
3. From the toolbar, click Move Up or Move Down to change the order in which tasks are performed. Click Move Right to make a task a subtask of another task; click Move Left to promote a task.
4. Repeat the previous steps as necessary.

Associating Help Files with Tasks

Task: ***To associate a help file with a task:***

1. Open the **Process Template Editor**.
2. From the Workflows tree, select the task with which you want to associate the help file. The Task Properties view appears for the selected task.
3. In the Help File field, enter the name and location of the help file, or click Browse and navigate to it.
   - The help file can either be local, or you can use a URL (for example, http://www.mycompany.com/myURL.htm).
   - You can also click the Edit HTML button to the right of the Browse button to open a default HTML page in an HTML editor as a starting point.

Note • *Help files must be in HTML format.*

Deleting Tasks

Task: ***To delete an existing task:***

1. Open the **Process Template Editor**.
2. In the Workflows tree, right-click the task you want to remove and select Delete.
3. From the resulting dialog box, click Yes to confirm the deletion.

Adding New Tools from the Process Template Editor

Task: ***To add a new tool to the Tools list from the Process Template Editor:***

1. Open the **Process Template Editor**.
2. From the Workflows tree, select the task with which you want to associate the new tool. The Task Properties view appears for the selected task.
3. From the Tool list in the Task Properties view, select <Add Tool ...>. The Add New Tool dialog box opens.
4. In the **Add New Tool** dialog box, enter properties about the tool.
5. Click OK.

Creating and Using Projects

Projects, which are based on existing workflows, are the procedures followed to accomplish a set goal. Projects may include instructions describing what to do, and perhaps links to tools necessary to perform tasks. They also allow you to provide notes to help document issues that may arise during a project.

Tip • *If you update a workflow, all projects based on that workflow will reflect the changes made to the workflow.*

The following topics relate to creating and using projects:

- Creating Projects with the New Workflow Project Wizard
- Filtering Projects
- Executing Projects
- Running Associated Tools in Projects
- Deleting Projects

Creating Projects with the New Workflow Project Wizard

Task: ***To create a project using the New Workflow Project Wizard:***

1. Launch AdminStudio.
2. Click the **Process Assistants** tab.
3. Right-click in the Projects tree and select **New Project**. The **New Workflow Project Wizard** launches.
4. From the **Welcome Panel**, click **Next**. The **Workflow Selection Panel** appears.
5. From the **Workflow Selection Panel**, select the workflow on which you want to base the new project.
6. Provide a name for the new project and click **Next**. The **Source Package** panel appears.
7. From the **Source Package Panel**, specify the name and location of the source package used in this project. Alternatively, click Browse to navigate to it.
8. Click **Next**. The **Target Directory and Filename** panel appears.
9. From the **Target Directory and File Name Panel**, specify the Target Directory in which you want to store all files associated with this project.
10. In the **Target File Name** field, provide a name for the output file. Depending on the task being executed, the appropriate extension will be added to the file name.
11. Click **Finish**. The new Project is now listed.

Filtering Projects

Task: ***To display a specific project:***

1. From the Interface, click the **Process Assistants** tab.
2. From the drop-down menu above the Projects tree, select the project you want to display.

Executing Projects

Task: ***To execute a project:***

1. From the Interface, click the **Process Assistants** tab.
2. Display the project you want to execute. If you want to only display that project, use the filter above the Projects tree.
3. Click the first task in the project.

4. Perform the task.
5. When finished with the task, click the box to the left of the task.
6. Repeat for subsequent tasks in the project.

Running Associated Tools in Projects

Task: ***To run a tool associated with a task:***

1. Select the **Process Assistants** tab from the Interface.
2. Expand a Project in the Projects tree to display all of its tasks.
3. Right-click the task with which the tool is associated and select Run Task from the context menu.

Note • *In AdminStudio, tools are any external application or file that you can launch from a workflow or project. This typically is an application, but can be a simple document or batch file necessary to completing the project.*

Deleting Projects

Task: ***To delete an existing project:***

1. Click the **Process Assistants** tab in the Interface.
2. From the Projects tree, right-click the project you want to delete and select Delete Project from the context menu.
3. Confirm the deletion by clicking Yes in the resulting dialog box.

Saving Workflow and Project Changes

Because AdminStudio uses a database (the Application Catalog) to store information involving Workflows and Projects, all changes are stored immediately. There is no need to “save” your modifications; AdminStudio performs this automatically.

Workflow Project Example: Using the New Workflow Project Wizard

The following basic example covers creating a workflow and project which takes advantage of command-line functionality available in AdminStudio.

Prior to creating projects, you must create a workflow on which to base the project. This workflow might involve few steps, or it might cover as broad of a task as repackaging a legacy installation, editing it in InstallShield Editor, customizing it in Tuner, performing application isolation, identifying and resolving conflicts, distributing it, and entering information about it into a third-party tracking system.

In this example, you are going to create a basic workflow involving two steps: repackaging a legacy installation and opening the resulting InstallShield Editor project.

Creating a New Workflow

Task: ***To create a new workflow:***

1. Open the **Process Template Editor**.
2. Right-click in the Workflows tree pane and select New Workflow. A new Workflow is listed.
3. Name the workflow My Workflow Example.
4. Right-click My Workflow Example and select New Task.
5. Name the task Repackage a Legacy Setup.
6. With the Repackage a Legacy Setup task selected, from the Task Properties pane, select Repackager from the Tool menu. This associates Repackager with this task.
7. Change the Tool Configuration to Repackage a legacy setup. This associates a predefined command line (`-app "[SourcePackage]" -o "[TargetDir]" -pp "[TargetFileName]"`) with this task.
8. Right-click My Workflow Example and select New Task.
9. Name the task Open the Repackaged Setup.
10. With Open the Repackaged Setup selected, from the Task Properties pane, select Editor from the Tool menu. This associates InstallShield Editor with this task.
11. Change the Tool Configuration to Edit an InstallShield Editor project created with Repackager. This associates a predefined command line (`-"[TargetDir]\[TargetFileName].ism"`) with this task.

Creating a Project Based on the Workflow

Task: ***To create a project based on your new workflow:***

1. Click the **Process Assistants** tab.
2. Right-click in the Projects tree and select New Project. The **New Workflow Project Wizard** launches.
3. From the **Welcome Panel**, click **Next**. The **Workflow Selection Panel** appears.
4. From the **Workflow Selection Panel**, select My Workflow Example as the workflow on which you want to base the new project.
5. Name the new project My Sample Project and click **Next**. The Source Package panel appears.
6. From the **Source Package Panel**, click Browse and navigate to the following directory:

   AdminStudio Installation Directory\Tutorial\Repackager\PackageForTheWeb 3
7. Select PackageForTheWeb3.exe and click Open. This value (the directory and package name) are stored in the SourcePackage variable, which is used by the command line in Repackager set when you created the workflow.
8. Click **Next**. The **Target Directory and File Name Panel** opens.
9. From the Target Directory and File Name panel, specify the directory where you want to store files associated with your project. For this example, use C:\AdminStudio Shared\Test\WorkflowExample This value is written to the TargetDir variable used in the command line set for InstallShield Editor when creating the workflow.
10. Set the Target File Name to WorkflowProjectEx. This value is written to the TargetFileName variable used in the command line set for InstallShield Editor when creating the workflow.
11. Click Finish. The new Workflow Project is now listed.

Running the Workflow

Task: ***To run the workflow:***

1. Expand the workflow My Sample Project in the Projects tree.
2. Right-click the task Repackage a Legacy Setup and select Run Task.<

   When Repackager launches, it reads the value SourcePackage to determine the file to repackage. It also reads TargetDir and TargetFileName to determine where to place the output and what to call the output file.
3. After completing the steps above, right-click on the second step in the project and run the associated tool. InstallShield Editor launches, and opens the project file (.ism) created by Repackager.

Summary

This is just a brief example of how AdminStudio tools can be made aware of each other during a project. When crafting workflows, create command lines to streamline your projects.

Workflows, Projects, and Permissions

AdminStudio interface functionality (including workflows, projects, and the Tools Gallery) is directly influenced by user authorization and permissions. For example, Administrators can see all users and projects assigned to those users in AdminStudio. In the case of NT Groups, Administrators can see individual members of those groups in the **Process Assistants** tab. Further, Administrators can assign projects to users when running the **New Workflow Project Wizard**.

An example of how permissions affect workflows and projects is the availability of the **Process Template Editor**, which requires the **View Workflow Tab** permission. Likewise, only users with the **Create Project** permission can create projects. Even if you have permission to view and create workflows, you can only associate tools which you are permitted to use with tasks you create. If you are executing projects, you can only launch tools you have permissions to use, regardless of whether they are associated with a task in the workflow.

For more information, see Managing Roles and Permissions.

Frequently Asked Questions

The following is a list of questions frequently asked by AdminStudio users, including a link to the appropriate help topic.

General & Workflow

- How do I add a new tool to the Tools Gallery? See Adding New Tools to the Tools Gallery.
- How do I add a command line configuration to a tool? See Adding Command-Line Configurations for an Existing Tool.
- How do I specify a default Application Catalog? See Specifying a Default AdminStudio Application Catalog.

Application Isolation Wizard

- How do I isolate applications? See Isolating Applications Using Application Isolation Wizard.
- How do I modify the default isolation recommendations? See Modifying the Default Isolation Recommendations.

ConflictSolver

- What types of conflicts can ConflictSolver detect? See Identifying and Resolving Application Conflicts Using ConflictSolver.
- How do I change which conflicts are checked? See Changing Default Conflict Types Checked.
- How do I validate a package? See Validating Packages.
- How do I identify conflicts? See Checking for Conflicts Using the Conflict Wizard.
- How do I automatically resolve conflicts? See Automatically Resolving Conflicts.

Tuner

- What should I do If MSI prevalidation fails? See Handling Invalid Windows Installer Packages.
- How do I prevent a feature from displaying during custom installation? See Changing a Feature's Visibility.
- When should I use the Dialogs View instead of MSI command-line options? See Dialogs View vs. Command-Line Options.
- How do I create a `setup.exe` file for my package and transform? Creating a Setup.exe File for the Package and Transform.
- When do I use Tuner vs. InstallShield Editor? See Customizing Installations Using Tuner.

Repackager

- Why do people use a Repackager? See Repackaging Legacy Installations Using the Repackaging Wizard.
- Why is a "clean" system important for repackaging? See About Repackaging on Clean Systems.
- Should I repackage a Windows Installer (.msi) setup? See Repackaging Wizard Best Practices.

- How do I repackage a non-Windows Installer setup? Repackaging Legacy Installations Using the Repackaging Wizard and Converting Legacy Installations Using the Repackager Interface.
- How can I speed up repackaging? See Repackaging Wizard Best Practices.
- What can I do with a repackaged setup? See Repackaging Legacy Installations Using the Repackaging Wizard and Converting Legacy Installations Using the Repackager Interface.
- Where does Repackager store my repackaged files and the MSI packages it builds? See Set Target Project Information and Capture Settings Panel.
- How do I identify and fix WinINSTALL conversion problems? See Troubleshooting Guidelines for WinINSTALL Conversion.
- How do I identify and fix SMS conversion problems? See Troubleshooting Guidelines for SMS Conversion.
- What do I do if I receive a `ISDEV: fatal error -5023: Error building table file` error while using Repackager? See Resolving an "Error Building Table File" Error.

Distribution Wizard

- How do I create an administrative installation? See Creating Administrative Installations for Packages.

OS Snapshot Wizard

- Why do I need an OS Snapshot? See Taking OS Snapshots.

InstallShield Editor

Answers to common questions regarding InstallShield Editor can be found under **Frequently Asked Questions** in the InstallShield Editor Help Library.

AdminStudio Interface Reference

The AdminStudio interface reference is organized into the following areas:

- AdminStudio Start Page
- Tools Tab
- Process Template Editor
- Process Assistants Tab
- Enterprise Server Tab
- AdminStudio Menus and Toolbar
- Dialog Boxes
- Wizards

AdminStudio Start Page

The AdminStudio Start Page provides quick access to product information, to recently opened files, and to InstallShield resources.

AdminStudio Start Page Tabs

The AdminStudio Start Page, which is designed to help you quickly get started evaluating and using AdminStudio, provides process information on how to perform key tasks using AdminStudio tools. Information is organized into the following tabs:

- **Getting Started**—Describes the main tasks that you can use AdminStudio to accomplish, and provides links to additional information. See Getting Started Tab.
- **Test for Application Compatibility**—Provides a flowchart that outlines how to use Compatibility Solver to test for application readiness on Microsoft Windows 7 32-bit and 64-bit platforms, as well as compatibility with Internet Explorer 8.0. See Test for Application Compatibility Tab.
- **Migrate to Application Virtualization**—Provides a flowchart that outlines the steps required to migrate your application portfolio into virtual applications that are ready for deployment within the enterprise. See Migrate to Application Virtualization Tab.
- **Migrate to Windows Installer**—Provides a flowchart that outlines the steps required to migrate legacy setups (such as `.exe` files) to deployable Windows Installer packages (`.msi`). See Migrate to Windows Installer Tab.
- **Set Up Infrastructure**—Lists the infrastructure setup steps that you need to perform prior to using AdminStudio for the first time: connect to an Application Catalog, configure virtual machines, set e-mail notification settings, and configure Compatibility Solver. See Set Up Infrastructure Tab.
- **Help & Support**—Provides sources for user documentation, support, and product information.

AdminStudio Views

The AdminStudio interface is organized into the following tabs, which appear across the top of the Start Page:

Figure 3-2: AdminStudio Interface Tabs

Click on these tabs to access the following AdminStudio views:

- **Start Page**—Initial view of AdminStudio.
- **Tools**—The Tools Tab includes the Tools Gallery and information on the selected tool.
- **Process Assistants**—On the Process Assistants Tab, you can create, execute, and delete projects, and access existing projects, which are the procedures followed to accomplish a set goal.

- **Report Center**—On the Report Center Tab, you can view the Application Readiness Dashboard. This dashboard report provides a snapshot of the current status of packages in your Application Catalog including deployment type breakdown, virtualization readiness, and package quality and conflict testing summary information.
- **Enterprise Server**—On the Enterprise Server Tab, you can use Job Manager to automate tasks and Report Center to generate reports. You can also manage Users, Roles, and Permissions.

Tools Tab

The Tools tab, which is accessed from the AdminStudio **Start Page**, includes the Tools Gallery and the Content Pane.

Tools Gallery

You can launch each tool in the AdminStudio suite by double-clicking on the appropriate icon in the Tools Gallery. Individual AdminStudio applications can also be launched by clicking on the tool in the Tools Gallery on the **Start Page**:

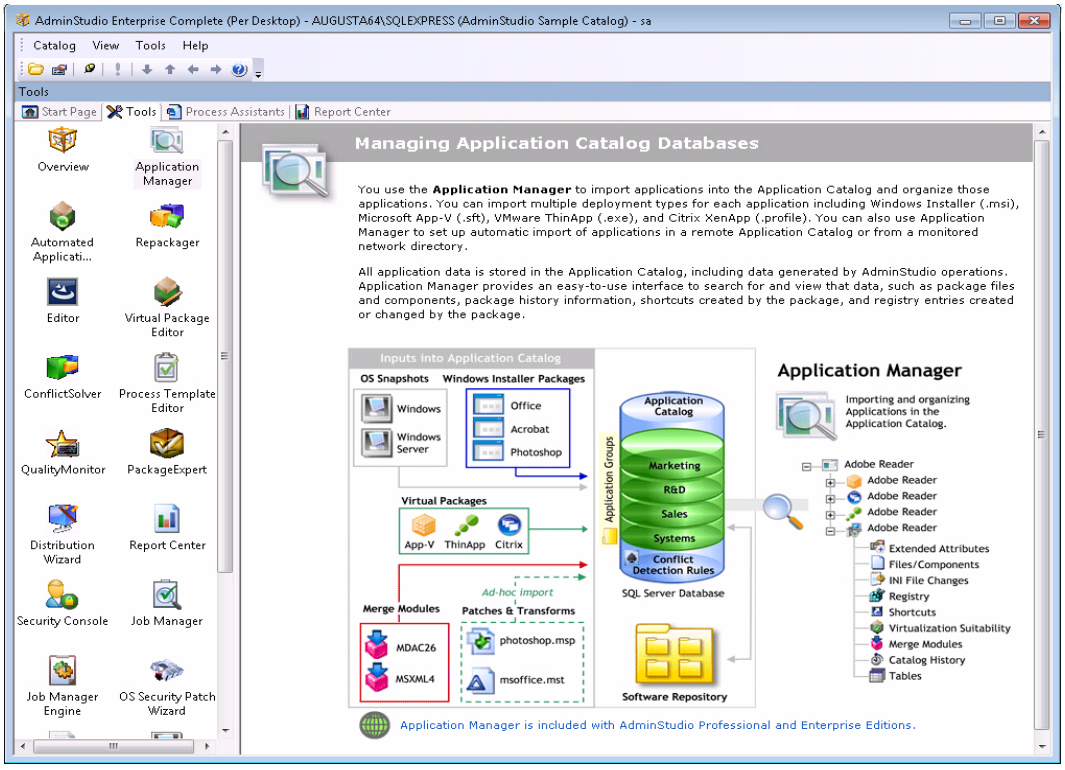


Figure 3-3: Tools Gallery on the Tools Tab

Note • *If a core AdminStudio tool is not in the Tools Gallery or available from the menu, it may mean an administrator has restricted tool access based on user permissions.*

By right-clicking in the Tools Gallery, you can launch the **Add Tool Wizard**, from which you can add new tools into the gallery and make them available for task assignments in workflows and projects.

Content Pane

The first icon in the Tools Gallery, Overview, is selected by default. When selected, the Content Pane lists a description of each of the AdminStudio tools. If you single-click on an AdminStudio tool icon in the Tools Gallery on the **Tools** tab, additional information about the selected tool is displayed in the Content Pane.

Process Assistants Tab

Purpose

From the Process Assistants tab, you can create, execute, and delete projects. You can also access existing projects. The drop-down filter above the Project pane allows you to view all available projects, or only a specific project.

Each project must be based on an existing workflow. In this way, projects are similar to photocopies from a master instruction sheet—all planning and design of the procedure is done to the workflow. The project is a copy of that workflow, and multiple projects can be based on the same workflow if you are performing the same procedure.

Integration with AdminStudio Workflow Manager

AdminStudio Workflow Manager is a Web-based application that manages the application lifecycle, incorporating standards (data) and methodologies (process). AdminStudio Workflows and Workflow Manager Workflows can be integrated, so that an AdminStudio Project can be a Workflow Phase in a Workflow Manager Workflow.

When an AdminStudio Project is linked to a Workflow Manager Workflow, please note the following indications on the Process Assistants tab:

- When an integrated AdminStudio project is selected on the Process Assistants tab, the name of its associated Workflow Manager Application is displayed in the Project Properties.
- When an AdminStudio Project is linked to a Workflow Manager Workflow and the workstation is not currently connected to the Workflow Manager Server, the following icon appears in the bottom right of the Process Assistants tab view:

Report Center Tab

On the **Report Center** tab, you can view the Application Readiness Dashboard. This dashboard report provides a snapshot of the current status of packages in your Application Catalog including deployment type breakdown, virtualization readiness, and package quality and conflict testing summary information. You can click on any chart to open more detailed reports.

The Application Readiness Dashboard includes the following information:

- **Deployment Type Breakdown**—Provides a graph listing the percentage of packages in the Application Catalog by deployment type (Windows Installer, App-V, XenApp, ThinApp, or Legacy).
- **Error Category Breakdown**—Shows the number of AdminStudio Best Practices and Conflict errors that have been identified in Windows Installer and App-V packages, by category.
- **Virtualization Readiness**—Shows a summary of the virtualization status of packages in the Application Catalog, identifying packages as being Ready for Virtualization, Not Suitable for Virtualization, Already Virtualized, or Not Tested.
- **Windows Installer/App-V Package Quality**—Shows the Microsoft Validation status of Windows Installer packages, and the AdminStudio Best Practices test status of Windows Installer and App-V packages.
- **Windows Installer/App-V Package Conflicts**—Shows the AdminStudio Conflict test status of Windows Installer and App-V packages.

AdminStudio uses Microsoft SQL Reporting Services to generate this dashboard (which is also visible in Application Manager). For more information, see Viewing Application Catalog Enhanced Reporting.

Enterprise Server Tab

Edition • *The **Enterprise Server** tab is available in AdminStudio Enterprise Edition.*

AdminStudio Enterprise Server is a security console and set of Web tools that are closely integrated with AdminStudio. The Enterprise Server tools include the Security Console, Job Manager, Report Center, Configuration Manager Web Console, and Workflow Manager. For more information, see the following topics:

- Generating and Viewing Reports in Report Center
- Automating Tasks Using Job Manager
- Distributing Packages Using Configuration Manager Web Console
- Managing Roles and Permissions
- Managing Users, Directory Services, and User Logins

Workflow Manager Tab

AdminStudio Workflow Manager is a Web-based application that manages the application lifecycle, incorporating standards (data) and methodologies (process). AdminStudio Workflows and Workflow Manager Workflows can be integrated, so that an AdminStudio Project can be a Workflow Phase in a Workflow Manager Workflow.

If you have purchased AdminStudio Enterprise Edition and Workflow Manager, the Workflow Manager icon will be displayed in the AdminStudio Tools Gallery on the AdminStudio Tools tab and will be available as a tab on the AdminStudio Enterprise Server interface.

***Note** • For more information on Workflow Manager, see **Using Workflow Manager to Manage Enterprise Software Packaging** in the AdminStudio Enterprise Server Help Library.*

Process Template Editor

Workflows, which can be created and modified using the Process Template Editor, are the basis for all projects in AdminStudio. These workflows consist of defined tasks, with which instructions (in the form of HTML files) and tools can be associated. Users can then create projects based on these workflows, and execute them—following the specific steps defined in the workflow. This allows you to create specific, repeatable procedures to accomplish your application migration goals.

Tasks

Once you create a workflow, you can add tasks to it. Tasks are discrete steps in your overall process. Each task has the following configurable options:

Table 3-3 • Task Options

Option	Description
Tool	If needed, you can pick a tool to associate with the task. When a user runs the workflow, the tool can be launched from the workflow step. By default, the AdminStudio tools are included in this list. If you have added tools to the Tools Gallery, they also appear in this list. If you want to add a tool directly from the Process Template Editor, select the <New Tool> option to display the Add New Tool dialog box. This adds the tool to the Tools Gallery and makes it available for the current task. ***Note** • If, after adding a new tool for a task that is not included in the Tools Gallery, you assign a different tool or no tool to the task, the tool you added will no longer be available. To avoid this, when possible, add tools to the Tools Gallery*

Table 3-3 • Task Options (cont.)

Option	Description
Tool Configuration	This list contains all available command-line configurations for the selected tool. If you do not need a configuration, select <None>. Click Configure to add new configurations to the tool, which you can then select from this list.
Help File	You can associate a help file (in HTML format) with the task to provide instructions for performing the task. Enter the path and help file in this field, or use the Browse button to navigate to it. If you have yet to create an HTML page, click the Edit HTML button to the right of the Browse button to open a default page in an HTML editor.
Notes	Add any notes you want associated with this task. This field can only hold 255 characters, so additional information should be added to your help file.

AdminStudio Menus and Toolbar

The following commands and toolbar buttons are available in the AdminStudio interface.

Table 3-4 • AdminStudio Menus and Toolbar

Menu	Command	Shortcut	Button	Description
Catalog	**Connect**	Ctrl+O		Displays the Connect Application Catalog Dialog Box, where you can open an existing SQL Server Application Catalog or the AdminStudio Enterprise Server Application Catalog.
Catalog	**Create**	Ctrl+N		Displays the Application Catalog Wizard, where you can create a new SQL Server Application Catalog database.
Catalog	**Disconnect**	Ctrl+D		Closes the currently open Application Catalog.
Catalog	**Properties**			Opens the Application Catalog Properties Dialog Box where you can configure the Software Repository.
Catalog	**Change AES Password**			Change the password of the current user to log in to the AdminStudio Enterprise Server.
Catalog	**Logout**			Log out of the AdminStudio Enterprise Server.

Table 3-4 • AdminStudio Menus and Toolbar (cont.)

Menu	Command	Shortcut	Button	Description
Catalog	**Replication**	Ctrl+R		Use to access the SQL Server **Application Catalog Replication** feature. These menu items are selectively enabled depending on whether the current user is connected to an SQL Server database and has been assigned `sysadmin` privileges in SQL Enterprise Manager on that database server. If the open database is a Subscriber database, the **Subscription Manager** option will be enabled, and if it is a Publisher database, the **Publication Manager** option will be enabled.
Catalog	**Exit**	Alt+C+X		Exits AdminStudio and returns you to the Windows desktop.
View	**Toolbar**	Alt+V+T		Toggles the Toolbar.
View	**Status Bar**	Alt+V+S		Toggles the Status Bar.
View	**Always On Top**	Alt+V+A		When checked, the AdminStudio Interface remains on top of all other windows.
View	**Start Page**			Select to open the AdminStudio Start Page.
View	**Process Assistant**			Select to open the Welcome Page of the Process Assistant.
View	**Tools**			Select to open the Tools tab, which includes the Tools Gallery and information on the selected tool.
View	**Report Center**			Select to open the Report Center tab, which displays the Application Readiness Dashboard.
Tools	**Check for Updates**	Alt+T+U		Determine if there any updates or messages available for AdminStudio.
Tools	**Options**	Alt+T+O		Displays the **Options** dialog box, from which you can configure the location of shared resources and the frequency AdminStudio checks for updates.
Help	**Contents**	Alt+H+C		Launches the online Help Library and displays the Contents tab.

Table 3-4 • AdminStudio Menus and Toolbar (cont.)

Menu	Command	Shortcut	Button	Description
Help	**Index**	Alt+H+I		Launches the online Help Library and displays the Index tab.
Help	**Search**	Alt+H+S		Launches the online Help Library and displays the Search tab.
Help	**Support Central**	Alt+H+U		Connects to the AdminStudio Support Web site.
Help	**Web Community**	Alt+H+M		Connects to the AdminStudio Web Community.
Help	**ReadMe**	Alt+H+R		Displays the AdminStudio Release Notes file.
Help	**Feedback**	Alt+H+F		Connects to an online form, through which you can provide feedback about AdminStudio.
Help	**Flexera Software on the Web**	Alt+H+W		Connects to the Flexera Software Web site.
Help	**About AdminStudio**	Alt+H+A		Displays the About dialog box with version information for AdminStudio.
Context Menu	**New Workflow**			Create a new Workflow.
Context Menu	**New Task**			Create a new Task.
Context Menu	**Rename**			Rename selected Workflow or Task.
Context Menu	**Delete**			Delete selected Workflow or Task.
Context Menu	**New Project**		Ctrl-P	Create a new Project.
Context Menu	**Del Project**			Delete selected Project.
Context Menu	**Run Task**			Runs the tool associated with the selected task in the project.

Table 3-4 • AdminStudio Menus and Toolbar (cont.)

Menu	Command	Shortcut	Button	Description
Context Menu	**Move Up**			Moves the selected task up in the task order.
Context Menu	**Move Down**			Moves the selected task down in the task order.
Context Menu	**Move Left**			Moves the selected task left in the task order.
Context Menu	**Move Right**			Moves the selected task right in the task order.

Dialog Boxes

The following dialog boxes can be opened from the AdminStudio Interface:

- About AdminStudio Dialog Box
- Add New Tool Dialog Box
- Command Line Properties Dialog Box
- Options Dialog Box
- Tool Properties Dialog Box

About AdminStudio Dialog Box

The **About AdminStudio** dialog box can be opened by selecting About AdminStudio from the **Help** menu. This dialog box displays information about the product, including the full version number (essential if you need technical support). If you have not registered AdminStudio, click the **Register** button to connect to the InstallShield Web site to begin the Product Registration process. Registering your product offers you expert technical support, new product announcements and special offers, plus notification of product upgrades.

Add New Tool Dialog Box

The Add New Tool dialog box is displayed if you select <New Tool> from the Tool list while designing a workflow. This dialog box allows you to provide information about a tool you want accessible from AdminStudio, particularly to use in workflow tasks.

This dialog box contains the following options:

Table 3-5 • Add New Tool Dialog Box Options

Option	Description
Target	Enter the location of the tool or Browse to the application or file you want added to the tools pane for use in AdminStudio.
Name in Tools Gallery	This description is used as the display name for the application in the Tools Gallery.
Command Line Arguments	Enter any command-line arguments for the tool. Because you may have different uses for applications, you can add the same application multiple times to the tools pane, with each instance using different command line arguments.
Working Directory	If this tool requires a working directory, enter it here or click Browse to locate it.
Comments	Enter any comments about this tool in this field.
HTML Explanation File	Enter the location and name of an HTML file you want displayed when you single-click on the tool in the tools pane. Alternatively, click Browse and navigate to it. If you have yet to create one, click the Edit HTML button ( ) below the field to open a default page in an HTML editor:
Add to Tools Gallery Check Box	When this box is checked, the tool will be added to the Tools Gallery. If unchecked, it is only available for the task where it was added.

Command Line Properties Dialog Box

The Command Line Properties dialog box is displayed when you create or edit a command-line configuration for a tool.

The Command Line Properties dialog box has two configurable options:

Table 3-6 • Command Line Properties Dialog Box Options

Option	Description
Description	Provide a description for the configuration. This assists you in differentiating similar command-line options.
Command Line	Provide the actual command-line parameters for the tool. The arrow to the left allows you to select one of the following variables to include in the command-line: • **InstallLocation**—The location where AdminStudio is installed. • **DevLocation**—The location where InstallShield Editor is installed. • **SharedPoint**—The AdminStudio shared directory. • **SourcePackage**—The name and location of the source package. • **TargetDir**—The directory where output from the selected project is stored. • **TargetFileName**—The name of the output file. • **ProjectName**—The name of the current project.

Options Dialog Box

From the **Options** dialog box, you can configure settings including application catalog settings, shared locations settings, and the frequency AdminStudio checks for updates. The dialog box consists of the following tabs:

- Application Catalog Tab
- Locations Tab
- Updates Tab
- Quality Tab: Customer Experience Improvement Program
- Virtual Packages Tab
- Notification Settings Tab

Application Catalog Tab

From this tab, you can configure the following options:

Table 3-7 • Options Dialog Box/Application Catalog Tab Options

Option	Description
Shared AdminStudio Application Catalog	This field displays the name and location of the shared AdminStudio Application Catalog. Click the **Change** button to select another Application Catalog using the Select Application Catalog Dialog Box.
AdminStudio Enterprise Server URL	Enter the URL for the AdminStudio Enterprise Server, available in the AdminStudio Enterprise Edition.
Make this the default shared AES URL	If you have more than one AdminStudio Enterprise Servers, select this option to make the URL in the **AdminStudio Enterprise Server URL** field the default.
Allow automatic authentication	Select this option to allow automatic authentication by AdminStudio Enterprise Server.
Install AES	If AdminStudio Enterprise Server is not installed and you want to install it on this machine, click this button to launch the installation.
Compatibility Solver Database	In order to integrate Compatibility Solver test results into the AdminStudio Application Catalog—making them available to display in AdminStudio report views—click **Change** and specify the location of the Compatibility Solver Database. ***Edition*** • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.* ***Edition*** • *AdminStudio enhanced reports are included with AdminStudio Enterprise Edition.* ***Note*** • *For more information, see Viewing the Application Compatibility Dashboard.*

Locations Tab

From this tab, you can configure the following options:

Table 3-8 • Options Dialog Box/Locations Tab Options

Option	Description
AdminStudio Shared Location	Enter or browse to the shared location for AdminStudio. This location will contain shared information for repackaging and conflict identification. To maintain consistency when creating workflows, it is recommended that you set this shared location the same for each AdminStudio seat. ***Tip*** • *The **AdminStudio Shared Location** is defined during installation and normally does not need to be changed. It is usually assigned to a network folder, preferably a UNC path.*
Task Help Page Location	Enter or browse to the directory where you want to store all HTML pages that serve as workflow task instructions.
Configuration Manager Web Console URL	Enter the URL for the Configuration Manager Web Console Web site.

Updates Tab

You can configure the following options from this tab:

Table 3-9 • Options Dialog Box/Updates Tab Options

Option	Description
Check for software updates	Specify how often you want AdminStudio to check for updates. Your options are: • Never • Once every 15 days • Once every 30 days (default) • Once every 60 days

Quality Tab: Customer Experience Improvement Program

The Customer Experience Improvement Program (CEIP) helps AdminStudio improve the quality, reliability and performance of our software and services.

If you choose to participate in the Customer Experience Improvement Program, we will collect anonymous information about how you use AdminStudio. This information helps us identify trends and usage patterns.

All information collected is anonymous, and this data collection will not affect the performance of AdminStudio tools. You will never be prompted to complete a survey, and no one from our company will contact you. You can continue working with AdminStudio without interruption.

Your membership status in the Customer Experience Improvement Program is specified on the **Quality** tab of the AdminStudio **Options** dialog box. If you initially select to participate but later you change your mind, you can opt-out of this program by changing the selection on the **Quality** tab.

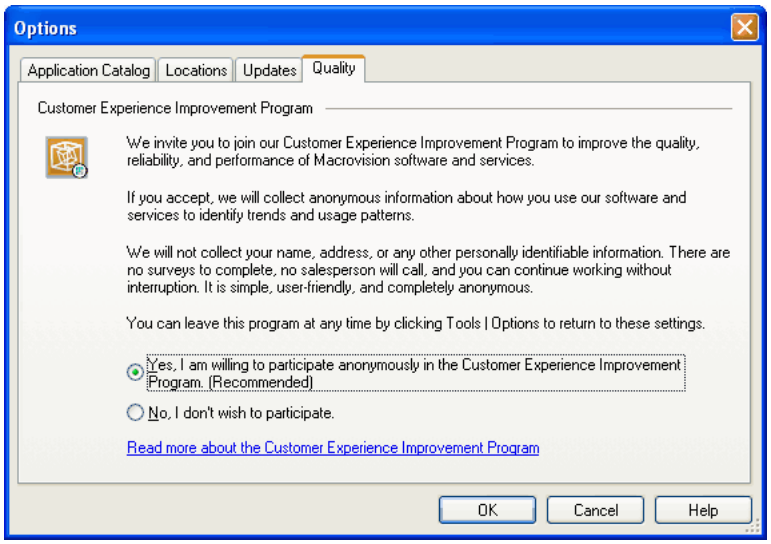


Figure 3-4: Quality Tab of the AdminStudio Options Dialog Box

Participation in the Customer Experience Improvement Program is not mandatory, but your input is appreciated.

For more information on the Customer Experience Improvement Program, visit the AdminStudio Web Site.

Virtual Packages Tab

On the **Virtual Packages** tab, you can control which of the available virtual package formats are displayed in the AdminStudio toolset—Microsoft App-V, VMware ThinApp, and/or Citrix XenApp—and you can choose whether to include the App-V Application Launcher with App-V applications.

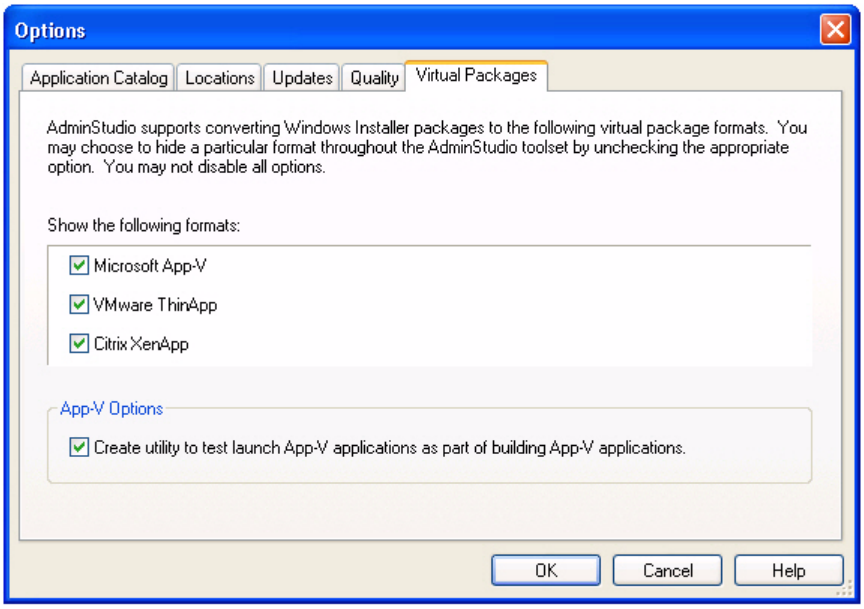


Figure 3-5: Virtual Packages Tab of the Options Dialog Box

The **Virtual Packages** tab of the **Options** dialog box has the following options:

Table 3-10 • Virtual Packages Tab of the Options Dialog Box

Option	Description
Microsoft App-V	Select to display all Microsoft App-V virtual package format options throughout AdminStudio and InstallShield Editor. If this check box is not selected, the options for the virtual package format will be hidden.
VMware ThinApp	Select to display all VMware ThinApp virtual package format options throughout AdminStudio and InstallShield Editor. If this check box is not selected, the options for the virtual package format will be hidden.
Citrix XenApp	Select to display all Citrix XenApp virtual package format options throughout AdminStudio and InstallShield Editor. If this check box is not selected, the options for the virtual package format will be hidden.
Create utility to test launch App-V applications as part of building App-V applications.	Select this option to include the App-V Application Launcher with each App-V application that you build. You can use the App-V Application Launcher to locally test a newly built App-V application before moving it to a deployment server. For more information, see Launching Packages for Testing. ***Note*** • *This option is only visible when the Microsoft App-V format is selected.*

Important • *You may not disable all options.*

Notification Settings Tab

On the **Notification Settings** tab, you can configure your SMTP notification settings. This will enable AdminStudio to send you e-mail notifications during various processes.

Currently, e-mail notifications are sent when soft time-outs are encountered while using Automated Application Converter to repackage an application on a virtual machine.

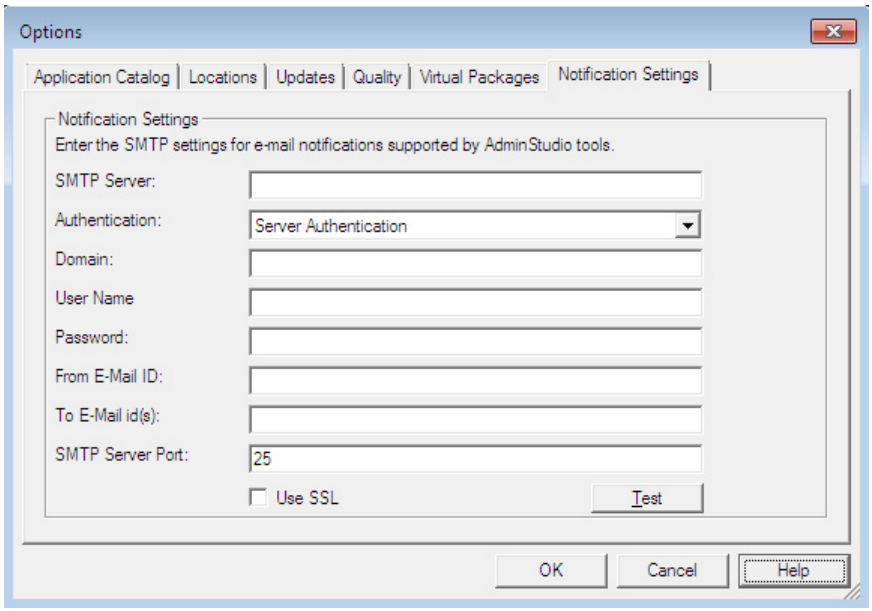


Figure 3-6: Notification Settings Tab of the Options Dialog Box

The **Notification Settings** tab of the **Options** dialog box has the following options:

Table 3-11 • Notification Settings Tab of the Options Dialog Box

Option	Description
SMTP Server	Enter the address of your e-mail server, such as: `smtp.yourcompany.com`
Authentication	Specify how your e-mail is authenticated by selecting one of the following options: • **Server Authentication**—Select this option if you want to perform server authentication on your AdminStudio e-mail. • **Anonymous**—Select this option if you do not want to perform authentication on your AdminStudio e-mail.

Table 3-11 • Notification Settings Tab of the Options Dialog Box

Option	Description
Domain	Enter the Domain of the user account listed in the **User Name** field.
User Name	Enter the name of an existing user account in the Domain specified in the **Domain** field. This user must have permission to send e-mail.
Password	Enter the password of the user account defined in the **User Name** and **Domain** fields. ***Note*** • *If your network domain requires that user passwords are changed periodically, you will have to open this dialog box again to update this account's password. To avoid this, try to obtain a user account that has a password that does not expire.*
From E-Mail ID	Enter the e-mail address to serve as the identity of AdminStudio. All e-mails sent by AdminStudio will have this e-mail address in the **From** field.
To E-Mail ID(s)	Enter the e-mail address to serve as the system account for AdminStudio e-mail. All e-mails sent to AdminStudio will be sent to this address. ***Note*** • *Use a semi-colon to separate multiple e-mail addresses in the **To E-Mail ID(s)** field.*
SMTP Server Port	Enter the port of your SMTP server.
Use SSL	Select this option if you want to use SSL security for the AdminStudio e-mail account.

Tool Properties Dialog Box

The Tool Properties dialog box is displayed when you right-click on a tool in the Tools gallery and select Properties. This dialog box contains the following tabs:

- Properties Tab
- Configuration Tab

Properties Tab

The **Properties** tab of the T**ool Properties** dialog box contains information about the tool, including the name and location of the executable, the name of the tool as it appears in the Tools gallery, and the help file associated with it (if any).

The following options can be configured:

Table 3-12 • Tool Properties Dialog Box/Properties Tab Options

Option	Description
Target	Enter the location of this tool's executable. Alternatively, click Browse and navigate to it.
Name in Tools Gallery	Provide a name for the tool as it will appear in the Tools gallery.
Command Line Arguments	Enter any default command line arguments for this tool.
Working Directory	If this tool requires a working directory, enter it here or click Browse to locate it.
Comments	Provide any comments about this tool.
HTML Explanation File	Enter the location and name of an HTML file you want displayed when you single-click on the tool in the tools pane. Alternatively, click Browse and navigate to it. If you have yet to create one, click the Edit HTML button below the field (shown below) to open a default page in an HTML editor:

Configuration Tab

From the **Configuration** tab of the Tool Properties dialog box, you can **Add**, **Modify**, or **Delete** command-line configurations associated with the tool. Each tool can have multiple configurations associated with it, for different uses.

Table 3-13 • Tool Properties Dialog Box/Configuration Tab Options

Option	Description
Command Line List	Listing of all command lines defined for this tool.
Add	Click to open the Command Line Properties Dialog Box, where you can add a new command line. In the Description field, you provide a description for the configuration. This assists you in differentiating similar command-line options. In the Command Line field, you provide the actual command-line parameters for the tool. The arrow to the left allows you to select one of the following variables to include in the command-line: • **InstallLocation**—The location where AdminStudio is installed. • **DevLocation**—The location where InstallShield Editor is installed. • **SharedPoint**—The AdminStudio shared directory. • **SourcePackage**—The name and location of the source package. • **TargetDir**—The directory where output from the selected project is stored. • **TargetFileName**—The name of the output file. • **ProjectName**—The name of the current project.
Modify	Click to open the Command Line Properties Dialog Box, where you can modify the selected command line.
Delete	Click to delete the selected command line.

Note • *Most AdminStudio Tools have one or more command-line configurations already defined. Although you can change or remove these configurations, there is no way to automatically reset them to their default values.*

Ways to Assign a Command Line to a Tool

There are three ways a user can assign a command line to a tool:

- **Tool Properties Panel of Add Tool Wizard**—When you add a tool, you can assign a command line on the **Tool Properties Panel** of the **Add Tool Wizard**.
- **Properties Tab of the Tool Properties Dialog Box**—When you view the tool's properties, you can assign a command line on the **Properties Tab** of the **Tool Properties** dialog box.

- **Configuration Tab of the Tool Properties Dialog Box**—Using the **Configuration Tab** of the **Tool Properties** dialog box, you can create *multiple* command lines and can use AdminStudio variables in these command lines. Then, when you go to the Workflow tab and create a new Workflow, you can associate a Tool with a task and also select which command line configuration they want to use. Once you have done that, you can go to Project tab and create a new Project. When you create a new Project, you will have to specify the Source Package and the target directory and file name. Once the Project is created, when you execute the task, AdminStudio will execute the command line configuration you previously selected by replacing the AdminStudio variables in the command line.

Note *• A command Line entered by the user in* ***Properties Tab*** *of the* ***Tool Properties*** *dialog box play no role in the Workflow and Project tab. This command line is used only when you run the tool from the Tools Tab.*

Wizards

The AdminStudio interface includes the following Wizards:

- Add Tool Wizard
- New Workflow Project Wizard

Add Tool Wizard

The Add Tool Wizard allows you to add new tools that appear in the AdminStudio Tools gallery. You can specify the tool's executable, provide command-line options for the tool, and provide a link to information about the tool.

The Add Tool Wizard includes the following panels:

- Welcome Panel
- Tool Properties Panel
- Command-Line Configurations Panel

Welcome Panel

The Add Tool Wizard allows you to add new tools that appear in the AdminStudio Tools gallery. You can specify the tool's executable, provide command-line options for the tool, and provide a link to information about the tool.

Click **Next** to proceed to the **Tool Properties Panel**.

Tool Properties Panel

From the Tool Properties panel of the Add Tool Wizard, you can enter information about the tool you are adding to the gallery.

You can configure the following options:

Table 3-14 • Add Tool Wizard/Tool Properties Panel Options

Option	Description
Target	Enter the location and name of this tool's executable. Alternately, click Browse and navigate to it.
Name in Tools Gallery	Provide a name for the tool as it will appear in the Tools Gallery.
Command Line Arguments	Enter any default command line arguments for this tool.
Working Directory	If this tool requires a working directory, enter it here or click Browse to locate it.
Comments	Provide any comments about this tool.
HTML Explanation File	Enter the location and name of an HTML file you want displayed when you single-click on the tool in the tools pane. Alternatively, click Browse and navigate to it. If you have yet to create one, click the Edit HTML button below the field (shown below) to open a default page in an HTML editor:

Command-Line Configurations Panel

From this panel, you can add command-line configurations for the tool. Each tool can have multiple command-line configurations for different tool uses.

Click **Add** to add a new option from the **Command-Line Properties** dialog box. You can also click Modify to edit the selected command-line option, or Delete to remove the selected option.

Click **Finish** to add the tool to the Tools Gallery; click Back to return to the **Tool Properties Panel**.

New Workflow Project Wizard

The New Workflow Project Wizard assists you in creating a new project based on an existing workflow. The values you specify in this Wizard are stored in variables that can be accessed from tools, allowing for greater interoperability in AdminStudio. The New Workflow Project Wizard includes the following panels:

- Welcome Panel
- Workflow Selection Panel
- Source Package Panel
- Target Directory and File Name Panel

Welcome Panel

The New Workflow Project Wizard assists you in creating a new project based on an existing workflow. The values you specify in this Wizard are stored in variables that can be accessed from tools, allowing for greater interoperability in AdminStudio.

Workflow Selection Panel

From the Workflow Selection panel, you can specify the workflow on which you want to base this project. Available workflows appear in the workflows window.

Select the workflow you want to use, and provide the name for the new project (which is stored in the ProjectName predefined variable).

Source Package Panel

From the Source Package panel, you can specify the name and location of the source package used in this workflow. If you are creating a repackaging project, this is usually an executable, such as Setup.exe. The source package name and location is stored in the predefined variable SourcePackage.

***Note** • You must specify the name and location of a source package if it is required in the workflow (by using the SourcePackage variable). For example, if the workflow specifies to launch a package with certain command-line parameters, AdminStudio needs to know what package to launch.*

Source packages can also be non-setup files. For example, if you are creating a simple workflow that involves editing a Notepad file, the source package may be a .txt file.

Target Directory and File Name Panel

From the **Target Directory and File Name** panel, specify the **Target Directory** and **Target File Name** used in this project.

All output files (such as an INC file from Repackager or an MST file from Tuner) associated with the project will be stored in the Target Directory, and the value for this directory can include a predefined variable such as `SharedPoint`. The Target File Name is the name used for all files created by project tasks in this project, with the appropriate extension appended to it depending on the file type. The Target Directory is stored in the predefined variable TargetDir and the output file name is stored in the variable `TargetFileName`.

***Note** • If any workflow tasks use the TargetDir or TargetFileName variables, you must specify the target directory and package name. For example, if the workflow specifies to save a task's output, AdminStudio needs to know where to save it and what to call it.*

Click **Finish** to close the Wizard.

Log Files

AdminStudio tools generate the following log files:

Table 3-15 • AdminStudio Log Files

Log File	Location
AdminStudio.log	`C:\Program Files\AdminStudio\10.0 SP2`
predeploytest.log	`C:\Program Files\AdminStudio\10.0 SP2\PreDeployTest\Client`
licwrapserver.log	`C:\Program Files\AdminStudio\10.0 SP2\FLEXWrap\Server`
debug.log	`C:\Program Files\AdminStudio\10.0 SP2\FLEXWrap\Server`
distributer.log	`C:\Program Files\AdminStudio\10.0 SP2\Common`
islc.log	`C:\Program Files\AdminStudio\10.0 SP2\Repackager`

Part 2

Managing Users, Roles, Permissions, Application Catalogs

This part of the AdminStudio 10.0 SP2 User Guide includes the following chapters:

- Managing Users, Directory Services, and User Logins
- Managing Roles and Permissions
- Managing Application Catalog Databases

4

Managing Users, Directory Services, and User Logins

You can create a user account for each person that you want to have access to AdminStudio Enterprise Server, or you can import users or groups of users from a Windows Active Directory or Novell eDirectory Directory Service.

You can also configure various login methods to best suit your needs: AdminStudio account login, domain account login, single sign-on login, and guest account login.

Information is presented in the following main sections:

- Managing Users and Groups
- Managing Directory Services Configurations
- Managing User Logins

Managing Users and Groups

There are several methods for creating new users in AdminStudio Enterprise Server:

- **Create a user account**—Create a user account for each person that you want to have access to AdminStudio Enterprise Server. See Creating a New User Account.
- **Import Directory Service users**—Set up a Windows Active Directory or Novell eDirectory Directory Service Connection and import users from that Directory Service. See Importing Directory Services Users and Groups.
- **Import Directory Service group**—Set up a Windows Active Directory or Novell eDirectory Directory Service Connection and import a group from that Directory Service. This allows you to provide dynamic access to all of the members of that group as the members of that group changes. For more information, see Importing Directory Services Users and Groups.

***Note** • For more information on the methods for logging into AdminStudio Enterprise Server, and the how authentication is performed, see Managing User Logins.*

Creating a New User Account

To create a new user account, perform the following steps.

Task: ***To create a new user account:***

1. Open AdminStudio Enterprise Server and open the **Administration** tab.
2. Open the **Users & Groups** tab. The **User Administration** page opens.
3. Click **Add**. The **User Details** view opens.
4. Select the new user's **Company** from the list.
5. Enter a **User Name** and a **Password**, with the **User Name** in the format of:

   username`@`*companyname*`.com`
6. Optionally, enter the user's **Email address**.
7. Optionally, enter a geographic **Location** that you can use to group users together, such as `New York Office` or `Midwest Region`, etc. Assigning a geographical location to a user enables you to display this information in reports.
8. All of the **Roles** that have been defined for the selected Company are listed. Select the Roles that you want to assign to this user.

   Note • *The Roles that a user is assigned to determine what portions of AdminStudio Enterprise Server and Workflow Manager functionality they will have access to. Therefore, you must assign at least one Role to a user.*
9. Click **Update** to save your entries and close the **User Details** view, or click **Update and Add New** to remain in the **User Details** view so you can add another user.

Importing Directory Services Users and Groups

If you have defined a Directory Service, as described in Adding a Directory Service Connection, you can choose to add Users or Groups from that Directory Service into AdminStudio Enterprise Server.

When a user is added from a Directory Service into AdminStudio Enterprise Server, only information that uniquely identifies the user in the Directory Service is stored in AdminStudio Enterprise Server. Information such as telephone number and email address will always be queried at run time, so that the most current information is obtained.

AdminStudio Enterprise Server supports Windows Active Directory and Novell eDirectory directory services.

Note • *For more information on AdminStudio Enterprise Server support for Directory Services, see Managing Directory Services Connections and Managing User Logins.*

Task: ***To import users or groups from a Directory Service into AdminStudio Enterprise Server:***

1. Open AdminStudio Enterprise Server and open the **Administration** tab.
2. Open the **Users & Groups** tab. The **User Administration** page opens.
3. Click **Directory Service User/Group Import**. The **Directory Services User/Group Add** view opens.
4. From the **Select a Directory Service** list, select the Directory Service that you want to import users or groups from.
5. Next to **Select Group or User**, select the **User** or **Group** option.
6. Next to **Filter Directory Service** list, select one of the following options:
   - **Show All**—Select this option to include all Users or Groups in the list.
   - **Filter list by**—Select this option, enter text in the box, and click **Find** to filter the list by the specified criteria.

   For example, to search for all of the users that start with the letter **P**, use the asterisk wildcard character (*) and enter **P*** in the **Filter list by** box.
7. From the **Select Group or User to Add** list, select the group or user to import to AdminStudio Enterprise Server. The **User Details** view opens, providing a read-only view of the imported user or group's User Name and Password.

   If you import a User that is a member of a Group that was previously imported, that User inherits the Roles that are assigned to that Group. You can then assign additional Roles to that User.
8. Click **Update** to save your entries and close the **User Details** view, or click **Update and Add New** to remain in the **User Details** view so you can add another user.

Disabling a User or Group Account

User and Group accounts have a status of **Active** or **Inactive**. If a User or Group has the status of **Inactive**, that User or a member of that Group:

- Cannot log into AdminStudio Enterprise Server.
- Cannot be assigned any work.
- Is not listed on the **User Administration** page.
- Does not get any email notifications.
- Cannot be selected as a **Customer Contact** or **Servicer Contact**.
- Cannot be viewed by Customers.

Deleting vs. Disabling a User Account

If a user is associated in any way with an open or completed Workflow Manager Application Request, you cannot delete that user from the system. In that case, you would instead disable that account by setting its **Status** to **Inactive**.

- **If a user IS NOT associated with an Application Request**, you can delete that user. This means that if you create a user by mistake, before that user has interacted with the application, you can delete it by opening that user on the **User Details** view and clicking **Delete**.
- **If a user IS associated with an Application Request**, However, since AdminStudio Enterprise Server stores the IDs of users on actions and assignments for historical, tracking and reporting purposes, you cannot delete a user that is associated with a Workflow Manager Application Request. In this case, you would set the user's **Status** to **Inactive** on the **User Details** view.

Task: ***To disable a user account:***

1. Open AdminStudio Enterprise Server and open the **Administration** tab.
2. Open the **Users & Groups** tab. The **User Administration** page opens.
3. Select the User or Group that you want to disable to open its **User Details** view.
4. Select **Inactive** from the **Status** list and click **Update**.

Deleting a User Account

You can delete a User or Group account from the **User Details** view.

Task: ***To delete a user account:***

1. Open AdminStudio Enterprise Server and open the **Administration** tab.
2. Open the **Users & Groups** tab. The **User Administration** page opens.
3. Select the User or Group that you want to delete to open its **User Details** view.
4. Click **Delete**. You are prompted to confirm the deletion.
5. Click **OK**. The deleted User or Group is no longer listed on the **User Administration** page.

Managing Directory Services Configurations

AdminStudio Enterprise Server can be integrated with Windows Active Directory and Novell eDirectory. This enables you to set up automatic login with AdminStudio Enterprise Server based upon Directory Services authentication.

If you import a Group, all members of that group could then login to AdminStudio Enterprise Server without requiring you to import the individual members of the Group. AdminStudio Enterprise Server can then retrieve user attributes, such as email address or telephone number, from the Directory Service dynamically.

- **Directory Service Connections**—Rather than creating a user account for each person who will use AdminStudio Enterprise Server, you can import users from Windows Active Directory or Novell eDirectory directory services. For information on integrating AdminStudio Enterprise Server with Directory Service users and groups, see Managing Directory Services Connections.
- **Directory Service Attributes—**When creating a Workflow Manager Template, you can set a Data Element's **Data Type** to **Directory Service**. This means that when a Directory Service authenticated user performs the Workflow Step that requests that Data Element, information is pulled from the Directory Service to automatically populate the field, such as that user's name, email address, or location. For information on enabling the use of Directory Service Attributes, see Managing Directory Services Attributes.

Note *• For more information about Directory Services and Lightweight Directory Access Protocol (LDAP), see Ldp Overview on the Microsoft TechNet Web site.*

Managing Directory Services Connections

Rather than creating a user account for each person who will use AdminStudio Enterprise Server, you can import users from Windows Active Directory or Novell eDirectory directory services. To integrate AdminStudio Enterprise Server with a Directory Service users and group, you set up a **Directory Service Connection**.

This section includes the following topics:

- Adding a Directory Service Connection
- Editing an Existing Directory Service Connection
- Deleting a Directory Service Connection

Adding a Directory Service Connection

Directory Services Connections are used to import users and groups into AdminStudio Enterprise Server, and to authenticate Active Directory or eDirectory users logging into AdminStudio Enterprise Server.

You can choose to have a Directory Service listed in the **Domain** list on the AdminStudio Enterprise Server login page. This enables users in this Directory Service to login to AdminStudio Enterprise Server using their enterprise network credentials.

To add a Directory Service connection, perform the following steps:

Task: ***To add a Directory Service connection:***

1. On the **Administration** tab, click **Directory Services**. The **Directory Services** page opens.
2. Click **Manage Directory Services Connections**. The **Directory Services List** page opens.
3. Click **Add**. The **Add Directory Service Connection** view opens.
4. Enter the following information:

Option	Description
Use to Authenticate Users?	Select **Yes** to include this Directory Service in the **Domain** list on the AdminStudio Enterprise Server login page. This enables users to login to AdminStudio Enterprise Server using their enterprise network credentials. You can create many Directory Service connections, but only those connections that have this field checked will be listed in the **Domain** list on the AdminStudio Enterprise Server login page. ***Note*** • *In an enterprise, there is usually only one Directory Service that is responsible for authenticating users. However, you can create additional Directory Service connections to import users and groups.*
Directory Service Name	Enter a name to identify this Directory Service in the Domain list on the AdminStudio Enterprise Server login page. It is recommended that the Directory Service Name match the Domain Name.
Description	Enter a description of this Directory Service.
Directory Service Type	Identify the type of Directory Service that you are adding a connection to: • Active Directory • Novell eDirectory
Directory Service Host	Enter the name or IP address of the Server on which this Directory Service is running.
Directory Service Port	Enter the port number of the Server on which this Directory Service is running. By default, AdminStudio Enterprise Server uses Port 389. ***Important*** • *The value for the Directory Service Port should be the port on the given server to which AdminStudio Enterprise Server should connect to send LDAP queries.*

Option	Description
Base Distinguished Name	Enter the Base Distinguished Name (DN) to identify the root node of this Directory Service. For example, for **MyCompany**, the Base DN could be: `dc="MyCompany", dc="com"`
Domain Name	Enter the Domain Name of this Directory Service. This Domain Name is used to create the right credentials, along with the Admin User Name. For example, you would enter `DomainName\AdminUserName` to connect to the specified Directory Service. ***Note*** • *To upgrade from AMS 2.5, enter the same Domain Name that was used in AMS 2.5. AdminStudio Enterprise Server will read this Domain Name and then try to authenticate the User using this Directory Service. For single sign-on to work on an upgrade scenario from 2.5 to 4.0, administrators have to configure at least one Directory Service.*
Use Secure Socket Layer (SSL)?	Select **Yes** if this Directory Service is configured to use Secure Socket Layer (SSL).
Connect Anonymously?	Select **No** if you do not want to permit anonymous connections. If you select **Yes** to permit anonymous connections, AdminStudio Enterprise Server may not be able to authenticate Directory Service users and may not be able to add Directory Service users/groups into AdminStudio Enterprise Server.

Admin Distinguished Name	Enter the Distinguished Name of a user who has permission to retrieve users/groups information and authenticate any user against this Directory Service. Enter the Distinguished Name in the format of:
	
	`domainname\username`
	
	such as:
	
	`abcinc\jsmith`

Password	Enter the password for the Admin Distinguished Name.
Group Class Name	Enter the object class names for groups for this Directory Service. Default values are: • **Active Directory**: `group` • **eDirectory**: `groupofnames`
Group Name Attribute	Enter an attribute used by this Directory Service to name groups. The default value for both Active Directory and eDirectory is `cn`.

Option	Description
Group Member Attribute	Enter an attribute used by this Directory Service to define member groups. Default values are: • **Active Directory**: `member` • **eDirectory**: `uniquemember`
User Class Name	Enter the object class name used by this Directory Service for users. Default values are: • **Active Directory**: `user` • **eDirectory**: `inetorgperson`
User Name Attribute	Enter the attribute used by this Directory Service to identify users. Default values are: • **Active Directory**: `samaccountname` • **eDirectory**: `uid`

5. Click **Test Connection** test to see if the settings that you entered can be used to successfully connect to this Directory Service.
6. Click **Update** to save your entries and return to the **Directory Services List** page or click Update and Import (User/Group) to save your entries and open the **Directory Services User/Group Add** view.

Editing an Existing Directory Service Connection

To edit an existing Directory Service connection, perform the following steps:

Task: ***To edit an existing Directory Service connection:***

1. On the **Administration** tab, click **Directory Services**. The **Directory Services** page opens.
2. Click **Manage Directory Services Connections**. The **Directory Services List** page opens.
3. Select the Directory Service connection that you want to edit. The **Edit Directory Service Connection** page opens.
4. Edit these entries as described in Adding a Directory Service Connection.

Deleting a Directory Service Connection

You can import a user/group into an AdminStudio Enterprise Server database using a Directory Service Connection. This will create a reference to this Directory Service Connection in the database. You can then import a Directory Service Attribute into the database.

In order to delete this Directory Service, user has to delete all the references to this Directory Service Connection. If there are references left (like users from this directory service or attributes from this directory service) you cannot delete this Directory Service Connection.

Managing Directory Services Attributes

When creating a Workflow Manager Template, you can assign a **Data Element** the **Data Type** of **Directory Service**. This means that when a Directory Service authenticated user performs the Workflow Step that requests that Data Element, information is pulled from the Directory Service to automatically populate the field, such as that user's name, email address, location, etc.

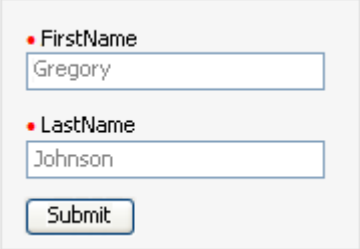


Figure 4-1: Example of Fields Populated With Directory Services Attributes

Note • *If the user is not connected using Directory Service authentication, then the fields will be left blank and enabled.*

On the Directory Services Attributes Administration Page, which is accessed by clicking **Manage Directory Services Attributes** on the **Directory Services** page, Administrators can map their customizations for the directory service.

Creating Directory Services Attributes

To add a new Directory Services attribute from the directory server into the AdminStudio Enterprise Server database, perform the following steps.

Task: ***To create a new Directory Services attribute:***

1. On the **Directory Services** page, click **Manage Directory Services Attributes**. The **Directory Services Attributes Administration** page opens.
2. Click **Add**. The **Add Directory Service Attributes** view opens.
3. Using the current user's credentials, the **Attribute Name** list lists all of the Directory Services attributes from the Directory Services server. Select the Directory Services attribute that you want to add to the AdminStudio Enterprise Server database. A Directory Services attribute is used in Directory Services protocol to access information directories, such as `employeeNumber` or `documentAuthor`.
4. In the **Attribute Alias** text box, enter the name that you want to use to identify the Directory Services attribute in AdminStudio Enterprise Server, such as Employee ID or Author.
5. Click Add to add the defined Directory Services attribute. If AdminStudio Enterprise Server is unable to connect to the server (and no attributes are retrieved), the Add button is disabled.

Deleting Directory Services Attributes

To delete a Directory Services Attribute, open the **Directory Services Attributes Administration** page and click **Delete** next to the attribute that you want to delete.

Managing User Logins

This section explains the four methods of logging into AdminStudio Enterprise Server, and how to set the session timeout value.

- Login Methods
- Using AdminStudio Enterprise Server Account Login
- Using Domain Account Login
- Using Single Sign-On Login
- Using Guest Account Login
- Setting the Session Timeout Value

Login Methods

There are four methods you can use to login to AdminStudio Enterprise Server:

Table 4-1 • AdminStudio Enterprise Server Login Methods

Method	Description	Related Topics
AdminStudio Enterprise Server Account Login	Users login to AdminStudio Enterprise Server using a User Account (User Name and Password) that was created in AdminStudio Enterprise Server by clicking **Add** on the **User Administration** page.	• Using AdminStudio Enterprise Server Account Login • Creating a New User Account
Domain Account Login	Users login to AdminStudio Enterprise Server using their Domain Account (User Name and Password), the same account they use to login to their network. To set this up, you import users or groups from a Directory Service: Active Directory or Novell eDirectory. ***Note*** • *For domain account login, the **Anonymous access** option on the IIS Authentication Methods Dialog Box must be selected.*	• Using Domain Account Login • Adding a Directory Service Connection • Importing Directory Services Users and Groups
Single Sign-On Login	You set up AdminStudio Enterprise Server so that if users are logged on to your network using their Domain Account (User Name and Password), they are automatically logged in to AdminStudio Enterprise Server. To set this up, you import users or groups from Active Directory ***Note*** • *For Single Sign-On, IIS performs user authentication. For all other login methods, AdminStudio Enterprise Server performs authentication.* ***Note*** • *Single Sign-on is not supported for Novell eDirectory users and groups.* ***Note*** • *For single sign-on login, the **Anonymous access** option on the IIS Authentication Methods Dialog Box must not be selected.*	• Using Single Sign-On Login • Adding a Directory Service Connection • Importing Directory Services Users and Groups

Table 4-1 • AdminStudio Enterprise Server Login Methods (cont.)

Method	Description	Related Topics
Guest Account Login	You set up a Guest account to permit users without AdminStudio Enterprise Server login credentials to access features – such as viewing Reports or searching for an Application Request – without entering a User ID or Password.	• Using Guest Account Login • Setting Up a Guest Account • Logging in as a Guest

IIS Authentication Methods Dialog Box

To set the Anonymous access option in IIS, perform the following steps.

Task: ***To open the IIS Authentication Methods dialog box:***

1. Open Internet Information Services (IIS).
2. Select the AdminStudio Enterprise Server Web site in the tree view.
3. On the **Action** menu, select **Properties**. The **Web Site Properties** dialog box opens.
4. Open the **Directory Security** tab and click **Edit** in the **Anonymous access and authentication control** area. The **Authentication Methods** dialog box opens.

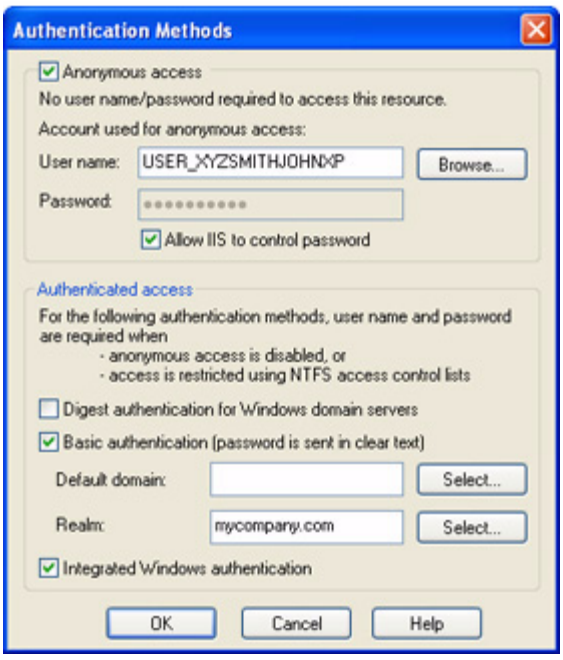


Figure 4-2: Anonymous Access option on the IIS Authentication Methods Dialog Box

Using AdminStudio Enterprise Server Account Login

To Set Up

To set up AdminStudio Enterprise Server User Account login, create a user account (**User Name** and **Password**) for each person that you want to have access to AdminStudio Enterprise Server. See Creating a New User Account.

To Login

Users with AdminStudio Enterprise Server accounts enter their AdminStudio Enterprise Server-generated **User Name** and **Password** on the AdminStudio Enterprise Server Login page. AdminStudio Enterprise Server performs the authentication.

More Information

For more information, see Creating a New User Account

Using Domain Account Login

To Set Up

To set up Domain Account login, you do the following:

- Set up a **Windows Active Directory** or **Novell eDirectory** Directory Service Connection. See Adding a Directory Service Connection.
- On your AdminStudio Enterprise Server IIS Server, open the IIS Authentication Methods Dialog Box and select the **Anonymous access** option. This specifies that AdminStudio Enterprise Server is going to authenticate the users.
- Import users or groups from that Directory Service. See Importing Directory Services Users and Groups.

To Login

Imported users or users belonging to the imported groups enter their Directory Service **User Name** and **Password** on the AdminStudio Enterprise Server Login page.

AdminStudio Enterprise Server then connects with the Directory Service using the credentials provided on the Add Directory Service Connection View, and then passes the login User Name and Password to the Directory Service so that it can authenticate the user.

Note • *When entering the User Name, it is not necessary to enter the Directory Service domain name.*

More Information

For more information, see:

- Adding a Directory Service Connection
- Importing Directory Services Users and Groups
- IIS Authentication Methods Dialog Box

Using Single Sign-On Login

To Set Up

To set up Single Sign-on login, you do the following:

- Set up a **Windows Active Directory** Directory Service Connection. See Adding a Directory Service Connection.
- On your AdminStudio Enterprise Server IIS Web Server, open the IIS Authentication Methods Dialog Box and clear the **Anonymous access** option. This specifies that IIS is going to authenticate the users.
- Import users or groups from that **Active Directory** Directory Service. See Importing Directory Services Users and Groups.

Note • *Regarding Single Sign-On, note the following:*

- *For Single Sign-On, IIS performs user authentication . For all other login methods, AdminStudio Enterprise Server performs authentication.*
- *Single Sign-on is not supported for Novell eDirectory users and groups.*

To Login

When a user with single sign-on opens AdminStudio Enterprise Server, the IIS Web Server checks to see if the user's domain **User Name** and **Password** are valid in the Active Directory Domain Server, and then automatically logs the user in to AdminStudio Enterprise Server. AdminStudio Enterprise Server does not need to connect to the Active Directory Domain Server.

For More Information

For more information, see:

- Adding a Directory Service Connection
- Importing Directory Services Users and Groups
- IIS Authentication Methods Dialog Box

Using Guest Account Login

This section describes how to setup a Guest Account and how to login using a Guest Account:

- Setting Up a Guest Account
- Logging in as a Guest

Setting Up a Guest Account

When a Guest Account is set up, AdminStudio Enterprise Server users without a User Account are automatically logged in using a Guest Account.

The AdminStudio Enterprise Server administrator can set up a Guest account to permit users without login credentials to access features such as viewing a report or searching for an Application Request . By using a Guest Account, Administrators do not have to add a new user account just to provide access for users who only need very limited functionality.

AdminStudio Enterprise Server system administrators should configure a Guest Account as follows:

Task: ***To set up a Guest Account:***

1. Create a new user account to use as the Guest Account, and add this user to the `web.config` file located in the AdminStudio Enterprise Server Web Application `wwwroot` directory. Enter the user name in the following location of the `web.config` file:

   ```
   <!-- Guest System Access -->
   <add key="GuestAccount" value="username@companyname.com" />
   ```

2. On the **Administration** tab, click **Users & Groups**. The **User Administration** page opens.
3. Assign the Guest Account user to Roles with limited permissions. See Managing Roles and Permissions for more information.

Guest System Access Recommendations

When setting up Guest System Access, please follow these guidelines:

- The AdminStudio Enterprise Server Administrator should assign the Guest account user to Roles with limited privileges. If the Guest account belongs to a Role with high privileges, then every user who logs on as a Guest will have high privileges.
- If the `GuestAccount` key does not exist in the `web.config` file, then the **Guest Access** option does not appear on the AdminStudio Enterprise Server Login page.
- If the `GuestAccount` key exists in the `web.config` file but the user name specified in the value is empty (such as `value=""`), then the **Guest Access** option does not appear on the AdminStudio Enterprise Server Login page.
- If the `GuestAccount` key exists in the `web.config` file, but the user name specified in the value does not exist in the AdminStudio Enterprise Server database, then AdminStudio Enterprise Server displays an error message when a user tries to log on as a Guest User.

Configuring a Guest Account on a Workflow Manager Portal Site

Please note the following regarding configuring a Guest account on a Workflow Manager Portal site:

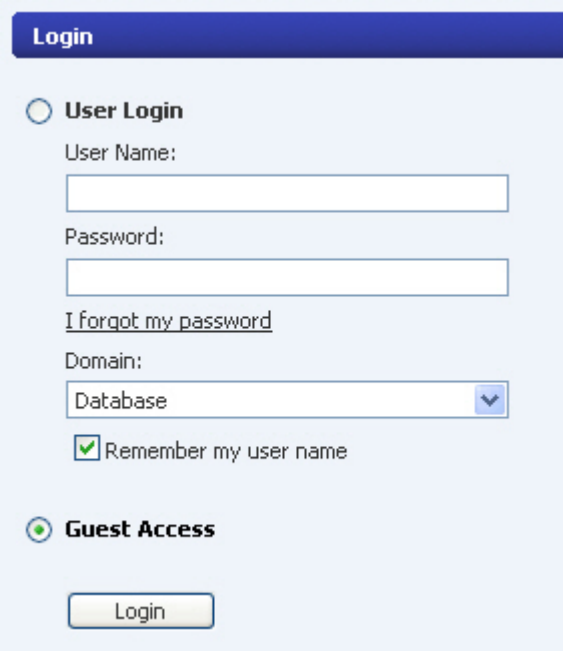
- If users are accessing Workflow Manager through a Workflow Manager portal, then the AdminStudio Enterprise Server administrator should add the `GuestAccount` key to the `web.config` file of the Portal site. By default, Workflow Manager will use the Guest account value configured at the portal site to log on the Guest user. If the portal site does not have any `GuestAccount` value configured, then Workflow Manager uses the `GuestAccount` value at the Workflow Manager Web Application site to log on the Guest account.
- If users are accessing the Workflow Manager site directly (not going through a portal), then Workflow Manager uses the `GuestAccount` value configured in the `web.config` file located in the AdminStudio Enterprise Server Web Application `wwwroot` directory to log on the Guest account.

***Note** • The `web.config` file installed by AdminStudio Enterprise Server should contain the GuestAccount key with an empty value. It is the responsibility of the System Administrator to fill in that value in the `web.config` file of the Portal site or the AdminStudio Enterprise Server Web Application site.*

Logging in as a Guest

Before anyone can log on to AdminStudio Enterprise Server as a Guest User, the System Administrator must configure a Guest Account, following the instructions in Setting Up a Guest Account. Then a user could log on to your AdminStudio Enterprise Server site as a guest using the appropriate method:

- **If Directory Service authentication is used at your site**, and the user's Directory Service user name does not exist in the AdminStudio Enterprise Server database, AdminStudio Enterprise Server automatically logs this user on using the Guest account.
- **If Directory Service authentication is not used at your site**, AdminStudio Enterprise Server opens the AdminStudio Enterprise Server Login page by default. The user then selects the **Guest Access** option and clicks **Login**.



The AdminStudio Enterprise Server Administrator assigns the Guest account to specific Roles, and the degree of permissions that the Guest account has depends upon which Roles the account is assigned to.

- **When a Role is created**, the administrator must indicate whether that Role has permission to create a Workflow Manager Application Request.
- **When Reports are created**, the administrator must indicate which Roles have permission to view them. Guest Users can view all Reports that their assigned Roles have permission to access.

Setting the Session Timeout Value

When AdminStudio Enterprise Server is installed, the session timeout value is set to 20 minutes. This means that after you login to the AdminStudio Enterprise Server, if your session remains idle for 20 minutes, you are prompted to login again.

You can change the session timeout value by editing two values in the `web.config` file: `sessionState timeout` and `LoginPageTimeOut`.

Task: ***To change the session timeout value:***

1. Open the `wwwroot` directory of the AdminStudio Enterprise Server installation and open `web.config` in a text editor.
2. Locate the line that contains `sessionState timeout`:

   ```
   <sessionState timeout="20" mode="InProc" stateConnectionString= "tcpip=127.0.0.1:42424"
   sqlConnectionString="data source=127.0.0.1; Trusted_Connection=yes" cookieless="false"  />
   ```

3. Change the value specified for `sessionState timeout="`*nn*`"` to the new timeout value.
4. Locate the following line that contains `LoginPageTimeOut`:

   ```
   <add key="LoginPageTimeOut" value="19.5" />
   ```

5. Change this `value` to match the timeout value you specified for the `sessionState`.

Users/Groups and Directory Services Reference

This section details the contents of the AdminStudio Enterprise Server pages that are used to manage users and directory services:

Table 4-2 • Users/Groups and Directory Services Tabs Reference

Page	Subpages
User Administration Page	• User Details View • Directory Services User/Group Add View
Directory Services Page	• Directory Services List Page • Add Directory Service Connection View • Directory Services Attributes Administration Page • Add Directory Service Attributes View

User Administration Page

This view lists all users defined in the system. From this view, you can add a new User, edit an existing User, or import a Directory Service User or Group.

Table 4-3 • User Administration Page Options

Option	Description
Add	Click to access the User Details View, where you can add a new User.
Directory Service User/Group Import	Click to access the Directory Services User/Group Add View, where you can import a User or Group from an Directory Service.
Status	To filter the list of users by Status, make a selection from this list: **Active**, **Inactive** or **All**.
User Name	Name user uses to log onto AdminStudio Enterprise Server, in the format of: *username@companyname.com* If a Directory Services Group has been imported, the Group name is listed in this column.
Roles	Lists the roles that this User has been assigned to.
Company	Name of user's Company.

From the **User Administration** page, you can perform the following tasks:

- **Create a AdminStudio Enterprise Server user account**—Create a user account for each person that you want to have access to AdminStudio Enterprise Server. See Creating a New User Account.
- **Import Directory Service Users or Groups**—Set up a Windows Active Directory or Novell eDirectory Directory Service Connection and the click **Directory Service User/Group Import** on the **User Administration** page to import Users or Groups from that Directory Service. See Importing Directory Services Users and Groups. Importing a group allows you to provide dynamic access to all of the members of that Group as the members of that Group changes.
- **Edit a user account**—To edit an existing account, click **Select** next to the **User Name** to open the **User Details** view.

Note • *For more information on the methods for logging into AdminStudio Enterprise Server and how authentication is performed, see Managing User Logins.*

User Details View

On this view, enter or edit user account and imported Directory Services account information. On this page, you can also delete a user account.

Table 4-4 • User Details View Options

Option	Description
Company	Select the Company that this user or group belongs to.
User Name	Perform one of the following: • **New**—When creating a new user, enter the name the user will use to log onto AdminStudio Enterprise Server in the format of: *username@companyname.com*. • **Importing**—When importing a user group from a Directory Service, the name listed here will be prepopulated and uneditable.
Password	Enter the User's password. ***Note*** • *This field is not shown when importing a Directory Service Group.*
Status	Select **Active** to make this user or group active in the AdminStudio Enterprise Server system, or select **Inactive** to disable this account. See Disabling a User or Group Account.
Email	User's email address. ***Note*** • *This field only appears when creating a new user account. For users imported from a Directory Service, this information is retrieved dynamically from the Directory Service when needed.*
Roles	Listing of all of the defined Roles at the selected Company. Select the Roles appropriate for this user or group.
Location	Optionally, enter a geographic Location that you can use to group users together, such as `New York Office` or `Midwest Region`, etc. Assigning a geographical location to a user enables you to display this information in reports.
Update	Click to save your entries and close the **User Details** view.
Update and Add New	Click to save your entries and remain in the **User Details** view, where you can enter another new user.
Delete	Click to delete this user account.
Cancel	Click to exit this view without saving your entries.

Directory Services User/Group Add View

This view is accessed by clicking **Directory Service User/Group Import** on the **User Administration** page. The following options are included:

Table 4-5 • Directory Services User/Group Add View

Option	Description
Select a Directory Service	Select the defined Directory Service that you want to import a user or group from. ***Note*** • *For information on defining a Directory Service Connection, see Adding a Directory Service Connection.*
Select Group or User	Specify whether you are importing a **User** or a **Group**.
Filter Directory Service List	Select **Show All** to include all Users or Groups in the list, or select **Filter list by**, enter text in the box, and click **Find** to filter the list by the specified criteria. For example, to search for all of the users that start with the letter **P**, use the asterisk wildcard character (*) and enter **P*** in the **Filter list by** box.
Select Group or User to Add	From the **Select Group or User to Add** list, select the group or user to import. ***Note*** • *If you import a User that is a member of a Group that was previously imported, that User inherits the Roles that are assigned to that Group. You can then assign additional Roles to that User.*
Update	Click to save your entries.

The selected user or group is then opened in the User Details View.

Directory Services Page

AdminStudio Enterprise Server can be integrated with Windows Active Directory and Novell eDirectory. This integration enables you to integrate Directory Services Users and Groups and also to use Directory Services attributes.

Table 4-6 • Directory Services Page Options

Option	Description
Manage Directory Service Connections	Click to open the Directory Services List Page where you can set up connections to Active Directory and eDirectory Directory Services.
Manage Directory Service Attributes	Click to open the Directory Services Attributes Administration Page where you can map Directory Service attributes, which can be used when defining Data Elements. When a Data Element is defined as a Directory Service attribute, when this Data Element appears during a data entry step in a Workflow, information about the logged in user will be pulled from the Directory Service to populate those fields, such as: Department, Location, Employee Number, etc.

Note • *For more information about Directory Services and Lightweight Directory Access Protocol (LDAP), see Ldp Overview on the Microsoft TechNet Web site.*

Directory Services List Page

Directory Services Connections are used to import users and groups into AdminStudio Enterprise Server, and to authenticate Active Directory or Novell eDirectory users logging into AdminStudio Enterprise Server.

You can choose to have a Directory Service listed in the **Domain** list on the AdminStudio Enterprise Server login page. This enables users in this Directory Service to login to AdminStudio Enterprise Server using their enterprise network credentials.

The information and options are included:

Table 4-7 • Directory Services List Page Options

Option	Description
Directory Service Name	The name that identifies this Directory Service in the **Domain** list on the AdminStudio Enterprise Server login page.
Description	A description of this Directory Service.

Table 4-7 • Directory Services List Page Options (cont.)

Option	Description
Directory Service Type	Identifies this Directory Service as either a Windows **Active Directory** or **Novell eDirectory** type.
Directory Service Host	The name or IP address of the Server on which this Directory Service is running.
Add	Click to add a new Directory Service connection.

Add Directory Service Connection View

On the **Add Directory Service Connection** view, which is opened by clicking **Add** on the Directory Services List Page, you enter the necessary settings to connect to an Active Directory or Novell eDirectory Directory Service.

Table 4-8 • Add Directory Service Connection View Options

Option	Description
Use to Authenticate Users?	Select **Yes** to include this Directory Service in the **Domain** list on the AdminStudio Enterprise Server login page. This enables users to login to AdminStudio Enterprise Server using their enterprise network credentials. ***Note*** • *For Single Sign-On, IIS performs user authentication. For all other login methods, AdminStudio Enterprise Server performs authentication.*
Directory Service Name	Enter a name to identify this Directory Service in the Domain list on the AdminStudio Enterprise Server login page. It is recommended that the Directory Service Name match the Domain Name.
Description	Enter a description of this Directory Service.
Directory Service Type	Identify the type of Directory Service that you are adding a connection to: • Active Directory • Novell eDirectory
Directory Service Host	Enter the name or IP address of the Server on which this Directory Service is running.
Directory Service Port	Enter the port number of the Server on which this Directory Service is running. By default, AdminStudio Enterprise Server uses Port 389. ***Important*** • *The value for the Directory Service Port should be the port on the given server to which AdminStudio Enterprise Server should connect to send LDAP queries.*

Table 4-8 • Add Directory Service Connection View Options (cont.)

Option	Description
Base Distinguished Name	Enter the Base Distinguished Name (DN) to identify the root node of this Directory Service. For example, for **MyCompany**, the Base DN could be: `dc="MyCompany", dc="com"`
Domain Name	Enter the Domain Name of this Directory Service. This Domain Name is used to create the right credentials, along with the Admin User Name. For example, you would enter `DomainName\AdminUserName` to connect to the specified Directory Service. ***Note*** • *To upgrade from AMS 2.5, enter the same Domain Name that was used in AMS 2.5. AdminStudio Enterprise Server will read this Domain Name and then try to authenticate the User using this Directory Service. For single sign-on to work on an upgrade scenario from 2.5 to 4.0, administrators have to configure at least one Directory Service.*
Use Secure Socket Layer (SSL)?	Select **Yes** if this Directory Service is configured to use Secure Socket Layer (SSL).
Connect Anonymously?	Select **No** if you do not want to permit anonymous connections. If you select **Yes** to permit anonymous connections, AdminStudio Enterprise Server may not be able to authenticate Directory Service users and may not be able to add Directory Service users/groups into AdminStudio Enterprise Server.
Admin Distinguished Name	Enter the Distinguished Name of a user who has permission to retrieve users/groups information and authenticate any user against this Directory Service. Enter the Distinguished Name in the format of: `domainname\username` such as: abcinc\jsmith
Password	Enter the password for the **Admin Distinguished Name**.
Group Class Name	Enter the object class names for groups for this Directory Service. Default values are: • **Active Directory**: `group` • **eDirectory**: `groupofnames`
Group Name Attribute	Enter an attribute used by this Directory Service to name groups. The default value for both Active Directory and eDirectory is `cn`.

Table 4-8 • Add Directory Service Connection View Options (cont.)

Option	Description
Group Member Attribute	Enter an attribute used by this Directory Service to define member groups. Default values are: • **Active Directory**: `member` • **eDirectory**: `uniquemember`
User Class Name	Enter the object class name used by this Directory Service for users. Default values are: • **Active Directory**: `user` • **eDirectory**: `inetorgperson`
User Name Attribute	Enter the attribute used by this Directory Service to identify users. Default values are: • **Active Directory**: `samaccountname` • **eDirectory**: `uid`
Update	Click to save your entries and return to the Directory Services List Page.
Update and Import (User/Group)	Click to save your entries and open the Directory Services User/Group Add View.
Test Connection	Click to test to see if the settings that you entered can be used to successfully connect to this Directory Service.

Directory Services Attributes Administration Page

On the **Directory Services Attributes Administration** page, which is accessed by clicking **Manage Directory Services Attributes** on the **Directory Services** page, Administrators can map their customizations for the directory services.

Directory Service attributes can be used when defining Data Elements. When a Data Element is defined as a Directory Service attribute, when this Data Element appears during a data entry step in a Workflow, information about the logged in user will be pulled from the Directory Service to populate those fields, such as: Department, Location, Employee Number, etc.

The following information is listed:

Table 4-9 • Directory Services Attributes Administration Page Options

Option	Description
Directory Service Attribute	Name of the LDAP (Lightweight Directory Access Protocol) Directory Service attribute that is used in Directory Services protocol to access information directories, such as employeeNumber or documentAuthor.

Table 4-9 • Directory Services Attributes Administration Page Options (cont.)

Option	Description
Attribute Alias	Name that you want to use to identify the Directory Service attribute in AdminStudio Enterprise Server, such as Employee ID or Author.
Delete	Click to delete the listed attribute.

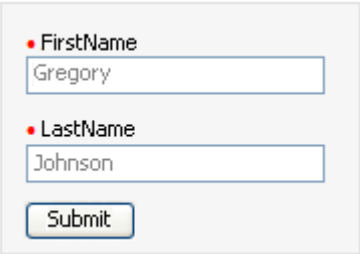
Add Directory Service Attributes View

When you click **Add** on the **Directory Services Attributes Administration** page to add a new Directory Service attribute into the AdminStudio Enterprise Server database, the following fields are displayed:

Table 4-10 • Add Directory Service Attributes View Options

Option	Description
Directory Service	Select the Directory Service that contains the attribute that you want to add.
Attribute Name	Lists all the attributes from the selected Directory Service server, using the currently logged user's credentials.
Attribute Alias	Enter the name that you want to use to identify the Directory Service attribute in AdminStudio Enterprise Server.
Add	Click to add the defined Directory Service attribute. If AdminStudio Enterprise Server is unable to connect to the server (and no attributes are retrieved), the **Add** button is disabled.

When a Data Element is defined as a Directory Service attribute during Workflow Manager Template creation, when this Data Element appears in the Workflow, information will be pulled from the Directory Service to populate those fields, such as:



However, if the user is not connected using Directory Service authentication, then the fields will be left blank and will be enabled.

5

Managing Roles and Permissions

All of the AdminStudio and Workflow Manager permissions are related to Roles. The amount of functionality that a user has is directly related to what Roles they are assigned to.

This section contains information on the following tasks:

- AdminStudio and Workflow Manager Roles and Permissions
- Creating a New Role
- Editing an Existing User Role
- Copying an Existing Role
- Roles Reference

AdminStudio and Workflow Manager Roles and Permissions

Permissions are assigned to Roles, and Roles are assigned to users. Therefore, all permissions in AdminStudio and Workflow Manager are controlled by Roles.

This section includes the following topics:

- Role Permission Lists
  - AdminStudio Client Tools Permissions
  - AdminStudio Enterprise Server Tools Permissions
  - Workflow Manager Permissions
- System Roles

Role Permission Lists

Permissions to perform all AdminStudio and Workflow Manager functions are assigned using Roles. A user has permission to perform only those tasks that are explicitly selected in the Role(s) that the user is assigned to.

This section describes all of the AdminStudio and Workflow Manager permissions:

- AdminStudio Client Tools Permissions
- AdminStudio Enterprise Server Tools Permissions
- Workflow Manager Permissions

AdminStudio Client Tools Permissions

The AdminStudio client tools have the following permissions:

Table 5-1 • AdminStudio Client Tools Permissions

Category	Permission
AdminStudio Process Assistants Tab	• **Edit**—Permits user to view and edit those Projects on the **Process Assistants** tab that are assigned to him. • **Create**—Permits user to create new Projects and assign them to users. Users with the **Create** permission see a list of all users and their associated Projects on the **Process Assistants** tab. Users with only the **Edit** permission cannot create new Projects and can only view and edit Projects that are assigned to him. • **Delete**—Permits user to delete a Project from the **Process Assistants** tab.
AdminStudio Process Template Editor	• **View**—Permits user to view the **Process Template Editor** and all of the existing Workflows. • **Edit**—Permits user to edit existing Workflows on the **Process Template Editor** • **Create**—Permits user to create a new Workflow on the **Process Template Editor** • **Delete**—Permits user to delete a Workflow from the **Process Template Editor**.
AdminStudio Tools Tab	• **Add**—Permits user to add a new tool to the **Tools** tab. • **Edit**—Permits user to edit the properties of an existing tool on the **Tools** tab. • **Delete**—Permits user to delete an existing tool on the **Tools** tab.

Table 5-1 • AdminStudio Client Tools Permissions (cont.)

Category	Permission
AdminStudio General	• **Modify Tools Options Dialog**—Permits user to set options on the **Locations**, **Updates** and **Quality** tabs of the AdminStudio **Options** dialog box. Users without this permission can view the **Options** dialog box but cannot make any changes. If a user has only the **Modify Tools Options Dialog** permission but does not also have the **Change Default Application Catalog** permission, that user cannot edit the **Shared AdminStudio Application Catalog** field on the **Application Catalog** tab of the **Options** dialog box. • **Change Default Application Catalog**—Permits user to edit the **Shared AdminStudio Application Catalog** field on the **Application Catalog** tab of the **Options** dialog box (if that user also has the **Modify Tools Options** dialog permission). This permission also permits user to edit the **Make this the default shared Application Catalog** option on the **Connect Application Catalog** dialog box
Application Manager / **Conflict Solver** Conflicts	• **Run Analysis**—Permits user to perform conflict analysis on a package. • **Resolve**—Permits user to resolve any automatically resolvable conflicts found during conflict analysis on a package. • **Modify Rules**—Permits user to open the **Rules Viewer** and create and edit new Rules. • **Modify Data**—Permits user to create new Groups, rename existing Groups, and modify Group Properties. Permits user to Cut and Paste a Group to a new location. Permits user to Copy/Cut and Paste a package into a new Group. Permits user to modify options on the **Resolution Options** dialog box. Permits user to edit a package **Description** on the **Products View.**
Application Manager / **Conflict Solver** Package	• **Delete**—Permits user to delete a package from the Application Catalog. • **Import**—Permits user to import a package into the Application Catalog. • **Modify Extended Attributes**—Permits user to modify a package's metadata on the Extended Attribute view. • **Delete History**—Permits user to delete a package's history log (which contains a record of any operation that materially changes a software package or the data associated with it).

Table 5-1 • AdminStudio Client Tools Permissions (cont.)

Category	Permission
Application Manager / Conflict Solver Other	• **Modify Tools Options Dialog**—Permits user to open the Application Manager/ConflictSolver **Options** dialog box and set options. • **Change Application Catalog**—Permits the user to connect to a different Application Catalog by selecting **Connect** on the **Catalog** menu. Without this permission, this selection is disabled. • **Run Merge Wizard**—Permits user to use the Merge Wizard to merge the application data of a specified Application Catalog into the currently open Application Catalog. • **Run Validation**—Permits user to use the Validation Wizard to validate a Windows Installer package against custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards. • **Scan for Dependencies**—Permits user to generates a list of all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog. • **Run Directory Monitoring**—Permits user to use the **Network Directory** option of **Package Auto Import** to monitor a directory location on the network (or a local directory) and automatically import or re-import packages in that directory at scheduled intervals. • **Run Best Practices**—Permits user to use the Conflict Wizard or the ConflictSolver Process Assistant to evaluate source packages to see if they meet Windows Installer best practices rules.
Patch Impact Manager	• **Import Patch**—Permits user to use **OS Security Patch Wizard** to import Windows operating system patches into the Application Catalog. • **Run Analysis**—Permits user to use the ConflictSolver **Patch Impact Analysis Wizard** to analyze the impact of installing an OS Security patch on user machines.
Software Repository	• **Overwrite**—Permits user to import a duplicate package into the Software Repository, overwriting the existing version.

AdminStudio Enterprise Server Tools Permissions

The AdminStudio Enterprise Server tools have the following permissions:

Table 5-2 • AdminStudio Enterprise Server Tools Permissions

Category	Permission
People	• **View**—Permits user to see the **Administration**/**Users & Groups** tab. • **Add**—Permits user to add new users and import a users list from Active Directory. • **Edit**—Permits user to modify existing users. • **Delete**—Permits user to delete existing or newly created users.
Roles	• **View**—Permits user to see the **Administration**/**Roles** tab. • **Add**—Permits user to add new Roles. • **Copy**—Permits user to copy an existing Role. • **Edit**—Permits user to modify an existing Role. • **Email Notify Enabled**—*Not currently in use*.
Directory Services	• **View**—Permits user to see the **Administration**/**Directory Services** tab and view the **Directory Services** page, the **Directory Services List** page, and the **Directory Services Attributes Administration** page. • **Add**—Permits user to add a new Directory Service. • **Edit**—Permits user to edit an existing Directory Service. • **Delete**—Permits user to delete existing or newly created Directory Services.
System Settings	• **View**—Permits user to open the **Administration**/**System Settings** tab. • **Edit**—Permits user to edit the fields on the **Administration**/**System Settings** tab: AdminStudio Enterprise Server User Account and the Workflow Manager/ AdminStudio Enterprise Server Serial Numbers.

Table 5-2 • AdminStudio Enterprise Server Tools Permissions (cont.)

Category	Permission
Job Manager	• **View**—Permits user to see the **Job Manager** tab only if that user also has at least one of the following Job Manager permissions: **Manage Jobs**, **Manage Templates**, or **Configuration**. If you only have the Job Manager **View** permission (and none of the other Job Manager permissions), you will not be able to view the **Job Manager** tab or any of its subtabs. • **Manage Templates**—Permits user to see the **Job Manager**/**Manage Templates** tab, view the list of Templates on the **Job Template List** page, and view the details of an existing Template on the **Edit Template** page. Users with only the **View** and **Manage Templates** permission cannot create a new Template or edit an existing Template. When a Role is assigned the **Manage Templates** permission, the **View** permission is automatically selected. • **Manage Jobs**—Permits user to see the **Job Manager**/**Manage Jobs** tab, view the list of Jobs on the **Jobs Queue** page, and view the details of a Job that they created on the **Edit Job** page. Users with only the **View** and **Manage Jobs** permission cannot create a new Job or edit an existing Job. When a Role is assigned this permission, the **View** permission is automatically selected. • **Manage Jobs for All Owners**—Adds the **Show All Jobs** check box option to the **Jobs Queue** page, which enables this user to see a list of Jobs created by all users. For users without this permission, only those Jobs that they created are listed. • **Configuration**—Permits the user to view the list of conflict detection rules that are selected on the **Conflict Detection Options** page. However, if this user does not also have the **Edit** permission, this user will not be able to make any changes to this list. When a Role is assigned this permission, the **View** permission is automatically selected. • **Edit**—If a user also has the **Manage Templates** and/or **Manage Jobs** permissions, this permission allows the user to create new Templates and Jobs, and also to edit existing Templates and Jobs. If this user has this permission and also has the **Configuration** permission, this user can modify the selections on the **Conflict Detection Options** page.

Table 5-2 • AdminStudio Enterprise Server Tools Permissions (cont.)

Category	Permission
Report Center / **All Reports**	• **Add**—Permits user to click **Add** on the **All Reports** page and create a new Custom Report using the Report Wizard. • **Edit**—Permits user to edit an existing Custom Report. • **Delete**—Permits user to delete an existing Custom Report.
Report Center / **Package Reports**	• **View**—Permits user to see the **Package Reports** tab and view Package Reports.
Software Distribution	• **Operate**—Permits user to use Configuration Manager Web Console (the **Distribute** tab of AdminStudio Enterprise Server) to configure package distribution options and distribute setup packages without going to Configuration Manager Server.

Workflow Manager Permissions

Workflow Manager has the following permissions:

Table 5-3 • Workflow Manager Permissions

Category	Permission
Application Request	• **View**—Permits user to view the **Properties** and **Issues** tabs on the **Application Progress** page of existing Application Requests. • **Add**—Permits user to create a new Application Request. • **Edit**—Permits user to edit an existing Application Request. • **Copy**—Permits user to copy an existing Application Request. • **Delete**—Permits user to delete an existing Application Request. • **Monitor Application Progress**—Permits user to see the **Application Progress**, **Uploaded Files**, and **Downloadable Files** tabs of the **Application Progress** page. Also permits user to complete Workflow Steps. • **View Related Applications**—Permits user to view the **Related Applications** tab of the **Application Progress** page. • **Add Related Applications**—Permits user to link one Application Request to another on the **Related Applications** tab. • **Delete Related Application Links**—Permits user to delete Related Application links. • **Edit Application Due Period**—Permits user to edit the **Application Due Period** field on the **Properties** tab of the **Application Progress** page.

Table 5-3 • Workflow Manager Permissions (cont.)

Category	Permission
Application Status Management	• **View**—Permits user to view the **Application Status Administration** page and to view (but not update) Application Status details on the **Edit Application Status** view. • **Add**—Permits user to add a new Application Status. • **Edit**—Permits user to update Application Status details on the **Edit Application Status** view. • **Delete**—Permits user to delete an existing Application Status
Calendar Settings	• **View**—Permits user to view the **Calendar Settings Administration** page. • **Edit**—Permits user to edit the settings on the **Calendar Settings Administration** page.
Issues	• **View**—Permits user to view the **Issues** tab of the **Application Progress** page and to view the issue details. • **Add**—Permits user to create a new **E-mail** or **Knowledge Base** issue . • **Respond**—Permits user to respond to an existing issue. • **Close**—Permits user to close an **E-mail** or **Knowledge Base** issue. • **Add/Close Critical Issue**—Permits user to create and close **Critical** issues.
My Notifications	• **View**—Permits user to view the **My Notifications** pages: **My Default Project Notifications** and **My Application Notifications**. • **Edit**—Permits user to edit the settings on the **My Notifications** pages: **My Default Project Notifications** and **My Application Notifications**.
Project Permissions	• **View**—Permits user to view (but not edit) the permission and email settings on the **Project Permission and Notification Settings** view. • **Edit**—Permits user to edit the permission and email settings on the **Project Permission and Notification Settings** view.
Projects	• **View**—Permits user to view the **Project Administration** page and view (but not update) Project details on the **Project Details** view. • **Add**—Permits user to create a new Project. • **Edit**—Permits user to edit the details of an existing Project. • **Delete**—Permits user to delete a Project.
Requestor Company	• **View**—Permits user to view a list of existing Requestor companies and view Requestor company details. • **Add**—Permits user to add new Requestor companies. • **Edit**—Permits user to update Requestor company details. • **Delete**—Permits user to delete an existing Requestor company.

Table 5-3 • Workflow Manager Permissions (cont.)

Category	Permission
Search	• **Simple Search**—Permits user to enter keywords in the **Application Search** box on the **Workflow Manager Home** page to search for an Application Request by name. • **Advanced Search**—Permits user to search for Application Requests by multiple criteria on the **Application Search** page.
Servicer Company	• **View**—Permits user to view a list of existing Servicer companies, view Servicer company details, and update those details. • **Add**—Permits user to create new Servicer companies.
Templates	• **View**—Permits user to view the Template Administration page and view Template details on the Template Details page. Also gives you permission to add an external data source on the **External Data Sources** subtab of the **Administration** tab. • **Add**—Permits user to edit an existing Template and create a new Template. • **Copy**—Permits user to copy an existing Template.
Terminology	• **View**—Permits user to view and modify AdminStudio Enterprise Server system terminology.
Work Assignment	• **View**—Permits user to view the **Working Queue View** and **Work Assignment Listing** pages to see a list of existing work assignments. • **Assign**—Permits user to assign work on the **Assignment Details** page.
Global Email Administration	• **View**—Permits user to view the **Email Settings Administration** and **External Email Address Administration** pages. • **Edit**—Permits user to edit the settings on the **Email Settings Administration** and **External Email Address Administration** pages.
Deployment Sites	• **View**—Permits user to see the **Deployment Site Administration** page. • **Add**—Permits user to add a new Deployment Site. • **Edit**—Permits user to edit an existing Deployment Site.

System Roles

AdminStudio and Workflow Manager come with default **System Roles** which cannot be modified. These Roles were created based upon the needs of typical AdminStudio and Workflow Manager users, providing each Role with only the permissions a user would need to perform their required tasks. You can either assign these **System Roles** to users, or you can copy **System Roles** and then modify them to customize them for your organization.

Copied System Roles or new Roles that you create have a **Role Type** of **User** (User Roles), while default Roles created during installation have a **Role Type** of **System** (System Roles). A Role's **Role Type** is listed on the **Role Administration** page and cannot be changed.

Information about System Roles is organized in the following sections:

- Superuser Role: AMSSuper
- Default System Roles
- Default System Users

Superuser Role: AMSSuper

When AdminStudio Enterprise Server and Workflow Manager are installed, the following user is created and is assigned the **AMSSuper** Role:

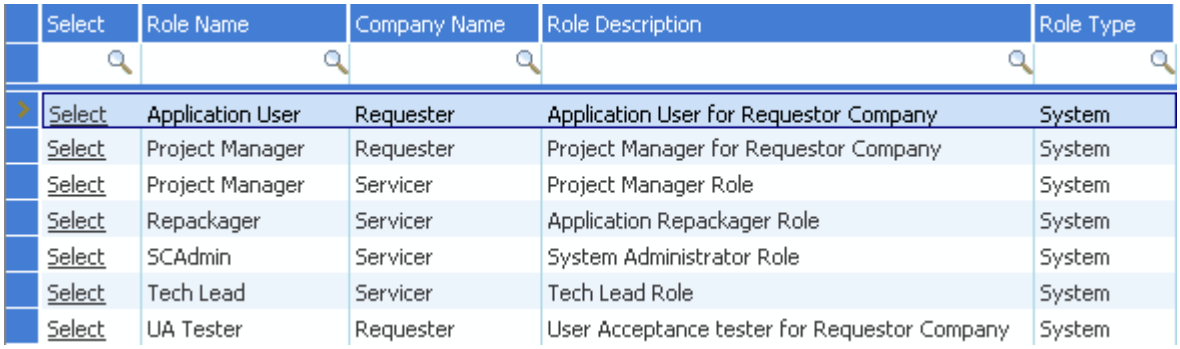
- **User Name**: `suams`
- **Password**: `suams`

A user assigned the **AMSSuper** Role can create Servicer companies and customize terminology. All other tasks should be performed by a Service Company administrator or by accounts created by a Service Company administrator.

Note • *The **AMSSuper** Role is not listed on the* `Role Administration` *page unless you are logged on using the* `suams` *account.*

Default System Roles

When AdminStudio Enterprise Server and Workflow Manager are installed, the following System Roles are created:



Select	Role Name	Company Name	Role Description	Role Type
Select	Application User	Requester	Application User for Requestor Company	System
Select	Project Manager	Requester	Project Manager for Requestor Company	System
Select	Project Manager	Servicer	Project Manager Role	System
Select	Repackager	Servicer	Application Repackager Role	System
Select	SCAdmin	Servicer	System Administrator Role	System
Select	Tech Lead	Servicer	Tech Lead Role	System
Select	UA Tester	Requester	User Acceptance tester for Requestor Company	System

Figure 5-1: Default System Roles

Note • *If you did not purchase Workflow Manager, the Roles associated with the Requester company are not listed:* ***Application User***, ***Project Manager***, *and* ***QA Tester***.

Note • *If you are connected to an Application Catalog that has been upgraded from AMS 2.5, all Roles will have a* ***Role Type*** *of* ***User***.

The following table lists the default System Roles:

Table 5-4 • Default System Roles

Role Name	Company Name	Description
Project Manager	**Servicer**	Project Manager for Servicer Company
Repackager	**Servicer**	Application Repackager for Servicer Company
SCAdmin	**Servicer**	System Administrator for Servicer Company
Tech Lead	**Servicer**	Tech Lead for Servicer Company
Application User	**Requester**	(Workflow Manager only) Application User for Requester Company
Project Manager	**Requester**	(Workflow Manager only) Project Manager for Requester Company
UA Tester	**Requester**	(Workflow Manager only) User Acceptance Tester for Requester Company

You can view the permissions of each of these Roles by selecting the Role on the **Role Administration** page, and then expanding the **Role Permissions** list.

Default System Users

When AdminStudio Enterprise Server and Workflow Manager are installed, a User Account is created for each of these System Roles:

Select	User Name	Roles	Company
Select	admin@servicer.com	SCAdmin	Servicer
Select	pm@requester.com	Project Manager	Requester
Select	pm@servicer.com	Project Manager	Servicer
Select	repackager@servicer.com	Repackager	Servicer
Select	techlead@servicer.com	Tech Lead	Servicer
Select	tester@requester.com	UA Tester	Requester
Select	user@requester.com	Application User	Requester

Figure 5-2: Default System Users

To see what functionality is available for one of these default System users, you could log on as that user and then observe what features are available. You could also select the System user's associated Role on the **Role Administration** page, and then expand the **Role Permissions** list.

Note • *By default, the password for each of these default System Users is the same as the text prior to the @ sign (such as **techlead** for techlead@servicer.com), except the password for pm@requester.com and pm@servicer.com is **projectmanager**.*

Creating a New Role

To create a new Role, perform the following steps.

Task: ***To create a new Role:***

1. On the **Administration** tab, click **Roles**. The **Role Administration** page opens.
2. Click the **Add** button. The **Role Details View** opens.
3. Enter a name in the **Role Name** field to uniquely identify this Role.
4. From the **Role Company** list, select the name of the Company whose users can be assigned this Role.
5. In the **Role Description** text box, describe the purpose of this new Role.
6. In the **Role Permissions** list, select the permissions you want to assign to this Role. For a detailed list of the available permissions, see Role Permission Lists.
7. Click the **Update** button. The new Role now appears in the list on the **Role Administration** page.

Editing an Existing User Role

You can edit all Roles with a **Role Type** of **User**, but System Roles cannot be modified.

Task: ***To edit an existing User Role:***

1. On the **Administration** tab, click **Roles**. The **Role Administration** page opens.
2. Select the Role that you want to edit. The **Role Details View** opens.
3. Make edits to the **Role Name** and **Role Description** fields, if desired.

   Note • *It is not recommended that you change a Role's* ***Role Company.***
4. In the **Role Permissions List**, edit the permissions assigned to this Role.
5. Click the **Update** button. You are returned to the **Role Administration** page.

Copying an Existing Role

You can make a copy of an existing User or System Role using a new Role Name. You can then modify it to customize it for your organization. You cannot modify a System Role. You can also copy an existing Role from one Company to another.

To copy an existing Role, perform the following steps.

Task: ***To copy an existing Role:***

1. On the **Administration** tab, click **Roles**. The **Role Administration** page opens.
2. Click the **Copy** button. The **Copy Role View** opens.
3. From the **Copy from Company** list, select the name of the Company whose Role you want to copy.
4. From the **Copy from Role** list, select the Role that you want to copy.
5. From the **Copy to Company** list, select the Company that you are creating this new Role for.
6. Enter a name in the **New Role Name** field to uniquely identify this Role.

   Note • *You are not permitted to have two Roles in the same Company with the same name. You are permitted to use the same Role name in more than one Company.*
7. From the **Role Company** list, select the name of the Company whose users can be assigned this Role.
8. In the **Role Description** text box, enter a description for the Role.
9. In the **Role Permissions List**, edit the permissions that were previously assigned to this Role.
10. Click the **Update** button. The new Role now appears in the Role list on the **Role Administration** page.

Deleting a Role

You can delete all Roles with a **Role Type** of **User**, but **System** Roles cannot be deleted.

Task: ***To delete an existing User Role:***

1. On the **Administration** tab, click **Roles**. The **Role Administration** page opens.
2. Select the user-defined Role that you want to delete. The **Role Details View** opens.
3. Click **Delete**. You are prompted to confirm the deletion.
4. Click **OK**. The **Role Details View** closes and the Role you deleted is no longer listed on the **Role Administration** page.

Roles Reference

Reference information for Roles is presented in the following sections:

- Role Administration Page
- Copy Role View
- Role Details View

Role Administration Page

The **Role Administration** page lists all defined Roles on the system. From this view, you can edit existing Roles or add new ones. The following options are available:

Table 5-5 • Role Administration Page

Option	Description
Select	Click to access the Role Details View, where you can edit this Role.
Role Name	Name of Role.
Role Company	Name of company that this Role was defined for.
Role Description	Description of the purpose of this Role.
Role Type	Identifies this role as one of the following: • **System**—Role was created during installation and cannot be deleted or modified. However, it can be copied and modified to create a User Role. • **User**—Role created by an administrator either by copying an existing System Role or by creating a new Role. For more information, see System Roles.
Add	Click to access the Role Details View, where you can add a new Role.
Copy	Click to access the Copy Role View, where you can copy an existing Role.

Copy Role View

On the **Copy Role** view of the **Role Administration** page you can copy an existing System or User Role.

Table 5-6 • Role Administration Page / Copy Role View Options

Option	Description
Copy from Company	Select the name of the Company that has a Role that you want to copy.
Copy from Role	Select the name of the Role that you want to copy. You can copy either System or User Roles.
Copy to Company	Select the name of the Company that this new Role is being created for.
New Role Name	Enter a name to identify this new Role.
Copy	Click to create this new Role and close the Copy Role view.
Cancel	Click to exit this view without creating a new Role.

Role Details View

Permissions are assigned to Roles, and Roles are assigned to users. Therefore, all permissions in AdminStudio and Workflow Manager are controlled by Roles.

On the Role Details view, you can enter or edit information to identify a Role and set the Role Permissions. You can also delete a user-defined Role.

Table 5-7 • Role Administration Page / Role Details View Options

Option	Description
Role Name	Enter a name to uniquely identify this Role
Role Company	Select the Company whose employees can be assigned the Role.
Role Description	Enter text to identify the purpose of this Role.
Role Permissions List	Select the specific functionality permissions that you want to assign to this Role. For more information, see Role Permission Lists and System Roles.
Update	Click to save your entries and close the Role Details view.
Delete	Click to delete this user-defined Role. ***Note*** • *If you are editing a System Role, the* ***Delete*** *button will be disabled.*
Cancel	Click to exit this view without saving your entries.

6

Managing Application Catalog Databases

Edition • *Application Manager is included with AdminStudio Professional and Enterprise Editions.*

You use Application Manager to manage your applications and their deployment types in Application Catalog databases. Tasks you perform using the Application Manager include importing packages, merging application catalogs, setting up automatic package import, organizing packages into groups, and viewing a package's deployment types, extended attributes, files/components, INI file changes, registry entries, shortcuts, merge modules, history, and tables.

Information about using Application Catalogs is organized into the following sections:

Table 6-1 • Topics Regarding Using Application Catalogs

Section	Description
Application Manager and ConflictSolver	Describes the functionality available using Application Manager and how it differs from ConflictSolver.
Managing Application Catalogs	Explains how to create, connect to, search for, organize, view information about, and delete packages from.
Importing Data	Explains how to import Windows Installer packages (`.msi`) with any associated transforms (`.mst`) or patches (`.msp`), merge modules (`.msm`), OS snapshots (`.osc`), Marimba NCP files (`.ncp`), and other setup types. It also explains how to import virtual applications in Microsoft App-V, Citrix XenApp, and VMware ThinApp formats. ***Edition*** • *Import support for Marimba Native Channel Packager (`.ncp`) files is available in AdminStudio Enterprise Edition.*

Table 6-1 • Topics Regarding Using Application Catalogs (cont.)

Section	Description
Viewing Application Catalog Enhanced Reporting	Describes the available Application Catalog summary reports—graphical representations of summary data concerning the readiness of Windows Installer and App-V packages for distribution—and how to create your own customizable reports.
Viewing Package Data	Describes how to view package metadata for Windows Installer and App-V packages in Application Manager.
Using the Software Repository	Explains how to manage a software package's associated installation files using the Software Repository.
Sharing Application Catalog Data	Describes the three methods for sharing Application Catalog data in your enterprise: merge, auto-import, and replication.
Taking OS Snapshots	Explains how to create OS Snapshots using the OS Snapshot Wizard.
Reference	Describes every view, dialog box, and Wizard available in Application Manager.

Application Manager and ConflictSolver

Application Manager and ConflictSolver are closely related tools. Both provide a comprehensive view of the Application Catalog. However, the tasks you can perform using each of these tools differ:

- **ConflictSolver** is used to perform conflict detection and resolution for Windows Installer and App-V packages.
- **Application Manager** is used for importing and storing packages in a database, and organizing those packages.

This section compares these tools and includes the following topics:

- Functionality Comparison
- Accessing the Tools
- Displaying ConflictSolver Options in Application Manager
- Creating and Connecting to Application Catalogs

Functionality Comparison

This table compares the functionality included in Application Manager and ConflictSolver.

Table 6-2 • Comparison of ConflictSolver and Application Manager Functionality

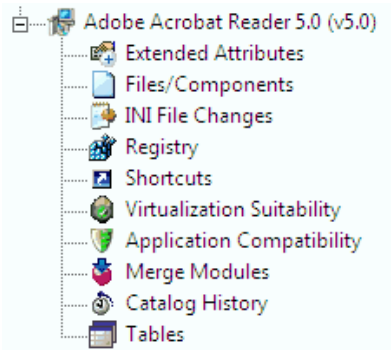
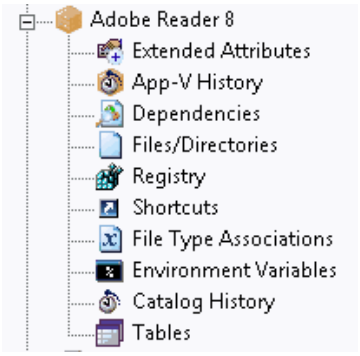
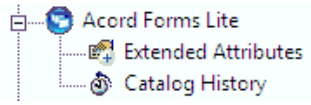
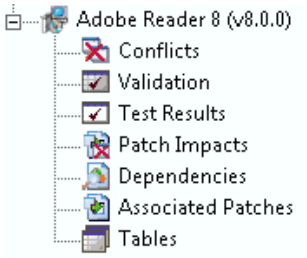
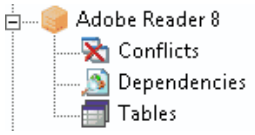
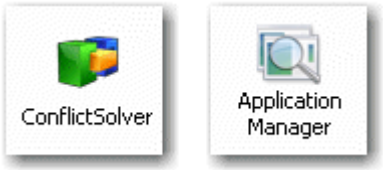
View	Major Functionality	Package Sub-Nodes
Application Manager	Use to perform the following tasks • **Import packages**—Import packages into the Application Catalog. • **Manage packages**—Organize packages into Groups, rename packages, and delete packages. • **Merge Wizard**—Merge data from a source Application Catalog into the currently open Application Catalog. • **Package Auto Import**—Set up the automatic import of packages. • **Software Repository**—Import packages into and check in and check out packages from the Software Repository. • **Associate a package with a Workflow Manager package** • **Launch applications**—Launch Predeployment Test Preparation Wizard, InstallShield Editor, and Distribution Wizard. • **Run Virtualization Readiness**—Run the Virtualization Readiness tests and view a package's subsequent Virtualization Suitability report. • **Reports**—View Package, File, and Registry reports. • **Application Compatibility**—View the results of Compatibility Solver application compatibility testing.	**Windows Installer Sub-Nodes**  **App-V Sub-Nodes**  **XenApp and ThinApp Sub-Nodes** 

Table 6-2 • Comparison of ConflictSolver and Application Manager Functionality

View	Major Functionality	Package Sub-Nodes
ConflictSolver	Use ConflictSolver to perform the following tasks: • **Conflict Analysis and Resolution**—Check one or more packages for conflicts against others in the Application Catalog, and resolve those conflicts. • **Validation**—Validate Windows Installer package against Microsoft ICEs (Internal Consistency Evaluators), • **Test Results**—View PackageExpert test results. • **Patch Impact Analysis Wizard**—Launch Patch Impact Analysis Wizard for selected package. • **Dependency Scan**—Generate a list of all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog. • **Associated Patches**—View a list of imported patches that, if installed, would update the selected product. • **Reports**—View Package, File, and Registry reports. For information on using ConflictSolver, see Identifying and Resolving Application Conflicts Using ConflictSolver.	**Windows Installer Sub-Nodes**  **App-V Sub-Nodes** 

Accessing the Tools

Both Application Manager and ConflictSolver can be opened from the Start Page, the Tools Gallery, or from a menu shortcut.



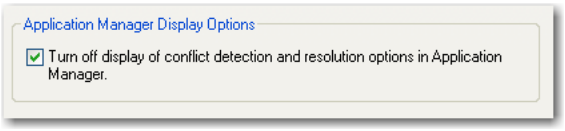
You can toggle between Application Manager and ConflictSolver by clicking a button in the toolbar:

When the ConflictSolver or Application Manager is active, **ConflictSolver** or **Application Manager** is listed in the Title bar and also in the top right corner of the interface.

Displaying ConflictSolver Options in Application Manager

By default, all functionality for interacting with the Application Catalog, including conflict detection and resolution, is available in Application Manager.

To turn off the display of conflict detection and resolution options in Application Manager, select the check box in the **Application Manager Display Options** area of the **General** tab of the ConflictSolver and Application Manager **Options** dialog box.



Creating and Connecting to Application Catalogs

You can connect to Application Catalogs from both Application Manager and ConflictSolver as described in Connecting to an Existing Application Catalog. However, you use the AdminStudio interface to create new application catalogs. See Creating New Application Catalogs.

Managing Application Catalogs

Package information is stored in Application Catalogs. When you use Application Manager, you will either automatically be logged into an Application Catalog, or you will have to connect to an existing catalog. Once you have connected to a catalog, you can manipulate data in Application Manager, including creating and organizing groups and entering extended attribute data.

The following topics relate to Application Catalog management:

- About AdminStudio Application Catalogs
- Using the AdminStudio Quick Start Evaluation Experience
- Connecting to an Existing Application Catalog
- Creating New Application Catalogs
- Specifying a Default AdminStudio Application Catalog
- Searching an Application Catalog
- Disconnecting from an Application Catalog
- Upgrading an Existing Application Catalog
- Organizing Your Application Catalog Using Groups
- Viewing and Editing Package Extended Attributes
- Viewing Package Catalog History
- Deleting Packages

About AdminStudio Application Catalogs

This section provides the following information:

- Application Manager Organization and Structure
- Overview of Application Catalogs
- Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog

Application Manager Organization and Structure

Starting with AdminStudio 10.0, Application Manager and ConflictSolver follow a new “application-centric” model. The tree has been restructured to display multiple deployment formats under a parent **Application** node. The following image illustrates how Application Manager’s tree structure has changed.

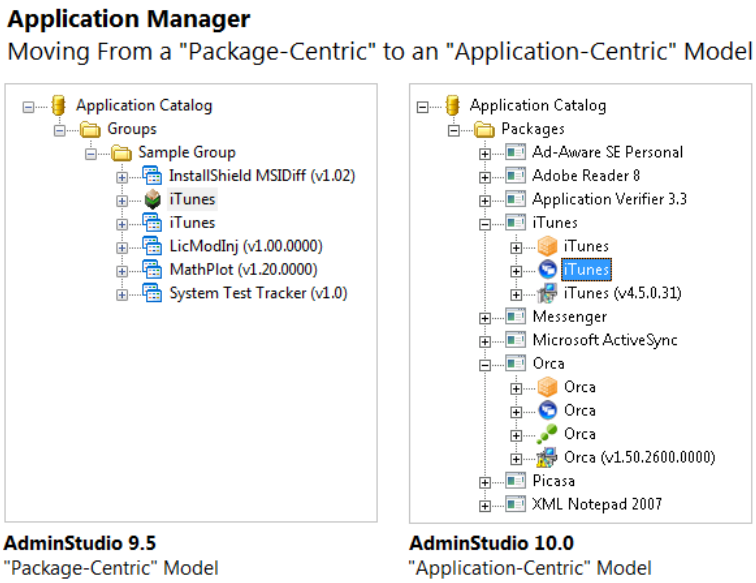


Figure 6-1: AdminStudio 10.0’s Application-Centric Model

You can use this new, unified Application Catalog to manage all deployment types for an application, enabling you to handle all packages—including virtual packages—with the same process and tools as those used for MSIs, eliminating the cost and complexity of having multiple solutions and repositories.

The restructured Application Manager provides the same full functionality as previous versions:

- **Automatic Application node creation**—An Application node will be created automatically each time a package is imported into the Application Catalog, or the user can manually create one in any Application Manager group.
- **Unique subnodes and context menus**—The Application node, as well as the various Deployment Type nodes, have unique subnodes and context menus appropriate for that type.
- **Software repository functionality**—Software repository functionality (such as check-in, check-out) is available for Windows Installer and App-V deployment types.

Application Manager's application-centric structure uses the following icons to represent applications and their deployment types:

Table 7 • Application Manager Tree Icons

Icon	Description
	Group
	Application
	Windows Installer Package (.msi)
	Microsoft App-V Package (.sft)
	Citrix XenApp Profile (.profile)
	VMware ThinApp Package (.exe)
	Legacy Application (.exe)

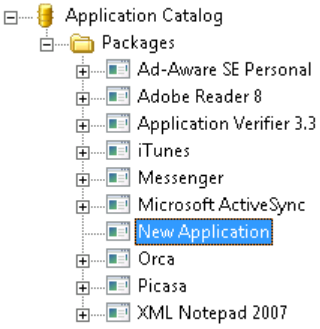
Deployment types are grouped under Application nodes, and Applications are grouped into Groups.

Creating a New Application in Application Manager

To create a new application in Application Manager, perform the following steps.

Task: ***To create a new Application in Application Manager:***

1. Launch Application Manager.
2. Select a group in the tree and select **New Application** from the context menu. A new application is created.



3. Enter an appropriate name for the application.
4. To import a deployment type for that application, select the application node and, from the context menu, point to **Import** and click **Package**.
5. Proceed with the steps in Importing Data.

Overview of Application Catalogs

To perform enterprise level data checking, it is very efficient if the data can be consolidated into a single database. The AdminStudio Application Catalog database consolidates data from many installation packages in a single location.

Application Catalogs can contain the complete contents of Windows Installer (`.msi`) packages, Merge Module (`.msm`) packages, and OS snapshots (`.osc`), and virtual packages in Microsoft App-V, Citrix XenApp, and VMware ThinApp formats. Depending upon the import options that are set, an Application Catalog can also include binary records. Application Catalogs can also store application, workflow, permissions, Predeployment Test, Microsoft Patch, PackageExpert, and Workflow Manager data.

You can use ConflictSolver and PackageExpert to test packages in the Application Catalog:

- **ConflictSolver**—You can use ConflictSolver's MSI ICEs (internal consistency evaluators) to evaluate specific packages, and can use ConflictSolver ACEs (application conflict evaluators) to extend the ICEs and evaluate data across packages.
- **PackageExpert**—You can use PackageExpert to analyze the integrity of packages in the Application Catalog and also identify and resolve any issues packages would have when deployed on the Windows Vista operating system.

Sharing Application Catalogs

In multi-user environments, the AdminStudio Application Catalog is typically shared. The AdminStudio Administrator can make a shared catalog the default catalog, which results in all AdminStudio users who use the same shared location using the same shared Application Catalog. If the shared Application Catalog is changed, users will automatically open up the new shared Application Catalog. For more information, see Specifying a Default AdminStudio Application Catalog.

Sharing Application Catalog Data

There are three methods for sharing Application Catalog data in your enterprise:

- **Merge Wizard**—Use to merge data from a source Application Catalog into the currently open Application Catalog.
- **Package Auto Import Wizard**—Offers two methods of automatically importing Windows Installer packages into your Application Catalog and keeping them updated.
- **Application Catalog Replication**— Enables you to distribute an Application Catalog database where it is needed across a large enterprise or multi-national corporation.

See Sharing Application Catalog Data for detailed information.

Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog

Edition • *AdminStudio Enterprise Server Tools are included with AdminStudio Enterprise Edition.*

You can connect to any standalone SQL Server Application Catalog, or you can connect to the AdminStudio Enterprise Server Application Catalog:

- **Standalone**—A Standalone Application Catalog is not associated with the Enterprise Server tools. The AdminStudio client tools connect directly to the database server hosting the standalone Application Catalog.
- **AdminStudio Enterprise Server**—All of the AdminStudio Enterprise Server tools—Security Console, Report Center, Job Manager, and Workflow Manager—are connected to the AdminStudio Enterprise Server Application Catalog. AdminStudio client tools can also connect to this Application Catalog, allowing you to store all data generated on a package in the same location, and to link packages in this Application Catalog to Workflow Manager Application Requests.

  Note • *AdminStudio Enterprise Tools are always connected to the AdminStudio Enterprise Server database.*

For more information on using Application Catalogs, see the following:

- Connecting AdminStudio Client Tools to a Standalone Application Catalog
- Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog

Using the AdminStudio Quick Start Evaluation Experience

When you initially open AdminStudio, because a default Application Catalog has not yet been set, the **Default Application Catalog** dialog box opens, prompting you to open an Application Catalog.

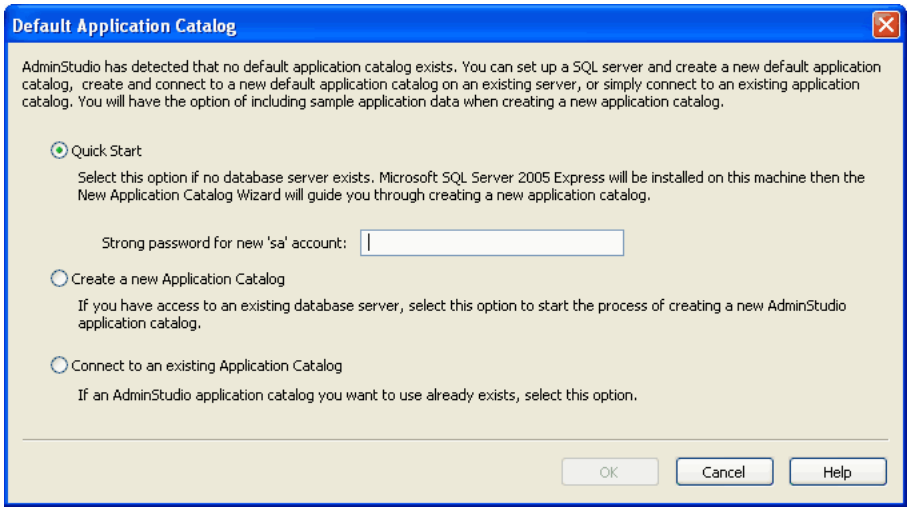


Figure 6-2: Default Application Catalog Dialog Box

To quickly get started using AdminStudio, select one of the following options:

- **If you do not have access to a database server**, but you want to quickly start using AdminStudio, select the **Quick Start** option. Microsoft SQL Server 2005 Express will be installed on your machine first, then you will have the option to create a new Application Catalog database on that server that is populated with sample data.
- **If you do have access to a database server**, but just do not have any sample data in it, select the **Create a new Application Catalog** option, and choose to populate this new Application Catalog with sample data by selecting the **Include Sample Data in New Catalog** option on the **Create Application Catalog** dialog box.

Connecting to an Existing Application Catalog

Edition • *AdminStudio Enterprise Server Tools are included with AdminStudio Enterprise Edition.*

From AdminStudio and the AdminStudio client tools, you usually connect to an Application Catalog by selecting **Connect** on the **Catalog** menu. AdminStudio supports SQL Server databases.

Note • *Starting with AdminStudio 9.01, Oracle databases are not supported.*

You can choose to connect to a standalone Application Catalog database or the AdminStudio Enterprise Server Application Catalog database.

- Connecting AdminStudio Client Tools to a Standalone Application Catalog
- Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog

Note • *See Standalone Application Catalog vs. the AdminStudio Enterprise Server Application Catalog for more information.*

Note • *AdminStudio Enterprise Tools are always connected to the AdminStudio Enterprise Server database.*

Connecting AdminStudio Client Tools to a Standalone Application Catalog

To connect to an existing Standalone Application Catalog from an AdminStudio client tool (Application Manager, ConflictSolver, or PackageExpert), perform the following steps.

Task: ***To connect to an existing Standalone Application Catalog from the AdminStudio client tools:***

1. Perform one of the following:
   - **Application Manager and ConflictSolver**—On the **Catalog** menu, click **Connect**.
   - **PackageExpert**—Select the **Application Catalog** node and select **Connect** from the context menu.

   The **Connect Application Catalog** dialog box opens, displaying three tabs: **Enterprise Server**, **Standalone**, and **Recent**.
2. Click the **Standalone** tab. The Standalone tab opens, prompting you to enter database connection information.
3. If you want this Application Catalog to be the default shared Application Catalog used in your organization, select the corresponding option at the bottom of the dialog box.
4. Select the **Server** where the Application Catalog is stored.
5. Specify how the database server should verify the authenticity of the login—either using **Windows Authentication** or **Server Authentication**. If you selected **Server Authentication**, enter the appropriate **Login ID** and **Password**.
6. In the **Catalog** box, enter the name of the Application Catalog you want to open.
7. Click **Test** to test the connection to the database.
8. Click **OK**.

Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog

The AdminStudio Enterprise tools (Report Center, Job Manager, and Workflow Manager) are configured during installation to connect to an Application Catalog, which is referred to as the Enterprise Server Application Catalog.

You can also connect the AdminStudio *client* tools to the Enterprise Server Application Catalog. This allows you to have all of the client and enterprise tools reference the same database.

To connect to the AdminStudio Enterprise Server application catalog from an AdminStudio client tool, perform the following steps.

Task: ***To connect to the AdminStudio Enterprise Server Application Catalog:***

1. Perform one of the following:
   - **Application Manager and ConflictSolver**—On the **Catalog** menu, click **Connect**.
   - **PackageExpert**—Select the **Application Catalog** node and select **Connect** from the context menu.

   The **Connect Application Catalog** dialog box opens, displaying three tabs: **Enterprise Server**, **Standalone**, and **Recent**.
2. Open the **Enterprise Server** tab.
3. The URL to the AdminStudio Enterprise Server is listed above the **Authentication** field. If the AdminStudio Enterprise Server has not yet been configured with the AdminStudio client tools (such as when it is set to its default value of `http://localhost`), click the URL link to open the **Select AdminStudio Enterprise Server URL** dialog box, and enter the URL for location of the AdminStudio Enterprise Server associated with this installation of AdminStudio.
4. From the **Authentication** list, select either **AdminStudio Enterprise Server User** or **Windows Authentication**.

   Important • *When using **AdminStudio Enterprise Server User** authentication, if Anonymous authentication is turned off in IIS, both the user's machine and the AdminStudio Enterprise Server need to be on the same domain in order for login to succeed.*
5. If you selected **AdminStudio Enterprise Server User**, enter your AdminStudio Enterprise Server **User Name** and **Password** (provided by your System Administrator).
6. Click **Login**. After a successful login, the **Provider**, **Server**, and **Catalog** name of the Enterprise Server database is listed.
7. Click **OK**.

Login Troubleshooting

If you are using a Web Portal with custom security zone settings, your AdminStudio Enterprise Server URL is using an IP address, and you receive Error 0x800A1518 when you attempt to login, change the AdminStudio Enterprise Server URL to the NetBios equivalent and then try again. For example, if you are connecting to `http://120.12.1.15`, the NetBios equivalent would be `http://wfmportal`.

Creating New Application Catalogs

You can create new Application Catalogs using the AdminStudio interface, or using scripts. This section includes the following topics:

- Creating New Application Catalogs Using the AdminStudio Interface
- Creating New Application Catalogs Using Scripts

Creating New Application Catalogs Using the AdminStudio Interface

To create a new SQL Server application catalog database, perform the following steps.

Task: ***To create a new standalone Application Catalog:***

1. On the AdminStudio **Catalog** menu, click **Create**. The **Welcome** panel of the **Application Catalog Wizard** opens.
2. If you want to include a few sample packages in this new Application Catalog, select the **Include Sample Data in New Catalog** option.
3. Click **Next**. The **Specify Database Information** panel opens.
4. Enter or select the name of the **Server** where this Application Catalog will be stored.
5. Specify how the database server should verify the authenticity of the login—either using **Windows Authentication** or **Server Authentication**. If you selected **Server Authentication**, enter the appropriate **Login ID** and **Password**.
6. In the **Catalog** box, enter the name of the Application Catalog you are creating.
7. If you are creating a new SQL Server database which is intended for use as a Subscriber for Application Catalog Replication, select the **Create catalog as a subscriber** option. If you do not intend to use the Replication functionality that AdminStudio provides for SQL Server, do not select this option. For more information, see About Creating a Catalog as a Subscriber and Replicating Application Catalogs.
8. Click **Test** to test the connection to the database.
9. Click Next. The **Select Software Repository Location** panel opens, prompting you to select the location where the Software Repository will store imported packages and their associated files. For more information, see Using the Software Repository.

10. If you want to store data associated with this Application Catalog's packages in the Software Repository, choose the **Enable the Software Repository** option, select a Software Repository Location, and enter a Login ID and Password of the **Proxy Account** that must be used to access this Repository.
11. Click Next. The **Creating Application Catalog** panel opens and reports on the creation progress. When the Application Catalog has been created, a message appears stating that the creation was successful.
12. Click Finish. The new Application Catalog now opens in AdminStudio.

About Creating a Catalog as a Subscriber

Please note the following about creating a Subscriber Application Catalog for Application Catalog Replication:

- You cannot create a Publication in a Subscriber database.
- If you *choose* the **Create catalog as a subscriber** option, making it a **Subscriber** database, the **Subscription Manager** option under **Replication** on the **Catalog** menu is enabled.
- If you *do not choose* this option, making it a **Publisher** database, the **Publication Manager** option under **Replication** on the **Catalog** menu is enabled.
- Once you have created a database, you cannot go back and edit the choice you made on this option.
- Just because you select this option does not mean that this database must contain subscribed data or that all of its data must be subscribed. It can contain all local data or both local and subscribed data.

For more information, see Replicating Application Catalogs.

Tip *• The first time AdminStudio is run, anyone can create an SQL Server database (providing they have access to aN SQL Server and database creation rights). However, once an AdminStudio Application catalog has been created, the database creator becomes the Application Catalog administrator, and security rights are in place in AdminStudio.*

Creating New Application Catalogs Using Scripts

Typically, users with administrative privileges in AdminStudio use **Create** on the **Catalog** menu to create a new Application Catalog.

However, because of security concerns, some Database Administrators may be hesitant to grant the database creation rights that are necessary to create an Application Catalog database using SQL Server to AdminStudio users. Consequently, the Database Administrator must manually create the database using scripts and provide the necessary read and write access for users to that database.

AdminStudio is shipped with database creation SQL scripts to make it easy for Database Administrators to create new Application Catalogs:

Table 6-1 • Scripts to Create Application Catalog Databases on SQL Server

Script File	Description
AMS_System_Schema.SQL	Primary Workflow Manager tables
AMSCreateIndex.SQL	Indexes for Workflow Manager tables
AMS_SamplePackagingTemplate.SQL	Sample Workflow Manager template data
AS_System_Schema.SQL	Primary AdminStudio client tools data
Seed_Data.SQL	Seed data required by AdminStudio
CustomReportWizard.SQL	Report Center data
Reporting.StoredProcedures.SQL	Application Catalog enhanced reporting data

These SQL scripts are located in the following directory:

AdminStudio Installation Directory\Support\SQL_Scripts

To create an Application Catalog database on SQL Server, perform the following steps:

Task: ***To use scripts to create an AdminStudio Application Catalog on SQL Server***

1. Log on to your SQL Server.
2. Launch the Enterprise Manager and Query Analyzer.
3. In Query Analyzer, execute a `CREATE DATABASE` command to create and identify the new Application Catalog database.
4. Select the newly created database in Query Analyzer.
5. Execute the following scripts in order:
   a. AMS_System_Schema.SQL
   b. AMSCreateIndex.SQL
   c. AMS_SamplePackagingTemplate.SQL
   d. AS_System_Schema.SQL
   e. Seed_Data.SQL
   f. CustomReportWizard.SQL
   g. Reporting.StoredProcedures.SQL

Specifying a Default AdminStudio Application Catalog

You can specify a default Application Catalog so that each time you open AdminStudio, you will be prompted to login to the same Application Catalog database.

You can also configure your enterprise so that *all of the users at your enterprise* will be prompted to login to the same Application Catalog each time they open AdminStudio.

Setting a Default Application Catalog for Yourself

Whenever you connect to an Application Catalog, you can designate it as the default Application Catalog by selecting the **Make this the shared default Application Catalog** option on the **Connect Application Catalog** dialog box.

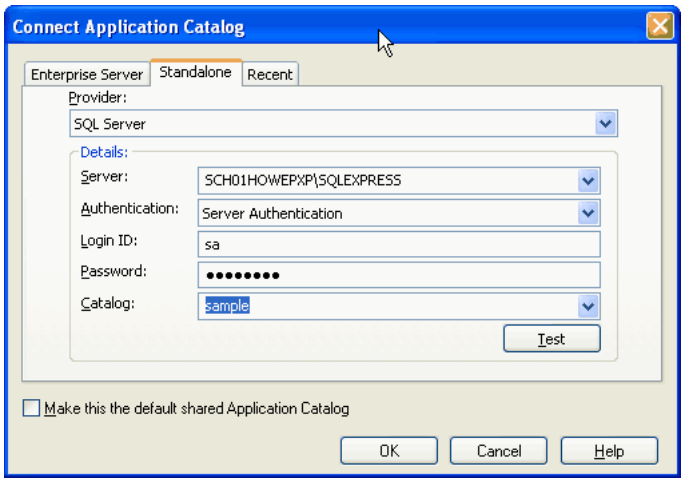


Figure 6-3: Default Shared Application Catalog Option on the Connect Application Catalog Dialog Box

Setting the Default Application Catalog for Your Enterprise

To configure your enterprise so that *all of the users at your enterprise* will be prompted to login to the same Application Catalog each time they open AdminStudio, all users need to be connected to the same `AdminStudio Shared.ini` file that is stored in a shared network location that is available to all users.

Methods to Set the Default Application Catalog for Your Enterprise

You can set a default Application Catalog for all AdminStudio users in your enterprise in two ways:

- Using the AdminStudio Interface

  or

- Editing the Shared AdminStudio.ini File

Using the AdminStudio Interface

To use the AdminStudio interface to set a default Application Catalog for all AdminStudio users at your enterprise, perform the following steps.

Task: ***To specify the default Application Catalog:***

1. First, the AdminStudio System Administrator needs to perform the following steps to set the default Application Catalog for the enterprise:

   a. Locate and copy the following file on the machine where you installed AdminStudio:

      `C:\AdminStudio Shared\Shared AdminStudio.ini`

   b. Copy this file to a shared network location that is accessible to all of the users in your enterprise.

   c. Launch AdminStudio.

   d. On the **Tools** menu, click **Options**. The AdminStudio **Options** dialog box opens.

   a. Open the **Locations** tab.

   b. Set the **AdminStudio Shared Location** to the network location where you copied the `Shared AdminStudio.ini` file, such as:

      `\\servername\AdminStudio Shared`

   c. Open the **Application Catalog** tab.

   d. Click **Change** next to the **Shared AdminStudio Application Catalog** field. The **Select Application Catalog** dialog box opens.

   e. Select the Application Catalog database that you want to be set as the default, following the instructions in Connecting to an Existing Application Catalog.

      After you connect to the database, that database name is listed in the **Shared AdminStudio Application Catalog** field.

   f. Click **OK** to close the **Options** dialog box.

2. Next, each AdminStudio user in the enterprise needs to perform the following steps to set the location of *their* **AdminStudio Shared** directory to the same shared network directory that the System Administrator configured.

   a. Launch AdminStudio.

   b. On the **Tools** menu, click **Options**. The AdminStudio **Options** dialog box opens.

   a. Open the **Locations** tab.

   b. Set the **AdminStudio Shared Location** to the shared network location provided by your System Administrator.

Caution *• If a user is not assigned to a Role that has the* ***Modify AdminStudio Tools Options Dialog*** *permission, they cannot change the location of the* ***AdminStudio Shared*** *directory on the* ***Options*** *dialog. In this situation, the location of the* ***AdminStudio Shared*** *directory would be set during installation.*

Editing the Shared AdminStudio.ini File

To set a default Application Catalog for all AdminStudio users at your enterprise by editing the `Shared AdminStudio.ini` file, perform the following steps.

Task: ***To specify the default Application Catalog by editing the Shared AdminStudio.ini file:***

1. First, the AdminStudio System Administrator needs to perform the following steps to set the default Application Catalog for the enterprise:
   a. Locate and copy the following file on the machine where you installed AdminStudio:

      `C:\AdminStudio Shared\Shared AdminStudio.ini`

   b. Copy this file to a shared network location that is accessible to all of the users in your enterprise.
   c. Open the `Shared AdminStudio.ini` file that you just copied to a shared network location.
2. Insert the following in the `[Database Settings]` section of the `Shared AdminStudio.ini` file:

   ```
   [Database Settings]
   DefaultDatabase=Provider=SQLOLEDB.1;User ID=userid; PWD=password;Initial
   Catalog=nameofdatabase;Data Source=nameofsqlserver;
   ```

3. Next, each AdminStudio user in the enterprise needs to perform the following steps to set the location of *their* **AdminStudio Shared Location** directory to the same shared network directory that the System Administrator configured.
   a. Launch AdminStudio.
   b. On the **Tools** menu, click **Options**. The AdminStudio **Options** dialog box opens.
   a. Open the **Locations** tab.
   b. Set the **AdminStudio Shared Location** to the shared network location provided by your System Administrator.

Caution *• If a user is not assigned to a Role that has the* ***Modify AdminStudio Tools Options Dialog*** *permission, they cannot change the location of the* ***AdminStudio Shared*** *folder on the* ***Options*** *dialog. In this situation, the location of the AdminStudio Shared location would be set during installation.*

Note • *The Roles assigned to a user determine that user's permissions:*

- *The **Create** and **Connect** options on the **Catalog** menu on the AdminStudio Interface are disabled for users that are not assigned to a Role that has permission to perform those actions.*
- *Users that are assigned to Roles that have the **Modify AdminStudio Tools Options Dialog** permission can change the location of the **AdminStudio Shared** folder on the **Options** dialog (accessed by selecting **Options** on the **Tools** menu from the AdminStudio Interface). For those users who do not have that permission, **Options** on the **Tools** menu is disabled, so they are unable to change the location of the **AdminStudio Shared** folder.*

Searching an Application Catalog

You can search for data in Application Catalog tables by using **Find** on the **Edit** menu. You can search all tables in all packages in a group, or just search one column in one table of one package.

Note • *This search is limited to string type columns.*

The tables that are searched depend upon what is selected when the **Find** dialog box is opened:

Table 6-2 • Application Catalog Search Options

If you select...	and specify these options...	this will be searched
Application Catalog	**All Tables** and **All Columns**	All tables and all columns in all of the packages in the Application Catalog.
Group	**All Tables** and **All Columns**	All tables and all columns in all of the packages in the selected group.
Package	**All Tables** and **All Columns**	All tables and all columns in the selected package.
Package	A table and **All Columns**	All columns of a specific table in the selected package.
Package	A table and a column	A specific column in a specific table in the selected package.

Task: ***To search the Application Catalog:***

1. Open an Application Catalog in Application Manager or ConflictSolver.
2. Select the node in the tree (Application Catalog, Groups, a specific group, a specific package, an OS snapshot, etc.) that you want to search.

3. On the **Edit** menu, click **Find**. The **Find** dialog box opens.

   You could also click the Find button ( ) on the toolbar, use Ctrl + F, or choose Find from the selected package or group's context menu.

4. In the **Find What** text box, enter the text that you want to search for.

   Note • *This search is limited to string type columns.*

5. On the **Look In Table** list, select the table that you would like to search, or select **<All Tables>**. When you select a table from this list, the **Look In Columns** list is populated with all of the columns in that table.

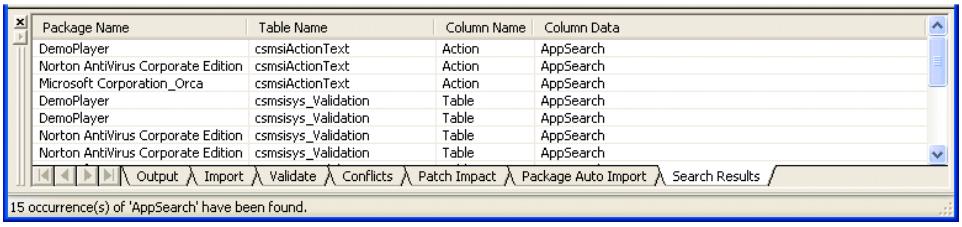
6. If you selected a table from the **Look in Table** list, all of the columns in that table are listed. Select the column that you would like to search, or select **<All Columns>**.

7. If you want to search for a partial match rather than an exact match, select the **Partial Match** option.

   - **If this option is not selected**, Application Manager will search for an exact match of the text you entered in the **Find What** text box. The search will be case sensitive.
   - **If this option is selected**, then Application Manager will use appropriate wild card characters so that a partial data match is performed. The search will be case insensitive

8. Click **Find** to initiate the search.

   The **Find** dialog box will close, and the data that is found is displayed in the **Search Results** tab of the Output Window, in the following format:



   If you double click on the data displayed in the Output Window, Application Manager will navigate to the appropriate record in the **Tables View**, and that record will be highlighted in red.

Disconnecting from an Application Catalog

To disconnect from the currently open Application Catalog, select **Disconnect** from the **Catalog** menu. You can disconnect from an Application Catalog while you are in ConflictSolver, Application Manager, PackageExpert, and also when you are in the AdminStudio Interface.

When you have disconnected from an Application Catalog in ConflictSolver or Application Manager, the following message appears:

```
ConflictSolver requires a connection to the AdminStudio Application Catalog. Select Connect from the
    Catalog menu to establish a connection.
```

Upgrading an Existing Application Catalog

When you attempt to open an AdminStudio 5.x to 9.x Application Catalog in AdminStudio 10.0 SP2, you are prompted to upgrade it to use the AdminStudio 10.0 SP2 schema.

Log files for the upgrade are created in the following directory:

AdminStudio Shared Directory\ConflictSolver\Logs

Note • *Note the following regarding upgrading an existing Application Catalog:*

- *The upgrade of AdminStudio 3.0, 3.01, and 3.5 databases is not supported by AdminStudio 7.0 or later.*
- *Starting with AdminStudio 8.0, Microsoft Access databases are not supported.*
- *Starting with AdminStudio 9.01, Oracle databases are not supported.*
- *When an SQL Server Application Catalog database is upgraded, the old tables are not dropped from the Application Catalog.*

Upgrading Pre-AdminStudio 5.0 Application Catalogs

Pre-AdminStudio 5.0 Application Catalogs cannot be upgraded automatically by AdminStudio 7.0 or later. However, you can upgrade them using the Legacy Upgrade Wizard, a standalone utility that was included with AdminStudio 7.0 and 7.5. The Legacy Upgrade Wizard utility is installed in the following directory:

C:\Program Files\InstallShield\AdminStudio\7.x\Common\LegacyUpgradeWizard.exe

If you do not have a copy of AdminStudio 7.0 or 7.5 available to you, contact Technical Support.

Organizing Your Application Catalog Using Groups

Within Application Manager and ConflictSolver, you can create groups to organize your applications, patches, and OS Snapshot images in the Application Catalog. This is especially useful for organizing your Application Catalog in ways consistent with how your company is organized.

For example, you could create a group representing a certain department's base image including the proper operating system and necessary applications. When you perform conflict analysis on new packages you are integrating into your environment, you can run only the relevant comparisons—saving you the time it would take to run the analysis against all packages in the Application Catalog, or the effort of manually determining the set of packages against which you want to run the analysis each time.

Tasks relating to groups include:

- Adding Groups
- Organizing Applications in Application Manager
- Deleting Application Manager Groups
- Editing Group Properties
- Copying Packages to Multiple Groups
- Moving Applications, OS Snapshots, and Groups

Adding Groups

To add additional groups or subgroups to an Application Catalog, perform the following steps.

Task: ***To add a group to Application Manager:***

1. In the Application Manager tree, right-click on the group to which the new group should belong and select **New Group**.
2. Provide a name for the new group.
3. Press **Enter**.

Organizing Applications in Application Manager

To move applications or groups into different groups, perform the following steps:

Task: ***To organize individual packages and groups of packages in Application Manager:***

1. In the Application Manager tree, select an application or group you want to move.
2. Drag the application or group onto a new group.

Note • *The following rules apply to drag and drop operations in the Application Manager Product View:*

- *You cannot drop a node on itself.*
- *You cannot drop a node on its parent. It is already a child of the parent.*
- *You cannot drop a group on its child groups.*

Deleting Application Manager Groups

To delete a group from an Application Catalog, perform the following steps:

Task: ***To delete a Application Manager group:***

1. Right-click the group you want to delete in the Application Manager tree and select **Delete** from the context menu.
2. From the resulting message box, confirm the deletion.

Caution • *When you delete a group, all subgroups and applications within that group are also removed from the Application Manager.*

Editing Group Properties

For each group, you can modify its name, and add a description or other comments. To edit this information, perform the following steps.

Task: ***To edit group properties:***

1. Right-click on the group in the Application Manager tree and select **Properties** from the context menu. The **Group Properties** dialog box opens, displaying the group **Name**, **Description**, and **Comments**.
2. Make any desired edits.
3. Click **OK**.

Copying Packages to Multiple Groups

More than one copy of the same package can exist in an Application Catalog. There are two methods of doing this:

- **Importing**—You can use the Import Wizard to import the same package into two different groups. Each time you import the package, Application Manager will create a new Application node in that group (if one does not already exist) to store that package.
- **Manually creating a new Application and then copying**—If you want to make a copy of an existing package in a new group in Application Manager, you first need to create a new Application node in the new group, and then you can cut and paste the package into that Application node.

Below are instructions for manually copying a package into multiple groups.

Task: ***To copy a package to multiple groups:***

1. Select the package that you want to copy and select **Copy** from the context menu (or press Ctrl+C).
2. Select the group where you want to place the copy of the package.
3. From the context menu, select **New Application**. A new application named `New Application` is created.
4. Rename the new application to match the name of the package you are copying.
5. Select the new, renamed Application node and select **Paste** from the context menu (or press Ctrl+V). The package is copied into the selected Application node.

Moving Applications, OS Snapshots, and Groups

You can move OS Snapshots, applications, or groups into other groups.

Task: ***To move an application, OS Snapshot, or group:***

1. Select the application, OS Snapshot, or group you want to move in the Application Manager tree.
2. Drag the item into a new group.

Deleting Packages

You can delete packages that have been imported into the Application Catalog.

Task: ***To delete a package from Application Manager:***

1. Right-click on the package in the **Application Manager Product View**, point to **Delete** on the context menu, and click **Package** (to delete the package from the selected Application) or **Package from all Applications** (to delete the package from all Applications in the Application Catalog).
2. Confirm the deletion.

Importing Data

You can use the Import Wizard to import multiple application deployment types either one at a time, all of the packages in a directory, or one or multiple packages directly from Microsoft System Center Configuration Manager.

- Data Types Supported By the Import Wizard
- Package Sources Supported by the Import Wizard

Data Types Supported By the Import Wizard

You can use the Import Wizard to import the following types of data into the Application Catalog:

Table 6-3 • Data Types Supported By the Import Wizard

Data Type	Description
Windows Installer Packages, Transforms, and Patches	You use the Application Manager Import Wizard to import Windows Installer packages (`.msi`) with any associated transforms (`.mst`) and patches (`.msp`) into the Application Catalog. See the following topics: • Importing a Single Windows Installer or Virtual Package • Importing a Directory of Windows Installer and/or App-V Packages • Importing Windows Installer and/or App-V Packages From Microsoft Configuration Manager
Microsoft App-V Package	You have several available options when importing Microsoft App-V packages into the Application Catalog. See the following topics: • Importing a Single Windows Installer or Virtual Package • Importing a Directory of Windows Installer and/or App-V Packages • Importing Windows Installer and/or App-V Packages From Microsoft Configuration Manager **App-V Packages Created By AdminStudio** Also, for App-V packages created by AdminStudio, you have several additional options: • **During the import of the Windows Installer package**—During the import of a Windows Installer package, if the Import Wizard searches for and finds a virtual package created by AdminStudio in a subdirectory of the directory containing that Windows Installer package, you are prompted to also import that virtual package. See Importing a Virtual Package During the Import of its Source Windows Installer Package. • **After the import of the Windows Installer package**—If you import an AdminStudio-created virtual package into an Application Catalog that already contains the virtual package's source Windows Installer package, you are prompted to associate the virtual package with the Windows Installer package. See Importing a Virtual Package After the Import of its Source Windows Installer Package. ***Note*** • *If you use the Import Wizard to import an App-V package directly into the Application Catalog, you can then use the* ***Associate Package*** *function to associate that App-V package with its source Windows Installer package. See Importing a Virtual Package Without Its Source Windows Installer Package and Manually Associating a Virtual Package with a Windows Installer Package.*

Table 6-3 • Data Types Supported By the Import Wizard (cont.)

Data Type	Description
Citrix XenApp and VMware ThinApp Packages	You have several options when importing VMware ThinApp and Citrix XenApp virtual packages into the Application Catalog: • **During the import of the Windows Installer package**—During the import of a Windows Installer package, if the Import Wizard searches for and finds a virtual package created by AdminStudio, you are prompted to also import that virtual package. See Importing a Virtual Package During the Import of its Source Windows Installer Package. • **After the import of the Windows Installer package**—If you import an AdminStudio-created virtual package into an Application Catalog that already contains the virtual package's source Windows Installer package, you are prompted to associate the virtual package with the Windows Installer package. See Importing a Virtual Package After the Import of its Source Windows Installer Package. • **By itself**—You can use the Import Wizard to import a virtual package directly into the Application Catalog. Then you can use the **Associate Package** function to associate the virtual package with its source package. See Importing a Virtual Package Without Its Source Windows Installer Package and Manually Associating a Virtual Package with a Windows Installer Package.
Ad-Hoc Import	You can also import transform files and patch files after the Windows Installer package that they are associated with has already been imported into the Application Catalog. This is referred to as an ad-hoc import. The imported transform or patch file is then stored in the Software Repository along with the rest of the imported package's associated files. See Using the Software Repository and Performing an Ad-Hoc Import of Transform Files or Patch Files. ***Note*** • *You can only perform an ad-hoc import if you are connected to a Software Repository-enabled Application Catalog.*
Merge Modules	A merge module (`.msm`) is a package containing all of the logic and files needed to install distinct pieces of application functionality such as run-time `.dll` files and virtual machines. Merge modules are built once and can be added to any installation project. Merge modules, which are imported into the Application Catalog using the Import Wizard, are used primarily during conflict detection. For optimal performance, Merge modules should be imported into an Application Catalog database prior to importing Windows Installer packages. This ensures that conflicts resulting from not using available merge modules are correctly identified. For more information, see Importing Merge Modules.

Table 6-3 • Data Types Supported By the Import Wizard (cont.)

Data Type	Description
Additional Data	Additionally, you can use the Import Wizard to import OS snapshots (`.osc`), Marimba Native Channel Packager (`.ncp`) files and other non-MSI based setup programs. See Importing OS Snapshots, Importing Marimba NCP Files, and Importing Other Setup Types.

Package Sources Supported by the Import Wizard

You specify the source of the package(s) that you want to import by making a selection on the **Select Package Source** panel of the Import Wizard:

Table 6-4 • Package Sources Supported by the Import Wizard

Source	Description
Browse Local Machine or Network for Package	Select to import a single installation package into the Application Catalog. If you select this option and click **Next**, the **File Selection** panel opens, prompting you to browse for the file.
Browse Local Machine or Network for Directory of Packages	Select to import a directory of packages (containing both Windows Installer and App-V packages, if desired) into the Application Catalog. If you select this option and click **Next**, the **Folder Selection** panel opens, prompting you to select the directory containing the packages you want to import.
Microsoft System Center Configuration Manager	Select to import applications from a Microsoft System Center Configuration Manager server. If you select this option and click **Next**, the **Connect to a MIcrosoft Configuration Manager Server** panel opens, prompting you to connect to a server. Next, the **Select Packages** panel opens, listing all of the packages in the connected Configuration Manager server and prompting you to select the packages you want to import.

Importing a Single Windows Installer or Virtual Package

You can import a Windows Installer package with all of its associated transform files and patches into the Application Catalog at the same time. The transform files and patches are immediately associated with the Windows Installer package. You can also import a single Microsoft App-V, Citrix XenApp, or VMware ThinApp package.

Tasks associated with importing Windows Installer packages, transforms, and patches include:

- Importing a Single Package
- Using Duplicate Package Identifiers
- Performing an Ad-Hoc Import of Transform Files or Patch Files

Once you import these files, they become available for conflict identification.

Note • *Although you can identify conflicts between an external package and packages already in the Application Catalog, it is strongly recommended you import the package first and perform conflict identification after import. Conflict information is only persisted for the last conflict identification performed between an external package and packages in the Application Catalog, and performance is better when performing conflict identification between internal packages. See Conflict Persistence for more information.*

Importing a Single Package

You can use the Import Wizard to import a single Windows Installer or virtual package. For a Windows Installer package, you can also import all of its associated transform and patch files at the same time.

Note • *Application Manager stores imported Windows Installer and Microsoft App-V packages for further conflict analysis. When you import a package into an Application Catalog, you can automatically run validation and conflict identification features, if configured through the ConflictSolver and Application Manager* ***Options*** *dialog box.*

Task: ***To import a single Windows Installer or virtual package into an Application Catalog:***

1. On the **Catalog** menu, click **Import Package**. The **Import Wizard Welcome** panel opens.
2. Click **Next**. The **Select Package Source** panel opens.
3. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
4. Click **Browse** and select the package that you want to import.
5. If you want to add the selected package to the Software Repository, select **Add the file(s) to the Software Repository** option. After you select this option, if AdminStudio determines that there are other files in that directory that could be associated with the selected file, the **Additional Dependencies of the Import File** list appears, enabling you to select any additional files to be managed by the Software Repository.

6. When you are finished making selections on the **File Selection** panel, click **Next**. The **MST Source Information** panel opens, where you may optionally select any additional transform (`.mst`) files to be imported along with the Windows Installer package.

   Note • *When the **MST Source Information** panel opens, all of the `.mst` files that are in the same directory as the Windows Installer file you are importing are automatically listed in the **Transform Files (.mst)** list, but only those `.mst` files that AdminStudio determines are probably applicable to this Windows Installer package are selected to be included in the import. If you do not want to import a selected `.mst` file, clear the selection.*

7. Should the Windows Installer package require transforms, click the **New** button ( ) in the **Transforms** area to browse to the location of the transform. If the package requires multiple transforms, you can repeat the procedure as necessary. The order in which transforms are applied can be changed by selecting a transform and clicking the **Move Up** and **Move Down** buttons. If you need to delete a transform you have added, select the transform and click the **Delete** button.

8. When you have finished adding transforms, click **Next**. The **MSP Source Information** panel opens, where you may optionally select any additional patch (`.msp` or `.exe`) files to be imported along with the Windows Installer package.

9. Should the package require patches, click the **New** button ( ) in the Patches area to browse to the location of the patch. If the package requires multiple patches, you can repeat the procedure as necessary. The order in which patches are applied can be changed by selecting a patch and clicking the **Move Up** and **Move Down** buttons. If you need to delete a patch you have added, select the patch and click the **Delete** button.

10. When you have finished adding patches, click **Next**. If the package that you are importing includes some custom tables, the **Import Options** panel opens, prompting you to specify whether you want to import these tables.

11. To add one of the listed tables to the **Ignore Tables** list on the **Import** tab of the ConflictSolver and Application Manager **Options** dialog box (so that it will not be imported into the Application Catalog during any import), select the table name and click **Add to Ignore List**.

12. Click Next. The **Destination Group** panel opens.

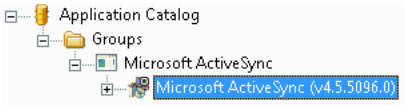
    Note • *If you opened the Import Wizard by selecting a Group in the Application Catalog and then selecting **Import Package** from the context menu, the package will be imported into the selected group, and this panel will not appear.*

13. Select one or more destination groups into which your package will be imported. You can also click **New** to create a new Group. When you click **Next**, the **Summary** panel opens.

14. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Finish** to start the import.

    A report of the import process appears on the **Import** tab of the **Output** window. Depending on whether options have been set from the **General** tab of the ConflictSolver and Application Manager **Options** dialog box (available from the **Tools** menu), validation and conflict checking may be performed during import.

The package appears under an **Application** node in the Application Manager tree:



About Importing Windows Installer Packages

Regarding importing Windows Installer packages, note the following:

- If you specify an `update.exe` patch file that was created by Developer/DevStudio/InstallShield Editor, Application Manager will extract the `.msp` file in the Temp folder and then perform the import.
- For patches to be applied to an MSI package, it is necessary to perform an Administrative install of the MSI package and then perform an Administrative install of each patch package one by one. This way, the content of each patch package is appended to the MSI package at the Administrative install location.
- You can also import a Windows Installer package by right-clicking the group into which you want to import the package and selecting **Import Package** from the context menu. The **MSI Source Information** panel is displayed in the Import Wizard, and you can proceed with the steps above. However, since you have already specified the group into which you want to import the package, the **Destination Group** panel does not appear.
- Although not required, it is recommended that you import only Windows Installer packages that have passed the Full MSI Validation Suite validation into the Application Catalog.

About Windows Installer Packages (.msi)

Application Manager supports the import of Windows Installer packages (`.msi`). A Windows Installer package contains all of the information that the Windows Installer requires to install or uninstall an application or product and to run the setup user interface. The `.msi` file can also contain one or more transform files (`.mst`) and one or more patches (`.msp`).

A Windows Installer package is organized around the concepts of components and features:

- A feature is a part of the application's total functionality that a user may decide to install independently.
- A component is a piece of the application or product to be installed.

The Windows Installer always installs or removes a component from a user's computer as a coherent piece. Components are usually hidden from the user. When a user selects a feature for installation, the installer determines which components must be installed to provide that feature.

Performing an Ad-Hoc Import of Transform Files or Patch Files

Edition *• You can only perform an ad-hoc import if you are connected to a Software Repository-enabled Application Catalog. The Software Repository feature is included in AdminStudio Enterprise Edition.*

In a large enterprise, transform and patch files are created frequently to modify packages for different situations in their organization. For example, to localize a software application, a transform file is often created for each language supported by that application. Systems Administrators in these large enterprises have to keep track of which software package each file is supposed to be applied to.

You cannot tell which package a transform or patch file is associated with by just looking at the file. However, you can use the Import Wizard to perform an **ad-hoc import** of these loose transform and patch files into your Application Catalog. Then, Application Manager will:

- Determine which of the packages in the Application Catalog the transform or patch file is associated with, and
- Store the files in the Software Repository along with that package's other associated files.

Note *• You can only perform an ad-hoc import if you are connected to a Software Repository-enabled Application Catalog. See Using the Software Repository for more information.*

Note *• To perform an ad-hoc import of a transform file or patch file, the package that the file is associated with must exist in the Application Catalog. If no associated package exists, the file cannot be imported.*

Task: ***To perform an ad-hoc import of a transform or patch file:***

1. Open **Application Manager**.
2. Connect to a Software Repository-enabled Application Catalog.
3. On the **Catalog** menu, click **Import Package**. The **Welcome** panel opens.
4. Click **Next**. The **File Selection** panel opens.
5. Select the transform or patch file that you want to import.
6. Select the **Add the file(s) to the Software Repository** option.

   After you select this option, the **Additional Dependencies of the Import File** list appears, enabling you to select any dependencies of the file you are importing. For example, a dependency might be a file that the transform file references that is stored externally in a separate CAB file.
7. Select dependencies, if desired, and click **Next**.
8. The **Target Package Information** panel opens and lists the package that Application Manager has determined is associated with this file.

Note • *For transforms and patches, if Application Manager determines that there are no packages associated with this file, you cannot proceed with the import.*

9. Select the file's associated package.
10. If you want to reimport a new version of this package along with the file you are importing, select the **Reimport new version of the package with this file change** option. If you do not select this option, the package will not be reimported.
11. Click **Next**. The **Summary** panel opens, listing the **Adhoc file to import**, the **Destination Packages for the file**, and whether you chose to reimport the package.
12. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Finish** to start the import. Progress messages appear in the **Output Window**.

About Transforms (.mst)

Application Manager supports the import of Windows Installer packages (`.msi`) with associated transforms (`.mst`). A transform is a collection of changes applied to an installation. By applying a transform to a base installation package, the installer can add or replace data in the installation database. The installer can only apply transforms during an installation.

The installer registers a list of transforms required by the product during the installation. The installer must apply these transforms to the product's installation package when configuring or installing the product.

A transform can modify information that is in any persistent table in the installer database. A transform can also add or remove persistent tables in the installer database. Transforms cannot modify any part of an installation package that is not in a database table, such as information in the summary information stream, information in substorages, information in nested installations, or files in embedded cabinets.

About Patches (.msp)

AdminStudio supports the import of Windows Installer packages (`.msi`) with associated patches (`.msp`). A Windows Installer patch (`.msp` file) is a file used to deliver updates to Windows Installer applications. A patch is a self-contained package that contains all the information required to update an application.

A patch package contains the actual updates to the application and describes which versions of the application can receive the patch. A patch package does not include a database like a regular installation package (`.msi` file). Patches contain at minimum two database transforms. One transform updates the information in the installation database of the application. The other transform adds information that the installer uses for patching files.

Importing a Directory of Windows Installer and/or App-V Packages

You can import a directory of packages (containing both Windows Installer and App-V packages, if desired) into the Application Catalog.

Task: ***To import a directory of Windows Installer and/or App-V packages:***

1. Open **Application Manager**.
2. On the **Catalog** menu, click **Import Package**. The **Import Wizard Welcome** panel opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse a Local Machine or Network for Directory of Packages** and click **Next**. The **Folder Selection** panel opens, prompting you to select the directory containing the packages you want to import.
5. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
6. Click **Browse** and select the directory containing the packages that you want to import.
7. To import the Windows Installer packages in the selected directory, select the **Import Windows Installer packages (.msi)** option.
8. To import the transform files that are located in the same directory as an `.msi` package, select the **Apply transform files (.mst) located in the same folder as an MSI file** option.
9. To import the Microsoft App-V (.sft) packages in the selected directory, select the **Import App-V packages (.sft)** option.
10. If you want to add these package to the Software Repository, select **Add the file(s) to the Software Repository** option. After you select this option, if AdminStudio determines that there are other files in that directory that could be associated with the selected file, the **Additional Dependencies of the Import File** list appears, enabling you to select any additional files to be managed by the Software Repository.
11. Click **Next**. The **Summary** panel opens.
12. Click **Finish**. Packages are imported into the Application Catalog. For information on how the Import Wizard decides which packages to import, see Import Wizard's Selection Rules When Importing Packages from a Directory.

Importing Windows Installer and/or App-V Packages From Microsoft Configuration Manager

You can use the Import Wizard to import Windows Installer and Microsoft App-V packages from a Microsoft System Center Configuration Manager server into the Application Catalog.

Task: ***To import Windows Installer and/or App-V packages from Microsoft Configuration Manager:***

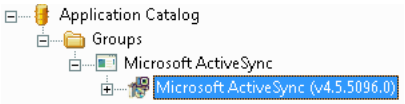
1. Open **Application Manager**.
2. On the **Catalog** menu, click **Import Package**. The **Import Wizard Welcome** panel opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse Microsoft System Center Configuration Manager** and click **Next**. The **Connect to a Microsoft Configuration Manager Server** panel opens.
5. Specify the connection information for a Microsoft Configuration Manager Server:

Option	Description
Server	Enter the name of the Microsoft Configuration Manager Server that you want to connect to.
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server login identification to log into this server. Then enter the appropriate **User name** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Microsoft Configuration Manager Server. ***Note*** • *After you successfully connect to a Microsoft Configuration Manager Server, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*

6. Click **Next**. The **Select Packages** panel opens, listing all of the packages in the connected Configuration Manager server and prompting you to select the packages you want to import.
7. Select the packages that you want to import and click **Next**. The **Summary** panel opens.
8. Click **Finish** to begin the import. The selected packages are imported into the Application Catalog.

   A report of the import process appears on the **Import** tab of the **Output** window. Depending on whether options have been set from the **General** tab of the ConflictSolver and Application Manager **Options** dialog box (available from the **Tools** menu), validation and conflict checking may be performed during import.

When the import is complete, the packages each appear under an **Application** node in the Application Manager tree:



Importing Virtual Packages

Edition • *Support for importing Microsoft App-V applications, ThinApp applications, and Citrix profiles into the Application Catalog is included in the AdminStudio Virtualization Pack.*

You can import Microsoft App-V, VMware ThinApp, and Citrix virtual packages into the Application Catalog and associate them with their source Windows Installer package.

Virtual packages are self-contained entities which ordinarily cannot be modified after they are created. By associating a virtual package with the Windows Installer package which originated it, you have the convenience of being able to easily locate the virtual package's originating Windows Installer package, modify the original Windows Installer package, and then regenerate the virtual package.

You have several options in how to import a virtual package:

- **By itself**—You can use the Import Wizard to import a virtual package directly into the Application Catalog. Then you can use the **Associate Package** function to associate the virtual package with its source package. See Importing a Virtual Package Without Its Source Windows Installer Package and Manually Associating a Virtual Package with a Windows Installer Package.
- **During the import of the Windows Installer package**—During the import of a Windows Installer package, if the Import Wizard searches for and finds a virtual package created by AdminStudio, you are prompted to also import that virtual package. See Importing a Virtual Package During the Import of its Source Windows Installer Package.
- **After the import of the Windows Installer package**—If you import an AdminStudio-created virtual package into an Application Catalog that already contains the virtual package's source Windows Installer package, you are prompted to associate the virtual package with the Windows Installer package. See Importing a Virtual Package After the Import of its Source Windows Installer Package.

Importing a Virtual Package Without Its Source Windows Installer Package

You can import a virtual package into the Application Catalog even if you have not yet imported its source Windows Installer package. If you then import the virtual package's source Windows Installer package at a later date, you can use the **Associate Package** function in Application Manager to manually associate them.

Task: ***To import a virtual package by itself:***

1. Open Application Manager.
2. On the **Catalog** menu, select **Import Package**. The Welcome panel of the **Import Wizard** opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
5. Click **Browse** and select the virtual package that you want to import:
   - **Microsoft App-V application**—Select the App-V file with an `.osd` extension, such as `MyApp.osd`.
   - **VMware ThinApp application**—Select the ThinApp file with an `.exe` extension, such as `MyApp.exe`.
   - **Citrix XenApp profile**—Select the Citrix XenApp file with a `.profile` extension, such as `MyApp.profile`.
6. Click **Next**. The **Summary** panel opens.
7. Click **Finish** to begin the import. The virtual package is imported and will be listed under its own **Application** node in the tree.

Importing a Virtual Package During the Import of its Source Windows Installer Package

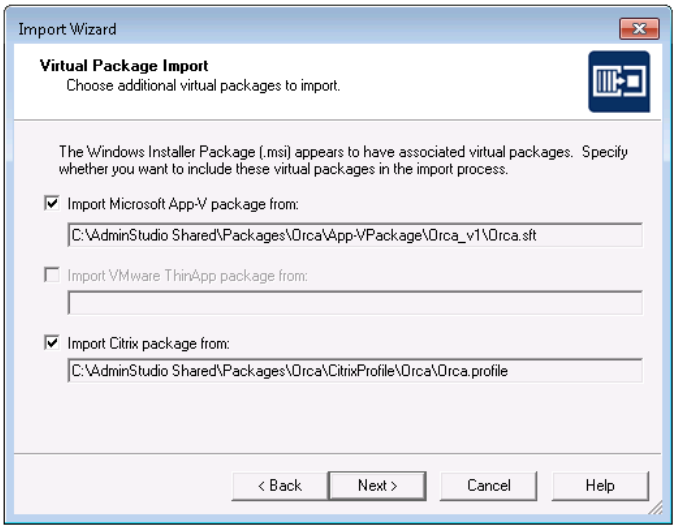
When you are importing a Windows Installer package, if the Import Wizard finds a virtual package in a subdirectory of the directory containing that Windows Installer package, you are prompted to also import that virtual package.

Task: ***To import a virtual package during the import of a Windows Installer package:***

1. Open Application Manager.
2. On the **Catalog** menu, select **Import Package**. The Welcome panel of the **Import Wizard** opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
5. Click **Browse** and select the Windows Installer package that has associated virtual packages that you want to import.

Important • *The Import Wizard will only prompt you to import a Windows Installer package's associated virtual packages if those virtual packages are in a subdirectory of the directory containing the Windows Installer package.*

6. Click **Next**. The **MST Source Information** panel opens.
7. Include any desired transform files, as described in Importing a Single Package, and click **Next**. The **MSP Source Information** panel opens.
8. Include any desired patch files, as described in Importing a Single Package, and click **Next**. The **Virtual Package Import** panel opens.



Note • *This panel only opens if the Import Wizard finds a virtual package in a subdirectory of the directory containing the Windows Installer package you have selected to import.*

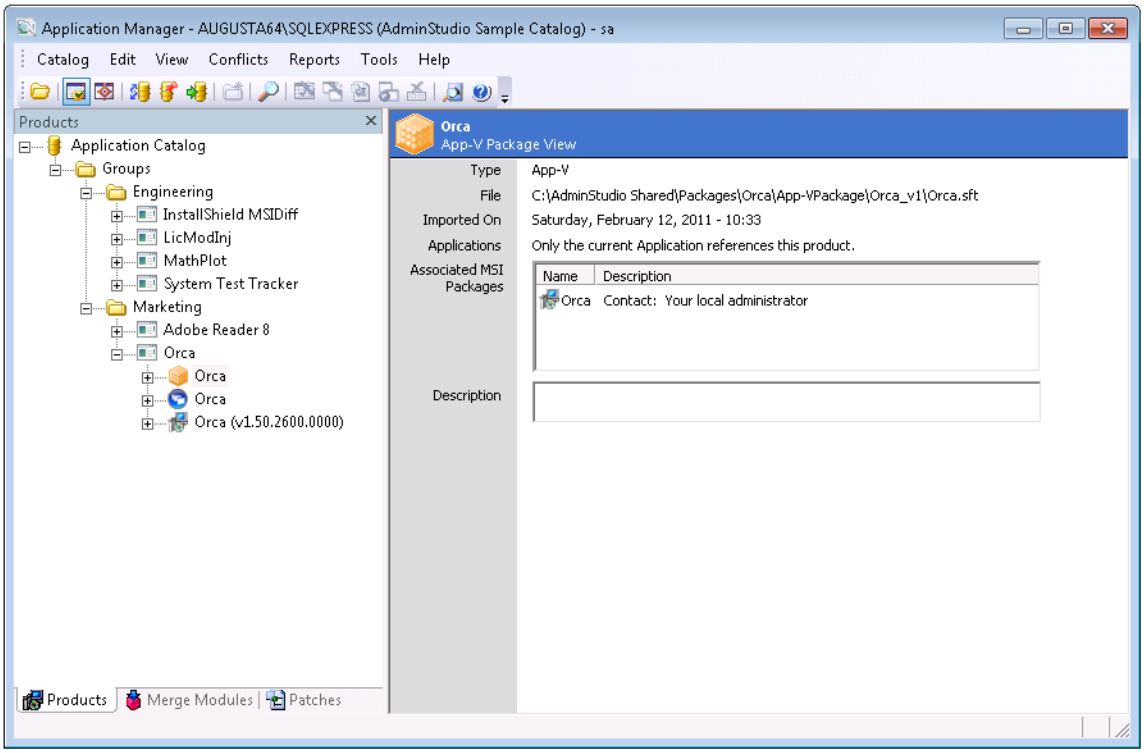
If a Windows Installer package's virtual package is not in that location, you can use the Import Wizard to import the virtual package separately, as described in Importing a Virtual Package After the Import of its Source Windows Installer Package, and then manually associate the two packages in Application Manager, as described in Manually Associating a Virtual Package with a Windows Installer Package.

9. Select the virtual package(s) you want to import and click **Next**. The **Destination Group** panel opens.

Note • *If you opened the Import Wizard by selecting a group folder and then selecting **Import** from the context menu, the **Destination Group** panel will not open.*

10. Select the group or groups where you want to import this package, as described in Importing a Single Package, and click **Next**. The **Summary** panel opens.

11. Click **Finish** to begin the import. The packages are imported. When import is complete, both the Windows Installer and the virtual package(s) are listed in the Application Catalog under the same Application node in the tree.



Importing a Virtual Package After the Import of its Source Windows Installer Package

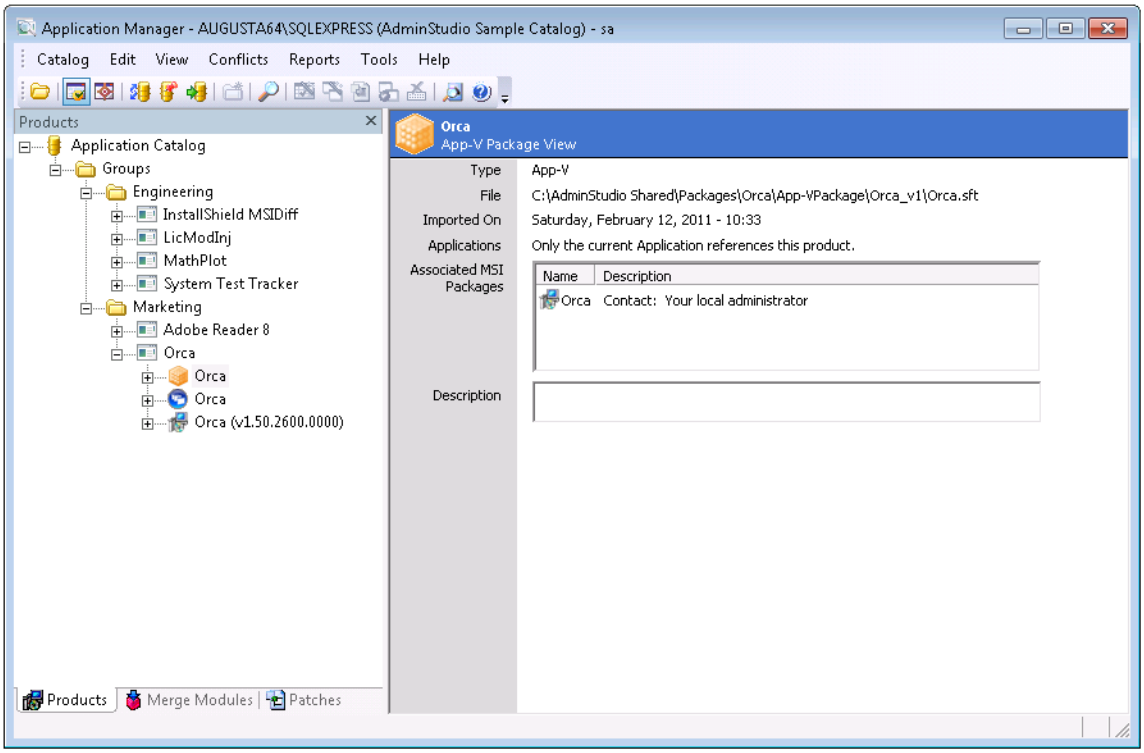
During the import of an AdminStudio-created virtual package into an Application Catalog that already contains the virtual package's source Windows Installer package, the **Associate Virtual to MSI Packages** panel opens and you are prompted to associate the virtual package with the Windows Installer package.

If this source Windows Installer package has been imported into more than one location in the Application Catalog, multiple packages are listed.

Important • *During virtual package conversion, AdminStudio 9.0 and later creates a file, metadata.ami, that contains metadata that identifies the original Windows Installer package that was used to create the virtual package. If a virtual package was created by a method other than the AdminStudio 9.x conversion process, this metadata file will not have been created. Therefore, when importing this type of virtual package, the Import Wizard will not prompt you to associate it with a Windows Installer package in the Application Catalog. Instead, you can perform the steps in Manually Associating a Virtual Package with a Windows Installer Package.*

Task: ***To import a virtual package after the import of its source Windows Installer package:***

1. Open Application Manager.
2. On the **Catalog** menu, select **Import Package**. The Welcome panel of the **Import Wizard** opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
5. Click **Browse** and select the virtual package that you want to import:
   - **Microsoft App-V application**—Select the App-V file with an `.osd` extension, such as `MyApp.osd`.
   - **VMware ThinApp application**—Select the ThinApp file with an `.exe` extension, such as `MyApp.exe`.
   - **Citrix profile**—Select the Citrix file with a `.profile` extension, such as `MyApp.profile`.
6. Click **Next**. The **Associate Virtual to MSI Packages** panel opens, prompting you to associate this virtual package with its source Windows Installer package.
7. Select the Windows Installer package and click **Next**. The **Summary** panel opens.
8. Click **Finish** to begin the import. The virtual package is imported. When import is complete, both the Windows Installer and the virtual package is listed in the Application Catalog under the same Application node in the tree.



Manually Associating a Virtual Package with a Windows Installer Package

You can use the Import Wizard to import a virtual package directly into the Application Catalog even if you have not yet imported its source Windows Installer package. If you then import the virtual package's source Windows Installer package at a later date, you can use the **Associate Package** function in Application Manager to manually associate the virtual package with its source package.

Caution • *After you have imported a virtual package into the Application Catalog, you are permitted to use the* ***Associate Package*** *function to associate it with* ***any*** *Windows Installer package in the Application Catalog, even one that is not its source package. Therefore, it is preferable to use the Import Wizard to import both the Windows Installer and virtual packages at the same time so that AdminStudio can create the proper associations.*

Task: ***To manually associate a virtual package with a Windows Installer package:***

1. Open Application Manager
2. Click on the virtual package you want to make an association for, and select **Associate Package** from the context menu. The **Virtual Package Association** dialog box opens, listing Windows Installer packages in the Application Catalog.
3. Select a Windows Installer package and click OK. The virtual package is now associated with the selected Windows Installer package.

Deleting a Virtual Package Association

You can delete a virtual package's association with a Windows Installer package by performing the following steps:

Task: ***To delete a virtual package association:***

1. Open Application Manager.
2. Right-click on a virtual package, point to **Delete** and click **Package Association** on the context menu. The **Delete Virtual Package Association** dialog box opens, prompting you to confirm the deletion.
3. Select the association that you want to delete and click **OK**. The association is deleted.

Importing Merge Modules

For optimal performance, Merge Modules should be imported into an Application Catalog database prior to importing Windows Installer packages. This ensures that conflicts resulting from not using available merge modules are correctly identified.

Task: ***To import Merge Modules:***

1. Open **Application Manager**.
2. Open the **Merge Modules** tab.
3. Select the root Merge Module or one of the imported Merge Modules and select **Import Merge Module** from the context menu. The **MSM Source Information** panel of the **Import Wizard** opens.
4. Click the Browse ( ) button in the **Merge Modules** area and select the merge module file that you want to import.
5. To import multiple patches, you can repeat the procedure as necessary.
6. The order in which merge modules are applied can be changed by selecting a merge module in the list and clicking the Move Up and Move Down arrows.
7. If you need to delete a merge module you have added, clear its check box.
8. If you want to store this merge module in the Software Repository, select the **Add the file(s) to the Software Repository** option.
9. Click **Next**. The **Summary** panel opens.
10. Click **Finish** to accept these options and begin the import.

    A report of the import process appears on the **Import** tab in the Output window.

Note • *You can also import a Merge Module into the Application Catalog by selecting* ***Import Package*** *on the* ***Catalog*** *menu. However, using that method, only one Merge Module can be imported at a time.*

About Merge Modules (.msm)

Application Manager supports the import of merge modules (.msm). Merge modules are essentially simplified Windows Installer `.msi` files.

A merge module cannot be installed alone because its lacks some vital database tables that are present in an installation database. Merge modules also contain additional tables that are unique to themselves. To install the information delivered by a merge module with an application, the module must first be merged into the application's .msi file.

Importing OS Snapshots

You can import OS Snapshot (`.osc`) files into the Application Catalog to use to determine conflicts between an operating system and a package. OS Snapshot files are files representing a particular computer system's contents. To generate an OS snapshot file, use the OS Snapshot Wizard to scan a computer's operating system and record the files, INI files, shortcuts, and registry entries present.

***Caution** • The OS Snapshot Wizard saves OS Snapshot information in two files: an `.osc` file (a collection of file type information) and an associated `.nir` file (registry information). The `.nir` file must be present in the same directory as the `.osc` file in order for import to be successful.*

Task: ***To import an OS Snapshot into an Application Catalog:***

1. Open Application Manager.
2. On the **Catalog** menu, select **Import Package**. The **Welcome** panel of the Import Wizard opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
5. Select the OS Snapshot (`.osc`) file that you want to import
6. Click **Browse** and select the file that you want to import.
7. If you want to add this OS Snapshot to the Software Repository, so that it stores the OS Snapshot's associated `.nir` file, select **Add the file(s) to the Software Repository** option.
8. Click **Next**. The **Destination Group** panel opens.

   ***Note** • If you opened the Import Wizard by selecting a Group in the Application Catalog and then selecting **Import Package** from the context menu, the OS Snapshot will be imported into the selected group, and this panel will not appear.*
9. Select one or more destination groups into which your OS snapshot will be imported. Click **Next**. The **Summary** panel opens.
10. Click **Finish** to accept these options and begin the import.

A report of the import process appears on the **Import** tab in the Output window.

About OS Snapshots (.osc)

Application Manager supports the import of OS Snapshot (`.osc`) files, which are files representing a particular computer system's contents. To generate an OS snapshot file, use the OS Snapshot Wizard to scan a computer's operating system and record the files, INI files, shortcuts, and registry entries present. The Wizard saves this information in an `.osc` file (a collection of file type information) and an associated `.nir` file (registry information).

When an OS Snapshot file is imported into Application Manager, it can be used as a reference point for conflict identification. See Taking OS Snapshots for more information.

To provide maximum flexibility during the OS Snapshot process, you can use the Exclusions Editor to create an exclusion list that identifies files, INI files, shortcuts, and registry entries that the OS Snapshot Wizard should disregard during the scan. Using this list, you can eliminate unnecessary files, shortcuts, or registry entries, and reduce the time it takes to perform the OS Snapshot. See Configuring Exclusions Using the Exclusions Editor for more information.

***Caution** • OS Snapshots should only be used for comparison in Application Manager. You should never attempt to convert an OS Snapshot into an MSI package.*

Importing Marimba NCP Files

***Edition** • Import support for Marimba Native Channel Packager (.ncp) files is available in AdminStudio Enterprise Edition.*

You can import Marimba NCP files into the Application Catalog. See About Marimba Native Channel Packager Files (.ncp) for more information.

To import Marimba NCP files into an Application Catalog, perform the following steps.

Task: ***To import a Marimba NCP file into an Application Catalog:***

1. Open Application Manager.
2. On the **Catalog** menu, select **Import Package**. The **Welcome** panel of the Import Wizard opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
5. Select a **Marimba NCP File** (**.ncp**) and click **Next**. The **Destination Group** panel opens.

***Note** • If you opened the Import Wizard by selecting a Group in the Application Catalog and then selecting **Import Package** from the context menu, the NCP File will be imported into the selected group, and this panel will not appear.*

6. Select one or more destination groups into which your package will be imported and click **Next**. The **Summary** panel opens.
7. Review the information in the **Summary** panel . If you are satisfied with the import options, click **Finish** to start the import.

   A report of the import process appears on the **Import** tab of the **Output** window. The Marimba NCP file package appears under the specified group in the **Application Manager Product View**.

Caution • *The Marimba NCP file that should be imported into an Application Catalog is always named manifest.ncp. If you attempt to import another file with an .ncp extension into an Application Catalog, the import will fail.*

About Marimba Native Channel Packager Files (.ncp)

Marimba NCP (Native Channel Packager) files are used to describe software components, their versions, their underlying structures, and their dependence on other components. Marimba's NCP file is derived from an OSD (Open Software Description) file and is currently used by Marimba's Castanet family of products.

Clients, such as Marimba Castanet Tuners, can use an NCP file to learn what files constitute a software package for particular software for a particular client platform and particular user preferences. Just as an HTML file can contain links to other HTML or media files, so can an OSD file contain links to other NCP files. These linked NCP files represent sub-packages or dependencies.

The following is an example of a very simple NCP file:

```
<SOFTPKG NAME=com.Installshield.www.AdminStudio"
     VERSION="1,0,0,0">
<TITLE>Solitaire></TITLE>
<ABSTRACT>AdminStudio by InstallShield Corporation</ABSTRACT>
<LICENSE HREF=http://www.installshield.com/AdminStudio
     license.html/>
<IMPLEMENTATION>
<OS VALUE-"WinNT">
<OSVERSION VALUE-"4,0,0,0"/></OS>
<OS VALUE-"Win95">
<PROCESSOR VALUE="x86"/>
<LANGUAGE VALUE="en"/>
<CODEBASE HREF=http://www.installshield.com/AdminStudio.cab/>
</IMPLEMENTATION>
</SOFTPKG>
```

Application Manager supports the import of Marimba NCP Version 4.7.2 files into the AdminStudio Application Catalog, making them available for conflict identification and resolution. Please note the following:

- The Marimba NCP file that should be imported into Application Manager is *always* named manifest.ncp. If you attempt to import another file with an .NCP extension into Application Manager, the import will fail.
- Information from NCP files is stored in OS Snapshot specific tables within the Application Catalog.

- An NCP file can be selected as a source and a target package for Conflict detection, using ACE23 and ACE24. However, reporting, validation, and conflict resolution are not supported for the NCP package type.
- Extended attributes are supported for NCP packages.
- You can import NCP files from the command line.

Importing Other Setup Types

AdminStudio supports the import of wide variety of other (i.e., non-MSI) setup formats into the Application Catalog. See About Other Setup Types for a description of those setup types.

To import a setup in one of these formats into an Application Catalog, perform the following steps:

Caution • *Package Auto Import, Application Catalog Replication, reporting, validation, conflict detection, and thus conflict persistence are not supported by the Other Setup Types package type. However, the Merge process does support this package type. See Merging Application Catalogs Using the Merge Wizard for more information.*

Task: ***To import Other Setup Types into an Application Catalog:***

1. Open Application Manager.
2. On the **Catalog** menu, select **Import Package**. The **Welcome** panel of the Import Wizard opens.
3. Click **Next**. The **Select Package Source** panel opens.
4. Select **Browse Local Machine or Network for Package** and click **Next**. The **File Selection** panel opens.
5. Select the non-MSI package that you want to import and click **Next**. The **Additional non-MSI Import Options** panel opens, where you can enter a name for the package and the base directory for the package (where all of its installation files are located).
6. Enter a **Product Name** to identify this package.
7. Click the **Explore** button and select the base **Directory** of this package where all of its associated files can be found.
8. If you want to import all the files in the specified directory folder and all of its subfolders, select the **Include entire directory hierarchy** check box. If you want to import only the files in the root of the specified directory folder, do not select this option.
9. Click Next. The **Destination Group** panel opens.

Note • *If you opened the Import Wizard by selecting a Group in the Application Catalog and then selecting* ***Import Package*** *from the context menu, the file will be imported into the selected group, and this panel will not appear.*

10. Select one or more destination groups into which this package will be imported and click **Next**. The **Summary** panel opens.
11. Review the information in the **Summary** panel. If you are satisfied with the import options, click **Finish** to start the import.

    A report of the import process appears on the **Import** tab of the **Output** window. The Other Setup Type package appears under the specified group in the **Application Manager Product View**.

About Other Setup Types

A wide variety of other (i.e., non-MSI) setup formats exist. Application Manager supports the import of non-MSI setup types (such as InstallShield Professional or ISMP installations) into the Application Catalog. By importing these other setup formats, you allow AdminStudio to manage these setups in a manner consistent with other MSI based packages.

When importing one of these setup formats using the Import Wizard, you specify a directory or a directory hierarchy, and Application Manager will scan any `setup.exe` or `isetup.exe` files for pertinent setup information.

***Note** • Package Auto Import, Application Catalog Replication, reporting, validation, conflict detection, and thus conflict persistence are not supported by the Other Setup Types package type. However, the Merge process does support this package type. See Merging Application Catalogs Using the Merge Wizard for more information.*

Using Duplicate Package Identifiers

When you import a package into an Application Catalog database, Application Manager checks specific identifiers that are selected on the **Duplicate Package** tab of the ConflictSolver and Application Manager **Options** dialog box to determine if that package has already been imported.

- **For Windows Installer files**, the following identifiers are listed on the **Duplicate Package** tab:
  - PackageCode
  - ProductCode
  - Product Language
  - ProductVersion
  - List of Transform Files

  If none of these identifiers are selected, Application Manager will use the Product Name identifier to perform a Duplicate Package check.

- **For App-V packages**, the following identifiers are listed on the **Duplicate Package** tab:
  - PackageGUID
  - VersionGUID

  If neither of these identifiers are selected, Application Manager will use the **Product Name** identifier to perform a Duplicate Package check.

If Application Manager determines that you are attempting to import a duplicate package (based upon the selected identifiers), you are given the opportunity to enter a name of your choosing to identify the duplicate package before the package is imported. If the package is stored in the Software Repository, you can also choose to overwrite the existing version or treat it as a new version of an existing package.

Business Case for Importing a Package Multiple Times

You might encounter this situation if you are importing the same package into the same Application Catalog database multiple times, each time with a different set of transforms. One common reason why you might want to import the same package into your Application Catalog database more than once would be if you wanted to use Tuner to create custom installation SKUs of a common MSI package to distribute to different departments in your organization, each installation including certain features that are appropriate for the department and excluding certain features that are not appropriate. For example, if you were distributing a copy of Microsoft Office, you could add transforms to the Microsoft Office MSI package so that:

- Accounting's installation would include only Word and Excel
- Marketing's installation would include only Word and PowerPoint, and
- Development's installation would include only Word and Access.

Therefore, you might want to import the same package into your database more than once, each time with a different set of transformations. What happens when you import the package the second time depends upon the identifiers you selected on the **Duplicate Package** Tab. In this example:

- If you select the **List of Transform Files** and **ProductCode** identifiers on the **Duplicate Package** tab of the ConflictSolver and Application Manager **Options** dialog box, Application Manager will not identify these two packages as duplicate, even though they have the same **ProductCode**, because they have a different set of transformations. Therefore, the package will be imported with the same display name as the first package.
- If you only select the **ProductCode** identifier on the **Duplicate Package** tab of the ConflictSolver and Application Manager **Options** dialog box, Application Manager will identify the second package as a duplicate because the two packages have the same **ProductCode**.

Duplicate Product Name Conventions

When it identifies a duplicate package, by default Application Manager generates a new name for that package by pre-pending the Product Name with the Manufacturer's name and, if necessary, appending the Product Name with numbers:

- **1st Package**: PowerPoint
- **2nd Package**: Microsoft Corporation_PowerPoint

You can edit the **Duplicate Package Naming Syntax** on the **Duplicate Package** tab of the **Options** dialog box.

When Duplicate Packages are Identified During Bulk Import

If Application Manager is performing a bulk import or reimport, it still identifies duplicate packages using the user-specified criterion. Application Manager will generate a unique displayed product name and allow the import to proceed without prompting the user.

Application Manager identifies duplicate packages based upon the identifiers that you selected on the **Duplicate Package** tab of the ConflictSolver and Application Manager **Options** dialog box. See Using Duplicate Package Identifiers for more information.

Identifying Duplicate Package Identifiers

To identify duplicate package identifiers, perform the following steps.

Task: ***To identify duplicate package identifiers:***

1. Select **Options** from the **Tools** menu. The ConflictSolver and Application Manager **Options** dialog box opens
2. Click the **Duplicate Package** tab.
3. Under **Duplicate Package Identification Options**, select the identifiers that you would like to use for Windows Installer package imports by selecting one or more of the following options:
   - **PackageCode Property**— Identifier of the package the product was installed from. No two non-identical .msi files should ever have the same package code.
   - **ProductCode Property**—Unique identifier for the particular product release, represented as a string GUID, for example:

     `{12345678-1234-1234-1234-123456789012}`

   - **Product Language**—The language the installer should use for any strings in the user interface that are not authored into the database.
   - **ProductVersion**—Version of the product in string format. The format of the string is: major.minor.build.
   - **List of Transform Files**—A list of the transformations associated with this package.
   - **[None Selected]**—If you do not select any of these five identifiers, Application Manager checks the Product Name identifier to determine if a package is a duplicate.
4. Under **Duplicate Virtual Package Identification Options**, select the identifiers that you would like to use for App-V package imports by selecting one or more of the following options:
   - **PackageGUID**—Unique identifier of App-V package.
   - **VersionGUID**—Unique identifier of App-V package version.
   - **[None Selected]**—If you do not select either of these identifiers, Application Manager checks the Product Name identifier to determine if a package is a duplicate.
5. Click **OK** to save your changes and exit the **Options** dialog box
6. Proceed with the package import, as described in Importing a Single Package or Importing Virtual Packages.

Note • *If Application Manager is performing a bulk import or reimport, it still identifies duplicate packages using the user-specified criterion. Application Manager will generate a unique displayed product name and allow the import to proceed without prompting the user.*

Note • *The options that you select on the* ***Duplicate Package*** *tab of the ConflictSolver and Application Manager* ***Options*** *dialog box apply globally to all packages that you attempt to import; you cannot apply different identifiers to different packages. Also, since these options are saved in the AdminStudio Shared Directory, everyone using AdminStudio at your organization will share the same Duplicate Package options.*

Viewing Application Catalog Enhanced Reporting

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

AdminStudio makes it easy for you to view Application Catalog summary information on the Windows Installer and App-V packages in your catalog, giving you insight into the readiness of those packages for distribution.

AdminStudio uses Microsoft SQL Reporting Services to provide the **Application Readiness Dashboard** and other summary reports. You can also create your own custom reports using Microsoft SQL Reporting Services.

- Viewing the Application Readiness Dashboard
- Viewing the Application Compatibility Dashboard
- Viewing the Virtualization Readiness Dashboard
- Viewing Other Summary Reports
- Exporting a Report in PDF, Excel, or Word Format
- Creating Customizable Reports

Viewing the Application Readiness Dashboard

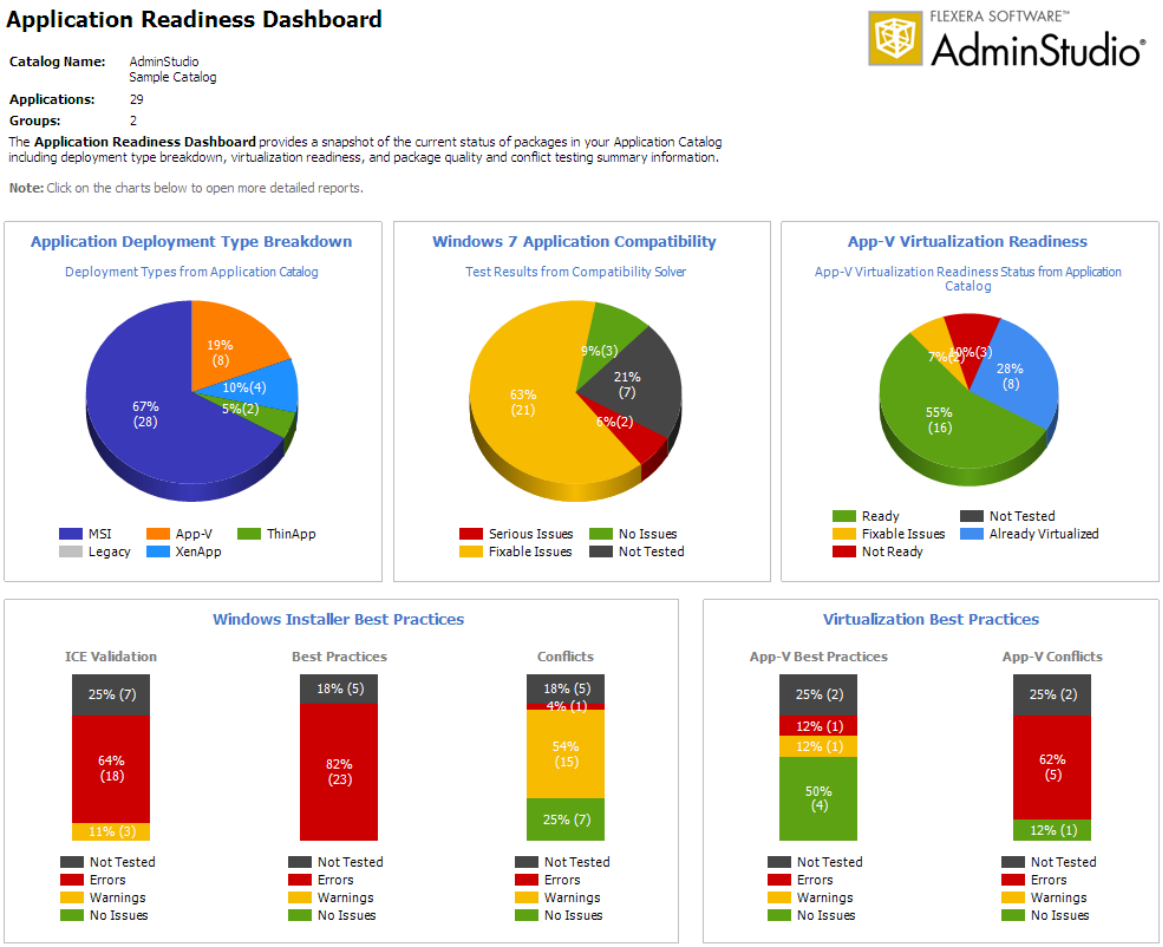
Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

AdminStudio uses Microsoft SQL Reporting Services to provide the **Application Readiness Dashboard**, a summary dashboard of Application Catalog data concerning the readiness of Windows Installer and App-V packages for distribution. This dashboard includes charts that provide a snapshot of the current status of packages in your Application Catalog including deployment type breakdown, App-V virtualization readiness, Windows 7 test results from Compatibility Solver, and the results of Windows Installer and App-V best practices testing. You can click any of the categories in these charts to open more detailed reports.

Task: ***To open the Application Readiness Dashboard:***

1. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.
   - Open the AdminStudio interface and select the **Report Center** tab.

2. Select **Enterprise Dashboard** from the **Select Report** list. The **Application Readiness Dashboard** opens.



The following information is listed at the top of the report:

Property	Description
Catalog Name	The name of the Application Catalog database that you are connected to.
Applications	The total number of applications in the Application Catalog.
Groups	The total number of groups defined in the Application Catalog.

3. Review the following charts:

- Application Deployment Type Breakdown Chart
- Windows 7 Application Compatibility Chart
- App-V Virtualization Readiness Chart
- Windows Installer Best Practices Chart

- Virtualization Best Practices Chart

Application Deployment Type Breakdown Chart

The **Application Deployment Type Breakdown** chart of the Application Readiness Dashboard lists the percentage of packages in the Application Catalog by deployment type: **MSI** (Windows Installer), **App-V**, **XenApp**, **ThinApp**, and **Legacy**.

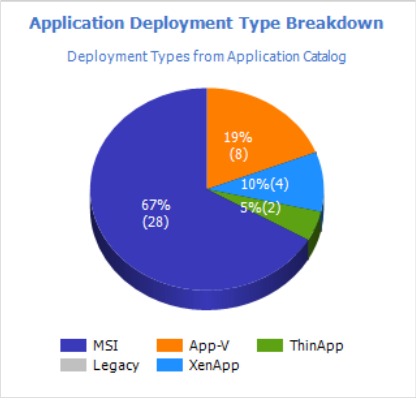
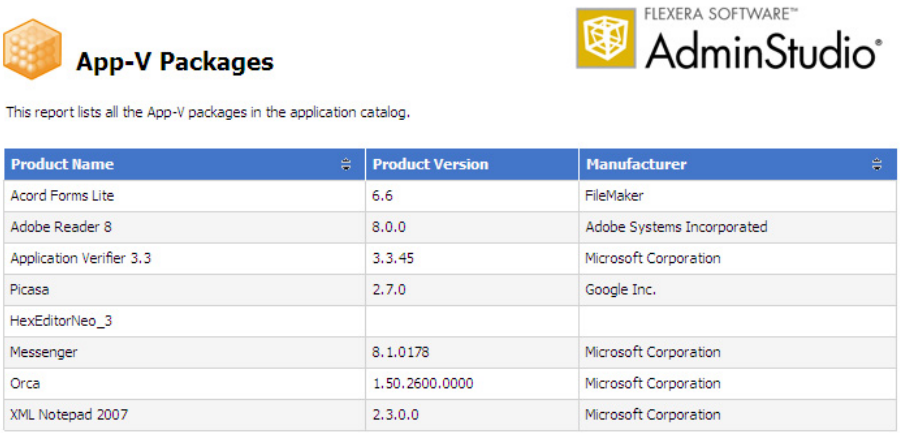


Figure 6-4: Application Deployment Type Breakdown Chart in the Application Readiness Dashboard

To view a list of packages of a specific deployment type, click on the appropriate category of the chart.



Product Name	Product Version	Manufacturer
Acord Forms Lite	6.6	FileMaker
Adobe Reader 8	8.0.0	Adobe Systems Incorporated
Application Verifier 3.3	3.3.45	Microsoft Corporation
Picasa	2.7.0	Google Inc.
HexEditorNeo_3		
Messenger	8.1.0178	Microsoft Corporation
Orca	1.50.2600.0000	Microsoft Corporation
XML Notepad 2007	2.3.0.0	Microsoft Corporation

Figure 6-5: App-V Packages Report

Windows 7 Application Compatibility Chart

Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.*

The **Windows 7 Application Compatibility** chart of the Application Readiness Dashboard lists the results of Windows 7 application compatibility tests that were performed in Compatibility Solver on packages that were loaded into the Compatibility Solver database. Packages are grouped according to the type of issues found: **Serious Issues**, **Fixable Issues**, **No Issues**, or **Not Tested**.

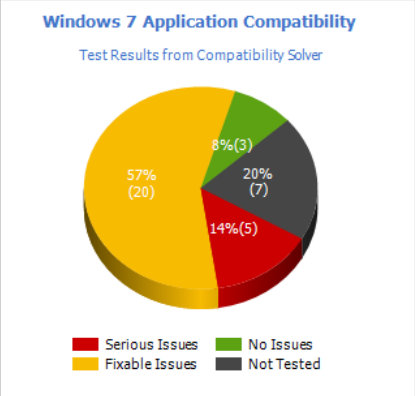


Figure 6-6: Windows 7 Application Compatibility Chart in the Application Readiness Dashboard

To view a list of packages in each group, click on the appropriate category of the chart.

Application Compatibility Results - Windows 7
Packages With Serious Issues Found

FLEXERA SOFTWARE™ AdminStudio®

Product Name	Manufacturer	Serious Issue Count	Fixable Issue Count
Messenger	Microsoft Corporation	1	44
Microsoft ActiveSync	Microsoft Corporation	1	6
PageMaker	Starbucks Corporation	1	8
Picasa	Google Inc.	5	4
Win2K	MyCompany	208	50

Figure 6-7: Packages With Serious Issues Found During Windows 7 Application Compatibility Testing

Note • *In order to display Compatibility Solver test results in AdminStudio reports, you need to specify the location of the **Compatibility Solver Database** on the **Application Catalog** tab of the AdminStudio **Options** dialog box. See Connecting to the Compatibility Solver Database.*

Note • *For information on viewing additional Compatibility Solver test results in AdminStudio, see Viewing the Application Compatibility Dashboard and Viewing Windows Installer Package Application Compatibility Issues.*

App-V Virtualization Readiness Chart

The **App-V Virtualization Readiness** chart of the Application Readiness Dashboard lists the percentage of and number of Windows Installer packages in the Application Catalog in each of the following App-V virtualization readiness categories: **Ready**, **Fixable Issues**, **Not Ready**, **Not Tested**. This chart also lists the number of App-V packages in the Application Catalog in the **Already Virtualized** category.

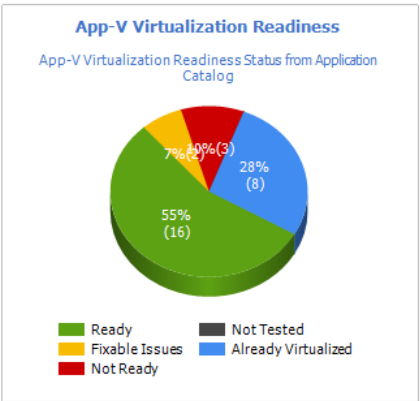


Figure 6-8: App-V Virtualization Readiness Chart of the Application Readiness Dashboard

Regarding the App-V Virtualization Readiness chart, note the following:

- **If an application contains a Windows Installer package but no App-V package**, the suitability of the Windows Installer package is evaluated and the application is listed in the **Ready**, **Fixable Issues**, **Not Ready**, or **Not Tested** category.
- **If an application contains just an App-V package or both a Windows Installer package and an App-V package**, it is listed in the **Already Virtualized** category.
- **If an application only contains a legacy package** (without a Windows Installer or an App-V package), it is ignored.

Note • *For more information, see Virtualization Readiness Categories.*

To view a list of packages in a specific category, click on the appropriate section of the chart.

Virtualization Readiness - App-V
Applications Already Virtualized

FLEXERA SOFTWARE™ AdminStudio®

This report lists applications that already have an App-V package.

Application	Product Name	Product Version	Manufacturer
Acord Forms Lite	Acord Forms Lite	6.6	FileMaker
Adobe Reader 8	Adobe Reader 8	8.0.0	Adobe Systems Incorporated
Application Verifier 3.3	Application Verifier 3.3	3.3.45	Microsoft Corporation
HexEditorNeo_3	HexEditorNeo_3		
Messenger	Messenger	8.1.0178	Microsoft Corporation
Orca	Orca	1.50.2600.0000	Microsoft Corporation
Picasa	Picasa	2.7.0	Google Inc.
XML Notepad 2007	XML Notepad 2007	2.3.0.0	Microsoft Corporation

Figure 6-9: Applications With App-V Packages

Virtualization Readiness Categories

The following table describes the criteria AdminStudio uses to categorize Windows Installer packages into each virtualization readiness status category:

Table 6-5 • Virtualization Readiness Status Categories

Status	Description
Ready	The application contains a Windows Installer package that is ready for virtualization to App-V format using the Automated Application Converter. Any issues discovered during virtualization readiness testing will be handled automatically during conversion.
Fixable Issues	The application contains a Windows Installer package that has issues that can be fixed by AdminStudio users prior to virtualization, such as if the package contains no shortcuts. If you used InstallShield Editor, for example, to fix the issue for this package, virtualization would be successful.
Not Ready	The application contains a Windows Installer package that may not be suitable for virtualization by Automated Application Converter. ***Note*** • *For information on issues that would cause AdminStudio to place a package in the* ***Not Ready*** *category, see Virtualization Suitability Issues.*
Not Tested	The application contains a Windows Installer package that has not yet been tested for virtualization readiness.
Already Virtualized	The application contains a Microsoft App-V virtual package.

Note • *Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog. To manually retest a package or group of packages for virtualization readiness, select a package, application, or group in the Application Manager tree and select* ***Run Virtualization Readiness*** *from the context menu. See also Viewing Windows Installer Package Virtualization Suitability Issues.*

Note • *To see a summary of not only the App-V virtualization readiness status, but also the Citrix XenApp and VMware ThinApp virtualization readiness status, open the* ***Virtualization Readiness Dashboard****. See Viewing the Virtualization Readiness Dashboard.*

Windows Installer Best Practices Chart

The **Windows Installer Best Practices** chart of the Application Readiness Dashboard lists the Microsoft ICE Validation status of Windows Installer packages, and the AdminStudio Best Practices and Conflict test status of Windows Installer packages.

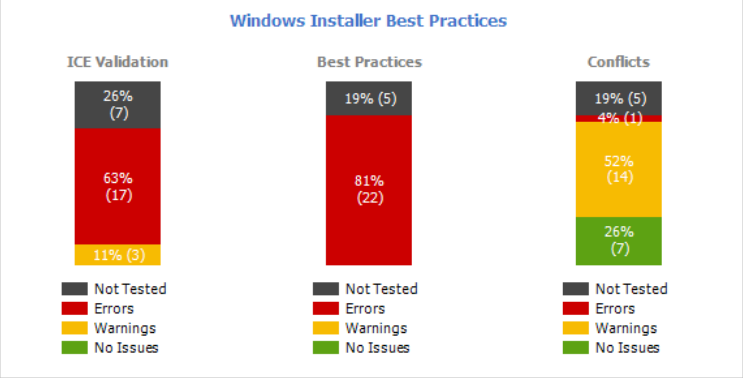


Figure 6-10: Windows Installer Best Practices Charts of the Application Readiness Dashboard

If you click on one of the categories in these charts, you see a list of packages that fall into that category.

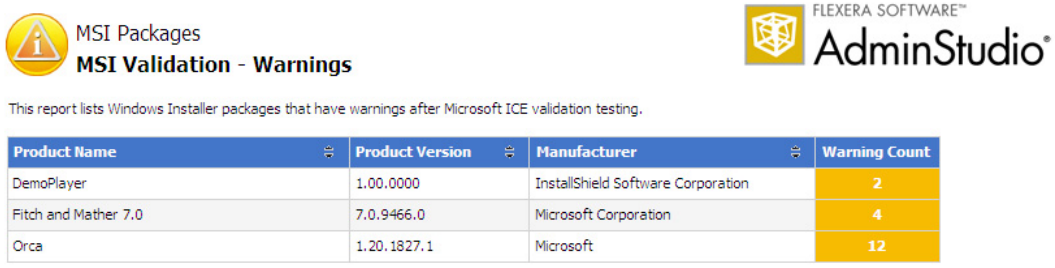
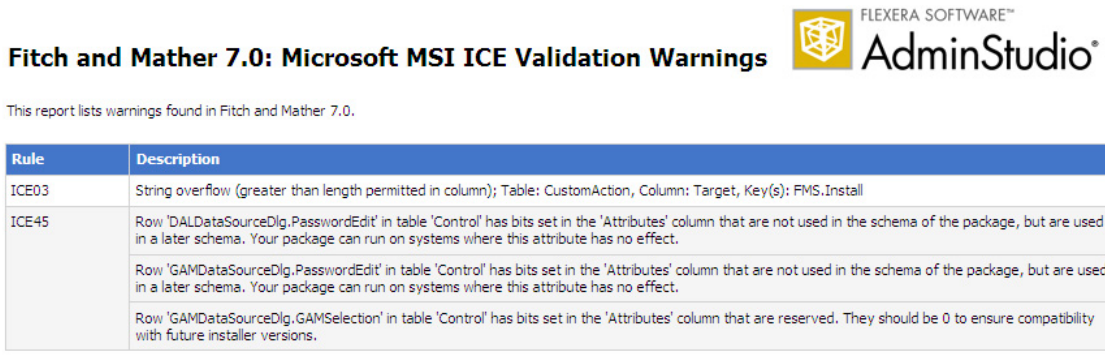


Figure 6-11: List of Packages in the “Warnings” Category of the ICE Validation Chart

You can then click on the count to see a list of the warnings or errors generated for that package.



Fitch and Mather 7.0: Microsoft MSI ICE Validation Warnings

FLEXERA SOFTWARE™ AdminStudio®

This report lists warnings found in Fitch and Mather 7.0.

Rule	Description
ICE03	String overflow (greater than length permitted in column); Table: CustomAction, Column: Target, Key(s): FMS.Install
ICE45	Row 'DALDataSourceDlg.PasswordEdit' in table 'Control' has bits set in the 'Attributes' column that are not used in the schema of the package, but are used in a later schema. Your package can run on systems where this attribute has no effect.
	Row 'GAMDataSourceDlg.PasswordEdit' in table 'Control' has bits set in the 'Attributes' column that are not used in the schema of the package, but are used in a later schema. Your package can run on systems where this attribute has no effect.
	Row 'GAMDataSourceDlg.GAMSelection' in table 'Control' has bits set in the 'Attributes' column that are reserved. They should be 0 to ensure compatibility with future installer versions.

Figure 6-12: ICE Warnings Generated for a Specific Windows Installer Package

The Windows Installer Best Practices area includes the following charts:

- ICE Validation Chart
- Best Practices Chart
- Conflicts Chart

ICE Validation Chart

The **ICE Validation** chart lists the current status of Windows Installer packages in the Application Catalog regarding Microsoft ICE validation. Validation—which can either be performed during import or by selecting a Windows Installer package in the Application Manager tree and clicking **Validate Package** from the context menu—compares the package against Microsoft ICEs (Internal Consistency Evaluators), custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards. A Windows Installer package would fall into one of the following validation categories:

Table 6-6 • Validation Categories

Category	Description
Errors	This package contains an error in database authoring that causes incorrect behavior. An example of an error would be duplicate component GUIDs, which would cause the Installer to incorrectly register components.
Warnings	This package contains an issue in database authoring that can cause incorrect behavior. Warnings can also report unexpected side-effects of database authoring. An example of a warning would be entering the same property name in two conditions that differ only by the case of letters in the name. Because the installer is case-sensitive, the installer treats these as different properties.
No Errors	Validation resulted in no errors or warnings.
Not Tested	Validation has not yet been performed on this package.

Note • *If all of a package's detected ICE errors are suppressed, the package is listed in the* ***No Errors*** *category. For more information, see Suppressing an ICE Error.*

Note • *For more information on validation, see Performing Validation and Validation View.*

Best Practices Chart

The **Best Practices** chart lists the current status of Windows Installer packages in the Application Catalog regarding Best Practice ACE rules. Best Practice ACEs are rules that internally perform checks against the structure of a Windows Installer package to enforce Microsoft best practices. Best Practice ACEs are similar to Microsoft ICEs. Both Microsoft ICEs (custom actions written by Microsoft used during validation) and Best Practice ACEs (custom actions included with ConflictSolver used during conflict analysis) are used to determine if an installation package is built according to Windows Installer standards.

When testing an installation package against a Best Practice ACE Rule, there is no target package involved. The source installation package is tested to see if it meets the standards defined in the ACEs. Best Practice testing is performed using the Conflict Wizard.

After Best Practices testing, a Windows Installer package would fall into one of the following categories:

Table 6-7 • Best Practices Rule Categories

Category	Description
Errors	Best Practices testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Best Practices testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).
No Errors	Best Practices testing resulted in no errors or warnings.
Not Tested	Best Practices testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the* ***Errors*** *category. For more information, see Resolving Conflicts, Automatically Resolving Conflicts, and Suppressing ACE Errors.*

Note • *For more information on Best Practice testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

Conflicts Chart

The **Conflicts** chart lists the current status of Windows Installer packages in the Application Catalog regarding Conflict ACE rules. Conflict ACE Rules are used to detect conflicts between two packages. You can use Conflict ACE Rules to detect conflicts between one or more source packages (Windows Installer and App-V) and one or more target packages in the Application Catalog. Conflict evaluation is done for each source package against each target package.

Conflict testing is performed using the Conflict Wizard.

After Conflict testing, a Windows Installer package would fall into one of the following categories:

Table 6-8 • Conflict Rule Categories

Category	Description
Errors	Conflict testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Conflict testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).
No Errors	Conflict testing resulted in no errors or warnings.
Not Tested	Conflict testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the* ***Errors*** *category. For more information, see Resolving Conflicts, Automatically Resolving Conflicts, and Suppressing ACE Errors.*

Note • *For more information on Conflict testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

Virtualization Best Practices Chart

The **Virtualization Best Practices** area of the Application Readiness Dashboard lists the AdminStudio Best Practices and Conflict test status of App-V packages.

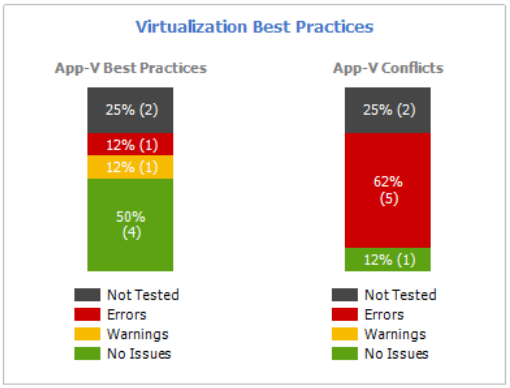


Figure 6-13: Virtualization Best Practices Charts of the Application Readiness Dashboard

If you click on one of the categories in these charts, you see a list of packages that fall into that category.

App-V Packages
AdminStudio Conflicts - Failed With Errors

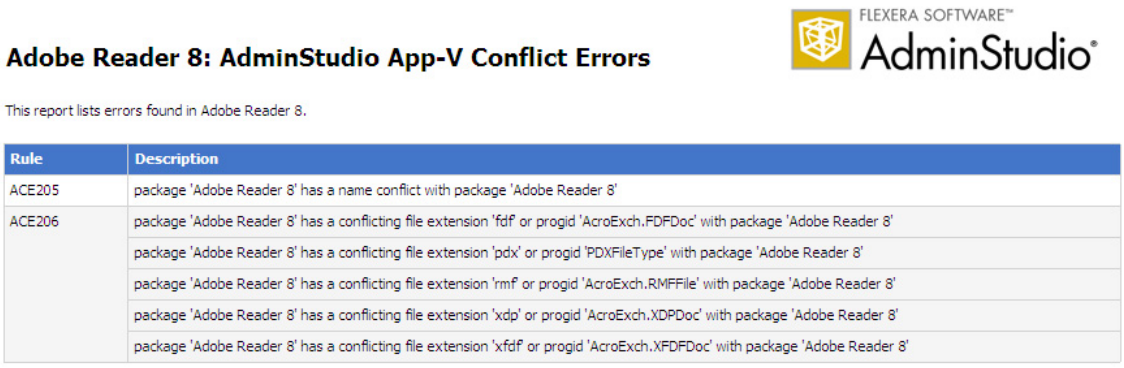
FLEXERA SOFTWARE™ AdminStudio®

This report lists App-V packages that have errors after AdminStudio App-V Conflict testing.

Product Name	Product Version	Manufacturer	Error Count	Warning Count
Acord Forms Lite	6.6	FileMaker	2	0
Adobe Reader 8	8.0.0	Adobe Systems Incorporated	6	0
Application Verifier 3.3	3.3.45	Microsoft Corporation	1	0
Messenger	8.1.0178	Microsoft Corporation	1	0
Orca	1.50.2600.0000	Microsoft Corporation	5	0

Figure 6-14: List of Packages in the "Errors" Category of the App-V Conflicts Chart

You can then click on the count to see a list of the ACE warnings or errors generated for that package.



Adobe Reader 8: AdminStudio App-V Conflict Errors

FLEXERA SOFTWARE™ AdminStudio®

This report lists errors found in Adobe Reader 8.

Rule	Description
ACE205	package 'Adobe Reader 8' has a name conflict with package 'Adobe Reader 8'
ACE206	package 'Adobe Reader 8' has a conflicting file extension 'fdf' or progid 'AcroExch.FDFDoc' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'pdx' or progid 'PDXFileType' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'rmf' or progid 'AcroExch.RMFFile' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'xdp' or progid 'AcroExch.XDPDoc' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'xfdf' or progid 'AcroExch.XFDFDoc' with package 'Adobe Reader 8'

Figure 6-15: ACE Conflict Errors Generated for a Specific App-V Package

The App-V Best Practices area includes the following charts:

- App-V Best Practices Chart
- App-V Conflicts Chart

App-V Best Practices Chart

The App-V **Best Practices** chart lists the current status of App-V packages in the Application Catalog regarding Best Practice ACE rules. Best Practice ACEs (custom actions included with ConflictSolver) are rules that internally perform checks against the structure of an App-V package to enforce best practices.

When testing an installation package against a Best Practice ACE Rule, there is no target package involved. The source installation package is tested to see if it meets the standards defined in the ACEs. Best Practice testing is performed using the Conflict Wizard.

After Best Practices testing, an App-V package would fall into one of the following categories:

Table 6-9 • Best Practices Rule Categories

Category	Description
Errors	Best Practices testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Best Practices testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).

Table 6-9 • Best Practices Rule Categories

Category	Description
No Errors	Best Practices testing resulted in no errors or warnings.
Not Tested	Best Practices testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the* ***Errors*** *category. For more information, see Resolving Conflicts, Automatically Resolving Conflicts, and Suppressing ACE Errors.*

Note • *For more information on Best Practice testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

App-V Conflicts Chart

The **Conflicts** chart lists the current status of App-V packages in the Application Catalog regarding Conflict ACE rules. Conflict ACE Rules are used to detect conflicts between two packages. You can use Conflict ACE Rules to detect conflicts between one or more source App-V packages and one or more target packages (Windows Installer and App-V) in the Application Catalog. Conflict evaluation is done for each source package against each target package.

Conflict testing is performed using the Conflict Wizard.

After Conflict testing, an App-V package would fall into one of the following categories:

Table 6-10 • Conflict Rule Categories

Category	Description
Errors	Conflict testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Conflict testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).

Table 6-10 • Conflict Rule Categories

Category	Description
No Errors	Conflict testing resulted in no errors or warnings.
Not Tested	Conflict testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the* ***Errors*** *category. For more information, see Resolving Conflicts and Suppressing ACE Errors.*

Note • *For more information on Conflict testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

Viewing the Application Compatibility Dashboard

Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.*

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

The **Application Compatibility Dashboard** displays a set of pie charts that provide a snapshot of the compatibility test results of the currently imported applications in Compatibility Solver. This dashboard not only displays Windows 7 application compatibility test results, but also Windows 64-bit, Windows Server 2008 and 2008 R2, Internet Explorer 7-8, and Internet Explorer 8 Presentation compatibility issues.

You can click on any section of the charts to open more detailed reports.

Task: ***To open the Application Compatibility Dashboard:***

1. Specify the location of the Compatibility Solver database on the **Application Catalog** tab of the AdminStudio **Options** dialog box, as described in Connecting to the Compatibility Solver Database.
2. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.
   - Open the AdminStudio interface and select the **Report Center** tab.
3. Select **Application Compatibility Dashboard** from the **Select Report** list. The **Application Compatibility Dashboard** opens.

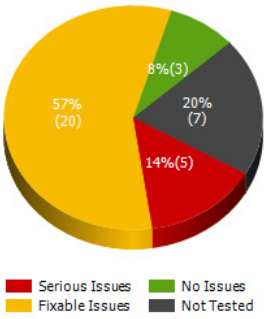
Application Compatibility Dashboard

FLEXERA SOFTWARE™ AdminStudio®

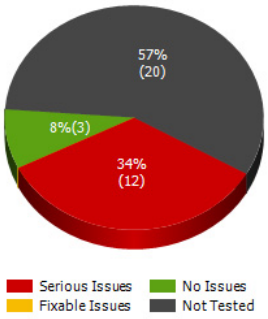
The **Application Compatibility Dashboard** provides a snapshot of the compatibility test results of the currently imported applications in Compatibility Solver. This includes the results of testing for compatibility with Windows operating systems, 64-bit Windows platforms, and certain Internet Explorer versions.

Note: Click on the charts below to open more detailed reports.

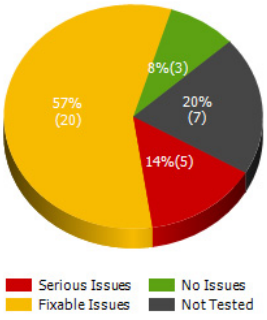
Windows 7 Compatibility



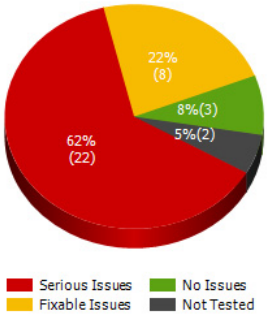
Windows 64-Bit Compatibility



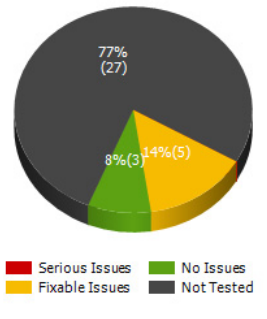
Windows Server 2008 Compatibility



Internet Explorer 7-8 Compatibility



Internet Explorer 8 Presentation Compatibility

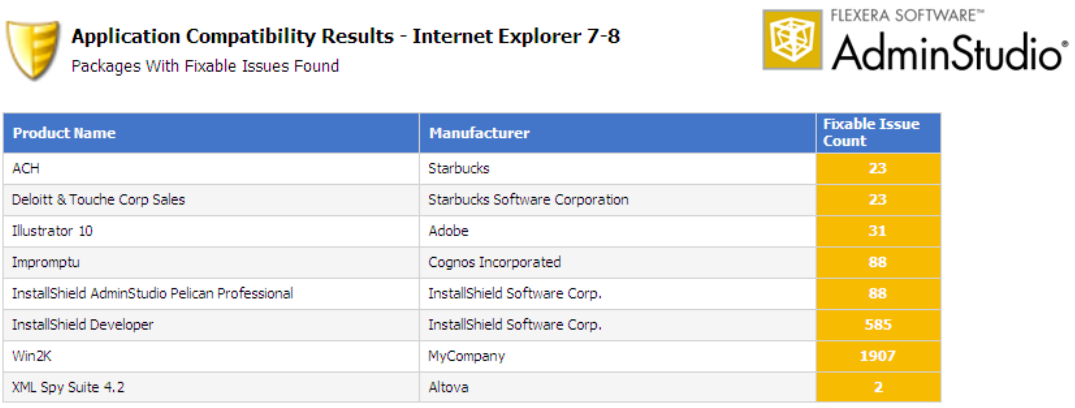


4. Review the following pie charts:

Chart	Description
Windows 7 Compatibility	Lists the Windows 7 application compatibility test results of the currently imported applications in Compatibility Solver.
Windows 64-Bit Compatibility	Lists the Windows 64-bit application compatibility test results of the currently imported applications in Compatibility Solver.
Windows Server 2008 Compatibility	Lists the Windows Server 2008 and 2008 R2 application compatibility test results of the currently imported applications in Compatibility Solver.
Internet Explorer 7-8 Compatibility	Lists the Internet Explorer 7-8 application compatibility test results of the currently imported applications in Compatibility Solver.
Internet Explorer 8 Presentation Compatibility	Lists the Internet Explorer 8 presentation application compatibility test results of the currently imported applications in Compatibility Solver.

In these charts, packages are grouped according to the type of issues found: **Serious Issues**, **Fixable Issues**, **No Issues**, or **Not Tested**.

5. To see a list of packages that fall into a category, click on that category.



Product Name	Manufacturer	Fixable Issue Count
ACH	Starbucks	23
Deloitt & Touche Corp Sales	Starbucks Software Corporation	23
Illustrator 10	Adobe	31
Impromptu	Cognos Incorporated	88
InstallShield AdminStudio Pelican Professional	InstallShield Software Corp.	88
InstallShield Developer	InstallShield Software Corp.	585
Win2K	MyCompany	1907
XML Spy Suite 4.2	Altova	2

Note • *In addition to viewing Compatibility Solver test results in the Application Compatibility Dashboard, you can view a summary of Windows 7 Application Compatibility Chart test results in Application Manager's* ***Application Readiness Dashboard****. You can also view Compatibility Solver test results for an individual package in the package's* ***Application Compatibility View****, as described in Viewing Windows Installer Package Application Compatibility Issues.*

Connecting to the Compatibility Solver Database

Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.*

In order to integrate Compatibility Solver test results into the AdminStudio Application Catalog—making them available to display in AdminStudio report views—you need to specify the location of the **Compatibility Solver Database** on the **Application Catalog** tab of the AdminStudio **Options** dialog box.

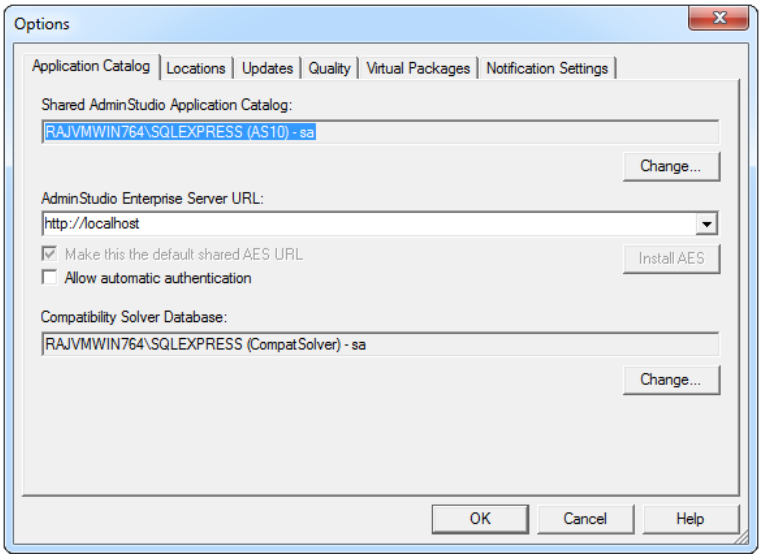


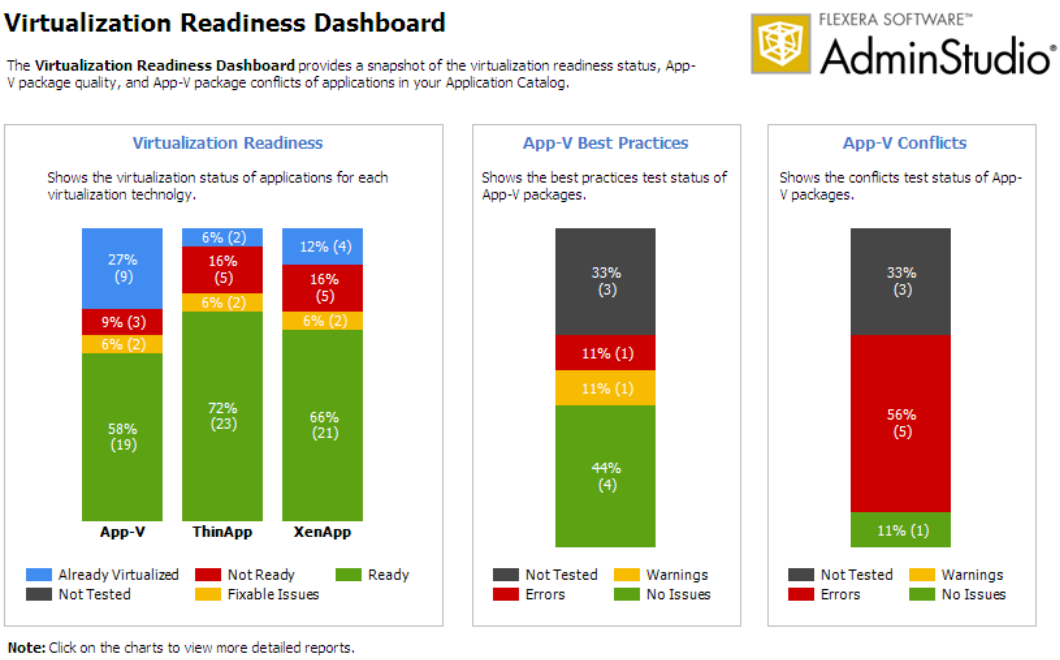
Figure 6-16: Compatibility Solver Database Property on AdminStudio Options Dialog Box

Viewing the Virtualization Readiness Dashboard

The **Virtualization Readiness Dashboard** provides a snapshot of the virtualization readiness status of packages in the Application Catalog for all three supported virtual formats (Microsoft App-V, Citrix XenApp, and VMware ThinApp). This dashboard also shows the best practices and conflict test status of App-V packages in the Application Catalog.

Task: ***To open the Virtualization Readiness Dashboard:***

1. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.
   - Open the AdminStudio interface and select the **Report Center** tab.
2. Select **Virtualization Readiness Dashboard** from the **Select Report** list. The **Virtualization Readiness Dashboard** opens.



Note: Click on the charts to view more detailed reports.

3. Review the following charts:

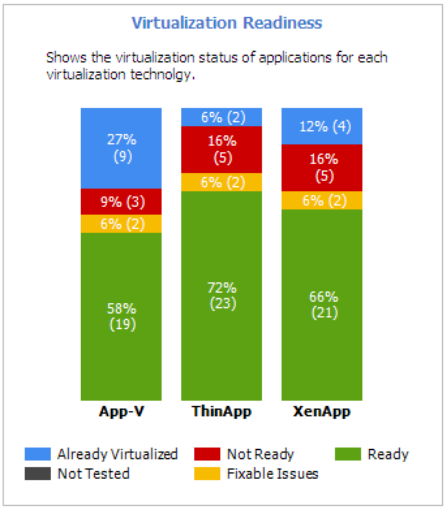
Chart	Description
Virtualization Readiness	These bar charts identify the virtualization status of Windows Installer packages in the Application Catalog for each of the supported virtualization technologies—App-V, ThinApp, and XenApp.  Packages are automatically tested at import and are assigned one of the following statuses: **Already Virtualized**, **Not Tested**, **Not Ready**, **Fixable Issues**, or **Ready**. You can click on any section of these charts to view a list of the packages in that category. ***Note*** *• Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog. To manually retest a package or group of packages for virtualization readiness, select a package, application, or group in the Application Manager tree and select* ***Run Virtualization Readiness*** *from the context menu.*

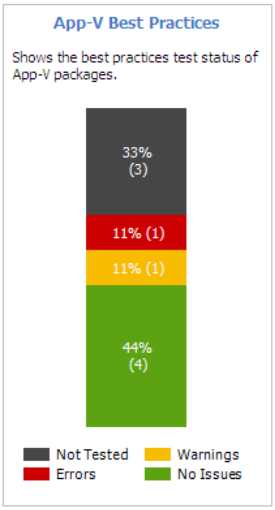
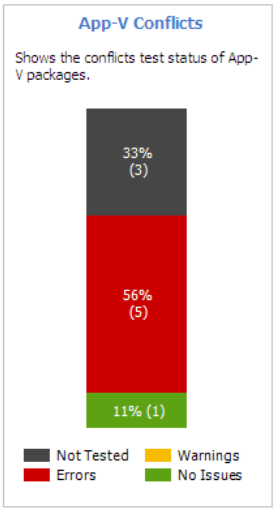
Chart	Description
App-V Best Practices	This bar chart summarizes the results of best practice testing of App-V packages in the Application Catalog.  Packages are tested either automatically at import or by using the Conflict Wizard, and are assigned one of the following statuses: **Not Tested**, **Errors**, **Warnings**, or **No Issues**. You can click on any section of these charts to view a list of the App-V packages in that category.

Chart	Description
App-V Conflicts	This bar chart summarizes the results of conflict analysis of App-V packages in the Application Catalog.  Packages are tested either automatically at import or by using the Conflict Wizard, and are assigned one of the following statuses: **Not Tested**, **Errors**, **Warnings**, or **No Issues**. You can click on any section of these charts to view a list of the App-V packages in that category.

Note • *To see a list of any virtualization suitability issues that have been detected for a specific Windows Installer package, see Viewing Windows Installer Package Virtualization Suitability Issues.*

Virtualization Readiness Status in Application Catalog vs. Automated Application Converter

The following table explains how the Application Catalog package statuses of **Ready**, **Not Ready**, and **Fixable Issues** correspond to statuses assigned to individual packages when they are added to Automated Application Converter:

Table 6-11 • Virtualization Readiness Status in Application Catalog vs. Automated Application Converter

Application Catalog Status	Automated Application Converter Status	Description
Ready	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.

Table 6-11 • Virtualization Readiness Status in Application Catalog vs. Automated Application Converter

Application Catalog Status	Automated Application Converter Status	Description
Not Ready	Virtualization not supported	Virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized in Automated Application Converter. In order to virtualize the package, you must first override the status and change it to **Ready** or **Requires repackaging**.* ***Note*** • *For more information, see Virtualization Suitability Issues.*
	Virtualization not recommended	This package is not recommended for virtualization due to one of the following issues: • Package includes a custom shell extension. • Package utilizes ClickOnce technology. ***Note*** • *For more information, see Virtualization Suitability Issues.*
Fixable Issues	Virtualization not recommended	This package is not recommended for virtualization because it does not contain a shortcut. It is possible to resolve this issue by manually creating a shortcut in the package before performing conversion. ***Note*** • *For more information, see Virtualization Suitability Issues.*

Manually Run Virtualization Readiness Tests

Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog.

You can also choose to manually retest a package or group of packages for virtualization readiness.

Task: ***To manually run virtualization readiness test:***

1. Open Application Manager.
2. Select a package, application, or group in the Application Manager tree and select **Run Virtualization Readiness** from the context menu. The Virtualization Readiness tests are run.
3. To view a summary of the virtualization readiness status of all applications in the Application Catalog, select the **Application Catalog** node and select **Virtualization Readiness Dashboard** from the **Select Report** list. See Viewing the Virtualization Readiness Dashboard.
4. To view the virtualization readiness status of an individual Windows Installer package, expand the Windows Installer package node in the tree and select the **Virtualization Suitability** node. The **Virtualization Suitability View** opens. See Viewing Windows Installer Package Virtualization Suitability Issues.

Viewing Other Summary Reports

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

AdminStudio makes it easy for you to view Application Catalog summary information on the Windows Installer and App-V packages in your catalog, giving you insight into the readiness of those packages for distribution.

In addition to the Application Readiness Dashboard, Application Compatibility Dashboard, and Virtualization Readiness Dashboard, AdminStudio offers the following additional summary reports:

- Viewing the Shared Extensions Report
- Viewing the Error Category Breakdown Report
- Viewing the App-V Dependencies/Dependents Reports

Viewing the Shared Extensions Report

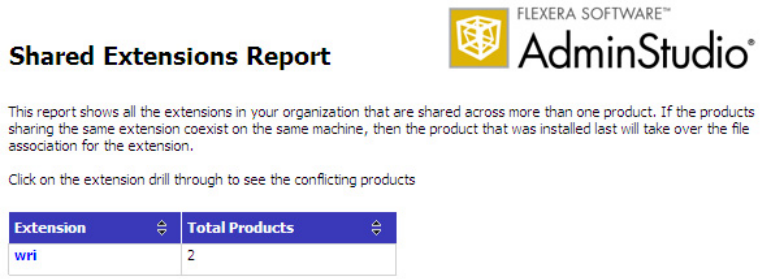
Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

The **Shared Extensions Report** lists all of the extensions used in the Application Catalog that are shared across more than one product.

Note • *If the products sharing the same extension coexist on the same machine, then the product that was installed last will take over the file association for the extension.*

Task: ***To open the Shared Extensions Report:***

1. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.
   - Open the AdminStudio interface and select the **Report Center** tab.
2. Select **Shared Extensions** from the **Select Report** list. The **Shared Extensions Report** opens.



3. To see a report that lists the conflicting products, click on an extension in the list.

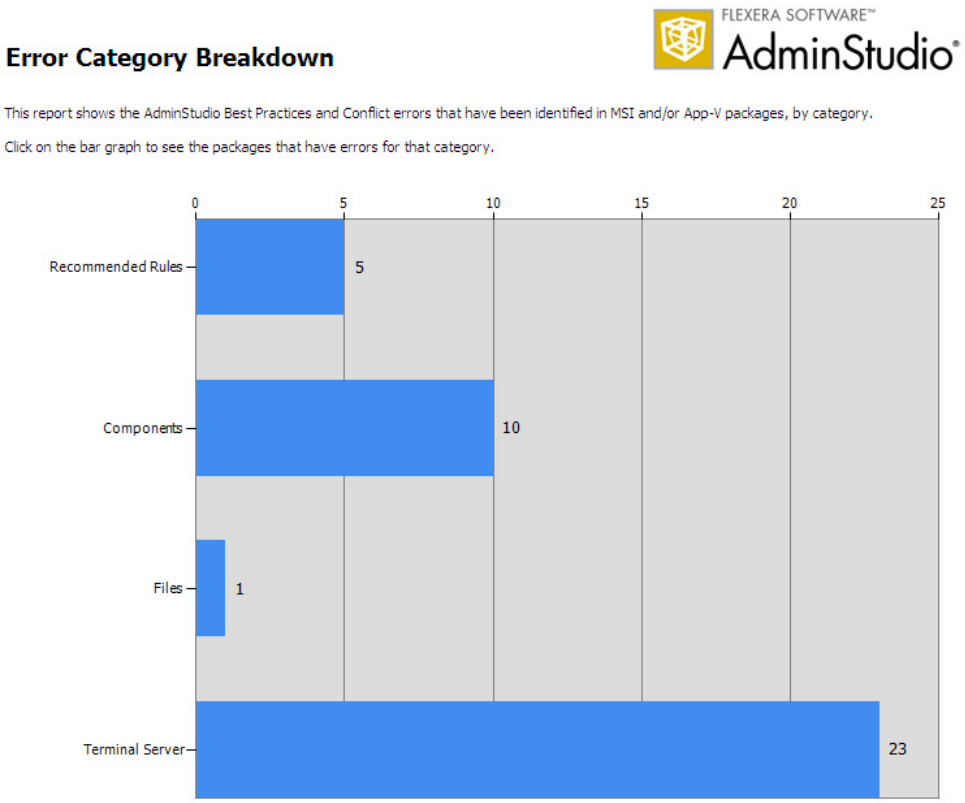
Viewing the Error Category Breakdown Report

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

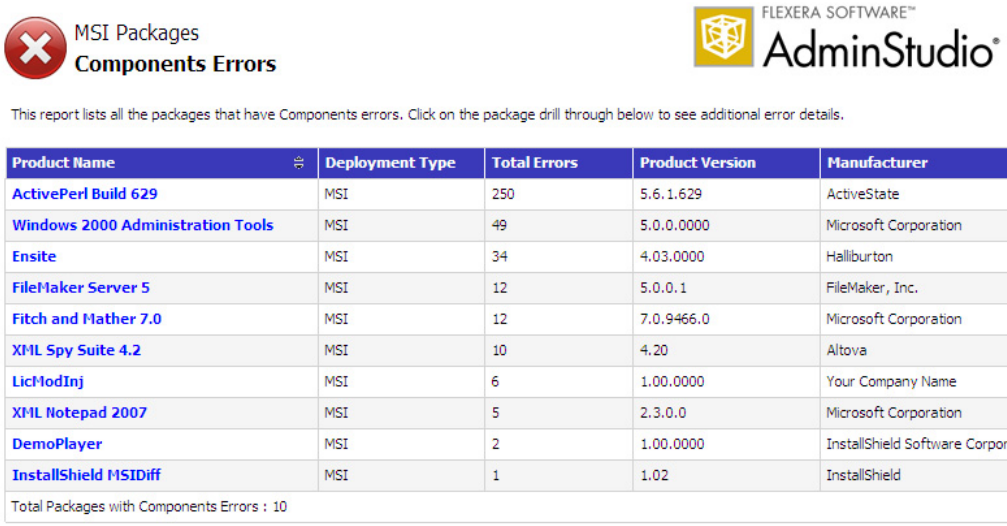
The **Error Category Breakdown** report shows the number of AdminStudio Best Practices and Conflict errors that have been identified in Windows Installer and App-V packages, by category.

Task: ***To open the Error Category Breakdown report:***

1. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.
   - Open the AdminStudio interface and select the **Report Center** tab.
2. Select **Error Category Breakdown** from the **Select Report** list. The **Error Category Breakdown** report opens.



3. To see a report that lists the packages that have errors for that category, click on a category.



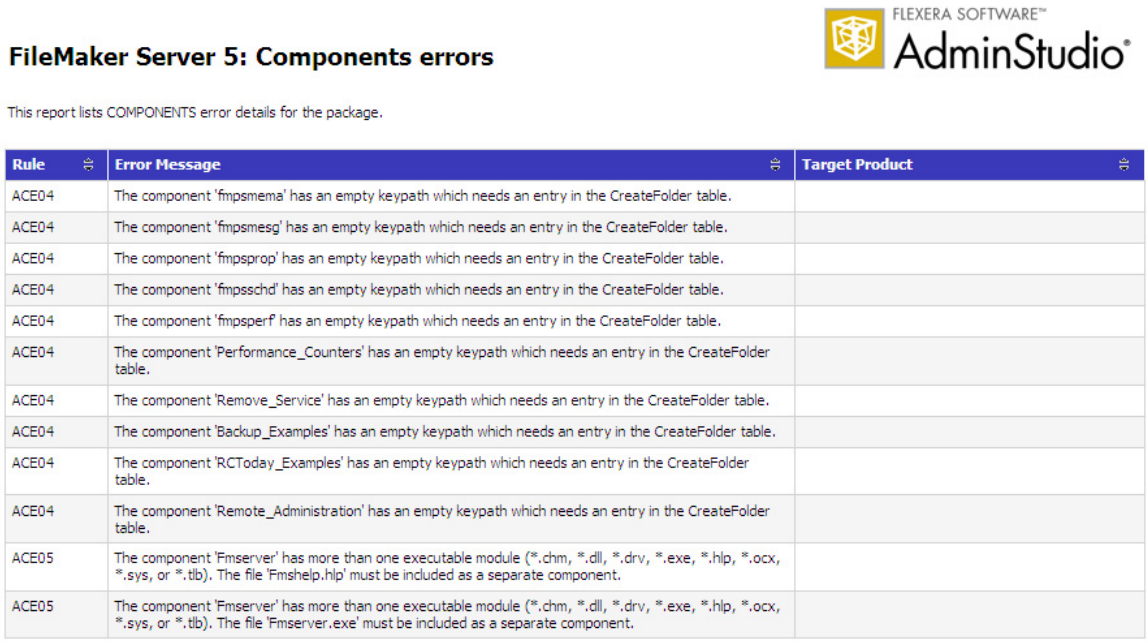
MSI Packages
Components Errors

FLEXERA SOFTWARE™ AdminStudio

This report lists all the packages that have Components errors. Click on the package drill through below to see additional error details.

Product Name	Deployment Type	Total Errors	Product Version	Manufacturer
ActivePerl Build 629	MSI	250	5.6.1.629	ActiveState
Windows 2000 Administration Tools	MSI	49	5.0.0.0000	Microsoft Corporation
Ensite	MSI	34	4.03.0000	Halliburton
FileMaker Server 5	MSI	12	5.0.0.1	FileMaker, Inc.
Fitch and Mather 7.0	MSI	12	7.0.9466.0	Microsoft Corporation
XML Spy Suite 4.2	MSI	10	4.20	Altova
LicModInj	MSI	6	1.00.0000	Your Company Name
XML Notepad 2007	MSI	5	2.3.0.0	Microsoft Corporation
DemoPlayer	MSI	2	1.00.0000	InstallShield Software Corporation
InstallShield MSIDiff	MSI	1	1.02	InstallShield
Total Packages with Components Errors : 10				

4. To see a report that lists the specific errors for a package, click on the **Product Name**.



FileMaker Server 5: Components errors

FLEXERA SOFTWARE™ AdminStudio

This report lists COMPONENTS error details for the package.

Rule	Error Message	Target Product
ACE04	The component 'fmpsmema' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsmesg' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsprop' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsschd' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsperf' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Performance_Counters' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Remove_Service' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Backup_Examples' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'RCToday_Examples' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Remote_Administration' has an empty keypath which needs an entry in the CreateFolder table.	
ACE05	The component 'Fmserver' has more than one executable module (*.chm, *.dll, *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb). The file 'Fmshelp.hlp' must be included as a separate component.	
ACE05	The component 'Fmserver' has more than one executable module (*.chm, *.dll, *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb). The file 'Fmserver.exe' must be included as a separate component.	

Viewing the App-V Dependencies/Dependents Reports

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

You can view a report that lists all App-V applications in the Application Catalog that have dependent applications or that are dependent applications, and a report that lists the recommended dependencies for an App-V package.

- Viewing the App-V Dependencies Report
- Viewing the App-V Dependents Report
- Viewing the App-V Recommended Dependencies Report

Viewing the App-V Dependencies Report

The **App-V Dependencies** report lists all of the App-V packages in the Application Catalog that have dependent App-V packages through the Dynamic Suite Composition feature of App-V.

Task: ***To open the App-V Dependencies report:***

1. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.
   - Open the AdminStudio interface and select the **Report Center** tab.
2. Select **App-V Dependencies** from the **Select Report** list. The **App-V Dependencies** report opens.

App-V Dependencies

FLEXERA SOFTWARE™ AdminStudio

This report lists all Applications in the Application Catalog with its dependencies.

Application	Dependency Applica	In Catalog?	DSC Server URL	Server URL	Mandatory?
Ad-Aware SE Personal	HyperCam	Yes	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	Yes
	ListMaker v1.15.5	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	Yes
	Winzip	Yes	RTSP://%SFT_SOFTGRIDSERVER %:554/Winzip/Winzip.sft	RTSP://%SFT_SOFTGRIDSERVER %:554/Winzip/Winzip.sft	Yes
AppVSample		No	RTSP://%SFT_SOFTGRIDSERVER %:554/Firefox/Firefox.sft		No
FontTest	JavaJRE	Yes	FILE://C:\Users\Test\Desktop\JavaJRE - VPE \JavaJRE_2.sft	rtsps://%SFT_SOFTGRIDSERVER %:322/JavaJRE.sft/JavaJRE_2.sft/JavaJRE_3.sft/JavaJ RE_4.sft	Yes
HyperCam	Ad-Aware SE Personal	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	snagit	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/snagit/snagit.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/snagit/snagit.sft	Yes
ListMaker v1.15.5	Ad-Aware SE Personal	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	ListMaker v1.15.5	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	Yes

Viewing the App-V Dependents Report

The **App-V Dependents** report lists all of the App-V packages in the Application Catalog that are dependent upon other App-V packages through the Dynamic Suite Composition feature of App-V.

Task: ***To open the App-V Dependents report:***

1. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.
   - Open the AdminStudio interface and select the **Report Center** tab.
2. Select **App-V Dependents** from the **Select Report** list. The **App-V Dependents** report opens.

App-V Dependents

FLEXERA SOFTWARE™ AdminStudio®

This report lists all Applications in the Application Catalog and its dependent applications.

Application	Dependent Application	In Catalog?	DSC Server URL	Server URL	Mandatory?
Ad-Aware SE Personal	HyperCam	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	ListMaker v1.15.5	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	snagit	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
Adobe Reader 8	Sample Kit	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/Adobe Reader 8/Adobe Reader 8_2.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/Adobe Reader 8/Adobe Reader 8_2.sft	Yes
Application Verifier 3.3	Sample Kit	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/Application Verifier 3/Application Verifier 3.3.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/Application Verifier 3/Application Verifier 3.3.sft	Yes
FontTest	Sample Kit	Yes	RTSPS://%SFT_SOFTGRIDSERVER %:322/FontTest.sft/FontTest_4.sft	RTSPS://%SFT_SOFTGRIDSERVER %:322/FontTest.sft/FontTest_4.sft	Yes
HyperCam	Ad-Aware SE Personal	Yes	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	Yes
JavaJRE_4	FontTest	Yes	FILE://C:\Users\Test\Desktop\JavaJRE - VPE \JavaJRE_2.sft	rtsps://%SFT_SOFTGRIDSERVER %:322/JavaJRE.sft/JavaJRE_2.sft/JavaJRE_3.sft/JavaJRE_4.sft	Yes
ListMaker v1.15.5	Ad-Aware SE Personal	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	Yes

Viewing the App-V Recommended Dependencies Report

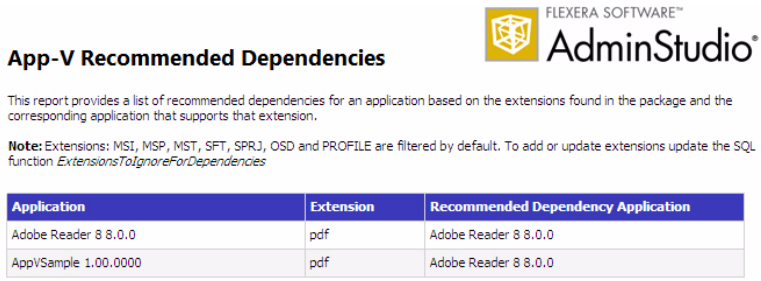
The **App-V Recommended Dependencies** report lists the recommended Dynamic Suite Composition dependencies for App-V packages based on cross-referencing the file extensions found in them with published file type associations from other App-V packages in the catalog.

Task: ***To open the App-V Recommended Dependencies report:***

1. Do one of the following:
   - Open Application Manager and select the **Application Catalog** node.

- Open the AdminStudio interface and select the **Report Center** tab.

2. Select **App-V Recommended Dependencies** from the **Select Report** list. The **App-V Recommended Dependencies** report opens.



App-V Recommended Dependencies

FLEXERA SOFTWARE™ AdminStudio®

This report provides a list of recommended dependencies for an application based on the extensions found in the package and the corresponding application that supports that extension.

Note: Extensions: MSI, MSP, MST, SFT, SPRJ, OSD and PROFILE are filtered by default. To add or update extensions update the SQL function *ExtensionsToIgnoreForDependencies*

Application	Extension	Recommended Dependency Application
Adobe Reader 8 8.0.0	pdf	Adobe Reader 8 8.0.0
AppVSample 1.00.0000	pdf	Adobe Reader 8 8.0.0

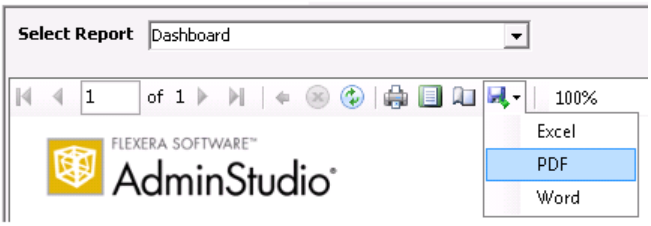
Exporting a Report in PDF, Excel, or Word Format

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

You can save any of the Application Catalog summary reports or any of their drill-through reports in PDF, Microsoft Excel, or Microsoft Word format.

Task: ***Saving a report:***

1. Open the report that you want to save.
2. In the toolbar, click the **Save** icon.



3. From the menu, select either **Excel**, **PDF**, or **Word**. The report is exported and you are prompted for a location to store the report.
4. Specify a location and click **Save**.

Note • *You can also print the currently viewed report by clicking the **Print** icon in the toolbar.*

Creating Customizable Reports

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

You can create your own custom reports that are generated using Microsoft SQL Reporting Services.

To create a custom report, you need to first create an `.rdlc` file (using either Microsoft SQL Server Business Intelligence Development Studio or Visual Studio 2010), copy that file to the AdminStudio installation directory, and then edit the AdminStudio Report Definition file (`AdminStudio.Reports.xml`) to add a reference to your custom report.

Task: ***To create a custom Microsoft SQL Reporting Services report:***

1. Create an `.rdlc` file.
2. Copy that file to the following directory on the machine where AdminStudio is installed:

   `[AdminStudio Installation Directory]\ConflictSolver\ReportDefinition\RDLC`
3. Open the `AdminStudio.Reports.xml` file, found in the following location, in a text editor:

   `[AdminStudio Installation Directory]\ConflictSolver\ReportDefinition`
4. In the `AdminStudio.Reports.xml` file, copy an existing `<Report>` element under the `<Reports>` element. Below are the `<Report>` elements for the `Shared Extensions` report and the `Shared Extensions By Product` drill-through report:

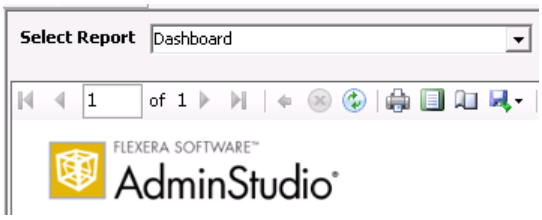
```
<Report Name="Shared Extensions" RdlcPathType="Relative"
    RdlcPath="RDLC\SharedExtensions.rdlc" IsView="">
    <DataSources>
        <DataSource DataSourceName="ds_asrpt_SharedExtensions"
        SqlCommandType="StoredProcedure" SqlCommandString="sp_asrpt_SharedExtensions">
            <SqlParameters>
                <SqlParameter />
            </SqlParameters>
        </DataSource>
    </DataSources>
        <Report Name="Shared Extensions By Product" RdlcPathType="Relative"
        RdlcPath="RDLC\SharedExtensionsByProduct.rdlc">
            <DataSources>
                <DataSource DataSourceName="ds_asrpt_SharedExtensionsByProduct"
                SqlCommandType="StoredProcedure"
                SqlCommandString="sp_asrpt_SharedExtensionsByProduct">
                    <SqlParameters>
                        <SqlParameter DrillThroughParameterName="Extension"
                        SqlParameterName="@extension" SqlParameterValue="" />
                    </SqlParameters>
                </DataSource>
            </DataSources>
        </Report>
</Report>
```

5. Replace the highlighted text above with the correct information for your new custom report: the data source and the parameters that are required for any drill through reports that you include.

   In this XML file, note the following parameter values:

Parameter	Values
`RdlcPathType`	Valid values are `Relative` or `Absolute`.
`SqlCommandType`	Valid values are `StoredProcedure` or `SQL`.
`SqlCommandString`	Enter the stored procedure name or an SQL string.

6. To view your custom report in Application Manager, select the name of your custom report from the **Select Report** drop down list on the Application Catalog View.



Viewing Package Data

Application Manager provides views of both Windows Installer and App-V package metadata:

- Viewing Windows Installer Package Data
- Viewing App-V Package Data

Viewing Windows Installer Package Data

Application Manager provides the following views of Windows Installer package data:

- Viewing and Editing Package Extended Attributes
- Viewing Windows Installer Package Files/Components
- Viewing Windows Installer Package INI File Changes
- Viewing Windows Installer Package Registry Information
- Viewing Windows Installer Package Shortcuts
- Viewing Windows Installer Package Virtualization Suitability Issues
- Viewing Windows Installer Package Application Compatibility Issues
- Viewing Windows Installer Package Merge Modules

- Viewing Package Catalog History
- Viewing Package Tables

Viewing and Editing Package Extended Attributes

Extended attributes are optional attributes for packages, defined by an Extended Attribute Description File (in XML format). Because you can manually create the description file, you have the flexibility to include information about each package that may be specific to your organization—such as the users or business groups that receive the package.

Extended attributes can be edited in the **Extended Attributes View** on a package-by-package basis.

The following tasks and concepts relate to extended attributes:

- Using Extended Attributes
- Extended Attribute Description File
- Integrating Package Extended Attribute Data with an Application Request

Note • *In addition to Windows Installer packages, you can also view and edit Extended Attributes for Microsoft App-V, Citrix XenApp, and VMware ThinApp virtual packages, as well as for non-Windows Installer legacy packages.*

Using Extended Attributes

Assuming you have created an extended attribute description file (or are using the default provided file), you can configure Application Manager to use it with the current Application Catalog.

Task: ***To use extended attributes in Application Manager:***

1. With Application Manager open, click **Options** on the **Tools** menu. The ConflictSolver and Application Manager Options Dialog Box opens.
2. Click the **Extended Attributes** tab.
3. Specify or browse to the extended attribute description file (`.xml`) containing the extended attributes you want to use for Application Manager.
4. Click **OK**.

The **Extended Attributes View** is available under each package in Application Manager.

Caution • *The default Extended Attribute description file is named EA_Default.xml, and is installed in the AdminStudio Shared folder. You can modify the data displayed in the* ***Extended Attributes*** *view, but to do this, do not edit the EA_Default.xml file. Instead, copy the EA_Default.xml file, rename it, make your edits to the new file, and*

*then enter the new file name and location in the Extended Attribute Description File field on the **Extended Attributes** tab of the ConflictSolver and Application Manager **Options** dialog box.*

Extended Attribute Description File

Application Manager uses an XML file to describe the data that appears in the **Extended Attributes** view. The name and location of this XML file can be specified from the **Extended Attributes** tab in the ConflictSolver and Application Manager **Options** dialog box.

***Caution** • The default Extended Attribute description file is named EA_Default.xml, and is installed in the AdminStudio Shared folder. You can modify the data displayed in the **Extended Attributes** view, but to do this, do not edit the EA_Default.xml file. Instead, copy the EA_Default.xml file, rename it, make your edits to the new file, and then enter the new file name and location in the Extended Attribute Description File field on the **Extended Attributes** tab of the ConflictSolver and Application Manager **Options** dialog box.*

Description File Properties

The description file, which is in XML format, contains tags for each extended attribute (up to a limit of 400 attributes). It supports text or file values. The following list explains each tag available in the description file.

Table 6-12 • Description File Tags

Attribute	Description
UniqueIdentifier	This value, which Application Manager uses to validate that the XML file is for extended attributes, must be set to ISASEA40.
Name	The name of the attribute as it appears in the Extended Attributes view. This cannot exceed 255 characters.
Type	The extended attribute type. This can be Text, File, or Selection. If no type is specified, then Application Manager defaults the attribute to text.
DefaultValue	This tag, available only for Text types, provides the default value for the attribute. This optional value cannot exceed 512 characters.
DefaultFileExtension	This tag, available only for File types, provides the default file extension when you browse for the file. Examples of this could be *.txt, *.bmp, *.doc, or *.* (representing all files).

Table 6-12 • Description File Tags

Attribute	Description
FileFilter	Provide the file types to populate the File type filter in the Browse dialog box. These must be in pairs, and in the format Longname (*.ext)|*.ext. Before the closing </FileFilter> tag, you must have two pipe symbols (||). For example, to include filters for text files, bitmaps, and all files, use the following line: `<FileFilter>Text Document (*.txt)|*.txt|Bitmap (*.bmp)|*.bmp|All Files (*.*)|*.*||</FileFilter>` This value cannot exceed 255 characters.
Caption	The caption for the Browse dialog box when using File types. This cannot exceed 255 characters.
Values	Used only for Selection types, this is a semicolon-delimited list of possible values for the selection. These will appear in a drop-down list for the extended attribute. The first value is used as the default. The total number of characters of all the values and necessary semicolons cannot exceed 255 characters.
HelpText	Text that appears below the value field for either Text or Selection attributes. You can use it to provide additional information to help users know what to input. This cannot exceed 512 characters.

Description File Format

An example of an extended attribute description file follows:

```
<Extended_Attribute UniqueIdentifier="ISASEA40">
   <AttributeDetails>
      <Name>Owner</Name>
      <Type>Text</Type>
      <DefaultValue></DefaultValue>
      <HelpText>Provide the name of the package's owner.
      </HelpText>
   </AttributeDetails>
   <AttributeDetails>
      <Name>Test Script</Name>
      <Type>File</Type>
      <DefaultFileExtension>*.*</DefaultFileExtension>
      <FileFilter>All Files (*.*)|*.*||</FileFilter>
      <Caption>Test Script Files</Caption>
   </AttributeDetails>
   <AttributeDetails>
      <Name>Program Type</Name>
      <Type>Selection</Type>
      <Values>Office Application;Utility;
      Graphic Application;Programming Application;Game;Other
      </Values>
      <HelpText>Select the type of application from the
```

```
      above list.
      </HelpText>
   </AttributeDetails>
</Extended_Attribute>
```

Integrating Package Extended Attribute Data with an Application Request

Note • *AdminStudio Workflow Manager is a Web-based application management system that has integrated functionality with AdminStudio.*

Application Manager allows you to integrate extended attributes with AdminStudio Workflow Manager. This option is enabled by selecting the **Integrate with Workflow Manager** option on the **Extended Attributes** of the ConflictSolver and Application Manager **Options** dialog box.

When the **Integrate with Workflow Manager** option is selected, you can associate extended attribute data for packages in Application Manager with Application Requests in Workflow Manager. This is accomplished by right-clicking on the package name in the Application Manager Product View and selecting **Associate with Workflow Manager Application** from the context menu.

Task: ***To associate a package with a Workflow Manager Application Request:***

1. Open Application Manager.
2. Connect to the AdminStudio Enterprise Server Application Catalog. See Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog.
3. On the **Tools** menu, click **Options**. The **Options** dialog box opens.
4. On the **Extended Attributes** tab, confirm that the **Integrate with Workflow Manager** option is selected.
5. In the Application Manager **Product View**, select the product that you want to link to Workflow Manager and select **Associate with Workflow Manager Application** from the context menu.

   The **Associate with Workflow Manager Application** dialog box opens.
6. Pick the application in Workflow Manager with which you want to associate this product.

Assuming that Workflow Manager is configured to use the same extended attributes file, if you enter new data into the Extended Attributes view in Application Manager, Workflow Manager automatically detects it; if changes are made in Workflow Manager, they are automatically reflected in Application Manager.

By design, extended attributes data in Application Manager and Workflow Manager data have a one-to-one relationship. You can only associate one Workflow Manager application with a product in Application Manager; once the application is associated, it is no longer available for association with other Application Manager products.

Note • *Another integration feature between AdminStudio and Workflow Manager is that when you associate a package with a Workflow Manager Application Request and then view that Application Request's Application*

Report, there is a link to open the Package Report of its associated package. There is also a link on the Package Report to open the Application Report of its associated Application Request.

Viewing Windows Installer Package Files/Components

To display the files and components in a Windows Installer package, expand the Windows Installer package in the Application Manager tree and select the **Files/Components** node.

Task: ***To view Windows Installer package Files/Components:***

1. Open Application Manager.
2. Expand a Windows Installer package in the Application Manager tree and select the **Files/Components** node. The **Files/Components View** opens.
3. Review the following information:

Column	Description
Component	Name of component that the file listed in the FileName column is associated with.
FileName	Name of file.
FileSize	Size of the file listed in the FileName column.
Version	Version of the file listed in the FileName column.
Path	Installation location of the file listed in the FileName column.

Viewing Windows Installer Package INI File Changes

To display any INI file changes made by a Windows Installer package, expand the Windows Installer package in the Application Manager tree and select the **INI File Changes** node.

Task: ***To view Windows Installer package INI file changes:***

1. Open Application Manager.
2. Expand a Windows Installer package in the Application Manager tree and select the **INI File Changes** node. The **INI File Changes View** opens.
3. Review the following information:

Column	Description
Component	Name of component that makes an entry in the INI File.

Column	Description
FileName	Name of INI File that the component listed in the Component column makes an entry in.
DirProperty	The directory location where the INI File will be installed.
Section	The section of the INI file where this entry is made.
Key	The Key used in the INI File entry
Value	The Value used in the INI File entry.

Viewing Windows Installer Package Registry Information

To display any registry entries created or changed by a Windows Installer package, expand the Windows Installer package in the Application Manager tree and select the **Registry** node.

Task: ***To view Windows Installer package Registry information:***

1. Open Application Manager.
2. Expand a Windows Installer package in the Application Manager tree and select the **Registry** node. The **Registry View** opens.
3. Review the following information:

Column	Description
Component	The name of the component that is creating a Registry entry.
Root	Default value of Key.
Key	Key of the Registry Entry that this component is making.
Name	Name of the Registry Entry that this component is making.
Value	Value of the Registry Entry that this component is making.

Viewing Windows Installer Package Shortcuts

To display any shortcuts created by a Windows Installer package, expand the Windows Installer package in the Application Manager tree and select the **Shortcuts** node.

Task: ***To view Windows Installer package Shortcuts:***

1. Open Application Manager.
2. Expand a Windows Installer package in the Application Manager tree and select the **Shortcuts** node. The **Shortcuts View** opens.
3. Review the following information:

Column	Description
Component	Name of the component that the shortcut listed in the Name column is associated with.
Name	Name of the shortcut.
Directory_	Directory where the shortcut will exist.
Target	Directory and executable that the shortcut invokes.

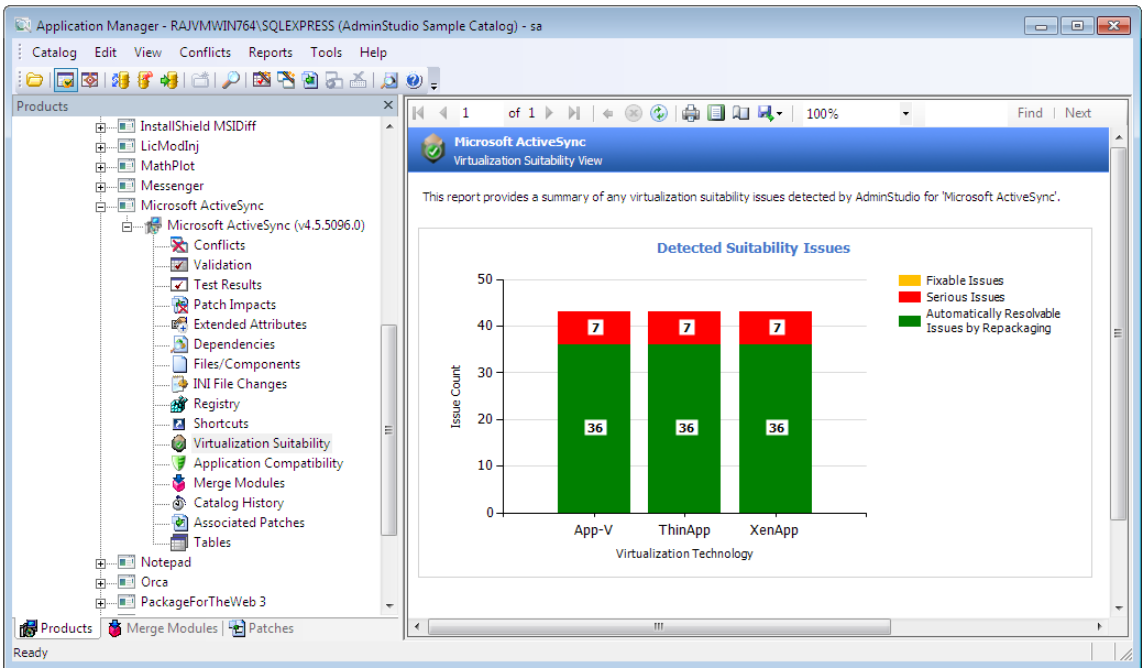
Viewing Windows Installer Package Virtualization Suitability Issues

AdminStudio performs virtualization suitability testing to test a Windows Installer package for suitability to convert it to Microsoft App-V, Citrix XenApp, and VMware ThinApp virtual formats. A Windows Installer package's **Virtualization Suitability View** lists any virtualization suitability issues that have been detected during testing.

To view the results of virtualization suitability testing of a Windows Installer package, expand the Windows Installer package in the Application Manager tree and select the **Virtualization Suitability** node.

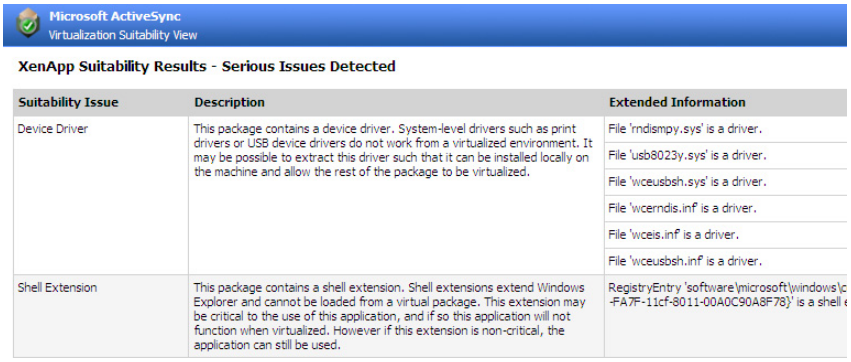
Task: ***To view Windows Installer package Virtualization Suitability View:***

1. Open Application Manager.
2. Expand a Windows Installer package in the Application Manager tree and select the **Virtualization Suitability** node. The **Virtualization Suitability View** opens.



Issues are grouped into three categories: **Fixable Issues**, **Serious Issues**, and **Automatically Resolvable Issues by Repackaging.**

3. To view a detailed report of issues in one of the categories, click on one of the categories of the bar charts.



Tip • *Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog. To manually retest a package or group of packages for virtualization readiness, select a package, application, or group in the Application Manager tree and select* ***Run Virtualization Readiness*** *from the context menu.*

Viewing Windows Installer Package Application Compatibility Issues

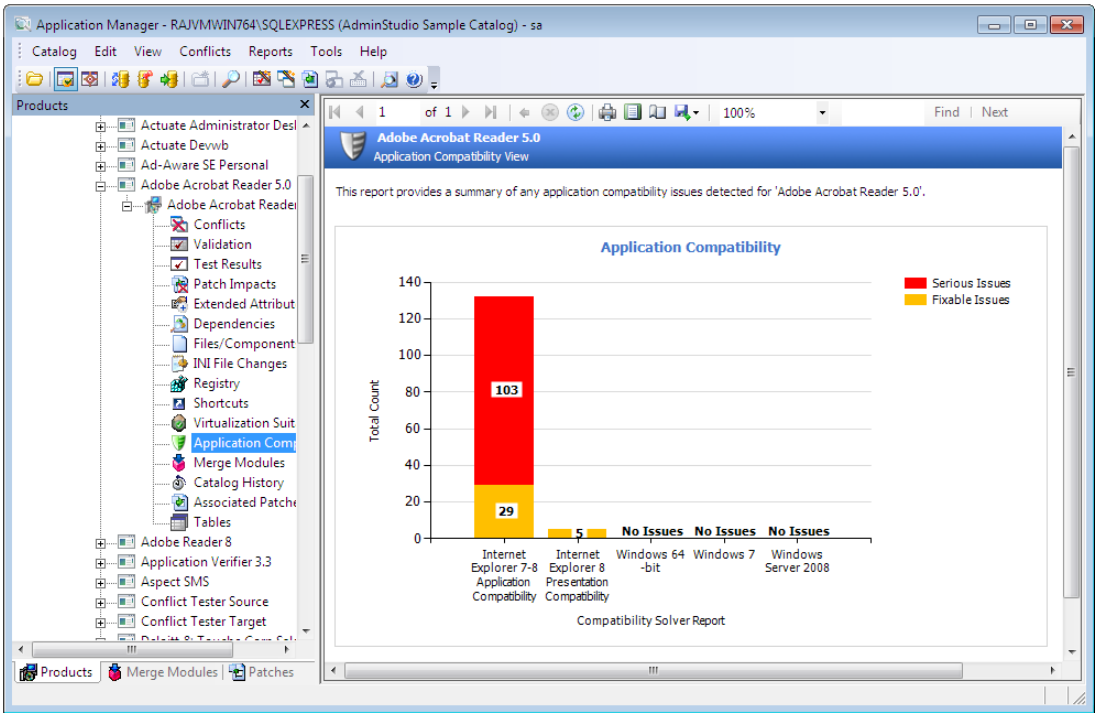
Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.*

To view Compatibility Solver test results for a selected Windows Installer package, expand the Windows Installer package in the Application Manager tree and select the **Application Compatibility** node.

Note • *To view Compatibility Solver test results in AdminStudio, you need to first specify the location of the Compatibility Solver database on the **Application Catalog** tab of the AdminStudio **Options** dialog box, as described in Connecting to the Compatibility Solver Database.*

Task: ***To view Windows Installer package Application Compatibility issues:***

1. Open Application Manager.
2. Expand a Windows Installer package in the Application Manager tree and select the **Application Compatibility** node. The **Application Compatibility View** opens.

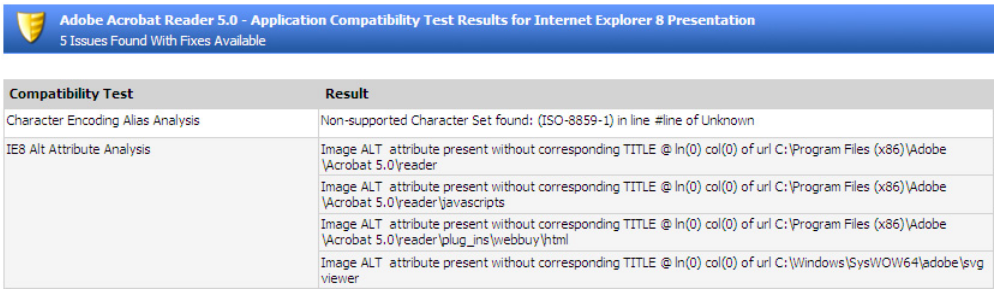


3. Review the following information that is displayed on the **Application Compatibility View**:
   - **Serious and Fixable issues**—This view lists the number of serious and fixable issues for the following sets of compatibility tests: Windows 64-bit, Windows 7, Windows Server 2008 and 2008 R2, Internet

Explorer 7-8 Application Compatibility, and Internet Explorer 8 Presentation Compatibility. If no issues were found, `No Issues` is displayed.

- **No tests run in Compatibility Solver**—If the package has been loaded into Compatibility Solver, but no tests have been run, `Not Tested` is displayed.
- **Package not loaded in Compatibility Solver**—If the package has not been loaded into Compatibility Solver, no results will be displayed.

4. To open more detailed reports, click on any section of a bar chart.



Viewing Windows Installer Package Merge Modules

To display any merge modules included with a Windows Installer package, expand the Windows Installer package in the Application Manager tree and select the **Merge Modules** node.

Task: ***To view a Windows Installer package's Merge Modules:***

1. Open Application Manager.
2. Expand a Windows Installer package in the Application Manager tree and select the **Merge Modules** node. The **Merge Modules View** opens.
3. Review the following information:

Column	Description
Title	The title of the Merge Module included with this package.
ModuleID	The number which uniquely identifies the Merge Module listed in the Title column.
Version	The version of the Merge Module listed in the Title column.
Language	The language that the Merge Module listed in the Title column was written for.

Viewing Package Catalog History

The tracking of change history is a critical operation within the Enterprise environment. Maintaining this information, displaying it, and allowing it to be a filter will gives you the information you need to monitor and maintain the integrity of your software packages.

In AdminStudio, any operation that materially changes a software package or the data associated with the package is tracked, and can be viewed in the Application Manager **Catalog History** view.

Task: ***To view a package's catalog history:***

1. Open **Application Manager**.
2. In the tree, expand a Windows Installer or App-V package node and select the **Catalog History** node. The **Catalog History** view opens, displaying the following information:

Item	Description
Action	Name of the event which was logged: • Import/Reimport • Validation • Conflict Detection • Conflict Resolution • Extended Attribute Modification • Package Description Modification • Package Move/Copy • Patch Impact Analysis ***Note*** *• You can specify which events you want to be listed on the* ***Catalog History*** *view by making selections on the History Tab of the ConflictSolver and Application Manager Options Dialog Box. You can choose which events you want to log and which events you want to display in the* ***Catalog History*** *view.*
Date	Date and time logged event occurred.
User	User who performed the logged event.
Description	Description providing details of the logged event.

Note *• If a package was replicated into another Application Catalog, its history data would not be replicated.*

Deleting Package History

To delete all of the entries in a package's History Log File, perform the following steps:

Task: ***To delete package history:***

1. Open **Application Manager**.
2. Select the package and right-click to open the context menu.
3. On the context menu, point to **Delete** and click **History Log Information**.

Viewing Package Tables

The **Tables** view provides a way to view table data for a Windows Installer or App-V package in the Application Catalog.

Most tables are derived directly from standard MSI tables, as described in the Windows Installer SDK online help. When building your own ACE rules to use for conflict identification, it is important to understand the data available for packages so you can construct the necessary rule.

To display table information for a package, expand the package in the Application Manager tree and select the **Tables** node.

Task: ***To view package tables:***

1. Open Application Manager.
2. Expand a Windows Installer or App-V package in the Application Manager tree and select the **Tables** node. The **Tables** view opens.
3. Select the specific table you want to view from the **Tables** list at the top of the view.

Viewing App-V Package Data

AdminStudio provides comprehensive application data reports for App-V packages, including the following information:

- Viewing and Editing Package Extended Attributes
- Viewing App-V Package History
- Viewing App-V Package Dependencies
- Viewing App-V Package Files and Directories
- Viewing App-V Package Registry
- Viewing App-V Package Shortcuts

- Viewing App-V Package File Type Associations
- Viewing App-V Package Environment Variables
- Viewing Package Catalog History
- Viewing Package Tables

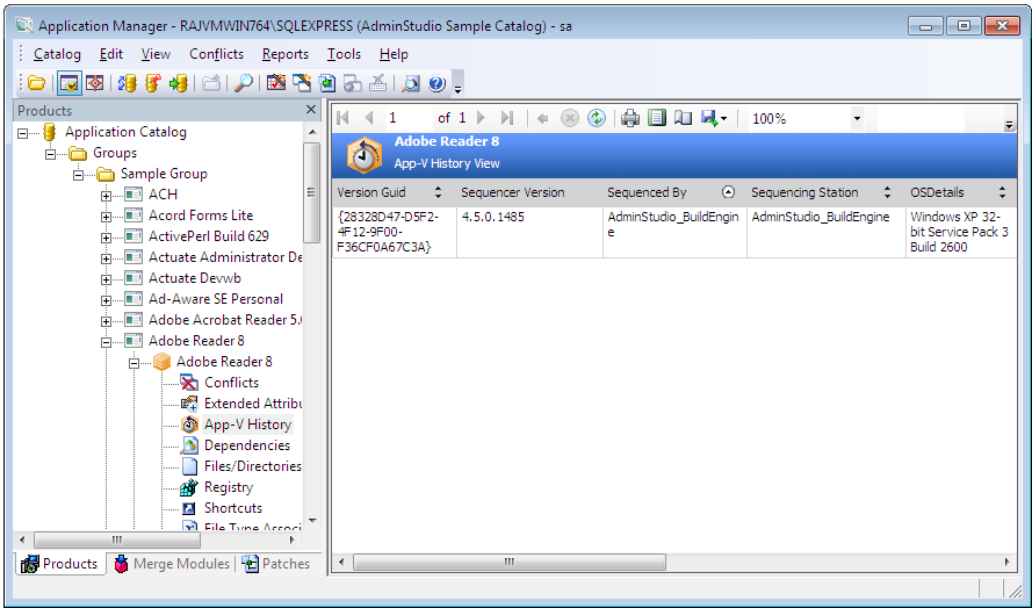
Note • *Many of these reports offer drill-down functionality, enabling you to view even more detailed information, and can be printed separately, or saved in PDF, Microsoft Excel, or Microsoft Word format. See Exporting a Report in PDF, Excel, or Word Format.*

Viewing App-V Package History

The **App-V History View** lists an entry for each time the App-V package has been saved. To view App-V package history, perform the following steps.

Task: ***To view App-V package history:***

1. Open **Application Manager**.
2. In the tree, expand an App-V package and select the **App-V History** node. The **App-V History View** opens, listing an entry for each time this App-V package has been saved.



For each entry, the following information is displayed:

- Version GUID
- Sequencer Version

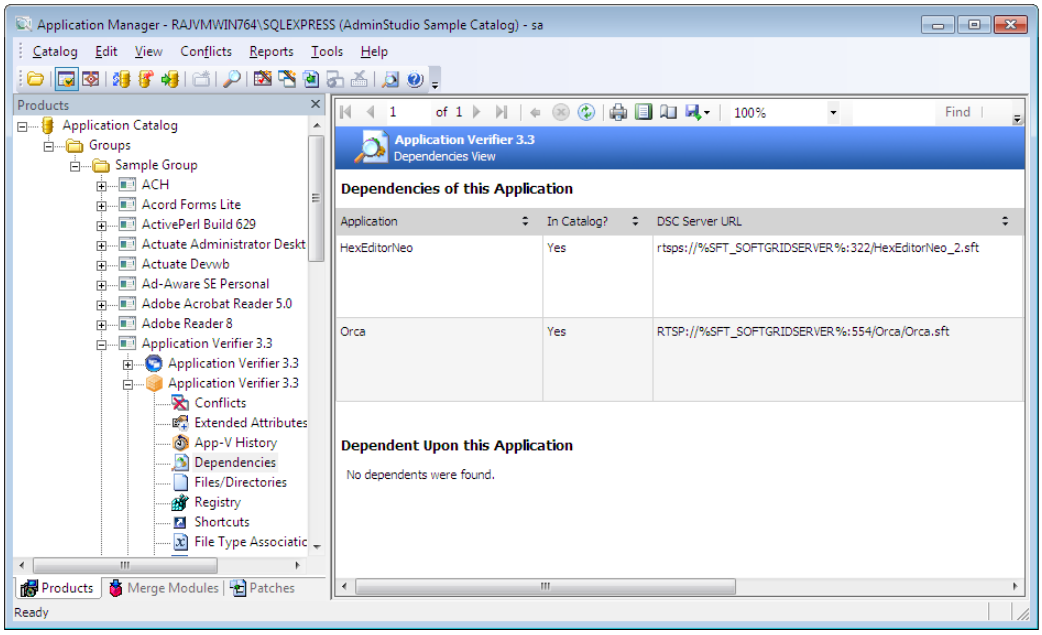
- Sequenced By
- Sequencing Station
- OSDetails
- System Folder
- Windows Folder
- User Folder
- .Net Framework Version
- IEVersion

Viewing App-V Package Dependencies

The App-V **Dependencies View** lists both the applications the App-V package is dependent on and the applications dependent upon this App-V package. To view App-V package dependencies, perform the following steps.

Task: ***To view App-V package dependencies:***

1. Open **Application Manager**.
2. In the tree, expand an App-V package and select the **Dependencies** node. The **Dependencies View** opens.



This view lists both the applications this App-V package is dependent on and the applications dependent upon this App-V package. For each dependency, the following information is listed:

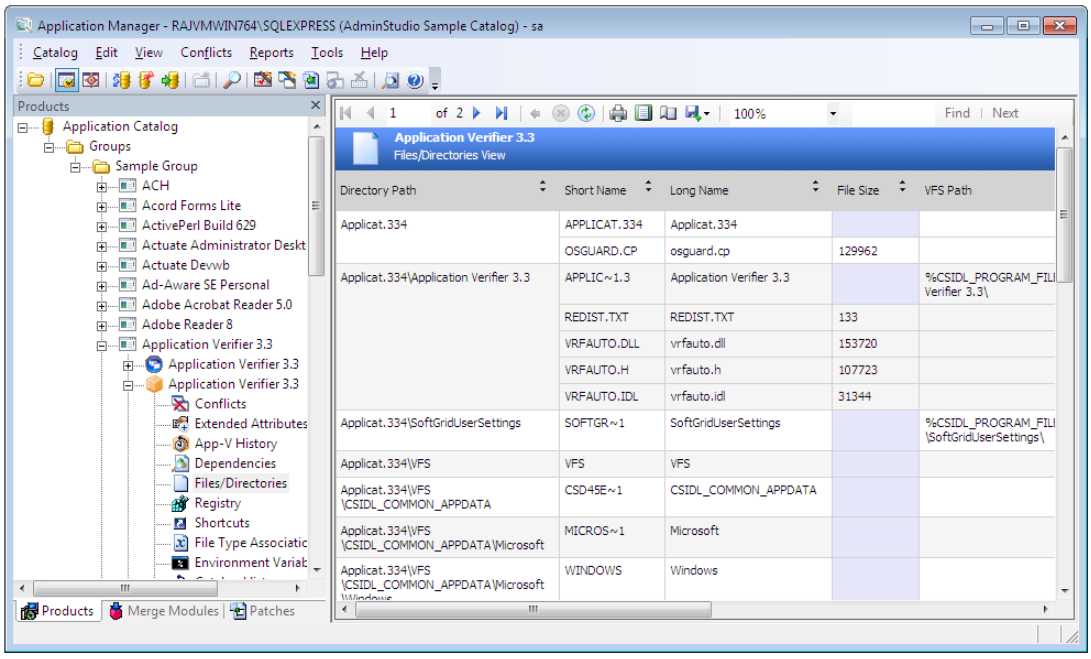
- Application
- In Catalog? (Yes / No)
- DSC Server URL
- Server URL
- Mandatory? (Yes / No)

Viewing App-V Package Files and Directories

The App-V **Files/Directories View** lists the files and directories included in the App-V package.To view App-V package files and directories, perform the following steps.

Task: ***To view App-V package files and directories:***

1. Open **Application Manager**.
2. In the tree, expand an App-V package and select the **Files/Directories** node. The **Files/Directories View** opens.



The following information is listed for each file/directory:

- Directory Path
- Short Name
- Long Name

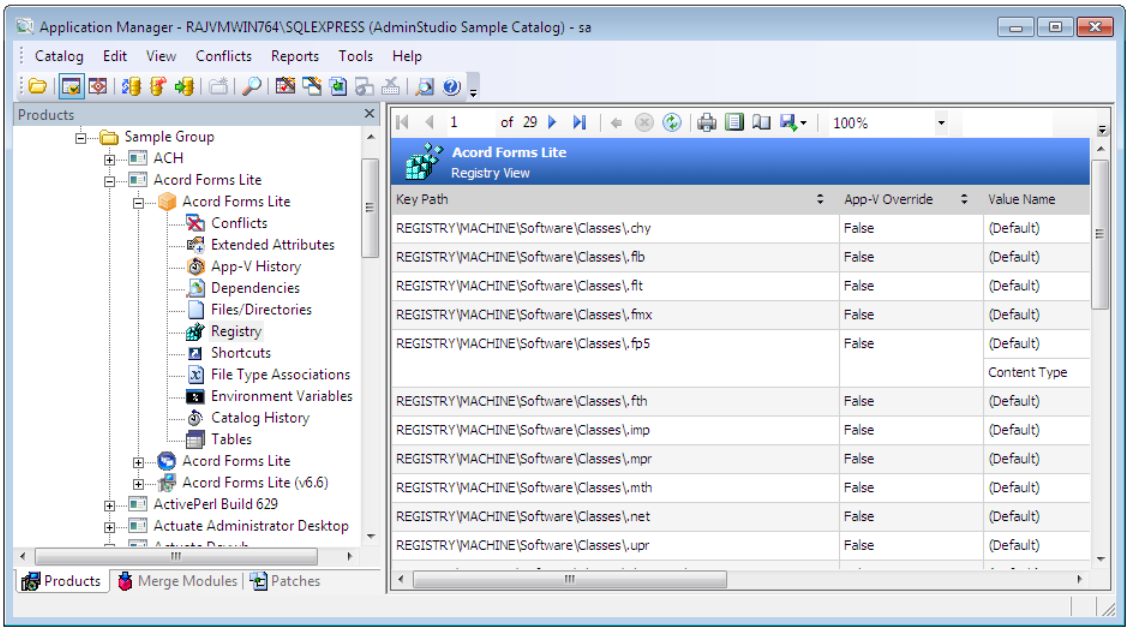
- File Size
- VFS Path
- Feature Block 1
- App-V Version
- App-V Data Type

Viewing App-V Package Registry

The App-V **Registry View** lists any registry entries created or changed by the App-V package. To view the App-V **Registry View**, perform the following steps.

Task: ***To view App-V package Registry information:***

1. Open **Application Manager**.
2. In the tree, expand an App-V package and select the **Registry** node. The **App-V Registry View** opens.



For each registry entry, the following information is listed:

- Key Path
- App-V Override
- Value Name
- Data

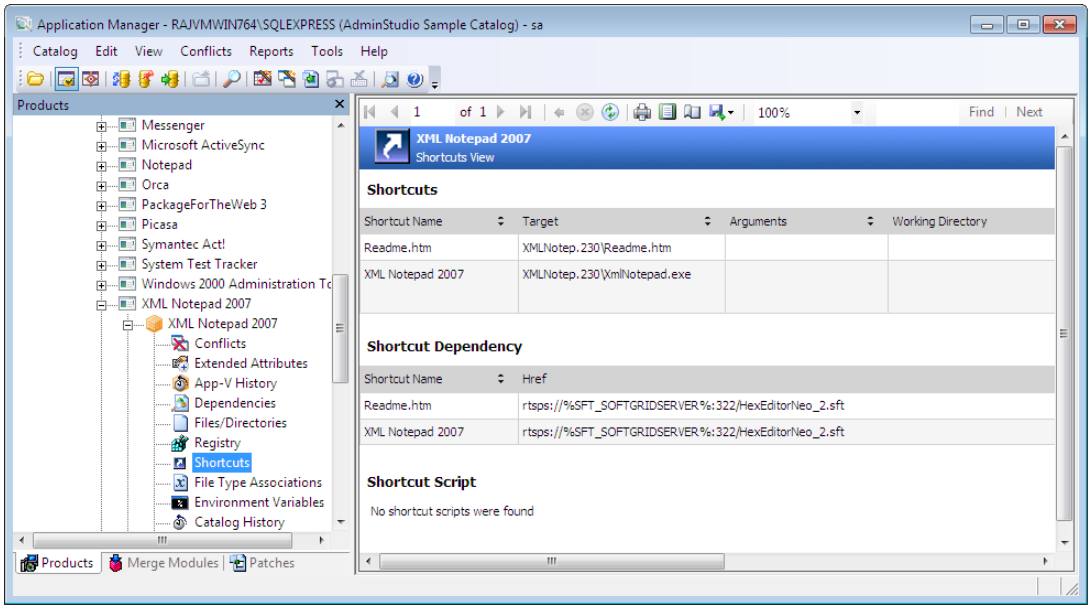
- Type

Viewing App-V Package Shortcuts

The App-V **Shortcuts View** lists any shortcuts created by the App-V package. To view the App-V **Shortcuts View**, perform the following steps.

Task: ***To view App-V package shortcuts:***

1. Open **Application Manager**.
2. In the tree, expand an App-V package and select the **Shortcuts** node. The **Shortcuts View** opens.



Under the **Shortcuts** subheading, the following information is listed for each shortcut:

- Shortcut Name
- Target
- Arguments
- Working Directory
- Target Version
- Location

Under the **Shortcut Dependency** subheading, the following information is listed for each shortcut dependency:

- Shortcut Name

- Href
- GUID
- Is Mandatory

Under the **Shortcut Script** subheading, the following information is listed for each shortcut script:

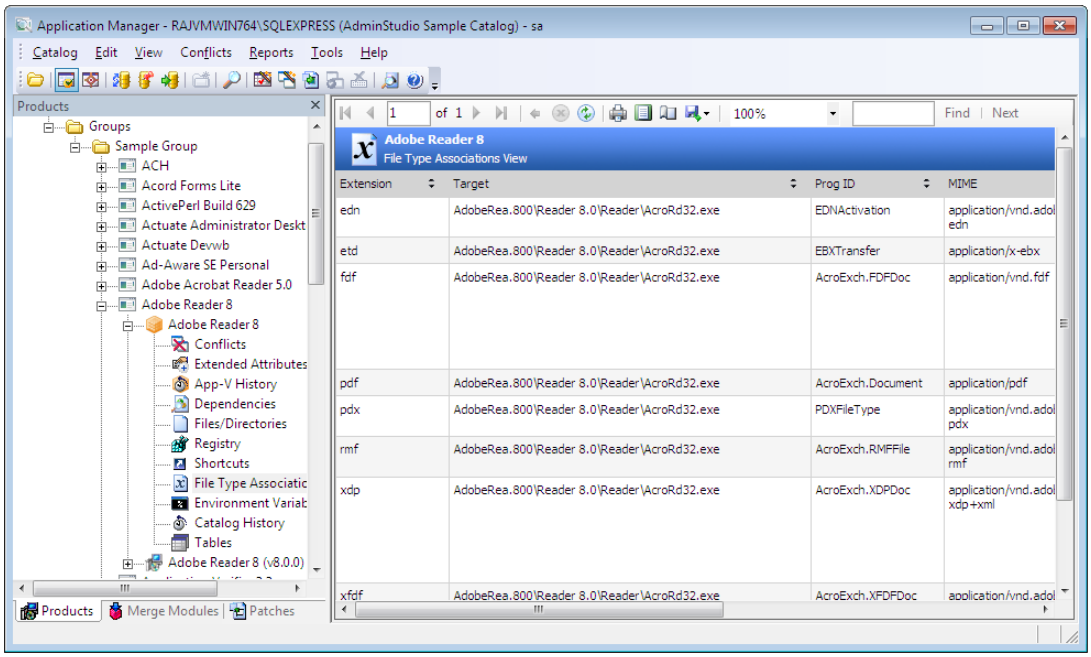
- Shortcut Name
- Body (of script)

Viewing App-V Package File Type Associations

The App-V **File Type Associations View** lists an App-V package's file type associations. To view the App-V **File Type Associations View**, perform the following steps.

Task: ***To view App-V package file type associations:***

1. Open **Application Manager**.
2. In the tree, expand an App-V node and select the **File Type Associations** node. The **File Type Associations View** opens.



The following information is listed for each file type association:

- Extension
- Target

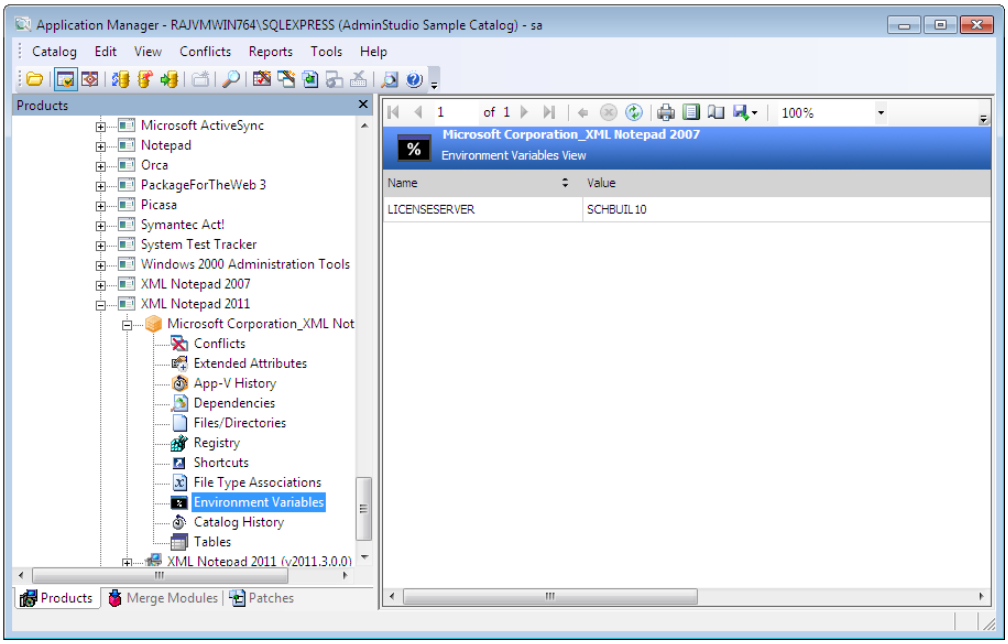
- Prog ID
- MIME
- Description
- Verb
- Arguments

Viewing App-V Package Environment Variables

The App-V **Environment Variables View** lists an App-V package's environment variables. To view the App-V **Environment Variables View**, perform the following steps.

Task: ***To view App-V package file type associations:***

1. Open **Application Manager**.
2. In the tree, expand an App-V node and select the **Environment Variables** node. The **Environment Variables View** opens.



The following information is listed for each variable:

- Name
- Value

Using the Software Repository

Edition • *The Software Repository feature is available in AdminStudio Enterprise Edition.*

A Windows Installer package is made up of many files that are executed when the setup is run. However, only the `.msi` file is imported into the Application Catalog database. Also, an App-V package is made up of many files that are used when the virtual application is run. However, only the `.sft` file is imported into the Application Catalog database.

To safeguard these additional files against alteration or being misplaced, you can choose to manage Windows Installer and Microsoft App-V packages using the Software Repository. The Software Repository gives you a secure, transparent storage system for your AdminStudio data, especially Windows Installer and App-V packages used in the enterprise.

The AdminStudio tools are tightly integrated into the Software Repository, particularly Application Manager and InstallShield Editor. Certain concepts within AdminStudio itself require the Software Repository in order to operate:

- **Package version management and history**—You can store multiple versions of a package in the Software Repository. Instead of reimporting a package that has changed, you can check out a package and then check it back in either as a new package version or you can overwrite the existing version. When a package is checked out, it cannot be modified by another user. You can also view version history and include it in reports. See Using Version Management Features and Viewing Package Catalog History for more information.
- **Ad-hoc import of transform and patch files**—You can perform an ad-hoc import of transform and patch files into the Application Catalog and Application manager will determine which software package in the Application Catalog the transform or patch file is associated with. The file will then be stored in the Software Repository along with that package's other associated files. See Performing an Ad-Hoc Import of Transform Files or Patch Files for more information.

When a Windows Installer or App-V package is managed within the Software Repository, its `.msi` or `.sft` file and all of its other associated files and subfolders are imported into a subfolder of the Software Repository location identified for that Application Catalog.

Information on the AdminStudio Software Repository is presented in the following sections:

- Enabling the Software Repository in Application Catalogs
- Methods to Import Packages into the Software Repository
- Identifying Software Repository Packages in Application Manager
- Using Version Management Features
- Software Repository Integration into Other AdminStudio Tools

Enabling the Software Repository in Application Catalogs

To be able to import packages into the Software Repository of an Application Catalog, the Application Catalog must first have the Software Repository enabled. You can choose to enable the Software Repository feature when you are creating a new Application Catalog, and you can also enable it in existing Application Catalogs.

When an Application Catalog is enabled for the Software Repository, a Software Repository directory location must be specified.

When an Application Catalog has the Software Repository feature enabled, that does not mean that all of its existing packages and all newly imported packages will be automatically added to the Software Repository. Each time you import a package, you can choose whether you want to add it to the Software Repository.

Note • *If a package is not added to the Software Repository when it is imported, the only way to add it to the Software Repository after that is to import it again, and to choose to overwrite the existing package version.*

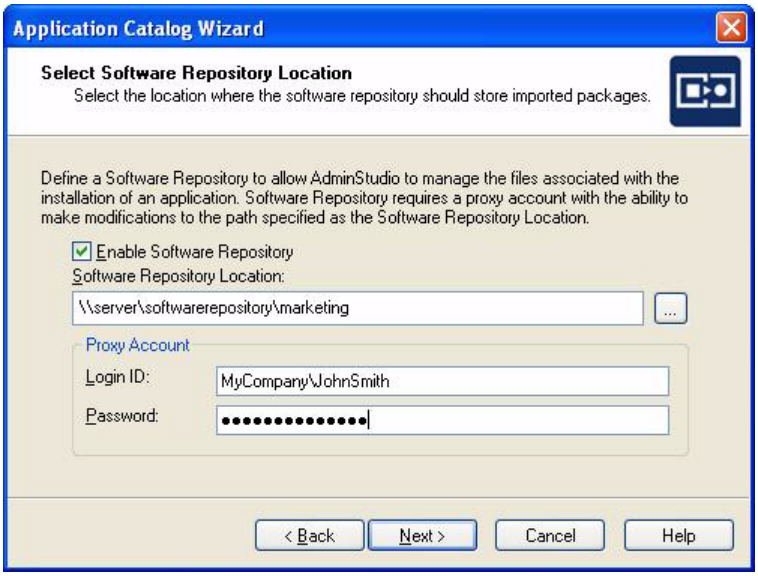
Enabling the Software Repository in a New Application Catalog

To enable the Software Repository in a new Application Catalog, perform the following steps.

Task: ***To enable the Software Repository in a new Application Catalog:***

Create the new Application Catalog as described in Creating New Application Catalogs, but on the Select Software Repository Location Panel, do the following:

1. Choose the **Enable the Software Repository** option.
2. Select a **Software Repository Location**.
3. Enter a Login ID and Password of the **Proxy Account** that will be used to access this location.



Enabling the Software Repository in an Existing Application Catalog

To enable the Software Repository feature in an existing Application Catalog, perform the following steps.

Task: ***To enable the Software Repository in an existing Application Catalog:***

1. Open the Application Catalog to which you want add Software Repository support in AdminStudio, ConflictSolver, or Application Manager.
2. On the **Catalog** menu, click **Properties**. The **Application Catalog Properties** dialog box opens.
3. Open the **Software Repository** tab.
4. Select the **Enable Software Repository** check box. The **Software Repository Location** option is enabled.

5. Browse to the location where you want to store the Software Repository for this Application Catalog.
6. Under **Proxy Account**, specify a **Login ID** and **Password** that will be used to access this location.
7. Click **OK**.

Methods to Import Packages into the Software Repository

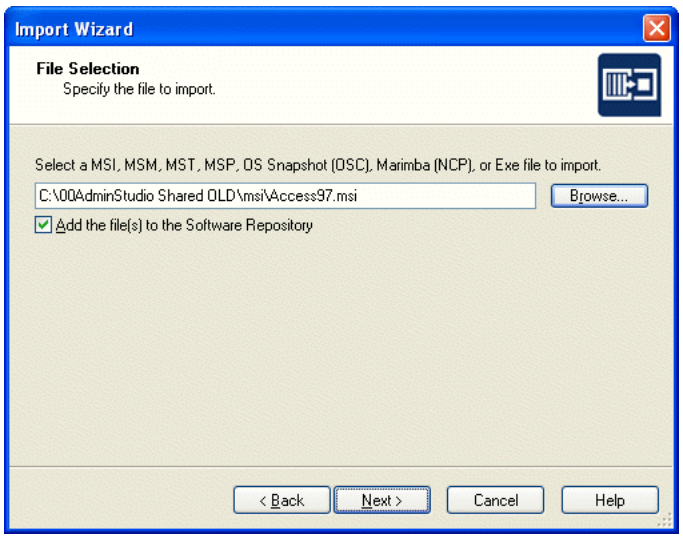
You can import packages into the Software Repository using the following methods:

- Importing a Package into the Software Repository Using the Import Wizard
- Adding a Package to the Software Repository by Overwriting Original Package
- Software Repository Auto Import Method

Importing a Package into the Software Repository Using the Import Wizard

When you import a package into the Software Repository, all of its associated installation files are stored in the Software Repository location, not just the installation package itself.

To have the package and its associated files managed within the Software Repository, launch the Import Wizard as described in Importing Data, and when the **File Selection** panel opens, select the **Add the file(s) to the Software Repository** option. If you are importing Merge Modules on the **Merge Modules** tab, this option is on the **MSM Source Information** panel.



When you click **Finish** on the Import Wizard, the selected package is imported into the Application Catalog, and all of that package's associated files and subfolders are copied into the Software Repository location specified for the Application Catalog. Progress messages are displayed in the Output Window stating that each file is being copied into the Software Repository.

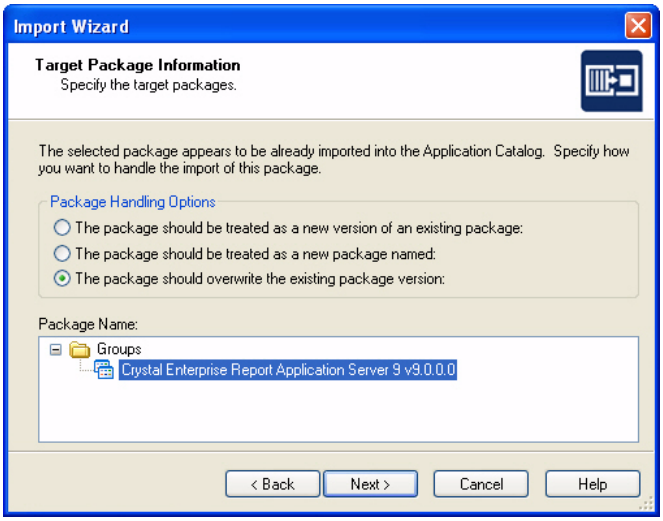
Adding a Package to the Software Repository by Overwriting Original Package

If a package is not added to the Software Repository when it is imported, the only way to add it to the Software Repository after that is to reimport it and overwrite the original package (assuming that the Application Catalog is Software Repository-enabled).

Task: ***To add a package to the Software Repository by overwriting the original package:***

Reimport the package again by following the instructions in Importing a Single Package, but this time:

1. Choose the **Add the file to the Software Repository** option on the **File Selection** panel
2. Choose the **The package should overwrite the existing package version** option on the **Target Package Information** panel:



Note • *For more information on managing multiple versions of a package in the Software Repository, see Using Version Management Features.*

Software Repository Auto Import Method

Use the following command line options to perform a bulk import of Windows Installer packages into the Software Repository:

```
[General]
ImportIntoTheSoftwareRepository=Y
SuppressDuplicatePackageImport=N
```

Identifying Software Repository Packages in Application Manager

In Application Manager, packages that are managed within the Software Repository have a different icon than those that are not:

Table 6-13 • Application Manager Product View Icons Identifying Software Repository

Icon	Description
	Package is not managed within the Software Repository.
	Package is managed within the Software Repository.
	Package is managed within the Software Repository and is checked out.
	Merge Module is managed within the Software Repository.

Note • *Similar overlays are displayed on App-V package icons that are listed under the **Application** node.*

Note • *These icons are not displayed in when viewing these packages in ConflictSolver.*

Also, when a Software Repository package is selected in the Application Manager **Product View**, the statement `Managed within the Software Repository` is listed for the **File** and **Original File** fields. Below is an example of a Software Repository package in the Application Manager Product View:

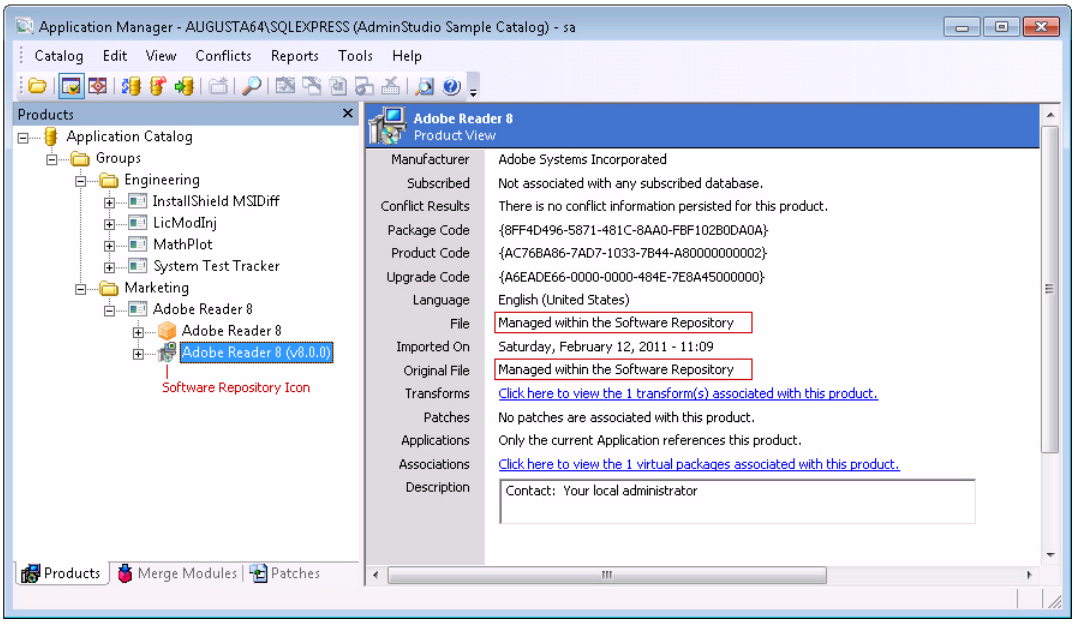


Figure 6-17: MSI Package Managed Within the Software Repository

Using Version Management Features

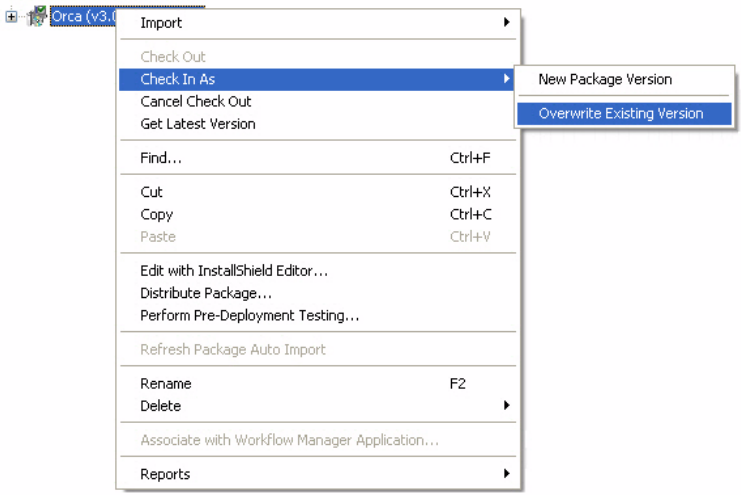
For those packages which are part of the Software Repository, you can store multiple versions of a package. Instead of reimporting a package that has changed, you can check out a package and then check it back in either as a New Package Version or you can Overwrite Existing Version.

The Software Repository version management features are described in the following topics:

- Checking-Out and Checking-In Packages
- Cancel Check Out
- Getting a Copy of the Latest Version of a Package
- Viewing Package Version History

Checking-Out and Checking-In Packages

For those packages which are part of the Software Repository, you can store multiple versions of a package. Instead of reimporting a package that has changed, you can check out a package and then check it back in either as a New Package Version or you can Overwrite Existing Version.



When you check out a package, AdminStudio makes a local copy of the package files into the following directory:

`C:\Documents and Settings\UserName\My Documents\AdminStudio\PackageName_PackageVersion`

When a package is checked out, an icon is displayed on the Application Manager Product View to indicate that the package is checked out. If the current user is not the user who checked out the package, then the path portion of the property is not displayed.

When you check in a package, it will be imported into the Application Catalog and either the new package will be added as a new version of an existing package or the old package will be overwritten.

Cancel Check Out

If you checked out a package and then decide that you do not want to modify it, you can choose to cancel the check out by selecting the package, and then selecting **Cancel Check Out** from the context menu.

Unless a user is assigned to a Role with advanced Software Repository permissions, only the user who checked out the package will have the **Cancel Check Out** enabled. After the user confirms the operation, the local package files are deleted.

Getting a Copy of the Latest Version of a Package

If you want to get a copy of a package that is in the Software Repository, but you do not want to check it out, you can get a copy of that package by selecting the package, and then selecting **Get Latest Version** from the context menu. The package and all of its associated files will then be copied to the user's profile directory:

`C:\Documents and Settings\`*`UserName`*`\My Documents\AdminStudio\`*`PackageName`*

For example:

`C:\Documents and Settings\JohnSmith\My Documents\AdminStudio\Adobe Reader`

Note • *The location where Software Repository files are copied to when you select* ***Get Latest Version*** *cannot be changed. The files are always copied to a subdirectory of the current user's* `My Documents` *directory.*

Viewing Package Version History

If you are connected to a Software Repository-enabled Application Catalog and you select a package that has more than one version, a **View Package Version History** link appears in the header of the view.

If you click on this link, the **Package Versions** dialog box opens, listing all of the versions of the selected package.

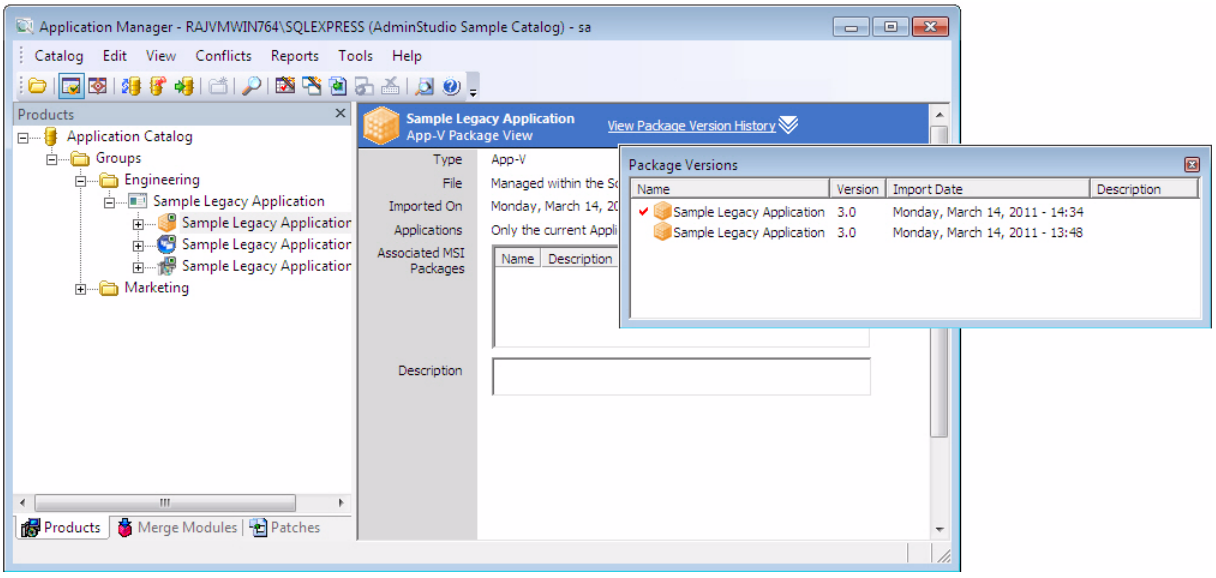


Figure 6-18: View Package Version History

Software Repository Integration into Other AdminStudio Tools

The Software Repository feature is integrated into several other AdminStudio tools:

- InstallShield Editor
- ConflictSolver
- PackageExpert
- Distribution Wizard
- Predeployment Test Preparation Wizard
- Automated Application Converter

InstallShield Editor

The Software Repository feature is integrated with InstallShield Editor in the following ways:

- You can launch InstallShield Editor from the Application Manager.
- From InstallShield Editor, you can browse for packages that are stored in the Software Repository and select them for edit. You can either check a file out for edit or simply get the latest version of the file to edit. You can also undo a check out from InstallShield Editor.
- From the InstallShield Editor, you can add a package to the Software Repository.
- You can add a package to the Software Repository via the InstallShield Editor build process.

For more information on InstallShield Editor's integration with AdminStudio, see InstallShield Editor Integration with Application Manager and the Software Repository.

ConflictSolver

If you are performing conflict analysis on packages that are a part of the Software Repository, the resolution process will only operate against Checked Out packages by the current user. If a package is not checked out, you will be unable to automatically resolve any conflicts that are found or to modify the package in any way.

PackageExpert

If you are testing packages that are a part of the Software Repository, the resolution process will only operate against Checked Out packages by the current user. If a package is not checked out, you will be unable to automatically resolve any errors that are found or to modify the package in any way.

Distribution Wizard

You can launch the Distribution Wizard from Application Manager by selecting the package and then selecting **Distribute Package** from the context menu. When you do this, the package name in the **Windows Installer Package (.msi)** field of the **Installation Package Information** panel is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository:

- **Not in the Software Repository**—The full name and path of the Windows Installer file is displayed, and you can edit this entry or click **Browse** and select a different package.
- **In the Software Repository**—Only the name of the Windows Installer file is displayed (not the full path) and this entry cannot be edited or changed.

Predeployment Test Preparation Wizard

You can launch the Predeployment Test Preparation Wizard from Application Manager by selecting the package and then selecting **Perform Predeployment Testing** from the context menu. When you do this, the name in the **Source .MSI Package** field of the **Windows Installer File Selection** panel is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository:

- **Not in the Software Repository**—The full name and path of the Windows Installer file is displayed, and you can edit this entry or click **Browse** and select a different package.
- **In the Software Repository**—Only the name of the Windows Installer file is displayed (not the full path) and this entry cannot be edited or changed.

The **Target .MSI Package** field on the same panel is automatically populated with the name of the package specified in the **Source .MSI Package** field with `_analysis` appended to the end of the name. You can edit the name of this file, if desired. If the package is not in the Software Repository, the full name and path of the **Target .MSI Package** is displayed, but if the package is in the Software Repository, only the name of the **Target .MSI Package** is displayed (not the full path).

Automated Application Converter

When publishing packages from Automated Application Converter to the Application Catalog, if you select the **Add the file(s) to the Software Repository** option on the **Select Group** panel of the **Package Publish Wizard**, the files will be added to the Software Repository.

Sharing Application Catalog Data

The section describes the three methods for sharing Application Catalog data in your enterprise: merge, auto-import, and replication.

Table 6-14 • Sharing Application Catalog Data Help Topics

Section	Description
Merging Application Catalogs Using the Merge Wizard	You can use the Merge Wizard to merge data from a source Application Catalog into the currently open Application Catalog.
Automatically Importing Packages	You can use the Package Auto Import feature to link to Windows Installer and App-V packages in a Remote Application Catalog, import the data into your local Application Catalog, and maintain consistency between the packages.
Replicating Application Catalogs	Using Application Catalog Replication simplifies software management by allowing you to selectively distribute subsets of data from a master Application Catalog to multiple IT groups at your organization.
Using Microsoft SQL Server to Perform Merge Replication of Application Catalogs	AdminStudio supports merge replication of Application Catalog databases using Microsoft SQL Server 2005. You can use merge replication to synchronize Application Catalog databases distributed in multiple locations across a large enterprise, ensuring that all databases are kept up-to-date.

AdminStudio provides three options for sharing Application Catalog Data:

- **Merging Application Catalog Information**—Use the **Merge Wizard** to merge the source Application Catalog's data into the currently open Application Catalog. The merged data is not linked in any way to the source Application Catalog.
- **Automatically Importing Packages**—The **Package Auto Import Wizard** offers two methods of automatically importing Windows Installer and App-V packages into your Application Catalog:
  - **Linking to Packages in a Remote Application Catalog**—Using the Remote Application Catalog option of the Package Auto Import Wizard, you can monitor Windows Installer and App-V packages in a Remote Application Catalog, and automatically import or re-import those packages at scheduled intervals. Application Manager can link to one or multiple Application Catalogs, and can link to one or multiple *packages* within those Application Catalogs.
  - **Automatically Importing Packages from a Network Directory**—Using the Network Directory option of the Package Auto Import Wizard, you can automatically import or re-import all Windows Installer and App-V packages in a specific folder into your Application Catalog at scheduled intervals.
- **Replicating Application Catalogs**—Use Application Catalog Replication's **Publication** and **Subscription** Wizards to distribute an Application Catalog database where it is needed across a large enterprise or multi-national corporation. Data is replicated between a Publisher Database Server and a Subscriber Database Server, which do not need to be available on the same network.

Comparison Chart

The following chart compares the three methods of sharing Application Catalog data:

Table 6-15 • Sharing Application Catalog Data Comparison Chart

Topic	Merging (Merge Wizard)	Auto Import (Package Auto Import)	Replicating (Subscription and Publication)
AdminStudio Edition	Professional Edition	Professional Edition	Enterprise Edition
Configuration	Connection between two Application Catalog databases on the same network/domain.	Connection between two Application Catalogs Databases on the same network/domain or between an Application Catalog and a directory on the same network/domain.	Connection between: Publisher and Publisher Database Server, Publisher Database Server and Subscriber Database Server, and Subscriber and Subscriber Database Server
How It Works	The source Application Catalog's data is merged into the currently open Application Catalog. This is a one-time-only database merge. The merged data is *not* linked in any way to the source Application Catalog.	With the Remote Application Catalog option, you link to individual packages in a remote Application Catalog, and these packages are imported into your local Application Catalog. When the remote packages are modified, the packages in your local Application Catalog are automatically re-imported. When using the Network Directory option, if a modified version of a previously imported package is placed in the monitored Network Directory, the package is automatically re-imported.	An Application Catalog database is distributed across a large enterprise. Data is replicated between the Publisher Database Server and Subscriber Database Server, which do not need to be available on the same network.

Table 6-15 • Sharing Application Catalog Data Comparison Chart (cont.)

Topic	Merging (Merge Wizard)	Auto Import (Package Auto Import)	Replicating (Subscription and Publication)
Customers Who Would Use This	Merging could be used in an organization where multiple people test packages. After the packages have been tested, all of the "tested" applications could be consolidated into one centralized Application Catalog.	Package Auto Import could be used to maintain consistency between applications in various Application Catalogs in organizations where they are satisfied with domain security and have both source and target Application Catalog databases (or source directory and target Application Catalog database) on an accessible domain.	Replication could be used at large enterprises that want to maintain a central Application Catalog, with smaller Application Catalogs at individual offices (such as New York, London, etc.). This method moves the data closer to the user (which improves performance), and allows the user to subscribe to only the data that they need (which makes it easier to work with).

Table 6-15 • Sharing Application Catalog Data Comparison Chart (cont.)

Topic	Merging (Merge Wizard)	Auto Import (Package Auto Import)	Replicating (Subscription and Publication)
Advantages	Very simple. Data is merged in with the data in the local Application Catalog.	Can select the packages you want to link to. Linked applications can be updated to match the source. Automatically imported data coexists with data in local Application Catalog.	Replication provides optimal security. Publisher Database Server and Subscriber Database Server do not need to be available on the same network. Operates within the SQL Server Security Model, making it a very secure way to replicate data—because the subscriber need not have access to the remote database. AdminStudio encapsulates functionality from SQL Server to support “Snapshot” type replication. Using AdminStudio to perform Snapshot replication is easier than trying to perform it using SQL Server Enterprise Manager. Can select individual packages in a Publication to subscribe to. Replication is very efficient. It operates within the SQL Server Job Refresh Model, which means that the replication/ subscription process is run automatically. Automatically imported data coexists with data in the local Application Catalog.

Table 6-15 • Sharing Application Catalog Data Comparison Chart (cont.)

Topic	Merging (Merge Wizard)	Auto Import (Package Auto Import)	Replicating (Subscription and Publication)
Disadvantages	Local user needs a physical connection to the database and has to have permissions on that remote database. This could allow the user performing the merge to modify the source database. This could cause a Security issue. This is a one time merge with no link to the source data and no updating.	When using the Remote Application Catalog option, the local user needs a physical connection to the database and has to have permissions on that remote database. This could allow the user performing the link to modify the source database. This could cause a Security issue.	You can subscribe to multiple subscriptions, but only one subscription can be active at any one time.

Merging Application Catalogs Using the Merge Wizard

In some circumstances, you may need to merge data from two Application Catalogs. The reasons for this can vary, but usually involve increasing the number of packages available for conflict identification or the necessity of having the same data available in multiple geographic locations. You can merge data from two different Application Catalogs using the Merge Wizard.

Note • *You can use the Merge Wizard to merge Windows Installer packages (.msi), legacy applications (.exe), and Microsoft App-V applications (.sft). However, the Merge Wizard does not support merging Citrix XenApp profiles or VMware ThinApp applications.*

Task: ***To merge data from two Application Catalogs:***

1. Open **Application Manager** and connect to the Application Catalog that is to contain the combined data.
2. On the **Catalog** menu, click **Merge** to open the **Merge Wizard**. The **Welcome** panel opens.
3. Click **Next**. The **Source Application Catalog** panel opens.
4. Select the Application Catalog that contains the data that you want to merge.

   If the Application Catalog that you select requires additional authentication, the **Select Application Catalog** dialog box opens. Connect to that catalog as described in Connecting to an Existing Application Catalog.
5. Click **Next** to advance to the **Groups** panel.
6. Select the Windows Installer packages, legacy applications, and App-V applications that you want to merge. Selecting a Group icon selects all of the applications in that group. You can also select to **Replace duplicate application data**.

7. Click **Next**. The **Progress** panel opens, where you can view the merge progress.
8. Click **Finish** to exit the Wizard.

Note • *For a comparison of the three methods for sharing Application Catalog Data in AdminStudio, see Sharing Application Catalog Data.*

Automatically Importing Packages

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

For a large distributed enterprise, accurate and synchronized data is required for shared Windows Installer and Microsoft App-V packages. If a corporate level team maintains a database of packages used throughout the corporation in New York, then all the regional offices should be able to link to the corporate level packages and get the different pieces of data.

Package Auto Import offers two methods of automatically importing Windows Installer and Microsoft App-V packages into your Application Catalog and keeping them updated:

Table 6-16 • Package Auto Import Methods

Method	Description
Automatically Importing Packages in a Remote Application Catalog	Using the **Remote Application Catalog** option of the Package Auto Import Wizard, you can monitor Windows Installer and App-V packages in a Remote Application Catalog, and automatically import or re-import those packages at scheduled intervals. Application Manager can link to one or multiple Application Catalogs, and can link to one or multiple *packages* within those Application Catalogs.
Automatically Importing Packages from a Network Directory	Using the **Network Directory** option of the Package Auto Import Wizard, you can automatically import or re-import all Windows Installer and App-V packages in a specific Network Directory into your Application Catalog at scheduled intervals.

Caution • *When automatically importing or re-importing packages from a Remote Application Catalog or a Network Directory using Package Auto Import, validation will not be performed even if the Check for Validation Errors option on the **Import** tab of the ConflictSolver or Application Manager Options dialog box is selected. If validation was performed during automatic import and if a package failed validation, it would not be imported into the Application Catalog, which would disrupt the automatic import process. Instead, you can perform validation after the package has been imported. See Validating After Import.*

Tip *• When you link to a Windows Installer or App-V package in a Remote Application Catalog, Application Manager connects to the Remote Application Catalog and imports that package into the local Application Catalog. In order for Package Auto Import to work properly, both the Remote Application Catalog and the Remote package file must be accessible to the linked users at all times. In other words, both the Remote Application Catalog and all of the package files that have been imported into it must be able to be accessed by linked users via a shared network drive.*

Note *• You can use the Package Auto Import Wizard to automatically import Windows Installer packages (.msi) and Microsoft App-V applications (.sft). However, the Package Auto Import Wizard does not support the automatic import of Citrix XenApp profiles or VMware ThinApp applications.*

Note *• For a comparison of the different methods for sharing Application Catalog Data in AdminStudio, see Sharing Application Catalog Data.*

Automatically Importing Packages in a Remote Application Catalog

Note *• To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

Using the Remote Application Catalog option of Package Auto Import, you can monitor Windows Installer and App-V packages in a Remote Application Catalog, and automatically import or re-import those packages at scheduled intervals.

The following topics are included in this section:

- Linking to a Remote Application Catalog
- Editing a Link to a Remote Application Catalog
- Manually Updating a Linked Package
- Deleting a Link to a Remote Application Catalog
- Viewing Remote Package Links on the Application Manager Product View

Linking to a Remote Application Catalog

Using the **Remote Application Catalog** option of **Package Auto Import**, you can monitor Windows Installer and App-V packages in a Remote Application Catalog, and automatically import or re-import those packages at scheduled intervals. Follow these procedures to configure a link to a Remote Application Catalog.

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

Task: ***To link to a Remote Application Catalog:***

1. Open **Application Manager**.
2. On the **Catalog** menu, select **Package Auto Import**. The **Package Auto Import dialog box** opens.
3. Click **Add**. The **Package Auto Import Wizard Welcome** panel opens.
4. Click **Next**. The **Package Auto Import Type Selection** panel opens, prompting you to specify whether you want to monitor packages in a **Remote Application Catalog** or monitor a **Network Directory**.
5. Select **Remote Application Catalog** and click **Next**. The **Remote Link Application Catalog** panel opens and lists Application Catalogs that you have recently accessed.
6. From the **Catalogs** list, select the Application Catalog you want to link to or click **Browse** to open the **Select Application Catalog** dialog box.

   Connect to the selected catalog as described in Connecting to an Existing Application Catalog.
7. Click **Next**. The **Groups** panel opens, listing all of the applications in the selected Application Catalog.
8. Select the Windows Installer and App-V packages within the Application Catalog that you want to link to.
9. Click **Next**. The **Schedule** panel opens.
10. To enable automatic import of the linked packages at scheduled intervals, select the **Schedule AdminStudio to update the changed packages** option. The scheduling fields are enabled.
11. From the **Run the process list**, select **Daily** or **Weekly**.
12. Select the time of day that you want the automatic import to occur by selecting the hour, minutes, seconds, or AM/PM setting and then clicking the up and down arrows to adjust the setting.
13. From the **Starting on** list, click on the arrow to open the Calendar and then select the first day that you want automatic importing to begin.
14. Click **Next**. The **Summary** panel opens.
15. Review the summary of the options that were selected in the previous panels of the Wizard and click **Finish**. You are returned to the **Package Auto Import** dialog box and the new **Remote Application Catalog** link is now listed.

Note • *Note that on the **Package Auto Import** dialog box, the **Start** button is disabled when this new Remote Application Catalog link is selected. This button is disabled until the linked packages have been imported into the local Application Catalog for the first time. The first automatic import of linked packages in a Remote Application Catalog is initiated when you click the **Close** button to exit the Package Auto Import dialog box.*

16. Click **Close** to close the Package Auto Import dialog box.

Initial and Subsequent Automatic Imports

Import is initiated automatically when you set up the Remote Application Catalog connection, and then subsequent imports occur automatically at the scheduled date and time.

Initial Automatic Import

When you initially set up a connection to a Remote Application Catalog, automatic import is initiated when you click **Close** to exit the Package Auto Import dialog box. Import results are displayed in the **Output** window. The packages are imported into the **Package Auto Import Group** of the Application Catalog.

Subsequent Automatic Imports

After a package has been imported into the Application Catalog, an update check will be performed:

- At the scheduled interval (if the **Schedule AdminStudio to update the changed packages** option is selected), *and*
- Whenever you click **Start** on the **Package Auto Import dialog box**.

Preferred Authentication Modes

AdminStudio has preferred authentication modes when connecting to SQL Server Application Catalogs.

When you open AdminStudio, by default AdminStudio will attempt to connect to your SQL Server Application Catalog using NT authentication mode. If that fails, AdminStudio will display a login dialog box with the Server, Login ID, and Database fields pre-populated with your last selection.

If you use SQL Server Authentication Mode to login to your SQL Server Application Catalog, AdminStudio will not save SQL login information. Therefore, each time you want to refresh the data from this Application Catalog, you will have to supply AdminStudio with the login information. Therefore, it is recommended that you use NT authentication mode to login to SQL Server Application Catalogs.

Editing a Link to a Remote Application Catalog

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

On the Groups tab of the Package Auto Import Properties (Remote Application Catalog) Dialog Box, you can link or unlink to different packages *within* an Application Catalog, and on the **Options** tab, you can modify update options and adjust the scheduled update interval.

Task: ***To edit a Remote Application Catalog link:***

1. On the **Catalog** menu, select **Package Auto Import**. The **Package Auto Import** dialog box opens.
2. Select the **Remote Application Catalog** link that you would like to edit and click **Properties**.

   The **Groups** tab of the **Package Auto Import Properties (Remote Application Catalog)** dialog box opens, displaying all of the groups, applications, and deployment type packages in the selected Remote Application Catalog that you can link to. Any package that you have already linked to is already selected.
3. Select or deselect check boxes to link or unlink to packages within this Remote Application Catalog.
4. If you want to specify options that directly affect the way the linked packages will be updated, select the **Options** tab.
5. On the **Options** tab, select one or more of the following **Refresh Options**:
   - **Update package when its source is modified**
   - **Delete package when the source package is deleted**

   See Manually Updating a Linked Package for a detailed explanation of these two options.
6. To enable automatic import of the linked packages at scheduled intervals, select the **Schedule** option. The scheduling fields are enabled.

   If you want to disable automatic importing of this package if it is modified in its Remote Application Catalog, leave the **Schedule** option unselected. With this option unselected, the only way that this package can be updated is to select the **Application Catalog** link on the **Package Auto Import** dialog box and then click **Start** to force an update.
7. From the **Run the process** list, select **Daily** or **Weekly**.
8. Select the time of day that you want the automatic import to occur by selecting the hour, minutes, seconds, or AM/PM setting and then clicking the up and down arrows to adjust the setting.
9. From the **Starting on** list, click on the arrow to open the Calendar and then select the first day that you want automatic importing to begin.
10. Click **OK** to close the **Package Auto Import Properties** dialog box and then click **Close** to close the **Package Auto Import** dialog box.

Manually Updating a Linked Package

***Note** • To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

If you select the Remote Application Catalog link on the **Package Auto Import** dialog box and then click the **Start** button, the linked packages in the *local* Application Catalog are updated with the same packages in the *remote* Application Catalog. When you click on **Start**, how the packages are refreshed depends upon which options you have selected on the Package Auto Import Properties (Remote Application Catalog) Dialog Box:

Table 6-17 • Import Options

Option	Description
Update package when its source is modified	**If you have selected this option:** Application Manager will go and check if any of the linked packages have been modified in the Application Catalog. • **If the linked package has been modified**, then the local Application Catalog database is updated to reflect the changes. • **If the linked package has not been modified**, you receive a message stating that “No refresh is necessary”. **If you have not selected this option:** Application Manager will still check to see if any of the linked packages have been modified: • **If the linked package has been modified**, Application Manager will inform you that the source package has been modified but that it will not update the local Application Catalog database because this option is not selected. • **If the linked package has not been modified**, you receive a message stating that “No refresh is necessary”.
Delete package when the source package is deleted	**If you have selected this option**, Application Manager will delete any linked data in the local database if the data is deleted from the remote database. **If you have not selected this option**, even if the source package is deleted in the remote database, your copy of the package will not be deleted.

As the linked packages are being re-imported, output messages will be displayed in the Package Auto Import tab of the Application Manager **Output** window.

Deleting a Link to a Remote Application Catalog

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

Task: ***To delete a link to a Remote Application Catalog:***

1. On the **Catalog** menu, select **Package Auto Import**. The **Package Auto Import** dialog box appears.
2. Select the **Remote Application Catalog** link that you would like to delete and click **Remove**.

Viewing Remote Package Links on the Application Manager Product View

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

The Application Manager Product View shows all the groups and packages present in an Application Catalog. The groups and packages that have been linked to using the Remote Application Catalog method of the Package Auto Import Wizard are displayed using a special icon:

If a linked product in the local Application Catalog has been updated in the Remote Application Catalog and needs to be re-imported, an exclamation point is displayed on the icon:

Automatically Importing Packages from a Network Directory

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

Using the Network Directory option of Package Auto Import, you can monitor a directory location on the network (or a local directory) and automatically import or re-import packages in that directory at scheduled intervals.

The following topics are included in this section:

- Linking to a Network Directory
- Editing a Link to a Network Directory
- Forcing an Import of Packages in a Network Directory
- Deleting a Link to a Network Directory

Linking to a Network Directory

Using the **Network Directory** option of **Package Auto Import**, you can monitor a directory location on the network (or a local directory) and automatically import or re-import packages in that directory at scheduled intervals. Follow these procedures to configure a link to a network directory.

Task: ***To link to a network directory:***

1. On the **Catalog** menu, select **Package Auto Import**. The **Package Auto Import** dialog box opens.
2. Click **Add**. The **Package Auto Import Wizard Welcome** panel opens.
3. Click **Next**. The **Package Auto Import Type Selection** panel opens, prompting you to specify whether you want to monitor packages in a **Remote Application Catalog** or monitor a **Network Directory**.
4. Select **Network Directory** and click **Next**. The **Folder** panel opens.
5. Use the Browse button to select the network directory Folder that you want to monitor.

Note • *While you are permitted to enter a local directory (rather than a network directory) as the monitored directory, this might cause problems if a person monitoring this directory has the same directory on his local machine. For example, if a directory named `C:\MyAppCatalogs` is chosen to be the monitored directory, and a person who choose to monitor this directory also has a directory on his local machine named `C:\MyAppCatalogs`, Application Manager would monitor that person's local directory rather than the directory that was chosen to be the monitored directory.*

6. To also monitor the selected directory's sub-directories, select the **Also include sub-directories** option.

   If you select this option, when packages in subdirectories are imported into your Application Catalog, they will be placed in groups that maintain the directory hierarchy. For example, if you are monitoring a directory named **Marketing**, and that directory has a subdirectory named **Graphics** that contains packages, when those packages are imported into your Application Catalog, they will be placed in the **Graphics** subgroup of the destination group.

7. Click **Next**. The **Destination Group** panel opens, prompting you to select the Application Catalog group into which the packages will be imported.

8. Select a destination group and click **Next**. The **Schedule** panel opens.

9. To enable automatic import of the linked packages at scheduled intervals, select the **Schedule AdminStudio to update the changed packages** option. The scheduling fields are enabled.

   If you do not select this option, packages will only be imported when you select this entry on the Package Auto Import dialog box and then click **Start**.

10. From the **Run the process** list, select **Daily** or **Weekly**.

11. Select the time of day that you want the automatic import to occur by selecting the hour, minutes, seconds, or AM/PM setting and then clicking the up and down arrows to adjust the setting.

12. From the **Starting on** list, click on the arrow to open the Calendar and then select the first day that you want automatic importing to begin.

13. Click **Next**. The **Summary** panel opens.

14. Review the summary of the options that were selected in the previous panels of the Wizard and click **Finish**. You are returned to the **Package Auto Import dialog box** and the new Network Directory link is now listed as a monitored activity.

15. To force an import of the packages in the monitored Network Directory, select the **Network Directory** entry and click **Start**. Import results will be displayed in the **Output** window. The packages will be imported into the specified Group in the Application Catalog:

Note • *If you do not choose to perform a forced import, packages in the monitored Network Directory will be imported at the next scheduled interval.*

Note • *To modify Network Directory link settings, select the link on the Package Auto Import Dialog Box and click* ***Properties****. See Editing a Link to a Remote Application Catalog.*

Editing a Link to a Network Directory

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

On the **Groups** tab of the **Package Auto Import Properties (Network Directory)** dialog box, you can choose to monitor subdirectories and can adjust the scheduled update interval.

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

Task: ***To edit a link to a Network Directory:***

1. On the **Catalog** menu, select **Package Auto Import**. The **Package Auto Import** dialog box opens.
2. Select the **Network Directory** link that you would like to edit and click Properties. The **Package Auto Import Properties (Network Directory)** dialog box opens.
3. To choose to monitor the selected folder's subfolders, select the **Also include sub-directories** option.

   If you select this option, when packages in subdirectories are imported into your Application Catalog, they will be placed in groups that maintain the directory hierarchy. For example, if you are monitoring a directory named `Marketing`, and that directory has a subdirectory named `Graphics` that contains packages, when those packages are imported into your Application Catalog, they will be placed in the `Graphics` subgroup of the destination group.
4. Click **Next**. The **Destination Group** panel opens, prompting you to select the Application Catalog group into which the packages will be imported.
5. Select a destination group and click **Next**. The **Schedule** panel opens.
6. To enable automatic import of the packages in the Network Directory at scheduled intervals, select the **Schedule** option. The scheduling fields are enabled.

   If you want to disable automatic importing of packages in the Network Directory, leave the **Schedule** option unselected. With this option unselected, the only way that packages in that directory will be imported into the Application Catalog is to select the **Network Directory** link on the **Package Auto Import** dialog box and then click **Start** to force an import.
7. From the **Run the process** list, select **Daily** or **Weekly**.
8. Select the time of day that you want the automatic import to occur by selecting the hour, minutes, seconds, or AM/PM setting and then clicking the up and down arrows to adjust the setting.
9. From the **Starting on** list, click on the arrow to open the Calendar and then select the first day that you want automatic importing to begin.

10. Click **OK** to close the **Package Auto Import Properties** dialog box and then click **Close** to close the **Package Auto Import** dialog box.

Forcing an Import of Packages in a Network Directory

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

Select a monitored Network Directory on the Package Auto Import dialog box and then click the **Start** button to force an import of the packages in that Network Directory. As the packages are being imported (or re-imported if they have been modified), output messages are displayed in the Package Auto Import tab of the Application Manager Output Window.

Deleting a Link to a Network Directory

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

Task: ***To delete a link to a remote package:***

1. On the **Catalog** menu, click **Package Auto Import**. The **Package Auto Import** dialog box appears.
2. Select the monitored Network Directory that you would like to delete and click **Remove**.

   The packages that have already been imported into your Application Catalog from this Network Directory will remain, but they will no longer be updated if a modified version of the package is placed in the Network Directory that they were imported from.

Package Auto Import and Duplicate Package Names

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

During manual importing of packages, if you are attempting to import a package that has been identified as a duplicate to a package already in the Application Catalog—based upon the identifiers you have selected on the **Duplicate Package** tab of the ConflictSolver and Application Manager Options Dialog Box (PackageCode, ProductCode, Product Language, ProductVersion or List of Transform Files), the Target Package Information Panel opens, proposing an automatically generated new name for the package and prompting you to accept the new name or enter one of your choosing.

However, if AdminStudio encounters a duplicate package name during auto import, the package you are importing will be automatically renamed by pre-pending the Product Name with the Manufacturer's name and, if necessary, appending the Product Name with numbers:

- **1st Package:** `Application 2004`
- **2nd Package:** `ABC Corporation_Application 2004`

The import is allowed to proceed without prompting the user.

You can edit the **Duplicate Package Naming Syntax** on the **Duplicate Package** tab of the **Options** dialog box.

Replicating Application Catalogs

Edition • *Application Catalog replication is available in AdminStudio Enterprise Edition.*

AdminStudio Application Catalog Replication makes it possible to distribute an AdminStudio Application Catalog database where it is needed across a large enterprise or multi-national corporation. One location can publish software packages to IT teams at other sites, keeping all teams current while enabling each to manage their software independently.

The following topics are included in this section:

Table 6-18 • Application Catalog Replication Topics

Topic	Description
About Application Catalog Replication	The benefits of using Replication, an overview of creating and managing Publisher and Subscriber Application Catalogs, and how to configure the AdminStudio Shared Location for Replication.
Publication/Subscription Lifecycle	Diagram illustrating how Application Catalogs are replicated.
Specialized User Scenarios	Describes how to subscribe to more than one Publication at a time, override the scheduled update day and time, and replicate Application Catalogs in controlled SQL database environments.
Creating and Managing Publications	Describes how to create, edit, publish, and delete a Publication. This section also explains how to set a Publication schedule and how to edit a Publication access list.
Creating and Managing Subscriptions	Describes how to create, enable/disable, manually update, and delete a Subscription.

Note • *Only AdminStudio users who have been assigned* `sysadmin` *privileges to the currently connected SQL database server through the SQL Enterprise Manager can create Publications or set up Subscriptions using that database. For all other users, the Application Catalog Replication options are disabled.*

About Application Catalog Replication

This section lists the benefits of using Replication, provides an overview of creating and managing Publisher and Subscriber Application Catalogs, and explains how to configure the AdminStudio Shared Location for Replication.

- Benefits of Using Application Catalog Replication
- Creating and Managing Publisher and Subscriber Application Catalogs
- Configuring the AdminStudio Shared Location for Replication
- Initial Configuration Checklist

Benefits of Using Application Catalog Replication

The benefits of using Application Catalog replication include the following:

- Simplifies Software Package Management
- Allows You to Distribute Work Geographically Across the World
- Provides Optimal Security
- Automatically Performs SQL Server Snapshot Replication

Simplifies Software Package Management

Using Application Catalog Replication simplifies software management by allowing you to selectively distribute subsets of data from a master Application Catalog to multiple IT groups at your organization.

One organization could have hundreds of applications, which would require multiple IT groups using AdminStudio to perform conflict detection, customization, repackaging, and testing. It would be unwieldy and confusing for all of those AdminStudio users to connect to the same Application Catalog.

In this situation, it would be very beneficial for one AdminStudio user to maintain a master Application Catalog in a central location, and then to replicate (distribute) subsets of the data to the other IT groups. By allowing an IT group to work with just the data that they need, performance and efficiency is improved.

Allows You to Distribute Work Geographically Across the World

For large multi-national enterprises, using Application Catalog Replication allows you to distribute work geographically across the world. A large enterprise could maintain a central Application Catalog that is updated as applications are generated and verified by multiple remote IT groups on local Application Catalogs. The verified applications could then be distributed globally from the master Application Catalog to all the distributed work groups of the enterprise.

Provides Optimal Security

Because the Publisher Database Server and Subscriber Database Server do not need to be available on the same network, Application Catalog Replication provides optimal security. Replication operates within the SQL Server Security Model, making it a very secure way to replicate data—because the Subscriber need not have any access to the remote database. To configure the Replication feature, the user must be assigned (using SQL Enterprise Manager) `sysadmin` privileges to the database server.

Automatically Performs SQL Server Snapshot Replication

AdminStudio Application Catalog Replication uses SQL Server Snapshot replication method. Snapshot replication operates within the SQL Server Job Refresh model, which means that the replication/refresh process is run automatically. Using AdminStudio to perform Snapshot replication is much easier than performing it using SQL Server Enterprise Manager.

***Note** • For more information on Snapshot Replication, see the Snapshot Replication topic on the Microsoft MSDN Web site.*

Creating and Managing Publisher and Subscriber Application Catalogs

When you create an Application Catalog, it is designated as either a Publisher or a Subscriber catalog, and Application Manager is used to manage those subscriptions and publications.

- Creating Publisher and Subscriber Application Catalogs
- Managing Publisher and Subscriber Application Catalogs

Creating Publisher and Subscriber Application Catalogs

When creating a new SQL Server Application Catalog, you must choose (by selecting a check box) whether to define the Application Catalog as a Subscriber.

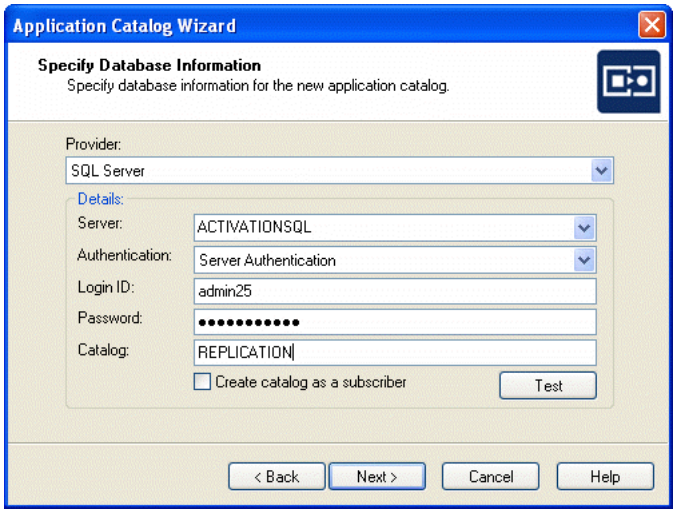


Figure 6-19: Create Database as a Subscriber Option

When this option is not selected, or when an existing database is upgraded, the Application Catalog is defined as a Publisher. The AdminStudio interface selectively enables/disables the Replication menu items based upon the type of Application Catalog that is currently opened.

Note • *You can review your distributor properties in SQL Server Enterprise Manager on the Replication tab of the SQL Server Properties dialog box.*

Managing Publisher and Subscriber Application Catalogs

In AdminStudio Application Catalog Replication, data is replicated between Publisher and Subscriber databases. You manage the Publisher and Subscriber databases using the following two dialog boxes:

- **Publication Manager dialog box**—Use to create and manage publications created on an AdminStudio Application Catalog database that is configured as a Publisher. See Managing Publications.
- **Subscription Manager dialog box**—Use to create and manage Subscriptions to Publications on an AdminStudio Application Catalog configured as a Subscriber. See Managing Subscriptions.

To see a diagram illustrating how data is replicated from the Publisher database to the Subscriber database, see Publication/Subscription Lifecycle.

***Note** • The **Publication Manager** and **Subscription Manager** options under **Replication** on the AdminStudio **Catalog** menu are only enabled when a user with* `sysadmin` *privileges on an SQL Server database server connects to a database on that server.*

- *If that database was defined as a **Publisher** when it was created, the **Publication Manager** option is enabled.*
- *If that database was defined as a **Subscriber** when it was created, the **Subscription Manager** option is enabled.*

See Application Catalog Wizard for information on how to identify databases as Subscribers or Publishers.

***Caution** • You can create as many Subscriptions or Publications as you want to, but because any previously replicated data is deleted whenever new Subscription data is received, the Subscriber should have only one Subscription active at any one time. If you want to have more than one Subscription set up at one time and switch between them, see Using the "Disable" Feature to Subscribe to More Than One Publication for more information.*

Configuring the AdminStudio Shared Location for Replication

When Publishing data using Application Catalog Replication, the Publication Server needs access to two script files that are installed in the AdminStudio Shared Directory Location. Therefore, the AdminStudio Shared directory location should refer to a network folder, making it accessible to the Publication Server. See Specifying the Shared AdminStudio Application Catalog.

Initial Configuration Checklist

Before you can use Application Catalog Replication, you need to perform the following steps to configure the Subscriber and Distributor databases at your organization. If the Subscriber and Distributor databases are not configured in this manner, Replication will fail.

***Note** • The Publisher server is often configured as its own Distributor server.*

Task:** **To configure the Subscriber and Distributor servers:

1. Check the SQL Server Agent logins configured on the Subscriber and Distributor databases, and make sure that the SQL Server Agent on the Subscriber database can login to the Distributor database via the Distributor SQL Server Agent using the `distributor_admin` login.
2. Configure a Distributor database for the servers that are publishing data.

Publication/Subscription Lifecycle

The Application Catalog database replication process is detailed in the diagram and notes below, including creating and publishing a Publication, and subscribing to that Publication.

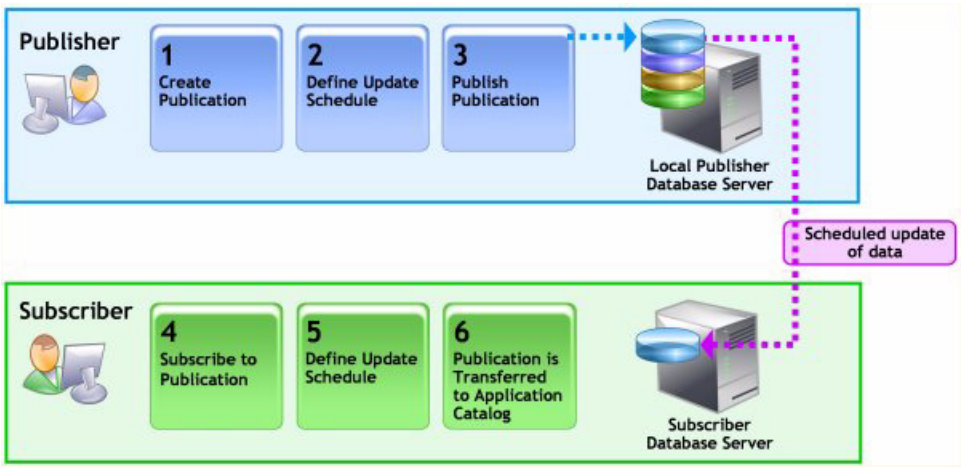


Figure 6-20: Application Catalog Database Replication Process

The following steps are included in the Application Catalog Database Replication Process:

Table 6-19 • Application Catalog Database Replication Process

Step	Description
PUBLISHER STEPS	
Step 1: Create Publication	The Publisher—an AdminStudio user who wants to make part of his Application Catalog available for Publication—creates a Publication on the local Publisher Database Server. The Publisher must have been assigned `sysadmin` privileges in SQL Enterprise Manager on the Publisher Database Server.
Step 2: Define Update Schedule	The Publisher defines the automatic update schedule for the Publication. At the specified date and time, the Publisher Database Server will check the Application Catalog to see if any published data has changed. If newer data exists, the Publication will be updated.
Step 3: Publish Publication	The Publisher generates the initial Publication by clicking the Run button on the Publication Manager dialog box. The Publication is now available for Subscription.

Table 6-19 • Application Catalog Database Replication Process (cont.)

Step	Description
SUBSCRIBER STEPS	
Step 4: Subscribe to Publication	The Subscriber—an AdminStudio user who wants to receive data from the Publisher's Application Catalog—creates a Subscription to that Publication. The Subscriber must have been assigned `sysadmin` privileges in SQL Enterprise Manager on the Subscriber Database Server.
Step 5: Define Update Schedule	The Subscriber defines the automatic update schedule for the Subscription. At the specified date and time, the Subscriber Database Server will check to see if the Subscription on the Publisher Database Server has been updated. If it has been updated, the Subscription on the Subscriber Database Server will be updated.
Step 6: Data is Copied to Application Catalog	The subscribed Publication is copied to the Subscriber Database Server, and is listed under the Replication group in the Subscriber's Application Catalog (when viewed in the Application Manager Product View).

Terminology Review

The following terminology was used in this diagram and is used throughout the Application Catalog Replication help topics:

Table 6-20 • Application Catalog Replication Terminology

Term	Definition
Publisher	Person who creates the Publication.
Local Publisher Database Server	A Database Server making data available for replication.
Publication	The actual data made available for replication.
Subscriber	Person who subscribes to a publication.
Subscriber Database Server	A Database Server that receives the replicated data.
Subscription	A request to receive a Publication from a Distributor.

Specialized User Scenarios

The following topics explain how to use the Application Catalog Database Replication feature to best meet your organization's needs. The following topics are included:

- Using the "Disable" Feature to Subscribe to More Than One Publication
- Overriding the Scheduled Update Day and Time
- Replicating Application Catalogs in Controlled SQL Database Environments

Using the "Disable" Feature to Subscribe to More Than One Publication

At your organization, it could be beneficial to have more than one Subscription configured in the Subscription Manager, allowing you to switch from one Subscription to another, even though only one Subscription can be active and listed in your Application Catalog at any one time: the one that has been executed (updated) the most recently.

You might want to set up multiple Subscriptions in a situation where an IT group wants to work on a specific group of applications at a time. For example, an IT group might want to perform conflict analysis on Accounting applications one week and on Marketing applications the next week. When the IT group finishes work on the Accounting applications, the "Accounting" Subscription would be disabled and the "Marketing" Subscription would be enabled. By having multiple Subscriptions configured, local users can quickly switch between tasks.

On the **Subscription Manager** dialog box, Subscriptions that are enabled are preceded by a check mark (✔), while disabled Subscriptions are preceded by a red X (✖).

Note • *You use the **Enable**/**Disable** button on the **Subscription Manager** dialog box to set the status of each Subscription to be either Enabled or Disabled. When you disable a Subscription, you are turning off its automatic update feature. Since Subscriptions are updated automatically, and since all of the currently active Subscription's data is replaced when a Subscription is updated, if you have multiple Subscriptions, it is to your advantage to disable all except the one you are currently using. This allows you to work on one set of data and control when that data is changed. If you have more than one Subscription enabled at a time, when the second enabled Subscription is automatically updated at its scheduled time, it will replace the data of the Subscription you were currently using.*

Overriding the Scheduled Update Day and Time

If you want to manually update either Publication data or Subscription data before its scheduled update time, you can use the Run button on the Subscription Manager and Publication Manager dialog boxes. Using the **Run** button forces the selected Publication or Subscription to refresh itself.

Replicating Application Catalogs in Controlled SQL Database Environments

Some companies that have a controlled SQL database environment prefer to have their database administrators make major changes to their databases. In this case, you could select the Generate publication as script option on the **Publication Summary Panel** of the Publication Wizard or the Generate subscription as script option on the **Subscription Summary Panel** of the Subscription Wizard so that the Wizard would generate an SQL script of the code necessary to create this Publication or Subscription rather than actually create it.

When this option is selected, the script is the only output of the Wizard and the new item is not listed on the Publication Manager or Subscription Manager. The Database Administrator could then run this script to create the Publication or Subscription.

To create Publications and Subscriptions using SQL Server Enterprise Manager, perform the following steps:

Task: ***To create publications and subscriptions using SQL Server Enterprise Manager:***

1. Use the **Publication Wizard** or **Subscription Wizard** to generate an SQL script by selecting the **Generate as script** option on the **Summary** panel.
2. Copy the script to your server.
3. Open **Query Analyzer**.
4. Select the database you want to work with.
5. Open the script file that was generated, and examine the `sp_addpublication` stored procedure to make sure that the `PreReplication.sql` and `PostReplication.sql` scripts are accessible from the system running Enterprise Manager. If they are not accessible, copy the scripts to an accessible location, and update the `sp_addpublication` stored procedure with the revised path names.
6. Execute the script.
7. Return to **SQL Server Enterprise Manager**.
8. If you created a Publication, open the database that you just generated the publication for and publish it.

Creating and Managing Publications

A Publication describes a subset of a database that you want to copy to a specific location to make it available to Subscribers. When a Publication is published, the data is copied to that accessible location. You create Publications to meet the needs of your subscribers. The Publication can be a partial Application Catalog—perhaps containing only 10 out of the 200 applications in your Application Catalog—or it can be an exact replica of your Application Catalog.

This section describes how to create, edit, publish, and delete a Publication, and also how to set a Publication schedule and edit a Publication access list. This section covers the following tasks:

- Managing Publications
- Creating a New Publication
- Publishing a Publication
- Setting a Publication Schedule
- Editing a Publication
- Editing a Publication Access List
- Deleting a Publication

Managing Publications

You create, publish, edit, and delete Publications using the **Publication Manager** dialog box.

The Publication Manager, which lists all Publications in the SQL Server database you are currently connected to, is accessed by pointing at **Replication** on the **Catalog** menu and selecting **Publication Manager**.

Note • *The **Publication Manager** option under **Replication** on the AdminStudio **Catalog** menu is only enabled if the current user is connected to an SQL Server database, has been assigned* `sysadmin` *privileges in SQL Enterprise Manager on that database server, and the database was identified when it was created as being intended for use as a **Publisher**. See Application Catalog Wizard for information on how to identify databases as Subscribers or Publishers.*

The Publication Manager lists the last time that each of the Publications was updated (published to the Publisher Database Server) and the next scheduled update time for that Publication. From the Publication Manager, you can perform the following tasks:

- **Create a New Publication**—Click the New button to create a new Publication using the **Publication Wizard**.
- **Edit an Existing Publication**—Select a Publication and click Edit to access the **Publication Properties** dialog box, where you can edit the selected Publication's description, update schedule, user access list, and the packages included in the Publication.
- **Publish a Publication**—Select a Publication and click Run to manually update the selected Publication. The Publication data is replicated from your Application Catalog to the Publisher Database Server, making it available to Subscribers.
- **Delete a Publication**—Select a Publication and click Delete to delete the Publication data from the Publisher Database Server, making it no longer available for Subscription. When the list on the Publication Manager is refreshed, the Publication will no longer be listed.

Creating a New Publication

To create a Publication, you use the **Publication Wizard** to select the packages that you want to include in the Publication, specify the users that you want to be able to access the Publication, and schedule how often you want the Publication to be automatically updated. When the Publication Wizard is complete, the SQL Server is configured for this Publication, and the name of the Publication is listed on the **Publication Manager** dialog box.

Task: ***To create a new Publication:***

1. On the Catalog menu, point to **Replication** and select **Publication Manager**. The **Publication Manager** dialog box appears.

   Note • *The **Publication Manager** option under **Replication** on the AdminStudio **Catalog** menu is only enabled if the current user is connected to an SQL Server database, has been assigned* `sysadmin` *privileges in SQL Enterprise Manager on that database server, and the database was identified when it was created as being intended for use as a **Publisher**. See **Application Catalog Wizard** for information on how to identify databases as Subscribers or Publishers.*

2. Click **New**. The **Welcome** panel of the Publication Wizard appears.
3. Click **Next**. The **Publication Details** panel appears.
4. In the **Name** text box, enter a name to identify this new Publication. This name will be listed on the Publication Manager dialog box.

   Caution • *The Publication Name cannot be a T-SQL reserved word.*

5. In the **Description** text box, enter text to describe the purpose of this new Publication. This description will be listed on the **Publication Manager** dialog box.
6. Click **Change** next to the **Schedule** text box. The **Task Scheduling** dialog box appears.
7. In the **Run the process every** list, select how frequently you want this Publication to be automatically updated on the Publisher Database Server: **Day**, **Week**, or **Month**.
   - If you selected **Week**, select the day of the week that you want to update this Publication.
   - If you selected **Month**, select the day of the month that you want to update this Publication.
8. From the Starting on list, select the day or date that you want this Publication to begin being automatically updated on the day or date you selected from the Run the process every list.

   For example, if you want the Publication to be automatically updated on the 15th of each month, you would select the first day of the first month that you want the automatic updates to begin, not the first day that the Publication should be updated. If it is Monday and you want the Publication to be updated every Friday, to have the Publication begin being updated on the upcoming Friday, you would select the current day of the week (Monday), but if you didn't want the Publication to be updated until the Friday after next, you would select Saturday.

9. Click OK to exit the Task Scheduling dialog box.
10. Click **Next**. The **Publication Data Options** panel appears.
11. Select the types of data you want to include in the Publication:
    - **Groups and Package Data**—Select this option if you want to select a specific set of packages from the various Groups in the Application Catalog to include in the Publication. If you select this option, the **Publication Packages** panel will appear, prompting you to select which packages you want to include in the Publication. If you do not select this option, all Groups and all packages within those Groups will be included in the Publication.
    - **Merge Module Data**—Select this option if you would also like to include all Merge Modules associated with the packages you have selected in the Publication. If you do not select this option, the Merge Module data will not be included in the Publication.
12. Click **Next**. The **Publication Access List** panel appears.
13. On this panel, specify the list of users allowed to access this Publication. It initially shows a list of users who have the necessary privileges to access the Publication: those users who have been assigned `sysadmin` privileges to this database server through the SQL Enterprise Manager. To remove a user, select it and click Remove.

    Caution • *Only users that have been assigned* `sysadmin` *privileges in SQL Enterprise Manager to the connected database server can establish the Publisher-Subscriber relationship between the databases and are therefore listed on this panel. You can remove a user from this list (so that they will not be able to access this Publication), but you cannot add additional users to this list using AdminStudio. To add new users, you must use SQL Enterprise Manager.*
14. Click **Next**. If you selected the **Groups and Package Data** option, the **Publication Packages** panel appears.
15. On the **Publication Packages** panel, the Groups and Packages in the Publisher Application Catalog database you are connected to are listed. Select the Groups and packages that you want to include in this new Publication and click **Next.**

    The **Publication Summary** panel appears, providing a detailed summary of the options that were selected in previous panels of the Wizard.
16. If you would like to generate an SQL script of the code necessary to create this Publication rather than actually create it, select the Generate publication as script option, and enter a path name for the script file.

    Note • *You might want to use this option if your organization has a controlled SQL database environment that prefers to have their database administrators make major changes to their databases. See Replicating Application Catalogs in Controlled SQL Database Environments for more information.*
17. To choose to publish this Publication immediately upon completion of the Publication Wizard (instead of publishing it manually or waiting until its scheduled update), select the **Publish the publication immediately after creation** option.

18. Click **Finish** to complete the creation of this Publication.
    - If you did not select the **Generate publication as script** option, the new Publication is listed on the Publication Manager.
    - If you selected the **Generate publication as script** option, you will receive a message that the script was generated properly. When you close that message, you will see that this Publication is not listed on the Publication Manager. Request that your Database Administrator run the SQL script that you just generated in order to create this Publication.
19. Proceed with steps under Publishing a Publication.

Publishing a Publication

When you publish a Publication, you are initiating a manual replication of the Publication data from your Application Catalog to the Publisher Database Server, making it available for Subscription. After its initial Publication, the Publication is also automatically "re-published" (updated) according to the schedule specified when the Publication was created.

Task: ***To publish a Publication:***

1. On the **Publication Manager**, select the Publication that you want to publish.

   Note • *If you did not select the Generate publication as script option on the* ***Publication Summary*** *panel of the Publication Wizard, the Publication should be listed on the Publication Manager as soon as the Publication Wizard finishes. If the Publication you want to publish is not listed here, make sure that the Publication Wizard was completed and, if the* ***Generate publication as script*** *option was selected, check with your Database Administrator to see if the script has been run.*

2. Click **Run**.

   The Publication data is replicated from your Application Catalog to the Publisher Database Server, making it available for Subscription.

Setting a Publication Schedule

When you create a Publication, you specify how frequently you want this Publication to be automatically updated on the Publisher Database Server. If you would like to change this frequency, you can edit the Publication from the Publication Manager dialog box.

Task: ***To set a Publication schedule:***

1. On the **Publication Manager**, select the Publication that you want to edit and click Edit. The **Publication Properties** dialog box appears.
2. Click the Details tab.
3. Click Change next to the Schedule text box. The **Task Scheduling** dialog box appears.
4. In the Run the process every list, select how frequently you want this Publication to be automatically updated on the Publisher Database Server: Day, Week, or Month.
   - If you selected Week, select the day of the week that you want to update this Publication.
   - If you selected Month, select the day of the month that you want to update this Publication.
5. From the Starting on list, select the day or date that you want this Publication to begin being automatically updated on the day or date you selected from the Run the process every list.

   For example, if you want the Publication to be automatically updated on the 15th of each month, you would select the first day of the first month that you want the automatic updates to begin, not the first day that the Publication should be updated. If it is Monday and you want the Publication to be updated every Friday, to have the Publication begin being updated on the upcoming Friday, you would select the current day of the week (Monday), but if you didn't want the Publication to be updated until the Friday after next, you would select Saturday.
6. Click OK to exit the Task Scheduling dialog box.
7. Click OK to exit the Publication Properties dialog box.

Editing a Publication

You can edit an existing Publication from the Publication Manager dialog box.

Task: ***To edit a Publication:***

1. On the **Publication Manager**, select the Publication that you want to edit and click Edit. The **Publication Properties** dialog box appears.
2. Click the Details tab.
3. In the Description text box, enter text to describe the purpose of this Publication. This description will be listed on the Publication Manager and Subscription Manager dialog boxes.

4. In the Schedule field, specify how often you would like this Publication to be automatically updated. Click the Change button to access the **Task Scheduling** dialog box, where you can edit how frequently this Publication will be automatically updated.
5. If the Groups and Package Data option was selected on the **Publication Data Options** panel when this Publication was created, you can click the Packages tab and select or unselect packages for inclusion in the Publication.
6. Click OK to exit the Publication Properties dialog box.

Editing a Publication Access List

When you create a Publication, you specify the list of users allowed to access this Publication. If you would like to alter this list, you can edit the Publication from the Publication Manager dialog box.

Task: ***To edit a Publication Access List:***

1. On the **Publication Manager**, select the Publication that you want to edit and click Edit.

   The **Publication Properties** dialog box appears.
2. Click the **Access List** tab. The **Access List** tab lists users who have the necessary privileges to access the Publication: those users who have been assigned `sysadmin` privileges to this database server through the SQL Enterprise Manager.
3. Select the users allowed to access this Publication. To remove a user, select it and click **Remove**.

   Note • *Only users that have been assigned* `sysadmin` *privileges in SQL Enterprise Manager to the connected database server can establish the Publisher-Subscriber relationship between the databases and are therefore listed on this panel. You can remove a user from this list (so that they will not be able to access this Publication), but you cannot add additional users to this list using AdminStudio. To add new users, you must use SQL Enterprise Manager.*
4. Click OK to exit the Publication Properties dialog box.

Deleting a Publication

When you no longer want to have a Publication available for Subscription, you can delete it using the Delete button on the **Publication Manager** dialog box.

Task: ***To delete a Publication:***

1. On the **Publication Manager** dialog box, select the Publication that you want to delete.
2. Click Delete. A dialog box opens asking you to confirm the deletion.
3. Click OK.

   The Publication data is deleted from the Publisher Database Server, and when the list on the Publication Manager is refreshed, the Publication will no longer be listed.

Creating and Managing Subscriptions

The topics in this section explain how to create a Subscription, edit its update schedule, enable or disable it, and update it manually. Please note the following about Subscriptions:

- **Subscribed data can co-exist with local data**—A Subscriber's Application Catalog can contain its own packages and can also contain subscribed Publications containing packages. The subscribed Publication is listed under the Replication group in the Subscriber's Application Catalog (when viewed in the Application Manager Product View).
- **Subscribed data is effectively read-only**—You should consider a Subscription as read-only data. Each time Subscription data is updated, it is completely replaced, so all changes that were made on a local copy of the Subscription data would be lost.

This section covers the following tasks:

- Managing Subscriptions
- Creating a New Subscription
- Enabling/Disabling a Subscription
- Manually Updating a Subscription
- Deleting a Subscription

Managing Subscriptions

You use the **Subscription Manager** dialog box to view Subscription information, create new Subscriptions, disable a Subscription so that it is not automatically updated, manually update a Subscription, and delete a Subscription.

The Subscription Manager, which lists all Publications that are currently subscribed to, is accessed by pointing at Replication on the Catalog menu and selecting Subscription Manager.

***Note** • The **Subscription Manager** option under **Replication** on the AdminStudio **Catalog** menu is only enabled if the current user is connected to an SQL Server database, has been assigned* `sysadmin` *privileges in SQL Enterprise Manager on that database server, and the database was identified when it was created as being intended for use as a **Subscriber**. See Application Catalog Wizard for information on how to identify databases as Subscribers or Publishers.*

The Subscription Manager lists the last time that each of the Subscriptions was updated, the next scheduled update time for that Subscription, and whether the Subscription is enabled (active) or disabled. From the Subscription Manager, you can perform the following tasks:

- **Create a New Subscription**—Click the **New** button to create a new Subscription using the **Subscription Wizard**.
- **Enable or Disable a Subscription**—Click the Enable/Disable button to toggle the selected Subscription's status between Enabled and Disabled. When you disable a Subscription, you are turning off its automatic update feature.

  ***Note** • While you can subscribe to multiple Subscriptions, only one Subscription can be active and listed in your Application Catalog at any one time: the one that has been executed (updated) the most recently. For information on how to use the Disable feature to subscribe to more than one Application Catalog, see Specialized User Scenarios.*

- **Manually Update a Subscription**—Click **Run** to manually update the selected Subscription. The Subscription Manager will then replace the data of the Application Catalog's currently active Subscription with the most recent version of the selected Subscription's data.
- **Delete a Subscription**—Click **Delete** to delete the selected Subscription from the Subscription Manager. When the list on the Subscription Manager is refreshed, the Subscription will no longer be listed.

Creating a New Subscription

To create a Subscription, use the Subscription Wizard to select the Publisher database that you want to subscribe to, and schedule how often you want the Subscription to be automatically updated. When the Subscription Wizard is complete, the replicated Publication data is copied into your Application Catalog, and the name of the Subscription is listed in the **Subscription Manager** dialog box.

Task: ***To create a new Subscription:***

1. On the Catalog menu, point to **Replication** and select **Subscription Manager**. The Subscription Manager Dialog Box opens.

   Note • *The **Subscription Manager** option under **Replication** on the AdminStudio **Catalog** menu is only enabled if the current user is connected to an SQL Server database, has been assigned* `sysadmin` *privileges in SQL Enterprise Manager on that database server, and the database was identified when it was created as being intended for use as a **Subscriber**. See Application Catalog Wizard for information on how to identify databases as Subscribers or Publishers.*

2. Click **New**. The Welcome Panel of the Subscription Wizard opens.
3. Click **Next**. The Subscription Details Panel opens, listing Application Catalogs that you have recently accessed.

   Note • *If you are connected to a standalone Application Catalog and you are not logged on to AdminStudio Enterprise Server, the Login Required Dialog Box opens before the Subscription Details panel opens, prompting you to log in.*

4. From the **Catalogs** list, select the Publisher Application Catalog you want to subscribe to, or click **Browse** to open the Select Application Catalog Dialog Box and connect to an existing Publisher Application Catalog as described in Connecting to an Existing Application Catalog.
5. To schedule how often the data will be updated, click **Change** next to the **Schedule** box. The Task Scheduling Dialog Box appears.
6. In the **Run the process every** list, select how frequently you want this Subscription to be automatically updated: **Day**, **Week**, or **Month**.
   - If you selected **Week**, select the day of the week that you want to update this Publication.
   - If you selected **Month**, select the day of the month that you want to update this Publication.
7. From the **Starting on** list, select the day or date that you want this Publication to begin being automatically updated on the day or date you selected from the **Run the process every** list.

   For example, if you want the Publication to be automatically updated on the 15th of each month, you would select the first day of the first month that you want the automatic updates to begin, not the first day that the Publication should be updated. If it is Monday and you want the Publication to be updated every Friday, to

have the Publication begin being updated on the upcoming Friday, you would select the current day of the week (Monday), but if you didn't want the Publication to be updated until the Friday after next, you would select Saturday.

8. Click **OK** to exit the **Task Scheduling** dialog box. You are returned to the **Subscription Details** panel, and the schedule you specified is now listed in the **Schedule** field.
9. Click **Next**. The Subscription Selection Panel appears, which displays the Publications available from the selected Publisher database that satisfy the Publication Access List Panel requirements for this user. In other words, it lists the Publications that you have permission to subscribe to.
10. Select one Publication from this list to subscribe to and click **Next**. The Subscription Summary Panel appears, providing a detailed summary of the options that were selected in previous panels of the Wizard.
11. If you would like to generate an SQL script of the code necessary to create this Subscription rather than actually create it, select the **Generate subscription as script** option, and enter a path name for the script file.

Note • *You might want to use this option if your organization has a controlled SQL database environment that prefers to have their database administrators make major changes to their databases. See Replicating Application Catalogs in Controlled SQL Database Environments for more information.*

12. Click **Finish** to complete the creation of this Subscription.

Enabling/Disabling a Subscription

You can use the Enable/Disable button on the **Subscription Manager** dialog box to set the status of each Subscription to be either Enabled or Disabled. When you disable a Subscription, you are turning off its automatic update feature. Subscriptions that are enabled are preceded by a check mark, while disabled Subscriptions are preceded by a red X.

Note • *For information on how you can use the Disable feature to subscribe to more than one Publication, see Specialized User Scenarios.*

Task: ***To enable or disable a Subscription:***

1. On the Catalog menu, point to Replication and select Subscription Manager. The **Subscription Manager** dialog box appears, listing all of your Subscriptions. Each Subscription's status (Enabled or Disabled) is displayed in the Last Run Status column.

   Note • *The **Subscription Manager** option under **Replication** on the AdminStudio **Catalog** menu is only enabled if the current user is connected to an SQL Server database, has been assigned* `sysadmin` *privileges in SQL Enterprise Manager on that database server, and the database was identified when it was created as being intended for use as a **Subscriber**. See Application Catalog Wizard for information on how to identify databases as Subscribers or Publishers.*

2. To disable an enabled Subscription, select it and click Disable.

   The Subscription's status in the Last Run Status column is changed to Disabled, meaning that the Subscription will still be listed in the Subscription Manager, but will not be automatically updated.

3. To enable a disabled Subscription, select it and click Enable.

   Note • *This button toggles between Enable and Disable, depending upon the Status of the selected Subscription.*

Manually Updating a Subscription

If you want to update a subscribed Publication to match the Publication on the Publisher Database Server and you do not want to wait until the next scheduled automatic update is performed, you can use the Run button on the **Subscription Manager** dialog box to manually update a Publication.

Task: ***To delete a Subscription:***

1. On the **Subscription Manager**, select the Publication that you want to update.
2. Click Run. The Publication is replicated from the Publisher Database Server, replacing the earlier version.

   Note • *You can only update a subscribed Publication that has already been published at least once. If the Publication has never been published, the update will fail.*

Deleting a Subscription

When you no longer want to subscribe to a Publication, you can Delete it from the Subscription Manager dialog box.

Task: ***To delete a Subscription:***

1. On the **Subscription Manager**, select the Subscription that you want to delete.
2. Click Delete. A dialog box opens asking you to confirm the deletion.
3. Click OK. The Subscription's name is deleted from the Subscription Manager List.

Using Microsoft SQL Server to Perform Merge Replication of Application Catalogs

AdminStudio supports merge replication of Application Catalog databases using Microsoft SQL Server 2005. You can use merge replication to synchronize Application Catalog databases distributed in multiple locations across a large enterprise, ensuring that all databases are kept up-to-date. One location can publish software packages to IT teams at other sites, keeping all teams current while enabling each to manage their software independently.

To set up merge replication, first you use the AdminStudio interface to create an Application Catalog to serve as a Publisher database. You then use SQL Server Management Studio to configure this database as a Publisher, by creating a Publication. Lastly, you use SQL Server Management Studio to create one or more Subscriptions (and Subscriber databases).

Perform the following steps to implement merge replication of AdminStudio Application Catalogs:

- Step 1: Creating the Publisher Database
- Step 2: Generating the Publication Script for the Publisher Database
- Step 3: Creating the Publication (Running the Script)
- Step 4: Creating a Subscription and a Subscribing Database
- Step 5: Starting the Snapshot Agent
- Step 6: Verifying the Existence and Integrity of the Subscriber Database:

Step 1: Creating the Publisher Database

To create the Publisher database, perform the following steps;

Task: ***To create the Publisher database:***

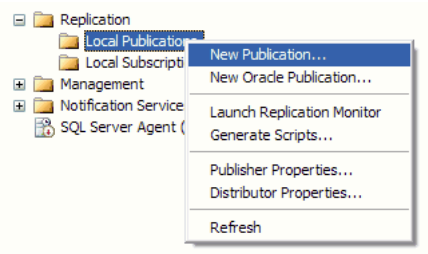
1. Open AdminStudio and create a new AdminStudio Application Catalog on an SQL Server 2005 database server, such as the AdminStudio Application Catalog database server).
2. Close AdminStudio, and verify in SQL Server Explorer/Management Studio the existence of the newly created database.

Step 2: Generating the Publication Script for the Publisher Database

To generate the Publication script for the Publisher database (Wizard), perform the following steps:

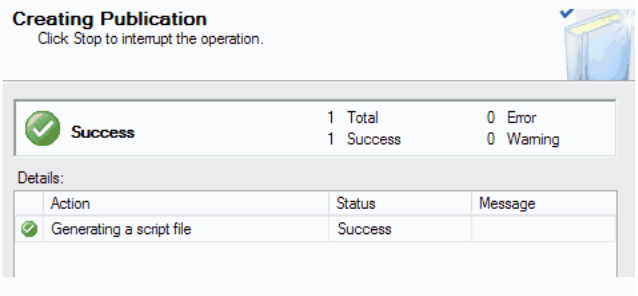
Task: ***To generate the Publication script for the Publisher database:***

1. Locate the **Replication** node in Management Studio for the given database server.
2. Expand **Replication**, right-click on **Local Publications**, and select **New Publication Wizard** from the context menu:



3. When prompted to select the **Publication Database**, select the database you created in Step 1: Creating the Publisher Database.
4. When prompted to select **Publication Type**, select **Merge Replication**.
5. When prompted to select **Subscriber Types**, verify that **SQL Server 2005** is the only option selected. (This document is targeted for SQL 2005 users only.)
6. On the **Articles** panel, select **Tables**, **Stored Procedures**, and **Views**, and click **Next**.
7. Continue by clicking **Next** on the **Article Issues** panel.
8. Continue by clicking **Filter Table Rows** on the **Article Issues** panel.
9. Configure the Snapshot agent as per your requirement. Leave defaults for default scheduling. (this setting can be changed later.)
10. On the **Agent Security** panel, click **Security Settings**, select **Run under the SQL Server Agent Service Account** option, and click **OK**.
11. Click **Next** on the Wizard.
12. On the **Wizard Actions** panel, clear the **Create the Publication** option and select the **Generate a script file** option.
13. Click **Next**.
14. Under **Script File Properties**, enter a valid path to store the script generated by this Wizard.
15. On the **Complete the Wizard** panel, enter a name for the Publication that you are creating (such as `ASPublication`), and click **Finish**.

16. Verify that the operation is reported as successful, as shown below:



Step 3: Creating the Publication (Running the Script)

To create the Publication, run the Publication script by performing the following steps:

Task: ***To create the Publication:***

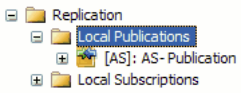
1. Open the script created in Step 2: Generating the Publication Script for the Publisher Database in a text editor.
2. Replace this code:

   ```
   @identityrangemanagementoption = N'auto', @pub_identity_range = 10000,
   @identity_range = 1000, @threshold = 80
   ```

   with this code:

   ```
   @identityrangemanagementoption = N'auto', @pub_identity_range = 5000000,
   @identity_range = 5000000, @threshold = 80
   ```

3. Save the changes. There should be approximately 281 changes.
4. Run the script against the Publisher Database (created in Step 1: Creating the Publisher Database)
5. Verify that the script runs without any errors or warnings.
6. Verify that a new publication is listed under **Replication**/**Local Publications**, as shown below. If no publication is listed, then refresh the **Replication** node.

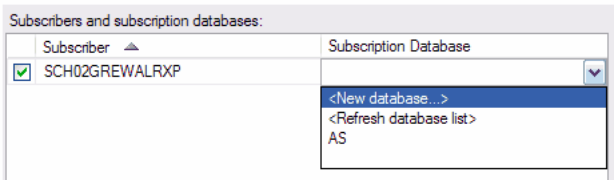


Step 4: Creating a Subscription and a Subscribing Database

To create a subscription and a subscribing database, perform the following steps:

Task: ***To create a subscription and a subscribing database:***

1. Right-click on **Local Subscriptions** and select **New Subscription Wizard** from the context menu.
2. When prompted to select a **Publication**, choose the publication we created in Step 3: Creating the Publication (Running the Script).
3. Under **Merge Agent Location**, choose **Run all agents at the Distributor** . This setting can be the default, and can be changed by an Administrator to manage the server load.
4. When prompted to select **Subscribers and subscription databases**, select the server under **Subscriber** on the left side, and select **<New database...>** under **Subscription Database** on the right side, as shown below:



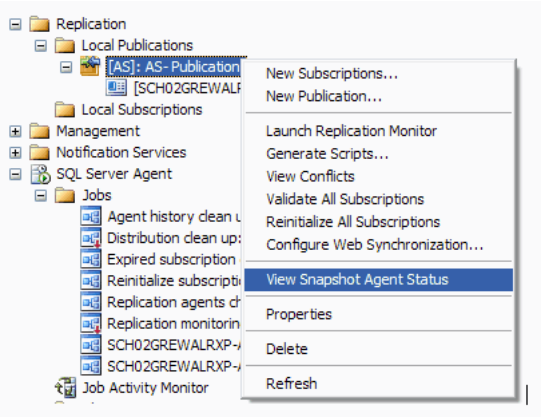
5. When prompted, enter a name for the new database (such as `AS-Subs 01`) and continue the wizard.
6. Under **Merge Agent Security**, provide the settings so that the Merge Agent runs under the default SQL Server Agent service account.
7. Under **Synchronization Schedule**, select **Run Continuously** under **Agent Schedule**. An administrator can modify this and provide a scheduling option instead of continuous.
8. Verify that **Initialize** is selected, and that **At first Synchronization** under **Initialize When** is selected. Click **Next**
9. Click **Next**. Under **Wizard Actions**, select the **Create the subscription** option and then finish the Wizard.
10. Verify that there are no errors or warnings in the report.
11. Close the Wizard.
12. Refresh the **Publication** node under **Local Publications** and expand it. There should be a new **Subscription** listed under the **AS-Publication** publication.

Step 5: Starting the Snapshot Agent

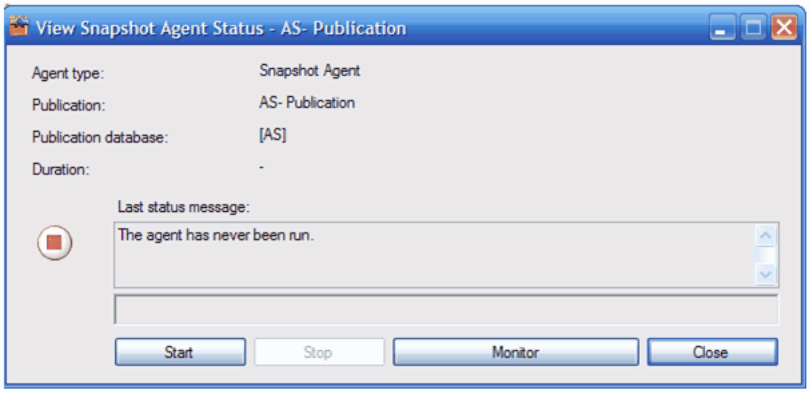
To start the Snapshot Agent, perform the following steps:

Task: ***To start the Snapshot Agent:***

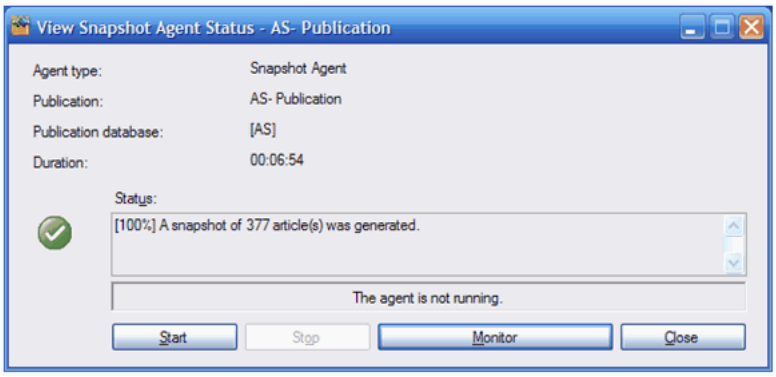
1. Right-click on **AS-Publication** and select **View Snapshot Agent Status** from the context menu.



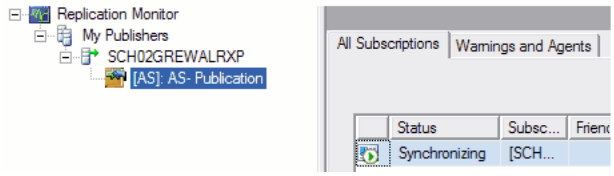
The **Snapshot Agent** opens:



2. Wait until the Snapshot Agent is done taking the initial snapshot.



3. Right-click on **Local Publications** and select **Launch Replication Monitor** from the context menu. Verify in the monitor that for the **AS-Publication** publication, there is one Subscriber and it is in **Synchronizing** state, as shown below:



Step 6: Verifying the Existence and Integrity of the Subscriber Database:

To verify the existence and integrity of the Subscriber database, perform the following steps:

Task: ***To verify the existence and integrity of the Subscriber database:***

1. Refresh and expand the **Database** node in Management Studio.
2. Verify that a new database (created in Step 1: Creating the Publisher Database) is created. Expand the **Tables** under this database and verify that a list of AdminStudio tables appears.
3. Launch AdminStudio and try to connect to this catalog.
4. Launch Application Manager and try to import a package in this catalog.

Note • *To create more subscribers, repeat Step 4: Creating a Subscription and a Subscribing Database.*

Taking OS Snapshots

The OS Snapshot Wizard provides a simple way to capture your basic operating system configuration. Using the OS Snapshot Wizard, you can scan your computer's operating system and record the files, INI files, shortcuts, and registry entries present. The Wizard then creates an .osc file representing the system contents. You can import this snapshot file into the an Application Catalog database to identify potential conflicts between Windows Installer-based setups and your operating system.

To provide maximum flexibility during the OS Snapshot process, you can create an exclusion list (similar to the Repackager exclusion list) that identifies files, INI files, shortcuts, and registry entries that the OS Snapshot Wizard should disregard during the scan. Using this list, you can eliminate unnecessary files, shortcuts, or registry entries, and reduce the time it takes to perform the OS Snapshot.

The following topics are included in this section:

- OS Snapshot Best Practices
- Configuring OS Snapshot Analysis Options
- Capturing an OS Snapshot

Caution • *OS Snapshots should only be used for comparison in ConflictSolver. You should never attempt to convert an OS Snapshot into an MSI package.*

OS Snapshot Best Practices

Before capturing OS Snapshots, consider the following:

Table 6-21 • OS Snapshot Best Practices

Guideline	Description
Capture on a Clean System	For optimal OS Snapshots, you should only capture on a clean system. This ensures an accurate baseline of files necessary for the operating system. This also means that you should never attempt to capture other packages in addition to the operating system. Use Repackager when you need to capture applications.
Exit Other Applications	Shut down all other applications besides OS Snapshot prior to capture. Ideally, this should be done from the Windows Task Manager. This ensures that files are not locked during capture, and unnecessary temporary files are not inadvertently captured.
Only Use OS Snapshot for Import into Application Manager	Never attempt to convert an OS Snapshot file into a Windows Installer package to install an operating system. AdminStudio does not support this use of OS Snapshots.

Table 6-21 • OS Snapshot Best Practices (cont.)

Guideline	Description
OS Snapshots Take Time	Depending on the operating system configuration, OS Snapshot often takes a significant amount of time to capture the base OS state. Consider that many typical OS installations exceed 500MB and contain tens of thousands of files, translating into a lengthy operation of cataloging and converting the files into an OS Snapshot file.
Take Multiple OS Snapshots	If your environment contains either multiple operating systems, or variations on the same operating system, take snapshots of each OS or variation. You can store all of these in your Application Catalog, allowing you to make comparisons between MSI packages and each OS.
OS Snapshots and Repackaging Are Not the Same	OS Snapshots, as the name implies, is for snapshots of the operating system only. Repackaging is only for traditional installation packages. These operations, although similar, are still very specialized and should only be used for their respective purposes.

Configuring OS Snapshot Analysis Options

Task: ***To configure analysis options for OS snapshot captures:***

1. Launch the **OS Snapshot Wizard** by clicking on its icon in the AdminStudio Tools Gallery. The **Welcome** panel opens.
2. Click **Next**. The **Project Information** panel opens.
3. Click Edit. The **Analysis Options** dialog box opens.
4. Configure the types of data you want to capture (Files, INI files, Shortcuts, and/or Registry data), and specify if you want to restrict the snapshot to a specific drive.
5. Click **OK** to return to the **Project Information** panel.
6. Click **Start**.

Capturing an OS Snapshot

You use the OS Snapshot Wizard to capture an OS Snapshots.

Task: ***To perform an OS Snapshot:***

1. Launch the **OS Snapshot Wizard** by clicking on its icon in the AdminStudio Tools Gallery. The **Welcome** panel opens.
2. Click Next. The **Project Information** panel opens.
3. Provide the OS Snapshot project name and OS Snapshot project folder for the OS Snapshot file.
4. Optionally, click **Edit** to configure analysis options. See Configuring OS Snapshot Analysis Options.
5. Click **Start** to perform the OS Snapshot.
6. On completion of the OS Snapshot, review the results in the **Summary** panel.
7. Click **Finish**.

Note • *The OS Snapshot is stored as an OSC file in the folder defined in the* ***Project Information*** *panel.*

Reference

This section contains information on the all of the dialog boxes and wizards that are used when managing, connecting to, import data into, or sharing application catalogs.

- Application Manager Views
- Dialog Boxes
- Wizards
- Database Server Permissions
- Application Manager Command-Line Functionality

Application Manager Interface

The Application Manager user interface consists of three areas: the Navigation window, the Details pane, and the Output Window. Both the Navigation window and the Output Window are dockable.

- **Navigation window**—The Navigation window consists of three tabs: **Products**, **Merge Modules**, and **Patches**.
- **Details pane**—When you select different items in these tabs, the Details pane displays corresponding information about that item.

- **Output window**—The Output Window consists of tabs where output is displayed during different Application Manager processes.

Application Manager also supports many context menus in addition to the menu bar and toolbar functionality.

This section details the Application Manager interface and includes the following topics:

- Menus and Toolbar
- Product View Icons
- Merge Modules View Icons
- Context Menus
- Output Window

Note • *By default, Application Manager displays all conflict detection and resolution options. The display of the conflict detection and resolution options can be turned off by selecting the* ***Turn off display of conflict detection and resolution options in Application Manager*** *option on the* ***General*** *tab of the ConflictSolver and Application Manager* ***Options*** *dialog box. This section of the documentation describes Application Manager functionality. For information on ConflictSolver functionality, see ConflictSolver Interface.*

Menus and Toolbar

The following commands and toolbar buttons are available in Application Manager. Additionally, several commands can be accessed through Context Menus.

Table 6-22 • Menus & Toolbars

Menu	Command	Shortcut	Button	Description
Catalog	**New**			Opens the Application Catalog Wizard, which you can use to create a new Application Catalog.
Catalog	**Connect**	Ctrl+O		Displays the Connect Application Catalog dialog box, where you can open an existing Application Catalog.
Catalog	**Refresh**	Ctrl+R		Refreshes the current Product and Merge Module views. This is particularly useful if multiple people are working on the same Application Catalog.
Catalog	**Disconnect**			Closes the currently open Application Catalog.
Catalog	**Properties**			Open the Application Catalog Properties dialog box.

Table 6-22 • Menus & Toolbars (cont.)

Menu	Command	Shortcut	Button	Description
Catalog	**Change AES Password**			Change the password of the current user to log in to the AdminStudio Enterprise Server.
Catalog	**Log Out**			Log out of AdminStudio Enterprise Server.
Catalog	**Package Auto Import**	Ctrl+E		Link/unlink to one or multiple Application Catalogs. This allows you to copy packages between various Application Catalogs to maintain consistency between the copied packages. You can also choose to manually refresh the Application Catalog so that it remains consistent with other linked Application Catalogs.
Catalog	**Merge**	Alt+M		Launches the Merge Wizard, which you can use to select a different Application Catalog and import its contents into the current Application Catalog.
Catalog	**Import Package**	Ctrl+I		Launches the Import Wizard, allowing you to import Windows Installer packages, transforms, patches, merge modules, OS snapshots, Marimba NCP files, and virtual packages (Microsoft App-V, VMware ThinApp, and Citrix). You can also select Other Setup Types to import other non-MSI based setups.
Catalog	**Exit**			Closes Application Manager.
Edit	**Find**	Ctrl+F		Use to search for data in various tables in the Application Catalog.
View	**Toolbar**	Ctrl+0		Toggles the Toolbar.
	Status Bar	Ctrl+1		Toggles the Status Bar.
	Output Window	Ctrl+2		Toggles the Output Window.
	Application Manager	Ctrl+3		Select to switch to Application Manager.
	ConflictSolver	Ctrl+4		Select to switch to the ConflictSolver.

Table 6-22 • Menus & Toolbars (cont.)

Menu	Command	Shortcut	Button	Description
Conflicts	**Rules Viewer**			Displays the Rules Viewer dialog box, from which you can create new ACE rules.
Tools	**Customize**	Alt+O		Opens the Customize dialog box, where you can customize toolbar settings.
	Options	Alt+N		Opens the ConflictSolver and Application Manager Options Dialog Box, where you can change various settings including default conflicts to identify, the default location for Windows Installer packages, report locations, the default validation file, and duplicate package identification options.
Help	**Contents**			Launches the online Help Library and displays the Contents tab.
	Index			Launches the online Help Library and displays the Index tab.
	Search			Launches the online Help Library and displays the Search tab.
	Support Central			Connects to the AdminStudio Support Web site.
	Web Community			Connects to the AdminStudio Web Community.
	ReadMe			Displays the AdminStudio Release Notes.
	Feedback			Connects to an online form, through which you can provide feedback about AdminStudio.
	Flexera Software on the Web			Connects to the Flexera Software Web site.
	About Application Manager			Displays the About dialog box with version information.
	Help Library			Launches the online Help Library.

Product View Icons

These icons are used on the ConflictSolver and Application Manager Product Views:

Table 6-23 • ConflictSolver and Application Manager Product View Icons

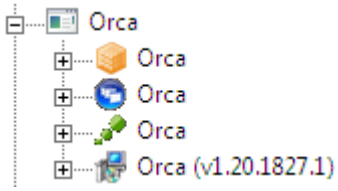
Name	Icon	Description
Application Catalog		The Application Catalog that you are connected to. When you select it, one of the following views opens: • Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards • Application Catalog View (Professional Edition)
Group		A group, which is used to organize your data.
Product		An application that has been imported into the Application Catalog database. The application can include multiple deployment types—such as a Windows Installer package, a Microsoft App-V package, a VMware ThinApp package, etc. Each deployment type is indicated by a separate subnode of this application node. In the following example, the application includes a Microsoft App-V package, a Citrix XenApp package, a VMware ThinApp package, and a Windows Installer package. 
		Microsoft App-V package that has been imported into the Application Catalog.
		Citrix XenApp package that has been imported into the Application Catalog.
		VMware ThinApp package that has been imported into the Application Catalog.
		An “X” overlay on an icon indicates that that the package has an unresolved conflict, or that one of an application’s deployment types has an unresolved conflict.

Table 6-23 • ConflictSolver and Application Manager Product View Icons (cont.)

Name	Icon	Description
Product (continued)		An "!" overlay on an icon indicates that the package has an unresolved question with its associated files (or that one of an application's deployment types has an unresolved question with its associated files). One or more files, transforms, or patch files associated with a package is either missing from the original import directory or its last modified date does not match the last modified date stored in the Application Catalog. When a package icon with this overlay is selected, a message appears in the view identifying the files in question and prompting you to take action to resolve the problem.
		An icon that contains both an "!" and an "X" overlay icon indicates that it is a package with unresolved conflicts that also has an unresolved question with its associated files.
		This overlay icon indicates that the package is managed within the Software Repository. This overlay icon is only displayed in Application Manager.
		This overlay icon indicates that the package is managed within the Software Repository and is checked out. This overlay icon is only displayed in Application Manager.
		This overlay icon indicates that the package is linked to a Remote Application Catalog. See Automatically Importing Packages.
		These overlay icons indicate that the package is linked in a local Application Catalog that has been updated on the Remote Server and needs to be refreshed in order to update its data with the data on the Remote Server. See Manually Updating a Linked Package.
Transforms		A transform.
		A transform that has an unresolved question. It is either missing from the original import directory or its last modified date does not match the last modified date stored in the Application Catalog. When this icon is selected, a message appears in the Application Manager Product View prompting you to take action to resolve the problem.
OS Snapshot		An OS snapshot, which is a file representing a particular computer system's contents.
Conflicts		Click to access the package's Conflicts View, where you can access information pertaining to the last execution of the Conflict Wizard.

Table 6-23 • ConflictSolver and Application Manager Product View Icons (cont.)

Name	Icon	Description
Validation		Click to access the package's Validation View, which lists all ICE Errors and Warnings that were generated the last time the selected package was validated.
Patch Impacts		Click to access the package's Patch Impacts Analysis View, which lists patches for which there is patch impact data persisted against the product, and identifies the patch that caused the impact.
Dependencies		Click to display the Dependencies associated with the package. See Dependencies View for more information.
Extended Attributes		Click to display the optional Extended Attributes associated with the package. See Extended Attributes View for more information.
Files, Files/ Components, Files/ Directories		Click to display the Files, Files/Components, or Files/Directories view for this package.
		Click to display the Files, Files/Components, or Files/Directories view for this package. The "!" overlay icon indicates that a file is missing from the original import directory. When this icon is selected, a message appears in the view describing the missing files and prompting you to locate the missing files.
INI File Changes		Click to display the INI File Changes View, listing any INI file changes made by the product.
Merge Modules		Click to display any merge modules included the package.
Associated Patches		Click to display the product's Associated Patches View, which displays a list of imported patches that, if installed, would update the selected package.
Registry		Click to display any registry entries created or changed by the package.
Shortcuts		Click to display any shortcuts created by the package.
Tables		Click to view the data for a given package in the Application Catalog.
Catalog History		Click to view a list of logged events for the selected package.
Virtualization Suitability		Click to view the results of a Virtualization Readiness analysis of that package. See Virtualization Suitability View.

Table 6-23 • ConflictSolver and Application Manager Product View Icons (cont.)

Name	Icon	Description
Application Compatibility		Click to view Compatibility Solver test results for a selected Windows Installer package. See Application Compatibility View.
App-V History		Click to open a view which lists an entry for each time this App-V package has been saved.
File Type Associations		Click to view a list of the file type associations for this App-V package.
Environment Variables		Click to display the environment variables used in this App-V package.

Merge Modules View Icons

The following icons are used on the Merge Modules tab:

Table 6-24 • Merge Modules View Icons

Name	Icon	Description
Merge Modules		Indicates both the All Merge Modules view, the root node of the merge modules explorer containing a list of all merge modules in the Application Catalog, and an individual Merge Module.
Merge Module Managed Within the Software Repository		A Merge Module managed within in the Software Repository.
Components		Select to display any components created or changed by the merge module.
Dependency		Select to display any dependencies in the merge module.
Exclusion		Select to display any exclusions in the merge module.
Files		Select to display any files in the merge module
Products		Select to display any products in the Application Catalog that use the merge module.

Context Menus

Application Manager includes several context menus which can be accessed by right-clicking on nodes on the **Products**, **Merge Modules**, and **Patches** tabs. These menus provide specific functionality in relation to what is clicked.

- Groups
- Packages
- Merge Modules Node
- Individual Merge Modules
- Patch View on the Patches Tab

Groups

When you right-click on a group in the Application Manager Product View, the following commands are available through the context menu:

Table 6-25 • Group Context Menu Commands

Command	Description
Import	Allows you to import a package or an operating system patch into the Application Catalog. You have two options: • **Package**—Select this option to launch the Import Wizard to import Windows Installer packages, transform files, merge modules or patch files (.msp) into the Application Catalog. • **Patches**—Select this option to launch OS Security Patch Wizard so that you can import Windows operating system and Microsoft application patches into the Application Catalog.
New Group	Creates a new group within the selected group.
Rename	Allows you to rename the selected group.
Delete	Removes the selected group from the Application Catalog, including any groups, products, or OS Snapshots contained within it.
Run Virtualization Readiness	Select to manually retest a package or group of packages for virtualization readiness. See Virtualization Suitability View. ***Note*** • *Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog.*
Find	Select to perform a search for data in the various tables of the Application Catalog. See Searching an Application Catalog for more information.

Table 6-25 • Group Context Menu Commands (cont.)

Command	Description
Cut	Cuts the selected group from the Application Catalog. You can then paste it using the Paste command.
Paste	Pastes a cut group from Application Manager.
Properties	Opens the Group Properties dialog box.

Packages

When you right-click on a package in the Application Manager Product View, the following commands are available through the context menu:

Table 6-26 • Package Context Menu Commands

Command	Description
Import	Allows you to import a package or an operating system patch into the Application Catalog. You have two options: • **Package**—Select this option to launch the Import Wizard to import Windows Installer packages, transform files, merge modules or patch files (.msp) into the Application Catalog. • **Patches**—Select this option to launch OS Security Patch Wizard so that you can import Windows operating system and Microsoft application patches into the Application Catalog.
Reimport the package	Reimports the package from its source location (if known and present). ***Note*** • *Supported for Windows Installer packages only.*
Find	Select to perform a search for data in the various tables of the Application Catalog. See Searching an Application Catalog for more information.
Cut	Cuts the selected package from the Application Catalog. You can use the paste command to paste it to a new location.
Copy	Copies the selected package. You can use the paste command to paste it to a new location.
Paste	Allows you to paste a package that has been cut or copied.

Table 6-26 • Package Context Menu Commands (cont.)

Command	Description
Edit with InstallShield Editor	Open this package in InstallShield Editor, where you can directly edit the package or create a transform. ***Note*** • *Supported for Windows Installer packages only.*
Edit with Virtual Package Editor	(App-V packages only) Open this App-V package in Virtual Package Editor, where you can view and edit package data.
Distribute Package	Launch the Distribution Wizard. ***Note*** • *Supported for Windows Installer packages only.*
Perform Predeployment Testing	Launch the Predeployment Test Preparation Wizard. ***Note*** • *Supported for Windows Installer packages only.*
Refresh Package Auto Import	Rather than waiting until the next scheduled automatic update is performed, select this option to manually re-import a Windows Installer or App-V package that is being monitored in a Remote Application Catalog.
Rename	Select to rename the selected package.
Delete	Select one of the following: • **Package**—Removes the product from the current Application. • **Package from all Applications**—Removes the product from all Applications and removes it from the Application Catalog. • **All Extended Attributes**—Removes all extended attributes from the selected package. • **Persisted Conflict Information**—Removes persisted conflict information for the selected package. • **Persisted Validation Information**—Removes persisted validation information for the selected package. • **History Log Information**—Removes all change history data for the selected package.

Table 6-26 • Package Context Menu Commands (cont.)

Command	Description
Associate with Workflow Manager Application	Launches the **Associate with Workflow Manager Application** dialog box, from which you can pick a package in Workflow Manager with which to associate the extended attribute data for the selected product. ***Note*** • *Supported for Windows Installer packages only.* ***Note*** • *AdminStudio Workflow Manager is a Web-based application management system that has integrated functionality with AdminStudio.*
Reports	Select one of the following: • **Package**—Creates the Package Report. • **Files**—Creates the Files Report. • **Registry**—Creates the Registry Report. • **Crystal Reports Conflicts**—Creates the pre-defined Crystal Reports Conflicts Report. • **Crystal Reports Files**—Creates the pre-defined Crystal Reports Files Report. • **Crystal Reports Registry**—Creates the pre-defined Crystal Reports Registry Report. ***Note*** • *Supported for Windows Installer packages only.*

Merge Modules Node

When you right-click on the root-level Merge Modules node in the merge modules explorer, the following command is available through the context menu:

Table 6-27 • Merge Modules Node Context Menus

Command	Description
Import Merge Module	Launches the Import Wizard directly to the **MSM Source Information** panel.

Individual Merge Modules

When you click on an individual merge module in the merge modules explorer, the following commands are available through the context menu:

Table 6-28 • Individual Merge Modules Context Menus

Command	Description
Import Merge Module	Launches the Import Wizard directly to the MSM Source Information panel.
Delete	Removes the selected merge module from the Application Catalog.

Patch View on the Patches Tab

When you click on an individual patch in the Patch View, the following commands are available through the context menu:

Table 6-29 • Patch View Context Menus

Command	Description
Rename	Rename the selected patch.
Delete	Removes the selected patch from the Application Catalog.
Cut	Cuts the selected patch from the Application Catalog. You can use the paste command to paste it to a new location.
Copy	Copies the selected patch. You can use the paste command to paste it to a new location.
Paste	Allows you to paste a patch that has been cut or copied.
Generate Report	Select to generate a Patch Report for that patch.
Properties	Launch the Patch Properties dialog box for that patch.

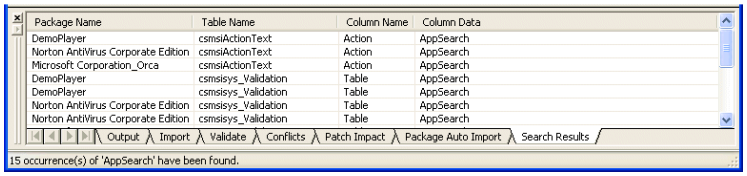
Output Window

When the Import and Package Auto Import Wizards are run, and when a Search is performed, the output messages and results of those wizards are displayed in the various tabs of the Output Window.

Table 6-30 • Output Window Context Menus

Tab	Description
Import Tab	When the Import Wizard is run to import a package into the Application Catalog, messages about the Import process are displayed in this tab.

Table 6-30 • Output Window Context Menus (cont.)

Tab	Description
Package Auto Import Tab	When the Package Auto Import Wizard is run to automatically import packages in a Remote Application Catalog or Network Directory, messages about the import process are displayed in this tab.
Search Results Tab	When Find is used to search for data in Application Catalog tables, the data that is found is displayed in this tab, in the following format:  If you double click on this data, Application Manager will navigate to the appropriate record in the Tables View, and that record will be highlighted.

User Permissions in Application Manager

Security and permissions can be assigned to Application Manager users to restrict the tasks that can be performed in Application Manager. The following table describes the various Application Manager permissions.

Table 6-31 • Application Manager Permissions

Permission	Explanation
Import Applications	Can import packages or merge modules into an Application Catalog, including re-importing packages resulting from unlinked data or conflict resolution
Delete	Can delete individual packages from the Application Catalog
Merge Database	Can merge information from two different Application Catalogs
Change Data	Can write to the Application Catalog. Without this permission, any Application Manager permission that allows users to write to the Application Catalog will not function; Application Manager is essentially read-only for roles without this permission.
Change Extended Attributes	Can change the XML file used to configure the extended attribute definition
Change Database	Can open another Application Catalog

Additionally, AdminStudio Administrators can restrict access to the Application Manager completely by disabling the Application Manager tool permission.

Note • *Typically only AdminStudio Administrators and a few select users will have access to create new Application Catalogs, upgrade Application Catalogs, or otherwise affect multiple users.*

Application Manager Views

The following views are associated with the Application Manager **Products** tab:

- Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards
- Application Catalog View (Professional Edition)
- Group View
- Product View
- App-V Package View
- Citrix / ThinApp Package View
- OS Snapshot View
- Other Setup Types View
- Marimba NCP Files Views
- Patches Tab Views
- Merge Module Views

Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards

AdminStudio provides an enhanced reporting infrastructure. AdminStudio has incorporated Microsoft SQL Reporting Services to provide a summary dashboard of Application Catalog readiness data and to enable you to create customizable reports.

- Application Readiness Dashboard
- Application Compatibility Dashboard
- Virtualization Readiness Dashboard
- Error Category Breakdown
- Shared Extensions Report
- App-V Dependencies Report
- App-V Dependents Report
- App-V Recommended Dependencies Report
- Customizable Reports

Note • *For information on the Application Catalog View that is displayed in AdminStudio Professional Edition, see Application Catalog View (Professional Edition).*

Application Readiness Dashboard

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

AdminStudio uses Microsoft SQL Reporting Services to provide the **Application Readiness Dashboard**, a summary dashboard of Application Catalog data concerning the readiness of Windows Installer and App-V packages for distribution. This dashboard includes charts that provide a snapshot of the current status of packages in your Application Catalog including deployment type breakdown, App-V virtualization readiness, Windows 7 test results from Compatibility Solver, and the results of Windows Installer and App-V best practices testing. You can click any of the categories in these charts to open more detailed reports.

To open the Application Readiness Dashboard, select the **Application Catalog** node in the Application Manager tree or open the AdminStudio **Report Center** tab, and then select **Enterprise Dashboard** from the **Select Report** list.

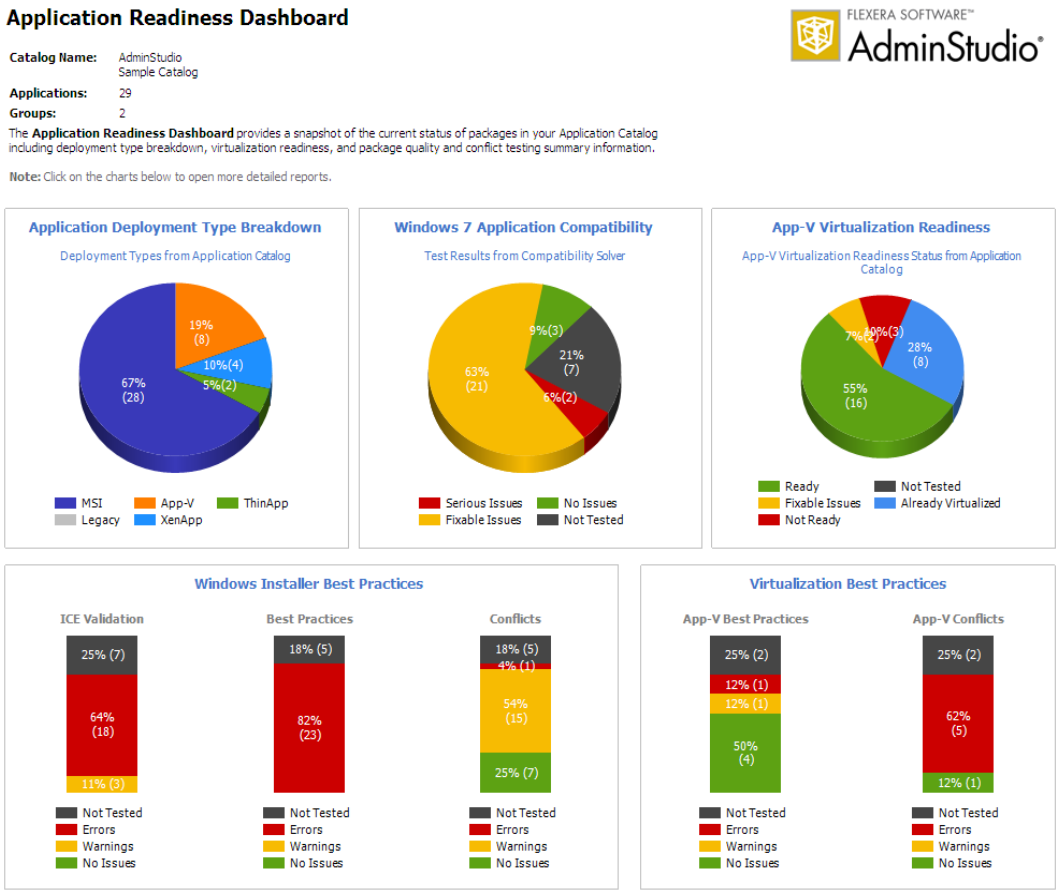


Figure 6-21: Application Readiness Dashboard

The following information is listed at the top of the report:

Table 6-32 • Summary Information on Application Readiness Dashboard

Property	Description
Catalog Name	The name of the Application Catalog database that you are connected to.
Applications	The total number of applications in the Application Catalog.
Groups	The total number of groups defined in the Application Catalog.

The Application Readiness Dashboard includes the following charts:

- Application Deployment Type Breakdown
- Windows 7 Application Compatibility
- App-V Virtualization Readiness
- Windows Installer Best Practices

- Virtualization Best Practices

Application Deployment Type Breakdown

The **Application Deployment Type Breakdown** chart of the Application Readiness Dashboard lists the percentage of packages in the Application Catalog by deployment type: **MSI** (Windows Installer), **App-V**, **XenApp**, **ThinApp**, and **Legacy**.

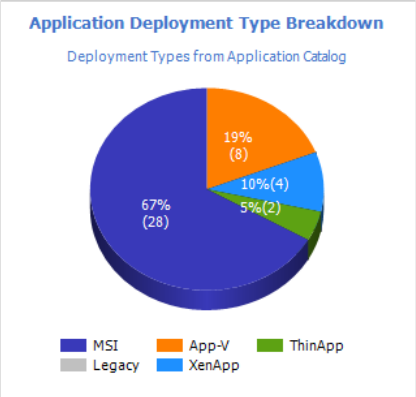


Figure 6-22: Application Deployment Type Breakdown Chart in the Application Readiness Dashboard

To view a list of packages of a specific deployment type, click on the appropriate category of the chart.

App-V Packages

FLEXERA SOFTWARE™ AdminStudio®

This report lists all the App-V packages in the application catalog.

Product Name	Product Version	Manufacturer
Acord Forms Lite	6.6	FileMaker
Adobe Reader 8	8.0.0	Adobe Systems Incorporated
Application Verifier 3.3	3.3.45	Microsoft Corporation
Picasa	2.7.0	Google Inc.
HexEditorNeo_3		
Messenger	8.1.0178	Microsoft Corporation
Orca	1.50.2600.0000	Microsoft Corporation
XML Notepad 2007	2.3.0.0	Microsoft Corporation

Figure 6-23: App-V Packages Report

Windows 7 Application Compatibility

Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.*

The **Windows 7 Application Compatibility** chart of the Application Readiness Dashboard lists the results of Windows 7 application compatibility tests that were performed in Compatibility Solver on packages that were loaded into the Compatibility Solver database. Packages are grouped according to the type of issues found: **Serious Issues**, **Fixable Issues**, **No Issues**, or **Not Tested**.

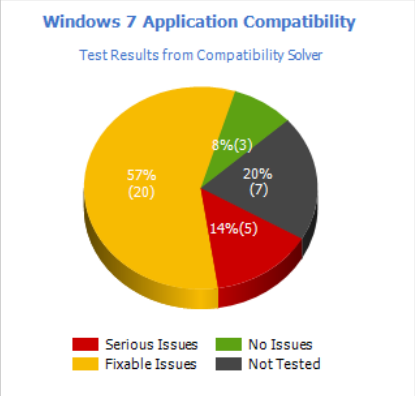


Figure 6-24: Windows 7 Application Compatibility Chart in the Application Readiness Dashboard

To view a list of packages in each group, click on the appropriate category of the chart.

Application Compatibility Results - Windows 7
Packages With Serious Issues Found

FLEXERA SOFTWARE™ AdminStudio®

Product Name	Manufacturer	Serious Issue Count	Fixable Issue Count
Messenger	Microsoft Corporation	1	44
Microsoft ActiveSync	Microsoft Corporation	1	6
PageMaker	Starbucks Corporation	1	8
Picasa	Google Inc.	5	4
Win2K	MyCompany	208	50

Figure 6-25: Packages With Serious Issues Found During Windows 7 Application Compatibility Testing

Note • *In order to display Compatibility Solver test results in AdminStudio reports, you need to specify the location of the **Compatibility Solver Database** on the **Application Catalog** tab of the AdminStudio **Options** dialog box. See Connecting to the Compatibility Solver Database.*

Note • *For information on viewing additional Compatibility Solver test results in AdminStudio, see Viewing the Application Compatibility Dashboard and Viewing Windows Installer Package Application Compatibility Issues.*

App-V Virtualization Readiness

The **App-V Virtualization Readiness** chart of the Application Readiness Dashboard lists the percentage of and number of Windows Installer packages in the Application Catalog in each of the following App-V virtualization readiness categories: **Ready**, **Fixable Issues**, **Not Ready**, **Not Tested**. This chart also lists the number of App-V packages in the Application Catalog in the **Already Virtualized** category.

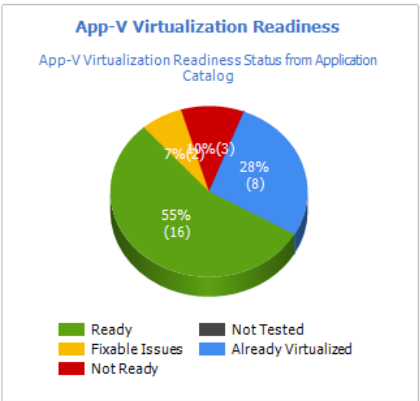


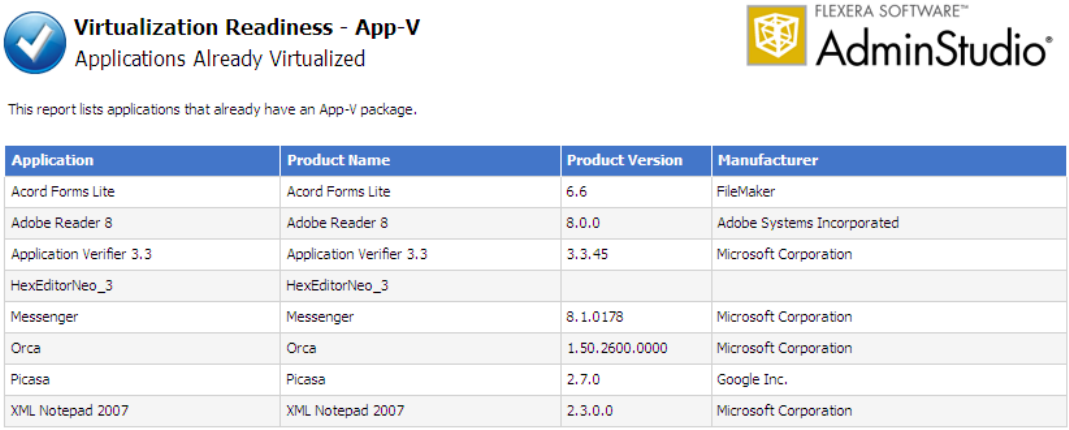
Figure 6-26: App-V Virtualization Readiness Chart of the Application Readiness Dashboard

Regarding the App-V Virtualization Readiness chart, note the following:

- **If an application contains a Windows Installer package but no App-V package**, the suitability of the Windows Installer package is evaluated and the application is listed in the **Ready**, **Fixable Issues**, **Not Ready**, or **Not Tested** category.
- **If an application contains just an App-V package or both a Windows Installer package and an App-V package**, it is listed in the **Already Virtualized** category.
- **If an application only contains a legacy package** (without a Windows Installer or an App-V package), it is ignored.

Note • *For more information, see Virtualization Readiness Categories.*

To view a list of packages in a specific category, click on the appropriate section of the chart.



Virtualization Readiness - App-V
Applications Already Virtualized

FLEXERA SOFTWARE™ AdminStudio®

This report lists applications that already have an App-V package.

Application	Product Name	Product Version	Manufacturer
Acord Forms Lite	Acord Forms Lite	6.6	FileMaker
Adobe Reader 8	Adobe Reader 8	8.0.0	Adobe Systems Incorporated
Application Verifier 3.3	Application Verifier 3.3	3.3.45	Microsoft Corporation
HexEditorNeo_3	HexEditorNeo_3		
Messenger	Messenger	8.1.0178	Microsoft Corporation
Orca	Orca	1.50.2600.0000	Microsoft Corporation
Picasa	Picasa	2.7.0	Google Inc.
XML Notepad 2007	XML Notepad 2007	2.3.0.0	Microsoft Corporation

Figure 6-27: Applications With App-V Packages

Note • *Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog. To manually retest a package or group of packages for virtualization readiness, select a package, application, or group in the Application Manager tree and select* ***Run Virtualization Readiness*** *from the context menu.*

Note • *To see a summary of not only the App-V virtualization readiness status, but also the Citrix XenApp and VMware ThinApp virtualization readiness status, open the Virtualization Readiness Dashboard.*

Windows Installer Best Practices

The **Windows Installer Best Practices** chart of the Application Readiness Dashboard lists the Microsoft ICE Validation status of Windows Installer packages, and the AdminStudio Best Practices and Conflict test status of Windows Installer packages.

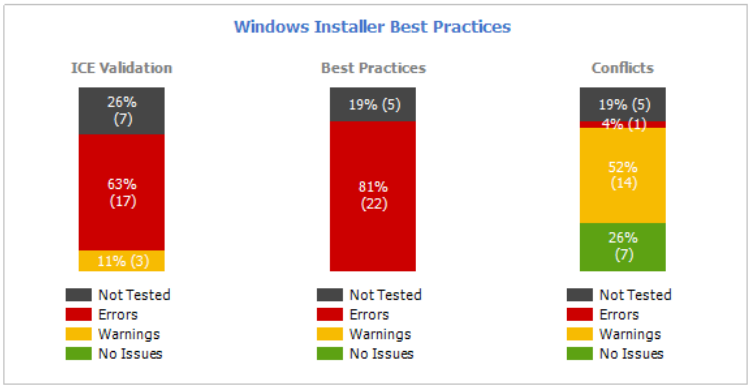
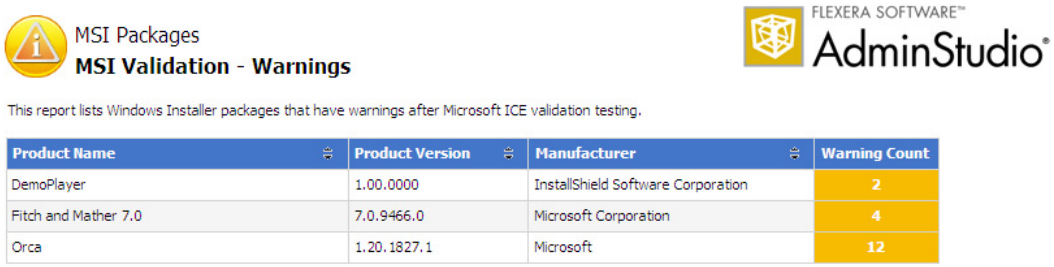


Figure 6-28: Windows Installer Best Practices Charts of the Application Readiness Dashboard

If you click on one of the categories in these charts, you see a list of packages that fall into that category.



MSI Packages
MSI Validation - Warnings

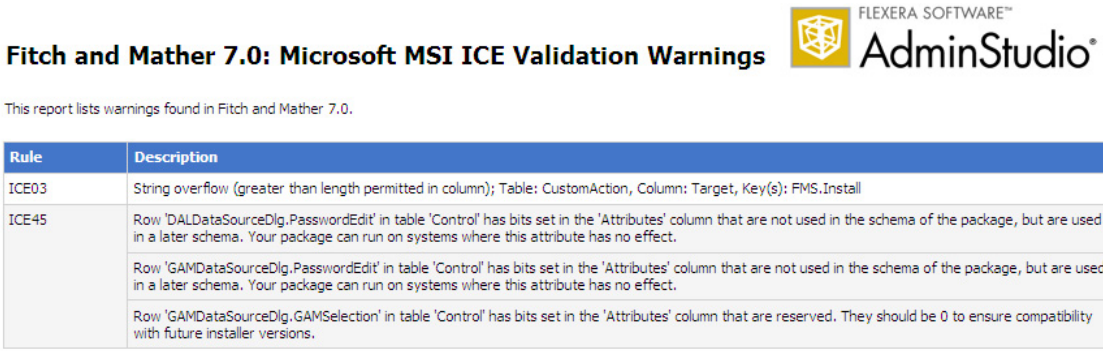
FLEXERA SOFTWARE™ AdminStudio®

This report lists Windows Installer packages that have warnings after Microsoft ICE validation testing.

Product Name	Product Version	Manufacturer	Warning Count
DemoPlayer	1.00.0000	InstallShield Software Corporation	2
Fitch and Mather 7.0	7.0.9466.0	Microsoft Corporation	4
Orca	1.20.1827.1	Microsoft	12

Figure 6-29: List of Packages in the “Warnings” Category of the ICE Validation Chart

You can then click on the count to see a list of the warnings or errors generated for that package.



Fitch and Mather 7.0: Microsoft MSI ICE Validation Warnings

FLEXERA SOFTWARE™ AdminStudio®

This report lists warnings found in Fitch and Mather 7.0.

Rule	Description
ICE03	String overflow (greater than length permitted in column); Table: CustomAction, Column: Target, Key(s): FMS.Install
ICE45	Row 'DALDataSourceDlg.PasswordEdit' in table 'Control' has bits set in the 'Attributes' column that are not used in the schema of the package, but are used in a later schema. Your package can run on systems where this attribute has no effect.
	Row 'GAMDataSourceDlg.PasswordEdit' in table 'Control' has bits set in the 'Attributes' column that are not used in the schema of the package, but are used in a later schema. Your package can run on systems where this attribute has no effect.
	Row 'GAMDataSourceDlg.GAMSelection' in table 'Control' has bits set in the 'Attributes' column that are reserved. They should be 0 to ensure compatibility with future installer versions.

Figure 6-30: ICE Warnings Generated for a Specific Windows Installer Package

The Windows Installer Best Practices area includes the following charts:

- ICE Validation Chart
- Best Practices Chart
- Conflicts Chart

ICE Validation Chart

The **ICE Validation** chart lists the current status of Windows Installer packages in the Application Catalog regarding Microsoft ICE validation. Validation—which can either be performed during import or by selecting a Windows Installer package in the Application Manager tree and clicking **Validate Package** from the context menu—compares the package against Microsoft ICEs (Internal Consistency Evaluators), custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards. A Windows Installer package would fall into one of the following validation categories:

Table 6-33 • Validation Categories

Category	Description
Errors	This package contains an error in database authoring that causes incorrect behavior. An example of an error would be duplicate component GUIDs, which would cause the Installer to incorrectly register components.

Table 6-33 • Validation Categories

Category	Description
Warnings	This package contains an issue in database authoring that can cause incorrect behavior. Warnings can also report unexpected side-effects of database authoring. An example of a warning would be entering the same property name in two conditions that differ only by the case of letters in the name. Because the installer is case-sensitive, the installer treats these as different properties.
No Errors	Validation resulted in no errors or warnings.
Not Tested	Validation has not yet been performed on this package.

Note • *If all of a package's detected ICE errors are suppressed, the package is listed in the **No Errors** category. For more information, see Suppressing an ICE Error.*

Note • *For more information on validation, see Performing Validation and Validation View.*

Best Practices Chart

The **Best Practices** chart lists the current status of Windows Installer packages in the Application Catalog regarding Best Practice ACE rules. Best Practice ACEs are rules that internally perform checks against the structure of a Windows Installer package to enforce Microsoft best practices. Best Practice ACEs are similar to Microsoft ICEs. Both Microsoft ICEs (custom actions written by Microsoft used during validation) and Best Practice ACEs (custom actions included with ConflictSolver used during conflict analysis) are used to determine if an installation package is built according to Windows Installer standards.

When testing an installation package against a Best Practice ACE Rule, there is no target package involved. The source installation package is tested to see if it meets the standards defined in the ACEs. Best Practice testing is performed using the Conflict Wizard.

After Best Practices testing, a Windows Installer package would fall into one of the following categories:

Table 6-34 • Best Practices Rule Categories

Category	Description
Errors	Best Practices testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Best Practices testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).
No Errors	Best Practices testing resulted in no errors or warnings.
Not Tested	Best Practices testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the* ***Errors*** *category. For more information, see Resolving Conflicts, Automatically Resolving Conflicts, and Suppressing ACE Errors.*

Note • *For more information on Best Practice testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

Conflicts Chart

The **Conflicts** chart lists the current status of Windows Installer packages in the Application Catalog regarding Conflict ACE rules. Conflict ACE Rules are used to detect conflicts between two packages. You can use Conflict ACE Rules to detect conflicts between one or more source packages (Windows Installer and App-V) and one or more target packages in the Application Catalog. Conflict evaluation is done for each source package against each target package.

Conflict testing is performed using the Conflict Wizard.

After Conflict testing, a Windows Installer package would fall into one of the following categories:

Table 6-35 • Conflict Rule Categories

Category	Description
Errors	Conflict testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Conflict testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).
No Errors	Conflict testing resulted in no errors or warnings.
Not Tested	Conflict testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the* ***Errors*** *category. For more information, see Resolving Conflicts, Automatically Resolving Conflicts, and Suppressing ACE Errors.*

Note • *For more information on Conflict testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

Virtualization Best Practices

The **Virtualization Best Practices** area of the Application Readiness Dashboard lists the AdminStudio Best Practices and Conflict test status of App-V packages.

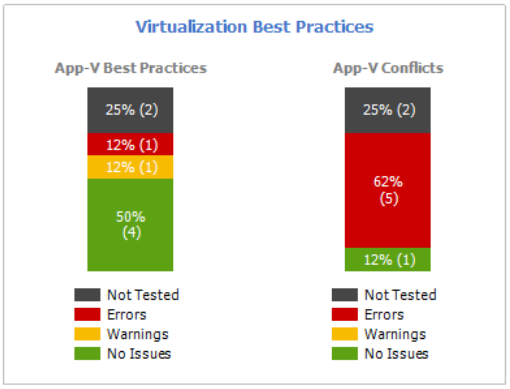


Figure 6-31: Virtualization Best Practices Charts of the Application Readiness Dashboard

If you click on one of the categories in these charts, you see a list of packages that fall into that category.

App-V Packages
AdminStudio Conflicts - Failed With Errors

FLEXERA SOFTWARE™ AdminStudio®

This report lists App-V packages that have errors after AdminStudio App-V Conflict testing.

Product Name	Product Version	Manufacturer	Error Count	Warning Count
Acord Forms Lite	6.6	FileMaker	2	0
Adobe Reader 8	8.0.0	Adobe Systems Incorporated	6	0
Application Verifier 3.3	3.3.45	Microsoft Corporation	1	0
Messenger	8.1.0178	Microsoft Corporation	1	0
Orca	1.50.2600.0000	Microsoft Corporation	5	0

Figure 6-32: List of Packages in the “Errors” Category of the App-V Conflicts Chart

You can then click on the count to see a list of the ACE warnings or errors generated for that package.

Adobe Reader 8: AdminStudio App-V Conflict Errors

FLEXERA SOFTWARE™ AdminStudio®

This report lists errors found in Adobe Reader 8.

Rule	Description
ACE205	package 'Adobe Reader 8' has a name conflict with package 'Adobe Reader 8'
ACE206	package 'Adobe Reader 8' has a conflicting file extension 'fdf' or progid 'AcroExch.FDFDoc' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'pdx' or progid 'PDXFileType' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'rmf' or progid 'AcroExch.RMFFile' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'xdp' or progid 'AcroExch.XDPDoc' with package 'Adobe Reader 8'
	package 'Adobe Reader 8' has a conflicting file extension 'xfdf' or progid 'AcroExch.XFDFDoc' with package 'Adobe Reader 8'

Figure 6-33: ACE Conflict Errors Generated for a Specific App-V Package

The App-V Best Practices area includes the following charts:

- App-V Best Practices Chart
- App-V Conflicts Chart

App-V Best Practices Chart

The App-V **Best Practices** chart lists the current status of App-V packages in the Application Catalog regarding Best Practice ACE rules. Best Practice ACEs (custom actions included with ConflictSolver) are rules that internally perform checks against the structure of an App-V package to enforce best practices.

When testing an installation package against a Best Practice ACE Rule, there is no target package involved. The source installation package is tested to see if it meets the standards defined in the ACEs. Best Practice testing is performed using the Conflict Wizard.

After Best Practices testing, an App-V package would fall into one of the following categories:

Table 6-36 • Best Practices Rule Categories

Category	Description
Errors	Best Practices testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Best Practices testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).
No Errors	Best Practices testing resulted in no errors or warnings.
Not Tested	Best Practices testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the **Errors** category. For more information, see Resolving Conflicts, Automatically Resolving Conflicts, and Suppressing ACE Errors.*

Note • *For more information on Best Practice testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

App-V Conflicts Chart

The **Conflicts** chart lists the current status of App-V packages in the Application Catalog regarding Conflict ACE rules. Conflict ACE Rules are used to detect conflicts between two packages. You can use Conflict ACE Rules to detect conflicts between one or more source App-V packages and one or more target packages (Windows Installer and App-V) in the Application Catalog. Conflict evaluation is done for each source package against each target package.

Conflict testing is performed using the Conflict Wizard.

After Conflict testing, an App-V package would fall into one of the following categories:

Table 6-37 • Conflict Rule Categories

Category	Description
Errors	Conflict testing resulted in errors. Errors should normally not be ignored, or at least an active determination should be made whether it is safe to ignore certain error rules in general or specifically for a given package. • **Suppress**—If you determine that a particular error can be safely ignored for a specific package, then you should suppress it, as described in Suppressing ACE Errors. If all of a package's detected errors are suppressed, the package will no longer be listed in the **Errors** category. • **Not run**—If you determine that a particular error can be ignored safely for all packages, then you should consider not running that particular rule, as described in Changing Default Conflict Types Checked.
Warnings	Conflict testing resulted in warnings. Warnings can be safely ignored (even though they are a violation of best practices).
No Errors	Conflict testing resulted in no errors or warnings.
Not Tested	Conflict testing has not yet been performed on this package.

Note • *If all of a package's detected errors are resolved or if they are suppressed, the package will no longer be listed in the **Errors** category. For more information, see Resolving Conflicts and Suppressing ACE Errors.*

Note • *For more information on Conflict testing, see Types of ACE Rules: Best Practice and Conflict and Perform Conflict Detection.*

Application Compatibility Dashboard

Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.*

Edition • *The AdminStudio Application Readiness Dashboard and other enhanced reports are included with AdminStudio Enterprise Edition.*

The **Application Compatibility Dashboard** provides a snapshot of the compatibility test results of the currently imported applications in Compatibility Solver. This report not only displays Windows 7 application compatibility test results, but also Windows 64-bit, Windows Server 2008 and 2008 R2, Internet Explorer 7-8, and Internet Explorer 8 Presentation compatibility issues.

To open the Application Compatibility Dashboard, select the **Application Catalog** node in the Application Manager tree or open the AdminStudio **Report Center** tab, and then select **Application Compatibility Dashboard** from the **Select Report** list.

Note • *Before you can view the Application Compatibility Dashboard, you need to specify the location of the Compatibility Solver database on the **Application Catalog** tab of the AdminStudio **Options** dialog box, as described in Connecting to the Compatibility Solver Database.*

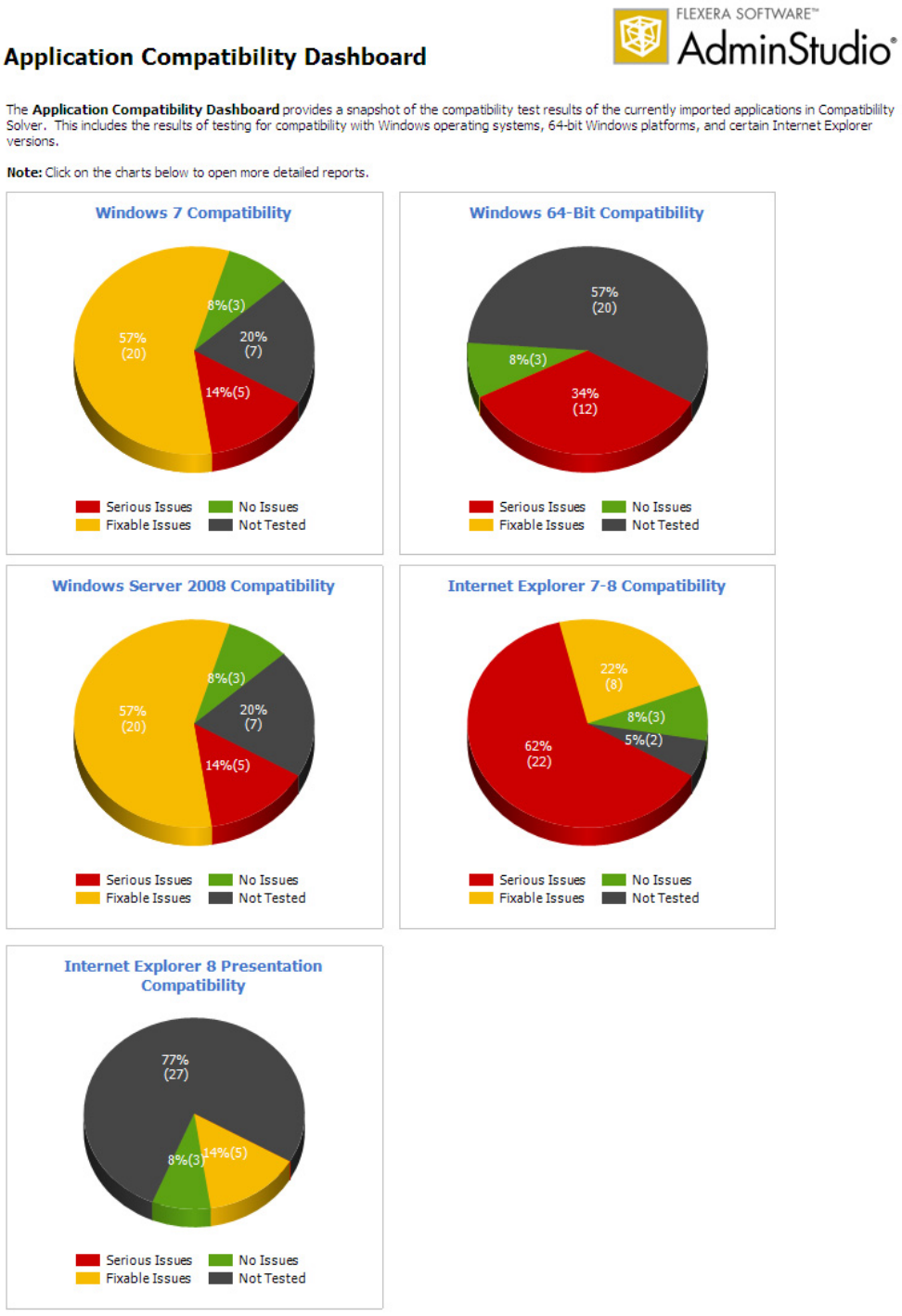


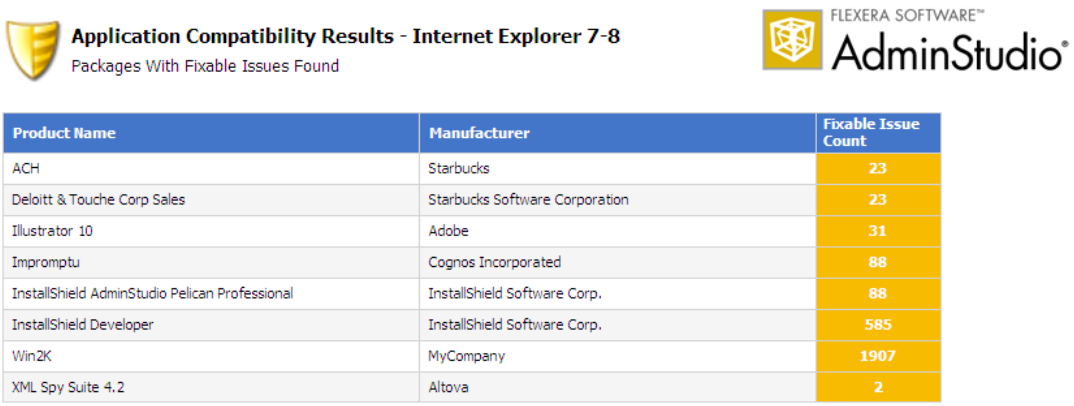
Figure 6-34: Application Compatibility Dashboard

The Application Compatibility Dashboard includes the following charts:

Table 6-38 • Charts on the Application Compatibility Dashboard

Chart	Description
Windows 7 Compatibility	Lists the Windows 7 application compatibility test results of the currently imported applications in Compatibility Solver.
Windows 64-Bit Compatibility	Lists the Windows 64-bit application compatibility test results of the currently imported applications in Compatibility Solver.
Windows Server 2008 Compatibility	Lists the Windows Server 2008 and 2008 R2 application compatibility test results of the currently imported applications in Compatibility Solver.
Internet Explorer 7-8 Compatibility	Lists the Internet Explorer 7-8 application compatibility test results of the currently imported applications in Compatibility Solver.
Internet Explorer 8 Presentation Compatibility	Lists the Internet Explorer 8 presentation application compatibility test results of the currently imported applications in Compatibility Solver.

To see a list of packages that fall into a category, click on that category.



Product Name	Manufacturer	Fixable Issue Count
ACH	Starbucks	23
Deloitt & Touche Corp Sales	Starbucks Software Corporation	23
Illustrator 10	Adobe	31
Impromptu	Cognos Incorporated	88
InstallShield AdminStudio Pelican Professional	InstallShield Software Corp.	88
InstallShield Developer	InstallShield Software Corp.	585
Win2K	MyCompany	1907
XML Spy Suite 4.2	Altova	2

Figure 6-35: Internet Explorer 7-8 - Packages With Fixable Issues Found

Note • *In addition to viewing Compatibility Solver test results in the Application Compatibility Dashboard, you can view a summary of Windows 7 Application Compatibility Chart test results in Application Manager's* ***Application Readiness Dashboard****. You can also view Compatibility Solver test results for an individual package in the package's* ***Application Compatibility View****, as described in Viewing Windows Installer Package Application Compatibility Issues.*

Virtualization Readiness Dashboard

The **Virtualization Readiness Dashboard** provides a snapshot of the virtualization readiness status of packages in the Application Catalog for all three supported virtual formats (Microsoft App-V, Citrix XenApp, and VMware ThinApp). This dashboard also shows the best practices and conflict test status of App-V packages in the Application Catalog.

To open the Virtualization Readiness Dashboard, select the **Application Catalog** node in the Application Manager tree or open the AdminStudio **Report Center** tab, and then select **Virtualization Readiness Dashboard** from the **Select Report** list.

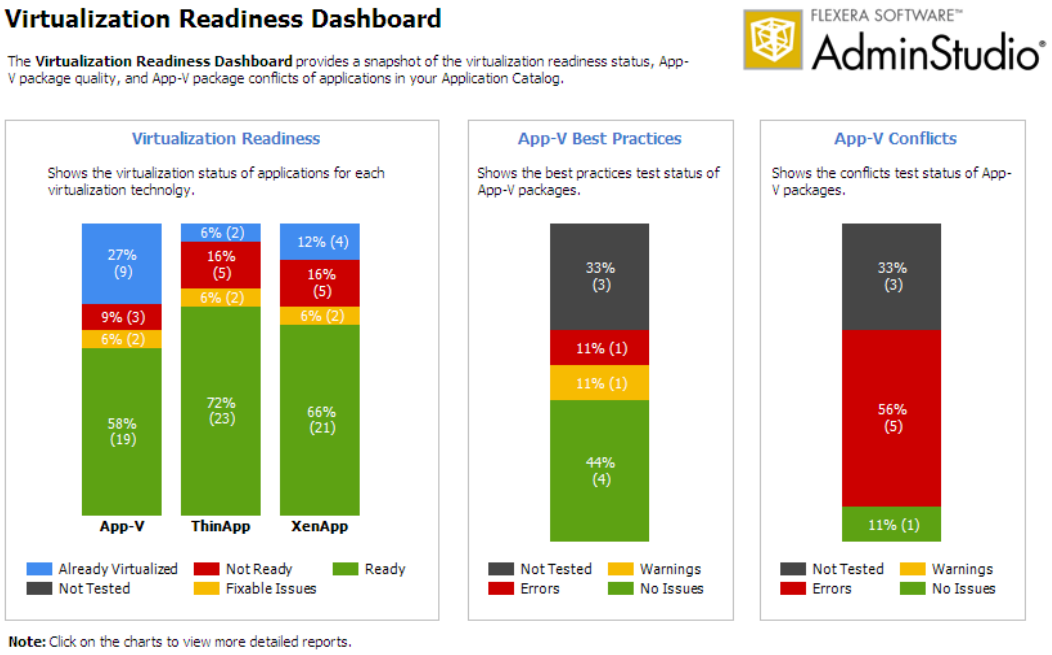


Figure 6-36: Virtualization Readiness Dashboard

Virtualization Readiness Charts

The bar charts in the **Virtualization Readiness** area of the **Virtualization Readiness Dashboard** identify the virtualization status of Windows Installer packages in the Application Catalog for each of the supported virtualization technologies—App-V, ThinApp, and XenApp.

Packages are automatically tested at import and are assigned one of the following statuses: **Already Virtualized**, **Not Tested**, **Not Ready**, **Fixable Issues**, or **Ready**.

You can click on any section of these charts to view a list of the packages in that category.

Note • *Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog. To manually retest a package or group of packages for virtualization readiness, select a package, application, or group in the Application Manager tree and select* ***Run Virtualization Readiness*** *from the context menu.*

Note • *To see a list of any virtualization suitability issues that have been detected for a specific Windows Installer package, see the package's Virtualization Suitability View.*

Note • *For a comparison between virtualization readiness status in the Application Catalog vs. in Automated Application Converter, see Virtualization Readiness Status in Application Catalog vs. Automated Application Converter.*

App-V Best Practices and App-V Conflicts Charts

The bar charts in the **App-V Best Practices** and **App-V Conflicts** areas of the **Application Readiness Dashboard** summarize the results of best practice testing and conflict analysis of App-V packages in the Application Catalog.

Packages are tested either automatically at import or by using the Conflict Wizard, and are assigned one of the following statuses: **Not Tested**, **Errors**, **Warnings**, or **No Issues**.

You can click on any section of these charts to view a list of the App-V packages in that category.

Error Category Breakdown

The Error Category Breakdown report, which you open by selecting **Error Category Breakdown** from the **Select Report** list of the Application Catalog View, shows the number of AdminStudio Best Practices and Conflict errors that have been identified in Windows Installer and App-V packages, by category.

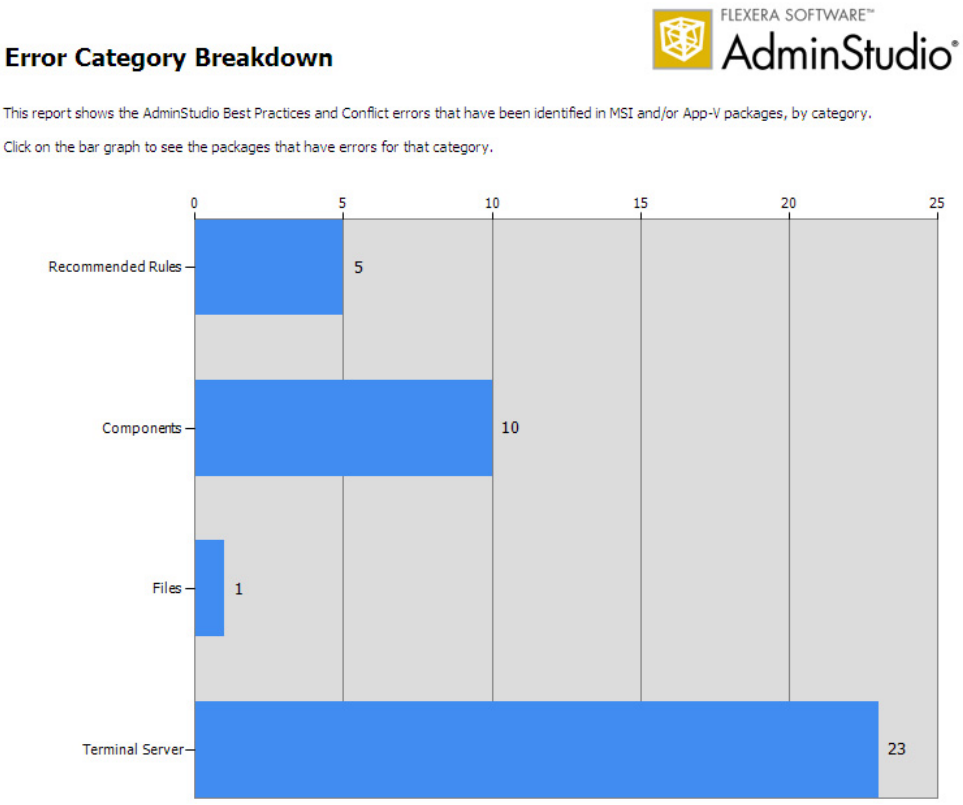
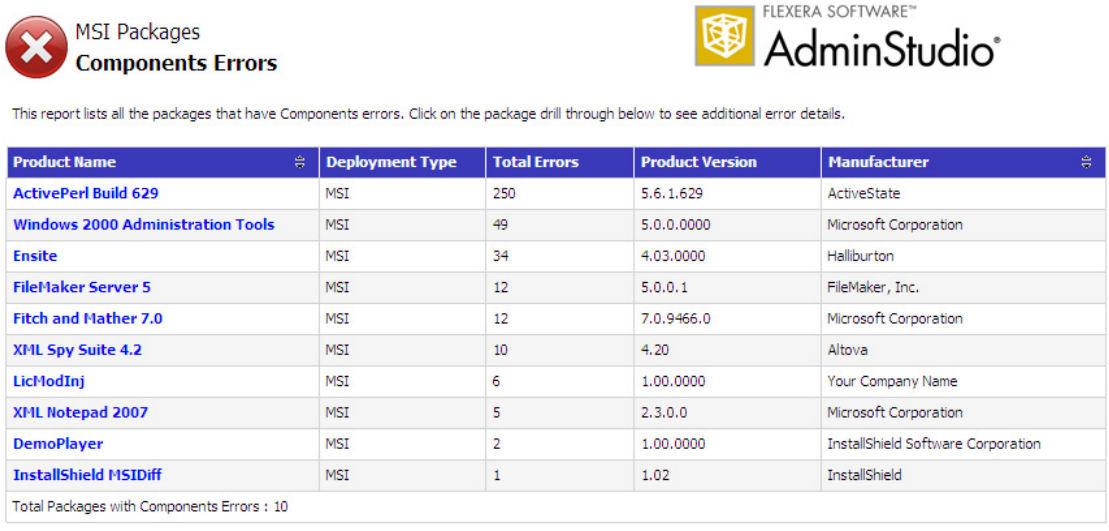


Figure 6-37: Error Category Breakdown

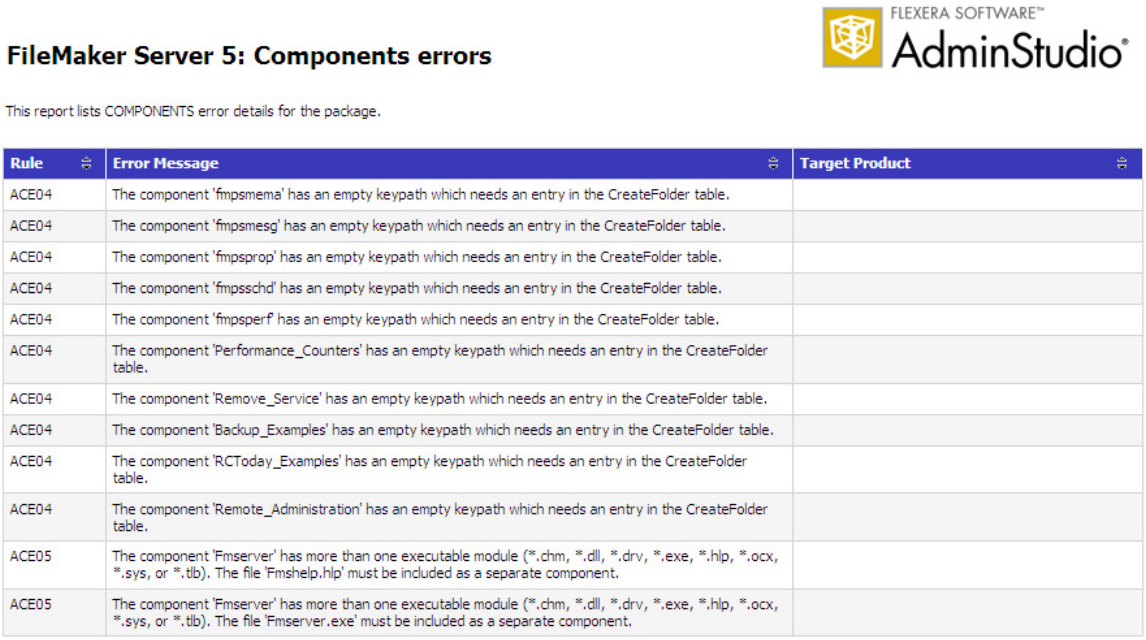
To see a report that lists the packages that have errors for that category, click on a category.



Product Name	Deployment Type	Total Errors	Product Version	Manufacturer
ActivePerl Build 629	MSI	250	5.6.1.629	ActiveState
Windows 2000 Administration Tools	MSI	49	5.0.0.0000	Microsoft Corporation
Ensite	MSI	34	4.03.0000	Halliburton
FileMaker Server 5	MSI	12	5.0.0.1	FileMaker, Inc.
Fitch and Mather 7.0	MSI	12	7.0.9466.0	Microsoft Corporation
XML Spy Suite 4.2	MSI	10	4.20	Altova
LicModInj	MSI	6	1.00.0000	Your Company Name
XML Notepad 2007	MSI	5	2.3.0.0	Microsoft Corporation
DemoPlayer	MSI	2	1.00.0000	InstallShield Software Corporation
InstallShield MSIDiff	MSI	1	1.02	InstallShield
Total Packages with Components Errors : 10				

Figure 6-38: MSI Packages Components Errors Report

To see a report that lists the specific errors for a package, click on the **Product Name**:



FileMaker Server 5: Components errors

FLEXERA SOFTWARE™ AdminStudio®

This report lists COMPONENTS error details for the package.

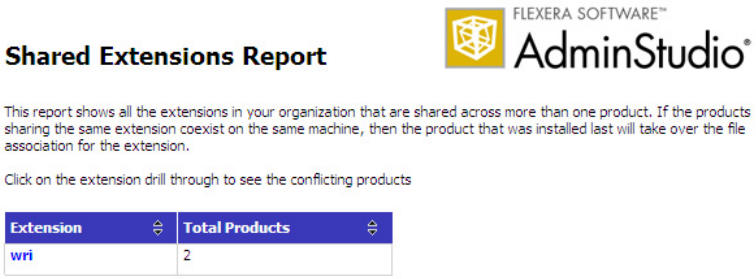
Rule	Error Message	Target Product
ACE04	The component 'fmpsmema' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsmesg' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsprop' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsschd' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'fmpsperf' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Performance_Counters' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Remove_Service' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Backup_Examples' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'RCToday_Examples' has an empty keypath which needs an entry in the CreateFolder table.	
ACE04	The component 'Remote_Administration' has an empty keypath which needs an entry in the CreateFolder table.	
ACE05	The component 'Fmserver' has more than one executable module (*.chm, *.dll, *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb). The file 'Fmshelp.hlp' must be included as a separate component.	
ACE05	The component 'Fmserver' has more than one executable module (*.chm, *.dll, *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb). The file 'Fmserver.exe' must be included as a separate component.	

Figure 6-39: Components Errors for a Specific Product

Note • *In AdminStudio 10.0, the **Error Category Breakdown** chart was displayed within the Application Readiness Dashboard. Starting in AdminStudio 10.0 SP1, the Error Category Breakdown is displayed in a separate report.*

Shared Extensions Report

The **Shared Extensions Report**, which you open by selecting **Shared Extensions** from the **Select Report** list of the Application Catalog View, lists all of the extensions used in the Application Catalog that are shared across more than one product.



Shared Extensions Report

FLEXERA SOFTWARE™ AdminStudio®

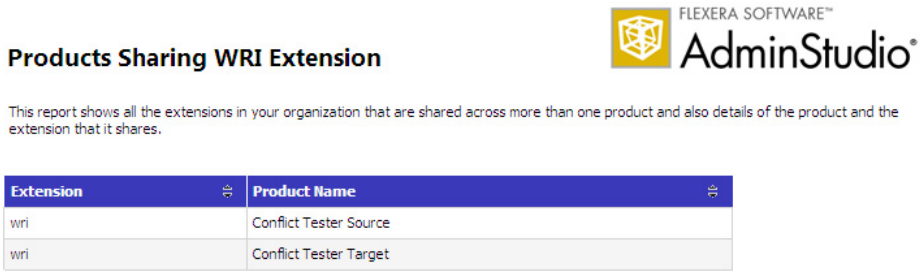
This report shows all the extensions in your organization that are shared across more than one product. If the products sharing the same extension coexist on the same machine, then the product that was installed last will take over the file association for the extension.

Click on the extension drill through to see the conflicting products

Extension	Total Products
wri	2

Figure 6-40: Shared Extensions Report

To see a report that lists the conflicting products, click on an extension in the list.



Products Sharing WRI Extension

FLEXERA SOFTWARE™ AdminStudio®

This report shows all the extensions in your organization that are shared across more than one product and also details of the product and the extension that it shares.

Extension	Product Name
wri	Conflict Tester Source
wri	Conflict Tester Target

Figure 6-41: Products That Contain Shared Extension

Note • *If the products sharing the same extension coexist on the same machine, then the product that was installed last will take over the file association for the extension.*

App-V Dependencies Report

The **App-V Dependencies** report, which you open by selecting **App-V Dependencies** from the **Select Report** list of the Application Catalog View, lists all of the App-V packages in the Application Catalog that have dependent App-V packages through the Dynamic Suite Composition feature of App-V.

App-V Dependencies

FLEXERA SOFTWARE™ AdminStudio®

This report lists all Applications in the Application Catalog with its dependencies.

Application	Dependency Applica	In Catalog?	DSC Server URL	Server URL	Mandatory?
Ad-Aware SE Personal	HyperCam	Yes	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	Yes
	ListMaker v1.15.5	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	Yes
	Winzip	Yes	RTSP://%SFT_SOFTGRIDSERVER %:554/Winzip/Winzip.sft	RTSP://%SFT_SOFTGRIDSERVER %:554/Winzip/Winzip.sft	Yes
AppVSample		No	RTSP://%SFT_SOFTGRIDSERVER %:554/Firefox/Firefox.sft		No
FontTest	JavaJRE	Yes	FILE://C:\Users\Test\Desktop\JavaJRE - VPE \JavaJRE_2.sft	rtsps://%SFT_SOFTGRIDSERVER %:322/JavaJRE.sft/JavaJRE_2.sft/JavaJRE_3.sft/JavaJRE_4.sft	Yes
HyperCam	Ad-Aware SE Personal	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	snagit	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/snagit/snagit.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/snagit/snagit.sft	Yes
ListMaker v1.15.5	Ad-Aware SE Personal	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	ListMaker v1.15.5	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	Yes

Figure 6-42: App-V Dependencies Report

App-V Dependents Report

The **App-V Dependents** report, which you open by selecting **App-V Dependents** from the **Select Report** list of the Application Catalog View, lists all of the App-V packages in the Application Catalog that are dependent upon other App-V packages through the Dynamic Suite Composition feature of App-V.

App-V Dependents

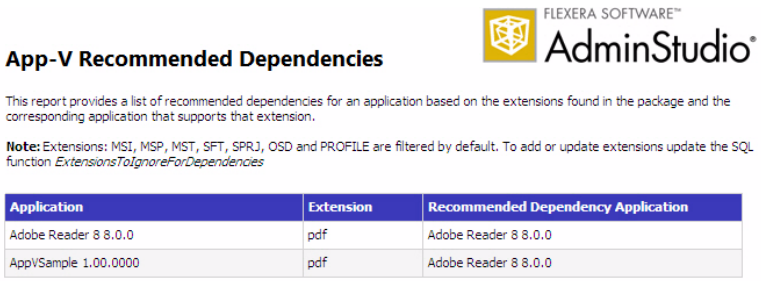
This report lists all Applications in the Application Catalog and its dependent applications.

Application	Dependent Application	In Catalog?	DSC Server URL	Server URL	Mandatory?
Ad-Aware SE Personal	HyperCam	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	ListMaker v1.15.5	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
	snagit	Yes	FILE://C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	FILE:///C:\AdAware\App-VPackage\Ad-Aware SE Personal.sft	Yes
Adobe Reader 8	Sample Kit	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/Adobe Reader 8/Adobe Reader 8_2.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/Adobe Reader 8/Adobe Reader 8_2.sft	Yes
Application Verifier 3.3	Sample Kit	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/Application Verifier 3/Application Verifier 3.3.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/Application Verifier 3/Application Verifier 3.3.sft	Yes
FontTest	Sample Kit	Yes	RTSPS://%SFT_SOFTGRIDSERVER %:322/FontTest.sft/FontTest_4.sft	RTSPS://%SFT_SOFTGRIDSERVER %:322/FontTest.sft/FontTest_4.sft	Yes
HyperCam	Ad-Aware SE Personal	Yes	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	RTSP://%SFT_SOFTGRIDSERVER %:554/HyperCam/HyperCam.sft	Yes
JavaJRE_4	FontTest	Yes	FILE://C:\Users\Test\Desktop\JavaJRE - VPE \JavaJRE_2.sft	rtsps://%SFT_SOFTGRIDSERVER %:322/JavaJRE.sft/JavaJRE_2.sft/JavaJRE_3.sft/JavaJRE_4.sft	Yes
ListMaker v1.15.5	Ad-Aware SE Personal	Yes	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	RTSP://%SFT_SOFTGRIDSERVER%:554/ListMaker v1.15/ListMaker v1.15.5.sft	Yes

Figure 6-43: App-V Dependents Report

App-V Recommended Dependencies Report

The **App-V Recommended Dependencies** report, which you open by selecting **App-V Recommended Dependencies** from the **Select Report** list of the Application Catalog View, lists the recommended Dynamic Suite Composition dependencies for App-V packages based on cross-referencing the file extensions found in them with published file type associations from other App-V packages in the catalog.



App-V Recommended Dependencies

This report provides a list of recommended dependencies for an application based on the extensions found in the package and the corresponding application that supports that extension.

Note: Extensions: MSI, MSP, MST, SFT, SPRJ, OSD and PROFILE are filtered by default. To add or update extensions update the SQL function *ExtensionsToIgnoreForDependencies*

Application	Extension	Recommended Dependency Application
Adobe Reader 8 8.0.0	pdf	Adobe Reader 8 8.0.0
AppVSample 1.00.0000	pdf	Adobe Reader 8 8.0.0

Figure 6-44: App-V Recommended Dependencies Report

Customizable Reports

In AdminStudio 10.0, you can create your own XML-driven custom reports that are generated using Microsoft SQL Reporting Services. To create a custom report, you just need to create an XML file in which you specify the RDLC file, the data source, and the parameters that are required for the drill through reports.

For more information, see Creating Customizable Reports.

Application Catalog View (Professional Edition)

The Application Catalog View (Professional Edition) is displayed on the right side of the Professional Edition of Application Manager whenever the Application Catalog node is selected.

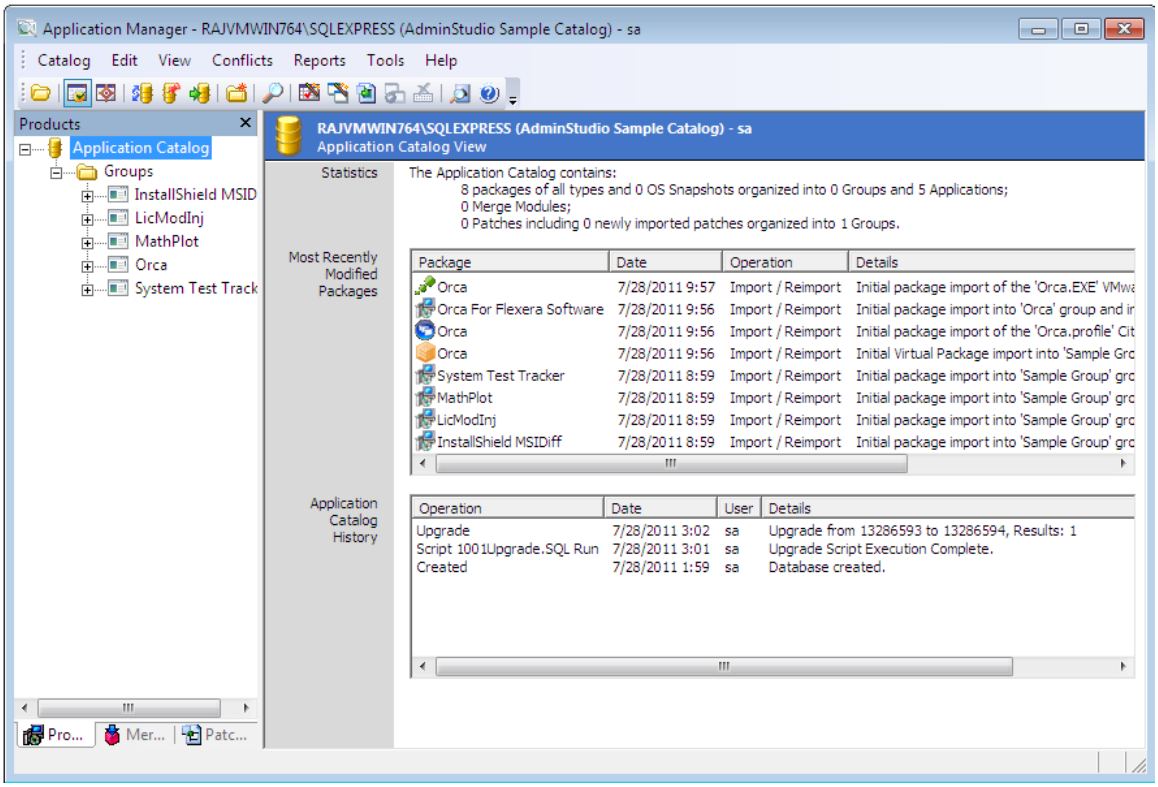


Figure 6-45: Application Catalog View (Professional Edition)

The following information is displayed:

Table 6-39 • Application Catalog View (Professional Edition)

Item	Description
Name	The name of the Application Catalog is displayed in the title bar.

Table 6-39 • Application Catalog View (Professional Edition)

Item	Description
Statistics	The number of packages, merge modules, OS Snapshots, patches, and groups is listed.
Most Recently Modified Packages	Packages that have been modified recently are listed with the most recently modified package listed first. If you double-click on a package, the Product View for that package is opened.
Application Catalog History	Operations that have been performed recently on the Application Catalog are listed, with the most recently performed listed first.

Note • *For information on the Application Catalog View that is displayed in AdminStudio Enterprise Edition, see Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards.*

Group View

The Group view is displayed on the right side of Application Manager whenever a group is selected.

If you select a group or product in a group, properties for that group or product are displayed in the Group View.

Additionally, if you double-click an item in the Group View, the corresponding item in the Application Manager Product View is selected. You can sort the products by either Name or Description by clicking on the column title.

Product View

When you select a Windows Installer Package (.msi) in the tree, details about that product are displayed in the right pane of the user interface.

The following information is included:

Table 6-40 • Application Manager Product View

Field	Description
Manufacturer	The manufacturer of the package.
Subscribed Database	If this package is associated with a subscribed database, the name of that database is listed here. If it is not, the following statement appears: `Not associated with any subscribed database.`
Conflict Results	If any persisted conflict data exists, this is a hyperlink to the package's Conflicts View.

Table 6-40 • Application Manager Product View

Field	Description
Package Code	The globally unique identifier (GUID) for the setup package.
Product Code	A string that uniquely identifies the product.
Upgrade Code	A string used to upgrade the application. The upgrade code for a package groups that package into a specific product family.
Language	The language of this package.
File	Identifies the location of this package. It can be either a hard-coded path or a UNC path. If the package is part of the Software Repository, the following statement appears: `Managed within the Software Repository`
Original File	When a user selects a Windows Installer package (`.msi`) along with one or more patch files (`.msp`) to import, AdminStudio first performs an administrative installation to merge the `.msi` and `.msp` files into one `.msi` file, and then imports the merged `.msi` file into the Application Catalog. In this instance, the **Original File** field lists the name and path of the original Windows Installer package that the patches were applied to, while the **File** field lists the name and path of the merged `.msi` file that was imported. If the package is part of the Software Repository, the following statement appears: Managed within the Software Repository ***Note*** • *When a Windows Installer package that was imported without a patch is selected, the entry of the **Original File** and **File** fields is identical.*
Imported On	The date and time the package was imported.
Transforms	This can be either a hard-coded path or a UNC path.
Patches	The patch files that were used to get to the final imported MSI package.
Groups	Any groups to which the package belongs.
Description	You can edit this with additional information about the product.

Table 6-40 • Application Manager Product View

Field	Description
View Package Version History	If you are connected to a Software Repository-enabled Application Catalog and you select a package that has more than one version, a **View Package Version History** link appears in the header of this view. If you click on this link, the **Package Versions** dialog box opens, listing all of the versions of the selected package.

Note • *If the package and/or transforms are no longer in their original import directory, you can locate the file(s) from the provided hyperlink. You are also informed if the last modified date for the package in the Application Catalog does not match the last modified date of the package in its external location. You are given the opportunity to reimport the package to keep it synchronized in the Application Catalog.*

If you click on the plus sign to expand a package in the Application Manager Product View, a node is listed for each available constituent view. For example, for a Windows Installer package, the following nodes are listed:

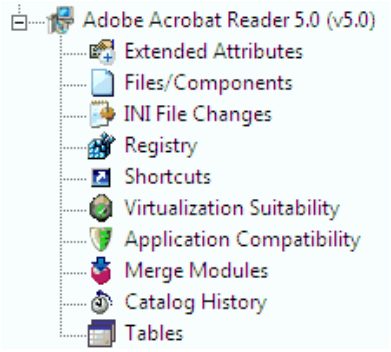


Figure 6-46: Windows Installer Package Nodes

When you select one of these nodes, a constituent view opens in the right pane:

- Extended Attributes View
- Files/Components View
- INI File Changes View
- Registry View
- Shortcuts View
- Virtualization Suitability View
- Application Compatibility View
- Merge Modules View
- Catalog History View

- Tables View

Note • *The **Only Display View Nodes With Data** option on the **General** tab of the ConflictSolver and Application Manager Options Dialog Box controls whether product nodes (constituent views) appear if no data is contained in that view. If you select the option, products containing views without data will not display those views. For example, if a product has no Dependencies, then the **Dependencies** node is not displayed for that product.*

Extended Attributes View

The Application Manager Extended Attributes View displays the optional extended attributes associated with the package. To open a package's Extended Attributes View, select the **Extended Attributes** node under that package in the package tree.

These attributes are dynamically populated based on an external Extended Attribute Description File (in XML format). You can specify the name and location of this file from the Extended Attributes Tab of the ConflictSolver and Application Manager Options Dialog Box.

On the left side of the view, the name for each attribute is displayed; the right side displays the value for the attribute. These values are in read-only fields, from which you can highlight and copy text, or, in the case of file links, launch the linked file. The file location displayed represents the location from which the file was originally imported; the file is actually stored within the Application Catalog and extracted into a temp directory when you click on the file link. This temp directory is purged when Application Manager closes, as long as the file is not locked by another application or process.

If you click on the attribute name, you can either provide the value in the Group Properties Dialog Box (for Text attributes) or browse for a file in a Browse dialog (for File attributes).

Note • *You can also view and edit a package's extended attributes on the Extended Attributes Page of the ConflictSolver Process Assistant.*

Files/Components View

The **Files/Components View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Files/Components** node, displays the files and components in the package. The following information is displayed:

Table 6-41 • File/Components View Information

Column	Description
Component	Name of component that the file listed in the FileName column is associated with.
FileName	Name of file.
FileSize	Size of the file listed in the FileName column.

Table 6-41 • File/Components View Information

Column	Description
Version	Version of the file listed in the FileName column.
Path	Installation location of the file listed in the FileName column.

INI File Changes View

The **INI File Changes View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **INI File Changes** node, displays any INI file changes made by that package. The following information is displayed:

Table 6-42 • INI File Changes View Information

Column	Description
Component	Name of component that makes an entry in the INI File.
FileName	Name of INI File that the component listed in the Component column makes an entry in.
DirProperty	The directory location where the INI File will be installed.
Section	The section of the INI file where this entry is made.
Key	The Key used in the INI File entry
Value	The Value used in the INI File entry.

Registry View

The **Registry View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Registry** node, displays any registry entries created or changed by the package. The following information is displayed:

Table 6-43 • Registry View Information

Column	Description
Component	The name of the component that is creating a Registry entry.
Root	Default value of Key.
Key	Key of the Registry Entry that this component is making.
Name	Name of the Registry Entry that this component is making.
Value	Value of the Registry Entry that this component is making.

Shortcuts View

The **Shortcuts View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Shortcuts** node, displays any shortcuts created by the package. The following information is displayed:

Table 6-44 • Shortcuts View Information

Column	Description
Component	Name of the component that the shortcut listed in the Name column is associated with.
Name	Name of the shortcut.
Directory_	Directory where the shortcut will exist.
Target	Directory and executable that the shortcut invokes.

Virtualization Suitability View

AdminStudio performs virtualization suitability testing to test a Windows Installer package for suitability to convert it to Microsoft App-V, Citrix XenApp, and VMware ThinApp virtual formats. A Windows Installer package's **Virtualization Suitability View** displays the results of a Virtualization Readiness analysis of that package, listing any virtualization suitability issues that have been detected during testing.

To open the **Virtualization Suitability View**, expand a Windows Installer package in the Application Manager tree and select the **Virtualization Suitability** node.

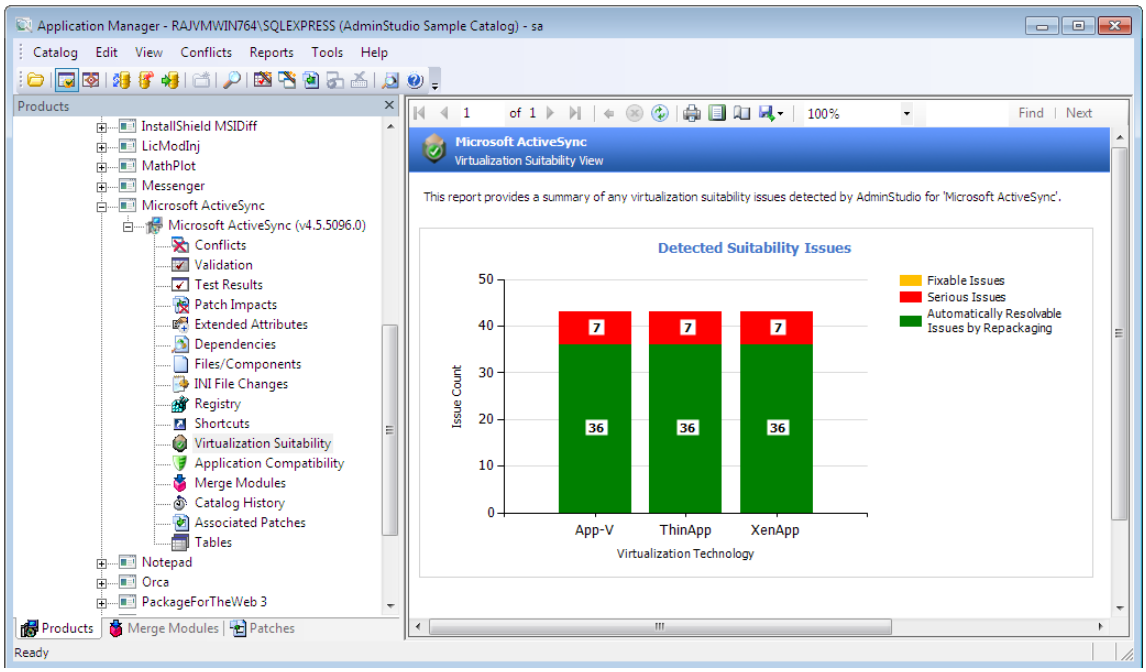


Figure 6-47: Virtualization Suitability View

These bar charts identify the number of issues detected during Virtualization Readiness testing of this Windows Installer package for each of the supported virtualization technologies—App-V, ThinApp, and XenApp.

Issues are grouped into three categories: **Fixable Issues**, **Serious Issues**, and **Automatically Resolvable Issues by Repackaging.**

To view a detailed report of issues in one of the categories, click on one of the categories of the bar charts.

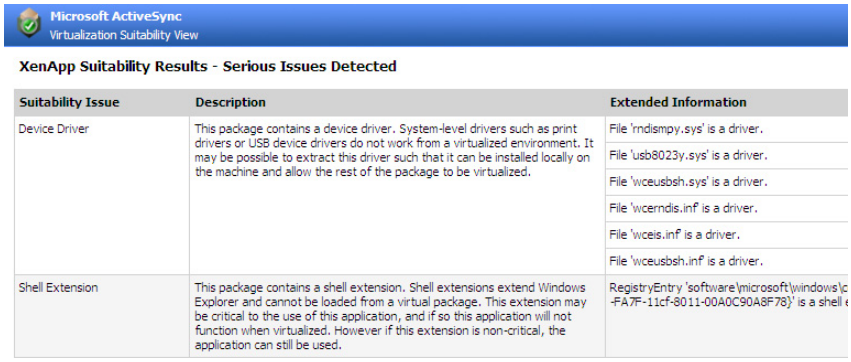


Figure 6-48: XenApp Suitability Results - Serious Issues Detected

Tip • *Packages are automatically tested for virtualization readiness during import into the Application Catalog or when you upgrade an existing Application Catalog. To manually retest a package for virtualization readiness, select a package in the Application Manager tree and select* ***Run Virtualization Readiness*** *from the context menu.*

Application Compatibility View

Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack.*

You can view Compatibility Solver test results for a selected Windows Installer package on the **Application Compatibility View** of a Windows Installer package.

To open the **Application Compatibility View**, expand a Windows Installer package in the Application Manager tree and select the **Application Compatibility** node.

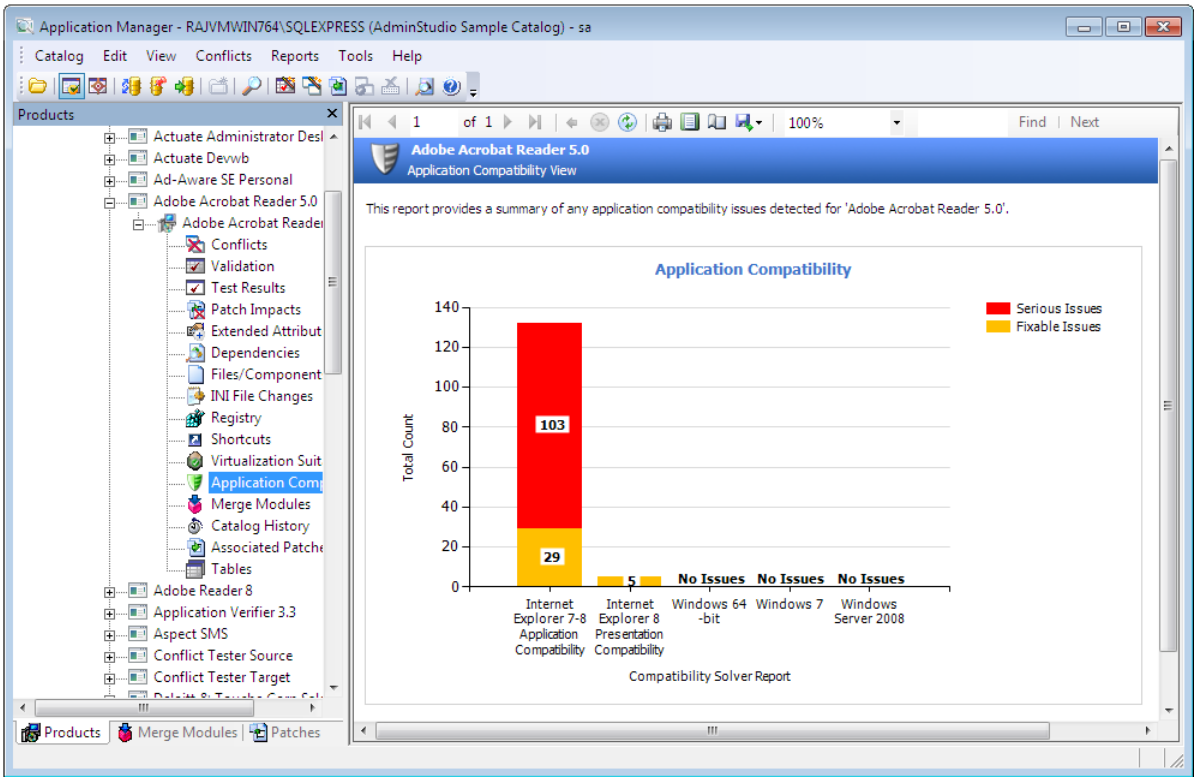


Figure 6-49: Application Compatibility View

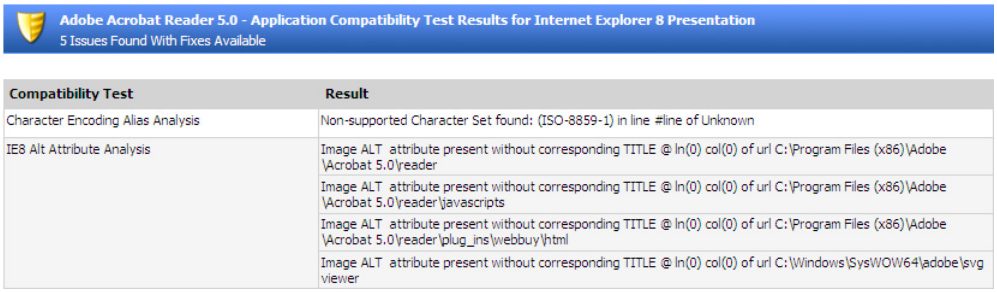
Note • *To view Compatibility Solver test results in AdminStudio, you need to first specify the location of the Compatibility Solver database on the **Application Catalog** tab of the AdminStudio **Options** dialog box, as described in Connecting to the Compatibility Solver Database.*

The **Application Compatibility View** includes the following information:

- **Serious and Fixable issues**—This view lists the number of serious and fixable issues for the following sets of compatibility tests: Windows 64-bit, Windows 7, Windows Server 2008 and 2008 R2, Internet Explorer 7-8 Application Compatibility, and Internet Explorer 8 Presentation Compatibility. If no issues were found, `No Issues` is displayed.

- **No tests run in Compatibility Solver**—If the package has been loaded into Compatibility Solver, but no tests have been run, `Not Tested` is displayed.
- **Package not loaded in Compatibility Solver**—If the package has not been loaded into Compatibility Solver, no results will be displayed.

To open more detailed reports, click on any section of a bar chart.



Adobe Acrobat Reader 5.0 - Application Compatibility Test Results for Internet Explorer 8 Presentation
5 Issues Found With Fixes Available

Compatibility Test	Result
Character Encoding Alias Analysis	Non-supported Character Set found: (ISO-8859-1) in line #line of Unknown
IE8 Alt Attribute Analysis	Image ALT attribute present without corresponding TITLE @ ln(0) col(0) of url C:\Program Files (x86)\Adobe\Acrobat 5.0\reader
	Image ALT attribute present without corresponding TITLE @ ln(0) col(0) of url C:\Program Files (x86)\Adobe\Acrobat 5.0\reader\javascripts
	Image ALT attribute present without corresponding TITLE @ ln(0) col(0) of url C:\Program Files (x86)\Adobe\Acrobat 5.0\reader\plug_ins\webbuy\html
	Image ALT attribute present without corresponding TITLE @ ln(0) col(0) of url C:\Windows\SysWOW64\adobe\svg viewer

Figure 6-50: Application Compatibility Test Results for Internet Explorer 8 Presentation - Issues Found with Fixes Available

Merge Modules View

The **Merge Modules View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Merge Modules** node, displays any merge modules included the package. The following information is displayed:

Table 6-45 • Merge Modules View Information

Column	Description
Title	The title of the Merge Module included with this package.
ModuleID	The number which uniquely identifies the Merge Module listed in the Title column.
Version	The version of the Merge Module listed in the Title column.
Language	The language that the Merge Module listed in the Title column was written for.

Catalog History View

The **Catalog History View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Catalog History** node, displays a list of logged events for that package. The following information is included:

Table 6-46 • Information Displayed in the Catalog History View

Item	Description
Action	Name of the event which was logged: • Import/Reimport • Validation • Conflict Detection • Conflict Resolution • Extended Attribute Modification • Package Description Modification • Package Move/Copy • Patch Analysis ***Note*** *• You can specify which events you want to be listed on the* ***Catalog History*** *view by making selections on the History Tab of the ConflictSolver and Application Manager Options Dialog Box. You can choose which events you want to log and which events you want to display in the Catalog* ***History*** *view.*
Date	Date and time logged event occurred.

Table 6-46 • Information Displayed in the Catalog History View (cont.)

Item	Description
User	User who performed the logged event.
Description	Description providing details of the logged event.

Tables View

The **Tables View**, which you open by expanding a Windows Installer package in the Application Manager tree and selecting the **Tables** node, provides a way to view the data for a given package in the Application Catalog.

Select the specific table you want to view from the **Tables** list at the top of the view.

Most tables are derived directly from standard MSI tables, as described in the Windows Installer SDK online help. When building your own ACE rules to use for conflict identification, it is important to understand the data available for packages so you can construct the necessary rule.

App-V Package View

When you select an App-V package in the Application Catalog, the **App-V Package View** opens. The following information is displayed:

Table 6-47 • App-V Package View

Option	Description
Type	Identifies the virtual package type as **App-V**.
File	Identifies the location of this package. It can be either a hard-coded path or a UNC path. If the package is part of the Software Repository, the following statement appears: `Managed within the Software Repository`
Imported On	Date and time the virtual package was imported into the Application Catalog.
Applications	Lists the Applications that reference this App-V package.

Table 6-47 • App-V Package View

Option	Description
Associated MSI Packages	Lists the App-V package's source Windows Installer package. By associating a virtual package with the Windows Installer package which originated it, you have the convenience of being able to easily locate the virtual package's originating Windows Installer package, modify the original Windows Installer package, and then regenerate the virtual package. ***Note*** • *To quickly locate one of these listed Windows Installer packages in the Application Catalog, double-click on it.* In order to be listed here, the Windows Installer package has to have already been imported into the Application Catalog and must be associated with this virtual package. There are several ways to associate a virtual package with its source Windows Installer package: • Importing a Virtual Package During the Import of its Source Windows Installer Package • Importing a Virtual Package After the Import of its Source Windows Installer Package • Manually Associating a Virtual Package with a Windows Installer Package
Description	You can edit this field and add additional information about the virtual package.
View Package Version History	If you are connected to a Software Repository-enabled Application Catalog and you select a package that has more than one version, a **View Package Version History** link appears in the header of this view. If you click on this link, the **Package Versions** dialog box opens, listing all of the versions of the selected package.

If you expand the App-V node in the tree, you can view its constituent views:

- **Extended Attributes**—The Extended Attributes view for an App-V package is identical to the Extended Attributes View shown when a Windows Installer package is selected. See Extended Attributes View in the Application Manager Product View section.
- **App-V History**—See App-V History View.
- **Dependencies**—See App-V Dependencies View.
- **Files/Directories**—See App-V Files/Directories View.
- **Registry**—See App-V Registry View.
- **Shortcuts**—See App-V Shortcuts View.
- **File Type Associations**—See App-V File Type Associations View.
- **Environment Variables**—See App-V Environment Variables View.

- **Catalog History**—The Catalog History view for an App-V, XenApp, or ThinApp virtual package is identical to the Catalog History view shown when a Windows Installer package is selected. See Catalog History View in the Application Manager Product View section.
- **Tables**—See App-V Tables View.

App-V History View

The **App-V History View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **App-V History** node, lists an entry for each time this App-V package was saved.

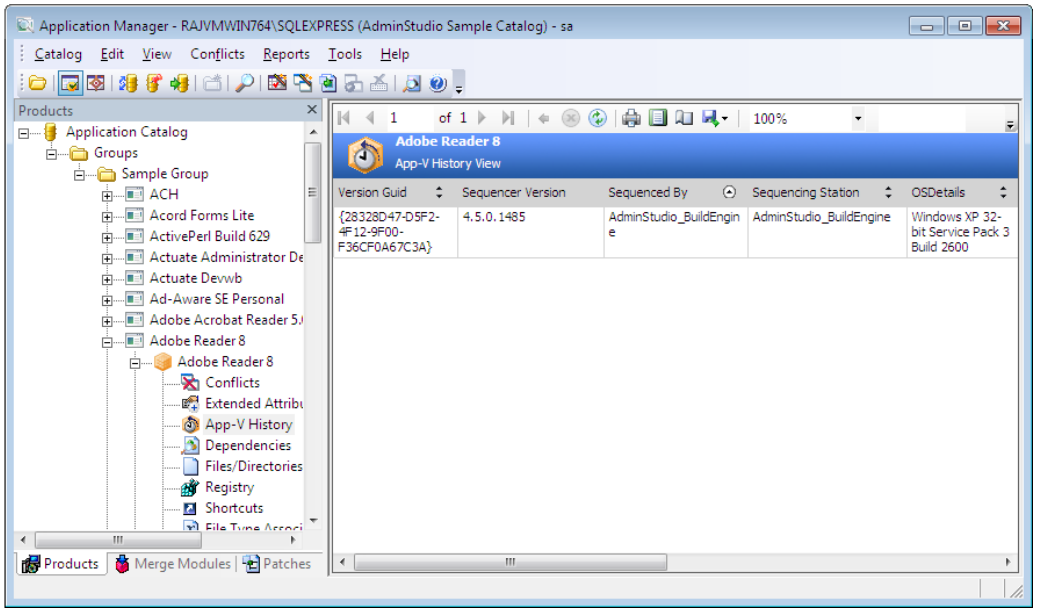


Figure 6-51: App-V History View

For each entry, the following information is displayed:

- Version Guid
- Sequencer Version
- Sequenced By
- Sequencing Station
- OSDetails
- System Folder
- Windows Folder
- User Folder
- .Net Framework Version
- IEVersion

App-V Dependencies View

The **App-V Dependencies View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **Dependencies** node, lists both the applications the App-V package is dependent on and the applications dependent upon this App-V package.

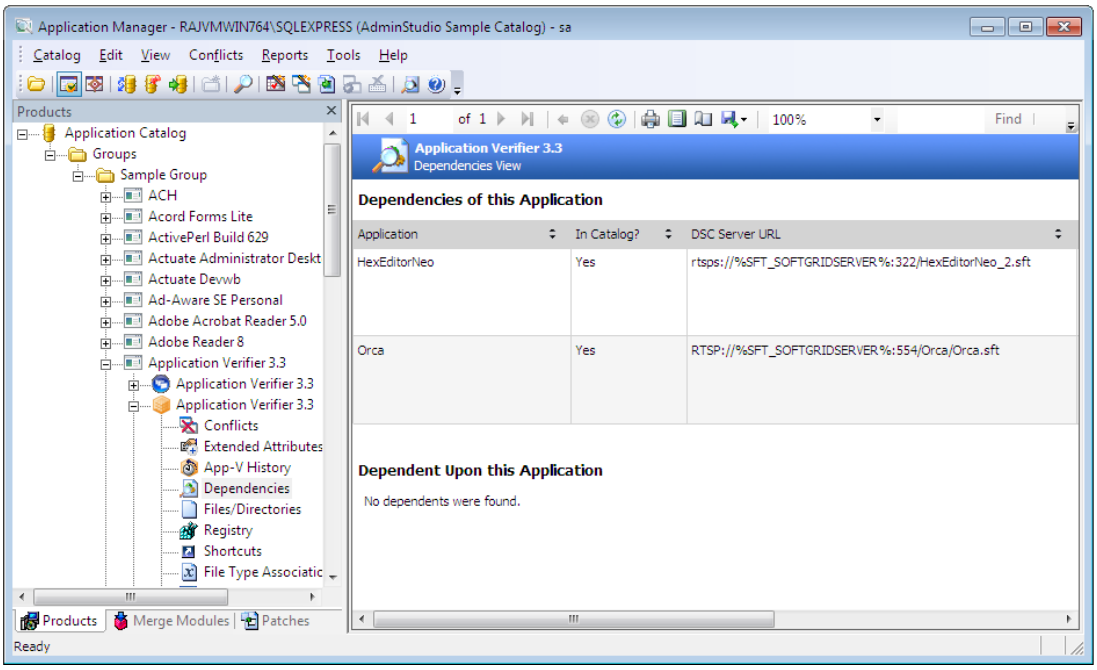


Figure 6-52: App-V Dependencies View

For each dependency, the following information is listed:

- Application
- In Catalog? (Yes / No)
- DSC Server URL
- Server URL
- Mandatory? (Yes / No)

App-V Files/Directories View

The **App-V Files/Directories View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **Files/Directories** node, lists the files and directories included in the App-V package.

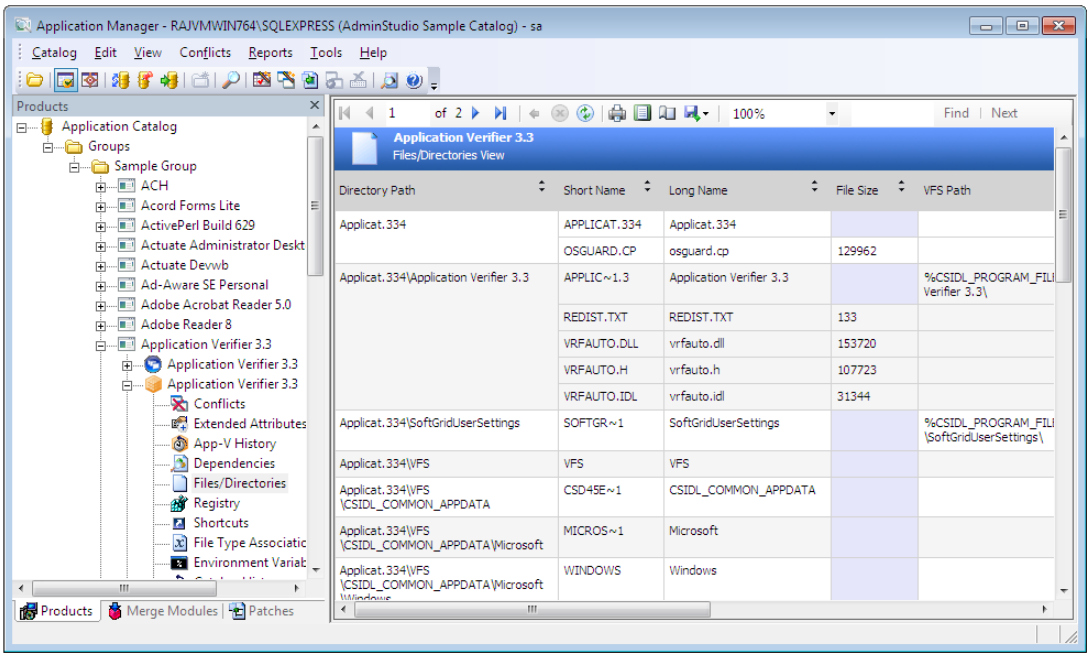


Figure 6-53: App-V Files/Directories View

For each file/directory, the following information is listed:

- Directory Path
- Short Name
- Long Name
- File Size
- VFS Path
- Feature Block 1
- App-V Version
- App-V Data Type

App-V Registry View

The **App-V Registry View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **Registry** node, lists any registry entries created or changed by the package.

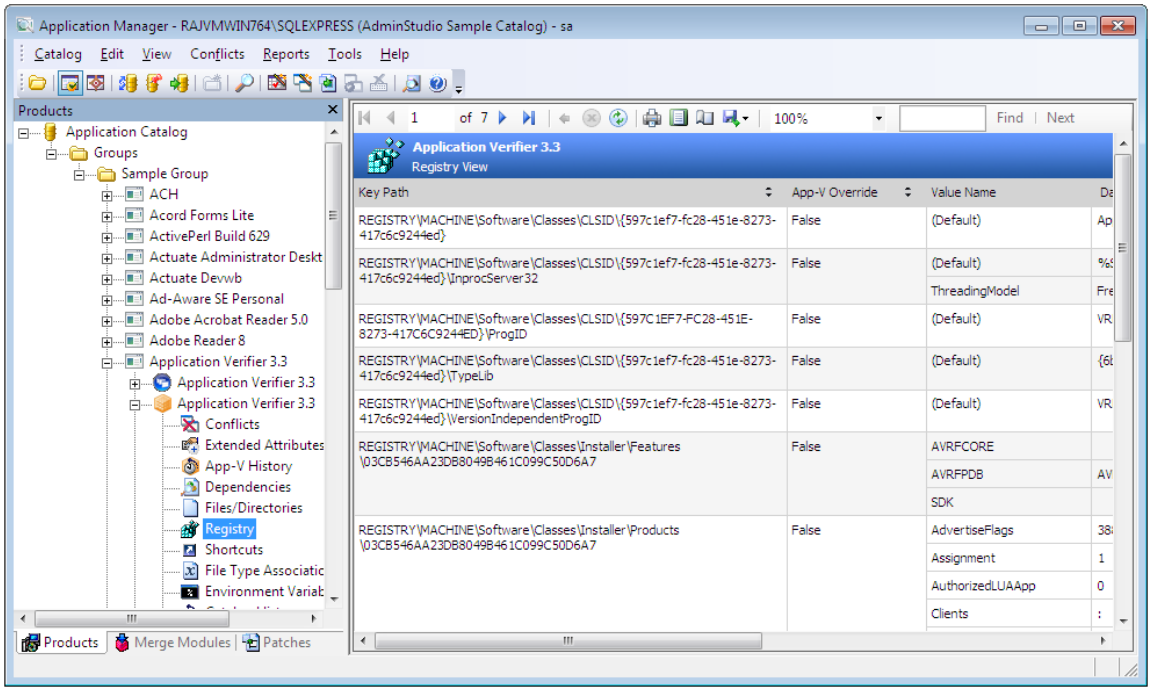


Figure 6-54: App-V Registry View

For each registry entry, the following information is listed:

- Key Path
- App-V Override
- Value Name
- Data
- Type

Note • *The information displayed on this view is the similar to the **Registry** view for Windows Installer packages. See Registry View.*

App-V Shortcuts View

The **App-V Shortcuts View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **Shortcuts** node, displays any shortcuts created by the product.

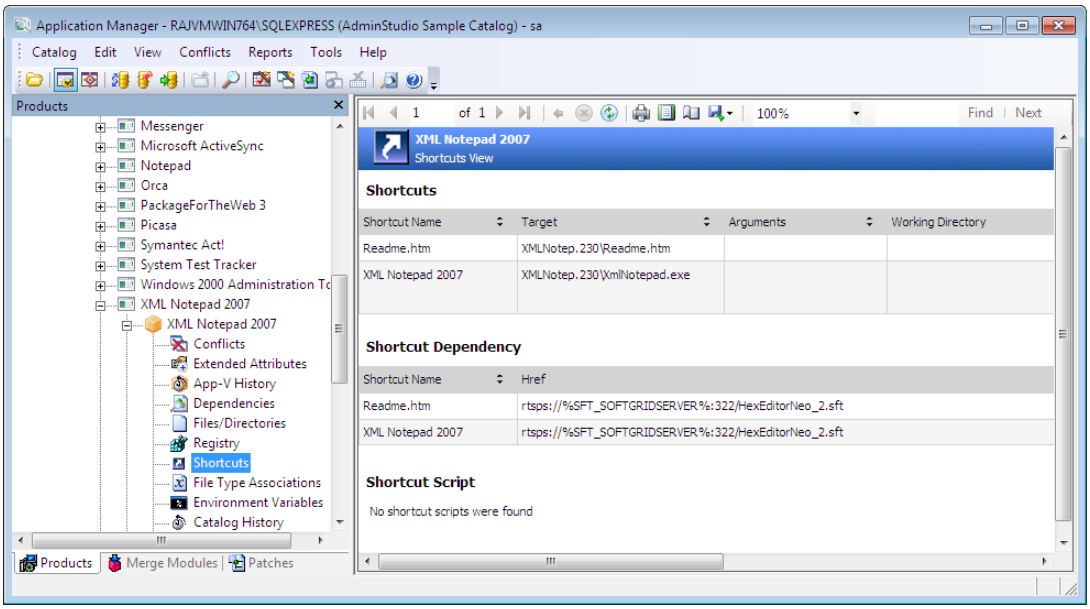


Figure 6-55: App-V Shortcuts View

Under the **Shortcuts** subheading, the following information is listed for each shortcut:

- Shortcut Name
- Target
- Arguments
- Working Directory
- Target Version
- Location

Under the **Shortcut Dependency** subheading, the following information is listed for each shortcut dependency:

- Shortcut Name
- Href
- GUID
- Is Mandatory

Under the **Shortcut Script** subheading, the following information is listed for each shortcut script:

- Shortcut Name
- Body (of script)

App-V File Type Associations View

The **App-V File Type Associations View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **File Type Associations** node, displays a list of this package's file type associations.

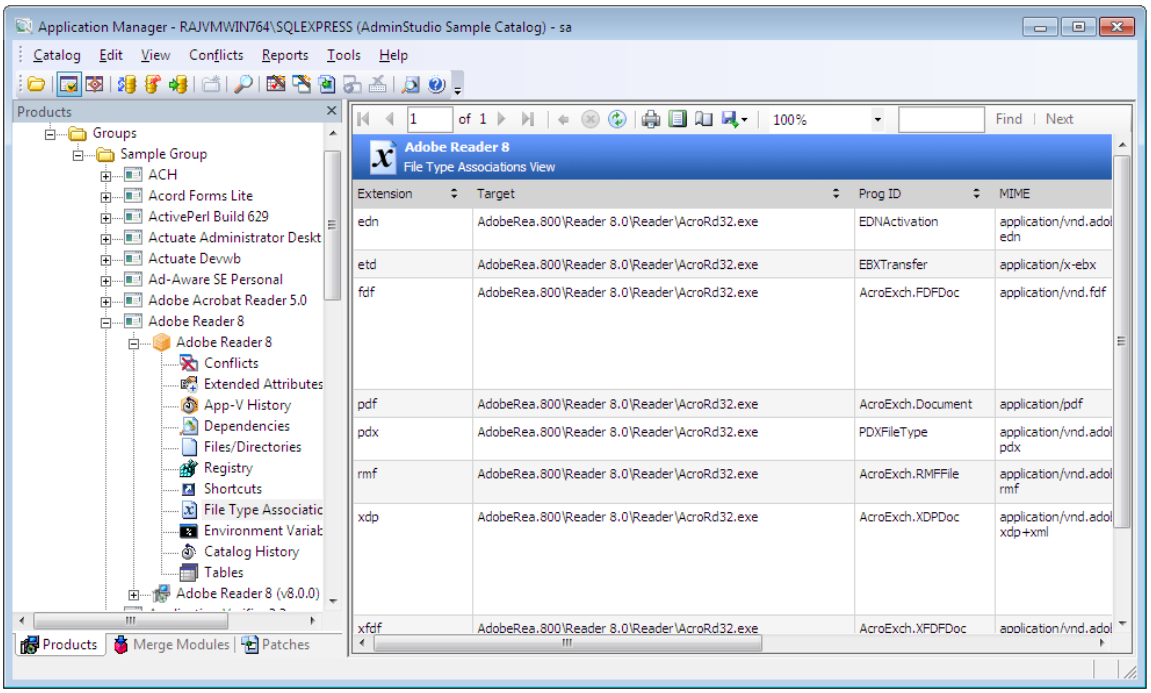


Figure 6-56: App-V File Type Associations View

For each file type association, the following information is listed:

- Extension
- Target
- Prog ID
- MIME
- Description
- Verb
- Arguments

App-V Environment Variables View

The **App-V Environment Variables View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **Environment Variables** node, lists the environment variables used in this App-V package.

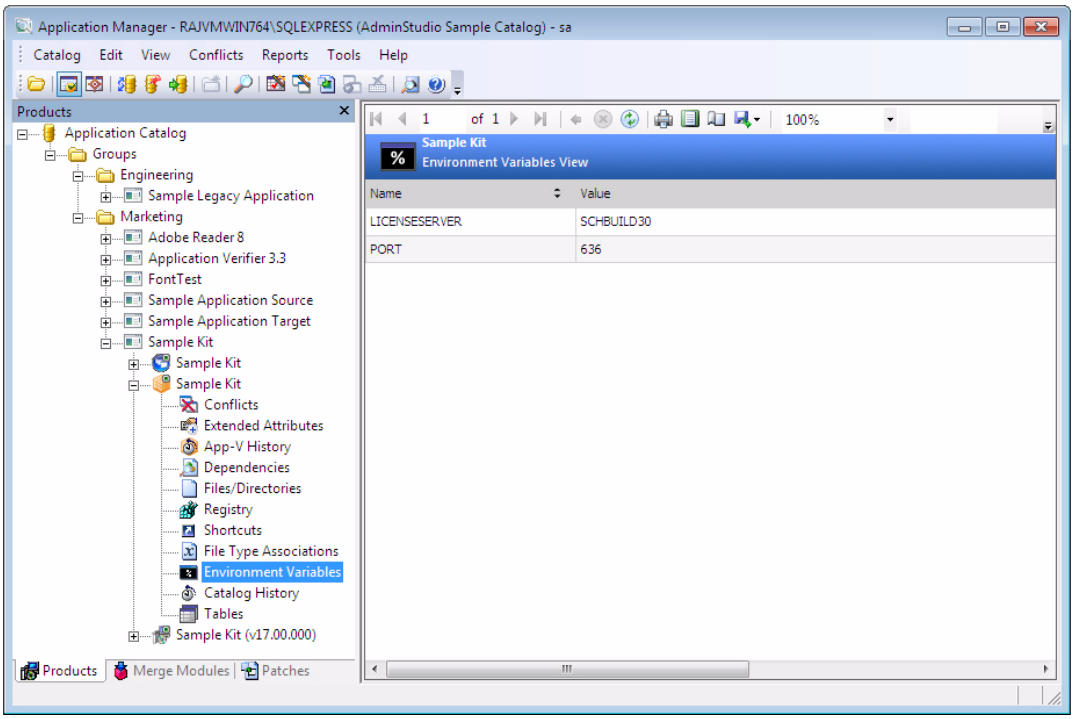


Figure 6-57: App-V Environment Variables View

For each environment variable, the following information is listed:

- Name
- Value

App-V Tables View

The **App-V Tables View**, which you open by expanding an App-V package in the Application Manager tree and selecting the **Tables** node, provides a way to view the data for this specific App-V package. Select the specific table you want to view from the **Tables** list at the top of the view.

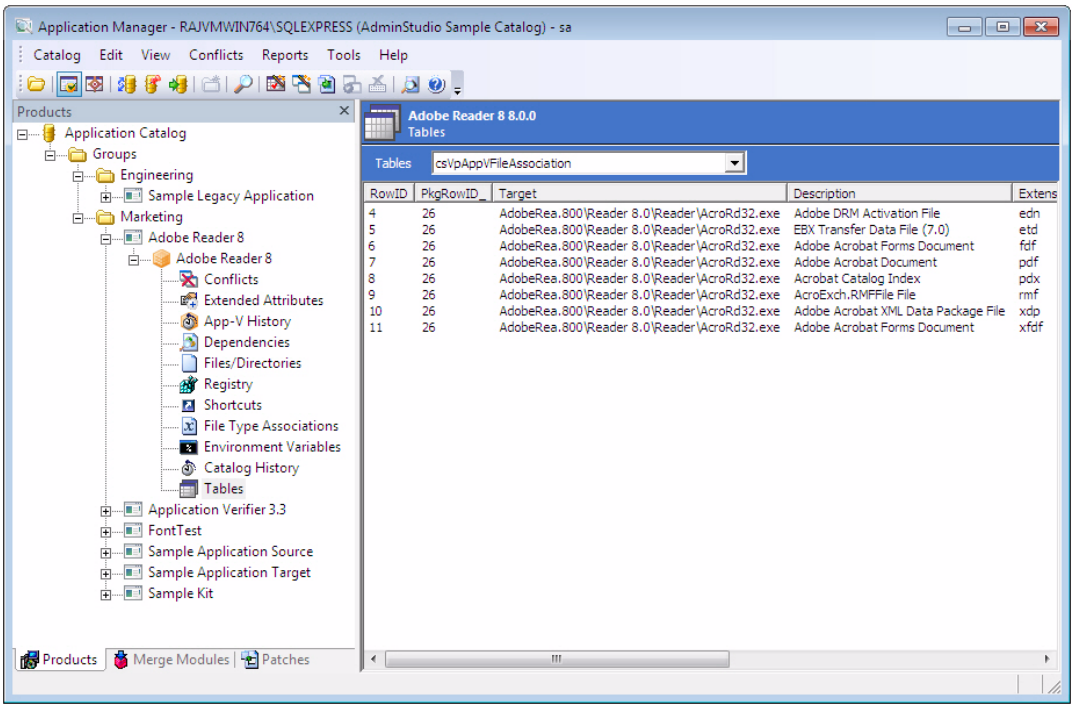


Figure 6-58: App-V Tables View

When building your own ACE rules to use for conflict identification, it is important to understand the data available for App-V packages so you can construct the necessary rule.

Citrix / ThinApp Package View

When you select a Citrix XenApp or VMware ThinApp package in the Application Catalog, the its Package view opens. The following information is displayed:

Table 6-48 • Citrix / ThinApp Package View

Option	Description
Type	Identifies the virtual package type as **Citrix** or **ThinApp**.
File	Identifies the location of this package. It can be either a hard-coded path or a UNC path. If the package is part of the Software Repository, the following statement appears: `Managed within the Software Repository`
Imported On	Date and time the virtual package was imported into the Application Catalog.
Associated MSI Packages	Lists the virtual package's source Windows Installer package. By associating a virtual package with the Windows Installer package which originated it, you have the convenience of being able to easily locate the virtual package's originating Windows Installer package, modify the original Windows Installer package, and then regenerate the virtual package. ***Note*** • *To quickly locate one of these listed Windows Installer packages in the Application Catalog, double-click on it.* In order to be listed here, the Windows Installer package has to have already been imported into the Application Catalog and must be associated with this virtual package. There are several ways to associate a virtual package with its source Windows Installer package: • Importing a Virtual Package During the Import of its Source Windows Installer Package • Importing a Virtual Package After the Import of its Source Windows Installer Package • Manually Associating a Virtual Package with a Windows Installer Package
Description	You can edit this field and add additional information about the virtual package.

If you expand the Citrix or ThinApp node in the tree, you can view its constituent views:

- **Extended Attributes**—The Extended Attributes view for a Citrix or ThinApp virtual package is identical to the Extended Attributes View shown when a Windows Installer package is selected. See Extended Attributes View in the Application Manager Product View section.
- **Catalog History**—The Catalog History View for a Citrix or ThinApp virtual package is identical to the Catalog History View shown when a Windows Installer package is selected. See Catalog History View in the Application Manager Product View section.

OS Snapshot View

When you click on an OS Snapshot in the Application Manager Product View, details about the snapshot appear in the right pane of the user interface.

The following information is displayed:

Table 6-49 • OS Snapshot View Information

Field	Description
Version	Version of the operating system of the OS Snapshot, such as Windows XP - 5.1.2600.
Language	The language the operating system was written for.
File	This can be either a hard-coded path or a UNC path.
Imported On	The date and time the OS Snapshot was imported.
Description	You can edit this with additional information about the OS Snapshot.

Click the plus sign next to the OS Snapshot icon to view these OS Snapshot constituent views:

- Extended Attributes View for OS Snapshots
- Files View for OS Snapshots
- INI File Changes View for OS Snapshots
- Registry View for OS Snapshots
- Shortcuts View for OS Snapshots

Extended Attributes View for OS Snapshots

The Extended Attributes View for an OS Snapshot is identical to the Extended Attributes View shown when a Product is selected. See Extended Attributes View in the Application Manager Product View section.

Files View for OS Snapshots

When you expand an OS Snapshot in the Application Manager Product View, you can click on Files to display a list of the files included in the OS Snapshot.

The following information is displayed for each of the files included in the OS Snapshot:

Table 6-50 • Files View Information

Column	Description
FileName	Name of the file.

Table 6-50 • Files View Information

Column	Description
csFilePath	Path
FileSize	Size of the OS Snapshot file.
Version	Version of the OS Snapshot file.
Language	Language that the OS Snapshot file was written for.
Attributes	Any attributes associated with the file.

INI File Changes View for OS Snapshots

When you expand an OS Snapshot in the Application Manager Product View, you can click on INI File Changes to display any INI file changes made by the snapshot.

The following information is displayed for each change to the INI file that is made by the snapshot:

Table 6-51 • INI File Changes View Information

Column	Description
FileName	Name of INI File that the OS Snapshot makes an entry in.
csFilePath	Path
Section	The section of the INI File where this entry is made.
Key	The Key used in the INI File entry
Value	The Value used in the INI File entry.

Registry View for OS Snapshots

When you expand an OS Snapshot in the Application Manager Product View, you can click on **Registry** to display any registry entries created or changed by the product.

The following information is displayed for each Registry Entry:

Table 6-52 • Registry View Information

Column	Description
Root	Default value of Key.
Key	Key of the Registry Entry that this component is making.

Table 6-52 • Registry View Information

Column	Description
Name	Name of the Registry Entry that this component is making.
Value	Value of the Registry Entry that this component is making.

Shortcuts View for OS Snapshots

When you expand an OS Snapshot in the Application Manager Product View, you can click on Shortcuts to display any shortcuts that were found on this OS Snapshot's operating system.

The following information is displayed:

Table 6-53 • Shortcuts View Information

Column	Description
Name	Name of shortcut.
Directory	Location of shortcut.
Target	The application executable that the shortcut points to.

Tables View for OS Snapshots

The Tables View for an OS Snapshot is identical to the Tables View shown when a Product is selected. See Tables View in the Application Manager Product View section.

Other Setup Types View

When you click on an Other Setup Types package (a package containing non-MSI based setup files) in the Application Manager Product View, details about that setup are displayed in the right pane of the user interface.

The following information is included:

Table 6-54 • Other Setup Types View Information

Field	Description
Main Directory	The original location of the non-MSI based setup files when they were imported into the Application Catalog.
Imported On	The date and time the setup was imported.

Table 6-54 • Other Setup Types View Information

Field	Description
Files	A listing of the individual files making up the setup are listed, allowing you to view or delete files by making a selection from the context menu. • To view the contents of a file, select View from the context menu. You can only view files supported by applications installed on your workstation. If you attempt to view a non-supported file, no View window will appear. If you attempt to view an executable, that executable will be launched. • To delete a file from this setup, select Delete from the context menu.

If you expand an Other Setup Type, you can view its constituent view, the Extended Attributes View for Marimba NCP Files.

Extended Attributes View for Other Setup Types

The Extended Attributes View for an Other Setup Types package is identical to the Extended Attributes View shown when a Product is selected. See Extended Attributes View in the Application Manager Product View section.

Marimba NCP Files Views

Edition • *Import support for Marimba Native Channel Packager (.ncp) files is available in AdminStudio Enterprise Edition.*

When you select a Marimba NCP file (`.ncp`) from the Application Manager Product View, details about that file are displayed in the right pane of the user interface.

The following information is included:

Table 6-55 • NCP Views Information

Column	Description
Conflict Results	Date that this package was last checked for conflicts. Click the link to jump to the Conflicts View for Marimba NCP Files.
Version	Version of the software package
File	Name and path of the NCP file that was imported
Imported On	Date when the file was imported
Description	After the NCP file is imported, you can edit this with additional information.

If you expand an NCP file, you can view its constituent views:

- Extended Attributes View for Marimba NCP Files
- Files View for Marimba NCP Files
- INI File Changes View for Marimba NCP Files
- Registry View for Marimba NCP Files
- Shortcuts View for Marimba NCP Files
- Tables View for Marimba NCP Files

Extended Attributes View for Marimba NCP Files

The Extended Attributes View for an NCP File is identical to the Extended Attributes View shown when a Product is selected. See Extended Attributes View in the Application Manager Product View section.

Files View for Marimba NCP Files

When you expand a Marimba NCP File in the Application Manager Product View, you can click on Files to display the files included in the NCP File.

The following information is displayed for each file:

Table 6-56 • Files View Information

Column	Description
FileName	Name of the file.
csFilePath	Path
FileSize	Size of the NCP file.
Version	Version of the NCP file.
Language	Language that the NCP file was written for.
Attributes	Any attributes associated with the file.

INI File Changes View for Marimba NCP Files

When you expand an Marimba NCP File in the Application Manager Product View, you can click on INI File Changes to display any INI file changes made by the NCP file.

The following information is displayed for each change that is made to the INI file by the NCP file:

Table 6-57 • INI File Changes View Information

Column	Description
FileName	Name of the file.
csFullPath	Path
Section	Section of INI file that is changed.
Key	Key that is changed.
Value	Value that is changed.

Registry View for Marimba NCP Files

When you expand a Marimba NCP File in the Application Manager Product View, you can click on Registry to display any registry entries created or changed by the NCP file.

The following information is displayed for each registry entry:

Table 6-58 • Registry View Information

Column	Description
Root	Default value of Key.
Key	Key of the Registry Entry that this NCP file is making.
Name	Name of the Registry Entry that this NCP file is making.
Value	Value of the Registry Entry that this NCP file is making.

Shortcuts View for Marimba NCP Files

When you expand a Marimba NCP file in the Application Manager Product View, you can click on Shortcuts to display any shortcuts created by the NCP file.

The following information is displayed for each shortcut created by the NCP file:

Table 6-59 • Shortcuts View Information

Field	Description
Name	Name of the shortcut.
Directory	Directory where the shortcut will exist.
Target	Directory and executable that the shortcut invokes.

Tables View for Marimba NCP Files

The Tables View for an NCP file is identical to the Tables View shown when a Product is selected. See Tables View in the Application Manager Product View section.

Patches Tab Views

The **Patches** tab includes the following views which provide content and impact information about Microsoft patches that have been imported into the Application Catalog.

Table 6-60 • Patches Tab Views

View	Description
Patches Group View	Opens when the root group in the **Patches** tab is selected, and lists all of the groups that have been created in the **Patches** tab.
New Patches Group View	All new patches are imported into the **New Patches** group, and this view lists all of the patches in that group.
Group View of a Selected Group	Opens when a group other than the root group in the **Patches** tab is selected. For each selected group, a list of all of the patches in that group is displayed.
Patch View	Lists general content information on a selected patch.

Patches Group View

The Patches Group View opens when the root group in the **Patches** tab is selected. The Patches Group View lists all groups that have been created in the Patches tab.

All new patches are imported into the **New Patches** group, and then you can organize the patches into other groups according to your business needs. See Organizing Your Application Catalog Using Groups.

The **New Patches** group is automatically created during installation. While it can be renamed, it cannot be deleted.

Context Menu Options

When the root group in the Patches Tab is selected, the following items are available on the context menu:

- **Refresh**—Refresh the patch listing to reflect the most recent modifications.
- **Import Patches**—Opens the OS Security Patch Wizard so that you can import patches into the Application Catalog.
- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Properties**—Open the **Group Properties** dialog box.

New Patches Group View

The New Patches Group View opens when the **New Patches** group on the Application Manager or ConflictSolver **Patches** tab is selected. All new patches are imported into the **New Patches** group. You can then organize the patches into other groups according to your business needs. See Organizing Your Application Catalog Using Groups.

The **New Patches** group is automatically created during installation. While it can be renamed, it cannot be deleted.

The New Patches Group View displays a list of all of the patches in that group, including the following information:

Table 6-61 • New Patches Group View Information

Option	Description
Name	Name of patch file.
Description	Description of the patch file.
Release Date	Date the patch was released by Microsoft.
Import Date	Date the patch was imported into the Application Catalog.

If you select a patch in this list, detailed patch properties are displayed in the area to the left of the list, including the **Title**, **Summary**, and **Release Date** of the patch.

Context Menu Options

When the **New Patches** group in the **Patches Tab** is selected, the following items are available on the context menu:

- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Properties**—Open the **Group Properties** dialog box.

Group View of a Selected Group

The Group View of a selected group opens when a group other than the root group or the **New Patches** group in the **Patches** tab is selected. For each selected group, a list of all of the patches in that group is displayed, including the following information:

Table 6-62 • Group View of a Selected Group Information

Option	Description
Name	Name of patch file.
Description	Description of the patch file.
Release Date	Date the patch was released by Microsoft.
Import Date	Date the patch was imported into the Application Catalog.

If you select a patch in this list, detailed patch properties are displayed in the area to the left of the list, including the **Title**, **Summary**, and **Release Date** of the patch.

Context Menu Options

When a group other than the root group or the **New Patches** group in the **Patches Tab** is selected, the following items are available on the context menu:

- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Delete**—Delete the selected group.
- **Cut**—Copy the selected group to the clipboard so that you can remove it from its current location and paste it into another location.
- **Paste**—Insert the patch or the group on the clipboard into the selected location.
- **Properties**—Open the **Group Properties** dialog box.

Patch View

The **Patch View**, which is displayed when a patch is selected on the **Patches** tab, lists general information on the selected patch. The following information is included:

Table 6-63 • Patch View Information

Option	Description
ID	Microsoft Security Bulletin ID. Click this link to view this bulletin on the Microsoft Web site.
Title	Title of patch.

Table 6-63 • Patch View Information

Option	Description
Release Date	Date the patch was released by Microsoft.
KB Article	Microsoft Knowledge Base article ID. Click this link to view this article on the Microsoft Web site.
Imported On	Date patch was imported into the Application Catalog
Groups	List of all of the groups that this patch is included in.
Description	You can enter a description of the patch in this field.

In the Application Manager, you can view additional detailed patch information by selecting a patch and then selecting **Properties** from the context menu.

Context Menu Options

When a patch in the **Patches Tab** is selected, the following items are available on the context menu:

- **Rename**—Rename the selected patch.
- **Delete**—Delete the selected patch.
- **Cut**—Copy the selected patch to the clipboard so that you can remove it from its current location and paste it into another location.
- **Copy**—Copy the selected patch to the clipboard so that you can paste a copy of it into another location.
- **Paste**—Insert the patch on the clipboard into the selected location.
- **Generate Report**—Generate a Patch Impact Analysis Report for that patch. See Generating the Patch Report.
- **Properties**—Open the **Patch Properties** dialog box for that patch.

Merge Module Views

From the Merge Modules tab, you can access the following views:

- **All Merge Modules View**—Select this option to view a list of all of the Merge Modules in your catalog. See All Merge Modules View.
- **Merge Module View**—Select an individual Merge Module to see detailed information on that module. See Merge Module View.

All Merge Modules View

The All Merge Modules view is the root node of the merge modules explorer. It contains a list of all merge modules in the Application Catalog, including titles, versions, languages, and identifiers for each module.

Double-click on a merge module to see information about it in the Merge Module View.

Merge Module View

When you select a merge module in the merge modules explorer, details are displayed in this view.

These details include:

Table 6-64 • Merge Module View Information

Field	Description
Version	The version of the Merge Module.
Language	The language that the Merge Module was written for.
Identifier	String that uniquely identifies the Merge Module.
File	The path and file name of the Merge Module file that was imported.
Imported On	The date and time the Merge Module was imported.

Click the plus sign next to the Merge Module to view these Merge Module constituent views:

- Merge Module / Components View
- Merge Module / Dependency View
- Merge Module / Exclusion View
- Merge Module / Files View
- Merge Module / Products View

Merge Module / Components View

When you expand a merge module in the merge modules explorer, you can click on Components to display any components created or changed by the merge module.

The following information is displayed for each component included in this merge module:

- Component
- ComponentId
- Directory_
- csFullPath

Merge Module / Dependency View

When you expand a merge module in the merge modules explorer, you can click on Dependency to display any dependencies in the merge module.

The following information is displayed for each dependency included in this merge module:

- ModuleLanguage
- RequiredID
- RequiredLanguage
- RequiredVersion

Merge Module / Exclusion View

When you expand a merge module in the merge modules explorer, you can click on Exclusion to display any exclusions in the merge module.

The following information is displayed for each exclusion included in this merge module:

- ModuleLanguage
- ExcludedID
- ExcludedLanguage
- ExcludedMaxVersion
- ExcludedMinVersion

Merge Module / Files View

When you expand a merge module in the merge modules explorer, you can click on Files to display any files in the merge module.

The following information is displayed:

Table 6-65 • Merge Module Files View Information

Column	Description
Component_	Name of component that this Merge Module file is associated with.
File	Name of this Merge Module file.
FileName	File name of this Merge Module file.
FileSize	Size of this Merge Module file.
Version	Version of this Merge Module file.

Merge Module / Products View

When you expand a merge module in the merge modules explorer, you can click on Products to display any products in the Application Catalog that use the merge module.

The following information is displayed:

Table 6-66 • Merge Module Products View Information

Column	Description
ProductName	Name of product associated with this Merge Module.
ProductVersion	Version of product associated with this Merge Module.
Manufacturer	Manufacturer of the product associated with this Merge Module.

Dialog Boxes

Application Manager includes the following dialog boxes:

- Application Catalog Properties Dialog Box
- Associate with Workflow Manager Application Dialog Box
- Command-Line Parameters Dialog Box
- Command-Line Parameters Dialog Box

- Connect Application Catalog Dialog Box
- Extended Attribute Property Dialog Box
- Find Dialog Box
- Login Required Dialog Box
- Group Properties Dialog Box
- Package Auto Import Dialog Box
- Publication Manager Dialog Box
- Publication Processing Dialog Box
- Publication Properties Dialog Box
- Select AdminStudio Enterprise Server URL Dialog Box
- Subscription Manager Dialog Box
- Task Scheduling Dialog Box

Application Catalog Properties Dialog Box

The Application Catalog Properties Dialog box, which is opened by selecting **Properties** on the **Catalog** menu, is used to enable the Software Repository, select its location, and create a Proxy Account that AdminStudio can use to make modifications to the Software Repository.

Software Repository Tab

A Windows Installer package is made up of many files that are executed when the setup is run. You only import the `.msi` file into the Application Catalog, not all of the files necessary for installation. With the Software Repository, when you import an installation package into the Application Catalog, all of the files associated with that package are copied into the Software Repository location, a directory that you specify. This allows you to manage those files, preventing them from getting modified or lost.

On the **Software Repository tab**, you can choose to **Enable the Software Repository** for the new Application Catalog, and specify a **Proxy Account** for AdminStudio to use to make modifications to the directory path selected as the **Software Repository Location**.

This tab includes the following options:

Table 6-67 • Application Catalog Properties / Software Repository Tab Options

Option	Description
Enable Software Repository	Select this option to enable the Software Repository feature for this Application Catalog.
Software Repository Location	Specify the directory location of the Software Repository for this Application Catalog.

Table 6-67 • Application Catalog Properties / Software Repository Tab Options (cont.)

Option	Description
Proxy Account	You need to specify a **Proxy Account** for AdminStudio to use to access the Application Catalog on the database server. You cannot use Windows Authentication for this Proxy Account. Specify a **Login ID** and **Password** for AdminStudio to use to log on to the database server that contains this Application Catalog when modifying the Software Repository Location directory.

Associate with Workflow Manager Application Dialog Box

Note • *AdminStudio Workflow Manager is a Web-based application management system that has integrated functionality with AdminStudio.*

The Associate with Workflow Manager Application dialog box is displayed when you select the corresponding command from the context menu when right-clicking on a product. It allows you to associate extended attribute data for a product with an application already in an AdminStudio Workflow Manager application catalog. You can select the available packages from the list in the dialog box.

Note • *This dialog (and its corresponding command) are only available if you select the **Integrate with Workflow Manager** option on the **Extended Attributes Tab** of the ConflictSolver and Application Manager **Options Dialog Box**.*

Note • *Be sure to select a Workflow Manager application that uses a template containing at least one major data group that was specified with the group's extended attribute description file, as described in the AdminStudio Workflow Manager user documentation.*

Change Enterprise Server Password Dialog Box

This dialog box, which is opened by selecting **Change AES Password** on the **Catalog** menu, allows you to change your password to connect to the AdminStudio Enterprise Server Application Catalog (which is the same password you use to log in to the AdminStudio Enterprise Server). The **Change AES Password** selection is enabled when you are connected to the AdminStudio Enterprise Server Application Catalog.

Note • *If you are not connected to the AdminStudio Enterprise Server Application Catalog, the* ***Change AES Password*** *selection on the* ***Catalog*** *menu is disabled.*

Table 6-68 • Change Enterprise Server Password Dialog Box

Option	Description
User Name	(Read Only) User name of user who is connected to the AdminStudio Enterprise Server Application Catalog.
Old Password	Enter the existing password for current user.
New Password	Enter the new password.
Confirm New Password	Enter the new password a second time.

Command-Line Parameters Dialog Box

This dialog box is displayed when running Application Manager from the command line using the -? parameter. Information in this dialog box is covered in the Application Manager Command-Line Functionality topic.

Connect Application Catalog Dialog Box

The Connect Application Catalog dialog box opens when you choose to open an existing Application Catalog. This dialog box has three tabs:

- **Enterprise Server**—Select this tab to open the AdminStudio Enterprise Server Application Catalog database. See Enterprise Server Tab.
- **Standalone**—Select this tab to open an Application Catalog database other than the AdminStudio Enterprise Server Application Catalog. See Standalone Tab.
- **Recent**—Provides a list of recently opened Application Catalogs. When you select an Application Catalog and click **OK**, either the Application Catalog opens or you are prompted for login information (if you need authentication to the Application Catalog). See Recent Tab.

Making this the Default Shared Application Catalog

If you select the **Make this the default shared Application Catalog** option, the Application Catalog you are opening will become the default Application Catalog (and be recorded as such in the **AdminStudio Shared** directory).

If the Application Catalog is made the default, all other AdminStudio users that use the same shared directory will automatically connect to the default Application Catalog when AdminStudio is launched. Therefore, you should only set this option if you want to affect all AdminStudio users who access that shared directory.

Note • *In the AdminStudio Enterprise Edition, only the AdminStudio Administrator or users with the Change Default Database permission will see the* ***Make this the default shared Application Catalog*** *option. This allows the AdminStudio Administrator to configure the default Application Catalog, and then subsequent installations of AdminStudio will automatically connect to the default Application Catalog if they use the same shared directory.*

Enterprise Server Tab

To connect to the AdminStudio Enterprise Server Application Catalog, you log in on the **Enterprise Server** tab of the **Connect Application Catalog** dialog box.

Table 6-69 • Connect Application Catalog / Enterprise Server Tab Options

Option	Description
URL	Location of the AdminStudio Enterprise Server associated with this installation of AdminStudio. If the AdminStudio Enterprise Server has not yet been configured with the AdminStudio client tools (such as when it is set to its default value of `http://localhost`), click the URL link to open the **Select AdminStudio Enterprise Server URL** dialog box, and enter the URL for location of the AdminStudio Enterprise Server associated with this installation of AdminStudio.
Authentication	Select one of the following options: • Windows Authentication • AdminStudio Enterprise Server User ***Note*** • *When using* ***AdminStudio Enterprise Server User*** *authentication, if Anonymous authentication is turned off in IIS, both the user's machine and the AdminStudio Enterprise Server need to be on the same domain in order for login to succeed.*
User Name and Password	If you selected **AdminStudio Enterprise Server User**, enter your AdminStudio Enterprise Server `User Name` and **Password** (provided by your System Administrator).

Login Troubleshooting

If you are using a Web Portal with custom security zone settings, your AdminStudio Enterprise Server URL is using an IP address, and you receive Error 0x800A1518 when you attempt to login, change the AdminStudio Enterprise Server URL to the NetBios equivalent and then try again. For example, if you are connecting to `http://120.12.1.15`, the NetBios equivalent would be `http://wfmportal`.

Standalone Tab

On the **Standalone** tab of the **Connect Application Catalog** dialog box and the **Specify Database Information** panel of the Application Catalog Wizard, enter the information required to login to the specified Application Catalog or enter the name of the Application Catalog that you are creating.

Table 6-70 • Connect Application Catalog / Standalone Tab Options

Option	Description
Server	Select one of the available SQL Servers on the network from this list. You can also manually enter the name of the SQL Server to which you want to connect.
Authentication	Select one of the following options: • **Windows Authentication**—Choose to use Windows network authentication (your network login ID) to log into this Application Catalog. • **Server Authentication**—Choose to use SQL Server login identification for authentication. • **Login ID and Password**—If you chose Server Authentication, enter the appropriate Login ID and Password.
Catalog	Do one of the following: • If you are connecting to an **existing** Application Catalog, select the catalog from those available on the Server. • If you are creating a **new** Application Catalog, enter a name for this new catalog.
Test	Click this button to test whether a connection can be made to the database.
Make this the default shared Application Catalog Connect Application Catalog Dialog Box Only	When this option is selected, the Application Catalog you are trying to open or create will become the default Application Catalog (and be recorded as such in the **AdminStudio Shared** directory). If the Application Catalog is made the default, all other AdminStudio users that use the same shared directory will automatically connect to the default Application Catalog when AdminStudio is launched. Therefore, you should only set this option if you want to affect all AdminStudio users who access that shared directory.

Table 6-70 • Connect Application Catalog / Standalone Tab Options (cont.)

Option	Description
Create database as a subscriber Create Application Catalog Wizard Only	Select this check box to designate this new Application Catalog as a Subscriber database in Application Catalog Replication (a database that receives replicated data). When this option is not selected, or when an existing database is upgraded, the Application Catalog is defined as a Publisher database in Application Catalog Replication (a database that makes data available for replication). The AdminStudio interface selectively enables/disables the Replication menu items on the Catalog menu based upon the type of Application Catalog that is currently opened. ***Edition*** • *Application Catalog Replication is included with the AdminStudio Enterprise Edition. If you have AdminStudio Standard or Professional Editions, this check box is disabled. Contact your AdminStudio Sales Representative for more information.* ***Note*** • *This check box only appears when creating a new Application Catalog, not when logging in to an existing Application Catalog.*

Recent Tab

The **Recent** tab displays a list of all Application Catalogs that have recently been open. To login to one of these Application Catalogs, select it and click OK.

- If you are opening a standalone SQL Server Application Catalog, you will be prompted for login information.
- If you are opening the AdminStudio Enterprise Server Application Catalog, you are prompted for AdminStudio Enterprise Server login information before the Application Catalog will open.

Default Application Catalog Dialog Box

When you initially open AdminStudio, because a default Application Catalog has not yet been set, the **Default Application Catalog** dialog box opens, prompting you to open an Application Catalog.

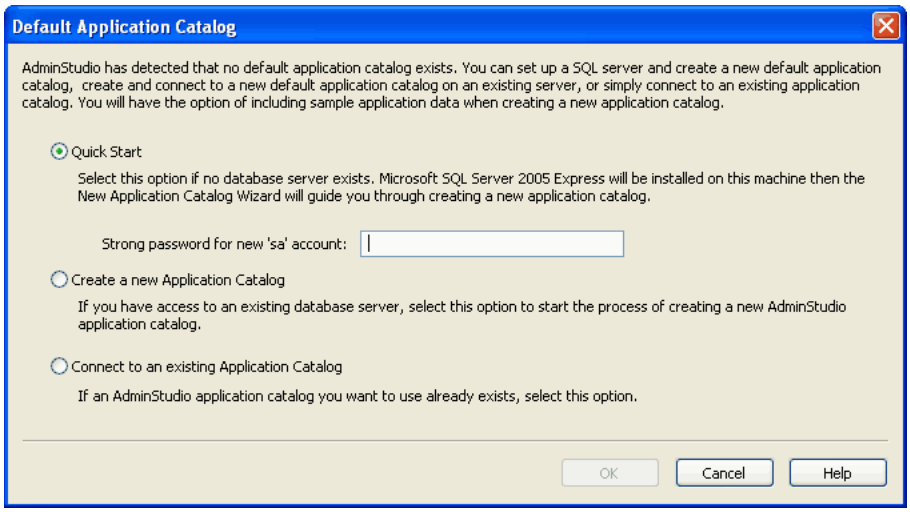


Figure 6-59: Default Application Catalog Dialog Box

You can select any of the following options:

Table 6-71 • Default Application Catalog Dialog Box Options

Option	Description
Quick Start	Select this option if you do not have access to a database server. Microsoft SQL Server 2005 Express will be installed on your machine first, then you will have the option to create a new Application Catalog database on that server that is populated with sample data. When Microsoft SQL Server 2005 Express is installed, a user named sa is automatically created. Therefore, you need to enter a password for this account in the **Strong password for new 'sa' account** box. A strong password: • **must be** at least 7 characters long, • **must be** a combination of letters, numbers, and symbol characters, • **must not be** a dictionary word, a command name, a person's name, or a system user name • **must not contain** any of the following characters: `[ ] { } ( ) , ; ? * ! @`

Table 6-71 • Default Application Catalog Dialog Box Options

Option	Description
Create a new Application Catalog	Select this option to create a new, empty Application Catalog on an existing SQL Server database server that you have access to. You will have the option of populating this new Application Catalog with sample data.
Connect to an existing Application Catalog	Select this option to connect to an existing Application Catalog on an SQL Server database server.

Extended Attribute Property Dialog Box

If you use extended attributes, and you click on a text extended attribute label, this dialog box opens.

Within it, you can provide the value for the extended attribute. When you click OK, the value is automatically displayed in the Extended Attributes View next to the corresponding label.

Find Dialog Box

You can use the **Find** dialog box, which is accessed by selecting **Find** from the **Edit** menu, to search for data in Application Catalog tables.

Note • *This search is limited to string type columns.*

The tables that are searched depend upon what is selected when the **Find** dialog box is opened:

Table 6-72 • Search Options

If you select...	and specify these options...	this will be searched
Group	**All Tables and All Columns**	All tables and all columns in all of the Packages in the selected Group
Package	**All Tables and All Columns**	All tables and all columns in the selected Package
Package	**A Table and All Columns**	All columns of a specific table in the selected Package
Package	**A Table and a Column**	A specific column in a specific table in the selected Package

Also, if you want to search for a partial match rather than an exact match, you can use the Partial Match option on the Find dialog box.

The Find dialog box can be accessed in four ways:

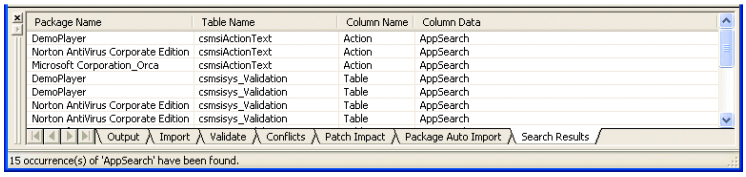
- Select **Find** on the **Edit** menu.
- Press **Ctrl** + **F**.
- Click the **Find** button ( )on the toolbar.
- Select the node in the tree (Group, MSI Package, OS Snapshot, Other Setup, etc.) that you want to search, and select **Find** from the context menu.

The **Find** dialog box contains the following fields and buttons:

Table 6-73 • Find Dialog Box Properties

Properties	Description
Find What	Enter the text that you want to search for.
Look In Table	• If a package was selected when the **Find** dialog box was opened, all of the tables in that package are listed. Select the table that you would like to search, or select **<All Tables>**. When you select a table from this list, the **Look In Columns** list is populated with all of the columns in that table. • If a group is selected when you opened the **Find** dialog box, **<All Tables>** is the only option listed.
Look In Columns	• If you selected a table from the **Look in Table** list, all of the columns in that table are listed. Select the column that you would like to search, or select **<All Columns>**. • If a group was selected when you opened the **Find** dialog box, **<All Columns>** is the only option listed.
Partial Match	• If this option is not selected, Application Manager will search for an exact match of the text you entered in the **Find What** text box. The search will be case sensitive. • If this option is selected, then Application Manager will use appropriate wild card characters so that a partial data match is performed. The search will be case insensitive.

Table 6-73 • Find Dialog Box Properties

Properties	Description
Find Button	Click to initiate the search. The **Find** dialog box will close, and the data that is found is displayed in the **Search Results** tab of the Output Window, in the following format:  If you double click on this data, Application Manager will navigate to the appropriate record in the Tables View, and that record will be highlighted in red.

Login Required Dialog Box

If you chose to open an existing Application Catalog that was listed on the **Recent** tab of the Connect Application Catalog Dialog Box, and you are not currently logged in to that database, this dialog box opens prompting you to log in.

Group Properties Dialog Box

The Group Properties dialog box opens when you right-click a group in the Groups tree and select Properties from the context menu.

You can provide both a name and description for the group, as well as any comments. Information provided in this dialog box is displayed in the Properties area of the Group view (displayed whenever a group is selected in the main Application Manager Product View).

Table 6-74 • Group Properties Dialog Box Properties

Option	Description
Name	Provide a name for the group. This name cannot exceed 40 characters.
Description	Enter any descriptive information about the group. This cannot exceed 80 characters.
Comments	Provide any comments about the group. Comments cannot exceed 255 characters in length.

Package Auto Import Dialog Box

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

The Package Auto Import dialog box lists the Application Catalogs and Network Directories that are being monitored.

The Package Auto Import dialog box includes the following options:

Table 6-75 • Package Auto Import Dialog Box Options

Option	Description
Monitored Activities	List of all Remote Application Catalogs and Network Directories that are being monitored for new or updated packages. Only those items linked to the currently open Application Catalog will be listed. ***Note*** • *When you initially set up a connection to a Remote Application Catalog, until the linked packages have been imported into the local Application Catalog for the first time, the Start button is disabled when that link is selected on the Package Auto Import dialog box. The first automatic import of linked packages in a Remote Application Catalog is initiated when you click the Close button to exit the Package Auto Import dialog box.*
Occurrence	Lists how often the Application Catalog or Network Directory is going to be checked for updated or new packages. This corresponds to the current selection made on the Options tab of the Package Auto Import Properties dialog box for that item.
Update On	Lists the last time the Remote Application Catalog or Network Directory was checked for new or updated packages.
Add	Click to access the Package Auto Import Wizard, where you can set up a connection to a Remote Application Catalog or Network Directory.
Remove	Click to remove the selected Remote Application Catalog or Network Directory connection. The packages that have already been imported into your Application Catalog will remain, but they will no longer be automatically updated.
Properties	Click to access the Package Auto Import Properties (Remote Application Catalog) Dialog Box or the Package Auto Import Properties (Network Directory) Dialog Box, where you can change the update schedule interval and other options for the selected link. This button is disabled for Remote Application Catalog entries until the linked packages have been imported into the Application Catalog for the first time.

Table 6-75 • Package Auto Import Dialog Box Options (cont.)

Option	Description
Start	Click to force an update of the selected link. Package Auto Import checks the monitored Remote Application Catalog or Network Directory to see if there were any updated or new packages, and then initiates an auto-import of those packages. This button is disabled for Remote Application Catalog entries until the linked packages have been imported into the Application Catalog for the first time.
Close	Click to exit this dialog box. When you initially set up a connection to a Remote Application Catalog or Network Directory, automatic import will be initiated when you click the Close button.

Note • *If you first use the Package Auto Import Wizard to set up a connection to a package, and then use the Package Auto Import Wizard to set up a connection to another package in the same Application Catalog, those connections will be listed as two separate entries on the Package Auto Import dialog box. However, after the packages have been imported into the Application Catalog, only one entry will appear for that Application Catalog.*

Package Auto Import Properties (Remote Application Catalog) Dialog Box

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

The Package Auto Import Properties (Remote Application Catalog) dialog box is accessed by selecting an Application Catalog on the Package Auto Import Dialog Box and clicking Properties.

Note • *See also the Package Auto Import Properties (Network Directory) Dialog Box.*

Groups Tab

The Groups tab displays the packages within a Remote Application Catalog that you can link to. Select or deselect the check boxes to link/unlink to the packages. Any package that is already linked to will display the check box as selected.

Options Tab

The settings on the Options tab directly affect the way linked packages will be refreshed (update the contents of packages on your local Application Catalog with packages on Remote Application Catalogs).

Table 6-76 • Options Tab Properties

Properties	Description
Update package when its source is modified	**If you have selected this option:** When the scheduled automatic import occurs or when you select Start on the Package Auto Import dialog box, Application Manager will go and check if any of the linked packages have been modified in the Application Catalog. • **If the linked package has been modified**, then the local Application Catalog database is updated to reflect the changes. • **If the linked package has not been modified**, you receive a message stating that no refresh is necessary. **If you have not selected this option:** When the scheduled automatic import occurs or when you select Start on the Package Auto Import dialog box, Application Manager will still check to see if any of the linked packages have been modified: • **If the linked package has been modified**, Application Manager will inform you that the source package has been modified but that it will not update the local Application Catalog database because this option is not selected. • **If the linked package has not been modified**, you receive a message stating that no refresh is necessary.
Delete package when the source package is deleted	**If you have selected this option**, when the scheduled automatic import occurs or when you select Start on the Package Auto Import dialog box, Application Manager will delete any linked data in the local database if the data is deleted from the remote database. **If you have not selected this option**, when the scheduled automatic import occurs or when you select Start on the Package Auto Import dialog box, even if the source package is deleted in the remote database, your copy of the package will not be deleted.
Schedule	Select this option to enable automatic import of the linked packages at scheduled intervals. To disable automatic import, leave this option unselected. When this option is selected, the scheduling fields are enabled.
Run the Process	From this list, select Daily or Weekly to specify how often you want automatic import to be performed.

Table 6-76 • Options Tab Properties (cont.)

Properties	Description
At	Select the time of day that you want the automatic import to occur by selecting the hour, minutes, seconds, or AM/PM setting and then clicking the up and down arrows to adjust the setting.
Starting on	Click on the arrow to open the Calendar and then select the first day that you want automatic importing to begin.

Package Auto Import Properties (Network Directory) Dialog Box

Note • *To use the Package Auto Import feature, you must have AdminStudio Administrator permission. For all other users, the Package Auto Import options will be disabled.*

The Package Auto Import Properties (Network Directory) dialog box is accessed by selecting a network directory on the Package Auto Import Dialog Box and clicking Properties. On this dialog you can choose to monitor sub-directories of the selected directory, and can schedule the monitor interval for this directory.

Note • *See also the Package Auto Import Properties (Remote Application Catalog) Dialog Box .*

This dialog box contains the following options:

Table 6-77 • Package Auto Import Properties (Remote Catalog)

Option	Description
Folder	Displays a read-only view of the network directory that is being monitored.
Monitor sub-directories also	Select this option if you want to check sub-directories of the selected directory for Windows Installer and App-V packages to import.
Schedule	Select this option to turn on automatic monitoring of the selected directory. If you do not select this option, the network directory will only be monitored when Start is clicked on the Package Auto Import dialog box. When the Schedule option is selected, the following fields are activated, where you can specify the monitor interval for this directory. • **Run the process**—Select whether you want to run the monitoring process Daily or Weekly. • **At**—Specify the time that you want to run the monitoring process. • **Starting on**—Select the initial day that you want to begin the monitoring process.

Publication Manager Dialog Box

The Publication Manager lists all Publications in the SQL Server database you are currently connected to. Use this dialog box to create, edit, publish, or delete a Publication.

Note • *The **Publication Manager** option under **Replication** on the AdminStudio **Catalog** menu is only enabled if the current user is connected to an SQL Server database, has been assigned* `sysadmin` *privileges in SQL Enterprise Manager on that database server, and the database was identified when it was created as being intended for use as a **Publisher**. See Application Catalog Wizard for information on how to identify databases as Subscribers or Publishers.*

Table 6-78 • Publication Manager Dialog Box Properties

Properties	Description
Name	Name of the Publication.
Description	Description of the Publication.
Status	The status of the last attempted update of the Publication: Succeeded or Failed. If the update is currently being executed, Executing appears in this column.
Last Action	Text describing the last action that was performed on this Publication.
Last Run	The date and time that the Publication was last updated.
Next Run	The next scheduled update time of that Publication.
New	Click to create a new Publication using the Publication Wizard.
Edit	Click to access the **Publication Properties** dialog box, where you can edit the selected Publication's name, description, scheduled update day and time, the type of data you want to include in the Publication, the list of users allowed to access the Publication, and the packages included in the Publication.
Run	Click to manually update the selected Publication. The Publication data is replicated from your Application Catalog to the Publisher Database Server, making it available for Subscription.
Delete	Click to delete the selected Publication from the Publication Manager.

Caution • *If an attempt to create a Publication fails, it could be because your Publication server does not have a Distributor server configured. For more information, see Initial Configuration Checklist.*

Publication Processing Dialog Box

The Publication Processing dialog box opens after you click Finish on the **Publication Summary** panel of the Publication Wizard.

The Publication Processing dialog box tracks the progress of the Publication Wizard as the Publication is created, the snapshot agent is created, the permissions are added to the Publication Access List, and articles are added to the Publication.

When processing is finished, the new Publication is listed on the **Publication Manager** dialog box.

Publication Properties Dialog Box

The Publication Properties dialog box, which is accessed by clicking Edit on the **Publication Manager** dialog box, allows you to edit the selected Publication's name, description, scheduled update day and time, the type of data you want to include in the Publication, the list of users allowed to access the Publication, and the packages included in the Publication.

The Publication Properties dialog box contains the following tabs:

Table 6-79 • Publication Properties Dialog Box Tabs

Tab	Properties
Details Tab	This tab contains the following properties: • **Name**—Name that identifies this Publication. This field is read-only. • **Description**—Enter a description of the purpose or intended subscribers to this Publication. This description will be listed on the Publication Manager. • **Schedule**—In this field, you specify how often you would like this Publication to be automatically updated. Click the Change button to access the **Task Scheduling** dialog box, where you can specify how frequently the Subscription will be automatically updated.
Options Tab	Specify the following Publication data type options: • **Groups and Package Data**—Select this option if you want to select a specific set of packages from the various groups in the Application Catalog to include in the Publication. If you select this option, the Publication Packages panel will appear, prompting you to select packages to include in the Publication. If you do not select this option, all groups and all packages within those groups will be included in the Publication. This field is read-only. • **Merge Module Data**—Select this option if you would also like to include all Merge Modules associated with the packages you have selected in the Publication.
Access List Tab	Specify the list of users allowed to access this Publication. It initially shows a list of users who have the necessary privileges to access the Publication: those users who have been assigned `sysadmin` privileges to this database server through the SQL Enterprise Manager. You can remove users from the list. See Publication Access List Panel for more information.

Table 6-79 • Publication Properties Dialog Box Tabs (cont.)

Tab	Properties
Packages Tab	If Groups and Package Data is selected on the Options tab, this tab lists the Groups and Packages in the SQL database you are connected to. Select the packages that you want to include in this Publication. See Publication Packages Panel for more information.

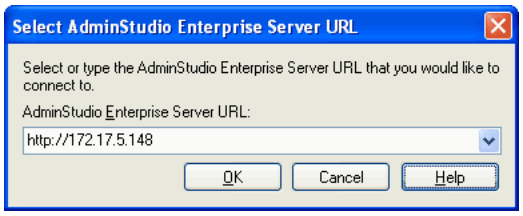
Select Application Catalog Dialog Box

The **Select Application Catalog** dialog box opens when you are attempting to connect to an existing Standalone Application Catalog that requires database authentication by selecting it from a list of recently used Application Catalogs.

- **Enterprise Server**—Select this tab to open the AdminStudio Enterprise Server Application Catalog database. See Enterprise Server Tab.
- **Standalone**—Select this tab to open an Application Catalog other than the AdminStudio Enterprise Server database. See Standalone Tab.
- **Recent**—Provides a list of recently opened Application Catalogs. When you select an Application Catalog and click **OK**, either the Application Catalog opens or you are prompted for login information (if you need authentication to the Application Catalog). See Recent Tab.

Select AdminStudio Enterprise Server URL Dialog Box

If you click the HTTP link on an AdminStudio Enterprise Server Login dialog, this dialog box opens prompting you to identify the AdminStudio Enterprise Server URL that you would like to connect to.



Subscription Manager Dialog Box

The Subscription Manager lists all Publications that you are subscribed to in the SQL Server database you are currently connected to, and displays their Enable/Disable status.

Use this dialog box to view Subscription information, create a new Subscription, disable a Subscription so that it is not automatically updated, manually update a Subscription, and delete a Subscription.

While you can subscribe to multiple Subscriptions, only one Subscription can be active and listed in your Application Catalog at any one time: the one that has been executed (updated) the most recently. For information on how to use the Disable feature on the Subscription Manager to subscribe to more than one Application Catalog, see Specialized User Scenarios.

Note • *The **Subscription Manager** option under **Replication** on the AdminStudio **Catalog** menu is only enabled if the current user is connected to an SQL Server database, has been assigned* `sysadmin` *privileges in SQL Enterprise Manager on that database server, and the database was identified when it was created as being intended for use as a **Subscriber**. See Application Catalog Wizard for information on how to identify databases as Subscribers or Publishers.*

The Subscription Manager dialog box contains the following properties:

Table 6-80 • Subscription Manager Dialog Box Properties

Properties	Description
Publisher	Name of the Publisher Database Server.
Publisher Database	Name of the database on the Publisher Database Server containing the Publication.
Publication	Name of the subscribed Publication.
Description	Description of the subscribed Publication that was entered by the creator of the Publication.
Status	The status of the last attempted update of the Publication: Succeeded or Failed. If the update is currently being executed, Executing appears in this column.
Last Action	Text describing the last action that was performed on this Subscription.
Last Run	The date and time that the Subscription was last updated.
Next Run	The next scheduled update time of that Subscription.
New	Click to create a new Subscription using the **Subscription Wizard**.
Enable/Disable	Click to toggle the selected Subscription's status between Enabled and Disabled. When you disable a Subscription, you are turning off its automatic update feature. Subscriptions that are enabled are preceded by a check mark, while disabled Subscriptions are preceded by a red X. For more information, see Specialized User Scenarios.
Run	Click to manually update the selected Subscription. The Subscription Manager will then replace the data of the Application Catalog's currently active Subscription with the most recent version of the selected Subscription's data. When a disabled Subscription is selected, the Run button is disabled. For more information, see Specialized User Scenarios.
Delete	Click to delete the selected Subscription from the Subscription Manager.

Caution • *If you are unable to receive a Subscription, the most likely cause is that the Publication has not been published. Check to find out if the Publication has been published.*

Task Scheduling Dialog Box

This dialog box is used to specify how frequently a Publication or Subscription will be automatically updated. Both the Publication Manager and Subscription Manager require schedule information. This dialog box is invoked by clicking the **Change** button on the **Details** panel of the **Publication** and **Subscription Wizards**, and the **Details** tab of the **Publication Properties** dialog box.

Note • *A Publication's update frequency can be changed after the Publication is created by accessing the Details tab of the **Publication Properties** dialog box. However, a Subscription's update frequency is specified when the Subscription is created and cannot be changed.*

Table 6-81 • Task Scheduling Dialog Box Properties

Properties	Description
Run the process every:	Select how frequently you want this Publication or Subscription to be automatically updated: Day, Week, or Month.
on	This list offers different selections depending upon the choice you made in the Run the process every field: • Day—Only one selection is listed: Every Day. • Week—Select the day of the week that you want to update this Publication or Subscription. • Month—Select the day of the month that you want to update this Publication or Subscription.
At:	Enter the time of day that you want the update to occur.
Starting on:	Select the day or date that you want this Publication to begin being automatically updated on the day or date you selected from the Run the process every list. For example, if you want the Publication to be automatically updated on the 15th of each month, you would select the first day of the first month that you want the automatic updates to begin, not the first day that the Publication should be updated. If it is Monday and you want the Publication to be updated every Friday, to have the Publication begin being updated on the upcoming Friday, you would select the current day of the week (Monday), but if you didn't want the Publication to be updated until the Friday after next, you would select Saturday.

Click OK to exit the Task Scheduling dialog box.

Test Connection Dialog Box

When you are creating a new Application Catalog using the Application Catalog Wizard, a **Test** button is provided on the Specify Database Information Panel to allow you to test the connection to the selected database server before completing the rest of the Wizard panels. When you click **Test**, this dialog box opens and informs you if the connection was successful.

Virtual Package Association Dialog Box

You can choose to use the **Associate Package** function in Application Manager to manually associate a virtual package with its source Windows Installer package after both packages have been imported into the Application Catalog.

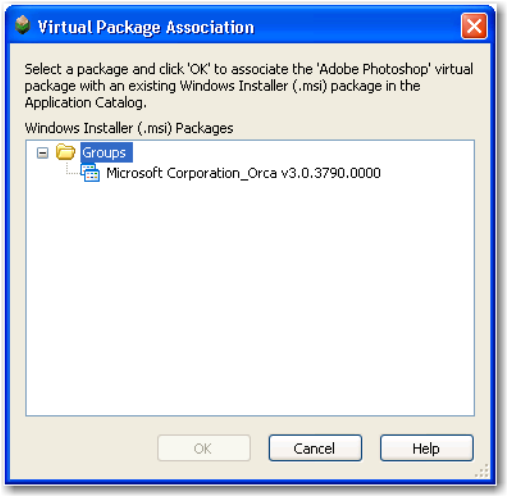


Figure 6-60: Virtual Package Association Dialog Box

If you select a virtual package in the Application Manager tree and then select **Associate Package** from the context menu, the **Virtual Package Association** dialog box opens, listing all of the Windows Installer packages in the Application Catalog. Select the virtual package's source Windows Installer package and click **OK**. The Windows Installer package will now be listed in the **Associated Packages** list in the Application Manager **Virtual Packages** view.

Important • *After you have imported a virtual package into the Application Catalog, you can use the* ***Associate Package*** *function to associate it with any Windows Installer package in the Application Catalog, even one that is not its source package. Therefore, it is preferable to use the Import Wizard to import both the Windows Installer and virtual packages at the same time so that AdminStudio can create the proper associations.*

Wizards

Application Manager includes the following Wizards:

- Application Catalog Wizard
- Import Wizard
- Package Auto Import Wizard
- Merge Wizard
- Publication Wizard
- Subscription Wizard
- Upgrade Wizard

Application Catalog Wizard

You use the Application Catalog Wizard to create a new SQL Server Application Catalog database. This Wizard includes the following panels:

- Welcome Panel
- Specify Database Information Panel
- Select Software Repository Location Panel
- Creating Application Catalog Panel

Welcome Panel

The first panel of the Application Catalog Wizard welcomes you to the Wizard.

This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled.

Table 6-82 • Welcome Panel Options and Buttons

Button	Description
Include Sample Data in New Catalog	Select this option to automatically populate the new Application Catalog with sample package data. This allows you to immediately test AdminStudio features using the packages that have already been imported.
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.

Table 6-82 • Welcome Panel Options and Buttons

Button	Description
Help	Brings up help specific to the Import Wizard panel currently displayed.

Specify Database Information Panel

On the **Specify Database Information** panel of the Application Catalog Wizard and the **Standalone** tab of the Connect Application Catalog Dialog Box, enter the information required to login to the specified Application Catalog.

Table 6-83 • Application Catalog Wizard / Specify Database Information Panel Options

Option	Description
Server	The list of available SQL Servers on the network. You can also manually enter the name of the SQL Server to which you want to connect.
Authentication	Select one of the following options: • **Windows Authentication**—Choose to use Windows network authentication (your network login ID) to log into this Application Catalog. • **Server Authentication**—Choose to use SQL Server login identification for authentication. • **Login ID and Password**—If you chose Server Authentication, enter the appropriate Login ID and Password.
Catalog	Select the catalog from those available on the Server.
Test	Click this button to test whether a connection can be made to the database.

Table 6-83 • Application Catalog Wizard / Specify Database Information Panel Options (cont.)

Option	Description
Create database as a subscriber	Select this check box to designate this new Application Catalog as a Subscriber database in Application Catalog Replication (a database that receives replicated data). When this option is not selected, or when an existing database is upgraded, the Application Catalog is defined as a Publisher database in Application Catalog Replication (a database that makes data available for replication). The AdminStudio interface selectively enables/disables the Replication menu items on the Catalog menu based upon the type of Application Catalog that is currently opened. ***Edition*** • *Application Catalog Replication is included with the AdminStudio Enterprise Edition. If you have AdminStudio Standard or Professional Editions, this check box is disabled. Contact your AdminStudio Sales Representative for more information.* ***Note*** • *This check box only appears when creating a new Application Catalog, not when logging in to an existing Application Catalog.*

Select Software Repository Location Panel

A Windows Installer package is made up of many files that are executed when the setup is run. You only import the `.msi` file into the Application Catalog, not all of the files necessary for installation. With the Software Repository, when you import an installation package into the Application Catalog, all of the files associated with that package are copied into the Software Repository location, a directory that you specify. This allows you to manage those files, preventing them from getting modified or lost.

On the **Select Software Repository Location** panel, you can choose to **Enable the Software Repository** for the new Application Catalog, and specify a **Proxy Account** for AdminStudio to use to make modifications to the directory path selected as the **Software Repository Location**.

This panel includes the following options:

Table 6-84 • Select Software Repository Location Panel Options

Option	Description
Enable Software Repository	Select this option to enable the Software Repository feature for this new Application Catalog.
Software Repository Location	Specify the directory location of the Software Repository for this Application Catalog.

Table 6-84 • Select Software Repository Location Panel Options (cont.)

Option	Description
Proxy Account	Specify a **Login ID** and **Password** for AdminStudio to use when modifying the Software Repository Location directory.

Creating Application Catalog Panel

This panel displays the progress while your new Application Catalog is being created. If the Application Catalog cannot be created, an error message will be displayed.

Import Wizard

The Import Wizard allows you to import the following installation package types into the Application Catalog: Windows Installer packages and associated transform and patch files, virtual packages (Microsoft App-V, VMware ThinApp, and Citrix), merge modules, OS snapshots, Marimba Native Channel Packager files, and other non-MSI setup formats (such as InstallShield Professional or ISMP installations).

The Import Wizard consists of the following panels:

- Welcome Panel
- Select Package Source Panel
- Connect to a Microsoft Configuration Manager Server
- Select Packages Panel
- File Selection Panel
- Folder Selection Panel
- Associate Virtual to MSI Packages Panel
- MST Source Information Panel
- MSP Source Information Panel
- MSM Source Information Panel
- Virtual Package Import Panel
- Target Package Information Panel
- Target Package Information Panel (Software Repository)
- Target Package Information Panel (Transforms/Patches)
- Import Options Panel
- Additional non-MSI Import Options Panel
- Destination Group Panel

- Summary Panel

When run, Application Manager displays its progress messages in the **Import** tab of the **Output Window**.

Note • *Depending on configurations set in the ConflictSolver and Application Manager* ***Options*** *dialog box, validation and conflict identification may take place during import. See Validating During Import and Checking for Conflicts During Import.*

Welcome Panel

You can use the Import Wizard to import the following installation package types into the Application Catalog:

- Windows Installer packages (.msi)
- Merge modules
- Marimba Native Channel Packager files (.ncp)
- Microsoft App-V applications (.sft)
- Citrix profiles (.profile)
- ThinApp applications (.exe)
- OS snapshots (.osc)
- Other setup types

This panel, and others in the Wizard, have four buttons located at the bottom of the panel. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 6-85 • Welcome Panel Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help specific to the Import Wizard panel currently displayed.

Select Package Source Panel

On the **Select Package Source** panel of the Import Wizard, select the source that contains the applications that you want to import.

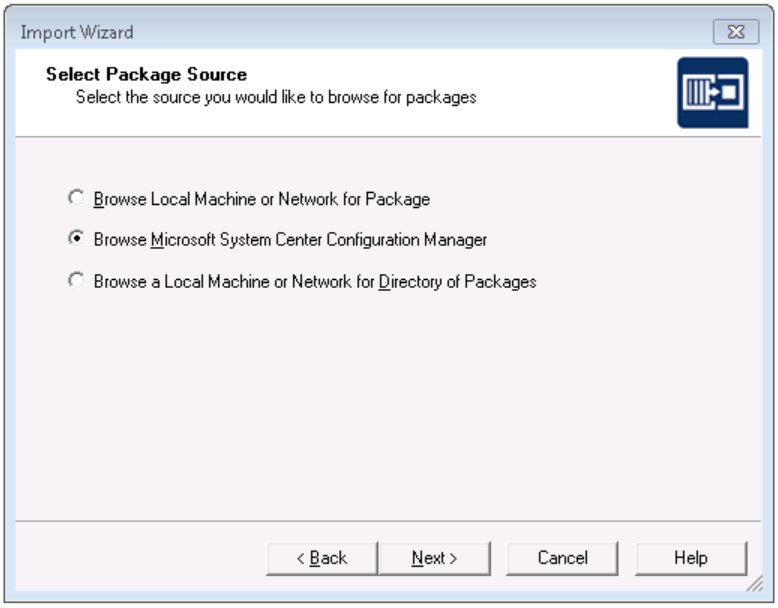


Figure 6-61: Import Wizard / Select Package Source Panel

On the **Select Package Source** panel, select one of the following options:

Table 6-86 • Select Package Source Panel

Option	Description
Browse Local Machine or Network for Package	Select this option to import a single installation package into the Application Catalog. If you select this option and click **Next**, the **File Selection** panel opens, prompting you to browse for the file.
Browse Microsoft System Center Configuration Manager	Select this option if you want to import applications from a Microsoft System Center Configuration Manager server. If you select this option and click **Next**, the **Connect to a MIcrosoft Configuration Manager Server** panel opens, prompting you to connect to a server.
Browse a Local Machine or Network Directory for Packages	Select this option to import a directory of packages into the Application Catalog. If you select this option and click **Next**, the **Folder Selection** panel opens, prompting you to select the folder containing the Windows Installer and/or App-V packages to import.

Connect to a Microsoft Configuration Manager Server

On the **Connect to a Microsoft Configuration Manager Server** panel, which opens if you select **Browse Microsoft System Center Configuration Manager** on the **Select Package Source** panel, you enter connection information to connect to the Microsoft System Center Configuration Manager server that you want to import applications from.

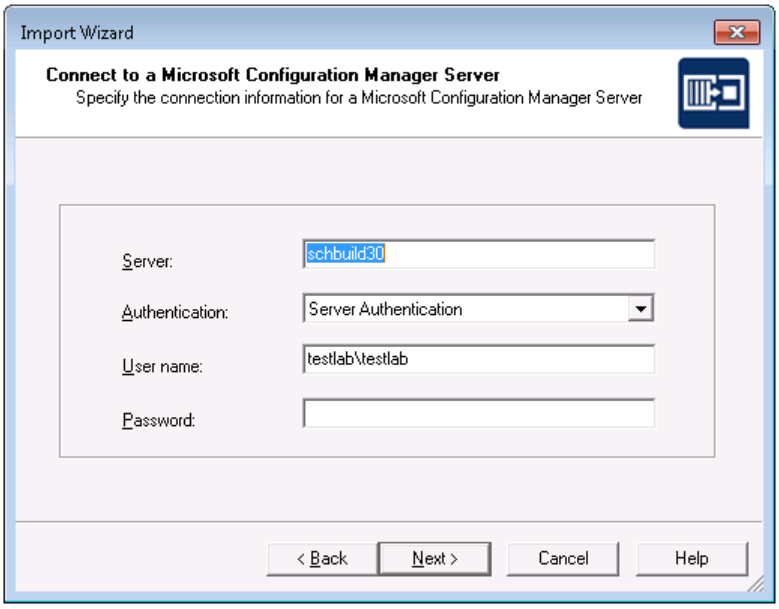


Figure 6-62: Import Wizard / Connect to a Microsoft Configuration Manager Server Panel

On the **Connect to a Microsoft Configuration Manager Server** panel, enter the following information:

Table 6-87 • Connect to a Microsoft Configuration Manager Server Panel

Option	Description
Server	Enter the name of the Microsoft Configuration Manager Server that you want to connect to.

Table 6-87 • Connect to a Microsoft Configuration Manager Server Panel

Option	Description
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server login identification to log into this server. Then enter the appropriate **User name** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Microsoft Configuration Manager Server. ***Note*** • *After you successfully connect to a Microsoft Configuration Manager Server, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*

After you successfully connect to a Configuration Manager Server and click **Next**, the **Select Package** panel opens, where you will be prompted to select the packages on the server that you want to import into the Application Catalog.

Select Packages Panel

The **Select Packages** panel, which opens after you successfully connect to a Configuration Manager server, lists all of the packages in the connected Configuration Manager server, in a tree format.

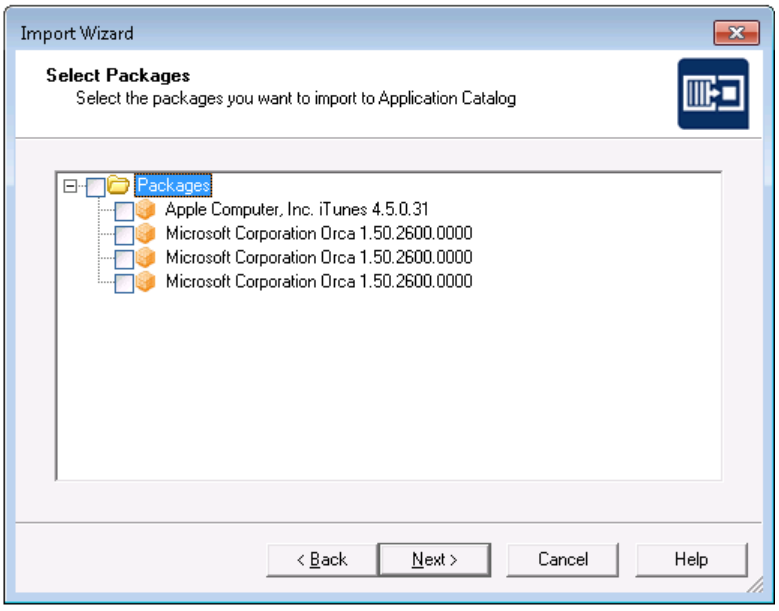


Figure 6-63: Import Wizard / Select Packages Panel

Select the packages that you want to import and click **Next** to continue. The **Summary** panel opens.

File Selection Panel

On the **File Selection** panel, select the package file to import, and click **Next**. The following types of packages can be imported:

Table 6-88 • Import Package Types

Package Type	Description
Windows Installer Files (*.msi)	File that contains all of the information that the Windows Installer requires to install or uninstall an application or product and to run the setup user interface. The `.msi` file can also contain one or more transform files (`.mst`) and one or more patches (`.msp`).
Windows Installer Merge Module Files (*.msm)	Simplified Windows Installer `.msi` files. A merge module cannot be installed alone because its lacks some vital database tables that are present in an installation database. Merge modules also contain additional tables that are unique to themselves. To install the information delivered by a merge module with an application, the module must first be merged into the application's .msi file.
OS Snapshot Files (*.osc)	Files representing a particular computer system's contents. To generate an OS snapshot file, use the OS Snapshot Wizard to scan a computer's operating system and record the files, INI files, shortcuts, and registry entries present.
Native Channel Packager Files (*.ncp)	Files that are used to describe software components, their versions, their underlying structures, and their dependence on other components. Marimba's NCP file is derived from an OSD (Open Software Description) file and is currently used by Marimba's Castanet family of products.
Microsoft App-V (*.sft)	An application that runs within the Microsoft App-V virtual environment. When a Microsoft App-V application is created, the installation data from the original Windows Installer package is converted into an `.sft` file that is compatible with Microsoft App-V version 4.5. The installation shortcuts are converted into `.osd` files, which provide access to programs within the `.sft` file.
Citrix Profiles (*.profile)	An application that runs within a Citrix virtual environment, which prevents it from interfering with other software running on the same machine.
ThinApp Applications (.exe)	A virtual application that runs within a ThinApp virtual environment.
Other Non-MSI Setup Types	Non-MSI setup types (such as InstallShield Professional or ISMP installations).

Adding a File to the Software Repository

A Windows Installer package is made up of many files that are executed when the setup is run. However, only the `.msi` file is imported into the Application Catalog database. To safeguard these additional files against alteration or being misplaced, you can choose to manage these files using the Software Repository.

If you want to add the file to the Software Repository, select **Add the file(s) to the Software Repository** option on the **File Selection Panel**.

Note • *The **Add the file(s) to the Software Repository** option is only displayed if you are connected to an Application Catalog that has the Software Repository enabled. See Enabling the Software Repository in an Existing Application Catalog.*

After you select this option, if AdminStudio determines that there are other files in that directory that could be associated with the selected file, the **Additional Dependencies of the Import File** list appears, enabling you to select any additional files to be managed by the Software Repository.

See Using the Software Repository for more information.

Folder Selection Panel

On the **Folder Selection** panel of the Import Wizard, which opens if you select **Browse a Local Machine or Network for Directory of Packages** on the **Select Package Source** panel, you are prompted to select the directory that contains the Windows Installer and/or Microsoft App-V packages that you want to import into the Application Catalog.

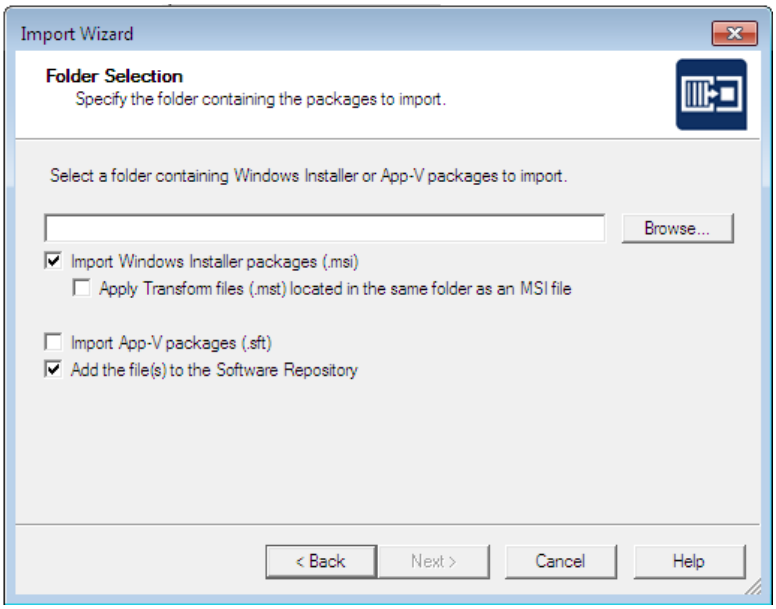


Figure 6-64: Import Wizard / Folder Selection Panel

On the **Folder Selection** panel, enter the following information:

Table 6-89 • Folder Selection Panel

Option	Description
Select a folder containing Windows Installer or App-V packages to import	Click **Browse** and select the directory that contains the Windows Installer and/or App-V packages that you want to import.
Import Windows Installer packages (.msi)	Select this option to import all of the Windows Installer packages in the selected directory.
Apply Transform files (.mst) located in the same folder as an MSI file	Select this option to also import all of the transform (.mst) files that are located in the same directory as a Windows Installer (.msi) file.
Import App-V packages (.sft)	Select this option to import all of the Microsoft App-V (sft) applications in the selected directory.
Add the file(s) to the Software Repository	Select this option if you want to add the imported packages to the Software Repository.

The Import Wizard then searches the selected directory and its subdirectories to locate the packages to import.

Import Wizard's Selection Rules When Importing Packages from a Directory

When importing packages from a directory, it is recommended that you organize the packages you want to import in one root directory, with each package in its own first level subdirectory, such as:

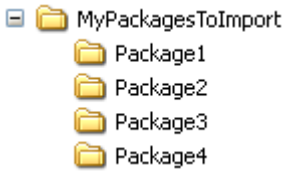


Figure 6-65: Recommended Directory Structure When Adding Packages from a Directory

When you select a folder (such as `MyPackagesToImport`), the Import Wizard scans that folder's first-level subfolders (such as `Package1`, `Package2`, `Package3`, etc.) and uses specific rules to determine which packages it will select for import:

- **All .msi, and .sft files are selected for import**—All `.msi` and `.sft` files in the first-level subfolders are selected for import.
- **If a first-level subfolder does not contain any .msi or .sft files, its subfolders are scanned** —If a first-level subfolder does not contain any `.msi` or `.sft` files, the Import Wizard will scan its child subfolders to locate

package files. However, if a first-level subfolder does contain an `.msi` or `.sft` file, its subfolders are not scanned.

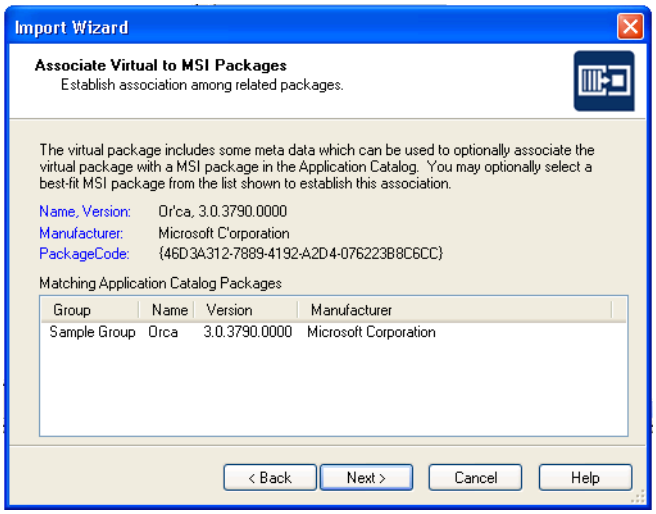
Note • *Once a Windows Installer or Microsoft App-V package is found, further descent through that subfolder tree is blocked. Therefore, no other packages in that subfolder tree are imported.*

Associate Virtual to MSI Packages Panel

This panel opens if you are importing a virtual package and its source Windows Installer package exists in the Application Catalog. If this Windows Installer package has been imported into more than one location in the Application Catalog, multiple packages are listed.

Virtual packages are self-contained entities which ordinarily cannot be modified after they are created. By associating a virtual package with the Windows Installer package which originated it, you have the convenience of being able to easily locate the virtual package's originating Windows Installer package, modify the original Windows Installer package, and then regenerate the virtual package.

Select the package that you want to associate with the virtual package you are importing and click **Next** to continue.



Important • *During virtual package conversion, AdminStudio 9.0 and later creates a file, metadata.ami, that contains metadata identifying the original Windows Installer package that was used to create the virtual package. However, if the virtual package was created by a method other than the AdminStudio 9.0+ conversion process, this metadata file will not have been created and this* ***Associate Virtual to MSI Packages*** *panel will not be displayed.*

MST Source Information Panel

If you are importing a Windows Installer package, you can include additional transforms to be imported along with it.

When the **MST Source Information** panel opens, all of the `.mst` files that are in the same directory as the Windows Installer file you are importing are automatically listed in the **Transform Files (.mst)** list, but only those `.mst` files that AdminStudio determines are probably applicable to this Windows Installer package are selected to be included in the import. If you do not want to import a selected `.mst` file, clear the selection.

You can also add additional transform files:

Task: ***To add transform files:***

1. Click the Browse ( ) button in the Transforms area and select the transform file that you want to import.
2. If the package requires multiple transforms, you can repeat the procedure as necessary.
3. The order in which transforms are applied can be changed by selecting a transform and clicking the up and down arrows.
4. If you need to delete a transform you have added, clear the check box.
5. Click **Next** to continue.

MSP Source Information Panel

If you are importing a Windows Installer package, you can include additional patches to be imported along with it. If a patch file was already in the same directory as the Windows Installer file you are importing, that patch file will automatically appear in the **Patches (.msp or .exe)** list. If you do not want to import it, clear the selection.

Note • *If you specify an update.exe patch file that was created by Developer/DevStudio/InstallShield Editor, Application Manager will extract the .msp file in the Temp folder and then perform the import.*

You can also add additional patch files:

Task: ***To add patch files:***

1. Click the Browse ( ) button in the Patches area and select the patch file that you want to import.
2. If the package requires multiple patches, you can repeat the procedure as necessary.
3. The order in which patches are applied can be changed by selecting a patch and clicking the up and down arrows.
4. If you need to delete a patch you have added, clear the check boxes.

5. For patches to be applied to a Windows Installer package, it is necessary to perform an Administrative install of the Windows Installer package and then perform an Administrative install of each patch package one by one. This way, the content of each patch package is appended to the Windows Installer package at the Administrative install location.

   Therefore, next to the **Specify the location for Administrative install** box, click the Browse button and specify a directory to provide Application Manager with the location where an Administrative Install of selected Windows Installer package will be performed.

6. Click **Next** to continue.

MSM Source Information Panel

If you are importing a Merge Module by selecting a Merge Module on the **Merge Modules** tab and selecting **Import Merge Module** from the context menu, this panel opens, allowing you to import multiple Merge Modules into the Application Catalog at one time.

Task: ***To import merge modules from the Merge Modules tab:***

1. Open **Application Manager**.
2. Open the **Merge Modules** tab.
3. Select the root Merge Module or one of the imported Merge Modules and select **Import Merge Module** from the context menu. The **MSM Source Information** panel of the **Import Wizard** opens.
4. Click the Browse ( ) button in the **Merge Modules** area and select the merge module file that you want to import.
5. To import multiple patches, you can repeat the procedure as necessary.
6. The order in which merge modules are applied can be changed by selecting a merge module in the list and clicking the Move Up and Move Down arrows.
7. If you need to delete a merge module you have added, clear its check box.
8. If you want to store this merge module in the Software Repository, select the **Add the file(s) to the Software Repository** option.
9. Click **Next**. The **Summary** panel opens.
10. Click **Finish** to accept these options and begin the import.

    A report of the import process appears on the **Import** tab in the Output window.

Note • *You can also import a Merge Module into the Application Catalog by selecting **Import Package** on the **Catalog** menu. However, using that method, only one Merge Module can be imported at a time.*

Virtual Package Import Panel

This panel opens if the Import Wizard finds a virtual package in the same directory as the Windows Installer package you have selected to import.

The Import Wizard searches the selected package's folder and subfolders for virtual packages created by AdminStudio. If a virtual package is found that appears to have been created by AdminStudio, then it will be listed on this panel.

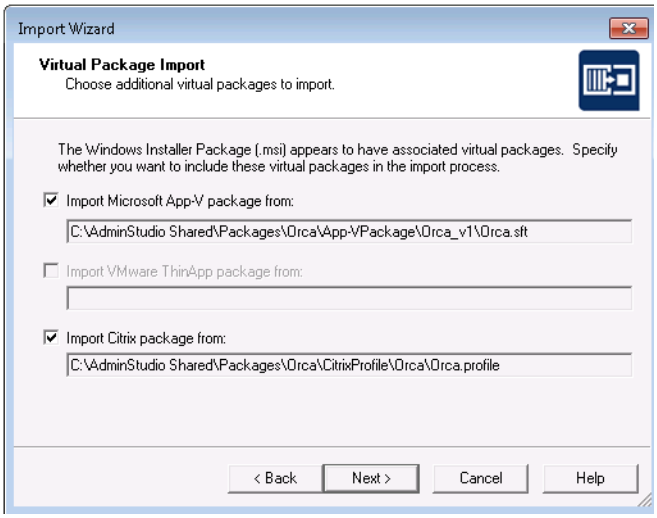


Figure 6-66: Import Wizard / Virtual Package Import Panel

Select the package or packages you want to import into the Application Catalog and click **Next** to continue.

Note • *If a virtual package created by AdminStudio exists in a directory other than the one that contains the selected Windows Installer package, it will not be listed on this panel. After the import of the selected Windows Installer package is complete, you can use the Import Wizard to import virtual packages separately, as described in Importing Virtual Packages. You would then be prompted to associate the virtual package with its already-imported source Windows Installer package.*

Target Package Information Panel

This panel opens if you are attempting to import a package that has been identified as a duplicate to a package that is already in the Application Catalog (and is not stored in the Software Repository). The identification of a duplicate package is based upon the identifiers selected on the **Duplicate Package** tab of the ConflictSolver and Application Manager Options Dialog Box.

An automatically generated new name is displayed in the **Package Name** field. You can either accept this generated name or enter a name of your choosing to identify the duplicate package before the package is imported.

Note • *The name that is entered on the **Package Name** field is the name that will be displayed in the product tree on the left side of the ConflictSolver and Application Manager views. However, changing this "display" name does not change the Product Name that appears in the title bar of the ConflictSolver and Application Manager Product Views.*

Target Package Information Panel (Software Repository)

This panel opens if you are attempting to import a package that has been identified as a duplicate to a package that is already in the Application Catalog (and is stored in the Software Repository). The identification of a duplicate package is based upon the identifiers selected on the **Duplicate Package** tab of the ConflictSolver and Application Manager Options Dialog Box.

An automatically generated new name is displayed in the **Package Name** field. Because this package is part of the Software Repository, you have several options:

- The package should be treated as a new version of the existing package
- The package should be treated as a new package named [the name entered in the **Package Name** field]
- The package should overwrite the existing package version

Make your selection and click **Next**.

For more information, see Using the Software Repository.

Note • *The name that is entered on the **Package Name** field is the name that will be displayed in the product tree on the left side of the ConflictSolver and Application Manager views. However, changing this "display" name does not change the Product Name that appears in the title bar of the ConflictSolver and Application Manager Product Views.*

Target Package Information Panel (Transforms/Patches)

This panel opens if you are attempting to import a transform or patch file that has been identified as a duplicate to a file that is already in the Application Catalog (and is stored in the Software Repository).

- **Updated file**—If this file is an updated version of a transform or patch file that has already been imported into the Application Catalog, select the **Reimport new version of the package with this file change** option.
- **File associated with a different package**—If this file has the same name as a transform or patch file that has already been imported into the Application Catalog, but is not associated with the same Windows Installer package as the existing file, select the group that contains the Windows Installer package that this file is associated with.

Import Options Panel

The **Import Options** panel allows you to select custom tables that are not part of a standard Windows Installer package, and direct Application Manager to import them. Indicate which custom tables you want to import by selecting the check box next to the table name. You can also use the Select All and Clear All buttons to make selections.

To add one of the listed tables to the **Ignore Tables** list on the **Import** tab of the ConflictSolver and Application Manager Options Dialog Box (so that it will not be imported into the Application Catalog during any import), select the table name and click **Add to Ignore List**.

Additional non-MSI Import Options Panel

If you are importing a non-Windows Installer package, it is necessary to enter a name for the package and the base directory for the package (where all of its installation files are located).

Table 6-90 • Import Wizard / Additional non-MSI Import Options

Options	Description
Product Name	Enter a name to identify this package in AdminStudio.
Directory	Click the Explore button to select the base directory of this package where all of its associated files can be found.
Include entire directory hierarchy	Select this option to include the selected directory and all of its subdirectories.

Destination Group Panel

The Destination Group panel allows you to specify the group (or groups) into which you want to import the MSI package.

You also have the option of selecting all groups by clicking the Select All button, or deselecting groups by clicking the Clear All button. If a group you want to import the package into does not exist, click the New button to create and configure it in the Group Properties dialog box.

***Note** • This panel is only displayed when launching the Import Wizard from the Catalog menu. If you right-click on a group in the Application Manager Product View and import a package or snapshot, the destination group will be group on which you right-clicked.*

Summary Panel

Before executing the import, review the information in the Summary panel about the options selected in the previous panels.

Depending on the import type and how the Import Wizard was invoked, clicking Back returns you to the **Destination Group** panel, **MSM Source Information** panel, **OS Snapshot Information** panel, or **Other Setup Installation Files** panel. Click **Finish** to begin import.

Package Auto Import Wizard

Note • *To use the Package Auto Import feature, you must have AdminStudio 10.0 SP2 Administrator permission. For all other users, the Package Auto Import options will be disabled.*

The Package Auto Import feature allows you to copy packages between various Application Catalogs and maintain consistency between the copied packages. This allows you to maintain multiple Application Catalogs that match your corporate organization and optimize the performance of AdminStudio.

The Package Auto Import Wizard consists of the following panels:

- Welcome Panel
- Package Auto Import Type Selection Panel
- Remote Link Application Catalog Panel
- Folder Panel
- Groups Panel
- Schedule Panel
- Summary Panel

When run, Application Manager displays the output report in the Package Auto Import tab of the Output Window.

Note • *You can use the Package Auto Import Wizard to copy Windows Installer packages (.msi), legacy applications (.exe), and Microsoft App-V applications (.sft) between various Application Catalogs. However, the Package Auto Import Wizard does not support copying Citrix XenApp profiles or VMware ThinApp applications.*

Tip • *When you link to a package in a remote Application Catalog, Application Manager connects to the remote Application Catalog and imports a package into the local Application Catalog. In order for Package Auto Import to work properly, both the remote Application Catalog and the remote package file must be accessible to the linked users at all times. In other words, both the remote Application Catalog and all of the package files that have been imported into it must be able to be accessed by linked users via a shared network drive.*

Welcome Panel

The first panel of the Package Auto Import Wizard welcomes you to the Wizard.

This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 6-91 • Welcome Panel Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help about the specific Package Auto Import Wizard panel.

Package Auto Import Type Selection Panel

On this panel, select the type of monitoring process that you want to configure. Select one of the following options:

- **Remote Application Catalog**—Monitor packages in a Remote Application Catalog and automatically import/re-import those packages at scheduled intervals. Application Manager can link to one or multiple Application Catalogs, and can link to one or multiple packages within those Application Catalogs.
- **Network Directory**—Automatically import or re-import all packages in a monitored network directory into your Application Catalog at scheduled intervals.

Note • *You can use the Package Auto Import Wizard to copy Windows Installer packages (.msi), legacy applications (.exe), and Microsoft App-V applications (.sft) between various Application Catalogs. However, the Package Auto Import Wizard does not support copying Citrix XenApp profiles or VMware ThinApp applications.*

Remote Link Application Catalog Panel

The Remote Link Application Catalog panel of the Package Auto Import Wizard lists Application Catalogs that you have recently accessed.

To select the Application Catalog that you want to link to, you can perform either of these options:

- Select the Application Catalog you want to link to from the **Catalogs** list.
- Click **Browse** to open the Select Application Catalog Dialog Box and connect to an existing Application Catalog as described in Connecting to an Existing Application Catalog.

During the linking process of an SQL Server Application Catalog, Application Manager will save the name of the database server, authentication mode, and name of the Application Catalog. Passwords are not saved.

Folder Panel

On the **Folder** panel, use the **Browse** button to select the network directory folder that you want to monitor.

To also monitor the selected folder's subfolders, select the Also include sub-directories option.

Note • *While you are permitted to enter a local directory (rather than a network directory) as the monitored directory, this might cause problems if a person monitoring this directory has the same directory on his local machine. For example, if a directory named `C:\MyAppCatalogs` is chosen to be the monitored directory, and a person who choose to monitor this directory also has a directory on his local machine named `C:\MyAppCatalogs`, Application Manager would monitor that person's local directory rather than the directory that was chosen to be the monitored directory.*

Note • *If you select this option, when packages in subdirectories are imported into your Application Catalog, they will be placed in groups that maintain the folder hierarchy. For example, if you are monitoring a directory named `Marketing`, and that directory has a subdirectory named `Graphics` that contains packages, when those packages are imported into your Application Catalog, they will be placed in the `Graphics` subgroup of the destination group.*

Groups Panel

The **Groups** panel of the Package Auto Import Wizard lists the group structure and the packages that are in the specified Remote Application Catalog. Choose the packages that you want to link to by selecting the check box next to the package name.

Click **Next** to continue.

Note • *You can use the Package Auto Import Wizard to copy Windows Installer packages (.msi), legacy applications (.exe), and Microsoft App-V applications (.sft) between various Application Catalogs. However, the Package Auto Import Wizard does not support copying Citrix XenApp profiles or VMware ThinApp applications.*

Destination Group Panel

When you are setting up Network Directory monitoring, the **Destination Group** panel opens, prompting you to select the Application Catalog group into which the packages will be imported.

Select the group in the Application Catalog where you want these packages to be imported, and click Next.

Note • *Packages that are imported using Remote Application Catalog monitoring are imported into the Package Auto Import group in the Application Catalog.*

Schedule Panel

On the **Schedule** panel of the Package Auto Import Wizard you can choose to enable automatic import of linked packages (or packages in a monitored Network Directory), and can also specify how often the Remote Application Catalog or Network Directory will be checked for updates.

The following options are included:

Table 6-92 • Schedule Panel Properties

Option	Description
Schedule AdminStudio to update the changed packages	Select this option to enable automatic import of the linked packages (or packages in the monitored Network Directory) at scheduled intervals. When this option is selected, the rest of the scheduling fields on this panel are enabled.
Run the process	Select **Daily** or **Weekly** to specify how often you want the Remote Application Catalog or Network Directory to be checked for updates.
At	Select the time of day that you want the automatic import to occur by selecting the hour, minutes, seconds, or AM/PM setting and then clicking the up and down arrows to adjust the setting.
Starting on	Click on the arrow to open the Calendar and then select the first day that you want automatic importing to occur.

Summary Panel

The **Summary** panel provides a detailed summary of the options that were selected in the previous panels of the Wizard. Click **Finish** to complete the remote link. Clicking **Back** returns you to the previous panel.

Merge Wizard

The Merge Wizard, available by clicking **Merge** on the **Catalog** menu, allows you to merge application data between two Application Catalogs.

The source Application Catalog's data is merged into the currently open Application Catalog. This is a one-time-only database merge. The merged data is not linked in any way to the source Application Catalog.

Merging could be used in an organization where multiple people test packages. After the packages have been tested, all of the "tested" applications could be consolidated into one centralized Application Catalog. Another reason to merge Application Catalogs would be a need to have the same data available in multiple geographic locations.

The following panels are included in the Merge Wizard:

- Welcome Panel
- Source Application Catalog Panel
- Groups Panel
- Progress Panel

Note • *You can use the Merge Wizard to merge Windows Installer packages (.msi), legacy applications (.exe), and Microsoft App-V applications (.sft). However, the Merge Wizard does not support merging Citrix XenApp profiles or VMware ThinApp applications.*

Welcome Panel

The Merge Wizard allows you to merge application data between two Application Catalogs. The source Application Catalog's data is merged into the currently open Application Catalog.

The first panel in the Merge Wizard is the **Welcome** panel. Click **Next** to proceed to the **Source Application Catalog** panel.

Source Application Catalog Panel

The **Source Application Catalog** panel lists of all of the Application Catalogs that the user has permission to view. Select the Application Catalog that contains the data that you want to merge.

If the Application Catalog that you select requires additional authentication, the Select Application Catalog Dialog Box opens. Connect to that catalog as described in Connecting to an Existing Application Catalog.

Groups Panel

From the **Groups** panel, you can select the Windows Installer packages, legacy applications, and Microsoft App-V applications that you want to merge. Selecting a Group icon selects all of the applications in that group.

The available groups displayed are from the source Application Catalog, and data from the selected groups will be merged into the current Application Catalog. If you select the option to **Replace duplicate application data**, if a duplicate package exists in the current Application Catalog, the source Application Catalog's package will overwrite the duplicate.

Note • *You can use the Merge Wizard to merge Windows Installer packages (.msi), legacy applications (.exe), and Microsoft App-V applications (.sft). However, the Merge Wizard does not support merging Citrix XenApp profiles or VMware ThinApp applications.*

Progress Panel

The **Progress** panel displays messages from the merge process.

When the merge is complete, review the log as necessary and click **Finish** to exit the Wizard.

OS Snapshot Wizard

The following topics contain information about each Wizard panel and dialog box available through the OS Snapshot Wizard. The help topics in this reference are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a Wizard or dialog box.

Select one of the following links for OS Snapshot reference information:

- Welcome Panel
- Project Information Panel
- Analyzing Panel
- OS Snapshot Summary Panel
- Analysis Options Dialog Box
- ISSnapshot.ini File

Welcome Panel

The **Welcome** panel appears when you first launch the OS Snapshot Wizard, providing some introductory information about the use of the OS Snapshot Wizard.

The **Next** button advances you to the **Project Information** panel.

Project Information Panel

The Project Information panel gathers information necessary for taking the OS Snapshot.

You must provide the following information before the Start button is enabled, allowing you to take the snapshot.

Table 6-93 • Project Information Panel Options

Option	Description
OS Snapshot project name	Provide a name for the snapshot file (.osc).
OS Snapshot project folder	Provide the directory in which snapshot data will be stored. Either enter the path in the field, or click the Browse (...) button to navigate to it. If the directory already exists, a confirmation dialog box opens when you click the Next button.

If you want to review or change current capture settings, click Edit to display the Analysis Options dialog box.

Analyzing Panel

The **Analyzing** panel appears while the OS Snapshot Wizard analyzes your system.

Following the snapshot, the **Summary** panel appears.

OS Snapshot Summary Panel

At the end of the OS Snapshot process, the Summary panel is displayed, containing information about the OS Snapshot that was just performed.

Prior to clicking Finish, review the information to ensure the snapshot contains the data you expected.

Following the OS Snapshot process, you can import the snapshot into the Application Catalog and use it as a baseline to which setups can be compared.

Caution • *OS Snapshots should only be used for comparison in ConflictSolver. You should never attempt to convert an OS Snapshot into an MSI package.*

Analysis Options Dialog Box

The **Analysis Options** dialog box, accessible by clicking **Edit** from the **Project Information** panel, allows you to specify capture types for the OS snapshots.

You can select the following:

- Files
- INI files
- Shortcuts
- Registry data

Additionally, you can restrict directory analysis to specific directories, which can significantly improve OS Snapshot Wizard performance. Click New to add a directory restriction, edit to modify an existing restriction, or delete to remove a restriction.

Options set in this dialog box apply to the current and subsequent snapshot sessions.

ISSnapshot.ini File

The ISSnapshot.ini file is the default exclusion file for the OS Snapshot Wizard. It contains exclusions to be applied when capturing an OS snapshot, and mainly focuses on specific items that should not be included in applications, such as InstallShield Professional-specific COM settings and OS Snapshot–specific registry entries.

The file is located in the Windows folder, and can be edited using the Exclusions Editor, or using a text editor. See Exclusions Editor Interface.

Note • *It is strongly recommended that you not modify this file, as it increases the likelihood of either inadvertently omitting necessary pieces of the OS snapshot, or including registry entries or files that should not be part of the snapshot.*

Publication Wizard

The Publication Wizard for Application Catalog Replication walks you through the process of creating an SQL Server Publication from an Application Catalog database.

The Publication Wizard consists of the following panels:

- Welcome Panel
- Publication Details Panel
- Publication Data Options Panel
- Publication Access List Panel
- Publication Packages Panel
- Publication Summary Panel

Welcome Panel

The first panel of the Publication Wizard welcomes you to the Wizard.

This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 6-94 • Welcome Panel Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help about the specific Wizard panel.

Publication Details Panel

On the **Publication Details** panel, you are prompted to enter basic Publication properties of name, description, and the schedule. The **Publication Details** panel contains the following fields:

Table 6-95 • Publication Wizard / Publication Details Panel Options

Option	Description
Name	Enter a name to identify this Publication. If you enter a duplicate name of an existing Publication, you will receive an error message. ***Caution*** • *The Publication Name cannot be a T-SQL reserved word.*
Description	Enter a description of the purpose or intended subscribers to this Publication. This description will be listed on the Publication Manager.
Schedule	In this field, you specify how often you would like this Publication to be automatically updated. When a Publication is updated, all of its data is replaced by replicated data from the Publisher's local Application Catalog database. This field is not directly editable. Click the Change button to access the **Task Scheduling** dialog box, where you can specify how frequently a Publication will be automatically updated.

Publication Data Options Panel

The **Publication Data Options** panel prompts you to specify the particular types of data to include in the Publication. This panel contains the following fields:

Table 6-96 • Publication Wizard / Publication Data Options Panel Properties

Properties	Description
Groups and Package Data	Select this option if you want to select a specific set of packages from the various groups in the Application Catalog to include in the Publication. If you select this option, the **Publication Packages** panel opens, prompting you to select packages to include in the Publication. If you do not select this option, all groups and all packages within those groups will be included in the Publication.
Merge Module Data	Select this option if you would also like to include all Merge Modules in the Application Catalog in the Publication.

Publication Access List Panel

On the **Publication Access List** panel, you specify the list of users allowed to access this Publication. It initially shows a list of users who have the necessary privileges to access the Publication: those users who have been assigned `sysadmin` privileges to this database server through the SQL Enterprise Manager. You can remove users from the list.

The **Publication Access List** panel contains the following options:

Table 6-97 • Publication Wizard / Publication Access List Panel Options

Option	Description
Type	Displays an icon to identify each listing as either a user or a group.
Name	Lists the name of the user or group.
Remove	Click to remove a user or group from the list. If a user is removed from the list, they will no longer be able to access the Publication, but they still would be able to be a user of the SQL Server and Publication database.

Caution • *Only users that have been assigned* `sysadmin` *privileges in SQL Enterprise Manager to the connected database server can establish the Publisher-Subscriber relationship between the databases and are therefore listed on this panel. You can remove a user from this list (so that they will not be able to access this Publication), but you cannot add additional users to this list using AdminStudio. To add new users, you must use SQL Enterprise Manager.*

Click Next to access the **Publication Packages** panel (if you selected the **Groups and Package Data** option on the **Publication Data Options** panel) or the **Publication Summary** panel if you did not select that option.

Publication Packages Panel

On the **Publication Packages** panel, the Groups and Packages in the Application Catalog you are connected to are listed. Select the packages that you want to include in this Publication.

Caution • *You do not have to include all of the packages in the Application Catalog in this Publication. However, when this Publication is replicated in the Subscriber's Application Catalog, all of the Groups in the Publisher's Application Catalog will be listed, even if they do not contain any packages.*

Caution • *When a user subscribes to this Publication, they are subscribing to all of the packages in all of the Groups that the Publisher has included. However, if the Publisher adds a new package to one of these same Groups after he has created the Publication, the new package is not automatically added to the existing Publication, making*

it unavailable for Subscription. To publish this new package, a new Publication would have to be created or an existing Publication would have to be edited.

***Note** • This panel only appears if you selected the **Groups and Package Data** option on the **Publication Data Options** panel.*

Publication Summary Panel

This panel provides a detailed summary of the options that were selected in the previous panels of the Wizard

The **Publication Summary** panel contains the following options:

Table 6-98 • Publication Wizard / Publication Summary Panel Options

Option	Description
Generate publication as script	To generate an SQL script of the code necessary to create this Publication rather than actually create it, select this option and enter a path name for the script file. For more information, see Replicating Application Catalogs in Controlled SQL Database Environments.
Publish the publication immediately after creation	To publish this Publication immediately after the Publication Wizard is finished, select this option. The name of this Publication would be listed on the Publication Manager with a status of Executing, and the Publication data would be replicated from your Application Catalog to the Publisher Database Server—making it available for Subscription immediately. If you do not select this option, you would have to click Run on the Publication Manager to publish this Publication manually, or the Publication would automatically be published on its scheduled date.

Click Finish to complete the creation of this Publication.

Subscription Wizard

The Subscription Wizard for Application Catalog Replication walks you through the process of creating an SQL Server Subscription to an Application Catalog database.

The Subscription Wizard consists of the following panels:

- Welcome Panel
- Subscription Details Panel
- Subscription Selection Panel
- Subscription Summary Panel

Welcome Panel

The first panel of the Subscription Wizard welcomes you to the Wizard. This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 6-99 • Welcome Panel Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help about the specific Wizard panel.

Subscription Details Panel

On the **Subscription Details** panel, you are prompted to enter connection information for the Publisher database and scheduling information. This panel contains the following fields:

Table 6-100 • Subscription Details Panel Properties

Properties	Description
Database	From the **Catalogs** list, select the Publisher Application Catalog you want to subscribe to, or click **Browse** to open the Select Application Catalog Dialog Box and connect to an existing **Publisher** Application Catalog as described in Connecting to an Existing Application Catalog ***Note*** • *You must connect to a SQL Server database that was identified as being intended for use as a Publisher when it was created. See Application Catalog Wizard.*
Schedule	In this field, you specify how often you would like this Subscription to be automatically updated. When a Subscription is updated, all of its data is replaced by replicated data from the Publisher database. This field is not directly editable. Click the **Change** button to access the **Task Scheduling** dialog box, where you can specify how frequently the Subscription will be automatically updated.

Subscription Selection Panel

The **Subscription Selection** panel displays the Publications available from the selected Publisher database that satisfy the **Publication Access List** requirements for the currently connected user. In other words, it lists the Publications that this user has permission to subscribe to.

Select one Publication from this list to subscribe to and click **Next** to continue to the **Summary** panel.

Subscription Summary Panel

This panel provides a detailed summary of the options that were selected in the previous panels of the Wizard.

To generate an SQL script of the code necessary to create this Subscription rather than actually create it, select the Generate subscription as script option and enter a path name for the script file. For more information, see Replicating Application Catalogs in Controlled SQL Database Environments.

Click **Finish** to complete the creation of this Subscription.

Upgrade Wizard

When you attempt to open an AdminStudio 5.0, 5.5, 6.0, 7.0, or 7.5 Application Catalog in AdminStudio 10.0 SP2, you are prompted to upgrade it to use the AdminStudio 10.0 SP2 schema.

Log files for the upgrade are created in the following directory:

AdminStudio Shared Directory\ConflictSolver\Logs

***Note** • Note the following regarding upgrading an existing Application Catalog:*

- *The upgrade of AdminStudio 3.0, 3.01, and 3.5 databases is not supported by AdminStudio 7.0 or later.*
- *Starting with AdminStudio 8.0, Microsoft Access databases are not supported.*
- *Starting with AdminStudio 9.01, Oracle databases are not supported.*
- *When an SQL Server Application Catalog database is upgraded, the old tables are not dropped from the Application Catalog.*

The Upgrade Wizard consists of the following panels:

- **Welcome Panel**—Initial panel displayed when the Upgrade Wizard is launched. Click **Next** to proceed with the upgrade.
- **Progress Panel**—Displays the results of the upgrade. Click **Finish** to exit the Upgrade Wizard.

Upgrading Pre-AdminStudio 5.0 Application Catalogs

Pre-AdminStudio 5.0 Application Catalogs cannot be upgraded automatically by AdminStudio 7.0 or later. However, you can upgrade them using the Legacy Upgrade Wizard, a standalone utility that was included with AdminStudio 7.0 and 7.5. The Legacy Upgrade Wizard utility is installed in the following directory:

```
C:\Program Files\InstallShield\AdminStudio\7.x\Common\LegacyUpgradeWizard.exe
```

If you do not have a copy of AdminStudio 7.0 or 7.5 available to you, contact Technical Support.

Database Server Permissions

In order to operate some AdminStudio tools, AdminStudio users require specific database permissions. Depending upon the type of user, you may wish to be more selective in the permissions you assign to these users.

If you have AdminStudio Enterprise Edition, you can assign permissions to individual users using the Role functionality in AdminStudio Enterprise Server, as described in AdminStudio and Workflow Manager Roles and Permissions. Otherwise, you can provide more selective restrictions at the database server level using the information in the following table, AdminStudio Database Server Permissions.

Every AdminStudio user will need at a minimum read privilege to every table in the Application Catalog. The minimum permissions are described below, based upon the type of operation you want the user to perform.

Table 6-101 • AdminStudio Database Server Permissions

Type	Description
General User Administrative Process	General administrative processes cover a range of activities such as adding groups, moving packages around, adding comments, updating extending attributes, etc. For example, these tables include **cstblPackage**, **cstblGroups**, and **cstblGroupPackages** Any Application Catalog table which is not referenced explicitly in the discussion for the other AdminStudio processes should be considered a general user administrative table. Most AdminStudio users should have write access to these tables.

Table 6-101 • AdminStudio Database Server Permissions (cont.)

Type	Description
Import Process	The user importing MSI packages, merge modules, or just about anything will require write access to a significant set of Application Catalog tables depending on the type of import. For example: • **MSI package file**—For MSI package file import operations, those Application Catalog tables with a **csmsi** prefix are populated. • **Merge modules**—For merge module import operations, the **csmsm** prefixed tables are used. • **Patches**—For patch import operations, the **cspch** prefix tables are used. • **OS snapshots**—For OS snapshot import operations, the **osc** prefix tables are used. Occasionally, the Import process will need to create new tables in the Application Catalog to support the import of custom table data. Either the necessary server privileges need to be given for these users, or the import of custom table data needs to be disabled within the Application Manager **Options** dialog box.
Validation Process	For this process, the user will need to be able to write entries into the **cstblValidationResults** and **cstblValidationConfiguration** tables.
Dependency Scanning Process	For this process, the user will need to be able to write entries into the **cstblPackageExeDependencies** table.
Conflict Detection and Resolution Process	For this process, the user will need to be able to write entries into the **cstblConflict** prefixed table names.
Patch Impact Analysis Process	For this process, the user will need to be able to write entries into the **cstblPatchConflict** prefixed table names. This process will create and delete some temporary tables and, as such, the user performing this process should have the necessary server privileges to perform these operations.
Package Auto Import Process	The Package Auto Import process will ultimately generate a series of Import operations, and so the user performing these operations should have the Import process rights described above. If the user wants to edit these operations in the Wizard, then they will need write accession to the **cstblSubscribed** prefixed tables.
Workflow Operations Process	For this process, the user will need to be able to write entries into the **wftbl** prefixed table names.
Tools Properties Operations	For this process, the user will need to be able to write entries into the **wftblTools** table name.

Table 6-101 • AdminStudio Database Server Permissions (cont.)

Type	Description
Job Manager Process	For this process, the user will need to be able to write entries into the **As** prefixed table names.
Pre-Deployment Testing	For this process, the user will need to be able to write entries into the **pdt** prefixed table names.

Note • *A number of processes within AdminStudio generate data which can subsequently be deleted by the AdminStudio user. Any discussion of the minimum privileges required for a specific AdminStudio process will also imply the privileges to delete this same data.*

Application Manager Command-Line Functionality

Note • *If you invoke Application Manager using the command line* `iscmide.exe` *without passing any parameters, Application Manager will launch in UI mode.*

Tip • *By default, all packages are imported into the root group. If you want to import packages to specific groups, you must use a configuration file (*`-C` *parameter), and specify a group for each package.*

ConflictSolver and Application Manager support the following command-line parameters. These parameters are case-insensitive, and must be preceded with either a dash symbol (-) or a forward slash (/).

Table 6-102 • Command Line Parameters

Command Line Parameter & Examples	Description
-? `iscmide.exe -?`	Displays version information and help text for command-line parameters.

-C"configuration_file_name"
 	
`iscmide.exe`
`-I`
`-C"C:\MyConfigs\myconfig.ini"`
	The name and location of a configuration file (.ini) containing any required parameters for import. If you are using a configuration file, the only necessary parameters to pass at the command line are `-I` and `-C"configuration_file_name"`.
	
	You can include all other parameters inside the INI file.

Table 6-102 • Command Line Parameters

Command Line Parameter & Examples	Description

-D"application_catalog_name"
 	
`iscmide.exe`
`-S"mysql\sql1"`
`-U"admin"`
`-P"admin"`
`-D"mycatalog"`
`-IF"c:\mypackages\mymsi.msi;`
`c:\mypackages\mytrans.mst;"`
	The name of the Application Catalog.
	
	This parameter is only used for SQL Server Application Catalogs, and is a required parameter when using a SQL Server–based Application Catalog.

-I
 	
`iscmide.exe`
`-I`
`-C"c:\mypackages\myconfig.ini"`
	This option indicates that a bulk import operation is to be performed. If you are using -IF or -IMM, you do not need to specify this, as these parameters inform Application Manager to perform an import operation.

-IF"msi_file_name[; mst1; mst2...]"
-IF"msi_file_name[; msp1; msp2...]"
-IF"msi_file_name[; mst1; msp1; msp2...]"
 	
`iscmide.exe`
`-F"c:\mycatalogs\conflict.mdb"`
`-IF"c:\mypackages\mymsi.msi;`
`c:\mypackages\mytrans.mst;"`
 	
`iscmide.exe`
`-F"c:\mycatalogs\conflict.mdb"`
`-IF"c:\mypackages\mymsi.msi;`
`c:\mypackages\mypatch.msp;`
`c:\mypackages\mytrans.mst;"`
 	
`iscmide.exe`
`-F"c:\mycatalogs\conflict.mdb"`
`-IF"c:\mypackages\mymsi.msi`
	The name and full path of the MSI file to be imported, and optionally a semicolon-delimited list of transforms or patches to be applied before importing the package.

-IMM"merge_module_file_name"
 	
`iscmide.exe`
`-F"c:\mycatalogs\conflict.mdb"`
`-IMM"c:\mymodules\crystal.msm"`
	The name and full path of the merge module to be imported.

-INCP
 	
`iscmide.exe`
`-INCP"C:\Acrobat.ncp"`
	Use to specify the Native Channel Packager (NCP) file that is to be imported.

-L"logfile_name"
 	
`iscmide.exe`
`-I`
`-C"c:\mypackages\myconfig.ini"`
`-L"c:\mypackages\mylog.txt"`
	The output log file name.

Table 6-102 • Command Line Parameters

Command Line Parameter & Examples	Description

-P"password"
 	
`iscmide.exe`
`-S"mysql\sql1"`
`-U"admin"`
`-P"admin"`
`-D"mycatalog"`
`-IF"c:\mypackages\mymsi.msi;`
`c:\mypackages\mytrans.mst;"`
	The password for the SQL Server Application Catalog.
	
	This parameter is only used for SQL Server Application Catalogs, and is a required parameter when using a SQL Server–based Application Catalog. The only exception to this requirement is if you want to take advantage of SQL integrated security (Windows Authentication). In this case, do not use the -P parameter.

-Q `iscmide.exe -Q`	Starts Application Manager as a system tray icon application without showing the full Application Manager Interface. When this option is specified with bulk import options, Application Manager exits once import is complete.

-S"server_name[\instance_name]"
 	
`iscmide.exe`
`-S"mysql\sql1"`
`-U"admin"`
`-P"admin"`
`-D"mycatalog"`
`-IF"c:\mypackages\mymsi.msi;`
`c:\mypackages\mytrans.mst;"`
	The SQL Server where the Application Catalog resides. If this parameter is passed, the -F parameter is ignored.
	
	This parameter is only used for SQL Server Application Catalogs, and is a required parameter when using a SQL Server–based Application Catalog.

-U"login_id"
 	
`iscmide.exe`
`-S"mysql\sql1"`
`-U"admin"`
`-P"admin"`
`-D"mycatalog"`
`-IF"c:\mypackages\mymsi.msi;`
`c:\mypackages\mytrans.mst;"`
	The user name for the SQL Server Application Catalog
	
	This parameter is only used for SQL Server Application Catalogs, and is a required parameter when using a SQL Server–based Application Catalog. The only exception to this requirement is if you want to take advantage of SQL integrated security (Windows Authentication). In this case, do not use the -U parameter.

Specifying User Interface Mode via Command Line

There are three modes of operation for ConflictSolver: Application Manager, ConflictSolver, and ConflictSolver Process Assistant. The following Command Line options launch ConflictSolver in these modes of operation.

Table 6-103 • User Interface Mode Command Line Options

Option	Description
-A	Process Assistant Mode
-CS	ConflictSolver Mode
-APP	Application Manager Mode

Using a Configuration File

This section explains how to define a configuration file to use during the import process and how to use a configuration file with command-line options. The following topics are included:

- Application Manager Configuration File
- Using a Configuration File with Command-Line Options

Application Manager Configuration File

In addition to supporting individual command-line parameters, Application Manager can also use a configuration file (when specified using the `-C"configuration_file_name"` parameter) to perform a bulk import of packages into the Application Catalog. This INI file can contain the values for all required parameters during the import process.

- Configuration File Examples
- Parameter Explanation
- Deprecated Parameters

Important • *To accommodate the bulk import of App-V virtual packages using the Application Manager configuration file, the parameters for specifying the files to import were updated starting in AdminStudio 10.0. While configuration files written for previous versions of AdminStudio are still valid in AdminStudio 10.0 and its Service Packs, their use is deprecated because it is simpler to use the new parameters. For information on the deprecated parameters, see Deprecated Parameters.*

Configuration File Examples

The following configuration file examples are provided:

- Configuration File for Performing a Bulk Import
- Configuration File to Perform Bulk Import into an Application Catalog for a Named User
- Configuration File to Perform Bulk Import into an Application Catalog for a Trusted User
- Configuration File for Applying Transforms and/or Patches During Command-Line Import

Configuration File for Performing a Bulk Import

The following is an example of a configuration file for performing a bulk import:

```
[GENERAL]
DatabaseType=SQL
LogFile=D:\scratch\TestImport.txt
PackageFile=2

[SQL]
Server=ConflictSolverSQL2K
UserID=myAccountName
Password=myPassword
Database=myDBName

[PackageFile-1]
File=D:\scratch\xmlnotepad2007\XmlNotepad.msi
Group=RootGroup\SubGroup
Transform1=\\server\Data1.mst
EA1=Business Criticality;Business Critical
EA2=Installation Instructions;\\fileserver\docs\VirtualizationSuitabilityNotes.docx

[PackageFile-2]
File=D:\scratch\Test\Test.sft
Group=RootGroup\SubGroup
EA1=Business Criticality;Business Critical
EA2=Installation Instructions;\\fileserver\docs\AppVNotes.docx
```

Configuration File to Perform Bulk Import into an Application Catalog for a Named User

When writing a configuration file to perform bulk import into an Application Catalog for a named user, specify the `[SQL]` section as follows:

```
[SQL]
Server=ConflictSolverSQL2K
UserID=Admin
Password=mypassword
Database=AdminStudio
```

Configuration File to Perform Bulk Import into an Application Catalog for a Trusted User

When writing a configuration file to perform bulk import into an Application Catalog for a trusted user, specify the [SQL] section as follows:

```
[SQL]
Server=ConflictSolverSQL2K
Database=AdminStudio70
```

Configuration File for Applying Transforms and/or Patches During Command-Line Import

The following is an example of a configuration file that applies transforms and patches during a command-line import:

```
[General]
DatabaseType=SQL
LogFile=c:\temp\importlog.txt
PackageFile=4

[PackageFile-1]
File=\\server\Data1.MSI
Transform1=\\server\Data1a.MST
Transform2=\\server\Data1b.MST
Patch1=\\server\Data1p.MSP
Patch2=\\server\Data2p.MSP
AdminInstallLocation=\\Server\Shared\Data1

[PackageFile-2]
File=\\server\Data2.MSI
Patch1=\\server\Data1p.MSP
AdminInstallLocation=\\Server\Shared\Data2

[PackageFile-3]
File=\\server\Data3.MSI
Transform1=\\server\Data3a.MST
Patch1=\\server\Data1p.MSP
AdminInstallLocation=\\Server\Shared\Data3

[PackageFile-4]
SetupName=AdminStudio for Macintosh
SetupDirectory=C:\AdminStudio\MacFiles
FullDirectory=1
```

Note • *The [PackageFile-4] section of this example is specifying an other (non-MSI and non-SFT) setup type. See Importing Other Setup Types and About Other Setup Types.*

Parameter Explanation

This section provides a description of the parameters used in each section of the configuration file:

- [General] Section
- [SQL] Section
- [PackageFile-n] Section

[General] Section

The following table describes the parameters used in the `[General]` section of the configuration file.

Table 6-104 • Parameter Explanation: [General] Section

Parameter	Description
DatabaseType	Using this parameter, you can specify the Application Catalog database type. This value must be set to `SQL`. ***Note*** • *In previous releases, AdminStudio supported Microsoft Access and Oracle databases. However, these two database types are no longer supported.*
LogFile	Using this parameter, you can specify the name and location of the output log file. The LogFile parameter corresponds to the -L command-line parameter.
PackageFile	Use this parameter to indicate the total number of Windows Installer (`.msi`), merge module (`.msm`), App-V (`.sft`), and other setup type packages to be imported. Each package is denoted in subsequent INI file sections named `[PackageFile-n]`.

[SQL] Section

The following table describes the parameters used in the `[SQL]` section of the configuration file.

Table 6-105 • Parameter Explanation: [SQL] Section

Parameter	Description
Server	Using this required parameter, you can provide the name of the SQL Server. It corresponds to the -S command-line parameter.
UserID	Using this parameter, you can provide the login name for the SQL Server. It corresponds to the -U command-line parameter. This is required for non-trusted logins.
Password	Using this parameter, you can provide the password for the SQL Server. It corresponds to the -P command-line parameter. This is required for non-trusted logins.

Table 6-105 • Parameter Explanation: [SQL] Section

Parameter	Description
Database	Use the Database parameter to provide the catalog name for the SQL Server. It corresponds to the -D command-line parameter. If a value is not specified, the default SQL Server Application Catalog for the specified login will be used.

[PackageFile-n] Section

Each Windows Installer, merge module, App-V, or other setup type package to be imported into the Application Catalog must be described in its own section, numbered sequentially (`[PackageFile-1]`, `[PackageFile-2]`, etc.). Each section must contain the name and location of the file, and any transforms or patches to apply to the file prior to import.

Table 6-106 • Parameter Explanation: [PackageFile-n] Section

Parameter	Description
File	Use this parameter to specify the name and location of the file (`.msi`, `.msm`, `.sft`) you are importing. This parameter is required.
Group	You can use the Group parameter to specify the group into which the package should be imported. Use a "\" to indicate a group hierarchy. If no group is specified, the package is imported into the root group. It is possible to specify multiple groups by separating each group with a semicolon, such as: `Group=RootGroup\SubGroup;RootGroup\Subgroup2`
Transform*n*	Use this parameter to specify the name and location of a transform to apply to the Windows Installer package prior to import. Each subsequent transform increases the value of *n* (`Transform1`, `Transform2`, `Transform3`, etc.). ***Note*** • *This parameter is only applicable when importing Windows Installer packages.*

EAn	Use this parameter to specify extended attributes and values for the file. Each subsequent attribute increases the value of *n* (`EA1`, `EA2`, `EA3`, etc.). Specify extended attributes using the following syntax:
	`EA`*`n`*`=`*`AttributeName`*`;`*`AttributeValue`*
	For example:
	`EA1=Business Criticality;Business Critical`
	`EA2=Installation Instructions;\\schfiler\doc\VirtualizationNotes.docx`
	The attribute names should already be specified in the `EA_Default.xml file.`
	The `EAn` parameter is supported for Windows Installer (`.msi`) and App-V (`.sft`) packages.

Table 6-106 • Parameter Explanation: [PackageFile-n] Section

Parameter	Description
Patch*n*	Use this parameter to specify the name and location of a patch to apply to the Windows Installer package prior to import. Each subsequent patch increases the value of *n* (`Patch1`, `Patch2`, `Patch3`, etc.). ***Note*** • *This parameter is only applicable when importing Windows Installer packages.*
AdminInstallLocation	When applying a patch to an MSI package, it is necessary to perform an Administrative install of the MSI package and then perform an Administrative install of each patch package one by one. Use this parameter to specify the location where the Administrative install will be performed. ***Note*** • *This parameter is required if patches are specified.*
SetupName	Identifies the name of the imported setup package (an other setup type). ***Note*** • *Only valid when importing a file with an unsupported file extension. See About Other Setup Types.*
SetupDirectory	Identifies the location of the other setup type files. ***Note*** • *Only valid when importing a file with an unsupported file extension. See About Other Setup Types.*
FullDirectory	Specifies whether to import files in the selected directory and all subdirectories (1) or just the files in the selected directory (0). ***Note*** • *Only valid when importing a file with an unsupported file extension. See About Other Setup Types.*

Deprecated Parameters

To accommodate the bulk import of App-V virtual packages using the Application Manager configuration file, the parameters for specifying the files to import were updated starting in AdminStudio 10.0.

In versions of AdminStudio prior to version 10.0, you had to individually specify the number of MSI, MSM, and Other Setup Type files that you were going to import in the `[General]` section of the configuration file (using the `MSIFile=n`, `MSMFile=n` and `OtherSetupFile=n` parameters), and then specify the parameters for each package to import in its own section that started with `[MSIFile-n]`, `[MSMFile-n]` or `[OtherSetupFile-n]`.

In AdminStudio 10.0, the total number of packages (of any deployment type) to import can now be specified using one parameter in the `[General]` section: `PackageFile=n`. And the parameters for each package (of any deployment type) to import are specified in a separate section entitled `[PackageFile-n]`. For example, the following configuration file would import two files: one Windows Installer package and one App-V package:

```
[GENERAL]
DatabaseType=SQL
LogFile=D:\server10\ImportLog.txt
PackageFile=2
.
.
.
[PackageFile-1]
File=D:\server10\xmlnotepad2007\XmlNotepad.msi
Group=RootGroup\SubGroup1
Transform1=\\server\Data1.mst
EA1=Business Criticality;Business Critical
EA2=Installation Instructions;\\fileserver\docs\VirtualizationNotes.docx

[PackageFile-2]
File=D:\server10\packages\app2000.sft

Group=RootGroup\SubGroup2
```

While configuration files written for previous versions of AdminStudio are still valid in AdminStudio 10.0 and its Service Packs, their use is deprecated because it is simpler to use the new parameters. This section lists those deprecated parameters.

- Deprecated Parameters in the [General] Section
- [MSIFile-n] Section (Deprecated)
- [MSMFile-n] Section (Deprecated)
- [OtherSetupFile-n] Section (Deprecated)

Deprecated Parameters in the [General] Section

The `MSIFile=n`, `MSMFile=n`, and `OtherSetupFile=n` parameters in the `[General]` section of the Application Manager Configuration file have been deprecated in AdminStudio 10.0; they have been replaced with the `PackageFile=n` parameter. The following is an example of how these deprecated parameters were used:

```
[General]
DatabaseType=SQL
LogFile=c:\temp\importlog.txt
MSIFile=3
MSMFile=1
OtherSetupFile=1
```

The following table describes these parameters.

Table 6-107 • Deprecated Parameter Explanation: [General] Section

Parameter	Description
MSIFile	Use this parameter to indicate the number of MSI files to be imported. Each MSI file is denoted in subsequent INI file sections named `[MSIFile-n]`. See [MSIFile-n] Section (Deprecated).
MSMFile	Use this parameter to indicate the number of merge modules to be imported. Each merge module file is denoted in subsequent INI file sections named `[MSMFile-n]`. See [MSMFile-n] Section (Deprecated).
OtherSetupFile	Use this parameter to indicate the number of other setup type files to be imported. Each other setup type file is denoted in subsequent INI file sections named `[OtherSetupFile-n]`. See [OtherSetupFile-n] Section (Deprecated). ***Note*** • *See Importing Other Setup Types and About Other Setup Types.*

[MSIFile-n] Section (Deprecated)

Each Windows Installer package to be imported into the Application Catalog must be described in its own section, numbered sequentially (`[MSIFile-1]`, `[MSIFile-2]`, etc.). Each section must contain the name and location of the file, and any transforms or patches to apply to the file prior to import.

Table 6-108 • Parameter Explanation: [MSIFile-n] Section

Parameter	Description
File	Using the File parameter, you can specify name and location of the Windows Installer package (.msi) you are importing. This parameter is required.
Transform*n*	Use this parameter to specify the name and location of a transform to apply to the Windows Installer package prior to import. Each subsequent transform increases the value of *n* (Transform1, Transform2, Transform3, etc.).
Patch*n*	Use this parameter to specify the name and location of a patch to apply to the Windows Installer package prior to import. Each subsequent patch increases the value of *n* (Patch1, Patch2, Patch3, etc.).
AdminInstallLocation	When applying a patch to an MSI package, it is necessary to perform an Administrative install of the MSI package and then perform an Administrative install of each patch package one by one. Use this parameter to specify the location where the Administrative install will be performed. ***Note*** • *This parameter is required if patches are specified.*
Group	You can use the Group parameter to specify the group into which the package should be imported. Use a "\" to indicate a group hierarchy. If no group is specified, the package is imported into the root group. It is possible to specify multiple groups by separating each group with a semicolon, such as: `Group=RootGroup\SubGroup;RootGroup\Subgroup2`

[MSMFile-n] Section (Deprecated)

Each merge module to be imported into the Application Catalog must be described in its own section, numbered sequentially (`[MSMFile-1]`, `[MSMFile-2]`, etc.). Each section must contain the name and location of the file.

Table 6-109 • Parameter Explanation: [MSMFile-n] Section

Parameter	Description
File	Use this parameter to specify the name and location of the merge module (.msm) you are importing.

[OtherSetupFile-n] Section (Deprecated)

Each other setup type file to be imported into the Application Catalog must be described in its own section, numbered sequentially (`[OtherSetupFile-1]`, `[OtherSetupFile-2]`, etc.). Each section must contain the name and location of the file.

Table 6-110 • Parameter Explanation: [OtherSetupFile-n] Section

Parameter	Description
File	Use this parameter to specify the name and location of the other setup type file you are importing.
SetupName	Identifies the name of the imported setup package (an other setup type). ***Note*** • *Only valid when importing a file with an unsupported file extension. See About Other Setup Types.*
SetupDirectory	Identifies the location of the other setup type files. ***Note*** • *Only valid when importing a file with an unsupported file extension. See About Other Setup Types.*
FullDirectory	Specifies whether to import files in the selected directory and all subdirectories (1) or just the files in the selected directory (0). ***Note*** • *Only valid when importing a file with an unsupported file extension. See About Other Setup Types.*

Using a Configuration File with Command-Line Options

Application Manager supports use of a configuration file to pass parameters during command-line import. This is extremely useful if you are importing multiple packages and/or merge modules simultaneously. Use the following command line to use a configuration file:

```
iscmide.exe -C"configuration_file_name"
```

Replace **configuration_file_name** with the name and location of the configuration file to use.

Importing

This section includes topics on how to perform import operations from the command line. The following topics are included:

- Applying Transforms and Patches During Command-Line Import
- Importing Multiple Windows Installer Packages Simultaneously
- Importing Multiple Merge Modules Simultaneously
- Simultaneously Importing Windows Installer Packages and Merge Modules
- Using the Command Line to Import All Packages in a Directory
- Running Import Silently
- Creating a Log File During Command-Line Import

Applying Transforms and Patches During Command-Line Import

Application Manager provides two mechanisms for applying transforms or patches to a Windows Installer package during import. First, if you are only importing a single package into Application Manager from the command line, you can pass transforms or patches after the file name, as in the following command lines:

Table 6-111 • Applying Transforms and Patches During Import

Function	Command
TRANSFORM:	iscmide.exe -IF"msi_file;[mst1;mst2;]"
PATCH:	iscmide.exe -IF"msi_file;[msp1;msp2;]"
BOTH:	iscmide.exe -IF"msi_file;[mst1;msp1;]"

Replace **msi_file** with the name and location of the Windows Installer package, and **mst1** and **mst2** or msp1 and **msp2** with the names and locations of the transforms or patches to apply. For example, if your MSI file was named `data1.msi` and your two MST files were named `alpha.mst` and `gamma.mst`, your command line would look like the following:

```
iscmide.exe -IF"data1.msi;alpha.mst;gamma.mst;
```

Note • *Depending on the situation, it may be necessary to pass additional command-line parameters.*

If you are importing multiple Windows Installer packages, and applying transforms or patches to them, use a configuration file in which you can specify the names and locations of the packages and associated transforms or patches. Use the following command line to use a configuration file:

```
iscmide.exe -C"configuration_file_name"
```

Replace **configuration_file_name** with the name and location of the configuration file to use.

Importing Multiple Windows Installer Packages Simultaneously

The best way of importing multiple Windows Installer packages simultaneously into an Application Catalog is via a configuration file. Use the following command line to call your configuration file, in which you can specify multiple Windows Installer packages to import.

```
iscmide.exe -C"configuration_file"
```

Replace **configuration_file** with the name and location of the configuration file containing the names and locations of the packages you want to import.

Importing Multiple Merge Modules Simultaneously

The best way of importing multiple merge modules simultaneously into an Application Catalog is via a configuration file. Use the following command line to call your configuration file, in which you can specify multiple merge modules to import.

```
iscmide.exe -C"configuration_file"
```

Replace **configuration_file** with the name and location of the configuration file containing the names and locations of the merge modules you want to import.

Simultaneously Importing Windows Installer Packages and Merge Modules

To simultaneously import both Windows Installer packages and merge modules into an Application Catalog, use a configuration file from the command line. In this configuration file, you can specify the names and locations of both merge modules and Windows Installer packages you want to import.

Use the following command-line to specify your configuration file:

```
iscmide.exe -C"configuration_file_name"
```

Replace **configuration_file_name** with the name and location of the configuration file to use.

Using the Command Line to Import All Packages in a Directory

It is possible from the command line to import all packages in a directory. This can be done by placing the following line in a batch file (modifying it as necessary for your specific environment):

```
for %%a in (*.msi) do ISCMIDE
   -S"server_name"
   -U"userid"
   -P"password"
   -D"Application_Catalog_Name"
   -IF"%%a"
   -Q
   -L"%%aLog.txt"
```

The above statement runs Application Manager, importing each MSI package in the Application Catalog, and creates a log file with the name of the `.msi` file prepended to the log file name.

However, the above command makes the following assumptions:

- `ISCMIDE.exe` is in the path; otherwise, include the full path to `ISCMIDE.exe`.
- All packages are being imported into the specified SQL Server Application Catalog (provide the server name, userid, password, and Application Catalog name with the appropriate parameters `-S`,`-U`,`-P`, and `-D`, respectively).
- The above command will start Application Manager, import a file, exit Application Manager, and then restart Application Manager for each MSI file. If this is not acceptable, use a configuration file instead.
- No transforms are applied to the imported packages. If you need to apply transforms, use a configuration file instead.
- Application Manager starts in quiet mode. If it is switched to full mode, it is your responsibility to end the Application Manager process so the next file can be imported.

Running Import Silently

Application Manager can be run silently in system tray mode by using the following command line:

```
iscmide.exe -Q
```

An icon for Application Manager, with a corresponding context menu available by right-clicking on the icon, appears in the system tray. If you pass the -Q parameter in addition to other parameters, progress is displayed in the tool tip available when you mouse over the icon. This is beneficial if you are importing a sizeable amount of packages, yet want to monitor the progress periodically.

In quiet mode, when import from the command line ceases, Application Manager automatically exits.

Creating a Log File During Command-Line Import

It may be necessary or beneficial to create a log file during package/merge module import from the command line. You may want to see the results of the import, or determine why certain packages were not processed as expected. To create a log file, use the following command-line:

```
iscmide.exe -L"log_file_name"
```

Replace `log_file_name` with the name and location of the log file you want to create.

Note • *It may be necessary to pass additional command-line parameters, depending on the task you are performing from the command line.*

Connecting to Standalone Application Catalogs

This section includes the following topics:

- Connecting to a Specific Standalone Application Catalog Using Command-Line Options
- Creating Shortcuts to Specific Standalone Application Catalogs

Connecting to a Specific Standalone Application Catalog Using Command-Line Options

Using the following command line, you can connect to a specific SQL Server Application Catalog:

```
iscmide.exe -S"sql_server_name" -U"user_id" -P"password"
```

Replace **sql_server_name** with the specific SQL Server name (including path information). Replace **user_id** and **password** with the UserID and password for the server.

Note • *Depending on the Application Catalog configuration, it may be necessary to pass additional command-line parameters, such as the Application Catalog name.*

Creating Shortcuts to Specific Standalone Application Catalogs

For convenience, you may want to create a Windows shortcut to launch ConflictSolver and automatically connect to a specific Application Catalog.

Task: ***To create the shortcut:***

1. On the Windows Desktop, right-click, point to New, and select Shortcut. The Create Shortcut Wizard appears.
2. Click the Browse button and navigate to the ConflictSolver executable (`ISCMIDE.exe`). Typically, it is in the *`AdminStudio Installation Directory`*`\ConflictSolver` directory. When you have located it, click OK and then click Next.
3. Type a name for the shortcut and click Finish. The new shortcut appears on the Desktop.
4. Right-click on the shortcut and select Properties.
5. In the **Properties** dialog box, append the necessary command-line parameters to the end of the target, such as:

   ```
   iscmide.exe -S"sql_server_name" -U"user_id" -P"password"
   ```

6. Click OK to dismiss the dialog box.

Note • *If you are creating a shortcut to a Microsoft SQL Server Application Catalog, you need to provide values for the SQL Server name, UserID, password, and Application Catalog name using the -S or -O, -U, -P, and -D parameters. See Application Manager Command-Line Functionality or Connecting to a Specific Standalone Application Catalog Using Command-Line Options.*

Application Catalog Replication Command Line Functionality

Note • *These options are provided in case you would like to run Application Catalog Replication using the command line. However, this functionality can be invoked from the main AdminStudio interface.*

The following Application Catalog Replication command-line options are supported by AdminStudio:

Table 6-112 • Replication Command Line Options

Option	Description
-c"Connection String"	Optional, if not present we attempt to connect to the default database. Otherwise, we will prompt the user for a connection of an existing database.
-q	Silent launch without splash dialog box.
-h[number]	Parent window handle. (Optional)

Virtualization Suitability Issues

AdminStudio uses the following set of tests to determine if a Windows Installer package is a suitable candidate for virtualization to Microsoft App-V, VMware ThinApp, or Citrix XenApp formats.

Table 6-113 • Virtualization Suitability Issues

Rule	Description
Generic	Package is not a valid Windows Installer package. Usually this issue is found in legacy installations that require repackaging. In rare cases, it could also be found in a Windows Installer (`.msi`) package which has become corrupted. However since the contents are not known, this package has not been analyzed for virtualization suitability.
No Shortcut	Package contains no shortcuts. • **App-V suitability**—For conversion to App-V packages, this issue is acceptable in some scenarios, such as packages which provide dependencies to others which dynamically suite it. However, if this package merely provides a plug-in to another application, it must contain a shortcut to launch that application in this package's virtual context. • **ThinApp and XenApp suitability**—For conversion to ThinApp and XenApp formats, shortcuts are necessary to define the entry point into the virtual application. One potential resolution to this issue is to use InstallShield Editor to add shortcut(s) to the Windows Installer package.
ClickOnce	Package contains a ClickOnce application. ClickOnce is a per-user installation format which is often incompatible with the per-machine nature of virtual package deployment. A ClickOnce application also may try to automatically update itself, which results in invalid versioning in the application virtualization client.
Shell Extension	Package contains a shell extension. Shell extensions extend Windows Explorer and cannot be loaded from a virtual package. This extension may be critical to the use of this application, and, if so, this application will not function when virtualized. However, if this extension is non-critical, this application may function when virtualized.
OS Integrated	This package contains files that are closely integrated with the operating system. The files that make up applications like Internet Explorer or Windows Media Player, or frameworks like the .NET Framework, do not make good candidates for virtualization. These files should instead be installed locally on the machine.

Table 6-113 • Virtualization Suitability Issues

Rule	Description
Boot Service	This package contains a service which starts at boot-time. Virtualized services are limited to the lifetime of the virtual application, so services that must start at boot time do not make good candidates for virtualization to App-V or XenApp formats. It may be possible to extract this service such that it can be installed locally on the machine and allow the rest of the package to be virtualized. ***Note*** • *Boot time services do not cause a problem when converting a package to a ThinApp application.*
Too Large	This package contains more than 4 GB of files. App-V and XenApp do not support packages containing more than 4 GB of files. This application cannot be successfully virtualized. ***Note*** • *ThinApp supports packages containing more than 4 GB of files.*
COM Surrogate DLLs	This package contains a COM DLL using surrogate virtualization. App-V, XenApp, and ThinApp do not support COM DLL surrogate virtualization, so this package may not work correctly if virtualized.
COM Plus	This package contains a COM Plus component. App-V, XenApp, and ThinApp do not support COM Plus components, so this package may not work correctly if virtualized.
Device Driver	This package contains a device driver. System-level drivers such as print drivers or USB device drivers do not work from a virtualized environment. It may be possible to extract this driver such that it can be installed locally on the machine and allow the rest of the package to be virtualized.
64-bit Package	This is a 64-bit package. XenApp and ThinApp do not support virtualization of 64-bit packages. ***Note*** • *App-V supports virtualization of 64-bit packages.*

Part 3

Repackaging and Customizing Installations

This part of the AdminStudio 10.0 SP2 User Guide includes the following chapters:

- Repackaging Legacy Installations Using the Repackaging Wizard
- Converting Legacy Installations Using the Repackager Interface
- Performing Automated Repackaging and Virtualization Using the Automated Application Converter
- Using the Virtual Package Editor
- Creating Customized Virtual Applications
- Customizing and Authoring Installations Using InstallShield
- Customizing Installations Using Tuner

7

Repackaging Legacy Installations Using the Repackaging Wizard

Installations created for the Windows Installer service dramatically differ from traditional installations, making reusing legacy installations impossible without using a repackaging tool. You can use Repackager's Repackaging Wizard to capture the data placed on your system during installation and convert it into a Windows Installer (`.msi`) package, which you can then customize and distribute according to your organization's needs.

Documentation regarding using the Repackaging Wizard is presented in the following sections:

Table 7-1 • Using the Repackaging Wizard

Section	Description
About Repackaging	Introduces you to repackaging, explains various repackaging methods, lists Repackaging Best Practices, explains how to include the InstallScript Engine with a Windows installer package, and reviews Repackager options.
Repackaging Methods	Describes the methods of repackaging that the Repackaging Wizard supports.
Configuring Repackager to Ensure Optimal Installation Capture	Describes how to configure Repackager in order to get optimal results when capturing an installation.
Repackaging Legacy Installations Using the Repackaging Wizard	Explains how to use the Repackaging Wizard to convert the following installations: • InstallShield Professional 1.x to 5.1.x • InstallShield Professional 5.5 to 7.x • InstallShield InstallScript MSI • InstallShield DevStudio 9.x InstallScript • InstallShield Editor InstallScript

Table 7-1 • Using the Repackaging Wizard (cont.)

Section	Description
Repackaging Wizard Reference	Describes each of the dialog boxes and Wizard panels that you might encounter when using the Repackaging Wizard.

Note • *For information on other Repackager features, see Converting Legacy Installations Using the Repackager Interface.*

About Repackaging

This section introduces you to repackaging, lists Repackaging Best Practices, and explains how to set Repackager options.

- Purpose of Repackaging Applications
- Supported Legacy Installation Types
- Repackaging 64-Bit Applications
- Repackaging Options Comparison
- Repackaging Wizard Best Practices
- About Repackaging on Clean Systems
- Including the InstallScript Engine With a Windows Installer Package

Purpose of Repackaging Applications

Installations created for the Windows Installer service dramatically differ from traditional installations, making reusing legacy installations impossible without using a repackaging tool. Repackager assists you by capturing the data placed on your system during installation and converting it into a Windows Installer (`.msi`) package, which you can then customize and distribute according to your organization's needs.

Repackaging an installation into a Windows Installer package provides the following benefits:

- **Can customize it using InstallShield Editor or Tuner**—You can further configure or customize the Windows Installer package to meet your specific needs by editing the `.msi` file in InstallShield Editor or by creating transforms in InstallShield Editor or Tuner.
- **Can perform conflict analysis and resolution**—You can use ConflictSolver to check the Windows Installer package for conflicts that may exist between it and other known Windows Installer packages in an Application Catalog database, ensuring the proper installation and functioning of your installations.

- **Can implement application repair and feature advertising**—Finally, once converted to a Windows Installer package, the installation can take advantage of Windows Installer functionality such as application repair and feature advertisement.

Supported Legacy Installation Types

You can use both the Repackaging Wizard and the Repackager interface to create Repackager projects. The tool that you use depends upon the type of installation you are converting:

Table 7-2 • Methods of Creating Repackager Projects

Tool	Installation Source
Repackaging Wizard	You can use the Repackaging Wizard to convert the following installations: • InstallShield Professional 1.x to 5.1.x • InstallShield Professional 5.5 to 7.x • InstallShield InstallScript MSI • InstallShield DevStudio 9.x InstallScript • InstallShield Editor InstallScript See Repackaging Legacy Installations Using the Repackaging Wizard.
Repackager Interface	You can use the Repackager interface to convert the following installations: • Repackager 3.x output (.inc) • Microsoft SMS projects (.ipf) • Novell ZENworks 3.0, 3.1, or 4.0 projects (.axt/.aot) • WinINSTALL projects (.txt) (6.0, 6.5, 7.x) • Wise installation projects (.wse) • InstallShield Professional log files (.isl) See Converting Legacy Installations Using the Repackager Interface. ***Edition*** •

Repackaging 64-Bit Applications

Repackager has the capability to repackage both 32-bit and 64-bit applications, as well as hybrid applications (both 32-bit and 64-bit). The Repackaging Wizard remains a 32-bit application, but can be run on both 32-bit (x86) and 64-bit (x64) Windows operating systems. The following table lists the operating systems to use to repackage both 32-bit and 64-bit applications, and the operating systems those repackaged applications will run on.

Table 7-3 • Repackaging 32-Bit and 64-Bit Applications

Application Type	Repackage on ...	Will run on ...
64-bit application	Windows 64-bit OS ***Note*** • *You can use either the* ***Installation Monitoring*** *or* ***Snapshot*** *method to repackage a 64-bit application on a 64-bit operating system.*	Windows 64-bit OS
32-bit application	Windows 32-bit OS ***Important*** • *While it is possible to repackage a 32-bit application on a 64-bit OS, it is recommended that you use a 32-bit OS, to avoid inadvertently capturing any 64-bit data. If Repackager captures any 64-bit data, it will flag the package as a 64-bit application, meaning that it will only run on a 64-bit OS. See Excluding 64-Bit Data.*	Windows 64-bit OS or Windows 32-bit OS

Excluding 64-Bit Data

It is strongly recommended that you repackage 32-bit applications on a 32-bit OS. However, if you choose to repackage a 32-bit application on a 64-bit OS, you need to make sure that you exclude any unnecessary 64-bit data, such as data from a 64-bit Windows Service that could be running or 64-bit files (stored in the `System64Folder`, `ProgramFiles64Folder`, or `CommonFiles64Folder` directories) or 64-bit registry entries (any entries stored in a node other than `WOW6432Node`).

Repackaging Options Comparison

The following table details the different options available to you when using Repackager, based upon source type, product and version:

Table 7-4 • Repackaging Options Comparison Chart

Source	Product / Version	Repackaging Method	Result
Media	IS Professional 1.x to 5.1.x	**Repackaging Wizard** Installation Monitoring **or** Snapshot	Repackager project with no feature delineation
	IS Professional 5.5 to 7.x	**Repackaging Wizard** Installation Monitoring **or** Snapshot **and** SmartScan Wizard	Repackager project with feature delineation, including registry entries and shortcuts
	IS InstallScript MSI	**Repackaging Wizard** Installation Monitoring **or** Single Step Snapshot	Repackager project with feature delineation, including registry entries and shortcuts
	IS Editor InstallScript IS DevStudio 9.*x* InstallScript	**Repackaging Wizard** InstallShield Professional Logging Method	Repackager project with feature delineation, including registry entries and shortcuts. Also captures path variables and converts them into properties (using text substitution rather than hard-coded path names).
Project	Repackager 3.x output (`.inc`) Microsoft SMS projects (`.ipf`) Novell ZENworks 3.0, 3.1, or 4.0 projects (`.axt/.aot`) WinINSTALL projects (`.txt`) (6.0, 6.5, 7.x) Wise installation projects (`.wse`)	**Repackager Interface** Select **Open** on the **File** menu to have Repackager automatically convert file to a Repackager project	Repackager project with no feature delineation
	InstallShield Editor Pro log files (`.isl`)	**Repackager Interface** Select **Open** on the **File** menu to have Repackager automatically convert file to a Repackager project	Repackager project with feature delineation

Once you have created a Repackager project, you can visually examine the files, `.ini` files, shortcuts, and registry data from the installation, and exclude any non-essential items. Then, you can build the Repackager project into an InstallShield Editor project (`.ism`) for further editing, or create a Windows Installer package (`.msi`).

Repackaging Wizard Best Practices

To ensure optimal performance of the Repackaging Wizard during repackaging and when working with Repackager projects, the following best practices are recommended:

- Repackage on a Clean System
- Launch Repackager Remotely or Install Repackager on the Clean Machine
- Use the Repackager Interface to Exclude Unwanted Items
- Exit All Other Applications
- Only Repackage Non-Windows Installer Setups

Repackage on a Clean System

It is essential that you repackage applications on a "clean" system to ensure you capture all changes made by the installation. A clean system typically consists of a computer with only the operating system and necessary service packs installed on it. Repackaging on a clean system provides the following benefits:

- **Prevents you from capturing Repackager files**—By repackaging on a clean system, you are ensuring that you do not inadvertently capture Repackager files during repackaging.
- **Ensures that you capture all of the necessary setup files**—If you do not repackage on a clean system, you may not capture all of the necessary files for the setup because the files may already be installed on the system.

Note • *For more information, see About Repackaging on Clean Systems.*

Launch Repackager Remotely or Install Repackager on the Clean Machine

Because it is best to keep the number of packages installed on the clean machine to a minimum, you should launch Repackager remotely from the clean machine or install Repackager on the clean machine:

- **Launch Repackager Remotely**—You could install Repackager on a shared network drive and then launch Repackager remotely from the clean machine. See Launching Repackager Remotely.
- **Install Repackager on clean machine**—You could install a copy of Repackager onto the clean machine. While it is preferable to launch Repackager remotely from the clean machine, if you do not have network access to an installation of the AdminStudio client tools, this is your best option. See Installing Repackager on a Clean Machine.

Both of these options are explained in Configuring Repackager to Ensure Optimal Installation Capture.

Use the Repackager Interface to Exclude Unwanted Items

You should repackage using the provided exclusions and then use the Repackager interface to visually remove unwanted items from the capture.

Because this occurs post-capture, you do not need to recapture the legacy setup if you inadvertently exclude items from the Windows Installer package you are building.

Note • *Since Windows Installer does not support packaging device drivers, you would need to create Custom Actions to install device drivers. See Using Custom Actions in the Windows Installer help section for more information.*

Exit All Other Applications

Other applications may lock files or directories, and may hinder the performance of the setup and repackaging. Therefore, exit all applications prior to repackaging.

Only Repackage Non-Windows Installer Setups

Windows Installer setups should not be repackaged. They should either be edited in InstallShield Editor, or, as Microsoft recommends, by creating a transform. This can be done using InstallShield Editor or Tuner.

You should not repackage Windows Installer (`.msi`) packages for the following reasons:

- Repackaging a Windows Installer package is against Microsoft Best Practices.
- If you make changes to a Windows Installer package, vendors will no longer provide support for that product.
- If you repackage a Windows Installer package, the component codes within the package are not retained and hence future patching or upgrades will not work.
- Traditionally, repackaging tools will ignore the Windows Installer-specific data in the Registry. This will result in an incomplete package.

Also, Repackager is not intended for repackaging operating system installations or service packs, or deeply integrated operating system components such as Internet Explorer. Moreover, components such as MDAC or DCOM should be included in the clean image, or installed by a setup using the vendor's redistributable.

Exception to This Rule

In general, due to the reasons listed above, it is not recommended to repackage a vendor-created Windows Installer package to create a new Windows Installer package. However, some IT organizations may elect to repackage Windows Installer packages in order to simplify them, which should make them more reliable and less likely to violate the organization's and Microsoft's recommended best practices.

If you choose to repackage a Windows Installer package, you need to keep in mind that you may no longer be able to:

- Directly deploy vendor-provided patches for this package, OR
- Use any vendor-provided automatic updating service for this package.

Therefore, you should only consider repackaging a Windows Installer package if your IT staff is also willing to invest resources into periodically repackaging that application's vendor patches into an updated Windows Installer package.

Note • *Tightly-controlled organizations probably would not want to have automatically-updating software, so the inability to use an automatic updating service may not be of concern to them.*

About Repackaging on Clean Systems

For optimal results when using Repackager or OS Snapshot, you should perform these processes on a clean system. A clean system typically consists of a computer with only the operating system and necessary service packs installed on it. It is the baseline system that the computer requires to run.

Although it may be tempting to consider basic software, such as Microsoft Office, as part of the clean system, this can result in poor snapshots and repackaged setups. Each application you install on the baseline system adds to the DLLs, changes versions of files, makes new registry entries, etc. This may cause Repackager or the OS Snapshot Wizard to miss these during capture, which ultimately may lead to missing files or registry entries in repackaged setups, or unexpected conflicts between the operating system and Windows Installer packages.

Note • *For more information on setting up a clean system to repackage on, see Configuring Repackager to Ensure Optimal Installation Capture.*

Alternate-Language Repackaging on Clean Machines

The standalone Repackager setup for clean machines does not install any language resources other than US English. Therefore, if you are Repackaging a setup on a clean system in a language other than in US English, you need to ensure you point to the correct template in the Repackaged Output View.

This can be on a mapped network drive, or you can copy the language-populated template (for example, `ISProjBlankTpl.ism`) to your clean system from the *AdminStudio Installation Directory*`\Editor\Support` directory.

Language-specific templates are available when you purchase InstallShield Editor Language Packs.

Including the InstallScript Engine With a Windows Installer Package

Should you need to include the InstallScript engine with your setup, all the major releases of the InstallScript engine are available in the `InstallScript_Engines` folder on the AdminStudio installation CD. For more information, see the *Update to the Latest InstallShield Installation Engines* Knowledge Base article at:

http://consumer.installshield.com/kb.asp?id=Q108322

Repackaging Methods

Repackager supports three methods of repackaging:

Table 7-5 • Repackaging Methods

Repackaging Method	Description
Installation Monitoring Method	Repackager monitors system changes as an application is installed, and that data can be converted into a Windows Installer package. Installation Monitoring is the default method. ***Edition*** •
Snapshot Method	Repackager compares a system snapshot before and after an installation, determines the changes that were made, and that data can be converted to a Windows Installer package.
InstallShield Professional Logging Method	Repackager reads logged output of InstallShield Editor and DevStudio 9.*x* InstallScript installations to obtain additional information that would not be captured by repackaging, such as path variables and a feature tree.

Installation Monitoring Method

When using the **Installation Monitoring** method, the Repackaging Wizard monitors a system for any processes that are created during an installation. The Installation Monitoring method determines the dynamic interdependencies between files.

By monitoring these processes in the background, the Repackaging Wizard can identify files, shell links, and registry entries that are added, modified, or removed by the installation. The resulting files and Repackager output file can be converted into a Windows Installer package.

Installation Monitoring Method Considerations

Consider the following about Installation Monitoring when selecting a repackaging method:

- **Faster than Snapshot**—Installation Monitoring is significantly faster than the Snapshot repackaging method.
- **Clean system not required**—Although it is still a good practice to repackage on a clean system, it is not as important when using Installation Monitoring technology as it is when you use the Snapshot method.
- **Can exclude processes from the project**—When using the Installation Monitoring method, you can specify the processes that you want to exclude from the Installation Monitoring.
- **Enhanced system reboot handling**—On Windows Vista and newer, system reboots are almost instantaneous and do not allow running applications to properly shut down, which may result in a loss of data.

When using the Installation Monitoring method, Repackager successfully handles a system reboot and delays it until you click the Reboot button on the Repackaging Wizard.

- **Windows side-by-side support**—The Repackager Installation Monitoring method scans and detects changes made to the Windows SxS (Side-by-Side) store and automatically includes the proper merge modules.

Snapshot Method

When using the **Snapshot** method, the Repackaging Wizard takes a reference snapshot of a system as a baseline configuration, performs the installation, and then takes a second snapshot.

The difference between the two snapshots is stored in a directory you specify, along with the Repackager output file (`.inc`). This file can then be converted into a Windows Installer package (`.msi`) using Repackager.

Snapshot Method Considerations

Consider the following about Snapshot technology when selecting a repackaging method:

- **Slower than Installation Monitoring**—The Snapshot method is significantly slower than the Installation Monitoring repackaging method.
- **Clean system is required**—When repackaging using Snapshot technology, you should use a clean system, with a baseline configuration for your target environment. If you do not repackage on a clean system, you may not capture all of the necessary files for the setup because the files may already be installed on the system.
- **Exclude anti-virus software directories**—Any machine that you use to repackage most likely has anti-virus software installed on it, even a “clean” machine. While you are repackaging an application, the real-time virus detection feature of anti-virus software could automatically update various cached files in its directories. In order to avoid repackaging errors when using the **Snapshot** method, you should exclude these directories. See Excluding Directories and Subdirectories for more information.

Note • *Anti-virus software does not affect repackaging using the **Installation Monitoring** method.*

InstallShield Professional Logging Method

Note • *The InstallShield Professional Logging Method is supported for InstallShield Editor and DevStudio 9.x InstallScript installations.*

Using the **InstallShield Professional Logging Method**, Repackager can read logged output of InstallShield Editor and DevStudio 9.x InstallScript installations. This enables Repackager to capture additional information from those installations that would not be captured by the **Installation Monitoring** or **Snapshot** methods: path variables and the feature tree.

Path Variables

When using the InstallShield Professional Logging Method, path variables are captured and converted into properties (using text substitution rather than hard-coded path names).

For example, if you were using a standard repackaging method, the path name of the application's executable file would be captured as follows:

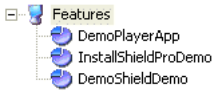
```
C:\Program Files\CompanyName\ApplicationName\ProgramName.exe
```

When using the InstallShield Professional Logging Method, the executable path name would be captured using a path variable instead of a hard-coded path name:

```
[INSTALLDIR]\ProgramName.exe
```

Feature Tree

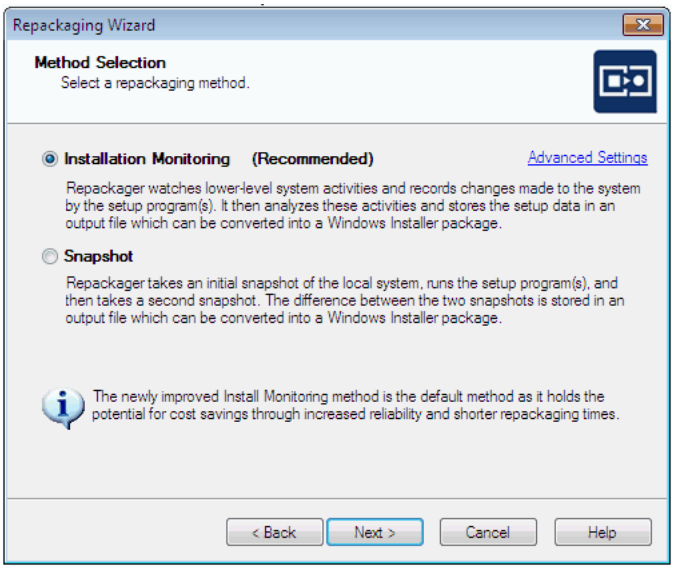
The InstallShield Professional Logging Method is able to group files, shortcuts and registry entries into features corresponding to InstallShield Professional components. Any items (files, folders, shortcuts, or registry entries) that are attached to a component directly or indirectly (through File Groups), are attached to corresponding features.



If you do not use the InstallShield Professional Logging Method to repackage an InstallShield Editor or DevStudio 9.x InstallScript installation, all files, shortcuts and registry entries would be installed together as one feature.

Selecting the InstallShield Professional Logging Method

You access the InstallShield Professional Logging Method through the Repackaging Wizard, but **InstallShield Professional Logging Method** is *not* offered as a choice on the **Method Selection Panel** of the Repackaging Wizard. Only **Installation Monitoring** and **Snapshot** are listed:



When you are repackaging an InstallShield Editor or DevStudio 9.x InstallScript installation and want to use the InstallShield Professional Logging Method, you can select either **Installation Monitoring** or **Snapshot** on the **Method Selection Panel**. However, if you select the **Snapshot** method, you must select **Single Step** on the **Snapshot Method Panel** (which would open next).

Caution • *If you select the Snapshot / Multiple Steps repackaging method, Repackager will not recognize the setup as an InstallShield Editor or DevStudio 9.x InstallScript installation and the InstallShield Professional Setup Panel will not open.*

After you select the **Installation Monitoring** or **Snapshot** method, and then specify an InstallShield Editor or DevStudio 9.x InstallScript installation on the Repackaging Wizard's Collect Product Information Panel, Repackager will automatically detect the InstallScript installation and will display the InstallShield Professional Setup Panel. On this panel, you can choose to use the **InstallShield Professional Logging Method** instead of the repackaging method you chose on the Method Selection Panel.

For instructions on how to use the InstallShield Professional Logging Method when repackaging InstallShield Editor or DevStudio 9.x InstallScript installations, see Repackaging Using the InstallShield Professional Logging Method and InstallShield Professional Setup Panel.

Configuring Repackager to Ensure Optimal Installation Capture

Both repackaging methods, **Installation Monitoring** and **Snapshot**, involve installing an application and recording the system changes made by that installation. To ensure that you capture *all* changes made by the installation, you should, ideally, install the application onto a "clean machine" (a computer with only the operating system installed), as described in About Repackaging on Clean Systems.

Depending upon your network connectivity, you should configure Repackager on a clean machine in one of the following ways:

Table 7-6 • Methods to Configure Repackager

Repackager Configuration	Description
Launching Repackager Remotely	If you have connectivity from a clean machine to a computer or network location that contains an installation of Repackager, you should launch Repackager remotely.
Installing Repackager on a Clean Machine	If you do not have any network connectivity on the clean machine, you should install Repackager on the clean machine.

***Note** • You cannot repackage remotely from a Windows NT4 system because a file required during repackaging on that operating system, `MSCVRT.dll`, must be installed locally. Therefore, when repackaging on a Windows NT4 system, install Standalone Repackager on that machine, as described in Installing Repackager on a Clean Machine.*

Launching Repackager Remotely

Because you want to avoid installing applications on the clean machine, you should launch Repackager remotely from the clean machine.

To launch Repackager remotely, perform the following tasks:

- Sharing Directories on a Machine with an Installation of AdminStudio
- Run the Remote Repackager Setup on the Clean Machine
- Launching Remote Repackager on the Clean Machine

Sharing Directories on a Machine with an Installation of AdminStudio

To share directories on a machine where AdminStudio is installed, perform the following steps:

Task: ***To share the Repackager and AdminStudio Shared folders:***

1. Locate a production machine with network access that has AdminStudio installed on it.

   ***Tip** • Check to make sure that this installation of AdminStudio has already been activated before proceeding.*

2. Open Windows Explorer and locate the following directory:

   `[ADMINSTUDIO INSTALLATION DIRECTORY]\Repackager`

3. Right-click the `Repackager` directory and select **Properties** from the context menu. The **Repackager Properties** dialog box opens.
4. Open the **Sharing** tab and turn on sharing for this folder; configure the sharing rights as necessary.
5. Repeat the steps above to also share the `AdminStudio Shared` directory used by that installation of AdminStudio.

Run the Remote Repackager Setup on the Clean Machine

To run the Remote Repackager setup on the clean machine, perform the following steps:

Task: ***To run the Remote Repackager setup on the clean machine:***

1. On this clean machine with network access, launch Windows Explorer.
2. Select **Map Network Drive** from the **Tools** menu. The **Map Network Drive** dialog box opens.
3. Specify the drive letter you want use to represent the shared location.
4. Click **Browse**. The **Browse for Folder** dialog box opens.
5. Select the shared `Repackager` directory on the production machine (that you configured in Sharing Directories on a Machine with an Installation of AdminStudio) and click OK.
6. Click **Finish** to exit the **Map Network Drive** dialog box.
7. Repeat the steps above to also map the `AdminStudio Shared` directory on the same production machine (that you configured in Sharing Directories on a Machine with an Installation of AdminStudio).
8. From Windows Explorer, navigate to the drive mapped to the shared `Repackager` directory on the production machine and open the `Remote Repackager` folder.
9. Run the `Setup.exe` file in the `Remote Repackager` folder to install Remote Repackager on the clean machine.

Launching Remote Repackager on the Clean Machine

To launch Repackager on the clean machine, perform the following steps:

Task: ***To launch Remote Repackager on the clean machine:***

1. On the clean machine, double-click the **Repackager** shortcut on the desktop (that you created in Run the Remote Repackager Setup on the Clean Machine). The Repackager Start Page opens.

   Important • *Because you are running Repackager remotely, the online help topics cannot be viewed. However, you can view a version of AdminStudio Help Library online at:*

   `http://helpnet.flexerasoftware.com`

2. Click the **Capture an Installation Using Repackaging Wizard** link. The **Welcome** panel of the **Repackaging Wizard** opens.
3. Continue using the Repackaging Wizard to capture a legacy setup, following the instructions in Repackaging Legacy Installations Using the Repackaging Wizard.

   Caution • *On the **Set Target Project Information and Capture Settings** panel of the Repackaging Wizard, do not set the **Project path to store files** field to a location on the clean machine; instead choose a network location.*

Installing Repackager on a Clean Machine

It is essential that you repackage applications on a “clean" system to ensure you capture all changes made by the installation. A clean system typically consists of a computer with only the operating system and necessary service packs installed on it. It is the baseline system that the computer requires to run.

While you want to avoid installing applications on the clean machine, if the clean machine does not have network connectivity to an installation of Repackager (which is required in order to run Repackager remotely), you have to install Repackager locally on a clean machine by running the Repackager installation.

To install a standalone version of Repackager on a clean machine, perform the following steps.

Task: ***To install Repackager on a clean machine:***

1. Build a “clean machine”—a computer with only the operating system and necessary service packs installed on it.
2. On the AdminStudio installation DVD, locate the standalone Repackager installation in the `Tools` directory:

   `\Tools\StandaloneRepackager.exe`
3. Copy `StandaloneRepackager.exe` to the clean machine.
4. Launch the setup. The **Welcome Panel** opens.
5. Click **Next**. The **License Agreement** panel opens.
6. Select the **I accept the terms of the license agreement** option and click **Next**. The **Customer Information** panel opens.
7. Enter a **User Name** and **Organization** name to identify this installation of Repackager.
8. Enter the **Serial Number** you received for the Edition of AdminStudio that you purchased.
9. Click **Next**. The **Destination Folder** panel opens.
10. If you want to install Repackager in the specified directory, click **Next**. If you want to select a different directory, click **Change**, select a new directory, and then click **Next**. The **AdminStudio Shared Location** panel opens.
11. Specify the location of your organization’s `AdminStudio Shared` directory, and click **Next**. The **Ready to Install** panel opens.
12. Click **Install** to begin the installation process. The **Installing Repackager** panel opens. When installation is complete, the **InstallShield Wizard Completed** panel opens.
13. Click **Finish** to exit the Wizard. A Repackager shortcut will be added to the Windows **Start** menu under **AdminStudio**, **AdminStudio Tools**.

Repackaging Legacy Installations Using the Repackaging Wizard

One frequently used method of creating a Repackager project is to repackage a legacy setup. Fundamentally, this involves monitoring the execution of a non-Windows Installer setup and converting changes made by the setup into a Windows Installer file.

Repackager provides the Repackaging Wizard for accomplishing this task. Using this Wizard, you can select the repackaging method (either Snapshot or Installation Monitoring), specify the setup(s) you want to repackage, and run the setup(s). When the Repackaging Wizard has finished its analysis, Repackager automatically creates a Repackager project (`.irp`) file, which can be modified in Repackager. You can then convert this file to an InstallShield Editor project (`.ism`) for further editing, or convert it directly to a Windows Installer package (`.msi`).

For InstallShield Editor or DevStudio 9.x InstallScript installations, you can choose to use the InstallShield Professional Logging Method instead of the Snapshot or Installation Monitoring methods.

***Caution** • It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

When using the Repackaging Wizard to repackage a legacy setup, you can use any of the following methods:

Table 7-7 • Repackaging Methods

Repackaging Method	Description
Installation Monitoring Method	Repackager monitors system changes as an application is installed, and that data can be converted into a Windows Installer package. Installation Monitoring is the default method. See Repackaging Using the Installation Monitoring Method. ***Edition** •*
Snapshot Method	Repackager compares a system snapshot before and after an installation, determines the changes that were made, and that data can be converted to a Windows Installer package. This is the default method. See Repackaging Using the Snapshot Method.
InstallShield Professional Logging Method	Repackager reads logged output of InstallShield Editor and DevStudio 9.*x* InstallScript installations, allowing you to get additional information that would not be captured by standard repackaging. See Repackaging Using the InstallShield Professional Logging Method.

Table 7-7 • Repackaging Methods (cont.)

Repackaging Method	Description
Using InstallScript Scan	You can use the Repackaging Wizard and InstallScript Scan to convert an InstallScript MSI installation to a Basic MSI package with InstallScript support. InstallScript Scan preserves the original components and much of the InstallScript installation logic, architecture, and maintainability of the original installation package. See Repackaging an InstallScript MSI Setup to a Basic MSI Setup

Repackaging Using the Installation Monitoring Method

When you choose the **Installation Monitoring** method of repackaging, Repackager monitors system changes as an application is installed, and then you can convert that data into a Windows Installer package. Installation Monitoring is the default method.

Caution • *It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

To repackage an installation using the Installation Monitoring method, perform the following steps:

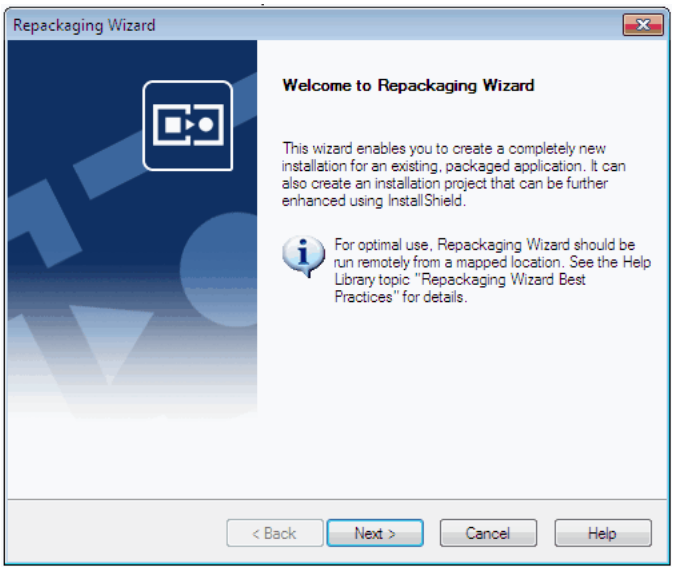
- Step 1: Selecting the Repackaging Method.
- Step 2: Excluding Processes (Optional)
- Step 3: Collecting Product Information
- Step 4: Adding Additional Setup Programs (Optional)
- Step 5: Set Target Project Information
- Step 6: Set Capture Settings (Optional)
- Step 7: Beginning the Repackaging Process

Step 1: Selecting the Repackaging Method

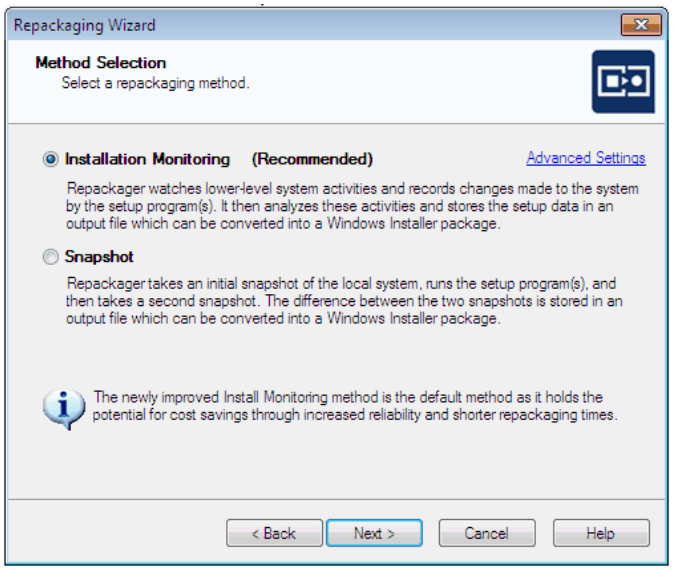
In this step, you launch the Repackaging Wizard and select the Installation Monitoring repackaging method.

Task: ***To select a repackaging method:***

1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.



2. Click **Next**. The Method Selection Panel opens.



3. Select **Installation Monitoring**.

Note • *The InstallShield Professional Logging Method, which should be used to repackage InstallShield Editor and DevStudio 9.x InstallScript installations, is not offered as a choice on the Method Selection panel. See Repackaging Using the InstallShield Professional Logging Method for more information on using that method.*

4. Continue with Step 2: Excluding Processes (Optional).

Step 2: Excluding Processes (Optional)

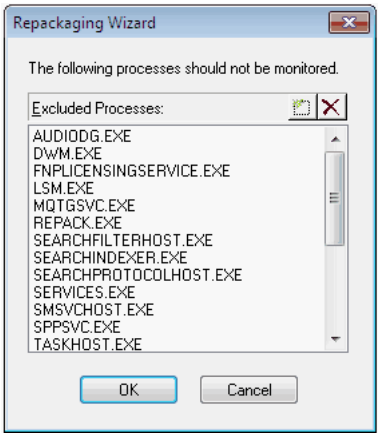
During Installation Monitoring, Repackager captures all of the activity of each service or process running on the machine, and then processes this collected data. However, many services running on a machine may have nothing to do with the installation being repackaged.

- **If you want to modify the default excluded processes list**, perform the following steps.
- **If you do not want to modify the default excluded processes list**, continue with Step 3: Collecting Product Information.

Tip • *If you know that the installation that you are capturing is from a self-extracting `.exe` file and if you want to use the **Installation Monitoring** method, you should add the name of that `.exe` file to the excluded processes list.*

Task: ***To exclude processes from Installation Monitoring:***

1. On the Method Selection Panel, click the **Advanced Settings** link. The Excluded Processes dialog box opens, listing a default set of processes.



2. To add a process to this list, click the New ( ) button to add a new blank line to this list, and enter the name of the process that you want to exclude.

3. To delete a process from this list, select the process and click the Delete ( ) button.

***Note** • The changes you make to the excluded processes list are persisted for future Repackaging sessions. Therefore, once you have entered an appropriate set of processes to exclude for your machine, you can skip this optional step.*

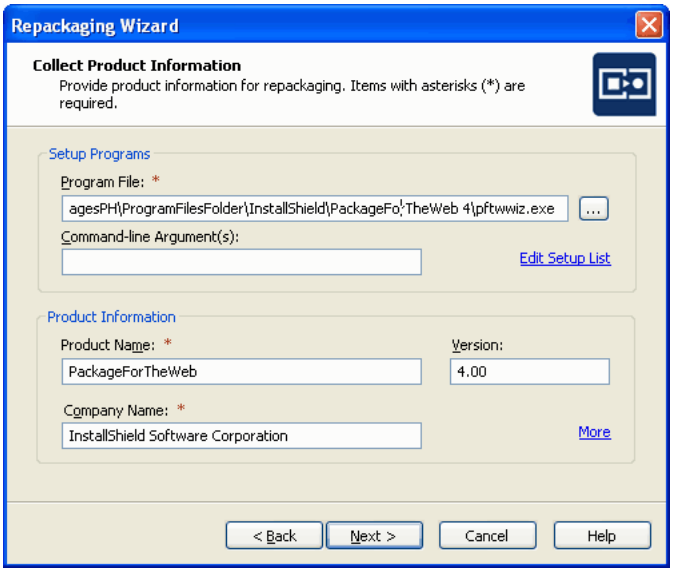
4. Click **OK** to return to the Method Selection Panel.
5. Continue with Step 3: Collecting Product Information.

Step 3: Collecting Product Information

In this step, you will specify the installation you want to repackage and enter any command-line arguments to be used when the installation is run.

Task: ***To enter product information:***

1. On the Method Selection Panel, click **Next**. The Collect Product Information Panel opens.



2. Click the Browse (...) button next to the **Program File** field and select the installation program that you are repackaging.
3. In the **Command-line Argument(s)** field, enter any command-line arguments to be used when the installation is run.
4. In the **Product Information** area, modify the **Product Name**, **Version**, and **Company Name**, as necessary.
5. If you want to associate Web sites with this installation, click the **More** link to open the Additional Product Information dialog box, enter the **Product URL** and **Support URL** for the application you are repackaging, and click OK.

6. Continue with Step 4: Adding Additional Setup Programs (Optional).

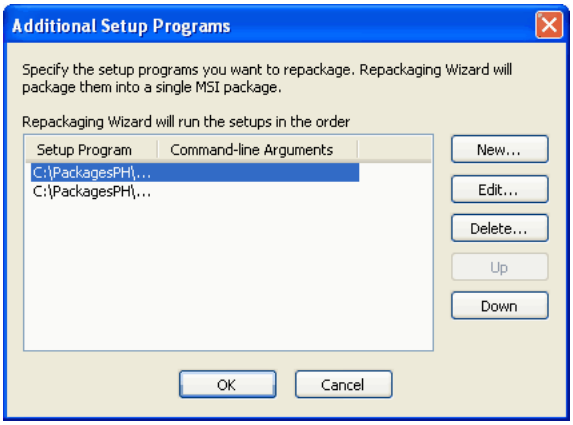
Step 4: Adding Additional Setup Programs (Optional)

You can specify additional setup programs to repackage together with this installation. Additional setup programs share the same product name, version number, and company name in the repackaged installation. However, as you locate each additional setup program to repackage, you can specify command-line parameters pertaining only to that setup. You can also specify the order in which the setups are run, should it be necessary.

- **If you want to add additional setup programs**, perform the following steps.
- **If you do not want to add additional setup programs**, continue with Step 5: Set Target Project Information.

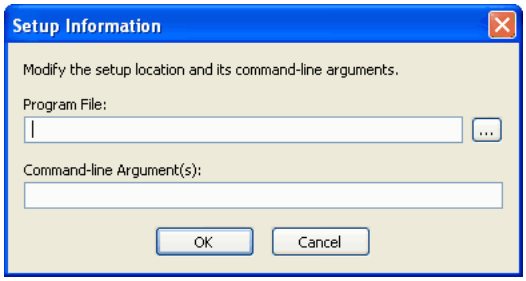
Task: ***To add additional setup programs, perform the following steps:***

1. On the Collect Product Information Panel, click the **Edit Setup List link**. The **Additional Setup Programs** dialog box opens.



2. If you want to **add** a setup program, perform the following steps:

   a. Click **New**. The **Setup Information** dialog box opens.



   b. Click the Browse (...) button next to the **Program File** field and select the setup program that you want to add.

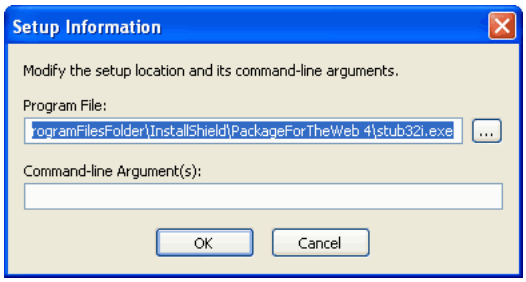
c. In the **Command-line Argument(s)** field, enter any command-line arguments to be used when this setup is run.

d. Click **OK** to return to the **Additional Setup Programs** dialog box.

e. If necessary, click the **Up** and **Down** buttons to change the order in which the setups are run.

3. If you want to **edit** an existing setup program, perform the following steps:

a. On the Additional Setup Programs dialog box, select the program that you want to edit and click **Edit**. The Setup Information dialog box opens.



b. Modify the **Program File** and **Command-line Argument(s)** fields.

c. Click **OK** to return to the Additional Setup Programs dialog box.

4. If you want to **delete** a listed setup program, perform the following steps:

a. Select the program that you want to delete and click **Delete**. A dialog box opens prompting you to confirm the deletion.

b. Click **OK** to confirm the deletion and return to the Additional Setup Programs dialog box, where the deleted program is no longer listed.

5. Click **OK** to return to the **Collect Product Information** panel.

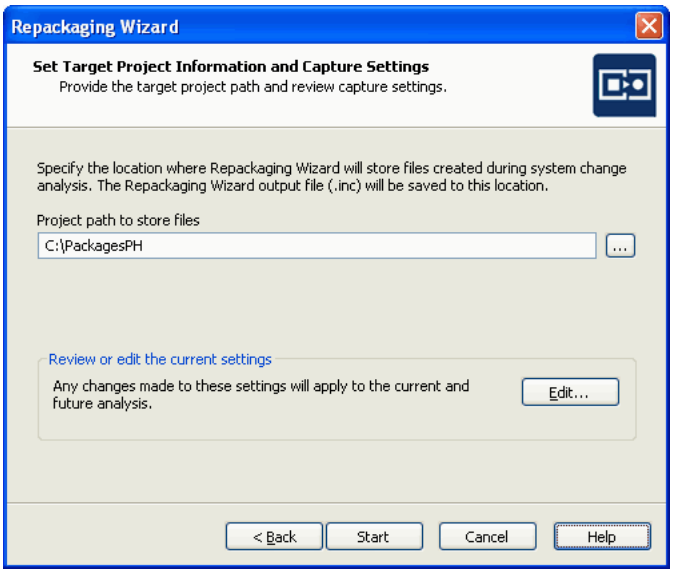
6. Continue with Step 5: Set Target Project Information.

Step 5: Set Target Project Information

In this step, you identify the location where you want files created by Repackager to be stored. For the Installation Monitoring repackaging method, it is recommended that this location not be located on your clean machine, but rather on the same machine as the Repackager executable (most likely on your administrator machine).

Task: ***To set target project information and capture settings:***

1. On the Collect Product Information Panel, click **Next**. The Set Target Project Information and Capture Settings Panel opens.



Note • *If you specified an InstallShield Editor or DevStudio 9.x InstallScript installation in the* ***Program File*** *field of the Collect Product Information Panel, the InstallShield Professional Setup Panel appears instead of the Set Target Project Information and Capture Settings Panel. See Repackaging Using the InstallShield Professional Logging Method.*

2. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

   You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

3. Continue with Step 6: Set Capture Settings (Optional).

Step 6: Set Capture Settings (Optional)

From the Set Target Project Information and Capture Settings Panel, you can specify the following capture types for the repackaging session:

- Files and deleted files
- .ini files and .ini files with non-.ini extensions
- Shortcuts

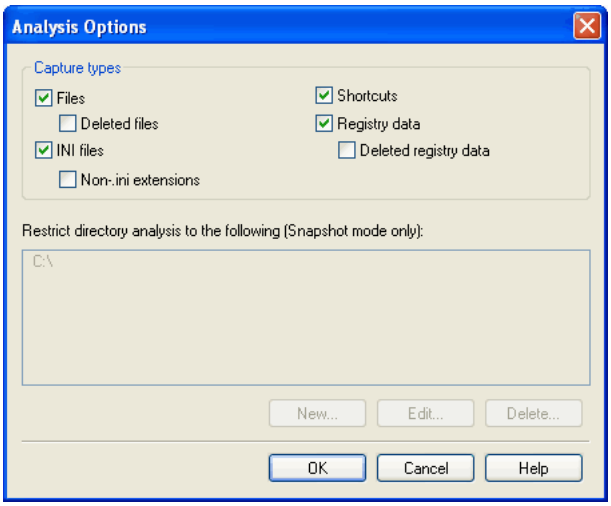
- Registry data and deleted registry data

Options set in this dialog box apply to the current and subsequent repackaging sessions.

- **If you want to set capture settings**, perform the following steps.
- **If you do not want to set capture settings**, continue with Step 7: Beginning the Repackaging Process.

Task: ***To set capture settings:***

1. On the Set Target Project Information and Capture Settings Panel, click **Edit**. The Analysis Options dialog box opens.



Note • *Options set in this dialog box apply to the current and subsequent repackaging sessions.*

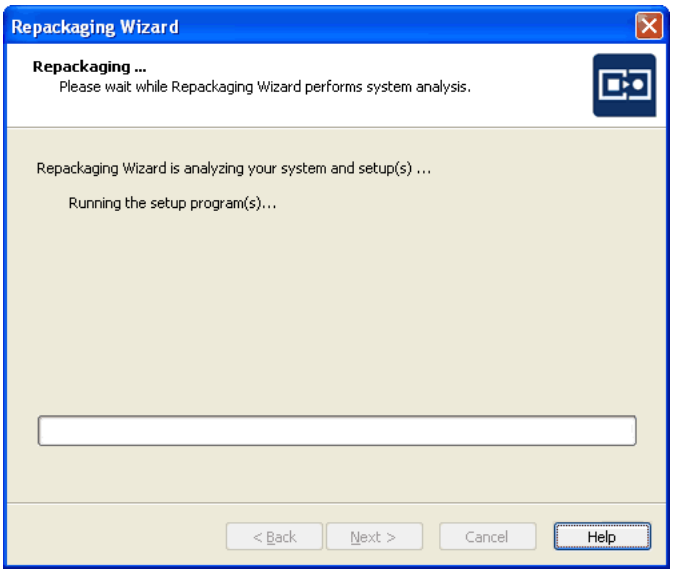
2. Select the capture types that you want to use for this repackaging session:
   - **Files**
   - **Deleted files**
   - **INI files**
   - (INI files with) **Non-.ini extensions**
   - **Shortcuts**
   - **Registry data**
   - **Deleted registry data**
3. Click **OK** to return to the Set Target Project Information and Capture Settings Panel.
4. Continue with Step 7: Beginning the Repackaging Process.

Step 7: Beginning the Repackaging Process

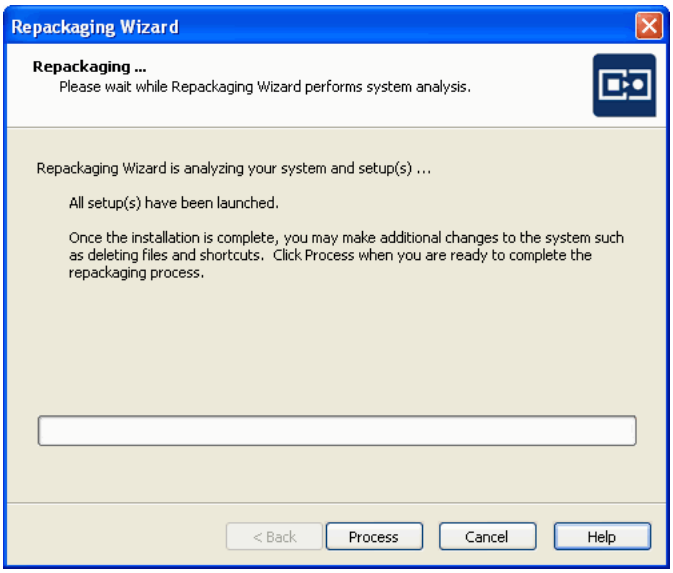
In this step you will begin the repackaging process.

Task: ***To begin the repackaging process:***

1. To begin the repackaging process, click **Start** on the Set Target Project Information and Capture Settings Panel. The **Repackaging Panel** opens and the Repackaging Wizard captures the initial system status. Then, the selected setup program will be launched.

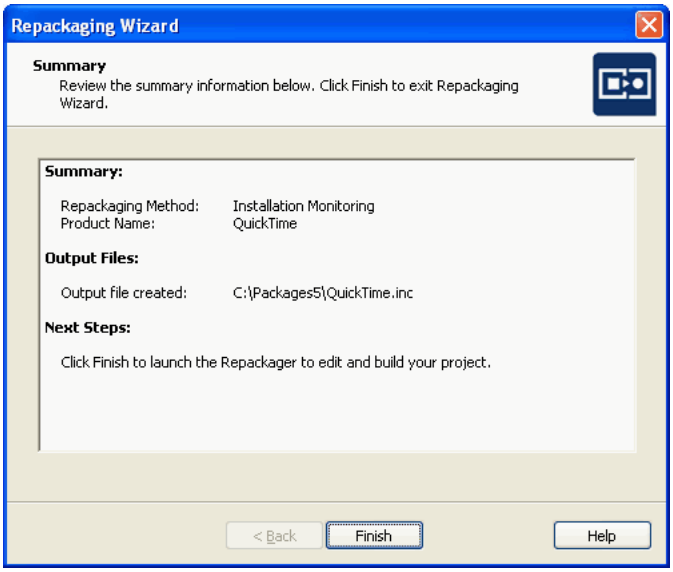


2. Follow the prompts until the installation has completed. When the installation is complete, you are prompted to make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.



3. When you are ready to complete the repackaging process, click **Process**. The Repackaging Wizard then analyzes the system and setup data that it collected.

   Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful.



4. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.
5. Continue with the instructions in Working With Repackager Projects.

Repackaging Using the Snapshot Method

When using the **Snapshot** method of repackaging, the Repackaging Wizard takes a reference snapshot of a system as a baseline configuration, performs the installation, and then takes a second snapshot.

The difference between the two snapshots is stored in a directory you specify, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and the source files. The Repackager project file can then be converted into a Windows Installer package (`.msi`).

Caution • *It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

Types of Snapshot Repackaging

There are two types of Snapshot repackaging:

Single Step

When Repackaging in a single step:

- You specify at least one setup program to repackage.
- Repackager first takes an initial system snapshot.
- Repackager then runs the setup program(s) you selected.
- Then Repackager takes a second snapshot to create the script file that can be converted into a Windows Installer package.

You also have the option of requiring the Repackager to prompt you before running the setup program(s), allowing you the opportunity to make changes to your system that you want included in the final package.

See Performing Single Step Snapshot Repackaging.

Multiple Step

When repackaging in multiple steps:

- You run the Repackager to obtain an initial system snapshot, after which the Repackager exits.
- You can then perform any modifications to the system, such as changing configurations, running installations, and so forth.
- After making the necessary modifications, you would then run the Repackager again to analyze system status changes.
- Repackager compares the final snapshot to the initial snapshot to determine the system changes that were made, and then records that information in a script file.

See Performing Multiple Step Snapshot Repackaging.

Performing Multiple Step Snapshot Repackaging

To repackage an installation using the **Multiple Step Snapshot** method, perform the following steps:

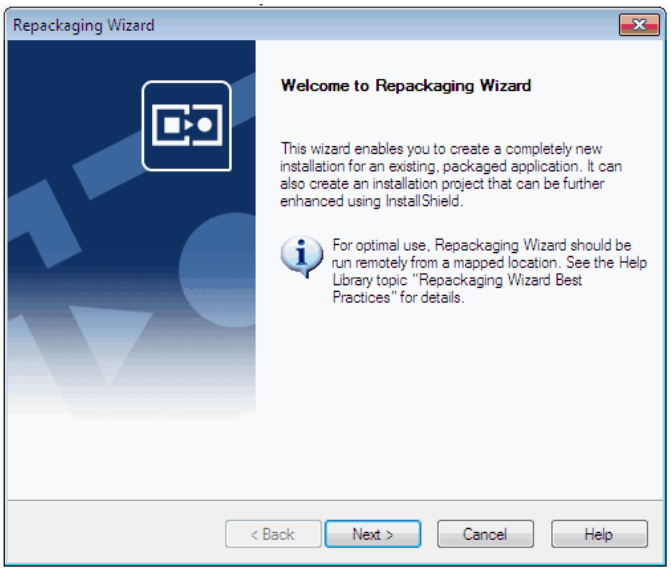
- Step 1: Selecting the Repackaging Method.
- Step 2: Initial Analysis
- Step 3: Install Setup and Make Manual System Changes
- Step 4: Entering Product Information
- Step 5: Set Target Project Information
- Step 6: Set Capture Settings (Optional)
- Step 7: Beginning the Repackaging Process

Step 1: Selecting the Repackaging Method

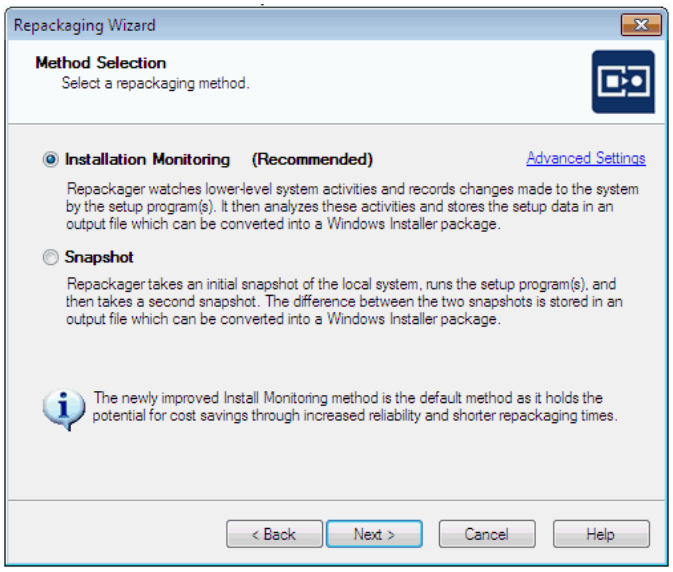
In this step, you launch the Repackaging Wizard and select the **Snapshot** repackaging method.

Task: ***To select a repackaging method:***

1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.

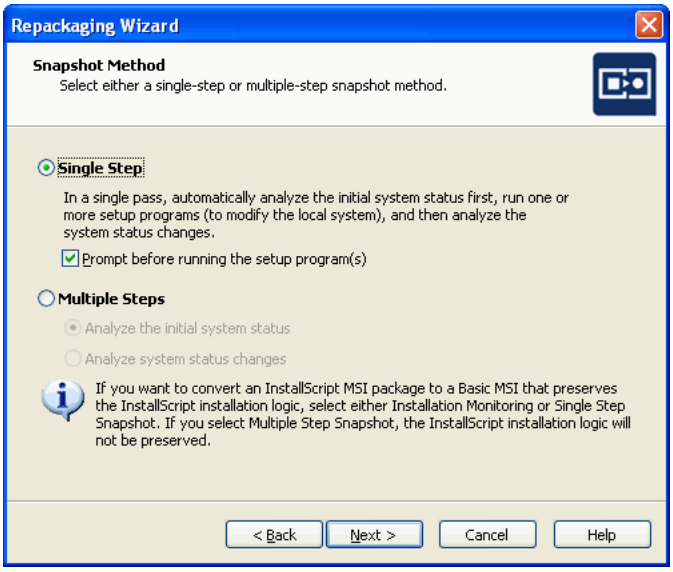


2. Click **Next**. The Method Selection Panel opens.



Note • *The InstallShield Professional Logging Method, which should be used to repackage InstallShield Editor and DevStudio 9.x InstallScript installations, is not offered as a choice on the Method Selection panel. See Repackaging Using the InstallShield Professional Logging Method for more information on using that method.*

3. Select **Snapshot** and click Next. The Snapshot Method panel opens.



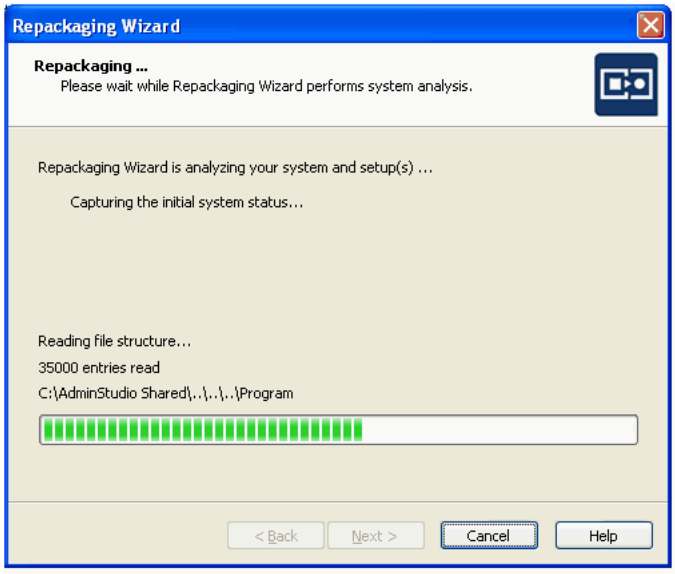
4. On the Snapshot Method panel, select **Multiple Steps**. The **Analyze the initial system status** option is enabled.
5. Select the **Analyze the initial system status** option.
6. Continue with Step 2: Initial Analysis.

Step 2: Initial Analysis

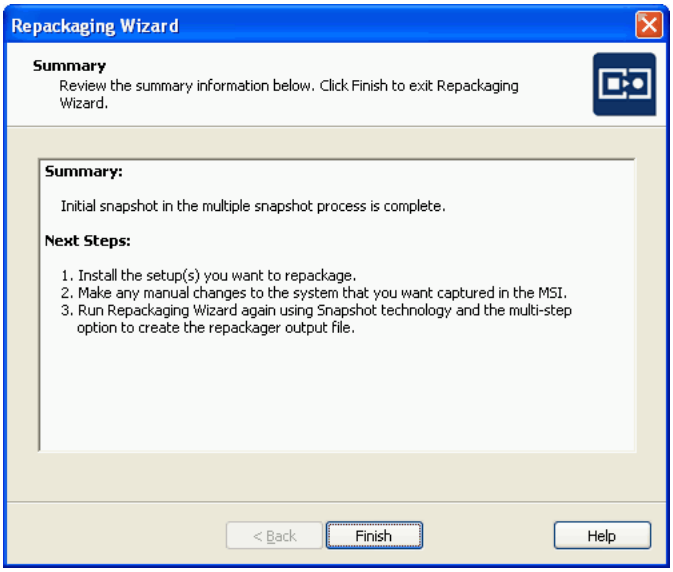
In this step, the Repackaging Wizard takes an initial snapshot of your system.

Task: ***To perform initial analysis:***

1. On the Snapshot Method panel, click **Next**. The Repackaging Panel of the Repackaging Wizard opens, displaying the progress of the initial system status capture.



When Repackager finishes capturing the initial system status, the **Summary** panel opens, prompting you to install the application you are repackaging.



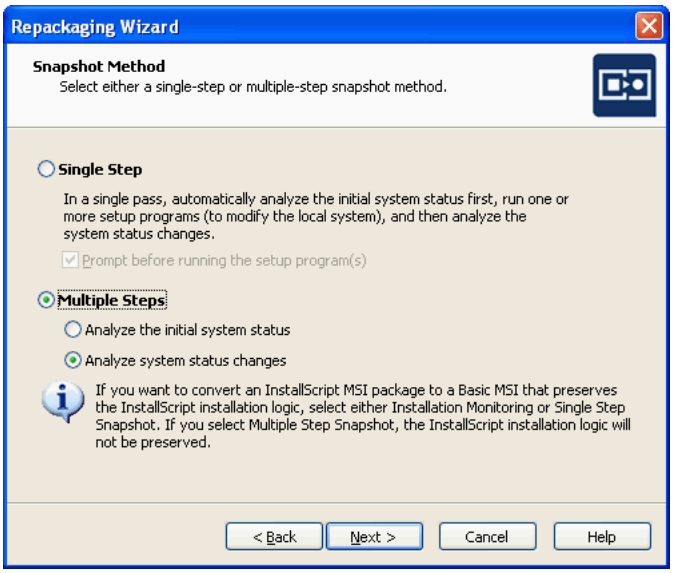
2. Click **Finish** to close the Repackaging Wizard.
3. Continue with Step 3: Install Setup and Make Manual System Changes.

Step 3: Install Setup and Make Manual System Changes

In this step, you will manually launch the installation of the application you are repackaging, and then you will make any manual changes to the system that you want captured in the Windows Installer package.

Task: ***To install setup and make manual system changes:***

1. Launch the installation program of the application you are repackaging.
2. Follow the prompts until the installation has completed.
3. When the installation is complete, make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.
4. Launch the Repackaging Wizard again. The **Welcome Panel** opens.
5. Click **Next**. The **Method Selection Panel** opens.
6. Select **Snapshot** and click **Next**. The **Snapshot Method Panel** opens with **Multiple Steps** already selected, and the **Analyze system status changes** option now enabled and selected.



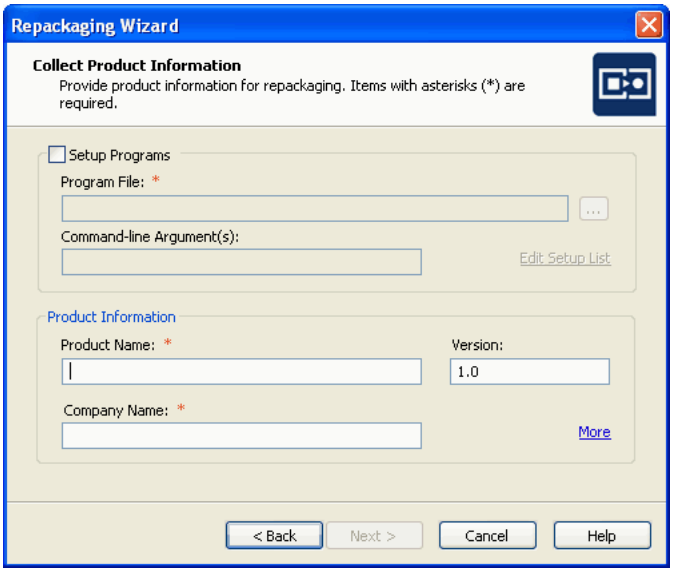
7. Continue with Step 4: Entering Product Information.

Step 4: Entering Product Information

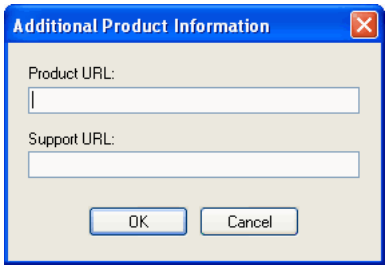
In this step, you will enter product information for the application that you just installed.

Task: ***To enter product information:***

1. On the **Snapshot Method Panel**, click **Next**. The **Collect Product Information Panel** opens. Because you are now performing the second step of a multiple-step Snapshot, the **Setup Programs** area is disabled (because you have already installed the application you are repackaging).



2. In the **Product Information** area, modify the **Product Name**, **Version**, and **Company Name**, as necessary.
3. If you want to associate Web sites with this installation package, perform the following steps:
   a. Click the **More** link. The **Additional Product Information** dialog box opens.



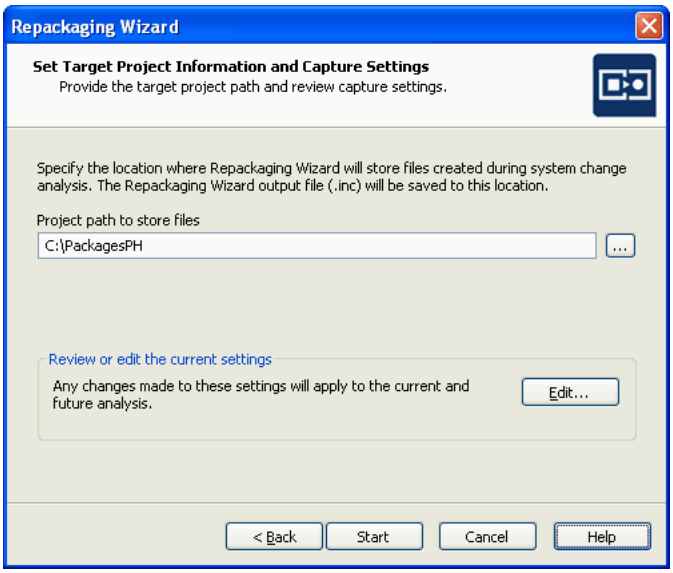
   b. Enter the **Product URL** and **Support URL** for the application you are repackaging.
   c. Click **OK**.
4. Continue with Step 5: Set Target Project Information.

Step 5: Set Target Project Information

In this step, you identify the location where you want files created by Repackager to be stored.

Task: ***To set target project information and capture settings:***

1. On the **Collect Product Information Panel**, click **Next**. The **Set Target Project Information and Capture Settings Panel** opens.



2. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

   You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

3. Continue with Step 6: Set Capture Settings (Optional).

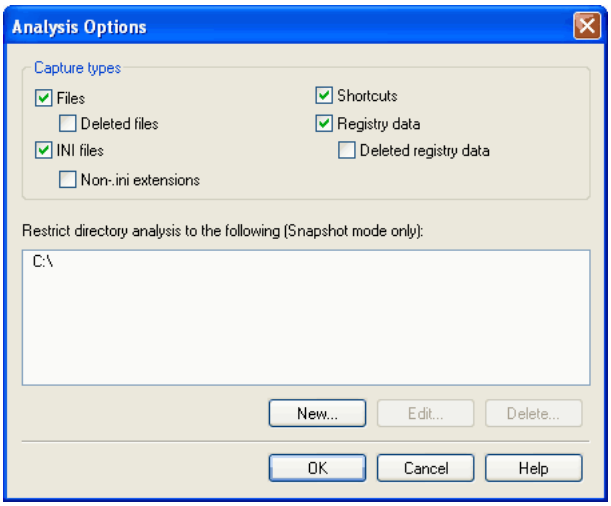
Step 6: Set Capture Settings (Optional)

From the **Set Target Project Information and Capture Settings Panel**, you can specify capture types for the repackaging session such as files, .ini files, shortcuts, and Registry data. You can also restrict directory analysis to specific directories, which can significantly improve repackaging performance.

- **If you want to modify the default capture settings**, perform the following steps.
- **If you do not want to modify the default capture settings**, click **Next** and continue with Step 7: Beginning the Repackaging Process.

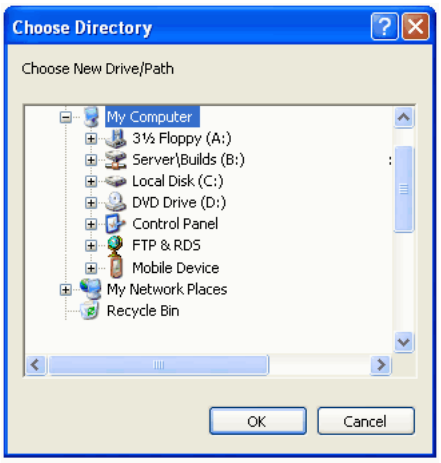
Task: ***To modify capture settings:***

1. On the **Set Target Project Information and Capture Settings Panel**, click **Edit**. The **Analysis Options** dialog box opens.



Note • *Options set in this dialog box apply to the current and subsequent repackaging sessions.*

2. Select the capture types that you want to use for this repackaging session:
   - **Files**
   - **Deleted files**
   - **INI files**
   - (INI files with) **Non-.ini extensions**
   - **Shortcuts**
   - **Registry data**
   - **Deleted registry data**
3. If you want to restrict directory analysis to specific directories, click **New**. The **Choose Directory** dialog box opens.



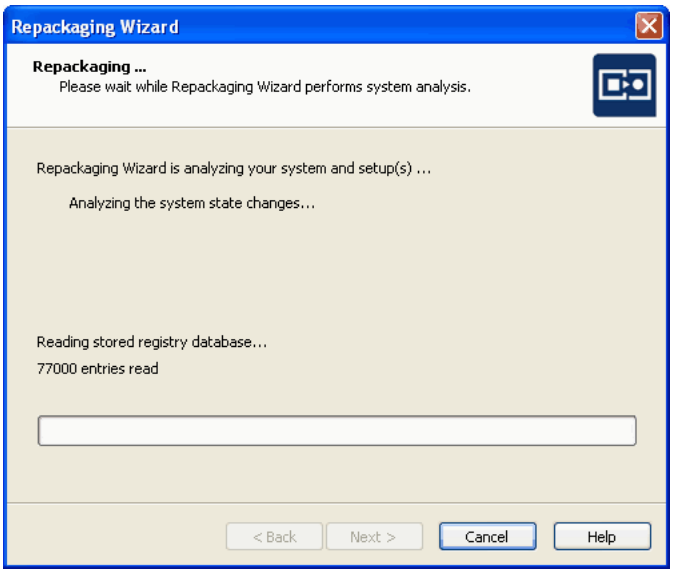
4. Select a directory and click **OK**. The selected directory is now listed on the Analysis Option dialog box.
5. If you want to modify an existing restriction, or delete a restriction, select the directory and click **Edit** or **Delete**.
6. Click **OK** to return to the **Set Target Project Information and Capture Settings Panel**.
7. Continue with Step 7: Beginning the Repackaging Process.

Step 7: Beginning the Repackaging Process

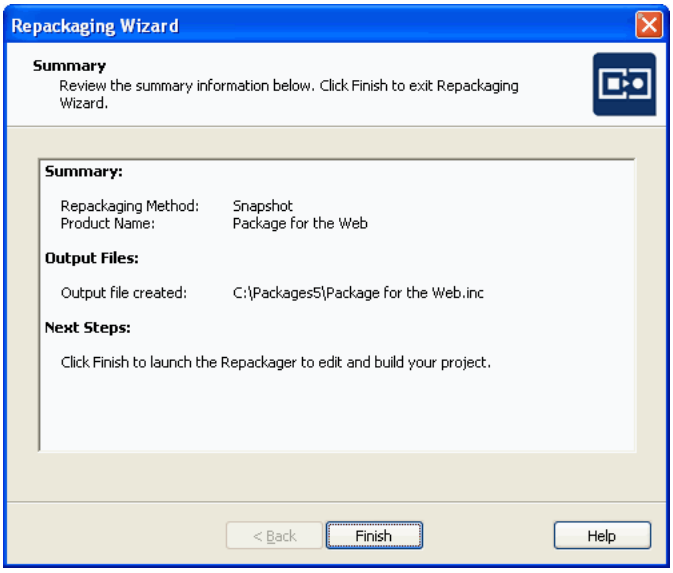
In this step you will begin the repackaging process.

Task: ***To begin the repackaging process:***

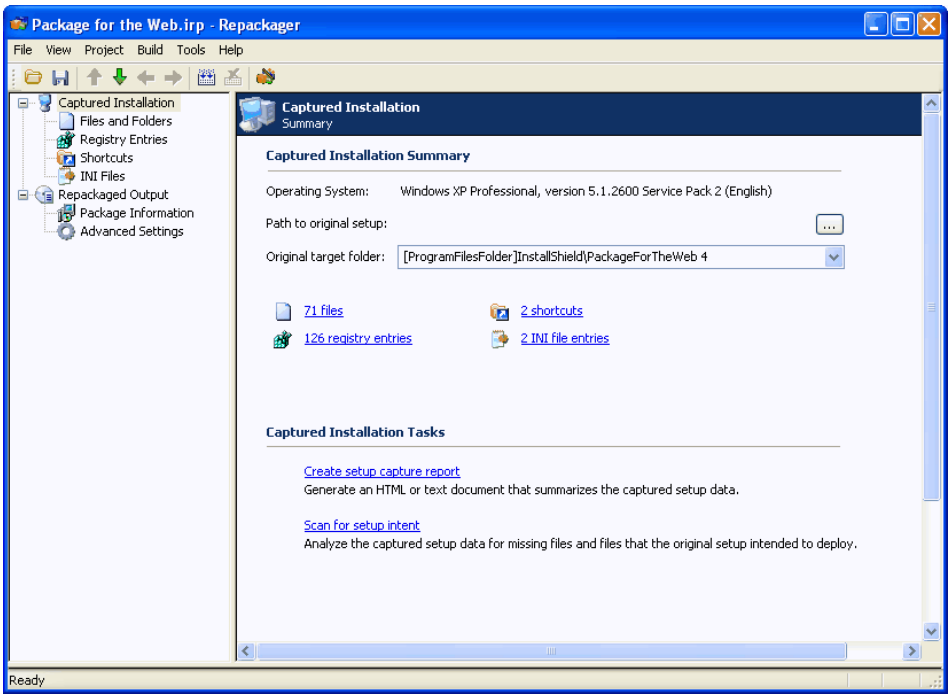
1. To begin the repackaging process, click **Start** on the Set Target Project Information and Capture Settings Panel. The **Repackaging Panel** opens and the Repackaging Wizard captures the system state changes.



When the Repackaging Wizard has finished analyzing the system state changes and creating the Repackager project, the Summary Panel opens, providing confirmation that the repackaging was successful and listing the location of your new Repackager project.



2. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.



3. Continue with the instructions in Working With Repackager Projects.

Performing Single Step Snapshot Repackaging

To repackage an installation using the **Single Step Snapshot** method, perform the following steps:

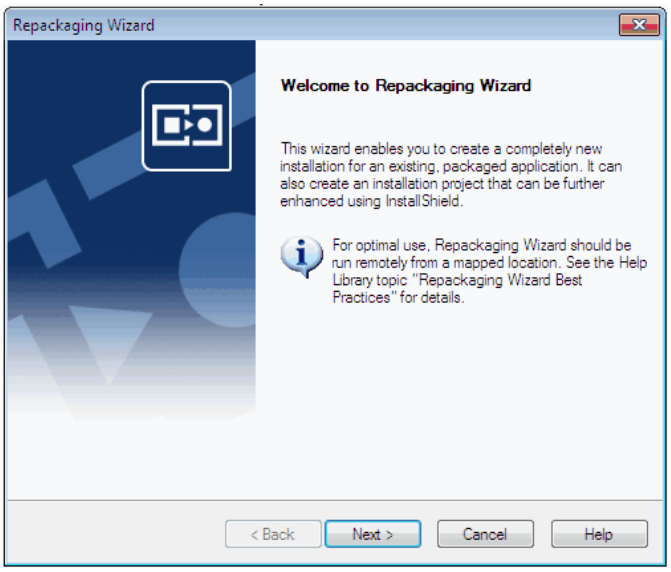
- Step 1: Selecting the Repackaging Method.
- Step 2: Collecting Product Information
- Step 3: Set Target Project Information
- Step 4: Set Capture Settings (Optional)
- Step 5: Beginning the Repackaging Process

Step 1: Selecting the Repackaging Method

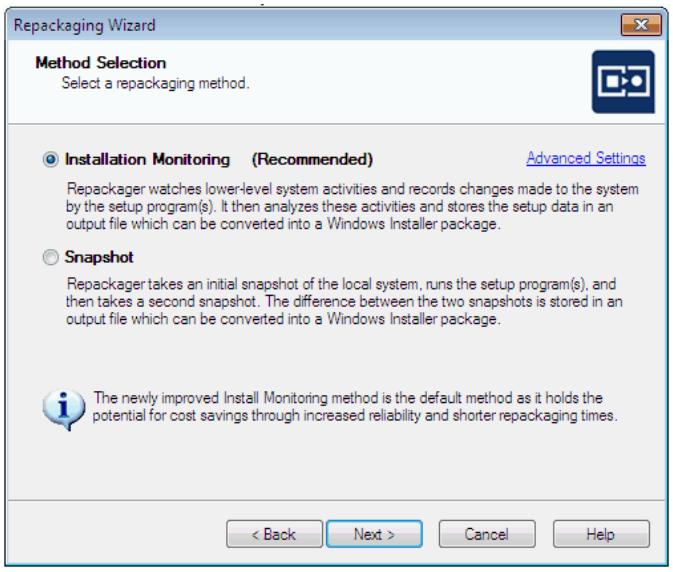
In this step, you launch the Repackaging Wizard and select the **Snapshot** repackaging method.

Task: ***To select a repackaging method:***

1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.

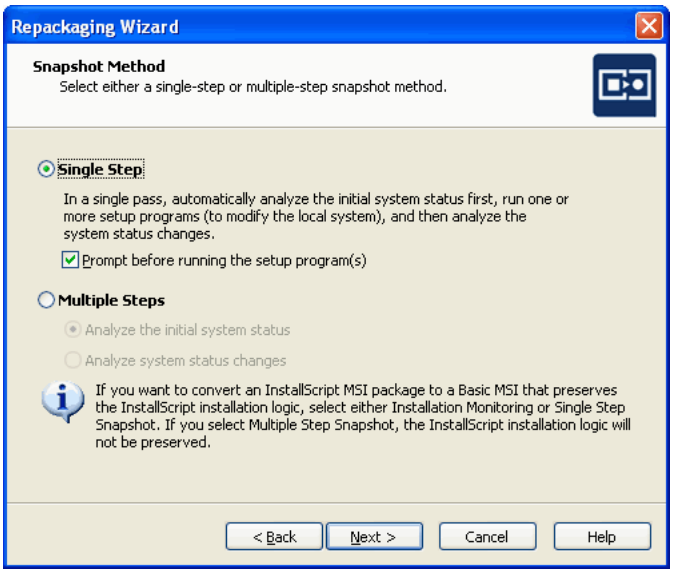


2. Click **Next**. The Method Selection Panel opens.



Note • *The InstallShield Professional Logging Method, which should be used to repackage InstallShield Editor and DevStudio 9.x InstallScript setups, is not offered as a choice on the Method Selection panel. See Repackaging Using the InstallShield Professional Logging Method for more information on using that method.*

3. Select **Snapshot** and click Next. The Snapshot Method panel opens.



4. On the Snapshot Method panel, select **Single Step**.
5. If you want to be prompted before the selected setup program is launched, select the **Prompt before running the setup program(s) option**. If you do not select this option, the setup program will automatically be launched as soon as the Repackaging Wizard has finished analyzing the system status.
6. Continue with Step 2: Collecting Product Information.

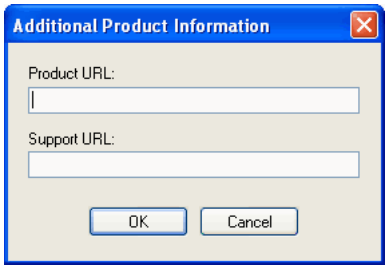
Step 2: Collecting Product Information

In this step, you will specify the installation you want to repackage and enter any command-line arguments to be used when the installation is run.

Task: ***To enter product information:***

1. On the Snapshot Method panel, click **Next**. The Collect Product Information Panel opens with the **Setup Programs** and **Product Information** areas enabled.

2. Click the Browse (...) button next to the **Program File** field and select the installation program that you are repackaging.
3. In the **Command-line Argument(s)** field, enter any command-line arguments to be used when the installation is run.
4. In the **Product Information** area, modify the **Product Name**, **Version**, and **Company Name**, as necessary.
5. If you want to associate Web sites with this installation package, perform the following steps:
   a. Click the **More** link. The **Additional Product Information** dialog box opens.



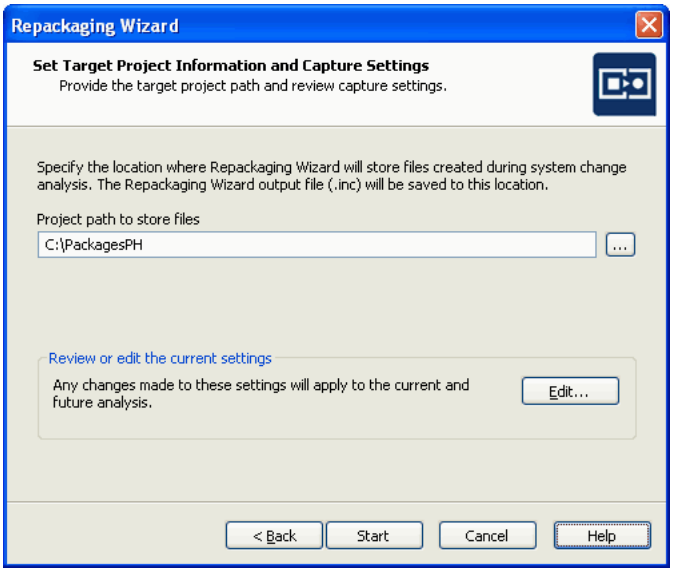
   b. Enter the **Product URL** and **Support URL** for the application you are repackaging.
   c. Click **OK**.
6. Continue with Step 3: Set Target Project Information .

Step 3: Set Target Project Information

In this step, you identify the location where you want files created by Repackager to be stored.

Task: ***To set target project information:***

1. On the Collect Product Information Panel, click **Next**. The **Set Target Project Information and Capture Settings Panel** opens.



2. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

   You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

3. Continue with Step 4: Set Capture Settings (Optional).

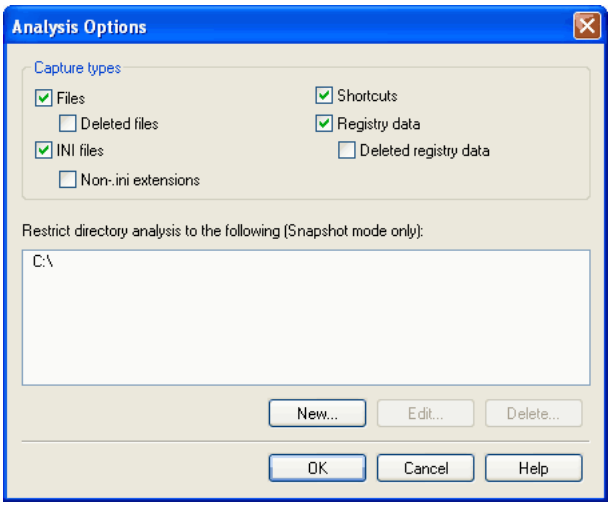
Step 4: Set Capture Settings (Optional)

From the **Set Target Project Information and Capture Settings Panel**, you can specify capture types for the repackaging session such as files, .ini files, shortcuts, and Registry data. You can also restrict directory analysis to specific directories, which can significantly improve repackaging performance.

- **If you want to modify the default capture settings**, perform the following steps.
- **If you do not want to modify the default capture settings**, click **Next** and continue with Step 7: Beginning the Repackaging Process.

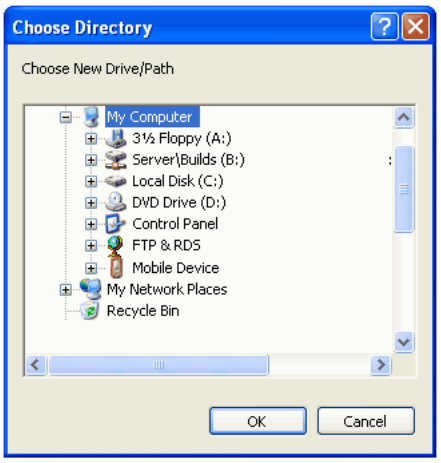
Task: ***To modify capture settings:***

1. On the **Set Target Project Information and Capture Settings Panel**, click **Edit**. The **Analysis Options** dialog box opens.



Note • *Options set in this dialog box apply to the current and subsequent repackaging sessions.*

2. Select the capture types that you want to use for this repackaging session:
   - **Files**
   - **Deleted files**
   - **INI files**
   - (INI files with) **Non-.ini extensions**
   - **Shortcuts**
   - **Registry data**
   - **Deleted registry data**
3. If you want to restrict directory analysis to specific directories, click **New**. The **Choose Directory** dialog box opens.



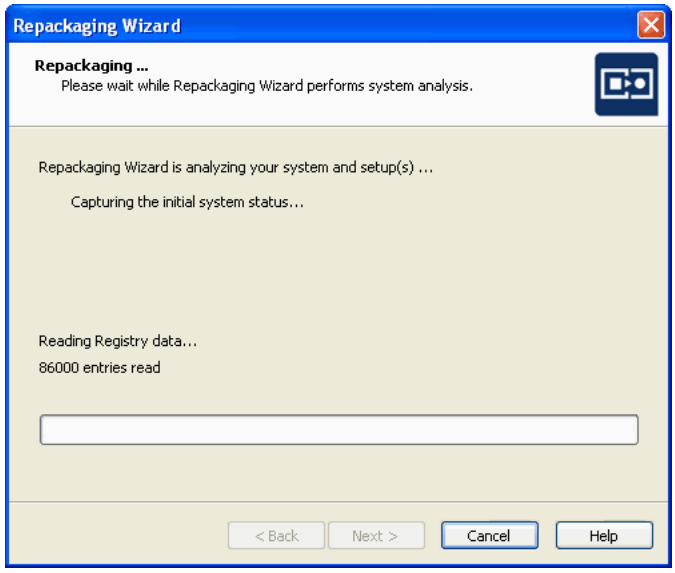
4. Select a directory and click **OK**. The selected directory is now listed on the Analysis Option dialog box.
5. If you want to modify an existing restriction, or delete a restriction, select the directory and click **Edit** or **Delete**.
6. Click **OK** to return to the **Set Target Project Information and Capture Settings Panel**.
7. Continue with Step 7: Beginning the Repackaging Process.

Step 5: Beginning the Repackaging Process

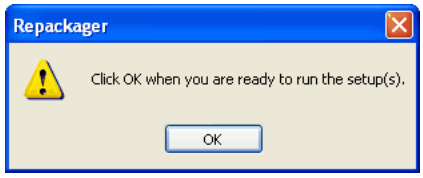
In this step you will begin the repackaging process.

Task: ***To begin the repackaging process:***

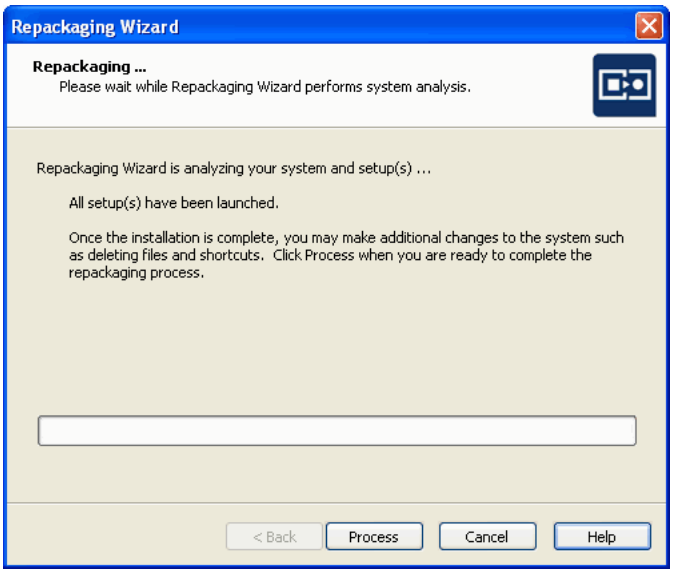
1. To begin the repackaging process, click **Start** on the **Set Target Project Information and Capture Settings Panel**. The **Repackaging Panel** opens and the Repackaging Wizard captures the initial system status.



Depending upon whether you chose the **Prompt before running the setup program(s)** option on the Snapshot Method Panel, either the installation that you selected will start or you will be prompted to start it.

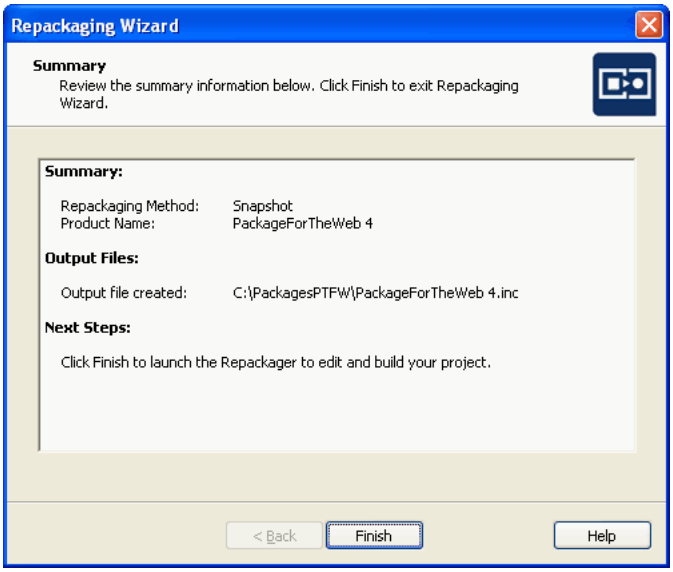


2. Install the application by following the prompts until the installation has completed.
3. When the installation is complete, you are prompted to make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.



4. When you are ready to complete the repackaging process, click **Process**. The Repackaging Wizard then analyzes the system and setup data that it collected.

   Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful and listing the location of the Repackager project that was just created.



5. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.
6. Continue with the instructions in Working With Repackager Projects.

Repackaging Using the InstallShield Professional Logging Method

If you are repackaging an InstallShield Editor or DevStudio 9.x InstallScript installation, you can use the InstallShield Professional Logging Method to read logged output of those setups to obtain additional information that would not be captured by the Installation Monitoring or Snapshot methods: Path Variables and the Feature Tree.

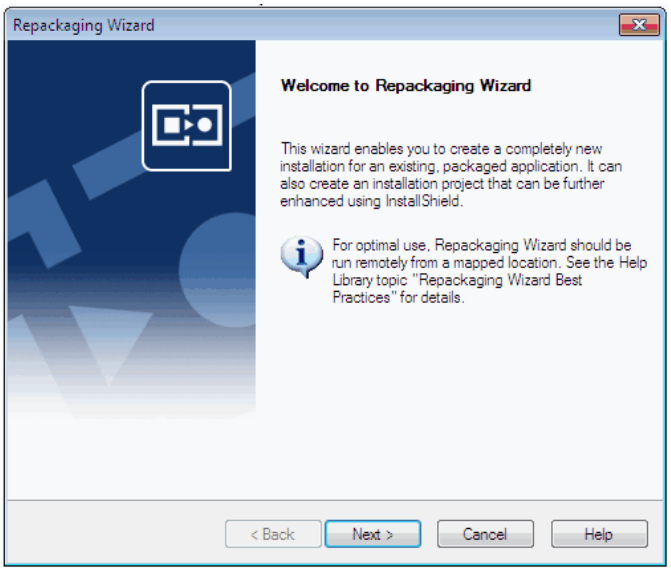
See InstallShield Professional Logging Method for more information.

Using the InstallShield Professional Logging Method

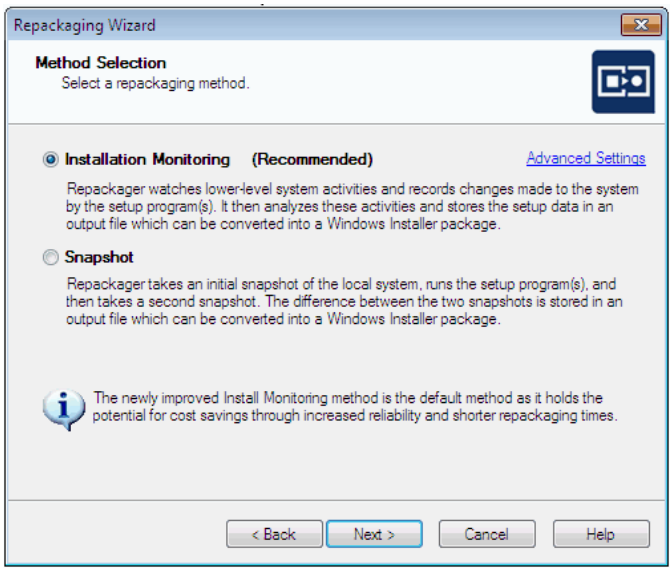
To repackage an InstallShield Editor or DevStudio 9.x InstallScript installation using the InstallShield Professional Logging Method, perform the following steps:

Task: ***To repackage using the InstallShield Professional Logging Method:***

1. Launch the **Repackaging Wizard** from Repackager. The **Welcome Panel** opens.



2. Click **Next**. The Method Selection Panel opens.



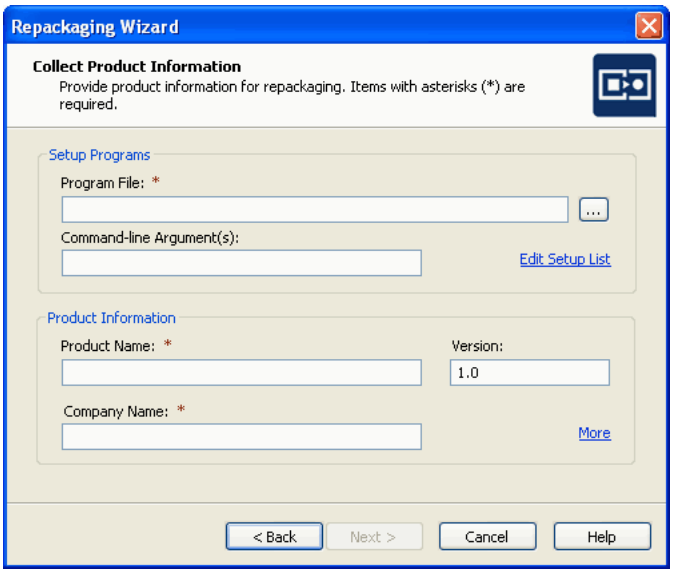
3. Select the **Installation Monitoring** method.

   The **InstallShield Professional Logging Method** is not offered as a choice on this panel, but when you select an InstallShield Editor or DevStudio 9.x InstallScript installation on the **Collect Product Information Panel** (the next panel in the Repackaging Wizard), Repackager will automatically detect the InstallScript installation and will display the **InstallShield Professional Setup Panel**. On this panel you can choose to use the **InstallShield Professional Logging Method** instead of the method you chose on the Method Selection Panel.

   Caution • *When you are attempting to repackage a setup using the InstallShield Professional Logging Method, you are permitted to select either **Installation Monitoring** or **Snapshot** on the Method Selection Panel, but it is recommended that you select **Installation Monitoring**.*

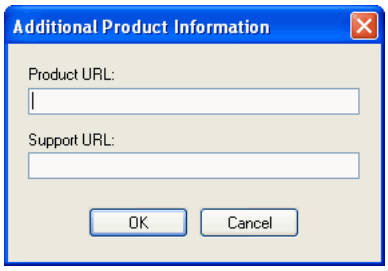
   *If you select the **Snapshot** method, an additional panel opens, entitled Snapshot Method. On this panel, if you select **Multiple Steps** instead of **Single Step**, Repackager will not recognize the setup as an InstallShield Editor or DevStudio 9.x InstallScript installation and the **InstallShield Professional Setup Panel** will not open. You can avoid this extra and possibly troublesome step by selecting **Installation Monitoring** on the Method Selection Panel.*

4. On the **Method Selection Panel**, click **Next**. The **Collect Product Information Panel** opens.

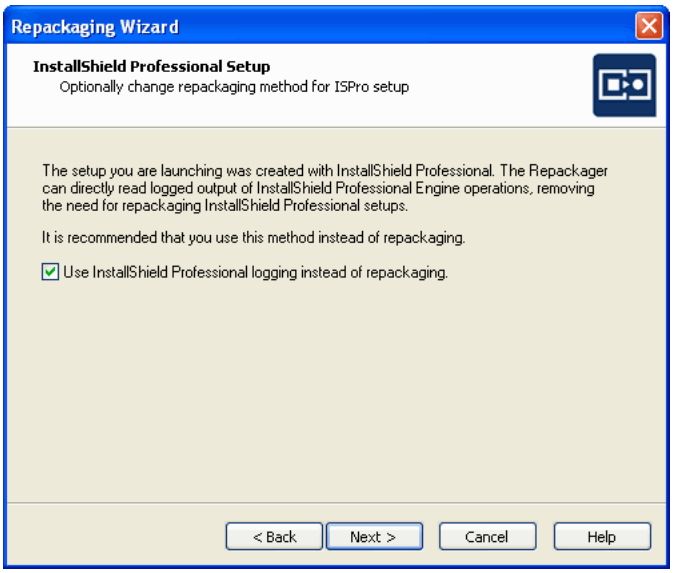


Caution • *Do not use the **Edit Setup List** link to add an additional setup program to this project. If you do, Repackager will not recognize the specified installation as an InstallShield Editor or DevStudio 9.x InstallScript installation and the **InstallShield Professional Setup Panel** will not open.*

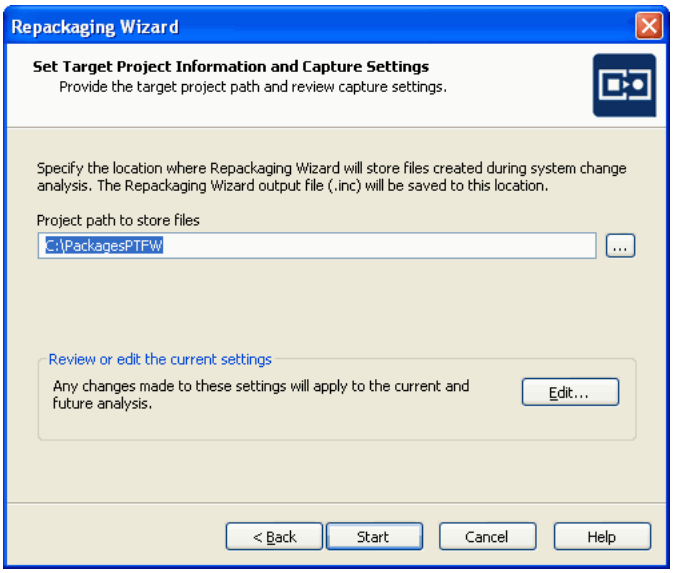
5. Click the Browse (...) button next to the **Program File** field and select the InstallShield Editor or DevStudio 9.*x* InstallScript installation that you are repackaging.
6. In the **Command-line Argument(s)** field, enter any command-line arguments to be used when the installation is run.
7. In the **Product Information** area, modify the **Product Name**, **Version**, and **Company Name**, as necessary.
8. If you want to associate Web sites with this installation package, perform the following steps:
   a. Click the **More** link. The **Additional Product Information** dialog box opens.



   b. Enter the **Product URL** and **Support URL** for the application you are repackaging.
   c. Click **OK**.
9. Click **Next**. Because you specified an InstallShield Editor or DevStudio 9.x InstallScript installation in the **Program File** field, the InstallShield Professional Setup Panel opens.



10. Select the **Use InstallShield Professional logging instead of repackaging** option.
11. Click **Next**. The **Set Target Project Information and Capture Settings Panel** opens.



12. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

    You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.
13. To begin the repackaging process, click **Start** on the Set Target Project Information and Capture Settings Panel. The selected InstallShield Editor or DevStudio 9.x InstallScript installation is launched.
14. Follow the prompts until the installation has completed.

Repackager then launches and opens the Repackager project file (`*.irp`) that you just created.

15. Continue with the instructions in Working With Repackager Projects.

Repackaging an InstallScript MSI Setup to a Basic MSI Setup

InstallScript MSI installations use a Windows Installer database for storage of all file/registry information, but the actual user interface, and much of the installation logic is driven by the InstallScript engine via a `setup.exe` file. This type of installation architecture can cause difficulties during deployment, such as:

- inability to customize or transform the application
- inability to perform conflict detection
- inability to suppress the user interface
- difficulty patching or upgrading the application

Also, if an InstallScript MSI installation is repackaged using traditional methods (OS Snapshot or Installation Monitoring), significant platform-specific or custom installation, maintenance, and uninstallation logic, and user interface information is lost because those methods only record the installation activities for the specific platform used during repackaging.

Therefore, it is recommended that you use InstallScript Scan to convert an InstallScript MSI installation to a Basic MSI package with InstallScript support. InstallScript Scan preserves the original components and much of the InstallScript installation logic, architecture, and maintainability of the original installation package.

***Note** • If you want to convert an InstallScript MSI package to a Basic MSI package that preserves the InstallScript installation logic, and you are using the Snapshot method, you must select Single Step rather than Multiple Steps. If you select Multiple Steps, the InstallScript installation logic will not be preserved.*

Task: ***To convert an InstallScript MSI Setup to a Basic MSI Setup with InstallScript support:***

1. Launch the **Repackaging Wizard** from Repackager. The **Welcome Panel** opens.
2. Click Next. The **Method Selection Panel** opens.
3. Select a **repackaging method**: Installation Monitoring or Snapshot.
4. Click Next. If you selected **Snapshot** on the **Method Selection Panel**, the **Snapshot Method Panel** appears. (If you selected **Installation Monitoring**, skip to Step 6.)
5. Select **Single Step** and click **Next**. The **Collect Product Information Panel** opens.

Caution • *Because you are converting an InstallScript MSI package to a Basic MSI package with InstallScript support, you must select the Single Step Snapshot method (or use the Installation Monitoring method). If you select Multiple Step Snapshot, the InstallScript installation logic will not be preserved.*

6. On the **Collect Product Information Panel**, select the InstallScript MSI setup file and enter other product information.

Caution • *While it is possible to click the Edit Setup List button and select additional setups, because you are converting an InstallScript MSI package, do not select additional setups.*

7. Click Next. Repackager will automatically determine if this is an InstallScript-based setup. If it is an InstallScript-based setup, the **InstallScript MSI Identified Panel** opens, informing you that the Repackaging Wizard has identified this setup as being an InstallScript MSI setup and prompting you to use InstallScript Scan to convert this setup.
8. Select Yes and click Next. The **InstallScript MSI Conversion Output Panel** opens.
9. In the Project Path to store files field, specify the location where you want the Repackaging Wizard to store files created during InstallScript Scan Analysis and where it will save the converted MSI package.

Note • *To specify capture types for the repackaging session, click the Edit button to access the* ***Analysis Options*** *dialog box.*

10. Click Next. The **Repackaging Panel** appears, displaying the progress of the repackaging operation.
11. Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful.
12. Click Finish to launch the Repackager to edit and build your project. See Working With Repackager Projects.

Running the Repackaging Wizard from the Command Line

To run the Repackaging Wizard from the command line, perform the following steps.

Task: ***To run the Repackaging Wizard from the command line:***

1. Open a command-line prompt.
2. Type `Repack.exe` followed by any command-line options you want to pass. See Repackaging Wizard Command-Line Options.
3. Press Enter.

An example of a typical command line is as follows:

```
Repack.exe -app Setup.exe -o C:\MyRepackagedApps\Output
     -pp SomeApp -cs Custom -cf MyOptions.ini -sb
```

In the above example, the following options are used:

Table 7-8 • Repackager Command-Line Options used in Example

Option	Description
-app	Specifies the name of the setup.
-o	Specifies the location of the output directory
-pp	Specifies the name of the product (and the name of the Repackager output file).
-cs	Specifies the name of the custom analysis options file to use.
-cf	Name of the analysis options file to use.
-sb	Allows you to run Repackager silently, with no user interaction.

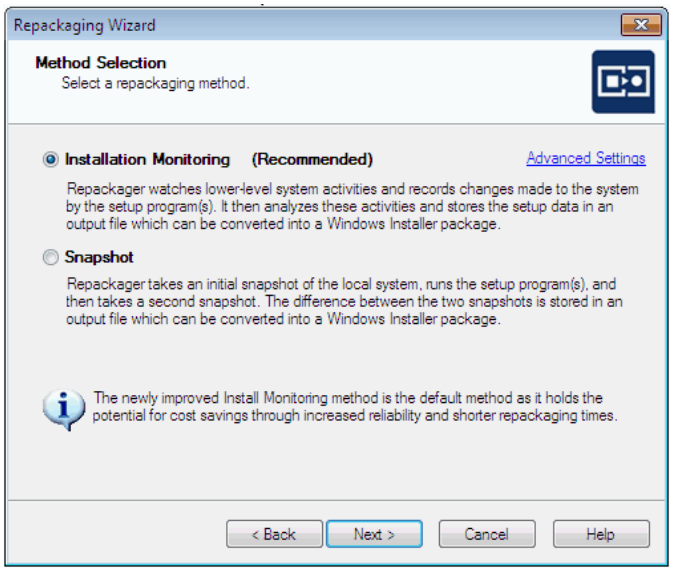
Repackaging a Windows Installer (.msi) Package

While it is not recommended that you repackage a Windows Installer (`.msi`) package, it sometimes may be necessary to repackage a Windows Installer package in order to convert it to a virtual package (perhaps due to the use of custom actions or other features that are not supported in application virtualization).

To repackage a Windows Installer (`.msi`) package, perform the following steps:

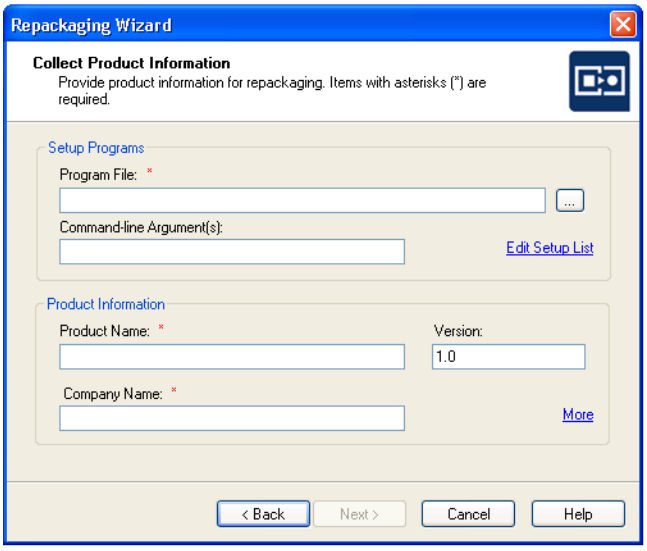
Task: ***To repackage a Windows Installer package:***

1. From the Repackager interface, launch the **Repackaging Wizard** by clicking on the link or by selecting **Repackaging Wizard** from the **Tools** menu. The Welcome Panel opens.
2. Click **Next**. The **Method Selection Panel** opens.

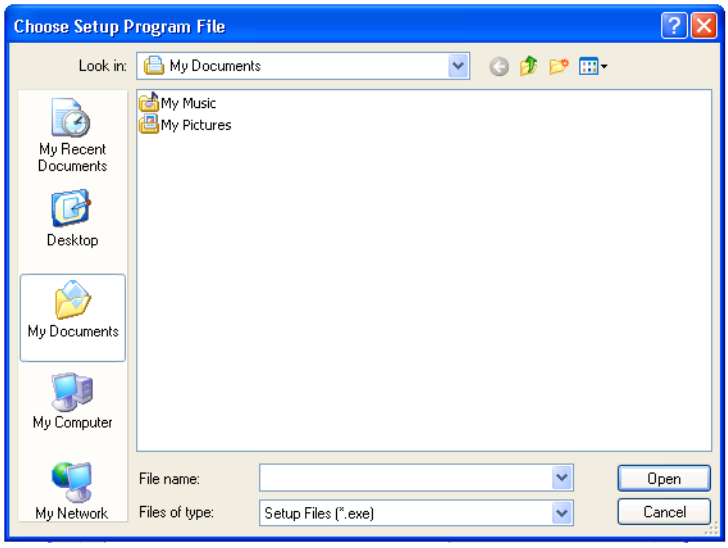


3. Select **Installation Monitoring** and click **Next**. The **Collect Product Information Panel** opens.

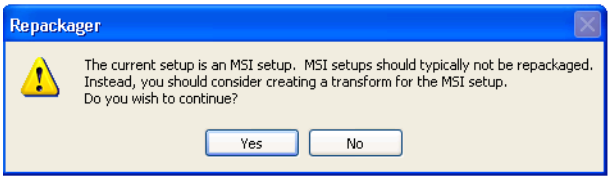
Note • *The **Installation Monitoring** method is recommended, but you may also choose the **Snapshot** method when repackaging a Windows Installer package. The **Installation Monitoring** method was used in the instructions that follow. For instructions on using the **Snapshot** method, see Repackaging Using the Snapshot Method*



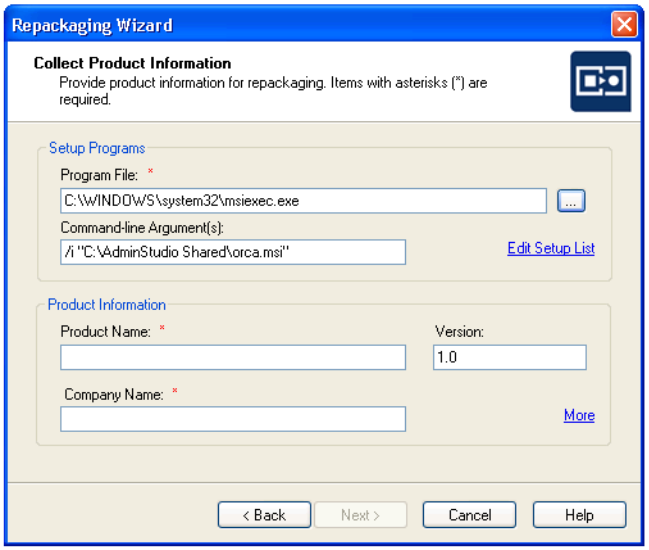
4. Click the Browse (...) button next to the **Program File** field to open the **Choose Setup Program File** dialog box.



5. From the **Files of type** list, select **All Files (*.*)**. All files in the selected directory are listed.
6. Click **Open** and select the Windows Installer package (.msi) that you are repackaging. A message appears warning you that MSI setups should not typically be repackaged.



7. Click **Yes** to close the message. Several fields in the Collection Product Information panel have been populated with the commands necessary to repackage a Windows Installer package.

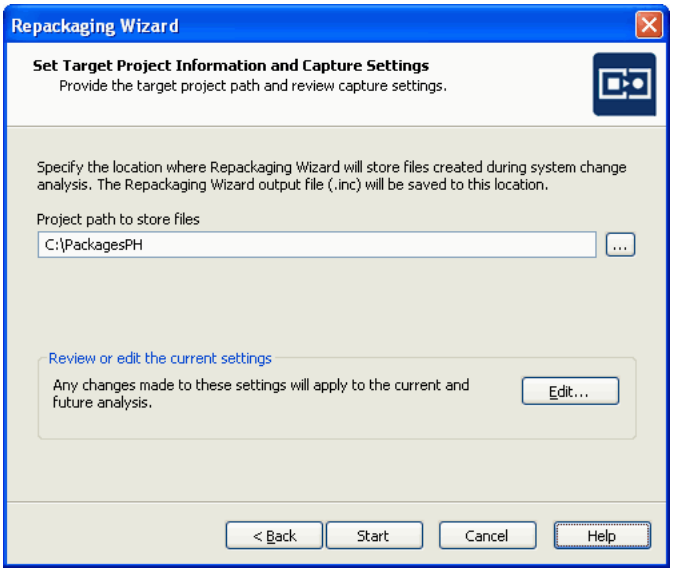


The following information was filled in:

Field	Entry
Program File	`C:\WINDOWS\system32\msiexec.exe`
Command line Argument(s)	`/i "C:\DIRECTORYPATH\PACKAGENAME.msi"`

Caution • *Do not edit the entries in the Program File or Command line Argument(s) fields.*

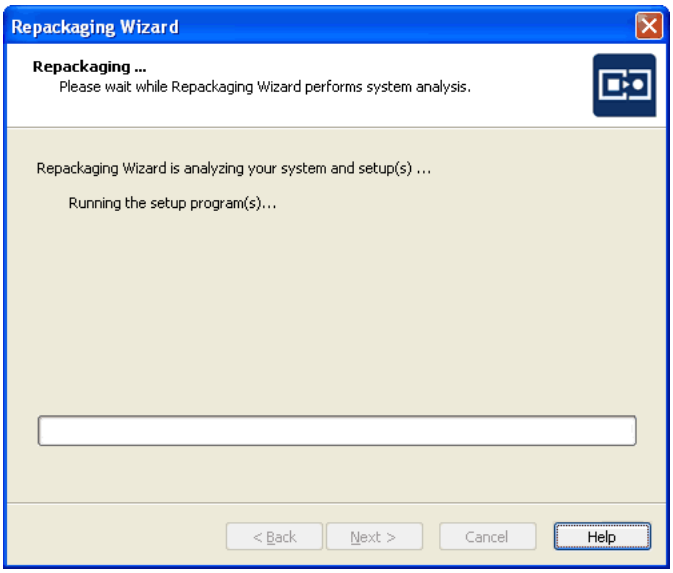
8. In the **Product Information** area, make entries in the **Product Name**, **Version**, and **Company Name** fields.
9. Click **Next**. The **Set Target Project Information and Capture Settings Panel** opens.



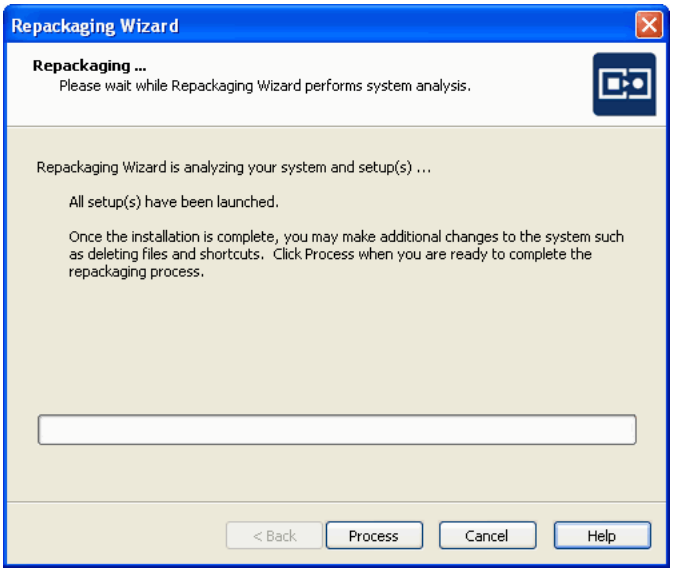
10. Click the Browse (...) button next to the **Project path to store files** field and select the directory where you want the Repackaging Wizard to place its output, including the Repackager project file (`.irp`), the Repackaging Wizard output files, and source files.

    You can also enter the name of a new folder in the **Project path to store files** field, and you will be prompted to create it when you exit this panel.

11. To begin the repackaging process, click **Start** on the Set Target Project Information and Capture Settings Panel. The **Repackaging Panel** opens and the Repackaging Wizard captures the initial system status. Then, the selected setup program will be launched.

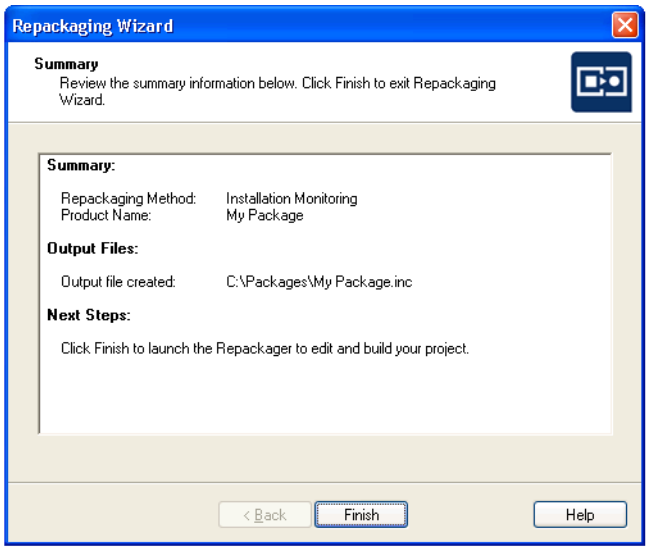


12. Follow the prompts until the installation has completed. When the installation is complete, you are prompted to make any additional changes to the system (such as deleting files and shortcuts) that you want to be recorded in this repackaged installation.



13. When you are ready to complete the repackaging process, click **Process**. The Repackaging Wizard then analyzes the system and setup data that it collected.

    Following repackaging, the **Summary Panel** is displayed, providing confirmation that the repackaging was successful.



14. Click **Finish**. Repackager launches and opens the Repackager project file (`*.irp`) that you just created.
15. Continue with the instructions in Working With Repackager Projects.

Repackaging Wizard Reference

This section describes each of the dialog boxes and Wizard panels that you might encounter when using the Repackaging Wizard. The help topics in the Repackager Reference are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog box.

Reference information is organized as follows:

Table 7-9 • Organization of Repackager Reference Section

Section	Description
Repackaging Wizard	This section provides a panel-by-panel description of the Repackaging Wizard.
Additional Repackaging Wizard Dialog Boxes	This section describes the dialog boxes that can be accessed from the Repackaging Wizard.
Repackaging Wizard Command-Line Options	This section lists the command-line options that are supported by the Repackaging Wizard.
Reboot Handling in the Repackaging Wizard	This section describes how the Snapshot Method and Installation Monitoring Method handle required reboots during repackaging.

Repackaging Wizard

Repackager provides the Repackaging Wizard to convert a legacy setup into a Repackager project. Using this Wizard, you can select the repackaging method (either Snapshot or Installation Monitoring), specify the setup(s) you want to repackage, and run the setup(s). When the Repackaging Wizard has finished its analysis, Repackager automatically creates a Repackager project (`.irp`) file, which can be modified in Repackager. You can then convert this file to an InstallShield Editor project (`.ism`) for further editing, or convert it directly to a Windows Installer package (`.msi`).

The Repackaging Wizard includes the following panels:

- Welcome Panel
- Method Selection Panel
- Snapshot Method Panel
- Collect Product Information Panel
- InstallScript MSI Identified Panel
- InstallShield Professional Setup Panel
- Set Target Project Information and Capture Settings Panel
- InstallScript MSI Conversion Output Panel
- Repackaging Panel
- Summary Panel
- Additional Repackaging Wizard Dialog Boxes

Welcome Panel

The Welcome panel appears when you first launch the Repackaging Wizard, providing some introductory information about the use of the Wizard, including that it is for use with traditional (non-Windows Installer-based) installations.

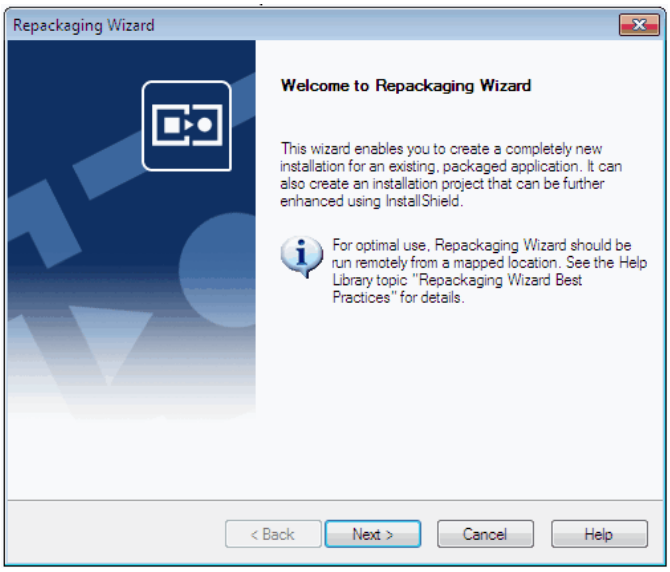


Figure 7-1: Repackaging Wizard Welcome Panel

Method Selection Panel

From the Method Selection panel, select the method(s) you want to use for repackaging.

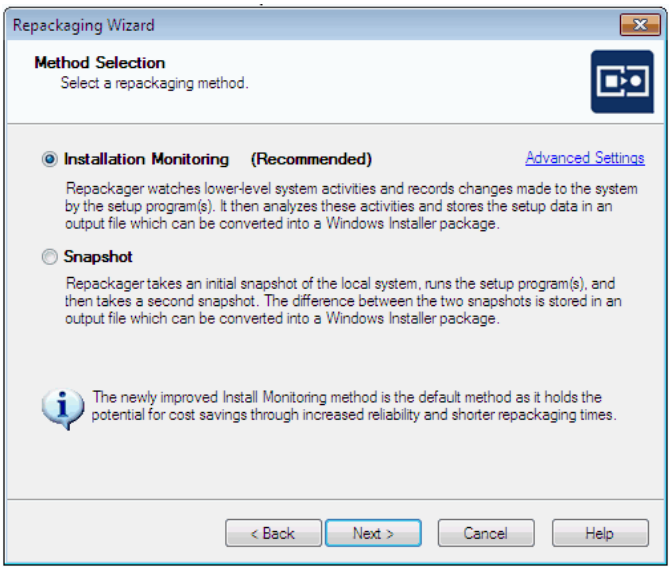


Figure 7-2: Repackaging Wizard Method Selection Panel

The available choices include:

Table 7-10 • Method Selection Panel Options

Options	Description
Snapshot	The Snapshot method involves taking system snapshots before and after an installation, and then creating the Windows Installer package from the difference between them. Any configurations you make between snapshots is also included in the generated Windows Installer package.
Installation Monitoring	Installation Monitoring watches all activities generated by an installation, and then determines the files, .ini files, registry entries and shortcuts that should be included in the generated Windows Installer package. Installation Monitoring is significantly faster than the Snapshot repackaging method. If there are services running on the machine that have nothing to do with the installation being repackaged, click the **Advanced Settings** link to open the Excluded Processes Dialog Box, where you can choose to exclude those processes. ***Tip*** • *If you know that the installation that you are capturing is from a self-extracting* `.exe` *file and if you want to use the* ***Installation Monitoring*** *method, you should click* ***Advanced Settings*** *and add the name of that* `.exe` *file to the excluded processes list.*

Using the InstallShield Professional Logging Method

The InstallShield Professional Logging Method, which should be used to repackage InstallShield Editor and DevStudio 9.x InstallScript installations, is not offered as a choice on the **Method Selection** panel.

However, if you specify an InstallShield Editor or DevStudio 9.x InstallScript installation on the **Collect Product Information** panel (the next panel in the Repackaging Wizard), Repackager will automatically detect the InstallScript installation and will display the **InstallShield Professional Setup** panel. On this panel, you can choose to use the **InstallShield Professional Logging Method** instead of the **Installation Monitoring** or **Snapshot** methods.

It does not matter which option you select on the **Method Selection** panel, but if you select the **Snapshot** method, you must select **Single Step** on the **Snapshot Method** panel (which appears next). If you instead select **Multiple Steps**, Repackager will not recognize the setup as an InstallShield Editor or DevStudio 9.x InstallScript installation and the **InstallShield Professional Setup** panel will not appear.

System Changes Captured by Repackager

Regardless of the repackaging method used, Repackager captures system changes made to the following:

- Application Paths
- Environment Variables

- Files
- INI Files
- NT Services
- ODBC Data Sources
- ODBC Drivers
- Printer Drivers
- Registry Entries
- Shortcuts

If you are using the InstallShield Professional Logging Method to repackage an InstallShield Editor or DevStudio 9.x InstallScript installation, the following additional information is also collected:

- Path Variables
- Feature/Component Tree

Snapshot Method Panel

The Snapshot Method Panel, which is only displayed if you use the snapshot technology, allows you to specify the way in which you perform repackaging.

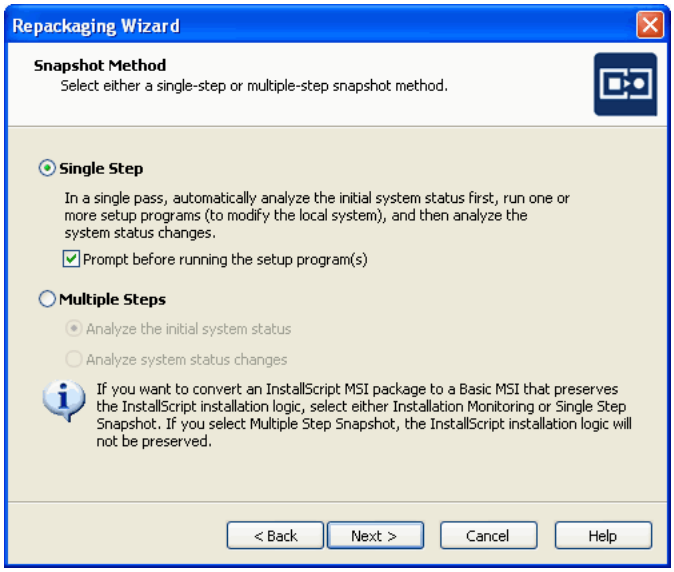


Figure 7-3: Repackaging Wizard Snapshot Method Panel

On the Snapshot Method Panel, you have the following two options:

Table 7-11 • Snapshot Method Panel Options

Option	Description
Single Step	Repackaging in a single step requires you specify at least one setup program to repackage. The Repackager first takes an initial system snapshot, then runs the setup program(s) you specify, and then takes a second snapshot to create the script file that can be converted into a Windows Installer package. You also have the option of requiring the Repackager to prompt you before running the setup program(s), allowing you the opportunity to make changes to your system that you want included in the final package. ***Note*** • *If you are repackaging an InstallShield Editor or DevStudio 9.x InstallScript installation and want to use the InstallShield Professional Logging Method, select* ***Single Step****.*
Multiple Steps	Repackaging in multiple steps allows you to run the Repackager to obtain an initial system snapshot, after which the Repackager exits. You can then perform any modifications to the system, such as changing configurations, running installations, and so forth. After making the necessary modifications, run the Repackager again to analyze system status changes. The difference between the second Repackager execution and the first results in the script file that ultimately can be converted into a Windows Installer package.

The single step method is very straightforward if you are repackaging applications and not performing many system changes. The multiple step method allows greater flexibility because a setup is not required. This allows you to capture system configurations within the Repackager output, and ultimately within a Windows Installer package. For example, you could modify the screen color depth and create an MSI package for just that configuration.

If Single Step is selected, the **Collect Product Information Panel** is displayed when you click Next. If Multiple Steps is selected and you are performing the initial snapshot, the Collect Product Information panel is displayed, but the Setup Programs area is disabled. If you are performing a system status change analysis, the **Repackaging Panel** appears when you click **Next**.

Collect Product Information Panel

The Collect Product Information panel gathers information necessary for repackaging the installation(s).

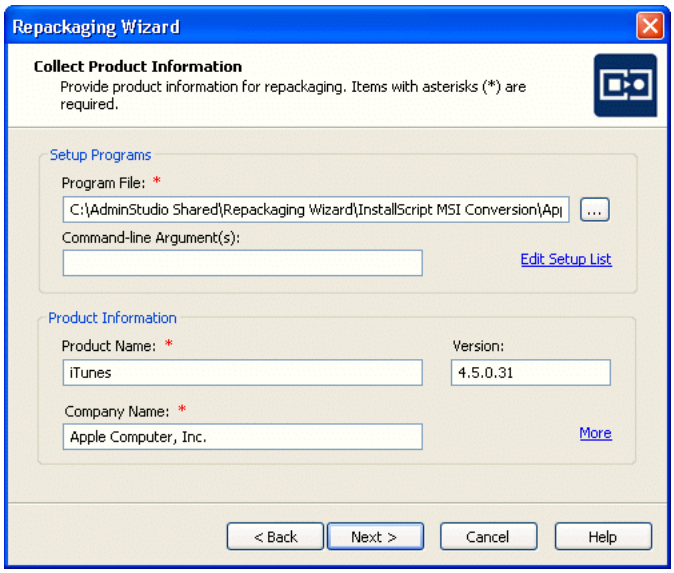


Figure 7-4: Repackaging Wizard Collect Product Information Panel

The information on the Collect Product Information Panel is divided into two sections: **Setup Programs** and **Product Information**.

Setup Programs Area

The Setup Programs area contains information about the setup you are repackaging. Repackager uses this information to launch the setup correctly following pre-analysis. The information collected includes:

Table 7-12 • Setup Programs Options

Properties	Description
Program File	The name and location of the setup executable. Click the Browse ... button to locate this file. This is a required field.
Command-Line Argument(s)	Any command-line arguments to be used when the setup is run.

Table 7-12 • Setup Programs Options (cont.)

Properties	Description
Edit Setup List	Click to display the **Additional Setup Programs** dialog box, from which you can enter additional installations to repackage together with this installation. Additional setups share the same product name, version number, and company name in the repackaged installation. However, as you locate each additional setup to repackage, you can specify command-line parameters pertaining only to that setup. You can also specify the order in which the installations are run, should it be necessary. ***Caution*** • *If you are repackaging an InstallShield Editor or DevStudio 9.x InstallScript installation and want to use the InstallShield Professional Logging Method, specify only one Program File. If you specify multiple setup program files, the* ***InstallShield Professional Setup Panel*** *will not appear, and the Repackaging method that you chose on the* ***Method Selection Panel*** *will be used instead.*

Product Information Area

In the Product Information area, you identify the repackaged installation's **Product Name**, **Version Number**, and **Company Name**.

Table 7-13 • Product Information Options

Field	Description
Product Name	Enter the name for final repackaged installation. This could be the name of the original installation (for example, Tuner), the name of a collective group of products (for example, Microsoft Applications), or another name of your selection (for example, My Apps). This is a required field.
Version Number	Enter the version of the product.
Company Name	Enter the name of the company.

Product Support Information

If you want to associate Web sites with this installation, click the **More** link in the Product Information area to open the Additional Product Information dialog box, where you can enter the **Product URL** and **Support URL** for the application you are repackaging.

InstallScript MSI Identified Panel

This panel opens if the Repackaging Wizard identifies an installation as an InstallScript MSI installation created with InstallShield Editor, InstallShield DevStudio, or InstallShield Developer.

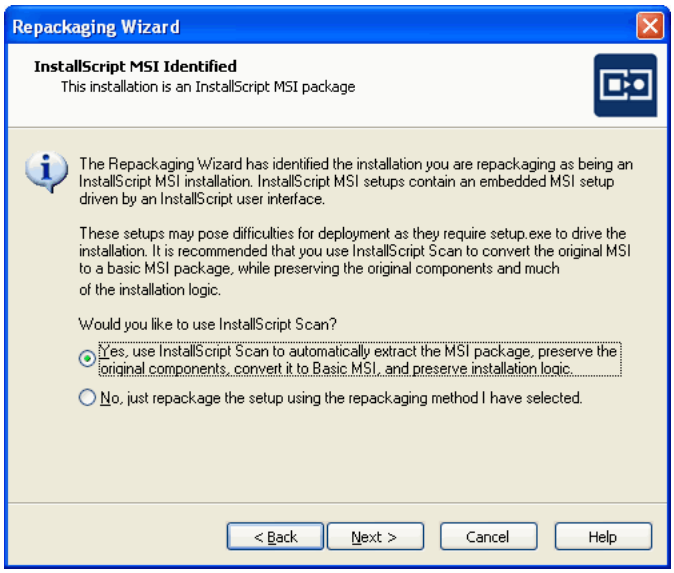


Figure 7-5: Repackaging Wizard InstallScript MSI Identified Panel

InstallScript MSI installations use a Windows Installer database for storage of all file/registry information, but the actual user interface, and much of the installation logic is driven by the InstallScript engine via a `setup.exe` file. This type of installation architecture can cause difficulties during deployment, such as:

- inability to customize or transform the application
- inability to conflict detect
- inability to suppress the user interface
- difficulty patching or upgrading the application

Also, if an InstallScript MSI installation is repackaged using traditional methods (OS Snapshot or Installation Monitoring), significant platform-specific or custom installation, maintenance, and uninstallation logic, and user interface information is lost because those methods only record the installation activities for the specific platform used during repackaging.

Therefore, it is recommended that you use InstallScript Scan to convert an InstallScript MSI installation to a Basic MSI package with InstallScript support. InstallScript Scan preserves the original components and much of the InstallScript installation logic, architecture, and maintainability of the original installation package.

Select one of the following options:

- **Yes**—Use InstallScript Scan to automatically extract the MSI package and convert it to Basic MSI, while preserving the original components and installation logic. This is the default selection.
- **No**—Repackage the installation using the repackaging method selected on the **Method Selection Panel** (Installation Monitoring or Snapshot).

Click **Next** to proceed.

InstallShield Professional Setup Panel

This panel appears in the Repackaging Wizard if you specified an InstallShield Editor or DevStudio 9.x InstallScript installation in the **Program File** field of the Collect Product Information Panel, and if:

- You specified only one program file on the Collect Product Information Panel, *and*
- You selected the **Installation Monitoring** method on the Method Selection Panel, *or*
- You selected the **Snapshot** method on the Method Selection Panel and then selected **Single Step** on the Snapshot Method Panel.

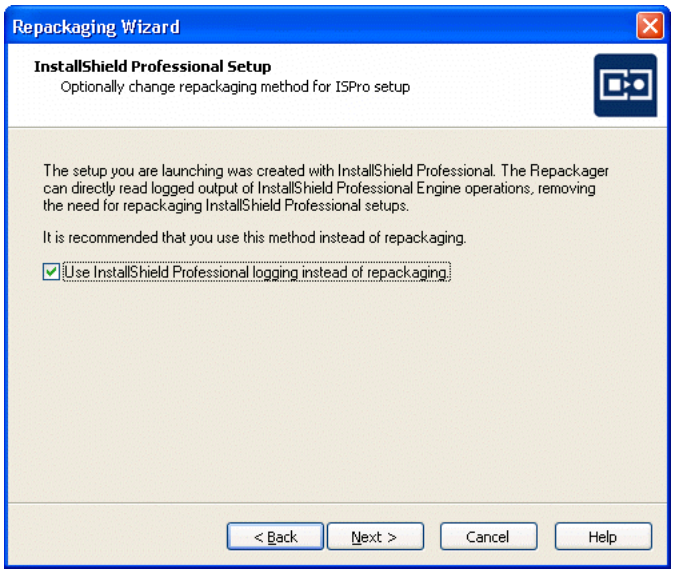


Figure 7-6: Repackaging Wizard InstallShield Professional Setup Panel

If you want to use the InstallShield Professional Logging Method, select the Use InstallShield Professional Logging instead of repackaging check box.

Using the InstallShield Professional Logging Method, Repackager can read logged output of InstallShield Editor and DevStudio 9.x InstallScript installations. This method replaces the other repackaging methods (Installation Monitoring and Snapshot) for InstallShield Editor or DevStudio 9.x InstallScript installations. By using this method, you will be able to get additional information that would not be captured by repackaging, including path variables and a feature tree. For more information, see InstallShield Professional Logging Method.

Set Target Project Information and Capture Settings Panel

The location where you want files created by Repackager stored is defined in the **Project path to store files** field on the **Set Target Project Information Panel**.

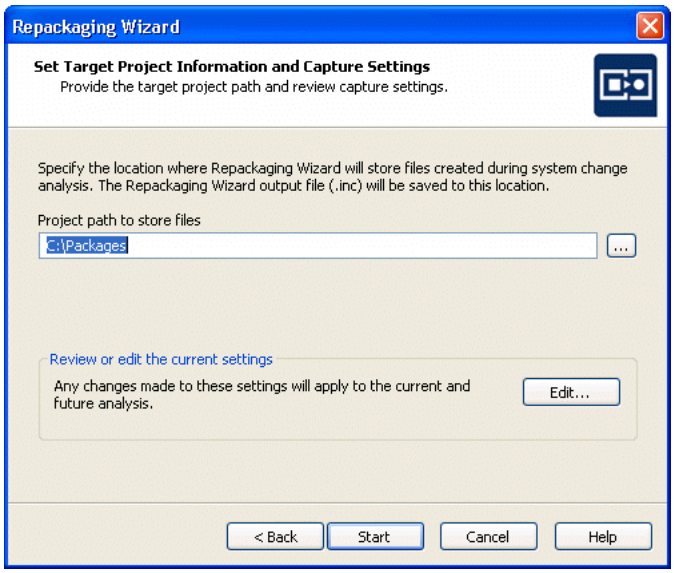


Figure 7-7: Repackaging Wizard Set Target Project Information and Capture Settings

It is recommended that this location not be located on your clean machine, but rather on the same machine as the Repackager executable (most likely on your administrator machine).

You can also review or edit current capture settings by clicking Edit, which displays the **Analysis Options** dialog box.

Click **Start** to begin repackaging and display the **Repackaging Panel**.

InstallScript MSI Conversion Output Panel

On this panel, specify the location where you want Repackager to store the files it creates during InstallScript Scan analysis. The converted Windows Installer MSI package will be saved to this location.

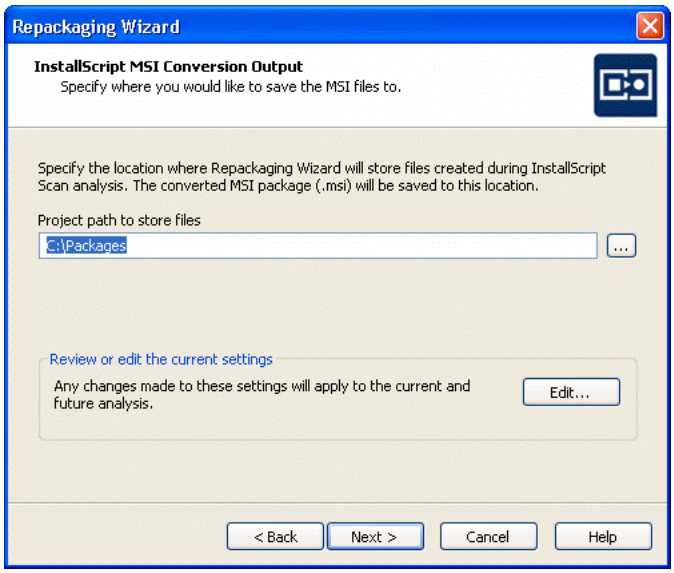


Figure 7-8: Repackaging Wizard InstallScript MSI Conversion Output Panel

It is recommended that this location not be located on your clean machine, but rather on the same machine as the Repackager executable (most likely on your administrator machine).

You can also review or edit current settings by clicking **Edit** to open the **Analysis Options** dialog box. On the **Analysis Options** dialog box, you can specify capture types for the repackaging session, and, for snapshot-mode captures, you can restrict directory analysis to specific directories.

Click **Start** to begin repackaging and display the **Repackaging Panel**.

Repackaging Panel

The Repackaging panel appears while Repackager analyzes your system.

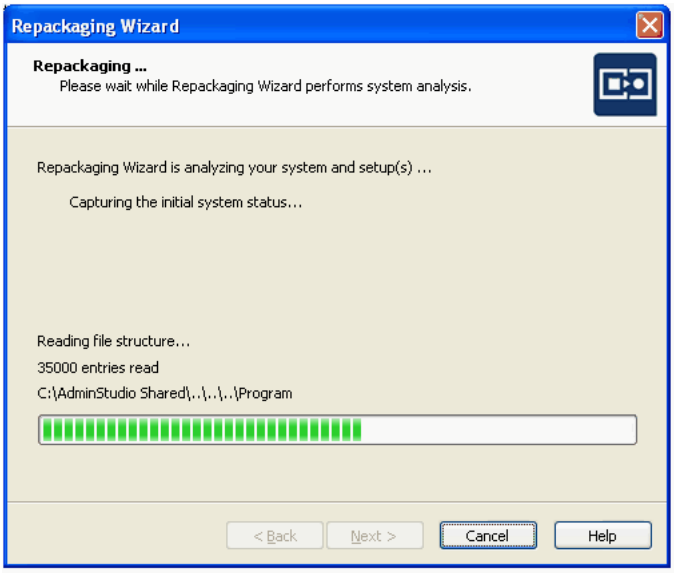


Figure 7-9: Repackaging Wizard Repackaging Panel 1

Depending on settings configured before starting repackaging, the analysis may stop following the initial phase, and again after setup has been run.

After the setups have been completed, you are prompted to click the **Process** button to complete the repackaging process.

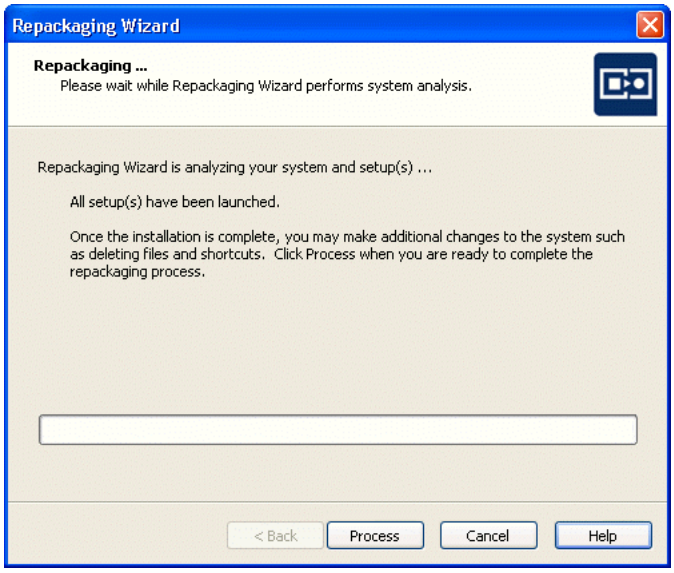


Figure 7-10: Repackaging Wizard Repackaging Panel 2

When you click **Process**, the repackaging is performed and its progress is displayed.

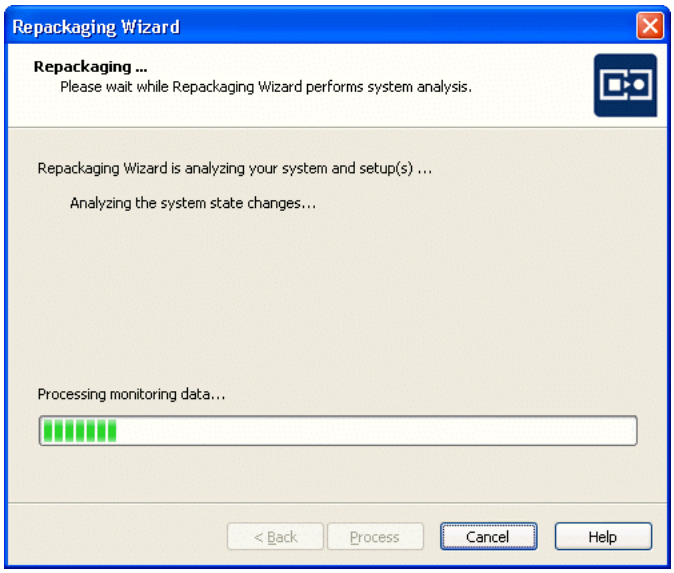


Figure 7-11: Repackaging Wizard Repackaging Panel 3

Following repackaging, the **Summary Panel** is displayed.

Summary Panel

The final panel displayed by Repackager is the Summary panel.

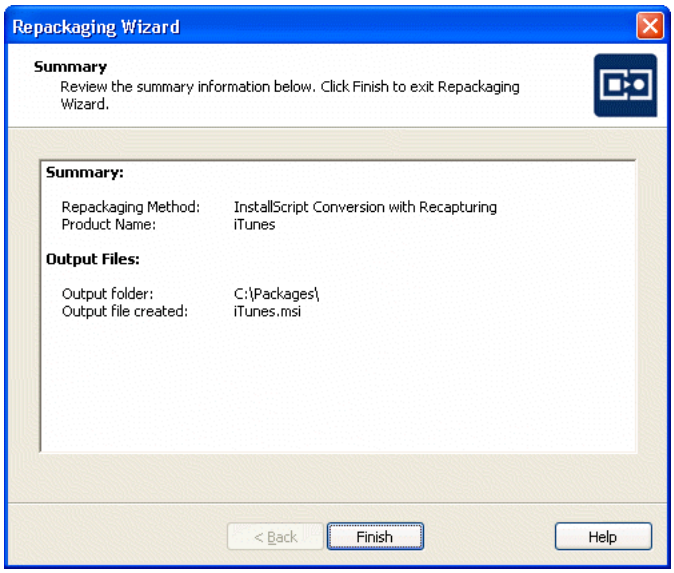


Figure 7-12: Repackaging Wizard Summary Panel

This panel provides confirmation that repackaging was successful, and provides the location of the source setup program(s), the Windows Installer package, and the InstallShield Editor project.

Additional Repackaging Wizard Dialog Boxes

The following dialog boxes can be accessed from the Repackaging Wizard:

- Additional Setup Programs Dialog Box
- Setup Information Dialog Box
- Excluded Processes Dialog Box
- Analysis Options Dialog Box

Additional Setup Programs Dialog Box

This dialog box, which is accessed by clicking the **Edit Setup List** button on the **Collect Product Information Panel** of the Repackaging Wizard, displays a list of additional setup programs you want to add to the final Windows Installer package.

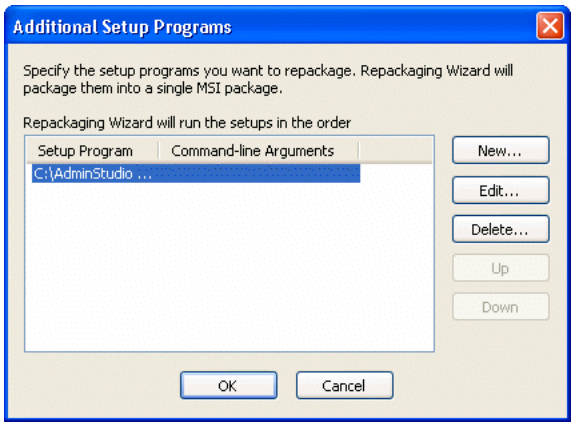


Figure 7-13: Repackaging Wizard's Additional Setup Programs Dialog Box

Essentially, this is a list of the other executables to run, in the order they are to be run, prior to final analysis. The following buttons are available:

Table 7-14 • Additional Setup Programs Dialog Box Buttons

Button	Description
New	Brings up the **Setup Information** dialog box to enter information about the setup programs.
Edit	Displays the **Setup Information** dialog box to edit information about the currently selected setup.
Delete	Removes the currently selected setup.
Up	Moves the selected setup up in the setup programs list.

Table 7-14 • Additional Setup Programs Dialog Box Buttons (cont.)

Button	Description
Down	Moves the selected setup down in the setup programs list.

Setup Information Dialog Box

The **Setup Information** dialog box allows you to enter or edit information pertaining to the installations you are repackaging.

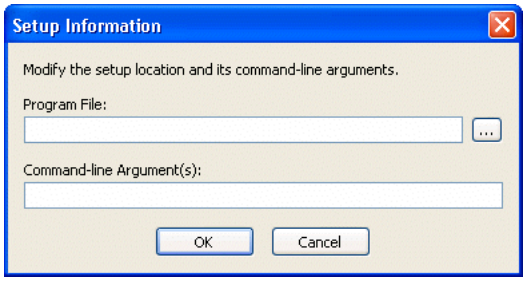


Figure 7-14: Setup Information Dialog Box

Accessible from the **Additional Setup Programs** dialog box, you can provide the name and location of an additional setup program, and any command-line arguments for the setup.

Excluded Processes Dialog Box

During Installation Monitoring, Repackager captures all of the activity of each service or process running on the machine, and then processes this collected data. However, many services running on a machine may have nothing to do with the installation being repackaged. Therefore, you may choose to exclude those processes by adding them to the list on the Excluded Processes dialog box.

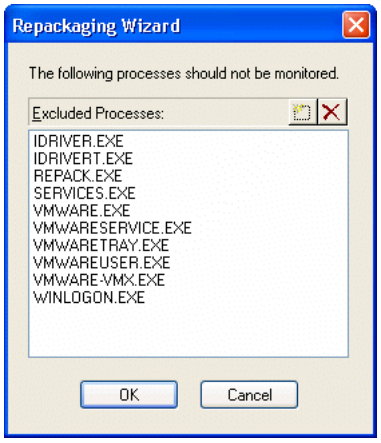


Figure 7-15: Repackaging Wizard Excluded Processes Dialog Box

You can open the Excluded Processes dialog box by clicking the **Advanced Settings** link on the Repackaging Wizard Method Selection Panel. The Excluded Processes dialog box initially lists a default set of processes.

- **To add a process to this list**, click the New ( ) button to add a new blank line to this list, and enter the name of the process that you want to exclude.
- **To delete a process from this list**, select the process and click the Delete ( ) button.

Analysis Options Dialog Box

The Analysis Options dialog box, accessible by clicking Edit from the **Set Target Project Information and Capture Settings Panel** or the **InstallScript MSI Conversion Output Panel**, allows you to specify capture types for the repackaging session.

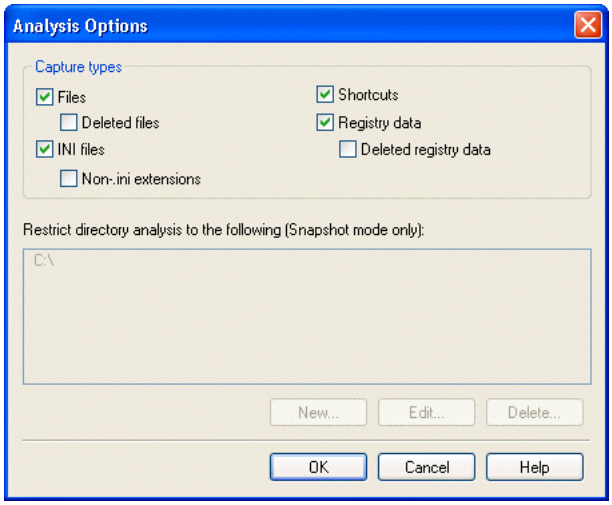


Figure 7-16: Analysis Options Dialog Box

You can select the following:

- Files
- Deleted files
- .ini files
- .ini files with non-.ini extensions
- Shortcuts
- Registry data
- Deleted registry data

Additionally, for snapshot-mode captures, you can restrict directory analysis to specific directories, which can significantly improve repackaging performance. Click **New** to add a directory restriction, **Edit** to modify an existing restriction, or **Delete** to remove a restriction.

Options set in this dialog box apply to the recurrent and subsequent repackaging sessions.

Repackaging Wizard Command-Line Options

The following command-line options are supported by the Repackaging Wizard:

Table 7-15 • Repackaging Wizard Command-Line Options

Option	Description
`-?`	Displays a dialog box containing usage information for all Repackager command line options: • **If a option name is provided**, detailed help for the specified option will be displayed. • **If no option name is provided,** a dialog box containing general usage information for all options is displayed.
`-app <setup program list>`	Enables you to provide a pipe (|) delimited list of setups to run during repackaging. You can also pass command-line arguments to the setup by separating them from the setup name with a semicolon. If entering multiple setups, proper double quoting—including escaping nested quotes—is necessary: `-app "\"exe1path\";cmdline1|\"exe2path"|...exeN; cmdlineN`
`-b [<project file path and name>.ism]`	Specifies the build/release destination folder. You can, optionally, specify a location for the `.ism` file that will be created. If this is not specified, then the `.ism` and `.msi` files will be created in a sub-folder of the Repackager project files location called `MSI_Package`. The project name must end in `.ism`. ***Note*** • *In addition, the `-o` and `-pp` options should, at a minimum, be specified to allow for the location of the Repackager project files.*

Table 7-15 • Repackaging Wizard Command-Line Options (cont.)

Option	Description
`-buildonly`	If this "build MSI only" command line option is used, the Repackaging Wizard will skip the repackaging step and just build a Windows Installer package based on existing Repackager project files (`.inc`, `.irp`, etc.). • The `-b` option, which specifies the build/release destination folder, also needs to be used in conjunction with the `-buildonly` option. • The `-o` option, which specifies a folder path not including the filename, and the `-pp` option, which sets the product name, also need to be used in conjunction with the `-buildonly` option. The following is a sample command line that uses the `-b`, `-o`, `-pp`, and `-buildonly` command line options: `repack.exe -b [optional path to ISM file to create] -o <required path to folder of existing Repackager project files> -pp <required name of product - should match existing INC and IRP file names> -buildonly [optional -bv parameter] [-sb or -sn]`
`-bv appv`	Build App-V application.
`-bv thinapp`	Build ThinApp application.
`-bv citrix`	Build Citrix profile.
`-cf <config.ini>`	This option allows you to select your own configuration template containing exclusions. A sample of this type of file (named `Default.ini`) can be found in the following directory: *`AdminStudio Installation Directory`*`\Repackager` This particular file contains the default exclusion information.
`-cs <configuration type>`	This option allows you to select the configuration file type for exclusions. Possible values are: • **Shared**—Use shared settings from those stored in the `AdminStudio Shared` directory. • **Custom**—Use a custom configuration file (in conjunction with `-cf`).

Table 7-15 • Repackaging Wizard Command-Line Options (cont.)

Option	Description
`-is`	Regarding the Repackaging an InstallScript MSI Setup to a Basic MSI Setup procedure, use this parameter in the command line using the following syntax: `Repack.exe -app "c:\setup.exe" -o C:\apps\output -mm -is` In the above example, the user wants to repackage `c:\setup.exe` using the Installation Monitoring repackaging method (as specified by `-mm`) and InstallScript conversion (as specified by `-is`). Repackager would perform the InstallScript conversion process and produce a Basic MSI package with InstallScript support as output. Without the -is parameter, Repackager would perform repackaging without performing InstallScript conversion, and would only create a Repackager `.inc` file as the output. ***Note*** • *The command line parameter -is will be considered only if the setup to be repackaged is a InstallScript MSI setup. If user specified any other legacy setup that is not a InstallScript MSI setup then -is will be ignored.* ***Note*** • *If user chooses to use the Multiple Step Snapshot repackaging method, then the -is parameter will be ignored. Even if the setup is an InstallScript MSI setup, -is will still be ignored when using the Multiple Step Snapshot repackaging method.*
`-mode <snapshot mode>`	Repackager supports the following repackaging modes for snapshots: • **single**—Single step repackaging that creates an INC file as its output. • **pre**—Pre-scanning only scans the local drive for a baseline snapshot of the system. • **post**—Post-scanning only scans the local drive and compares the result with the pre-scan. The differences are written to the INC file as output.
`-mm`	Instructs Repackager to use installation monitoring as the repackaging technology.
`-mp`	Instructs Repackager to use the InstallShield Professional Logging Method as the repackaging technology.
`-ms`	Instructs Repackager to use snapshots as the repackaging technology.
`-o <.inc path name>`	Specifies a folder path not including the filename. The file name is derived from the Product Name unless overridden with the -of switch.
`-of <inc file name>`	Specifies the .inc file name that should be used instead of the product name. Use -o to specify the path.

Table 7-15 • Repackaging Wizard Command-Line Options (cont.)

Option	Description
`-onp`	When using the **Installation Monitoring** method via command line to perform repackaging on a 64-bit operating system, you can use the `-onp` command line option to cause the **Installation Monitoring** method to only monitor new processes created on the system and to ignore any existing/ running ones. This option is useful to optimize the monitoring process on a 64-bit operating system.
`-pc <company name>`	Allows you to set the company name.
`-pp <product name>`	Allows you to set the product name. This will be the same name as the generated Repackager output file (`.inc`).
`-pv <product version>`	Allows you to set the product version.
`-sb`	This option allows you to run Repackager silently, with no user interaction. A progress dialog box is displayed. If no .ini file is specified using the -i parameter, Repackager uses `Repack.ini` as the default input file. If an output folder is not specified using -o, the default output folder is `C:\Packages`.
`-sn`	This option allows you to run Repackager silently, with no user interaction and no progress dialog box. If no .ini file is specified using the -i parameter, Repackager uses `Repack.ini` as the default input file. If an output folder is not specified using `-o`, the default output folder is `C:\Packages`.
`-version`	This option displays standard version information for Repackager, including the full version and copyright information.

Tip • *To open a help topic from the command line that lists command line options, enter the following:*

```
repack.exe /?
```

Note • *In addition to the - sign for command-line arguments, you can also use the / symbol.*

Reboot Handling in the Repackaging Wizard

During repackaging, a setup may require a reboot. For example, some operations may require a file which is in use be replaced, which can only be done after a reboot. Some nuances exist depending on the repackaging technology you are using (Snapshot or Installation Monitoring). In either case, when the Repackaging Wizard detects that a reboot is necessary, the Repackaging Wizard saves the appropriate data and waits until you confirm that you are ready to reboot the machine.

For Snapshot repackaging, the operating system completes the reboot operation. During startup, the operating system restarts all applications and processes and performs any pending file operations. One of the applications that restarts is Repackager. Before you continue processing in Repackager, be patient and ensure all processes and applications have restarted. This may take a minute or two. After the applications and processes have been launched, you can continue repackaging by clicking Process.

For Installation Monitoring, on reboot the operating system launches the Repackaging Wizard, which in turn launches applications and processes and waits until these are finished before prompting you to continue repackaging. However, in some cases the processes or applications launched by the Repackaging Wizard will launch other applications and processes. As in Snapshot repackaging, it is generally a good idea to wait a minute or two before clicking Process.

In both circumstances, waiting helps ensure the setup is fully installed and that captured data contains the necessary information to properly rebuild the setup as an MSI installation.

***Note** • On Windows Vista and newer, system reboots are almost instantaneous and do not allow running applications to properly shut down, which may result in a loss of data. When using the **Installation Monitoring** method, Repackager successfully handles a system reboot and delays it until you click the **Reboot** button on the Repackaging Wizard.*

8

Converting Legacy Installations Using the Repackager Interface

A Repackager project file (.irp) can be built into an InstallShield Editor project (.ism) or a Windows Installer package (.msi). You can use the Repackager interface to create and modify Repackager project files. You can also use it to build an isolated Windows Installer package and to configure the exclusions used when repackaging a legacy installation.

Information about the Repackager interface is presented in the following sections:

Table 8-1 • Using the Repackager Interface

Section	Description
About the Repackager Interface	Explains how to launch the Repackager interface and how to set options.
Creating Repackager Projects	Explains how to create a Repackager project file (.irp), which can then be built into an InstallShield Editor project (.ism) or a Windows Installer package (.msi).

Table 8-1 • Using the Repackager Interface (cont.)

Section	Description
Working With Repackager Projects	Explains how to build an InstallShield Editor project and Windows Installer package from a Repackager project. The topics in this section include: • Building an InstallShield Editor Project • Building a Windows Installer Package • Automatically Generating a Virtual Application During Repackager Project Build • Viewing Repackager Project Properties • Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project • Creating a Setup Capture Report for a Project • Saving Repackager Projects • Opening InstallShield Editor from Repackager
Isolating Windows Installer Packages	Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested. This section reviews isolation concepts and options, and explains how to build an isolated Windows Installer package.
Configuring Exclusions	Explains how to use Repackager and the Exclusions Editor to configure the exclusions used when repackaging a legacy installation.
Scanning InstallShield Professional Setups for Additional Data	Explains how to scan InstallShield Professional setups for possible additional files.
Creating an InstallShield Editor Template to Use Within Repackager	Explains how to create an InstallShield Editor template to use to speed up the Repackaging process.
Repackager Interface Reference	Describes each of the views and dialog boxes that you might encounter when using the Repackager interface. The help topics in this section are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog box.

Note • *For information on other Repackager features, see Repackaging Legacy Installations Using the Repackaging Wizard.*

About the Repackager Interface

Information about using the Repackager interface is presented in this section:

- Launching the Repackager Interface
- Setting Repackager Options

Launching the Repackager Interface

Repackager can be launched from within the AdminStudio interface. Additionally, if you install Repackager on a network, use Windows Explorer to browse to the `islc.exe` executable on the shared drive.

Task: ***To launch Repackager from the AdminStudio interface:***

1. Launch AdminStudio.
2. Click the **Tools** tab.
3. From the Tools Gallery, click the **Repackager** icon on the left side.

The Repackager Start Page opens and you can begin the repackaging process.

Note • *You can also launch Repackager directly from the Windows* ***Start*** *menu by pointing to* ***All Programs****, AdminStudio,* ***AdminStudio 10.0 SP2 Tools****, and clicking* ***Repackager****.*

Caution • *It is highly recommended that you repackage applications on a "clean" system. See Configuring Repackager to Ensure Optimal Installation Capture for more information.*

Setting Repackager Options

On the Options Dialog Box, which is opened by selecting **Options** from the **Tools** menu, you can specify the following Repackager options:

- Selecting Data Display Colors
- Specifying Additional Merge Module Directories
- Controlling the Display of ICE Validation Warnings

Selecting Data Display Colors

On the **Colors** tab of the Repackager Options dialog box, you can configure the color of scanned items and deleted items in Repackager's exclusion views (Files, `.ini` Files, Registry Data, and Shortcuts).

Task: ***To change the way excluded and included data is displayed in Repackager:***

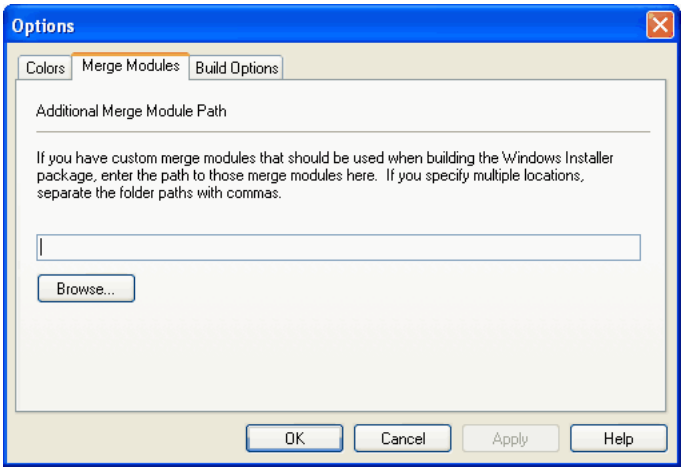
1. Open the Repackager interface.
2. From the **Tools** menu, select **Options**. The **Colors** tab of the **Options** dialog box opens.
3. Configure the display colors for **Excluded**, **SmartScan**, and **Setup Intent** items.
4. Click OK.

Specifying Additional Merge Module Directories

If you have custom merge modules that should be used when building a Windows Installer package, you need to specify the directories that contain those custom merge modules on the **Merge Modules** tab of the Options dialog box.

Task: ***To specify directories of additional Merge Modules:***

1. Open the Repackager interface.
2. From the Tools menu, select **Options**. The **Colors** tab of the Options dialog box opens.
3. Open the **Merge Modules** tab.



4. Enter the directory paths to the custom merge modules. To specify multiple directories, separate the folder paths with commas.

Note • *You can click* ***Browse*** *and navigate to a directory, but if you browse to a second directory, its directory path will replace the one you initially selected. Therefore, if you want to specify multiple directories separated by commas, you need to manually enter the directory paths.*

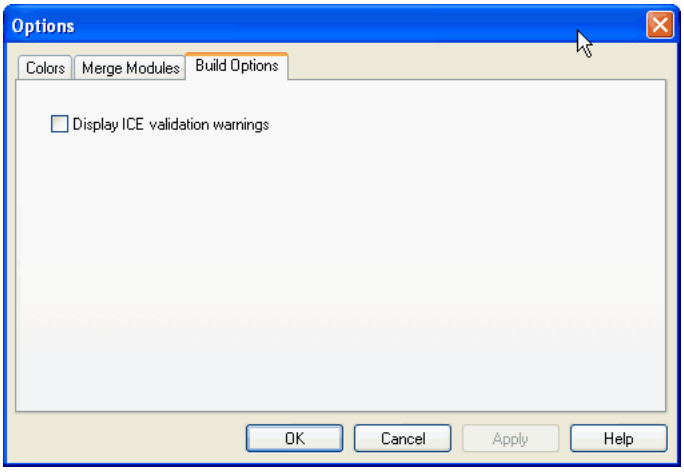
5. Click **OK**.

Controlling the Display of ICE Validation Warnings

On the **Build Options** tab of the Options Dialog Box, you can specify whether or not you want to list ICE validation warnings in the Repackager output window during the Build process.

Task: ***To set the display of ICE validation warnings during builds:***

1. From the Repackager interface, select **Options** from the **Tools** menu. The Options dialog box opens.
2. Open the **Build Options** tab.



3. To display any ICE validation warnings that occur during the Repackager Build process, select the **Display ICE validation warnings** option. By default, this option is not selected.

Creating Repackager Projects

Repackager projects (`.irp`) allow you to visually analyze the files, `.ini` files, shortcuts, and registry entries captured or changed during the conversion of a legacy setup into a Windows Installer package. You can also exclude files, shortcuts, registry entries, and `.ini` files from the resulting Windows Installer package, without affecting the original setup data.

There are two methods of creating Repackager projects:

Table 8-2 • Methods of Creating Repackager Projects

Method	Installation Source
Repackaging Wizard	You can use the Repackaging Wizard to convert the following installations: • InstallShield Professional 1.x to 5.1.x • InstallShield Professional 5.5 to 7.x • InstallShield InstallScript MSI • InstallShield DevStudio 9.x InstallScript • InstallShield Editor InstallScript See Repackaging Legacy Installations Using the Repackaging Wizard.
Repackager Interface	You can use the Repackager interface to convert the following installations: • Repackager 3.x output (.inc) • Microsoft SMS projects (.ipf) • Novell ZENworks 3.0, 3.1, or 4.0 projects (.axt/.aot) • WinINSTALL projects (.txt) (6.0, 6.5, 7.x) • Wise installation projects (.wse) • InstallShield Professional log files (.isl) See Converting Legacy Installations Using the Repackager Interface.

Converting Legacy Installations Using the Repackager Interface

In addition to repackaging a legacy installation using the Repackaging Wizard, you can also convert many setup types directly to Repackager projects (`.irp`)—and ultimately to InstallShield Editor projects (`.ism`) and Windows Installer packages (`.msi`). Repackager can directly convert the following setup types:

- Converting Repackager 3.x Output Files
- Converting a Microsoft SMS Project to a Repackager Project
- Converting Novell ZENworks Projects
- Converting WinINSTALL Projects
- Converting Wise Installation Projects
- Converting InstallShield Professional Log Files

Converting Repackager 3.x Output Files

To convert a Repackager 3.x output file to a Repackager project, perform the following steps.

Task: ***To convert a Repackager 3.x output file (.inc) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. On the **File** menu, click **Open**. The **Open** dialog box opens.
3. Change the **Files of type** filter to **Legacy Repackager Files (*.inc)**.
4. Browse to locate the Repackager 3.x output file you want to convert.
5. Select the file and click **OK**.

The Repackager 3.x project is updated to the Repackager project (`.irp`) format. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting a Microsoft SMS Project to a Repackager Project

To convert a Microsoft SMS project to a Repackager project, perform the following steps.

Task: ***To convert a Microsoft SMS project (.ipf) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the File menu, select Open.
3. In the Open dialog box, change the Files of type filter to SMS Installer (*.ipf).
4. Browse to locate the SMS project you want to convert.
5. Select the project, and click OK.

The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting Novell ZENworks Projects

You can convert Novell ZENworks 3.0, 3.1, or 4.0 projects (`.axt/.aot`) to Windows Installer packages (`.msi`) one at a time or in bulk:

- **Repackager Interface**—You can convert a ZENworks project to a Windows Installer package using the Repackager interface. See Converting a Novell ZENworks Project Using the Repackager Interface.

- **Command Line**—You can use the Command Line to bulk convert multiple ZENworks projects to Windows Installer packages. See Converting Multiple Novell ZENworks Projects Using the Command Line.

Note • *In order to convert an .aot file, the ZENworks Desktop Management Agent 6.5 or later (zenlite.dll) must be installed on the workstation where Repackager is installed. If this agent is not installed, Repackager can only convert ZENworks .axt files. See About .axt and .aot Application Object Template Files for more information.*

Converting a Novell ZENworks Project Using the Repackager Interface

Using Repackager, you can convert Novell ZENworks 3.0, 3.1 or 4.0 projects (`.axt/.aot`) to Windows Installer packages (`.msi`).

About .axt and .aot Application Object Template Files

In ZENworks Desktop Management, the `snAppShot` utility generates application object template files—with either an `.axt` or `.aot` extension—that contain the details that are required for the Application Launcher to be able to distribute an application to a workstation:

- registry entries to be added
- files to be copied
- changes to be made in the `.ini` files and system text files (`autoexec.bat` and `config.sys`)

Because an `.axt` file is a text file that can be edited with a text editor in order to modify it after it has been created, it can be opened and converted by Repackager.

However, in order to convert a `.aot` file (which is not a text file), the ZENworks Desktop Management Agent 6.5 or later (`zenlite.dll`) must be installed on the workstation where Repackager is installed. If this agent is not installed, Repackager can only convert ZENworks `.axt` files.

Note • *For information on installing the ZENworks Desktop Management agent (version 6.5 or later) to a workstation, see one of the following:*

- ***Novell ZENworks 6.5 Desktop Management Installation Guide***
  http://www.novell.com/documentation/zenworks65/dminstall/data/front.html
- ***Novell ZENworks 7 Desktop Management Installation Guide***
  http://www.novell.com/documentation/zenworks7/index.html?page=/documentation/zenworks7/dm7install/data/front.html

Task: ***To convert a Novell ZENworks project (.axt/.aot) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. On the **File** menu, click **Open**.

3. In the **Open** dialog, change the **Files of type** filter to either **Novell ZENworks (*.axt)** or **Novell ZENworks (*.axt/*.aot)**.

Note • *If the ZENworks Desktop Management Agent 6.5 or later (zenlite.dll) is installed on the workstation where Repackager is installed, the Files of type filter will be Novell ZENworks (*.axt/*.aot). If this agent is not installed, the Files of type filter will be Novell ZENworks (*.axt) and you will be unable to select .aot files as the legacy setup source. See About .axt and .aot Application Object Template Files for more information.*

4. Browse to locate the ZENworks project you want to convert.
5. Select the project, and click **OK**.

The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting Multiple Novell ZENworks Projects Using the Command Line

To perform a bulk conversion of ZENworks 3.0, 3.1, or 4.0 projects to Windows Installer packages, you use the `-Z` command line switch.

Task: ***To convert multiple Novell ZENworks projects (.axt/.aot) to a Windows Installer package, a Repackager project, or an InstallShield Editor project:***

1. Create an `.ini` file using the following format:

```
[General]
OutputFormat=MSI|INC|ISM

[AXT]
C:\myData\Project1.axt
C:\myData\Project2.axt
C:\myData\Project3.axt

[AOT]
C:\myData\Project1.aot
C:\myData\Project2.aot
C:\myData\Project3.aot
C:\myData\Project4.aot
```

The following table describes the elements of this file:

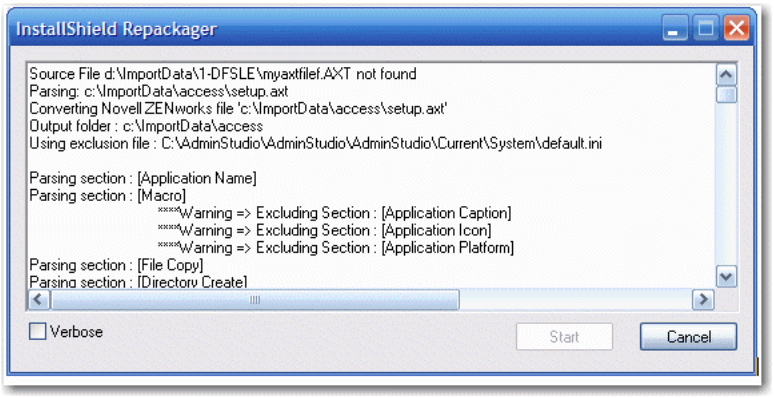
Section	Description
[General]	Controls the output format of the entire conversion process. Select one of the following to identify the output format: • **MSI**—Windows Installer package • **INC**—Repackager output file • **ISM**—InstallShield Editor project file
[AXT]	List the names and locations of the legacy ZENworks projects (`.axt`) you want to convert. Include the paths (absolute or relative) to the `.axt` files.
[AOT]	List the names and locations of the ZENworks `.aot` projects you want to convert. Include the paths (absolute or relative) to the `.aot` files.

2. Run the repackaging process from the command line using the `-Z` parameter:

```
ISLC.exe -Z"C:\DirectoryName\FileName.ini"
```

Caution • *You must enter a fully qualified path to identify the location of your .ini file.*

Repackager loads the `.ini` file and begins the conversion process. A dialog box opens to display progress messages.



To limit the volume of messages listed, clear the **Verbose** check box.

3. When the repackaging process is complete, the **Cancel** button changes to a **Close** button. Click **Close** to close this dialog box.

You will find the converted files in the location specified in the `.ini` file as the location of the `.aot/.axt` input files.

Converting WinINSTALL Projects

Task: ***To convert a WinINSTALL 6.0, 6.5, or 7.x project (.txt) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the **File** menu, select **Open**.
3. In the **Open** dialog box, change the **Files of type** filter to **WinINSTALL (*.txt)**.
4. Browse to locate the WinINSTALL project you want to convert.
5. Select the project, and click **OK**.
6. If the **WinINSTALL Conversion** dialog box opens, fill in the WinINSTALL-specific variables and click **OK**.

   The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Note • *WinINSTALL projects must be converted to .txt files prior to conversion to Repackager projects.*

Converting Wise Installation Projects

Task: ***To convert a Wise Installation project (.wse) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the File menu, select Open.
3. In the Open dialog box, change the Files of type filter to Wise Projects (*.wse).
4. Browse to locate the Wise Installer project you want to convert.
5. Select the project, and click OK.

   The legacy project is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Converting InstallShield Professional Log Files

You can convert an InstallShield Professional log file (`.isl`) to a Repackager project if you have access to the original setup media. When you open the log file, following the steps below, Repackager will try to find the original setup media automatically (in the location specified in the log file), but if it cannot, it will allow you to browse to it before continuing. If you do not have access to the original setup media, the conversion will fail.

Task: ***To convert an InstallShield Professional Log File (.isl) to a Repackager project (which can subsequently be built into a Windows Installer package):***

1. Launch Repackager.
2. From the File menu, select Open.
3. In the Open dialog box, change the Files of type filter to InstallShield Pro Log Files (*.isl).
4. Browse to locate the InstallShield Professional log file you want to convert.
5. Select the file, and click OK.

   The log file is converted to a Repackager project. Files, `.ini` files, shortcuts, and registry entries within the project are visible through the appropriate views in the Repackager Interface.

Working With Repackager Projects

After creating a Repackager project—by Repackaging Legacy Installations Using the Repackaging Wizard or by Converting Legacy Installations Using the Repackager Interface—you can perform the following tasks:

- Building an InstallShield Editor Project
- Building a Windows Installer Package
- Automatically Generating a Virtual Application During Repackager Project Build
- Viewing Repackager Project Properties
- Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project
- Creating a Setup Capture Report for a Project
- Saving Repackager Projects
- Opening InstallShield Editor from Repackager

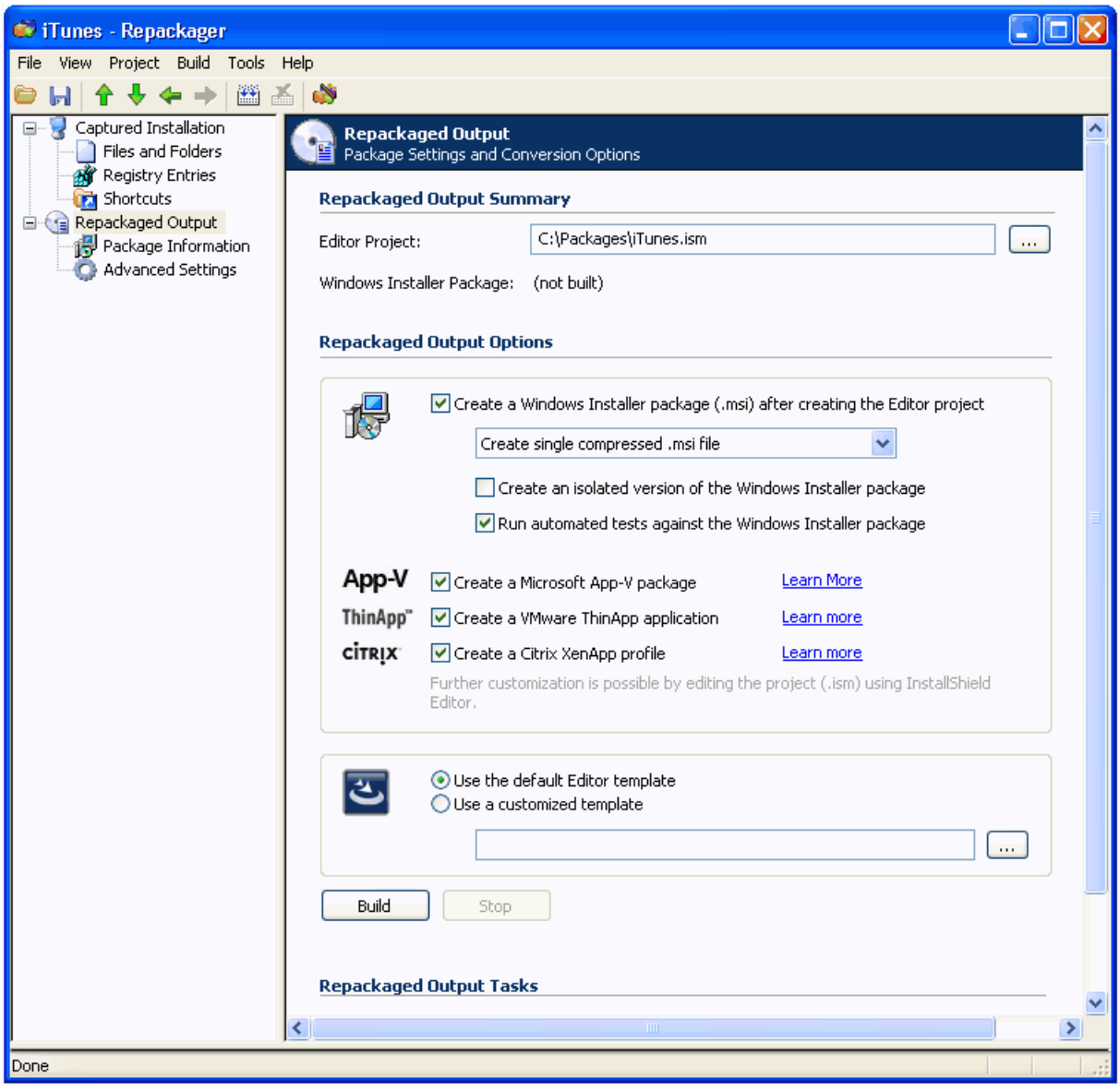
Building an InstallShield Editor Project

You can build an InstallShield Editor project (`.ism`) from your Repackager project (`.irp`).

You can also choose to build just an InstallShield Editor project, so that you can open it in InstallShield Editor and make some modifications prior to building.

Task: ***To build an InstallShield Editor project (.ism):***

1. In the Repackager interface, open the Repackager project that you want to convert to an InstallShield Editor project.
2. Select **Repackaged Output** from the View List. The **Repackaged Output View** opens.



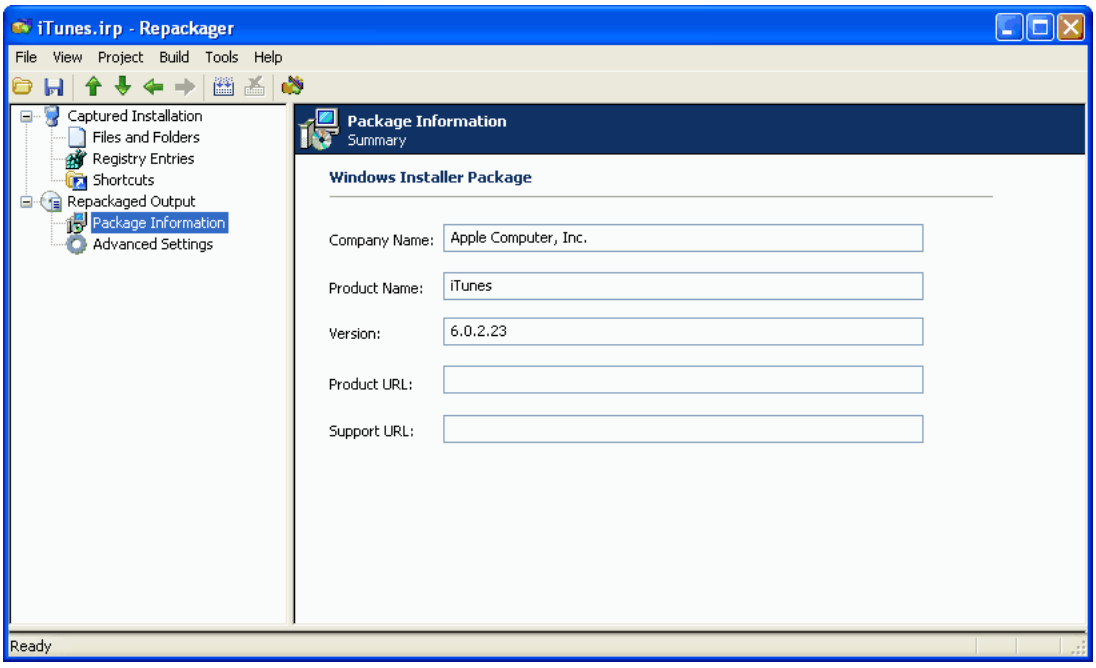
3. In the **Editor Project** field, enter the name and location of the InstallShield Editor Project file you want to create.
4. If you do not want to **Create a Windows Installer package (.msi) after creating the Editor project**, clear this option. If you want to create a Windows Installer Package, see Building a Windows Installer Package.

5. A project template contains all of the default settings and design elements that you want to use as a starting point when you create an installation project. In the **Repackaged Output Options** area, select the InstallShield Editor Project Template you want to use when creating the project:

   - **Use the default Editor template**—Select this option to use the default InstallShield Editor Project Template.
   - **Use a customized template**—Select this option to use a customized InstallShield Editor Project Template.

     For example, if you wanted all of your InstallShield Editor projects to have a special custom dialog, a set of required redistributables, and a particular SQL script, you could create a project template that has all of those settings. Then, any time that you wanted to create a new project, you could base it off of your custom template. This enables you to avoid re-creating the custom dialog, re-adding the redistributables, and re-adding the SQL script every time that you create a new InstallShield Editor Project.

6. Select **Package Information** from the View List. The Package Information View opens, where you can specify information for the Windows Installer package that you build from the Repackager project. Much of this information may be prepopulated based on settings used in the Repackaging Wizard.



7. Enter the following information:

   a. **Company Name**—The name of the company that developed the product you are repackaging.

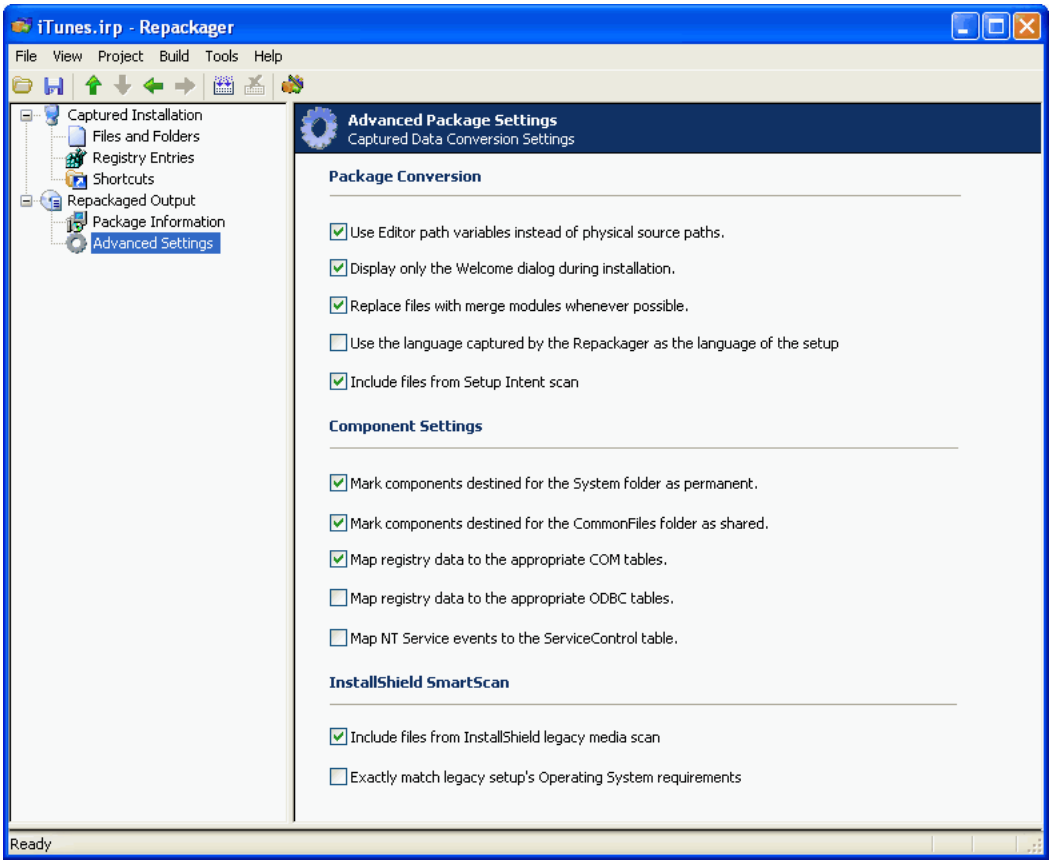
   b. **Product Name**—The name of the product you are repackaging.

   c. **Version**—The product's version number.

   d. **Product URL**—The URL for product information. This appears in **Add/Remove Programs** in the Control Panel.

e. **Support URL**—A URL for support information. This also appears in **Add/Remove Programs** in the Control Panel, and is often changed during repackaging to provide an internal support URL.

8. Select **Advanced Settings** from the View List. The Advanced Settings View opens.



9. Select the options that you want to use, as described in Configuring Advanced Conversion Options.

10. Select **Repackaged Output** on the View List. The **Repackaged Output View** opens.

11. Click the **Build** button. The build process begins, and its progress is reported in the output window.

    When the build process is complete, a `Conversion completed` message appears in the output window, and a link to the build log file is provided.

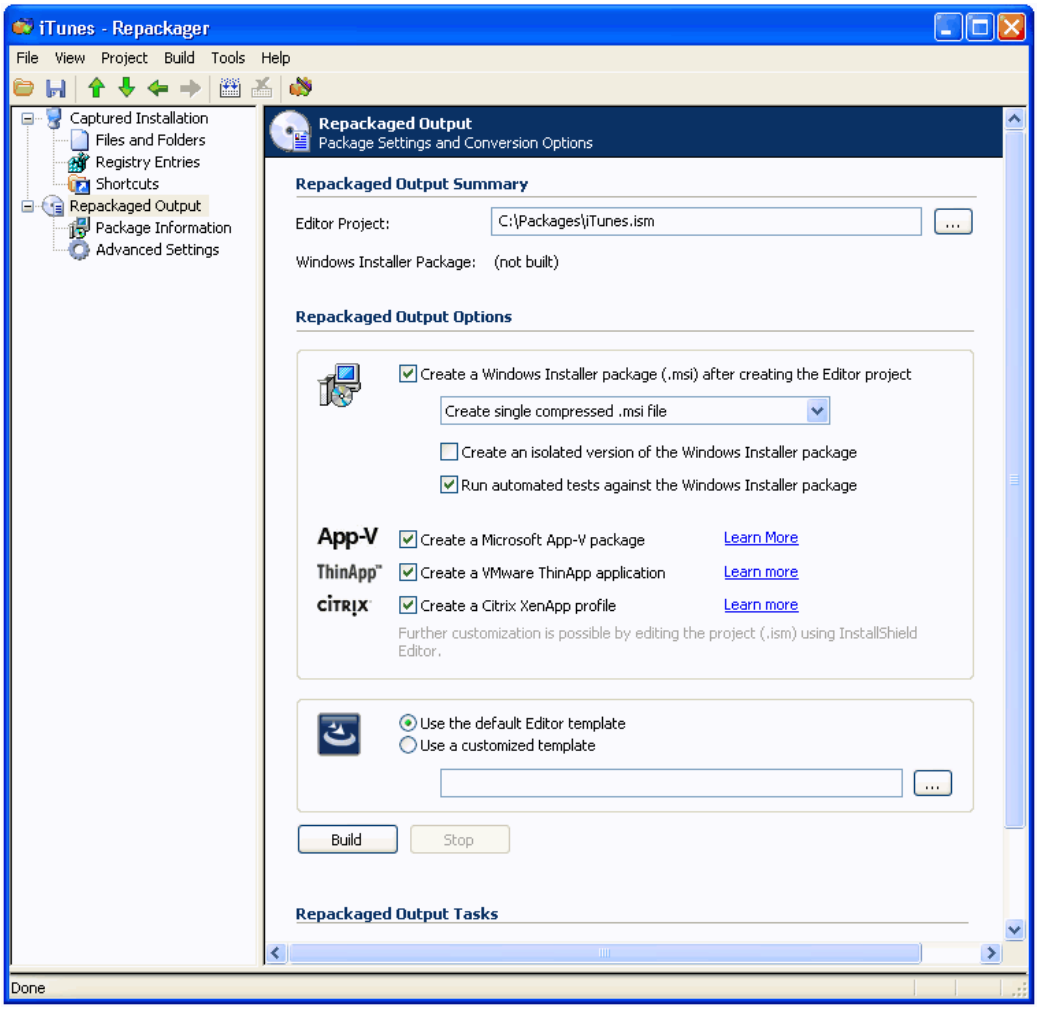
Building a Windows Installer Package

You can simultaneously build an InstallShield Editor project (`.ism`) and a Windows Installer package (`.msi`) from your Repackager project (`.irp`). However, before you do so, you must configure options in your Repackager project necessary for the build.

Note • *For information on building a virtual application, see Automatically Generating a Virtual Application During Repackager Project Build.*

Task: ***To build an InstallShield Editor project (.ism) and a Windows Installer package (.msi):***

1. In the Repackager interface, open the Repackager project that you want to convert to an InstallShield Editor project and build a Windows Installer package.
2. Select **Repackaged Output** from the View List. The **Repackaged Output View** opens.



3. In the **Editor Project** field, enter the name and location of the InstallShield Editor Project file you want to create.
4. Select the **Create a Windows Installer package (.msi) after creating the Editor project** option, and select the following additional options:

a. The compression option that you select for this package depends upon the size of your application's installation and your delivery method.

Neither `Setup.exe` nor your `.msi` file can be spanned across multiple disks. So, if the source files associated with your Windows Installer package cannot fit on the same disk as the `setup.exe` and `.msi` file, you will need to include them in `.cab` files on other disks. But if you are performing a network installation and have unlimited space, there is no need to compress files or include additional files in .cab files.

From the list, select one of the following options:

Option	Description
Create single compressed .msi file	Select this option if you want to compress all necessary files inside the .msi package, as opposed to storing them outside of the .msi database.
Create single compressed setup.exe file	Select this option if you want to compress all files inside a setup.exe file, including the .msi file and all other necessary files.
Create .msi file + external compressed .cab file	Select this option if you want to create an .msi file and want to compress the rest of the necessary files in an external .cab file. For example, you might have an installation that contains three features—each containing a 1.5 MB file, Setup.exe, and the installation files for Windows NT—and you want to create a custom media type that is 2 MB in size. The build will span multiple disks. • Disk one will contain Setup.exe, InstMsiW.exe (which contains the logic to install the Windows Installer service on Windows NT machines), Setup.ini (which is required for installations that include Setup.exe), and your .msi file. • The remaining disks will contain .cab files that store compressed copies of all your source files.
Create .msi file + external compressed .cab file + setup.exe	Select this option if you want to create an .msi file and a setup.exe file, and want to compress all the rest of the necessary files in an external .cab file.
Create uncompressed .msi file	Select this option if you want to create an uncompressed .msi file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi file.
Create uncompressed .msi file and setup.exe	Select this option if you want to create an uncompressed .msi file along with a setup.exe file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi and setup.exe files.

b. To reduce versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested, select the **Create an**

isolated version of the Windows Installer package. An additional Windows Installer package will be created in the same directory as the `.ism` file and the other `.msi` file, with the naming convention of:

appname`.isolated.msi`

For more information on how Repackager isolates applications and the available isolation options, see Isolating Windows Installer Packages.

c. Select the **Run automated tests against the Windows installer package** option to automatically run PackageExpert tests against the newly built Windows Installer package to determine if it is built according to Windows Installer standards, and if it is in compliance with the installation requirements of the Windows Vista operating system. All of the tests that are currently selected on the PackageExpert **Configuration View** are run, and those selected tests that are automatically resolvable, will be resolved.

d. To build a virtual application, select the **Create a Microsoft App-V application**, **Create a VMware ThinApp application**, and/or **Create a Citrix XenApp profile** option. See Automatically Generating a Virtual Application During Repackager Project Build.

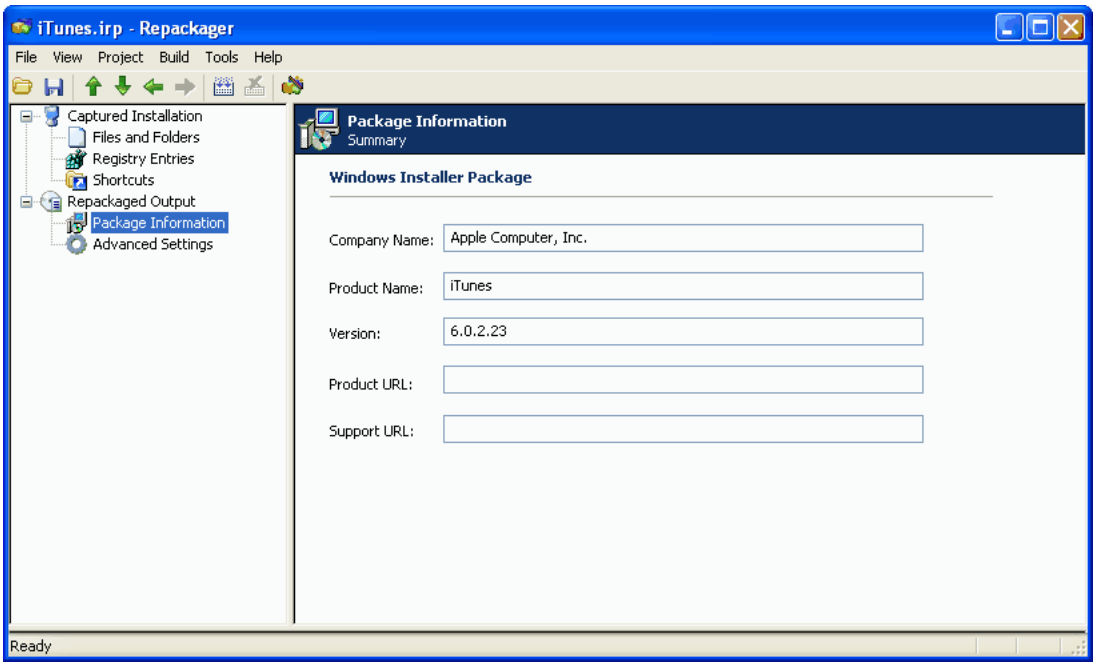
Note • *In order to select one of these virtualization options, you must have already selected the* ***Create a Windows Installer package (.msi) after creating the Editor project*** *option.*

5. A project template contains all of the default settings and design elements that you want to use as a starting point when you create an installation project. In the **Repackaged Output Options** area, select the InstallShield Editor Project Template you want to use when creating the project:

- **Use the default Editor template**—Select this option to use the default InstallShield Editor Project Template.
- **Use a customized template**—Select this option to use a customized InstallShield Editor Project Template.

  For example, if you wanted all of your InstallShield Editor projects to have a special custom dialog, a set of required redistributables, and a particular SQL script, you could create a project template that has all of those settings. Then, any time that you wanted to create a new project, you could base it off of your custom template. This enables you to avoid re-creating the custom dialog, re-adding the redistributables, and re-adding the SQL script every time that you create a new InstallShield Editor Project.

6. Select **Package Information** from the **View List**. The **Package Information View** opens, where you can specify information for the Windows Installer package that you build from the Repackager project. Much of this information may be prepopulated based on settings used in the Repackaging Wizard.



7. Enter the following information:

   a. **Company Name**—The name of the company that developed the product you are repackaging.

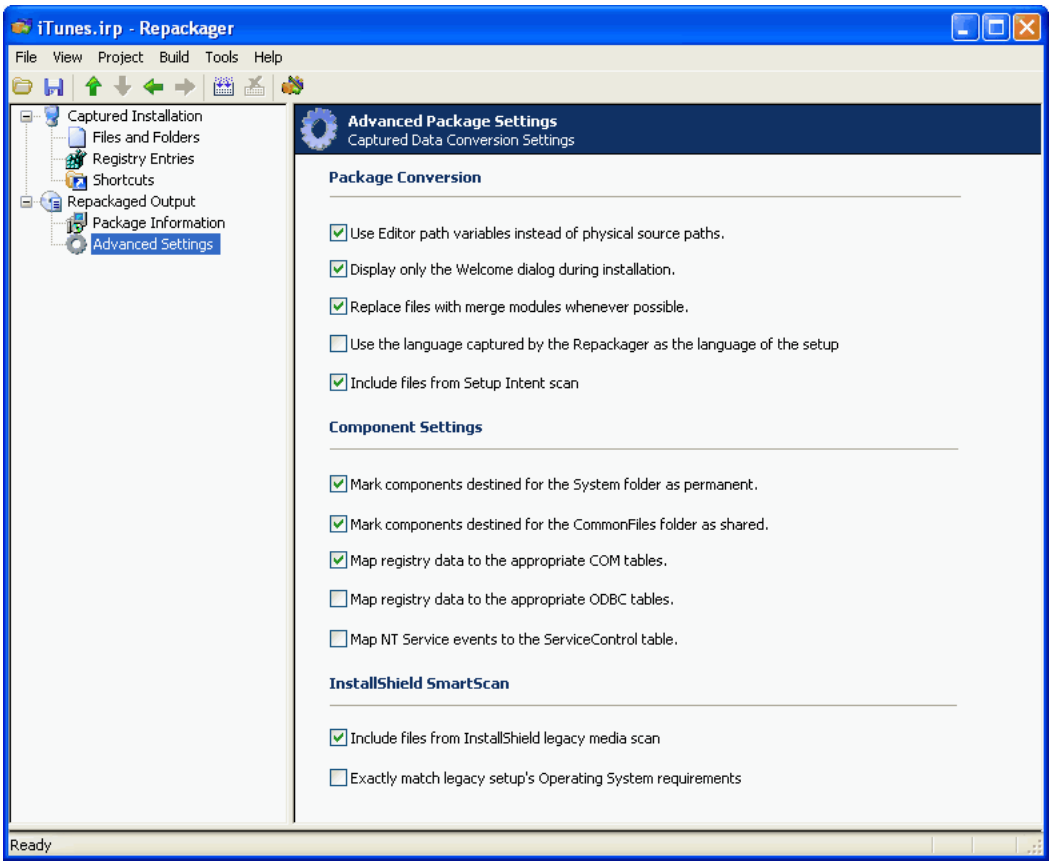
   b. **Product Name**—The name of the product you are repackaging.

   c. **Version**—The product's version number.

   d. **Product URL**—The URL for product information. This appears in **Add/Remove Programs** in the Control Panel.

   e. **Support URL**—A URL for support information. This also appears in **Add/Remove Programs** in the Control Panel, and is often changed during repackaging to provide an internal support URL.

8. Select **Advanced Settings** from the View List. The Advanced Settings View opens.



9. Select the options that you want to use, as described in Configuring Advanced Conversion Options.
10. Select **Repackaged Output** on the View List. The **Repackaged Output View** opens.
11. Click the **Build** button. The build process begins, and its progress is reported in the output window.

    When the build process is complete, a `Conversion completed` message appears in the output window, a link to the build log file is provided, and the location of the `.msi` file is listed. For example:

    ```
    Output file: C:\1516261\WinZip.msi
    ```

About the Context.msi File

When some Windows Installer packages are repackaged, some of their data (such as files or registry entries) are excluded according to the normal Repackager exclusion settings. For example, files destined for the `\Windows\Installer` folder are typically excluded. However, this type of information is occasionally necessary in order to successfully convert a Windows Installer package to a virtual package.

To address this issue, when Repackager builds a Windows Installer package, it now produces two `.msi` files: `packagename.msi` and `packagename.context.msi`.

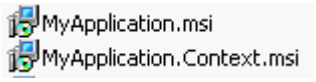


Figure 8-1: Repackaged Output: application.msi and application.context.msi

The `.context.msi` file contains context data that is necessary in order to convert a `.msi` file to a virtual package. When creating a virtual package, Repackager combines the data in both the main `.msi` file and the `.context.msi` file to produce the final virtual package.

For more information on the `.context.msi` file, see Capturing Virtualization Context.

Important • *If you are not converting a package to a virtual package, you can ignore its `.context.msi` file.*

Note • *Context data is not displayed in the Repackager interface when viewing captured Files/Registry details.*

Configuring Advanced Conversion Options

To set package conversion, component settings, and InstallShield SmartScan options in your Repackager project, perform the following steps.

Task: ***To configure advanced conversion options:***

1. Select **Advanced Settings** from the Repackager View List. The **Advanced Settings View** opens.
2. Under **Package Conversion**, select the package conversion options you want to use during conversion:

Option	Description
Use Editor path variables instead of physical source paths	When storing files in the InstallShield Editor project (`.ism`), the Wizard uses path variable locations whenever possible.
Display only the Welcome dialog box during installation	Only the Welcome dialog box is displayed when the Windows Installer package is run on a target machine. If this option is unchecked, the default UI sequence is displayed when the setup is installed.
Replace files with merge modules wherever possible	Following best practice rules, Repackager replaces components with comparable merge modules whenever possible.
Use the language captured by the Repackager as the language of the setup	When selected, the target package's language will be the language detected by Repackager (as displayed in the Captured Installation view).

Option	Description
Include files from Setup Intent scan	Any files identified when running the Setup Intent Wizard will be included in the package (unless you have manually excluded them from the project).

3. Under **Component Settings**, select the component settings options you want to use during conversion:

Option	Description
Mark components destined for the System folder as permanent	Executable files installed to the system folder (System32Folder) are marked as Permanent files and will not be uninstalled when the package is uninstalled. This eliminates ICE09 validation errors.
Mark components destined for the CommonFiles folder as shared	Executable files installed to the CommonFilesFolder (or a subfolder of CommonFilesFolder) are marked as shared files. This ensures that these components can coexist with DLLs installed by previous setups.
Map registry data to the appropriate COM tables	Setting this option reduces the number of ICE33 warnings that can occur during package validation, resulting from data not being mapped to the appropriate MSI tables.
Map registry data to the appropriate ODBC tables	If selected, ODBC-related registry data is mapped to ODBC tables instead of the Registry table. This data will only function correctly if Windows Installer supports the ODBC resource being mapped; it is recommended that you do not enable this option if you are unsure whether the ODBC resources are supported correctly by Windows Installer.
Map NT Service events to the ServiceControl table	If selected, NT Service–related registry data is mapped to ServiceControl table instead of the Registry table.

4. Under **InstallShield SmartScan**, select the InstallShield SmartScan options you want to use during conversion:

Option	Description
Include files from InstallShield legacy media scan	Files identified in the Media Scan Wizard will be included in the package (unless you have manually excluded them from the project).

Option	Description
Exactly match legacy setup's Operating System requirements	For SmartScan projects or projects using the Professional Logging Method: • **if this option is selected**, component conditions will store specific operating system information: if the filter was NT4, the condition will be (VersionNT=4). • **If this option is not selected**, component conditions will store a grouping of the operating system: if the filter was NT4, the condition will be (VersionNT). By default, this option is not selected.

Note • *For more information on InstallShield SmartScan, see Scanning InstallShield Professional Setups for Additional Data.*

Automatically Generating a Virtual Application During Repackager Project Build

Edition • *The features on the Repackaged Output View are available in AdminStudio Standard, Professional, and Enterprise Editions.*

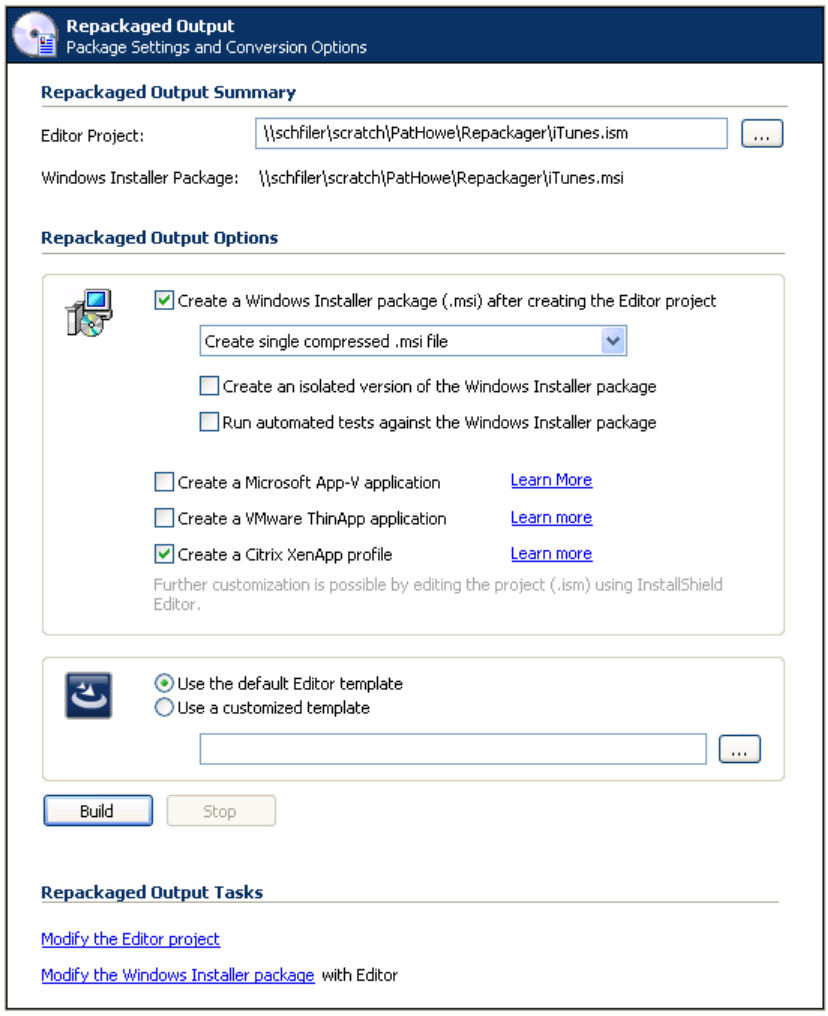
You can simultaneously build an InstallShield Editor project (`.ism`), a Windows Installer package (`.msi`), a Microsoft App-V application, a ThinApp application, and a Citrix profile from your Repackager project (`.irp`). To do this, you need to select options on the Repackager **Repackaged Output** view.

Task: ***To automatically generate a virtual application during Repackager project build:***

1. In the Repackager interface, open a Repackager project.
2. Select **Repackaged Output** from the View List. The **Repackaged Output** view opens.
3. In the **Editor Project** field, enter the name and location of the InstallShield Editor Project file you want to create.
4. Select the **Create a Windows Installer package (.msi) after creating the Editor project** option, and select the associated compression, isolation, and automated test options as described in Building a Windows Installer Package.

Important • *When building a virtual package, the* ***Create a Windows Installer package (.msi) after creating the Editor project*** *option* ***must*** *be selected. If it is not selected, the virtualization options are disabled.*

5. Select one or more of the virtual application options:
   - **Create a Microsoft App-V application**
   - **Create a VMware ThinApp application**
   - **Create a Citrix XenApp profile**



Note • *If you would like to further customize the virtual application using the InstallShield Microsoft App-V Assistant, ThinApp Assistant, or Citrix Assistant, you can click the* ***Modify the Editor project*** *link below to*

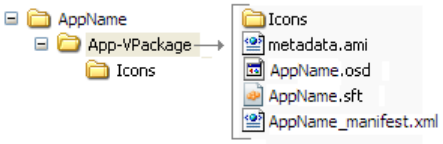
open this project in InstallShield Editor. This option is not available until after you build the Repackager project the first time.

Note • *You can also use the Automated Application Converter to convert a Windows Installer package to a virtual package. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

6. Select whether to use the default Editor template or a customized template, as described in Building a Windows Installer Package.
7. Select **Package Information** from the View List and set Package Information options as described in Building a Windows Installer Package.
8. Select **Advanced Settings** from the View List and select the options that you want to use, as described in Configuring Advanced Conversion Options.
9. Select **Repackaged Output** on the View List. The **Repackaged Output View** opens.
10. Click the **Build** button. The build process begins, and its progress is reported in the output window.

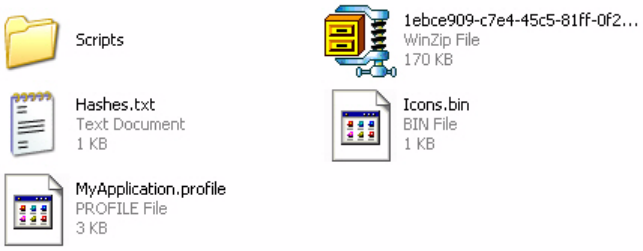
    When the build process is complete, a `Conversion completed` message appears in the output window, and a link to the build log file is provided.

    - **If you chose the App-V application option**, a folder named `App-VPackage` was created in the location you specified in the **Editor Project** field. This folder contains the App-V application for this package and all of its associated files:



    - **If you chose the ThinApp application option**, a folder named `ThinAppPackage` was created in the location you specified in the **Editor Project** field. This folder contains the ThinApp application for this package and all of its associated files:

    - **If you chose the Citrix XenApp profile option**, a folder named `CitrixProfile` was created in the location you specified in the **Editor Project** field. This folder contains the Citrix profile for this package and all of its associated files:

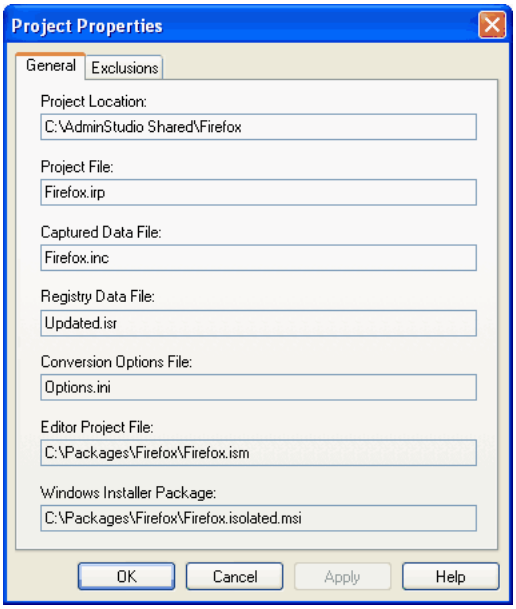


Viewing Repackager Project Properties

You can view the properties of the currently open Repackager project by opening the Project Properties dialog box.

Task: ***To view properties for the current Repackager project:***

1. Open a project in the Repackager interface.
2. From the **Project** menu, select **Properties**. The **General** tab of the **Project Properties** dialog box opens.



The following properties are listed:

- **Project Location**—The full path of the directory where the current Repackager project file (`.irp`) is located.
- **Project File** —The name of the current Repackager project file.
- **Captured Data File**—The name and location of the captured data file (`.inc`), which was either created by the Repackaging Wizard or during conversion of a Novell ZENworks project, Microsoft SMS project, or WinINSTALL project. The path is relative to the current Repackager project file.

- **Registry Data File**—The name and location of the file containing captured registry data. The path is relative to the current Repackager project file.
- **Conversion Options File**—The name and location of the `Options.ini` file, which contains an exhaustive list of all options you can use during conversion of the Repackager project to an InstallShield Editor project and Windows Installer package.
- **Editor Project File**—The name and location of the InstallShield Editor project file as set in the Product View (MSI Package). The path is relative to the current Repackager project file.
- **Windows Installer Package**—The name and location of the Windows Installer package. The path is relative to the current Repackager project file.

3. When finished viewing properties in the **General** tab, click OK.

Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project

Although an installation may have intended to install certain files, these files sometimes may not be installed—often because the files already exist on the target machine (either as the same version or a newer version). These files, although not installed or updated, are needed for the product to execute properly when the setup is run on a system that does not already have these files.

You can use the **Setup Intent Wizard** to detect file dependencies that may not be included in your Repackager project (`.irp`). The **Setup Intent Wizard** scans a setup to identify files that may not have been captured during repackaging—effectively recognizing the installation's intent for these files.

To use the **Setup Intent Wizard**, perform the following steps:

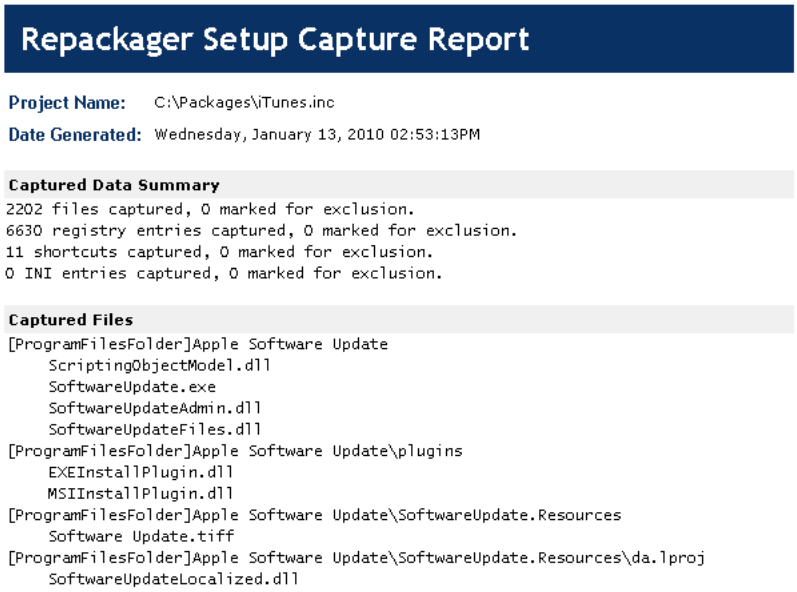
Task: ***To detect file dependencies:***

1. From the Project menu, select Setup Intent Wizard. The **Welcome Panel** opens.
2. From the **Welcome** panel, click **Next**. The **Scanning Project Panel** opens.
3. Once scanning is finished, the **Results Panel** opens, listing new files that your setup requires.
4. From the **Results Panel**, select the files you want added to your Repackager project and click **Finish**.
5. Save your Repackager project.

Note • *Because the Setup Intent Wizard analyzes files in the Repackager project and searches for dependent files, you must run the Setup Intent Wizard from the same machine where repackaging was performed (with the Repackaging Wizard). You can then save the Repackager project and transfer it to another machine.*

Creating a Setup Capture Report for a Project

You can generate an HTML or text document that summarizes the data that was captured when a setup was repackaged.



Repackager Setup Capture Report

Project Name: C:\Packages\iTunes.inc

Date Generated: Wednesday, January 13, 2010 02:53:13PM

Captured Data Summary

2202 files captured, 0 marked for exclusion.
6630 registry entries captured, 0 marked for exclusion.
11 shortcuts captured, 0 marked for exclusion.
0 INI entries captured, 0 marked for exclusion.

Captured Files

[ProgramFilesFolder]Apple Software Update
    ScriptingObjectModel.dll
    SoftwareUpdate.exe
    SoftwareUpdateAdmin.dll
    SoftwareUpdateFiles.dll
[ProgramFilesFolder]Apple Software Update\plugins
    EXEInstallPlugin.dll
    MSIInstallPlugin.dll
[ProgramFilesFolder]Apple Software Update\SoftwareUpdate.Resources
    Software Update.tiff
[ProgramFilesFolder]Apple Software Update\SoftwareUpdate.Resources\da.lproj
    SoftwareUpdateLocalized.dll

Figure 8-2: Sample Repackager Setup Capture Report

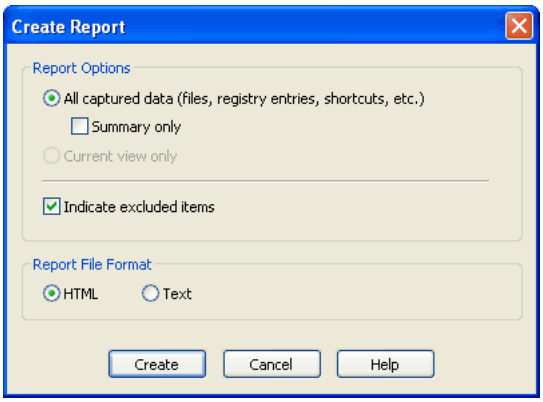
The following information is available to be displayed in this report:

- Captured files
- Captured shortcuts
- Captured .ini file entries
- Captured Registry entries

The report also indicates which of the files, shortcuts, `.ini` file entries, or Registry entries, if any, have been marked for exclusion. Those marked for exclusion are not included in the Repackager project.

Task: ***To create a report detailing captured data:***

1. From the **Project** menu, select **Create Report**. The **Create Report** dialog box opens.



2. Select whether you want the report to contain **All captured data** (all of the data collected during the entire capture), or just the **Current view**.
3. If you want the report to contain data from the entire capture, specify whether you want to just display summary information.
4. Specify whether you want to display excluded items in the report.
5. Select the file format for the report. You can generate an **HTML** report or a **Text** report.
6. Click **Create**. A Save As dialog box opens.
7. From the resulting Save As dialog box, browse to the location where you want to save the file, and provide a name for the report.
8. Click **Save**. The report is saved to the specified location and automatically opens.

Saving Repackager Projects

Edition •

To save a Repackager project, perform the following steps:

Task: ***To save the current Repackager project:***

1. Select **Save** from the **File** menu.

   or

2. Click the Save button (💾) on the toolbar.

Task: ***To save the current Repackager project under a different name:***

Select Save As from the File menu.

Opening InstallShield Editor from Repackager

After building your Repackager project into a Windows Installer package and/or an InstallShield Editor project, you may want to launch InstallShield Editor for additional modifications.

Task: ***To launch the generated InstallShield Editor project (.ism) in InstallShield Editor:***

From the Repackager **Project** menu, select **Edit InstallShield Project**. If installed, InstallShield Editor opens the project file.

Task: ***To launch the generated Windows Installer package (.msi) in InstallShield Editor:***

From the Repackager **Project** menu, select **Edit Windows Installer Package**. If installed, InstallShield Editor opens the package in Direct MSI Edit mode.

Isolating Windows Installer Packages

Application isolation is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

When building a Windows Installer package from your Repackager project, you can also choose to create an isolated version of that package by selecting an option on the Repackaged Output View.

Information about application isolation is presented in the following topics:

- About Application Isolation
- About Assemblies
- About Manifests
- About Digital Certificates
- Setting Isolation Options
- Building an Isolated Windows Installer Package

About Application Isolation

Application isolation, which is a technique used to minimize the dependencies of an application on system components or dynamic elements, is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components and dynamic elements with which it was originally developed and tested.

Isolation is accomplished by:

- Providing DLLs and other shared components for specific applications, *and*
- Placing information traditionally stored in the Registry into other files that specify the locations of these isolated components.

Application isolation provides increased stability and reliability for applications because they are unaffected by changes caused by installation and ongoing maintenance of other applications on the system.

Depending on the isolation options chosen, you can partially or totally isolate an application. When using assemblies and manifests to isolate applications for Windows XP systems, the assemblies can be updated following deployment without necessitating application reinstallation.

Reasons to Isolate Applications

You would want to isolate an application if:

- You want to resolve incompatibilities between different versions of shared components.
- You want to reduce the complexity of the installation by storing COM activation data in a manifest instead of the registry.
- You want to insulate the application from changes to shared components.

Reasons Not to Isolate an Application

You would not want to isolate an application if, following application isolation, you discover that the application no longer works because of an internal dependency on a component that has been moved during the isolation process.

Tip • *Following isolation, you can use the Dynamic Dependency Scanner in InstallShield Editor to verify isolated files are loaded from a different directory.*

Isolating Windows Installer Packages Using Application Isolation Wizard

In addition to being able to generate an isolated version of a repackaged setup immediately after the build step in Repackager, you can also use Application Isolation Wizard to isolate a Windows Installer package.

Application Isolation Wizard is a stand-alone tool which accepts a Windows Installer package as input and outputs a new, isolated Windows Installer package.

The Application Isolation Wizard provides a user interface experience that allows the user to extend the initial "dependency scanning" process for identifying file isolation candidates, while in Repackager you specify your assembly and digital signing isolation options on the Isolation Options dialog box, and then those selections are applied to all isolated packages created by Repackager.

For more information, see Isolating Applications Using Application Isolation Wizard.

About Assemblies

Assemblies are DLLs or other portable executable files that applications require to function. Under Windows XP, these can be either shared or private. Private assemblies are typically stored in the same directory as the application they support. Shared assemblies are stored in the `WinSxS` directory, and are digitally signed.

By creating manifests for assemblies, Repackager allows you to create self-contained applications that can use different versions of the same DLL or other portable executable, without any version conflicts.

Shared Assemblies

Shared assemblies are assemblies available to multiple applications on a computer. Applications that require these assemblies specify their dependence within a manifest. Multiple versions of shared assemblies can be used by different applications running simultaneously.

These assemblies are stored in the `WinSxS` directory, and must be digitally signed for authenticity. After deployment, the version of shared assemblies can be changed, allowing for changes in dependencies.

Private Assemblies

Private assemblies are assemblies created for exclusive use by an application. They are accompanied by an assembly manifest, which contains information normally stored in the registry. Private assemblies allow you to totally isolate an application, eliminating the possibility that dependent files may be overwritten by other applications.

These assemblies are always stored in the same location as their associated executable.

About Manifests

Manifests, which are used during isolation, are XML files that describe an application. Repackager can create two types of manifests: application manifests and assembly manifest.

Application Manifests Describe an Isolated Application

Application manifests are XML files that describe an isolated application. This descriptive information includes the relationship between the application and its dependent files.

Typically, the naming convention for a manifest is:

```
ApplicationName.Extension.manifest
```

For example, if the application was `HelloWorld.exe`, the manifest file is called:

```
HelloWorld.exe.manifest
```

Assembly Manifests Describe an Application's Assemblies

Assembly manifests are XML files that describe an application's assemblies. This includes components such as DLLs.

Information stored in the assembly manifest, such as COM registration information, ProgIDs, etc., is usually stored in the Registry. However, by making it independent from the Registry, only that application can use the dependent files described in the manifest. This enables you to have multiple versions of the same DLL or other portable executable file on a system without generating compatibility conflicts.

Typically, the naming convention for a manifest is:

```
AssemblyName.Extension.manifest
```

For example, if the component was `Goodbye.dll`, the manifest file is called:

```
Goodbye.dll.manifest
```

Manifests as New Components

When you create manifests, Repackager supports putting them into new components. If you do not select the **Create new component for each assembly** option on the **Manifest Options** tab of the **Isolation Options** dialog box, the manifest will be added to the same component as the assembly.

About Digital Certificates

Digital certificates identify you and/or your company to end users to assure them the assembly they are about to use has not been altered. They are issued by a certification authority such as VeriSign, or created using a combination of software publishing credentials (`.spc`) and a private key (`.pvk`), both also issued by a certification authority. The certificate includes the public cryptograph key, and, when used in combination with a private key, can be used by end users to verify the authenticity of the signor.

The following digital certificate concepts are defined in this topic:

- Private Keys
- Software Publishing Credentials
- Using a Certificate Store
- Creating a Certificate File

Private Keys

A private key (a file with the extension .pvk) is granted by a certification authority. Repackager uses the private key you enter in the **Digital Signature** tab of the Isolation Options dialog box to digitally sign your shared assembly and ensure end users of its content's authenticity.

The .spc (Software Publishing Credentials) file and .pvk file you enter in the Digital Signature tab compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Software Publishing Credentials

You must supply a certification authority with specific information about your company and software to obtain software publishing credentials in the form of an .spc file. Your software publishing credentials are used to generate a digital signature for your assembly.

The `.spc` file and `.pvk` (private key) file you enter as in the Digital Signature tab of the Advanced Options dialog box compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Using a Certificate Store

To perform code signing, both private key and software publishing credential information must be supplied. This must occur each time a package is signed. Most server operating systems store a certificate locally on the computer that the user used to request the credential information.

Instead of having to store credential files on each of the user computers, you can create a Certificate Store, a storage location which will have numerous certificates, which enables all users or computers with adequate permissions to retrieve the certificate as needed.

Using a Certificate Store allows you to associate the same credentials and private key files with multiple packages. This simplification is particularly useful when isolating applications, as typically the code signing information will be identical for all shared assemblies. Ultimately, the Certificate Store removes the burden of managing private key and software publishing credential information.

Creating a Certificate File

You can create a certificate file from the constituent PVK and SPC files and import it into the Certificate Store using the PVK Digital Certificate Files Importer. You can then export the certificate (`.cer`) file for use outside of the Certificate Store.

Caution • *Certificate files must be 2048-bit or higher. For more information, see the article: Assembly Signing Example on the Microsoft Developer Network Web site.*

Setting Isolation Options

Application isolation is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

On the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the following Repackager isolation options:

- **Assembly Options**—Specify the type of assemblies Repackager will create: private side-by-side assemblies in the application folder or shared side-by-side assemblies in the `WinSxS` folder. You can also specify the assembly naming conventions. See Specifying Manifest Options.
- **Digital Signature Options**—You can configure the certificate information required when using shared assemblies. This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly. See Setting Digital Signature Options for Shared Assemblies.

Specifying Manifest Options

On the **Manifest Options** tab of the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the following options:

- Selecting the Assembly Type
- Specifying the Assembly Naming Conventions

Note • *For more information on assemblies and manifests, see About Assemblies and About Manifests.*

Selecting the Assembly Type

On the **Manifest Options** tab of the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the type of assemblies Repackager will create: private side-by-side assemblies in the application folder or shared side-by-side assemblies in the `WinSxS` folder.

Task: ***To select the assembly type:***

1. Open the Repackager interface.
2. From the **Tools** menu, select **Isolation Options**. The **Manifest Options** tab of the **Isolation Options** dialog box opens.
3. Select one of the following **Assembly Type** options:
   - Create private side-by-side assemblies in the application folder.
   - Create shared side-by-side assemblies in `WinSxS` directory.

Note • *Manifests for shared assemblies must be digitally signed. See Setting Digital Signature Options for Shared Assemblies.*

Note • *The modifications you make on the **Isolation Options** dialog box will be recorded in the `isolationconfig.ini` file, which is stored in the `AdminStudio Shared` directory.*

Specifying the Assembly Naming Conventions

On the **Manifest Options** tab of the Isolation Options Dialog Box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the type of naming conventions Repackager will use when creating assemblies.

Task: ***To set the default naming convention for assemblies:***

1. Open the Repackager interface.
2. From the **Tools** menu, select **Isolation Options**. The **Manifest Options** tab of the **Isolation Options** dialog box opens.
3. In the **Assembly Naming Conventions** area, enter your **Company** name and **Division**. These two fields create the default assembly naming convention (in the form `Company.Division.Assembly` followed by a number).
4. If you want to create a new component for each assembly, select the **Create new component for each assembly** option.

Assembles created during application isolation will follow the naming convention as specified.

Note • *The modifications you make on the **Isolation Options** dialog box will be recorded in the `isolationconfig.ini` file, which is stored in the `AdminStudio Shared` directory.*

Setting Digital Signature Options for Shared Assemblies

You can configure the certificate information required when using shared assemblies on the **Digital Signature** tab of the **Isolation Options** dialog box.This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly.

Note • *The modifications you make on the Isolation Options dialog box will be recorded in the `isolationconfig.ini` file, which is stored in the `AdminStudio Shared` directory.*

Note • *For more information, see About Digital Certificates.*

Task: ***To set digital signature options:***

1. Open the Repackager interface.
2. From the **Tools** menu, select **Isolation Options**. The **Manifest Options** tab of the **Isolation Options** dialog box opens.

3. Open the **Digital Signatures** tab.

4. Click the Browse (...) button next to the **Certificate File** field and navigate to the certificate file you are using to sign assemblies.

   A digital certificate identifies you and/or your company to end users and assures them the data they are about to receive has not been altered.

5. In the **Code Signing Technology** area, select the type of code signing technology you want to use for the digital signature. You can use one of the following technologies:

Technology	Description
Credentials	Select this option to use credential files as the code signing technology. If you select this option, you must supply the name and location of both your software publishing credential files: • **SPC File**—Specify the name and location of your software publishing credentials file (`.spc`). • **PVK**—Specify the name and location of your private key file (`.pvk`). In order to receive a software publishing credentials and a private key, you must supply a certification authority, such as VeriSign with specific information about your company and software.
Certificate Name in the store	Select this option to use the name of an existing certificate file in the Certificate Store as the code signing technology. The Certificate Store is a central repository for certificate files. Using a Certificate Store allows you to reuse the certificate files for different purposes as necessary. As an alternative to providing `.spc` and `.pvk` files, you can specify the certificate name as it appears in the certificate store.

Building an Isolated Windows Installer Package

To reduce versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested, select the **Create an isolated version of the Windows Installer package** option on the Repackager **Repackaged Output** view. An additional Windows Installer package will be created in the same directory as the `.ism` file and the other `.msi` file, with the naming convention of:

appname`.isolated.msi`

For more information on how Repackager isolates applications and the available isolation options, see Isolating Windows Installer Packages.

Configuring Exclusions

Repackaging exclusions refer to exclusions made during repackage time using the Repackaging Wizard. Any files, registry entries, .ini files, or shortcuts excluded at this point are not included in the Repackager project.

There are two methods of configuring exclusions:

- Configuring Exclusions Using Repackager
- Configuring Exclusions Using the Exclusions Editor

Configuring Exclusions Using Repackager

There are three types of exclusions used when repackaging a legacy installation:

Table 8-3 • Repackager Exclusion Types

Exclusion Type	Description
Repackaging Exclusions	Repackaging exclusions refer to exclusions made during repackage time using the Repackaging Wizard. Any files, registry entries, .ini files, or shortcuts excluded at this point are not included in the Repackager project. Therefore, if you exclude a directory you later need, you need to repackage the legacy setup again. The Repackager best practice is to capture everything using the Repackaging Wizard, and then exclude visually in the Repackager Interface. This way, you avoid having to run the Repackaging Wizard again if you accidentally exclude necessary files. In some cases, you may want to avoid capturing specific data types during repackaging. For example, your organization may never want to capture shortcuts. You can disable capture of shortcuts during repackage time, thereby eliminating the need to exclude them later. In Snapshot mode, you may want to limit the analysis to a certain directory to reduce the time it takes to capture the initial and final snapshot.
Project Exclusions	Each Repackager project can use a project exclusion list which marks files, registry entries, shortcuts, and .ini files as excluded in the Repackager project. If your process dictates that you capture everything and only exclude items in the Repackager Interface, then you should set up commonly captured but unnecessary items from the project by default. Because all the data from the original capture is intact, if you accidentally exclude necessary files, you can always reinclude them from the Repackager Interface and quickly rebuild your Windows Installer package.
Individual Project Exclusions	Because each project is different, and may require you to make decisions as to whether certain captured data is necessary, you can also selectively exclude or reinclude items on a per-package basis. These individual project exclusions allow you a fine-level of control as you prepare to build your Windows Installer package from the Repackager project.

Excluding Files

To exclude a captured file from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude a captured file from the InstallShield Editor project and Windows Installer package:***

1. Select **Files and Folders** from the View List. The **Files and Folders View** opens.
2. Expand the directory tree and select the directory containing the file you want to exclude.
3. In the file list, right-click the file and then click **Exclude**.

Excluding All Files in a Directory

To exclude all captured files in a directory from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude all captured files in a directory from the InstallShield Editor project and Windows Installer package:***

1. Select **Files and Folders** from the View List. The **Files and Folders View** opens.
2. Expand the directory tree and select the directory containing the files you want to exclude.
3. Right-click the directory and then click **Exclude**.

Excluding Directories and Subdirectories

To exclude all captured files and subdirectories within a directory from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude all captured files and subdirectories within a directory from the InstallShield Editor project and Windows Installer package:***

1. Select **Files and Folders** from the View List. The **Files and Folders View** opens.
2. Expand the directory tree to display the directory containing the files and subdirectories you want to exclude.
3. Right-click the directory and then click **Exclude All**.

Excluding Registry Keys

To exclude a registry key from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude a registry key from the InstallShield Editor project and Windows Installer package:***

1. Select **Registry Entries** from the View List. The **Registry Entries View** opens.
2. Expand the Registry tree to display the registry key you want to exclude.
3. Right-click the registry key and then click **Exclude**.

Excluding Registry Values

To exclude a captured registry value from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude a captured registry value from the InstallShield Editor project and Windows Installer package:***

1. Select **Registry Entries** from the View List. The **Registry Entries View** opens.
2. Expand the Registry tree and select the registry key containing the value you want to exclude.
3. In the **Registry Value** list, right-click the value and then click **Exclude**.

Excluding .ini Files

To exclude a captured .ini file from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude a captured .ini file from the InstallShield Editor project and Windows Installer package:***

1. Select **INI Files** from the View List. The **INI Files View** opens.
2. Expand the **INI Files** tree to display the .ini file you want to exclude.
3. Right-click the .ini file and then click **Exclude**.

Excluding .ini File Sections

To exclude a section in a captured .ini file from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude a section in a captured .ini file from the InstallShield Editor project and Windows Installer package:***

1. Select **INI Files** from the View List. The **INI Files View** opens.
2. Expand the **INI Files** tree to display the .ini file containing the section you want to exclude.
3. Right-click the section and then click **Exclude**.

Excluding Shortcuts

To exclude a captured shortcut from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude a captured shortcut from the InstallShield Editor project and Windows Installer package:***

1. Select **Shortcuts** from the View List. The **Shortcuts View** opens.
2. Expand the Shortcuts tree to display the shortcut you want to exclude.
3. Right-click the shortcut and then click **Exclude**.

Excluding All Shortcuts in a Directory

To exclude all captured shortcuts in a directory from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude all captured shortcuts in a directory from the InstallShield Editor project and Windows Installer package:***

1. Select **Shortcuts** from the View List. The **Shortcuts View** opens.
2. Expand the Shortcuts tree to display the directory containing the shortcuts you want to exclude.
3. Right-click the directory and then click **Exclude**.

Excluding Shortcuts from Subdirectories

To exclude all captured shortcuts within a directory or its subdirectories from the InstallShield Editor project and Windows Installer package, perform the following steps.

Task: ***To exclude all captured shortcuts within a directory or its subdirectories from the InstallShield Editor project and Windows Installer package:***

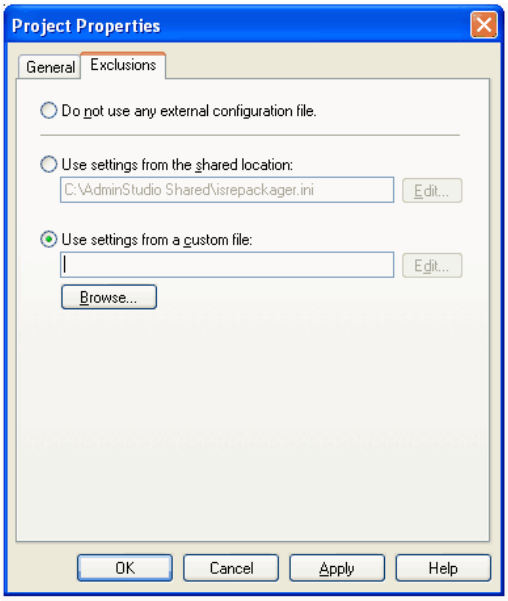
1. Select **Shortcuts** from the View List. The **Shortcuts View** opens.
2. Expand the **Shortcuts** tree to display the directory containing the shortcuts and/or subdirectories containing shortcuts you want to exclude.
3. Right-click the directory and then click **Exclude All**.

Specifying the External Configuration File

To specify an external configuration file which you want to use as a filter when converting legacy setups, perform the following steps.

Task: ***To specify an external configuration file which you want to use as a filter when converting legacy setups:***

1. From the Repackager **Project** menu, select **Properties**. The **General** tab of the **Project Properties** dialog box opens.
2. Open the **Exclusions** tab.



3. Select the **Use settings from the shared location** or the **Use settings from a custom file** option. The **Browse** button for that option is activated.
4. Click **Browse** and select the configuration file you want to use.

Tip • *After you select a configuration file, the **Edit** button is activated, enabling you to open the file in the Exclusions Editor.*

5. Click **OK**.

When you apply a configuration file, Repackager automatically updates all views to reflect the configuration file's exclusions. However, if you have already excluded items using Repackager, item states are retained.

Modifying External Configuration Files

To configure an external configuration file, perform the following steps.

Task: ***To configure an external configuration file:***

1. From Repackager's **Project** menu, click **Properties**. The **General** tab of the **Project Properties** dialog box opens.
2. Select the **Exclusions** tab.
3. Select the file you want to modify in either the **Use settings from the shared location** option or **Use settings from a custom file** options.
4. Click **Edit**. The **Exclusions Editor** opens.
5. Make necessary modifications using the **Exclusions Editor**.
6. When you finish editing the configuration file, click **OK**.
7. Click **OK** to close the **Project Properties** dialog box.

When you apply a configuration file, Repackager automatically updates all views to reflect the configuration file's exclusions. However, if you have already excluded items using Repackager, item states are retained.

Configuring Exclusions Using the Exclusions Editor

The Exclusions Editor allows you to configure three types of exclusions: Repackaging, Project, and OS Snapshot.

Repackaging Exclusions

Repackaging exclusions refer to exclusions made during repackage time using the Repackaging Wizard. Any files, registry entries, `.ini` files, or shortcuts excluded at this point are not included in the Repackager project. Therefore, if you exclude a directory you later need, you need to repackage the legacy setup again.

The Repackager best practice is to capture everything using the Repackaging Wizard, and then exclude visually in the Repackager Interface. This way, you avoid having to run the Repackaging Wizard again if you accidentally exclude necessary files.

In some cases, you may want to avoid capturing specific data types during repackaging. For example, your organization may never want to capture shortcuts. You can disable capture of shortcuts during repackage time, thereby eliminating the need to exclude them later. In Snapshot mode, you may want to limit the analysis to a certain directory to reduce the time it takes to capture the initial and final snapshot.

Project Exclusions

Each Repackager project can use a project exclusion list which marks files, registry entries, shortcuts, and `.ini` files as excluded in the Repackager project. If your process dictates that you capture everything and only exclude items in the Repackager Interface, then you should set up commonly captured but unnecessary items from the project by default. Because all the data from the original capture is intact, if you accidentally exclude necessary files, you can always reinclude them from the Repackager Interface and quickly rebuild your Windows Installer package.

OS Snapshot Exclusions

Like pre-capture repackaging exclusions, you can use the Exclusions Editor to configure exclusions to apply during the capture of OS snapshots. However, to maximize the usefulness of OS snapshots, you should avoid editing the default snapshot exclusion list (ISSnapshot.ini).

Exclusions and Repackager

Exclusions in Repackager refer to files, registry entries, shortcuts, and `.ini` files that are marked as excluded in the Repackager Interface by default when you open a Repackager project or if you change your exclusions file. The captured data is only marked as excluded and not ignored or discarded during capture. You can create an exclusion file in the Exclusions Editor, and link it to Repackager from the **Exclusions Tab** of the **Project Properties** dialog box in Repackager.

Exclusions and the OS Snapshot Wizard

When using the Exclusions Editor to configure analysis options for capturing OS snapshots, you are creating an exclusion list for files, directories, `.ini` files, `.ini` file sections, and registry data. Items in the exclusion list are not captured during the OS snapshot process, and will not be included in the OS snapshot file which is created.

Launching Exclusions Editor

The Exclusions Editor can be launched either within the Repackager interface or outside of Repackager. You can edit the default exclusions file, `isrepackager.ini`, using either interface.

However, if you want to create a new, custom exclusions file, you must launch the Exclusions Editor outside of Repackager.

- Launching Exclusions Editor Outside of Repackager
- Launching Exclusions Editor Within Repackager

Launching Exclusions Editor Outside of Repackager

To launch the Exclusions Editor outside of the Repackager interface, perform the following steps.

Task: ***To add a file to the exclusion list:***

1. Launch the Exclusions Editor by locating and executing the following file:

   AdminStudio Installation Directory\Repackager\AnalysisOptions.exe

   The **Files** tab of the Exclusions Editor opens.

2. Perform one of the following to open an exclusions file:

   - **Shared Exclusions**—To edit the shared exclusions file, on the **Files** menu, point to **Open** and click **Shared Exclusions**. The exclusions in the shared exclusions file are now listed on the **Files** tab.
   - **Custom Exclusions**—To create a new custom exclusions file, on the **Files** menu, click **New**. A default set of exclusions is listed.

3. Make edits to the file.
4. Save the file by selecting **Save** on the **File** menu.
5. If you were creating a custom exclusions file, specify a name and location for this exclusions file and click **Save**.

Launching Exclusions Editor Within Repackager

To add a file to the exclusion list, perform the following steps.

Task: ***To add a file to the exclusion list:***

1. Launch Repackager and open a project.
2. On the **Project** menu, click **Properties**. The **Project Properties** dialog box opens.
3. On the **Exclusions** tab, do one of the following:

   - To edit the default exclusions file, select **Use settings from the shared location** and click **Edit**.
   - To edit a custom exclusions file, select **Use settings from a custom file**, browse to the file you want to open (if it is not listed), and click **Edit**.

   The **Files** tab of the Exclusions Editor opens, with the appropriate exclusions file open.

4. Make edits to the file.
5. Save the file and close the Exclusions Editor by clicking **OK**.

Note • *Note that when opening the Exclusions Editor from within Repackager, there is no* ***File*** *menu displayed, meaning that you can only edit an existing exclusions file; you cannot create a new exclusions file.*

Excluding Files

You use the Exclusions Editor to create an exclusion list for files so that those files are not captured during the OS snapshot process, and will not be included in the OS snapshot file.

Task: ***To add a file to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor.
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.
3. Enter or browse to the directory **Path** containing the file you want to exclude.
4. Enter the name of the file you want to exclude, or browse to it by clicking the Browse (...) button to the right of the **Excluded Files** field. If you want to exclude multiple files from the same directory, separate them with pipes (|). If you want to exclude all files in a directory, enter an asterisk (*).
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusion appears in the **Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor.

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Excluding Files with Specific Extensions

To exclude files with specific extensions from the exclusion list, perform the following steps.

Task: ***To exclude files with specific extensions from the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.
3. In the **File Exclusion Information** dialog box, enter or browse to the directory containing the files you want to exclude. If you want to exclude files with a certain extension from all directories, enter an asterisk (*) for the **Path** value.
4. Enter an asterisk followed by the extension you want excluded in the **Excluded Files** field. For example, if you want to exclude all `.bak` files, enter `*.bak`. If you want to exclude multiple file types from the same directory (or from all directories), separate each exclusion with a pipe (|).
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusion appears in the **Files** tab.

6. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Excluding Directories

To add a directory to the exclusion list, perform the following steps.

Task: ***To add a directory to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.
3. In the **File Exclusion Information** dialog box, enter or browse to the directory **Path** containing the files you want to exclude.
4. Enter an asterisk (*) in the **Excluded Files** field.
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusion appears in the **Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Editing Existing File Exclusions

To edit an existing file exclusion, perform the following steps.

Task: ***To edit an existing file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Select the appropriate exclusion and click **Edit**. The **File Exclusion Information** dialog box opens.
3. In the **File Exclusion Information** dialog box, modify the **Path** and **Excluded Files** information.

4. Click **OK** to close the **File Exclusion Information** dialog box. The edited exclusion is listed in the **Files** tab.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Removing File Exclusions

To remove an existing file exclusion, perform the following steps.

Task: ***To remove an existing file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Select the appropriate exclusion and click **Delete**.
3. Confirm the exclusion by clicking **OK**. The deleted exclusion is removed from the list.
4. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring file exclusions for Repackager, you are only configuring Repackager to automatically mark the file as excluded; this can be changed from within Repackager on a file-by-file basis. However, when configuring file exclusions for the OS Snapshot Wizard, files in the exclusion list are not captured in the OS snapshot file.*

Excluding .ini Files

To add an .ini file to the exclusion list, perform the following steps.

Task: ***To add an .ini file to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **INI Files** tab.
3. Click **New**. The **INI File Exclusion Information** dialog box opens.
4. Enter or browse to the `.ini` file you want to exclude.

5. If there are specific sections you want to exclude from the `.ini` file, put the section names in brackets ([]) and separate them with pipes (|) in the **Excluded Sections** field. If you want to exclude all sections, put an asterisk (*) in the **Excluded Sections** field.
6. Click **OK** to close the **INI File Exclusion Information** dialog box. The new exclusion appears in the list on the **INI Files** tab.
7. Save the exclusions file as described in Launching Exclusions Editor.

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.*

Excluding Sections from .ini Files

To add a specific .ini file section to the exclusion list, perform the following steps.

Task: ***To add a specific .ini file section to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **INI Files** tab.
3. Click **New**. The **INI File Exclusion Information** dialog box opens.
4. Enter or browse to the `.ini` file containing the section you want to exclude.
5. To exclude a specific `.ini` file section, enter the section name in brackets ([]) in the **Excluded Sections** field. If there are multiple sections, separate them with pipes (|).
6. Click **OK** to close the **INI File Exclusion Information** dialog box. The new exclusion appears in the INI Files and Sections Excluded During Analysis dialog box.
7. Click **OK** to close the **INI File Exclusion Information** dialog box. The new exclusion appears in the list on the **INI Files** tab.
8. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.*

Editing Existing .ini File Exclusions

To edit an existing .ini file exclusion, perform the following steps.

Task: ***To edit an existing .ini file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor.
2. Open the **INI Files** tab.
3. Select the appropriate exclusion and click **Edit**. The **INI File Exclusion Information** dialog box opens.
4. In the **File Exclusion Information** dialog box, modify the **INI File** and **Excluded Sections** information.
5. Click **OK** to close the **INI File Exclusion Information** dialog box. The edited exclusion appears in the list on the **INI Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.*

Removing .ini File Exclusions

To delete an existing .ini file exclusion, perform the following steps.

Task: ***To delete an existing .ini file exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **INI Files** tab.
3. Select the appropriate exclusion and click **Delete**.
4. Confirm the exclusion by clicking **OK**. The deleted exclusion is removed from the list.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring .ini file exclusions for Repackager, you are only configuring Repackager to automatically mark the .ini file and/or sections as excluded; this can be changed from within Repackager on an .ini file by .ini file*

basis. However, when configuring .ini file exclusions for the OS Snapshot Wizard, .ini files in the exclusion list are not captured in the OS snapshot file.

Excluding Registry Data

To add registry data to the exclusion list, perform the following steps.

Task: ***To add registry data to the exclusion list:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **Registry** tab.
3. Click **New**. The **Choose Registry Key** dialog box opens.
4. Enter or browse to the registry key you want to exclude and click **OK**. The key is added to the list on the **Registry** tab.
5. If you want to exclude a certain value in the key, select it from the list and click Edit. The **Edit Registry Key** dialog box opens.
6. Provide the **Value Name** you want to exclude, and click **OK** to close the dialog box. The exclusion information is reflected in the list on the **Registry** tab.
7. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring registry exclusions for Repackager, you are only configuring Repackager to automatically mark the registry entry and/or values as excluded; this can be changed from within Repackager on an registry key by registry key basis. However, when configuring registry exclusions for the OS Snapshot Wizard, registry data in the exclusion list is not captured in the OS snapshot file.*

Editing Existing Registry Exclusions

To edit existing registry exclusions, perform the following steps.

Task: ***To edit an existing registry exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **Registry** tab.
3. Select the registry key that you want to edit and click **Edit**. The **Edit Registry Key** dialog box opens.

4. Modify the exclusion as necessary and click **OK**. The edited information is reflected in the list on the **Registry** tab.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring registry exclusions for Repackager, you are only configuring Repackager to automatically mark the registry entry and/or values as excluded; this can be changed from within Repackager on an registry key by registry key basis. However, when configuring registry exclusions for the OS Snapshot Wizard, registry data in the exclusion list is not captured in the OS snapshot file.*

Removing Registry Exclusions

To delete an existing registry exclusion, perform the following steps.

Task: ***To delete an existing registry exclusion:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. Open the **Registry** tab.
3. Select the registry key that you want to delete and click **Delete**.
4. Confirm the deletion by clicking **OK**. The deleted exclusion is removed from the list.
5. Save the exclusions file as described in Launching Exclusions Editor:

Note • *When configuring registry exclusions for Repackager, you are only configuring Repackager to automatically mark the registry entry and/or values as excluded; this can be changed from within Repackager on an registry key by registry key basis. However, when configuring registry exclusions for the OS Snapshot Wizard, registry data in the exclusion list is not captured in the OS snapshot file.*

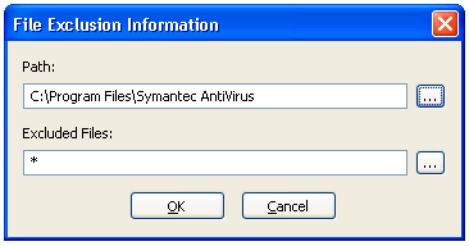
Repackaging and Anti-Virus Software

Any machine that you use to repackage most likely has anti-virus software installed on it, even a “clean” machine. During repackaging, the real-time virus detection feature of anti-virus software could automatically update various cached files in its directories.

Therefore, in order to avoid repackaging errors when using the Snapshot repackaging method, you should exclude the software directories containing your anti-virus software.

Task: ***To exclude anti-virus software directories:***

1. Launch the Exclusions Editor and open an exclusions file by performing the steps listed in Launching Exclusions Editor:
2. On the **Files** tab, click **New**. The **File Exclusion Information** dialog box opens.



3. Enter or browse to the directory **Path** containing the anti-virus files that you want to exclude. For example, if you wanted to exclude Symantec AntiVirus software, you would select the following directory:

   `C:\Program Files\Symantec AntiVirus`

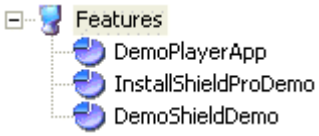
4. Enter an asterisk (*) in the **Excluded Files** field.
5. Click **OK** to close the **File Exclusion Information** dialog box. The new exclusions appear on the **Files** tab.
6. Save the exclusions file as described in Launching Exclusions Editor:

Important • *It is strongly recommended that you leave your anti-virus software running during repackaging.*

Scanning InstallShield Professional Setups for Additional Data

If you are repackaging a legacy installation that was originally created using InstallShield Professional 5.5 and later, you can use the **SmartScan Wizard** to scan the setup for possible additional files, `.ini` files, shortcuts, or registry data (such as for language-specific setups or platform-specific setups).

The SmartScan Wizard is also able to group files, shortcuts and registry entries into features corresponding to InstallShield Professional components. Any items (files, folders, shortcuts, or registry entries) that are attached to a component directly or indirectly (through File Groups), are attached to their corresponding features.



Automatic Launch of the SmartScan Wizard

For your convenience, if a Repackaging project is opened (either by using Open on the File menu or by using the Repackaging Wizard), and the project meets certain conditions, the SmartScan Wizard is automatically launched. The SmartScan Wizard is automatically launched:

- if the original setup was an InstallShield Professional 5.5 or later installation, *and*
- the file has not already been repackaged using the **InstallShield Professional Logging Method** (available for InstallShield Editor and DevStudio 9.x InstallScript installations only), *and*
- the SmartScan Wizard has not already been run on this project

Task: ***To scan an InstallShield Professional setup:***

1. From the **Project** menu, select **SmartScan Wizard**. The **Welcome Panel** appears.
2. Click **Next**. The **Original InstallShield Professional Setup Panel** appears.
3. The **Specify the path of the original setup executable** field will be filled in by default if that information exists. Browse to the InstallShield Professional setup you want to scan, and specify a password if required.
4. In the **Original target folder** field, enter the directory where the product was installed during the repackaging process. In most cases, this will be a subdirectory of `[ProgramFilesFolder]`.
5. Click **Next**. The **Scanning Panel** appears.
6. When scanning is complete, click **Next**. The **Setup Feature Tree Panel** appears, listing the features and components that were configured in the InstallShield Professional project.
7. Select the **Add feature tree** option if you want to use the feature information defined in the Professional media.
8. Click **Next**. The **Scanning Media Panel** appears, showing the results of Repackager's search for any extra setup information in the media.
9. Click **Finish** to apply the results to the project.
   - The setup is scanned, and any additional files, shortcuts, `.ini` files, or registry data is added to the appropriate view.
   - The data captured by the SmartScan Wizard appears in the color designated in the **SmartScan items** field on the **Colors** tab of the **Options** dialog box.
   - If you selected the **Add feature tree** option, the files, shortcuts and registry entries are grouped into features corresponding to the InstallShield Professional components.

Note • *After you have scanned an InstallShield Professional project and included data from it in your Repackager project, you can save the project. When you reopen it, this collected data no longer retains the "scanned" color, since it is now part of the Repackager project.*

Creating an InstallShield Editor Template to Use Within Repackager

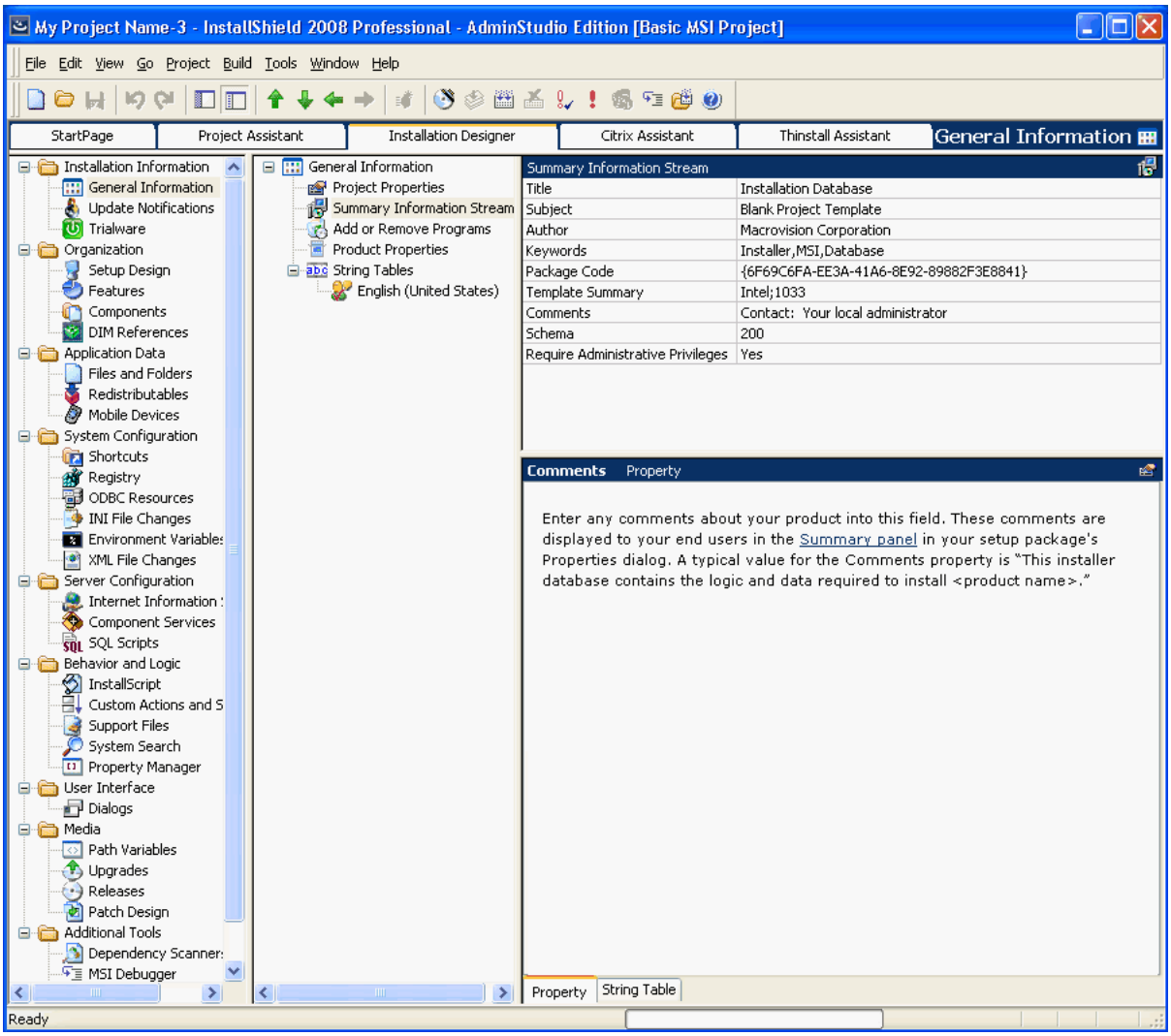
One of the main reasons you use AdminStudio is to significantly reduce the time it takes to package an application for deployment. You can use the following procedure to speed up the packaging process even more.

You can create an InstallShield Editor template that you can use within the Repackager interface to save additional time when customizing a package. By using this template, all future InstallShield Editor `.ism` project files generated by Repackager will contain the company-specific default settings that were specified in the template. Using a template is also beneficial for organizations with multiple packagers, since it helps enforce consistency by enabling all packagers to make the same standard customizations to packages.

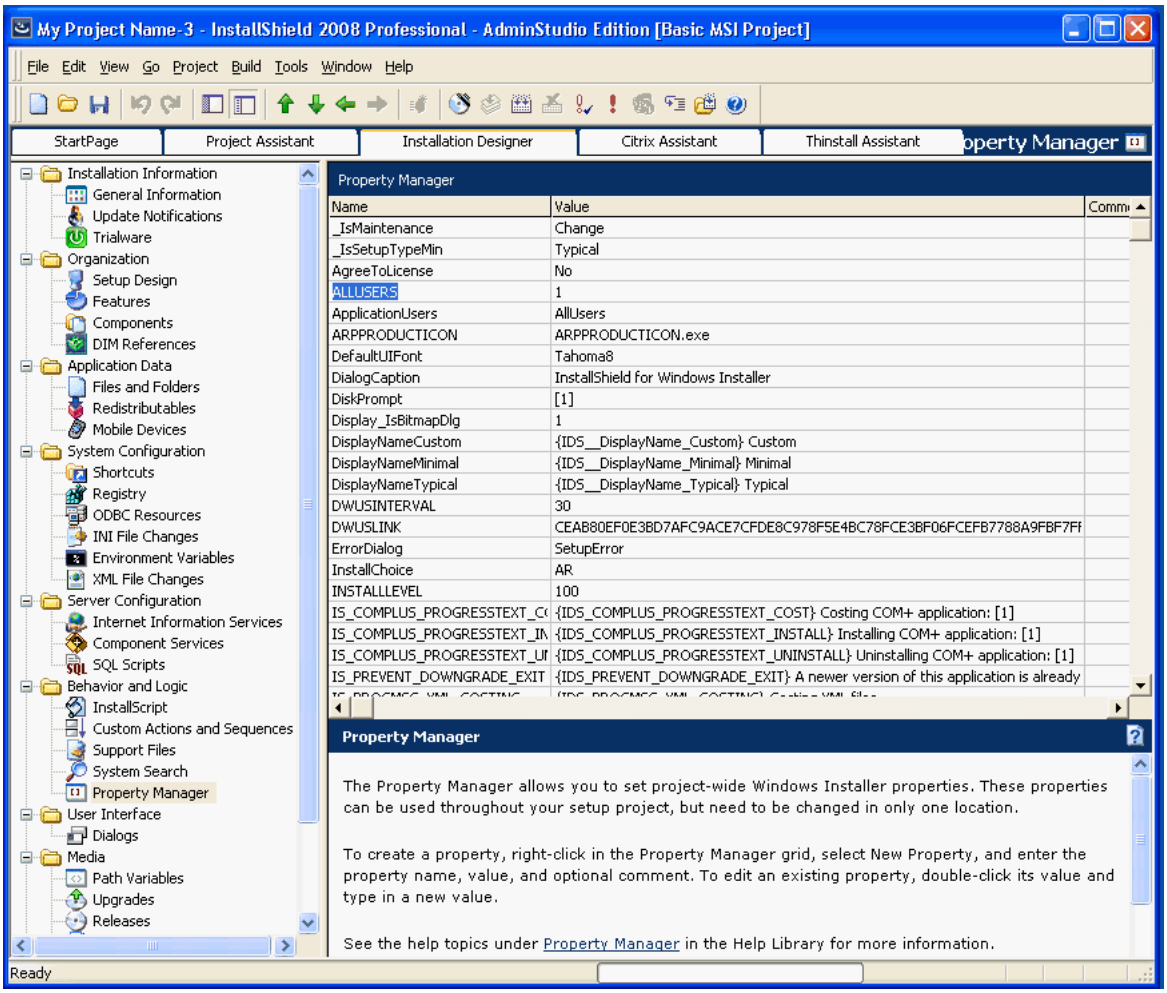
To create an InstallShield Editor template to use within Repackager, perform the following steps.

Task: ***To create a customized InstallShield Editor template:***

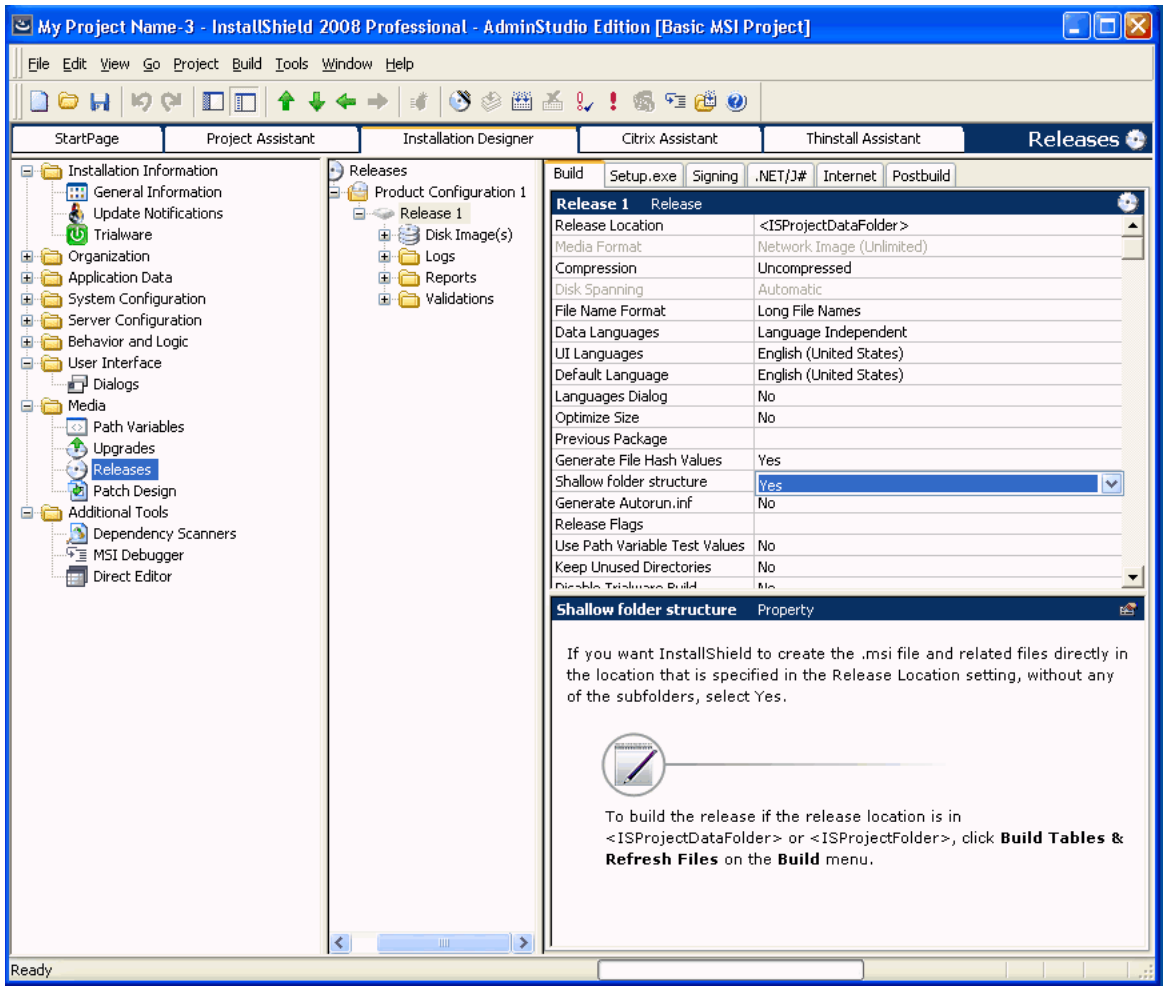
1. Create a new **Basic MSI Project** in the InstallShield Editor.
2. On the **Installation Designer** tab, select the **General Information** node under **Installation Information**, and enter your company-specific information as required.



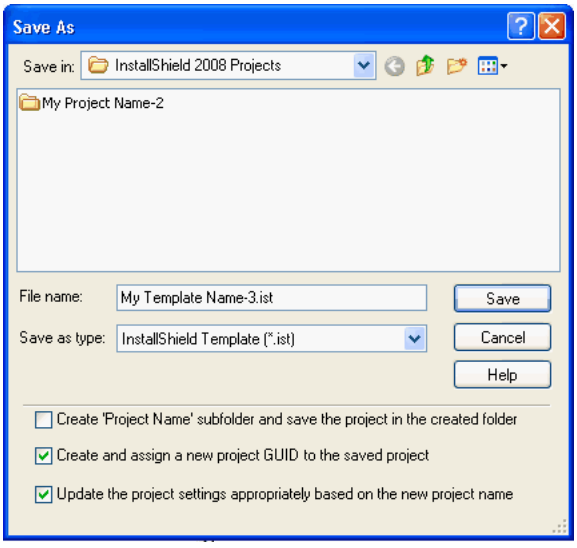
3. Under **Behavior and Logic**, select the **Property Manager** node and add the required properties like ALLUSERS, ISSCRIPTDRIVEN, etc.



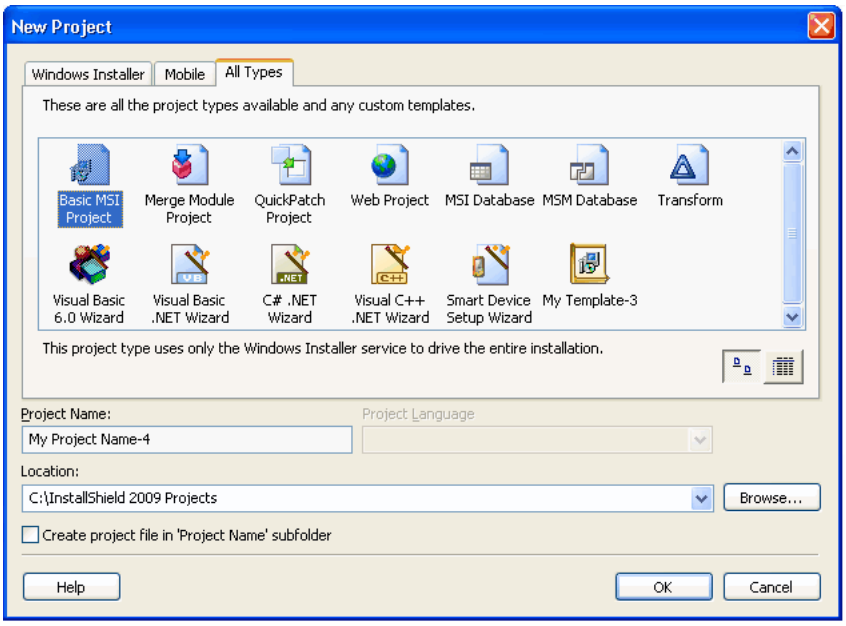
4. You can also optionally set **Shallow Folder Structure** to **Yes** in the **Releases** view under **Media**.



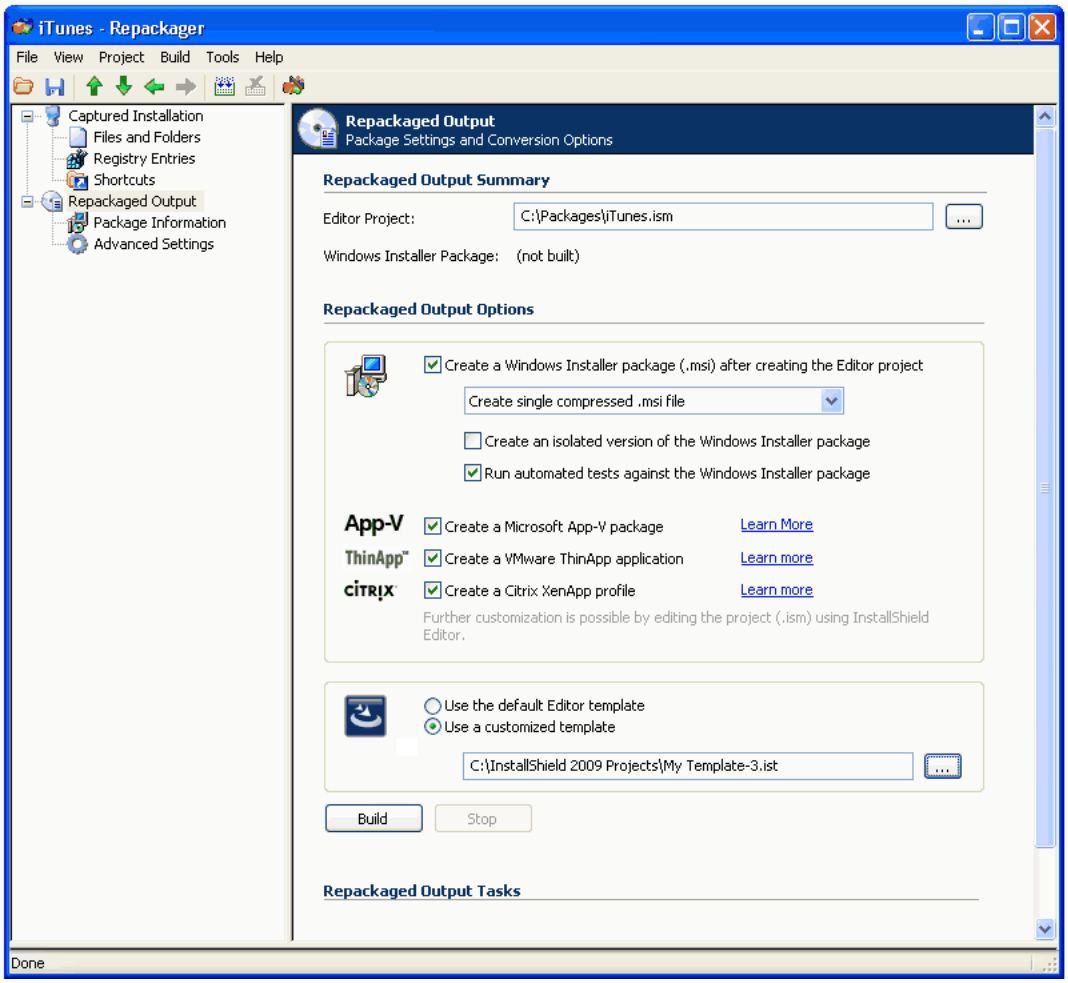
5. After making all required changes, save the project as an InstallShield Editor Template (`.ist`) type.



6. This new template should now be available along with other project types in the InstallShield Editor.



7. From within the Repackager interface, you can start using this customized template by selecting the **Use a customized template** option in the **Repackaged Output** view, and selecting the InstallShield Editor template that you just created.



Repackager Interface Reference

This section describes each of the dialog boxes and Wizard panels that you might encounter when using the Repackager interface. The help topics in this section are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog box.

- Repackager Interface
- SmartScan Wizard
- Setup Intent Wizard
- VMware Repackaging Wizard
- Exclusions Editor Interface
- Options.ini File

- Files Associated with Repackager
- Using InstallShield to Chain Multiple Windows Installer Packages Together
- Troubleshooting

Repackager Interface

From the Repackager Interface, you can:

- Open the Repackaging Wizard and repackage legacy setups.
- Open the Exclusions Editor and configure exclusions.
- Convert Novell ZENworks, Microsoft SMS, and WinINSTALL projects into Repackaging projects.
- Create a package exclusion list.
- Build a Repackager project into an InstallShield Editor project and Windows Installer package.

The Interface consists of several menus, a toolbar, the status bar, the output window, the View List, and several associated views.

- Menus and the toolbar are discussed in the Menus and Toolbar topic.
- Individual views are covered in their respective help topics.
- The status bar, output window, and View List are described in the following table.

Table 8-4 • Repackager Interface Elements

Interface Element	Description
Status Bar	The status bar, which can be toggled from the View menu, displays information when you hover over buttons in the toolbar.

Table 8-4 • Repackager Interface Elements

Interface Element	Description
View List	The View List allows you to navigate to different views in the Repackager project. The corresponding view is displayed when you select an item in the tree. You can also use the Forward, Back, Navigate Up, and Navigate Down buttons in the View List. The View List includes the following views: • Captured Installation View • Files and Folders View • Registry Entries View • Shortcuts View • INI Files View • Deleted Files View • Deleted Registry Entries View • Repackaged Output View • Package Information View • Advanced Settings View
Output Window	When you open Repackager 3.x output, Novell ZENworks projects, Microsoft SMS projects, WinINSTALL projects, or Wise installation projects in the Repackager Interface, conversion information appears in the Output window. This window can be toggled from the View menu.

Repackager Start Page

When you first launch Repackager, the **Repackager Start Page** opens.

This page gives you a brief overview of Repackager functionality and uses, and gives you links to launch the Repackaging Wizard, convert a Windows Installer package to an application virtualization format, open an existing Repackager project, convert a legacy setup to a Repackager project, upgrade a legacy Repackager file, and open a recently accessed package.

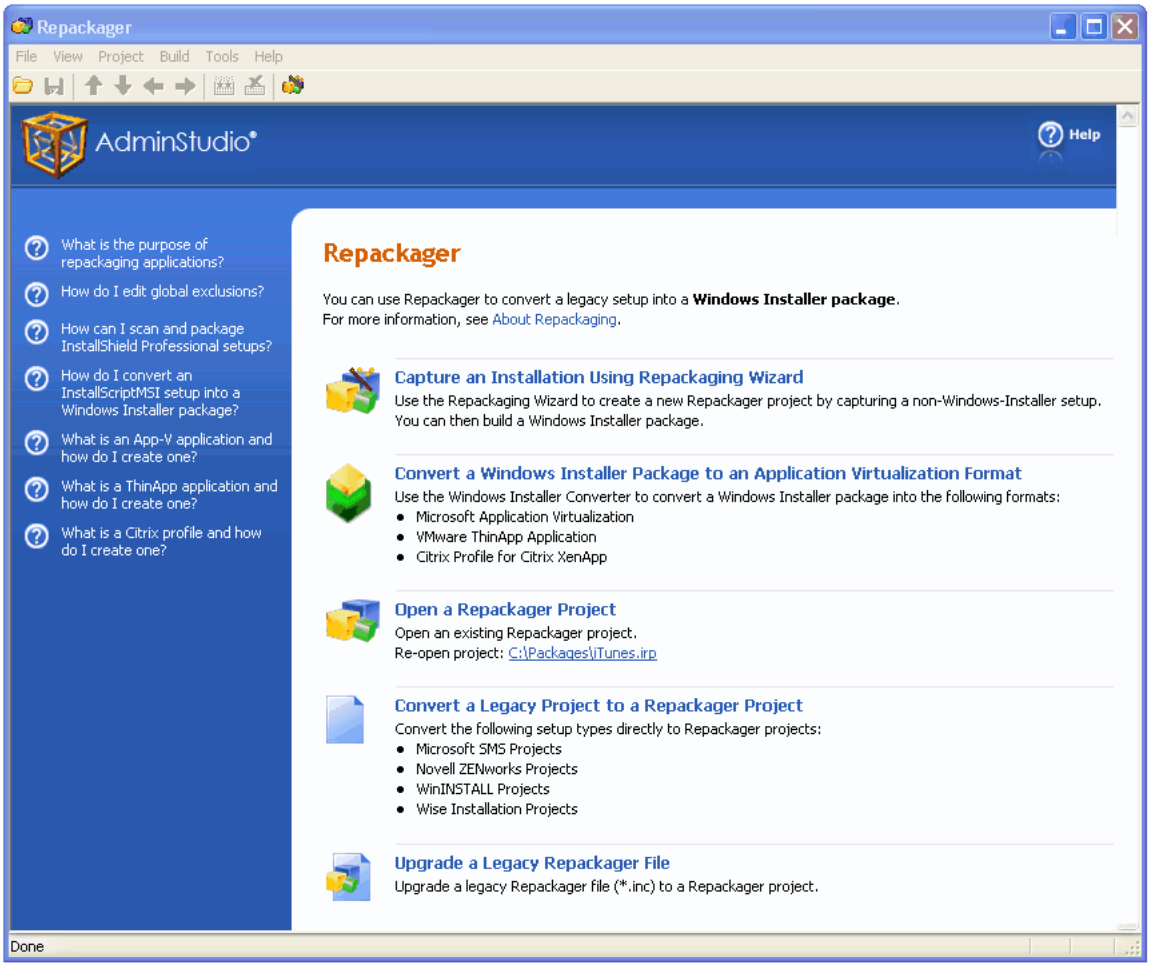


Figure 8-3: Repackager Start Page

Menus and Toolbar

The following table provides a description of each menu command and toolbar button:

Table 8-5 • Repackager Menus and Toolbars

Menu	Command	Toolbar Button	Keyboard Shortcuts	Description
File	**Open**		Ctrl+O	Allows you to open: • An existing Repackager project (`.irp`) • Repackager 3.x output (`.inc`) • Novell ZENworks 3.0, 3.1, or 4.0 project (`.axt/.aot`) • Microsoft SMS project (`.ipf`) • WinINSTALL converted project (`.txt`) (6.0, 6.5, or 7.x) • Wise Installer project (`.wse`)
File	**Save**		Ctrl+S	Saves the current project.
File	**Save As**			Saves the current project using the name and location you specify.
File	**1,2,3,4**			Allows you to open the four most recently accessed Repackager projects.
File	**Exit**			Exits Repackager.
View	**Toolbar**			Toggles display of the toolbar.
View	**Status Bar**			Toggles display of the status bar.
View	**Output**			Toggles display of the Output window.
View	**Refresh**		F5	Refreshes the current view.
Project	**Edit Windows Installer Package**			Once you build the Repackager project into a Windows Installer package (`.msi`), opens the package in InstallShield Editor (in Direct MSI Edit mode).
Project	**Edit InstallShield Project**			Once you build the Repackager project into an InstallShield Editor project (`.ism`), opens the project in InstallShield Editor.

Table 8-5 • Repackager Menus and Toolbars

Menu	Command	Toolbar Button	Keyboard Shortcuts	Description
Project	**Setup Intent Wizard**			Launches the Setup Intent Wizard.
Project	**SmartScan Wizard**			Launches the **SmartScan Wizard**, which allows you to scan an InstallShield Professional setup for additional files that may not have been captured during repackaging because they may be language- or platform-specific.
Project	**Create Report**		Ctrl+R	Allows you to create a report for the project in text or HTML format.
Project	**Properties**			Displays properties for the current project, including exclusion information.
Build	**Build**		F7	Builds the Repackager project into an InstallShield Editor project and a Windows Installer package.
Build	**Stop Build**		Ctrl+Break	Terminates an in-process build.
Tools	**Repackaging Wizard**			Launches the Repackaging Wizard.
Tools	**VMware Repackaging Wizard**			Launches the VMware Repackaging Wizard.
Tools	**Options**			Displays the Options dialog box.
Tools	**Isolation Options**			Displays the Isolation Options dialog box, where you can specify assembly and digital signature isolation options.
Help	**Contents**			Launches the Help Library, displaying the Contents tab.
Help	**Index**			Launches the Help Library, displaying the Index tab.
Help	**Search**			Launches the Help Library, displaying the Search tab.

Table 8-5 • Repackager Menus and Toolbars

Menu	Command	Toolbar Button	Keyboard Shortcuts	Description
Help	**Support Central**			Accesses the AdminStudio Support Web site.
Help	**Web Community**			Accesses the AdminStudio Web Community.
Help	**ReadMe**			Displays the AdminStudio ReadMe file.
Help	**Feedback**			Accesses the feedback form on the AdminStudio Web site.
Help	**AdminStudio on the Web**			Accesses the AdminStudio Web site.
Help	**About Repackager**			Displays the About Repackager dialog box.
	Up			Moves you up one view in the View List.
	Down			Moves you down one view in the View List.
	Back			Displays the previously displayed view in the View List.
	Forward			Returns you to the view from which you selected the Back button.

Dialog Boxes

Repackager includes the following dialog boxes to assist you in your project creation:

- Create Report Dialog Box
- Isolation Options Dialog Box
- Options Dialog Box
- Password Required Dialog Box
- Project Properties Dialog Box
- WinINSTALL Conversion Dialog Box

About Repackager Dialog Box

This dialog box available by selecting **About Repackager** from the **Help** menu, displays version information for Repackager.

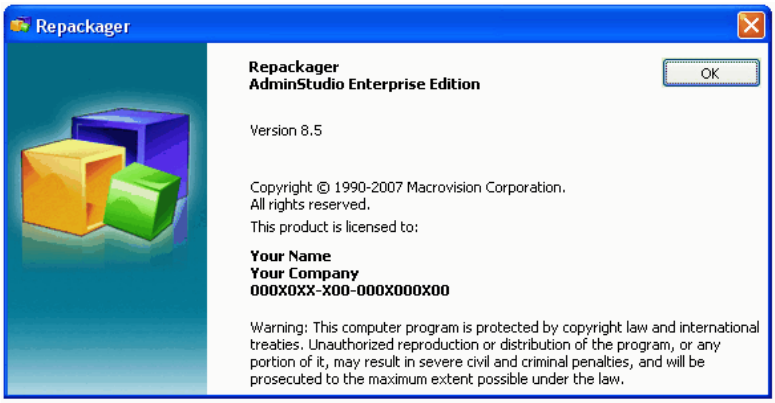


Figure 8-4: About Repackager Dialog Box

Create Report Dialog Box

The Create Report dialog box, available by selecting Create Report from the Project menu, allows you to configure a report for the current Repackager project, or a specific subset of captured data.

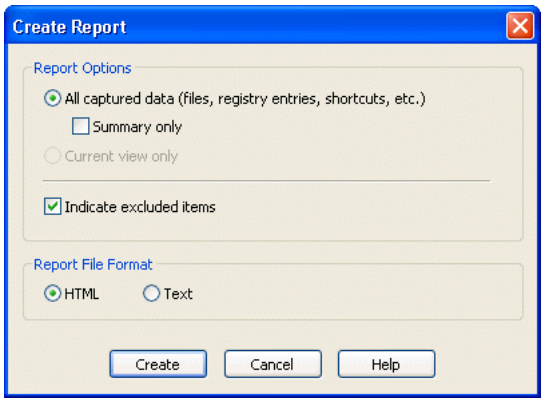


Figure 8-5: Create Report Dialog Box

This dialog box contains the following options:

Table 8-6 • Create Report Dialog Box Options

Option	Description
All captured data (files, registry entries, shortcuts, etc.)	Select to have the report include all captured data.

Table 8-6 • Create Report Dialog Box Options

Option	Description
Summary only	If you select All captured data, you can select this option to only display summary information in the report (the number of items captured and the number of items excluded for files, .ini files, registry data, and shortcuts).
Current view only	Select this option to include only the currently selected view in the report.
Indicate excluded items	Select to display items that have been marked as excluded in Repackager.
Report File Format	Select the file format for Repackager reports: HTML or Text.
Create	When you click Create, you are prompted for a name and location for the outputted report.

Isolation Options Dialog Box

Application isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

On the Isolation Options dialog box, which is opened by selecting **Isolation Options** from the **Tools** menu, you can specify the following Repackager isolation options:

- **Assembly Options**—Specify the type of assemblies Repackager will create, and the assembly naming conventions. See Manifest Options Tab
- **Digital Signature Options**—Configure the certificate information required when using shared assemblies. See Digital Signature Tab.

Note • *The modifications you make on the Isolation Options dialog box will be recorded in the `isolationconfig.ini file`, which is stored in the `AdminStudio Shared` directory.*

Manifest Options Tab

The Manifest Options tab allows you to configure several settings associated with manifests. The following settings are included:

Table 8-7 • Isolation Options Dialog Box / Manifest Options Tab

Option	Description
Assembly Type	This option allows you to select the type of assemblies that Repackager will create and use: • Create private side-by-side assemblies in the application folder • Create shared side-by-side assemblies in the WinSxS folder (Default) ***Note*** • *Manifests for shared assemblies must be digitally signed. This can be done in the Digital Signature Tab.* ***Note*** • *A 2048-bit key is required to sign a Windows XP assembly/manifest being installed to the WinSxS folder.*
Assembly Naming Conventions	Specify your company and division information to define the default naming convention that Repackager will use when creating assemblies during application isolation By default, assembly names are specified in the form of: `Company.Division.Assembly` ***Note*** • *See see About Assemblies and About Manifests for more information.*
Create a new component for each assembly	Select this option if you want to create a new component for each assembly created during isolation. This check box applies to all assemblies created. ***Caution*** • *If you are creating assemblies for applications files within multiple components, this option must be selected for successful application isolation.* *If you are planning to deploy this isolated package to operating systems prior to Windows XP, always check this box.*

Digital Signature Tab

On the **Digital Signature** tab, you can configure the certificate information required when using shared assemblies.This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly.

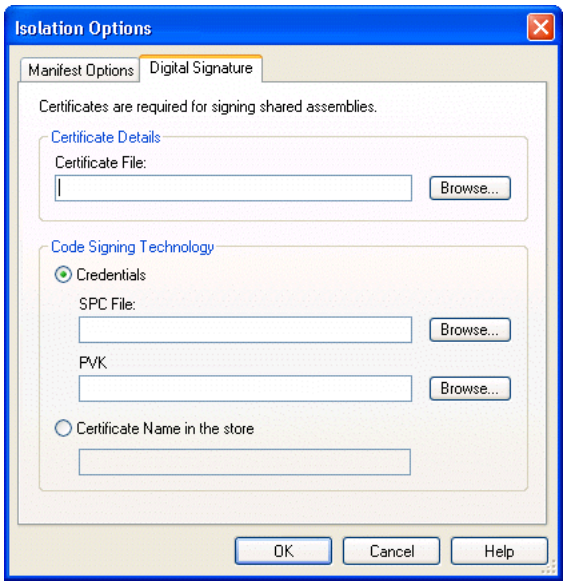


Figure 8-6: Isolation Options Digital Signature Options Tab

Caution • *Repackager uses timestamping when signing global assemblies. Consequently, you must have an Internet connection on the computer when you create a global assembly.*

You must configure the following options when signing these assemblies:

Table 8-8 • Isolation Options Dialog Box / Digital Signatures Tab

Item	Description
Certificate File	Click the Browse (...) button next to the field and navigate to the certificate file you are using to sign assemblies. A digital certificate identifies you and/or your company to end users and assures them the data they are about to receive has not been altered.
Credentials	Select this option to use credential files as the code signing technology. If you select this option, you must supply the name and location of both your software publishing credential files: **SPC File** and **PVK** File. ***Note*** • *In order to receive a software publishing credentials and a private key, you must supply a certification authority, such as VeriSign, with specific information about your company and software.*
SPC File	Specify the name and location of your software publishing credentials file (`.spc`).
PVK	Specify the name and location of your private key file (`.pvk`).

Table 8-8 • Isolation Options Dialog Box / Digital Signatures Tab

Item	Description
Certificate Name in the Store	Select this option to use the name of an existing certificate file in the Certificate Store as the code signing technology. The Certificate Store is a central repository for certificate files. Using a Certificate Store allows you to reuse the certificate files for different purposes as necessary.

Note • *For more information, see About Digital Certificates.*

Options Dialog Box

The Options dialog box, available from the **Tools** menu, presents options on three tabs: **Colors**, **Merge Modules**, and **Build Options**.

Colors Tab

On the **Colors** tab, you can configure the color of scanned items and deleted items in Repackager's exclusion views (**Files**, **.ini Files**, **Registry Data**, and **Shortcuts**).

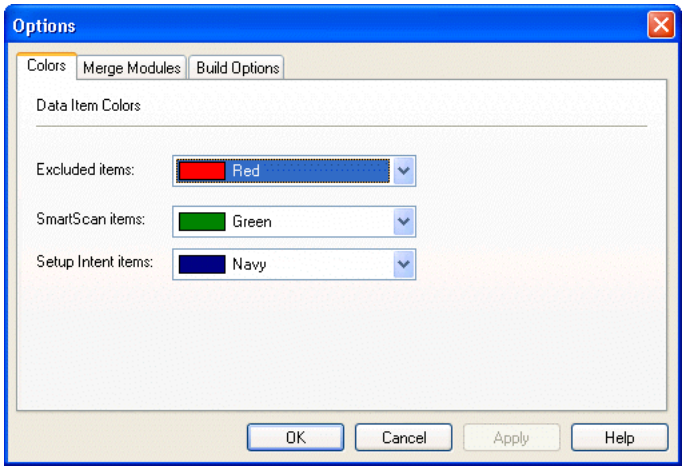


Figure 8-7: Colors Tab of the Options Dialog Box

Merge Modules Tab

On the **Merge Modules** tab, you can specify additional directories containing custom merge modules to use during repackaging.

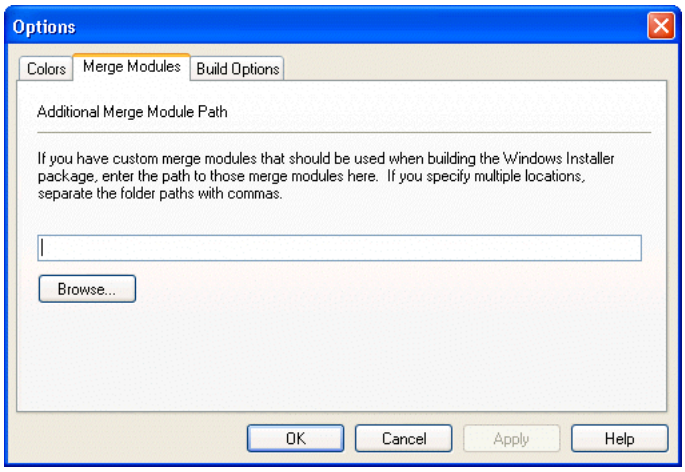


Figure 8-8: Merge Modules Tab of the Options Dialog Box

Build Options Tab

On the **Build Options** tab, you can specify whether or not you want to list ICE validation warnings in the Repackager output window during the Build process.

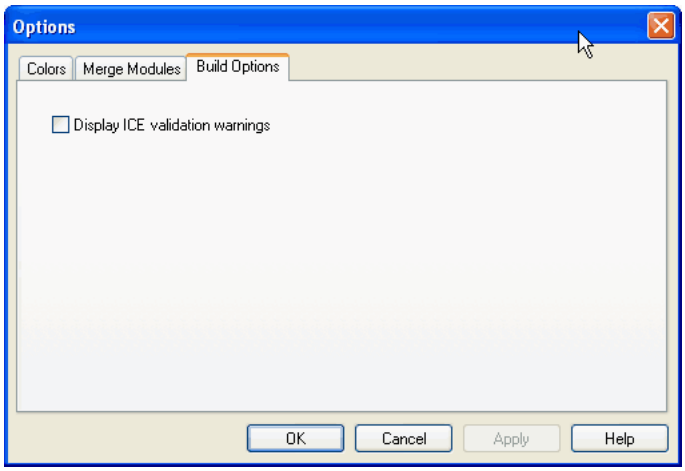


Figure 8-9: Build Options Tab of the Options Dialog Box

To display any ICE validation warnings that occur during the Repackager Build process, select the **Display ICE validation warnings** option. By default, this option is not selected.

Password Required Dialog Box

The Password Required dialog box is displayed when attempting to run the SmartScan Wizard on an InstallShield Professional setup which requires a password. Provide the password for the setup and click OK to proceed.

For InstallShield Professional 6.x and 7.x setups, the SmartScan Wizard can bypass password protection, and will not prompt you for a password even if one is required to run the installer.

Project Properties Dialog Box

The Project Properties dialog box, accessed by selecting Properties from the Projects menu, contains two tabs:

Table 8-9 • Project Properties Dialog Box Tabs

Tab	Description
General Tab	Allows you to view properties for the current Repackager project.
Exclusions Tab	Use to configure the location of the default exclusion file.

General Tab

The General tab of the Project Properties dialog box displays information about the current Repackager project (`.irp`).

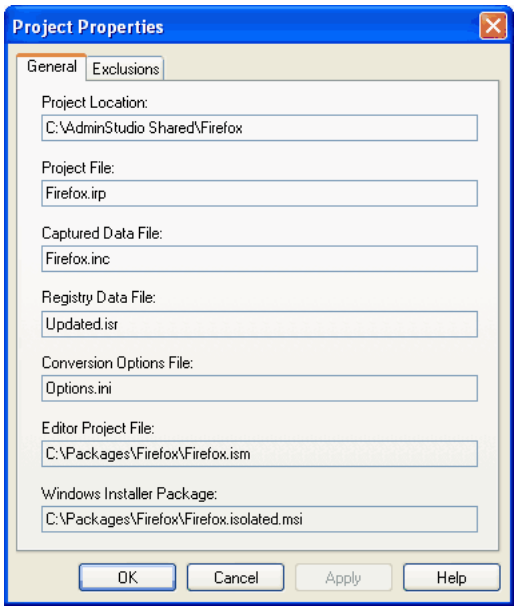


Figure 8-10: Project Properties Dialog Box General Tab

The following options are displayed:

Table 8-10 • General Tab Options

Option	Description
Project Location	The full path of the current Repackager project file (`.irp`).
Project File	The name of the current Repackager project file.

Table 8-10 • General Tab Options

Option	Description
Captured Data File	The name and location of the captured data file (`.inc`), which was either created by the Repackaging Wizard or during conversion of a Novell ZENworks project, Microsoft SMS project, or WinINSTALL project. The path is relative to the current Repackager project file.
Registry Data File	The name and location of the file containing captured registry data. The path is relative to the current Repackager project file.
Conversion Options File	The name and location of the Options.ini file, which contains an exhaustive list of all options you can use during conversion of the Repackager project to an InstallShield Editor project and Windows Installer package.
Editor Project File	The name and location of the InstallShield Editor project file as set in the Product View (MSI Package). The path is relative to the current Repackager project file.
Windows Installer Package	The name and location of the Windows Installer package. The path is relative to the current Repackager project file.

Exclusions Tab

The Exclusions tab allows you to select an exclusion file to use as a filter when importing captured data into a Repackager project.

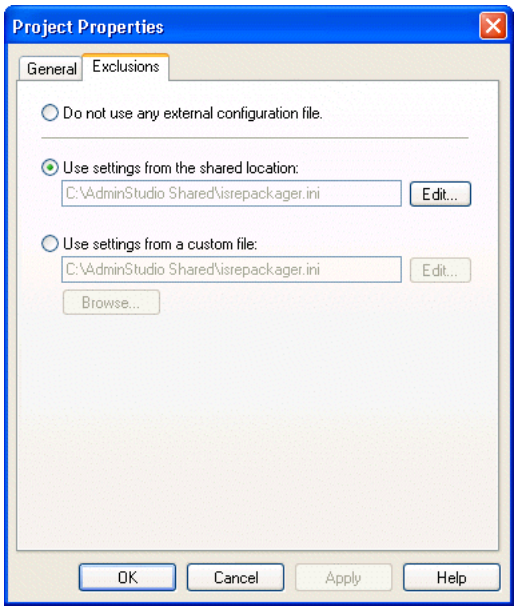


Figure 8-11: Project Properties Dialog Box Exclusions Tab

Select one of the following options for the configuration file:

Table 8-11 • Exclusions Tab Properties

Option	Description
Do not use any external configuration	Repackager will import all captured data into the Repackager project.
Use settings from the shared location	Repackager will use the settings contained in isRepackager.ini in the `AdminStudio Shared` directory (configured during installation). Use this option when you are working in a team environment where the exclusion list needs to be stored in a centralized location.
Use settings from InstallShield defaults	Repackager will use the settings contained in the default.ini file in the Repackager folder. These are the InstallShield Editor-recommended exclusions. It is recommended that you do not modify these exclusions so you can return to them if you need to restart your exclusion list.
Edit	Click to open the Exclusions Editor, which you can use to exclude files, registry entries, .ini files, or shortcuts from the Repackager project. See Configuring Exclusions Using the Exclusions Editor and Exclusions Editor Interface for more information.
Use settings from a custom file	Specify or browse to a file created with the Exclusions Editor that you want to use as your filter during conversion to a Repackager project. You would create a custom exclusion file based upon your company's requirements.

Caution • *Using the custom settings option, it is possible to use the local settings file (`isRepackager.ini`) in the Windows directory. This file is also used for default exclusions for the Repackaging Wizard. By modifying this file, you introduce the possibility of excluding data at repackaging time in subsequent Repackaging Wizard executions, as opposed to marking items as excluded in a Repackager project (which does not affect the captured data). For this reason, it is highly recommended that you do not use the `isRepackager.ini` configuration file in the local Windows folder for your Repackager exclusions.*

WinINSTALL Conversion Dialog Box

When you convert a WinINSTALL project to a Repackager project, this dialog box appears to allow you to set WinINSTALL-specific variables. These variables are:

Table 8-12 • WinINSTALL Variables

Variable	Description
@Server	The machine name of the server where the WinINSTALL directory is located.

Table 8-12 • WinINSTALL Variables

Variable	Description
@WinstallDir	The location of the directory where the WinINSTALL executables are located.

Repackager Views

Repackager includes several views, from which you can examine the captured data that will be used to create an InstallShield Editor project (`.ism`) and Windows Installer package (`.msi`). Depending on the presence or absence of certain data types, some views may not be displayed. For example, if the setup does not include any `.ini` files, the INI Files view will not be displayed in the View List.

The following views are available in Repackager:

- Captured Installation View
- Files and Folders View
- Registry Entries View
- Shortcuts View
- INI Files View
- Deleted Files View
- Deleted Registry Entries View
- Repackaged Output View
- Package Information View
- Advanced Settings View

Note • *Information listed in the views (such as files, .ini files, or registry entries) is limited to 267 characters in length. Anything longer than this limit will be truncated in the view. The full value can be viewed in InstallShield Editor.*

Captured Installation View

From the **Captured Installation** view, you can review summary information about the setup you are converting into a Windows Installer package.

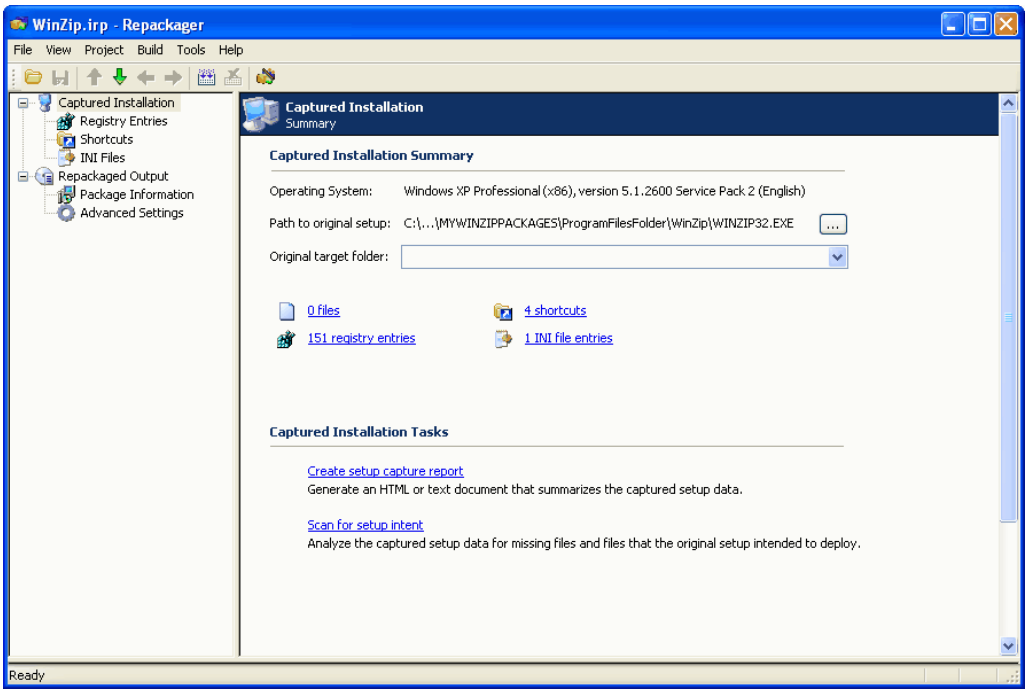


Figure 8-12: Repackager Captured Installation View

The **Captured Installation** view lists the following information:

Table 8-13 • Repackager Captured Installation View

Property	Description
Operating System	Identifies the operating system—including version, service pack, and processor type (32 or 64-bit)—of the machine where the capture was performed.
Path to original setup	Location of setup that was repackaged. Click the browse (...) button to select a different setup.
Original target folder	From this list, select the original target folder for the installation. In most cases, this will be a subdirectory of `[ProgramFilesFolder]`. Alternatively, you can enter your own target. This value will be set as the value for `INSTALLDIR`, and is a mandatory property. ***Note*** • *Information about the provided install locations can be found in the SystemFolder Property topic of the Windows Installer Help Library.*

Table 8-13 • Repackager Captured Installation View

Property	Description
Number of Files, Shortcuts, Registry Entries, and INI File Entries	Links that list the number of files, shortcuts, and registry entries captured, and the number of `.ini` file changes made. Click these links to open the following subviews: • Files and Folders View • Registry Entries View • Shortcuts View • INI Files View Each subview of this view allows you to view the names and associated information of each item captured, and selectively exclude (or reinclude) these items from the ultimate Windows Installer package. If no entries were captured of a particular type, the corresponding view does not appear in the View List. For example, if no `.ini` file changes were captured, the **INI Files** view is not displayed.
Create setup capture report	Click to generate the **Setup Capture Report**, an HTML or text document that summarizes the data that was captured when a setup was repackaged. For more information, see Creating a Setup Capture Report for a Project. ***Edition*** • *The Setup Capture Report feature is included with AdminStudio Standard, Professional, and Enterprise Editions.*
Scan for setup intent	Click to launch the **Setup Intent Wizard**, which you can use to scan a setup to identify files that may not have been captured during repackaging—effectively recognizing the installation's intent for these files. For more information, see Using the Setup Intent Wizard to Detect File Dependencies in a Repackager Project. ***Edition*** • *The **Setup Intent Wizard** is included with AdminStudio Standard, Professional, and Enterprise Editions.*

Automatic Launch of the SmartScan Wizard

If you are repackaging a legacy installation that was originally created using InstallShield Professional 5.5 or later, you can use the **SmartScan Wizard** to scan the setup for possible additional files, `.ini` files, shortcuts, or registry data (such as for language-specific setups or platform-specific setups). The **SmartScan Wizard** is able to group files, shortcuts and registry entries into features corresponding to InstallShield Professional components. Any items (files, folders, shortcuts, or registry entries) that are attached to a component directly or indirectly (through File Groups), are attached to their corresponding features.

For your convenience, if a Repackaging project is opened (either by using **Open** on the **File** menu or by using the Repackaging Wizard), and the project meets the following conditions, the SmartScan Wizard is automatically launched if:

- the original setup was an InstallShield Professional 5.5 or later installation, *and*
- the file has not already been repackaged using the InstallShield Professional Logging Method (available for InstallShield Editor and DevStudio 9.x InstallScript installations only), *and*
- the SmartScan Wizard has not already been run on this project.

Files and Folders View

From the **Files and Folders** view, you can examine information about each captured file, selectively exclude files or directories from the package you are creating, or reinclude files that you previously excluded.

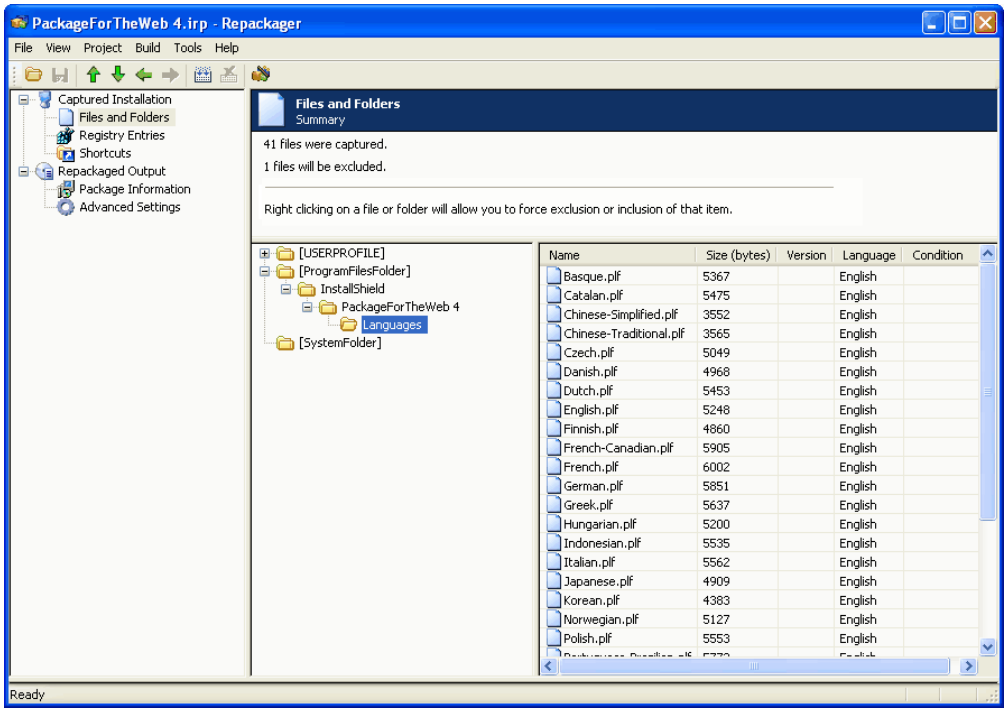


Figure 8-13: Repackager Files and Folders View

The upper pane displays the number of files captured and how many of these files will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see where files will be installed and the names of the files.

When you select a file from the tree, the lower-right pane displays attributes for that file. These attributes are:

Table 8-14 • File Attributes

Attribute	Description
Name	The file's name.
Size	The file's size in bytes.

Table 8-14 • File Attributes (cont.)

Attribute	Description
Version	The file's version.
Short Name	The short name for the file (if the file's author defined it).
Language	The file's language.

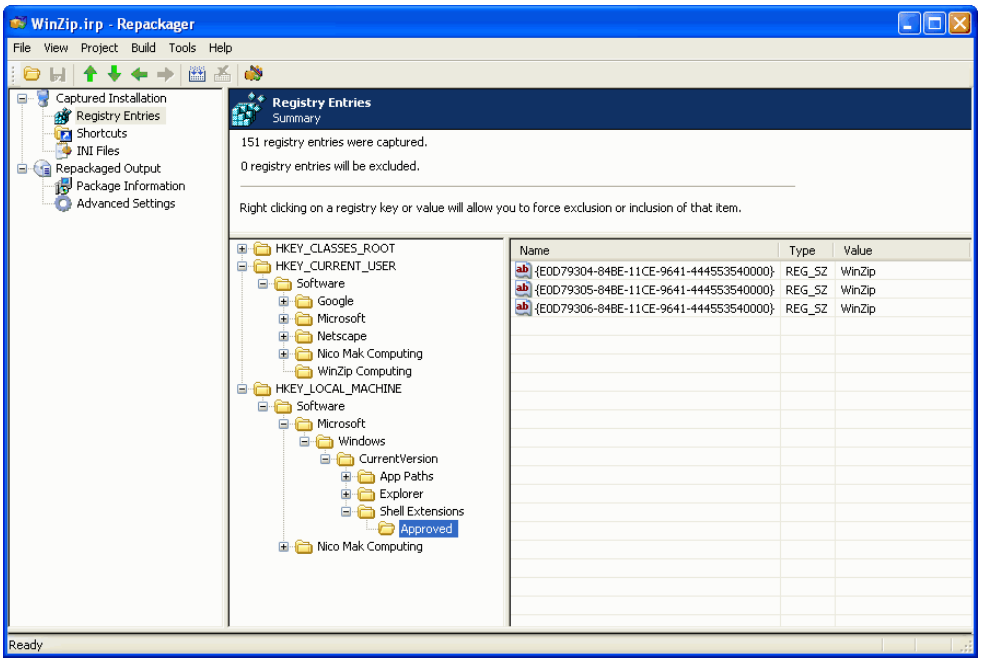
Excluding Files and Subdirectories

To specify which files and subdirectories you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, and **Include All** commands on the context menu:

- **To exclude a captured file**, select the file you want to exclude and select **Exclude**.
- **To exclude captured files within a directory**, select the directory containing the files you want to exclude and select either **Exclude** (to exclude only the files in the selected directory) or **Exclude All** (to exclude all of the files in the selected directory and all of its subdirectories).
- **To include a captured file that had previously been excluded**, select the file you want to include and select **Include**.
- **To include captured files within a directory that had previously been excluded**, select the directory containing the files you want to include and select either **Include** (to include only the files in the selected directory) or **Include All** (to include all of the files in the selected directory and all of its subdirectories).

Registry Entries View

From the **Registry Entries** view, you can examine information about each captured registry entry, selectively exclude registry values or registry keys from the package you are creating, or reinclude registry values that you previously excluded.



The upper pane displays the number of registry entries captured and how many of these entries will be excluded from the Windows Installer package when built. The lower-left pane provides a tree displaying the registry keys and subkeys captured. When you select a key from the tree, the lower-right pane displays any registry values for that key. Displayed information includes:

Table 8-15 • Registry Attributes

Attribute	Description
Name	The registry value name.
Type	The registry value type. This can be either a string value, an expandable string value, a multistring value, a dword value, or a binary value.
Value	The content of the registry value.

Excluding Registry Entries

To specify which registry entries you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, and **Include All** commands on the context menu:

- **To exclude a registry entry**, select the registry entry you want to exclude and select **Exclude**.
- **To exclude registry entries within a registry key or registry hive**, select the key or hive from the tree and select either **Exclude** (to exclude the registry entries in the selected hive or key only) or **Exclude All** (to exclude all of the registry entries in the selected hive or key and all of its keys and subkeys).
- **To include a registry entry that had previously been excluded**, select the registry entry and select **Include**.

- **To include registry entries within a registry key or registry hive that had previously been excluded**, select the key or hive from the tree and select either **Include** (to include the registry entries in the selected hive or key only) or **Include All** (to include all of the registry entries in the selected hive or key and all of its keys and subkeys).

Shortcuts View

From the Shortcuts view, you can examine information about each captured shortcut, selectively exclude shortcuts from the package you are creating, or reinclude shortcuts that you previously excluded.

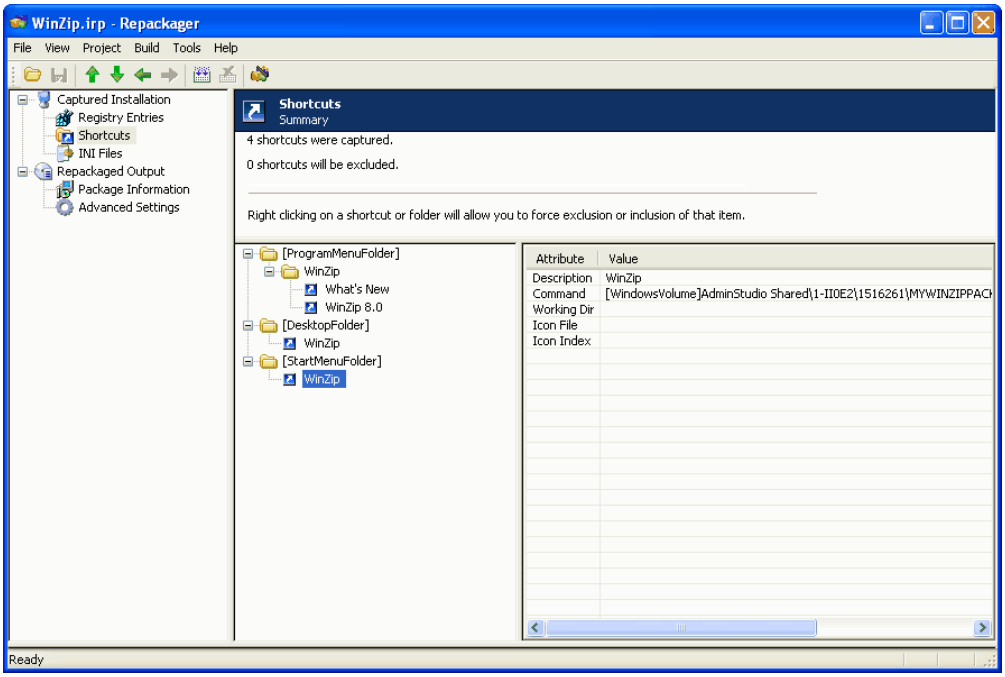


Figure 8-14: Repackager Shortcuts View

The upper pane displays the number of shortcuts captured and how many of these shortcuts will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see where shortcuts will be installed and the names of the shortcuts. When you select a shortcut from the tree, the lower-right pane displays attributes for that shortcut. These attributes are:

Table 8-16 • Shortcuts View Attributes

Attribute	Description
Description	The name of the shortcut as it appears on the desktop.
Command	The fully-qualified path and name of the file to which the shortcut points.

Table 8-16 • Shortcuts View Attributes

Attribute	Description
Working Dir	The shortcut's working directory, which may need to be specified so required files can load. This is equivalent of the Start in value found when right-clicking a shortcut from the desktop and selecting Properties.
Icon File	The name of the file containing the shortcut's icon.
Icon Index	The index number for the icon in the icon file.

Note • *Shortcuts can be excluded from the Windows Installer package you are building on an individual shortcut basis or by directory.*

Excluding Shortcuts

To specify which shortcuts you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, and **Include All** commands on the context menu:

- **To exclude a shortcut**, select the shortcut you want to exclude and select **Exclude**.
- **To exclude shortcuts within a directory**, select the directory containing the shortcuts you want to exclude and select either **Exclude** (to exclude only the shortcuts in the selected directory) or **Exclude All** (to exclude all of the shortcuts in the selected directory and all of its subdirectories).
- **To include a shortcut that had previously been excluded**, select the shortcut you want to include and select **Include**.
- **To include shortcuts within a directory that had previously been excluded**, select the directory containing the shortcuts you want to include and select either **Include** (to include only the shortcut in the selected directory) or **Include All** (to include all of the shortcuts in the selected directory and all of its subdirectories).

INI Files View

From the INI Files view, you can examine information about each captured `.ini` file, selectively exclude `.ini` files or `.ini` file sections from the package you are creating, or reinclude `.ini` files or sections that you previously excluded.

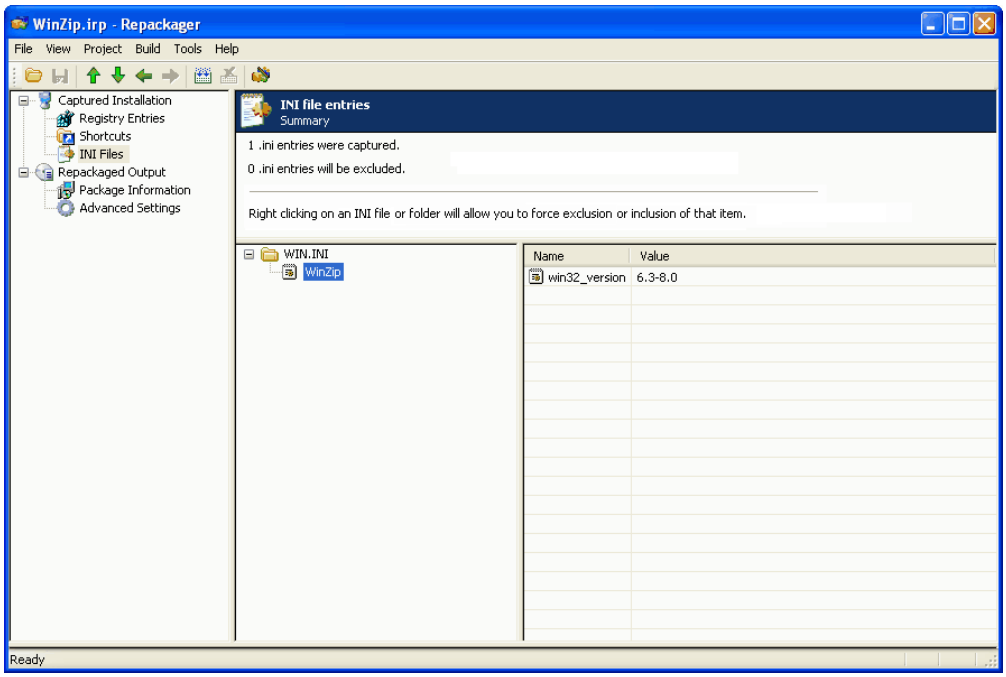


Figure 8-15: Repackager INI Files View

The upper pane displays the number of `.ini` files captured and how many of these `.ini` files will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see the full path to captured `.ini` files and sections contained within the `.ini` files. When you select a section from the tree, the lower-right pane displays name/value pairs in that section.

Excluding INI Files

To specify which INI files you want to include in the package, use the **Exclude**, **Exclude All**, **Include**, and **Include All** commands on the context menu:

- **To exclude an INI file**, select the INI file you want to exclude and select **Exclude**.
- **To exclude INI files within a directory**, select the directory containing the INI files you want to exclude and select either **Exclude** (to exclude only the INI files in the selected directory) or **Exclude All** (to exclude all of the INI files in the selected directory and all of its subdirectories).
- **To include an INI file that had previously been excluded**, select the INI file you want to include and select **Include**.
- **To include INI files within a directory that had previously been excluded**, select the directory containing the INI files you want to include and select either **Include** (to include only the INI file in the selected directory) or **Include All** (to include all of the INI files in the selected directory and all of its subdirectories).

Deleted Files View

From the Deleted Files view, you can examine information about each file deleted during repackaging, selectively exclude files or directories from the package you are creating, or reinclude previously excluded files.

The upper pane displays the number of files captured and how many of these files will be excluded from the Windows Installer package when built. The lower-left pane provides a tree from which you can see where files will be installed and the names of the files. When you select a file from the tree, the lower-right pane displays attributes for that file. These attributes are:

Table 8-17 • Deleted Files View Attributes

Attribute	Description
Name	The file's name.
Size	The file's size in bytes.
Version	The file's version.
Short Name	The short name for the file (if the file's author defined it).
Language	The file's language.

Excluding Files and Subdirectories

To specify which files and subdirectories you want to include in the package, use the Exclude, Exclude All, Include, and Include All buttons:

- To exclude a captured file from the package, select the file you want to exclude and click Exclude.
- To exclude all captured files and subdirectories within a directory from the package, select the directory containing the files and subdirectories you want to exclude and click Exclude All.
- To include a captured file in the package that had previously been excluded, select the file you want to include and click Include.
- To include all captured files and subdirectories within a directory, select the directory containing the files and subdirectories you want to include and click Include All.

Deleted Registry Entries View

From the Deleted Registry Entries view, you can examine information deleted from the registry repackaging, selectively exclude registry keys from the package you are creating, or reinclude previously excluded data.

The upper pane displays the number of deleted registry entries captured and how many of these entries will be excluded from the Windows Installer package when built. The lower-left pane provides a tree displaying the registry keys and subkeys captured. When you select a key from the tree, the lower-right pane displays any registry values for that key. Displayed information includes:

Table 8-18 • Deleted Registry Entries View Attributes

Attribute	Description
Name	The registry value name.

Table 8-18 • Deleted Registry Entries View Attributes

Attribute	Description
Type	The registry value type. This can be either a string value, an expandable string value, a multistring value, a dword value, or a binary value.
Value	The content of the registry value.

Excluding Registry Entries

To specify which registry entries you want to include in the package, use the Exclude, Exclude All, Include, and Include All buttons:

- To exclude a registry entry from the package, select the registry entry you want to exclude and click Exclude.
- To exclude all registry entries and subdirectories within a directory from the package, select the directory containing the registry entries you want to exclude and click Exclude All.
- To include a registry entry in the package that had previously been excluded, select the registry entry you want to include and click Include.
- To include all registry entries and subdirectories within a directory, select the directory containing the shortcuts and subdirectories you want to include and click Include All.

Repackaged Output View

From this view, you can configure build options for the project, including whether to build an MSI package automatically following conversion.

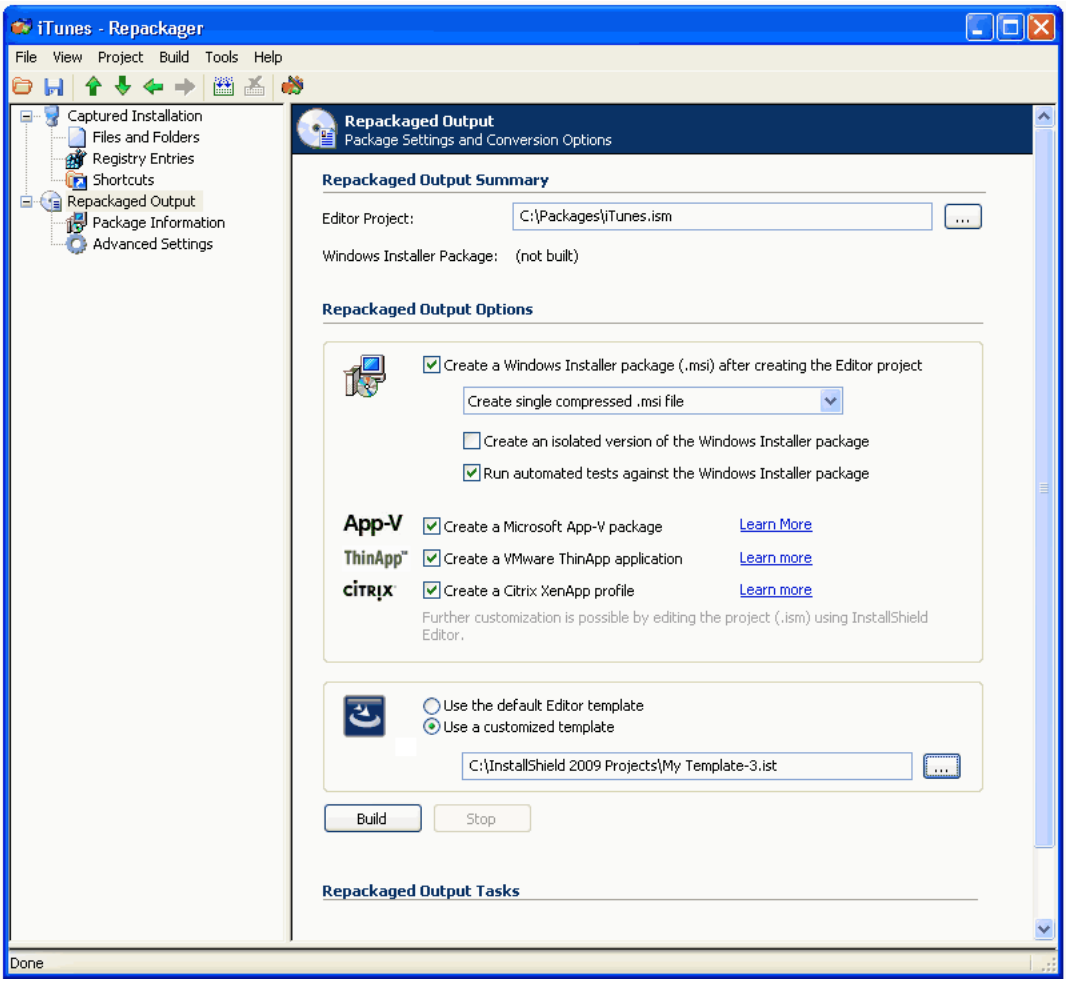


Figure 8-16: Repackager Repackaged Output View

The following properties are available for configuration:

Table 8-19 • Repackaged Output View Options

Option	Description
Editor project	Provide the name and location of the InstallShield Editor project (`.ism`) file.
Windows Installer package	The name and location of the Windows Installer package (`.msi`). If a Windows Installer package has not yet been built from this Repackager project, `(not built)` is listed.
Create a Windows Installer package (.msi) after creating the Editor project	If this option is selected, after creating the InstallShield Editor project file (`.ism`), a Windows Installer (.msi) file will also be built.

Table 8-19 • Repackaged Output View Options

Option	Description
Windows Installer Package Options	If you have selected the **Create a Windows Installer package after creating the Editor project** option, you need to also select one of the following options: • **Create single, compressed .msi file**—Select this option if you want to compress all necessary files inside the .msi package, as opposed to storing them outside of the .msi database. • **Create single, compressed setup.exe file**—Select this option if you want to compress all files inside a setup.exe file, including the .msi file and all other necessary files. • **Create .msi file + external compressed .cab file**—Select this option if you want to create an .msi file and want to compress the rest of the necessary files in an external .cab file. • **Create .msi file + external compressed .cab file + setup.exe**—Select this option if you want to create an .msi file and a setup.exe file, and want to compress all the rest of the necessary files in an external .cab file. • **Create uncompressed .msi file**—Select this option if you want to create an uncompressed .msi file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi file. • **Create uncompressed .msi file + setup.exe**—Select this option if you want to create an uncompressed .msi file along with a setup.exe file. All of the rest of the necessary files, in uncompressed format, would be shipped with the .msi and setup.exe files.

Table 8-19 • Repackaged Output View Options

Option	Description
Create an isolated version of the Windows Installer package	Select this option to create a second, isolated version of the Windows Installer package when the Windows Installer package is built. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested. If this option is selected, an additional Windows Installer package will be created in the same directory as the `.ism` file and the other `.msi` file, with the naming convention of: *appname*`.isolated.msi` For more information on how Repackager isolates applications and the available isolation options, see Isolating Windows Installer Packages. ***Note*** • *This option is only enabled when the **Create a Windows Installer package (.msi) after creating the Editor project** option is selected and one of the following values is chosen:* • *Create single compressed .msi file* • *Create .msi file + external compressed .cab file* • *Create uncompressed .msi file*
Run automated tests against the Windows Installer package	Select this option to automatically run PackageExpert tests against the newly built Windows Installer package to determine if it is built according to Windows Installer standards, and if it is in compliance with the installation requirements of the Windows Vista operating system. All of the tests that are currently selected on the PackageExpert **Configuration View** are run, and those selected tests that are automatically resolvable, will be resolved. ***Note*** • *This option is only enabled when the **Create a Windows Installer package (.msi) after creating the Editor project** option is selected and any of the values except for **Create single compressed setup.exe file** is chosen.*
Create a Microsoft App-V application	If this option is selected, after building a Windows Installer (.msi) file, a Microsoft App-V application will also be built. ***Note*** • *This option requires that you build a Windows Installer package.*

Table 8-19 • Repackaged Output View Options

Option	Description
Create a VMware ThinApp application	If this option is selected, after building a Windows Installer (.msi) file, a VMware ThinApp application will also be built. ***Note*** • *This option requires that you build a Windows Installer package.*
Create a Citrix XenApp profile	If this option is selected, after building a Windows Installer (.msi) file, a Citrix profile compatible with Citrix XenApp will also be built. ***Note*** • *This option requires that you build a Windows Installer package.*
Use the default Editor template	When building an InstallShield Editor project, select this option to use the default InstallShield Editor template. A project template contains all of the default settings and design elements that you want to use as a starting point when you create an installation project.
Use a customized template	When building an InstallShield Editor project, select this option to specify a customized InstallShield Editor Project Template to use. For example, if you wanted all of your InstallShield Editor projects to have a special custom dialog, a set of required redistributables, and a particular SQL script, you could create a project template that has all of those settings. Then, any time that you wanted to create a new project, you could base it off of your custom template. This enables you to avoid re-creating the custom dialog, re-adding the redistributables, and re-adding the SQL script every time that you create a new InstallShield Editor Project.
Build	Click to initiate the build process to build a Windows Installer package.
Repackaged Output Tasks	After an InstallShield Editor project and a Windows Installer package has been built, you can use these links to perform the following tasks: • **Modify the Editor Project**—Open this Repackager project's associated InstallShield Editor project in InstallShield Editor. • **Modify the Windows Installer package with Editor**—Open this Repackager project's associated Windows Installer package in InstallShield Editor.

Once you have built the Windows Installer package and/or InstallShield Editor file, you can launch InstallShield Editor from the **Repackaged Output** area of the view.

Package Information View

The Package Information view allows you to specify information for the Windows Installer package that you build from the Repackager project. Much of this information may be prepopulated based on settings used in the Repackaging Wizard.

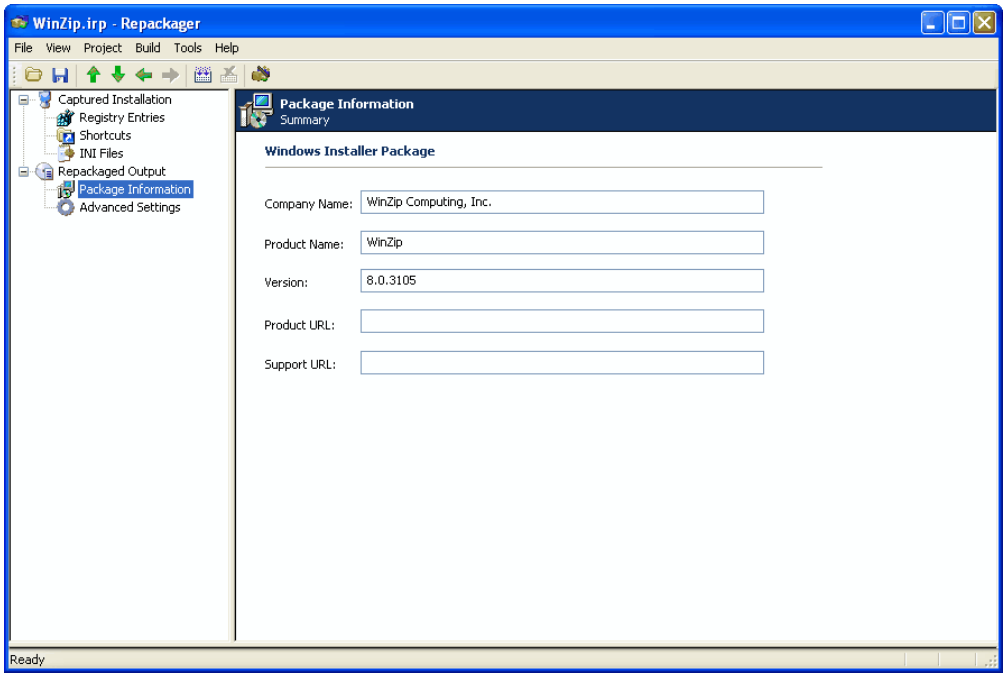


Figure 8-17: Repackager Package Information View

You can configure the following options:

Table 8-20 • Package Information View Options

Option	Description
Company Name	The name of the company that developed the product you are repackaging.
Product Name	The name of the product you are repackaging.
Version	The product's version number.
Product URL	The URL for product information. This appears in Add/Remove Programs in the Control Panel.
Support URL	A URL for support information. This also appears in Add/Remove Programs in the Control Panel, and is often changed during repackaging to provide an internal support URL.

Advanced Settings View

From the Advanced Settings view, you can configure several additional settings that may apply to your repackaged setup.

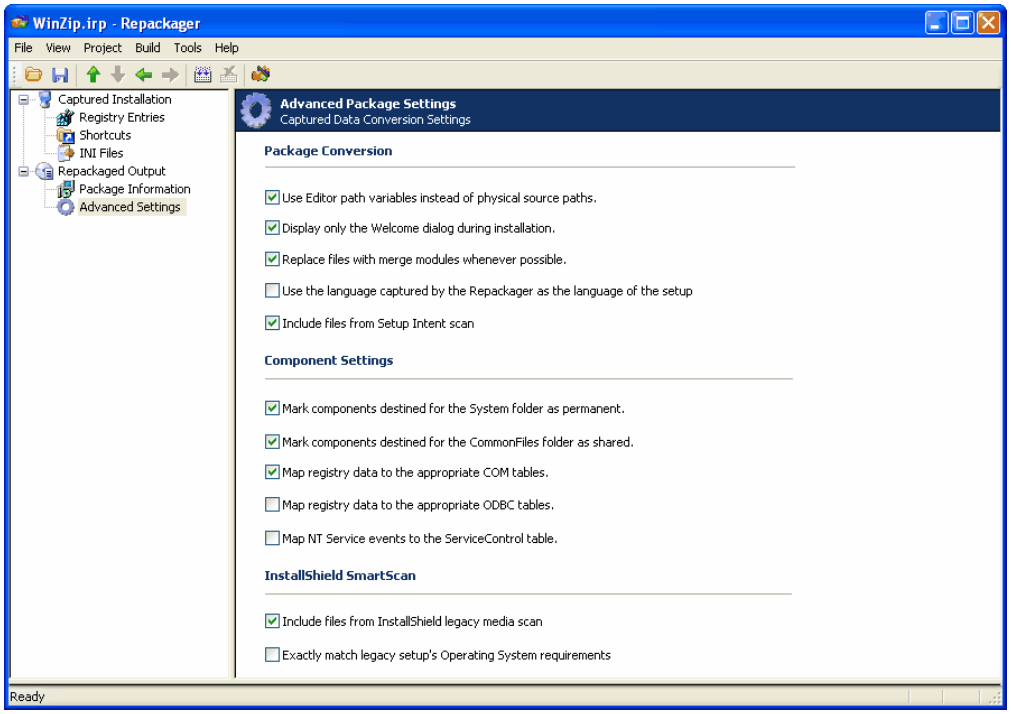


Figure 8-18: Repackager Advanced Settings View

Select the appropriate options:

Package Conversion Options

The following package conversion options are available:

Table 8-21 • Package Conversion Options

Option	Description
Use Editor path variables instead of physical source paths	When storing files in the InstallShield Editor project (`.ism`), the Wizard uses path variable locations whenever possible.
Display only the Welcome dialog box during installation	Only the Welcome dialog box is displayed when the Windows Installer package is run on a target machine. If this option is unchecked, the default UI sequence is displayed when the setup is installed.
Replace files with merge modules wherever possible	Following best practice rules, Repackager replaces components with comparable merge modules whenever possible.

Table 8-21 • Package Conversion Options (cont.)

Option	Description
Use the language captured by the Repackager as the language of the setup	When selected, the target package's language will be the language detected by Repackager (as displayed in the Captured Installation view).
Include files from Setup Intent scan	Any files identified when running the Setup Intent Wizard will be included in the package (unless you have manually excluded them from the project).

Component Settings Options

The following component settings options are available:

Table 8-22 • Component Settings Options

Option	Description
Mark components destined for the System folder as permanent	Executable files installed to the system folder (System32Folder) are marked as Permanent files and will not be uninstalled when the package is uninstalled. This eliminates ICE09 validation errors.
Mark components destined for the CommonFiles folder as shared	Executable files installed to the CommonFilesFolder (or a subfolder of CommonFilesFolder) are marked as shared files. This ensures that these components can coexist with DLLs installed by previous setups.
Map registry data to the appropriate COM tables	Setting this option reduces the number of ICE33 warnings that can occur during package validation, resulting from data not being mapped to the appropriate MSI tables.
Map registry data to the appropriate ODBC tables	If selected, ODBC-related registry data is mapped to ODBC tables instead of the Registry table. This data will only function correctly if Windows Installer supports the ODBC resource being mapped; it is highly recommended that you do not enable this option if you are unsure whether the ODBC resources are supported correctly by Windows Installer.
Map NT Service events to the ServiceControl table	If selected, NT Service–related registry data is mapped to ServiceControl table instead of the Registry table.

InstallShield SmartScan Options

The following InstallShield SmartScan options are available:

Table 8-23 • InstallShield SmartScan Options

Option	Description
Include files from InstallShield legacy media scan	Files identified in the Media Scan Wizard will be included in the package (unless you have manually excluded them from the project).
Exactly match legacy setup's Operating System requirements	For SmartScan projects or projects using the Professional Logging Method: • **Selected**—if this option is selected, component conditions will store specific operating system information: if the filter was NT4, the condition will be (VersionNT=4). • **Unselected**—If this option is not selected, component conditions will store a grouping of the operating system: if the filter was NT4, the condition will be (VersionNT). • **Default**—By default, this option is not selected.

SmartScan Wizard

The SmartScan Wizard is designed to scan original, InstallShield Professional 5.x and later media for setup information that may not have been captured during repackaging, such as possible additional files, `.ini` files, shortcuts, or registry data, which is often included in language-specific or platform-specific setups. This primarily involves OS-dependent and language-dependent files that may be included in header files. Ultimately, this makes the installation more portable.

Tip • *Any files found will be displayed in Repackager in a different color (as specified in the Color tab of the Options dialog box).*

The SmartScan Wizard is also able to group files, shortcuts and registry entries into features corresponding to InstallShield Professional components. Any items (files, folders, shortcuts, or registry entries) that are attached to a component directly or indirectly (through File Groups), are attached to their corresponding features.

Automatic Launch of the SmartScan Wizard

For your convenience, if a Repackaging project is opened (either by using Open on the File menu or by using the Repackaging Wizard), and the project meets certain conditions, the SmartScan Wizard is automatically launched. The SmartScan Wizard is launched:

- if the original setup was an InstallShield Professional 5.5 or later installation, *and*
- the file has not already been repackaged using the InstallShield Professional Logging Method (available for InstallShield Editor and DevStudio 9.x InstallScript installations only), *and*

- the SmartScan Wizard has not already been run on this project

SmartScan Wizard Panels

The SmartScan Wizard includes the following panels:

- Welcome to the SmartScan Wizard Panel
- Original InstallShield Professional Setup Panel
- Scanning Panel
- Setup Feature Tree Panel
- Scanning Media Panel

Welcome to the SmartScan Wizard Panel

The SmartScan Wizard is designed to scan original, InstallShield Professional 5.x and later media for setup information that may not have been captured during repackaging, such as possible additional files, `.ini` files, shortcuts, or registry data (such as for language-specific setups or platform-specific setups). This primarily involves OS-dependent and language-dependent files that may be included in header files. Ultimately, this makes the installation more portable.

The SmartScan Wizard is also able to group files, shortcuts and registry entries into features corresponding to InstallShield Professional components. Any items (files, folders, shortcuts, or registry entries) that are attached to a component directly or indirectly (through File Groups), are attached to their corresponding features.

The first panel in the SmartScan Wizard is the **Welcome** panel.

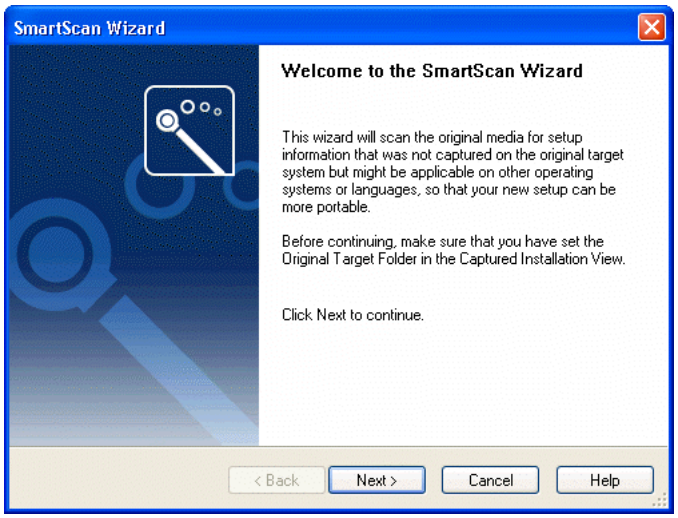


Figure 8-19: SmartScan Wizard Welcome Panel

When you click **Next**, you advance to the **Original InstallShield Professional Setup Panel**, where you specify the InstallShield Professional setup you want to scan.

Note • *For your convenience, if a Repackaging project is opened (either by using Open on the File menu or by using the Repackaging Wizard), and the project meets certain conditions, the SmartScan Wizard is automatically launched. The SmartScan Wizard is launched:*

- *if the original setup was an InstallShield Professional 5.5 or later installation, AND*
- *the file has not already been repackaged using the InstallShield Professional Logging Method (available for InstallShield Editor and DevStudio 9.x InstallScript installations only), AND*
- *the SmartScan Wizard has not already been run on this project*

Original InstallShield Professional Setup Panel

On this panel, you specify the InstallShield Professional setup executable that you want to scan, and the original target directory of that executable.

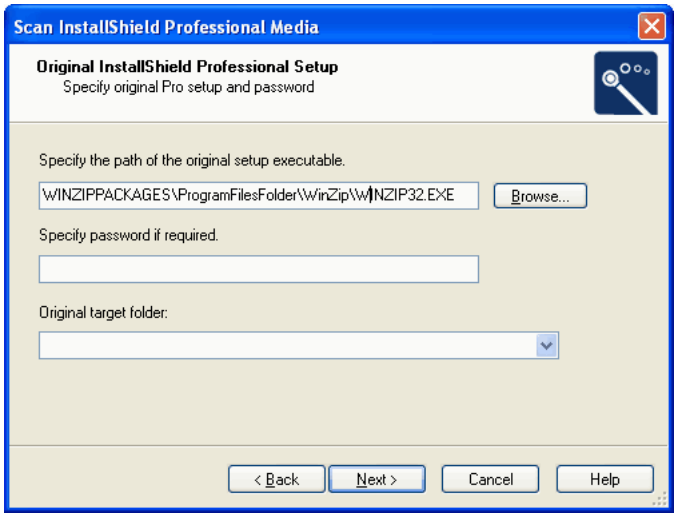


Figure 8-20: SmartScan Wizard Original InstallShield Professional Setup Panel

Enter the following options:

Table 8-24 • Original InstallShield Professional Setup Panel Options

Option	Description
Specify the path of the original setup executable	This field is filled in by default if that information exists. Click Browse to specify the InstallShield Professional setup executable that you want to scan. If you enter an invalid path name, the Next button will be disabled.
Specify password if required	Specify a password of the path of the original setup executable, if required.

Table 8-24 • Original InstallShield Professional Setup Panel Options (cont.)

Option	Description
Original target folder	By default, this field is populated with information found in the original setup package. Enter the directory where the product was installed during the repackaging process. In most cases, this will be a subdirectory of `[ProgramFilesFolder]`. Alternatively, you can enter any target where the product was installed during the repackaging process. For example, if you installed this product to `C:\Program Files\MyCompany\MyProduct Folder`, you would enter the following in this field: `[ProgramFilesFolder]MyCompany\MyProduct`

Click Next to proceed to the **Scanning Panel**, which shows the output and progress of the scan.

Scanning Panel

This panel displays the progress that SmartScan Wizard is making as it scans an InstallShield Professional setup.

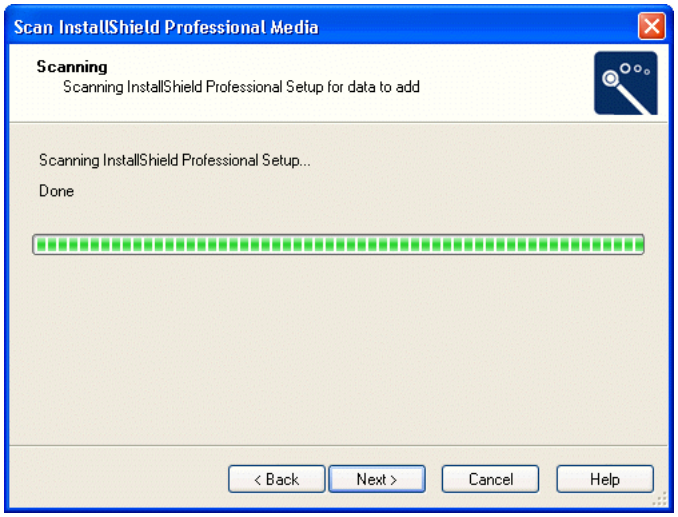


Figure 8-21: SmartScan Wizard Scanning Panel

Setup Feature Tree Panel

The Setup Feature Tree Panel lists the features and components that were configured in the original InstallShield Professional project. If you want to use this feature and component information, select the **Add feature tree** option.

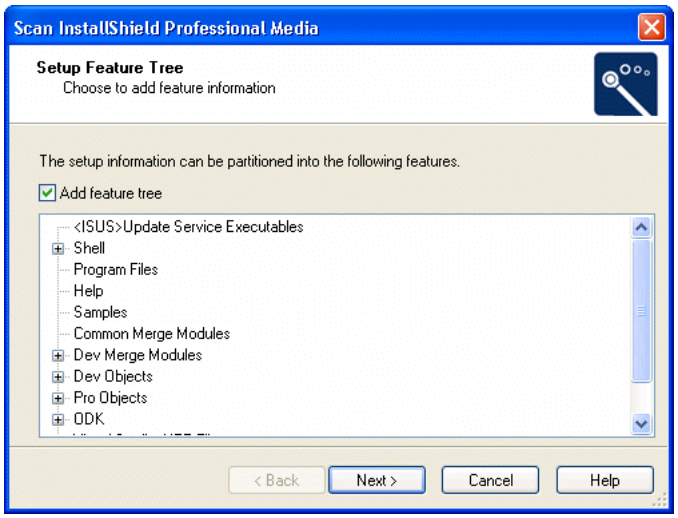


Figure 8-22: SmartScan Setup Feature Tree Panel

Click **Next** to proceed to the **Scanning Media Panel**, which first scans the original media, then displays the results.

Scanning Media Panel

Following the actual scan of the original setup media, the SmartScan Wizard displays a log of files to be included in the project.

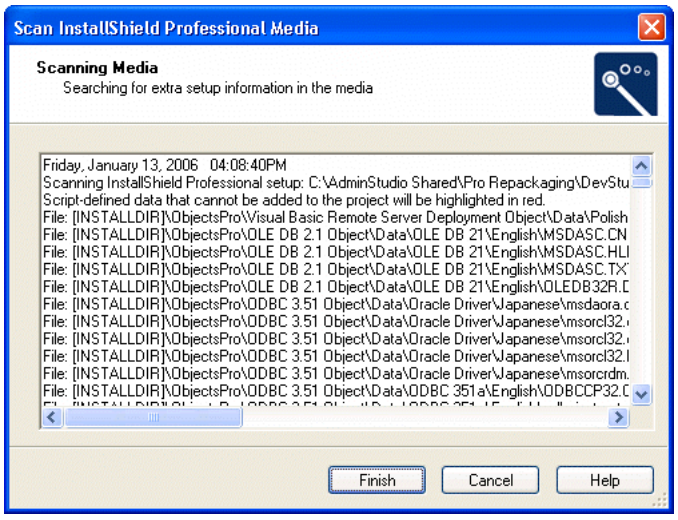


Figure 8-23: SmartScan Wizard Scanning Media Panel

Click **Finish** to add these to the project. Any default exclusions based on the exclusions file will be applied.

Setup Intent Wizard

Although an installation may have intended to install certain files, these files sometimes may not be installed—often because the files already exist on the target machine (either as the same version or a newer version). These files, although not installed or updated, are needed for the product to execute properly when the setup is run on a system that does not already have these files.

The Setup Intent Wizard allows you to scan a setup to identify files that may not have been captured during repackaging—effectively recognizing the installation's intent for these files.

Tip • *Any files found will be displayed in Repackager in a different color (as specified in the **Color** tab of the Options dialog box).*

The Setup Intent Wizard consists of the following panels:

- Welcome Panel
- Scanning Project Panel
- Results Panel

Welcome Panel

The first panel in the Setup Intent Wizard informs you the purpose of the Wizard, and warns you the source files for your project must be present for successful scanning.

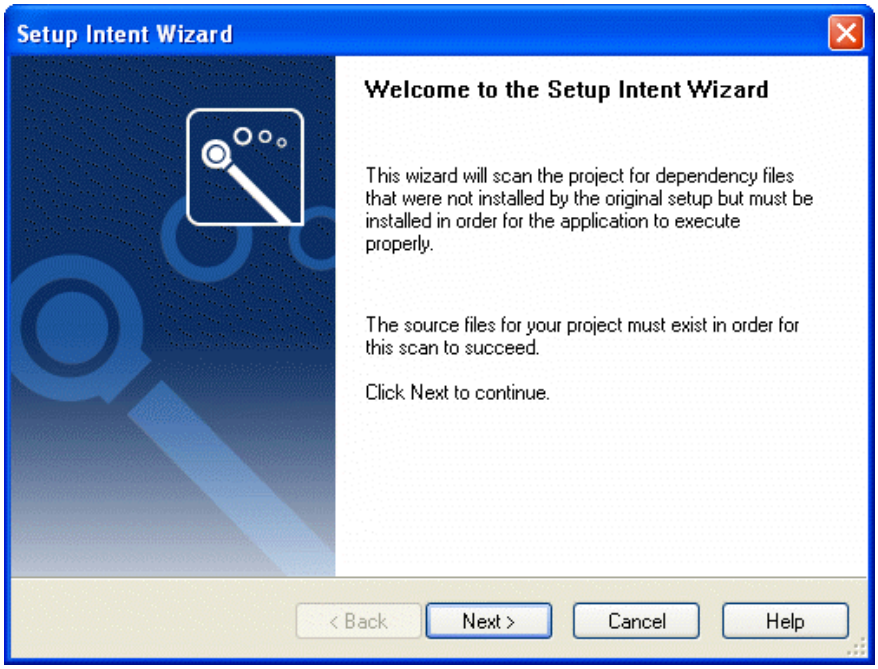


Figure 8-24: Setup Intent Wizard Welcome Panel

Click Next to start the scan and display the **Scanning Project Panel**.

Scanning Project Panel

The Scanning Project Panel is displayed while scanning is in progress. Each file scanned is listed, and a progress bar displays the overall scan progress.

When the scan is complete, the **Results Panel** opens, listing new files that your setup required.

Results Panel

The final panel in the Setup Intent Wizard allows you to view and select new files detected by the Wizard, but not already included in your Repackaging project.

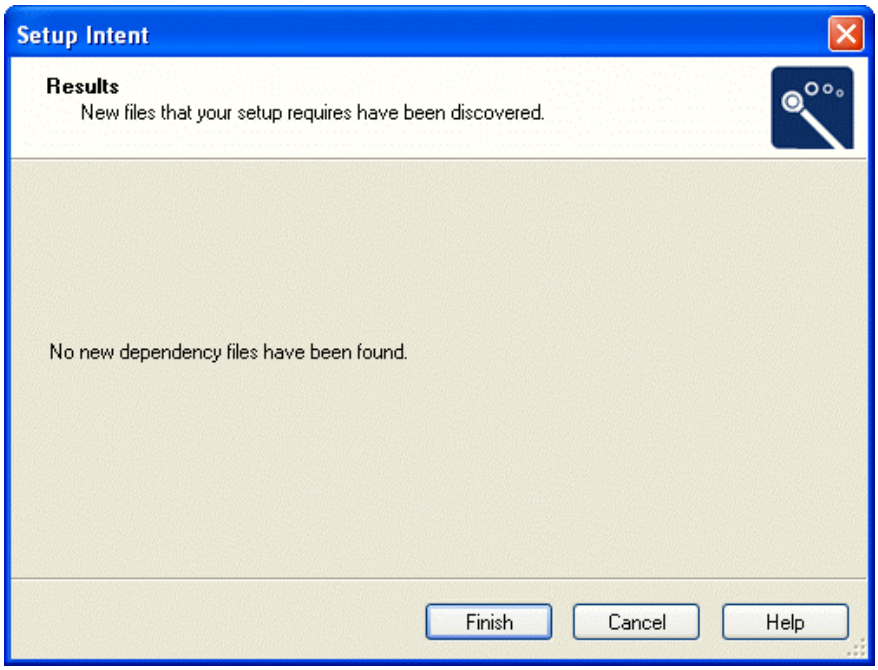


Figure 8-25: Setup Intent Wizard Results Panel

Select the files you want to include in your project which were not identified during repackaging.

Click **Finish** to exit the Setup Intent Wizard and return to the Repackaging project (with selected files automatically added to the project), or click **Back** to return to the **Scanning Project Panel**.

VMware Repackaging Wizard

Repackager includes integration with VMware Workstation's virtual machine technology. This provides you with the ability to launch a VMware session for repackaging purposes, and run different operating systems on the same computer. By using VMware, you are able to forego the traditional "ghosting" for clean images each time a new application is repackaged by simply electing not to save changes to the VMware session. You can then reload the clean state of the operating system, and proceed to the next package.

Note • *AdminStudio supports VMware 3.0 and later.*

Note • *The **VMware Repackaging Wizard** menu item on the **Tools** menu is enabled if Repackager finds VMware 3.0 and later installed on the workstation, and if a VMware image exists on that machine. If no VMware images are found, the VMware Repackaging Wizard menu item will be disabled. Repackager reads the information about VMware images from:*

`<Application Folder>\VMware\Virtual Machines.vmls`

`Virtual Machines.vmls` *is a text file that contains information about individual VMware images and where the configuration file for each image is located. This file should contain information for at least one VMware image for the **VMware Repackaging Wizard** menu item to be enabled.*

Using the VMware Repackaging Wizard, you select an available VMware operating system, and then Repackager automatically launches the selected operating system within a VMware session.

The VMware Repackaging Wizard includes two panels:

- Welcome Panel
- VMware Virtual Machines Panel

Welcome Panel

The first panel displayed in the VMware Repackaging Wizard is the Welcome panel. It explains the purpose of this Wizard: to display available VMware images on the current workstation, allowing you to select and launch the one you need.

VMware Virtual Machines Panel

On the VMware Virtual Machines panel, you select a VMware virtual machine available on the current workstation. Repackager automatically launches the selected virtual machine operating system within a VMware session so that you can begin repackaging in that environment.

Note • *AdminStudio supports VMware 3.0 and later.*

Click Back to return to the **Welcome Panel**; click Launch to launch the selected VMware image.

Exclusions Editor Interface

The following topics cover each tab, menu, and dialog box in the Exclusions Editor:

- Menus
- Files Tab
- .ini Files Tab
- Registry Tab
- File Exclusion Information Dialog Box
- INI File Exclusion Information Dialog Box
- Choose Registry Key Dialog Box
- Edit Registry Key Dialog Box
- About Exclusions Editor Dialog Box

Menus

Menus are not available when running the Exclusions Editor from within Repackager. They are only available when you launch the Exclusions Editor by opening the following file:

```
AdminStudio Installation Directory\Repackager\AnalysisOptions.exe
```

Note • *See Launching Exclusions Editor for more information.*

The following table provides a description of each menu command:

Table 8-25 • Exclusions Editor Menu Commands

Menu	Command	Keyboard Shortcut	Description
File	**New**	Ctrl+N	Creates a new, blank settings file.
File	**Open | Shared Exclusions**		Opens the settings file (`isrepackager.ini`) from the `AdminStudio Shared` directory. Open this settings file when working in a team environment where the exclusion list needs to be stored in a centralized location.
File	**Open | Custom Exclusions**		Allows you to browse to an Exclusions Editor settings file and open it. You would create a custom exclusion file based upon your company's requirements.

Table 8-25 • Exclusions Editor Menu Commands

Menu	Command	Keyboard Shortcut	Description
File	**Save**	Ctrl+S	Saves the current Exclusions Editor settings file.
File	**Save As**		Saves the current Exclusions Editor settings file to the name and location specified.
File	**Exit**		Exits the Exclusions Editor.
Help	**Help Library**		Displays the online Help Library.
Help	**About Exclusions Editor**		Displays the About Exclusions Editor dialog box.

Files Tab

File exclusions for Repackager indicate which files are automatically marked as excluded in the Repackager project. File exclusions in the OS Snapshot Wizard indicate files that will be excluded from the captured OS snapshot.

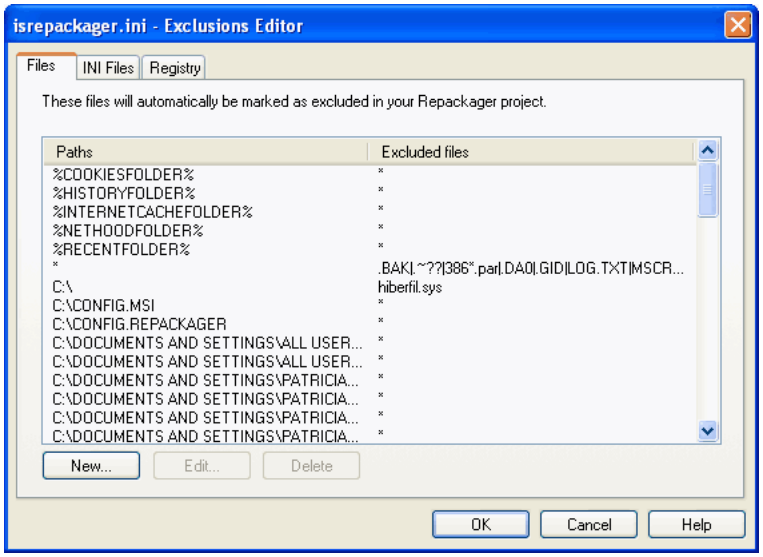


Figure 8-26: Exclusions Editor Files Tab

This **Files** tab contains a list of paths and files currently excluded from the capture process. Specific files, file extensions, and the entire contents of specified directories can be excluded.

The following three buttons allow you to add, edit, and remove files and directories from the exclusion list:

Table 8-26 • Exclusions Editor / Files Tab Buttons

Button	Description
New	Displays the **File Exclusion Information** dialog box from which you can specify additional file exclusions.
Edit	Brings up a dialog box from which you can change settings for the currently selected path in the exclusion list.
Delete	Deletes the currently selected path from the exclusion list.

Note • *It is highly recommended that you do not edit the default exclusions for the OS Snapshot Wizard.*

.ini Files Tab

.ini file exclusions for Repackager indicate which .ini files and sections are automatically marked as excluded in the Repackager project. `.ini` file exclusions in the OS Snapshot Wizard indicate `.ini` files and sections that will be excluded from the captured OS snapshot.

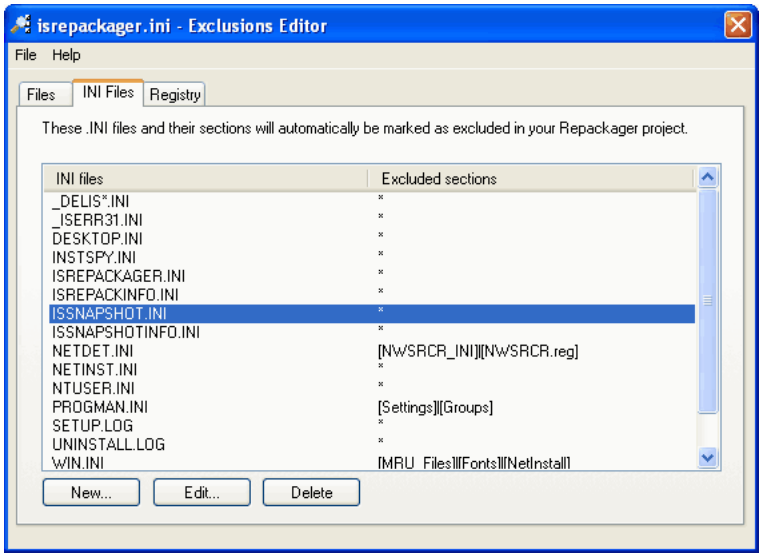


Figure 8-27: Exclusions Editor INI Files Tab

The **INI Files** tab contains a list of the `.ini` files and sections within `.ini` files excluded during analysis. If all sections are excluded, an asterisk (`*`) is used in the Excluded Sections column.

The following three buttons allow you to add, edit, and remove `.ini` files from the exclusion list:

Table 8-27 • Exclusions Editor / .ini Files Tab Buttons

Button	Description
New	Displays the **INI File Exclusion Information** dialog box from which you can specify additional .ini file exclusions.
Edit	Brings up a dialog box from which you can edit currently excluded .ini files.
Delete	Deletes the selected .ini file from the exclusion list.

Note • *It is highly recommended that you do not edit the default exclusions for the OS Snapshot Wizard.*

Registry Tab

Registry exclusions for Repackager indicate which registry keys are automatically marked as excluded in the Repackager project. Registry exclusions in the OS Snapshot Wizard indicate registry keys that will be excluded from the captured OS snapshot.

The **Registry** tab contains a list of keys and values to be excluded during registry analysis. For keys that have specific values excluded, the value name appears in the Value column. For keys that have all values excluded, an asterisk (*) represents the entire key in the Value column.

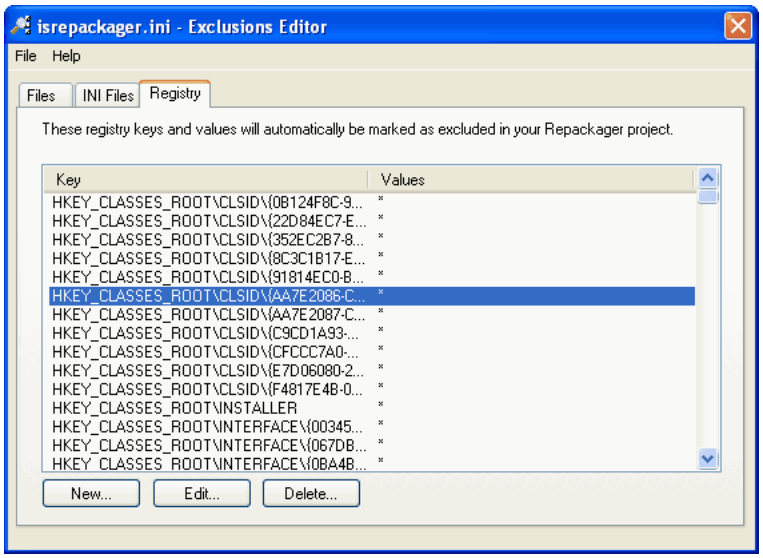


Figure 8-28: Exclusions Editor Registry Tab

There are three buttons available from this dialog box that are used to add, edit, or remove keys from the exclusion list:

Table 8-28 • Exclusions Editor / Registry Tab Buttons

Button	Description
New	Displays the **Choose Registry Key** dialog box, from which you can select registry keys and values for exclusion during analysis.
Edit	Brings up a dialog box from which you can modify the selected key's exclusion settings.
Delete	Removes the selected key from the exclusion list.

Note • *It is highly recommended that you do not edit the default exclusions for the OS Snapshot Wizard.*

File Exclusion Information Dialog Box

The **File Exclusion Information** dialog box, which is accessed by clicking **New** or **Edit** on the Files Tab, allows you to specify files to be excluded from analysis by the capture tool.

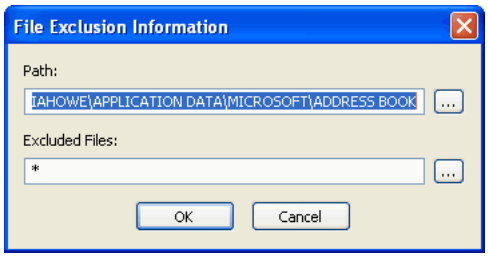


Figure 8-29: File Exclusion Information Dialog Box

Click the Browse button next to the **Path** field and select the directory that contains the file or files you want to exclude. Then, click the Browse button next to the **Excluded Files** field and select the file or files you want to exclude. In the **Excluded Files** field, you can specify files to exclude in the following ways:

- To exclude multiple files from the same directory, separate the file names with pipes (|), such as:

  `file.dll|myfile.exe|anotherfile.exe`

- To exclude all files with a certain extension in the selected directory, enter an asterisk (*) plus the extension, such as:

  `*.txt`

- To exclude all files in the selected directory, enter an asterisk (*) .

Click **OK** to return to the Files Tab.

INI File Exclusion Information Dialog Box

The INI File Exclusion Information dialog box, which is accessed by clicking **New** or **Edit** on the .ini Files Tab, allows you to specify `.ini` files to be excluded from analysis by the capture tool.

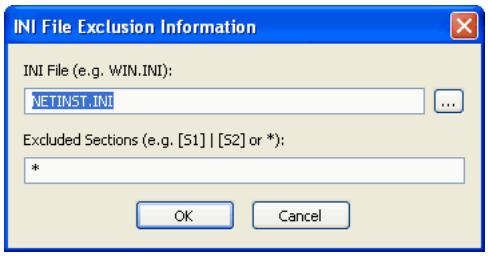


Figure 8-30: INI File Exclusion Information Dialog Box

Enter or browse to the `.ini` file you want to exclude, and provide the section(s) to be excluded. Sections must be enclosed in square brackets ([]), and separated by vertical bars (|) if more than one section in an `.ini` file is to be excluded (for example, [Groups],[Settings]). You can also exclude all `.ini` file sections by only entering an asterisk in the Excluded Sections field.

Click **OK** to return to the .ini Files Tab.

Choose Registry Key Dialog Box

The Choose Registry Key dialog box, which is accessed by clicking **New** on the Registry Tab, provides a way for you to select registry keys that you want excluded from analysis.

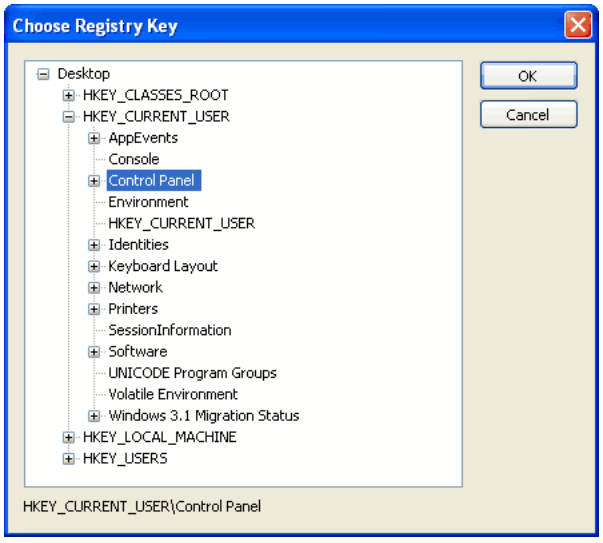


Figure 8-31: Choose Registry Key Dialog Box

Navigate through the tree until you find the key you want to exclude and click **OK** to return to the Registry Tab.

By default, all values in that key are excluded. To modify this, select the key from the Registry Tab and click **Edit** to display the **Edit Registry Key** dialog box.

Tip • *You can also select a registry hive to exclude. As with individual registry keys, all values (and keys) contained in the hive are excluded by default.*

Edit Registry Key Dialog Box

When you select a registry key on the Registry Tab and click **Edit**, the **Edit Registry Key** dialog box opens.

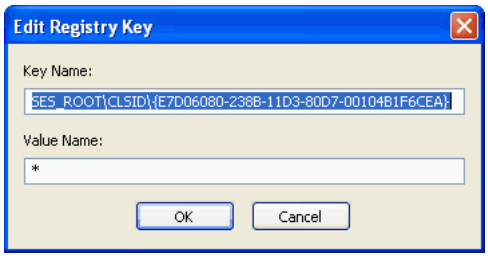


Figure 8-32: Edit Registry Key Dialog Box

You can modify the **Key Name** and/or **Value Name** excluded during analysis.

Click OK to return to the Registry Tab.

About Exclusions Editor Dialog Box

The About Exclusions Editor dialog box displays version and copyright information for the Exclusions Editor. This may be useful if you need to report a problem encountered when using the Exclusions Editor.

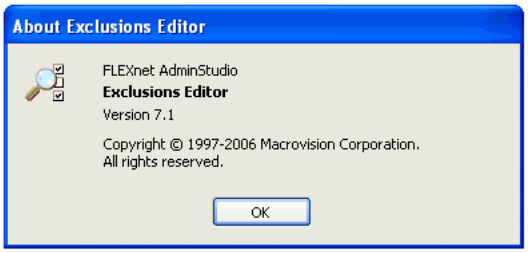


Figure 8-33: About Exclusions Editor Dialog Box

Options.ini File

The `Options.ini` file is created by Repackager and is used during the conversion of Repackager output into an InstallShield Editor project (`.ism`). It includes basic project settings which are required by Repackager. Information about this file is presented in the following sections:

- [MMExclusions] Section
- [General] Section
- [IgnoreShortcuts] Section
- Options.ini File Defaults

Note • *Although many of these settings have a one-to-one correspondence with settings available in the Repackager interface, some can only be accessed by editing this .ini file directly.*

[MMExclusions] Section

This section lists the merge module GUIDs that should not be included in your package. This section only applies if you have selected to replace files with merge modules during conversion.

[General] Section

Following are descriptions of properties that can be set in the [General] section of the Options.ini file.

Table 8-29 • Options.ini File/General Section Properties

Properties	Description
AddIMMSearchPath	Use to specify additional directories containing custom merge modules to use during repackaging.
ALLUSERS	If this option is set to **Y** and if the template file (specified using the `ProjectTemplate` option) does not contain **ALLUSERS** in its Property table, then a property named **ALLUSERS** with a value of 2 will be added to the Property table. This will cause silent installs to behave as non-silent installs do (non-silent installs rely on a custom action to set this property). This option is set to **Y** by default.
ARPPublisher	This populates the **Publisher** field in **Add/Remove Programs** in the Control Panel.
ARPPublisherURL	This populates the **Publisher URL** field in **Add/Remove Programs** in the Control Panel.
ARPSupportURL	This populates the **Support URL** field in **Add/Remove Programs** in the Control Panel.
AutoUpgrade	Upgrades the InstallShield Editor template project file (if used) if needed. This option is set to **Y** by default.

Table 8-29 • Options.ini File/General Section Properties (cont.)

Properties	Description
BuildCompressed	This option, set to **Y** by default, compresses all necessary files inside the MSI package, as opposed to storing them outside of the MSI database.
BuildFeatures	If the SmartScan Wizard is specified to include feature information in this project, this option will be set to **Y**. This option is set to **Y** by default.
BuildMSI	Specifies whether or not to build the MSI package after building ISM. This option is set to **Y** by default.
BuildProduct	Identifies the InstallShield Editor Product configuration to build.
BuildProScannedFiles	Files identified in the Media Scan Wizard will be included in the package (unless you have manually excluded them from the project). This option is set to **Y** by default.
BuildRelease	Identifies which InstallShield Editor Release configuration to build.
BuildStaticScannedFiles	Any files identified when running the Setup Intent Wizard will be included in the package (unless you have manually excluded them from the project). This option is set to **Y** by default.
COMMapping	When this option is set to **Y**, registry data pertaining to COM information will be mapped to the appropriate MSI tables whenever possible. This reduces the number of ICE33 warnings that can occur during package validation. This option is set to **Y** by default.
CreateSetupExe	This option, which is set to **N** by default, allows you to automatically create a `Setup.exe` file to begin the installation.
EnablePathVariables	Set this option to **Y** to use path variables. If enabled, the repackaged setup is significantly more portable between computers (with dependencies to the system where the setup was repackaged removed). This option is set to **Y** by default.

Table 8-29 • Options.ini File/General Section Properties (cont.)

Properties	Description
ExtraHKCRPermanent	When this option is set to **Y**, any changes made to existing registry data during repackaging which cannot be identified as belonging to a file installed by the setup are placed in permanent components, which are not removed by default when the repackaged setup is uninstalled. This prevents inadvertently removing registry entries required by other applications that were not originally made by the repackaged setup. By default, this option is set to **Y**, and it is strongly recommended that you retain this setting to prevent unexpected results when the package is uninstalled.
INSTALLDIR	This value will be used for `INSTALLDIR` (the installation directory) and can use a Windows Installer property such as `[ProgramFilesFolder]\MyProgram`
ISProSetup	If one of the original setups that was repackaged was created by InstallShield Professional 5.5 or later, this option will be set to **Y**. This option is set to **N** by default.
LimitedUI	Set this option to **Y** display only the InstallWelcome dialog box when the MSI package is run. This option is set to **Y** by default.
MultiUserShortcuts	When this option is set to **Y**, non-advertised shortcuts will work for all users on the target system. This will generate ICE43 warnings when validation is run. If you know the installation is for a single-user environment, change this option to **N** to avoid these warnings. This option is set to **Y** by default.
MMPathVersion	When including merge modules, if this option is set to **Y**, compare path and version information. This option is set to **Y** by default.
NewInstallDir	Value for INSTALLDIR variable.

Table 8-29 • Options.ini File/General Section Properties (cont.)

Properties	Description
ODBCMapping	If selected, ODBC-related registry data is mapped to ODBC tables instead of the Registry table. This data will only function correctly if Windows Installer supports the ODBC resource being mapped; it is recommended that you do not enable this option if you are unsure whether the ODBC resources are supported correctly by Windows Installer. This option is set to **N** by default.
OSGranular	For SmartScanned or ProLogged Projects: • **If this is set to Y**, component conditions will store specific operating system information. For example, if the filter is NT4, the condition will be (VersionNT=4). • **If this is set to N**, component conditions will store a grouping of the operating system. For example, if the filter is NT4, the condition will be (VersionNT). This is set to **N** by default.

OtherComponentFileExtensions	Specify additional extensions to use when defining components. MSI has rules governing component creation for file types. For example, portable executable (PE) files must have separate components. Therefore, certain extensions have been defined (EXE, DLL, etc.). Additional extensions can be defined in the options.ini file in the format of:
	
	`Type1:Extension1|Type2:Extension2`
	
	where Type is one of the following numbers:
	
	`0 = other`
	`1 = PE`
	`2 = help`
	`3 = font`
	`4 = INI`
	
	This option is set to 1:QTX|1:AX by default.

OtherFilesNewComponents	When this option is set to Y, one component will be created for every file in your setup. Otherwise, new components will only be created for each portable executable file. This option is set to N by default.

Table 8-29 • Options.ini File/General Section Properties (cont.)

Properties	Description
PermanentSystemFiles	Set this option to Y to mark portable executable files installed to a system folder (System32Folder) as Permanent files (will not be uninstalled). This option is set to Y by default.
PermanentSystemFilesSubfolders	Set this option to Y to mark files installed to a subfolder of a system folder as Permanent files (will not be uninstalled). This option is set to N by default.
ProductName	The name of the product. You must provide a value for this option either in this file or in Repackager.
ProductVersion	The version of the product. You must provide a value for this option either in this file or in Repackager.
Project	Name of InstallShield Editor project file.
ProjectTemplate	The name and location of the default InstallShield Editor project template (`.ism`) used in the conversion process.
ServiceControlEvents	When this option is set to Y, the ServiceControl table will be populated for NT Services. This option is set to N by default.
SharedCommonFiles	Set this option to Y to mark portable executable files installed to the CommonFilesFolder (or subfolder) as Shared files. This option is set to Y by default.
SISAuthor	This option populates the Author field of the Summary Information Stream (accessible from the package's properties). This option is set to Repackager by default.
SISSubject	This option populates the Subject field of the Summary Information Stream (accessible from the package's properties).
SkipMMIfShortcut	Merge Modules that have files pointed to by shortcuts should be skipped even if they are not in the exclusion list. This option is set to Y by default.

Table 8-29 • Options.ini File/General Section Properties (cont.)

Properties	Description
SmartScannedOnce	Specifies whether or not the SmartScan Wizard was run for this project. This option is set to N by default.
UseAdvertisedShortcuts	Create advertised shortcuts where applicable. This option is set to Y by default.
UseHKCUProxy	Set this option to Y to copy all registry entries in HKEY_CURRENT_USER to HKEY_USERS\.default. This option is set to N by default.
UseLanguage	When selected, the target package's language will be the language detected by Repackager (as displayed in the Captured Installation view). This option is set to N by default.
UseMergeModules	Set this option to Y to replace files with merge modules whenever possible during conversion. Exceptions are listed under the [MMExclusions] section. This option is set to Y by default.
UseSrcFolder	Set this option to Y to make the InstallShield Editor project (`.ism`) folder default to the Repackager output project (`.inc`) folder. This option is set to Y by default.

[IgnoreShortcuts] Section

Shortcuts that refer to executables listed in this section will be ignored during conversion.

Options.ini File Defaults

This section lists the default settings in the `Options.ini` file that is shipped with Repackager:

```
[MMExclusions]

[General]
UseSrcFolder=Y
EnablePathVariables=Y
UseHKCUProxy=N
LimitedUI=Y
SISAuthor=InstallShield Repackager
OtherFilesNewComponents=N
UseMergeModules=Y
SharedCommonFiles=Y
```

```
PermanentSystemFiles=Y
PermanentSystemFilesSubfolders=N
ExtraHKCRPermanent=Y
COMMapping=Y
ODBCMapping=N
ServiceControlEvents=N
ALLUSERS=Y
ProjectTemplate=
BuildCompressed=Y
CreateSetupExe=N
MultiUserShortcuts=Y
SmartScannedOnce=N
ISProSetup=N
BuildFeatures=Y
OtherComponentFileExtensions=1:QTX|1:AX
OSGranular=N
MMPathVersion=N

[IgnoreShortcuts]
TargetExe1=isuninst.exe
TargetExe2=uninst.exe
TargetExe3=setup.exe
TargetExe4=uninst.dll
TargetExe5=rnuninst.exe
```

Files Associated with Repackager

Several files are associated with Repackager. Some are output files, and some contain default information for Repackager to function. These files are described in the tables below.

Files Used By the Repackaging Wizard

The following files are used by the Repackaging Wizard.

Table 8-30 • Files Used by the Repackaging Wizard

File	Location	Description
`Repack.ini`	*Windows* If the file is not found in the Windows directory, then the Repackaging Wizard extracts a default file from the resource and stores it in the Windows directory.	This is an input file for the Repackaging Wizard. It contains a list of the exclusions for the files, folders, `.ini` files and registry entries for the last used configuration of Repackager. During the Snapshot and Install Monitoring modes of repackaging, the entries in this file are filtered out from the repackaged output. See Repack.ini File for more information.

Table 8-30 • Files Used by the Repackaging Wizard (cont.)

File	Location	Description
`Options.ini`	Repackager output directory (specified in the Set Target Project Information and Capture Settings Panel). The Repackaging Wizard makes a copy of the default `options.ini` that is present in the following directory: *`AdminStudio Installation Directory`*`\Repackager` and saves this file in the same location as the current repackaged output file (`.inc`). Additionally, the `UseSrcFolder` flag can be used to store the created InstallShield Editor file in the same directory as the `.inc` file.	This is an output file from the Repackaging Wizard. It contains configuration information about the repackaged setup, including whether to use path variables, whether to display a limited user interface during installation of the repackaged setup and whether every file will go into its own component.
`productname``.inc`	Created in the `Repackager` output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard. It contains the locations of files, `.ini` files, and shortcuts detected by Repackager as having been created, modified, or removed during repackaging. Also, it contains a link to the `standard.nir` and `deleted.isr` files for registry information.
`updated.isr`	Created in the `Repackager` output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard when the Install Monitoring method is used. It contains registry additions and modifications detected during repackaging using installation monitoring only.
`deleted.isr`	Created in the `Repackager` output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard. It contains registry deletions detected during repackaging using Installation Monitoring and Snapshot.
`standard.nir`	Created in the Repackager output directory (specified in the Set Target Project Information and Capture Settings Panel).	This is an output file from the Repackaging Wizard when the Snapshot method is used. It contains registry additions and modifications detected during repackaging using the Snapshot method.

Table 8-30 • Files Used by the Repackaging Wizard (cont.)

File	Location	Description
`*.spy`	Created in the following folder: *`WindowsDrive`*`\InstallHook`	This is an output file from the Repackaging Wizard when the Install Monitoring method is used. It contains API call logs for installation monitoring.
`Default.ini`	*`AdminStudio Installation Directory`*`\Repackager`	Contains the default configuration for Repackager, including default exclusion information.
`Repack.log`	*WindowsFolder*	Log file created by the Repackaging Wizard.

Files Used By the Repackager Interface

Table 8-31 • Files Used by the Repackager Interface

File	Location	Description
`*.irp`	Saved in the same location as the `.inc` file.	This is a Repackager project file. It is the main file for each repackaged or converted setup. It contains information about the .inc files referred to and also stores the file, folder, `.ini` files and registry exclusions made in the Repackager Interface.
`<Exclusion List>.ini`	varies	This is an input file for the conversion of the .inc file to an MSI package. It contains the list of files, folder, `.ini` files and registry entries exclusions. Users can choose a different exclusion file from the Repackager Interface and the exclusions will be reflected in the Interface.
`Options.ini`	Saved in the same location as the `.inc` file.	This is an input file for the conversion of the `.inc` file to an MSI package. It contains configuration information about the repackaged setup, including whether to use path variables, whether to display a limited user interface during installation of the repackaged setup, and whether every file will go into its own component. Additionally, the `UseSrcFolder` flag can be used to store the created InstallShield Editor file in the same directory as the `.inc` file.

Repack.ini File

The `Repack.ini` file is the default capture exclusion file for the Repackaging Wizard. It contains exclusions to be applied during repackaging, and mainly focuses on specific items that should not be included in applications, such as InstallShield Professional-specific COM settings, OS settings, and Internet Explorer settings. Any item excluded during capture will not be available for exclusion/inclusion in the Repackager project file.

The file is located in the Windows folder, and can be edited using the Exclusions Editor, or using a text editor.

Note • *It is strongly recommended that you not modify this file, as it increases the likelihood of either inadvertently omitting necessary pieces of applications you are repackaging, or including registry entries or files that should not be part of the repackaged application. In the first scenario, you may need to recapture your application; in the second, you may need to exclude more from the Repackager project.*

Instead, capture your application using the default exclusions in the Repackaging Wizard, and then selectively exclude captured data using the Repackager Interface. This way, if you inadvertently exclude a necessary piece, you need only reinclude it in Repackager—not recapture the application entirely.

Using InstallShield to Chain Multiple Windows Installer Packages Together

If your application includes more than one Windows Installer (`*.msi`) package, you can use InstallShield Editor to chain them together using a nested MSI Custom Action. This enables you to run multiple MSI files within a single setup process.

To do this, you open the InstallShield Editor **Custom Actions View** and use the **Custom Action Wizard**.

Task: ***To add a Nested MSI Custom Action:***

1. Launch **InstallShield Editor**.
2. Open your Windows Installer package in Direct Edit Mode.
3. In the Installation Designer, expand the **Behavior and Logic** tree and select the **Custom Actions** node. The Custom Actions View opens.
4. In the middle pane, right-click **Custom Actions** and then click **Custom Action Wizard**.
5. Follow the **Nested Installations** procedure in the InstallShield Editor user documentation to create a nested MSI Custom Action.

Troubleshooting

Repackager Troubleshooting information is presented in the following topics:

- Troubleshooting Guidelines for WinINSTALL Conversion
- Troubleshooting Guidelines for SMS Conversion
- Resolving an “Error Building Table File” Error

Troubleshooting Guidelines for WinINSTALL Conversion

Use the following troubleshooting guidelines to identify and fix WinINSTALL conversion problems.

- **Repackager tool supports 6.0, 6.5, and 7.x project formats only.** For all other formats, please use the WinINSTALL LE tool available as a free download in Windows 2000 to convert to 7.x files.
- **Repackager tool cannot convert WinINSTALL .NAI files**—It can only convert WinINSTALL projects that have been converted to text (`.txt`).

- **All files must be available**—All the files that were available to the original WinINSTALL installation project must be available to the converted installation at the exact same locations.
- **Not all elements of a WinINSTALL installation are converted**—Because WinINSTALL installations are based on a different technology than Windows Installer, not all elements of a WinINSTALL installation are converted. Only the installation of files, registry changes, and other system changes are converted.
- **Custom logic is not converted**—Custom logic written in WinINSTALL's custom scripting language is not converted.
- **WinINSTALL environment variable assignments are not converted**—To re-add environment variable assignments in a Windows Installer installation, open the converted project in InstallShield Editor and use the Environment Variable view.
- **WinINSTALL variables are converted to a Windows Installer variable**—If the target path of a file contains a WinINSTALL variable, then the WinINSTALL variable is converted to a Windows Installer variable.
- **Specify @ variables at conversion time**—If the source path of a file in WinINSTALL contains either the @Server or @Wininstall variable, you can specify the values of these two variables at conversion time in the Repackager.
- **The WinINSTALL Preinstall and Postinstall scripts are not converted.**

Troubleshooting Guidelines for SMS Conversion

Use the following troubleshooting guidelines to identify and fix SMS conversion problems.

- **All files must be available**—All the files that were available to the original SMS installation project must be available to the converted installation at the exact same locations.
- **Not all elements of an SMS installation are converted**—Because SMS installations are based on a different technology than Windows Installer, not all elements of a SMS installation are converted. Only the installation of files, registry changes, `.ini` Files, ODBC, NT Services, Fonts, Shortcuts, Variables, and other system changes are converted.
- **Custom logic is not converted**—Custom logic written in SMS's custom scripting language is not converted.
- **SMS environment variable assignments are not converted**—To re-add environment variable assignments in a Windows Installer installation, open the converted project in InstallShield Editor and use the Environment Variable view.

Resolving an "Error Building Table File" Error

When building with Repackager, if you have received the following error message during the build:

```
ISDEV: fatal Error 5023: Error building table File
```

your first step is to go to the Repackager Interface and check whether the number of files installed by this setup is greater than 32,767. If it is, this error occurs because Windows Installer supports 32,767 files in the File table but the package being built exceeds this limit. See Authoring a Large Package in Windows Installer Help for more information.

If you want to fix this error using Repackager, perform the steps listed below.

Task: ***To fix this error using Repackager:***

1. Browse to the appropriate directory:
   - If you are using the standalone Repackager, browse to the Repackager folder.
   - If you are using the Repackager on a machine where AdminStudio is fully installed, browse to the following directory:

     `<AdminStudio INSTALLDIR>\Editor\Support\0409`
2. Locate the `IsMsiPKg.itp` and `IsMsiPKgLarge.itp` files in this directory.
3. Rename `IsMsiPKg.itp` to `IsMsiPKg.itp.bak`.
4. Make a copy of `IsMsiPKgLarge.itp` and rename the copy `IsMsiPKg.itp`.
5. Perform the conversion and create the MSI.
6. Delete `IsMsiPKg.itp`.
7. Rename `IsMsiPKg.itp.bak` back to `IsMsiPKg.itp`, thereby restoring the original file.

Note • *Transforms and patches cannot be created between two packages with different column types.*

Note • *For more information, see the Authoring a Large Package and File Table topics in the Windows Installer Help.*

9

Performing Automated Repackaging and Virtualization Using the Automated Application Converter

You can use the Automated Application Converter to convert a single package or a group of packages into Microsoft App-V, VMware ThinApp, and Citrix XenApp virtual application formats.

When converting a Windows Installer package to a virtual package, you often need to repackage it prior to being able to successfully convert it. The reason for this is that it is not possible to determine the run-time behavior of certain Windows Installer package elements—such as custom actions, conditional components, and launch conditions—without actually installing the package.

You can use the Automated Application Converter to:

- Examine a group of selected setups.
- Perform automated virtualization of setups that can be cleanly virtualized.
- Perform automated repackaging of those setups that cannot be cleanly virtualized (due to custom actions, etc.), and then perform automated virtualization of those repackaged MSIs.

Information about using the Automated Application Converter is presented in the following sections:

Table 9-1 • AdminStudio Automated Application Converter Help Library

Section	Description
About the Automated Application Converter	Describes the benefits of using the Automated Application Converter to perform automated repackaging and virtualization, and provides an overview of its workflow.
Getting Started With the Automated Application Converter	Explains how to use the Application Conversion Project Wizard to get started using the Automated Application Converter to perform automated repackaging and virtualization.
Managing Virtual Machines	Explains how to use the Virtual Machine Import wizard to add new virtual images to the Automated Application Converter and how to manage virtual machines on the **Machines** tab.

Table 9-1 • AdminStudio Automated Application Converter Help Library

Section	Description
Managing Packages to Convert	Explains how to use the Package Import Wizard to add packages to the Automated Application Converter, and how to manage packages on the **Packages** tab.
Using the Application Conversion Wizard to Perform Automated Package Conversion	Explains how to use the Application Conversion Wizard to perform a conversion run using the selected packages and virtual machines, and how to view conversion run log report information.
Reference	Describes each of the user interface elements and Wizard panels that you might encounter when using the Automated Application Converter. The help topics in this Reference section are the same detailed documentation that is displayed when you press the F1 key or click the **Help** button while working in a dialog box.
Troubleshooting	Includes information to help you resolve typical problems that you might encounter when using the Automated Application Converter.

Note • *Automated Application Converter is available in a Single Application Version (one-at-a-time conversion) and a Multiple Application Version (batch conversion). For more information, see AdminStudio Editions and Components.*

Getting Started With Application Virtualization

AdminStudio provides support for the conversion of Windows Installer packages to the following virtual application formats:

- Microsoft App-V applications
- VMware ThinApp applications
- Citrix profiles, which can be deployed on Citrix XenApp

You have several options when deciding how you want to create a virtual application, depending upon your source files, whether you are an enterprise user or an ISV, and the degree of customization you want to perform:

Table 9-2 • Application Virtualization Support in AdminStudio

If you have ...	And want to ...	Use ...	Description
Windows Installer Package(s) Legacy Application(s) 	**Convert it to a virtual application with ...** • Customized App-V options • Default ThinApp and Citrix XenApp options • Default isolation options	**Automated Application Converter**	Use Automated Application Converter to convert a single or group of Windows Installer (.msi) and legacy (.exe) packages to virtual applications. For detailed information, see Using the Application Conversion Project Wizard. ***Note*** • *For more information, see Using Automated Application Converter vs. the InstallShield Virtual Assistants.*

Table 9-2 • Application Virtualization Support in AdminStudio (cont.)

If you have ...	And want to ...	Use ...	Description
Windows Installer Package or InstallShield Project	**Convert it to a virtual application with ...** • Modified package contents, registry settings, and shortcuts • Custom isolation options on folders and registry entries • Operating system and/or language requirements	**InstallShield Editor Microsoft App-V Assistant** **or** **InstallShield Editor Citrix Assistant** **or** **InstallShield Editor ThinApp Assistant**	Use the InstallShield Editor Microsoft App-V Assistant, ThinApp Assistant or Citrix Assistant to create a virtual application from an InstallShield project or a Windows Installer package. Customization options include: • Modifying package contents, registry settings, and shortcuts • Setting custom isolation options on folders and registry entries • Setting operating system and/or language requirements • Specifying deployment server ***Note*** • *For more information, see Using Automated Application Converter vs. the InstallShield Virtual Assistants.* For detailed information on how to use the Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant, see *Creating Customized Virtual Applications* in the InstallShield Help Library.
Repackager Project	**Convert it to a virtual application with ...** • No modifications • Default isolation options	**Repackager**	By selecting an option on the **Repackaged Output** view, you can simultaneously build an InstallShield Editor project, a Windows Installer package, an App-V application, a ThinApp application, and a Citrix profile from a Repackager project. For information on this feature, see Automatically Generating a Virtual Application During Repackager Project Build.

Using Automated Application Converter vs. the InstallShield Virtual Assistants

Whether you should choose to use Automated Application Converter or an InstallShield Virtual Assistant to perform virtualization could depend upon whether you are a system administrator for an enterprise or an independent software vendor (ISV):

- **Enterprises: Automated Application Converter**—Automated Application Converter is the tool of choice when doing a mass conversion of a variety of setups to virtual packages because it can operate on multiple packages in one project and handle repackaging when it is necessary. This scenario most often applies to enterprises.
- **ISVs: InstallShield Virtual Assistants**—InstallShield Virtual Assistants could be used when focusing on one particular Windows Installer package that does not need to be repackaged. This scenario most often applies to ISVs.

The InstallShield Virtual Assistants allow for customizing the various virtualization-related options for converting a Windows Installer package to a virtual package. In addition, it is possible to make modifications to the source Windows Installer `.msi` file.

Most of the package-level virtualization options—such as whether to compress the package or not—are also available in Automated Application Converter, but file and registry-specific isolation options are only available in the InstallShield Virtual Assistants.

Also, while Automated Application Converter enables you to customize the majority of App-V virtualization options, it does not enable you to set VMware ThinApp or Citrix XenApp conversion options. If the user needs to set some ThinApp or XenApp conversion options, then it would be necessary to use the InstallShield Virtual Assistant for VMware ThinApp or Citrix XenApp.

About Application Virtualization

Note • *This section provides a description of virtualization in general for those that are not familiar with it. It does not represent the architecture of any specific vendor.*

A typical Windows application has dependencies on components that are shared by multiple applications, such as registry entries or COM controls. When an installation author recognizes that their application requires a shared component—such as MDAC (Microsoft Data Access Components)—they include a merge module to install that component.

When one of these shared components is installed during an application's installation, it is possible that a previously-installed version of the same component could be overwritten, causing the existing application to break. Because of these possible problems, extensive compatibility testing needs to be performed before an application can be distributed in the enterprise environment. The following diagram provides an example of two conflicting installed applications.

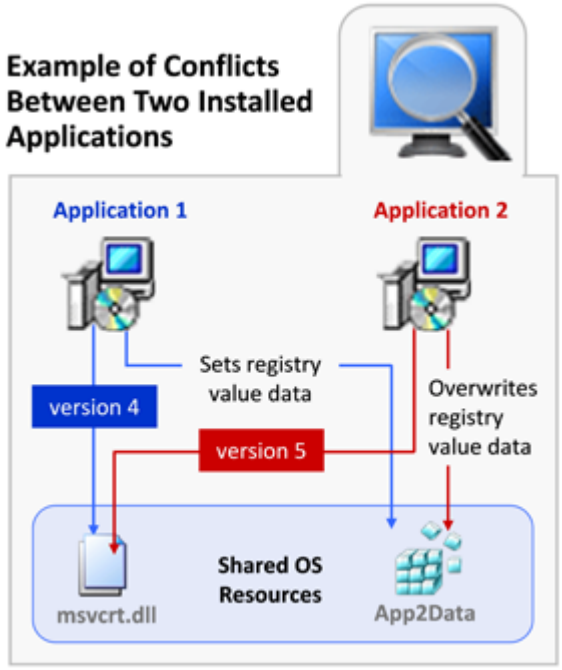


Figure 9-1: Example of Conflicts Between Two Installed Applications

Virtualization simplifies the situation by keeping the application layer and the operating system layer separate, so that the virtual application has no impact on the other applications. In application virtualization, a container or isolation environment is created around the application: a controlled virtual space for application execution that separates the interaction between an application and the underlying operating system's resources in order to protect applications from conflicting with each other.

The following diagram provides an example of how application virtualization would solve the conflicts shown in the previous example.

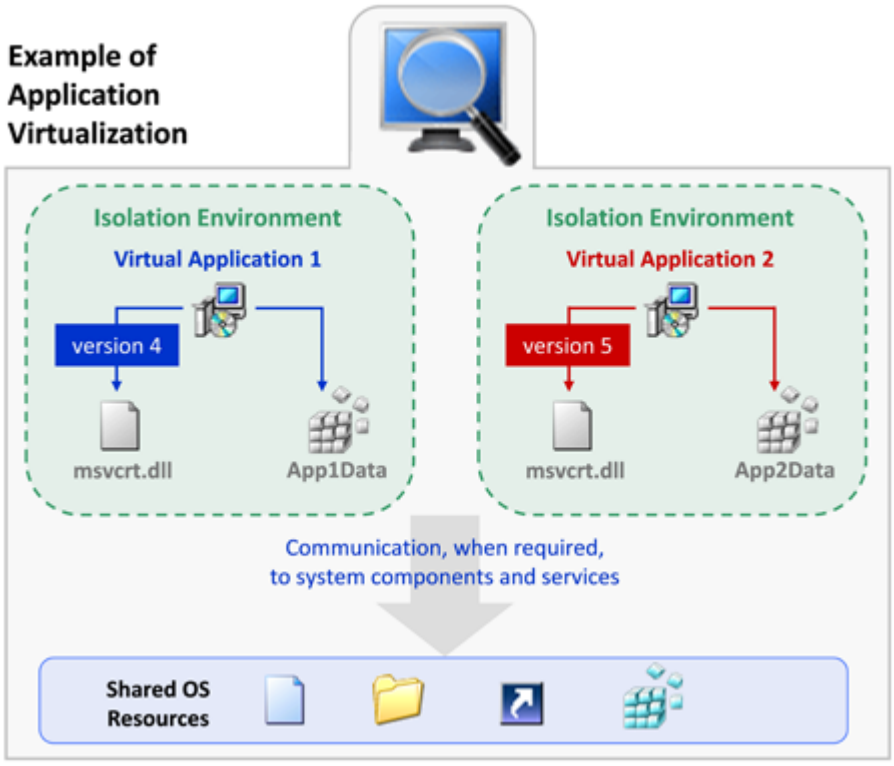


Figure 9-2: Example of Application Virtualization

Application virtualization allows the configuration of an application to be standardized to an isolation environment, rather than to an individual user's desktop machine. Application objects, files and registry settings are contained within this isolation environment. Critical application resources are managed locally by the isolation environment, thus minimizing resource dependencies between applications.

Application virtualization greatly reduces the scope for conflicts between applications and, therefore, simplifies compatibility testing.

About Microsoft Application Virtualization

Microsoft Application Virtualization (App-V) provides the capability to make applications available to end user computers without having to install the applications directly on those computers. Information on Microsoft Application Virtualization is presented in the following topics:

- About Microsoft Application Virtualization (App-V)
- Creating 64-Bit App-V Packages
- Components of an App-V Application

About Microsoft Application Virtualization (App-V)

Microsoft Application Virtualization (App-V) enables applications to run as network services, removing the need for local installation of the applications. An App-V application runs in a self-contained, virtual environment. The virtual environment contains the information necessary to run the application on the client without installing the application locally. Only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middle ware, plug-ins, and other applications.

Because App-V applications are not installed on the client, there is minimal impact on the host operating system or other applications. As a result, application conflicts and the need for regression testing are dramatically reduced.

Using Microsoft Application Virtualization enables you to centralize the installation and management of deployed applications, and control access to applications. The App-V client presents to the end user a list of applications to which that user has access.

Microsoft Application Virtualization Infrastructure

The Microsoft Application Virtualization (App-V) infrastructure includes:

- **App-V Sequencer**—The App-V Sequencer converts application data into a format which is compatible with the App-V server and client, producing an App-V application.
- **App-V Server**—An App-V application can be placed on one or more App-V servers so that it can be streamed down to the clients on demand and cached locally.
- **App-V Client**—The App-V Client is the system component that enables the end user to interact with the App-V applications that are available on the App-V server.

Benefits of Using the Automated Application Converter

Instead of using the App-V Sequencer to create App-V applications, you can use the Automated Application Converter, as shown in the following diagram:

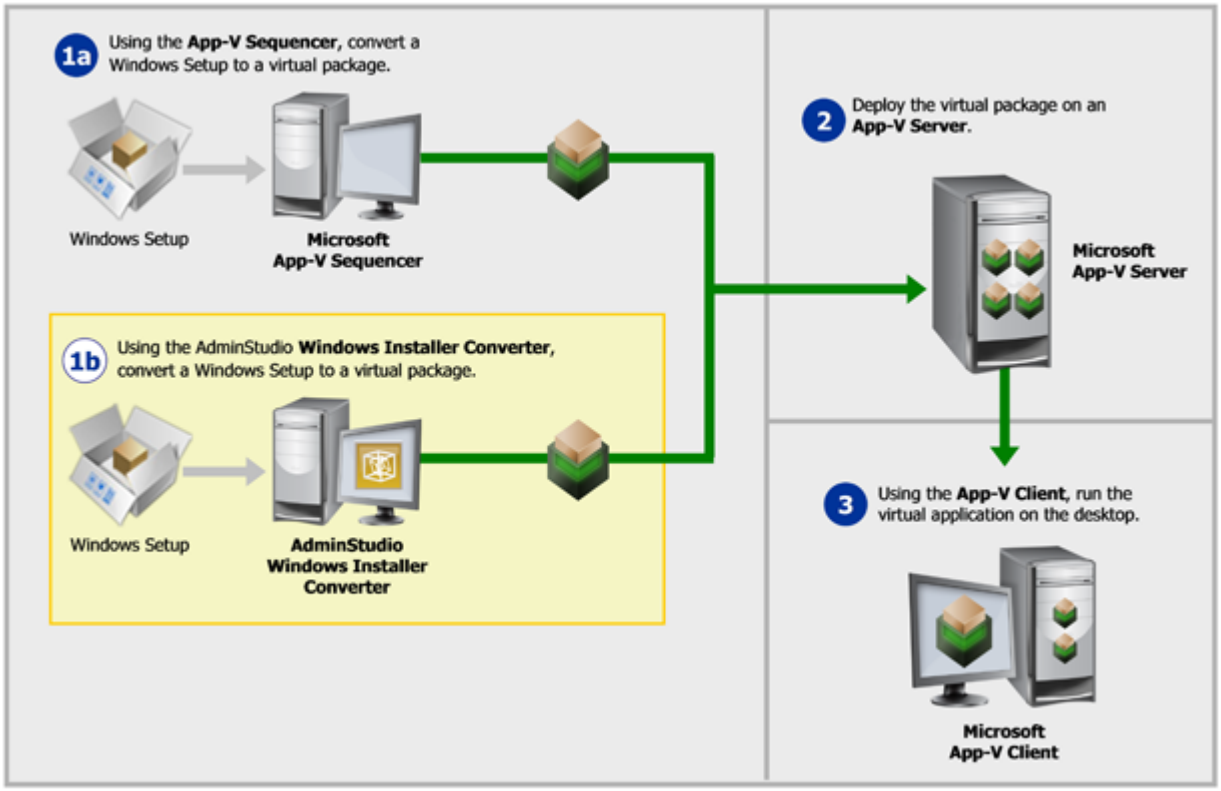


Figure 9-3: Using the Automated Application Converter to Create an App-V Package

Using the Automated Application Converter instead of the Microsoft App-V Sequencer to create an App-V application offers the following benefits:

Table 9-3 • Benefits of Using the Automated Application Converter to Create an App-V Application

Automated Application Converter Benefit	Problem	Solution
Product Installation on a Clean Machine Is Not Required	The Microsoft App-V Sequencer obtains the information it needs to create an App-V application by installing a package on a clean machine and then comparing the file system snapshot that it took prior to installation with one it takes after installation. To perform this task properly, there are two requirements: • **Product must be installed on a clean machine**—To ensure that all proper changes made by the installation are captured, sequencing needs to be performed on a clean machine (a computer with only the operating system, necessary service packs, and the App-V Sequencer installed on it). A new clean machine would need to be recreated for each application that is sequenced. • **Installation directory must be known before sequencing can begin**—In order to sequence the application effectively, you must have detailed knowledge of the how the installation is supposed to work. Prior to beginning the sequencing process, you are required to specify the installation directory for the application being sequenced. This information is often not readily available, and may require you to open the installation in an editing tool, such as InstallShield, in order to find it, or to run the installation one time prior to sequencing.	Instead of installing the package, the Automated Application Converter obtains the information it needs to create an App-V application directly from the installation. You are not required to have any knowledge of settings within the installation, such as the installation directory. Because there is no need to install the application to obtain this information, no permanent changes are made to the local machine and a clean machine is not required.

Table 9-3 • Benefits of Using the Automated Application Converter to Create an App-V Application

Automated Application Converter Benefit	Problem	Solution
Can Test the App-V Application Immediately After Conversion	To run an App-V application, the App-V Client must be installed on the machine. Because sequencing must be performed on a clean machine, which does not have the App-V Client installed, you cannot immediately test a newly-created App-V application on the same machine where you sequenced it.	The Automated Application Converter includes a launch utility that allows you to launch and test the App-V application locally immediately after conversion before distributing them to the App-V Server. ***Note*** • *This feature requires that the App-V Client is installed on the local machine.*
Can Perform Bulk Conversion	When using the App-V Sequencer to create App-V packages, you can only sequence one application at a time. For each application that is sequenced, you need to install the application on a clean machine.	You can use the Automated Application Converter (Multiple Application Version) to perform bulk conversions of multiple applications in a directory hierarchy. The Automated Application Converter has both a user interface and a command line interface.

Creating 64-Bit App-V Packages

AdminStudio supports converting a 64-bit Windows Installer package into a 64-bit Microsoft App-V package, which can be deployed on Windows 64-bit systems with the Microsoft App-V 4.6 64-bit client. This, in combination with our new 64-bit repackaging capability, enables you to convert any 64-bit installation into a 64-bit App-V package.

Note • *App-V packages created using earlier versions of AdminStudio are compatible with the new App-V 4.6 client. In some cases, it may be necessary to manually edit the* `.osd` *files to specify support for 64-bit operating systems.*

Important • *It is highly recommended that you perform the conversion of 64-bit Windows Installer packages to App-V packages on a Windows 64-bit machine. If you attempt conversion on a 32-bit Windows machine, it could result in a failure to extract COM information for 64-bit binaries. Also, in some cases, Windows Installer packages contain shortcuts that target executables not found in the package itself. If these shortcuts target executables found in 64-bit Windows folder locations, then these shortcuts will not be handled correctly on 32-bit machines.*

The following matrix lists the compatibility of App-V packages created using various versions of AdminStudio with the App-V 4.5 and App-V 4.6 clients:

Table 9-4 • App-V Package Compatibility Matrix

Version of AdminStudio Used to Create App-V Package	Application Type	App-V 4.5 Support	App-V 4.6 (32-bit) Support	App-V 4.6 (64-bit) Support
9.01 or 9.5	**32-bit**	Yes	Yes	Yes
	64-bit	Not Supported		
9.5 SP1, 9.5 SP2, 10.0	**32-bit**	Yes	Yes	Yes
	64-bit	No	No	Yes

Components of an App-V Application

When you use Automated Application Converter to build an App-V virtual package, the resources you generate are called App-V applications. The components of an App-V application are shown in the following diagram:

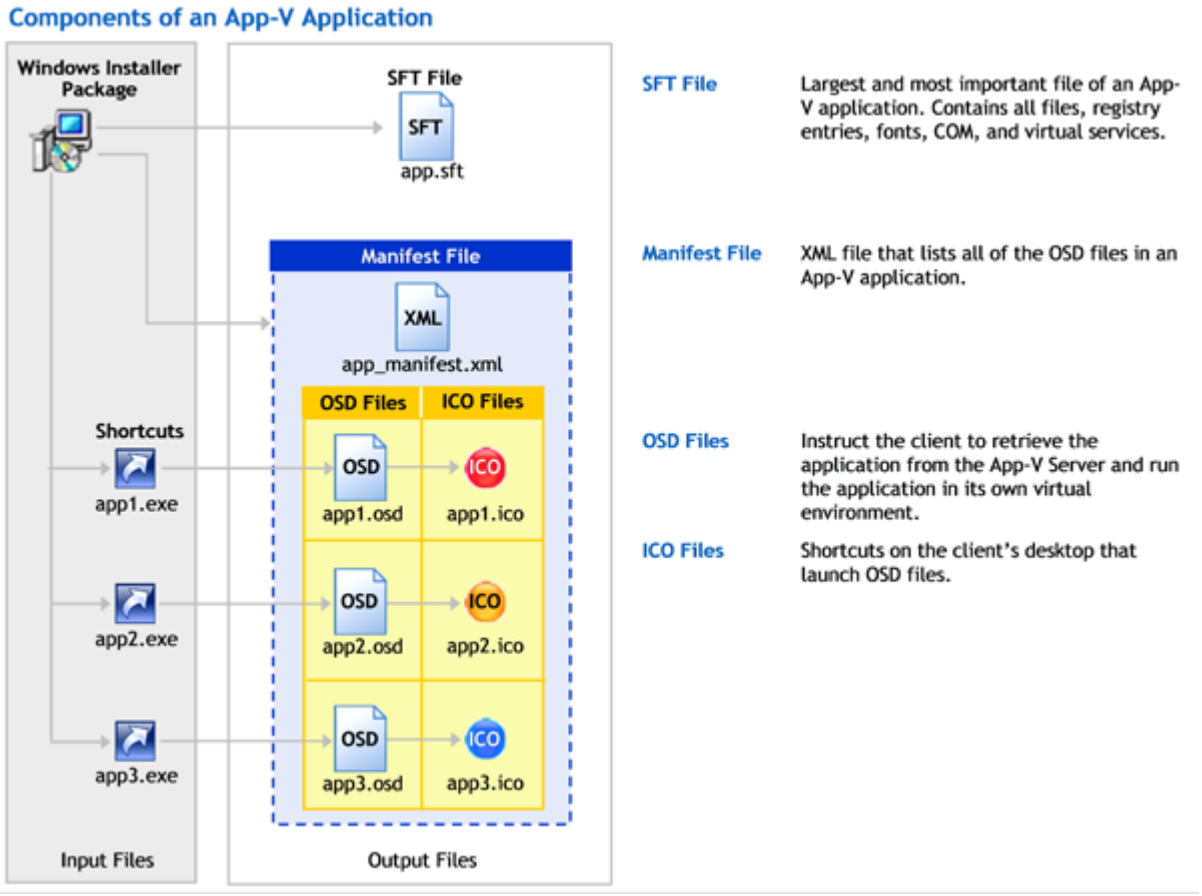


Figure 9-4: Components of an App-V Application

The following table describes the components of an App-V application:

Table 9-5 • Components of an App-V Application

File	Size	Definition
SFT	Up to 4 GB	The largest and most important file in an App-V application. All of the application's assets—including files, registry entries, fonts, COM, and virtual services—reside in this file.

Table 9-5 • Components of an App-V Application

File	Size	Definition
OSD File	2 KB	An XML-based file that instructs the client about the necessary details for retrieving the App-V application from the App-V Server and running the application in its own virtual environment. ***Important*** • *Each .osd file needs to identify the published location of its .sft file on the App-V server. You can specify this location on the Package Information page of the InstallShield Microsoft App-V Assistant. However, if you use the Automated Application Converter to generate an App-V application, you need to use a text editing application, such as Notepad, to individually edit each .osd file and enter the location of the App-V server. See Editing an OSD File to Identify the Location of the App-V Server.*
ICO File	24 KB	Shortcuts on the client's desktop that point to an OSD file and launch App-V applications.
Manifest File	2 KB	XML file that lists all of the OSD files in an App-V application.

When you generate an App-V application, its files are saved in a folder named `App-VPackage` that is created in the same directory as the Windows Installer package you converted. A typical App-V application consists of the following files:

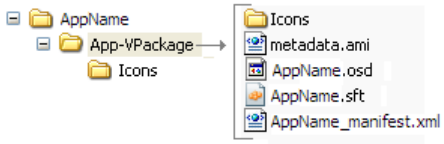


Figure 9-5: App-V Application Files and Directories

Editing an OSD File to Identify the Location of the App-V Server

An `.osd` file is an XML-based file that instructs the App-V client on how to retrieve the App-V application from the App-V server and how to run it in its virtual environment.

Each `.osd` file needs to contain the published location of its `.sft` file on the App-V server. If you set this location on the **Package Information** page of the InstallShield **Microsoft App-V Assistant**, it will automatically be included in all of the `.osd` files that InstallShield builds for that App-V application. However, if you use the Automated Application Converter to generate an App-V application, you need to use a text editing application, such as Notepad, to edit each `.osd` file and enter the published location of its `.sft` file

Task: ***To edit an OSD file by using a text editor:***

1. Open the OSD file using any XML or ASCII text editor—for example, Microsoft Notepad.

   Note • *Before modifying the `.osd` file, read the schema prescribed by the `.xsd` file in the install directory. Failing to follow this schema might introduce errors that prevent a sequenced application from starting successfully.*

2. Locate the `CODEBASE` element. Below is a sample `CODEBASE` element:

```
<CODEBASE HREF="HTTP://%SFT_SOFTGRIDSERVER%:80/orca.sft" GUID="A895355A-5883-41C6-A144-
    1BDA12242AAA" PARAMETERS="" FILENAME="{A895355A-5883-41C6-A144-1BDA12242AAA}\Orca.exe"
    SYSGUARDFILE="{A895355A-5883-41C6-A144-1BDA12242AAA}\osguard.cp" SIZE="2555268"/>
```

3. Locate the `HREF` attribute of the `CODEBASE` element and enter a valid URL to the published location of that App-V application's `.sft` file.

   Note • *When editing an `.osd` file, adhere to the prescribed schema and the following guidelines:*

   - *Ensure that named elements are nested within the `<SOFTPKG>` root element.*
   - *Ensure that element names are in all uppercase letters.*
   - *Be aware that attribute values are case sensitive.*
   - *Type carefully, and observe the XML specifications.*

How Windows Services Are Integrated into an App-V Application

When you use the Automated Application Converter to convert a Windows Installer package to an App-V application, references to Windows services that are encountered are integrated into the App-V application. In a Windows Installer package, a Windows service may be indicated by either an entry in its **ServiceInstall** table or by a Registry entry for Windows services.

- **ServiceInstall table**—If a Windows Installer package's use of a Windows service is indicated by an entry in the **ServiceInstall** table, Automated Application Converter will convert that entry to a standard Registry entry for Windows services.
- **Registry entry**—If a Windows Installer package's use of a Windows service is indicated by a Registry entry for Windows services (perhaps as the result of being repackaged), Automated Application Converter does not need to make any changes to support the application's use of the Windows service within the virtual environment.

Start Up and Shut Down Sequences

If an App-V application has an associated Windows service, App-V will start up the Windows service first, in the virtual environment, and then start up the App-V application. You will see the Windows service start up in the Task Manager as a separate process, but App-V will be running the service within the virtual environment.

Upon shut down, App-V will first shut down the App-V application and then shut down the Windows service.

About ThinApp Virtual Packages

You can use the Automated Application Converter to convert a Windows Installer package to a ThinApp virtual application.

Information about using the Automated Application Converter is presented in the following topics:

- About ThinApp Applications
- Prerequisites for Building a ThinApp Application

Note *• You can also convert a Windows Installer package to a ThinApp application using InstallShield Editor's ThinApp Assistant. Using the ThinApp Assistant, you can configure a ThinApp application's Active Directory settings, files, folders, shortcuts, registry settings, isolation options, and build options. See Getting Started With Application Virtualization and the InstallShield Help Library for information on the ThinApp Assistant.*

Note *• For information on how to simultaneously build an InstallShield Editor project, a Windows Installer package, and a ThinApp application from your Repackager project, see Automatically Generating a Virtual Application During Repackager Project Build.*

About ThinApp Applications

ThinApp is a self-contained application virtualization solution that requires no client-side agents or supporting server infrastructure. A ThinApp application runs within a virtual environment that prevents it from interfering with other software running on the same machine.

ThinApp applications can be deployed on a machine without modifying the local operating system or file system. They run in a "sandbox" (or virtual environment) which protects the local operating system from installation modifications that could affect stability or security. Also, ThinApp applications can be run safely from restricted user accounts without local installation.

Information about ThinApp applications is presented in the following sections:

- ThinApp Virtual Operating System
- Benefits of Deploying ThinApp Applications
- Components of a ThinApp Application

ThinApp Virtual Operating System

A ThinApp application runs in a virtual operating system—a small light-weight component which is embedded with each ThinApp application—that consists of a virtual file system and a virtual registry. When the ThinApp application is run, the virtual operating system environment is merged with the real system environment.

The virtual operating system technology enables entire applications to be packaged into a single `.exe` file that can be run without an installation process, and without modifying the resident operating system.

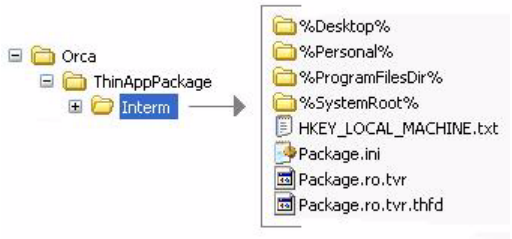
A ThinApp application can be run from a network or offline on the local machine.

Components of a ThinApp Application

When you use Automated Application Converter, Repackager, or InstallShield Editor to build a ThinApp virtual package, the resources you generate are called **ThinApp applications**.

When package conversion is complete, a `Conversion completed` message appears in the **Output** window and the path to the generated ThinApp application is listed, such as:

```
C:\AdminStudio Shared\My Application\ThinAppPackage
```



The ThinApp application is created in a folder named `ThinAppPackage` that is created in the same directory as the Windows Installer package you converted.

The number of files included in a ThinApp application depends upon how many shortcuts are defined:

Table 9-6 • Components of a ThinApp Application

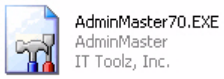
Number of Shortcuts	ThinApp Application Components	Description
1 shortcut	`ProductName.exe` 	The ThinApp application consists of a single executable (`.exe`) file: • **Launching the application**—This executable file is used to launch the ThinApp application. • **Location of application data**—This executable file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

Table 9-6 • Components of a ThinApp Application

Number of Shortcuts	ThinApp Application Components	Description

More than 1 shortcut	`ProductName.exe`
	`FeatureName.exe`
	`Package.DAT`
	
		The ThinApp application consists of two or more executable files and a `Package.DAT` file:
		• **Launching the application**—Each of the executables is used to launch the ThinApp application or a specific feature of the ThinApp application.
		• **Location of application data**—The `Package.DAT` file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

Metadata File	`metadata.ami`	A file created during AdminStudio 9.0+ package conversion that contains metadata identifying the original Windows Installer package that was used to create the virtual package. ***Note*** • *Because of this file, you are able to import this virtual package into the Application Catalog and associate it with its source Windows Installer package. For more information, see Importing Virtual Packages.*

Caution • *Modifying these files directly is **not recommended**. To make any modifications, use the InstallShield ThinApp Assistant.*

Intermediate Data Files: Interm Directory

When a ThinApp application is built, files that support the ThinApp application build process are extracted out of the Windows Installer package and saved in a subdirectory of the `ThinAppPackage` directory named the `Interm` directory.

Figure 9-6: Interm Subdirectory of the ThinAppPackage Directory

The data in this directory is then compiled into ThinApp application as part of the build process. The data in the `Interm` directory *does not* need to be distributed with the ThinApp application.

Benefits of Deploying ThinApp Applications

Deploying ThinApp applications provides the following benefits:

- **Reduces time to deployment and costs associated with testing**—Applications can be deployed and run in independent sandboxes, eliminating the need for expensive and time-consuming multi-application regression testing. This reduces the time to deployment and the costs associated with testing.
- **Fast, lightweight virtualization** —ThinApp does not use emulation, so all processes are executed natively at full speed.
- **Reduces the cost of maintaining secure locked-down desktops**—ThinApp applications can run in restricted user accounts without requiring any host modifications.
- **Enhances work-force mobility, business continuity and disaster recovery**—ThinApp applications can be run off-line, directly from any external media including USB Flash, CD-ROM, and off-line laptops.
- **No infrastructure changes needed**—ThinApp applications can be deployed using any existing software deployment systems including Active Directory and SMS. ThinApp has no client or server components to manage or maintain and ThinApp can transparently stream large applications from any network attached storage devices without server software.
- **Sandboxing prevents modifications**—ThinApp redirects all changes intended for the host computer's file system and registry to a private per-user sandbox. Sandboxes can be located on a network share, allowing application settings to follow users as they move from machine to machine. For mobile users, sandboxes can be stored on local USB flash drives, thus preventing damage to the host computer or accidental host storage of sensitive data.

Prerequisites for Building a ThinApp Application

AdminStudio will convert the package installation into a format compatible with ThinApp. However, the ThinApp build process requires the availability of certain ThinApp tools.

As a prerequisite to building a ThinApp application from AdminStudio, you must have installed ThinApp **and accepted any and all license agreements**.

Caution • *If you install ThinApp but you have not yet accepted the license agreement, the build process will fail. For more information, see the VMware Web site.*

About Citrix Virtual Packages

You can use the Automated Application Converter to convert a Windows Installer package to a Citrix profile for deployment on Citrix XenApp.

Information about using the Automated Application Converter is presented in the following topics:

- About Citrix XenApp and Citrix Profiles

- Benefits of Deploying Citrix Profiles

Note • *You can also convert a Windows Installer package to a Citrix profile using InstallShield Editor's Citrix Assistant. Using the Citrix Assistant, you can configure a Citrix profile's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options. See Getting Started With Application Virtualization and the InstallShield Help Library for information on the Citrix Assistant.*

Note • *For information on how to simultaneously build an InstallShield Editor project, a Windows Installer package, and a Citrix profile from your Repackager project, see Automatically Generating a Virtual Application During Repackager Project Build.*

About Citrix XenApp and Citrix Profiles

Citrix XenApp is an application delivery system for Windows applications. When you use Repackager or InstallShield Editor to prepare a Windows Installer package for deployment on Citrix XenApp, the resources you generate are called **profiles**.

Overview information about Citrix XenApp and Citrix profiles is presented in the following topics:

Table 9-7 • Overview of Citrix XenApp

Topic	Description
About Citrix XenApp	Provides an overview of how Citrix XenApp works and provides a diagram illustrating application delivery.
About Citrix Profiles (.profile)	Lists the files and directories that comprise a Citrix profile.

About Citrix XenApp

Citrix XenApp is an application delivery system for Windows applications that offers both application virtualization and application streaming. Applications are centralized on Citrix XenApp and then those applications are deployed to users throughout the enterprise. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine.

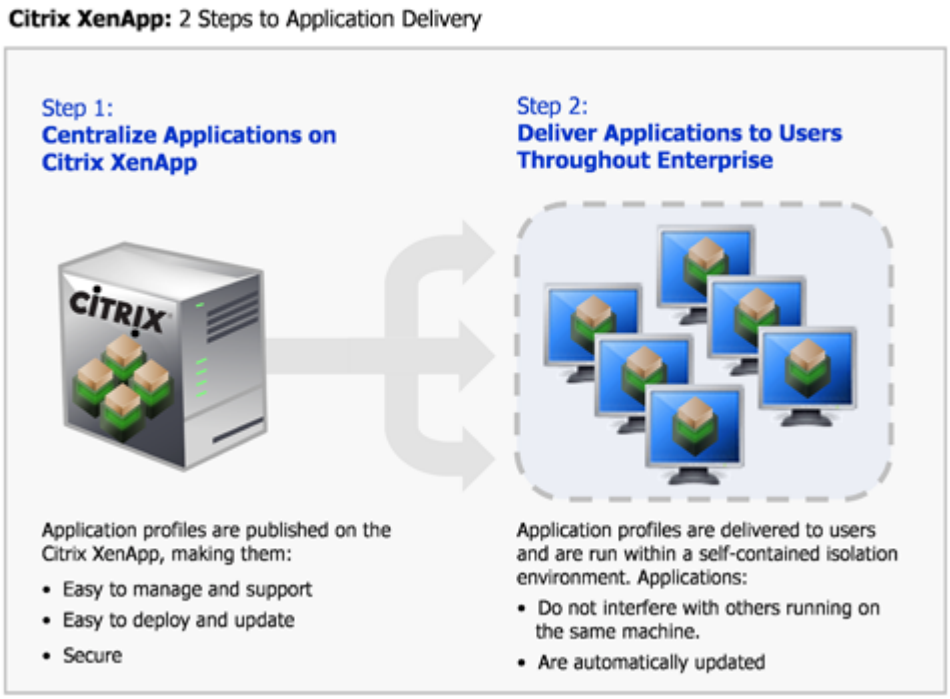


Figure 9-7: Citrix XenApp: Two Steps to Application Delivery

When applications are deployed on a Citrix XenApp, users can run those applications in an isolation environment, without installing, while connected or offline. Applications behave just like they were installed locally, but without any of the problems of installation, such as interfering with other applications on the same device. Files are saved locally and individual settings are preserved. Every time the application is run, it checks for errors or updates and they are delivered automatically.

Note • *For more information, see Benefits of Deploying Citrix Profiles.*

About Citrix Profiles (.profile)

When you use Automated Application Converter, Repackager, or InstallShield Editor to prepare a Windows Installer package for deployment on Citrix XenApp, the resources you generate are called **profiles**.

When package conversion is complete, the Automated Application Converter, Repackager, or InstallShield Editor displays the path to the generated virtual package, such as:

```
C:\AdminStudio Shared\MyPackage\CitrixProfile\MyPackage.profile
```

These files are saved in a subfolder of a folder named `CitrixProfile` that is created in the same directory as the Windows Installer package you converted. The profile, which is published on Citrix XenApp, consists of the following:

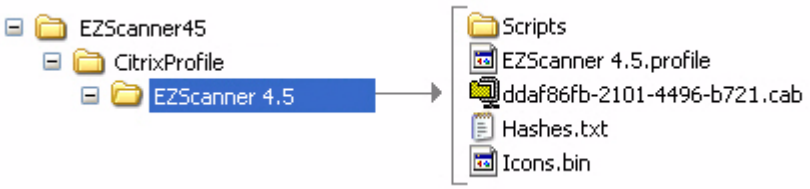


Figure 9-8: Profile Files and Directories

A profile contains the following files and directories:

Table 9-8 • Components of an Application Profile

Component	Name	Description
Profile Manifest File	`myapp.profile`	An XML file that defines the profile.
CAB File	`[alphanumeric_string].cab`	Compressed cabinet file that provides the isolation environment contents for the application.
Hashes File	`Hashes.txt`	Hash key file for digital signatures and signing profiles.
Icons File	`Icons.bin`	Icons repository.
Scripts Folder	`Scripts`	Folder containing any pre- launch or post-exit scripts that you have chosen to include.
Metadata File	`metadata.ami`	A file created during AdminStudio 9.0+ package conversion that contains metadata identifying the original Windows Installer package that was used to create the virtual package. ***Note*** • *Because of this file, you are able to import this virtual package into the Application Catalog and associate it with its source Windows Installer package. For more information, see Importing Virtual Packages.*

Caution • *Modifying these files directly is **not recommended**. To make any modifications, use the InstallShield Citrix Assistant.*

A profile can contain a single application or suite of applications.

Benefits of Deploying Citrix Profiles

Converting a Windows Installer package to a Citrix profile and deploying it on a Citrix XenApp offers the following benefits:

- Reduces Application Conflicts
- Enables Rapid, Low Cost Application Deployment
- Enables Automatic Software Updates
- Centralized Application Management Provides Controlled Access and Security
- Enables User-Based Application Access Rather Than Machine-Based Access

Reduces Application Conflicts

Traditionally to deploy an application throughout an enterprise, the application was installed on each user's desktop. Therefore, prior to installation, each application had to be tested for conflicts against each target desktop image (operating system with existing applications). After resolving conflicts that were found during testing, each application then had to be installed on each desktop. This process was very time consuming not only during initial installation, but also when applying patches or upgrading.

Citrix profiles run within isolation environments, which separate the interaction between an application and the underlying operating system's resources in order to prevent the applications from interfering with others running on the same machine. Because applications do not interact, the need to perform any conflict analysis and regression testing prior to deployment is eliminated. This not only results in rapid application deployment, but it also reduces the total cost of application delivery, due to decreased labor by IT.

Also, because users running applications in an isolation environment encounter no conflicts with other applications, user calls to the help desk are decreased.

Enables Rapid, Low Cost Application Deployment

Deploying Citrix profiles on Citrix XenApp simplifies the deployment of new applications, updates and patch deployment, regardless of the diversity of the access devices, software languages, computing architectures, and networks that are involved.

- **Only a single instance of the application is installed**—Instead of deploying, managing, updating and securing a vast array of heterogeneous client software on each individual user's access device, a single instance of the application is installed on Citrix XenApp. The IT department only has to test for one environment, and deploy and update in one place. This reduces the cost of application installation and support. Also, you can deploy a Citrix profile once on a Citrix XenApp and replicate it to other Citrix XenApps within the existing enterprise infrastructure.
- **Prevents application-specific server silos**—Deploying applications on Citrix XenApp prevents the build-up of application-specific server silos because you can safely install and reliably run multiple application versions and incompatible applications on the same server.
- **Enables you to quickly install and update software throughout your enterprise**—Because you can manage the delivery of all of your Windows-based applications from one centralized location, there is no need to go from desktop to desktop, travel from office to office, or wait for laptops to return to headquarters in order to install or update software. With Citrix XenApp, you can deliver applications and updates instantly anywhere,

any time—to offshore employees, outsourcers, new branch offices, new mergers and acquisitions, and mobile workforces.

Enables Automatic Software Updates

When an upgrade or patch needs to be deployed, you would only need to update the Citrix profile on Citrix XenApp, which will then automatically update all of the instances of that Citrix profile throughout the enterprise. This means that users always have the latest application updates and patches, automatically.

Centralized Application Management Provides Controlled Access and Security

With Citrix XenApp, you can centralize applications and data in secure data centers, which increases data security and ensures fast, reliable performance. Centralized application management using Citrix XenApp provides the following benefits:

- **Enhances security**—Enables you to control, protect, and retain intellectual property centrally to reduce the chance for data loss and theft. Citrix XenApp helps you prevent data from leaving the data center without your explicit permission, which supports regulatory compliance and security objectives. You can provide authorized access to appropriate users—such as employees, customers, and partners—while verifying the ongoing security of the environment.
- **Can provide managed access to applications to users outside of your organization**—You can standardize the use of applications, without having to standardize the machines that the applications use. This enables you to provide managed access to applications from computers that are not your own corporate assets, such as from contractor or consultant computers.
- **Monitors application usage and performance**—Citrix XenApp gives you end-to-end visibility into application usage and performance. It gives IT administrators the power to understand who is using what, how often, and to what extent. They can observe, monitor, measure, audit, report and archive all the dimensions of information flow throughout the computing environment. This enables informed decisions regarding application consolidation and retirement, capacity planning, service level agreements and departmental charge-back.
- **Enables identity-driven access**—Citrix XenApp enables you to provide identity-driven access tailored to any user environment. It automatically analyzes the user's permissions and then delivers the appropriate level of access to applications without compromising security. Depending on who and where users are and what device and network they're using, they may be granted different levels of access. You can also easily "decommission" applications by simply turning off a user's permission to it.

Enables User-Based Application Access Rather Than Machine-Based Access

Users can access their applications anywhere on the network, regardless of where they are or what device they are using.

About the Automated Application Converter

The AdminStudio Automated Application Converter combines the functionality of the Windows Installer Batch Converter with the additional capability to automatically repackage and convert Windows Installer packages, as well as setups in other formats, into virtual applications. You can also choose to automatically repackage setups into Windows Installer packages.

Information about the Automated Application Converter is presented in the following sections:

- Benefits of Using the Automated Application Converter
- Automated Application Converter Workflow Diagram
- Supported Operating Systems
- Supported Virtual Machines

Benefits of Using the Automated Application Converter

Previously, when converting a Windows Installer package to a virtual application, there were cases when you needed to capture its installation prior to being able to perform a successful conversion. Repackaging is sometimes required because it is not possible to determine the run-time behavior of certain Windows Installer package elements—such as conditional components and custom actions—without actually running the install. While converting a Windows Installer package to a virtual application is automated and is a batch process, repackaging setups is a manual process requiring a packager to individually repackage each setup on clean machines and then to convert them into virtual applications—which is a time consuming task requiring several hours of a packager's time.

The Automated Application Converter examines a group of setups to automatically determine which need to be repackaged and which can be virtualized without repackaging. It converts the setups that can be directly converted and then automatically repackages the others by launching virtual machines, running the setups, and capturing them prior to converting them into the target virtual formats.

The Automated Application Converter provides the following benefits:

- **Automated repackaging on virtual machines**—The Automated Application Converter provides an interface to provision and manage virtual machines, silently repackage installs on them, and create virtual packages for the resulting MSIs.
- **Ability to repackage non-MSI setups**—You can use the Automated Application Converter to repackage legacy (non-MSI) setups to create an MSI package that can be converted to a virtual package.

  ***Note** • These setups must support silent install mode.*
- **Conversion of packages from multiple sources**—Using the Automated Application Converter, you can convert setups from multiple sources:
  - AdminStudio Application Catalog
  - Microsoft System Center Configuration Manager
  - Specified directory or file
- **Efficiently manages repackaging queue on multiple virtual machines**—The Automated Application Converter efficiently manages the virtual machine queue, allowing setups to be packaged simultaneously on multiple machines.
- **Easy-to-understand reports**—The Automated Application Converter generates easy-to-understand HTML reports for each conversion run, providing detailed information on each package.
- **Easy-to-read progress indicators with one-click access to virtual machines**—The Automated Application Converter provides dashboard-type progress indicators with one-click access to open a virtual machine in a Remote Desktop session, enabling you to view the progress of a repackaging session and to quickly perform troubleshooting.

Automated Application Converter Workflow Diagram

You can use the Automated Application Converter to examine a group of setups to automatically determine which need to be repackaged and which can be virtualized without repackaging. It converts the setups that can be directly converted and then automatically repackages the others by launching virtual machines, running the setups, and capturing them prior to converting them into the target virtual formats.

The following diagram provides an overview of the AdminStudio Automated Application Converter workflow including:

- Input sources
- Virtualization readiness check
- Automated repackaging on virtual machines
- Conversion to virtual packages
- Output types

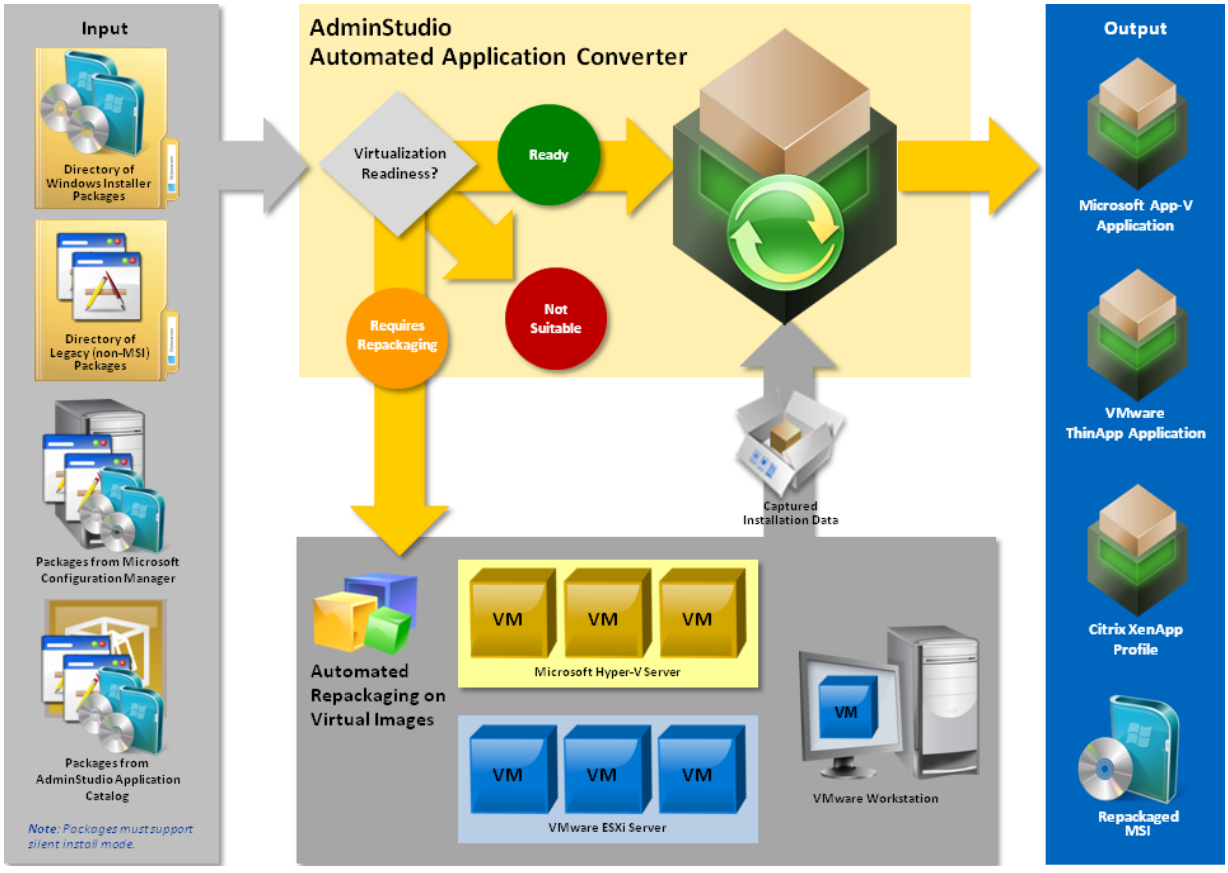


Figure 9-9: AdminStudio Automated Application Converter Workflow Diagram

Supported Operating Systems

The Automated Application Converter supports the following operating systems:

- Windows XP (32-bit)
- Windows Vista (32-bit and 64-bit)
- Windows 7 (32-bit and 64-bit)
- Windows Server 2008 (32-bit and 64-bit)
- Windows Server 2003 (32-bit and 64-bit)
- Windows Server 2008 R2

Supported Virtual Machines

The Automated Application Converter supports automated repackaging on virtual machines from the following platforms:

- Microsoft Hyper-V Server
- VMware ESX or ESXi Server
- VMware Workstation 6.5 or later

Launching the Automated Application Converter

The Automated Application Converter can be launched by doing either of the following:

- On the Windows Start Menu, point to **All Programs**, **AdminStudio**, **AdminStudio Tools**, and click **Automated Application Converter**.
- Launch **Repackager**, and then click the **Automatically Repackage Installations on Your Virtual Machines** link on the Repackager Home Page.

The **Open Project** panel of the Application Conversion Project Wizard opens.

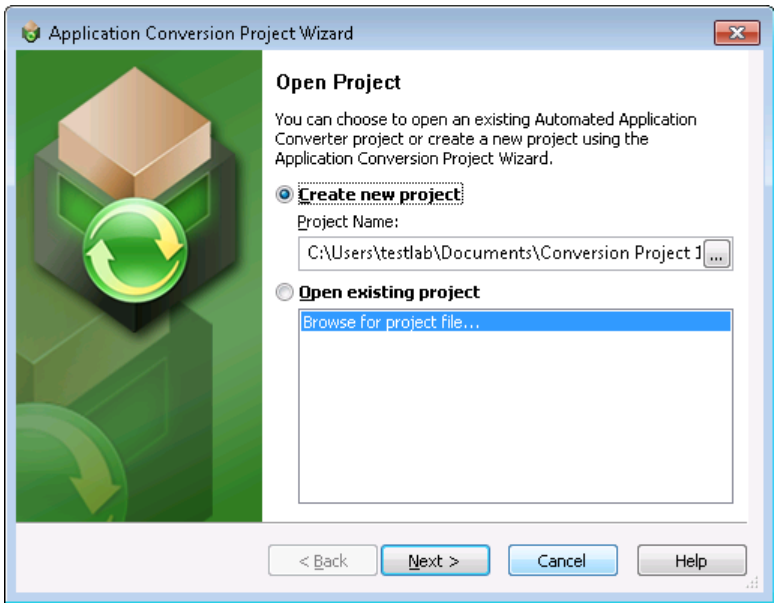


Figure 9-10: Open Project Panel of Application Conversion Project Wizard

On the **Open Project** dialog box, you can choose to create a new project or open an existing project. For more information, see Opening a Project.

Getting Started With the Automated Application Converter

The quickest way to get started using the Automated Application Converter is to use the end-to-end **Application Conversion Project Wizard**, which takes you through the three main steps in automated batch virtualization: selecting the packages to convert, selecting the virtual machines to use for repackaging, and converting the selected packages. See Using the Application Conversion Project Wizard for instructions.

You can also choose to perform each of these tasks separately by using one of the other three wizards that are provided:

Table 9-9 • Automated Application Converter Wizards

If you want to ...	Use this wizard ...	Description and Purpose
Add virtual machines	**Virtual Machine Import Wizard**	Add virtual machines to use to perform automated repackaging to Windows Installer packages. See Adding Virtual Machines Using the Virtual Machine Import Wizard.
Add packages	**Package Import Wizard**	Add packages from an AdminStudio Application Catalog, a Microsoft Configuration Manager server, or from a local or network file system. See Selecting Packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server and Selecting Packages from a Local Machine or Network.
Virtualize or repackage packages	**Application Conversion Wizard**	Virtualize packages to the virtual formats you specify. You can also perform repackaging. See Using the Application Conversion Wizard to Perform Automated Package Conversion.

Opening a Project

When you launch the Automated Application Converter, the **Open Project** dialog box opens, prompting you to either create a new project or open an existing project.

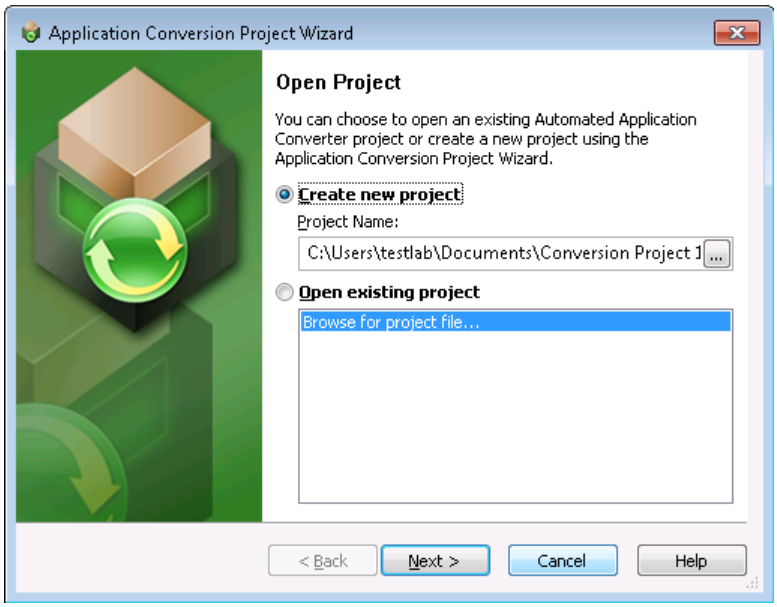


Figure 9-11: Open Project Dialog Box

The following procedures explain how to create a new project or open an existing project.

- Creating a New Project
- Opening an Existing Project

Creating a New Project

To create a new project, perform the following steps.

Task: ***To create a new project:***

1. Do one of the following depending upon whether the Automated Application Converter is open:
   - **Not open**—Launch the Automated Application Converter.
   - **Open**—On the **File** menu, click **New Project**.

   The **Open Project** dialog box opens.
2. Select **Create new project**.
3. Click the Browse button next to the **Project Name** field. The **Save As** dialog box opens.

4. Enter a name (with an `.aacx` extension) and location for the new project file and click **Save**. The new project name is now listed in the **Project Name** box.
5. Click **Next**. The **Select Package Source** panel of the Application Conversion Project Wizard opens.
6. Continue with the steps in Using the Application Conversion Project Wizard.

Opening an Existing Project

To open an existing project, perform the following steps.

Task: ***To open an existing project:***

1. If the Automated Application Converter is not open, perform the following steps:
   a. Launch the Automated Application Converter. The **Open Project** dialog box opens.
   b. Select **Open existing project**.
   c. From the list, either select a project name or select **Browse for project file**.
   d. Click **Finish**. One of the following occurs:
      - **If you selected an existing project** from the list, the project opens in the Automated Application Converter interface.
      - **If you selected Browse for project file**, the **Open** dialog box opens. Select a project file and click **Open**. The project opens in the Automated Application Converter interface.
2. If the Automated Application Converter is open, perform the following steps:
   a. On the **File** menu, click **Open**. The **Open** dialog box opens.
   b. Browse to the project file you want to open and click **Open**. The project opens in the Automated Application Converter interface.

Note • *If you had unsaved changes in the project file that was already open, you will be prompted to save those changes prior to opening the new project file.*

Using the Application Conversion Project Wizard

To get started using the Application Conversion Project Wizard, perform the following steps:

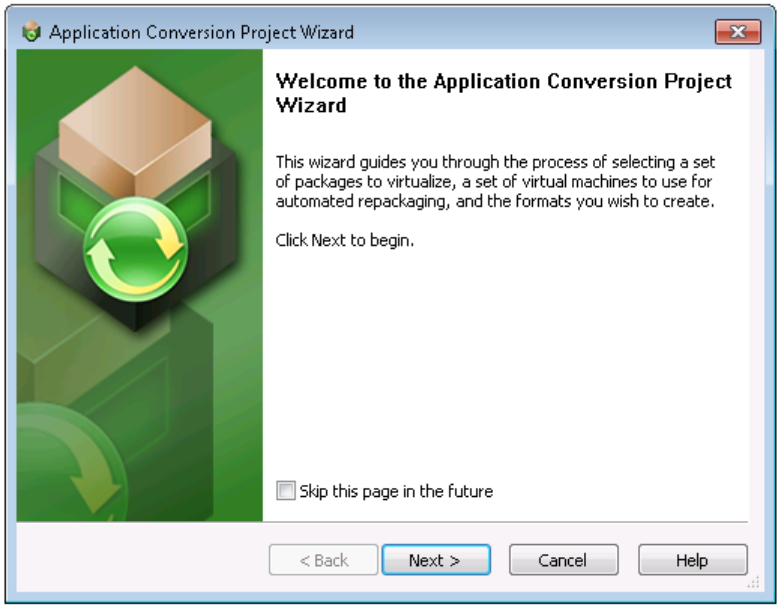
Task: ***To get started using the Application Conversion Project Wizard:***

1. Perform the steps in Preparing Your Virtual Machines for Use With the Automated Application Converter to prepare your virtual machines to use for automated repackaging.
2. Launch the Automated Application Converter by doing one of the following:
   - On the Windows Start Menu, point to **All Programs**, **AdminStudio**, **AdminStudio Tools**, and click **Automated Application Converter**.
   - Launch **Repackager**, and then click the **Automatically Repackage Installations on Your Virtual Machines** link on the Repackager Home Page.

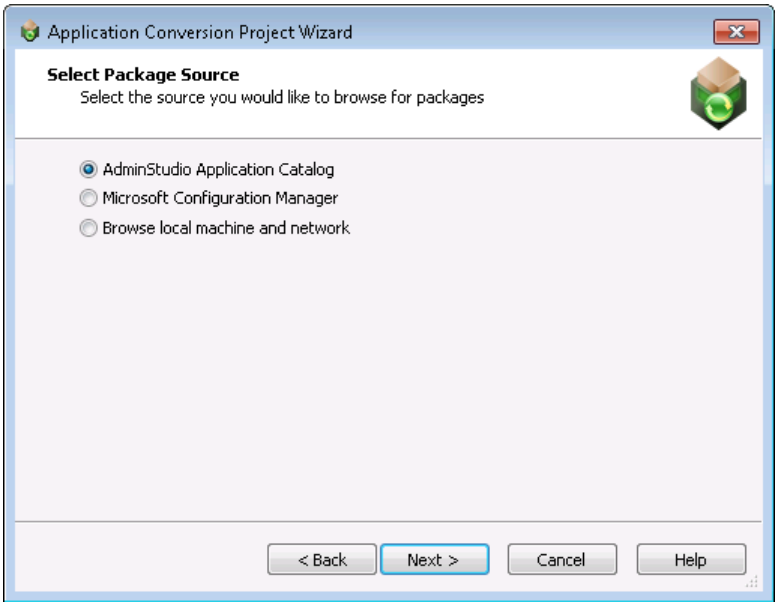
   The Automated Application Converter opens.

   Note • *See Automated Application Converter User Interface for more information.*

3. On the **Tools** menu, click **Project Wizard** (or click the icon in the toolbar). The **Welcome to the Application Conversion Project Wizard** panel opens.



4. Click **Next**. The **Select Package Source** panel opens, prompting you to select the source that contains the packages you want to convert.

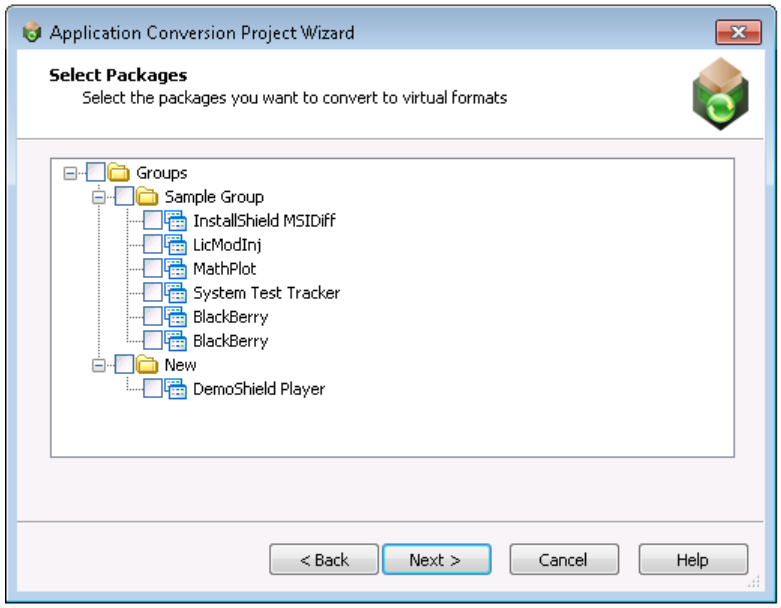


5. Select one of the following options:

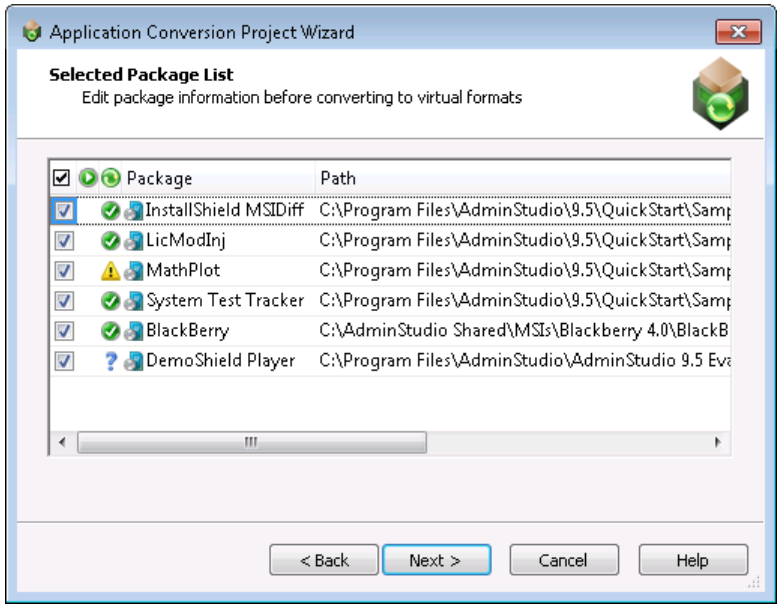
Option	Description
AdminStudio Application Catalog	Select this option to connect to an AdminStudio Application Catalog and add all of the installations in that catalog to the list of packages to convert. If you select this option, the Connect to an AdminStudio Application Catalog panel opens, prompting you to login to an Application Catalog.
Microsoft Configuration Manager	Select this option to connect to a Microsoft System Center Configuration Manager Server and add all of the installations on this server to the list of packages to convert. If you select this option, the **Connect to a Microsoft Configuration Manager Server** panel opens, prompting you to login to a Microsoft Configuration Manager Server.

Option	Description
Browse local machine and network	Select this option to browse a local or network machine to add installations to the list of packages to convert. If you select this option, the **Select Packages** panel opens, where you are prompted to select an installation file or a directory of installation files to add to the list of packages to convert. ***Note*** • *For information on the rules that the Automated Application Converter uses to determine which packages in the selected directory's subdirectories would be added to the list on the* ***Selected Package List*** *panel, see Automated Application Converter's Selection Rules When Adding Packages from a Directory.*

If you connected to an Application Catalog or Microsoft Configuration Manager Server, the **Select Packages** panel opens.



When you have finished this step, packages will be listed and selected on the **Selected Package List** panel, and an icon indicates each package's virtualization readiness status.

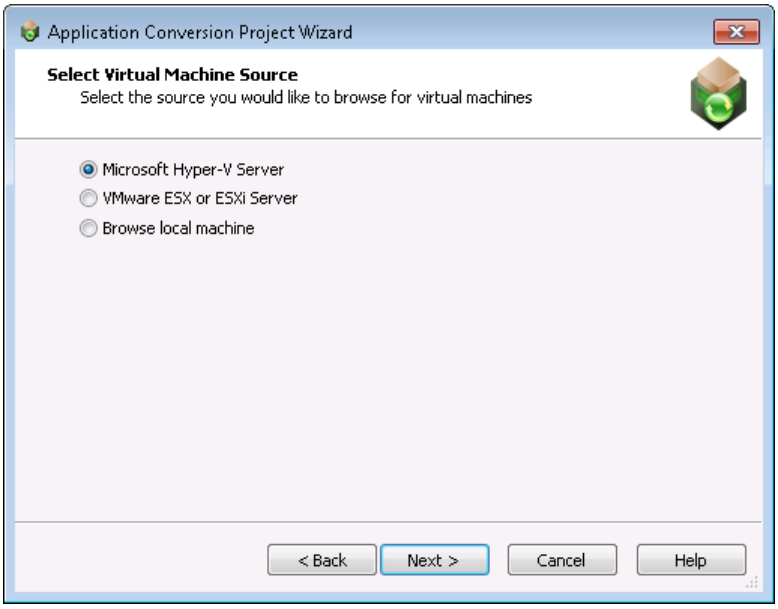


One of the following icons is listed in each package's Virtualization Readiness ( ) column:

Icon	Meaning	Description
	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** *• If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required.* *An example of an unsupported table is the* `IniFile` *table, which changes files on the target machine in ways that cannot be statically determined.*
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.

Icon	Meaning	Description
	Virtualization not supported	Automated Application Converter has determined that virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready to Virtualize** or **Requires Repackaging**.* ***Note*** • *For more information, see Virtualization Suitability Issues.*
	Virtualization not recommended	Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues: • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology. ***Note*** • *For more information, see Virtualization Suitability Issues.*
	Unknown	The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.

6. Click **Next**. The **Select Virtual Machine Source** panel opens, prompting you to select the type of virtual machine that you are going to use for automated repackaging.

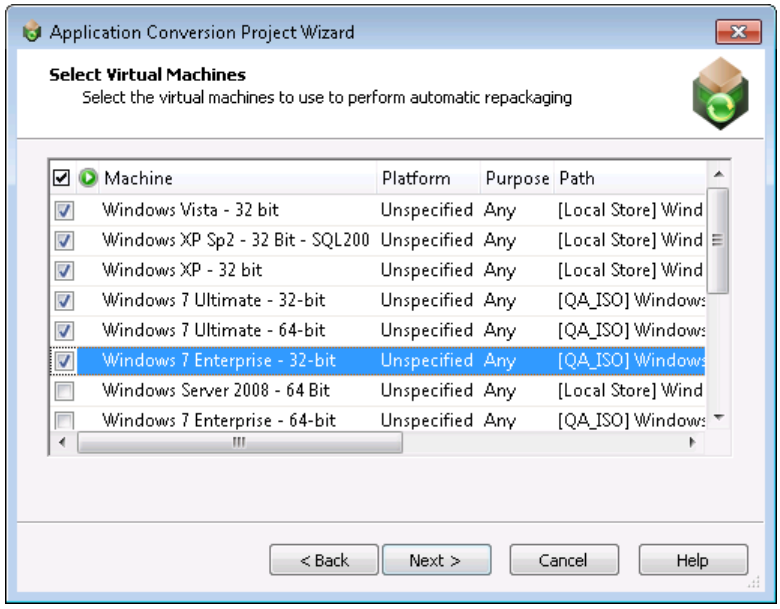


Note • *If none of the packages selected on the **Selected Package List** panel require repackaging in order to be converted into a virtual package, the **Select Virtual Machine Source** panel will not be displayed. Instead, the **Initial Configuration Complete** panel will open.*

Select one of the following options and click **Next**:

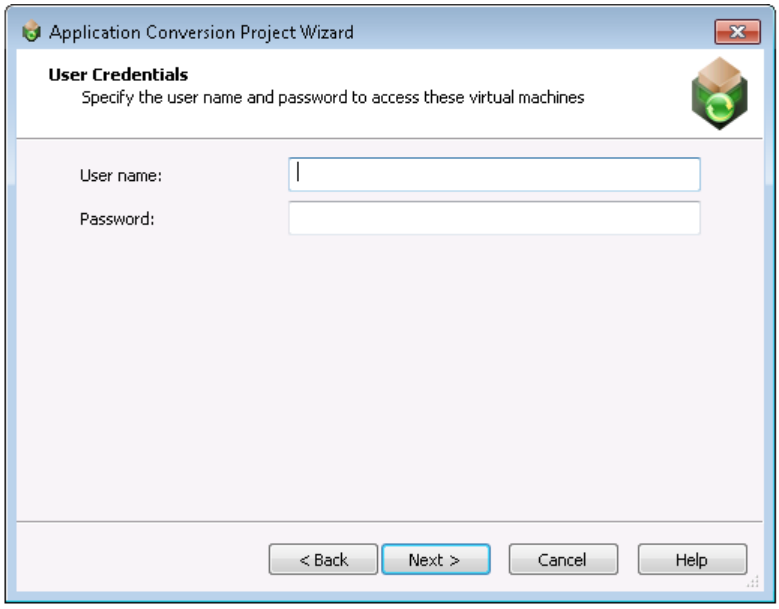
Option	Description
Microsoft Hyper-V Server	Select this option to connect to a Microsoft Hyper-V Server. You will then be prompted for login information on the Select Virtual Machines from a Microsoft Hyper-V Server panel.
VMware ESX or ESXi Server	Select this option to connect to a VMware ESX or ESXi Server. You will then be prompted for login information on the Select Virtual Machines from a VMware ESX or ESXi Server panel.
Browse local machine	Select this option to connect to a VMware Workstation virtual image installed locally. The Select Virtual Machines opens, where will be prompted to select either a VMware Workstation image or directory of images.

When you have finished this step, the virtual machines will be listed (but not selected) on the **Select Virtual Machines** panel.

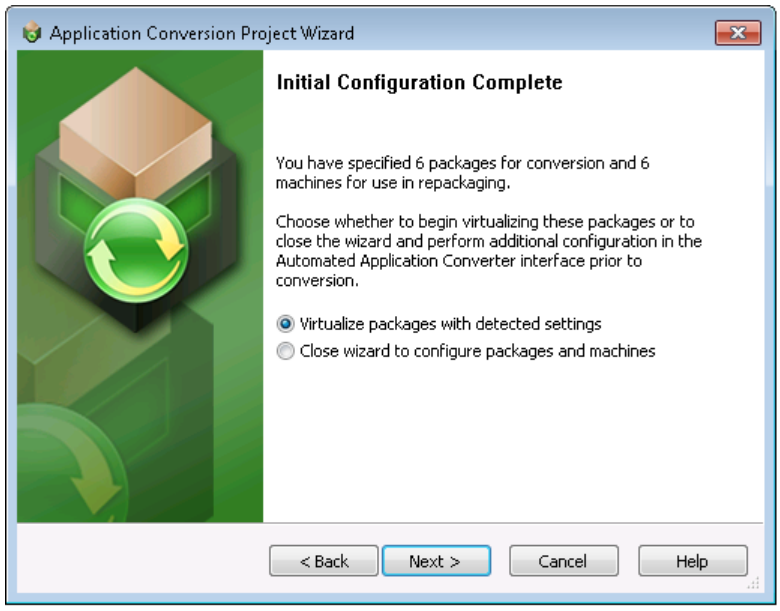


7. On the **Select Virtual Machines** panel, select the virtual machine images that you want to use to perform automated repackaging.
8. For each selected image, click in the **Platform** column and identify its platform.
9. Optionally, if you want to limit the use of a virtual machine to either repackaging only or testing only, click in the **Purpose** column and select **Repackaging** or **Testing** from the list. The default value is **Any**.

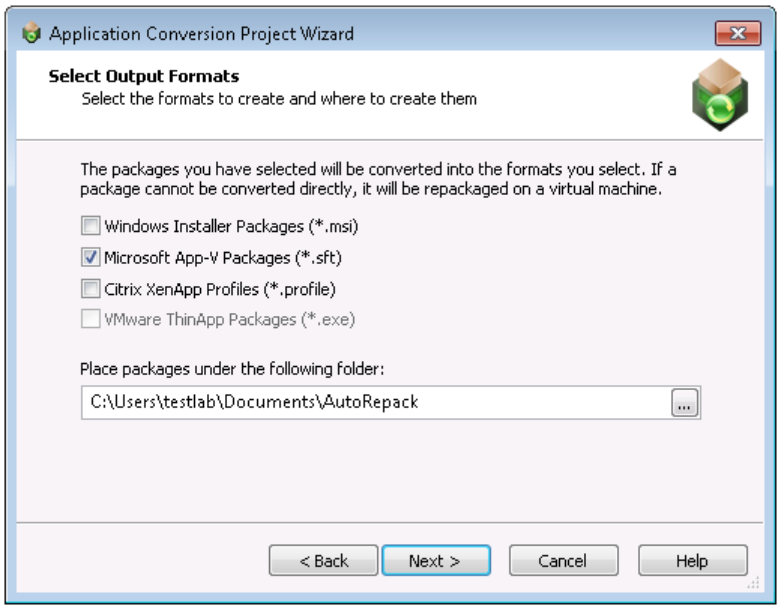
10. Click **Next**. The **User Credentials** panel opens, prompting you to specify the login credentials to use to access the selected virtual machines.



11. Enter login credentials and click **Next**. The **Initial Configuration Complete** panel opens, listing a summary of your selections, and prompting you to select whether you want to begin to **Virtualize packages with detected settings** or to **Close wizard to configure packages and machines**.



12. Select **Virtualize packages with detected settings** and click **Next**. The **Select Output Formats** panel opens, prompting you to select one or more output formats:



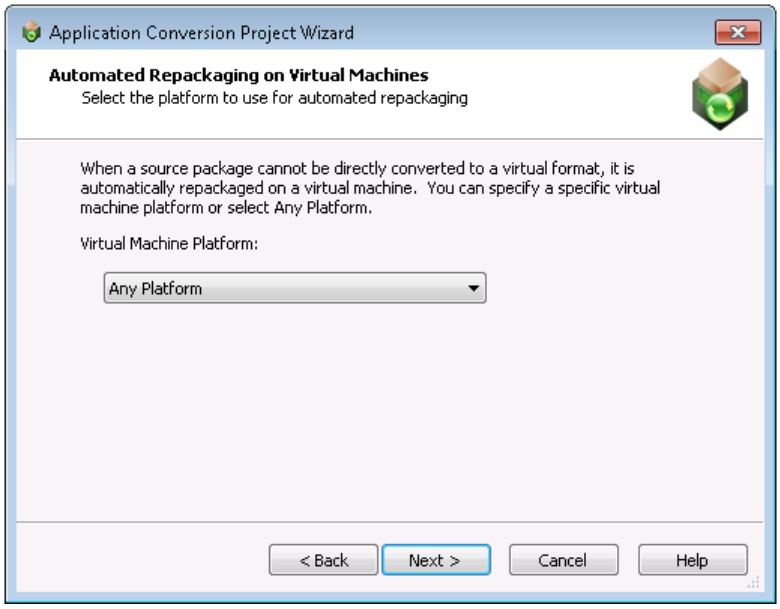
Note • *If you have selected Windows Installer packages on the **Selected Package List** panel, but those packages do not require repackaging prior to virtualization, the **Windows Installer Package (*.msi)** option on the **Select Output Formats** panel will be disabled. If you want to force the Automated Application Converter to repackage that package, return to the **Selected Package List**, click in that package's Virtualization Readiness column and select **Requires repackaging** from the list.*

13. On the **Select Output Formats** panel, select one or more of the following formats:

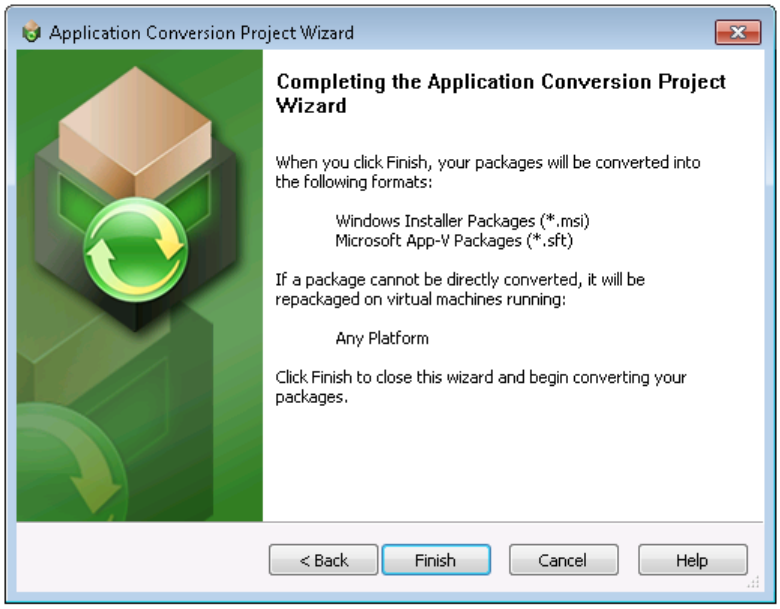
Option	Description
Windows Installer Packages (*.msi)	Select this option to repackage the selected packages into Windows Installer packages (.msi).
Microsoft App-V Packages (*.sft)	Select this option to convert the selected packages to Microsoft App-V virtual applications.
Citrix XenApp Profiles (*.profile)	Select this option to convert the selected packages to Citrix XenApp profiles.
VMware ThinApp Packages (*.exe)	Select this option to convert the selected packages to VMware ThinApp virtual applications.

14. Under **Place packages under the following folder**, select the output location where you want to store the packages.

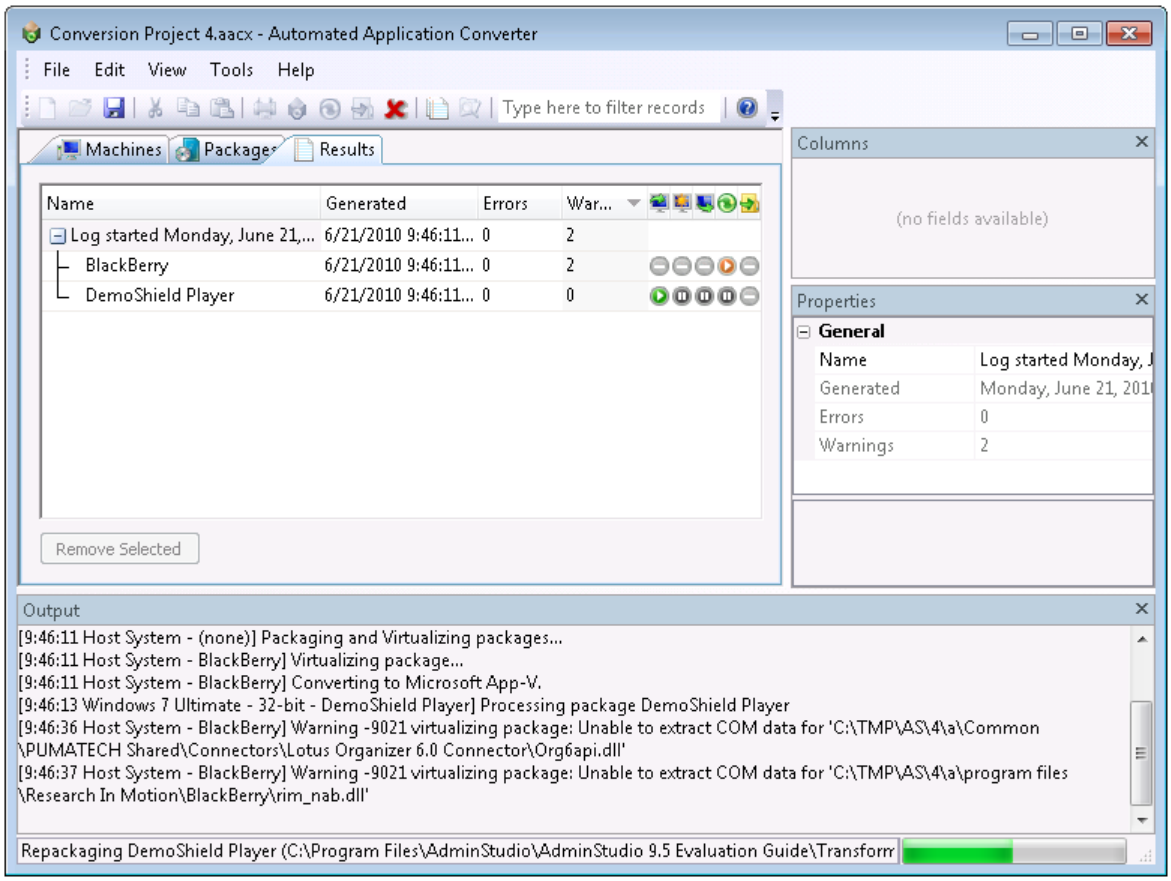
15. Click **Next**. The **Automated Repackaging on Virtual Machines** panel opens, prompting you to select the platform of the virtual machines that you want to use to perform automated repackaging during this conversion process.



16. From the **Virtual Machine Platform** list, select a platform, or leave **Any Platform** selected, and click **Next**. The **Application Conversion Project Wizard Complete** panel opens.



17. Click **Finish** to close the wizard and begin converting the selected packages.The conversion process begins. The **Results** tab opens and messages are displayed in the **Output** window.

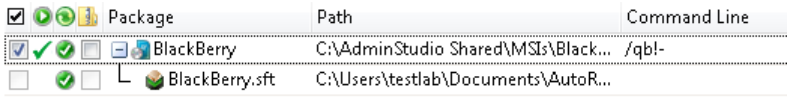


Icons displayed on the **Results** tab indicate each package's progress:

Column	Icon	Description
Copy In **Repackage** **Copy Out**		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed, but warnings were encountered. View the results AdminStudio Automated Application Converter Log Report for detailed information on these warnings.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) failed. • **Copy In**—Error could have been caused by not being able to connect to the virtual machine. • **Repackage**—Error means that repackaging has failed. • **Copy Out**—Error could mean that you ran out of hard drive space at the package source location or that there is a permission problem preventing you from writing to the selected directory. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was skipped. Possible reasons that the operation was skipped could be: • **Repackaging not required**—Because repackaging was not required, these three operations were not required. • **Could not connect to virtual machine**—The Automated Application Converter could not successfully connect to the virtual machine, so therefore the **Repackage** and **Copy Out** operations were skipped.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is currently being performed.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is still being performed even though a warning was generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation was cancelled

Column	Icon	Description
Conversion Column		Package was converted to a virtual application successfully.
		Package was converted to a virtual application, but warnings were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Package was converted to a virtual application, but errors were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		The Automated Application Converter was unable to convert this package to a virtual application.
		Conversion is in progress.
		Conversion is in progress, but a warning has been generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		An error was generated when converting one of the virtual formats which caused it to fail. However, the conversion to another one of the selected virtual formats continues.
		Conversion was cancelled

18. When conversion is complete, the virtual packages will be listed in a tree structure under the original package on the **Packages** tab.



19. To view the **AdminStudio Automated Application Converter Log** report, select the top level node of the conversion run log (such as `Log started Monday, June 21, 2010...`) on the **Results** tab and do one of the following:

    - Click the **Results** button on the toolbar.
    - Select **View Report** from the context menu.
    - Select **View Report** on the **Tools** menu.
    - Press Ctrl+R.

    See AdminStudio Automated Application Converter Log Report for more information.

20. Continue with the steps in Launching Packages for Testing and Publishing Converted Packages.

About Automated Application Converter Project Files

All of the selections that you make on wizard panels or in the Automated Application Converter interface are saved in an XML-based project file: `ProjectName.aacx`. You can also choose to modify project settings by editing this XML file.

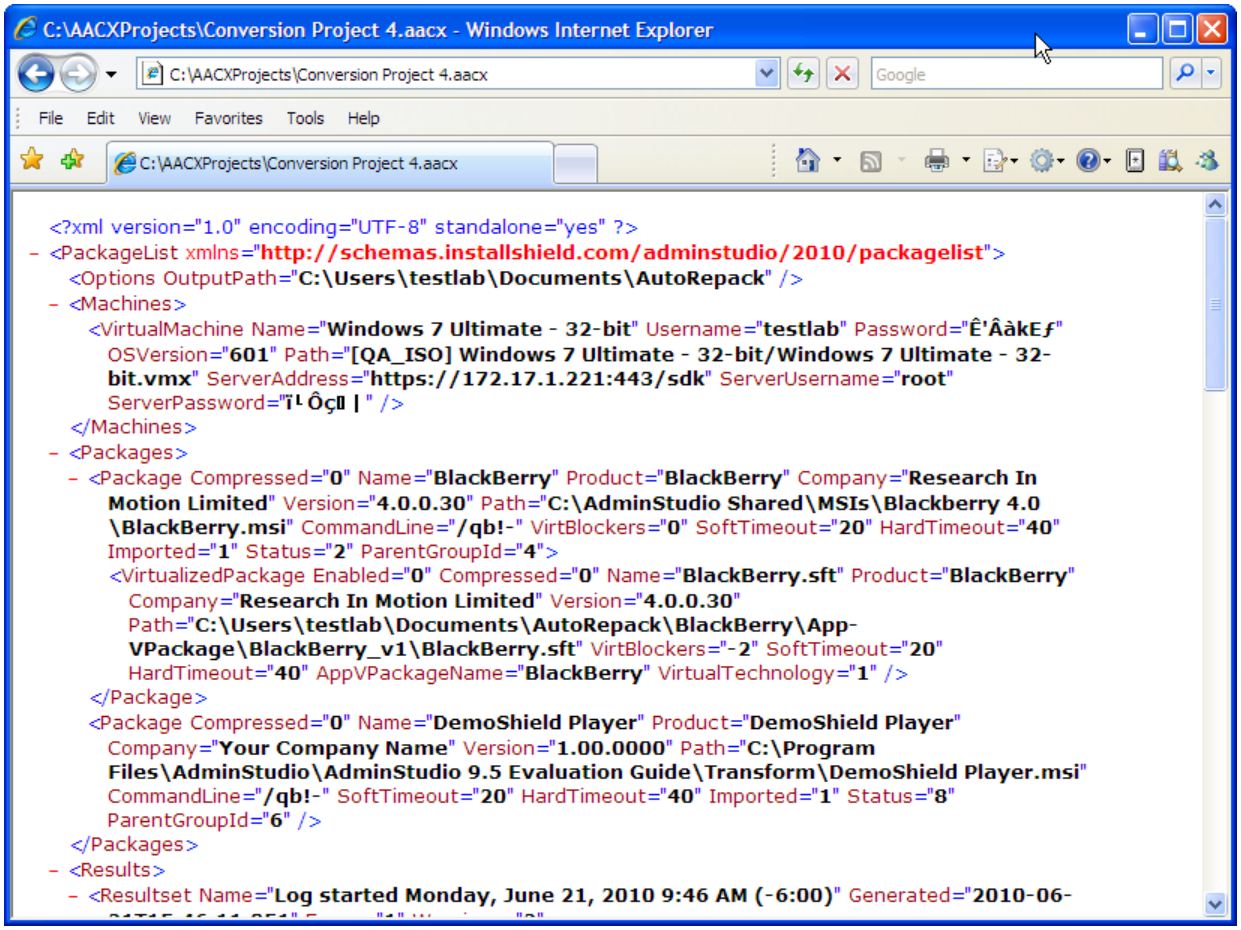


Figure 9-12: Sample Project File

Note • *To both launch the Automated Application Converter and open a project, you can double-click a project file in Windows Explorer.*

As shown in the following example, each project file is enclosed within a <PackageList> element, and the major sub-elements of a project file are <Options>, <Machines>, <Packages>, and <Results>. All of the settings specified in the Automated Application Converter interface appear in this file.

Table 9-10 • Elements Comprising an Automated Application Converter Project File

Elements

```
<PackageList>
    <Options />
    <Machines>
        <VirtualMachine />
        <VirtualMachine />
    </Machines>
    <Packages>
        <Package>
            <VirtualizedPackage />
        </Package>
        <Package>
            <RepackagedPackage />
        </Package>
    </Packages>
    <Results>
        <Resultset>
            <Machines>
                <UseMachine />
                <UseMachine />
            </Machines>
            <Packages>
                <UsePackage />
                <UsePackage />
            </Packages>
            <Messages>
                <LogItem />
                <LogItem />
            </Messages>
            <Result>
                <Messages>
                    <LogItem />
                    <LogItem />
                </Messages>
            </Result>
        </Resultset>
    </Results>
</PackageList>
```

The following table describes the major elements of a project file:

Table 9-11 • Major Elements of an Automated Application Converter Project File

Element	Description
PackageList	The root element of an Automated Application Converter project file is the <PackageList> element: <PackageList xmlns="http://schemas.installshield.com/adminstudio/2010/packagelist"> The PackageList element identifies the location of the XML namespace used by the application. XML namespaces provide a method to avoid element name conflicts.
Options	The <Options> element of a project file identifies the output location of the converted packages and identifies the currently selected output formats: <Options OutputPath="C:\Users\testlab\Documents\AutoRepack" Msi="1" Citrix="1" />
Machines	The <Machines> element contains multiple <VirtualMachine> elements, which identify the virtual machines that you have added to the project: <Machines> <VirtualMachine Name="Windows 7 Enterprise - 32-bit" Username="testlab" Password="Êé <VirtualMachine Name="Windows XP" Username="testlab" Password="Ê'ÂàkEƒ" Path="[</Machines>
Packages	The <Packages> element contains multiple <Package> elements, which identify the packages that you have added to the project: <Packages> <Package Compressed="0" Name="LicModInj" Product="LicModInj" Company="Your <VirtualizedPackage Enabled="0" Compressed="0" Name="LicModInj.sft" Pro <ErrorVirtualizedPackage Enabled="0" Compressed="0" Name="LicModInj.prof </Package> <Package Compressed="0" Name="MathPlot" Product="MathPlot" Company="Install <Package Compressed="0" Name="BlackBerry" Product="BlackBerry" Company="Res <VirtualizedPackage Enabled="0" Compressed="0" Name="BlackBerry.sft" Pr <VirtualizedPackage Enabled="0" Compressed="0" Name="BlackBerry.profile" </Package> </Packages> Each <Package> element can have multiple <RepackagedPackage>, <VirtualizedPackage>, and <ErrorVirtualizedPackage> elements.

Table 9-11 • Major Elements of an Automated Application Converter Project File

Element	Description

Results	The <Results> element contains multiple <Resultset> elements, each of which contains information on a conversion run:
	
	`<Results>`
	`  <Resultset Name="Log started Friday, June 11, 2010 2:56 PM (-6:00)" Generated="`
	`  <Machines>`
	`  <UseMachine Name="Windows 7 Ultimate - 32-bit" Path="[QA_ISO] Windows 7`
	`  <UseMachine Name="Windows 7 Ultimate - 64-bit" Path="[QA_ISO] Windows 7`
	`  </Machines>`
	`  <Packages>`
	`  <UsePackage Name="InstallShield MSIDiff" Path="C:\Program Files\AdminSt`
	`  <UsePackage Name="LicModInj" Path="C:\Program Files\AdminStudio\9.5\Qui`
	`  <UsePackage Name="MathPlot" Path="C:\Program Files\AdminStudio\9.5\Quic`
	`  </Packages>`
	`  <Messages>`
	`  <LogItem Id="4300" Flags="4" Message="Processing packages..." Time="201`
	`  <LogItem Id="4301" Message="Using virtual machine Windows 7 Ultimate -`
	`  <LogItem Id="4301" Message="Using virtual machine Windows 7 Ultimate -`
	`  </Messages>`
	`  <Result Name="InstallShield MSIDiff" Generated="2010-06-11T20:56:02.284" Er`
	`  <Messages>`
	`  <LogItem Id="4336" Flags="2" Message="Package InstallShield MSIDif`
	`  <LogItem Id="4345" Message="Virtualizing package..." Time="2010-06-`
	`  <LogItem Id="4347" Message="Converting to Microsoft App-V." Time="`
	`  </Messages>`
	`  </Result>`
	`  <Result Name="LicModInj" Generated="2010-06-11T20:56:02.285" Warnings="7" C`
	`  <Messages>`
	`  <LogItem Id="4336" Flags="2" Message="Package LicModInj will be dir`
	`  <LogItem Id="4345" Message="Virtualizing package..." Time="2010-06-`
	`  <LogItem Id="4347" Message="Converting to Microsoft App-V." Time="`
	`  </Messages>`
	`  </Result>`
	`  </Resultset>`
	`</Results>`
	
	• Each <Resultset> element of a <Results> element includes information on <Machines>, <Packages>, <Messages>, and multiple <Result> elements for that run.
	• Each <Result> element of a <Resultset> element contains information on the conversion run for an individual package.

Using Automated Application Converter in Evaluation Mode

When using the Automated Application Converter in Evaluation mode, you can only use one virtual machine and convert up to three packages during one repackaging/virtualization conversion run. Even if more than one virtual machine is selected, only one will be used, and even if more than three packages are selected, only three will be processed.

Managing Virtual Machines

The Automated Application Converter supports automated repackaging on virtual machines from the following platforms:

- Microsoft Hyper-V Server
- VMware ESX or ESXi Server
- VMware Workstation 6.5 or later

You can use the Virtual Machine Import Wizard to add "clean" virtual machines to the **Machines** tab of the Automated Application Converter, making them available for use during automated repackaging.

Information about managing virtual machines is presented in the following sections:

- Virtual Machine System Requirements
- Preparing Your Virtual Machines for Use With the Automated Application Converter
- VMware VIX API Requirement
- Adding Virtual Machines Using the Virtual Machine Import Wizard
- Editing Virtual Machine Properties on the Machines Tab

Virtual Machine System Requirements

Automated Application Converter performs automated repackaging on virtual machines. This section lists the virtual machine platform and virtual machine image system requirements.

- Supported Virtual Machine Platforms
- VMware Requirements
- Microsoft Hyper-V Server Requirements
- Virtual Machine Image Requirements

Supported Virtual Machine Platforms

The Automated Application Converter supports automated repackaging on virtual machines from the following platforms:

- VMware ESX/ESXi Server, Version 3.5 Update 3 or later
- VMware Workstation 6.5 or later
- Microsoft Hyper-V Server 2008 R2 or later

VMware Requirements

As described above, Automated Application Converter supports automated repackaging on VMware ESX/ESXi Server and VMware Workstation.

- VMware VIX API Requirement
- VMware ESX/ESXi Server Permission Requirements

VMware VIX API Requirement

In order for the Automated Application Converter to perform automated repackaging, it needs to communicate with the virtualization technology that you are using. If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation), you need to have the VMware VIX API installed on the same machine as the Automated Application Converter. You can do this by either installing VMware Workstation on that machine or by downloading and installing the VMware VIX API from the following location:

http://www.vmware.com/support/developer/vix-api

Note • *It is recommended that you always install the latest version of VMware VIX so that you have the most options and the latest bug fixes.*

VMware ESX/ESXi Server Permission Requirements

If you plan to use a VMware ESX/ESXi Server in conjunction with Automated Application Converter, make sure that the account that you use to log in to this server has the permissions/roles needed to automatically open a VM using VMware VIX API. The account needs to either have an administrator role assigned or, at least, have the following three roles assigned:

- All Privileges/Virtual Machine/State/Create Snapshot
- All Privileges/Virtual Machine/State/Delete Snapshot
- All Privileges/Virtual Machine/Interaction/Console Interaction

If the login account does not have these permissions/roles, Automated Application Converter will be unable to automatically boot up a virtual machine on that server.

Microsoft Hyper-V Server Requirements

As described above, Automated Application Converter supports automated repackaging on Microsoft Hyper-V Server. When preparing a Hyper-V Server for use with Automated Application Converter, make sure that the following conditions are met:

- **Configuration tools**—Verify that the Hyper-V configuration tools are installed on the Hyper-V server machine. These tools can be installed using the Microsoft Hyper-V Management Console.
- **Connection**—Verify that you can successfully connect to the Hyper-V Server from the machine where AdminStudio Automated Application Converter is installed.
- **Permissions**—Make sure that the Hyper-V Server user has the permissions required to perform operations on the Hyper-V machines.
- **Configuration settings**—Connecting to a WMI namespace on a remote computer running Windows Vista or Windows Server 2008 may require changes to configuration settings. Check the following configuration settings on the AdminStudio machine as well as on the Hyper-V Server machine:

- Windows Firewall Settings
- User Account Control (UAC) Settings
- DCOM Settings
- Common Information Model Object Manager (CIMOM) Settings

Note • *For detailed information, see Connecting to WMI Remotely at:*

http://msdn.microsoft.com/en-us/library/aa822854(VS.85).aspx

Virtual Machine Image Requirements

Automated Application Converter uses virtual machines to perform automated repackaging. These virtual machines have the following requirements:

Virtual Machine System Requirements

When creating a virtual machine image that will be hosted on one of the virtual machine platforms listed above, the recommended minimum requirements should meet those required by the applications you are trying to repackage. Since you repackage on the target deployment platform, the virtual machine image should closely resemble the target deployment environment.

Preparing Your Virtual Machines for Use With the Automated Application Converter

You need to prepare each virtual machine that you are going to use with the Automated Application Converter to perform automated repackaging by running the Virtual Machine Preparation setup and by creating a snapshot. For instructions, see Preparing Your Virtual Machines for Use With the Automated Application Converter.

Preparing Your Virtual Machines for Use With the Automated Application Converter

You need to prepare each virtual machine that you are going to use with the Automated Application Converter to perform automated repackaging by performing the following steps:

Task: ***To prepare a virtual machine:***

1. On each virtual machine you are going to use with the Automated Application Converter, run the Virtual Machine Preparation setup, as described in Running the Virtual Machine Preparation Setup.
2. At the end of the Virtual Machine Preparation setup, you will be prompted to restart the virtual machine. Restart the virtual machine and verify that you are automatically logged in and that `GuestAgent.exe` is launched:

Note • *The Guest Agent (GuestAgent.exe) is a tool that is launched on a virtual image that enables the Automated Application Converter to manipulate the virtual machine in ways that may be unsupported by its automation APIs. In particular, this enables launching and monitoring the AdminStudio Repackager in an automated fashion.*

3. Shut down the virtual machine.
4. Take a snapshot, as described in Taking a Snapshot. If your virtualization technology supports named snapshots, name the snapshot `AutoRepack_Base`, which is the default name that the Automated Application Converter will be looking for. If you assign a snapshot name other than `AutoRepack_Base`, after you add the virtual machine to the Automated Application Converter, you need to enter that snapshot name in the **Snapshot Name** field in the **Properties** window of the **Machines** tab for that virtual machine.

Running the Virtual Machine Preparation Setup

On each virtual machine that you are going to use to perform automated repackaging, you need to run the Virtual Machine Preparation setup, an application that will enable automatic login. When you install AdminStudio, you will find the Virtual Machine Preparation setup in the following location:

```
C:\Program Files\AdminStudio\10.0 SP2\Repackager\VirtualMachinePrep\VMCfg.exe
```

You need to run this application one time on all of the virtual machines that you are going to use with the Automated Application Converter.

Note • *If you do not run the Virtual Machine Preparation setup on the virtual machines you want to use, the Automated Application Converter will be unable to connect to them.*

Taking a Snapshot

After you have run the Virtual Machine Preparation setup on a virtual machine, you need to shut it down and create a snapshot. This enables the Automated Application Converter to revert the virtual image to a clean state after each repackaging run.

Note • *If you do not take a snapshot of the virtual image, the Automated Application Converter will be unable to revert the image to a clean state after completing a repackaging run. Therefore, while the first repackage on the virtual machine would be on a clean image, all subsequent repackaging runs would be run on a "dirty" virtual image.*

Links to instructions on how to create a snapshot of a virtual machine are presented in the following table:

Table 9-12 • Instructions for Taking a Snapshot of a Virtual Machine

Platform	Link
VMware Workstation	***To take a snapshot:*** 1. Open VMware Workstation. 2. On the **VM** menu, point to **Snapshot** and click **Take Snapshot**. 3. Name the snapshot `AutoRepack_Base`. 4. You can optionally add a description to record notes about the virtual machine state captured in the snapshot. 5. Click **OK**.
VMware ESX or ESXi Server	To take a snapshot on the VMware ESX or ESXi server, perform the following steps: ***To take a snapshot:*** 1. Open the VMware infrastructure client. 2. On the **Inventory** menu, point to **Virtual Machine** and **Snapshot**, and then click **Take Snapshot**. 3. Name the snapshot `AutoRepack_Base`. 4. You can optionally add a description to record notes about the virtual machine state captured in the snapshot. 5. Click **OK**. ***Note*** • *For more information, see the VMware Knowledge Base article entitled Understanding virtual machine snapshots in VMware ESX:* *http://kb.vmware.com/kb/1015180*

VMware-Specific Snapshot Configuration Option

When configuring a VMware Workstation or VMware ESX/ESXi Server image, there is a setting that controls what VMware does with a virtual machine when it is powered off. In VMware Workstation, the option is set on the **Options** tab of the **Virtual Machine Settings** dialog box:

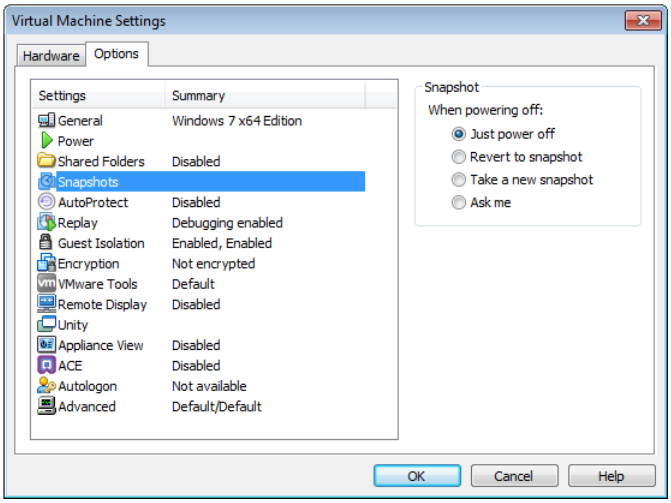


Figure 9-13: Snapshots Options on VMware Workstation Virtual Machine Settings Dialog Box

It is recommended that you set the **Snapshot** / **When powering off** option to **Just power off**. Do not select the **Ask me** option; selecting this option would block the use of this virtual image by Automated Application Converter until the prompt is dismissed by the user.

VMware VIX API Requirement

In order for the Automated Application Converter to perform automated repackaging, it needs to communicate with the virtualization technology that you are using. If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation 6.5 or later), you need to have the VMware VIX API installed on the same machine as the Automated Application Converter. You can do this by either installing VMware Workstation on that machine or by downloading and installing the VMware VIX API from the following location:

http://www.vmware.com/support/developer/vix-api

Adding Virtual Machines Using the Virtual Machine Import Wizard

You can add one or multiple virtual machines to the Automated Application Converter to use to perform automated repackaging during conversion to virtual packages.

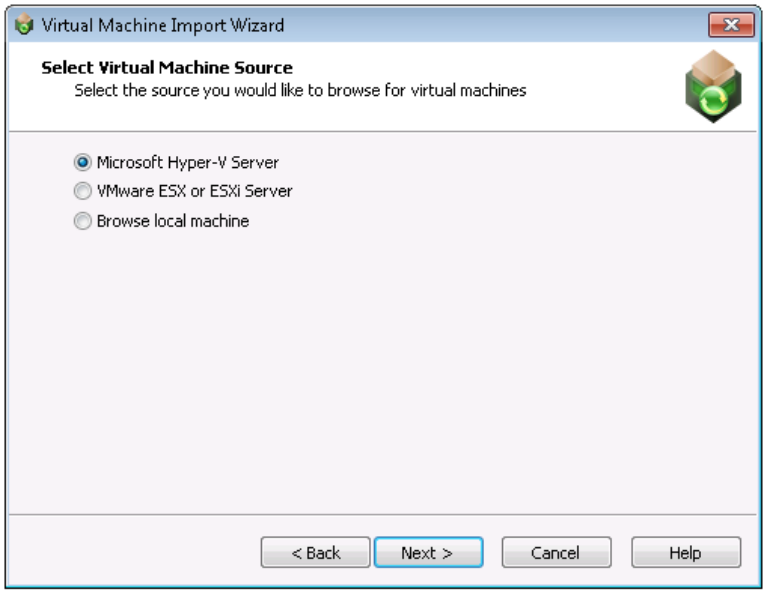
You have the option of selecting just one virtual machine to use for all repackaging, or selecting an operating system group of multiple virtual machines that can be used simultaneously to speed up the repackaging of multiple setups.

If you have specified a group of multiple virtual machines, a package in the conversion list is assigned to each virtual machine. Then, when a virtual machine finishes repackaging a package, it is reverted to its clean snapshot image, and then starts repackaging the next package in the list.

To add virtual machines to the **Machines** tab using the Virtual Machine Import Wizard, perform the following steps.

Task: ***To add virtual machines:***

1. Open the **Machines** tab of the Automated Application Converter.
2. Click **Add Machine**. The **Welcome** panel of the **Virtual Machine Import Wizard** opens.
3. Click **Next**. The **Select Virtual Machine Source** panel opens, prompting you to select the type of virtual machine that you are adding.

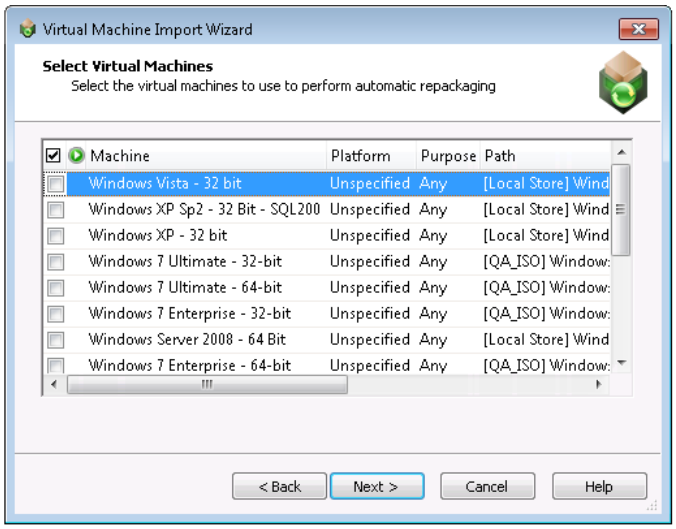


4. Select one of the following options and click **Next**.
   - **Microsoft Hyper-V Server**—Select this option to add a virtual image from a Microsoft Hyper-V Server.
   - **VMware ESX or ESXi Server**—Select this option to add a virtual image from a VMware ESX or ESXi Server.
   - **Browse local machine**—Select this option to add a virtual image from a local installation of VMware Workstation

5. Based upon your selection on the **Select Virtual Machine Source** panel, enter the following information:

Virtual Machine Source	Steps to Take
Microsoft Hyper-V Server	On the **Select Virtual Machines from a Microsoft Hyper-V Server** panel, enter the following information: • **Server Name**—Enter the server name of the Microsoft Hyper-V Server that you want to connect to. • **Authentication**—Select **Windows Authentication** if you want to use the credentials of the logged in user to login to the Hyper-V Server. Select **Server Authentication** if you want to connect to the Hyper-V Server using the specified **User name** and **Password**.
VMware ESX or ESXi Server	On the **Select Virtual Machines from VMware ESX or ESXi Server** panel, enter the following information: • **Server Name**—Enter the name of the VMware ESX or ESXi server. • **User name**—Enter the login ID for the VMware ESX or ESXi server. • **Password**—Enter the password for the VMware ESX or ESXi server.
Browse local machine	On the **Select Virtual Machines** panel, do one of the following: ***To add an individual virtual machine:*** 1. Click **Browse Files**. The **Select Virtual Machine Image File** dialog box opens. 2. Select the virtual machine image you want to add to the project and click **Open**. ***To add all of the virtual machines in a specific directory:*** 1. Click **Browse Folders**. The **Browse for Folder** dialog box opens. 2. Select a directory that contains the virtual machine images that you want to add to your project and click **OK**.

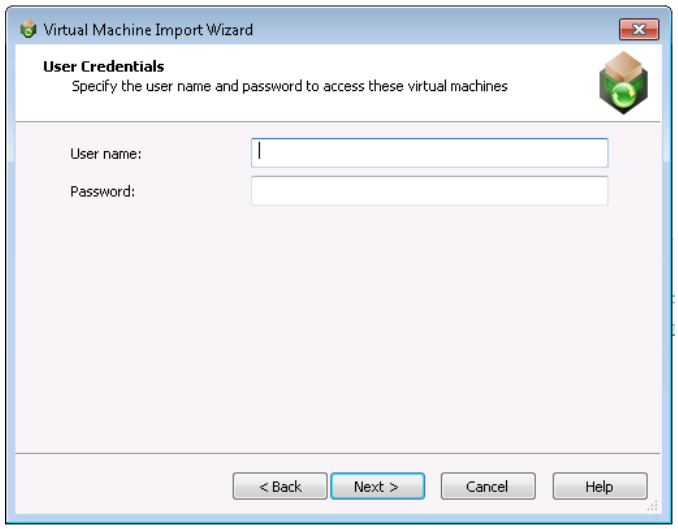
When you have finished this step, the virtual machines will be listed (but not selected) on the **Select Virtual Machines** panel.



6. On the **Select Virtual Machines** panel, select the virtual machine images that you want to use to perform automated repackaging.
7. For each selected image, click in the **Platform** column and identify its platform.
8. By default, virtual machines that you add to the **Machines** tab will be available for use for both automated repackaging of packages and for testing packages. However, if you want to specify that a virtual machine should be used for only repackaging or for only testing, click in the **Purpose** column of that virtual machine and select one of the following options:
   - **Repackaging**—Virtual machine will only be used to perform automated repackaging.
   - **Testing**—Virtual machine will only be used to test packages. You test a package by selecting it on the **Packages** tab and selecting **Launch Package for Testing** from the context menu. You will then be prompted to install and run that package on a virtual machine.
   - **Any**—Make this virtual machine available for use during both automated repackaging and package testing.

Note • *The **Launch Package for Testing** functionality will primarily be useful to test converted packages. However, if a problem occurs during conversion, it is also possible to use this function to install and launch the source package for testing.*

9. Click **Next**. The **User Credentials** panel opens, prompting you to specify the login credentials to use to access the selected virtual machines.



10. Enter the user credentials and click **Next**. The Virtual Machine Import Wizard Complete panel opens.
11. Click **Finish** to close the wizard and add the selected virtual machines to your project.

Editing Virtual Machine Properties on the Machines Tab

By default, the list of machines on the **Machines** tab lists the **Machine**, **Platform**, **Purpose**, and **Path** columns. Additional properties can be viewed and edited in the **Properties** window.

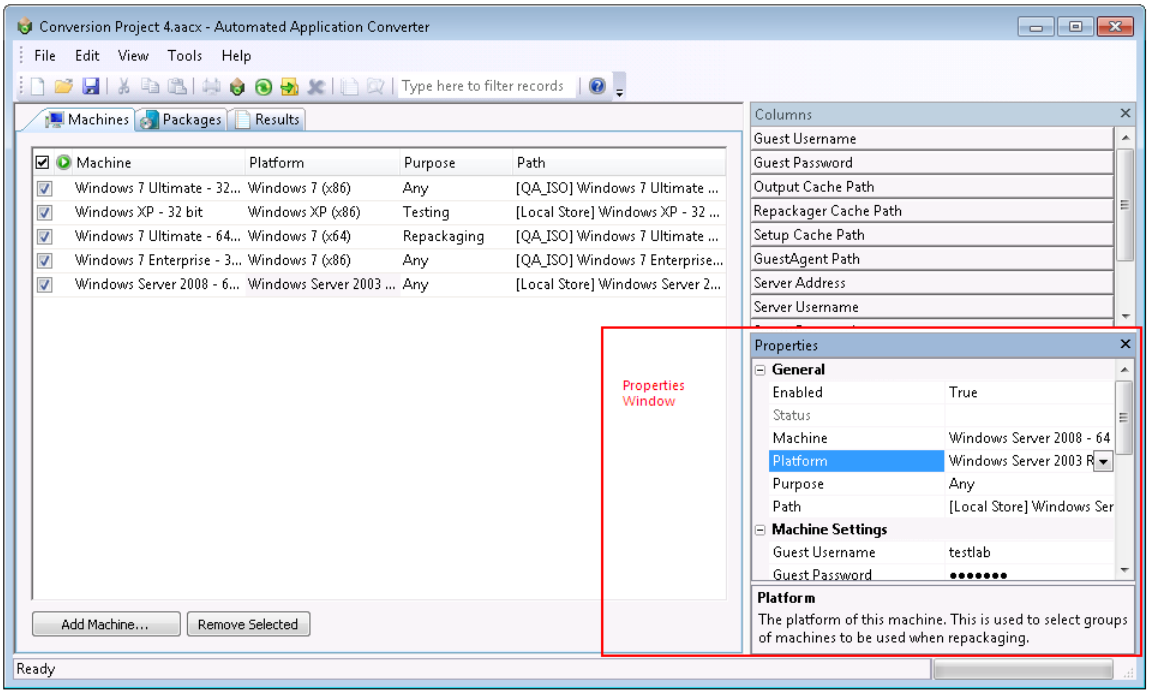


Figure 9-14: Properties Window on the Machines Tab

To edit the properties of a virtual machine, perform the following steps:

Task: ***To edit a virtual machine's properties:***

1. Open the **Machines** tab.
2. Select a virtual machine in the list.
3. Click in the list or in the Properties window to edit the following properties:

Property	Description
Machine	Name of the virtual machine image.
Platform	Field that identifies the operating system platform of the virtual machine. When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in this field on the **Machines** tab and making a selection from the list. When you perform a conversion run, you are given the opportunity (on the **Automated Repackaging on Virtual Machines** panel) to either select a specific platform to use for the repackaging of the selected packages, or to select **Any Platform**, meaning that all of the selected virtual machines will be used for repackaging.
Purpose	By default, virtual machines that you add to the **Packages** tab will be available for use for both automated repackaging of packages and for testing packages. However, if you want to specify that a virtual machine should be used for only repackaging or for only testing, click in the **Purpose** column of that virtual machine and select one of the following options: • **Repackaging**—Virtual machine will only be used to perform automated repackaging. • **Testing**—Virtual machine will only be used to test packages. You test a package by selecting it on the **Packages** tab and selecting **Launch Package for Testing** from the context menu. You will then be prompted to install and run that package on a virtual machine. • **Any**—Make this virtual machine available for use during both automated repackaging and package testing. This is the default value. ***Important*** *• If the* ***Purpose*** *column is not listed in the* ***Machines*** *list, you can edit the* ***Purpose*** *value in the Properties window.* ***Note*** *• The* ***Launch Package for Testing*** *functionality will primarily be useful to test converted packages. However, if a problem occurs during conversion, it is also possible to use this function to install and launch the source package for testing.*

Property	Description
Path	Path on the server or file system to the virtual machine image file.
Guest Username	The user name to use to login to this virtual machine.
Guest Password	The password to use to login to this virtual machine.
Output Cache Path	Specify the location for the repackaged output on the virtual machine. By default, this value is `C:\AutoRepack`.
Repackager Cache Path	Specify the location where Repackager will be installed on the virtual machine. By default, this value is `C:\Repackager`.
Setup Cache Path	Specify the location where the package will be copied to on the virtual machine. By default, this value is `C:\AppSetup`.
GuestAgent Path	Specify the location where the `GuestAgent.exe` file will be installed on the virtual machine. By default, this value is `C:\GuestAgent.exe`.
Server Address	The address of the virtual machine server on which this virtual machine is found. This may be a host name or a URL.
Server Username	The user name of the account used to access the virtual machine server.
Server Password	The password of the account used to access the virtual machine server.
Snapshot Name	Name of the snapshot to revert to before starting an automated repackaging session. This is only used if the virtualization technology supports named snapshots. If this value is not specified, but named snapshots are supported on the virtualization technology, the default name of `AutoRepack_Base` will be used.
Virtualization Technology	The virtualization technology powering this virtual machine.
Add Machine	Click to launch the Virtual Machine Import Wizard, which you can use to add virtual machines to the **Machines** tab.
Remove Selected	Click to remove the selected virtual machine from this list. ***Note*** • *A virtual machine is selected for removal when you click on it and it becomes highlighted, not by selecting the virtual machine's check box. Use the* ***Ctrl*** *key to select multiple machines.*

Connecting to Active Virtual Machines

When the Automated Application Converter is connected to a virtual machine and it is performing repackaging, you can use Remote Desktop to open that virtual machine directly from the Automated Application Converter interface to check on the progress of the repackaging run.

Task: ***To open a virtual machine from the Automated Application Converter interface:***

1. Add virtual machines to your project file, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
2. Begin a conversion as described in Using the Application Conversion Project Wizard or Performing a Conversion Using the Application Conversion Wizard.
3. Open the **Machines** tab or the **Packages** tab.
   - On the **Machines** tab, select a machine that is currently performing repackaging. The machine will have a status of Running (), which means that the Automated Application Converter has connected to the virtual machine and the `GuestAgent.exe` is running.
   - On the **Packages** tab, select a package that is currently being repackaged on a virtual machine. The package will have a status of Running ().
4. On the context menu, select **Connect to Machine**. The virtual machine opens in a Remote Desktop window.

Tip • *If you have selected a package that has a status of Running but which does not require repackaging, the* ***Connect to Machine*** *selection will be disabled.*

Managing Packages to Convert

You can use the **Package Import Wizard** to add packages to your Automated Application Converter project for conversion.

You can also modify package properties on the **Packages** tab.

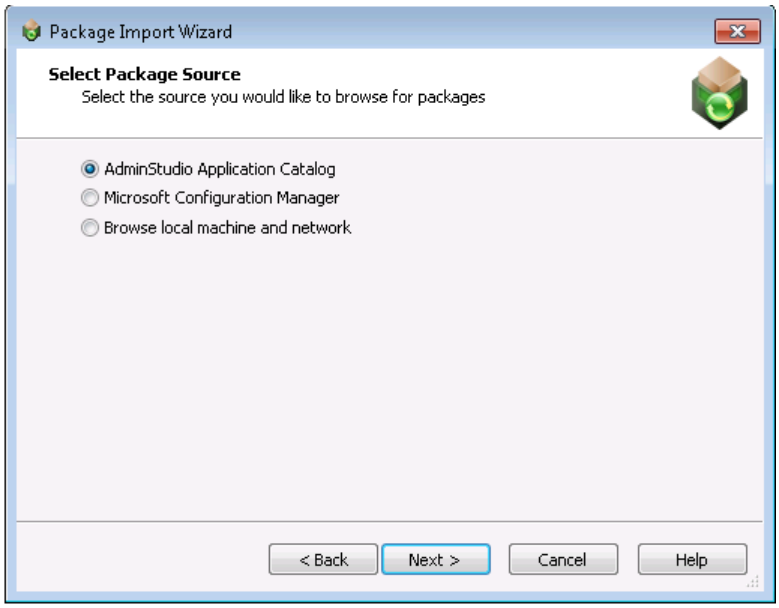
- Selecting Packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server
- Selecting Packages from a Local Machine or Network
- Editing Package Properties on the Packages Tab

Selecting Packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server

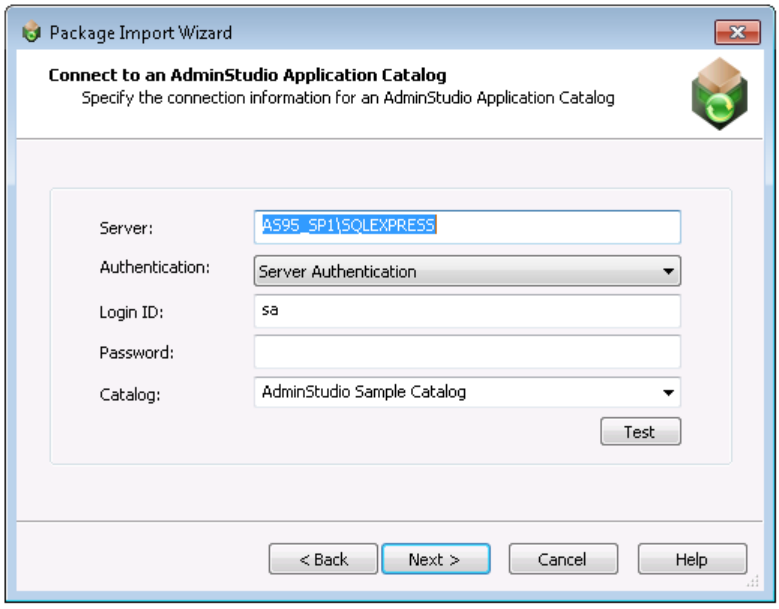
To select packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server to convert to virtual applications, perform the following steps:

Task: ***To select packages from an AdminStudio Application Catalog:***

1. Launch the Automated Application Converter.
2. Add one or more virtual machines, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
3. Open the **Packages** tab.
4. Click **Add Packages**. The **Package Import Wizard** opens.
5. Click **Next**. The **Select Package Source** panel opens.

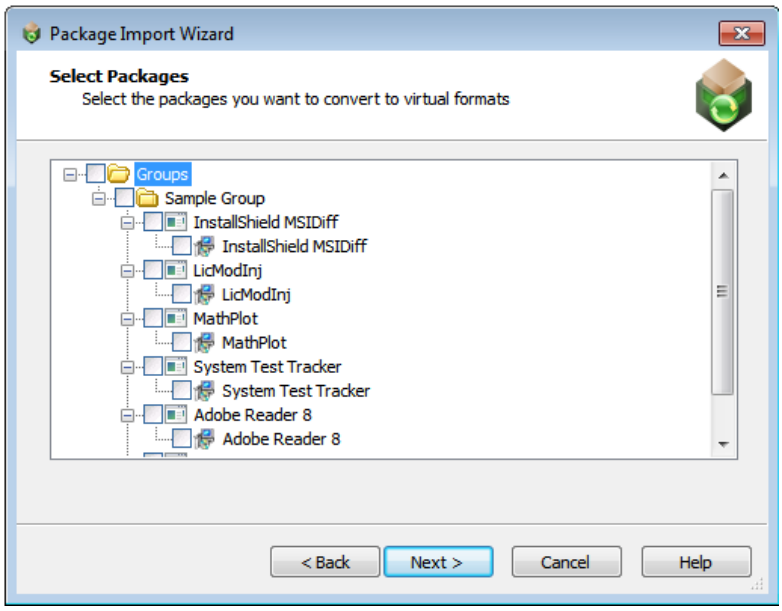


6. Select **AdminStudio Application Catalog** or **Microsoft Configuration Manager** and click **Next**. The **Connect to an AdminStudio Application Catalog** or **Connect to a Microsoft Configuration Manager Server** panel opens.

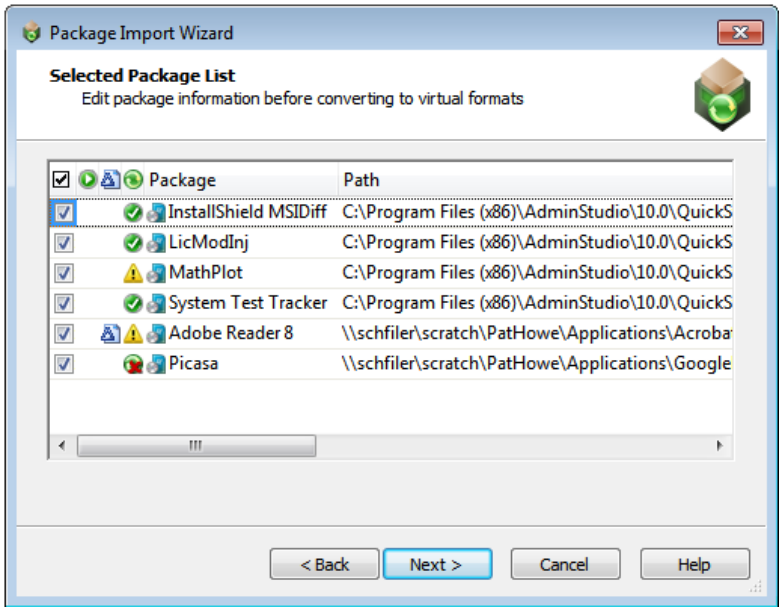


7. In the **Server** field, enter the name of the Server that you want to connect to.
8. From the **Authentication** list, select one of the following options:
   - **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Application Catalog/Microsoft Configuration Manager Server.
   - **Server Authentication**—Choose this option if you want to use server login identification to log into this Application Catalog/Microsoft Configuration Manager Server. Then enter the appropriate **Login ID**/**User name** and **Password**.
9. If you are connecting to an AdminStudio Application Catalog, enter the name of the existing AdminStudio Application Catalog database that you want to connect to in the **Catalog** field.

10. Click **Next**. The **Select Packages** panel opens, listing all of the packages found in the Application Catalog/ Microsoft Configuration Manager Server, but with none of them selected.



11. Select the packages that you want to add to this project and click **Next**. The **Selected Package List** panel opens.



An icon in the Virtualization Readiness column identifies whether the package requires repackaging prior to conversion to a virtual application:

Icon	Meaning	Description
	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required.* *An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.
	Virtualization not supported	Automated Application Converter has determined that virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready to Virtualize** or **Requires Repackaging**.* ***Note*** • *For more information, see Virtualization Suitability Issues.*
	Virtualization not recommended	Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues: • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology. S ***Note*** • *For more information, see Virtualization Suitability Issues.*
	Unknown	The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.

Note • *If you want to override this setting, click in the Virtualization Readiness column and make a selection from the list.*

12. If a transform file (`.mst`) is located in the same directory as the selected Windows Installer package, one of the following icons is listed in the Transform column:

    - One transform is being added with this package.
    - Multiple transforms are being added with this package.

    If multiple transforms are associated with this package, you should click the browse button in the **Transform** column to open the **MST** dialog box and specify which transform files you want to add and the order that you want the transforms applied. For more information, see MST Dialog Box.

Note • You can also edit the listed transform files by clicking the browse button in the **Transform** field in the **Properties** window of the **Packages** tab after you have added the packages.

13. Make sure that the packages that you want to convert are selected and click **Next**. The **Package Import Wizard Complete** panel opens.
14. Click **Finish** to close the wizard and add the selected packages to your project.

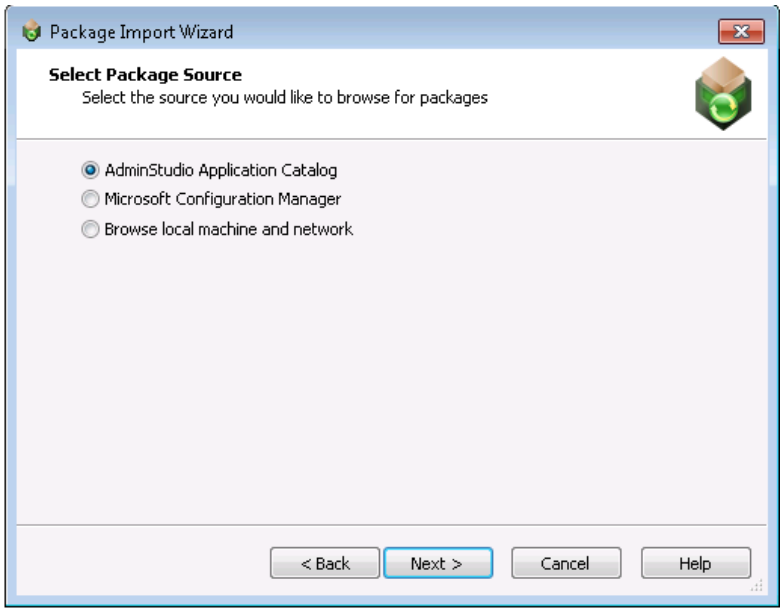
Selecting Packages from a Local Machine or Network

To select packages from a a local machine or network to convert to virtual applications, perform the following steps:

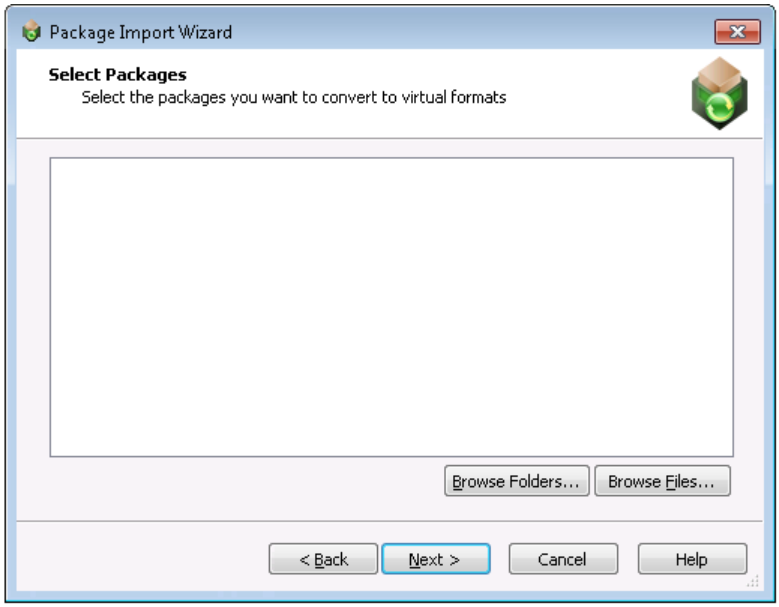
Task: ***To select packages from a local machine or network:***

1. Launch the Automated Application Converter.
2. Add one or more virtual machines, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
3. Open the **Packages** tab.
4. Click **Add Packages**. The **Package Import Wizard** opens.

5. Click **Next**. The **Select Package Source** panel opens.



6. Select **Browse local machine and network** and click **Next**. The **Select Packages** panel opens with no packages listed.



7. If you want to select one package to add, perform the following steps:

   a. Click **Browse Files**. The **Select Package Installation File** dialog box opens, prompting you to select the package you want to convert.

   b. Select the installation file (`.msi or .exe`) or installation script (`*.vbs`, `*.bat`, `*.cmd`, or `*.ps1`) you want to convert and click **Open**.The Automated Application Converter adds the selected package to the list on the **Select Packages** panel.

   Note • *You can use installation scripts to run more complex installation scenarios.*

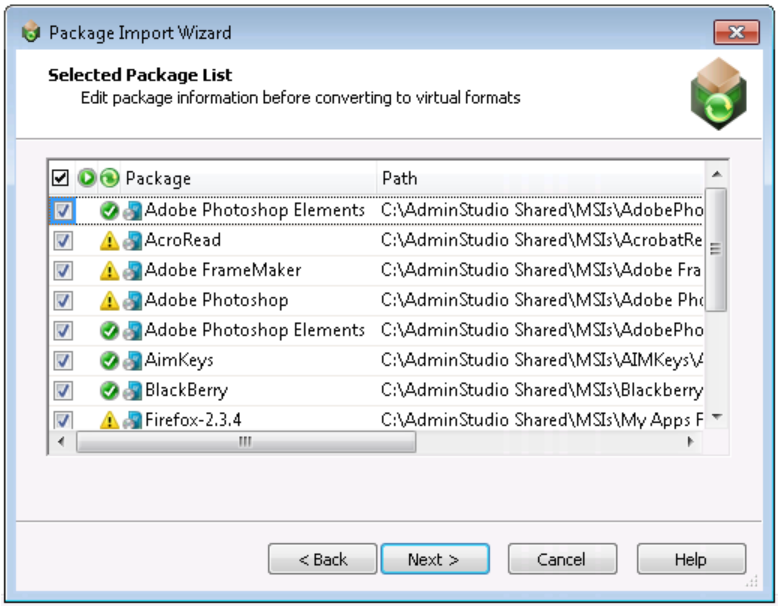
   If you want to select a directory of packages to add, perform the following steps:

   a. Click **Browse Folders.** The **Browse for Folder** dialog box opens, prompting you to select the directory containing the packages you want to convert.

   b. Select the directory that contains the installation files (`.msi or .exe`) and/or installation scripts (`*.vbs`, `*.bat`, `*.cmd`, or `*.ps1`) you want to convert and click **Open**. The Automated Application Converter searches the selected directory and its subdirectories to locate the installation files and/or scripts and adds them to the list on the **Select Packages** panel.

   Important • *The Automated Application Converter uses specific rules to determine which packages in the selected directory and its subdirectories would be added to the list on the* ***Select Packages*** *panel, and which of those files are automatically selected. See Automated Application Converter's Selection Rules When Adding Packages from a Directory for more information.*

8. On the **Select Packages** panel, click **Next**. The **Selected Package List** panel opens.



9. On the **Selected Package List** panel, an icon in the Virtualization Readiness column identifies whether the package requires repackaging prior to conversion to a virtual application:

Icon	Meaning	Description
	Ready	Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required.* *An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*
	Requires repackaging	Package must be repackaged before it can be successfully virtualized.
	Virtualization not supported	Automated Application Converter has determined that virtualization is not supported due to one of the following issues: • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready to Virtualize** or **Requires Repackaging**.* ***Note*** • *For more information, see Virtualization Suitability Issues.*
	Virtualization not recommended	Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues: • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology. ***Note*** • *For more information, see Virtualization Suitability Issues.*
	Unknown	The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.

Note • *If you want to override this setting, click in the Virtualization Readiness column and make a selection from the list.*

10. Make sure that the packages that you want to convert are selected and click **Next**. The **Package Import Wizard Complete** panel opens.
11. Click **Finish** to close the wizard and add the selected packages to the project.

Editing Package Properties on the Packages Tab

By default, the list of packages on the **Packages** tab lists the **Package**, **Path**, and **Command Line** properties. Additional properties can be viewed and edited in the **Properties** window.

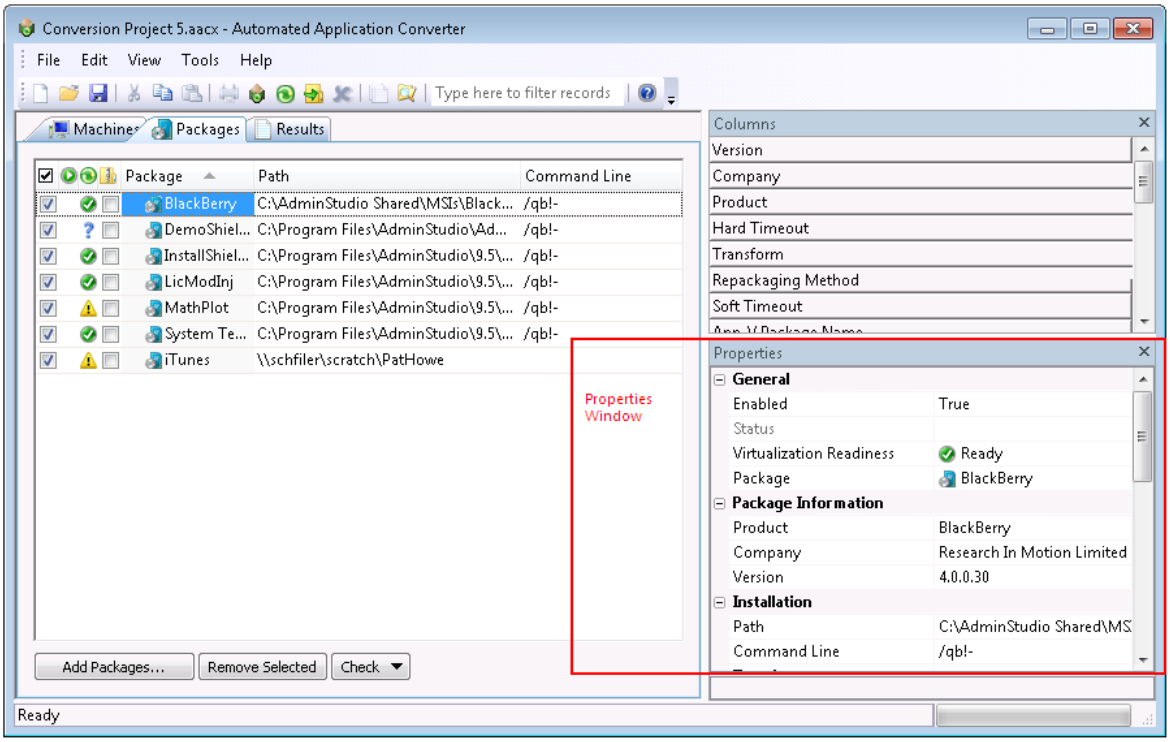


Figure 9-15: Properties Window on the Packages Tab

To edit the properties of a package, perform the following steps:

Task: ***To edit package properties:***

1. Open the **Packages** tab.
2. Click in the list or in the Properties window to edit the following properties:

Property	Description
Virtualization Readiness	When you add a package to the **Packages** tab, the Automated Application Converter does a quick check to identify that package's virtualization readiness status and assigns it one of the following icons: • **Ready** —Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined. • **Requires repackaging** —Package must be repackaged before it can be successfully virtualized. • **Virtualization not supported** —Automated Application Converter has determined that virtualization is not supported. • **Virtualization not recommended** —Automated Application Converter has determined that this package is not recommended for virtualization. • **Unknown** —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging. ***Note*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready** or **Requires repackaging**.* ***Note*** • *You can click on the icon in this column to override the Virtualization Readiness status that was automatically assigned to this package by the Automated Application Converter.*
Packages	Lists the name of the package. This is used as part of the output path for repackaged or virtualized results. It is also used in reports to refer to this package. If the package has been repackaged or converted to a virtual application, those output files are also listed below the source file in a tree structure, with an icon identifying the file type. See Icons Used on the Packages Tab.

Property	Description
Path	Lists the location from where the package was selected locally or from where it was originally imported into the AdminStudio Application Catalog or Microsoft Configuration Manager Server. ***Note*** • *It is recommended that you use UNC path when importing packages into the Application Catalog or publishing packages to Microsoft Configuration Manager Server.* ***Note*** • *If you are adding packages from an AdminStudio Application Catalog or a Microsoft Configuration Manager Server installed on a machine other than the machine where the Automated Application Converter is installed, make sure that the package source path listed here is accessible to the Automated Application Converter machine.*
Transform	This field can contain a semicolon-delimited list of transforms used to modify or install a Windows Installer package silently. To add a transform to this list, click the Browse button and select a transform (`.mst`) file, or enter the transform name and location in the box.
Command Line	Editable field that lists the command line parameters that will be used to run this installation silently during repackaging.
Product	Name of the application as provided by the company who manufactured it.
Company	Name of the company who manufactured this application.
Version	Version of the application.
Compressed	Indicates the compressed status of the package: • **False**—Indicates that the source `.msi` or `.exe` file is uncompressed. If this package is repackaged, the Automated Application Converter will copy all of the files in the same folder as the installation file to the virtual machine. • **True**—Indicates that the source `.msi` or `.exe` file is compressed. If this package is repackaged, the Automated Application Converter will copy only this single installation file to the virtual machine.
Repackaging Method	Indicates the repackaging method that will be used to repackage this package: • **Installation monitoring**—Repackager monitors system changes as a package is installed, and that data is converted into a Windows Installer package. • **Single-step snapshot**—Repackager first takes an initial system snapshot, then runs the installation, and then takes a second snapshot to create the script file that can be converted into a Windows Installer package.

Property	Description
Soft Timeout	Number of minutes allotted for the package to install before the user would be notified. After this time period elapses, the user will be notified, just in case there are pending dialogs for the user to dismiss or if some other user interaction is required. The default value is 20.
Hard Timeout	Number of minutes allotted for the package to install before it is considered a failure. If this time period elapses, the Automated Application Converter would consider the installation a failure and would move to the next package. The default value is 40.
Virtualization Technology	Indicates the virtualization format of the virtual package. Options are: • Microsoft App-V • Citrix XenApp • VMware ThinApp

3. Save the project.

About Repackaging Windows Installer Packages

As a general rule, Windows Installer setups should not be repackaged. Instead, they should either be edited in InstallShield Editor, or, as Microsoft recommends, by creating a transform.

However, some IT organizations may elect to repackage Windows Installer packages in order to simplify them, which should make them more reliable and less likely to violate the organization's and Microsoft's recommended best practices. You can use the Automated Application Converter to automatically repackage a group of Windows Installer packages by selecting the **Windows Installer Packages (*.msi)** option on the **Select Output Formats** panel of the Application Conversion Project Wizard and the Application Conversion Wizard.

If you choose to repackage a Windows Installer package, you need to keep in mind that you may no longer be able to:

- Directly deploy vendor-provided patches for this package, OR
- Use any vendor-provided automatic updating service for this package.

Therefore, you should only consider repackaging a Windows Installer package if your IT staff is also willing to invest resources into periodically repackaging that application's vendor patches into an updated Windows Installer package.

Note • *Tightly-controlled organizations probably would not want to have automatically-updating software, so the inability to use an automatic updating service may not be of concern to them.*

Using the Application Conversion Wizard to Perform Automated Package Conversion

You can use the Application Conversion Wizard to perform a conversion run using the selected packages and virtual machines. You can also view conversion run log report information.

- Performing a Conversion Using the Application Conversion Wizard
- Viewing Conversion Results

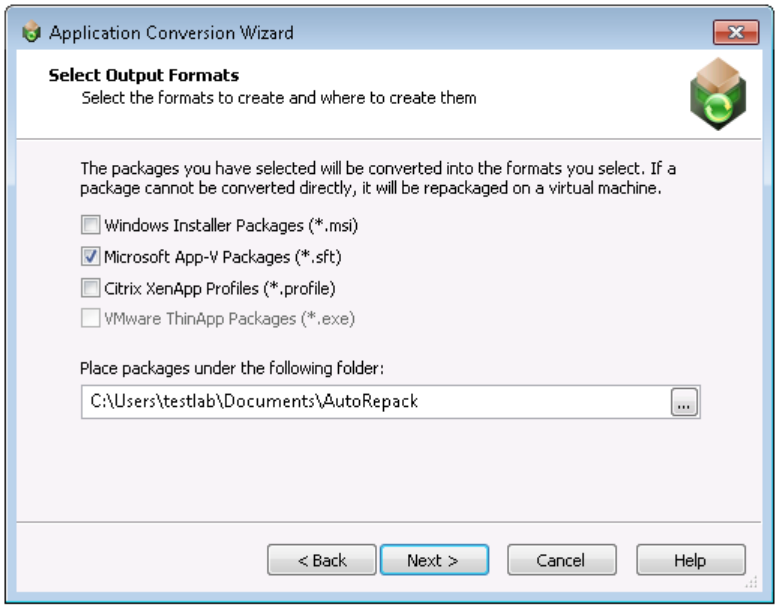
Performing a Conversion Using the Application Conversion Wizard

To use the Application Conversion Wizard to perform a conversion run using the selected packages and virtual machines, perform the following steps.

Task: ***To perform a conversion using the Application Conversion Wizard:***

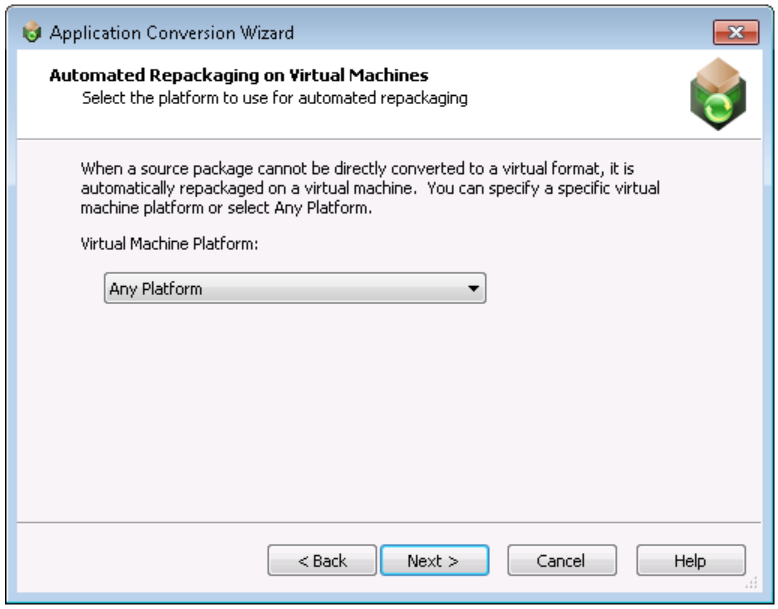
1. Select virtual machines to use for automated repackaging, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard.
2. Select packages to convert to virtual applications, as described in Selecting Packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server or Selecting Packages from a Local Machine or Network.
3. On the **Tools** menu, select **Application Conversion Wizard**. The **Application Conversion Wizard Welcome** panel opens.

4. Click **Next**. The **Select Output Formats** panel opens.



5. Select one or more of the following output formats:
   - Windows Installer Packages (*.msi)
   - Microsoft App-V Packages (*.sft)
   - Citrix XenApp Profiles (*.profile)
   - VMware ThinApp Packages (*.exe)
6. In the **Place packages under the following folder** field, specify the directory where you want to save the output packages.

7. Click **Next**. The **Automated Repackaging on Virtual Machines** panel opens.



8. From the **Virtual Machine Platform** list, select one of the following:

   - **Any Platform**—The Automated Application Converter will use any of the virtual machines that you have selected on the **Machines** tab to perform automated repackaging, regardless of platform.
   - **OS Platform**—If you select a specific operating system, the Automated Application Converter will use only those virtual machines that you have selected on the **Machines** tab that are of the selected operating system to perform automated repackaging.

   Important • *When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in the **Platform** field on the **Machines** tab and making a selection from the list.*

9. Click **Next**. The **Application Conversion Wizard Complete** panel opens.
10. Click **Finish** to close the wizard and begin converting the selected packages using the selected virtual machines.

Note • *If you have selected multiple virtual machines, the Automated Application Converter will attempt to connect to the first virtual machine in the list. If it successfully connects, conversion will proceed on that machine. If it fails to connect, it will move on to the next machine in the list.*

Viewing Conversion Results

To view the conversion results on the **Results** tab and in the AdminStudio Automated Application Converter Log report, perform the following steps:

Task: ***To view conversion results:***

1. Perform conversion as described in Using the Application Conversion Project Wizard or Performing a Conversion Using the Application Conversion Wizard.
2. Open the **Results** tab.
3. For each listed package, view the information in the **Errors**, **Warnings**, and Results Icons ( ) columns, as described in Results Tab.
4. Select the top level node of a conversion run log (such as `Log started Monday, June 21, 2010...`).
5. Do one of the following:
   - Click the **Results** button on the toolbar.
   - Select **View Report** from the context menu.
   - Select **View Report** on the **Tools** menu.
   - Press Ctrl+R.

   The AdminStudio Automated Application Converter Log report opens. See AdminStudio Automated Application Converter Log Report for more information.
6. Open the **Packages** tab.
7. In the tree, locate one of the source packages that you converted and click the plus sign to expand the listing. The converted packages in the formats you selected are listed.
8. Continue with the steps in Launching Packages for Testing and Publishing Converted Packages.

Launching Packages for Testing

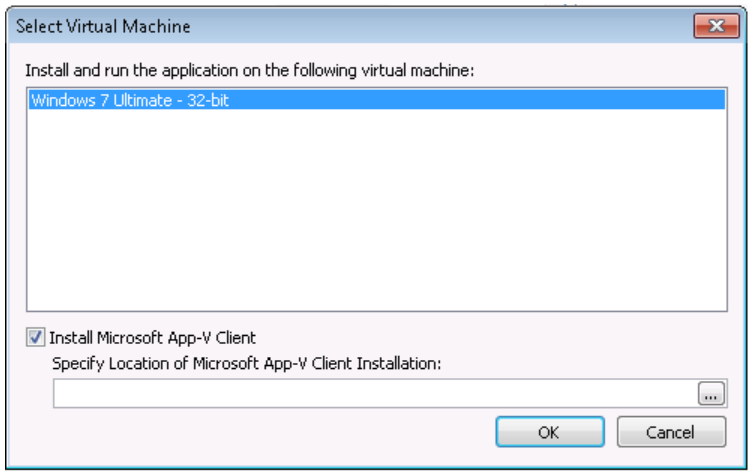
You can quickly launch a package for testing by selecting a package on the **Packages** tab and selecting **Launch Package for Testing** from the context menu. You can test any of the following package types:

- **Virtual package**—A virtual package that was converted from a Windows Installer package using the Automated Application Converter.
- **Repackaged MSI package**—A repackaged Windows Installer package that was converted from a source Windows Installer package using the Automated Application Converter.
- **Source package**—A source Windows Installer package that you have added to the **Packages** tab.

To launch packages for testing, perform the following steps:

Task: ***To launch packages for testing:***

1. Perform package conversion as described in Using the Application Conversion Project Wizard or Performing a Conversion Using the Application Conversion Wizard. Converted packages are listed under their source package.
2. On the **Packages** tab, select a package and select **Launch Package for Testing** from the context menu. The **Select Virtual Machine** dialog box opens, prompting you to select the virtual machine that you want to use to test the selected package.



3. Select a virtual machine from the list.
4. If you are testing an App-V package and the Microsoft App-V client is not yet installed on the selected virtual image, select the **Install Microsoft App-V Client** option and browse to the location of the Microsoft App-V Client installation file. Make sure that it is in a location that is accessible to the virtual machine.
5. Click **OK**. The Automated Application Converter will connect to the selected virtual machine, install the App-V client (if selected), and launch the selected package.

Testing App-V Applications Using the App-V Application Launcher

The App-V Application Launcher is a convenient testing tool that makes it possible for you to reliably and accurately test your App-V applications on your local machine or any other system that has the App-V client installed before moving it to the App-V server.

App-V Application Launcher Utility Location

When an App-V application is built, the App-V Application Launcher utility (`AppVLauncher.exe`) is placed in the same folder as the App-V application (the same directory that contains the `.sft` and `.osd` files).

Figure 9-16: App-V Application Launcher in the App-V Application Output Directory

Opening the App-V Application Launcher Utility

To open the App-V Application Launcher, double-click on the App-V Application Launcher utility (`AppVLauncher.exe`). The App-V Application Launcher will attempt to launch that application. If there are multiple shortcuts (`.osd` files) in this App-V application, the **Launch App-V Application** dialog box opens, where you are prompted to select the shortcut you want to launch from a list of all of the shortcuts.

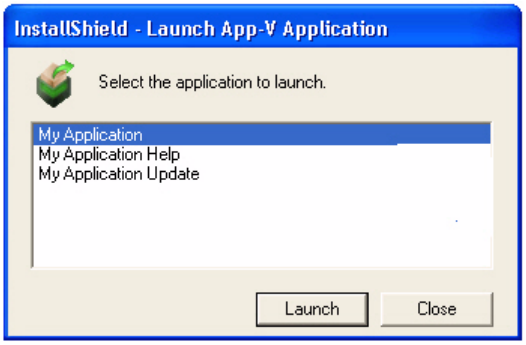


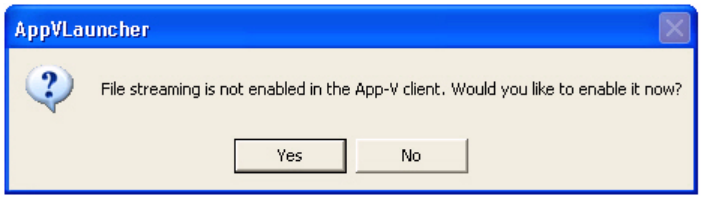
Figure 9-17: Launch App-V Application Dialog Box

Requirements for Using the App-V Application Launcher

There are two requirements for using the App-V Application Launcher:

- **Microsoft Virtualization client must be installed**—In order to test an App-V application using the App-V Application Launcher, you must have the Microsoft Virtualization Client installed on the machine.

- **File streaming must be enabled**—If you attempt to launch the App-V Application Launcher without having App-V file streaming enabled, an error message will open prompting you to enable the file streaming option. Click **Yes** to enable it.



Controlling Whether to Include the App-V Application Launcher in Output

If you do not want to include the App-V Application Launcher with each App-V application that you build, clear the **Create utility to test launch App-V applications as part of building App-V applications** check box on the **Virtual Packages** tab of the AdminStudio **Options** dialog box.

Publishing Converted Packages

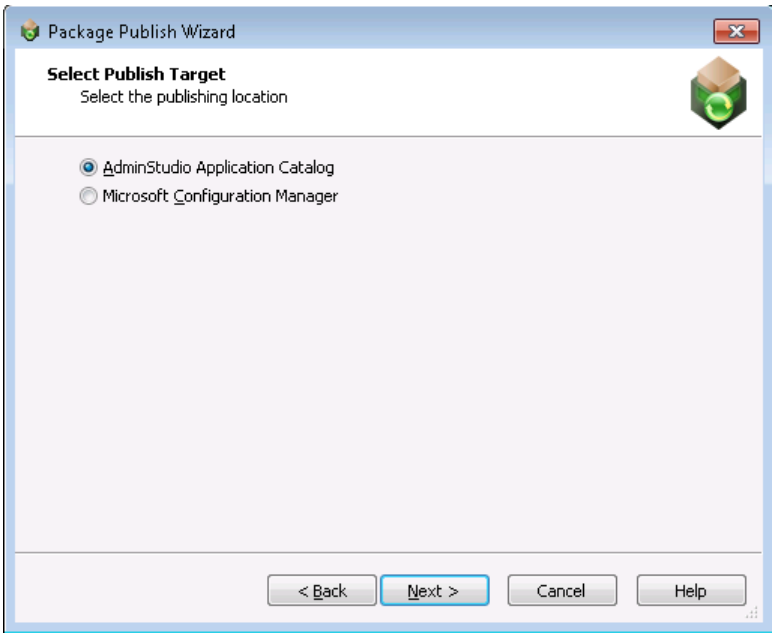
After converting packages to virtual packages or repackaged Windows Installer packages, you can publish them to a Microsoft Configuration Manager Server or AdminStudio Application Catalog using the **Package Publish Wizard**.

Task: ***To publish converted packages:***

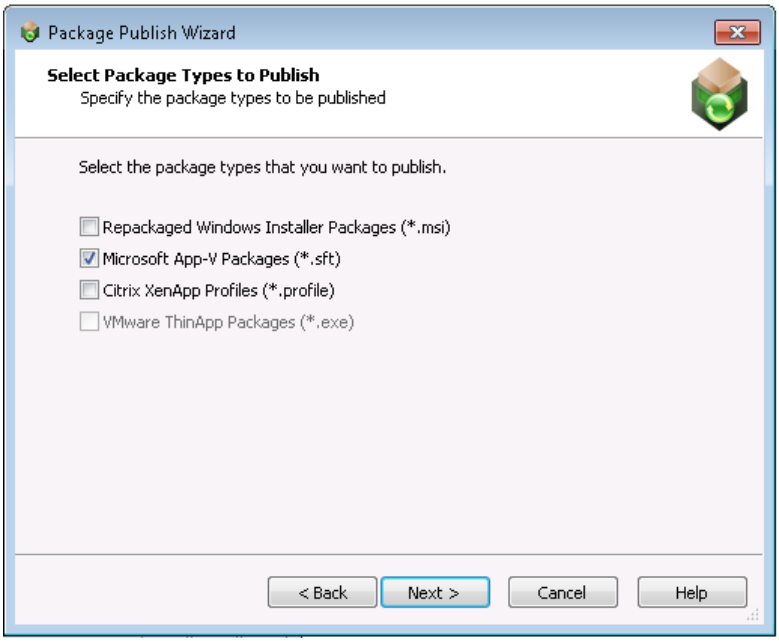
1. On the **Packages** tab, select the packages that you want to publish.
2. On the **Tools** menu, select **Package Publish Wizard**. The **Package Publish Wizard Welcome** panel opens.

   Note • *You can also open the **Package Publish Wizard** by clicking the Package Publish Wizard icon in the toolbar or by selecting a package on the **Packages** tab and selecting **Package Publish Wizard** from the context menu.*

3. Click **Next**. The **Select Publish Target** panel opens.



4. Select **AdminStudio Application Catalog** or **Microsoft Configuration Manager** and click **Next**. The **Connect to an AdminStudio Application Catalog** panel or the **Connect to a Microsoft Configuration Manager Server** panel opens.
5. Enter the login credentials, as described in Connect to an AdminStudio Application Catalog or Connect to a Microsoft Configuration Manager Server and click **Next**. One of the following panels open
   - **AdminStudio Application Catalog**—The Select Group panel opens, prompting you to specify the Application Catalog group where you want to publish the selected packages and whether to add the published files to the Software Repository.
   - **Microsoft Configuration Manager Server**—The Select Destination Folder panel opens, prompting you to select a location that the Microsoft Configuration Manager Server has access to where you want to publish the selected packages.
6. Specify the location where you want to publish the packages and click **Next**. The **Select Package Types to Publish** panel opens, prompting you to select the package formats that you want to publish (**Repackaged MSI**, **Microsoft App-V Packages**, **Citrix XenApp Profiles**, and **VMware ThinApp Packages**).



7. Select the package formats that you want to publish and click **Next**. The **Package Publish Wizard Complete** panel opens.
8. Click **Finish** to close the wizard and publish the selected packages.

Setting Automated Application Converter Options

On the **Project Options** dialog box, which is opened by selecting **Options** on the **Tools** menu, you can specify project-wide default options.

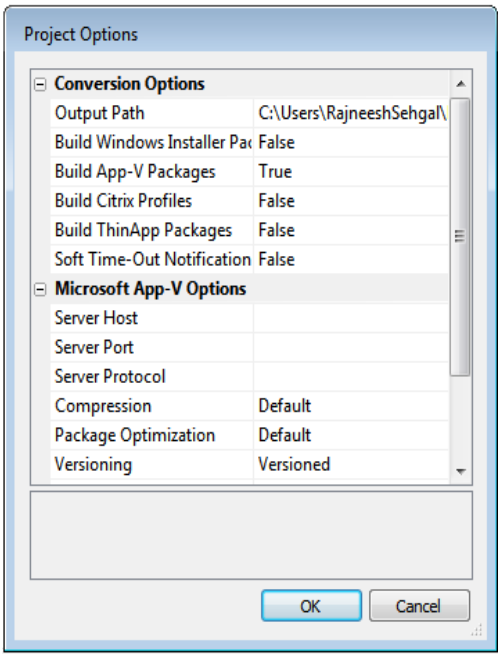


Figure 9-18: Project Options Dialog Box

To set project-wide default options, perform the following steps:

Task: ***To set project-wide default options.***

1. On the **Tools** menu, select **Options**. The **Project Options** dialog box opens.
2. Set project options as described in Project Options Dialog Box.
3. Click **OK** to save your selections.

Tip • *You can also specify global default settings for any App-V virtual setting in the* ***ISVirtualPackage*** *table by editing the* `settings.xml` *file. For more information, see Specifying Global Default Virtual Conversion Settings.*

Capturing Virtualization Context

Sometimes it is necessary to repackage a Windows Installer package before you can successfully virtualize it (as described in Virtualization Conversion Error Messages).

When some Windows Installer packages are repackaged, some of their data (such as files or registry entries) are excluded according to the normal Repackager exclusion settings. For example, files destined for the `\Windows\Installer` folder are typically excluded. However, this type of information is occasionally necessary in order to successfully convert a Windows Installer package to a virtual package.

To address this issue, when Repackager builds a Windows Installer package, it now produces two `.msi` files: `packagename.msi` and `packagename.context.msi`.

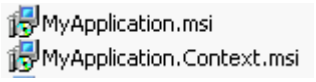


Figure 9-19: Repackaged Output: application.msi and application.context.msi

Important • *If you are not converting a package to a virtual package, you can ignore its `.context.msi` file.*

Note • *Context data is not displayed in the Repackager interface when viewing captured Files/Registry details.*

How is the Context Data Configured

Context data is configured in a settings file in the `AdminStudio Shared` folder called `isrepackager.context.ini`. It is identical in syntax to the familiar `isrepackager.ini` file which is used for exclusion settings. Data that matches the settings in the `context.ini` file is captured—not into the main application `.msi` file, but rather into a separate `context.msi` file. When creating a virtual package, Repackager combines the data in both the main `.msi` file and the `.context.msi` file to produce the final virtual package.

Reference

This section describes each of the user interface elements and Wizard panels that you might encounter when using the Automated Application Converter. The help topics in this Reference section are the same detailed documentation that is displayed when you press the F1 key or click the **Help** button while working in a dialog box.

Reference information is organized into the following sections:

Table 9-13 • Organization of Automated Application Converter Reference Section

Section	Description
Automated Application Converter User Interface	Contains information about the main Automated Application Converter interface, including tabs, menus, and the toolbar.
Wizards	Contains a panel-by-panel reference for each Wizard in the Automated Application Converter.
Dialog Boxes	Provides specific help for each dialog box in the Automated Application Converter.
Command Line Support	Explains how to run the Automated Application Converter project file via command line.
Specifying Global Default Virtual Conversion Settings	Explains how to set global default settings for any App-V virtual setting in the **ISVirtualPackage** table by editing the `settings.xml` file.

Automated Application Converter User Interface

Information on the Automated Application Converter user interface is presented in the following sections:

- Packages Tab
- Machines Tab
- Results Tab
- Menus & Toolbar Buttons
- Output Window
- Column Selector and Properties Windows
- AdminStudio Automated Application Converter Log Report
- Using List Features

Packages Tab

On the **Packages** tab, you select the packages that you want to virtualize/repackage. On this tab, you can also set package properties and view package status.

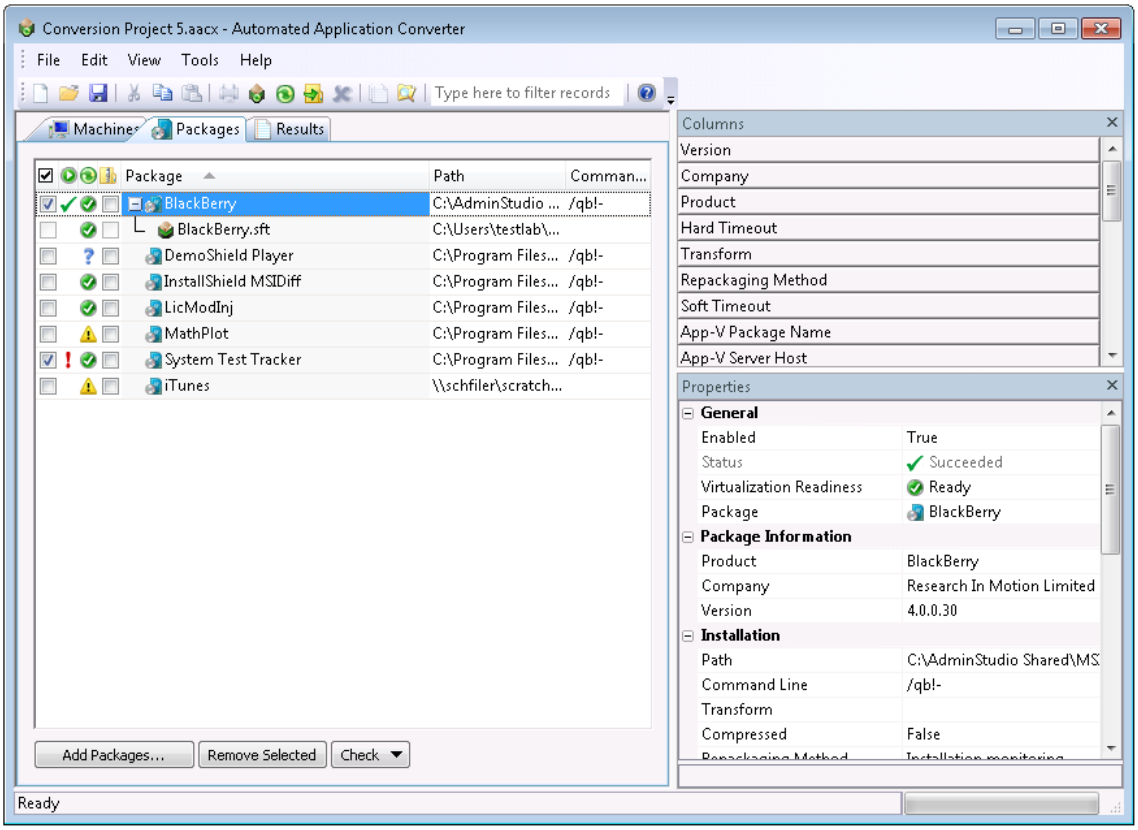


Figure 9-20: Packages Tab

This section includes the following information about the **Packages** tab:

- Adding Packages to the List
- Viewing Package Information on the Packages Tab
- Packages Tab Properties
- Icons Used on the Packages Tab
- Context Menu Commands on Packages Tab

Adding Packages to the List

You add packages to this list using the Package Import Wizard or the Application Conversion Project Wizard. For instructions, see the following topics:

- Selecting Packages from an AdminStudio Application Catalog or Microsoft Configuration Manager Server
- Selecting Packages from a Local Machine or Network

Viewing Package Information on the Packages Tab

By default, the **Packages** tab lists the **Status**, **Virtualization Readiness**, **Path**, and **Command Line** columns for each selected package. Additional columns of information can be viewed by selecting one of the fields in the Column Selector area and dragging it onto the list. Also, the values for these fields for the selected package can be viewed in the **Properties** window.

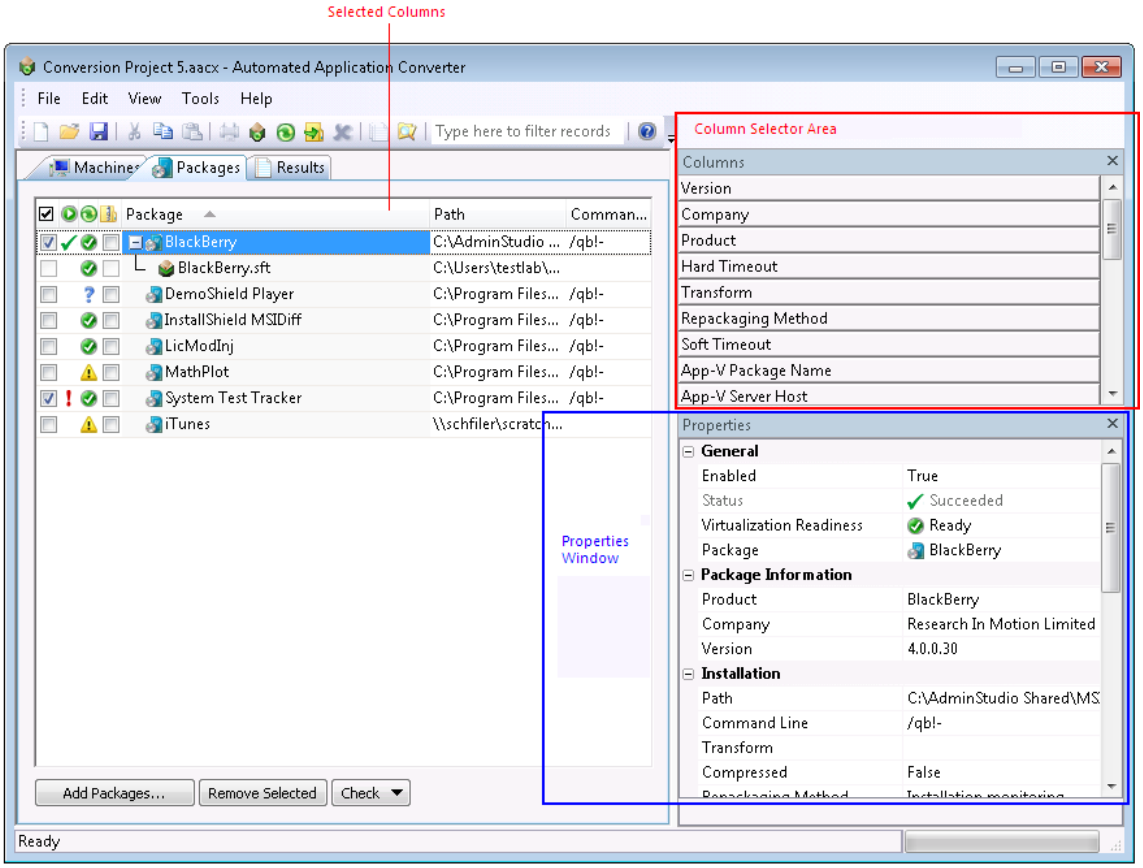


Figure 9-21: Column Selector Area on the Packages Tab

Note • *You can sort these lists, change the columns that are displayed, change the column order, resize the columns, and group these lists by a specific column. See Using List Features for more information.*

Packages Tab Properties

The **Packages** tab includes the following properties, many of which can be edited in the Package list or in the Properties Window.

Table 9-14 • Packages Tab

Property	Description
☑	Selection column. To select a package for conversion, click the check box in this column. ***Note*** • *This field corresponds with the **Enabled** field under **General** in the Properties window, which can be set to either **True** or **False**.*
Status 	Displays an icon to indicate the status of the package when it is being repackaged or virtualized, that the process has completed, or that the Automated Application Converter encountered an error during the process. See Icons Used on the Packages Tab.
Transforms 	Indicates whether any transforms are associated with the listed Windows Installer package. Automated Application Converter automatically adds all of the `.mst` files located in the same directory as the selected `.msi` file. If a transform is associated with the selected package, one of the following two icons are displayed in this column: • One transform is being added with this package. • Multiple transforms are being added with this package. If multiple transforms are associated with this package, you should click the browse button in the **Transform** column (or the browse button in the **Transform** field in the **Properties** window) to open the **MST** dialog box and specify which transform files you want to add and the order that you want the transforms applied. For more information, see MST Dialog Box.

Table 9-14 • Packages Tab (cont.)

Property	Description
Virtualization Readiness	When you add a package to the **Packages** tab, the Automated Application Converter does a quick check to identify that package's virtualization readiness status and assigns it one of the following icons: • **Ready** —Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined. • **Requires repackaging** —Package must be repackaged before it can be successfully virtualized. • **Virtualization not supported** —Automated Application Converter has determined that virtualization is not supported. • **Virtualization not recommended** —Automated Application Converter has determined that this package is not recommended for virtualization. • **Unknown** —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging. ***Note*** • *You can click on the icon in this column to override the Virtualization Readiness status that was automatically assigned to this package by the Automated Application Converter.* ***Note*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready** or **Requires repackaging**.*
Package	Lists the name of the package, as determined by the value in the **Product** property under **Package Information**. This is used as part of the output path for repackaged or virtualized results. It is also used in reports to refer to this package. If the package has been repackaged or converted to a virtual package, those output files are also listed below the source file in a tree structure, with an icon identifying the file type. See Icons Used on the Packages Tab.
Product	Name of the package as provided by the company who manufactured it.
Company	Name of the company who manufactured this package.
Version	Version of the package.

Table 9-14 • Packages Tab (cont.)

Property	Description
Path	Lists the location from where the package was selected locally or from where it was originally imported into the AdminStudio Application Catalog or Microsoft Configuration Manager Server. ***Note*** • *It is recommended that you use UNC path when importing packages into the Application Catalog or publishing packages to Microsoft Configuration Manager Server.* ***Note*** • *If you are adding packages from an AdminStudio Application Catalog or a Microsoft Configuration Manager Server installed on a machine other than the machine where the Automated Application Converter is installed, make sure that the package source path listed here is accessible to the Automated Application Converter machine. If it is also accessible to the virtual machines, repackaging can be performed more quickly.*
Command Line	Editable field that lists the command line parameters that will be used to run this installation silently during repackaging.
Transform	This field can contain a semicolon-delimited list of transforms used to modify or install a Windows Installer package silently. To add a transform to this list, click the Browse button to open the **MST** dialog box. For more information, see MST Dialog Box. ***Note*** • *If the transform file is located in the same directory as the `.msi` file, only the `.mst` file name is listed in the **Transform** field. If you have added a transform file from another directory, the full path is listed in this field.*
Compressed Indicates the compressed status of the package: • **False**—Indicates that the source `.msi` or `.exe` file is uncompressed. If this package is repackaged, the Automated Application Converter will copy all of the files in the same folder as the installation file to the virtual machine. • **True**—Indicates that the source `.msi` or `.exe` file is compressed. If this package is repackaged, the Automated Application Converter will copy only this single installation file to the virtual machine.	
Repackaging Method	Indicates the repackaging method that will be used to repackage this package: • **Installation monitoring**—Repackager monitors system changes as a package is installed, and that data is converted into a Windows Installer package. • **Single-step snapshot**—Repackager first takes an initial system snapshot, then runs the installation, and then takes a second snapshot to create the script file that can be converted into a Windows Installer package.

Table 9-14 • Packages Tab (cont.)

Property	Description
Soft Timeout	Number of minutes allotted for the package to install before the user would be notified. After this time period elapses, the user will be notified, just in case there are pending dialogs for the user to dismiss or if some other user interaction is required. The default value is 20.
Hard Timeout	Number of minutes allotted for the package to install before it is considered a failure. If this time period elapses, the Automated Application Converter would consider the installation a failure and would move to the next package. The default value is 40.
Virtualization Technology	Indicates the virtualization format of the virtual package. Options are: • Microsoft App-V • Citrix XenApp • VMware ThinApp • Windows Installer Package
App-V Package Name	Enter a name (a maximum of 64 characters) to override the name of the App-V package. By default, this matches the value of the **Product** property under **Package Information**. ***Note*** • *In this field, you can specify the single name to use for an App-V application package that contains multiple applications. For example,* `Microsoft Office` *could be used to label a package that contains Microsoft Word and Microsoft Excel applications that run in the same virtual environment.*
App-V Server Host	Specify the virtual application server or the load balancer in front of a group of virtual application servers that will stream the software package to an Application Virtualization Desktop Client. You must complete this item to create a sequenced application package, but you can change from the default `%SFT_SOFTGRIDSERVER%` environment variable to the actual hostname or IP address of a virtual application server. ***Note*** • *If you choose not to specify a static hostname or IP address, on each Application Virtualization Desktop Client you must set up an environment variable called* `SFT_SOFTGRIDSERVER`. • *Its value must be the hostname or IP address of the virtual application server or load balancer that is this client's source of applications.* • *You should make this environment variable a system variable rather than a user variable.* • *Any Application Virtualization Desktop Client session that is running on this computer during your assignment of this variable must be closed and then opened so that the resumed session will be aware of its new application source.*

Table 9-14 • Packages Tab (cont.)

Property	Description
App-V Server Port	Specify the port on which the virtual application server or the load balancer will listen for an Application Virtualization Desktop Client's request for the package. • This information is required to create a package, but you can change it. • The default port is 554.
App-V Server Path	Specify the relative path on the virtual application server where the software package is stored and from which it will be streamed. ***Note*** • *This information is required to create a package if the `.sft` file will be stored in a subdirectory of `CONTENT`; otherwise, this information is not required.*
App-V Server Protocol	Enables you to select the protocol that will stream the sequenced application package from a virtual application server to an Application Virtualization Desktop Client. The following protocols are available: • **RTSP**—The default, it specifies that the Real-Time Streaming Protocol controls the exchange of virtualization-enabled applications. • **RTSPS**—Specifies that the Real-Time Streaming Protocol with Transport Layer Security controls the exchange of a sequenced application package. • **FILE**—Specifies that the sequenced application will be streamed from a file share. • **HTTP**—Specifies that Hypertext Transport Protocol controls the exchange of a package. • **HTTPS**—Specifies that Secure Hypertext Transport Protocol controls the exchange of a package.
App-V Root Folder Name	Enter a name to identify the directory that will contain the App-V application's `.sft` file. When an App-V application is configured on a client machine, it is mounted on the computer's `Q:\` drive in the folder you specify in this setting. The default value for the Root Folder Name setting is based on the `[ProductName]` and `[ProductVersion]` properties of the App-V application's associated Windows Installer file using the 8.3 file naming convention. For example: • If `[ProductName]` is `MyApplication` and `[ProductVersion]` is `1.12.3.1`, the Default Folder Name will be `MyApplic.112`. • If `[ProductName]` is `MyApp` and `[ProductVersion]` is `1`, then the Default Folder Name will be `MyApp.1`. • If `[ProductName]` is `MyBlueApp` and `[ProductVersion]` is `1.2.3.4`, then the Default Folder Name will be `MyBlueAp.123`. If you happen to have another App-V application that has identical values for the `[ProductName]` and `[ProductVersion]` properties, you can avoid conflicts by entering a unique value in the Root Folder Name setting.

Table 9-14 • Packages Tab (cont.)

Property	Description
App-V Comments	Enter a short description of the purpose of this App-V application and any special information that you wish to document. ***Note*** • *The description you enter in this text box will appear in the OSD file ABSTRACT element.*
App-V Supported OS	Specify versions of Windows supported by this App-V package by selecting **True** or **False** next to the OS name. If none are selected, the package is OS independent, and will run on any version of Windows.
App-V Dynamic Suites	Enter a semicolon delimited list of OSD or SFT files to be dynamically suited with this package, or click the browse button and select the OSD or SFT files to be suited. Append `:MANDATORY` to the file if it must be present for this package to function.
App-V Compression	Specify whether to compress this App-V package by selecting **Default**, **Compressed**, or **Uncompressed**.
App-V Package Optimization	Specify whether to optimize package for offline use (entire package is included in Feature Block 1) by selecting **Offline**, or streaming use (only shortcut targets are included in Feature Block 1) by selecting **Stream**.
App-V Upgrade Package	Click the Browse button and specify the previous package of this package. When specified, this package will be built as an upgrade to the previous package.
App-V Versioning	Select **Versioned** to append the package version to the SFT file name. Select **Unversioned** to leave the package version off of the SFT file name. Select **Default** to use the setting that is defined on the **Project Options** dialog box.

Table 9-14 • Packages Tab (cont.)

Property	Description
App-V Diagnostics	Specify the tools to include in this App-V package by selecting **True** or **False**. • **Launcher**—Select this option to include the App-V Application Launcher with each App-V application that you build. You can use the App-V Application Launcher to test a newly built App-V application before moving it to a deployment server. • **Filesystem**—Select this option if you want to include the Windows Command Prompt application with your App-V application so that you can browse the virtual file system at runtime from within the virtual environment. If this option is selected, a file named `Virtual File System.osd` will be created in the App-V Package folder, which can be used to display the files and folders within the virtual environment. You can use `Virtual File System.osd` to view the existing files and folders on the computer plus the files and folders for the virtual package. • **Registry**—Select this option if you want to include the Registry Editor (`regedit.exe`) with your App-V application so that you can browse the registry at runtime from within the virtual environment. If this option is selected, a file named `Virtual Registry.osd` will be created in the App-V Package folder, which can be used to display the registry within the virtual environment. You can use `Virtual Registry.osd` to view the existing registry on the computer plus the registry for the virtual package.
App-V Runtime Drive	Specify the App-V client runtime drive.
Add Packages	Click to launch the Package Import Wizard, which you can use to add packages to the **Packages** tab.
Remove Selected	Click to remove the selected package from this list. You can also click the **Delete** key. ***Note*** • *A package is selected for removal when you click on it and it becomes highlighted, not by selecting the package's check box. Use the **Ctrl** key to select multiple packages.*

Icons Used on the Packages Tab

The following icons are used to display package status on the **Packages** tab:

Table 9-15 • Select Packages Panel

Column	Icon	Description
Status		Package is in the process of being repackaged.
		Package has been successfully repackaged.
		Repackaging has failed.
		Package is waiting in line to be repackaged or to be virtualized.
		A soft timeout has occurred, meaning that the package's **Soft Timeout** time period has elapsed. This could occur because the package is very large and is taking an unusually long time to repackage, or because you have set the **Soft Timeout** value too low, or because the installer is waiting for some kind of user input (meaning that the installation was not silent).
		The last conversion run of this package was cancelled.
Package		Identifies the source package (`.msi` or `.exe` file).
		Identifies the repackaged `.msi` file.
		Identifies the virtual application that was successfully created.

Table 9-15 • Select Packages Panel

Column	Icon	Description
Virtualization Readiness		**Ready** Package is ready to virtualize; no repackaging is required. ***Note*** • *If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined.*
		Requires repackaging Package must be repackaged before it can be successfully virtualized.
		Undetermined The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.
		Virtualization not supported Automated Application Converter has determined that virtualization is not supported due to one of the following issues (described in Virtualization Suitability Issues): • Package contains DLL surrogates. • Package installs boot services. • Package contains OS integrated files. • Package relies on a system-level driver. • Package's .sft file name is over 56 characters in length. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready** or **Requires repackaging**.*
		Virtualization not recommended Automated Application Converter has determined that this package is not recommended for virtualization due to one of the following issues (described in Virtualization Suitability Issues): • Package does not contain a shortcut. • Package includes a custom shell extension. • Package utilizes ClickOnce technology.

Context Menu Commands on Packages Tab

When you make a selection in the **Package** list on the **Packages** tab, the following commands are available on the context menu:

Table 9-16 • Context Menu Commands on Packages Tab

Command	Description
Explore	Open the directory that contains the selected file.
Remove	Permanently removed the selected packages from the project.
Test Virtualization Readiness	Select to test the selected package for virtualization readiness. The possible results are: • —Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined. • —Package must be repackaged before it can be successfully virtualized. • —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.
Launch Package for Testing	Select to install and run the selected application on a virtual machine. See Launching Packages for Testing.
Connect to Machine	When the Running icon is displayed in the **Status** column, indicating that the package is in the process of being repackaged, you can select **Connect to Machine** from the context menu to connect to the virtual machine via Remote Desktop on which this package is being repackaged.
Package Import Wizard	Select this option to launch the Package Import Wizard to import packages to this project.
Package Publish Wizard	Select this option to launch the Package Publish Wizard to publish virtual/repackaged packages to a Microsoft Configuration Manager Server or AdminStudio Application Catalog.

Machines Tab

On the **Machines** tab of the Automated Application Converter, you add a list of clean virtual machine images to use during automated repackaging.

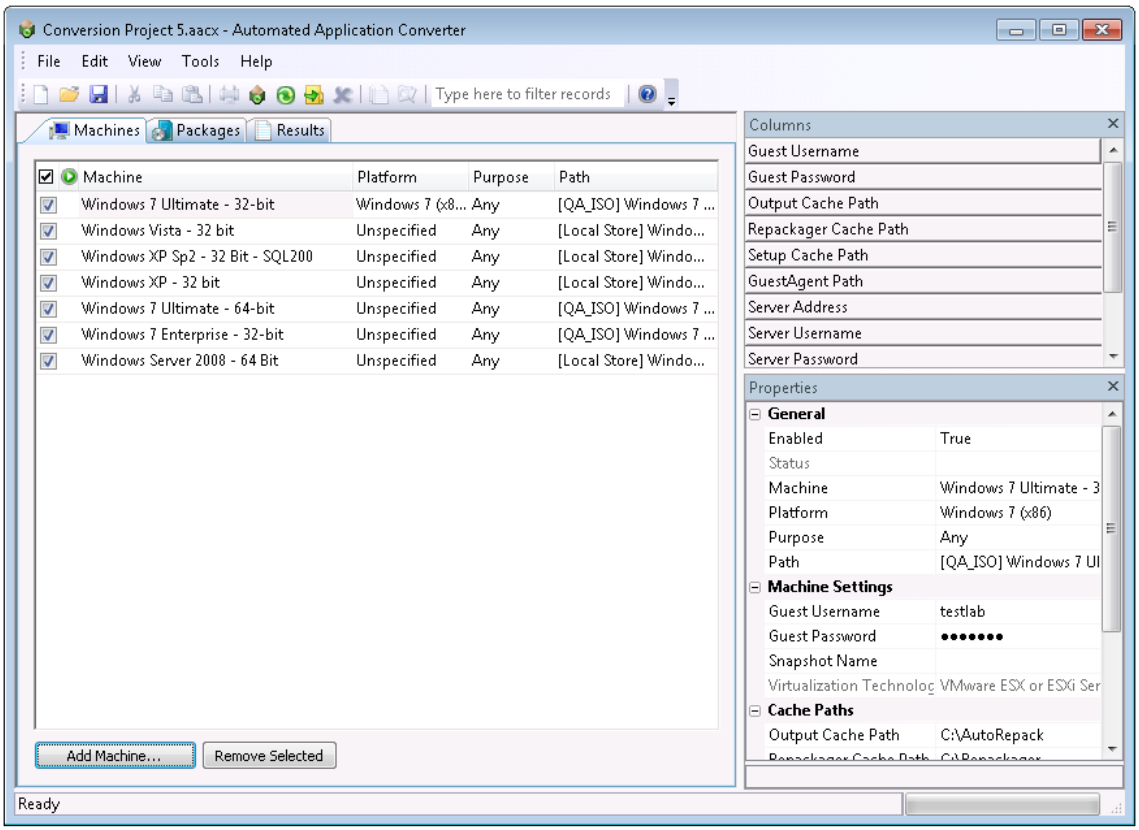


Figure 9-22: Machines Tab

This section includes the following information about the **Machines** tab:

- Adding Virtual Machines to the List
- Viewing Virtual Machine Information on the Machines Tab
- Machines Tab Properties
- Context Menu Commands on Machines Tab

Adding Virtual Machines to the List

To add a virtual machine to the list, click **Add Machine** to open the Virtual Machine Import Wizard, as described in Adding Virtual Machines Using the Virtual Machine Import Wizard. You will then be prompted for login information and other relevant data required to prepare the machine.

***Note** • Before you add a machine to this list, you need to perform the steps listed in Preparing Your Virtual Machines for Use With the Automated Application Converter to enable automatic login and to create a clean snapshot.*

To perform repackaging, you have the option of selecting one virtual machine or multiple virtual machines (that can be used simultaneously to speed up the repackaging of multiple setups). You can also specify that you want to use only virtual machines of a specific operating system platform.

When each virtual machine finishes repackaging a package, it is reverted to its clean snapshot image, and then starts repackaging the next package in the list.

Viewing Virtual Machine Information on the Machines Tab

By default, the **Machines** tab lists the **Status**, **Machine**, **Platform**, **Purpose**, and **Path** columns for each machine. Additional columns of information can be viewed by selecting one of the fields in the **Column** selector area and dragging it onto the list. Also, the properties for these **Columns** for the selected machine can be viewed in the **Properties** window when that column is selected in the **Column** selector area.

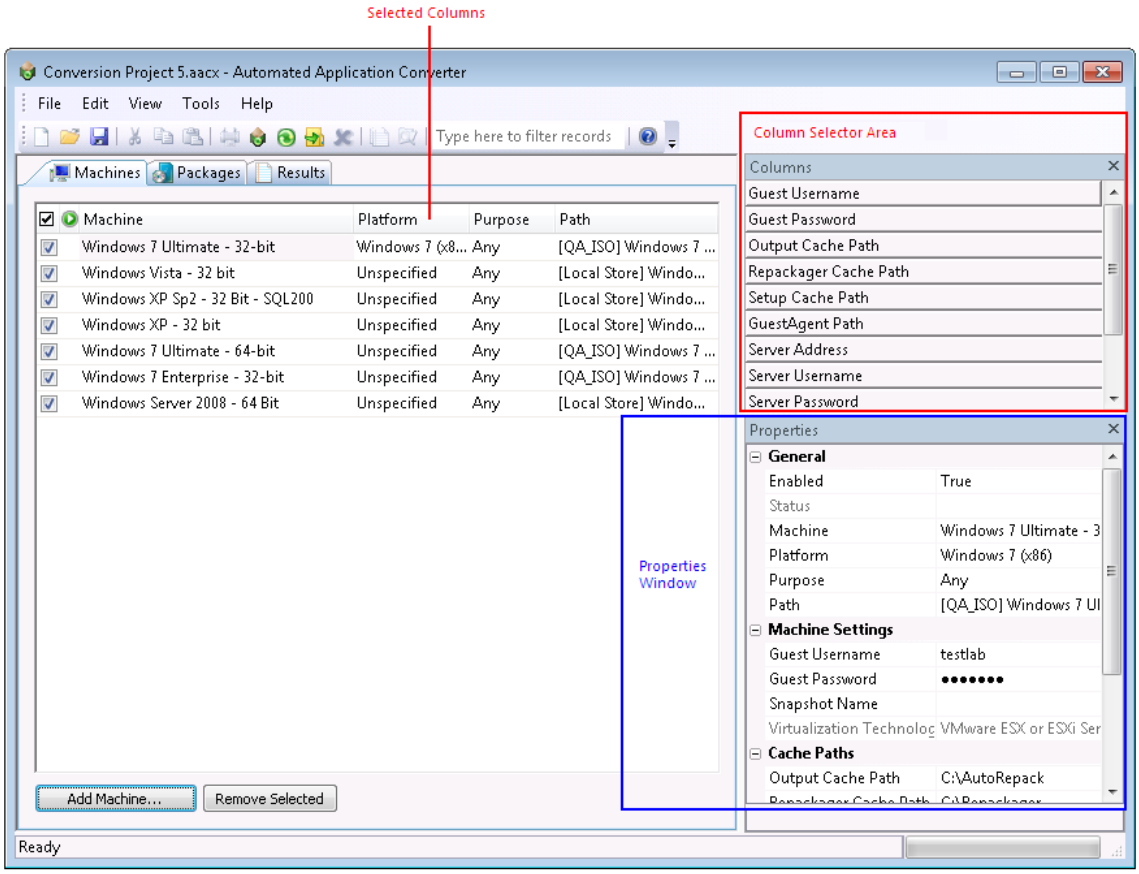


Figure 9-23: Column Selector Area on the Machines Tab

Note • *You can sort this list, change the columns that are displayed, change the column order, resize the columns, and group the list by a specific column. See Using List Features for more information.*

Machines Tab Properties

The **Machines** tab includes the following properties and information:

Table 9-17 • Machines Tab

Property	Description	
☑	Selection column. To select a virtual machine to use for automated repackaging, click the check box in this column.	
Status	The icon displayed in this column indicates the status of the virtual machine:	
		Virtual machine is in use.
		The Automated Application Converter encountered an error when attempting to connect to this virtual machine. or when rolling back to a snapshot on this virtual machine.
		The Automated Application Converter is waiting for the virtual machine to boot up.
	(No icon)	Virtual machine is not currently in use.
		The last conversion run on this virtual machine was cancelled.
Machine	Name of the virtual machine image.	
Platform	Field that identifies the operating system platform of the virtual machine. When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in this field on the **Machines** tab and making a selection from the list. When you perform a conversion run, you are given the opportunity (on the **Automated Repackaging on Virtual Machines** panel) to either select a specific platform to use for the repackaging of the selected packages, or to select **Any Platform**, meaning that all of the selected virtual machines will be used for repackaging.	
Path	Path on the server or file system to the virtual machine image file.	

Table 9-17 • Machines Tab (cont.)

Property	Description
Purpose	By default, virtual machines that you add to the **Packages** tab will be available for use for both automated repackaging of packages and for testing packages. However, if you want to specify that a virtual machine should be used for only repackaging or for only testing, click in the **Purpose** column of that virtual machine and select one of the following options: • **Repackaging**—Virtual machine will only be used to perform automated repackaging. • **Testing**—Virtual machine will only be used to test packages. You test a package by selecting it on the **Packages** tab and selecting **Launch Package for Testing** from the context menu. You will then be prompted to install and run that package on a virtual machine. • **Any**—Make this virtual machine available for use during both automated repackaging and package testing. This is the default value. ***Important*** • *If the **Purpose** column is not listed in the **Machines** list, you can select it in the **Columns** area and drag it to the list, or you can edit the **Purpose** value in the Properties window.* ***Note*** • *The **Launch Package for Testing** functionality will primarily be useful to test converted packages. However, if a problem occurs during conversion, it is also possible to use this function to install and launch the source package for testing.*
Guest Username	The user name to use to login to this virtual machine.
Guest Password	The password to use to login to this virtual machine.
Output Cache Path	Specify the location for the repackaged output on the virtual machine. By default, this value is `C:\AutoRepack`.
Repackager Cache Path	Specify the location where Repackager will be installed on the virtual machine. By default, this value is `C:\Repackager`.
Setup Cache Path	Specify the location where the package will be copied to on the virtual machine. By default, this value is `C:\AppSetup`.
GuestAgent Path	Specify the location where the `GuestAgent.exe` file will be installed on the virtual machine. By default, this value is `C:\GuestAgent.exe`.
Server Address	The address of the virtual machine server on which this virtual machine is found. This may be a host name or a URL.
Server Username	The user name of the account used to access the virtual machine server.

Table 9-17 • Machines Tab (cont.)

Property	Description
Server Password	The password of the account used to access the virtual machine server.
Snapshot Name	Name of the snapshot to revert to before starting an automated repackaging session. This is only used if the virtualization technology supports named snapshots. If this value is not specified, but named snapshots are supported on the virtualization technology, the default name of `AutoRepack_Base` will be used.
Virtualization Technology	The virtualization technology powering this virtual machine.
Add Machine	Click to launch the Virtual Machine Import Wizard, which you can use to add virtual machines to the **Machines** tab.
Remove Selected	Click to remove the selected virtual machine from this list. ***Note*** • *A virtual machine is selected for removal when you click on it and it becomes highlighted, not by selecting the virtual machine's check box. Use the **Ctrl** key to select multiple machines.*

Context Menu Commands on Machines Tab

When you make a selection in the **Machines** list on the **Machines** tab, the following commands are available on the context menu:

Table 9-18 • Context Menu Commands on Machines Tab

Command	Description
Connect to Machine	When the Running icon is displayed in the **Status** column, indicating that this virtual machine is currently being used to perform repackaging, you can select **Connect to Machine** from the context menu to connect to this virtual machine via Remote Desktop.
Remove	Select to remove the selected machine from this project.
Machine Import Wizard	Select to add virtual machines to this project using the Virtual Machine Import Wizard.

Results Tab

On the **Results** tab, the results of each virtualization conversion run for this project are listed.

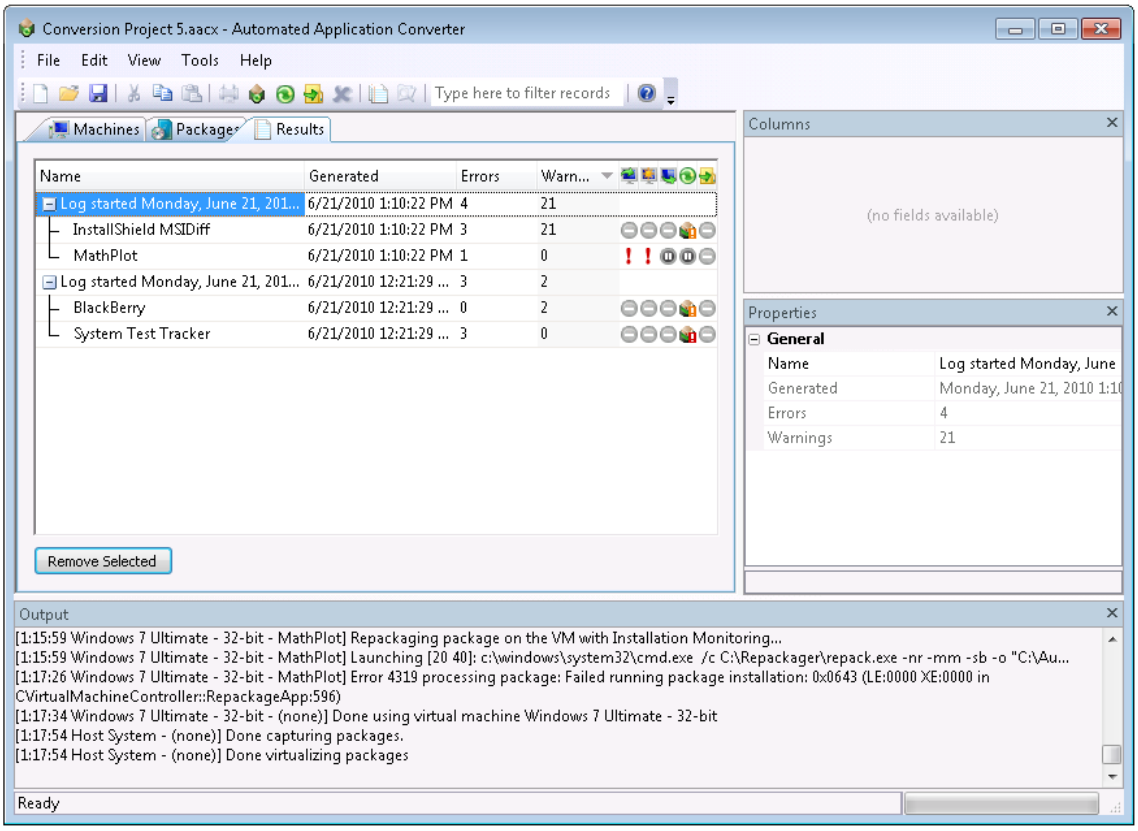


Figure 9-24: Results Tab

This section includes the following information about the **Results** tab:

- Results Tab Properties
- Icons Used on the Results Tab
- Context Menu Commands on Results Tab

Results Tab Properties

The **Results** tab includes the following properties and information:

Table 9-19 • Results Tab

Property	Description
Name	List of logged results for each run of this project. The log is identified by the date and time it was started, and the packages that were part of this run are listed in a tree structure under the log title. Click the plus sign to expand the listing.
Generated	Date and time the conversion of each package began.
Errors	The number of errors generated for each package in this run is listed in this column next to each package. The cumulative sum of all errors generated for all of the packages in the run is listed in this column next to the parent Log row.
Warnings	The number of warnings generated for each package in this run is listed in this column next to each package. The cumulative sum of all warnings generated for all of the packages in the run is listed in this column next to the parent Log row.
Results Icons	Icons in these columns indicate the status of each of the steps of the repackaging and conversion process. See Icons Used on the Results Tab for detailed information.

Icons Used on the Results Tab

The following icons are used on the **Results** tab:

Table 9-20 • Icons Used on Results Tab

Column	Icon	Description
Copy In **Repackage** **Copy Out**		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was successfully performed, but warnings were encountered. View the results AdminStudio Automated Application Converter Log Report for detailed information on these warnings.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) failed. • **Copy In**—Error could have been caused by not being able to connect to the virtual machine. • **Repackage**—Error means that repackaging has failed. • **Copy Out**—Error could mean that you ran out of hard drive space at the package source location or that there is a permission problem preventing you from writing to the selected directory. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) was skipped. Possible reasons that the operation was skipped could be: • **Repackaging not required**—Because repackaging was not required, these three operations were not required. • **Could not connect to virtual machine**—The Automated Application Converter could not successfully connect to the virtual machine, so therefore the **Repackage** and **Copy Out** operations were skipped.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is currently being performed.
		Operation (copying to virtual machine, repackaging, or copying from virtual machine back to source location) is still being performed even though a warning was generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Operation was cancelled.

Table 9-20 • Icons Used on Results Tab (cont.)

Column	Icon	Description
Conversion Column		Package was converted to a virtual application successfully.
		Package was converted to a virtual application, but warnings were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		Package was converted to a virtual application, but errors were generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		The Automated Application Converter was unable to convert this package to a virtual application.
		Conversion is in progress.
		Conversion is in progress, but a warning has been generated. View the results AdminStudio Automated Application Converter Log Report for detailed information on the errors encountered.
		An error was generated when converting one of the virtual formats which caused it to fail. However, the conversion to another one of the selected virtual formats continues.
		Conversion was cancelled.

Context Menu Commands on Results Tab

When you select a log node on the **Results** tab, the following commands are available on the context menu:

Table 9-21 • Context Menu Commands on Results Tab

Command	Description
View Report	Select to view the AdminStudio Automated Application Converter Log Report for the selected run.
Explore	Select to open the directory where the selected log file is located.
Remove	Select to delete the selected log file.

Note • *If you have selected a child node under the parent Log node, the context menu is disabled.*

Menus & Toolbar Buttons

The Automated Application Converter user interface includes the following menus, commands, and toolbar icons:

Table 9-22 • Automated Application Converter Menus and Commands

Menu	Command	Icon	Description
File	**New**		Click to open a new Automated Application Converter project file.
	Open...		Click to open an existing Automated Application Converter project file.
	Save		Click to save the open Automated Application Converter project file.
	Save As...		Click to save the open Automated Application Converter project file in a new location or using a different name.
	Recently Opened Items		Lists the most recently used list of Automated Application Converter projects. Click to open.
	Exit		Click to exit the Automated Application Converter.
Edit	**Copy**		Copy selected text. You can then paste it in an external program such as Notepad or Microsoft Word.
	Select All		Select all of the Packages, Machines, or Results in the list.
	Select None		Unselect all selected items.
View	**Toolbars**		Select one of the following options: • **Standard**—Toggles the display of the toolbar. • **Customize** —Select to customize which menu commands are displayed on the toolbar.
	Windows		Toggle the display of the **Columns** selector area, the **Properties** window and the **Output** window.
	Status Bar		Toggles the display of the status bar at the bottom of the interface.

Table 9-22 • Automated Application Converter Menus and Commands (cont.)

Menu	Command	Icon	Description
Tools	**Project Wizard....**		Click to open the Application Conversion Project Wizard, which guides you step-by-step through the entire virtualization process: adding virtual machines, adding packages, and virtualizing applications.
	Application Conversion Wizard...		Click to open the Application Conversion Wizard, which you can use to select the virtualization format you want to convert to and to perform conversion of the selected packages on the selected virtual machines.
	Explore		Open the directory containing the selected package.
	View Report		After selecting the top level node of a conversion run log on the **Results** tab (`Log started Monday, April 01, 2010...`), click this to open the AdminStudio Automated Application Converter Log report. See AdminStudio Automated Application Converter Log Report for more information.
	Test Virtualization Readiness		Click to test the selected package for virtualization readiness. The possible results are: • —Package is ready to virtualize; no repackaging is required. If a Windows Installer package does not contain any custom actions, conditional components, or unsupported tables, repackaging prior to virtualization is not required. An example of an unsupported table is the `IniFile` table, which changes files on the target machine in ways that cannot be statically determined. • —Package must be repackaged before it can be successfully virtualized. • —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging.
	Cancel Virtualization		Cancel current conversion run.

Table 9-22 • Automated Application Converter Menus and Commands (cont.)

Menu	Command	Icon	Description
Help	**Contents**		Launches the Help Library, displaying the **Contents** tab.
	Index		Launches the Help Library, displaying the **Index** tab.
	Search		Launches the Help Library, displaying the **Search** tab.
	About the Automated Application Converter		Displays the **About the Automated Application Converter** dialog box.

Output Window

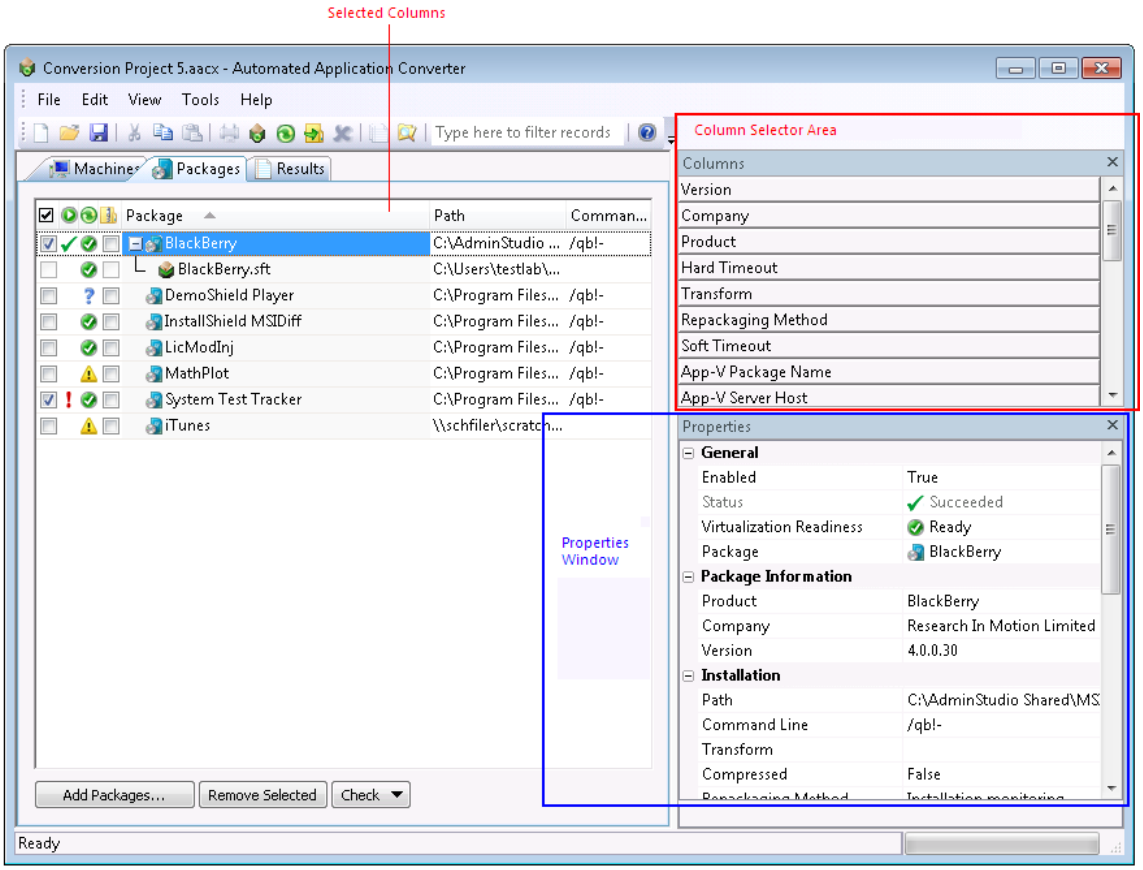
When a virtualization run is performed, the output messages and results are displayed in the Output Window.

You can copy the results in the Output window and paste them in an outside location, such as Notepad or Microsoft Word.

All of the messages and results listed in the Output window can also be viewed in the AdminStudio Automated Application Converter Log report. See AdminStudio Automated Application Converter Log Report.

Column Selector and Properties Windows

By default, the **Packages** and **Machines** tabs list several columns of information. However, additional columns of information can be viewed by selecting one of the fields in the **Column Selector** area and dragging it onto the list. Also, the values for these fields for the selected package or machine can be viewed in the **Properties** window.



A description of each of these properties can be found in Packages Tab and Machines Tab.

Note • *You can sort the lists on the **Package** and **Machines** tabs, change the columns that are displayed, change the column order, resize the columns, and group the lists by a specific column. See Using List Features for more information.*

AdminStudio Automated Application Converter Log Report

The AdminStudio Automated Application Converter Log is an HTML report you can view that lists the following information for each conversion run:

- **Machines**—List of the virtual machines used in the conversion run.
- **Packages**—List of the packages that included in this conversion run. The packages are linked to the Package Conversion Messages section of the report for that package.
- **Log Results** / **General Messages**—Start time, number of errors and warnings generated, and general processing messages.
- **Log Results** / **Package Conversion Messages**—Conversion messages for each package that the Automated Application Converter attempted to convert.

The following is an example of an AdminStudio Automated Application Converter Log report:

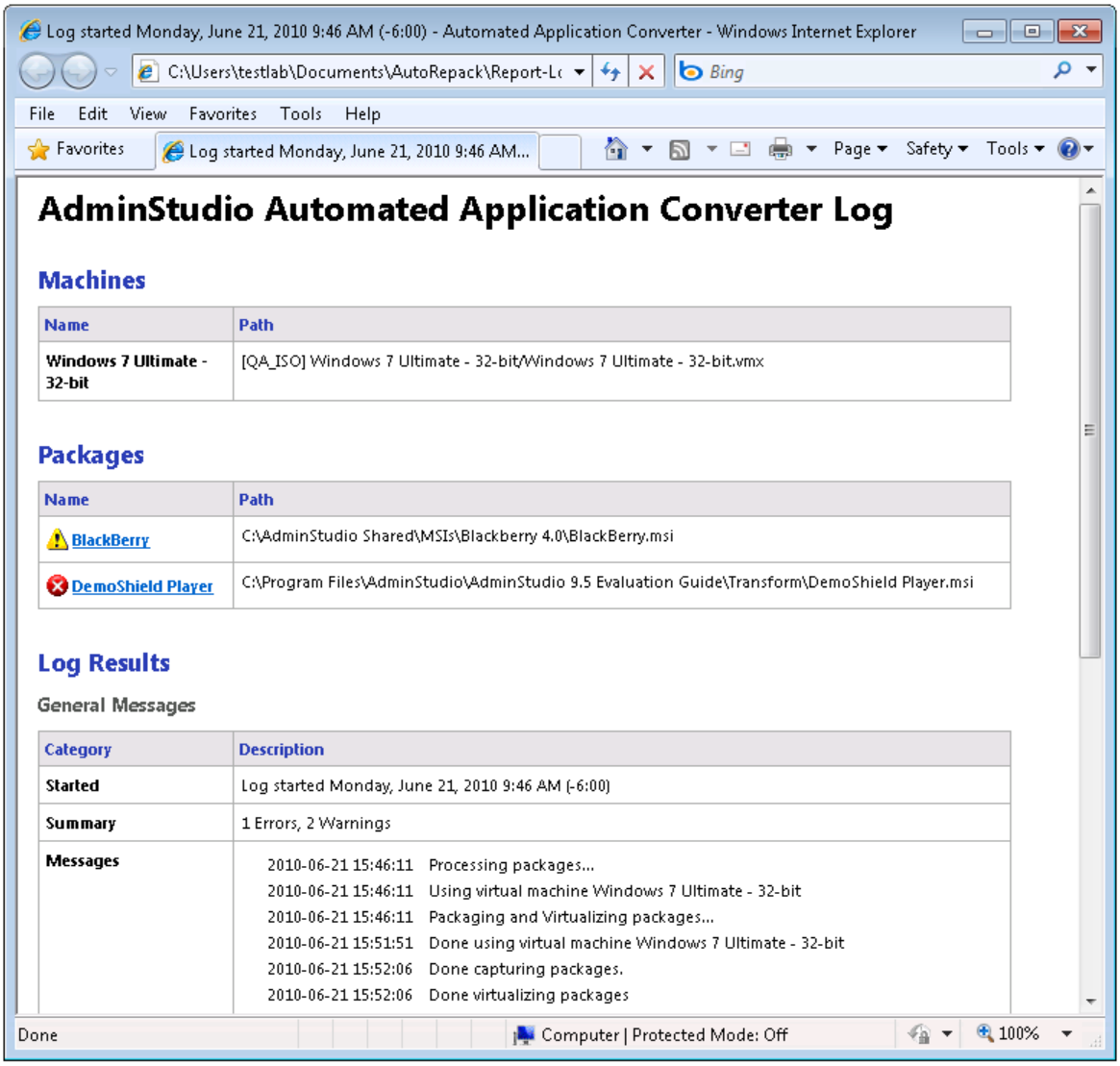


Figure 9-25: AdminStudio Automated Application Converter Log Report

Viewing an AdminStudio Automated Application Converter Log Report

To view an AdminStudio Automated Application Converter Log report, perform the following steps:

Task: ***To view an AdminStudio Automated Application Converter Log report:***

1. Open the **Results** tab.
2. Select the top level node of a conversion run log (`Log started Monday, June 21, 2010...`), and do one of the following:
   - Click **View Report** on the **Tools** menu.
   - Select **View Report** from the context menu.
   - Click the **Reports** icon on the toolbar.
   - Press Ctrl+R.

   The report opens in a new browser window.

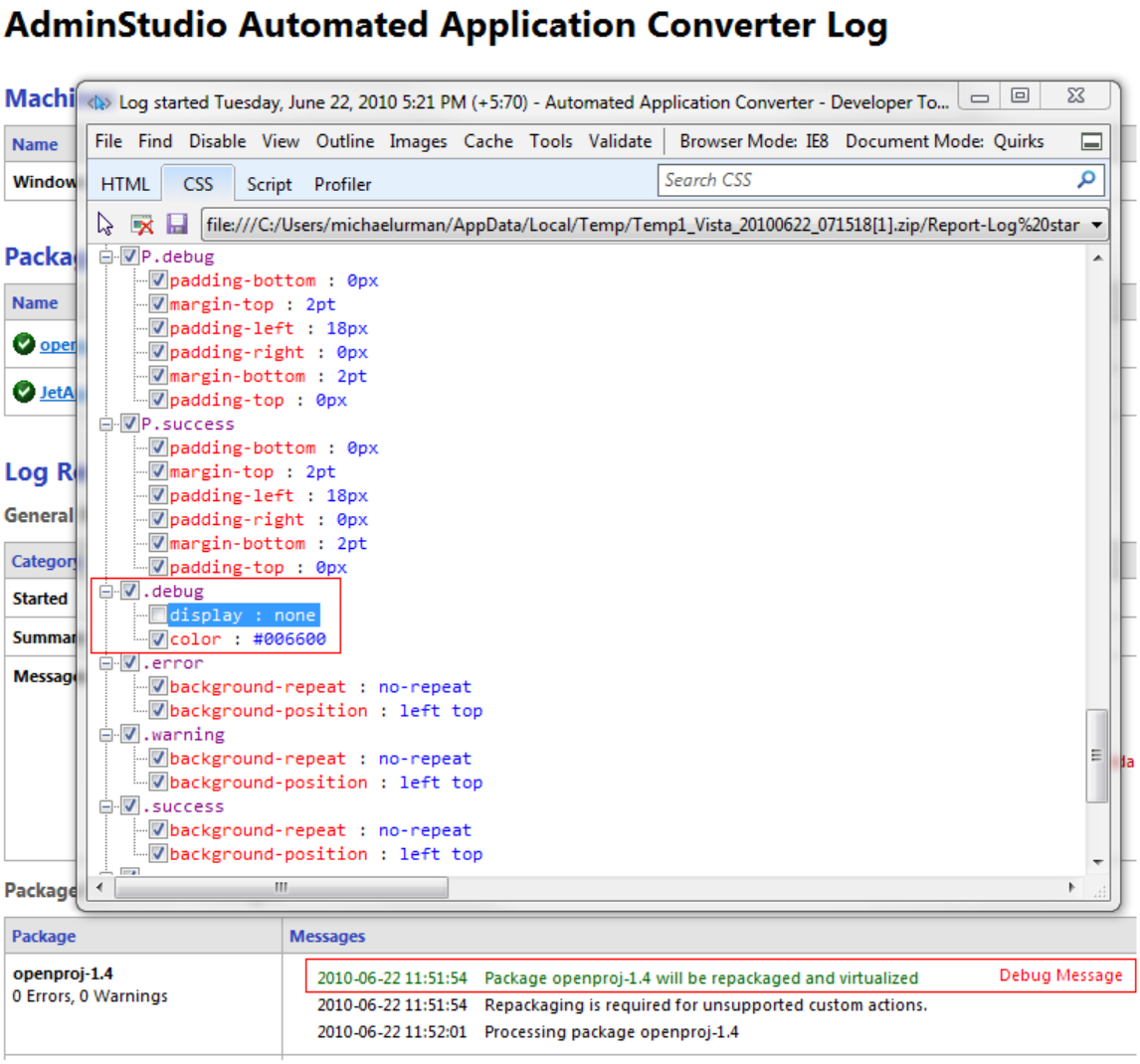
Viewing Debug Messages

By default, debug messages that occur during a conversion run are saved in the log report, but the display of those debug messages is turned off. However, if you are using Microsoft Internet Explorer 8 as your default browser, you can choose to view those debug messages by performing the following steps:

Task: ***Viewing debug mesages in an Automated Application Converter Log Report:***

1. Open an Automated Application Converter Log Report as described in Viewing an AdminStudio Automated Application Converter Log Report.
2. Press F12. The **Developer Tools** window opens.
3. Click on the **CSS** tab. The CSS file for the log report opens, displaying checkmarks next to each class and each property.
4. Scroll down to locate the following class: `.debug` (not `P.debug`).

5. Clear the selection of the `display : none` property, as shown below:



6. Return to the log report window. The debug messages are now displayed in green.

Using List Features

All of the lists displayed in the Automated Application Converter user interface—including the lists shown on Wizard panels—implement the same list features, which allow you to group a list by any column, sort a list by any column, resize list columns, change which columns are displayed, and change column order.

- Sorting Lists
- Changing Which List Columns Are Displayed
- Changing Column Order
- Resizing List Columns
- Grouping Lists

Sorting Lists

You can sort lists by any column by clicking on the header of the column you want to sort by or by selecting a column header and making a selection from the context menu.

Task: ***To sort a list by a column heading:***

1. Open the **Machines**, **Packages**, or **Results** tab.
2. To sort by a column heading, click on the column heading to toggle through the three sort order states, which are identified by a visual indicator:
   - **Sorted in ascending order**—When the column is sorted in ascending order, an up arrow is displayed in the header row.
   - **Sorted in descending order**—When the column is sorted in descending order, a down arrow is displayed in the header row.
   - **Not sorted**—When the column is not sorted (meaning that the list is either sorted by another column or is just listed in the default order that the records appear in the database), no arrow is displayed.

   Tip • *Another way to do this is to select the column header of the column you want to sort, and then select* ***Sort Ascending*** *or* ***Sort Descending*** *from the context menu.*
3. To sort just the children of the top level items (not the top level items), select the column header of the column you want to sort by, and then select **Sort Children** from the context menu.

Changing Which List Columns Are Displayed

To improve readability or clarity, you can choose to remove a column from a list. When you remove a column from a list, you are just turning off the display of that column, not deleting the data that was in that column. You can restore a removed column to the list at any time.

Adding/Restoring a Column to a List

To restore the display of a hidden column to a list, perform the following steps.

Task: ***To restore the display of a deleted column to a list:***

1. To restore the display of a deleted column to a list, point the cursor anywhere in the heading row.
2. Point to **Columns** in the context menu. A list of all of the available columns for this list is displayed.
3. Select the name of the column that you want to restore to the list.

Tip • *To add a column to the list, you can also click and drag a column header from the* ***Column Selector*** *area to the header row of the list.*

Removing a Column from a List

To remove a column from a list, perform the following steps.

Task: ***To remove a column from a list:***

1. To remove a column from a list, point the cursor anywhere in the heading row.
2. Point to **Columns** in the context menu. A list of all of the available columns for this list is displayed, with those that are currently selected for display indicated by a check mark.
3. Select the name of the column to clear the selection.

   The column is now hidden.

Note • *To remove a column to the list, you can also click and drag a column header from the header row of the list to the* ***Column Selector*** *area.*

Changing Column Order

To help compare the values of columns, you can click and drag to change the order of columns in a list.

Task: ***To change column order:***

1. Click on the column header of the column you wish to relocate.
2. While holding the mouse button down, drag the column header on top of the rule between two columns.
3. When the red arrows appear, release the mouse button to perform the move.

Resizing List Columns

To improve the readability of the values in a column of a list, you can click and drag to resize a column.

Task: ***To resize a column:***

1. Position your cursor at the right side of the column header of the column that you want to resize and click. After you click, the cursor turns into a double-arrow icon:
2. While holding the mouse button down, drag the edge of the column left or right until it is the desired width.

Grouping Lists

This section explains how to group a list by a column, ungroup a list, and create subgroupings.

- Grouping an Ungrouped List
- Ungrouping a List

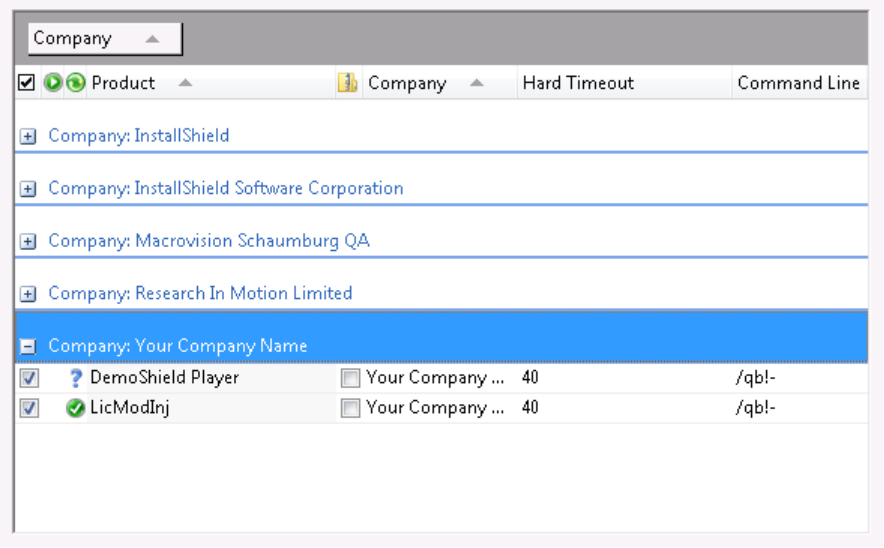
Grouping an Ungrouped List

If a list is not grouped by a column, no Group By Box is displayed.

To group by a column, perform the following steps.

Task: ***To group a list by a column heading:***

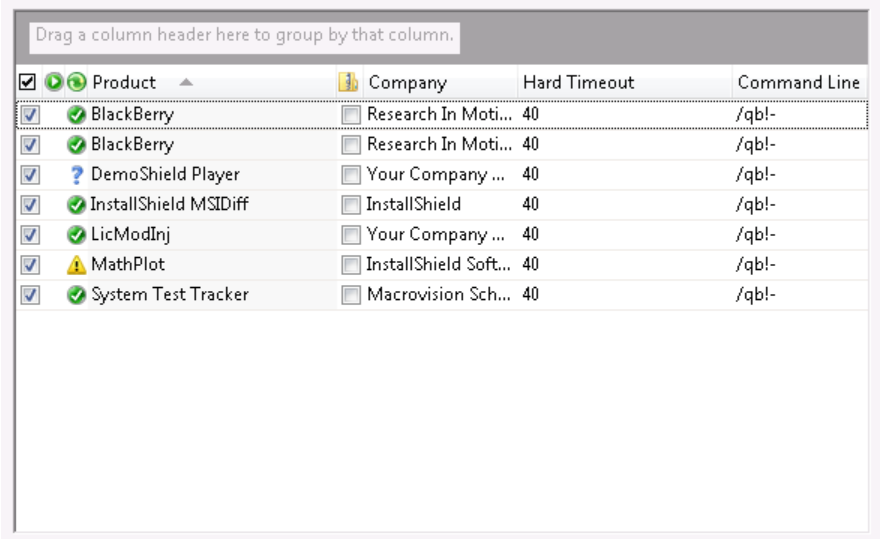
1. Click on the column header that you want to group by and select **Group By [Column Name]** from the context menu.



When you group by a column header, the following occurs:

- The name of the column header that you have chosen now appears in the Group By Box.
- The name of the column header that you selected and each of its values (in the format of `Column Name: Value`) is now listed in a bar at the left of the list, with all records associated with that value of that column grouped underneath that bar.
- One group (bar) appears for each of the values of the selected column. Click the plus sign to expand the list.

2. To display the Group By box without performing any grouping, select **Show Group Header** from the context menu. The text `Drag a column header here to group by that column.` is displayed in the **Group By** box:



Note • *When the Group By box is displayed, you can perform grouping by dragging a column header to the Group By box.*

Ungrouping a List

Perform the following steps to ungroup a list.

Task: ***To ungroup a list or change a list's Group By column:***

1. Click on the name of the column header in the Group By Box and drag it back to the header row in the list to the location where you want the column to be displayed.

Tip • *Another way to do this is to click on the column header name and clearing the* ***Group By [Column Name]*** *selection on the context menu.*

The list is now ungrouped.

2. If you want to choose another column to use to group the list by, follow the steps listed above under Grouping an Ungrouped List.

Wizards

The Application Conversion Project Wizard guides you step-by-step through the entire virtualization process: adding virtual machines, adding packages, and virtualizing packages. You can also choose to perform each of these tasks separately by using one of the other three wizards that are provided:

Table 9-23 • Automated Application Converter Wizards

If you want to ...	Use this wizard ...	Description and Purpose
Add packages	**Package Import Wizard**	Add packages from an AdminStudio Application Catalog, a Microsoft Configuration Manager server, or from a local or network file system.
Add virtual machines	**Virtual Machine Import Wizard**	Add virtual machines to use to perform automated repackaging of Windows Installer packages.
Virtualize packages	**Application Conversion Wizard**	Virtualize packages to the virtual formats you specify.

You can also use the Package Publish Wizard to publish converted packages to an AdminStudio Application Catalog or Microsoft Configuration Manager Server.

Application Conversion Project Wizard

When using the Automated Application Converter to perform batch conversion to virtual packages, there are three main procedures that you perform:

- **Step 1: Select packages**—Select packages to virtualize and/or repackage.
- **Step 2: Select machines**—Select the virtual machines that you want to use during automated repackaging.
- **Step 3: Select formats and perform conversion**—Select the virtualization formats you want to convert to and perform the conversion.

You can use the **Application Conversion Project Wizard** to perform all three of these steps in one guided procedure.

You can launch the Application Conversion Project Wizard in one of two ways:

- **Creating a new project upon product launch**—The **Open Project** panel opens automatically when you launch the Automated Application Converter or when you select **New Project** on the **File** menu.
- **Creating a new project after product launch**—Select **Project Wizard** on the **Tools** menu or when you click the Project Wizard icon on the toolbar, or select **New Project** on the **File** menu.

The Application Conversion Project Wizard includes the following panels:

- Open Project Panel
- Application Conversion Project Wizard Welcome
- Select Package Source
- Connect to an AdminStudio Application Catalog
- Connect to a Microsoft Configuration Manager Server
- Select Packages
- Selected Package List
- Select Virtual Machine Source
- Select Virtual Machines from a Microsoft Hyper-V Server
- Select Virtual Machines from a VMware ESX or ESXi Server
- Select Virtual Machines
- User Credentials
- Initial Configuration Complete
- Select Output Formats
- Automated Repackaging on Virtual Machines
- Application Conversion Project Wizard Complete Panel

Automated Application Converter's Other Wizards

Each of the Automated Application Converter's other three wizards—Virtual Machine Import Wizard, Package Import Wizard, and Application Conversion Wizard—consist of a subset of the panels included in the Application Conversion Project Wizard. The following table lists the panels in each of these three wizards.

Table 9-24 • Breakdown of Panels in the Automated Application Converter Wizards

Panel Name	Application Conversion Project Wizard	Package Import Wizard	Virtual Machine Import Wizard	Application Conversion Wizard
Select Package Source	X	X		
Connect to an AdminStudio Application Catalog	X	X		
Connect to a Microsoft Configuration Manager Server	X	X		
Select Packages	X	X		
Selected Package List	X	X		
Select Virtual Machine Source	X		X	
Select Virtual Machines from a Microsoft Hyper-V Server	X		X	
Select Virtual Machines from a VMware ESX or ESXi Server	X		X	
Select Virtual Machines	X		X	
User Credentials	X		X	
Initial Configuration Complete	X			
Select Output Formats	X			X
Automated Repackaging on Virtual Machines	X			X

Open Project Panel

The **Open Project** panel opens automatically when you launch the Automated Application Converter or when you select **New Project** on the **File** menu.

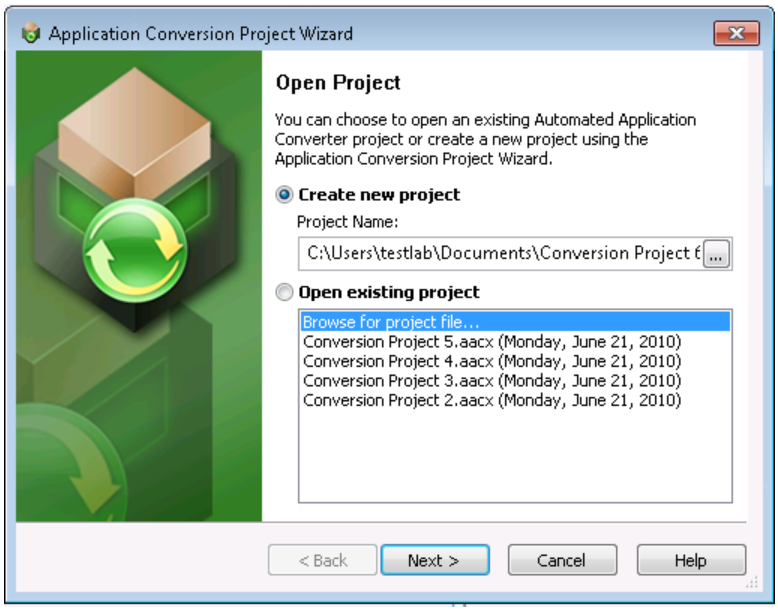


Figure 9-26: Open Project Panel

You have the following options:

- **Create a new project**—If you select the **Create a new project** option, click **Next** to continue with the wizard. You will be prompted to name and save the project when you begin conversion or exit the Automated Application Converter.
- **Select an existing project**—If you select an existing project from the list, click **Finish** to open the project.

Application Conversion Project Wizard Welcome

The **Application Conversion Project Wizard Welcome** panel opens when you select **Project Wizard** on the **Tools** menu or when you click the Project Wizard icon on the toolbar.

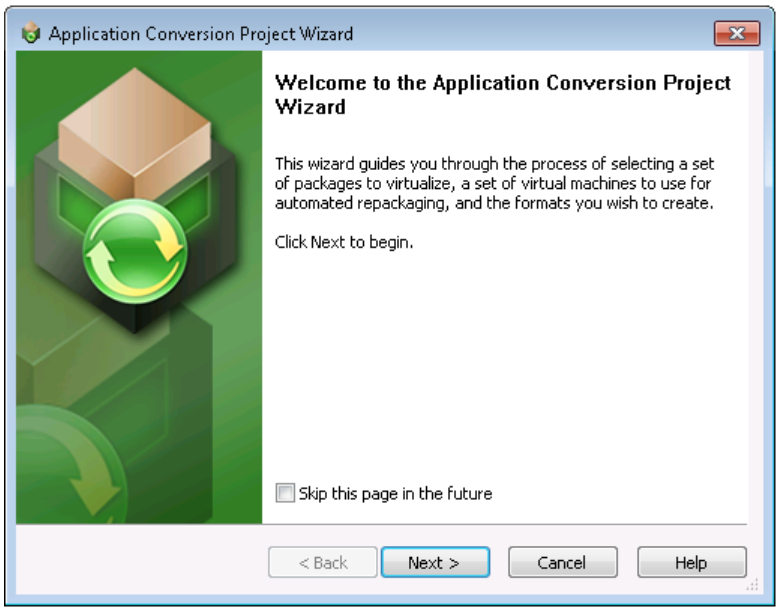


Figure 9-27: Application Conversion Project Wizard Welcome Panel

If you would prefer to perform each of these steps separately, you can instead use the following wizards:

Table 9-25 • Automated Application Converter Wizards

Wizard	Description
Virtual Machine Import Wizard	Use to add virtual machines to your project which can be used to perform automated repackaging into Windows Installer packages.
Package Import Wizard	Use to select packages from a Microsoft Configuration Manager Server, AdminStudio Application Catalog, or from a file system to virtualize.
Application Conversion Wizard	Use to select the virtualization format you want to convert to and to perform the conversion.

Select Package Source

On the **Select Package Source** panel, you select the source that contains the packages that you want to virtualize and/or repackage.

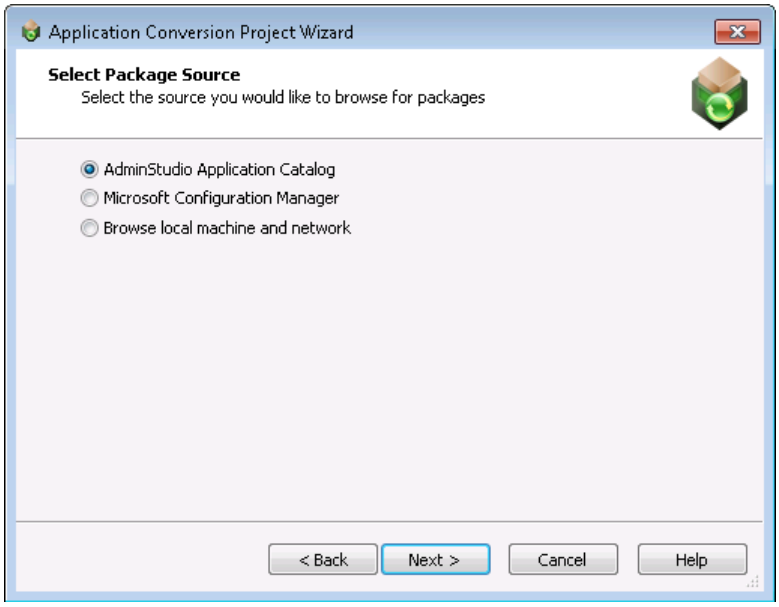


Figure 9-28: Select Package Source Panel

The **Select Package Source** panel includes the following options:

Table 9-26 • Select Package Source Panel

Option	Description
AdminStudio Application Catalog	Select this option to connect to an AdminStudio Application Catalog and add all of the installations in that catalog to the list of packages to convert. If you select this option, the Connect to an AdminStudio Application Catalog panel opens, prompting you to login to an Application Catalog.
Microsoft Configuration Manager	Select this option to connect to a Microsoft System Center Configuration Manager Server and add all of the installations on this server to the list of packages to convert. If you select this option, the Connect to a Microsoft Configuration Manager Server panel opens, prompting you to login to a Microsoft Configuration Manager Server.

Table 9-26 • Select Package Source Panel

Option	Description
Browse local machine and network	Select this option to browse a local or network machine to add installations to the list of packages to convert. If you select this option, the Selected Package List panel opens, where you are prompted to select an installation file or a directory of installation files to add to the list of packages to convert.

Connect to an AdminStudio Application Catalog

On the **Connect to an AdminStudio Application Catalog** panel, which opens if you select **AdminStudio Application Catalog** on the **Select Package Source** panel, you enter connection information to connect to an AdminStudio Application Catalog SQL database.

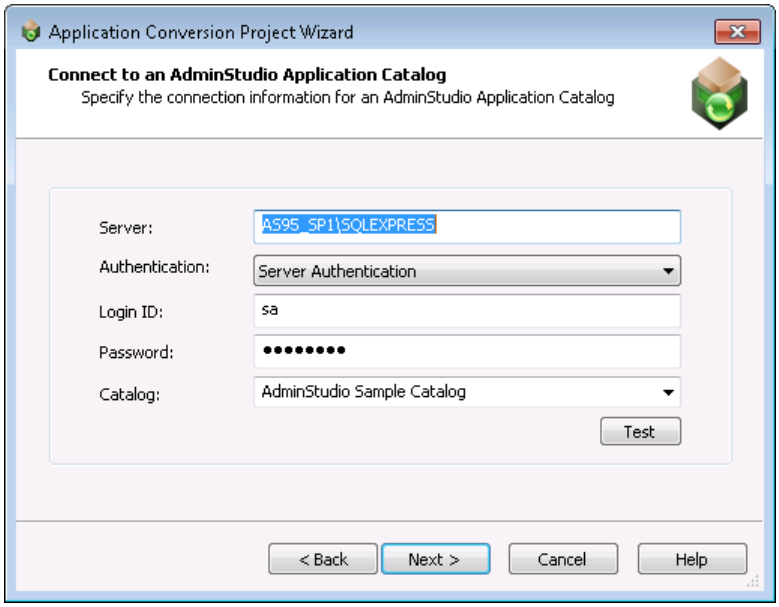


Figure 9-29: Connect to an AdminStudio Application Catalog Panel

On the **Connect to an AdminStudio Application Catalog** panel, enter the following information:

Table 9-27 • Connect to an AdminStudio Application Catalog Panel

Option	Description
Server	Enter the name of the SQL Server that you want to connect to.

Table 9-27 • Connect to an AdminStudio Application Catalog Panel

Option	Description
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use SQL Server login identification to log into this Application Catalog. Then enter the appropriate **Login ID** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Application Catalog. ***Note*** • *After you successfully connect to an Application Catalog, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*
Catalog	Enter the name of the existing AdminStudio Application Catalog database that you want to connect to

Connect to a Microsoft Configuration Manager Server

On the **Connect to a Microsoft Configuration Manager Server** panel, which opens if you select **Microsoft Configuration Manager** on the **Select Package Source** panel, you enter connection information to connect to a Microsoft System Center Configuration Manager server.

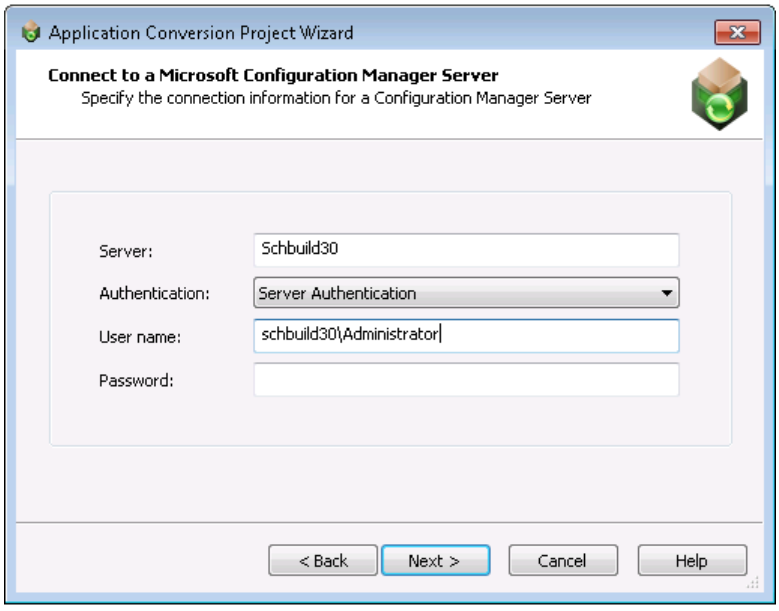


Figure 9-30: Connect to a Microsoft Configuration Manager Server Panel

On the **Connect to a Microsoft Configuration Manager Server** panel, enter the following information:

Table 9-28 • Connect to a Microsoft Configuration Manager Server Panel

Option	Description
Server	Enter the name of the Microsoft Configuration Manager Server that you want to connect to.
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server login identification to log into this Configuration Manager Server. Then enter the appropriate **User name** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Microsoft Configuration Manager Server. ***Note*** • *After you successfully connect to a Microsoft Configuration Manager Server, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*

Select Packages

The contents of the **Select Packages** panel depends upon the selection you made on the **Select Package Source** panel:

- **Browse local machine and network**—If you selected this option on the **Select Package Source** panel, there are no packages listed on the **Select Packages** panel. You need to click **Browse Folders** or **Browse Files** to select packages to convert. The **Browse For Folder** or **Select Package Installation File** dialog box would open. See Automated Application Converter's Selection Rules When Adding Packages from a Directory for more information.

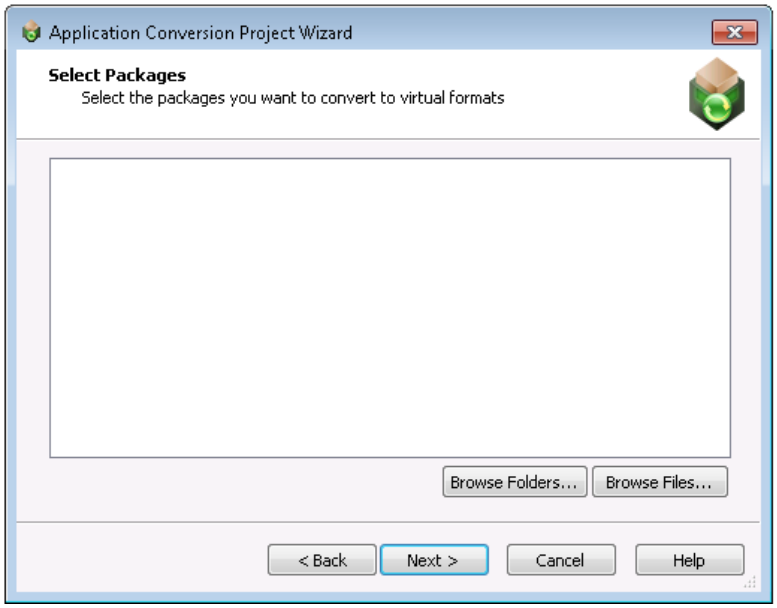


Figure 9-31: Select Packages Panel / No Packages Listed

- **AdminStudio Application Catalog** or **Microsoft Configuration Manager**—If you selected either of these options on the **Select Package Source** panel, the **Select Packages** panel lists all of the packages in the connected package source, in a tree format.

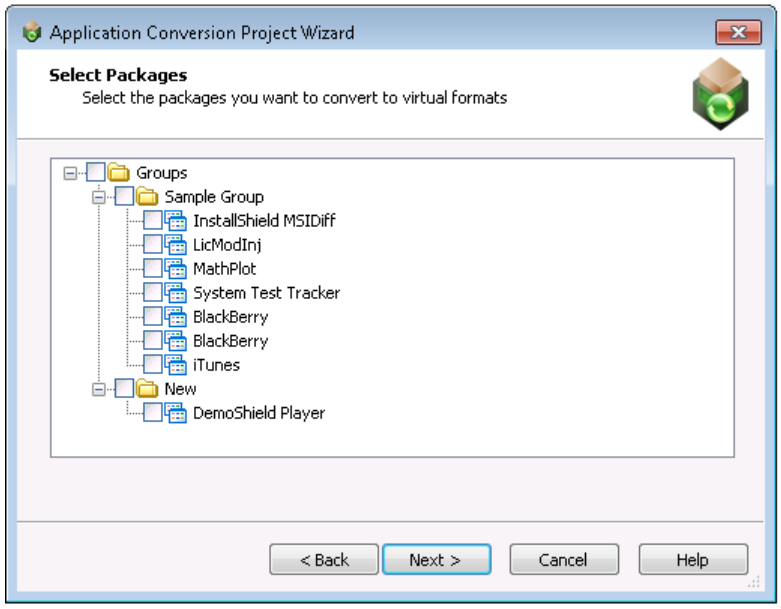


Figure 9-32: Select Packages Panel / Packages Listed

Select the packages that you want to convert and click **Next** to continue.

Selected Package List

The **Selected Package List** panel lists all of the packages you selected on the Select Packages panel.

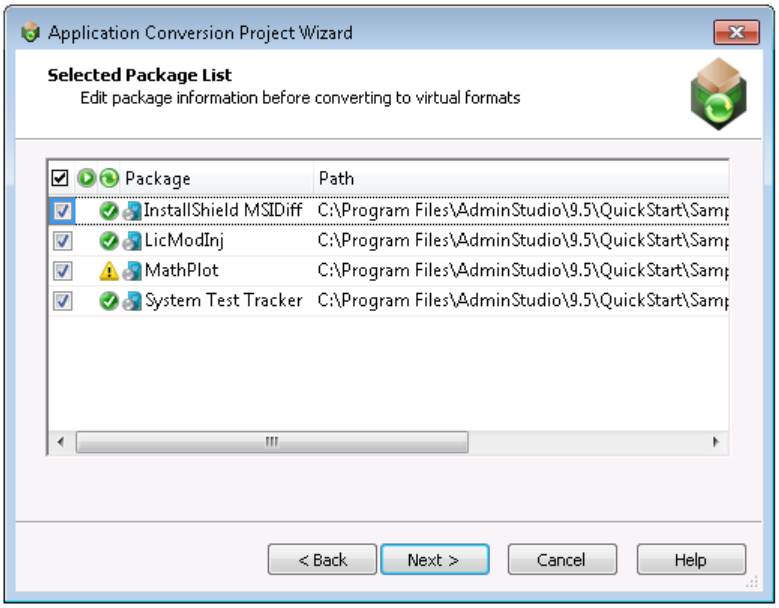


Figure 9-33: Selected Package List Panel

The **Selected Package List** panel includes the following options:

Table 9-29 • Selected Package List Panel

Option	Description
☑	Selection column. To select a package for conversion, click the check box in this column.
Status	Indicates the status of the package. On this panel, no status is indicated, but when this column is shown on the **Packages** tab, status will be indicated by an icon. See Packages Tab for more information.
Transforms	Indicates whether any transforms are associated with the listed Windows Installer package. Automated Application Converter automatically adds all of the `.mst` files located in the same directory as the selected `.msi` file. If a transform is associated with the selected package, one of the following two icons is displayed in this column: • One transform is being added with this package. • Multiple transforms are being added with this package. You may need to specify the order that you want these transforms to be applied. See Packages Tab and MST Dialog Box for more information.

Table 9-29 • Selected Package List Panel (cont.)

Option	Description
Virtualization Readiness	When you add a package to the **Selected Packages List** panel, the Automated Application Converter does a quick check to identify that package's virtualization readiness: whether the package can be virtualized directly or whether it requires repackaging before virtualization. An icon is displayed in this column to indicate the virtualization readiness: You can click on the icon in this column to override the Virtualization readiness status that was automatically been assigned to this package by the Automated Application Converter. The choices are: • **Ready** —Package is ready to virtualize; no repackaging is required. • **Requires repackaging** —Package must be repackaged before it can be successfully virtualized. • **Virtualization not supported** —Automated Application Converter has determined that virtualization is not supported. • **Virtualization not recommended** —Automated Application Converter has determined that this package is not recommended for virtualization. • **Unknown** —The Automated Application Converter was unable to determine whether this package is ready to be virtualized directly or whether it requires repackaging. ***Important*** • *Packages with a status of **Virtualization not supported** will not be virtualized. In order to virtualize the package, you must first override the status and change it to **Ready** or **Requires repackaging**.* ***Note*** • *You can click on the icon in this column to override the Virtualization readiness status that has automatically been assigned to this package.*
Package	Lists the name of the Windows Installer file or legacy installation file that you have added to the **Select Packages** panel.

Table 9-29 • Selected Package List Panel (cont.)

Option	Description
Path	Lists the path from where the package was selected locally or from where it was originally imported into the AdminStudio Application Catalog or Microsoft Configuration Manager Server. ***Note*** • *It is recommended that you use UNC path when importing packages into the Application Catalog or publishing packages to Microsoft Configuration Manager Server.* ***Note*** • *If you are adding packages from an AdminStudio Application Catalog or a Microsoft Configuration Manager Server installed on a machine other than the machine where the Automated Application Converter is installed, make sure that the package source path listed here is accessible to the Automated Application Converter machine.*
Command Line	Editable field that lists the command line parameters required to run this installation silently.

Automated Application Converter's Selection Rules When Adding Packages from a Directory

Instead of adding packages from an AdminStudio Application Catalog or a Microsoft Configuration Manager Server, you can choose to add a directory of packages from your local machine or network by doing the following:

- On the **Select Package Source** panel, select **Browse local machine and network** and click **Next**. The **Select Packages** panel opens.
- On the **Select Packages** panel, click **Browse Folders** and select a directory that contains multiple Windows Installer files (`.msi`), installation script files (`.vbs`, `.bat`, `.cmd`, `.ps1`), and/or legacy setups (`.exe`).

When adding packages from a directory, it is recommended that you organize the packages you want to convert in one root directory, with each package in its own first level subdirectory, such as:

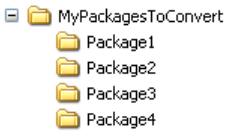


Figure 9-34: Recommended Directory Structure When Adding Packages from a Directory

When you click the **Browse Folders** button and select a folder (such as `MyPackagesToConvert`), the Automated Application Converter scans that folder's first-level subfolders (such as `Package1`, `Package2`, `Package3`, etc.) and uses specific rules to determine which packages it will add to the list on the **Select Packages** panel and which of those packages will be selected:

- **All .msi, .exe, and script files are added to the list**—All `.msi` files, `.exe` files, and script files in the first-level subfolders are added to the list.

- **Only some of the packages are selected**—The Automated Application Converter uses the following rules to determine which of the packages that it adds to the list are selected:
  - `.msi` files are always selected.
  - `.exe` files are only selected if there are no .msi files in that folder.
  - Script files are only selected if there are neither `.msi` files nor `.exe` files in that folder.
- **If a first-level subfolder does not contain any .msi, .exe, or script files, its subfolders are scanned**—If a first-level subfolder does not contain any `.msi`, `.exe`, or script files, the Automated Application Converter will scan its child subfolders to locate package files. However, if a first-level subfolder does contain an `.msi`, `.exe`, or script file, its subfolders are not scanned.

The following table demonstrates these rules:

Table 9-30 • Automated Application Converter's Selection Rules When Adding Packages from a Directory

If the root subdirectory contains...	What are added to the list? Which are selected?	Continue to search subdirectories?
MSI files only	☑ MSIs (added and selected)	No
MSI files and EXE or script files	☑ MSIs (added and selected) ☐ EXEs (added, not selected) ☐ Scripts (added, not selected)	No
EXE files only	☑ EXEs (added and selected)	No
EXE and script files	☑ EXEs (added and selected) ☐ Scripts (added, not selected)	No
Script files only	☑ Scripts (added and selected)	No
None of the above	None	Yes

The following diagram gives a visual representation of these rules in action:

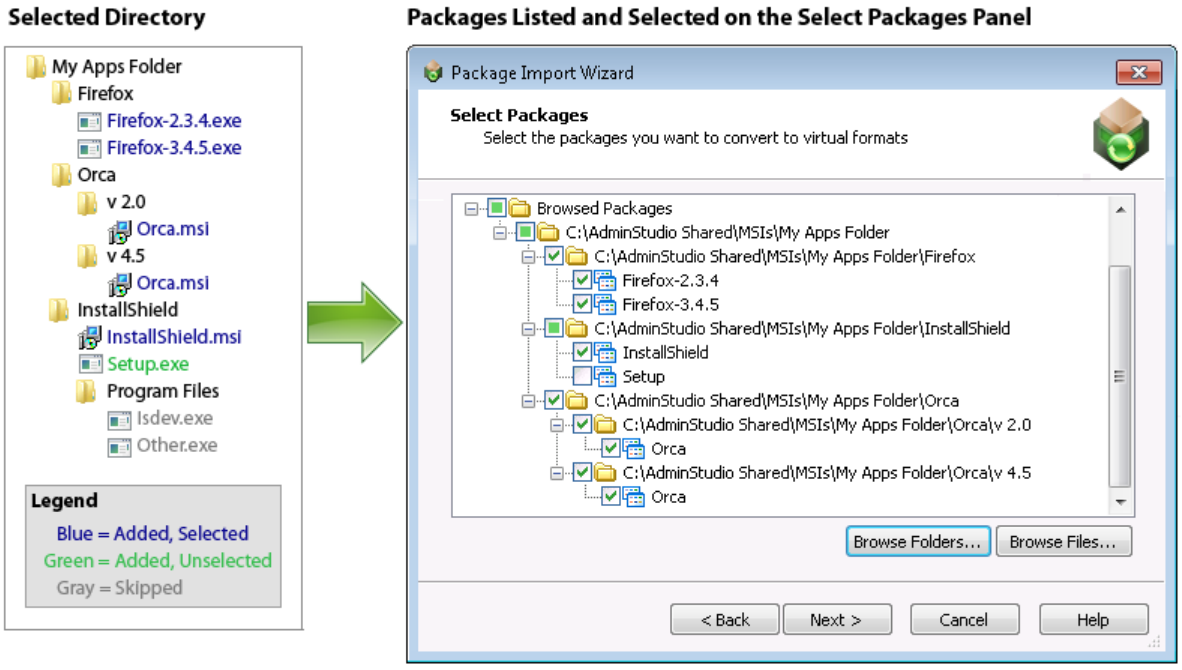


Figure 9-35: Example of How Packages are Added to List When a Directory is Selected

Select Virtual Machine Source

On the **Select Virtual Machine Source** panel, you select the source location of the virtual machines you want to use with the Automated Application Converter to perform automated repackaging.

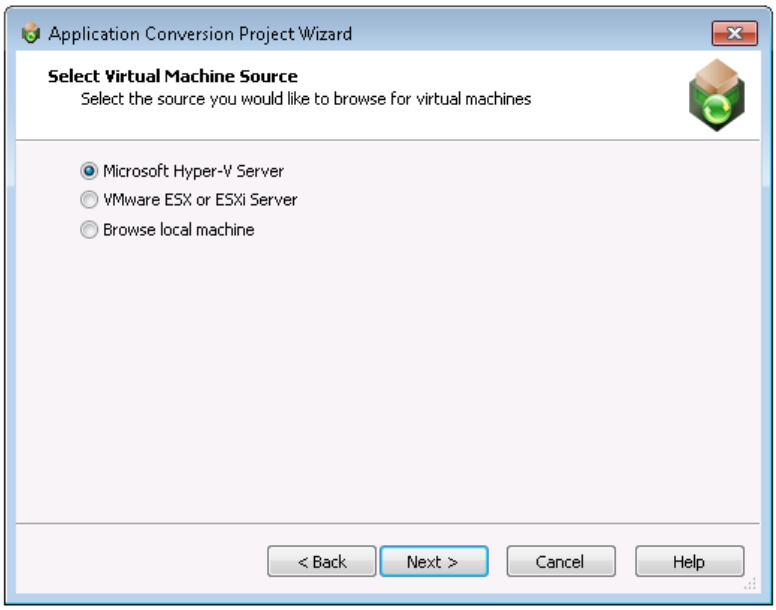


Figure 9-36: Select Virtual Machine Source Panel

Note • *If none of the packages selected on the **Selected Package List** panel require repackaging in order to be converted into a virtual package, the **Select Virtual Machine Source** panel will not be displayed. Instead, the **Initial Configuration Complete** panel will open.*

On the **Select Virtual Machine Source** panel, select one of the following options:

Table 9-31 • Select Virtual Machine Source Panel

Option	Description
Microsoft Hyper-V Server	Select this option to connect to a Microsoft Hyper-V Server. You will then be prompted for login information on the Select Virtual Machines from a Microsoft Hyper-V Server panel.
VMware ESX or ESXi Server	Select this option to connect to a VMware ESX or ESXi Server. You will then be prompted for login information on the Select Virtual Machines from a VMware ESX or ESXi Server panel.
Browse local machine	Select this option to connect to a VMware Workstation virtual image installed locally. The Select Virtual Machines opens, where will be prompted to select either a VMware Workstation image or directory of images.

Select Virtual Machines from a Microsoft Hyper-V Server

On the **Select Virtual Machines from a Microsoft Hyper-V Server** panel, you enter a server name and the login information to connect to a Microsoft Hyper-V Server.

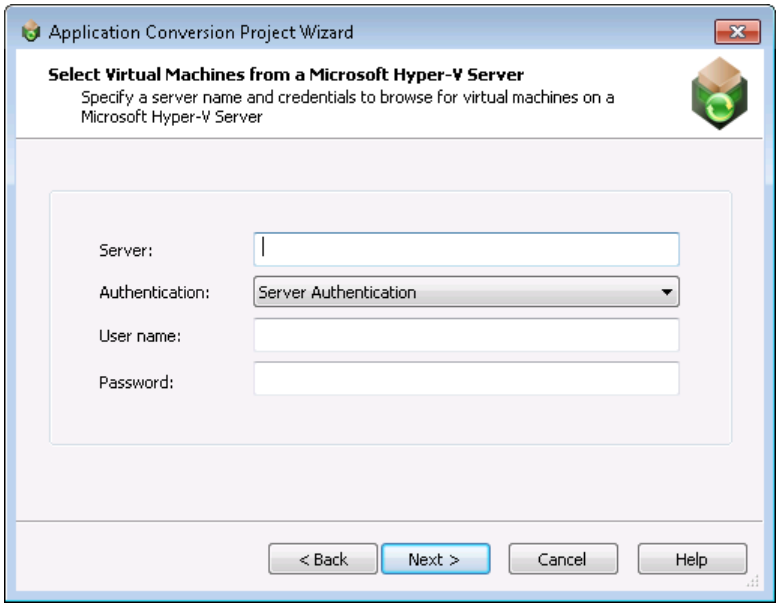


Figure 9-37: Select Virtual Machines from a Microsoft Hyper-V Server Panel

On the **Select Virtual Machines from a Microsoft Hyper-V Server** panel, enter the following information:

Table 9-32 • Select Virtual Machines from a Microsoft Hyper-V Server

Option	Description
Server Name	Enter the server name of the Microsoft Hyper-V Server that you want to connect to.
Authentication	Select one of the following options: • **Server Authentication**—Select this option if you want to connect to the Hyper-V Server using a **User name** and **Password** that you specify. • **Windows Authentication**—Select this option to use the credentials of the logged in user to login to the Hyper-V Server.

Select Virtual Machines from a VMware ESX or ESXi Server

On the **Select Virtual Machines from a VMware ESX or ESXi Server** panel, you enter a server name and the login information to connect to a VMware ESX or ESXi Server.

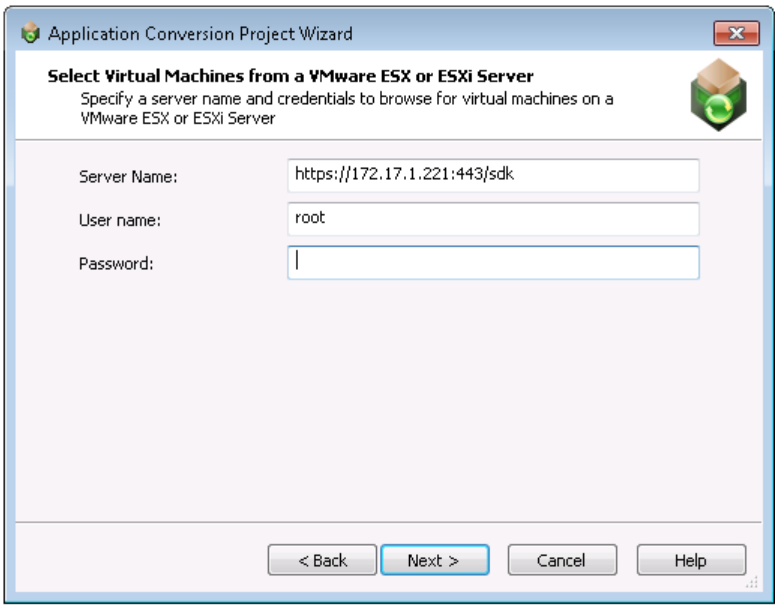


Figure 9-38: Select Virtual Machines from a VMware ESX or ESXi Server Panel

On the **Select Virtual Machines from a VMware ESX or ESXi Server** panel, enter the following information:

Table 9-33 • Select Virtual Machines from a VMware ESX or ESXi Server Panel

Option	Description
Server Name	Enter the name of the VMware ESX or ESXi server you want to connect to.
User name	Enter the login ID for the VMware ESX or ESXi server.
Password	Enter the password for the VMware ESX or ESXi server.

Select Virtual Machines

The contents of the **Select Virtual Machines** panel depends upon the selection you made on the **Select Virtual Machine Source** panel:

- **Browse local machine**—If you selected this option, there are no virtual machines listed on the **Select Virtual Machines** panel. You need to click **Browse Folders** or **Browse Files** to select virtual images. The Browse for Folder Dialog Box or Select Virtual Machine Image File Dialog Box would open.

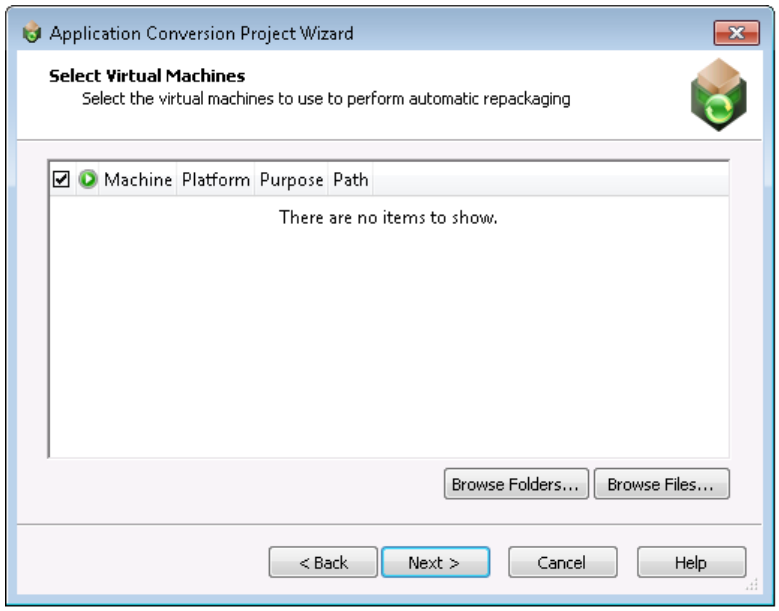


Figure 9-39: Select Virtual Machines Panel / No Machines Listed

- **Microsoft Hyper-V Server** or **VMware ESX or ESXi Server**—If you selected either of these options on the **Select Virtual Machine Source** panel, and have connected to the server, the **Select Virtual Machines** panel lists all of the virtual machines found on the selected server, but does not automatically select all of them.

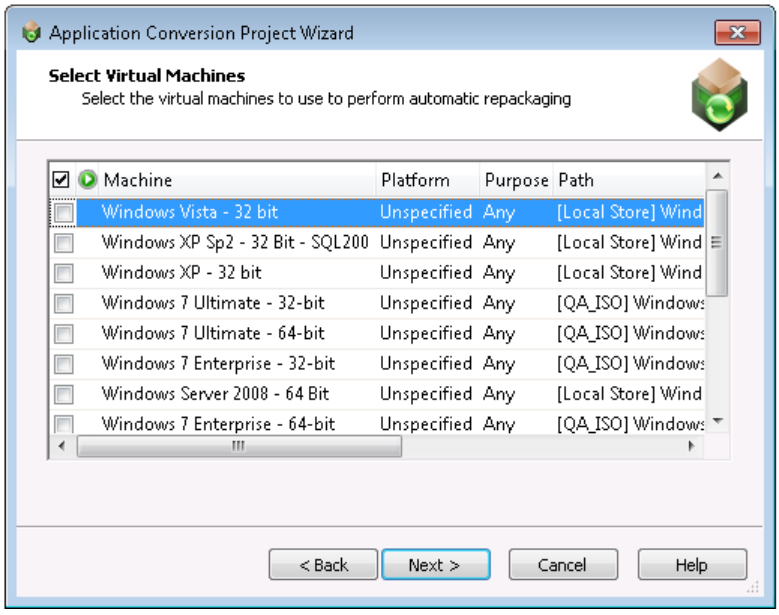


Figure 9-40: Select Virtual Machines Panel / Machines Listed

The **Select Virtual Machines** panel includes the following options:

Table 9-34 • Select Virtual Machines Panel

Option	Description
☑	Selection column. To select a virtual machine to use for automated repackaging, click the check box in this column.
Status	Indicates the status of the virtual machine. On the **Select Virtual Machines** panel, no status is indicated, but when this column is shown on the **Machines** tab, status will be indicated by an icon. See Machines Tab for more information.
Machine	Name of the virtual machine image.

Table 9-34 • Select Virtual Machines Panel (cont.)

Option	Description
Platform	Identifies the operating system platform of the virtual machine. When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the **Select Virtual Machines** panel or by clicking in this field on the **Machines** tab and making a selection from the list. When you perform a conversion run, you are given the opportunity (on the **Automated Repackaging on Virtual Machines** panel) to either select a specific platform to use for the repackaging of the selected packages, or to select **Any Platform**, meaning that all of the selected virtual machines will be used for repackaging.
Path	Path to the virtual machine on the virtual machine server or your local machine.

User Credentials

On the **User Credentials** panel, enter the user name and password to use to access the virtual machines you selected on the **Select Virtual Machines** panel.

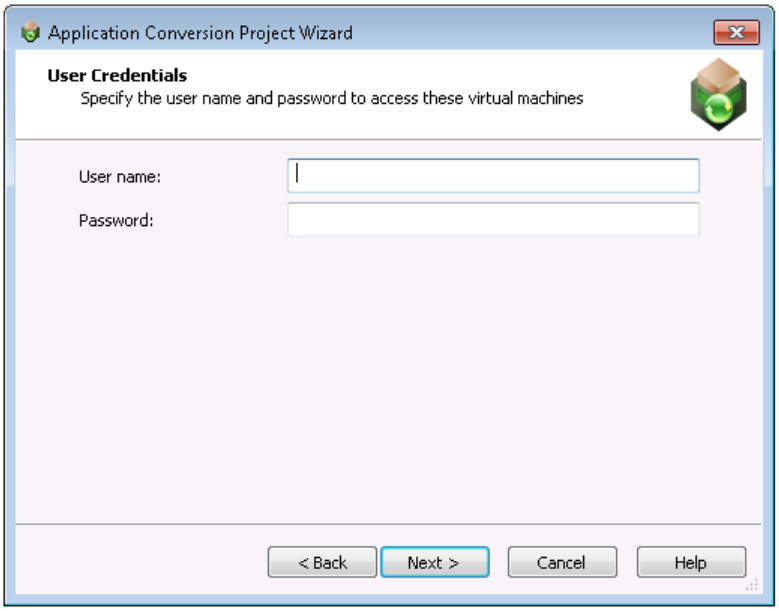


Figure 9-41: User Credentials Panel

The **User Credentials** panel includes the following options:

Table 9-35 • User Credentials Panel

Option	Description
User name	Enter the user name to use to access the virtual machines you selected on the **Select Virtual Machines** panel.
Password	Enter the password to use to access the virtual machines you selected on the **Select Virtual Machines** panel.

Important • *If the virtual machines you selected do not all use the same login credentials, you can add the appropriate credentials in the **Guest Username** and **Guest Password** properties on the Machines Tab after you have added the virtual machine.*

Initial Configuration Complete

The **Initial Configuration Complete** panel lists the packages and machines you have selected to add to your project. You can choose to either begin conversion or to close the wizard so that you can perform additional configuration of these packages and machines prior to conversion.

- **Virtualize packages with detected settings**—Select this option if you want to begin conversion of the selected packages using the selected virtual machines using the current settings.
- **Close wizard to configure packages and machines**—Select this option if you want to close this wizard and perform additional configuration of these packages and virtual machines on the **Packages** and **Machines** tabs prior to beginning conversion.

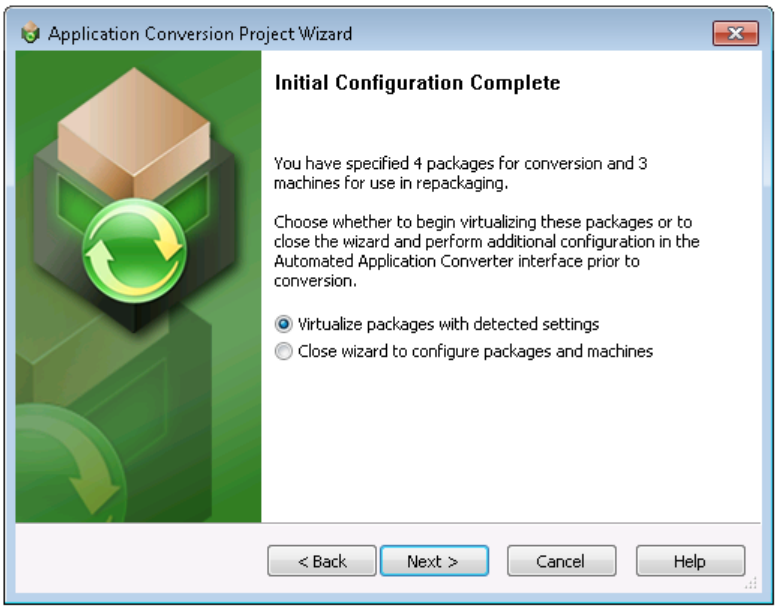


Figure 9-42: Initial Configuration Complete Panel

Select Output Formats

On the **Select Output Formats** panel, select the output formats you want to create and the output location for the packages.

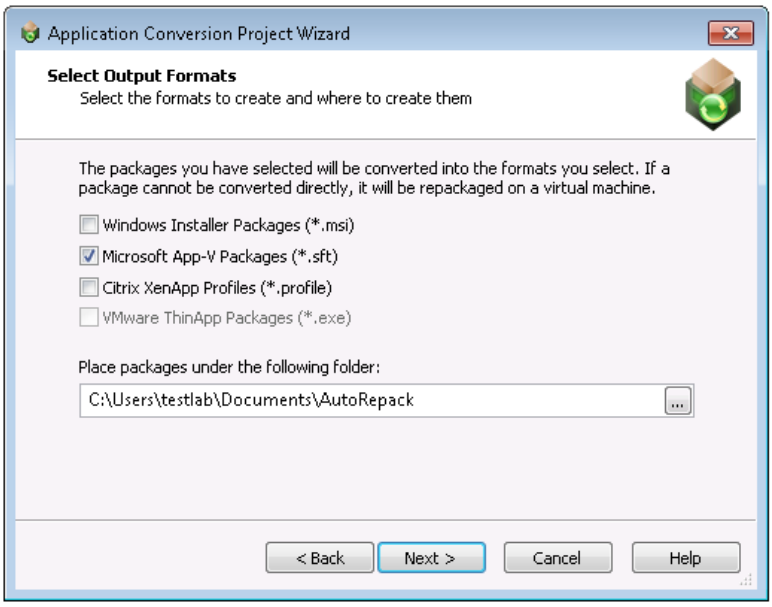


Figure 9-43: Select Output Formats Panel

The **Select Output Formats** panel includes the following options:

Table 9-36 • Select Output Formats Panel

Option	Description
Windows Installer Packages (*.msi)	Select this option to repackage the selected packages into Windows Installer packages (.msi).
Microsoft App-V Packages (*.sft)	Select this option to convert the selected packages to Microsoft App-V packages.
Citrix XenApp Profiles (*.profile)	Select this option to convert the selected packages to Citrix XenApp profiles.
VMware ThinApp Packages (*.exe)	Select this option to convert the selected packages to VMware ThinApp packages.
Place packages under the following folder	Select the location where you want to store the package output.

Automated Repackaging on Virtual Machines

On the **Automated Repackaging on Virtual Machines** panel, you specify which operating system platform you want to use to perform automated repackaging

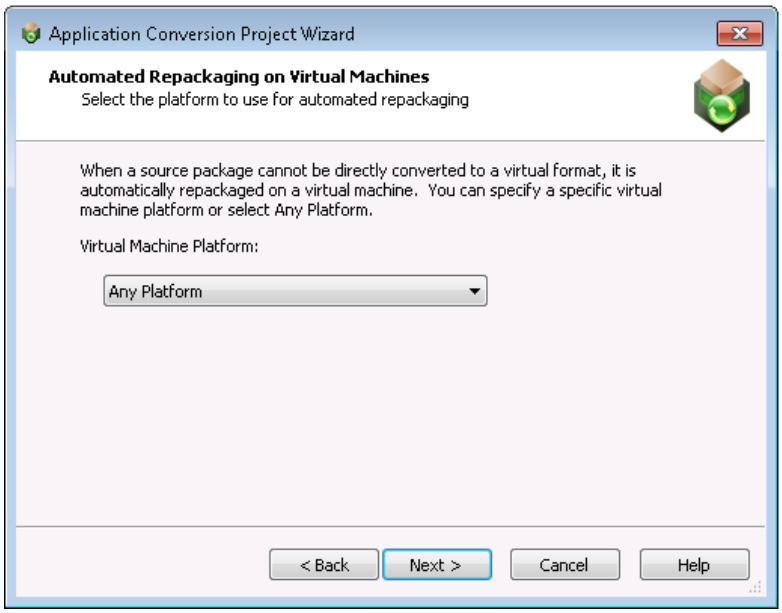


Figure 9-44: Automated Repacking on Virtual Machines Panel

On the **Automated Repackaging on Virtual Machines** panel, select one of the following options:

Table 9-37 •

Option	Description
Any Platform	The Automated Application Converter will use any of the virtual machines that you have selected on the **Machines** tab to perform automated repackaging, regardless of platform.
OS Platform	If you select a specific operating system, the Automated Application Converter will use only those virtual machines that you have selected on the **Machines** tab that are of the selected operating system to perform automated repackaging. ***Note*** *• When you select a virtual machine to add to the Automated Application Converter, you need to manually identify the operating system platform either on the* ***Select Virtual Machines*** *panel or by clicking in the* ***Platform*** *field on the* ***Machines*** *tab and making a selection from the list.*

Application Conversion Project Wizard Complete Panel

The **Application Conversion Project Wizard Complete** panel lists the virtual formats that your selected packages will be converted to, and the operating system platform of the virtual machine that will be used to perform repackaging if repackaging is required during conversion.

Click **Finish** to close the wizard and begin converting your packages.

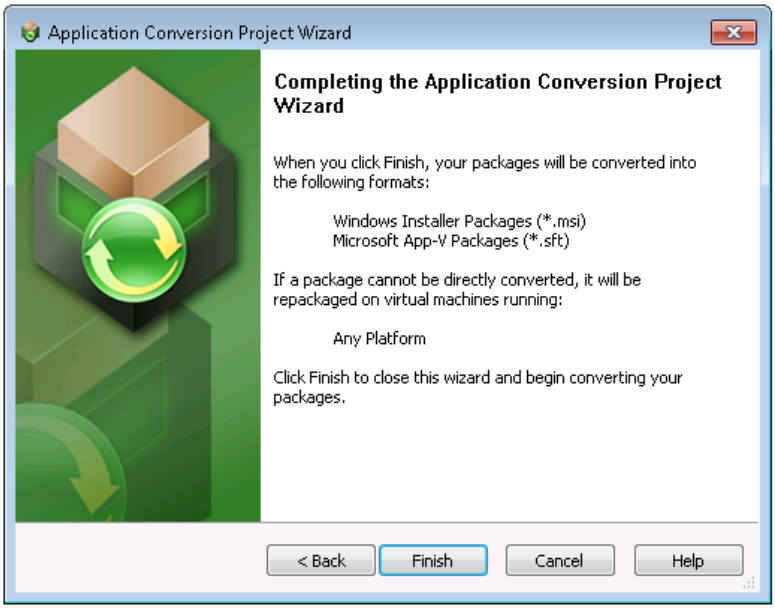


Figure 9-45: Application Conversion Project Wizard Complete Panel

Package Import Wizard

You can use the Package Import Wizard to select packages to convert to virtual formats. You can select packages from a specified AdminStudio Application Catalog, Microsoft Configuration Manager server, or from the file system of a local or network machine.

The Package Import Wizard, which is launched by clicking the **Add Packages** button on the Automated Application Converter **Packages** tab, consists of the following panels:

- Package Import Wizard Welcome
- Select Package Source
- Connect to an AdminStudio Application Catalog
- Connect to a Microsoft Configuration Manager Server
- Select Packages
- Selected Package List
- Package Import Wizard Complete

Note • *The main panels of the Package Import Wizard are also included in the end-to-end Application Conversion Project Wizard.*

Package Import Wizard Welcome

You can use the Package Import Wizard to select packages to convert to virtual formats. You can select packages from a specified AdminStudio Application Catalog, Microsoft Configuration Manager server, or from the file system of a local or network machine.

After you have added the packages, you can use the Virtual Machine Import Wizard to add virtual machines to use for automated repackaging, and then begin a conversion run using the Application Conversion Wizard.

Click **Next** to begin.

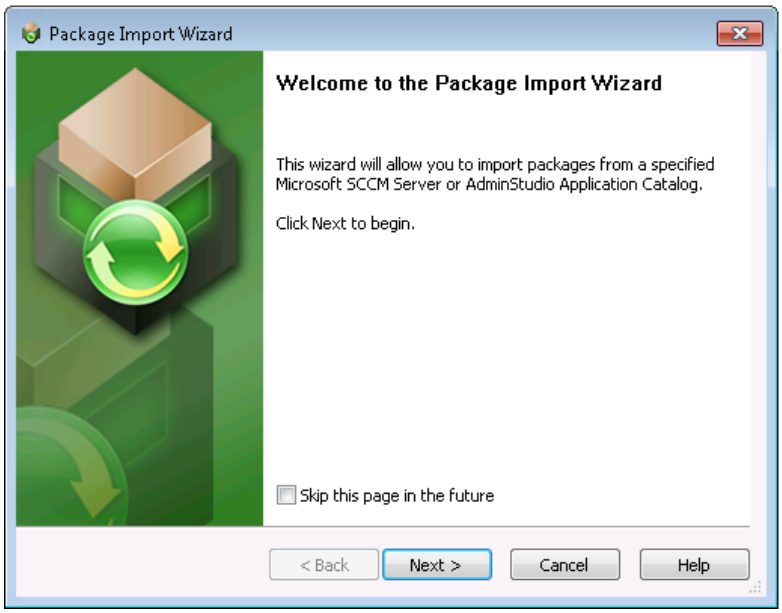


Figure 9-46: Package Import Wizard Welcome Panel

Package Import Wizard Complete

The **Package Import Wizard Complete** panel lists the number of packages you have added to your project for repackaging and virtualization. Click **Finish** to close the wizard and add these packages to your project.

To convert these packages to virtual applications, use the Application Conversion Wizard.

Figure 9-47: Package Import Wizard Complete Panel

Virtual Machine Import Wizard

The Virtual Machine Import Wizard consists of the following panels:

- Virtual Machine Import Wizard Welcome
- Select Virtual Machine Source
- Select Virtual Machines from a Microsoft Hyper-V Server
- Select Virtual Machines from a VMware ESX or ESXi Server
- Select Virtual Machines
- User Credentials
- Virtual Machine Import Wizard Complete

Note • *The main panels of the Virtual Machine Import Wizard are also included in the end-to-end Application Conversion Project Wizard.*

Virtual Machine Import Wizard Welcome

You can use the **Virtual Machine Import Wizard** to select virtual machines to use for automated repackaging. You can use virtual machines from a Microsoft Hyper-V Server, a VMware ESX or ESXi Server, or a local VMware Workstation.

After you have added the virtual machines, you can use the Package Import Wizard to add packages to convert, and then begin a conversion run using the Application Conversion Wizard.

Click **Next** to begin.

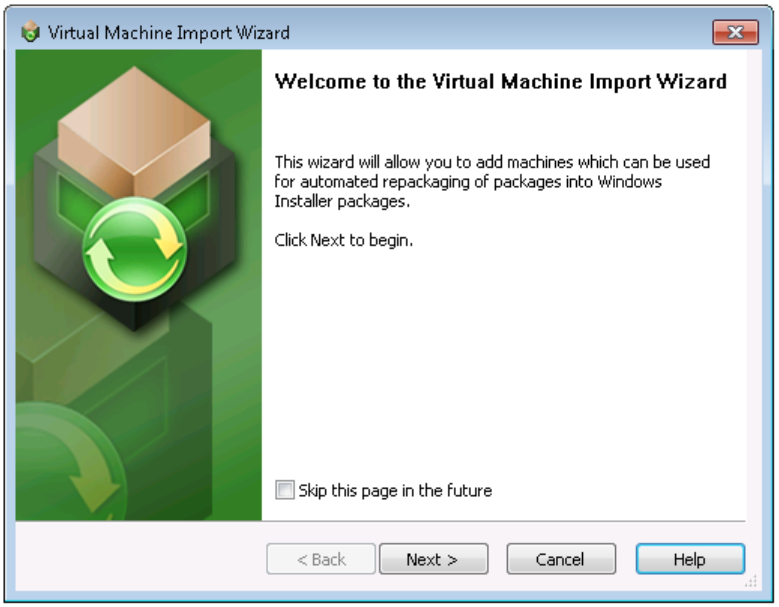


Figure 9-48: Virtual Machine Import Wizard Welcome Panel

Virtual Machine Import Wizard Complete

The **Virtual Machine Import Wizard Complete** panel lists the number of virtual machines you have added to your project for use in automated repackaging.

Click **Finish** to close the wizard and add these virtual machines.

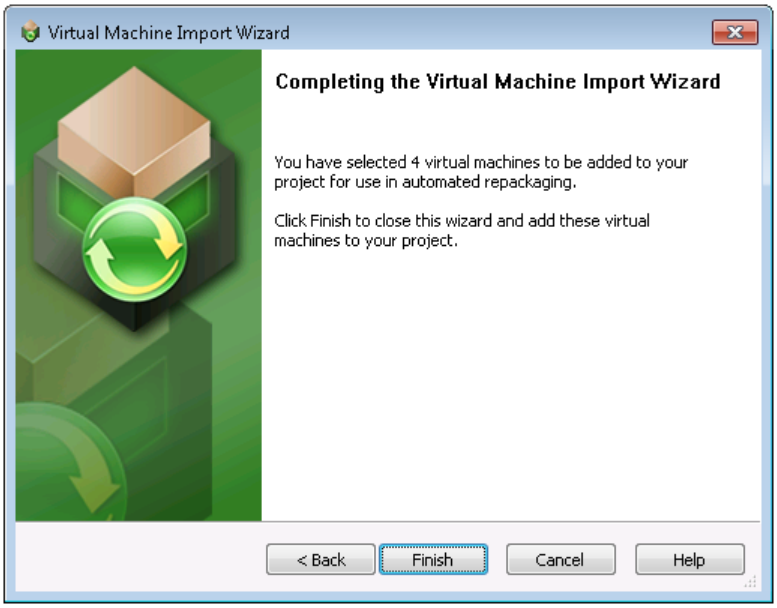


Figure 9-49: Virtual Machine Import Wizard Complete Panel

Application Conversion Wizard

The Application Conversion Wizard consists of the following panels:

- Application Conversion Wizard Welcome
- Select Output Formats
- Automated Repackaging on Virtual Machines
- Application Conversion Wizard Complete

Note • *The main panels of the Application Conversion Wizard are also included in the end-to-end Application Conversion Project Wizard.*

Application Conversion Wizard Welcome

You can use the **Application Conversion Wizard** to convert selected packages to virtual applications after you have used the Virtual Machine Import Wizard to add virtual machines to the project to use for automated repackaging and used the Package Import Wizard to add the packages you want to convert.

Click **Next** to begin.

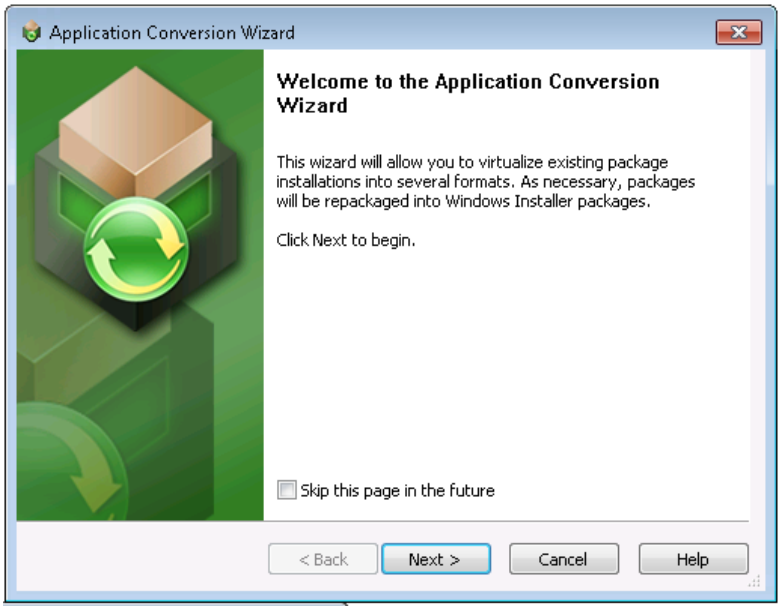


Figure 9-50: Application Conversion Wizard Welcome Panel

Application Conversion Wizard Complete

The **Application Conversion Wizard Complete** panel lists the virtual formats that your selected packages will be converted to, and the operating system platform of the virtual machine that will be used to perform repackaging if repackaging is required during conversion.

Click **Finish** to close the wizard and begin converting your packages.

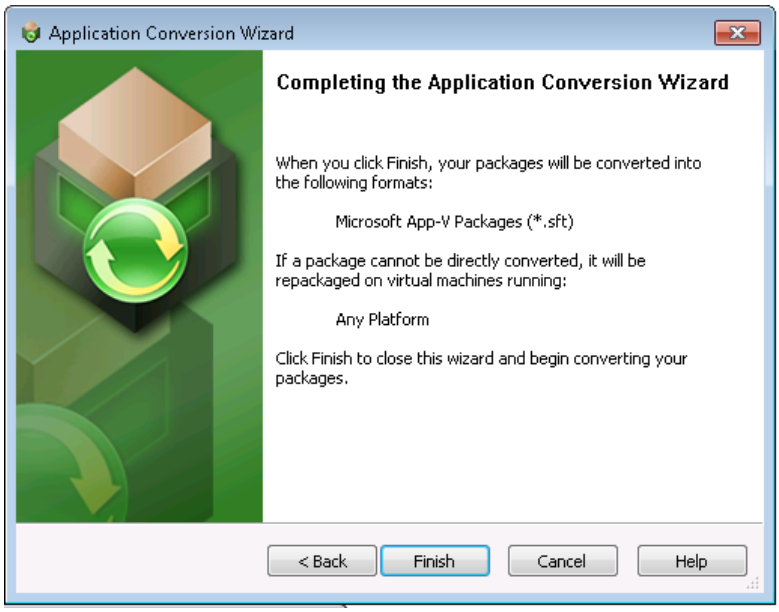


Figure 9-51: Application Conversion Wizard Complete Panel

Package Publish Wizard

After converting applications to virtual packages, you can publish them to a Microsoft Configuration Manager Server or AdminStudio Application Catalog using the **Package Publish Wizard**.

The **Package Publish Wizard** consists of the following panels:

- Package Publish Wizard Welcome
- Select Publish Target
- Connect to an AdminStudio Application Catalog
- Connect to a Microsoft Configuration Manager Server
- Select Destination Folder
- Select Group
- Select Package Types to Publish
- Package Publish Wizard Complete

Package Publish Wizard Welcome

You can use the Package Publish Wizard to publish the virtualized/repackaged packages you have generated using the Automated Application Converter to an AdminStudio Application Catalog or Microsoft Configuration Manager Server.

Click **Next** to continue.

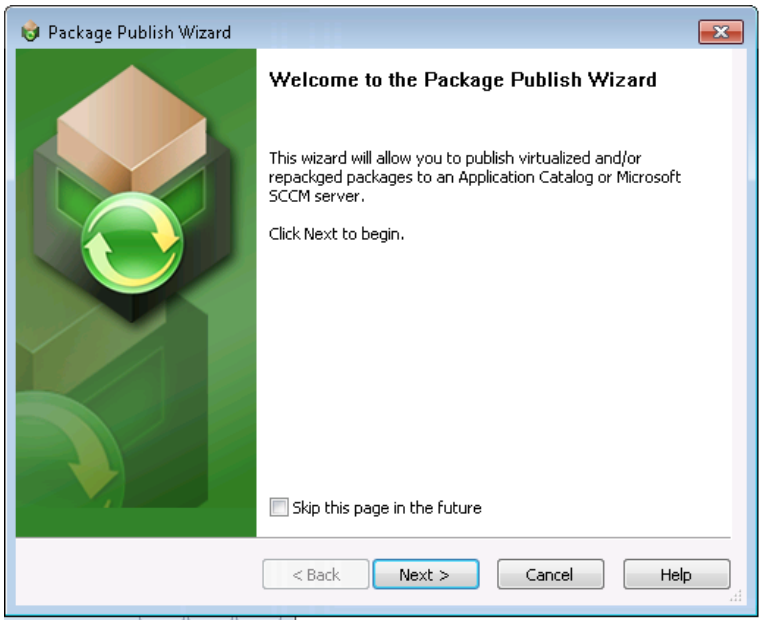


Figure 9-52: Package Publish Wizard Welcome Panel

Select Publish Target

On the **Select Publish Target** panel of the **Package Publish Wizard**, select the target location where you want to publish the selected virtual or repackaged packages. Select either **AdminStudio Application Catalog** or **Microsoft Configuration Manager** and click **Next** to continue.

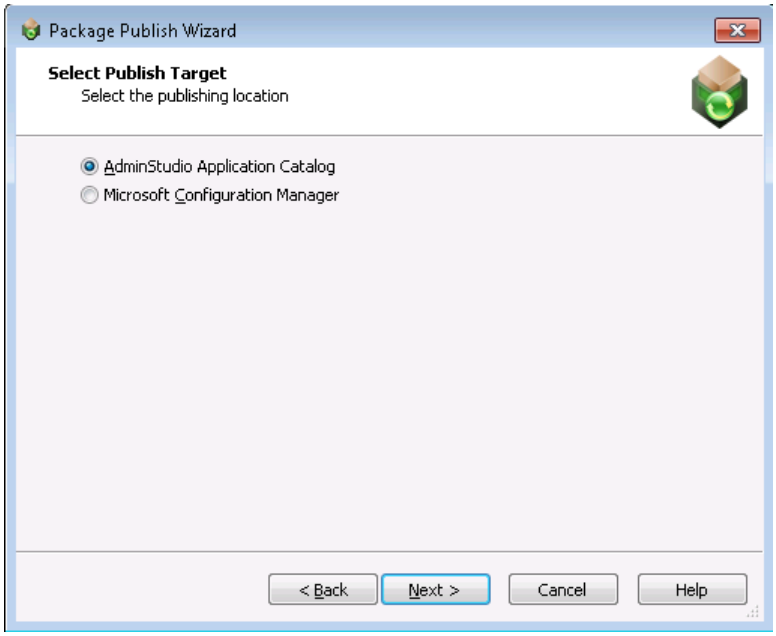


Figure 9-53: Select Publish Target Panel

Connect to an AdminStudio Application Catalog

On the **Connect to an AdminStudio Application Catalog** panel, which opens if you select **AdminStudio Application Catalog** on the **Select Publish Target** panel, you enter connection information to connect to an AdminStudio Application Catalog SQL database.

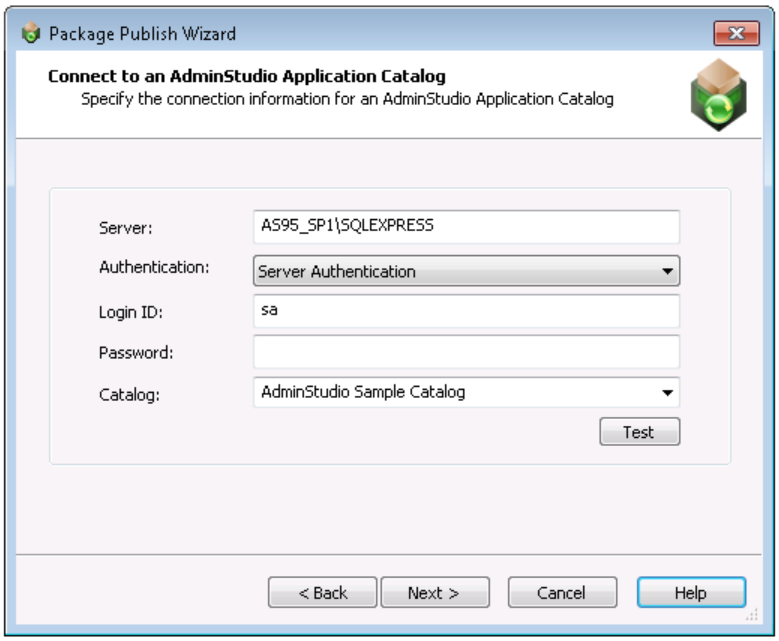


Figure 9-54: Connect to an AdminStudio Application Catalog Panel

On the **Connect to an AdminStudio Application Catalog** panel, enter the following information:

Table 9-38 • Connect to an AdminStudio Application Catalog Panel

Option	Description
Server	Enter the name of the SQL Server that you want to connect to.
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use SQL Server login identification to log into this Application Catalog. Then enter the appropriate **Login ID** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Application Catalog. ***Note*** • *After you successfully connect to an Application Catalog, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*

Table 9-38 • Connect to an AdminStudio Application Catalog Panel

Option	Description
Catalog	Enter the name of the existing AdminStudio Application Catalog database that you want to connect to

Connect to a Microsoft Configuration Manager Server

On the **Connect to a Microsoft Configuration Manager Server** panel, which opens if you select **Microsoft Configuration Manager** on the **Select Publish Target** panel, you enter connection information to connect to a Microsoft System Center Configuration Manager server.

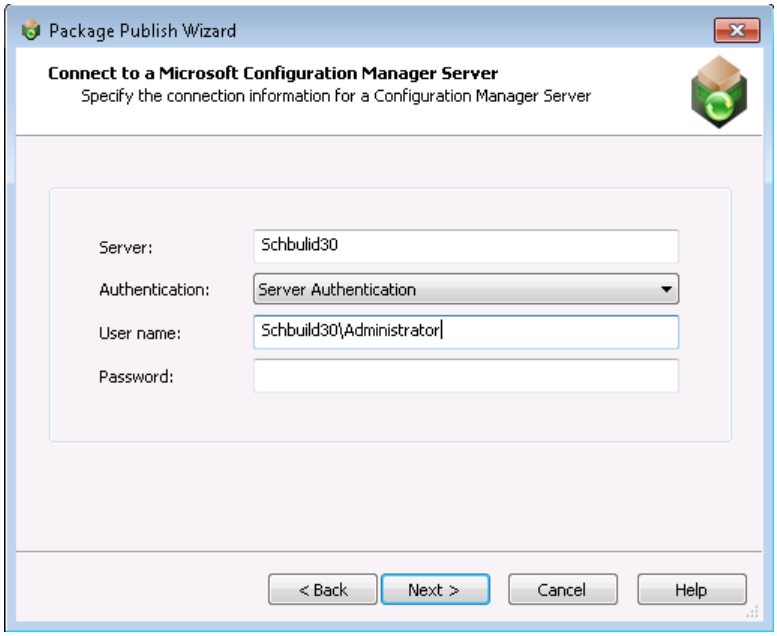


Figure 9-55: Connect to a Microsoft Configuration Manager Server Panel

On the **Connect to a Microsoft Configuration Manager Server** panel, enter the following information:

Table 9-39 • Connect to a Microsoft Configuration Manager Server Panel

Option	Description
Server	Enter the name of the Microsoft Configuration Manager Server that you want to connect to.
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server login identification to log into this server. Then enter the appropriate **User name** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Microsoft Configuration Manager Server. ***Note*** • *After you successfully connect to a Microsoft Configuration Manager Server, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*

Select Destination Folder

When publishing to a Microsoft Configuration Manager Server, the **Select Destination Folder** panel opens prompting you to select a location that the Microsoft Configuration Manager Server has access to where you want to publish the selected packages.

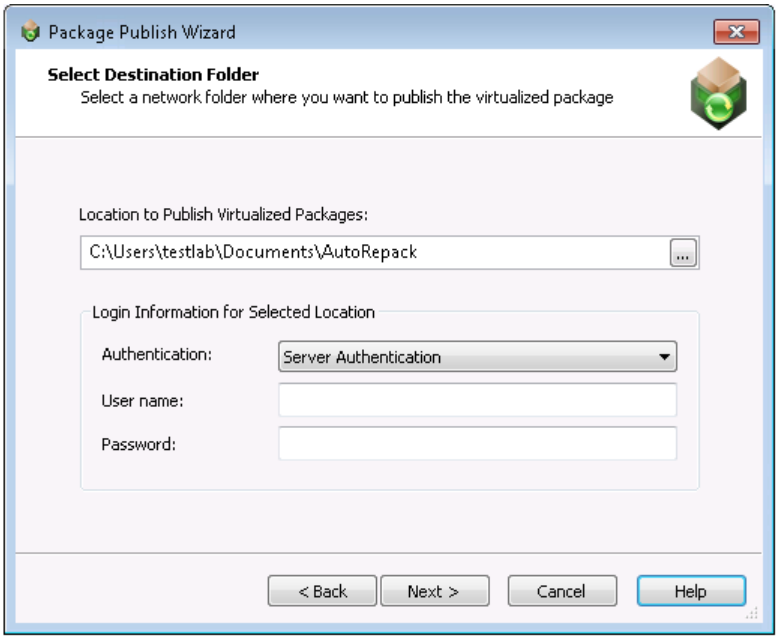


Figure 9-56: Select Destination Folder Panel

On the **Select Destination Folder** panel, enter the following information:

Table 9-40 • Select Destination Folder Panel

Option	Description
Location to Publish Virtualized Packages	Enter a target path, in UNC format (`\\Server\Share`), of the location where you want to publish the selected packages. Make sure that you enter a location that the Microsoft Configuration Manager Server has access to.
Authentication	Choose one of the following options: • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this location. • **SCCM Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server authentication (your Microsoft Configuration Manager Server login ID) to log into this location. • **Server Authentication**—Choose this option if you are publishing to an alternate file server that requires credentials. Then enter the appropriate **User name** and **Password**.

Select Group

When publishing to an AdminStudio Application Catalog, the **Select Group** panel opens, prompting you to specify the Application Catalog group where you want to publish the selected packages. You need to specify whether to attempt to publish the packages in their original source location or another specified location.

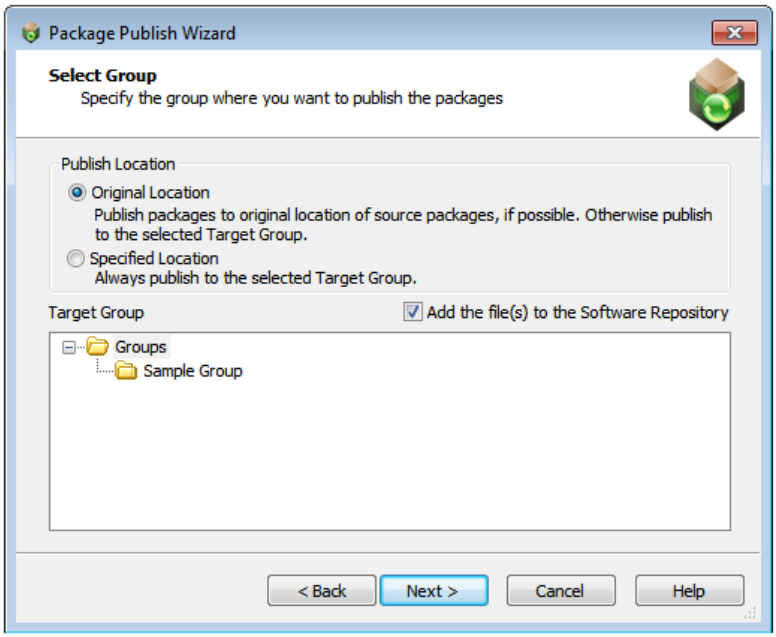


Figure 9-57: Select Group Panel

The **Select Group** panel includes the following options:

Table 9-41 • Select Group Panel Options

Option	Description
Original Location	Select this option if you want to publish the selected packages to the original location of their source package, if possible. If that location no longer exists, then publish the package to the location specified in **Target Group**.
Specified Location	Select this option if you want to publish the selected packages to the location specified in **Target Group**.
Target Group	If you selected the **Original Location** option, select the group where you want to publish the selected packages only if their original source location is unavailable. If you selected the **Specified Location** option, select the group where you want to publish the selected packages.
Add the file(s) to the Software Repository	Select this option to add the published files to the Software Repository.

Select Package Types to Publish

Select the package formats that you want to publish:

- **Repackaged MSI (*.msi)**
- **Microsoft App-V Packages (*.sft)**
- **Citrix XenApp Profiles (*.profile)**
- **VMware ThinApp Packages (*.exe)**

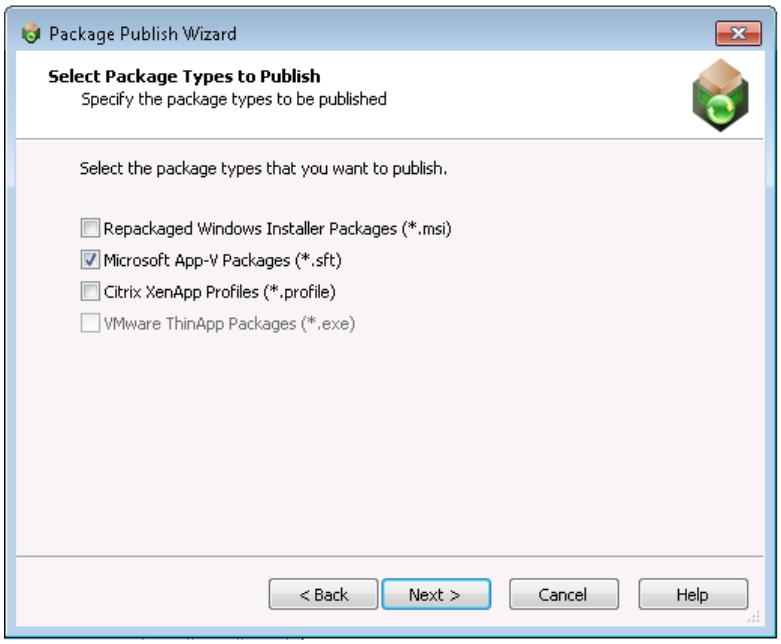


Figure 9-58: Select Package Types to Publish Panel

Package Publish Wizard Complete

The **Package Publish Wizard Complete** panel lists the number of packages you want to publish and the publish target location.

Click **Finish** to close the wizard and publish your packages.

Figure 9-59: Package Publish Wizard Complete Panel

Dialog Boxes

The Automated Application Converter provides the following dialog boxes:

- Browse for Folder Dialog Box
- Guest Agent
- Open Dialog Box
- Project Options Dialog Box
- Select Package Installation File Dialog Box
- Select Transform Dialog Box
- Select Virtual Machine Dialog Box
- Select Virtual Machine Image File Dialog Box

Browse for Folder Dialog Box

On the **Browse for Folder** dialog box, select the directory containing the packages to convert.

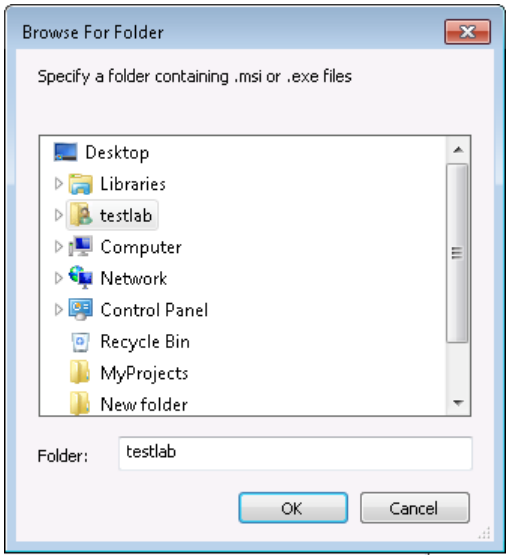


Figure 9-60: Browse for Folder Dialog Box

Select the directory that contains the Windows Installer files (`.msi`) and/or legacy package files (`.exe`) you want to convert and click **Open**. the Automated Application Converter searches the selected directory and its subdirectories to locate `.msi` and `.exe` files and adds them to the list on the **Select Packages** panel.

Important • *The Automated Application Converter uses specific rules to determine which packages in the selected directory and its subdirectories are added to the list on the* ***Select Packages*** *panel, and which of those files are automatically selected. See Automated Application Converter's Selection Rules When Adding Packages from a Directory for more information.*

Guest Agent

The Guest Agent (`GuestAgent.exe`) is a tool that is launched on a virtual image that enables the Automated Application Converter to manipulate the virtual machine in ways that may be unsupported by its automation APIs. In particular, this enables launching and monitoring the AdminStudio Repackager in an automated fashion.

Figure 9-61: Guest Agent Interface

Open Dialog Box

The **Open** dialog box opens when you select **Open** on the **File** menu or when you select the **Open existing project** option on the **Open Project** panel of the Application Conversion Project Wizard and then select **Browse for project file...** from the list.

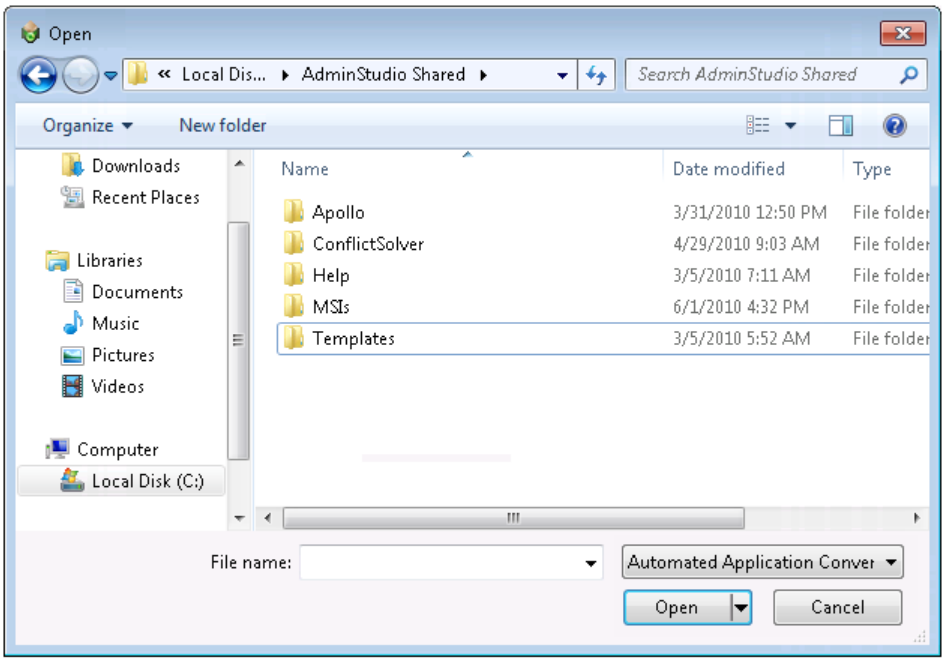


Figure 9-62: Open Dialog Box

Select an Automated Application Converter project file (`*.aacx`) and click **Open** to open the file.

MST Dialog Box

The **MST** dialog box, which opens when you click the Browse button in the **Transform** field in the **Properties** window of the **Packages** tab, lists the transforms that are associated with the selected Windows Installer package. Automated Application Converter automatically lists all of the `.mst` files located in the same directory as the selected `.msi` file.

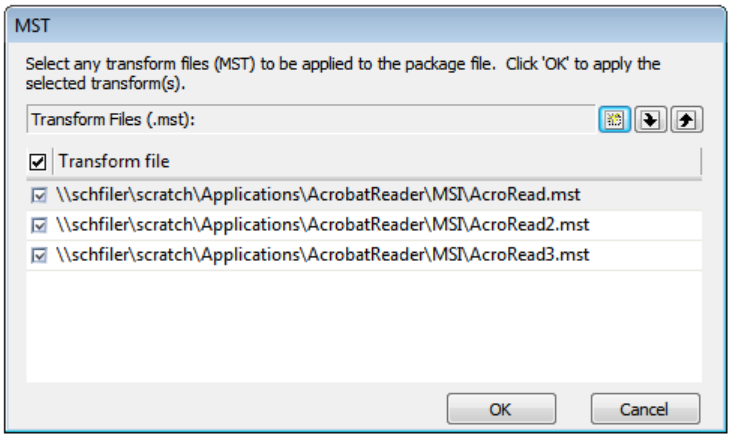


Figure 9-63: MST Dialog Box

Note *• You can also open the **MST** dialog box by clicking the Browse button in the **Transform** column of the package listing on the **Packages** tab, or by clicking the Browse button on the **Transform** column of the **Selected Package List** panel of the Package Import Wizard or the Application Conversion Project Wizard.*

On the **MST** dialog box, specify how transforms should be handled for the selected Windows Installer package:

- **Select transforms**—Select the transform (`.mst`) files that you want to import along with the Windows Installer package. If you do not want to import a selected `.mst` file, clear the selection.
- **Add additional transforms**—To add additional transforms that are not located in the same directory as the selected Windows Installer package, click the New button ( ) and browse to the location of the transform. If the package requires multiple transforms, you can repeat the procedure as necessary.
- **Order transforms**—If more than one transform is listed, use the up and down arrows to order the list of transforms in the order you want them applied.

Project Options Dialog Box

On the **Project Options** dialog box, which is opened by selecting **Options** on the **Tools** menu, you can specify project-wide default options.

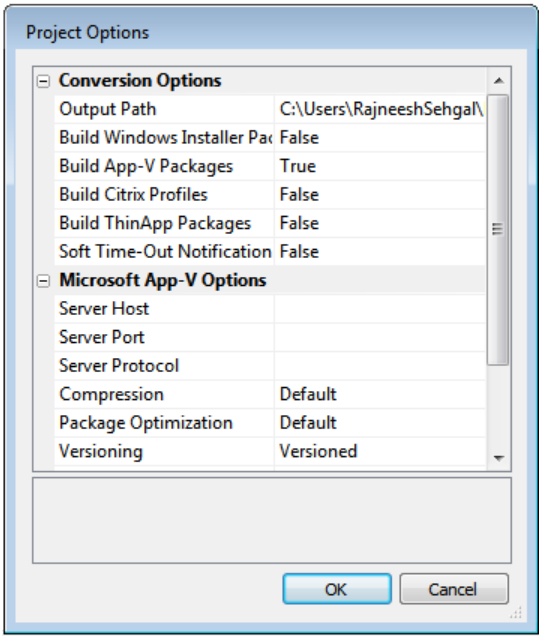


Figure 9-64: Project Options Dialog Box

The **Project Options** dialog box includes the following options:

Table 9-42 • Options on the Project Options Dialog Box

Option	Description
Output Path	Specify the default location that will populate the **Place packages under the following folder** field on the **Select Output Formats** panel of the Application Conversion Project Wizard and Application Conversion Wizard.
Build Windows Installer Packages **Build App-V Packages** **Build Citrix Profiles** **Build ThinApp Packages**	If this option is set to **True**, the output format will be selected by default on the **Select Output Formats** panel of the of the Application Conversion Project Wizard and Application Conversion Wizard. If it is set to **False**, the output format will not be selected by default.
Soft Time-Out Notification	Set this option to **True** if you want AdminStudio to automatically send you an email notification when a soft time-out is encountered while using Automated Application Converter to repackage an application on a virtual machine. ***Note*** • *To enable email notification, you need to configure your SMTP notification settings on the **Notification Settings** tab of the AdminStudio **Options** dialog box.*

Table 9-42 • Options on the Project Options Dialog Box

Option	Description
Server Host	Specify the default App-V Server Host value, which is the virtual application server or the load balancer in front of a group of virtual application servers that will stream the software package to an App-V Application Virtualization Desktop Client. You must complete this item to create a sequenced application package, but you can change from the default `%SFT_SOFTGRIDSERVER%` environment variable to the actual hostname or IP address of a virtual application server. ***Note*** • *If you choose not to specify a static hostname or IP address, on each Application Virtualization Desktop Client you must set up an environment variable called `SFT_SOFTGRIDSERVER`.* • *Its value must be the hostname or IP address of the virtual application server or load balancer that is this client's source of applications.* • *You should make this environment variable a system variable rather than a user variable.* • *Any Application Virtualization Desktop Client session that is running on this computer during your assignment of this variable must be closed and then opened so that the resumed session will be aware of its new application source.*
Server Port	Specify the default App-V Server Port value, which is the port on which the virtual application server or the load balancer will listen for an Application Virtualization Desktop Client's request for the package. • This information is required to create a package, but you can change it. • The default port is 554.
Server Protocol	Specify the default App-V Server Protocol value, which is the protocol that will stream the sequenced application package from a virtual application server to an Application Virtualization Desktop Client. The following protocols are available: • **RTSP**—The default, it specifies that the Real-Time Streaming Protocol controls the exchange of virtualization-enabled applications. • **RTSPS**—Specifies that the Real-Time Streaming Protocol with Transport Layer Security controls the exchange of a sequenced application package. • **FILE**—Specifies that the sequenced application will be streamed from a file share. • **HTTP**—Specifies that Hypertext Transport Protocol controls the exchange of a package. • **HTTPS**—Specifies that Secure Hypertext Transport Protocol controls the exchange of a package.

Table 9-42 • Options on the Project Options Dialog Box

Option	Description
Compression	Specify the default App-V Compression value by selecting **Default**, **Compressed**, or **Uncompressed**.
Package Optimization	Specify the default App-V Package Optimization value, which specifies whether to optimize package for offline use (entire package is included in Feature Block 1) by selecting **Offline**, or streaming use (only shortcut targets are included in Feature Block 1) by selecting **Stream**.
Versioning	Specify the default App-V Versioning value. • Select **Versioned** to append the package version to the SFT file name. • Select **Unversioned** to leave the package version off of the SFT file name. • Select **Default** to use the setting that is defined on the **Project Options** dialog box.
Diagnostics	Specify the default Diagnostic tools to include in App-V packages by selecting **True** or **False**. • **Launcher**—Set this to **True** to include the App-V Application Launcher with each App-V application that you build. You can use the App-V Application Launcher to test a newly built App-V application before moving it to a deployment server. • **Filesystem**—Set this to **True** to if you want to include the Windows Command Prompt application with your App-V application so that you can browse the virtual file system at runtime from within the virtual environment. If this option is selected, a file named `Virtual File System.osd` will be created in the App-V Package folder, which can be used to display the files and folders within the virtual environment. You can use `Virtual File System.osd` to view the existing files and folders on the computer plus the files and folders for the virtual package. • **Registry**—Set this to **True** to if you want to include the Registry Editor (`regedit.exe`) with your App-V application so that you can browse the registry at runtime from within the virtual environment. If this option is selected, a file named `Virtual Registry.osd` will be created in the App-V Package folder, which can be used to display the registry within the virtual environment. You can use `Virtual Registry.osd` to view the existing registry on the computer plus the registry for the virtual package.
Runtime Drive	Specify the App-V client runtime drive.
Create Installation MSI	Specify whether to create a wrapper MSI that can install this App-V package.
Compress Installation MSI	Specify whether to compress the wrapper MSI.

Select Package Installation File Dialog Box

On the **Select Package Installation File** dialog box, select the installation file (`.msi or .exe`) or installation script (`*.vbs`, `*.bat`, `*.cmd`, or `*.ps1`) that you want to add to your project for conversion to a virtual application.

Note • *You can use installation scripts to run more complex installation scenarios.*

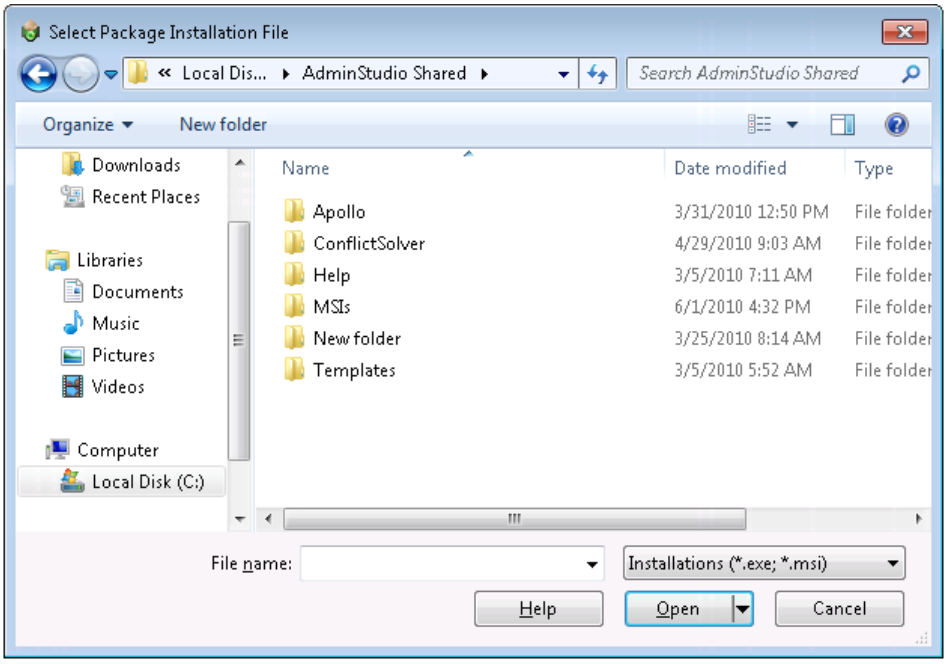


Figure 9-65: Select Package Installation File Dialog Box

Select Transform Dialog Box

On the **Select Transform** dialog box, which opens when you click in the **Transform** column/property on the **Packages** tab, you can select a transform file (`.mst`) to modify or install a Windows Installer package silently.

Note • *While the **Transform** property on the **Packages** tab can contain a semicolon-delimited list of transforms, when you browse to the transform file location using the **Select Transform** dialog box, you are only able to select one transform file. To include multiple transforms with a package, rather than browsing to the transform file location, you need to manually edit the **Transform** property on the **Packages** tab to enter multiple transform files, separated by a semicolon.*

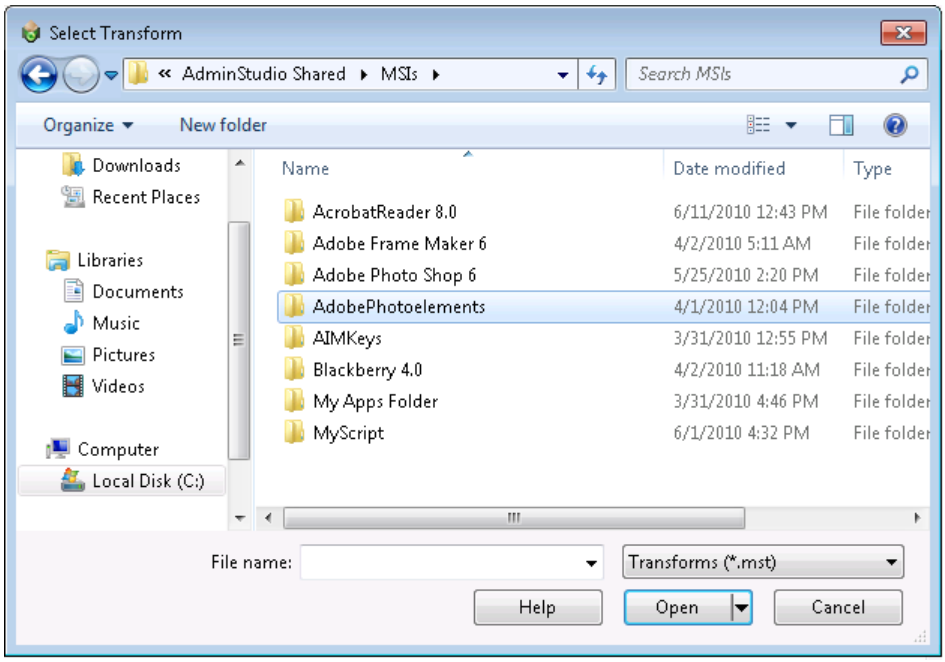


Figure 9-66: Select Transform Dialog Box

Select Virtual Machine Dialog Box

On the **Select Virtual Machine** dialog box, which opens when you select a package on the **Packages** tab and then select **Launch Package for Testing** from the context menu, you select the virtual machine that you want to use to test the selected package.

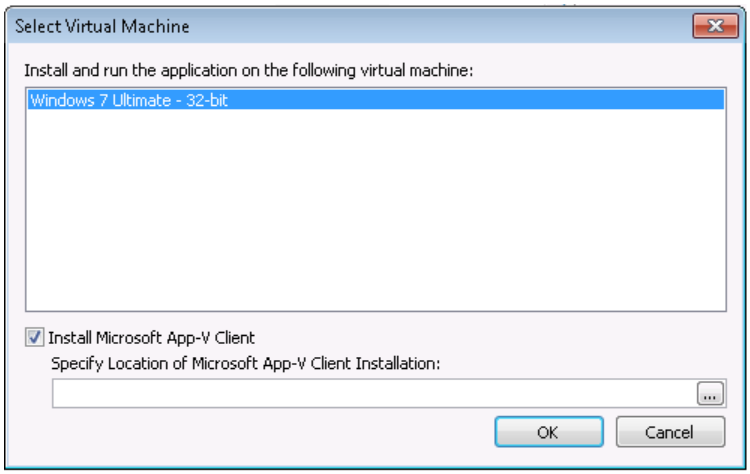


Figure 9-67: Select Virtual Machine Dialog Box

The following options are available:

Table 9-43 • Select Virtual Machine Dialog Box

Option	Description
Install and run the application on the following virtual machine	Select the virtual machine that you would like to use the test the selected package.
Install Microsoft App-V Client	If you are testing an App-V package, select this option to instruct the Automated Application Converter to install the App-V client on the selected virtual machine.
Specify Location of Microsoft App-V Client Installation	If you have selected the **Install Microsoft App-V Client** option, specify the location of the App-V client installation. Make sure that Automated Application Converter machine has access to the specified location.

Select Virtual Machine Image File Dialog Box

When you click **Browse Files** on the **Select Virtual Machines** panel, the **Select Virtual Machine Image File** dialog box opens, prompting you to select a VMware Workstation virtual machine image.

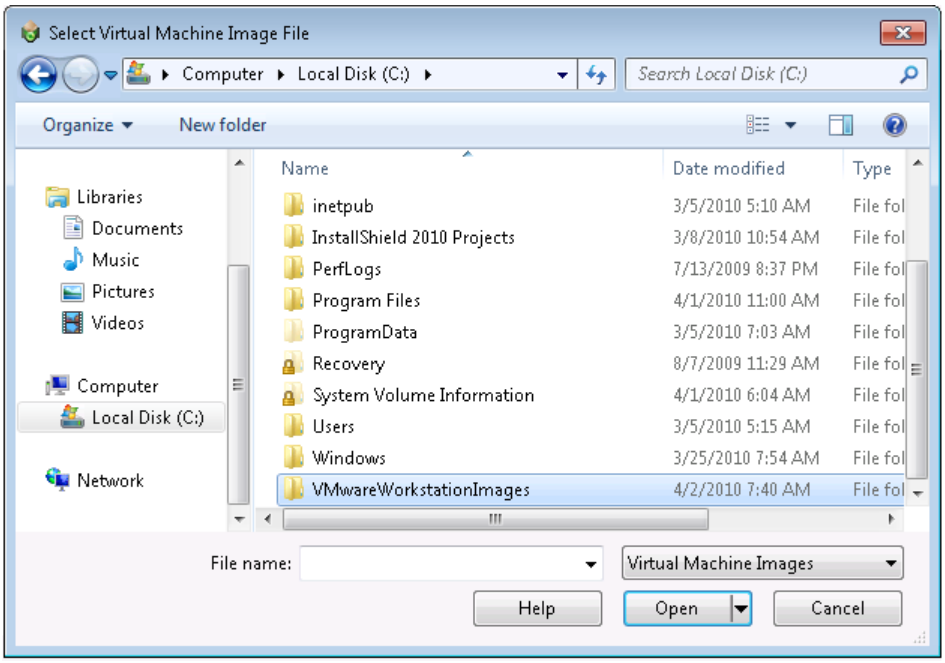


Figure 9-68: Select Virtual Machine Image File Dialog Box

Command Line Support

You can choose to run an Automated Application Converter project file via command line using the following command:

```
aacx.exe projectname.aacx
```

where `projectname.aacx` is the project file to load and execute. Results are displayed in console mode.

You can also use the following command line parameters to override project file settings:

Table 9-44 • Command Line Parameters

Parameter	Description

/create	Use the `/create` parameter to indicate the virtual formats to create, such as:
	`aacx.exe /create formattype projectname.aacx`
	where *formattype* can be any of the following (case-insensitive):
	`AppV`
	`Citrix`
	`ThinApp`
	`MSI`
	and *projectname.aacx* is the project file to load and execute.
	For example, to create an App-V package, enter:
	`aacx.exe /create AppV myproject.aacx`
	Multiple create commands can be specified in the same command line. For example, to create all virtual formats, enter:
	`aacx.exe /create AppV /create Citrix /create ThinApp myproject.aacx`
	Note • *Settings made using the `/create` parameter override the selections you made on the **Select Output Formats** panel, which are saved in the project file.*

/help
/?
	To view command line help, enter either of the following:
	`aacx.exe /?`
	`aacx.exe /help`

/log	To create a unicode text file to contain output messages, enter the `/log` parameter followed by an output file name: `aacx.exe /log output.txt myproject.aacx` ***Note*** • *These are the same output messages that would appear in the Output window when using the Automated Application Converter interface,*

Table 9-44 • Command Line Parameters (cont.)

Parameter	Description
/**options**	To specify an alternate `options.ini` file for repackaging with Automated Application Converter, enter the `/options` parameter followed by the path to the `options.ini` file that you want to use, such as: `/options=C:\options.ini` Using this option enables you to specify a different `options.ini` file when repackaging with Automated Application Converter than the `options.ini` file that you use when performing standard repackaging with Repackager (which could have custom options in it). The specified `options.ini` file will be copied to the guest image during repackaging and will overwrite the default Repackager `options.ini` file.
/**outdir**	To override the output directory for built and converted packages that was set in the project file on the **Select Output Formats** wizard panel, use the `/outdir` parameter: `aacx.exe /outdir "C:\output\aacxoutput" myproject.aacx` where `C:\output\aacxoutput` is the name of the directory that will contain the output.
/**report**	To specify the name of the HTML report that is generated after conversion, use the `/report` parameter: `aacx.exe /report reportname.html myproject.aacx`
/**showreport**	To specify the name of the HTML report that is generated after conversion and to automatically display that report, use the `/showreport` parameter: `aacx.exe /showreport reportname.html myproject.aacx`

Table 9-44 • Command Line Parameters (cont.)

<table>
<tr><th>Parameter</th><th colspan="2">Description</th></tr>
<tr><td>/vmplatform</td><td colspan="2">To specify the platform to use when performing automated repackaging, overriding the VMs selected in the project file, use the <code>/vmplatform</code> parameter:

<code>aacx.exe /vmplatform platformvalue proj.aacx</code>

where platformvalue is constructed from a version integer using the formula of <code>MajorVersion * 100 + MinorVersion</code> of the operating system (such as 600 for Windows Vista), followed optionally by s (for server) and/or x64 (for 64-bit). Examples are below.</td></tr>
<tr><td></td><td>Windows Vista 32-bit</td><td><code>aacx.exe /vmplatform 600 myproject.aacx</code></td></tr>
<tr><td></td><td>Windows Vista 64-bit</td><td><code>aacx.exe /vmplatform 600x64 myproject.aacx</code></td></tr>
<tr><td></td><td>Windows Server 2008 R2 64-bit</td><td><code>aacx.exe /vmplatform 601sx64 myproject.aacx</code></td></tr>
<tr><td></td><td>All enabled machines</td><td><code>aacx.exe /vmplatform any myproject.aacx</code></td></tr>
<tr><td></td><td colspan="2">
Note • The version integer described above is similar to the Windows Installer VersionNT property. See Operating System Property Values on the MSDN Web site.</td></tr>
</table>

Specifying Global Default Virtual Conversion Settings

In addition to the settings that can be specified on the Project Options Dialog Box, a default value can be specified for any virtual conversion setting that would normally be stored in the `ISVirtualPackage` table by editing the `settings.xml` file. The global value is used if no project-specific value is found.

To configure these global default values, locate the `settings.xml` file installed with InstallShield Editor and AdminStudio Repackager, and then find the `<Properties>` subelement of the `<Virtualization>` element:

```
<Virtualization>
    <Properties>
        <Property Name="AppVRuntimeDrive" Value="G:" />
        <Property Name="AppVServerURLPath" Value="%PackageName%_v%PackageVersion%" />
    </Properties>
</Virtualization>
```

To define a default value for any of the properties in the `ISVirtualPackage` table, create a `<Property>` in the `<Properties>` element and set a value. In the examples above, the `AppVRunTimeDrive` property is set to a default value of `G:`, and the `AppVServerURLPath` property is set to a default value of `%PackageName%_v%PackageVersion%`.

The following three replaceable parameters are only valid for the `AppVServerURLPath` property:

- **%PackageName%**—Name of the virtual package (which normally corresponds to the MSI `ProductName`).
- **%PackageVersion%**—Version number. (Each new upgrade increments this number.)
- **%PackageVersionedName%**—This is the `%PackageName%` for version one packages, and `%PackageName%_v%PackageVersion%` otherwise.

Troubleshooting

This section includes information to help you resolve typical problems that you might encounter when using the Automated Application Converter. The following sections are included:

- First Things to Check
- Problems and Solutions
- Best Practices for Optimal Performance
- How to Test a Virtual Machine
- Automated Application Converter Error Messages

First Things to Check

If you encounter a problem when performing package conversion, first scan this table to review of list of the most likely causes for conversion failure.

Table 9-45 • Most Likely Causes of Errors

Cause of Error	Resolution
Did not run the Virtual Machine Preparation Tool	On each virtual machine that you are going to use to perform automated repackaging, you need to run the Virtual Machine Preparation Tool, an application that will enable automatic login. See Preparing Your Virtual Machines for Use With the Automated Application Converter. ***Important*** • *If you do not run the Virtual Machine Preparation Tool on the virtual machines you want to use, the Automated Application Converter will be unable to connect to them.*
Did not install VMware VIX API	If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation 6.5 or later), you need to have the VMware VIX API installed on the same machine as the Automated Application Converter. See VMware VIX API Requirement.
Snapshot does not exist on the virtual image	After you run the Virtual Machine Preparation Tool on a virtual machine, you need to shut it down and create a snapshot. This enables the Automated Application Converter to revert the virtual image to a clean state after each repackaging run. See Taking a Snapshot. ***Important*** • *If a snapshot does not exist on the virtual machine, not only will repackaging on that virtual machine fail, but you will also be unable to use that virtual machine to perform testing (as described in Launching Packages for Testing).*

Table 9-45 • Most Likely Causes of Errors

Cause of Error	Resolution
Name of Snapshot on virtual image is not identified properly	If your virtualization technology supports named snapshots, you should name the snapshot `AutoRepack_Base`, which is the default name that the Automated Application Converter will be looking for. If you assign a snapshot name other than `AutoRepack_Base`, after you add the virtual machine to the Automated Application Converter, you need to specify that snapshot name in the **Snapshot Name** property in the **Properties** window of the **Machines** tab for that machine. See Editing Virtual Machine Properties on the Machines Tab. ***Important*** • *If the snapshot on the virtual machine is not identified properly in the Automated Application Converter, not only will repackaging on that virtual machine fail, but you will also be unable to use that virtual machine to perform testing (as described in Launching Packages for Testing).*
ThinApp client is not installed	If you choose to build ThinApp applications, AdminStudio will convert the package installation into a format compatible with ThinApp. However, the ThinApp build process requires the availability of certain ThinApp tools. As a prerequisite to building a ThinApp application from AdminStudio, you must have installed ThinApp and accepted any and all license agreements. For more information, see ThinApp on the VMware Web site.
Password of the virtual image has changed	When you add a virtual image to a project, you are prompted for the user name and password to logon to that machine. If you entered an incorrect password or if the password has recently changed, you need to edit that machine's **Guest Password** property on the **Machines** tab. See Editing Virtual Machine Properties on the Machines Tab.
Virtual machine is corrupted, or cannot be launched	If the Automated Application Converter is attempting to connect to a virtual machine that is corrupt or cannot be launched, conversion will fail. To make sure that your virtual machines are in proper working order, attempt to launch them manually (outside of the Automated Application Converter) using the configuration tool of the virtual technology.
Virtual machine does not have network connectivity	In order for the Automated Application Converter to use a virtual machine, the virtual machine must have connectivity to your network. From the host machine, try to manually browse to the C drive of the virtual machine by entering the following address: `\\virtual_machine_name\C$`

Table 9-45 • Most Likely Causes of Errors

Cause of Error	Resolution
Repackaging is taking a very long period of time	If the repackaging of a package is taking a very long period of time, you may want to verify that the value for that package's **Compressed** property is correct. If a package is in a directory that contains many other applications, and its **Compressed** property is set to **True**, the Automated Application Converter knows that only that one file needs to be copied to the virtual machine for repackaging. However, if the **Compressed** property set is set to **False**, there is no way to determine which of the files in that directory belong to the package, so all of the files in the directory must be copied to the virtual machine before repackaging can start. See Editing Package Properties on the Packages Tab. ***Tip*** • *It is recommended that each package be placed in its own directory to avoid problems such as this one.*

Problems and Solutions

The following chart lists some typical problems that you might encounter when using the Automated Application Converter and some suggested solutions.

Table 9-46 • Solutions to Common Problems

Problem	Possible Causes	Solution
Cannot connect to a virtual machine	Virtual machine has not been prepared.	Verify that the Virtual Machine Preparation Setup (`VMCfg.exe`) was run on the virtual machine to enable automatic login. See Running the Virtual Machine Preparation Setup for instructions. ***Tip*** • *A quick way to determine if the Virtual Machine Preparation Tool has been run on a virtual image is the presence of the `GuestAgent.exe` file on the root of the `C:` drive.*
	Specified **Guest Username** or **Guest Password** property is not specified correctly.	When you add a virtual machine to the Automated Application Converter, you specify the **User name** and **Password** on the **User Credentials** panel. If you entered an incorrect value or if one of these values has changed, you will be unable to connect to the virtual machine. Open the **Machines** tab and verify that the values in the **Guest Username** and **Guest Password** properties for the virtual machine are correct. ***Note*** • *When using a domain account, do not include the domain name in the **Guest Username** property.*
Cannot connect to a Windows 7 or Windows Server 2008 virtual machine	User Account Control (UAC) settings on a Windows 7 or Windows Server 2008 virtual machine could be causing problems during auto-login.	Make sure that you run the Virtual Machine Preparation Setup on the virtual machine to disable UAC. See Running the Virtual Machine Preparation Setup.

Table 9-46 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Unable to add packages from an AdminStudio Application Catalog or a Microsoft Configuration Manager Server	User does not have required login and/or view permissions on the Application Catalog or Configuration Manager Server.	Try to manually view packages on a Microsoft Configuration Manager Server or AdminStudio Application Catalog to see if you have the required view permissions. • **Microsoft Configuration Manager Server**—Use Configuration Manager Console to view packages on the Microsoft Configuration Manager Server that you were having trouble with. • **Application Catalog**—Use AdminStudio Application Manager to view the packages in the Application Catalog. Also, if using an AdminStudio Application Catalog with the Software Repository, make sure you have view permission to the Software Repository location of that Application Catalog.
Cannot connect to a Microsoft Configuration Manager Server or an AdminStudio Application Catalog	There are connection issues between domains due to user name.	Try to manually connect to the Application Catalog or Microsoft Configuration Manager Server to make sure that you are using the correct credentials or that there are no other networking issues.

Table 9-46 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Unable to publish packages to an AdminStudio Application Catalog or a Microsoft Configuration Manager Server	User does not have required permission to import/publish packages.	Try to manually publish and/or import a package to Microsoft Configuration Manager Server or to AdminStudio Application Catalog to see if you have the required permissions: • **Microsoft Configuration Manager Server**—Use the Configuration Manager Console to manually publish a package to the Microsoft Configuration Manager Server that you were having trouble with. This enables you to determine if you have import/publish permission. • **Application Catalog**—Use AdminStudio Application Manager Import Wizard to manually import a package into the Application Catalog. This enables you to determine if you have import permission. Also, if using an AdminStudio Application Catalog with the Software Repository, make sure you have write permission to the Software Repository location of that Application Catalog.
Cannot connect to a Microsoft Hyper-V Server or a Microsoft Configuration Manager Server	DCOM configuration settings need to be adjusted.	Adjust the DCOM settings, as described in this MSDN article, Connecting to WMI on a Remote Computer: http://msdn.microsoft.com/en-us/library/aa389290%28VS.85%29.aspx

Table 9-46 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Copy errors are generated during package conversion	User does not have permission to the **Output Path** location specified in the Automated Application Converter.	To make sure that you have permission to the Automated Application Converter output directory, perform the following steps: ***To specify Output Path directory:*** 1. Open the Automated Application Converter. 2. Select **Options** on the **Tools** menu. The **Project Options** dialog box opens. 3. Locate the **Output Path** setting under **Conversion Options**. By default, the path is `C:\Users\[UserName]\Documents\AutoRepack`. 4. Browse to that location using Windows Explorer to and attempt to copy a file to that location. 5. If you are unable to copy a file to that location, change the **Output Path** to a location that you do have write permission on.
Package status changes to soft or hard timeout	The installation is requesting user input.	In order for the Automated Application Converter to perform automated repackaging, packages must support silent installation mode and the silent installation mode command line parameters must be specified in the **Command Line** property for each package. To resolve this problem, open the **Packages** tab and make sure that the package's **Command Line** property contains the command line parameters to run the installation silently. ***Note*** • *The Automated Application Converter automatically populates the **Command Line** property for Windows Installer (`.msi`) packages that you add to the project. However, for non-MSI packages, you must manually enter the **Command Line** property.* If the **Command Line** property is specified, but you are still receiving a timeout error, you may want to try to manually run the installation using the command line parameters to make sure that the installation is not requesting any user input.

Table 9-46 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
App-V application does not launch	VM does not have App-V client installed	If you are able to connect to the virtual machine, but the App-V application will not launch, make sure that the App-V client is installed on the virtual machine. Also, make sure that App-V file streaming is enabled on the virtual machine. ***Note*** • *When you select **Launch Package for Testing**, you are prompted to **Install the Microsoft App-V Client** on the **Select Virtual Machine** panel. See Launching Packages for Testing.*
Cannot build a ThinApp package	ThinApp application is not installed.	To create a ThinApp application, you are required to have ThinApp installed on the same machine as the Automated Application Converter.
Cannot connect to Hyper-V server	Hyper-V configuration tools are not installed.	Make sure that the Hyper-V configuration tools are installed on the Hyper-V Server's Hyper-V Microsoft Management Console.
Virtualization Readiness status of a package is "Unknown" (?)	Location of source files is no longer accessible.	When you originally added the package to your Automated Application Converter project, you were able to access the source files, but now the source files are either no longer there or you no longer have permission to access them.

Table 9-46 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Repackaging is taking a long time	It is taking a long time to copy the files required for repackaging to the virtual machine and back to the host machine.	Try to copy the files manually from the host machine to the virtual machine to attempt to identify the cause of the delay. For VMware, try switching the **Network connection** setting between **Bridged** and **NAT** on the virtual machine to see if it helps to improve the copy speed.
	The package's **Compressed** property setting is incorrect.	If a package is in a directory that contains many other applications, and its **Compressed** property is set to **True**, the Automated Application Converter knows that only that one file needs to be copied to the virtual machine for repackaging. However, if the **Compressed** property set is set to **False**, there is no way to determine which of the files in that directory belong to the package, so all of the files in the directory must be copied to the virtual machine before repackaging can start. See Editing Package Properties on the Packages Tab. ***Tip*** • *It is recommended that each package be placed in its own directory to avoid problems such as this one.*

Table 9-46 • Solutions to Common Problems (cont.)

Problem	Possible Causes	Solution
Cannot connect to a VMware virtual machine	The latest version of the VMware VIX API is not installed on the same machine as the Automated Application Converter.	If you are using VMware virtualization technology (VMware ESX or ESXi Server or a local VMware Workstation 6.5 or later), you need to have the VMware VIX API installed on the same machine as the Automated Application Converter. See VMware VIX API Requirement.

	You are using VMware Workstation 7.01, but the `wrapper-config.txt` file does not identify Workstation 7.0.1	If running VMware Workstation 7.0.1, verify that the VIX `wrapper-config.txt` contains the following line that identifies a Workstation 7.0.1. For example:
		`# Workstation 7.0.1`
		`ws 9vmdb 7.0.1 Workstation-7.0.0`
		`player 9 vmdb 3.0.1 workstation-7.0.0`

	You are attempting to connect to a VMware ESX and a VMware Workstation image at the same time.	When performing repackaging on a VMware virtual machine using the Automated Application Converter, you should connect to either a VMware ESX/ESXi image or a VMware Workstation image, but not both at the same time.
	VIX API is not working properly on virtual machine.	If the VIX API is not working, verify that `vmrun.exe` works. You can enter **`vmrun.exe /?`** to get the parameters for this tool. For example, to power up a virtual image on a VMware ESX server, use this command: `C:\Program Files (x86)\VMware\VMware VIX>vmrun -h https://172.17.1.221/sdk -T esx -u root -p hostpassword -gu` *`UserName`* `-gp guestpassword start "[QA_ISO] Windows 7 Ultimate - 32-bit/Windows 7 Ultimate - 32-bit.vmx"`

Best Practices for Optimal Performance

When setting up the Automated Application Converter to perform automated repackaging and conversion to virtual packages, you should following these best practices:

Table 9-47 • Best Practices

Practice	Description
Machine containing virtual images should be dedicated for use with the Automated Application Converter	Machines that contain the virtual images that you will use with the Automated Application Converter should be dedicated for use with this tool only.
Virtual machine needs adequate resources	Make sure that the machine containing the virtual images has adequate resources.
Each package should be in its own folder	Each package that you add to an Automated Application Converter project should be in an individual folder, especially if they are uncompressed packages.
Manually test a virtual machine before adding it to an Automated Application Converter project	Before adding a virtual machine to your Automated Application Converter project, boot it up manually to make sure it is running properly. Make sure that no message boxes open (such as a `Failure to Start Service` error message) that would require user input.
Windows 7 is recommended	It is recommended that you install the Automated Application Converter on a machine running Windows 7, either 32-bit or 64-bit.
Disable unnecessary devices	On each virtual machine, disable devices that are not required, such as: • CD/DVD drives • USB Controller • Floppy drives • Sound cards
Turn off Windows Update on the virtual machines	To avoid problems with repackaging, turn off Windows Update on the virtual machines.

How to Test a Virtual Machine

If you are having trouble using a virtual machine with the Automated Application Converter, you may want to perform the following steps to manually test that virtual machine to see if it is in working order.

Task: ***To manually test a virtual machine:***

1. Connect to the technology provider: Microsoft Hyper-V Server, VMware ESX or ESXi Server, or VMware Workstation 6.5 or later.
2. Use Remote Desktop to connect to the virtual machine you want to test.
3. Launch the virtual machine to make sure that it boots up properly and that you can login using the user name and password you specified when you added the virtual machine to your Automated Application Converter project.

   Tip • *If you discover that you specified an incorrect user name or password, update the* ***Guest Username*** *and* ***Guest Password*** *properties on the* ***Machines*** *tab for this machine.*
4. Check to see if a snapshot exists and that snapshot name is either `AutoRepack_Base` or that you have specified an alternate name in the **Snapshot Name** property on the **Packages** tab for that virtual machine.
5. Manually copy the `Repackager` folder in the AdminStudio installation directory to the virtual machine to determine if you have write permission on the virtual machine.
6. Manually copy an application to the virtual machine.
7. Launch the Repackaging Wizard and repackage that application.
8. Copy the captured data from the virtual machine to the host machine in the output location specified for that package in the **Path** property on the **Packages** tab to test if you have permission to write to that location.
9. Repeat these steps for each of the virtual machines you are using with your project.

Automated Application Converter Error Messages

This section includes information on how to resolve the following error messages that could be generated by the Automated Application Converter:

- Error -4308: VM failed to start up
- Error -4309: VM failed to shut down
- Error -4310: Failed to connect to VM
- Error -4312: Failed to prepare Repackager
- Error -4313: Failed to access the package
- Error -4314: Failed to copy repackaged output from virtual machine
- Error -4315: Failed to send command to VM
- Error -4316: Failed getting response from VM
- Error -4317: Failed running pre-snapshot
- Error -4318: Failed running post-snapshot
- Error -4319: Failed running package installation
- Error -4320: Failed creating folder on VM
- Error -4333: Preparing command-line...
- Error -4380: Failed to prepare AppV
- Error -4388: Failed preparing for pre-snapshot
- Error -4389: Failed connecting to server
- Error -4390: Failed connecting to image
- Error -4391: Failed to reboot
- Error -4395: Failed to create VM directory
- Error -4409: Failed to delete package cache folder

Debug Messages in the Automated Application Converter Log Report

By default, debug messages that occur during a conversion run are saved in the AdminStudio Automated Application Converter Log report, but the display of those debug messages is turned off. However, if you are using Microsoft Internet Explorer 8 as your default browser, you can choose to view those debug messages. See Viewing Debug Messages for instructions.

Error -4308: VM failed to start up

The following table documents this message:

Table 9-48 • Error -4308: VM failed to start up

Category	Description
Message:	`Error -4308 controlling virtual machine: VM failed to start up`
Cause:	Automated Application Converter is unable to access this VMware virtual machine due to a failure to login to the virtual machine server or into the guest virtual machine.
Resolution:	Open the **Machines** tab and verify that the following properties are set correctly for this virtual machine: • **Machine Settings**—Verify the **Guest Username** and **Guest Password** properties. • **Virtual Machine Server**—Verify the **Server Username** and **Server Password** properties.

Error -4309: VM failed to shut down

The following table documents this message:

Table 9-49 • Error -4309: VM failed to shut down

Category	Description
Message:	`Error -4309 controlling virtual machine: VM failed to shut down`
Cause:	Automated Application Converter is unable to access this machine virtual machine in order to shut it down.
Resolution:	Open the **Machines** tab and verify that the following properties are set correctly for this virtual machine: • **Machine Settings**—Verify the **Guest Username** and **Guest Password** properties. • **Virtual Machine Server**—Verify the **Server Username** and **Server Password** properties.

Error -4310: Failed to connect to VM

The following table documents this message:

Table 9-50 • Error -4310: Failed to connect to VM

Category	Description
Message:	`Error -4310 controlling virtual machine: Failed to connect to VM`
Cause:	This error could be caused by the following reasons: • Virtual machine was unexpectedly shut down early. • Operating system on the virtual machine does not launch. • The Guest Agent is not running on the virtual machine. • Permissions to the virtual machine are incorrect. • Virtual machine `AutoRepack_Base` snapshot was taken before before the machine was powered off.
Resolution:	To resolve this error, try the following: • After you have run the Virtual Machine Preparation Tool on this virtual machine, verify that the snapshot launches, automatically logs into the virtual machine, and that the Guest Agent opens. • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Do not shut down the virtual machine in the middle of using it. • Make sure that you power off the virtual machine before taking the `AutoRepack_Base` snapshot.

Error -4312: Failed to prepare Repackager

The following table documents this message:

Table 9-51 • Error -4312: Failed to prepare Repackager

Category	Description
Message:	`Error -4312 controlling virtual machine: Failed to prepare Repackager`
Cause:	The Automated Application Converter could not read the Repackager source location or could not write to the Repackager cache location.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties under **Machine Settings** on the **Machines** tab for this virtual machine are set correctly. • Verify that the virtual machine's **Repackager Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4313: Failed to access the package

The following table documents this message:

Table 9-52 • Error -4313: Failed to access the package

Category	Description
Message:	`Error -4313 processing package: Failed to access the package`
Cause:	The Automated Application Converter could not read from the package source location or could not write to the package cache location.
Resolution:	To resolve this error, try the following: • Ensure that the host machine has access to the network and is visible to/accessible from the virtual machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • If the package is on a network share, verify that both the host machine and virtual machine have access to that network share. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Verify that the package does not already exist as read-only.

Error -4314: Failed to copy repackaged output from virtual machine

The following table documents this message:

Table 9-53 • Error -4314: Failed to copy repackaged output from virtual machine

Category	Description
Message:	`Error -4314 controlling virtual machine: Failed to copy repackaged output from virtual machine`
Cause:	The virtual machine could not read the output cache location, or could not write to the project output location.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the virtual machine's **Output Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • If the project **Output Path** is set to a network share, verify that the host machine has access to that network share.

Error -4315: Failed to send command to VM

The following table documents this message:

Table 9-54 • Error -4315: Failed to send command to VM

Category	Description
Message:	`Error -4315 controlling virtual machine: Failed to send command to VM`
Cause:	There was a network error sending a command from the host machine to the Guest Agent on the virtual machine.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the Guest Agent running on the virtual machine has not crashed. • Update the Guest Agent on the virtual machine by running the latest version of the Virtual Machine Preparation Tool (`VMCfg.exe`) and taking a new snapshot.

Error -4316: Failed getting response from VM

The following table documents this message:

Table 9-55 • Error -4316: Failed getting response from VM

Category	Description
Message:	`Error -4316 controlling virtual machine: Failed getting response from VM`
Cause:	The Automated Application Converter did not receive a response from the virtual machine.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the Guest Agent running on the virtual machine has not crashed. • Update the Guest Agent on the virtual machine by running the latest version of the Virtual Machine Preparation Tool (`VMCfg.exe`) and taking a new snapshot.

Error -4317: Failed running pre-snapshot

The following table documents this message:

Table 9-56 • Error -4317: Failed running pre-snapshot

Category	Description
Message:	`Error -4317 processing package: Failed running pre-snapshot`
Cause:	The virtual machine could not create the output cache location or could not run Repackager.
Resolution:	To resolve this error, try the following: • Ensure that the host machine has access to the network and is visible to/accessible from the virtual machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** and **Repackager Cache Path** properties on the **Machines** tab for this virtual machine are set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Consider changing the **Repackaging Method** property for the package to **Installation monitoring**, which would eliminate the need for a pre-snapshot.

Error -4318: Failed running post-snapshot

The following table documents this message:

Table 9-57 • Error -4318: Failed running post-snapshot

Category	Description
Message:	`Error -4318 processing package: Failed running post-snapshot`
Cause:	The virtual machine could not run Repackager.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** and **Repackager Cache Path** properties on the **Machines** tab for this virtual machine are set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Consider changing the **Repackaging Method** property for the package to **Installation monitoring**, which would eliminate the need for a post-snapshot.

Error -4319: Failed running package installation

The following table documents this message:

Table 9-58 • Error -4319: Failed running package installation

Category	Description
Message:	`Error -4319 processing package: Failed running package installation`
Cause:	The virtual machine could not run Repackager or the virtual machine could not launch the application.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4320: Failed creating folder on VM

The following table documents this message:

Table 9-59 • Error -4320: Failed creating folder on VM

Category	Description
Message:	`Error -4320 processing package: Failed creating folder on VM`
Cause:	Virtual machine was unable to create the Setup Cache folder. ***Note*** • *This error may be related to Error -4313: Failed to access the package.*
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4333: Preparing command-line...

The following table documents this message:

Table 9-60 • Error -4333: Preparing command-line...

Category	Description
Message:	`Error -4333 processing package: Preparing command-line...`
Cause:	The virtual machine could not query the associated program for the provided extension.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the package's source file (as specified in the **Setup Cache Path** property on the **Machines** tab) can run when double-clicked on the virtual machine. • If any special tools are required (script engines, etc.) to run the command line, install them on the virtual machine and retake the snapshot.

Error -4380: Failed to prepare AppV

The following table documents this message:

Table 9-61 • Error -4380: Failed to prepare AppV

Category	Description
Message:	`Error -4380 controlling virtual machine: Failed to prepare AppV`
Cause:	The virtual machine was unable to read the App-V client installation sources, or was unable to write to the Setup Cache Path.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify App-V Client installation folder was specified correctly on the Select Virtual Machine Dialog Box. • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Setup Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4388: Failed preparing for pre-snapshot

The following table documents this message:

Table 9-62 • Error -4388: Failed preparing for pre-snapshot

Category	Description
Message:	`Error -4388 processing package: Failed preparing for pre-snapshot`
Cause:	The virtual machine was unable to write to the Output Cache Path.
Resolution:	To resolve this error, try the following: • Ensure that the virtual machine has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive.

Error -4389: Failed connecting to server

The following table documents this message:

Table 9-63 • Error -4389: Failed connecting to server

Category	Description
Message:	`Error -4389 controlling virtual machine: Failed connecting to server`
Cause:	The Automated Application Converter was unable to connect to the virtual machine server. The server machine may be unavailable.
Resolution:	To resolve this error, try the following: • Verify that the **Server Address**, **Server Username**, and **Server Password** properties under **Virtual Machine Server** on the **Machines** tab are set correctly for this virtual machine. • Verify that the Hyper-V or VMware ESX/ESXi server is running. • Ensure that the virtual server has access to the network and is visible to/accessible from the host machine (if it is running on a different computer). • If you are using VMware and you received XE values 22002 (55F2) or 22003 (55F3), verify that the VMware VIX API is installed. See VMware VIX API Requirement.

Error -4390: Failed connecting to image

The following table documents this message:

Table 9-64 • Error -4390: Failed connecting to image

Category	Description
Message:	`Error -4390 controlling virtual machine: Failed connecting to image`
Cause:	The Automated Application Converter was unable to connect to the virtual machine. The virtual machine may be unavailable or the virtual machine credentials may be incorrect.
Resolution:	To resolve this error, try the following: • Verify that the virtual machine has not been deleted. • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • If you are using VMware ESXi Server and you received an XE value of 0020 (corresponding to a `VIX_E_LICENSE` error), this indicates that you may need to purchase a license for your VMware ESXi Server in order to use it with the Automated Application Converter.

Error -4391: Failed to reboot

The following table documents this message:

Table 9-65 • Error -4391: Failed to reboot

Category	Description
Message:	`Error -4391 controlling virtual machine: Failed to reboot`
Cause:	The Automated Application Converter could not access the virtual machine.
Resolution:	To resolve this error, try the following: • Verify that the **Server Address**, **Server Username**, and **Server Password** properties under **Virtual Machine Server** on the **Machines** tab are set correctly for this virtual machine. • Attempt to launch the virtual machine manually to ensure that it has not become corrupted.

Error -4395: Failed to create VM directory

The following table documents this message:

Table 9-66 • Error -4395: Failed to create VM directory

Category	Description
Message:	`Error -4395 controlling virtual machine: Failed to create VM directory`
Cause:	Virtual machine could not create a directory.
Resolution:	To resolve this error, try the following: • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the virtual machine's **Output Cache Path** property on the **Machines** tab for this virtual machine is set to a drive on the virtual machine, especially if the virtual machine does not have a `C:` drive. • Verify that the package's **Package** field does not include any characters that are invalid for a file name.

Error -4409: Failed to delete package cache folder

The following table documents this message:

Table 9-67 • Error -4409: Failed to delete package cache folder

Category	Description
Message:	`Error -4409 processing package: Failed to delete package cache folder`
Cause:	Virtual machine could not delete a directory. ***Note*** • *This error may be related to Error -4313: Failed to access the package.*
Resolution:	To resolve this error, try the following: • Verify that the **Guest Username** and **Guest Password** properties on the **Machines** tab for the virtual machine are set correctly. • Verify that the file or its containing folder is not locked due to being open in Windows Explorer, the Command Window, etc.

Virtualization Conversion Error Messages

When converting a Windows Installer package to a virtual application, error and warning messages are generated. Some of these messages are generic to package virtualization, and others are specific to the virtualization solution you are preparing packages for.

This section includes information on how to resolve error messages that could be generated by during virtualization using the Automated Application Converter, App-V Assistant, VMware ThinApp Assistant, and Citrix Assistant.

Error -9000: Unknown Exception

The following table documents this message:

Table 9-68 • Error -9000: Unknown Exception

Category	Description
Type:	Error
Message:	`An unknown exception occurred.`
Cause:	This is an unexpected internal error.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9001: Unknown COM

The following table documents this message:

Table 9-69 • Error -9001: Unknown COM

Category	Description
Type:	Error
Message:	`Internal error.`
Resolution:	Contact AdminStudio Technical Support.

Error -9002: Error Opening Package

The following table documents this message:

Table 9-70 • Error -9002: Error Opening Package

Category	Description
Type:	Error
Message:	`An error occurred when opening the package.`
Cause:	This is an unexpected internal error when reading the Windows Installer package.
Resolution:	Check to make sure that the package is accessible to the user. If the error persists and the package is on a network share, copy the package locally (to avoid any network connection issues) and try again. If this does not solve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9003: Error Saving Package

The following table documents this message:

Table 9-71 • Error -9003: Error Saving Package

Category	Description
Type:	Error
Message:	`An error occurred when saving the package.`
Cause:	This is an unexpected internal error when trying to save the Citrix profile.

Table 9-71 • Error -9003: Error Saving Package

Category	Description
Resolution:	Check to see if the user has proper access to the location the profile is being built to. If this does not solve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9004: Process Cancelled By User

The following table documents this message:

Table 9-72 • Error -9004: Process Cancelled By User

Category	Description
Type:	Error
Message:	`Process cancelled by user.`
Cause:	The user clicked the **Cancel** button to stop the build.
Resolution:	Restart the build process.

Error -9005: Error Creating Temporary Folder

The following table documents this message:

Table 9-73 • Error -9005: Error Creating Temporary Folder

Category	Description
Type:	Error
Message:	`An error occurred while creating a temporary folder`
Cause:	You encounter this error when the user does not have permission to write to `C:\TMP`, or the drive is out of disk space.
Resolution:	Obtain write access to `C:\TMP`, and free some disk space on the drive, and then rebuild the profile.

Error -9006: Error Decompressing Package

The following table documents this message:

Table 9-74 • Error -9006: Error Decompressing Package

Category	Description
Type:	Error
Message:	`An error occurred while decompressing the package 'PackageName'.`
Cause:	You encounter this error when the package is a compressed Windows Installer package (`.msi`) and errors were generated when AdminStudio attempted to perform an administrative installation to extract the files.
Resolution:	When this error occurred, you should have also received a return error code from Windows Installer. Look up that error code in the Windows Installer Help Library to determine the cause of the problem. If you did not receive a return error code from Windows Installer, this error could have been caused by the package not being authored properly. In the Windows Installer package, check to see if the **AdminExecuteSequence** table was defined. If that table is missing, the package cannot be decompressed.

Error -9007: File With Extension Not Found

The following table documents this message:

Table 9-75 • Error -9007: File With Extension Not Found

Category	Description
Type:	Error
Message:	`No file found with the extension 'ComponentKeyName'.`
Cause:	This is an unexpected error that occurred when file extensions were being processed.
Resolution:	Check to make sure that the executable for the file extension exists and that it is set as the key file in its component.

Error -9008: Error Extracting Icon

The following table documents this message:

Table 9-76 • Error -9008: Error Extracting Icon

Category	Description
Type:	Error
Message:	`An error occurred while extracting the icon 'IconKeyName'`
Cause:	This is an unexpected error that occurred when an icon listed in the **Icon** table was being extracted.
Resolution:	Verify that the **Icon** entry in the **Icon** table is valid. If necessary, replace it with a valid icon.

Error -9009: Unknown Provider

The following table documents this message:

Table 9-77 • Error -9009: Unknown Provider

Category	Description
Type:	Error
Message:	`The specified provider is unknown 'ProviderName'.`
Cause:	This is an unexpected internal error.
Resolution:	Invalid data may have been modified via the Direct Editor causing this error. Delete the **Release** you are building, and then create a new one and rebuild.

Error -9010: Invalid Target File Name

The following table documents this message:

Table 9-78 • Error -9010: Invalid Target File Name

Category	Description
Type:	Error
Message:	`The target file name is invalid. 'FileName'`
Cause:	This is an unexpected internal error.

Table 9-78 • Error -9010: Invalid Target File Name

Category	Description
Resolution:	Invalid data may have been modified via the Direct Editor causing this error. Verify the **Name** field on the Citrix Assistant / ThinApp Assistant **Profile Information** page and make sure the name does not contain any invalid file name characters.

Error -9011: Error Reading MSI Table

The following table documents this message:

Table 9-79 • Error -9011: Error Reading MSI Table

Category	Description
Type:	Error
Message:	`Unexpected error reading MSI table 'TableName'`
Cause:	This is an unexpected error that occurred when the specified Windows Installer table was being processed.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9012: Unexpected Error in Method

The following table documents this message:

Table 9-80 • Error -9012: Unexpected Error in Method

Category	Description
Type:	Error
Message:	`Unexpected error in method 'MethodName'`
Cause:	This is an unexpected internal error.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9013: Type Library Not Found

The following table documents this message:

Table 9-81 • Error -9013: Type Library Not Found

Category	Description
Type:	Error
Message:	`Type library not found: 'TypeLibraryName'`
Cause:	You encounter this error when a type library file was not found when trying to extract COM information.
Resolution:	Check to see if the type library file exists in the proper location when building the profile. If this does not resolve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9014: ShellExecute Failed

The following table documents this message:

Table 9-82 • Error -9014: ShellExecute Failed

Category	Description
Type:	Error
Message:	`ShellExecute failed: 'CommandLine'`
Cause:	You encounter this error when the specified command line failed to launch a process.
Resolution:	Check to see if the executable file name shown is a valid file and that the user has the proper access rights to run it. If this does not resolve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9015: Unable to Determine Full Path for Driver

The following table documents this message:

Table 9-83 • Error -9015: Unable to Determine Full Path for Driver

Category	Description
Type:	Warning
Message:	`Unable to determine the full path for driver 'DriverName'`
Cause:	You encounter this error when a driver referenced in the **ODBCDataSource** table is not being installed by the package.
Resolution:	This error can be resolved in one of two ways: **Editing the Windows Installer Package** 1. Edit the package using InstallShield Direct Edit Mode. 2. Navigate to the **ISVirtualPackage** table. 3. Create an entry as follows to identify the full path of the missing driver: **Name**: `<DriverName> Description` **Value**: `Path to Driver` **Manually Installing the Driver** Install the missing driver on your machine and then rebuild the Citrix profile.

Warning -9016: Contents of Table Ignored

The following table documents this message:

Table 9-84 • Warning -9016: Contents of Table Ignored

Category	Description
Type:	Warning
Message:	`Contents of table 'TableName' will be ignored`
Cause:	This error message identifies a known limitation of Citrix conversion.
Resolution:	If the contents of the table is deemed critical, repackage the application, and then rebuild the Citrix profile.

Warning -9017: .NET 1.x Assembly Not Supported

The following table documents this message:

Table 9-85 • Warning -9017: .NET 1.x Assembly Not Supported

Category	Description
Type:	Warning
Message:	`Assembly 'AssemblyName' is a .NET 1.x assembly and will not be converted correctly. Only .NET 2.0/3.0 assemblies are currently supported. You may wish to repackage this package first.`
Cause:	You encounter this error when attempting to convert a package containing a .NET 1.x assembly. Only .NET 2.0/3.0 assemblies are currently supported.
Resolution:	Repackage the application, and then rebuild the Citrix profile.

Warning -9018: Custom Actions Ignored

The following table documents this message:

Table 9-86 • Warning -9018: Custom Actions Ignored

Category	Description
Type:	Warning
Message:	`Custom action 'CustomActionName' will be ignored.`
Cause:	When converting a Windows Installer package to a Citrix profile, all custom actions are ignored. Any modifications to a target machine that a custom action in this Windows Installer package may create will not be propagated into the Citrix profile. ***Note*** • *When a custom action that does not modify the system or perform any part of the installation (such as an InstallShield Editor predefined custom action or a Type 19 custom action) is encountered, no message is generated. If a Type 51 custom action is encountered (which sets a property from a formatted text string), it is automatically resolved. If a Type 35 custom action is encountered, it is only resolved if it is referenced in the* ***Directory*** *table.*

Table 9-86 • Warning -9018: Custom Actions Ignored

Category	Description
Resolution:	The resolution that you should perform depends upon the purpose of the custom action: • If the custom action merely automatically enters a value or makes some other kind of minor modification, you can ignore this warning. • If the custom action does something which could change the behavior of the installation (such as setting a Property), you may need to resolve this issue. To resolve this issue, first attempt to launch the converted package on Citrix XenApp. If you receive any application errors, you need to repackage this application. ***To repackage a Windows Installer package to capture custom action functionality:*** 1. Use the Repackaging Wizard to repackage this application. The Repackaging Wizard monitors system changes as an application is installed, and then that data is converted into a Repackager project. 2. Build the Repackager project to generate a revised Windows Installer package. This new Windows Installer package does not contain any custom actions, but (as a result of being repackaged) it will include the functionality performed by the original custom action.

Warning -9019: Conditionalized Components

The following table documents this message:

Table 9-87 • Warning -9019: Conditionalized Components

Category	Description
Type:	Warning
Message:	`There exist one or more conditionalized components which may not be converted correctly`
Cause:	This warning is generated when attempting to convert conditionalized components because conditions on components are not evaluated.
Resolution:	Repackage the application on a machine that has a similar environment to the machines where the profile will be deployed. Then rebuild the Citrix profile. You can also evaluate the conditions on the listed components and remove the components you know are not needed for your target machines. Then rebuild the Citrix profile.

Error -9020: Directory With Null Parent

The following table documents this message:

Table 9-88 • Error -9020: Directory With Null Parent

Category	Description
Type:	Error
Message:	`Directory 'DirectoryName' has a null parent and will be ignored.`
Cause:	This error occurs if a directory table entry (other than `TARGETDIR`) is null.
Resolution:	Evaluate the ThinApp application to see if it works. If it does not work properly, you may want to consider repackaging the package.

Error -9021: Unable to Extract COM Data

The following table documents this message:

Table 9-89 • Error -9021: Unable to Extract COM Data

Category	Description
Type:	Error
Message:	`Unable to extract COM data for 'FileName'`
Cause:	This Windows Installer package has an entry in the **TypeLib** or **SelfReg** table that contains COM data that AdminStudio cannot convert to application data. Depending upon which file cannot be COM extracted, it is possible that this application will still work properly in Citrix XenApp isolation environment if you repackage this Windows Installer package with COM table mapping turned off. COM data is stored in the Windows Registry. So, if you repackage this Windows Installer package, the capture process will be able to obtain all of this data because it captures all modifications to the Registry and does not have to rely on COM extraction.
Resolution:	To resolve this issue, you need to repackage your Windows Installer package with COM table mapping turned off.

Error -9022: Complus Table

The following table documents this message:

Table 9-90 • Error -9022: Complus Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'Complus'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **Complus** table. During the conversion process, the **Complus** table is not read.
Resolution:	The **Complus** table contains information needed to install COM+ applications. While Citrix XenApp supports communicating with COM+ applications, it does not support *installing* COM+ services. Therefore, this application cannot be deployed on Citrix XenApp.

Error -9024: FileSFPCatalog

The following table documents this message:

Table 9-91 • Error -9024: FileSFPCatalog

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'FileSFPCatalog'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **FileSFPCatalog** table. During the conversion process, the **FileSFPCatalog** table is not read.
Resolution:	The **FileSFPCatalog** table associates specified files with the catalog files used by system file protection. If this file is necessary for the function of the application, you need to use Repackager to repackage the application.

Warning -9026: LaunchCondition Table

The following table documents this message:

Table 9-92 • Warning -9026: LaunchCondition Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'LaunchCondition'.`
Cause:	You encounter this warning when the Windows Installer package that you are converting includes a **LaunchCondition** table. During the conversion process, the **LaunchCondition** table is not read.
Resolution:	The **LaunchCondition** table contains a list of conditions that all must be satisfied for the installation to begin. For example, if an application requires Windows XP to run, Windows XP is listed in the **LaunchCondition** table. Because this table is not read, no check is performed. Therefore, when a user on an operating system other than Windows XP launches this Citrix profile, the application may not function properly. To resolve this issue, perform one of the following tasks: • **Option 1: Set Requirements on the Profile Requirements Page**—If the launch conditions only include operating system, service pack, and language requirements, open this package in the InstallShield Editor Citrix Assistant / ThinApp Assistant, and set those Operating System and Language requirements on the **Profile Requirements** page. Then deploy this application as a Citrix profile. • **Option 2: Determine if Launch Conditions are Met**—Review the launch conditions listed in the table, and determine if the desktops in your enterprise meet those requirements. If all of the desktops meet those requirements, you can deploy this application as a Citrix profile. If the desktops do not meet those requirements (such as having Internet Explorer 6.0 instead of 7.0), upgrade those desktops to meet these requirements, and then deploy this application as a Citrix profile.

Warning -9027: LockPermissions Table

The following table documents this message:

Table 9-93 • Warning -9027: LockPermissions Table

Category	Description
Type	Warning
Message:	`The conversion process does not support data in the MSI table 'LockPermissions'.`

Table 9-93 • Warning -9027: LockPermissions Table

Category	Description
Cause:	You encounter this warning when the Windows Installer package that you are converting includes a **LockPermissions** table. During the conversion process, the **LockPermissions** table is not read.
Resolution:	The **LockPermissions** table is used to secure individual portions of your application (files, registry keys, and created folders) in a locked-down environment. Citrix does not support permissions on files, registry keys, or created folders. You cannot modify permissions on any of the application's ACLs (access control lists). Because users will have full permissions when running this application in the isolation environment, this warning will not result in any problems.

Error -9028: MoveFile Table

The following table documents this message.

Table 9-94 • Error -9028: MoveFile Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'MoveFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **MoveFile** table. During the conversion process, the **MoveFile** table is not read.
Resolution:	This **MoveFile** table contains a list of files to be moved or copied from a specified source directory to a specified destination directory. Because this table is not read, you need to do one of the following to resolve this issue: • **Option 1: Edit the Windows Installer Package**—Open the Windows Installer package in InstallShield Editor and modify it to eliminate the use of the **MoveFile** table by installing additional files in the specified directories. • **Option 2: Repackage the Application**—Use the Repackaging Wizard to repackage this application, and then build the Repackager project to generate a revised Windows Installer package. • **Option 3: Write a Pre-Launch Script**—Write a pre-launch script that performs the file moving operations identified in the **MoveFile** table upon application launch.

Error -9029: MsiDriverPackages Table

The following table documents this message:

Table 9-95 • Error -9029: MsiDriverPackages Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'MsiDriverPackages'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **MsiDriverPackages** table. During the conversion process, the **MsiDriverPackages** table is not read.
Resolution:	The **MsiDriverPackages** table includes one record for each driver package component in the application. Citrix XenApp does not support any type of driver. For example, when installing a printer, you can install the printer software within the isolation environment, but not the printer drivers. Therefore, to resolve this issue, you need to install any required drivers outside of the isolation environment on the user desktop machines.

Warning -9030: ODBCTranslator Table

The following table documents this message:

Table 9-96 • Warning -9030: ODBCTranslator Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'ODBCTranslator'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ODBCTranslator** table. During the conversion process, the **ODBCTranslator** table is not read.
Resolution:	The **ODBCTranslator** table lists the ODBC translators belonging to the installation. ODBC translators translate one form of raw data into another form that can be used with a specific database type. Ignoring the **ODBCTranslator** table should not prevent your application from functioning properly. However, if it does, you need to use Repackager to repackage the application.

Warning -9031: RemoveFile Table

The following table documents this message:

Table 9-97 • Warning -9031: RemoveFile Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'RemoveFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **RemoveFile** table. During the conversion process, the **RemoveFile** table is not read. This warning is displayed only if the application installation removes files during install (not uninstall).
Resolution:	The **RemoveFile** table contains a list of files to be removed. If this file removal requirement is just a clean-up step, that does not impact the function of the application, you do not need to address this issue. However, if the existence of the files listed in the **RemoveFile** table prevents the application from functioning, you need to set the isolation option of the files to **Ignore** so that they are not visible to the isolation environment. The **Ignore** option directs the isolation environment to always look for the file on the system (ignoring the one inside the isolation environment).

Warning -9032: RemoveIniFile Table

The following table documents this message:

Table 9-98 • Warning -9032: RemoveIniFile Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'RemoveIniFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **RemoveIniFile** table. During the conversion process, the **RemoveIniFile** table is not read.
Resolution:	The **RemoveIniFile** table contains the information an application needs to delete from a `.ini` file. If the removal of this entry is necessary for the function of the application, you need to use Repackager to repackage the application.

Warning -9033: RemoveRegistry Table

The following table documents this message:

Table 9-99 • Warning -9033: RemoveRegistry Table

Category	Description
Type:	Warning
Message:	`The conversion process does not support data in the MSI table 'RemoveRegistry'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **RemoveRegistry** table. During the conversion process, the **RemoveRegistry** table is not read.
Resolution:	The **RemoveRegistry** table contains the registry information the application needs to delete from the system registry. If this removal requirement is just a clean-up step, that does not impact the function of the application, you do not need to address this issue. However, if the existence of the registry keys listed in the **RemoveRegistry** table prevents the application from functioning, you need to set the isolation option of the registry keys to **Ignore** so that they are not visible to the isolation environment. The **Ignore** option directs the isolation environment to always look for the registry key on the system (ignoring the one inside the isolation environment).

Error -9036: ISCEInstall Table

The following table documents this message:

Table 9-100 • Error -9036: ISCEInstall Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISCEInstall'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISCEInstall** table. During the conversion process, the **ISCEInstall** table is not read.
Resolution:	The **ISCEInstall** table is used to install Windows mobile applications. The conversion of mobile applications to Citrix profiles is not supported.

Error -9037: ISComPlusApplication Table

The following table documents this message:

Table 9-101 • Error -9037: ISComPlusApplication Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISComPlusApplication'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISComPlusApplication** table. During the conversion process, the **ISComPlusApplication** table is not read.
Resolution:	The **ISComPlusApplication** table contains information about COM+ applications. While Citrix XenApp supports communicating with COM+ applications, it does not support *installing* COM+ services. Therefore, this application cannot be deployed on Citrix XenApp.

Error -9038: ISPalmApp Table

The following table documents this message:

Table 9-102 • Error -9038: ISPalmApp Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISPalmApp'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISPalmApp** table. During the conversion process, the **ISPalmApp** table is not read.
Resolution:	The **ISPalmApp** table is used to install Palm mobile applications. The conversion of mobile applications to Citrix profiles is not supported.

Error -9039: ISSQLScriptFile Table

The following table documents this message:

Table 9-103 • Error -9039: ISSQLScriptFile Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISSQLScriptFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISSQLScriptFile** table. During the conversion process, the **ISSQLScriptFile** table is not read.
Resolution:	The **ISSQLScriptFile** table lists SQL scripts. When a Windows Installer package is installed, it can run an SQL script to update a database. An application running as a Citrix profile cannot update a database. To resolve this issue, you need to update the database prior to using the converted Citrix profile using one of the following methods: • Use scripts to update the database manually. • Update it by running the Windows Installer installation on one of the machines in your network that has access to that database.

Error -9040: ISVRoot Table

The following table documents this message:

Table 9-104 • Error -9040: ISVRoot Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISVRoot'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISVRoot** table. During the conversion process, the **ISVRoot** table is not read.
Resolution:	The **ISVRoot** table installs a Web site. An application running as a Citrix profile in an isolation environment cannot create a Web site. Therefore, creating Citrix profiles for applications that create Web sites during installation is not supported.

Error -9041: ISXmlFile Table

The following table documents this message:

Table 9-105 • Error -9041: ISXmlFile Table

Category	Description
Type:	Error
Message:	`The conversion process does not support data in the MSI table 'ISXmlFile'.`
Cause:	You encounter this error when the Windows Installer package that you are converting includes a **ISXmlFile** table. During the conversion process, the **ISXmlFile** table is not read.
Resolution:	The **ISXmlFile** table modifies XML files. If the modification of XML files is required in order for this application to operate properly, you need to use Repackager to repackage this application.

Error -9051: Package Decompression Canceled

The following table documents this message:

Table 9-106 • Error -9051: Package Decompression Canceled

Category	Description
Type:	Error
Message:	`Package decompression canceled by the user`
Cause:	The user cancelled the process of decompression of compressed MSI packages.

Error -9100: CreateInstance of Package Object Failed

The following table documents this message:

Table 9-107 • Error -9100: CreateInstance of Package Object Failed

Category	Description
Type:	Error
Message:	`CreateInstance of the Citrix package object failed.`
Cause:	Unexpected internal error.

Table 9-107 • Error -9100: CreateInstance of Package Object Failed

Category	Description
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9101: Create Operation of Package Object Failed

The following table documents this message:

Table 9-108 • Error -9101: Create Operation of Package Object Failed

Category	Description
Type:	Error
Message:	`Create operation on Citrix package object failed 'ObjectName'.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9102: Failed to Write Header Information

The following table documents this message:

Table 9-109 • Error -9102: Failed to Write Header Information

Category	Description
Type:	Error
Message:	`Failed to write package header information.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9103: Citrix Finalization Failed

The following table documents this message:

Table 9-110 • Error -9103: Citrix Finalization Failed

Category	Description
Type:	Error
Message:	`Citrix Finalization Failed`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9104: Citrix Save Failed

The following table documents this message:

Table 9-111 • Error -9104: Citrix Save Failed

Category	Description
Type:	Error
Message:	`Citrix Save Failed`
Cause:	Unexpected internal error. This error may sometimes occur when the profile is to be digitally signed.
Resolution:	Deselect the option to digitally sign the Citrix profile and then rebuild it.

Error -9105: Error Initializing Citrix Writer

The following table documents this message:

Table 9-112 • Error -9105: Error Initializing Citrix Writer

Category	Description
Type:	Error
Message:	`Unexpected error initializing Citrix writer`
Cause:	Unexpected internal error.

Table 9-112 • Error -9105: Error Initializing Citrix Writer

Category	Description
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9106: Error Initializing Citrix Package

The following table documents this message:

Table 9-113 • Error -9106: Error Initializing Citrix Package

Category	Description
Type:	Error
Message:	`Unexpected error initializing Citrix package`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9107: Error Writing Citrix File Entries

The following table documents this message:

Table 9-114 • Error -9107: Error Writing Citrix File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix file entries.`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9108: Error Determining Source File Path

The following table documents this message:

Table 9-115 • Error -9108: Error Determining Source File Path

Category	Description
Type:	Error
Message:	`Unexpected error determining source file path for 'FileName'`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9109: Error Writing Citrix Folder Entries

The following table documents this message:

Table 9-116 • Error -9109: Error Writing Citrix Folder Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix folder entries`
Cause:	Unexpected internal error.
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9110: Error Writing Citrix Registry Entries

The following table documents this message:

Table 9-117 • Error -9110: Error Writing Citrix Registry Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix registry entries`
Cause:	Unexpected internal error.

Table 9-117 • Error -9110: Error Writing Citrix Registry Entries

Category	Description
Resolution:	First check to see if the product was installed properly. Then, perform preliminary investigational steps and contact AdminStudio Technical Support.

Error -9113: Error Writing Citrix INI File Entries

The following table documents this message:

Table 9-118 • Error -9113: Error Writing Citrix INI File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix INI file entries`
Cause:	Unexpected internal error.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9114: Error Writing Citrix Shortcuts

The following table documents this message:

Table 9-119 • Error -9114: Error Writing Citrix Shortcuts

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix shortcuts`
Cause:	A catastrophic error has occurred while writing shortcuts to the profile.
Resolution:	Verify that shortcuts point to a valid file. Try to narrow down issue by only keeping one shortcut and then try to rebuild.

Error -9115: Error Saving Citrix Profile

The following table documents this message:

Table 9-120 • Error -9115: Error Saving Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error saving Citrix profile`
Cause:	A catastrophic error has occurred while saving the Citrix profile.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9116: Error Creating Empty Citrix Profile

The following table documents this message:

Table 9-121 • Error -9116: Error Creating Empty Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error creating empty Citrix profile`
Cause:	AdminStudio is unable to create a new internal instance of a Citrix profile.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9117: Error Creating Intermediate Folder

The following table documents this message:

Table 9-122 • Error -9117: Error Creating Intermediate Folder

Category	Description
Type:	Error
Message:	`Unexpected error creating intermediate folder`
Cause:	AdminStudio is unable to create the intermediate folder used for the build. This error could occur if the user does not have permission to write to `C:\TMP`.

Table 9-122 • Error -9117: Error Creating Intermediate Folder

Category	Description
Resolution:	Obtain write access to `C:\TMP` and then rebuild the profile.

Error -9118: Error Initializing Citrix Profile

The following table documents this message:

Table 9-123 • Error -9118: Error Initializing Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error initializing Citrix profile.`
Cause:	The initial values on the new profile could not be set.
Resolution:	Check the package name, description, version, and security settings for any possible causes.

Error -9119: Error Creating Default Target in Citrix Profile

The following table documents this message:

Table 9-124 • Error -9119: Error Creating Default Target in Citrix Profile

Category	Description
Type:	Error
Message:	`Unexpected error creating default target in Citrix profile`
Cause:	Initial target in the new profile could not be created.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9120: Error Deleting File From Profile

The following table documents this message:

Table 9-125 • Error -9120: Error Deleting File From Profile

Category	Description
Type:	Error

Table 9-125 • Error -9120: Error Deleting File From Profile

Category	Description
Message:	`Unexpected error deleting file 'FileName' from profile`
Cause:	Specified file could not be deleted from profile.
Resolution:	Check to see if the file exists and if the user has access rights to the file. If that did not resolve the problem, perform these additional investigational steps and then contact AdminStudio Technical Support.

Error -9121: Failed to Copy File into Citrix Profile

The following table documents this message:

Table 9-126 • Error -9121: Failed to Copy File into Citrix Profile

Category	Description
Type:	Error
Message:	`Failed to copy file into Citrix profile. Error: 'Name' File: 'Name'`
Cause:	Specified file could not be copied into profile.
Resolution:	Check to see if the file exists and if the user has access rights to the file. Also, when this error occurred, you should have also received a return error code from Windows Installer. Look up that error code in the Windows Installer Help Library to determine the cause of the problem.

Error -9122: Target Does Not Exist in Citrix Profile

The following table documents this message:

Table 9-127 • Error -9122: Target Does Not Exist in Citrix Profile

Category	Description
Type:	Warning
Message:	`The target for shortcut 'ShortcutName' does not exist in the Citrix profile. Excluding shortcut.`
Cause:	Actual file that shortcut points to is not included in the package.

Table 9-127 • Error -9122: Target Does Not Exist in Citrix Profile

Category	Description
Resolution:	Exclude the shortcut by clearing the selection on the Citrix Assistant **Profile Shortcuts** page, and then rebuild the profile.

Error -9124: No Shortcuts Created for this Profile

The following table documents this message:

Table 9-128 • Error -9124: No Shortcuts Created for this Profile

Category	Description
Type:	Error
Message:	`No shortcuts were created in the XenApp profile during the conversion because none were detected in the source package.`
Cause:	A XenApp profile must include at least one valid shortcut.
Resolution:	Add a shortcut on the Citrix Assistant **Profile Shortcuts** page, and then rebuild the profile.

Error -9125: Error Writing Citrix File Type Associations

The following table documents this message:

Table 9-129 • Error -9125: Error Writing Citrix File Type Associations

Category	Description
Type:	Error
Message:	`Unexpected error writing Citrix file type associations`
Cause:	Unable to write file type associations.
Resolution:	Perform preliminary investigational steps and then contact AdminStudio Technical Support.

Error -9126: Failed to Sign Profile Using Certificate

The following table documents this message:

Table 9-130 • Error -9126: Failed to Sign Profile Using Certificate

Category	Description
Type:	Error
Message:	`Failed to sign the profile using the supplied certificate`
Cause:	The certificate that is being used in invalid.
Resolution:	Obtain a valid certificate and rebuild the profile.

Error -9127: Could Not Create Script Execution

The following table documents this message:

Table 9-131 • Error -9127: Could Not Create Script Execution

Category	Description
Type:	Error
Message:	`Could not create script execution for 'ScriptName'`
Cause:	The specified script contains invalid data.
Resolution:	On the Citrix Assistant **Build Settings** page, delete the script from the profile, re-add it, and then rebuild the profile. If you are still having problems, perform these additional investigational steps and then contact AdminStudio Technical Support.

Warning -9128: Duplicate Shortcut

The following table documents this message:

Table 9-132 • Warning -9128: Duplicate Shortcut

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists in the profile. Excluding duplicate shortcut.`

Table 9-132 • Warning -9128: Duplicate Shortcut

Category	Description
Cause:	There are multiple shortcuts defined in this profile that refer to different Start Menu locations or to other locations in the package.
Resolution:	These shortcuts are not needed. On the Citrix Assistant **Profile Shortcuts** page, unselect these shortcuts, and then rebuild the profile.

Warning -9129: Duplicate Shortcut Names

The following table documents this message:

Table 9-133 • Warning -9129: Duplicate Shortcut Names

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists in the profile, but with different command line parameters. A new unique shortcut 'NewShortcutName(1)' will be created in the profile.`
Cause:	There are two shortcuts defined in this profile that have the same name, even though they have different command line parameters.
Resolution:	On the Citrix Assistant **Profile Shortcuts** page, rename one of these shortcuts and then rebuild the profile.

Warning -9130: Duplicate Shortcut Targets

The following table documents this message:

Table 9-134 • Warning -9130: Duplicate Shortcut Targets

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists in the profile, but with different target. A new unique shortcut 'NewShortcutName(1)' will be created in the profile.`
Cause:	There are two shortcuts defined in this profile that have the same name, even though they have different targets.
Resolution:	On the Citrix Assistant **Profile Shortcuts** page, rename one of these shortcuts and then rebuild the profile.

Warning -9131: Unable to Resolve Installer Variable

The following table documents this message:

Table 9-135 • Warning -9131: Unable to Resolve Installer Variable

Category	Description
Type:	Warning
Message:	`Unable to resolve an installer variable in the string 'StringName'`
Cause:	Not all Windows Installer variables can be resolved at build time. This can result in errors if your application requires a specific value.
Resolution:	Repackage this application and rebuild the profile, or use a constant value in the Windows Installer package.

Warning -9132: 16 Color Shortcut Icon Not Found

The following table documents this message:

Table 9-136 • Warning -9132: 16 Color Shortcut Icon Not Found

Category	Description
Type:	Warning
Message:	`No 16 color icon found for 'ShortcutName' shortcut. Shortcut icon may look distorted when published.`
Cause:	The icon used for this shortcut does not contain a 16-color image. Since Citrix currently does not support images with a higher number of colors, this icon may look distorted when shown and published in Citrix XenApp.
Resolution:	You can modify the shortcut to use a different icon or add a 16-color image to the one currently used.

Warning -9133: Shortcut Icon Not Found

The following table documents this message:

Table 9-137 • Warning -9133: Shortcut Icon Not Found

Category	Description
Type:	Warning

Table 9-137 • Warning -9133: Shortcut Icon Not Found

Category	Description
Message:	`No icon found for 'ShortcutName' shortcut. Using generic Windows application icon instead.`
Cause:	If no icon can be loaded for this shortcut, the generic Windows application icon is used. This can happen if the file used is corrupt or if extracting an image from it is not possible.
Resolution:	Modify the shortcut to use a different icon.

Warning -9134: Failure to Extract Icon from Executable

The following table documents this message:

Table 9-138 • Warning -9134: Failure to Extract Icon from Executable

Category	Description
Type:	Warning
Message:	`Failed to extract icon from executable 'filename'. Make sure the executable is not corrupt.`
Cause:	A corrupt icon file may cause this warning.
Resolution:	Modify the shortcut to use a different icon.

Error -9135: Shortcut Target is 16-Bit

The following table documents this message:

Table 9-139 • Error -9135: Shortcut Target is 16-Bit

Category	Description
Type:	Error
Message:	`The target for shortcut 'ShortcutName' is 16-bit. This application may not function properly in the Citrix Isolation Environment.`
Cause:	The file this shortcut points to is a 16-bit application.
Resolution:	Replace file with a newer 32-bit version. Can also test and see if the application works properly in the Citrix environment.

Warning -9136: Some Files May Not Be Decompressed

The following table documents this message:

Table 9-140 • Warning -9136: Some Files May Not Be Decompressed

Category	Description
Type:	Warning
Message:	`Some files may not be decompressed from this package because it contains features with a default install level of 0.`
Cause:	When installing a compressed Windows Installer package, the build engine runs an administrative installation of it to decompress it. One limitation of an administrative installation is that it does not decompress a file if the feature it is contained in has a default install level of **0**. If there are any files in any components contained within those features, AdminStudio will generate an error when it attempts to copy those files into the Citrix profile, because they will not exist in the source location.
Resolution:	To resolve this issue, edit the Windows Installer package and set the default install level of that feature to something other than **0**.

Warning -9137: Destination Directory Cannot Be Found

The following table documents this message:

Table 9-141 • Warning -9137: Destination Directory Cannot Be Found

Category	Description
Type:	Warning
Message:	`The destination directory for the 'FileName' file cannot be found. You should consider Repackaging this application before proceeding with the conversion process.`
Cause:	This is an internal error.
Resolution:	Contact AdminStudio Technical Support.

Warning -9138: Ignoring a DuplicateFile Table Entry

The following table documents this message:

Table 9-142 • Warning -9138: Ignoring a DuplicateFile Table Entry

Category	Description
Type:	Warning
Message:	`Ignoring a DuplicateFile table entry because unable to resolve the property used for the DestFolder: 'INVALIDPATH'`
Cause:	You might encounter this error when the Windows Installer package that you are converting includes one or more entries in the **DuplicateFile** table, and a property that is used in the DestFolder column for one of those entries in the **DuplicateFile** table cannot be resolved. For example, if the destination for a duplicate file is set by a custom action, that destination cannot be resolved during the conversion.
Resolution:	The **DuplicateFile** table contains a list of files that need to be duplicated during installation, either to a different directory than the original file or to the same directory but with a different name. Because a destination in this table cannot be resolved, you need to do one of the following to resolve this issue: • **Option 1: Edit the Windows Installer Package**—Open the Windows Installer package in InstallShield and modify it to eliminate the use of the problematic entry in the **DuplicateFile** table by including any additional copies of that file. • **Option 2: Repackage the Application**—Use the Repackaging Wizard to repackage this application, and then build the Repackager project to generate a revised Windows Installer package. • **Option 3: Write a Pre-Launch Script**—Write a pre-launch script that—upon application launch—performs the file copy operations for the problematic entry in the **DuplicateFile** table.

Error -9139: 64-Bit Executables (XenApp)

The following table documents this message:

Table 9-143 • Error -9139: 64-Bit Executables (XenApp)

Category	Description
Type:	Error
Message:	`The target for shortcut '[SHORTCUT_NAME]' is a 64-bit executable. XenApp does not support 64-bit applications.`
Cause:	XenApp does not support 64-bit applications.

Table 9-143 • Error -9139: 64-Bit Executables (XenApp) (cont.)

Category	Description
Resolution:	This 64-bit application cannot be converted to XenApp format.

Error -9200: ThinApp Must Be Installed

The following table documents this message:

Table 9-144 • Error -9200: ThinApp Must Be Installed

Category	Description
Type:	Error
Message:	`A licensed or demo version of ThinApp must be installed on this machine in order to successfully build ThinApp applications. (www.vmware.com)`
Cause:	ThinApp is not installed.
Resolution:	Install ThinApp.

Warning -9201: Extension for Shortcut Files Must Be “.exe”

The following table documents this message:

Table 9-145 • Warning -9201: Extension for Shortcut Files Must Be “.exe”

Category	Description
Type:	Warning
Message:	`The extension for the target for shortcut 'ShortcutName' is not '.exe'. Excluding shortcut.`
Cause:	Shortcuts that do not have a filename extension of `.exe` are excluded.
Resolution:	No action is required.

Error -9202: No Applications Were Created

The following table documents this message:

Table 9-146 • Error -9202: No Applications Were Created

Category	Description
Type:	Error
Message:	`No applications were created during the ThinApp conversion because no shortcuts were detected in the source package.`
Cause:	Either the Windows Installer package had no shortcuts or all shortcuts were excluded.
Resolution:	Make sure that the source Windows Installer `.msi` package has at least one shortcut to an `.exe` file.

Error -9203: ThinApp Tool is Missing

The following table documents this message:

Table 9-147 • Error -9203: ThinApp Tool is Missing

Category	Description
Type:	Error
Message:	`ThinApp: 'ToolName' was not found`
Cause:	One of the ThinApp tools required to build a ThinApp application was not found.
Resolution:	Reinstall ThinApp.

Error -9204: Duplicate Shortcut

The following table documents this message:

Table 9-148 • Error -9204: Duplicate Shortcut

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists. Excluding duplicate shortcut.`
Cause:	The source package has two shortcuts that both point to the same `.exe` target.

Table 9-148 • Error -9204: Duplicate Shortcut

Category	Description
Resolution:	No action is required.

Error -9205: Identically-Named Shortcut Already Exists, But With Different Parameters

The following table documents this message:

Table 9-149 • Error -9205: Identically-Named Shortcut Already Exists, But With Different Command Line Parameters

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists, but with different command line parameters. A new, unique shortcut will be created.`
Cause:	Two shortcuts in the package differed in arguments only.
Resolution:	No action is required.

Error -9206: Identically-Named Shortcut Already Exists, But With a Different Target

The following table documents this message:

Table 9-150 • Error -9206: Identically-Named Shortcut Already Exists, But With a Different Target

Category	Description
Type:	Warning
Message:	`'ShortcutName' shortcut already exists, but with a different target. A new, unique shortcut will be created.`
Cause:	Two shortcuts differed in the target pointed to only.
Resolution:	No action is required.

Error -9207: Error During Build Process (vregtool.exe)

The following table documents this message:

Table 9-151 • Error -9207: Error During Build Process (vregtool.exe)

Category	Description
Type:	Error
Message:	`An error occurred during the ThinApp build process (vregtool.exe).`
Cause:	An unexpected error occurred while running the vregtool.exe step of the ThinApp build process.
Resolution:	The cause of this error may be discernible by the progress messages that were displayed just before this error occurred. Also, make sure none of the files/folders in the build folder hierarchy are locked.

Error -9208: Error Occurred During Build Process (vftool.exe)

The following table documents this message:

Table 9-152 • Error -9208: Error Occurred During Build Process (vftool.exe)

Category	Description
Type:	Error
Message:	`An error occurred during the ThinApp build process (vftool.exe)`
Cause:	An unexpected error occurred while running the `vftool.exe` step of the ThinApp build process.
Resolution:	The cause of this error may be discernible by the progress messages that were displayed just before this error occurred. Also, make sure none of the files/folders in the build folder hierarchy are locked.

Error -9209: Error Occurred During ThinApp Build Process (tlink.exe)

The following table documents this message:

Table 9-153 • Error -9209: Error Occurred During ThinApp Build Process (tlink.exe)

Category	Description
Type:	Error
Message:	`An error occurred during the ThinApp build process (tlink.exe)`
Cause:	An unexpected error occurred while running the tlink.exe step of the ThinApp build process.
Resolution:	The cause of this error may be discernible by the progress messages that were displayed just before this error occurred. Also, make sure none of the files/folders of the build folder hierarchy are locked.

Error -9210: 64-Bit Executables (ThinApp)

The following table documents this message:

Table 9-154 • Error -9210: 64-Bit Executables (ThinApp)

Category	Description
Type:	Error
Message:	`The target for shortcut '[SHORTCUT_NAME]' is a 64-bit executable. ThinApp does not support 64-bit applications.`
Cause:	ThinApp does not support 64-bit applications.
Resolution:	This 64-bit application cannot be converted to ThinApp format.

Error -9300: Unhandled Exception During AdviseFile Operation

The following table documents this message:

Table 9-155 • Error -9300: Unhandled Exception During AdviseFile Operation

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the AdviseFile operation for rule 'RuleName'`

Table 9-155 • Error -9300: Unhandled Exception During AdviseFile Operation

Category	Description
Resolution:	Contact AdminStudio Technical Support.

Error -9301: Unhandled Exception During AdviseRegistry Operation

The following table documents this message:

Table 9-156 • Error -9301: Unhandled Exception During AdviseRegistry Operation

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the AdviseRegistry operation for rule 'RuleName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9302: Unhandled Exception During Command Action

The following table documents this message:

Table 9-157 • Error -9302: Unhandled Exception During Command Action

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the command action with the description 'CommandActionName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9303: Unhandled Exception During Alter File Action

The following table documents this message:

Table 9-158 • Error -9303: Unhandled Exception During Alter File Action

Category	Description
Type:	Error

Table 9-158 • Error -9303: Unhandled Exception During Alter File Action

Category	Description
Message:	`An unhandled exception occurred during the alter file action with the description 'FileName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9304: Unhandled Exception During Alter Registry Action

The following table documents this message:

Table 9-159 • Error -9304: Unhandled Exception During Alter Registry Action

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the alter registry action with the description 'RegistryName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9305: Unhandled Exception During Create Action

The following table documents this message:

Table 9-160 • Error -9305: Unhandled Exception During Create Action

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the create action with the description 'CreateName'`
Resolution:	Contact AdminStudio Technical Support.

Error -9306: Unhandled Exception During Execution of Rules Engine

The following table documents this message:

Table 9-161 • Error -9306: Unhandled Exception During Execution of Rules Engine

Category	Description
Type:	Error
Message:	`An unhandled exception occurred during the execution of the rules engine.`
Resolution:	Contact AdminStudio Technical Support.

Error -9401: Error Initializing App-V Writer

The following table documents this message:

Table 9-162 • Error -9401: Error Initializing App-V Writer

Category	Description
Type:	Error
Message:	`Unexpected error initializing App-V writer.`
Resolution:	Contact AdminStudio Technical Support.

Error -9402: Error Initializing App-V Package

The following table documents this message:

Table 9-163 • Error -9402: Error Initializing App-V Package

Category	Description
Type:	Error
Message:	`Unexpected error initializing App-V package.`
Resolution:	Contact AdminStudio Technical Support.

Error -9403: Error Writing App-V File Entries

The following table documents this message:

Table 9-164 • Error -9403: Error Writing App-V File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V file entries.`
Resolution:	Contact AdminStudio Technical Support.

Error -9404: Error Writing App-V Folder Entries

The following table documents this message:

Table 9-165 • Error -9404: Error Writing App-V Folder Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V folder entries`
Resolution:	Contact AdminStudio Technical Support.

Error -9405: Error Writing App-V Registry Entries

The following table documents this message:

Table 9-166 • Error -9405: Error Writing App-V Registry Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V registry entries.`
Resolution:	Contact AdminStudio Technical Support.

Error -9406: Error Writing App-V INI File Entries

The following table documents this message:

Table 9-167 • Error -9406: Error Writing App-V INI File Entries

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V INI file entries.`
Resolution:	Contact AdminStudio Technical Support.

Error -9407: Error Writing App-V Shortcuts

The following table documents this message:

Table 9-168 • Error -9407: Error Writing App-V Shortcuts

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V shortcuts.`
Resolution:	Contact AdminStudio Technical Support.

Error -9408: Error Writing App-V File Type Data

The following table documents this message:

Table 9-169 • Error -9408: Error Writing App-V File Type Data

Category	Description
Type:	Error
Message:	`Unexpected error writing App-V file type data.`
Resolution:	Contact AdminStudio Technical Support.

Error -9409: Error Saving App-V Data

The following table documents this message:

Table 9-170 • Error -9409: Error Saving App-V Data

Category	Description
Type:	Error
Message:	`Unexpected error saving App-V data.`
Resolution:	Contact AdminStudio Technical Support.

Error -9410: Error Determining Source File Path

The following table documents this message:

Table 9-171 • Error -9410: Error Determining Source File Path

Category	Description
Type:	Error
Message:	`Unexpected error determining source file path for 'FileName'.`
Cause:	The installation location of a file, which is determined by some run time property, cannot be determined by the App-V virtual converter.
Resolution:	Locate the file in InstallShield and provide a known directory.

Error -9411: OSD File Template Could Not Be Extracted

The following table documents this message:

Table 9-172 • Error -9411: OSD File Template Could Not Be Extracted

Category	Description
Type:	Error
Message:	`The Microsoft App-V OSD file template could not be extracted. The OSD file generation will not operate properly.`
Resolution:	Contact AdminStudio Technical Support.

Error -9412: OSD File Could Not Be Saved

The following table documents this message:

Table 9-173 • Error -9412: OSD File Could Not Be Saved

Category	Description
Type:	Error
Message:	`The Microsoft App-V OSD file could not be saved. The OSD file generation will not operate properly.`
Resolution:	Contact AdminStudio Technical Support.

Error -9413: App-V OSD Save

The following table documents this message:

Table 9-174 • Error -4313: App-V OSD Save

Category	Description
Type:	Error
Message:	`The Microsoft App-V OSD file could not be saved. The OSD file generation will not operate properly.`
Resolution:	Contact AdminStudio Technical Support.

Warning -9414: Local App-V Application Specified as a Dependency of the Primary Application

The following table documents this message:

Table 9-175 • Warning -9414: Local App-V Application Specified as a Dependency of the Primary Application

Category	Description
Type:	Warning
Message:	`A local App-V application was specified as a dependency of the primary application. The primary application may not run correctly if it is relocated to a different location.`
Cause:	The user specified a dependent App-V application that is either on the local drive or on a mapped drive. This is determined by examining the HREF attribute of the CODEBASE tag in the dependency application's OSD file.

Table 9-175 • Warning -9414: Local App-V Application Specified as a Dependency of the Primary Application

Category	Description
Resolution:	Dependency applications should be referenced by a portable mechanism using either a non-FILE protocol or by using a network URL.

Error -9415: Dependency Application Was Not Found

The following table documents this message:

Table 9-176 • Error -9415: Dependency Application Was Not Found

Category	Description
Type:	Error
Message:	`Dependency application was not found: 'ApplicationName'.`
Cause:	A specified App-V dependency application file was not found.
Resolution:	Check the path of the specified App-V dependency application.

Warning -9416: Invalid Primary Application Directory

The following table documents this message:

Table 9-177 • Warning -9416: Invalid Primary Application Directory

Category	Description
Type:	Error
Message:	`The specified Primary Application Directory, 'DirectoryName', does not exist.`
Cause:	This may be caused if the directories specified in the Windows Installer package have changed after a valid Primary Application Directory was specified.
Resolution:	Specify a valid Primary Application Directory using the supplied browse folder in InstallShield.

Error -9417: Dependency Application's OSD File Contains an Invalid HREF Value

The following table documents this message:

Table 9-178 • Error -9417: Dependency Application's OSD File Contains an Invalid HREF Value

Category	Description
Type:	Error
Message:	`Dependency application OSD file contains an invalid value for the HREF field of the CODEBASE tag: 'HREF_Field_Value'`
Cause:	The CODEBASE tag of the dependency application's OSD file may have an empty or non-existent HREF attribute.
Resolution:	Make sure that the CODEBASE tag of the dependency application's OSD file has a valid HREF attribute.

Error -9418: Error While Privatizing Side-By-Side Assemblies

The following table documents this message:

Table 9-179 • Error -9418: Error While Privatizing Side-By-Side Assemblies

Category	Description
Type:	Error
Message:	`An error occurred while privatizing Side-By-Side assemblies.`
Cause:	When converting to an App-V package, files installed to the win32 Sxs assembly cache need to be privatized so that the App-V runtime can find them. This error occurs if there was an unexpected failure in that process.
Resolution:	Contact AdminStudio Technical Support.

Error -9419: Error Inserting Watermark

The following table documents this message:

Table 9-180 • Error -9419: Error Inserting Watermark

Category	Description
Type:	Error

Table 9-180 • Error -9419: Error Inserting Watermark

Category	Description
Message:	`An error has occurred inserting the evaluation watermark into the App-V Package.`
Resolution:	Contact AdminStudio Technical Support.

Error -9420: Error During App-V Package Upgrade

The following table documents this message:

Table 9-181 • Error -9420: Error During App-V Package Upgrade

Category	Description
Type:	Error
Message:	`An error occurred while configuring App-V package upgrade.`
Cause:	This is a general failure to configure the new App-V package that is being built to be an upgrade for the previous one.
Resolution:	Verify that the previous App-V package is present and accessible at the location specified. Verify that the previous App-V package is not corrupt.

Warning -9421: 16-Bit Application

The following table documents this message:

Table 9-182 • Warning -9421: 16-Bit Application

Category	Description
Type:	Warning
Message:	`The target for shortcut '[SHORTCUT_NAME]' is 16-bit. This application will not function in 64-bit environments.`
Cause:	This warning helps to identify 16-bit applications. This is important for enterprises looking to deploy to 64-bit environments.
Resolution:	This warning can be ignored if the application will not be deployed to a 64-bit environment. Otherwise, it will be necessary to get a newer version of this application that is 32 or 64-bit.

Error -9422: Package Cannot Be Opened

The following table documents this message:

Table 9-183 • Error -9422: Package Cannot Be Opened

Category	Description
Type:	Error
Message:	`The previous package '[SFT_FILE_NAME]' could not be opened. Please verify that the package is a valid SFT file.`
Cause:	This error indicates that there was a problem reading information from the previous .sft file which is necessary to configure an upgrade. This could mean that the previous .sft file is corrupt, missing, or is not really an .sft file.
Resolution:	Verify that the file is present and accessible. Verify that the file is not corrupt. One way this could be done is by trying to deploy the App-V package on a machine with the App-V client.

Warning -9423: No Shortcuts Detected

The following table documents this message:

Table 9-184 • Warning -9423: No Shortcuts Detected

Category	Description
Type	Warning

Message	For App-V packages:
	`No shortcuts were detected in the source package. This can be a valid scenario for packages with file type association entry points and/or those that are meant to be dependencies of other App-V packages through the use of Dynamic Suite Composition functionality.`
	For XenApp and ThinApp packages:
	`This package contains no shortcuts. Shortcuts are necessary to define the entry point into the virtual application.`

Cause	Package contains no shortcuts. • **App-V**—For conversion to App-V packages, this issue is acceptable in some scenarios, such as packages which provide dependencies to others which dynamically suite it. However, if this package merely provides a plug-in to another application, it must contain a shortcut to launch that application in this package's virtual context. • **ThinApp and XenApp**—For conversion to ThinApp and XenApp formats, shortcuts are necessary to define the entry point into the virtual application.
Resolution:	One potential resolution to this issue is to use InstallShield Editor to add shortcut(s) to the Windows Installer package.

Warning -9500: Shortcut Missing

The following table documents this message:

Table 9-185 • Warning -9500: Shortcut Missing

Category	Description
Type:	Warning
Message:	`The target for shortcut 'FileName' does not exist. Excluding shortcut.`
Cause:	The target file of a shortcut in the project does not exist.
Resolution:	Repackage this application and then rebuild the virtual package.

Error -10000: Process Cancelled By User

The following table documents this message:

Table 9-186 • Error -10000: Process Cancelled By User

Category	Description
Type:	Error
Message:	`Process cancelled by user.`
Cause:	User clicked **Cancel** to cancel the profile conversion process.

Warning -10001: Suite File Missing

The following table documents this message:

Table 9-187 • Warning -10001: Suite File Missing

Category	Description
Type:	Warning
Message:	`The suite MSI file 'FileName' is missing and will be excluded from the conversion.`
Cause:	An MSI file that is part of a suite conversion was not found.
Resolution:	Make sure the input file for the suite conversion process exists.

Warning -10002: Suite File is Duplicate

The following table documents this message:

Table 9-188 • Warning -10002: Suite File is Duplicate

Category	Description
Type:	Warning
Message:	`The suite MSI file 'FileName' appears to be the same as the main MSI file and we will exclude this file from the conversion process.`
Cause:	A suite conversion was attempted where the main Windows Installer file (.msi) and one of the additional Windows Installer files specified were the same.
Resolution:	Specify unique Windows Installer files as part of the suite conversion process.

Warning -10003: Application File Missing

The following table documents this message:

Table 9-189 • Warning -10003: Application File Missing

Category	Description
Type:	Error
Message:	`Application file not found 'ApplicationName'`
Cause:	A file referenced by the installation was not found by the App-V virtual converter. It is likely that the file reference is broken in the installation.
Resolution:	Use InstallShield to locate the file in the installation and either fix the link or delete it.

Warning -10004: INI File Missing

The following table documents this message:

Table 9-190 • Warning: -10004: INI File Missing

Category	Description
Type:	Error
Message:	`INI file not found 'INI_File_Name'.`
Resolution:	Contact AdminStudio Technical Support.

Fix 11000: Excluding TCPIP Registry Entries

The following table documents this message:

Table 9-191 • Fix 11000: Excluding TCPIP Registry Entries

Category	Description
Type:	Fix
Message:	`Excluding TCPIP registry entries from the Citrix profile.`
Action:	Automated Application Converter will exclude all TCPIP registry entries from the Citrix profile.

Fatal Error 11001: Fail on VMware

The following table documents this message:

Table 9-192 • Fatal Error 11001: Fail on VMware

Category	Description
Type:	Fatal
Message:	`VMware cannot be virtualized.`
Cause:	Conversion will fail when the application being virtualized is VMware.
Action:	This error message is displayed: `VMware cannot be virtualized.`

Warning 11003: Control Panel Applet - Citrix

The following table documents this message:

Table 9-193 • Warning 11003: Control Panel Applet - Citrix

Category	Description
Type:	Warning
Message:	`The Control Panel Applet [AppletName] will not appear in Control Panel.`
Action:	Automated Application Converter will display a warning when the application contains a control panel applet.

Fix 11004: Control Panel Applet - ThinApp

The following table documents this message:

Table 9-194 • Fix 11004: Control Panel Applet - ThinApp

Category	Description
Type:	Fix
Message:	`Generating shortcut for the Control Panel Applet located at 'DirectoryPath'`
Action:	Automated Application Converter will create a default shortcut for ThinApp Control Panel applets.

Fatal Error 11005: QuickTime 7.4.1 Causes Fatal Error

The following table documents this message:

Table 9-195 • Error 11005: QuickTime 7.4.1 Causes Fatal Error

Category	Description
Type:	Fatal Error
Message:	`QuickTime 7.4.1 is known to have errors when running from a virtual package. Use QuickTime 7.4.5 instead.`
Cause:	QuickTime 7.4.1 cannot be virtualized correctly.
Resolution:	Obtain QuickTime 7.4.5 and repeat the conversion process.

Fix 11006: Adobe Distiller Exclude AdobePDFSettings

The following table documents this message:

Table 9-196 • Fix 11006: Adobe Distiller Exclude AdobePDFSettings

Category	Description
Type:	Fix
Message:	`Excluding the registry key Software\Adobe\Acrobat Distiller\AdobePDFSettings. Adobe Distiller will recreate these settings on first use.`
Action:	Automated Application Converter will exclude the AdobePDFSettings registry settings.

Fix 11007: Exclude URL Shortcut

The following table documents this message:

Table 9-197 • Fix 11007: Adobe Distiller Exclude AdobePDFSettings

Category	Description
Type:	Fix
Message:	`Excluding shortcut to .URL file. App-V does not launch these files properly.`
Action:	Automated Application Converter will exclude the shortcut to the .URL file.

Steps to Take Before Calling Technical Support

Before contacting AdminStudio Technical Support, perform the following steps to attempt to clearly identify the problem you are having:

- **Check package**—To determine if this error is caused by a problem with the specific package you are converting, try to build a Citrix profile of a simple package that contains only one file.
- **Check machine and OS**—To determine if this error is caused by a configuration on a particular machine or operating system, attempt to build this Citrix profile on another machine or operating system.
- **Check individual files**—To determine if this is error limited to a specific item, find out if removing or excluding a particular item will build error free.

Application Features Requiring Pre- or Post-Conversion Actions

Some application features are ignored when creating a Citrix profile. Therefore, some additional pre- or post-conversion actions must be taken in order for the application profile to run on Citrix XenApp.

One action you could take to try to include ignored features in a Citrix profile is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to a Citrix profile.

The following table lists the application features which require additional, manual conversion steps:

Table 9-198 • Application Features Ignored During Profile Conversion

Windows Installer Feature	Manual Conversion Steps
User-Defined Custom Actions	When converting a Windows Installer package to a Citrix profile, all custom actions are ignored. For user-defined custom actions, a warning message is generated. Any modifications to a target machine that a custom action in this Windows Installer package may create will not be propagated into the Citrix profile. The resolution that you should perform depends upon the purpose of the custom action: • If the custom action merely automatically enters a value or makes some other kind of minor modification, you can ignore this warning. • If the custom action does something which could change the behavior of the installation (such as setting a Property), you may need to resolve this issue. To resolve this issue, first attempt to launch the converted package on Citrix XenApp. If you receive any application errors, you need to repackage this application, by performing the following steps. ***To successfully convert a package with user-defined custom actions:*** 1. Use the Repackaging Wizard to repackage this application.The Repackaging Wizard monitors system changes as an application is installed, and then that data is converted into a Repackager project. 2. Build the Repackager project to generate a revised Windows Installer package. This new Windows Installer package does not contain any custom actions, but (as a result of being repackaged) it will include the functionality performed by the original custom action.
Services	Citrix XenApp does not support any type of services. Therefore, to resolve this issue, you need to install any required services outside of the isolation environment on the user desktop machines. ***To successfully convert a package with services:*** 1. If you have an application and a service bundled in the same Windows Installer package, you need to create a separate Windows Installer package containing just the service. 2. Install the service on each of the user desktop machines. The Citrix profile of this application should now be able to run in an isolation environment on machines that already have the service installed.

Table 9-198 • Application Features Ignored During Profile Conversion

Windows Installer Feature	Manual Conversion Steps
COM+	While Citrix XenApp supports communicating with COM+ applications, it does not support *installing* COM+ services. Therefore, an application that contains COM+ services cannot be deployed on Citrix XenApp.

10

Using the Virtual Package Editor

The Virtual Package Editor is a powerful tool that lets you edit App-V packages and perform tasks such as the following:

- Customize your App-V applications.
- Resolve virtualization best practice issues and application conflicts.
- Fix run-time problems.

The Virtual Package Editor documentation contains the following sections:

Table 10-1 • AdminStudio Virtual Package Editor Documentation Sections

Section	Description
About Virtualization	Provides background information about virtualization.
About the Virtual Package Editor	Introduces some basic concepts to help you get started with editing a virtual package that you converted from an installation or that you created from scratch.
Getting Started with the Virtual Package Editor	Contains information to help you become familiar with the Virtual Package Editor, begin creating or editing a virtual package, and customize the Virtual Package Editor user interface.
Editing Virtual Packages	Explains how to create edit virtual packages and guides you through every step of the process.
Virtual Package Editor Reference	Contains comprehensive reference information for the Virtual Package Editor user interface.

About Virtualization

Virtualization enables you to isolate an application in its own environment so that it does not conflict with existing applications or modify the underlying operating system.

Limitations of a Standard Installation Environment

A typical Windows-based application has dependencies on components that are shared by multiple applications. Applications access these shared system resources, such as the registry or Windows system files. When an installation author recognizes that their application references a shared system component, they include a merge module to install that component.

When one of these shared components is installed, it is possible that a previously installed version of the same component could be overwritten; this may cause the existing application to break. A similar problem could occur when one of these applications containing a shared component is uninstalled. Because of these possible problems, extensive compatibility testing needs to be performed before an application can be distributed in the enterprise environment.

The following diagram provides an example of two conflicting installed applications.

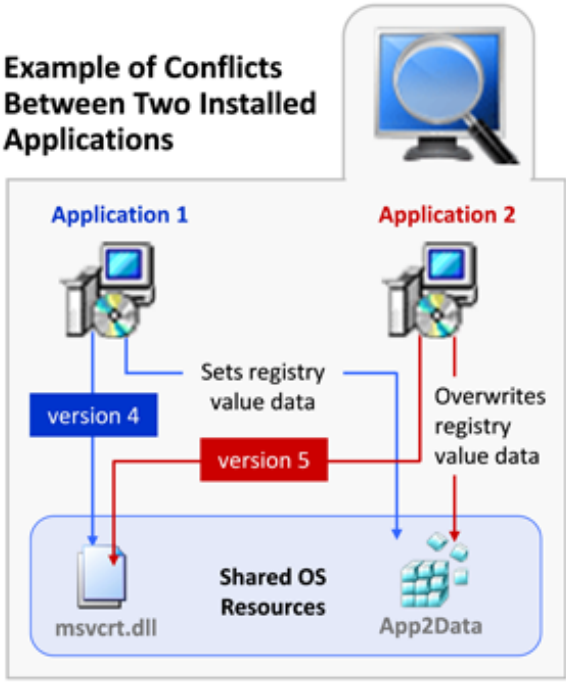


Figure 10-1: Example of Conflicts Between Two Installed Applications

Benefits of Application Virtualization

Virtual applications run in virtual environments that keep each application layer and the operating system layer separate. Each application includes its own configuration information in its virtual environment. As a result, many applications can run side-by-side with other applications on the same computer without any conflicts.

Even though virtual applications are not installed on the local machine, they exhibit the same functionality and access to local services as locally installed applications, and also nearly the same performance characteristics.

The following diagram provides an example of how application virtualization would solve the conflicts that are shown in the previous example.

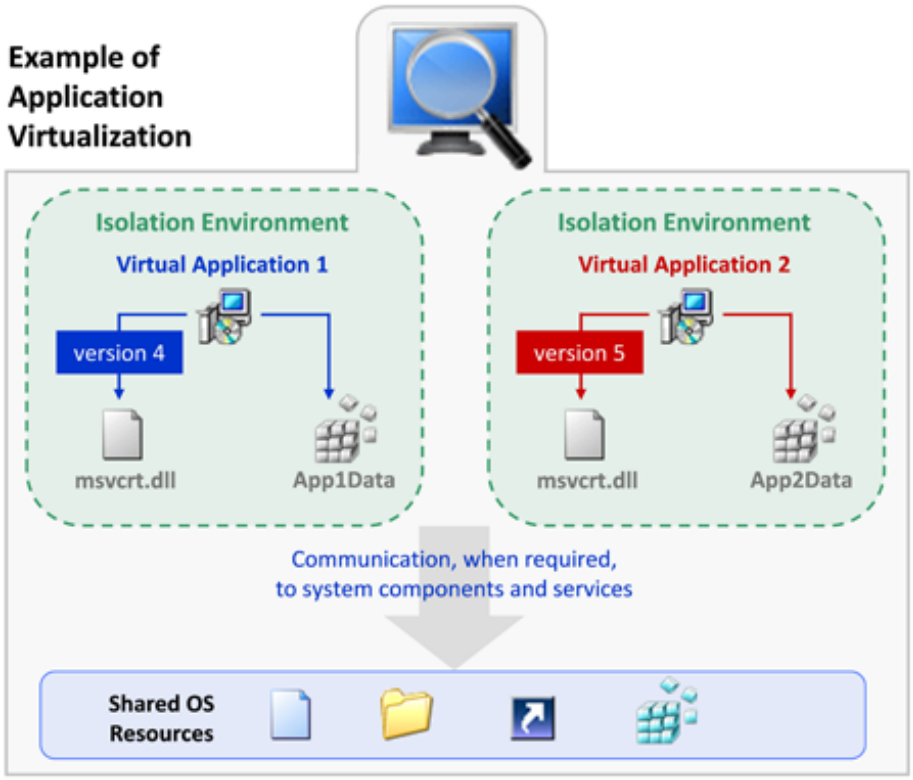


Figure 10-2: Example of Application Virtualization

Application virtualization allows the configuration of an application to be standardized to an isolation environment, rather than to an individual user's desktop machine. Application objects, files, and registry settings are contained within this isolation environment. Critical application resources are managed locally by the isolation environment, thus minimizing resource dependencies between applications.

Application virtualization greatly reduces the scope for conflicts between applications and, therefore, simplifies regression testing.

About the Virtual Package Editor

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally. Instead, only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

The Virtual Package Editor is a powerful tool that lets you edit App-V packages to customize your App-V applications, resolve virtualization best practice issues and application conflicts, and fix run-time problems. You can save your App-V packages as new packages that can be deployed alongside earlier versions of the virtual package in the same virtual environment; you can also create update packages that can upgrade earlier versions of your virtual applications.

Components of an App-V Package

The following table describes the components of an App-V package:

Table 10-2 • Components of an App-V Package

File	Description
.sft	The .sft file contains all of the files, registry information, and other configuration details of the package.
Manifest file	This file is an XML file that lists all of the .osd files in an App-V application.
.osd	The .osd files are XML-based files that describe the package's individual targets (or applications) that can be run.
.ico	The .ico files are icons files that are used for published shortcuts and file type associations.
.sprj	This file is the Microsoft App-V Sequencer project file. It contains references to the .sft and .osd files, and to a large number of settings related to the sequencing process.

Getting Started with the Virtual Package Editor

The Virtual Package Editor provides powerful features that make editing virtual packages easy. This section of the documentation contains information to help you become familiar with the Virtual Package Editor, begin creating or editing a virtual package, and customize the Virtual Package Editor user interface.

Starting the Virtual Package Editor

Task: ***To open the Virtual Package Editor, do one of the following:***

- On the **Tools** tab in AdminStudio, right-click **Virtual Package Editor** and then click **Launch Tool**.
- Launch **Application Manager**. On the **Products** tab, right-click an existing App-V package that you want to open, and then click **Edit with Virtual Package Editor**.

When you launch the Virtual Package Editor through the Tools tab, the Start Page opens. The Start Page provides access to product information, recently opened virtual packages, and product resources.

If you launch the Virtual Package Editor by opening a virtual package in the Application Manager, the Virtual Package Editor displays one of the views for the virtual package.

Creating a New Virtual Package

The Virtual Package Editor offers several ways to create a new virtual package.

Task: ***To create a new virtual package, do one of the following:***

- On the **File** menu, click **New**.
- Press CTRL+N.
- On the toolbar, click the **New** button.
- On the **Start Page** in the **Package Tasks** area, click the **Create a New Package** link.

The Virtual Package Editor also offers support for editing existing virtual packages. This functionality is useful if you converted a Windows Installer package to a virtual package in a tool such as the AdminStudio Automated Application Converter.

Opening an Existing Virtual Package

The Virtual Package Editor offers several ways to open an existing virtual package (.sft).

Task: ***To open an existing virtual package:***

1. Do one of the following:
   - On the **File** menu, click **Open**.
   - Press CTRL+O.
   - On the toolbar, click the **Open** button.
   - On the **Start Page** in the **Package Tasks** area, click the **Open an Existing Package** link.

   The **Open** dialog box opens.
2. Browse to the virtual package (.sft), and then click the **Open** button.

The Virtual Package Editor opens the virtual package, enabling you to edit it as needed.

Tip • *As an alternative, you can open a recently opened virtual package. To do so, perform one of the following tasks:*

- *On the **Start Page** in the **Package Tasks** area, click the **Open an Existing Package** link.*
- *On the **File** menu, click a recently opened .sft file name.*

*You can also open a virtual package from within Application Manager: On the **Products** tab, right-click an existing App-V package that you want to open, and then click **Edit with Virtual Package Editor**. Application Manager lets you check out the .sft file. When you save the virtual package in the Virtual Package Editor, your changes are saved in a temporary location. The version that is stored in Application Manager is updated when you check your changes in to the Application Manager.*

Saving a Virtual Package

The Virtual Package Editor offers several machine-wide, user-specific options for saving a virtual package (.sft). Before you save your virtual package as either a new package or an upgrade package, select the appropriate options.

Selecting the Appropriate Save Options

Task: ***To select the appropriate save options:***

On the **File** menu, point to **Save Options**, and then click the appropriate command to select or clear an option. The following table describes each option. If an option is selected, its menu command includes a check mark.

Table 10-3 • Save Options

Command	Description
Include App-V Launcher	If you want to use the AdminStudio App-V Application Launcher to test a newly built App-V package locally before moving it to a deployment server, select this command. To learn more, see Using the App-V Application Launcher to Test the Virtual Package. This command is selected by default.
Append Package Version	If you want the Virtual Package Editor to append the package version number to the name of the .sft file whenever you save an App-V package, select this command. This command is selected by default.

Table 10-3 • Save Options (cont.)

Command	Description
Build Wrapper MSI	If you want to build a Windows Installer package to assist in the distribution of each App-V package that you save in the Virtual Package Editor, select this command. If you enable this option, the Virtual Package Editor creates an InstallShield project (.ism file) and uses it to build an .msi package. If you run the .msi package, it "installs" the App-V application files in the local App-V client system cache. The wrapper .msi package can optionally include the App-V package file (.sft), depending on whether the Include SFT in Wrapper MSI command is selected. ***Note*** • *The Microsoft Application Virtualization Client must be installed on the local machine before you can install an App-V application through a wrapper .msi package. The installation determines whether the App-V client is present; if it is not, the installation displays an error message and exits.* Building a wrapper .msi file simplifies the deployment of an App-V application by enabling you to use enterprise distribution tools such as Microsoft System Center Configuration Manager or Novell ZENworks Configuration Management. This command is cleared by default.
Include SFT in Wrapper MSI	If you want to include the App-V package in the Windows Installer package that you save in the Virtual Package Editor, select this command. If you enable this option, the Virtual Package Editor includes the .sft file in the wrapper .msi package that it builds when you save the open App-V package. If you run the .msi package, it "installs" the App-V application files, including the .sft file, in the local App-V client system cache. If you disable this option, the contents of the .sft file are streamed from the App-V server as requested by the client. This command is available only if the Build Wrapper MSI command is selected. This command is cleared by default.
Compress Wrapper MSI	If you want to compress the App-V package files into the wrapper .msi package, select this command. This command is available only if the Build Wrapper MSI command and the Include SFT in Wrapper MSI option are selected. If you enable the former command but disable the latter command, the .msi package that is generated is always compressed. This command is selected by default.

You can save a virtual package in either of the following ways:

- Save the package as a new package. You can deploy a new package alongside earlier versions of the virtual package in the same virtual environment.
- Save the package as an update package. An update package can upgrade earlier versions of the virtual application.

Tip • *If you are using the Virtual Package Editor to edit a virtual package that is part of your application catalog, the Virtual Package Editor saves your changes in a temporary location. To update your application catalog with the latest changes to your virtual package, use Application Manager to check in your virtual package.*

Saving a Virtual Package as a New Package

Task: ***To save a virtual package as a new package, do one of the following:***

- On the **File** menu, click **Save**.
- Press CTRL+S.
- On the toolbar, click the **Save** button.

The Virtual Package Editor saves your virtual package as a new package. To open a Windows Explorer window that shows the App-V package, press CTRL+E, or on the View menu, click Show in Explorer.

Important • *If you want to deploy two copies of a package side by side, you must do some additional work before saving the package:*

- *In the General Information view, change the value of the Root Folder Name setting. This value must be unique because two packages with the same root folder name cannot be deployed simultaneously.*
- *In the General Information view, change the value of the Name setting. It is recommended that this value be different for each package.*
- *In the Shortcuts view, change the value of the Name setting, the Target Version setting, or both of those settings for each target in your package. The combination of the name and version must be unique for the targets in each new package; otherwise, the two packages cannot be deployed simultaneously.*

Saving a Virtual Package as an Update Package

Task: ***To save a virtual package as an update package:***

1. On the **File** menu, click **Save As**. The **Save As** dialog box opens.
2. Click the **Save as a new package** option.

The Virtual Package Editor saves your virtual package as an update package. To open a Windows Explorer window that shows the App-V package, press CTRL+E, or on the View menu, click Show in Explorer.

Saving a Virtual Package (an Update Package or a New Package) with a New Name and Location

1. On the **File** menu, click **Save As**. The **Save As** dialog box opens.
2. In the **Virtual Package** box, enter the path and file name that you want to use for the .sft file. As an alternative, you can click the ellipsis button (**...**) to browse to the file.
3. Click the **Save as an update package** option or the **Save as a new package** option.

The Virtual Package Editor saves your virtual package as specified. To open a Windows Explorer window that shows the App-V package, press CTRL+E, or on the View menu, click Show in Explorer.

Closing a Virtual Package

Task: ***To close a virtual package in the Virtual Package Editor:***

1. Select the tab of the .sft file that you want to close.
2. Do one of the following:
   - On the **File** menu, click **Close**.
   - Click the tab's **Close** button.

Working with the Virtual Package Editor Interface

The Virtual Package Editor interface is a graphical user interface with conventional Windows-based elements such as a menu bar, a toolbar, and dialog boxes. This section includes topics that explain how to perform basic tasks using these elements and how to customize the interface.

Configuring the Value of a Setting for More Than One Item at a Time

In many views of the Virtual Package Editor, you can select multiple items—such as files, registry keys, file extensions, or virtual services—and then change the value for one of the settings. The Virtual Package Editor lets you use the same value in that setting for all of the selected items in that view. This feature may save you time by enabling you to make extensive changes to multiple items simultaneously, instead of requiring you to edit the setting for each item individually.

For example, in the Files and Folders view, you may want to change the value of the App-V Override setting for a large number of files. If all of the files need to be configured the same way, you can simply select all of the pertinent files, and then change the value of the App-V Override setting as needed. Therefore, it is not necessary to separately select and configure each file that you want to modify.

Note that the values of some settings may not be equivalent for each selected item. For example, your virtual package may contain one file whose App-V Override setting is Yes, and another file whose App-V Override setting is No. If you select both of those files in the Files and Folders view, you will see the following unequal sign as the value of the App-V Override setting, indicating that the selected items have different values:

≠

In this example, you can select both files and select the appropriate value—Yes or No—to have the Virtual Package Editor use the same value for both files.

In some cases, the Virtual Package Editor does not allow you to change unequal values for more than one selected item. For example, if you select two files that are in the same folder, you cannot change the value of the Name setting for both of those files simultaneously, since each file in a folder must have a different file name.

Note • *If you select two or more items and you want to delete the entry in a setting that shows the unequal sign to indicate different values, you must first enter a value in the setting; then you can delete that value. For example, if you want to delete the value of the App-V VFS Path setting for two selected files, and those files have different values in the App-V VFS Path setting, you must first enter a value. Then you can delete it.*

The Virtual Package Editor provides several methods for selecting multiple items in a view.

Task: ***To select multiple items in a view so that you can configure some of their settings simultaneously, do one of the following:***

- To select multiple consecutive items that are near each other, drag your mouse pointer to create a box that surrounds each item that you want to select. When you do this, ensure that you start dragging your mouse pointer in empty space; otherwise, you may drag one or more item to a new location.
- To select multiple consecutive files or folders, select the first file or folder, press and hold SHIFT, and select the last file or folder.
- To select multiple nonconsecutive files or folders, select one file or folder, press and hold CTRL, and select each additional file or folder.

Showing or Hiding the Start Page in the Virtual Package Editor

The Virtual Package Editor Start Page is a tab that provides quick access to product information, to recently opened projects, and to Virtual Package Editor resources. You can show or hide this tab as necessary.

Task: ***To show the Start Page:***

On the **File** menu, click **Start Page**.

Task: ***To hide the Start Page, do one of the following:***

- On the **Start Page** tab, click the **Close** button.
- Click the **Start Page** tab. On the **File** menu, click **Close**.

Rearranging the Start Page and Virtual Package Tabs

Each virtual package that you have open in the Virtual Package Editor is displayed on a separate tab. The Start Page is also displayed on a separate tab. The Virtual Package Editor lets you change the order of these tabs.

Task: ***To change the order of the open tabs:***

Drag the tab that you want to move to the new location in the rows of tabs.

Showing or Hiding the Settings and Output Windows

The Settings window in the Virtual Package Editor contains a grid that lists information about the item that is selected in an open view. The Output window displays task-specific information such as details about the virtual package that you are opening. It also shows save information.

The Settings window and the Output window can be shown or hidden as necessary.

Task: ***To show or hide the Output window or the Settings window:***

On the **View** menu, click **Output Window** or click **Settings**.

If the window was visible, the Virtual Package Editor hides it. If the window was hidden, the Virtual Package Editor shows it.

Note that closing the Output window clears its contents. The Virtual Package Editor automatically shows the Output window whenever a task—such as saving or opening a virtual package—generates output.

Moving the Settings, Output, and Script Windows

The Settings window, the Output window, and the Script window can be moved to any side of the workspace in the Virtual Package Editor.

If you drag the Settings, Output, or Script window to the edge of a different side of the Virtual Package Editor interface, it becomes a docked window in that location.

Task: ***To move the Settings window, Output window, the or the Script window:***

Drag the title bar of the **Settings** window, the **Output** window, or the **Script** window to the new location. Resize the window as needed.

Showing or Hiding Toolbars

Task: ***To show or hide a toolbar, do one of the following:***

- Right-click a toolbar and select the toolbar that you want to be displayed or hidden.
- On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens. Select the check box for each toolbar that you want to be displayed. Clear the check box for each toolbar that you want to be hidden.

Adding Buttons and Menus to a Toolbar

Task: ***To add a button or menu to a toolbar:***

1. Ensure that the toolbar that you want to change is visible.
2. On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens.
3. Click the **Commands** tab.
4. In the **Categories** box, click the category for the button or menu that you want to add.
5. Drag the button or menu from the **Commands** box to the appropriate toolbar.

Tip • *To create your own custom toolbar, drag the button or menu to the empty gray area near the toolbars.*

Removing Buttons and Menus from a Toolbar

Task: ***To remove a button or menu from a toolbar:***

1. Ensure that the toolbar that you want to change is visible.
2. On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens.
3. Right-click the button or menu that you want to remove, and then click **Delete**.

Creating a Custom Toolbar

Task: ***To create a custom toolbar:***

1. On the **View** menu, point to **Toolbars**, and then click **Customize**. The **Customize** dialog box opens.
2. Click the **Tools** tab.
3. Click the **New** button. The **New Toolbar** dialog box opens.
4. In the **Toolbar name** box, enter a descriptive name for the toolbar, and click **OK**.
5. Customize the new toolbar by adding menus or buttons.

Editing Virtual Packages

Editing a virtual package involves performing some or all of the following tasks.

Table 10-4 • Editing Virtual Packages

Task	Description
Specifying Virtual Package Information	Basic information that you enter in the General Information view is used by the Microsoft Application Virtualization Client and the App-V server. The Dependencies view is where you specify other App-V packages that your App-V package requires.
Organizing Virtual Application Data	The Virtual Package Editor lets you manage the folders and files that will be available in the virtual environment. It also lets you define registry keys, values, and data for your virtual package. In addition, you can use the Virtual Package Editor to define the targets for your virtual application, and define entry points such as shortcuts for each target. These entry points enable end users to launch an App-V application from within the virtual environment.
Configuring Virtual Services	The Virtual Package Editor enables you to configure services that you want to include in your App-V package so that they are available in the virtual environment.
Testing and Troubleshooting Virtual Packages	Once you have made the necessary changes for the files, folders, shortcuts, services, and other elements of your virtual package, you are ready to test the virtual package, and identify potential conflicts and best practice violations between different App-V packages, and between App-V packages and Windows Installer–based installations. The Virtual Package Editor lets you add to a virtual package shortcuts that launch the Command Prompt window and the registry editor in order to debug issues in the virtual environment.

Specifying Virtual Package Information

When you create a virtual package or open an existing package in the Virtual Package Editor, you may need to view or specify important package information. This includes basic information such as the name of the virtual package and details such as the package GUID and version number. You may also want to see history information such as each date on which the package was saved.

Viewing History for a Virtual Package

The Virtual Package Editor shows read-only history information such as the following:

- The date and time when the package was saved
- The GUID of each saved package
- The user name of the person who saved the package
- The name of the machine on which the package was saved
- The version of the Virtual Package Editor that was used to save the package
- The version of App-V that was used when saving the package
- The operating system of the machine on which the package was saved

Each time that you save your .sft file, the Virtual Package Editor adds a new history entry to the History pane and shows such details.

Task: ***To view history for your virtual package:***

1. In the View List under **Package Information**, click **General Information**.
2. Review the information in the **History** pane.

Configuring General Information for a Virtual Package

In the General Information view, you can view and, if appropriate, edit basic information about your virtual package.

Task: ***To configure general information for your virtual package:***

1. In the View List under **Package Information**, click **General Information**.
2. In the **Settings** window, configure the settings as needed. For details about each setting, see General Information View.

Specifying a Virtual Package's Dependencies

A virtual package may rely on one or more other virtual packages in order to function properly. The Virtual Package Editor lets you specify other App-V packages that the open App-V package (the primary package) requires. This capability, called *Dynamic Suite Composition*, enables your virtual package to interact with the other virtual applications in the virtual environment. Dynamic Suite Composition enables you to deploy common system components once on each client system, making them available for use by many App-V applications, rather than requiring you to include them with each of the App-V applications that are dependent on them. This reduces redundancy in the local App-V cache and simplifies the construction and testing of the primary App-V application.

If you add a new dependency to your primary package, the Virtual Package Editor automatically associates each of the targets that are defined in the Shortcuts view with that new dependency. Similarly, if you add a new target to your primary package, the Virtual Package Editor automatically associates that target with each dependency that is defined in the Dependencies view. Each .osd file that defines a target contains a list of the other .sft files on which it depends. The Application Virtualization Client may cache this list; therefore, in most cases all of the primary package's targets should be associated with each dependency.

Adding a Dependency to a Virtual Package

The Virtual Package Editor lets you specify other App-V packages that the open App-V package (the primary package) requires.

Task: ***To add a dependency to your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. Right-click the **Dependencies** explorer and then click **Add Dependency**. The **Open** dialog box opens.
3. Browse to the .sft or .osd file for the required App-V package, and then click **Open**.

The Virtual Package Editor adds an .sft item to the Dependencies explorer. The .sft item may contain one or more targets. The targets are defined in the Shortcuts view of the primary package.

When a target with an associated dependency is launched, the Application Virtualization Client loads the dependency's environment and makes it available as part of the virtual environment of the target's package.

Configuring a Dependency in a Virtual Package

The Virtual Package Editor lets you view and configure settings for the dependencies in your virtual package. The settings display information such as the GUID and the server URL for the dependency.

Task: ***To configure a dependency in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, click the dependency that you want to configure.

3. In the **Settings** window, configure the settings as needed. For details about each setting, see Dependencies View.

Associating a Package's Targets with a Dependency in a Virtual Package

It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies. If you add a new dependency to your primary package, the Virtual Package Editor automatically associates each of the targets that are defined in the Shortcuts view with that new dependency. Similarly, if you add a new target to your primary package, the Virtual Package Editor automatically associates that target with each dependency that is defined in the Dependencies view. If you remove a target from a dependency in the Dependencies view, you may want to add it back.

Task: ***To associate a target with a dependency in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, right-click the .sft file with which you want to associate a target, and then click **Associate Target**.

   If one or more targets in the package are not associated with the dependency, the **Associate Targets with a Dependency** dialog box opens. Select the targets that you want to associate with the dependency.

   If all of the targets in the package are associated with the dependency, the Virtual Package Editor displays a message box informing you that all of the package's targets are already associated with the dependency.

The Virtual Package Editor adds one or more targets to the dependency if appropriate.

Specifying Whether a Dependency is Mandatory for a Target in a Virtual Package

The Virtual Package Editor lets you specify whether a dependency is mandatory in order for target in the primary package (the App-V package that you are editing in the Virtual Package Editor) to run properly. If the dependency is mandatory, the primary package cannot run without loading the required package. For example, a system DLL such as an MFC DLL is likely to be mandatory, but a reference to a document viewer such as Adobe Reader may not be mandatory.

Task: ***To specify whether a dependency is mandatory for a target in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, click the target that you want to configure.
3. In the **Settings** window, configure the **Mandatory** setting as needed.

Tip • *The Virtual Package Editor lets you configure the settings for more than one target at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Target from a Dependency in a Virtual Package

Important • *It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies.*

Task: ***To remove a target from a dependency in your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, right-click the target that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the target from the dependency.

Removing a Dependency from a Virtual Package

The Virtual Package Editor lets you remove a dependency from an App-V package.

Task: ***To remove a dependency from your virtual package:***

1. In the View List under **Package Information**, click **Dependencies**.
2. In the **Dependencies** explorer, right-click the .sft file dependency that you want to delete, and then click **Remove**.

The Virtual Package Editor removes the dependency from your App-V package.

Organizing Virtual Application Data

The primary objective of a virtual package is to keep the application layer and the operating system layer separate. Each application includes its own configuration information in its virtual environment. As a result, many applications can run side-by-side with other applications on the same computer without any conflicts.

The Virtual Package Editor lets you manage the folders and files that will be available in the virtual environment. It also lets you define registry keys, values, and data for your virtual package. In addition, you can use the Virtual Package Editor to define the targets for your virtual application, and define entry points such as shortcuts for each target. These entry points enable end users to launch an App-V application from within the virtual environment.

Including Files and Folders

The Virtual Package Editor lets you manage the files and folders that are in your virtual package. This includes the files and folders in the root folder, the virtual file system (VFS) folder, and the SoftGridUserSettings folder. The Virtual Package Editor also lets you extract folders and files from the App-V package file (.sft) to a location that you specify.

Adding a Predefined Folder to the VFS Folder in an App-V Package

The Virtual Package Editor lets you add various folders that use a CSIDL constant or an SFT constant to the VFS folder. Examples of such folders are CSIDL_APPDATA and SFT_PROGRAM_FILES_X64. At run time, the folder is mapped to the appropriate location in the virtual environment.

Task: ***To add a predefined folder to the VFS folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the **VFS** folder, point to **Add Predefined Folder**, and then click the appropriate folder.

The Virtual Package Editor adds the predefined folder to the VFS folder.

Adding a Folder to an App-V Package

The Virtual Package Editor lets you add folders to your virtual package.

Task: ***To add a folder to an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. Do one of the following:
   - To add an existing folder and all of its contents to the package, in the **Files and Folders** explorer, right-click the location where you want to add a new folder and click **Add Folder**. The **Browse For Folder** dialog box opens, enabling you to select the folder that you want to add.
   - To add a new empty folder to the package, in the **Files and Folders** explorer, right-click the location where you want to add a new folder and click **Add New Folder**. The Virtual Package Editor adds a new folder.

The Virtual Package Editor adds the folder to your virtual package.

Tip • *To change the name of the new folder, do one of the following:*

- *In the **Files and Folders** explorer, click the name of the new folder and then press F2. The Virtual Package Editor highlights the name of the folder, enabling you to edit it as needed.*
- *In the **Files and Folders** explorer, right-click the name of the new folder and then click **Rename**. The Virtual Package Editor highlights the name of the folder, enabling you to edit it as needed.*
- *Select the new folder, and in the **Settings** window, change the value of the **Name** setting.*

Adding a File to an App-V Package

The Virtual Package Editor lets you add files to your virtual package.

Task: ***To add a file to an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the location where you want to add a new file and click **Add Files**. The **Select files to add to the virtual package** dialog box opens.
3. Select the file that you want to add and then click **Open**.

Tip • *To select multiple files in a folder, hold down the CTRL key while clicking files.*

The Virtual Package Editor adds the file or files that you selected to the virtual package.

Configuring a File or Folder in an App-V Package

The Virtual Package Editor lets you configure settings for the files and folders in your virtual package. The settings set information such as file attributes, whether a file is part of feature block 1, and the file or folder data type (application data or user data).

Task: ***To configure the settings for a file or folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, click the file or folder that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Files and Folders View.

Tip • *The Virtual Package Editor lets you configure the settings for more than one file or folder at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Setting the VFS Path for the Contents of a Predefined Folder in an App-V Package

If a file or folder should exist outside the App-V package's root folder on the virtual file system, the Virtual Package Editor lets you specify the VFS path for that file or folder. You may want to specify a VFS path if your virtual application tries to access the files in a folder by referring to a system folder to find the files (for example, looking up the Programs File folder) instead of using a relative path. You may also want to specify a VFS path if end users need to be able to find files when using a file browse dialog box (for example, for templates that are stored in a common file folder).

You can manually specify the VFS path by editing the App-V VFS Path setting for each file and folder individually. As an alternative, you can let the Virtual Package Editor configure the path for all of the files and subfolders in a folder.

Task: ***To set the VFS path for the contents of a folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the predefined folder that contains the files and folders whose VFS path you want to configure, and then click **Set VFS Path on Contents**.

The Virtual Package sets the VFS path of each of the files and subfolders in the predefined folder.

Moving a File or Folder in an App-V Package

The Virtual Package Editor lets you move files and folders in your virtual package from one location to another using drag and drop functionality.

Task: ***To move a file or folder in an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, drag a file or folder that you want to move to the appropriate location.

Extracting Files and Folders from the App-V Package

When you are editing an App-V package in the Virtual Package Editor, you may want to extract one or more files and folders from the package and save them to a local or network location. Doing so enables you to view the physical files that are streamed within the .sft file. If you extract a folder that contains subfolders and files, the Virtual Package Editor uses the same folder structure when saving the folder and its contents.

Task: ***To extract a file from an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the file that you want to extract, and then click **Extract**. The **Save As** dialog box opens.
3. Browse to the location where you want to save the file.
4. In the **File name** setting, enter a new name for the file if you want to use a different one.
5. Click the **Save** button.

The Virtual Package Editor saves the file in the location that you specified.

Task: ***To extract a folder and its contents from an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the folder that you want to extract, and then click **Extract**. The **Browse for Folder** dialog box opens.
3. Select the folder that you want to contain the extracted folder and its contents.

The Virtual Package Editor saves the folder, its subfolders, and its files in the location that you specified.

Removing a File or Folder in an App-V Package

The Virtual Package Editor lets you remove files and folders from your App-V package. If you remove a folder, all of its contents—including any subfolders and files—are also removed.

Task: ***To remove a file or folder from an App-V package:***

1. In the View List under **Application Data**, click **Files and Folders**.
2. In the **Files and Folders** explorer, right-click the file or folder that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the file or folder from your virtual package.

Editing the Virtual Registry

The Registry view enables you to define registry keys, values, and data for your App-V package. This view also lets you configure isolation options for selected registry keys. Isolation options indicate how the isolation environment provides access to system resources that the application needs: you can choose to override one or more keys on the client system, or you can choose to create a merged view of one or more keys for the virtual environment.

Note that the registry entries that are configured in the Registry view affect only the application in your App-V package. They do not affect any other App-V packages that are streamed to the Application Virtualization Client, and they do not affect any products that are installed to the client system.

Keys, Value Names, and Value Data

The registry consists of machine data and user data. A key is a named location in the registry. A key can contain subkeys, a default value, and named values. A default value is a value without a name. All other values associate a name with some data: the value name identifies where to store it, and the value data is the data in that storage.

Note that the terms *key* and *subkey* are relative. In the registry, a key that is below another key can be referred to as a *subkey* or as a *key*, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding a Registry Key to a Virtual Package

The Virtual Package Editor enables you to add registry keys to your App-V package so that they are available in the virtual environment.

Task: ***To add a registry key to your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. Do one of the following:
   - To add machine registry data, expand the **MACHINE** node.
   - To add user registry data, expand the **USER** node. If your App-V package does not contain any user registry data, you may need to add the **USER** node. To do so, right-click the **Registry** explorer, point to **Add Predefined Key**, and click **USER**.
3. In the **Registry** explorer, right-click the registry entry that you want to contain the new key, and then click **Add Key**.

The Virtual Package Editor adds a registry key with a default REG_SZ value.

Tip • *To change the name of the new registry key, do one of the following:*

- *In the **Registry** explorer, click the name of the new registry key and then press F2. The Virtual Package Editor highlights the name of the key, enabling you to edit it as needed.*
- *In the **Registry** explorer, right-click the name of the new registry key and then click **Rename**. The Virtual Package Editor highlights the name of the key, enabling you to edit it as needed.*
- *Select the new registry key, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Registry Key in a Virtual Package

If your virtual package includes one or more registry keys, you can configure each key's settings to specify information such as the value data, as well as whether the key in the App-V package should override the corresponding key on the client system.

Task: ***To configure a registry key in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, click the registry key that you want to configure.
3. In the **Settings** window, configure the settings for the registry key as needed. For details about each setting, see Registry View.

Tip *• The Virtual Package Editor lets you configure the settings for more than one registry entry at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

The App-V Override setting lets you specify whether you want the selected registry key in the App-V package to either see only the registry content that is inside the App-V package for a key and all its subkeys, or see a merged view of the registry content inside the App-V package and of the registry content on the physical client system. If you want to change the value of this setting for all of a registry key's subkey simultaneously, see Configuring the App-V Override Setting for All of the Subkeys Under One or More Keys.

Configuring the App-V Override Setting for All of the Subkeys Under One or More Keys

If your virtual package includes a registry key that has multiple subkeys whose App-V Override setting should be configured with the same value, you can quickly change the value of the App-V Override setting for all of that key's subkeys simultaneously.

You can also quickly configure the App-V Override setting for all of the subkeys that belong to multiple parent keys.

Task: ***To configure the App-V Override setting for all of the subkeys under one or more keys:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, click the registry key that you want to configure.

   If you want to configure the **App-V Override** setting for all of the subkeys under multiple parent keys, select all of the applicable parent keys. To select multiple consecutive keys, select the first registry key, press and hold SHIFT, and select the last key. To select multiple nonconsecutive keys, select one key, press and hold CTRL, and select each additional key.
3. Right-click the selected key or keys and then click the appropriate command:

   - **Override Child Keys**—If you want to select **Yes** for the **App-V Override** setting of each subkey under the selected keys, select this option.

     The App-V application sees the registry content that is inside the App-V package for this key and all subkeys. Thus, the application does not see any registry content from the physical client system.
   - **Merge Child Keys**—If you want to select **No** for the **App-V Override** setting of each subkey under the selected keys, select this option.

     The App-V application sees a merged view of the registry content inside the App-V package and of the registry content on the physical client system. If the registry key has subkeys on the physical client system but not in the App-V package, these keys are merged into the registry view that is available to the App-V application. However, registry values that are on the physical client system and that are in registry keys that also exist in the App-V package are not merged into the App-V application's registry view.

Adding a Registry Value to a Registry Key in a Virtual Package

The Virtual Package Editor enables you to add registry values to registry keys to your App-V package so that they are available in the virtual environment.

Task: ***To add a registry value to a registry key in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, right-click the registry key that you want to contain the new value, and then click **Add Value**.

The Virtual Package Editor adds a registry value.

Tip • *To change the name of the new registry value, do one of the following:*

- *In the **Registry** explorer, click the name of the new registry value and then press F2. The Virtual Package Editor highlights the name of the value, enabling you to edit it as needed.*
- *In the **Registry** explorer, right-click the name of the new registry value and then click **Rename**. The Virtual Package Editor highlights the name of the value, enabling you to edit it as needed.*
- *Select the new registry value, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Registry Value and Its Value Data in a Virtual Package

If your virtual package includes one or more registry values, you can configure each value's settings to specify information such as the value data and the value type.

Task: ***To configure a registry value and its value data in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, click the registry value that you want to configure.
3. In the **Settings** window, configure the settings for the registry key as needed. For details about each setting, see Registry View.

Tip • *The Virtual Package Editor lets you configure the settings for more than one registry entry at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Registry Value from a Registry Key in a Virtual Package

Task: ***To remove a registry value a registry key in your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, right-click the registry value that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the registry value from the registry key in your virtual package.

Removing a Registry Key from a Virtual Package

The Virtual Package Editor lets you remove registry keys from your App-V package. If you remove a registry key, all of its subkeys and values are also removed.

Task: ***To remove a registry key from your virtual package:***

1. In the View List under **Application Data**, click **Registry**.
2. In the **Registry** explorer, right-click the registry key that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the registry key from your virtual package.

Defining Targets in a Virtual Application

The Virtual Package Editor lets you define each of the targets in your virtual package. Each target in your virtual package can contain one or more entry points, such as shortcuts, for each target. Entry points enable end users to launch each target in an App-V package from within the virtual environment.

Adding a Target to a Virtual Package

The Virtual Package Editor enables you to add one or more targets for your App-V package.

Task: ***To add a target to your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. Right-click the **Targets** explorer and then click **Add Target**.

The Virtual Package Editor adds a new target.

Tip • *To change the name of the new target, do one of the following:*

- *In the **Targets** explorer, click the name of the new target and then press F2. The Virtual Package Editor highlights the name of the target, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new target and then click **Rename**. The Virtual Package Editor highlights the name of the target, enabling you to edit it as needed.*
- *Select the new target, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Target in a Virtual Package

The Virtual Package Editor lets you configure settings for a target in your App-V package to specify information such as the file name and version number of the target file.

Task: ***To configure a target in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the target that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Target Settings.

Tip • *The Virtual Package Editor lets you configure the settings for more than one target at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Target from a Virtual Package

Task: ***To remove a target from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the target that you want to remove, and then click **Remove**.

Creating Shortcuts to the Virtual Application on the Client System

Shortcuts offer quick access to a virtual application. You can configure your virtual package so that it adds shortcuts for your virtual application on the desktop, the Start menu, and various other locations on the client system.

Each shortcut that you create is part of a target in your virtual package. Each target in your virtual package can contain one or more entry points, such as shortcuts, for each target. At the target level, the Virtual Package Editor enables you to configure information such as the file in your virtual application that you want to launch, the icon that should be used for the target, and the command-line arguments that should be used to launch the file. For a shortcut, the Virtual Package Editor enables you to configure the display name and location of the shortcut.

Adding a Shortcut for a Virtual Package

The Virtual Package Editor enables you to add to your App-V package a shortcut that points to your App-V application.

You can add a shortcut to any target in your App-V package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task: ***To add a shortcut to a target in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to contain the new shortcut, right-click the **Shortcuts** folder, and then click **Add Shortcut**.

The Virtual Package Editor adds the shortcut to the Targets explorer.

Tip • *To change the display name of the new shortcut, do one of the following:*

- *In the **Targets** explorer, click the name of the new shortcut and then press F2. The Virtual Package Editor highlights the name of the shortcut, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new shortcut and then click **Rename**. The Virtual Package Editor highlights the name of the shortcut, enabling you to edit it as needed.*
- *Select the new shortcut, and in the **Settings** window, change the value of the **Display Name** setting.*

Configuring a Shortcut in a Virtual Package

The Virtual Package Editor lets you configure the display name of a shortcut. It also lets you configure the shortcut's location, such as on the desktop, the Start menu, or various other locations on the client system.

Tip • *To configure information such as the file in your virtual application that you want to launch, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the shortcut. To learn more, see Configuring a Target in a Virtual Package.*

Task: ***To configure a shortcut in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the shortcut that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Shortcut Settings.

Tip • *The Virtual Package Editor lets you configure the settings for more than one shortcut at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Shortcut from a Virtual Package

Task: ***To remove a shortcut from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the shortcut that you want to remove, and then click **Remove**.

Using Environment Variables in a Virtual Environment

Environment variables are name and value pairs that can be accessed by your virtual application. Environment variables in a virtual package are stored in the virtual registry.

The Virtual Package Editor enables you to create environment variables that you want to be available to your virtual application in the virtual environment.

Note that the environment variables that are configured in the Shortcuts view of the Virtual Package Editor affect only the application in your App-V package. They do not affect any other App-V packages that are streamed to the Application Virtualization Client, and they do not affect any products that are installed to the client system.

Setting an Environment Variable in a Virtual Package

The Virtual Package Editor enables you to add to your App-V package add a shortcut that points to your App-V application.

You can add a shortcut to any target in your App-V package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task: ***To set an environment variable in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to be associated with the new environment variable, right-click the **Environment Variable** folder, and then click **Add Variable**.

The Virtual Package Editor adds the environment variable to the Targets explorer.

Tip • *To change the name of the new environment variable, do one of the following:*

- *In the **Targets** explorer, click the name of the new environment variable and then press F2. The Virtual Package Editor highlights the name of the environment variable, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new environment variable and then click **Rename**. The Virtual Package Editor highlights the name of the environment variable, enabling you to edit it as needed.*
- *Select the new environment variable, and in the **Settings** window, change the value of the **Name** setting.*

Configuring an Environment Variable in a Virtual Package

The Virtual Package Editor lets you set the name and value of an environment variable in your App-V package.

Task: ***To configure an environment variable in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the environment variable that you want to set.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Environment Variable Settings.

Tip • *The Virtual Package Editor lets you configure the settings for more than one environment variable at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing an Environment Variable from a Virtual Package

Task: ***To remove an environment variable from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the environment variable that you want to remove, and then click **Remove**.

Configuring File Extension Associations for the Virtual Application

The Virtual Package Editor enables you to associate file extensions with a target in your virtual package. Once you have associated a file extension with a target, you can set up one or more verbs, such as Open or Print, for the file extension. When an end user double-clicks a file with that file extension in the virtual environment, the file opens in your virtual application. If an end user right-clicks a file with that file extension in the virtual environment, the context menu shown by Windows Explorer includes the display names of the verbs that are set up for the file extension.

Each file extension that you create is part of a target in your virtual package. Each target in your virtual package can contain one or more entry points, such as file extensions and shortcuts. At the target level, the Virtual Package Editor enables you to configure information such as the file in your virtual application that you want to launch and the command-line arguments that should be used to launch the file. For a file extension, the Virtual Package Editor enables you to configure information such as the MIME type and the ProgId of the file extension.

Adding a File Extension to a Target in a Virtual Package

The Virtual Package Editor enables you to add a file extension to your App-V application.

You can associate a file extension with any target in your App-V package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task: ***To add a file extension to a target in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to contain the new file extension, right-click the **File Extensions** folder, and then click **Add File Extension**.

The Virtual Package Editor adds the file extension to the Targets explorer.

Tip • *To specify the file extension, do one of the following:*

- *In the **Targets** explorer, click the name of the new file extension and then press F2. The Virtual Package Editor highlights the name of the file extension, enabling you to enter the file extension as needed.*
- *In the **Targets** explorer, right-click the name of the new file extension and then click **Rename**. The Virtual Package Editor highlights the name of the file extension, enabling you to enter the file extension as needed.*
- *Select the new file extension, and in the **Settings** window, change the value of the **Extension** setting.*

*It is not necessary to enter the dot—for example, enter **txt** instead of **.txt**.*

Configuring a File Extension in a Virtual Package

The Virtual Package Editor lets you configure information such as the MIME type and the ProgId of the file extension.

Tip • *To configure information such as the file in your virtual application that you want to launch for the file extension, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the file extension. To learn more, see Configuring a Target in a Virtual Package.*

Task: ***To configure a file extension in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the file extension that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see File Extension Settings.

Tip • *The Virtual Package Editor lets you configure the settings for more than one file extension at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Adding a Verb to a File Extension in a Virtual Package

The Virtual Package Editor enables you to add a verb to a file extension to your App-V application.

Task: ***To add a verb to a file extension in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the appropriate target, right-click the file extension that you want to be associated with the new verb, and then click **Add Verb**.

The Virtual Package Editor adds the verb to the Targets explorer.

Tip • *To change the verb name, do one of the following:*

- *In the **Targets** explorer, click the name of the new verb and then press F2. The Virtual Package Editor highlights the name of the verb, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new verb and then click **Rename**. The Virtual Package Editor highlights the name of the verb, enabling you to edit it as needed.*
- *Select the new verb, and in the **Settings** window, change the value of the **Name** setting.*

Configuring a Verb for a File Extension in a Virtual Package

The Virtual Package Editor lets you configure information such as the display name and dynamic data exchange (DDE) settings of a file extension's verb.

Tip • *To configure information such as the file in your virtual application that you want to launch with the verb, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the verb. To learn more, see Configuring a Target in a Virtual Package.*

Task: ***To configure a verb in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the verb that you want to configure.

3. In the **Settings** window, configure the settings as needed. For details about each setting, see Verb Settings for a File Extension.

Tip • *The Virtual Package Editor lets you configure the settings for more than one verb at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Removing a Verb from a File Extension in a Virtual Package

Task: ***To remove a verb from a file extension in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the verb that you want to remove, and then click **Remove**.

Removing a File Extension from a Virtual Package

Task: ***To remove a file extension from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the file extension that you want to remove, and then click **Remove**.

Creating Scripts that Run Before or After the App-V Application Is Streamed or Launched

The Virtual Package Editor lets you add to your App-V package scripts that you want to be run at various stages: before or after the App-V application is streamed to the client, or before or after the App-V application is launched. You can create scripts that make changes that your application requires, either in the App-V environment or on the client system outside the virtual environment. For example, you may want to launch a script that ensures that a particular file or registry entry exists, or that synchronizes data inside the virtual environment with data outside the virtual environment.

Two different types of scripting are available:

- **Single command (HREF)**—The App-V package references an external script or an executable file. The contents of the script are launched directly on the client system. The Command Prompt window is not displayed unless the process that is being called opens it.
- **Command script (SCRIPTBODY)**—The contents of the script are stored in the App-V package and copied to a temporary .bat file in the root folder (typically under the Q drive) of the App-V package on the client system. The .bat file is launched from a visible Command Prompt window.

You can use either type of scripting to call an executable file that exists in the folder on the virtual application server where the App-V package is stored.

Each script that you create is associated with a target in your virtual package. Each target in your virtual package can contain one or more entry points, such as scripts and shortcuts. At the target level, the Virtual Package Editor enables you to configure information such as the file in your virtual application that you want to launch and the command-line arguments that should be used to launch the file. For a script, the Virtual Package Editor enables you to configure information such as when you want the script to be launched.

Adding a Script to a Target in a Virtual Package

The Virtual Package Editor enables you to add to your virtual package a script that you want to be run at various stages: before or after the App-V application is streamed to the client, or before or after the App-V application is launched.

You can associate a script with any target in your App-V package. To learn how to add a new target, see Adding a Target to a Virtual Package.

Task: ***To add a script to a target in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, under the target that you want to contain the new script, right-click the **Scripting** folder, and then click **Add Script**.

The Virtual Package Editor adds the script to the Targets explorer.

Configuring a Script in a Virtual Package

The Virtual Package Editor lets you configure information such as when you want the script to be launched.

Tip • *To configure information such as the file in your virtual application that you want to be associated with the script, the icon that should be used for the target, and the command-line arguments that should be used to launch the file, configure the settings for the target that contains the shortcut. To learn more, see Configuring a Target in a Virtual Package.*

Task: ***To configure a script in your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the script that you want to configure.
3. In the **Settings** window, configure the settings as needed. For details about each setting, see Scripting Settings.

   To learn how to trigger the appropriate behavior if the script fails, see Causing the App-V Application to Close After a Script Failure.
4. In the **Script** window, enter your script: either a one-line command or the body of the script.

Tip • *The Virtual Package Editor lets you configure the settings for more than one script at a time. To learn more, see Configuring the Value of a Setting for More Than One Item at a Time.*

Guidelines for Entering Script

Note the following guidelines when you are entering script in the Shortcuts view.

- If you are entering script for the command script (Scriptbody) type of script, you can use any script language that the client operating system supports.
- If you are entering script for the command script (Scriptbody) type of script, you can use command processor commands such as CHDIR and MOVE in your script.

  If you are entering script for the single command (HREF) type of script, command processor commands cannot be used, unless you launch `cmd.exe` to run the script.
- The Virtual Package Editor automatically adds the proper escape sequence for a newline character (\n), if appropriate, to the App-V package that it generates. Thus, to end a line and start a new one, simply press Enter; avoid entering a newline character (\n).

  The Virtual Package Editor also automatically adds the backslash character (\) if you enter a backslash, resulting in a double backslash (\\) in the App-V package. Thus, if you are specifying a path, do not use the escape character.
- If you are entering script for the single command (HREF) type of script, ensure that you enter only one line of script. If you enter more than one line, the Virtual Package Editor ignores all of the lines after the first line.

  Note that if you are using the command script (Scriptbody) type of script, you can enter more than one line of script.

Causing the App-V Application to Close After a Script Failure

If you have added a script that you want to be run for your App-V package, you can also specify the conditions under which the App-V package should be closed or the App-V package streaming should be stopped.

Note • *If you specify Post-shutdown for the Event setting of the script, any values that you specify for the Success Result setting and the Abort Result setting are ignored.*

Task: ***To specify success and abort behavior for a script in your App-V package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, click the script that you want to configure.
3. Enter the appropriate value in the **Success Result** setting or the **Abort Result** setting, as appropriate.

Note • *In some versions of App-V, unexpected results could occur if you specify values for both the Success Result setting and the Abort Result setting.*

Table 10-5 • Client Behavior for Various App-V Package Settings and Script Return Codes

Script Return Code	Value for the Success Result Setting	Value for the Abort Result Setting	Behavior of the Application Virtualization Client
0	Any value or empty	0	The Application Virtualization Client silently aborts the application startup.
1	1	0 or null	The Application Virtualization Client proceeds with the next part of the App-V application streaming or launching.
1	Null	0	The Application Virtualization Client proceeds with the next part of the App-V application streaming or launching.
2	1	1 or null	The Application Virtualization Client fails to stream the package or start the application, and it displays an error message. The message includes an error code in the format of xxxxxx-xxxxxx18-0000000*n*, where *n* represents the return code of the script. The error code is written to the log file.

Removing a Script from a Virtual Package

Task: ***To remove a debug tool from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the script that you want to remove, and then click **Remove**.

Configuring Virtual Services

Windows services are executable files that Windows–based systems run in the background to manage various system tasks. A service is an executable file, but it must be designed as a service; you cannot automatically use an arbitrary executable file as a service. Windows services can be configured to run every time that the system starts or on demand when needed. The Virtual Package Editor enables you to configure services that you want to include in your App-V package so that they are available in the virtual environment.

Adding a Virtual Service to a Virtual Package

The Virtual Package Editor enables you to add virtual services to your App-V package so that they are available in the virtual environment.

Task: ***To add a virtual service to your virtual package:***

1. In the View List under **System Configuration**, click **Virtual Services**.
2. Right-click the **Virtual Services** explorer and then click **Add Virtual Service**.

The Virtual Package Editor adds a new service.

Tip • *To change the name of the new service, do one of the following:*

- *In the **Targets** explorer, click the name of the new service and then press F2. The Virtual Package Editor highlights the name of the service, enabling you to edit it as needed.*
- *In the **Targets** explorer, right-click the name of the new service and then click **Rename**. The Virtual Package Editor highlights the name of the service, enabling you to edit it as needed.*
- *Select the new service, and in the **Settings** window, change the value of the **Display Name** setting.*

Configuring a Virtual Service in a Virtual Package

If your virtual package includes one or more virtual services, you can configure each service's settings to specify information such as the service name, the path to the executable file, and the type of service.

Task: ***To configure a virtual service in your virtual package:***

1. In the View List under **System Configuration**, click **Virtual Services**.
2. In the **Virtual Services** explorer, click the service that you want to configure.
3. Configure the settings for the virtual service as needed. For details about each setting, see Virtual Services View.

Removing a Virtual Service from a Virtual Package

Task: ***To remove a virtual service from your virtual package:***

1. In the View List under **System Configuration**, click **Virtual Services**.
2. In the **Virtual Services** explorer, right-click the service that you want to remove, and then click **Remove**.

The Virtual Package Editor removes the service from your virtual package.

Testing and Troubleshooting Virtual Packages

Once you have made the necessary changes for the files, folders, shortcuts, services, and other elements of your virtual package, you are ready to test the package. The Virtual Package Editor lets you optionally include the AdminStudio App-V Application Launcher with your App-V package if you want to test a newly saved App-V package locally before moving it to a deployment server.

ConflictSolver includes several Application Conflict Evaluators (ACEs) that may help you identify potential conflicts between different App-V packages, and between App-V packages and Windows Installer–based installations. Part of your testing strategy may involve using ConflictSolver to detect potential conflicts.

If you encounter issues when running the App-V package, you can add to the package shortcuts that launch the Command Prompt window and the registry editor. These types of shortcuts may help you debug problems with an App-V package, since they enable you to examine the file system and view the registry while the virtual application is running in the virtual environment.

Using the App-V Application Launcher to Test the Virtual Package

You can use the AdminStudio App-V Application Launcher to test a newly saved App-V package on a test machine before moving it to a deployment server.

If you want the Virtual Package Editor to include the App-V Application Launcher whenever you save the App-V package, enable the App-V Launcher save option. To learn more, see Saving a Virtual Package.

Requirements for Using the App-V Application Launcher

The machine on which you use the App-V Application Launcher to test an App-V package must meet the following requirements:

- The Microsoft Application Virtualization Client must be installed.
- The version of the Microsoft Application Virtualization Client that is present should be equal to or newer than the minimum client version of the App-V package. The Virtual Package Editor displays the minimum client version of the App-V package in the General Information view.
- File streaming must be enabled because the App-V Application Launcher publishes the App-V package from a local file path. If file streaming is not enabled, the App-V Application Launcher displays an informative message asking if it can enable this functionality.

Starting the App-V Application Launcher

When you save an App-V package in the Virtual Package Editor and the App-V Launcher save option is enabled, the Virtual Package Editor adds the App-V Application Launcher (`AppVLauncher.exe`) to the same folder as the App-V package every time that you save an App-V package.

Task: ***To use the App-V Application Launcher for testing a virtual package:***

1. In the Virtual Package Editor, open the App-V package that you want to test.
2. Do one of the following:
   - On the **View** menu, click **Show in Explorer**.
   - Press CTRL+E.
   - On the toolbar, click the **Explore** button.

   A Windows Explorer window opens. It shows the folder that contains the .sft file, the .xml manifest file, the `AppVLauncher.exe` file, possibly one or more .osd files, and possibly an icon folder.

   If you have saved the App-V package as a new version one or more times, the folder may also contain a subfolder for each earlier version. The subfolders are named bkup_*N*, where *N* represents the version number of the App-V package.
3. Copy the contents of the folder (except for the bkup_*N* folders) to a test machine that meets the aforementioned App-V Application Launcher requirements. The `AppVLauncher.exe` file should be in the same folder as the .sft file.
4. Double-click the `AppVLauncher.exe` file.

If the App-V package has one target defined in the Shortcuts view (that is, if the App-V package has only one .osd file), the App-V Application Launcher starts the App-V application.

If the App-V package has more than one target defined in the Shortcuts view (that is, if the App-V package has two or more .osd files), the App-V Application Launcher displays a dialog box that lists each target, and it lets you select the one that you want to launch.

Note • *The first time that you use the App-V Application Launcher to run an application in an App-V package, the entire package is published to that machine; this includes all of the package's shortcuts and file extension associations in the package. If you then use the App-V Application Launcher to run any application in the App-V package again, the App-V Application Launcher unpublishes the package (and its shortcuts and file extension associations) before republishing the package.*

Also note that the `AppVLauncher.exe` file requires elevation. If you want to be able to test your App-V package in a locked-down environment where end users will not have elevated privileged, you may want to use the App-V Application Launcher once to launch and publish your App-V package with elevated privileges. Once you have done that, you can use the published shortcuts and file extension associations to start your application.

Using Debug Tools with a Virtual Package

The Virtual Package Editor lets you incorporate the following tools in a virtual package:

- **Cmd.exe (x86)**—The 32-bit version of Cmd.exe on the local machine runs, and it has access to the virtual environment.
- **Cmd.exe (x64)**—The 64-bit version of Cmd.exe on the local machine runs, and it has access to the virtual environment. This requires Microsoft Application Virtualization Client 4.6 or later, and it also require that the App-V package be published in a 64-bit environment.
- **Regedit.exe**—Regedit.exe on the local machine runs, and it has access to the virtual environment.

These debug tools may help you troubleshoot issues in the virtual package.

Important • *It is recommended that you use the debug tools only for testing. Before you release your virtual package, remove these tools from the package.*

Task: ***To add a debug tool to your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. Right-click the **Targets** explorer, point to **Add Debug Tool**, and then point to the appropriate command:
   - Cmd.exe (x86)
   - Cmd.exe (x64)
   - Regedit.exe

The Virtual Package Editor adds the debug tool to the Targets explorer. The debug tool includes a shortcut that you can use to launch the tool in the virtual environment.

Adding a debug tool to a virtual package is similar to adding a target. Therefore, if appropriate, you can perform other tasks for the debug tool, just as you can for a target. For example, if you want to launch a script whenever you launch the Command Prompt window to simulate run-time behavior, you can add a script to the debug tool.

Task: ***To remove a debug tool from your virtual package:***

1. In the View List under **Application Data**, click **Shortcuts**.
2. In the **Targets** explorer, right-click the debug tool that you want to remove, and then click **Remove**.

Using the Virtual Package Editor to Resolve Application Conflict Evaluators (ACEs) in App-V Packages

You can use ConflictSolver to run Application Conflict Evaluators (ACEs) and identify potential conflicts between different App-V packages, and between App-V packages and Windows Installer packages. The following table lists ACE rules that pertain to App-V packages, as well as troubleshooting tips for resolving the issues in the Virtual Package Editor.

Table 10-6 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages

ACE Rule	Rule Type	Description
ACE200	Conflict	This ACE indicates that two or more packages contain a shortcut with the same display name and location. To resolve this ACE in an App-V package, use the Shortcuts view to do one of the following: • Select the shortcut, and then modify the value in the Display Name setting or the Location setting. • Remove the shortcut from the App-V package.
ACE201	Best Practice	This ACE indicates that a target in the package has a hard-coded path such as C:\...\, which may not be present in a virtual environment. To resolve this ACE in an App-V package, change the path of the target to use a variable instead of a hard-coded path: 1. In the View List under **Application Data**, click **Shortcuts**. 2. In the **Targets** explorer, select the target that contains the hard-coded path. 3. In the **Target** setting, replace the existing path with a path that uses a CSIDL constant or an SFT constant—such as CSIDL_APPDATA or SFT_PROGRAM_FILES_X64—instead of the hard-coded path. Note that if there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.

Table 10-6 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Rule	Rule Type	Description
ACE202	Best Practice	This ACE indicates that the command-line arguments for a target in the package include a hard-coded path such as C:\...\, which may not be present in a virtual environment. To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path: 1. In the View List under **Application Data**, click **Shortcuts**. 2. In the **Targets** explorer, select the target that contains the hard-coded path. 3. In the **Arguments** setting, replace the existing path with a path that uses a CSIDL constant or an SFT constant—such as CSIDL_APPDATA or SFT_PROGRAM_FILES_X64—instead of the hard-coded path. Note that if there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.
ACE203	Best Practice	This ACE indicates that the working directory for a target in the package include a hard-coded path such as C:\...\, which may not be present in a virtual environment. To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path: 1. In the View List under **Application Data**, click **Shortcuts**. 2. In the **Targets** explorer, select the target that contains the hard-coded path. 3. In the **Working Directory** setting, replace the existing path with a path that uses a CSIDL constant or an SFT constant—such as CSIDL_APPDATA or SFT_PROGRAM_FILES_X64—instead of the hard-coded path. Note that if there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.

Table 10-6 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Rule	Rule Type	Description
ACE204	Conflict	This ACE indicates that two or more packages have the same package GUID; therefore, the two packages cannot be deployed simultaneously as separate packages. If you are creating an update package that can upgrade earlier versions of the virtual package, the package GUID should stay the same. If you are creating a new package that can be deployed simultaneously as another package, the package GUID in one of the packages must be changed. To change the package GUID, save the package as a new package. To learn more, see Saving a Virtual Package.
ACE205	Conflict	This ACE indicates that two or more packages have the same name. This is not advisable from a best practice perspective, and it may cause some issues if you try to simultaneously deploy the App-V packages. To resolve this ACE: 1. In the View List under **Package Information**, click **General Information**. 2. In the **Name** setting, replace the duplicate name with a unique name.
ACE206	Conflict	This ACE indicates that two or more packages have support for the same file extension. However, a file extension can be registered with only one application at a time. To resolve this ACE, you may need to decide which package should contain the file extension association and which should not. Then you can use the Virtual Package Editor to remove the appropriate file extension.

Table 10-6 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Rule	Rule Type	Description
ACE207	Conflict	This ACE indicates that two or more packages have the same long or short name for the root folder. These names must be unique because two packages with the same root folder name cannot be deployed simultaneously. To resolve this ACE: 1. In the View List under **Package Information**, click **General Information**. 2. In the **Root Folder Name** setting, replace the duplicate folder name with a unique folder name. Note that instances of the old package's root folder name may still exist in location-related configuration data, such as in registry entries, .ini files, or XML files in the App-V package. The root folder name is not updated in those areas automatically if you change the root folder name in the General Information view. Therefore, if you know that the old package contains configuration data, you may need to identify where it is. Then you can use the Virtual Package Editor to update the root folder name as necessary. For example, you may want to use the Virtual Package Editor to extract a configuration file from the package. Next, you can update the root folder name in the file. In the Virtual Package Editor, you would then delete the old file from the App-V package, and add the updated file.
ACE208	Best Practice	This ACE indicates that an App-V package does not contain any shortcuts. You can ignore this ACE if one of the following is true: • This package is intended to be used as a dependency by a different App-V package through Dynamic Suite Composition. In this case, you need to edit the other App-V package and select this App-V package as a dependency in the Dependencies view. • This package is intended to be used as a plug-in. In this case, you need to create a shortcut to the application for which this is a plug-in. Some common examples include Office and Internet Explorer. If end users need to be able to launch this App-V package independently, consider adding a target to the App-V package if necessary, and then adding a shortcut to the target.

Table 10-6 • Troubleshooting Tips for Resolving ACE Issues in App-V Packages (cont.)

ACE Rule	Rule Type	Description
ACE215	Conflict	This ACE indicates that the App-V package contains a shortcut (App-V application) that uses the same name and version as one in another package. The combination of the name and version should be unique for shortcuts in different packages, since only one application is published and available at any given time. To resolve this ACE in an App-V package, use the Shortcuts view to do one of the following: • Select the target that contains the shortcut, and then modify the value in the Name setting or the Target Version setting. • Remove the shortcut from the App-V package.

Virtual Package Editor Reference

Reference information for the Virtual Package Editor is organized into the following sections:

Table 10-7 • Reference Sections

Section	Description
Virtual Package Editor Start Page	Provides information about the Start Page in the Virtual Package Editor.
Virtual Package Editor Menu, Toolbar, and Window Reference	Describes various components of the Virtual Package Editor user interface, including menus, toolbars, and windows.
Virtual Package Editor Dialog Box Reference	Contains reference information on each of the dialog boxes that are displayed in the Virtual Package Editor.
Virtual Package Editor View Reference	Describes each of the views that are displayed in the Virtual Package Editor.

Virtual Package Editor Start Page

The Virtual Package Editor Start Page is a tab that provides quick access to product information, to recently opened projects, and to Virtual Package Editor resources. The Start Page includes the following sections:

Table 10-8 • Sections on the Start Page

Section	Description
Package Tasks	Click a package task to quickly create a new virtual package or open an existing virtual package.
Help Topics	Frequently accessed help topics are listed in this section. To access the entire Virtual Package Editor Help Library from anywhere within the Virtual Package Editor, you can press F1, click the Help button, or click one of the appropriate commands on the Help menu.
Recent Packages	The section in the middle of the Start Page lists your most recently accessed virtual packages, their locations, and the dates on which they were last modified.
Resources	The Resources section contains links to connect you to helpful product information.

Virtual Package Editor Menu, Toolbar, and Window Reference

This section describes the various components of the Virtual Package Editor user interface, including menus, toolbars, and windows.

Menus in the Virtual Package Editor

The menus in the Virtual Package Editor are located on the menu bar, which is at the top of the Virtual Package Editor interface. Each menu contains a list of commands. Some of these commands have icons next to them so that you can quickly associate the command with the icon.

Each of the menus in the Virtual Package Editor is described in this section:

- File
- Edit
- View
- Window
- Help

File Menu in the Virtual Package Editor

The following table lists the File menu commands, as well as associated keyboard shortcuts and icons.

Table 10-9 • File Menu Commands

Command	Shortcut	Icon	Description
New	CTRL+N		Creates a new virtual package.
Open	CTRL+O		Opens an existing virtual package.
Close			Closes the currently selected tab.
Save	CTRL+S		Saves the currently selected virtual package as a new package. To learn about the various save options, see Saving a Virtual Package.
Save As			Enables you to save the currently selected virtual package with a new name and location. Also lets you specify whether you want to save the virtual package as an upgrade package or as a new package. To learn about the various save options, see Saving a Virtual Package.
Start Page			Opens or closes the Start Page.
Save Options			Provides several commands: • Include App-V Launcher • Append Package Version • Build Wrapper MSI • Include SFT in Wrapper MSI • Compress Wrapper MSI To learn about the various save options, see Saving a Virtual Package.
1, 2, 3, 4, 5, 6			Opens one of the recently accessed virtual packages.
Exit			Closes the open virtual packages and closes the Virtual Package Editor.

Edit Menu in the Virtual Package Editor

The following table lists the Edit menu commands, as well as associated keyboard shortcuts and icons.

Table 10-10 • Edit Menu Commands

Command	Shortcut	Icon	Description
Undo	CTRL+Z		Undoes the last action performed.
Redo	CTRL+Y		Reverses the last action that was performed with the Undo command.
Cut	CTRL+X		Removes the currently selected text and places it on the Clipboard.
Copy	CTRL+C		Copies the currently selected text to the Clipboard.
Paste	CTRL+V		Inserts the contents of the Clipboard at the insertion point, and replaces any selected text.
Remove	Delete		Deletes the currently selected text.
Refresh	F5		Refreshes the currently selected view.

View Menu in the Virtual Package Editor

The following table lists the View menu commands, as well as associated keyboard shortcuts and icons.

Table 10-11 • View Menu Commands

Command	Shortcut	Icon	Description
Toolbars			Provides the following commands: • **Standard**—Shows or hides the Standard toolbar. • **Customize**—Opens the Customize dialog box, which lets you add or remove toolbar buttons or implement other toolbar customization. If you created custom toolbars, they are also listed.
Settings			Opens or closes the Settings window.
Output Window			Opens or closes the Output window.
Status Bar			Opens or closes the status bar.

Table 10-11 • View Menu Commands (cont.)

Command	Shortcut	Icon	Description
Show in Explorer	CTRL+E		Opens the folder that contains the currently selected virtual package in a Windows Explorer window. If an .sft tab is not currently selected, or if you are creating a new .sft file but have not yet saved it, this command is disabled.

Window Menu in the Virtual Package Editor

The following table lists the Window menu commands.

Table 10-12 • Window Menu Commands

Command	Description
1, 2, 3	Opens the tab for the corresponding virtual package.

Help Menu in the Virtual Package Editor

The following table lists the Help menu commands.

Table 10-13 • Help Menu Commands

Command	Description
Contents	Displays the Contents tab of the help library.
Index	Displays the Index tab of the help library.
Search	Displays the Search tab of the help library.
About Virtual Package Editor	Displays the Virtual Package Editor dialog box, where you can find version information.

Standard Toolbar in the Virtual Package Editor

The Virtual Package Editor interface offers one built-in toolbar that gives you quick access to frequently used menu commands: the Standard toolbar.

The following table lists all of the buttons on the Standard toolbar.

Table 10-14 • Standard Toolbar Buttons

Button	Name	Shortcut	Description
	New	CTRL+N	Creates a new virtual package.
	Open	CTRL+O	Opens an existing virtual package.
	Save	CTRL+S	Saves the currently selected virtual package. To learn about the various save options, see Saving a Virtual Package.
	Cut	CTRL+X	Removes the currently selected text and places it on the Clipboard.
	Copy	CTRL+C	Copies the currently selected text to the Clipboard.
	Paste	CTRL+V	Inserts the contents of the Clipboard at the insertion point, and replaces any selected text.
	Explore	CTRL+E	Opens the folder that contains the currently selected virtual package in a Windows Explorer window. If an .sft tab is not currently selected, or if you are creating a new .sft file but have not yet saved it, this button is disabled.
	Help	F1	Opens the help library.

Script Window

The Virtual Package Editor displays a Script window when you select a script in the Shortcuts view. Use the Script window to enter the script that you want to be run at various stages: before or after the App-V application is streamed to the client, or before or after the App-V application is launched.

To learn how to configure a script's settings and enter your script in the Script window, see Configuring a Script in a Virtual Package.

Settings Window

The Settings window in the Virtual Package Editor contains a grid that lists information about the item that is selected in an open view.

The following table describes the buttons that are displayed above the settings in the Settings window.

Table 10-15 • Controls in the Settings Window

Name of Control	Icon	Description
Categorized		Sorts the settings according to categories.
Alphabetical		Sorts the settings alphabetically.

Output Window

The Output window displays task-specific information such as details about the virtual package that you are opening. It also shows save information.

Note that closing the Output window clears its contents. The Virtual Package Editor automatically shows the Output window whenever a task—such as saving or opening a virtual package—generates output.

Virtual Package Editor Dialog Box Reference

This section of the documentation describes dialog boxes that are displayed in the Virtual Package Editor.

- Browse for Folder
- Edit Value
- Save As
- Select a File
- Select a Folder
- Select Files to Add to the Virtual Package

Browse for Folder Dialog Box

The Virtual Package Editor displays the Browse for Folder dialog box when you click the Browse Your Computer button in settings such as the Working Directory setting for the target that is selected in the Shortcuts view. The Browse for Folder dialog box lets you select an existing folder on your machine or create a new folder.

The Browse for Folder dialog box is also displayed when you right-click an item in the Files and Folders view and then click Add Folder. In this case, the Browse for Folder dialog box lets you select a local or network folder that you want to add to your virtual package.

Edit Value Dialog Box

The Virtual Package Editor displays the Edit Value dialog box in the following scenarios:

- You click the ellipsis button (...) in the Value Data setting for a registry value in the Registry view. The ellipsis button is displayed in this setting for a value type of REG_BINARY or REG_MULTI_SZ. Use the Edit Value dialog box to specify the value data.
- You click the ellipsis button (...) in the Group Dependencies setting or the Service Dependencies setting for a service in the Virtual Services view. Use the Edit Value dialog box to specify the groups or services that are required by the virtual service in your virtual package.

Table 10-16 • Edit Value Dialog Box Settings

Setting	Description
Value Name	This read-only setting shows the name of the value that you are editing.
Value Data	Enter the value data (the registry value data, the group dependencies, or the service dependencies). Specify each value on a separate line.

Save As Dialog Box

The Save As dialog box lets you specify the name and location where you want to save your virtual package. This dialog box lets you specify whether you want to save the current virtual package as an update package or as a new package.

Task: ***To access the Save As dialog box:***

On the **File** menu, click **Save As**.

Table 10-17 • Save As Dialog Box Settings

Setting	Description
Virtual Package	Enter the path and file name that you want to use for the .sft file. As an alternative, you can click the ellipsis button (...) to browse to the file.
Save an update package	To save the virtual package as an update package that can upgrade earlier versions of the virtual application, click this option.

Table 10-17 • Save As Dialog Box Settings (cont.)

Setting	Description
Save as a new package	To save the virtual package as a new package that you can deploy alongside earlier versions of the virtual package in the same virtual environment, click this option.

Select a File Dialog Box

The Virtual Package Editor displays the Select a File dialog box when you click the Browse This Package button in settings such as the Target setting and the Icon setting for a target in the Shortcuts view. The Select a File dialog box displays the directory tree for your virtual package, enabling you to select the appropriate file.

Select a Folder Dialog Box

The Virtual Package Editor displays the Select a Folder dialog box when you click the Browse This Package button in settings such as the Working Directory setting for a target in the Shortcuts view. The Select a Folder dialog box displays the directory tree for your virtual package, enabling you to select the appropriate folder that you want to use as the working directory for the selected target.

Select Files to Add to the Virtual Package Dialog Box

The Virtual Package Editor displays the Select Files to Add to the Virtual Package dialog box when you right-click a folder in the Files and Folders view and then click Add Files. The dialog box lets you select local or network files that you want to add to your virtual package.

Virtual Package Editor View Reference

The View List in the left pane is a navigational element that consists of folders and subnodes that you can click to open various areas within the Virtual Package Editor. Each folder and subnode in the View List represents a view within the Virtual Package Editor. The Virtual Package Editor View Reference section describes each of those views.

Package Information View

The Package Information view contains links to other views that you can use to configure general settings about your virtual package, and identify dependencies.

Table 10-18 • Views under the Package Information View

View	Description
General Information	The General Information view is where you specify basic information such as the name of the virtual package and details such as the package GUID and version number. This view also shows history information such as each date on which the package was saved.
Dependencies	The Dependencies view is where you specify other App-V packages that the open App-V package requires.

General Information View

The General Information view contains basic information about your virtual package. It contains a History pane, plus a number of settings that you can configure.

History Pane

The History pane in the General Information view shows read-only information such as each date on which the package was saved, the GUID that corresponds with each saved version, and the version of App-V that was used when building the App-V package. Each time that you save your .sft file, the Virtual Package Editor adds a new history entry to the History pane.

General Information Settings

The General Information view settings are organized into the following main categories:

- App-V Settings
- App-V Server URL Settings

App-V Settings

Use the App-V area of the General Information view settings to view or specify basic information such as the name of the virtual package and details such as the package GUID and version number. The following settings are available in this area.

Table 10-19 • App-V Settings in the General Information View

Setting	Description
Name	Enter a name for the App-V package. ***Tip*** • *If your virtual package contains multiple applications, you can specify the name that identifies the entire package. For example, Microsoft Office could be used to identify a package that contains Microsoft Word and Microsoft Excel applications that run in the same virtual environment.*
Comments	Enter a short description of the App-V package. This setting is optional.
OS	The OS setting and its subsettings let you specify one or more operating systems on which the application can be run. If the application is operating system independent, select False for all of the OS subsettings.
Package GUID	This read-only setting shows the globally unique identifier (GUID) that is associated with the App-V package.
Package Version	This read-only setting shows the version number of the App-V package.
Minimum Client Version	This read-only setting shows the minimum version number of the Application Virtualization Client that is required to use the App-V package.
Root Folder Name	This setting specifies the root folder of the App-V package's file system. During run time, the Application Virtualization Client mounts the package's file system to the App-V virtual drive; the Q drive is the default. The long and short names of the root folder must be unique because two packages with the same root folder name cannot be deployed simultaneously.
Feature Block 1 Size	This read-only setting indicates the size of the primary feature block, the part of the App-V package that is required to start the application.
Total File Size	This read-only setting indicates the size of the entire package.
Compressed	Specify whether you want the App-V package's contents to be compressed.

Table 10-19 • App-V Settings in the General Information View (cont.)

Setting	Description
Enforce Security Descriptors	This read-only setting indicates whether security descriptors of the application in the App-V package are enforced after it is deployed to the client system.
Allow Local Interaction	Specify whether you want named objects (events, mutexes, semaphores, file mappings, and mailslots) and COM objects to be created in the global namespace. Available options are: • **Yes**—Named objects and COM objects are created in the global namespace, allowing virtual applications to interact with the applications of the client operating system. • **No**—Named objects and COM objects are isolated inside the virtual environment.

App-V Server URL Settings

Use the App-V Server URL area of the General Information view settings to specify the location from which the App-V package is streamed. The following settings are available in this area.

Table 10-20 • App-V Server URL Settings in the General Information View

Setting	Description
Protocol	Select the protocol that you want to use to stream the sequenced application package from the virtual application server to an Application Virtualization Client. Available options are: • **RTSP**—The real-time streaming protocol streams the App-V package. This is the default option. • **RTSPS**—The real-time streaming protocol with transport layer security streams the App-V package. • **FILE**—The App-V package are streamed from a file share. • **HTTP**—The hypertext transport protocol streams the App-V package. • **HTTPS**—The secure hypertext transport protocol streams the App-V package.

Table 10-20 • App-V Server URL Settings in the General Information View (cont.)

Setting	Description
Host	Specify the host—the virtual application server or the load balancer in front of a group of virtual application servers that stream the App-V package to the Application Virtualization Client. You can either specify a static host name or IP address, or you can enter `%SFT_SOFTGRIDSERVER%` to indicate an environment variable. ***Note*** • *If you enter `%SFT_SOFTGRIDSERVER%`, you must set up the SFT_SOFTGRIDSERVER system environment variable on each Application Virtualization Client. The value of this environment variable should be the name or IP address of the host.* *When you assign the variable on a client system, any Application Virtualization Client session that is running on the system must be closed and reopened; otherwise, the session is not aware of the new application source.*
Port	Specify the port on which the virtual application server or the load balancer listens for Application Virtualization Client requests for the package. The default port is 554.
Path	Specify the relative path on the virtual application server where the App-V package is stored. This is also the path from which the App-V package is streamed. If the App-V package is stored in a subdirectory of CONTENT, the path must be specified in this setting; otherwise, you can leave this setting blank.

Dependencies View

The Dependencies view is where you specify other App-V packages that the open App-V package requires.

Icons in the Dependencies View

The Dependencies explorer in the Dependencies view uses different icons to help you distinguish between different types of items. Following is a list of the possible icons in the Dependencies view.

Table 10-21 • Icons in the Dependencies View

Icon	Description
	This icon identifies the root node—the Dependencies explorer.
	This icon identifies an App-V package (.sft) that the open App-V package (the primary App-V package) requires.
	This icon identifies a target in the primary App-V package. It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies.

Settings in the Dependencies View

The settings that are displayed in the Dependencies view differ, depending on whether you select an App-V package (.sft file) or a target in this view:

- .sft Settings
- Target Settings

.sft Settings

When you select an .sft file in the Dependencies view, the following settings are available.

Table 10-22 • .sft Settings in the Dependencies View

Setting	Description
Name	This read-only setting shows the name of the required .sft file.
GUID	This read-only setting shows the globally unique identifier (GUID) that is associated with the required App-V package.
SysGuard File	This read-only setting shows the folder and name of the required App-V package's SysGuard file (`osguard.cp`). The SysGuard file describes how the virtual environment needs to be set up.
HREF	Enter the URL for the published location of the required App-V package on the virtual application server. Typically, this location matches the App-V server URL for the App-V package that contains the dependency.

Target Settings

When you select a target in the Dependencies view, the following settings are available.

Table 10-23 • Target Settings in the Dependencies View

Setting	Description
Name	This read-only setting shows the name of the target that is associated with the required App-V package.
Mandatory	Specify whether the required App-V package is mandatory in order for the primary package (the App-V package that you are editing in the Virtual Package Editor) to run properly. Note that if the dependency is mandatory, the primary package cannot run without loading the required package.

Application Data View

The Application Data view contains links to other views that you can use to specify the files, folders, and registry entries that make up your virtual application, and to configure entry points that launch your virtual application.

Table 10-24 • Views under the Application Data View

View	Description
Files and Folders	The Files and Folders view is where you specify the files and folders that are in the App-V package. This view also lets you extract folders and files from the App-V package file (.sft) to a location that you specify.
Registry	The Registry view enables you to define registry keys, values, and data for your App-V package.
Shortcuts	The Shortcuts view lets you define targets, as well as entry points that launch the targets in the virtual environment.

Files and Folders View

The Files and Folders view is where you specify the files and folders that are in the App-V package. This includes folders and files that are in the App-V package's root folder, the virtual file system (VFS) folder, and the SoftGridUserSettings folder. This view also lets you extract folders and files from the App-V package file (.sft) to a location that you specify.

Icons in the Files and Folders View

The Files and Folders view uses different icons to help you distinguish between different types of files and folders. Following is a list of the possible icons in the Files and Folders view.

Table 10-25 • Icons in the Files and Folders View

Icon	Description
	This icon identifies a folder.
	This icon identifies a folder that uses a VFS path.
	This icon identifies a folder that is configured to be overwritten when the App-V package is upgraded and streamed from the App-V server to the client.
	This icon identifies a folder that uses a VFS path and that is configured to be overwritten when the App-V package is upgraded and streamed from the App-V server to the client.
	This icon identifies a file.

Table 10-25 • Icons in the Files and Folders View (cont.)

Icon	Description
	This icon identifies a file that uses a VFS path.
	This icon identifies a file that is configured to be overwritten when the App-V package is upgraded and streamed from the App-V server to the client.
	This icon identifies a file that uses a VFS path and that is configured to be overwritten when the App-V package is upgraded and streamed from the App-V server to the client.
	This icon identifies a font file.
	This icon identifies a font file that uses a VFS path.
	This icon identifies a font file that is configured to be overwritten when the App-V package is upgraded and streamed from the App-V server to the client.
	This icon identifies a font file that uses a VFS path and that is configured to be overwritten when the App-V package is upgraded and streamed from the App-V server to the client.

Settings in the Files and Folders View

When you select a file or folder in the Files and Folders view, the following settings are available.

Table 10-26 • File and Folder Settings

Setting	Description
Name	Enter the name of the file or folder in the App-V package.
Short File Name	Enter the name of the file or folder using the 8.3 format.
Path	This read-only setting shows the location of the file or folder in the App-V package.
Size	This read-only setting shows the size of the file. This setting applies to files; it does not apply to folders.

Table 10-26 • File and Folder Settings (cont.)

Setting	Description
Attributes	To set various attributes for the selected file or folder, use the following settings: • **Read-Only**—Specify whether the file is read-only—protected from being changed or accidentally deleted. • **Hidden**—Specify whether the file or folder is visible in directory listings when default folder viewing options are enabled. • **System**—Specify whether the file or folder is a system file or folder that the operating system uses. • **Archive**—Specify whether the file or folder should be archived. Some applications use this attribute to determine whether to back up a file or folder. • **Normal**—Specify whether the file should have its other attributes configured. Selecting True for this setting is valid only if False is selected for the other True-False attributes. • **Not Content-Indexed**—Specify whether you want to avoid indexing the contents of the file or folder for faster searching.
Created	This read-only setting shows the date and time when the file or folder was created.
Modified	This read-only setting shows the date and time when the file or folder was last modified.
Source Path	If the file has not yet been saved as part of the App-V package, this read-only setting shows the fully qualified path of the source file. This setting applies to files; it does not apply to folders.
Register Font	Specify whether you want the font to be registered in the virtual environment.
App-V Feature Block 1	Specify whether the file is part of the primary feature block, the part of the App-V package that is required to start the application. This setting applies to files; it does not apply to folders.
App-V GUID	This read-only setting shows the globally unique identifier (GUID) that is associated with the file or folder.
App-V Version	For a file, this read-only setting shows the version number of the App-V package that corresponds with the last time that the file was modified. For any folder other than the root folder, this read-only setting shows the latest version number of the App-V package.

Table 10-26 • File and Folder Settings (cont.)

Setting	Description
App-V VFS Path	If the file or folder should needs to be available from a location outside the package's root folder, specify the path to which to virtualize the file or folder. Note that this path should include the name of the file or folder, and it should begin with a CSIDL constant such as %CSIDL_APPDATA%, if such a constant for the path is available. If a constant is not available, you can enter a hard-coded path starting with a drive letter.
App-V Data Type	Specify the data type of the file or folder. Available options are: • **Application Data**—Changes to the file or folder are saved for all users of the App-V package on the client system. • **User Data**—Changes to the file or folder are saved for only the logged-on user. • **Unspecified**—The data type of the file or folder is not configured.
App-V Override	Specify whether you want the Application Virtualization Client to overwrite the file or folder when the App-V package is upgraded and streamed from the App-V server to the client. If you select No, the Application Virtualization Client determines whether to overwrite the file or folder during an upgrade. In general, if User Data is selected for the App-V Data Type setting, the file or folder is not overwritten during an upgrade; otherwise, the file or folder is overwritten. ***Note*** • *According to Microsoft documentation, the override functionality is deprecated as of Application Virtualization Client 4.5.*

Registry View

The Registry view enables you to define registry keys, values, and data for your App-V package. This view also lets you configure isolation options for selected registry keys. Isolation options indicate how the isolation environment provides access to system resources that the application needs: you can choose to override one or more keys on the client system, or you can choose to create a merged view of one or more keys for the virtual environment.

Note that the registry entries that are configured in the Registry view affect only the application in your App-V package. They do not affect any other App-V packages that are streamed to the Application Virtualization Client, and they do not affect any products that are installed to the client system.

Icons in the Registry View

The Registry view uses different icons to help you distinguish between different types of registry entries. Following is a list of the possible icons in the Registry view.

Table 10-27 • Icons in the Registry View

Icon	Description
	This icon identifies for the root node—the Registry explorer. This icon also identifies the MACHINE and USER predefined keys.
	This icon identifies a registry key.
	This icon identifies a registry key that is configured to override the registry content on the physical client system.
	This icon identifies a REG_NONE, REG_SZ, REG_EXPAND_SZ, or REG_MULTI_SZ registry value.
	This icon identifies a REG_BINARY, REG_DWORD, or REG_QWORD registry value.

Settings in the Registry View

The Registry view contains the registry entries that are configured for your App-V package. When you select a registry key or value in the Registry view, the following settings are available.

Table 10-28 • Registry Key and Value Settings in the Registry View

Setting	Description
Name	Enter the name of the selected registry key or value.
Value Data	Enter the data for the selected registry value, or (for a registry key) enter the data for the selected key's default value.
Value Type	Select the appropriate type of registry data for the selected registry entry. If you select the REG_QWORD type, ensure that the operating system of the client system supports it. This setting applies to registry values; it does not apply to registry keys. The default value of a registry key is always the REG_SZ type of value.

Table 10-28 • Registry Key and Value Settings in the Registry View (cont.)

Setting	Description
App-V Override	Specify whether you want the selected registry key in the App-V package to override the corresponding key on the client system. Available options are: • **Yes**—The App-V application sees the registry content that is inside the App-V package for this key and all subkeys. Thus, the application does not see any registry content from the physical client system. • **No**—The App-V application sees a merged view of the registry content inside the App-V package and of the registry content on the physical client system. If the registry key has subkeys on the physical client system but not in the App-V package, these keys are merged into the registry view that is available to the App-V application. However, registry values that are on the physical client system and that are in registry keys that also exist in the App-V package are not merged into the App-V application's registry view. ***Tip*** • *If your virtual package includes a registry key that has multiple subkeys whose App-V Override setting should be configured with the same value, you can quickly change the value of the App-V Override setting for all of that key's subkeys simultaneously. You can also quickly configure the App-V Override setting for all of the subkeys that belong to multiple parent keys. To learn how, see Configuring the App-V Override Setting for All of the Subkeys Under One or More Keys.* This setting applies to registry keys; it does not apply to registry values.

Shortcuts View

The Shortcuts view lets you define the targets for your virtual application. This view also lets you define entry points, such as shortcuts, for each target. Entry points enable end users to launch each target in an App-V package from within the virtual environment.

Icons in the Shortcuts View

The Targets explorer in the Shortcuts view uses different icons to help you distinguish between different types of items. Following is a list of the possible icons in the Shortcuts view.

Table 10-29 • Icons in the Shortcuts View

Icon	Description
	This icon identifies the root node—the Targets explorer.
	This icon identifies a target in the primary App-V package. It is recommended that all of the targets in your virtual package be associated with each of the package's dependencies.

Table 10-29 • Icons in the Shortcuts View (cont.)

Icon	Description
	This icon identifies a container that holds the shortcuts that are associated with a target.
	This icon identifies a shortcut.
	This icon identifies a container that holds the environment variables that are associated with a target.
	This icon identifies an environment variable.
	This icon identifies a container that holds the file extensions that are associated with a target.
	This icon identifies a file extension.
	This icon identifies a verb for a file extension.
	This icon identifies a container that holds the scripts that are associated with a target.
	This icon identifies a script.

Settings in the Shortcuts View

Use the Targets explorer in the Shortcuts view to define each target in your App-V package. Under each target, you can configure associated shortcuts, environment variables, file extensions, and scripting. The settings that are displayed in the Shortcuts view differ, depending on what type of item you select in this view. For descriptions of each of the settings in the Shortcuts view, see the following:

- Target Settings
- Shortcut Settings
- Environment Variable Settings
- File Extension Settings
- Verb Settings for a File Extension
- Scripting Settings

Target Settings

When you select a target in the Shortcuts view, the following settings are available.

Table 10-30 • Target Settings in the Shortcuts View

Setting	Description
Name	Enter the name of the target.
Target	Enter the path and file name of the file in the App-V package or on the client system that should be launched when end users launch the target's shortcut or other entry point. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the target file.
Icon	Enter the path to the icon file (.ico, .exe, or .dll) that contains the icon resource for the shortcut. The location can be in the App-V package, or on your computer or the network. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the icon. If the icon file that you specify contains more than one icon resource, after the icon path, add a comma and then the index number. For example, 0 refers to the first icon in the file, 1 refers to the second icon, and 2 refers to the third icon. To specify the third icon in **C:\MyIcons.dll**, enter the following: `C:\MyIcons.dll,2`
Target Version	Enter the version number of the target.
Arguments	Enter the command-line arguments for the shortcut. These arguments work in the same way as any other command-line arguments. For example, you can link a file to an executable file or cause an executable file to run silently by passing command-line arguments. ***Note*** • *Verify that the syntax is correct because the Virtual Package Editor does not do this.*
Working Directory	Enter the working directory for the target. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to select or create the directory. The working directory is the default directory that is displayed in standard file-opening and file-saving dialog boxes, as well as the current directory used by the App-V application.
Terminate Children	Specify whether you want all of the applications and processes that were launched by the App-V application to be closed when the end user exits the App-V application.

Shortcut Settings

When you select a shortcut in the Shortcuts view, the following settings are available.

Table 10-31 • Shortcut Settings in the Shortcuts View

Setting	Description
Display Name	Enter the name of the shortcut as it should appear on the client system.
Location	Enter the path to the folder that contains the shortcut file. As an alternative to manually entering a value, you can click the Browse Your Computer button or select a predefined folder from the drop-down list.

Environment Variable Settings

When you select an environment variable in the Shortcuts view, the following settings are available.

Table 10-32 • Environment Variable Settings in the Shortcuts View

Setting	Description
Name	Enter the name of the environment variable that you want to configure for the App-V application.
Value	Enter the path or value for this environment variable. To enter multiple paths, separate the paths with a semicolon (;).

File Extension Settings

When you select a file extension in the Shortcuts view, the following settings are available.

Table 10-33 • File Extension Settings in the Shortcuts View

Setting	Description
Extension	To associate a file extension with the App-V application, enter the extension. It is not necessary to enter the dot—for example, enter `txt` instead of `.txt`.
Description	Enter the description text that you want to display for this file extension in the Application Virtualization Client.
MIME	Enter the MIME type that is associated with the file extension.

Table 10-33 • File Extension Settings in the Shortcuts View (cont.)

Setting	Description
ProgId	Enter the program identifier—ProgId, also known as *application identifier* or *tag name*—that you want to associate with the file extension. A file type's ProgId is an arbitrary string, but it should be unique on the client system. One ProgId naming convention is to append the word `file` to your extension without a dot—the .ext extension might use the ProgId *extfile*. Another convention is to name a file-type ProgId after the application that is used to open the file type, as in `SampleApp.Document`. For example, an **.xyz** file extension could point to an **xyzfile** ProgId, and all of the **.xyz** file-type information would be registered under **xyzfile**.
Icon	Enter the path to the icon file (.ico, .exe, or .dll) that contains the icon resource for the file extension. The location can be in the App-V package, or on your computer or the network. As an alternative to manually entering a value, you can click the Browse This Package button or the Browse Your Computer button to browse to the icon. If the icon file that you specify contains more than one icon resource, after the icon path, add a comma and then the index number. For example, 0 refers to the first icon in the file, 1 refers to the second icon, and 2 refers to the third icon. To specify the third icon in the icon file, enter the following after the icon file path: `, 2`
Perceived Type	Select the appropriate type of file.
File Type Attributes	To set various file type attributes for the file association, select the appropriate option for each of the following settings: • **Open Is Safe**—Specify whether the open verb can be used safely for downloaded files that have this file extension. • **Always Unsafe**—Specify whether files that have this file extension should be considered to be a possible security risk. • **Always Show Extension**—Specify whether you want the file extension to be displayed with the file name, even if the client system is configured to hide extensions for known file types. • **No Recent Documents**—Specify whether you want to exclude files with this extension from the Recent Documents folder.
Shell New Enabled	Specify whether you want to include this file type in the submenu that is displayed when end users click New on the context menu in Windows Explorer.

Verb Settings for a File Extension

The settings that are displayed when you select a verb under a file extension in the Shortcuts view are organized into the following main categories:

- General
- Dynamic Data Exchange

General Settings

Use the General area for a verb in the Shortcuts view to specify details such as the name and description of the verb.

Table 10-34 • General Settings for a Verb Under a File Extension in the Shortcuts View

Setting	Description
Name	Enter the name of the verb, such as **Open** or **Print**, that you want to be used when an end user right-clicks a file with the selected extension and then clicks the corresponding command. To include an underlined letter that indicates that end users can click the letter to select the command, precede that letter with an ampersand (&). For example, to display **Open** (with an underlined letter *O*) on the context menu for this file extension, enter the following: `&Open`
Display Name	Enter the text that you want to display for this verb on the context menu that Windows Explorer displays when an end user right-clicks a file with the associated file extension. To include an underlined letter that indicates that end users can click the letter to select the command, precede that letter with an ampersand (&). For example, to display **Open with SampleApp** (with an underlined letter *O*) on the context menu for this file extension, enter `&Open with SampleApp`. This setting is optional. If you do not specify a display name, the name of the verb as it appears in the Name setting is used on the context menu for a file with this file extension on the client system. Note that if you use one of the canonical verbs—such as open, print, or find—and you do not specify a display name, Windows automatically localizes the verb on each system.
Arguments	Enter the command-line arguments for the verb. ***Note*** • *Verify that the syntax is correct because the Virtual Package Editor does not do this.* ***Tip*** • *Use %1 in the argument in place of the file name. For example, if `-p %1` is the argument for the verb, and the end user right-clicks the file `C:\File.ext` and then clicks the command for this verb, the command-line argument becomes `-p C:\File.ext`. In some cases, it is necessary to enclose the %1 argument in quotation marks—as in `"%1"`—to correctly handle file names that contain spaces.*

Dynamic Data Exchange Settings

If your App-V application supports dynamic data exchange (DDE), use the Dynamic Data Exchange area for a verb in the Shortcuts view to specify DDE settings for the verb.

Table 10-35 • Dynamic Data Exchange Settings for a Verb Under a File Extension in the Shortcuts View

Setting	Description
DDE Command	Enter the DDE command for the verb. ***Note*** • *Verify that the syntax is correct because the Virtual Package Editor does not do this.* ***Tip*** • *Use %1 in the argument in place of the file name. In some cases, it is necessary to enclose the %1 argument in quotation marks—as in* ***"%1"****—to correctly handle file names that contain spaces.*
DDE Ifexec	Enter the DDE command that you want to use if the DDE conversation cannot be initiated.
DDE Application	Enter the application name that you want use to establish the DDE conversation. If you leave this setting blank, the DDE Command setting is used as the application name.
DDE Topic	Enter the name that you want to use as the topic name of the DDE conversation. If you leave this setting blank, *System* is used as the topic name.

Scripting Settings

When you select a script in the Shortcuts view, the following settings are available.

Table 10-36 • Scripting Settings in the Shortcuts View

Setting	Description
Event	Select the timing for the script that you want to launch. Available options are: • **Pre-stream**—The script or executable file runs after the end user launches the App-V application, but before feature block 1 of the application is streamed to the client system and before the virtual environment is set up. This type of script or executable file is run outside the virtual environment. • **Post-stream**—The script or executable file runs after the end user launches the App-V application and after feature block 1 of the application is streamed to the client system, but before the virtual environment is set up. This type of script or executable file is run either inside or outside the virtual environment. • **Pre-launch**—The script or executable file runs after the end user launches the App-V application, after feature block 1 of the application is streamed to the client system, and after the virtual environment is set up. This type of script or executable file is run either inside or outside the virtual environment. • **Post-launch**—The script or executable file runs after the App-V application is launched, but before the end user has access to the application. This type of script or executable file is run either inside or outside the virtual environment. • **Post-shutdown**—The script or executable file runs after the App-V application has been closed. This type of script is run outside the virtual environment.
Type	Specify the type of script that you want to be run. Available options are: • **Single command (HREF)**—The App-V package references an external script or an executable file. The contents of the script are launched directly on the client system. The Command Prompt window is not displayed unless the process that is being called opens it. • **Command script (SCRIPTBODY)**—The contents of the script are stored in the App-V package and copied to a temporary .bat file in the root folder (typically under the Q drive) of the App-V package on the client system. The .bat file is launched from a visible Command Prompt window. You can use either type of scripting to call an executable file that exists in the folder on the virtual application server where the App-V package is stored.

Table 10-36 • Scripting Settings in the Shortcuts View (cont.)

Setting	Description
Protect	Specify whether to run the script or executable file inside the virtual environment. Available options are: • **Yes**—The script or executable file is run inside the virtual environment. Protected scripts are useful for troubleshooting issues in the virtual environment. • **No**—The script or executable file is run outside the virtual environment. Unprotected scripts are useful for modifying the client system.
Wait	Specify whether to wait for the script or executable file to complete before continuing to the next scheduled task—either another script or the appropriate subsequent event.
Timeout	Enter the maximum number of seconds to wait for the script or executable file to complete before continuing. To wait until the script or executable file completes, enter the number 0 or leave this setting blank.
Success Result	Enter the return code that indicates that the script or executable file finished successfully. This setting is optional. For information on triggering the appropriate behavior if the script fails or succeeds, see Causing the App-V Application to Close After a Script Failure. ***Note*** • *If you specify Post-shutdown for the Event setting, any value that you specify for the Success Result setting is ignored.*
Abort Result	Enter the return code that indicates that the script or executable file failed. This setting is optional. For information on triggering the appropriate behavior if the script fails or succeeds, see Causing the App-V Application to Close After a Script Failure. ***Note*** • *If you specify Post-shutdown for the Event setting, any value that you specify for the Abort Result setting is ignored.*

System Configuration View

The System Configuration view contains links to another view that you can use to configure your virtual package so that the application can be run in the virtual environment.

Table 10-37 • Views under the System Configuration View

View	Description
Virtual Services	The Virtual Services view enables you to configure services that you want to include in your App-V package so that they are available in the virtual environment.

Virtual Services View

Windows services are executable files that Windows–based systems run in the background to manage various system tasks. A service is an executable file, but it must be designed as a service; you cannot automatically use an arbitrary executable file as a service. Windows services can be configured to run every time that the system starts or on demand when needed. The Virtual Services view enables you to configure services that you want to include in your App-V package so that they are available in the virtual environment.

The Virtual Services view shows the services that are configured for your App-V package. The Virtual Services view settings are organized into the following main categories:

- General Settings
- Error Handling Settings

General Settings

Use the General area of the Virtual Services view settings to specify information such as the name and location of the virtual service, as well as the type of service. This area is also where you specify when the service should be started. The following settings are available in this area:

Table 10-38 • General Settings in the Virtual Services View

Setting	Description
Name	Enter the name of the service. The name that you enter is used on the service's Properties dialog box. (To access an installed service's properties: In the Services administrative tool, right-click the service and then click Properties.) The maximum number of characters that is allowed is 256. The forward slash (/) and the backslash (\) are not valid characters for service names. The case of the name that you enter is preserved in the service control manager. Display name comparisons are always case-insensitive.

Table 10-38 • General Settings in the Virtual Services View (cont.)

Setting	Description
Display Name	Enter the name that you want to be displayed for this service in the service control manager and in other user interfaces. The maximum number of characters that is allowed is 256. The case of the display name that you enter is preserved in the service control manager. Display name comparisons are always case-insensitive.
Path to Executable File	Enter the fully qualified path to the executable file for the service. If the path contains one or more spaces, surround the path with quotation marks. You can include arguments that you want to be passed for a service that starts automatically. As an alternative to manually entering the path, you can click the Browse This Package button to browse to the executable file.
Service Type	Select the type of service that you are installing.
Service Is Interactive	Specify whether you want the service to be able to interact with users.
Startup Type	Specify when to start the service. Available options are: • **Automatic**—The service starts automatically when the system starts. • **On Demand**—The service starts when the service is requested through the service control manager. • **Never (Disabled)**—The service cannot be started.
Error Control	Select the appropriate severity of the error to indicate the action that the service control manager should perform if the service fails to start. Available options are: • **Ignore**—Ignore the error and continue with the service startup. • **Normal**—Log the error and continue with the service startup. • **Severe**—Log the error. If the last-known good configuration is being started, continue with the service startup. Otherwise, restart the system with the last-known good configuration. • **Critical**—Log the error. If the last-known good configuration is being started, the service startup fails. Otherwise, restart the system with the last-known good configuration.
Group	Enter the name of the load-ordering group, if any, of which this service is a member. Note that this setting can override the value of the Service Dependencies setting.

Table 10-38 • General Settings in the Virtual Services View (cont.)

Setting	Description
Group Dependencies	To specify one or more load-ordering groups that this service requires, click the ellipsis button (...) in this setting. When you do so, the Edit Value dialog box opens, enabling you to specify one or more groups. Enter each group on a separate line. The system attempts to start at least one member of the load-ordering group before starting this service.
Service Dependencies	To specify one or more services that this service requires, click the ellipsis button (...) in this setting. When you do so, the Edit Value dialog box opens, enabling you to specify one or more services. Enter each service on a separate line. The system attempts to start at least one member of the load-ordering group before starting this service.

Error Handling Settings

Use the Error Handling area of the Virtual Services view settings to specify what behavior should occur if the service fails.The following settings are available in this area:

Table 10-39 • Settings in the Virtual Services View

Setting	Description
Reset Period	Specify the amount of time (in seconds) between the reset intervals for the service's failure count. As an alternative, you can select one of the values from the list in this setting. The service control manager counts the number of times that the service has failed since the system was last restarted. When this interval has elapsed, the count is reset to the number 0 if the service has not failed during the reset period. When the service fails, the system performs an action that is specified for the First Error setting, the Second Error setting, or the Additional Errors setting, depending on how many errors have occurred since the last failure count reset or system restart. To indicate that the failure count should never be reset, select Never or enter a value of -1.
Reboot Message	Specify the message that should be displayed before the computer is restarted in response to an error. Note that Reboot the Computer must be listed as one of the action types for the First Error setting, the Second Error setting, or the Additional Errors setting; otherwise, the Reboot Message setting is ignored.
Command	Specify the command line that should be run if the Run a Command option is selected for the First Error, Second Error, or Additional Errors setting, and the first, second, or subsequent error occurs during service startup. Programs or scripts that you specify should not require input from end users. The command line that you specify is used to create a new process that runs under the same account as the service.
First Error	Select the action that you want the service control manager to perform the first time that the service fails.
First Action Delay	Specify the time (in milliseconds) that the service control manager should wait before performing the action that is specified in the First Error setting. As an alternative, you can select one of the values from the list in this setting.
Second Error	Select the action that you want the service control manager to perform the second time that the service fails.
Second Action Delay	Specify the time (in milliseconds) that the service control manager should wait before performing the action that is specified in the Second Error setting. As an alternative, you can select one of the values from the list in this setting.

Table 10-39 • Settings in the Virtual Services View (cont.)

Setting	Description
Additional Errors	Select the action that you want the service control manager to perform the third and subsequent times that the service fails.
Subsequent Action Delay	Specify the time (in milliseconds) that the service control manager should wait before performing the action that is specified in the Additional Errors setting. As an alternative, you can select one of the values from the list in this setting.

11

Creating Customized Virtual Applications

You can use AdminStudio to create customized virtual applications in the Microsoft App-V, VMware ThinApp, and Citrix XenApp virtual application formats.

Information about creating virtual applications is organized into the following sections:

- About Virtualization
- About the AdminStudio Virtualization Interface
- Creating Microsoft App-V Applications
- Creating ThinApp Applications
- Creating Citrix Profiles

***Note** • You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

About Virtualization

***Note** • This section provides a description of virtualization in general for those that are not familiar with it. It does not represent the architecture of any specific vendor.*

Virtualization enables you to isolate an application in its own environment so that it does not conflict with existing applications or modify the underlying operating system.

- Limitations of a Standard Installation Environment
- Benefits of Application Virtualization

Limitations of a Standard Installation Environment

A typical Windows application has dependencies on components that are shared by multiple applications. Applications access these shared system resources, such as the registry or Windows system files. When an installation author recognizes that their application references a shared system component, they include a merge module to install that component.

When one of these shared components is installed, it is possible that a previously installed version of the same component could be overwritten; this may cause the existing application to break. A similar problem could occur when one of these applications containing a shared component is uninstalled. Because of these possible problems, extensive compatibility testing needs to be performed before an application can be distributed in the enterprise environment.

The following diagram provides an example of two conflicting installed applications.

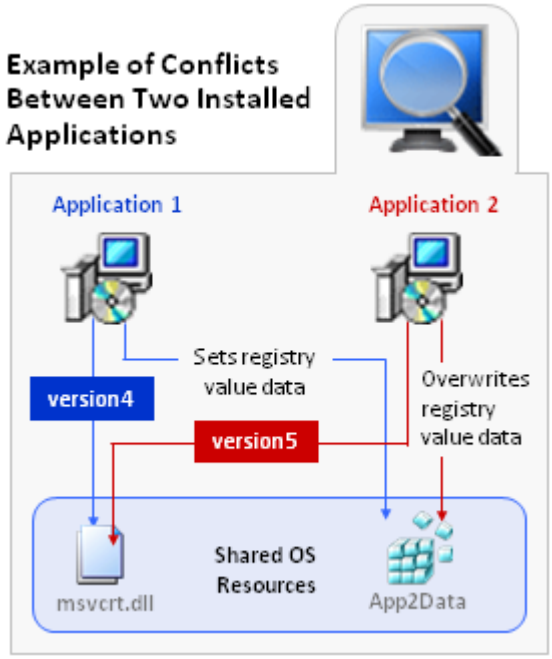


Figure 11-1: Example of Conflicts Between Two Installed Applications

Benefits of Application Virtualization

Virtual applications run in virtual environments that keep the application layer and the operating system layer separate. Each application includes its own configuration information in its virtual environment. As a result, many applications can run side-by-side with other applications on the same computer without any conflicts.

Even though virtual applications are not installed on the local machine, they exhibit the same functionality and access to local services as locally installed applications, and also nearly the same performance characteristics.

The following diagram provides an example of how application virtualization would solve the conflicts that are shown in the previous example.

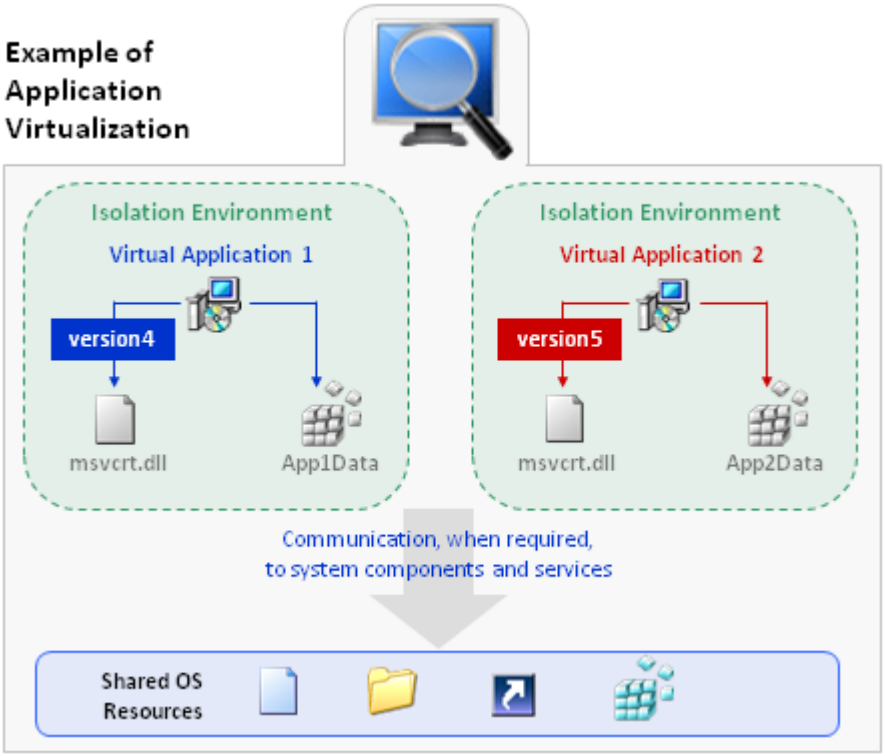


Figure 11-2: Example of Application Virtualization

Application virtualization allows the configuration of an application to be standardized to an isolation environment, rather than to an individual user's desktop machine. Application objects, files, and registry settings are contained within this isolation environment. Critical application resources are managed locally by the isolation environment, thus minimizing resource dependencies between applications.

Application virtualization greatly reduces the scope for conflicts between applications and, therefore, simplifies compatibility testing.

About the AdminStudio Virtualization Interface

Edition • *The Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant are included in the Virtualization Pack.*

Project • *The Microsoft App-V, ThinApp, and Citrix Assistants are available in the following project types:*

- *Basic MSI*
- *MSI Database (Direct Edit Mode)*
- *Transform (Direct MST Mode)*

AdminStudio provides the Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant to help you author a virtual application. You cannot configure a virtual application's options using the Installation Designer.

Information about the interface of these Assistants is organized in the following topics:

- About the Virtualization Assistant Tabs
- Using the More Options, Other Places, and Help Links Sections in a Virtualization Assistant
- Navigating in a Virtualization Assistant
- Opening the Installation Designer
- Showing or Hiding the Virtualization Assistants

About the Virtualization Assistant Tabs

Edition • *The Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant are included in the Virtualization Pack.*

When you create a new Basic MSI or MSI Database project, the **Microsoft App-V**, **VMware ThinApp**, and **Citrix XenApp** tabs are displayed in the AdminStudio interface. The home page of each Assistant has a diagram that illustrates the process of creating a virtual application using that technology.

You can work within these Assistants to create a project and configure its options and requirements. You can also use the Project Assistant or the Installation Designer to define the traditional Windows Installer version of your product installation.

How the Virtualization Assistants Work

When you create a new Basic MSI or MSI Database project, the **Microsoft App-V**, **VMware ThinApp**, and **Citrix XenApp** tabs are displayed in the AdminStudio interface.

The **Project Assistant** tab and the **Installation Designer** tab show the underlying framework for your product's Windows Installer–based installation. Some of these product elements are also displayed in the virtualization Assistants, where you can configure a virtual application's options and requirements.

Integration with the Project Assistant and the Installation Designer

Information that you enter in a virtualization Assistant is saved directly to the underlying project file. The Microsoft App-V Assistant, ThinApp Assistant, Citrix Assistant, Project Assistant, and Installation Designer run simultaneously. Any changes that you make in one are reflected instantly in the other. For example, if you remove a file in one of the virtualization Assistants, that file is no longer available in your project, and it does not appear in the Project Assistant or the Installation Designer.

Using the More Options, Other Places, and Help Links Sections in a Virtualization Assistant

Edition • *The Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant are included in the Virtualization Pack.*

The left column on each page of the virtualization Assistants contains one or more lists of links to help you in creating your installation and finding information:

- **More Options**—Provides additional configuration options relating to the specific virtualization Assistant page. These are less common options that complete the functionality of the Assistant.
- **Other Places**—The view in the Installation Designer that corresponds to the current virtualization Assistant page. Clicking the link launches the full Installation Designer and activates that view.
- **Help Links**—This list provides links to help topics pertinent to the current virtualization Assistant page.

Navigating in a Virtualization Assistant

Edition • *The Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant are included in the Virtualization Pack.*

Task: ***To navigate from one page of a virtualization Assistant to another, do one of the following:***

- To navigate directly to a specific page, click the appropriate icon in the navigation bar at the bottom of the page.
- To follow the assistant steps sequentially, do one of the following:
  - Click the Next or Back arrow buttons to move forward or backward.
  - Press CTRL+TAB to move to the next page and CTRL+SHIFT+TAB to move to the previous page.
- To move back to the Home page and view the overview diagram, click the Home button on the navigation bar.

Opening the Installation Designer

Edition • *The Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant are included in the Virtualization Pack.*

The **Installation Designer** tab displays the views in the AdminStudio interface. You can use this tab to configure your Windows Installer package. To open a view in the Installation Designer, click the **Installation Designer** tab.

Note • *The Installation Designer and the virtualization Assistants run simultaneously. Any changes that you make in one are reflected instantly in the other.*

Showing or Hiding the Virtualization Assistants

Edition • *The Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant are included in the Virtualization Pack.*

If you want to hide the Assistant for a virtualization technology that you do not use, you can hide it so that its tab is not displayed in the InstallShield interface. Similarly, if one of the virtualization Assistants is hidden, you can choose to display it.

Task: ***To show or hide a virtualization Assistant:***

On the **View** menu, click **Microsoft App-V Assistant**, **ThinApp Assistant**, or **Citrix Assistant**.

When an Assistant's command has a check mark next to it, the tab for that Assistant is shown in the InstallShield interface. When the check mark is not displayed, that Assistant is hidden.

Creating Microsoft App-V Applications

Edition • *The Microsoft App-V Assistant is included in the Virtualization Pack.*

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally. Instead, only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

Information about Microsoft App-V and creating Microsoft App-V applications is presented in the following sections:

- Overview of Microsoft Application Virtualization and the Microsoft App-V Assistant
- Using the Microsoft App-V Assistant to Create an App-V Application
- Microsoft App-V Assistant Reference

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

Overview of Microsoft Application Virtualization and the Microsoft App-V Assistant

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally. Instead, only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

The **Microsoft App-V Assistant**, which you can use to configure and build App-V applications, consists of the following pages:

Table 11-1 • Pages Comprising the Microsoft App-V Assistant

Page	Description
Microsoft App-V Assistant Home Page	Displays a diagram that illustrates the process of creating an App-V application.
Package Information Page	Enter the package name, enter a comment, specify any operating system requirements, and identify the deployment server.
Files Page	View existing files and folders, add and delete files, and set isolation options for selected files and folders. Isolation options specify how the virtual environment will provide access to files and folders requested by the App-V application.
Applications Page	Create, delete, include, exclude, or rename App-V application executables, which are derived from the shortcuts in its Windows Installer package.
Registry Page	Add, delete, or modify the registry settings, and set the isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to registry keys requested by the App-V application.
Dynamic Suite Composition Page	Use to control virtual application interaction between multiple App-V packages. On this page, you can select one or more packages that need to be linked to this App-V application in order for it to execute correctly.
Build Options Page	[Basic MSI Project mode] Select the releases that you want to build. [Direct Edit or Direct MST mode] To enable the Build function for an App-V application, select the **Build App-V application** option.

For information on Microsoft Application Virtualization and the App-V Assistant, see the following topics:

- About Microsoft Application Virtualization (App-V) and the App-V Assistant
- Components of an App-V Package
- About the Microsoft App-V Assistant

About Microsoft Application Virtualization (App-V) and the App-V Assistant

This section provides an overview of Microsoft Application Virtualization and its infrastructure, and explains the benefits of using the Microsoft App-V Assistant to create App-V packages:

- Overview
- Microsoft Application Virtualization Infrastructure
- Benefits of Using the Microsoft App-V Assistant

Overview

Microsoft Application Virtualization (App-V) enables you to deploy applications to end users without requiring the applications to be installed locally. Instead, only the App-V client needs to be installed on the client machines. Even though these virtual applications are never installed, they can communicate with the local operating system, middleware, plug-ins, and other applications. Using App-V enables you to centralize the deployment of applications and reduce application-to-application conflicts.

Because App-V applications are not installed on the client, there is minimal impact on the host operating system or other applications. As a result, application conflicts and the need for regression testing are dramatically reduced.

Using Microsoft Application Virtualization enables you to centralize the installation and management of deployed applications, and control access to applications. The App-V client presents to the end user a list of applications to which that end user has access.

Microsoft Application Virtualization Infrastructure

The Microsoft Application Virtualization (App-V) infrastructure includes:

- **App-V Sequencer**—The App-V Sequencer converts application data into a format that is compatible with the App-V server and client, producing an App-V application.
- **App-V Server**—An App-V application can be placed on one or more App-V servers so that it can be streamed down to the clients on demand and cached locally.
- **App-V Client**—The App-V Client is the system component that enables the end user to interact with the App-V applications that are available on the App-V server.

Benefits of Using the Microsoft App-V Assistant

Instead of using the App-V Sequencer to create App-V applications, you can use the InstallShield Microsoft App-V Assistant, as shown in the following diagram:

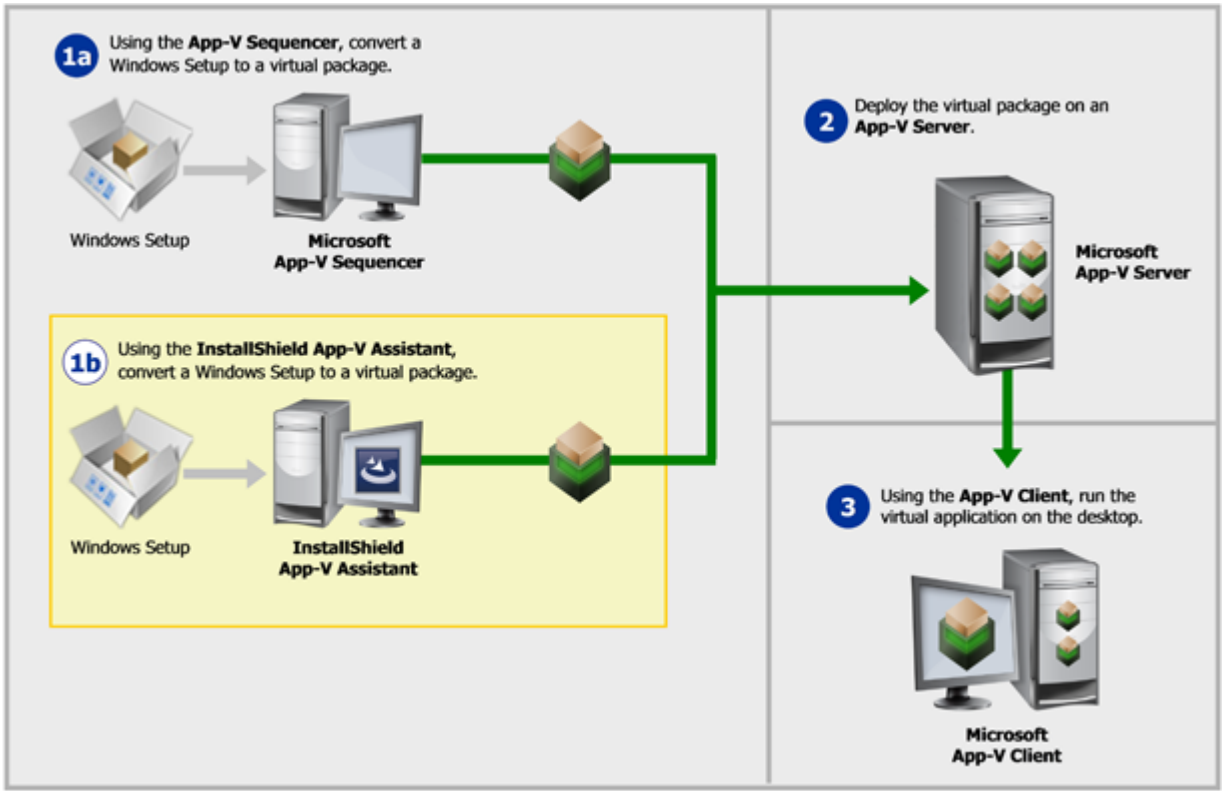


Figure 11-3: Using the App-V Assistant to Create an App-V Package

Using the App-V Assistant instead of the Microsoft App-V Sequencer to create an App-V application offers the following benefits:

Table 11-2 • Benefits of Using the App-V Assistant to Create an App-V Application

App-V Assistant Benefit	Problem	Solution
Product Installation on a Clean Machine Is Not Required	The Microsoft App-V Sequencer obtains the information it needs to create an App-V application by installing a package on a clean machine and then comparing the file system snapshot that it took prior to installation with one it takes after installation. To perform this task properly, there are two requirements: • **Product must be installed on a clean machine**—To ensure that all proper changes made by the installation are captured, sequencing needs to be performed on a clean machine (a computer with only the operating system, necessary service packs, and the App-V Sequencer installed on it). A new clean machine would need to be re-created for each application that is sequenced. • **Installation directory must be known before sequencing can begin**—In order to sequence the application effectively, you must have detailed knowledge of the how the installation is supposed to work. Prior to beginning the sequencing process, you are required to specify the installation directory for the application being sequenced. This information is often not readily available, and may require you to open the installation in an editing tool, such as InstallShield, in order to find it, or run the installation one time prior to sequencing.	Instead of installing the package, the App-V Assistant obtains the information it needs to create an App-V application directly from the installation. You are not required to have any knowledge of settings within the installation, such as the installation directory. Because there is no need to install the application to obtain this information, no permanent changes are made to the local machine and a clean machine is not required.

Table 11-2 • Benefits of Using the App-V Assistant to Create an App-V Application

App-V Assistant Benefit	Problem	Solution
Can Test the App-V Application Immediately After Conversion	To run an App-V application, the App-V Client must be installed on the machine. Because sequencing must be performed on a clean machine, which does not have the App-V Client installed, you cannot immediately test a newly created App-V application on the same machine where you sequenced it.	The App-V Assistant includes a launch utility that allows you to launch and test the App-V application locally immediately after conversion before distributing it to the App-V Server. ***Note*** • *This feature requires that the App-V Client is installed on the local machine.*
Can Include Diagnostic Tools	When running a virtual application in its virtual environment, you may at some point want to examine its contents to evaluate or debug it. However, the standard diagnostic tools that you use to examine installed applications (such as the Registry Editor and the Windows Command Prompt) are not normally available within the virtual environment. When a virtual application is running within its virtual environment, applications outside of that virtual environment cannot see into it.	When you use the InstallShield App-V Assistant to create an App-V application, you can choose to include diagnostic tools with the App-V application that enable you to use `Cmd.exe` and `Regedit.exe` on the local machine, with access to the virtual environment.
Can Perform Bulk Conversion	When using the App-V Sequencer to create App-V packages, you can only sequence one application at a time. For each application that is sequenced, you need to install the application on a clean machine.	You can use the Automated Application Converter to perform bulk conversions of multiple applications in a directory hierarchy. The Automated Application Converter has both a user interface and a command line interface. ***Edition*** • *The Automated Application Converter is included with the AdminStudio Virtualization Pack.*
Provides Easy-to-Use Interface for Dynamic Suite Composition	If you use the App-V Sequencer to create an App-V package, you have to open a separate Dynamic Suite Composition tool to edit the OSD file to create the inter-package dependencies.	The InstallShield Microsoft App-V Assistant provides a convenient, easy-to-use interface for dynamic suite composition. No separate tool is required.

Components of an App-V Package

The following table describes the components of an App-V package:

Table 11-3 • Components of an App-V Package

File	Definition
.sft	The .sft file contains all of the files, registry information, and other configuration details of the package.
Manifest file	This file is an XML file that lists all of the .osd files in an App-V application.
.osd	The .osd files are XML-based files that describe the package's individual targets (or applications) that can be run.
.ico	The .ico files are icons files that are used for published shortcuts and file type associations.
.sprj	This file is the Microsoft App-V Sequencer project file. It contains references to the .sft and .osd files, and to a large number of settings related to the sequencing process.

When you generate an App-V application, its files are saved in the following directory path:

App-VPackage*ProductName*_v*N*

The version number (*N*) of the App-V package is appended to the end of that folder path. Each time that you build an upgrade, InstallShield creates a new subfolder and increments the version number in the name of the subfolder.

The default location of the App-VPackage folder depends on whether you are building an App-V package from a Windows Installer package (.msi) or you are building an App-V package from an InstallShield project (.ism):

- If you are building an App-V package from a .msi file, InstallShield creates the App-VPackage folder in the same folder as the .msi file.
- If you are building an App-V package from an .ism file, InstallShield creates the App-VPackage folder in the following location:

  *InstallShield Project Folder**project name**product configuration**release name*\DiskImages\Disk1

A typical App-V application consists of the following files:

- ProductName.sft
- ProductName.osd
- ProductName_manifest.xml
- ProductName.sprj

InstallShield also creates a metadata.ami file in the same folder with the other files. This is an XML file that stores information about your product; AdminStudio uses the information in this file.

About the Microsoft App-V Assistant

Information about the Microsoft App-V Assistant is organized into the following sections:

- Process for Authoring an App-V Application Using the Microsoft App-V Assistant
- Supported InstallShield Project Types
- How Transforms are Included in an App-V Application
- How Windows Services Are Integrated into an App-V Application
- Quick Start for Microsoft App-V Sequencer Users

Process for Authoring an App-V Application Using the Microsoft App-V Assistant

You can use the Microsoft App-V Assistant to convert a Windows Installer package into an App-V application. During this process, you:

- **Package Information and Deployment Options**—Specify the package name, root folder name, enter a comment, specify any operating system requirements, and identify the deployment server.
- **Files, Folders, Shortcuts, Registry Settings**—Specify the files, folders, application shortcuts, and registry settings that will be included in the App-V application.
- **Isolation Options**—Set the isolation options for selected files, folders, and registry keys.
- **Build**—Specify build options and build an App-V application.

The following diagram illustrates the App-V application creation process:

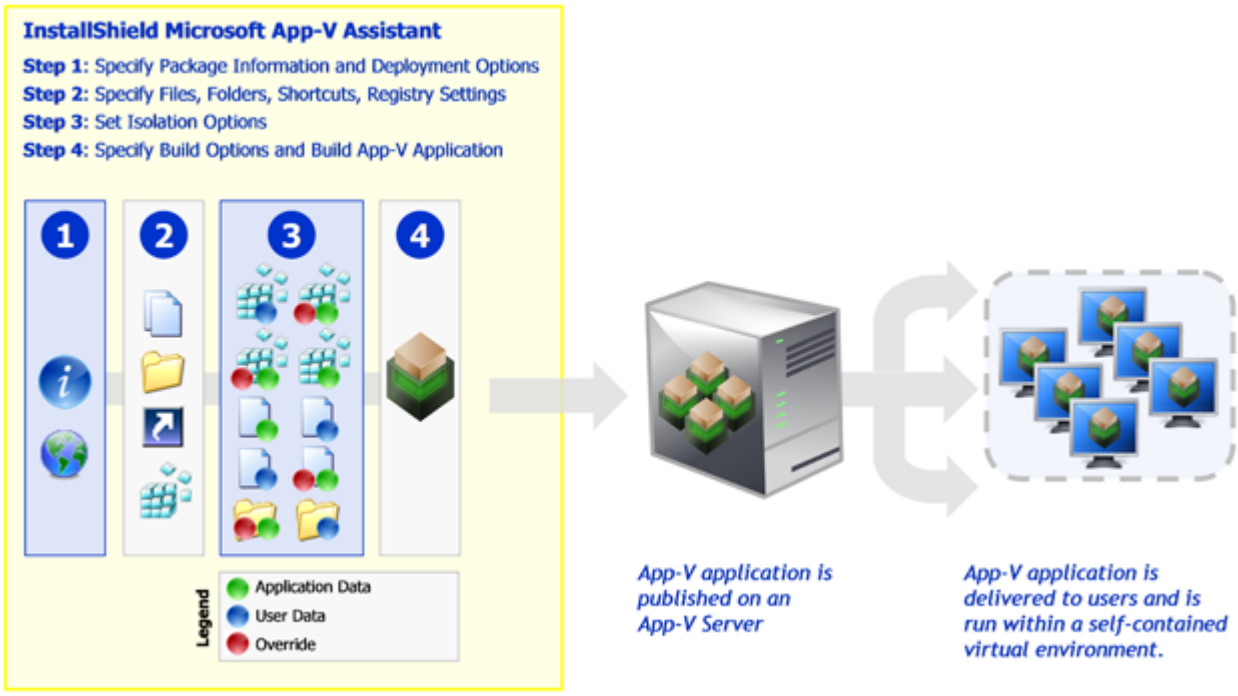


Figure 11-4: Creating an App-V Application

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

The process for authoring an App-V application using the Microsoft App-V Assistant is as follows:

Table 11-4 • Steps to Convert a Windows Installer Package to an App-V application

Step	Go To:	Actions
Getting Started	**InstallShield Start Page**	Create or open one of the following project types: • Basic MSI • MSI Database (Direct Edit Mode) • Transform (Direct MST Mode)
	InstallShield Start Page	Click on the **Microsoft App-V** tab to open the **Microsoft App-V Assistant Home** page.
	Microsoft App-V Assistant Home Page	Click **Package Information** in the navigation bar to open the **Package Information** page.

Table 11-4 • Steps to Convert a Windows Installer Package to an App-V application

Step	Go To:	Actions
Specifying Package Information and Deployment Options	**Package Information Page**	Specify the package name, root folder name, enter a comment, specify any operating system requirements, and identify the deployment server.
Managing Files in an App-V Application	**Files Page**	View existing files and folders, add and delete files.
Setting Isolation Options for Folders and Files	**Files Page**	Set isolation options for selected files and folders. Isolation options specify how the virtual environment will provide access to files and folders requested by the App-V application.
Modifying Shortcuts to the App-V Application's Executable Files	**Applications Page**	Create, delete, include, exclude, or rename App-V application executables, which are derived from the shortcuts in its Windows Installer package.
Modifying App-V Application Registry Settings	**Registry Page**	Add, delete, or modify the registry settings in your App-V application.
Setting App-V Application Registry Isolation Options	**Registry Page**	Set isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to registry keys requested by the App-V application.
Performing Dynamic Suite Composition	**Dynamic Suite Composition Page**	Use to virtualize applications separately from the plug-ins and middleware applications that they rely on, while still enabling them to communicate with those plug-ins and middleware applications within the virtual environment.
Modifying Build Options	**Build Options Page**	[Basic MSI Project mode] Select the releases that you want to build. [Direct Edit or Direct MST mode] To enable the Build function for an App-V application, select the **Build App-V application** option.

Table 11-4 • Steps to Convert a Windows Installer Package to an App-V application

Step	Go To:	Actions
Building an App-V Application	**Build on the Toolbar** OR **Build Virtual Package Button**	Click **Build** to build the active Release and create an App-V application. When you are in Direct Edit mode, click the **Build Virtual Package** button to save the Windows Installer package and create an App-V application.

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

Supported InstallShield Project Types

The **Microsoft App-V** tab is available when one of the following InstallShield project types is open:

- Basic MSI Project
- MSI Database (Direct Edit Mode)
- Transform (Direct MST Mode)

How Transforms are Included in an App-V Application

The Microsoft App-V Assistant supports the inclusion of transform files with Windows Installer packages in an App-V application.

- **How transforms are applied when building an App-V application**—When building an App-V application, transforms that you have specified are automatically applied to the base Windows Installer (`.msi`) package to create a temporary package, and then the App-V application is generated from that temporary package.
- **Creating a new transform**—You can create a new transform in InstallShield, and then build an App-V application from that transform file. When you create a new transform file in InstallShield, you specify the root `.msi` file in the **Open Transform** wizard. The steps you take to generate an App-V application after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.
- **Converting a Windows Installer package with existing transforms**—If you have a Windows Installer package and one or more existing transform files, and you want to include these transforms in the App-V application, you need to open one of the *transforms* in InstallShield (rather than the `.msi` file). The **Open Transform** wizard will open, and you will be prompted to specify the root `.msi` file and which of the existing `.mst` files you want to include. The steps you take to generate an App-V application after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.

Caution • *All of the transforms that you add to an App-V application must be located in the same folder as the Windows Installer .msi package so that they can be accessed when the App-V application is built.*

How Windows Services Are Integrated into an App-V Application

When you use the App-V Assistant to convert a Windows Installer package to an App-V application, references to Windows services that are encountered are integrated into the App-V application. In a Windows Installer package, a Windows service may be indicated by either an entry in its **ServiceInstall** table or by a Registry entry for Windows services.

- **ServiceInstall table**—If a Windows Installer package's use of a Windows service is indicated by an entry in the **ServiceInstall** table, the App-V Assistant will convert that entry to a standard Registry entry for Windows services.
- **Registry entry**—If a Windows Installer package's use of a Windows service is indicated by a Registry entry for Windows services (perhaps as the result of being repackaged), the App-V Assistant does not need to make any changes to support the application's use of the Windows service within the virtual environment.

Start Up and Shut Down Sequences

If an App-V application has an associated Windows service, App-V will start up the Windows service first, in the virtual environment, and then start up the App-V application. You will see the Windows service start up in the Task Manager as a separate process, but App-V will be running the service within the virtual environment.

Upon shut down, App-V will first shut down the App-V application and then shut down the Windows service.

Quick Start for Microsoft App-V Sequencer Users

The Microsoft App-V Assistant and the Microsoft App-V Sequencer use different directory structures for displaying files, folders, and registry entries.

- **Sequencer**—Items are shown in relation to the Primary Application Directory (also known as the mount point). See Microsoft App-V Sequencer.
- **App-V Assistant**—Items are shown in relation to the Windows Installer installation directory. Also, since the location of some directories is defined during installation, installation variables are used as placeholders. See App-V Assistant.

Microsoft App-V Sequencer

When viewing the files and folders of an App-V Sequencer project, they are displayed in relation to the Primary Application Directory. In the following graphic, the `myapp.exe` executable file is displayed in a subdirectory of `SoftcoMyApp.006`, the Primary Application Directory:

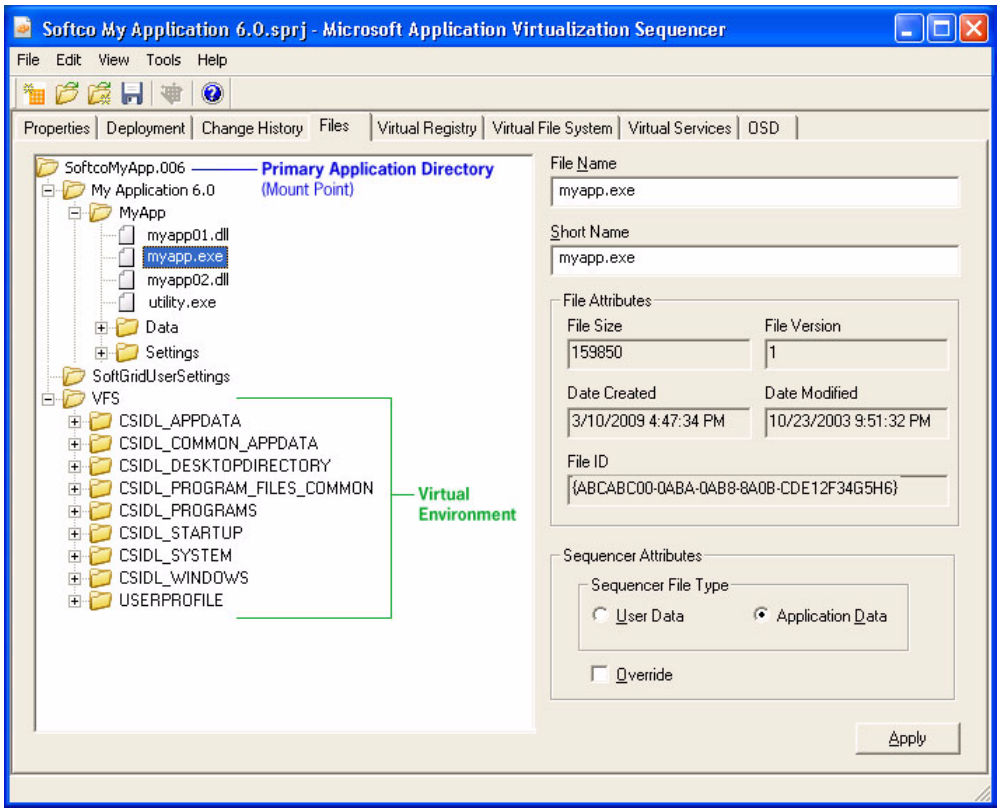


Figure 11-5: Microsoft App-V Sequencer Interface

When this App-V application is published, the `myapp.exe` executable file will be located in the following directory:

```
Q:\SoftcoMyApp.006\My Application 6.0\MyApp\myapp.exe
```

Note • *This assumes that virtual packages are mounted on the `Q:` drive.*

App-V Assistant

When editing a Windows Installer package or InstallShield project in the App-V Assistant, the directory structure of files, folders, and registry entries reflects the relative location of these items if they were installed. For example, on the **Files** page of the App-V Assistant, installation variables, such as `[ProgramFilesFolder]` or `[CommonFilesFolder]`, identify the installed location of their child elements.

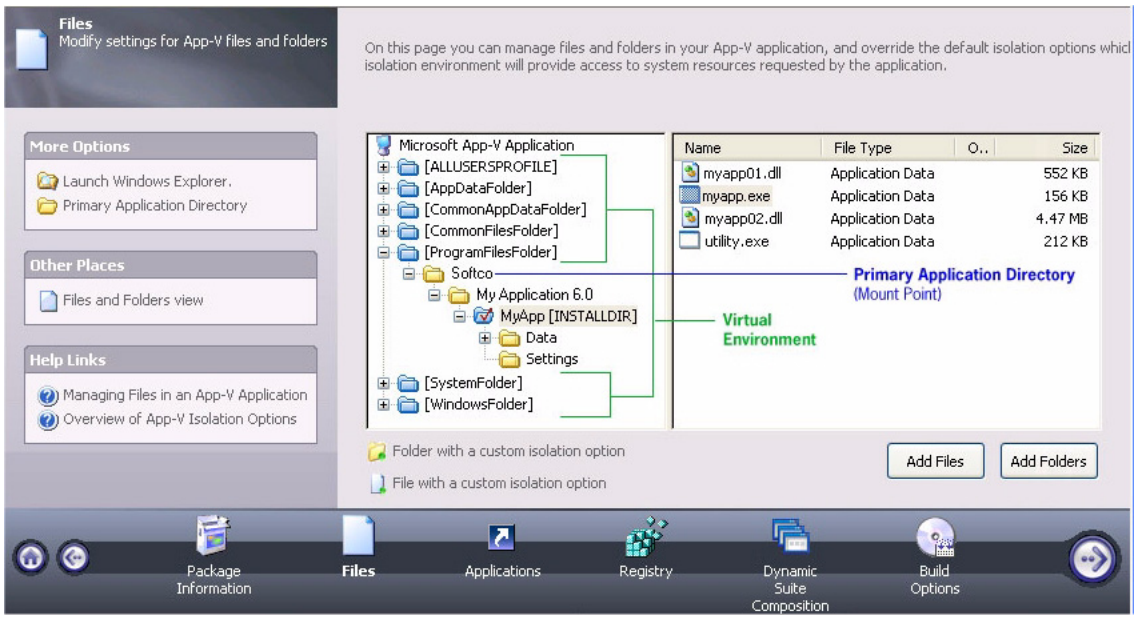


Figure 11-6: Microsoft App-V Assistant Interface

In this example, the file `myapp.exe` executable file is shown in the following directory:

```
[ProgramFilesFolder]\Softco\My Application 6.0\MyApp\myapp.exe
```

If this Windows Installer package were installed, the `myapp.exe` executable file would be installed in a subdirectory of the directory that is specified during installation. For example, if `C:\Program Files` is specified as the installation directory, `myapp.exe` would be installed in the following location:

```
C:\Program Files\Softco\My Application 6.0\MyApp\myapp.exe
```

However, when this Windows Installer package is converted to an App-V application, the build engine will map the files in the Primary Application Directory to the mount point. So,

```
C:\Program Files\Softco\My Application 6.0\MyApp\myapp.exe
```

becomes:

```
Q:\SoftcoMyApp.006\My Application 6.0\MyApp\myapp.exe
```

Also, all other references to this file—such as those in the Registry, INI files, and in Window Services—are rewired to point to the actual file location in the App-V package.

Note • *This assumes that virtual packages are mounted on the `Q:` drive.*

Comparison of Sequencer to App-V Assistant

To enable you to more clearly compare the Microsoft App-V Sequencer to the App-V Assistant, the following figure shows the location of selected folders and files in both interfaces.

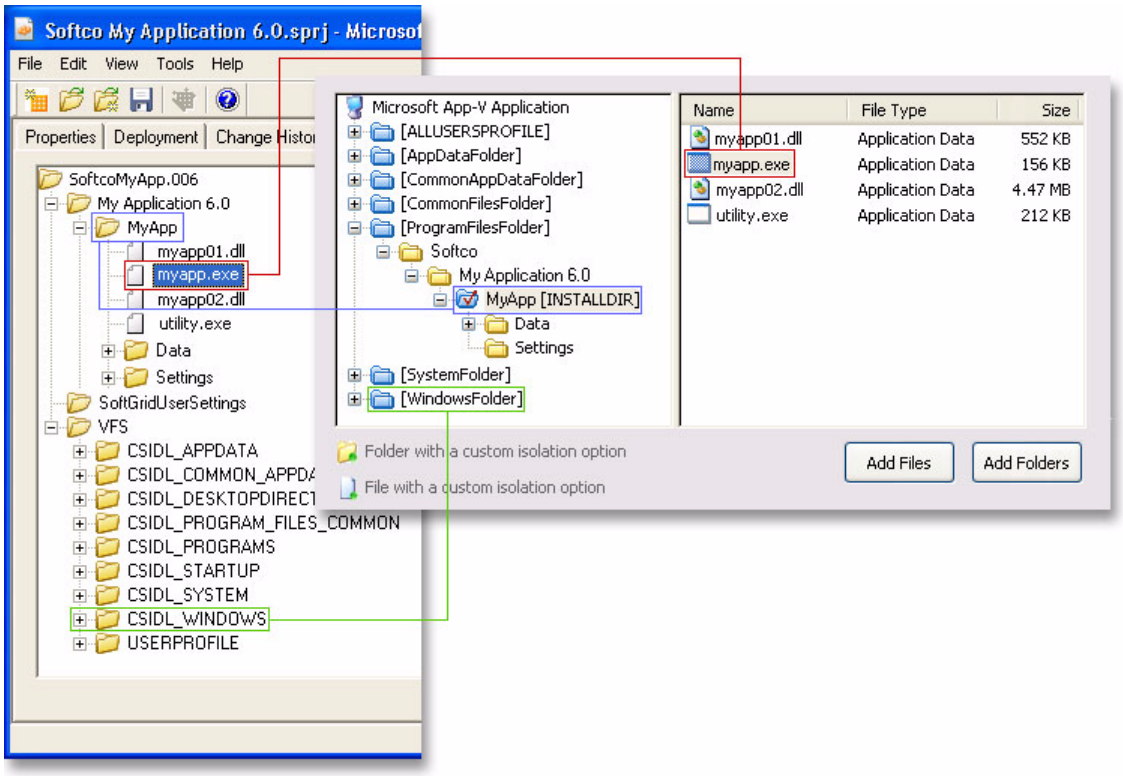


Figure 11-7: Comparison of Sequencer Interface to Microsoft App-V Assistant Interface

Using the Microsoft App-V Assistant to Create an App-V Application

The steps you need to take to create an App-V application are the following:

Table 11-5 • Steps to Take to Create an App-V Application Using the App-V Assistant

Step #	Description
Step 1	Specifying Package Information and Deployment Options
Step 2	Managing Files in an App-V Application
Step 3	Setting Isolation Options for Folders and Files
Step 4	Modifying Shortcuts to the App-V Application's Executable Files
Step 5	Modifying App-V Application Registry Settings
Step 6	Setting App-V Application Registry Isolation Options

Table 11-5 • Steps to Take to Create an App-V Application Using the App-V Assistant

Step #	Description
Step 7	Performing Dynamic Suite Composition
Step 8	Modifying Build Options
Step 9	Building an App-V Application
Step 10	Testing an App-V Application Using the App-V Application Launcher

Specifying Package Information and Deployment Options

When you are creating an App-V application, the first step is to specify the package name, root folder name, and enter a comment on the Package Information page. On this page, you can also specify any operating system requirements, identify the deployment server, and specify whether to include diagnostic tools with the virtual package. This page also lets you specify upgrade information for your App-V package if appropriate.

Specifying Package Information

The first step in creating an App-V application is to enter the package name, root folder name, and a comment. To specify package information for an App-V application, perform the following steps:

Task: ***To specify package information:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. In the **Package Name** field, enter a name for the virtual package.
3. In the **Root Folder Name** field, enter a name to identify the directory that will contain the App-V application's `.sft` file. When an App-V application is configured on a client machine, it is mounted on the computer's `Q:\` drive in the folder you specify in this field.
   - **Default value**—By default, the **Root Folder Name** is based on the `[ProductName]` and `[ProductVersion]` properties of the App-V application's associated Windows Installer file using the 8.3 file naming convention. For example:

If ProductName is ...	And ProductVersion is ...	Then default Root Folder Name will be ...
`MyApplication`	`1.12.3.1`	`MyApplic.112`
`MyApp`	`1`	`MyApp.1`
`MyBlueApp`	`1.2.3.4`	`MyBlueAp.123`

- **Conflicts caused by duplicate values**—If you happen to have another App-V application that has identical values for the `[ProductName]` and `[ProductVersion]` properties, you can avoid conflicts by entering a unique value in the **Root Folder Name** field.

4. In the **Comments** field, enter a description of the purpose of this App-V application and any special information that you wish to document.

Specifying Operating System Requirements

To specify operating system requirements for an App-V application, perform the following steps:

Task: ***To specify operating system requirements:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. Set the **Does your App-V application have any specific operating system requirements?** option to one of the following:
   - **No**—Select this option if this application will run on all of the listed operating systems. When this option is selected, the operating system check boxes are locked and cannot be changed.
   - **Yes**—Select this option if the application does not support one of the listed operating systems. When you select this option, the check boxes are unlocked and you can clear the selection of the unsupported operating systems.
3. If you selected **Yes**, select the operating systems that this application supports, and clear those that this application does not support.

Specifying Upgrade Package Information

InstallShield enables you to specify whether you want to create an upgrade for your App-V package. If you specify that you do want to create an upgrade, you can specify additional information about the upgrade, such as whether to append the version number to the App-V package file name.

Task: ***To specify upgrade information:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. In the **More Options** area, click **Upgrade Settings**. The **App-V Package Upgrade Settings** dialog box opens.
3. Do one of the following:
   - To create an upgrade for your App-V package, select the **Enable Upgrade** check box. Then specify which package should be upgraded. If you want InstallShield to include the version number in the package name, select the **Append version number to package name** check box.
   - To avoid creating an upgrade package, clear the **Enable Upgrade** check box.

Specifying the Deployment Server

To specify the deployment server for an App-V application, perform the following steps:

Task: ***To specify the deployment server:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. Under **Where will the App-V application be deployed?**, set the following **Server URL** options:

Option	Description
Protocol	Enables you to select the protocol that will stream the sequenced application package from a virtual application server to an Application Virtualization Desktop Client. The following protocols are available: • **RTSP**—The default, it specifies that the Real-Time Streaming Protocol controls the exchange of virtualization-enabled applications. • **RTSPS**—Specifies that the Real-Time Streaming Protocol with Transport Layer Security controls the exchange of a sequenced application package. • **FILE**—Specifies that the sequenced application will be streamed from a file share. • **HTTP**—Specifies that Hypertext Transport Protocol controls the exchange of a package. • **HTTPS**—Specifies that Secure Hypertext Transport Protocol controls the exchange of a package.
Host	Specify the virtual application server or the load balancer in front of a group of virtual application servers that will stream the software package to an Application Virtualization Desktop Client. You must complete this item to create a sequenced application package, but you can change from the default `%SFT_SOFTGRIDSERVER%` environment variable to the actual hostname or IP address of a virtual application server. ***Note*** • *If you choose not to specify a static hostname or IP address, on each Application Virtualization Desktop Client you must set up an environment variable called `SFT_SOFTGRIDSERVER`.* • *Its value must be the hostname or IP address of the virtual application server or load balancer that is this client's source of applications.* • *You should make this environment variable a system variable rather than a user variable.* • *Any Application Virtualization Desktop Client session that is running on this computer during your assignment of this variable must be closed and then opened so that the resumed session will be aware of its new application source.*

Option	Description
Port	Specify the port on which the virtual application server or the load balancer will listen for an Application Virtualization Desktop Client's request for the package. • This information is required to create a package, but you can change it. • The default port is 554.
Path	Specify the relative path on the virtual application server where the App-V package is stored. This is also the path from which the App-V package is streamed. Note that it is not necessary to include the .sft file name in this path. If the App-V package is stored in a subdirectory of CONTENT, the path must be specified in this setting; otherwise, you can leave this setting blank. If you leave this setting blank, the App-V package is stored in a folder that has the same name that is used for the package name and the .sft file name.

Including Diagnostic Tools With an App-V Application

On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **Package Information** page, you can choose to include the Registry Editor and Windows Command Prompt diagnostic tools with your App-V application.

If you include diagnostic tools with your App-V application, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running an App-V application and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Task: ***To include diagnostic tools with an App-V application:***

1. In the **Microsoft App-V Assistant**, open the **Package Information** page.
2. In the **More Options** list, click **Diagnostic Tools**. The **Diagnostic Tools** dialog box opens.
3. If you want to include the Registry Editor with your App-V application so that you can use `Regedit.exe` on the local machine and have access to the virtual environment, select the **Registry Diagnostics** option.

   If the **Registry Diagnostics** option is selected, a file named `Virtual Registry.osd` will be created in the App-V Package folder, which can be used to display the registry within the virtual environment.
4. If you want to include the Windows Command Prompt application with your App-V application so that you can use `Cmd.exe` on the local machine and have access to the virtual environment, select the **File System Diagnostics** option.

   If the **File System Diagnostics** option is selected, a file named `Virtual File System.osd` will be created in the App-V Package folder, which can be used to display the files and folders within the virtual environment.
5. Click **OK**.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the App-V application.

When the user runs this App-V application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

Managing Files in an App-V Application

The next step in creating an App-V application is to view existing files and folders, add and delete files and folders, and set isolation options for files and folders.

The following tasks are performed on the **Files** page.

- Adding, Deleting, and Moving Files and Folders in an App-V Application
- Controlling the Display of Predefined Folders
- Specifying the Primary Application Directory

Adding, Deleting, and Moving Files and Folders in an App-V Application

The directories in the destination tree on the **Files** page of the Microsoft App-V Assistant represent how your application will look when it is installed on to your customer's machine.

On the **Files** page, you can view all of the files and folders that are currently in your App-V application, add new files and folders to include in the App-V application, and delete files and folders from the App-V application.

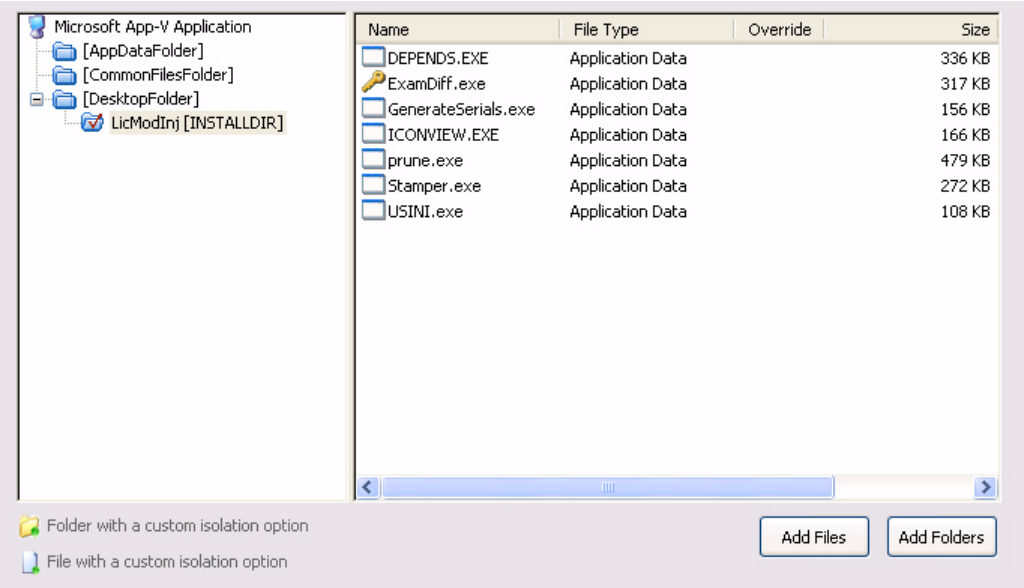
- Adding Files to an App-V application
- Adding an Existing Folder (and its Contents) to an App-V Application
- Creating a New Folder
- Moving Files and Folders
- Deleting Files and Folders

Adding Files to an App-V application

To add files to an App-V application, perform the following steps:

Task: ***To add a files to an App-V application:***

1. In the **Microsoft App-V Assistant**, open the **Files** page. The files and folders are listed in the **Microsoft App-V application** tree, organized by installation directory.



   Folders are listed in the column on the left, and all of the files in the selected folder are listed on the right. Blue folders are the supported MSI standard folders. The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

2. Browse through the folder tree to find the folder that you would like to add files to.
3. Select the folder and click the **Add Files** button. The **Open** dialog box opens.
4. Select the file or files that you want to add and click **Open**. The files you selected are now listed.

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Adding a File by Dragging and Dropping Files From Your System

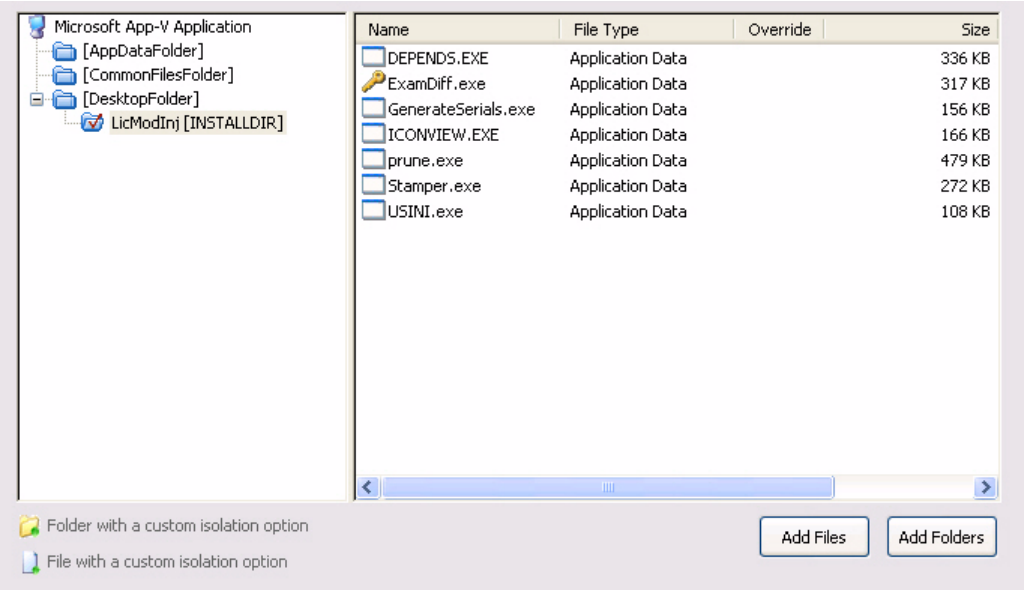
You can also add files or folders to your App-V application on the **Files** page by dragging them from a directory on your computer to the desired location in the tree.

Adding an Existing Folder (and its Contents) to an App-V Application

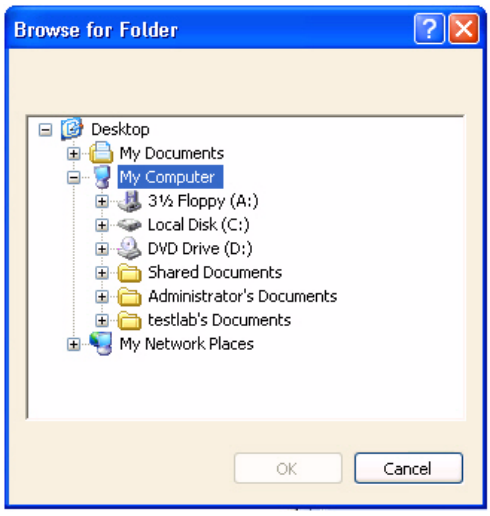
To add an existing folder and all of the files and subfolders within it to an App-V application, perform the following steps:

Task: ***To add an existing folder to an App-V application:***

1. In the **Microsoft App-V Assistant**, open the **Files** page. The files and folders are listed in the **App-V application** tree, organized by installation directory.

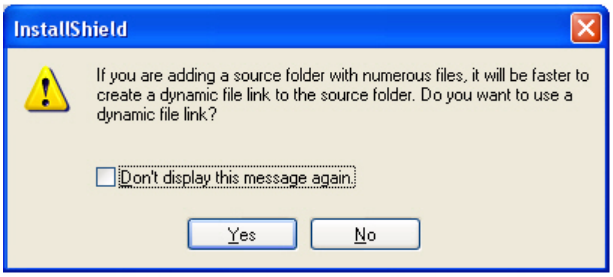


2. Browse through the folder tree to find the folder that you would like to add a folder into.
3. Select the folder and click the **Add Folders** button. The **Browse for Folder** dialog box opens, listing all of the directories available to your computer.



4. Select a folder and click **OK**.

   If you are editing an InstallShield project (not a Windows Installer package), you are prompted to choose whether you want to create a dynamic file link to the source folder.



5. Indicate whether you want to create a dynamic file link by selecting one of the following:

   - **No**—For more flexibility with App-V options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
   - **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

   The folder that you selected is now listed, along with of the files and folders within it.

Creating a New Folder

You can create a new, empty folder by selecting an existing folder in the tree and selecting **New Folder** from the context menu.

Task: ***To create a new folder:***

1. Right-click on a folder in the **Microsoft App-V application** tree and select **New Folder**. A new folder is created as a subfolder of the selected folder:

2. Enter a name for the new folder.

Moving Files and Folders

To change the folder's location in the App-V application folder tree structure, perform the following steps:

Task: ***To move a file or folder:***

1. Select the file or folder that you want to move.
2. With the mouse button down, drag the file or folder to the new location.
3. Release the mouse button.

Deleting Files and Folders

To delete a file or a folder (and all of its contents) from an App-V application, perform the following steps:

Task: ***To delete a file or folder:***

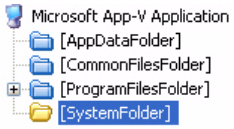
1. Select the file or folder in the **Microsoft App-V Application** tree that you want to delete.
2. Select **Delete** from the context menu. You are prompted to confirm the deletion.
3. Click **Yes**. The selected file or folder is deleted.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains from the entire Project, not just from the App-V application.*

Note • *You cannot delete predefined folders. You can only turn off the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Controlling the Display of Predefined Folders

On the **Files** page, the **Microsoft App-V application** tree initially displays the more commonly used predefined folders, such as `[CommonFilesFolder]` and `[ProgramFilesFolder]`.

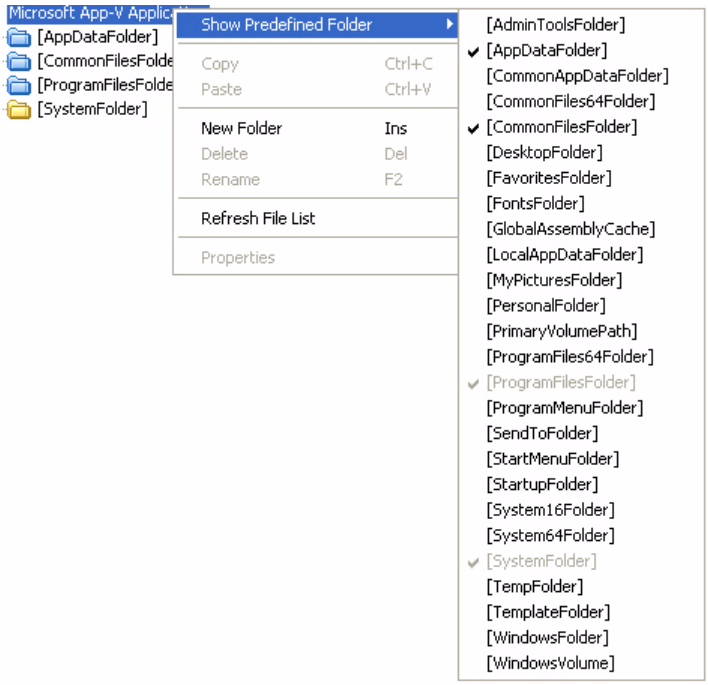


These predefined folders are dynamic, meaning that they do not use hard-coded paths. The value for each destination folder is obtained from the operating system of the target machine.

You can control which predefined folders are listed in this tree.

Task: ***To change which predefined folders are listed:***

1. In the **Microsoft App-V application** tree, select the **App-V application** node (or any of the files or folders that are listed, point to **Show Predefined Folder**. A list of predefined folders opens.



   Those folders that are already displayed are preceded by a check mark, and those that are not displayed do not have a check mark.

2. To add a folder to the tree listing, select a folder that is not currently listed in the tree.

   Note • *These predefined folders are always added to the root of the* ***Microsoft App-V application*** *tree, no matter what file or folder you had selected when you selected it from the Predefined Folders list.*

3. To remove a folder from the tree listing, select that folder name in this list (which is preceded by a check mark.

   Note • *You cannot turn off the display of the* `[ProgramFilesFolder]`.

Specifying the Primary Application Directory

When App-V applications are run on a machine that has Microsoft Virtualization Client installed, they are run from the Client's application cache drive.

For optimum performance, it is desirable to have the bulk of the application's files mounted to this drive at run-time. In order to achieve this, it is useful to determine an application's primary application directory so that folder can be mounted to the App-V Client's application literal cache drive when the App-V application is loaded.

When an App-V application is built using InstallShield or any AdminStudio tool, the following series of steps are used to determine an App-V application's primary application directory:

Table 11-6 • Steps to Automatically Determine Primary Application Directory

#	Step	Description
1	**Explicitly set primary application directory**	If a directory is specified on the InstallShield App-V Assistant's **Primary Application Directory** dialog box (as described in Explicitly Setting the Primary Application Directory), that directory will be used.
2	**Value of INSTALLDIR variable**	If the Windows Installer package has a value for `INSTALLDIR` (a system variable that specifies the root destination directory for an installation), that value will be used as the primary application directory. ***Note*** • *This step is not used when converting Windows Installer packages that have been repackaged using AdminStudio.* ***Note*** • *All Windows Installer packages created by InstallShield or AdminStudio will have a value for the INSTALLDIR variable.*
3	**Location of shortcut in a subdirectory of the ProgramFilesFolder**	If one of the `.exe` targets for a shortcut is in a subdirectory of `ProgramFilesFolder`, then the folder directly under `ProgramFilesFolder` will be used as the primary application directory. Typically, this would be: `C:\Program Files\YourApplication`
4	**Location of shortcut in a directory other than ProgramFilesFolder**	If no `.exe` targets are located in a subdirectory of `ProgramFilesFolder`, then the target directory of a shortcut that contains an `.exe` will be used.
5	**ProgramFilesFolder**	If none of the above can be found, then the primary application directory will be set to `ProgramFilesFolder`. Typically, this would be: C:\Program Files

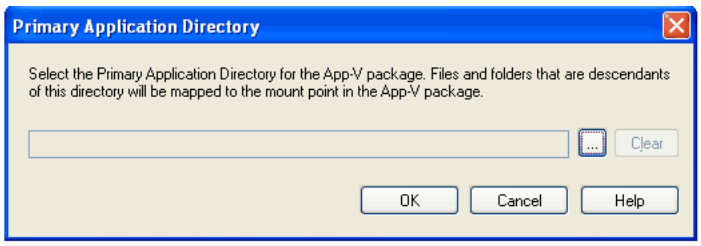
You can specify the Primary Application Directory for an App-V application by clicking **Primary Application Directory** in the **More Options** list on the **Files** page.

Explicitly Setting the Primary Application Directory

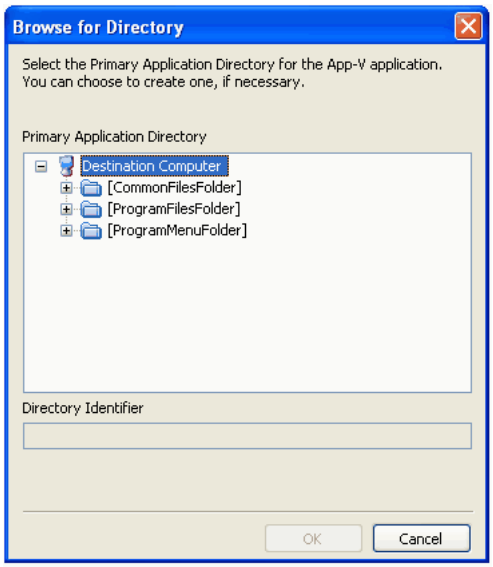
To specify the primary application directory for an App-V application, perform the following steps:

Task: ***To specify the primary application directory:***

1. In the **Microsoft App-V Assistant**, open the **Files** page.
2. In the **More Options** list, click **Primary Application Directory**. The **Primary Application Directory** dialog box opens, displaying the current primary application directory setting (if one has already been specified).



3. Click the browse (...) button. The **Browse for Directory** dialog box opens, listing all of the currently available destination directories for this App-V application.



4. Select one of the listed directories and click **OK**.

Setting Isolation Options for Folders and Files

When using the Microsoft App-V Assistant, you can set isolation options to control application compatibility and accessibility. The isolation option that is assigned to a file or folder specifies how the isolation environment will provide access to system resources requested by the application.

You set isolation options on the **Isolation Options** dialog box, which is opened by selecting a file or folder and then selecting **Isolation Options** from the context menu.

Information about setting isolation options is presented in the following topics:

- Overview of App-V Isolation Options
- Setting Isolation Options for Folders and Files
- Inheritance of Isolation Options from Folders to Files

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

Overview of App-V Isolation Options

When using the Microsoft App-V Assistant, you can set options that uses isolation environments use to control application compatibility and accessibility. The isolation option that is assigned to a file or folder specifies how the isolation environment will provide access to system resources requested by the application.

You set isolation options on the **Isolation Options** dialog box, which is opened by selecting **Isolation Options** on the context menu when you have a file or folder selected on the **Files** page or a registry key selected on the **Registry** page.

Isolation Options for Files and Folders

The following isolation options are available:

Table 11-7 • App-V Isolation Options for Files and Folders

Option	Description
File Type	When an App-V application performs a WRITE operation to a resource (a file or folder), the resource's File Type setting determines whether changes to that resource are saved for all users of the App-V application on that machine or only for the logged-in user. Select one of the following options: • **Application Data**—Changes to the resource are saved for all users of this App-V application on this machine. • **User Data**—Changes to the resource are saved only for the logged-in user.
Override	To have the virtualization client overwrite the resource (file or folder) when the application is upgraded and streamed from the App-V server to the client, select this check box. If Override check box is not selected, then the App-V client determines whether to overwrite the file or folder during an upgrade. In general, if the resource is marked as Application Data, the resource is overwritten during an upgrade; otherwise, it is not overwritten.

Isolation Options for Registry Keys

The following isolation options are available:

Table 11-8 • App-V Isolation Options for Registry Keys

Option	Description
Merge with local key	Select this option to specify that the App-V application will see a merged view of the registry entries for the selected key from both the local registry and from the App-V application's registry.
Override local key	Select this option to specify that the App-V application will see only the registry entries for the selected key that are part of that App-V application.

Setting Isolation Options for Folders and Files

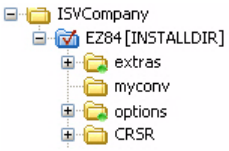
To set a file or folder's isolation options, perform the following steps:

Task: ***To set an isolation option on a folder or file:***

1. Open the **Files** page.
2. Browse through the folder tree to find the file or folder that you would like to modify.
3. Select the file or folder and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.
4. Specify the following options:

Option	Description
File Type	When an App-V application performs a WRITE operation to a resource (a file or folder), the resource's File Type setting determines whether changes to that resource are saved for all users of the App-V application on that machine or only for the logged-in user. Select one of the following options: • **Application Data**—Changes to the resource are saved for all users of this App-V application on this machine. • **User Data**—Changes to the resource are saved only for the logged-in user.
Override	To have the virtualization client overwrite the resource (file or folder) when the application is upgraded and streamed from the App-V server to the client, select this check box. If Override check box is not selected, then the App-V client determines whether to overwrite the file or folder during an upgrade. In general, if the resource is marked as Application Data, the resource is overwritten during an upgrade; otherwise, it is not overwritten.

5. Click **OK**. Files and folders that have an isolation setting other than default are marked with a special icon:



Inheritance of Isolation Options from Folders to Files

Isolation options for files and folders are always inherited. The App-V isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for `C:\Windows` and one for `C:\Windows\System32`. When the application requests `C:\Windows\System32\Notepad.exe`, then the `C:\Windows\System32` isolation rule will be applied because `C:\Windows\System32` is a more specific reference to `C:\Windows\System32\Notepad.exe` than is `C:\Windows`.

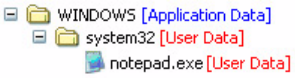


Figure 11-8: Example of Inheritance of Isolation Options from Folders to Files

Modifying Shortcuts to the App-V Application's Executable Files

You define application shortcuts to enable users to launch an App-V application from within the virtual environment.

By default, the **Microsoft App-V Assistant** creates App-V applications for all of the executable shortcuts that exist in your project (or Windows Installer package). These shortcuts are listed in a checklist on the **Applications** page.

When you select each shortcut, details about it are displayed:

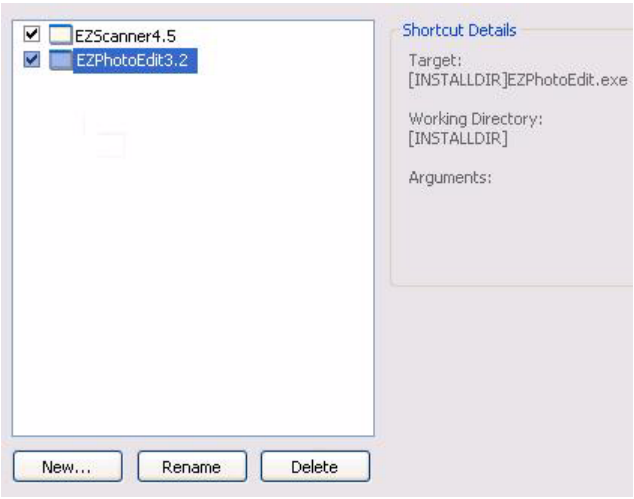


Figure 11-9: List of Shortcuts for an Application

Caution • *You must define at least one shortcut to enable users to launch the application from the isolation environment.*

On the **Applications** page, you can create, delete, include, exclude, or rename App-V application executables, which are derived from the shortcuts in its Windows Installer package.

- App-V Applications and the Virtual Environment
- App-V Shortcut Requirements
- Creating a New App-V Application
- Including an Existing App-V Application
- Excluding or Deleting an Existing App-V Application
- Excluding vs. Deleting App-V Application Shortcuts
- Renaming an App-V Application

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

App-V Applications and the Virtual Environment

On the **Applications** page of the Microsoft App-V Assistant, you define application shortcuts to enable users to launch an App-V application from within the virtual environment. By default, the Microsoft App-V Assistant creates App-V applications for all of the executable shortcuts that exist in your project.

To deploy an App-V application—on a local drive or a network share—systems administrators just need to give users access to the App-V application.

App-V Shortcut Requirements

For each App-V application, you are required to define **at least one** shortcut. You define application shortcuts to enable users to launch an App-V application from within the virtual environment. By default, the Microsoft App-V Assistant creates App-V applications for all of the executable shortcuts that exist in your project (or Windows Installer package).

If you build an App-V application that does not contain any shortcuts, users will not be able to launch the application.

Creating a New App-V Application

On the **Applications** page of the Microsoft App-V Assistant, you specify the executables that you want to create App-V applications for.

Task: ***To create a new App-V application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project (or Windows Installer package) are listed:
   - Those that are currently included in the App-V application are selected.
   - Those that are currently excluded from the App-V application are not selected.
2. Click **New**. The **Browse for a Shortcut Target File** dialog box opens and prompts you to select a file within this App-V application.
3. Select the file that you want to create a shortcut to.
4. Click **Open**. A new shortcut is listed, and it is named the same name as the selected file.
5. To include this shortcut in the App-V application, make sure that its check box is selected.

Including an Existing App-V Application

If you want to include a previously excluded shortcut in an App-V application, perform the following steps:

Task: ***To include an existing App-V application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To include a previously excluded shortcut, select the shortcut and select the check box.

Excluding or Deleting an Existing App-V Application

By default, the Microsoft App-V Assistant creates App-V applications for all of the executable shortcuts that exist in your project (or Windows Installer package). These shortcuts are listed in a checklist on the **Applications** page.

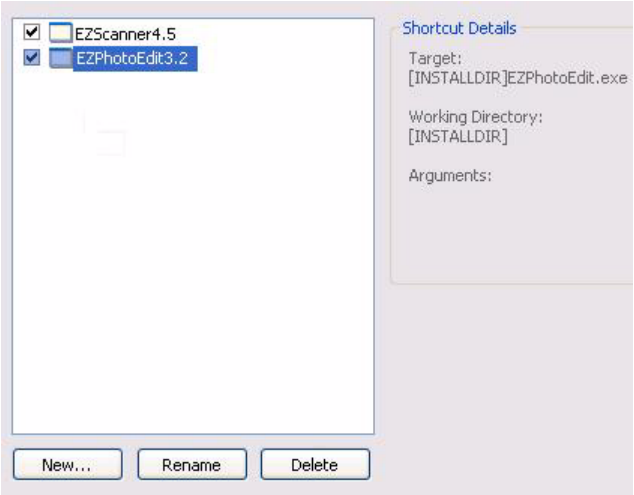


Figure 11-10: Initial List of Shortcuts for an Application

To prevent the shortcut from being created in the App-V application, you can choose to either delete or exclude it.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the App-V application, but it will remain in the InstallShield project. See Excluding an App-V Application.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the App-V application and the InstallShield project. See Deleting an App-V Application.

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

If you have any unnecessary shortcuts in your project, you can simply exclude them from the App-V application by unchecking them in the shortcuts list. If you like to permanently remove a shortcut, you can delete it from the shortcut list.

Excluding an App-V Application

If you want to exclude one of these shortcuts from being created in the App-V application, perform the following steps:

Task: ***To exclude an App-V application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To exclude a shortcut, select the shortcut and clear the check box.

Note • *When you exclude a shortcut, it will not be created in the App-V application, but it will remain in the InstallShield project.*

Deleting an App-V Application

To delete an App-V application, perform the following steps.

Task: ***To delete an App-V application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
2. Select the shortcut and click **Delete**.

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

Excluding vs. Deleting App-V Application Shortcuts

To prevent a shortcut from being created in the App-V application, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the App-V application, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding an App-V Application.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the App-V application and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting an App-V Application.

Renaming an App-V Application

To rename an App-V application, perform the following steps:

Task: ***To rename an App-V application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
2. Select the shortcut that you want to rename and click **Rename**. A box appears around the shortcut name, and the shortcut name becomes an editable field.
3. Enter a new name for the shortcut.

Modifying App-V Application Registry Settings

Using the **Microsoft App-V Assistant**, you can view existing registry keys, values, and data, and add or delete registry items in your App-V application.

You can also set isolation options for selected registry keys. Isolation options specify how the isolation environment will provide access to system resources requested by the application.

Information about modifying registry settings on the **Registry** page includes the following topics:

- About the Windows Registry
- Adding or Deleting Registry Keys and Values

About the Windows Registry

The Windows registry is a system-wide database that contains configuration information used by applications and the operating system. The registry stores all kinds of information, including the following:

- Application information such as company name, product name, and version number
- Path information that enables your application to run
- Uninstallation information that enables end users to uninstall the application easily without interfering with other applications on the system
- System-wide file associations for documents created by an application
- License information
- Default settings for application options such as window positions

Keys, Value Names, and Values

The registry consists of a set of keys arranged hierarchically under the My Computer explorer. Just under My Computer are several root keys. An installation can add keys and values to any root key of the registry. The root keys that are typically affected by installations are:

- `HKEY_LOCAL_MACHINE`
- `HKEY_USERS`
- `HKEY_CURRENT_USER`
- `HKEY_CLASSES_ROOT`

A key is a named location in the registry. A key can contain a subkey, a value name and value pair, and a default (unnamed) value. A value name and value pair is a two-part data structure under a key. The value name identifies a value for storage under a key, and the value is the actual data associated with a value name. When a value name is unspecified for a value, that value is the default value for that key. Each key can have only one default (unnamed) value.

Note that the terms key and subkey are relative. In the registry, a key that is below another key can be referred to as a subkey or as a key, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding or Deleting Registry Keys and Values

Editing the registry on the Registry page is performed much like it is performed in the InstallShield Registry view. To learn more, see Editing the Registry.

Setting App-V Application Registry Isolation Options

To set a registry key's isolation option, perform the following steps:

Task: ***To set an isolation option on a registry key:***

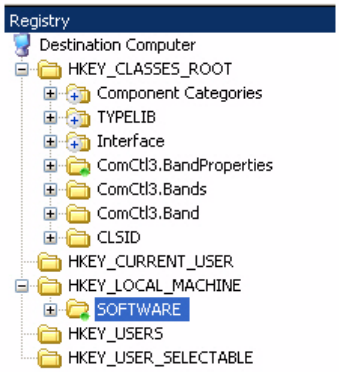
1. Open the **Registry** page.
2. Browse through the registry tree to find the key that you would like to modify.
3. Select the key and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.

Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

4. Select one of the following options:

Option	Description
Merge with local key	Select this option to specify that the App-V application will see a merged view of the registry entries for the selected key from both the local registry and from the App-V application's registry.
Override local key	Select this option to specify that the App-V application will see only the registry entries for the selected key that are part of that App-V application.

5. Click **OK**. Registry keys that have an isolation setting of **Override local key** are marked with a special icon:



Tip • *To import an existing registry (.reg) file, click the* ***Import a .reg file*** *option on the* ***More Options*** *list to open the Registry Import Wizard.*

Inheritance of Isolation Options in the Registry

Isolation options for registry keys are always inherited. The isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for the `Microsoft` registry key and one for `Microsoft\Windows` registry key. When the application requests `Microsoft\Windows\CurrentVersion`, then the `Microsoft\Windows` isolation rule will be applied because `Microsoft\Windows` is a more specific reference to `Microsoft\Windows\CurrentVersion` than is `Microsoft`.

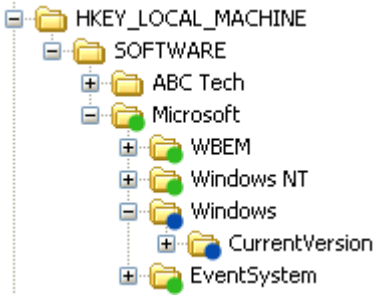


Figure 11-11: Example of Inheritance of Isolation Options from Folders to Files

Performing Dynamic Suite Composition

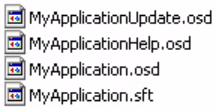
The point of application virtualization is to minimize the system dependencies that an application has on the underlying physical system. Many applications have common system dependencies on plug-ins or middleware, such as Adobe Reader or ODBC drivers.

Dynamic Suite Composition (DSC) is a Microsoft Application Virtualization feature that enables applications to be virtualized separately from the plug-ins and middleware applications that they rely on, while still enabling them to communicate with those plug-ins and middleware applications within the virtual environment. The primary App-V application and the dependency App-V applications in the dynamic suite will run and interact with one another as if they were all installed locally on a computer. You would only need to deploy common system components once on each client, making them available for use by many App-V applications, rather than to include them with each of the App-V applications that are dependent upon them. This reduces redundancy in the local App-V cache and simplifies the construction and testing of the primary App-V applications.

You specify the App-V applications that you want to include in a dynamic suite on the **Dynamic Suite Composition** page of the App-V Assistant.

Task: ***To compose a dynamic suite of App-V applications:***

1. In the App-V Assistant, open the **Dynamic Suite Composition** page.
2. To add a dependency App-V application, click the New ( ) button. The **Open** dialog box opens.
3. Open the directory where the dependency App-V application that you want to add is located. That application's `.osd` and `.sft` files are listed.



4. Select one of the following:
   - **One of the .osd files**—If this dependency App-V application is or is going to be published on a Server, select any one of the `.osd` files that are listed. If these `.osd` files were created properly, each of them should contain the information that will identify to the primary App-V application the published location of the dependency App-V application.

     Note • *It is not necessary to select more than one `.osd` file. All of them contain the same reference to the location of the dependency App-V application's `.sft` file, which is the only reference that is necessary in order for the primary App-V application to locate it.*

   - **The .sft file**—If this dependency App-V application is or is going to be published locally on the client or at an accessible network location, you may select just the `.sft` file.

   The selected App-V application is now listed in the **Dependency Applications** list and, by default, the **Mandatory** option is selected.
5. Set the status of the selected App-V application by doing one of the following:
   - **If the dependency App-V application is mandatory**—If the primary App-V application will not run unless it can locate this dependency App-V application, leave the **Mandatory** option selected. If a dependency App-V application that is configured as mandatory is not available, an error will be generated when someone attempts to run the primary App-V application.
   - **If the dependency App-V application is not mandatory**—If the primary App-V application will run even if it cannot locate this dependency App-V application, clear the **Mandatory** option.
6. Build the primary App-V application.

Deleting a Dependency Application from the List

To remove an App-V application from the Dependency Applications list, select the application and click the Delete ( ) button.

Modifying Build Options

On the Build Options page, you choose which releases of this InstallShield project you want to build an App-V application for when the project is built, and specify whether you want to include additional Windows Installer packages in the virtual package.

Also, if you are editing a Windows Installer package in Direct Edit mode (or Direct MST mode), you need to select the **Build App-V application** option on the Build Options page before you will be able to build an App-V application for that Windows Installer package.

Important • *You must create at least one Release (on the Releases view of the Installation Designer) before you will be able to select a Release on the Build Options page.*

Selecting Releases to Build

You select the releases that you want to build an App-V application for on the **Releases** tree of the **Build Options** page.

Important • *You cannot create or edit a release in the Microsoft App-V Assistant. If no releases exist, you can simply click the **Build** toolbar button to create a new release or open the **Releases** view of the InstallShield Installation Designer. You must create at least one release before you will be able to build an App-V application. For more information, see Creating and Building Releases.*

*If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Options** page is not displayed.*

Task: ***To select releases to build:***

1. Open the **Build Options** page.
2. Select the releases in the **Releases** tree that you want to build an App-V application for.

   Important • *When you select a release on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build an App-V application for that release. However, the releases that are selected on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Options** page:*

   - ***Active release selected****—A Windows Installer package and an App-V application would be built.*
   - ***Active release not selected****—Only a Windows Installer package would be built.*

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

Enabling App-V Application Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **Microsoft App-V Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build an App-V application for this Windows Installer package. To do this, perform the following steps:

Task: ***To enable App-V application building when in Direct Edit Mode:***

1. Open a Windows Installer package or a transform file in InstallShield. It will be opened in Direct Edit Mode or Direct MST Mode, and the Build function (**Build** on the **Build** menu and the **Build** toolbar button) will be disabled.
2. Open the **Build Options** page of the Microsoft App-V Assistant.
3. Select the **Build App-V application** option. After you select this option, the **Build App-V application** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Specifying Whether to Compress the Data Files in an App-V Package

InstallShield lets you specify whether you want to use zlib compression for the data files in the App-V package.

Task: ***To specify whether to compress the data files in the App-V package:***

1. Open the **Build Options** page.
2. Specify the appropriate option for the **Would you like to compress the data in the virtual package?** option.

Including Additional Windows Installer Packages in an App-V Application

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the App-V Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in an App-V application, set the **Would you like to include additional MSI files in the virtual package?** option on the **Build Options** page to **Yes**, and then select the packages that you want to add.

Task: ***To include additional Windows installer packages in an App-V application:***

1. Open the **Build Options** page.
2. Set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**.
3. Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
   - The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
   - Use the Delete button ( ) to delete a package from the list.

Building a Windows Installer Package to Assist in the Distribution of an App-V Application

You can choose to build a Windows Installer package to assist in the distribution of an App-V application. This simplifies the deployment of an App-V application by enabling you to use enterprise distribution tools such as Microsoft System Center Configuration Manager or Novell ZENworks Configuration Management.

To build a Windows Installer package with your App-V application, select the **Generate an installation package as part of the build output** check box on the Advanced Settings dialog box. You can access this dialog box by clicking the Advanced Settings link in the More Options area on the Build Options page of the Microsoft App-V Assistant. By default, this check box is not selected.

When you run this Windows Installer file, the minimally required App-V application files will be “installed” in the local App-V client system cache. (The `.sft` file remains on the App-V server until the client requests that it be downloaded when the application is launched for the first time.)

Note • *The App-V client must be installed on the local machine before you can install an App-V application. The installation will detect and warn if the App-V client is not available, and the installation will fail.*

To remove an installed App-V application, you need to use the Application Virtualization Client tool, which is available in the Administrative Tools of the Windows Control Panel.

Specifying Package Feature Block Optimizations

You can use the App-V Assistant **Package Optimizations** feature to control the performance and network traffic associated with running an App-V application. The package optimization option you select determines how quickly the App-V application will launch, and how often additional functionality will need to be streamed to the client while the App-V application is being used.

The files in an App-V application can be grouped into two feature blocks:

- **Feature block 1**—Feature block 1 must contain the core functionality of the App-V application that is necessary to launch the application. At application launch, all of the files in feature block 1 are streamed to the client in one unit.

- **Feature block 2**—Feature block 2 can contain additional functionality of the App-V application that is not necessary to launch the application. While the App-V application is being used, the files in feature block 2 can be streamed in small packets on an as-needed basis.

You can either choose to include all App-V application files in feature block 1 or, to improve launch speed, you can choose to group the files into two feature blocks: feature block 1 and feature block 2.

You indicate your package optimization preference on the **Package Optimizations** dialog box, which is opened by clicking the **Package Optimizations** link in the **More Options** menu of the **Build Options** page.

Figure 11-12: Package Optimizations Dialog Box

The **Package Optimizations** dialog box includes the following options:

Table 11-9 • Package Optimizations Options

Option	Description
Optimize for Streaming	If you choose this option, the App-V Assistant will perform a static analysis of the shortcuts in the application and decide which files should be in feature block 1 and which should be in feature block 2. This option provides a relatively quick launch time while limiting network traffic during application use. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent use of the application.*

Table 11-9 • Package Optimizations Options

Option	Description
Optimize for Offline Use	If you choose this option, all files in the App-V application will be included in feature block 1 and will be streamed to the client at start up in one file before the application launches. After that, no more streaming is done. All files are stored in the App-V cache, which means that the application is available for use even when the machine is not connected to the App-V server. Select this option if you want to enable users to use the App-V application when not connected to the App-V server and if you want to eliminate network traffic when the App-V application is being used. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent application use.*

Building an App-V Application

The method for building an App-V application depends upon what file you have open—an InstallShield project or a Windows Installer package.

- Building an App-V Application for an InstallShield Project
- Building an App-V Application for a Windows Installer Package

Building an App-V Application for an InstallShield Project

To build an App-V application for an InstallShield project, perform the following steps:

Task: ***To build an App-V application for an InstallShield project:***

1. Open the InstallShield project in InstallShield.
2. On the **Releases** view of the Installation Designer, make sure that at least one release has been created, and select the release that you want to build.

   Important • *You cannot create or edit a release in the Microsoft App-V Assistant. If no releases exist, or if you want to create a new release, open the* ***Releases*** *view of the InstallShield Installation Designer. You must create at least one release before you will be able to build an App-V application. For more information, see Creating and Building Releases.*
3. Open the **Build Options** page of the Microsoft App-V Assistant.
4. In the **Releases** tree, select the same release that is selected on the **Releases** view of the InstallShield Installation Designer. This is the release that you will build an App-V application for.

5. Click the **Build** toolbar button (or select **Build Release** on the **Build** menu) to start building the active release.

Important • *When you select a release on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build an App-V application for that release. However, the releases that are selected on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what was selected on the **Build Options** page:*

- ***Active release selected**—A Windows Installer package and an App-V application would be built.*
- ***Active release not selected**—Only a Windows Installer package would be built.*

To build more than one release at a time, perform a batch build. See Performing Batch Builds.

If you are building an App-V application for an InstallShield project, the output of the build is a Windows Installer package and an App-V application. For information on the files included in an App-V application, see Components of an App-V Package.

Building an App-V Application for a Windows Installer Package

To build an App-V application for a Windows Installer package, perform the following steps:

Task: ***To build an App-V application for a Windows Installer package:***

1. Do one of the following to open a Windows Installer package:
   - On the **File** menu, select **Open** and select a Windows Installer package (`.msi`).
   - On the **File** menu, select **Open** and select a transform file (`.mst`). The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
   - On the **File** menu, select **New** to open the **New Project** dialog box. Select **Transform** and click **OK**. The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
2. Use the Installation Designer to make any desired edits to the Windows Installer package or Transform file, and use the Microsoft App-V Assistant to set App-V application options.
3. Save the edits to the Windows Installer package or transform file by selecting **Save** on the **File** menu.
4. On the **Build Options** page of the Microsoft App-V Assistant, select the **Build App-V application** option. The **Build Virtual Package button is enabled.**
5. Click the **Build Virtual Package button** to start building the App-V application.

If you are building an App-V application for a Windows Installer package, the output of the build will be an App-V application. For information on the files included in an App-V application, see Components of an App-V Package.

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

Testing an App-V Application Using the App-V Application Launcher

You can use the InstallShield App-V Application Launcher to locally test a newly built App-V application before moving it to a deployment server.

To open the App-V Application Launcher, select **Test launch App-V application(s)** in the **More Options** list on the **Build Options** page. The App-V Application Launcher will attempt to launch that application. If there are multiple shortcuts (`.osd` files) in this App-V application, the **Launch App-V Application** dialog box opens, where you are prompted to select the shortcut you want to launch from a list of all of the shortcuts.

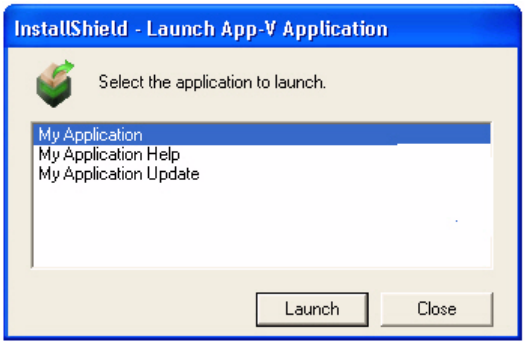


Figure 11-13: Launch App-V Application Dialog Box

Requirements for Using the App-V Application Launcher

The machine on which you use the App-V Application Launcher to test an App-V package must meet the following requirements:

- The Microsoft Application Virtualization Client must be installed.
- The version of the Microsoft Application Virtualization Client that is present should be equal to or newer than the minimum client version of the App-V package.
- File streaming must be enabled because the App-V Application Launcher publishes the App-V package from a local file path. If file streaming is not enabled, the App-V Application Launcher displays an informative message asking if it can enable this functionality.

App-V Application Launcher Utility Location

When an App-V application is built, the App-V Application Launcher utility (`AppVLauncher.exe`) is placed in the same folder as the App-V application (the same directory that contains the `.sft` and `.osd` files).

How the App-V Application Launcher Utility Works

On the **Package Information** page of the Microsoft App-V Assistant, you specify the URL of the App-V server where the `.sft` file of this App-V application will be deployed. This information is referenced in the `.osd` files in the generated App-V application. Therefore, testing an App-V application would seem to require that it be published on the server.

In order to make it possible to test the App-V package without having to publish it on the server or alter that URL path to the `.sft` file, a copy of the App-V Application Launcher utility is automatically copied to the output directory of each App-V application. Rather than looking for the `.sft` file in the location that you specified on the **Package Information** page, the App-V Application Launcher looks for the `.sft` file that is located in the same directory, ignoring the path set in the application.

***Note** • The first time that you use the App-V Application Launcher to run an application in an App-V package, the entire package is published to that machine; this includes all of the package's shortcuts and file extension associations in the package. If you then use the App-V Application Launcher to run any application in the App-V package again, the App-V Application Launcher unpublishes the package (and its shortcuts and file extension associations) before republishing the package.*

Also note that the `AppVLauncher.exe` file requires elevation. If you want to be able to test your App-V package in a locked-down environment where end users will not have elevated privileged, you may want to use the App-V Application Launcher once to launch and publish your App-V package with elevated privileges. Once you have done that, you can use the published shortcuts and file extension associations to start your application.

The App-V Application Launcher is a convenient testing tool that makes it possible for you to reliably and accurately test your App-V applications on your local machine or any other system that has the App-V client installed before moving it to the App-V server.

Controlling Whether to Include the App-V Application Launcher in Output

If you do not want to include the App-V Application Launcher with each App-V application that you build, clear the **Create utility to test launch App-V applications as part of building App-V applications** check box on the **Virtual Packages** tab of the AdminStudio **Options** dialog box.

Microsoft App-V Assistant Reference

Reference information about the Microsoft App-V Assistant is organized into the following sections:

- Pages
- Dialog Boxes
- Building App-V Applications Using the Command Line
- App-V Application Conversion Error and Warning Messages
- Application Features Requiring Pre- or Post-Conversion Actions

Pages

The Microsoft App-V Assistant is comprised of the following pages:

- Microsoft App-V Assistant Home Page
- Package Information Page
- Files Page
- Applications Page
- Registry Page
- Dynamic Suite Composition Page
- Build Options Page

Microsoft App-V Assistant Home Page

The Microsoft App-V Assistant Home page displays a diagram that illustrates the process of creating an App-V application.

Click the following icons in the navigation bar at the bottom of the page to navigate through the Microsoft App-V Assistant interface:

Table 11-10 • Navigation Bar Icons

Icon	Destination
	Package Information Page
	Files Page
	Applications Page
	Registry Page
	Build Options Page
	Go to next page.

Table 11-10 • Navigation Bar Icons

Icon	Destination
	Jump back to previous page.
	Microsoft App-V Assistant Home Page

Package Information Page

On the Package Information page, you specify the package name and enter a comment to document this virtual package. From this page, you can also choose to include diagnostic tools with the virtual package.

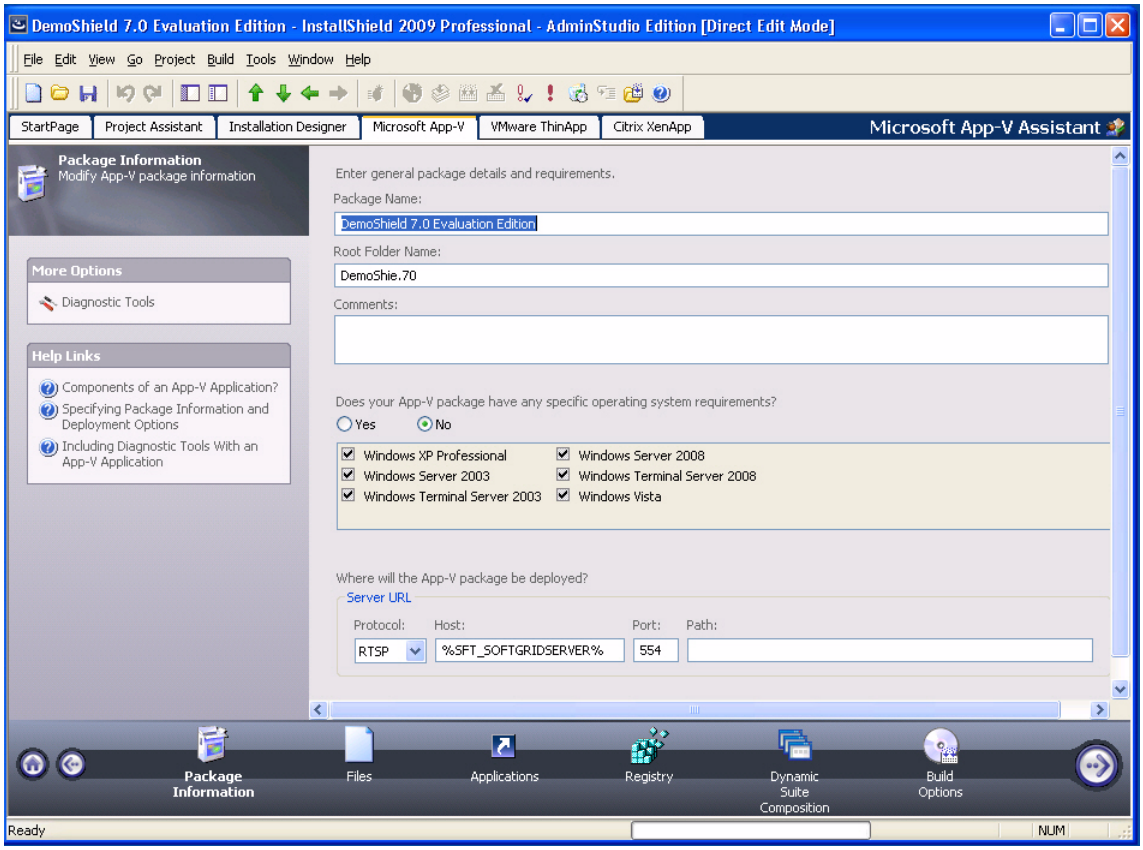


Figure 11-14: App-V Assistant Package Information Page

The Package Information page includes the following options:

Table 11-11 • Package Information Page

Option	Description
Package Name	Enter a name for the virtual package. ***Tip*** • *If your virtual package contains multiple applications, you can specify the name that identifies the entire package. For example, Microsoft Office could be used to identify a package that contains Microsoft Word and Microsoft Excel applications that run in the same virtual environment.*
Root Folder Name	Enter a name to identify the directory that will contain the App-V application's .sft file. When an App-V application is configured on a client machine, it is mounted on the computer's Q:\ drive in the folder you specify in this setting. The default value for the Root Folder Name setting is based on the **[ProductName]** and **[ProductVersion]** properties of the App-V application's associated Windows Installer file using the 8.3 file naming convention. For example: • If **[ProductName]** is `MyApplication` and **[ProductVersion]** is `1.12.3.1`, the Default Folder Name will be `MyApplic.112`. • If **[ProductName]** is `MyApp` and **[ProductVersion]** is `1`, then the Default Folder Name will be `MyApp.1`. • If **[ProductName]** is `MyBlueApp` and **[ProductVersion]** is `1.2.3.4`, then the Default Folder Name will be `MyBlueAp.123`. If you happen to have another App-V application that has identical values for the **[ProductName]** and **[ProductVersion]** properties, you can avoid conflicts by entering a unique value in the Root Folder Name setting.
Comments	Enter a short description of the purpose of this App-V application and any special information that you wish to document. ***Note*** • *The description you enter in this text box will appear in the OSD file ABSTRACT element.*

Table 11-11 • Package Information Page (cont.)

Option	Description
Does your App-V package have any specific operating system requirements?	Select one of the following: • **Yes**—Select this option if the application does not support one of the listed operating systems. When you select this option, the check boxes are unlocked and you can clear the selection of the unsupported operating systems. • **No**—Select this option if this application will run on all of the listed operating systems. When this option is selected, the operating system check boxes are locked and cannot be changed.
Operating System List	If you set the **Does your App-V package have any specific operating system requirements?** option to Yes, this list becomes enabled. To specify operating system requirements, select the operating systems that this application supports, and clear those that this application does not support.
Where will the App-V package be deployed?	Optionally identify the Server URL where this App-V package will be deployed, as described in Specifying the Deployment Server.
Diagnostic Tools	For testing purposes, you can choose to include diagnostic tools in your App-V application by clicking the Diagnostic Tools link in the More Options area. For more information, see App-V Diagnostic Tools Dialog Box.
Upgrade Settings	To specify upgrade information for your App-V package, click the Upgrade Settings link in the More Options area. For more information, see App-V Package Upgrade Settings Dialog Box.

Specifying the Deployment Server

To identify the server URL where this App-V package will be deployed, specify the following information:

Table 11-12 • Server URL Options

Option	Description
Protocol	Enables you to select the protocol that will stream the sequenced application package from a virtual application server to an Application Virtualization Desktop Client. The following protocols are available: • **RTSP**—The default, it specifies that the Real-Time Streaming Protocol controls the exchange of virtualization-enabled applications. • **RTSPS**—Specifies that the Real-Time Streaming Protocol with Transport Layer Security controls the exchange of a sequenced application package. • **FILE**—Specifies that the sequenced application will be streamed from a file share. • **HTTP**—Specifies that Hypertext Transport Protocol controls the exchange of a package. • **HTTPS**—Specifies that Secure Hypertext Transport Protocol controls the exchange of a package.
Host	Specify the virtual application server or the load balancer in front of a group of virtual application servers that will stream the software package to an Application Virtualization Desktop Client. You must complete this item to create a sequenced application package, but you can change from the default `%SFT_SOFTGRIDSERVER%` environment variable to the actual hostname or IP address of a virtual application server. ***Note*** • *If you choose not to specify a static hostname or IP address, on each Application Virtualization Desktop Client you must set up an environment variable called `SFT_SOFTGRIDSERVER`.* • *Its value must be the hostname or IP address of the virtual application server or load balancer that is this client's source of applications.* • *You should make this environment variable a system variable rather than a user variable.* • *Any Application Virtualization Desktop Client session that is running on this computer during your assignment of this variable must be closed and then opened so that the resumed session will be aware of its new application source.*
Port	Specify the port on which the virtual application server or the load balancer will listen for an Application Virtualization Desktop Client's request for the package. • This information is required to create a package, but you can change it. • The default port is 554.

Table 11-12 • Server URL Options

Option	Description
Path	Specify the relative path on the virtual application server where the App-V package is stored. This is also the path from which the App-V package is streamed. Note that it is not necessary to include the .sft file name in this path because the conversion process handles it automatically. If the App-V package is stored in a subdirectory of CONTENT, the path must be specified in this setting; otherwise, you can leave this setting blank. If you leave this setting blank, the App-V package is stored in a folder that has the same name that is used for the package name and the .sft file name.

Files Page

On the **Files** page of the Microsoft App-V Assistant, you can perform the following tasks:

- View Files and Folders
- Add Files and Folders
- Delete Files and Folders
- Set Isolation Options
- Modifying the Display of Predefined Folders

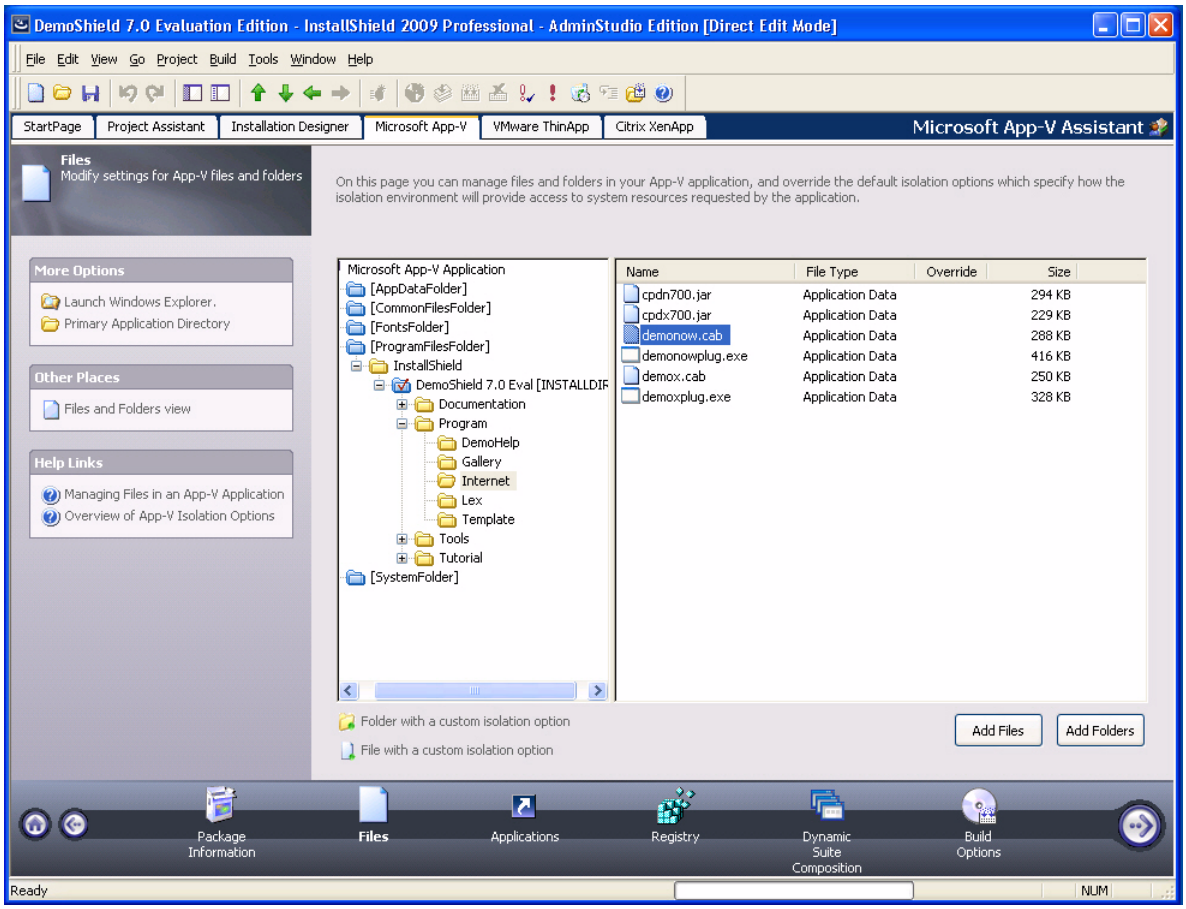


Figure 11-15: App-V Assistant Files Page

View Files and Folders

On the **Files** page, you can view all of the files and folders that are currently in your App-V application.

Folders are listed in the **Microsoft App-V application** tree on the left, and all of the files in the selected folder are listed on the right.

- The directories in the tree represent how your application will be organized within its secure compressed container.
- Blue folders are the supported MSI standard folders.
- The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

Add Files and Folders

On the **Files** page, you can use the **Add Files** and **Add Folders** buttons to add new files and folders to include in the App-V application. See Adding, Deleting, and Moving Files and Folders in an App-V Application.

If you are editing an InstallShield project (not a Windows Installer package), and you are adding a folder to this App-V application, you are prompted to choose whether you want to create a dynamic file link to the source folder.

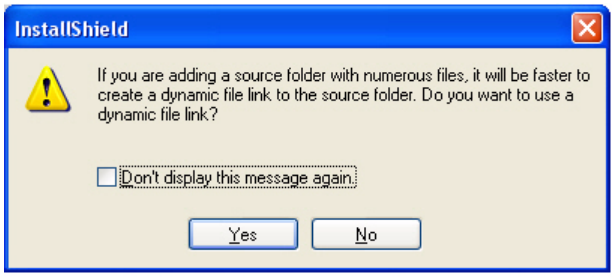


Figure 11-16: Dynamic File Link Option Dialog Box

Indicate whether you want to create a dynamic file link by selecting one of the following:

- **No**—For more flexibility with App-V options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
- **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

Delete Files and Folders

You can delete files and folders from the App-V application by selecting the file or folder you want to delete, and selecting **Delete** from the context menu. For more information, see Deleting Files and Folders.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains.*

Note • *You cannot delete predefined folders. You can only turn of the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Set Isolation Options

When using the Microsoft App-V Assistant, you can set isolation options to control application compatibility and accessibility. The isolation option that is assigned to a file or folder specifies how the isolation environment will provide access to system resources requested by the application.

You set isolation options on the **Isolation Options** dialog box, which is opened by selecting **Isolation Options** on the context menu when you have a file or folder selected on the **Files** page or a registry key selected on the **Registry** page.

You set isolation options by selecting a file or folder and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting Isolation Options for Folders and Files.

Modifying the Display of Predefined Folders

You can specify which of the Windows Installer predefined folders are listed in the **Microsoft App-V application** tree. See Controlling the Display of Predefined Folders.

Applications Page

You define shortcuts to enable users to launch an App-V application from within the virtual environment.

By default, the **Microsoft App-V Assistant** creates App-V applications for all of the executable shortcut that exist in your project. The project's shortcuts are listed in a checklist on the **Applications** page.

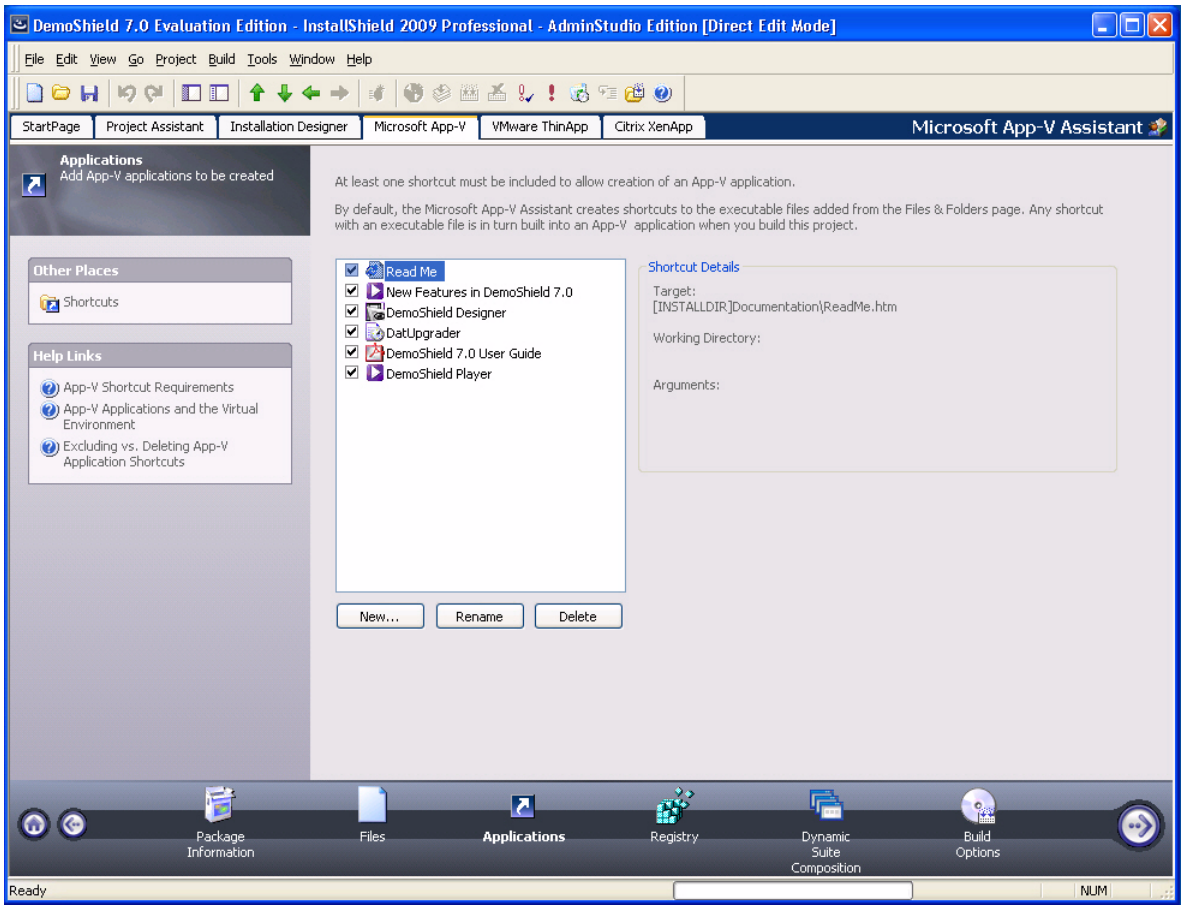


Figure 11-17: App-V Assistant Applications Page

Shortcut Requirements

For each App-V application, you are required to define **at least one** shortcut. You define application shortcuts to enable users to launch an App-V application from within the virtual environment. By default, the Microsoft App-V Assistant creates App-V applications for all of the executable shortcuts that exist in your project (or Windows Installer package).

If you build an App-V application that does not contain any shortcuts, users will not be able to launch the application.

Difference Between Deleting and Excluding a Shortcut

To prevent a shortcut from being created in the App-V application, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the App-V application, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding an App-V Application.
- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the App-V application and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting an App-V Application.

Managing Shortcuts

On the **Applications** page, you can create, delete, include, exclude, or rename an App-V application. For step-by-step instructions, see the following topics:

- Creating a New App-V Application
- Including an Existing App-V Application
- Excluding or Deleting an Existing App-V Application
- Renaming an App-V Application

Registry Page

On the **Registry** page, you can view existing registry keys, values, and data, and add or delete registry items. You can also override the default isolation options for a registry key. Isolation options specify how the virtual environment will provide access to system resources requested by the application.

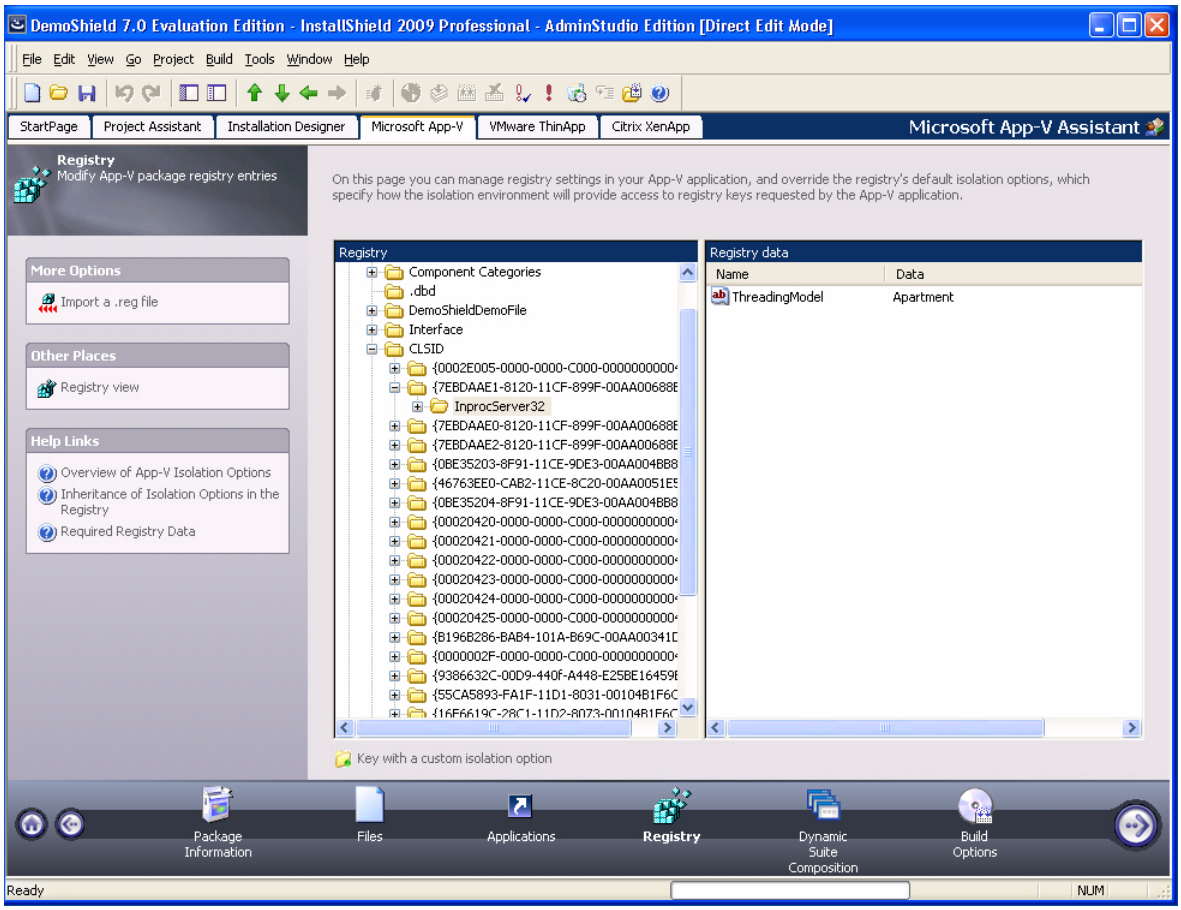


Figure 11-18: App-V Assistant Registry Page

The default settings for isolation options are built into the Microsoft App-V Assistant, and those defaults are adequate for most environments. However, you can override the default settings for selected registry keys to exert control over application interactions with client operating system resources.

You set isolation options by selecting a registry key and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting App-V Application Registry Isolation Options.

Registry items that are listed on this page will be included in the App-V application, and those that you delete will not. By default, all new registry keys are isolated.

Tip • *To import an existing registry (.reg) file, click the **Import a .reg file** option on the **More Options** list to open the Registry Import Wizard.*

Note • *You cannot set isolation options on root registry keys.*

Editing the registry on the Registry page is performed much like it is performed in the InstallShield Registry view. To learn more, see Editing the Registry.

Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

Dynamic Suite Composition Page

The point of application virtualization is to minimize the system dependencies that an application has on the underlying physical system. Many applications have common system dependencies on plug-ins or middleware, such as Adobe Reader or ODBC drivers.

Dynamic Suite Composition (DSC) is a Microsoft Application Virtualization feature that enables applications to be virtualized separately from the plug-ins and middleware applications that they rely on, while still enabling them to communicate with those plug-ins and middleware applications within the virtual environment. The primary App-V application and the dependency App-V applications in the dynamic suite will run and interact with one another as if they were all installed locally on a computer. You would only need to deploy common system components once on each client, making them available for use by many App-V applications, rather than to include them with each of the App-V applications that are dependent upon them. This reduces redundancy in the local App-V cache and simplifies the construction and testing of the primary App-V applications.

You specify the App-V applications that you want to include in a dynamic suite on the **Dynamic Suite Composition** page of the App-V Assistant.

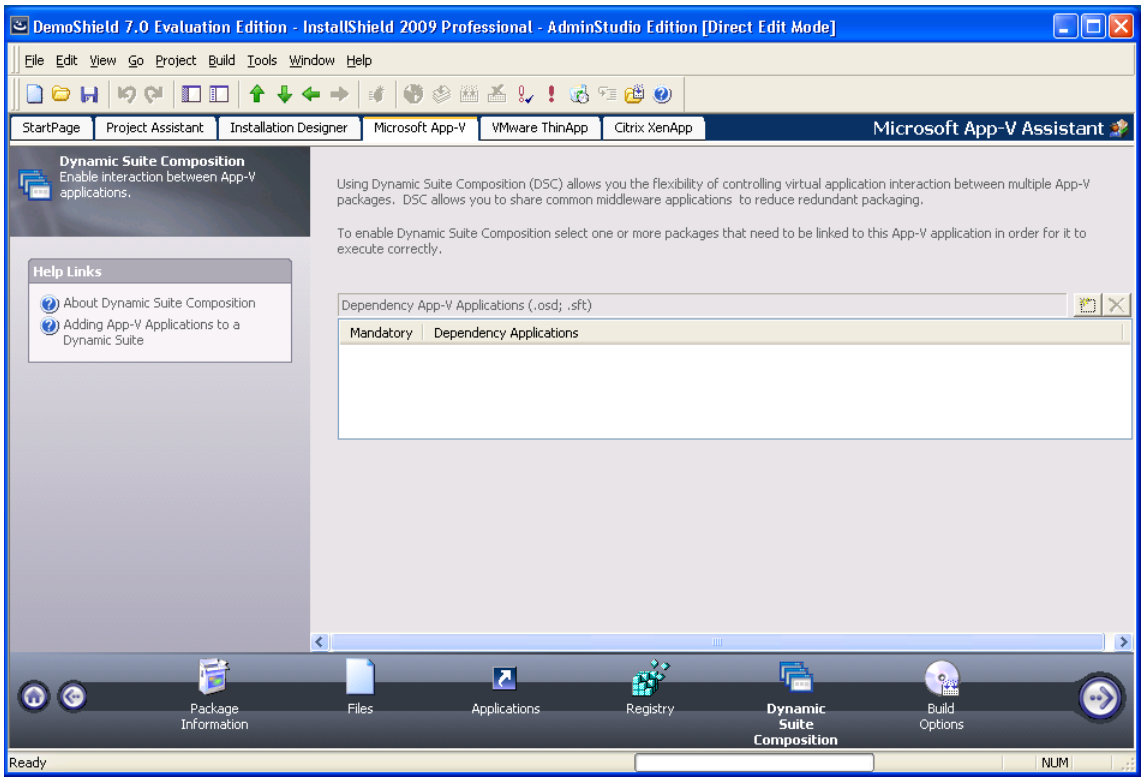


Figure 11-19: Microsoft App-V Assistant Dynamic Suite Composition Page

The **Dynamic Suite Composition** page has the following options:

Table 11-13 • Dynamic Suite Composition Page

Option	Description
Dependency App-V Applications List	List of all of the dependency App-V applications that have been selected for the primary App-V application.
Mandatory	Indicates whether to selected dependency App-V application is required in order for the primary App-V application to run. • **If the dependency App-V application is mandatory**—If the primary App-V application will not run unless it can locate this dependency App-V application, leave the **Mandatory** option selected. If a dependency App-V application that is configured as **Mandatory** is not available, an error will be generated when someone attempts to run the primary App-V application. • **If the dependency App-V application is not mandatory**—If the primary App-V application will run even if it cannot locate this dependency App-V application, clear the **Mandatory** option.

Table 11-13 • Dynamic Suite Composition Page

Option	Description
New Button	To add an App-V application to the **Dependency App-V Applications** list, click this button and select the App-V application (`.osd`, `.sft`) that you want to add. Select one of the following: • **One of the .osd files**—If this dependency App-V application is or is going to be published on a server, select any one of the `.osd` files that are listed. If these `.osd` files were created properly, each of them should contain the information that will identify to the primary App-V application the published location of the dependency App-V application. It is not necessary to select more than one `.osd` file. All of them contain the same reference to the location of the dependency App-V application's `.sft` file, which is the only reference that is necessary in order for the primary App-V application to locate it. • **The .sft file**—If this dependency App-V application is or is going to be published locally on the client or at an accessible network location, you may select just the `.sft` file. The selected reference App-V application is now listed in the **Dependency Applications** list and, by default, the **Mandatory** option is selected.
Delete Button	Click to delete the selected App-V application from the list.

Build Options Page

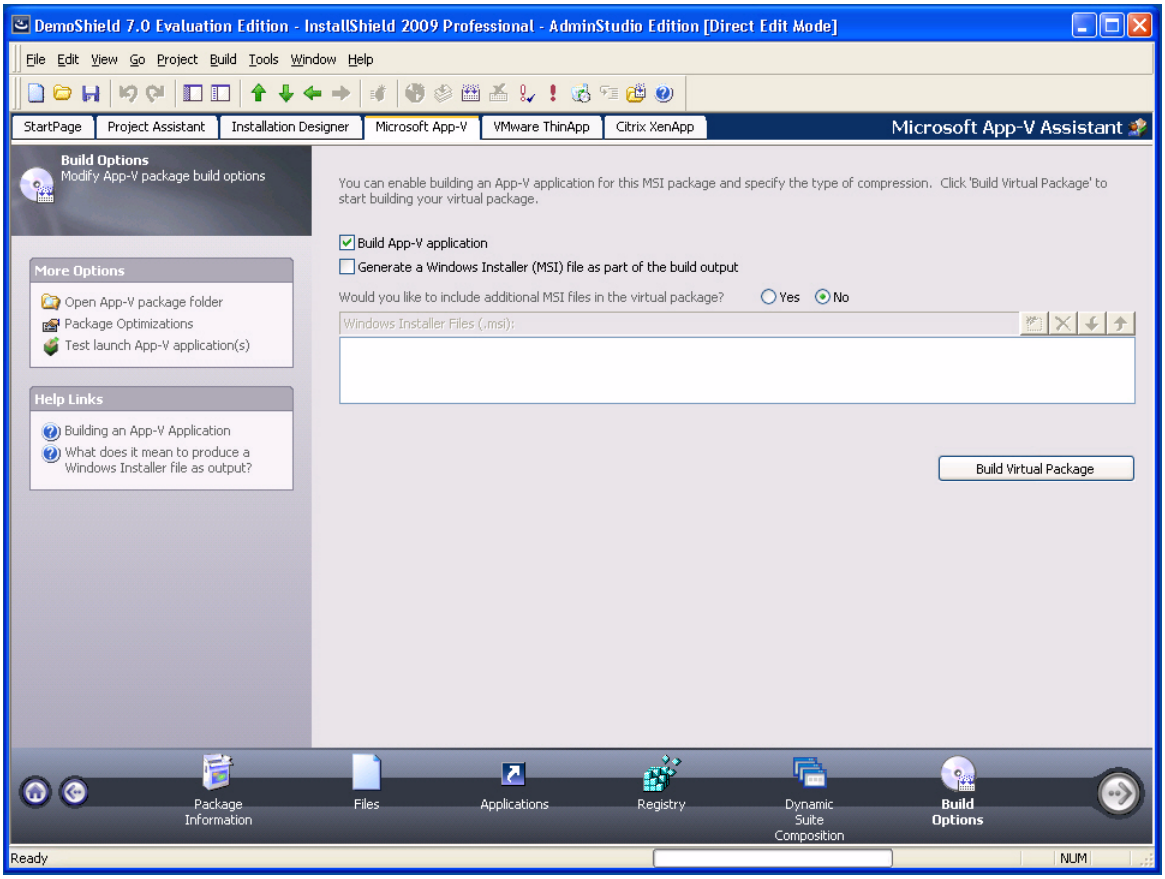


Figure 11-20: App-V Assistant Build Options Page

The options on the Build Options page vary depending upon whether you are editing an InstallShield project or a Windows Installer package:

- **InstallShield Project [Basic MSI Project Mode]**—When you open an InstallShield project in InstallShield, the following conditions are true:
  - The **Build Options** page includes a releases tree, and you select the release that you want to build.
  - To build the App-V application, you click the **Build** button on the toolbar.
- **Windows Installer Package [Direct Edit Mode]**—When you open a Windows Installer package in InstallShield, the following conditions are true:
  - Because you do not have to select a release for a Windows Installer package, there is no releases tree.
  - Because a Windows Installer package has already been built, InstallShield's standard build functionality is disabled. To build the App-V application, select the **Build App-V application** option and click the **Build Virtual Package** button.

Specifying Build Options

Compressing the Data in an App-V Package

To use zlib compression to compress the data in the App-V package, set the **Would you like to compress the data in the virtual package?** option to **Yes**.

Including Additional Windows Installer Packages in an App-V Application

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the Microsoft App-V Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in an App-V application, set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**, and then select the packages that you want to add.

- Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
- The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
- Use the Delete button ( ) to delete a package from the list.

Selecting Releases to Build

You select the releases that you want to build an App-V application for on the **Releases** tree of the **Build Options** page. By selecting a release, you are specifying that whenever that particular release is built, an App-V application will also be built.

Note • *If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the* ***Releases*** *tree on the* ***Build Options*** *page is not displayed.*

About Building Releases

When you select a release on the Releases tree on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build an App-V application for that release. However, the releases that are selected on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Options** page:

- **Active release selected**—A Windows Installer package and an App-V application would be built.
- **Active release not selected**—Only a Windows Installer package would be built.

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

About Creating Releases

You cannot create or edit a release in the Microsoft App-V Assistant. If no releases exist, you can simply click the **Build** toolbar button to create a new release or open the **Releases** view of the InstallShield Installation Designer. You must create at least one release before you will be able to build an App-V application. For more information, see *Creating and Building Releases.*

If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Options** page is not displayed.

Choosing to Build a Windows Installer Package to Assist in the Distribution of an App-V Application

You can choose to build a Windows Installer package to assist in the distribution of an App-V application by selecting the **Generate an installation package as part of the build output** check box on the Advanced Settings dialog box. You can access this dialog box by clicking the Advanced Settings link in the More Options area on the Build Options page of the Microsoft App-V Assistant. By default, this check box is not selected.

When you run this Windows Installer file, the minimally required App-V application files will be "installed" in the local App-V client system cache. (The `.sft` file remains on the App-V server until the client requests that it be downloaded when the application is launched for the first time.)

Note • *The App-V client must be installed on the local machine before you can install an App-V application. The installation will detect and warn if the App-V client is not available, and the installation will fail.*

This simplifies the deployment of an App-V application by enabling you to use enterprise distribution tools such as Microsoft System Center Configuration Manager or Novell ZENworks Configuration Management.

To remove an installed App-V application, you need to use the Application Virtualization Client tool, which is available in the Administrative Tools of the Windows Control Panel.

Enabling App-V Application Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **Microsoft App-V Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build an App-V application for this Windows Installer package. To enable the **Build** button to build just the App-V application, select the **Build App-V application** option on the **Build Options** page.

After you select this option, the **Build App-V application** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Building an App-V Application

The method for building an App-V application depends upon what file you have open—an InstallShield project or a Windows Installer package. For detailed instructions, see one of the following topics:

- Building an App-V Application for an InstallShield Project

- Building an App-V Application for a Windows Installer Package

Testing an App-V Application Using the App-V Application Launcher

From the **Build Options** page, you can choose to test your App-V application using the App-V Application Launcher. To open the App-V Application Launcher, select **Test launch App-V application(s)** in the **More Options** list on the **Build Options** page.

For more information, see Testing an App-V Application Using the App-V Application Launcher and Launch App-V Application Dialog Box.

Opening the App-V Application Folder

To quickly open the folder containing the App-V application files that were generated when this InstallShield project or Windows Installer package was built, click **Open App-V application folder** in the **More Options** list.

Dialog Boxes

The Microsoft App-V Assistant includes the following dialog boxes:

- App-V Diagnostic Tools Dialog Box
- File Isolation Options Dialog Box
- Folder Isolation Options Dialog Box
- Launch App-V Application Dialog Box
- Package Optimizations Dialog Box
- Primary Application Directory Dialog Box
- Registry Isolation Options Dialog Box

Advanced Settings Dialog Box

Edition • *The Microsoft App-V Assistant is included in the Virtualization Pack.*

The Advanced Settings dialog box opens when you click the Advanced Settings link in the More Options section on the Build Options tab. This dialog box is where you specify build and run-time options.

Table 11-15 • Advanced Settings Dialog Box Options

Option	Description
Generate an installation package as part of the build output	To build an installation package with your App-V application, select this check box. This check box is not selected by default. If you do select this check box, you can specify whether you want to load the installation package from the media location or from the shared location. Building an installation package enables you or your end users to use enterprise distribution tools such as Microsoft System Center Configuration Manager or Novell ZENworks Configuration Management to distribute your App-V application. When you run this Windows Installer file, the minimally required App-V application files will be "installed" in the local App-V client system cache. (The .sft file remains on the App-V server until the client requests that it be downloaded when the application is launched for the first time.) ***Note*** • *The App-V client must be installed on the local machine before you can install an App-V application. The installation will detect and warn if the App-V client is not available, and the installation will fail.* To remove an installed App-V application, you need to use the Application Virtualization Client tool, which is available in the Administrative Tools of the Windows Control Panel.
Load from Media Location	To load the installation package from the media location, select this option. If you select this option, the following check boxes are available for selection: • **Compress**—To build a compressed installation package, select this check box. If this check box is cleared, an uncompressed installation package is built. • **App-V Client Prerequisite (Generates Setup.exe)**—If you want to include the AdminStudio prerequisite that installs the App-V client on the target system, select this check box. Note that a Setup.exe setup launcher is required if the AdminStudio prerequisite needs to be included in the release.
Load from Shared Location	To load the installation package from the shared location, select this check box.
App-V application Test Launcher	To use the App-V application test launcher to locally test a newly built App-V application before moving it to a deployment server, select this check box. For more information, see Testing an App-V Application Using the App-V Application Launcher.

App-V Diagnostic Tools Dialog Box

On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **Package Information** page, you can choose to include the Registry Editor and Windows Command Prompt diagnostic tools with your App-V application.

If you include diagnostic tools with your App-V application, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running an App-V application and you encountered an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

The Registry Editor diagnostic tool lets you use `Regedit.exe` on the local machine and have access to the virtual environment. The Command Prompt diagnostic tool lets you use `Cmd.exe` on the local machine and have access to the virtual environment.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the App-V application.

When the user runs this App-V application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

App-V Package Upgrade Settings Dialog Box

The App-V Package Upgrade Settings dialog box is where you specify whether you want to create an upgrade for your App-V package. If you specify that you do want to create an upgrade, you can specify additional information about the upgrade.

Table 11-16 • App-V Package Upgrade Settings Dialog Box Settings

Setting	Description
Enable Upgrade	To create an upgrade for an earlier App-V package, select this check box. If you do not want to create an upgrade, clear this check box. When this check box is cleared, the other settings on the App-V Package Upgrade Settings dialog box are disabled. This check box is cleared by default.
Always upgrade latest built package	If you selected the Enable Upgrade check box and you want InstallShield to build an upgrade that updates the latest built App-V package, select this option.

Table 11-16 • App-V Package Upgrade Settings Dialog Box Settings (cont.)

Setting	Description
Choose package to upgrade	If you selected the Enable Upgrade check box and you want InstallShield to build an upgrade that updates a particular App-V package, select this option, and then specify the path of the earlier package that you want to be updated.
Append version number to package name	If you selected the Enable Upgrade check box and you want InstallShield to append the version number to the App-V package name, select this check box.

File Isolation Options Dialog Box

On the **File Isolation Options** dialog box, you can override the default isolation options for the selected file.

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, App-V, and registry settings.*

The **File Isolation Options** dialog box includes the following options:

Table 11-17 • File Isolation Options Dialog Box

Option	Description
File Type	When an App-V application performs a WRITE operation to a resource (a file or folder), the resource's **File Type** setting determines whether changes to that resource are saved for all users of the App-V application on that machine or only for the logged-in user. Select one of the following options: • **Application Data**—Changes to the resource are saved for all users of this App-V application on this machine. • **User Data**—Changes to the resource are saved only for the logged-in user.
Override	To have the virtualization client overwrite the file when the application is upgraded and streamed from the App-V server to the client, select this check box. If Override check box is not selected, then the App-V client determines whether to overwrite the file during an upgrade. In general, if the file is marked as Application Data, the file is overwritten during an upgrade; otherwise, it is not overwritten.

Folder Isolation Options Dialog Box

On the **Folder Isolation Options** dialog box, you can override the default isolation options for the selected folder.

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, App-V, and registry settings.*

The **Folder Isolation Options** dialog box includes the following options:

Table 11-18 • Folder Isolation Options Dialog Box

Option	Description
File Type	When an App-V application performs a WRITE operation to a resource (a file or folder), the resource's **File Type** setting determines whether changes to that resource are saved for all users of the App-V application on that machine or only for the logged-in user. Select one of the following options: • **Application Data**—Changes to the resource are saved for all users of this App-V application on this machine. • **User Data**—Changes to the resource are saved only for the logged-in user.
Override	To have the virtualization client overwrite the folder when the application is upgraded and streamed from the App-V server to the client, select this check box. If Override check box is not selected, then the App-V client determines whether to overwrite the folder during an upgrade. In general, if the folder is marked as Application Data, the file is overwritten during an upgrade; otherwise, it is not overwritten.

Launch App-V Application Dialog Box

When using the InstallShield App-V Application Launcher to locally test a newly built App-V application before moving it to a deployment server, the **Launch App-V Application** dialog box opens when an App-V application has more than one shortcut (`.osd` files). You are prompted to select the shortcut you want to launch from a list of all of the shortcuts.

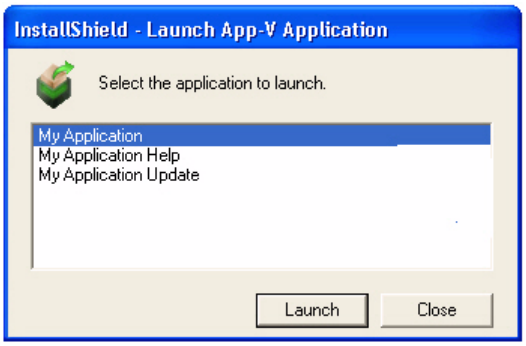


Figure 11-21: Launch App-V Application Dialog Box

To open the App-V Application Launcher, select **Test launch App-V application(s)** in the **More Options** list on the **Build Options** page.

Package Optimizations Dialog Box

You can use the App-V Assistant **Package Optimizations** feature to control the performance and network traffic associated with running an App-V application. The package optimization option you select determines how quickly the App-V application will launch, and how often additional functionality will need to be streamed to the client while the App-V application is being used.

You indicate your package optimization preference on the **Package Optimizations** dialog box, which is opened by clicking the **Package Optimizations** link in the **More Options** menu of the **Build Options** page.

Figure 11-22: Package Optimizations Dialog Box

The files in an App-V application can be grouped into two feature blocks:

- **Feature block 1**—Feature block 1 must contain the core functionality of the App-V application that is necessary to launch the application. At application launch, all of the files in feature block 1 are streamed to the client in one unit.
- **Feature block 2**—Feature block 2 can contain additional functionality of the App-V application that is not necessary to launch the application. While the App-V application is being used, the files in feature block 2 can be streamed in small packets on an as-needed basis.

You can either choose to include all App-V application files in feature block 1 or, to improve launch speed, you can choose to group the files into two feature blocks: feature block 1 and feature block 2.

The **Package Optimizations** dialog box includes the following options:

Table 11-19 • Package Optimizations Options

Option	Description
Optimize for Streaming	If you choose this option, the App-V Assistant will perform a static analysis of the shortcuts in the application and decide which files should be in feature block 1 and which should be in feature block 2. This option provides a relatively quick launch time while limiting network traffic during application use. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent use of the application.*
Optimize for Offline Use	If you choose this option, all files in the App-V application will be included in feature block 1 and will be streamed to the client at start up in one file before the application launches. After that, no more streaming is done. All files are stored in the App-V cache, which means that the application is available for use even when the machine is not connected to the App-V server. Select this option if you want to enable users to use the App-V application when not connected to the App-V server and if you want to eliminate network traffic when the App-V application is being used. ***Note*** • *When application files are streamed to a client either at launch or during application use, they are saved in the App-V cache and do not need to be streamed again during subsequent application use.*

Primary Application Directory Dialog Box

When App-V applications are run on a machine that has Microsoft Virtualization Client installed, they are run from the Client's application virtual drive.

For optimum performance, it is desirable to have the bulk of the application's files mounted to this drive at run-time. In order to achieve this, it is useful to determine an application's primary application directory so that folder can be mounted to the App-V Client's application virtual drive when the App-V application is loaded.

The InstallShield Microsoft App-V Assistant uses a series of steps to determine an App-V application's default primary application directory (as described in Determining the Default Primary Application Directory).

However, you can specify the primary application directory for an App-V application on the **Primary Application Directory** dialog box, which is opened by clicking **Primary Application Directory** in the **More Options** list on the **Files** page.

The **Primary Application Directory** dialog box displays the current primary application directory setting (if one has already been specified).

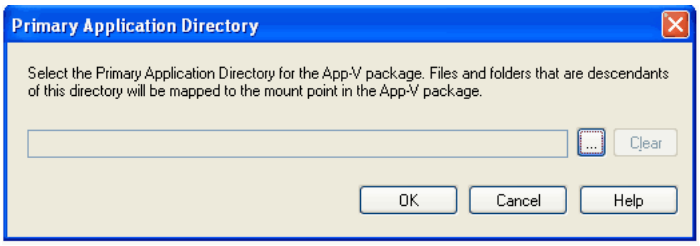


Figure 11-23: Primary Application Directory Dialog Box

When you click the browse (...) button, the **Browse for Directory** dialog box opens, listing all of the currently available destination directories for this App-V application. Select one of the listed directories and click **OK**.

Determining the Default Primary Application Directory

When an App-V application is built using InstallShield or any AdminStudio tool, the following series of steps are used to determine an App-V application's primary application directory:

Table 11-20 • Steps to Automatically Determine Primary Application Directory

#	Step	Description
1	**Explicitly set primary application directory**	If a directory is specified on the InstallShield App-V Assistant's **Primary Application Directory** dialog box (as described in Explicitly Setting the Primary Application Directory), that directory will be used.
2	**Value of INSTALLDIR variable**	If the Windows Installer package has a value for `INSTALLDIR` (a system variable that specifies the root destination directory for an installation), that value will be used as the primary application directory. ***Note*** • *This step is not used when converting Windows Installer packages that have been repackaged using AdminStudio.* ***Note*** • *All Windows Installer packages created by InstallShield or AdminStudio will have a value for the INSTALLDIR variable.*
3	**Location of shortcut in a subdirectory of the ProgramFilesFolder**	If one of the `.exe` targets for a shortcut is in a subdirectory of `ProgramFilesFolder`, then the folder directly under `ProgramFilesFolder` will be used as the primary application directory. Typically, this would be: `C:\Program Files\YourApplication`
4	**Location of shortcut in a directory other than ProgramFilesFolder**	If no `.exe` targets are located in a subdirectory of `ProgramFilesFolder`, then the target directory of a shortcut that contains an `.exe` will be used.

Table 11-20 • Steps to Automatically Determine Primary Application Directory

#	Step	Description
5	**ProgramFilesFolder**	If none of the above can be found, then the primary application directory will be set to `ProgramFilesFolder`. Typically, this would be: C:\Program Files

Registry Isolation Options Dialog Box

You open the **Registry Isolation Options** dialog box by selecting a registry key on the **Registry** page and clicking **Isolation Options** on the context menu. On the **Registry Isolation Options** dialog box, you can specify whether the App-V application will see only the registry entries for the selected key that are part of that App-V application, or will see a merged view of the registry entries for the selected key from both the local registry and from the App-V application's registry.

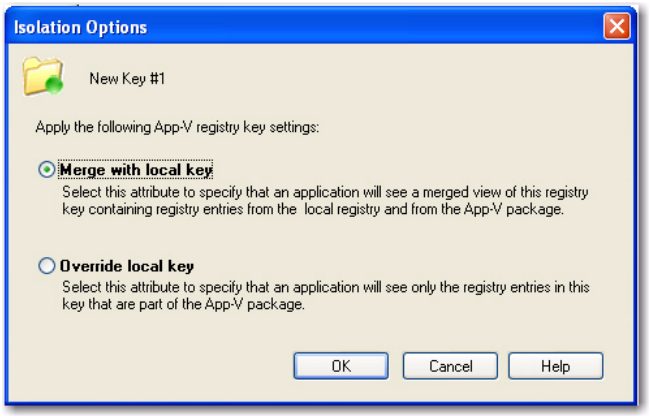


Figure 11-24: Registry Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

The **Registry Isolation Options** dialog box includes the following options;

Table 11-21 • Registry Isolation Options

Option	Description
Merge with local key	Select this option to specify that the App-V application will see a merged view of the registry entries for the selected key from both the local registry and from the App-V application's registry.
Override local key	Select this option to specify that the App-V application will see only the registry entries for the selected key that are part of that App-V application.

Building App-V Applications Using the Command Line

When you configure an App-V application in an InstallShield project and then build that project (using either the user interface or the command line), both the Windows Installer package and the App-V application are built. When you use the standard InstallShield command line build, you do not need to add any additional command line parameters. All of the App-V application settings are saved within the InstallShield project.

App-V Application Conversion Error and Warning Messages

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

Application Features Requiring Pre- or Post-Conversion Actions

Some application features are ignored when creating an App-V application. Therefore, some additional pre- or post-conversion actions must be taken in order for the App-V application to be created properly.

One action you could take to try to include ignored features in an App-V application is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to an App-V application.

For a list of ignored features, see Application Features Requiring Pre- or Post-Conversion Actions.

Creating ThinApp Applications

Edition • *The ThinApp Assistant is included in the Virtualization Pack.*

Important • *ThinApp support requires a separate purchase of VMware® ThinApp™.*

ThinApp (formerly Thinstall Virtualization Suite) is a self-contained application virtualization solution that requires no client-side agents or supporting server infrastructure. A ThinApp application runs within a virtual environment that prevents it from interfering with other software running on the same machine.

You can use the InstallShield **ThinApp Assistant** to configure and build a ThinApp application. Information about creating ThinApp applications using the ThinApp Assistant is presented in the following sections:

- Overview of the ThinApp Assistant
- Using the ThinApp Assistant to Create a ThinApp Application
- ThinApp Assistant Reference
- ThinApp Application Configuration File: package.ini

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

Overview of the ThinApp Assistant

A ThinApp application is a self-contained application virtualization solution that requires no client-side agents or supporting server infrastructure. A ThinApp application runs within a virtual environment that prevents it from interfering with other software running on the same machine. You can use the InstallShield **ThinApp Assistant** to configure and build a ThinApp application.

- About ThinApp Applications
  - The ThinApp Virtual Operating System
  - Benefits of Deploying ThinApp Applications
- About the ThinApp Assistant
  - Process for Authoring a ThinApp Application Using the ThinApp Assistant
  - Components of a ThinApp Application
  - Supported InstallShield Project Types
  - How Transforms are Included in a ThinApp Application
  - About Sandboxes

About ThinApp Applications

ThinApp applications can be deployed on a machine without modifying the local operating system or file system. They run in a “sandbox” (or virtual environment) which protects the local operating system from installation modifications that could affect stability or security. Also, ThinApp applications can be run safely from restricted user accounts without local installation.

Information about ThinApp applications is presented in the following sections:

- The ThinApp Virtual Operating System
- Benefits of Deploying ThinApp Applications

The ThinApp Virtual Operating System

A ThinApp application runs in a virtual operating system—a small light-weight component which is embedded with each ThinApp application—that consists of a virtual file system and a virtual registry. When the ThinApp application is run, the virtual operating system environment is merged with the real system environment.

The virtual operating system technology enables entire applications to be packaged into a single `.exe` file that can be run without an installation process, and without modifying the resident operating system.

A ThinApp application can be run from a network or offline on the local machine.

Benefits of Deploying ThinApp Applications

Deploying ThinApp applications provides the following benefits:

- **Reduces time to deployment and costs associated with testing**—Applications can be deployed and run in independent sandboxes, eliminating the need for expensive and time-consuming multi-application regression testing. This reduces the time to deployment and the costs associated with testing.
- **Fast, lightweight virtualization** —ThinApp does not use emulation, so all processes are executed natively at full speed.
- **Reduces the cost of maintaining secure locked-down desktops**—ThinApp applications can run in restricted user accounts without requiring any host modifications.
- **Enhances work-force mobility, business continuity and disaster recovery**—ThinApp applications can be run off-line, directly from any external media including USB Flash, CD-ROM, and off-line laptops.
- **No infrastructure changes needed**—ThinApp applications can be deployed using any existing software deployment systems including Active Directory and SMS. ThinApp has no client or server components to manage or maintain and ThinApp can transparently stream large applications from any network attached storage devices without server software.
- **Sandboxing prevents modifications**—ThinApp redirects all changes intended for the host computer's file system and registry to a private per-user sandbox. Sandboxes can be located on a network share, allowing application settings to follow users as they move from machine to machine. For mobile users, sandboxes can be stored on local USB flash drives, thus preventing damage to the host computer or accidental host storage of sensitive data.

About the ThinApp Assistant

Information about the ThinApp Assistant is organized into the following sections:

- Process for Authoring a ThinApp Application Using the ThinApp Assistant
- Components of a ThinApp Application
- Supported InstallShield Project Types
- How Transforms are Included in a ThinApp Application
- About Sandboxes

Process for Authoring a ThinApp Application Using the ThinApp Assistant

You can use the ThinApp Assistant to convert a Windows Installer package into a ThinApp application. During this process, you:

- **General Settings**—Specify sandbox and Active Directory settings.

- **Files, Folders, Shortcuts, Registry Settings**—Specify the files, folders, shortcuts, and registry settings that will be included in the ThinApp application.
- **Isolation Options**—Override the default isolation options for selected folders and registry keys.
- **Build**—Specify build options and build a ThinApp application.

The following diagram illustrates the ThinApp application creation process:

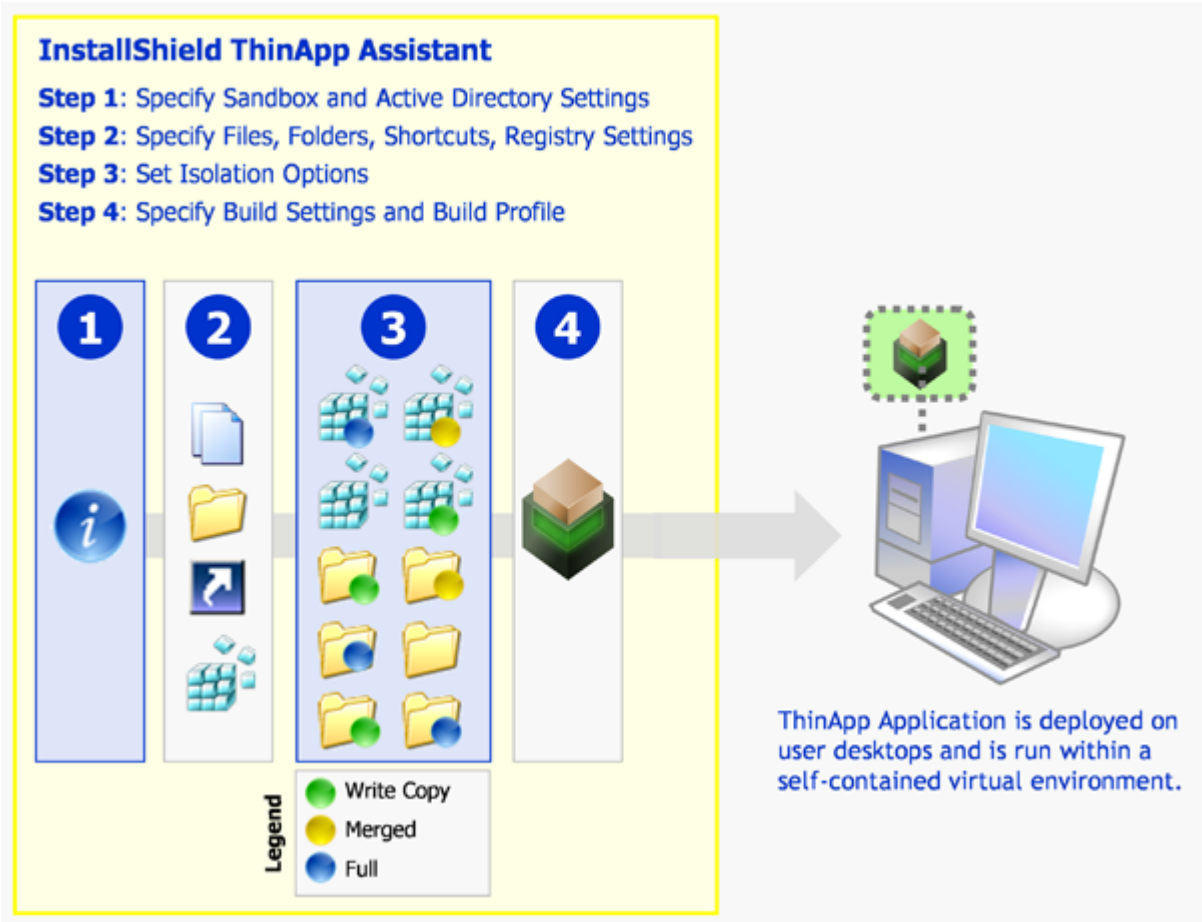


Figure 11-25: Creating a ThinApp Application

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

The process for authoring a ThinApp application using the ThinApp Assistant is as follows:

Table 11-22 • Steps to Convert a Windows Installer Package to a ThinApp Application

Step	Go To:	Actions
Getting Started	**InstallShield Start Page**	Create or open one of the following project types: • Basic MSI • MSI Database (Direct Edit Mode) • Transform (Direct MST Mode)
	InstallShield Start Page	Click on the **VMware ThinApp** tab to open the **ThinApp Assistant Home** page.
	ThinApp Assistant Home Page	Click **General Settings** in the navigation bar to open the **General Settings** page.
Specifying ThinApp General Settings	**General Settings Page**	Specify the sandbox name and sandbox options for the ThinApp application, control access to the ThinApp application via Active Directory, and specify whether to include diagnostic tools with the ThinApp application.
Managing Files and Folders in a ThinApp Application	**Files & Folders Page**	View existing files and folders, add and delete files.
Setting ThinApp Isolation Options	**Files & Folders Page**	Override the default isolation options for selected folders. Isolation options specify how the virtual environment will provide access to folders requested by the ThinApp application.
Modifying Shortcuts to the ThinApp Application's Executable Files	**Applications Page**	Create, delete, include, exclude, or rename ThinApp application executables, which are derived from the shortcuts in its Windows Installer package.
Modifying ThinApp Application Registry Settings	**Registry Page**	Add, delete, or modify the registry settings in your ThinApp application, and override the default isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to registry keys requested by the ThinApp application.

Table 11-22 • Steps to Convert a Windows Installer Package to a ThinApp Application

Step	Go To:	Actions
Modifying Build Options	**Build Options Page**	[Basic MSI Project mode] Select the releases that you want to build. [Direct Edit or Direct MST mode] To enable the Build function for a ThinApp application, select the **Build ThinApp application** option.
Building a ThinApp Application	**Build on the Toolbar** OR **Build Virtual Package Button**	Click **Build** to build the active Release and create a ThinApp application. When you are in Direct Edit mode, click the **Build Virtual Package** button to save the Windows Installer package and create a ThinApp application.

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

Components of a ThinApp Application

When you use the ThinApp Assistant to build a ThinApp virtual package, the resources you generate are called ThinApp applications. The number of files included in a ThinApp application depends upon how many shortcuts are defined in the project (or Windows Installer package) and whether you choose to include diagnostic tools with the ThinApp application.

Table 11-23 • Components of a ThinApp Application

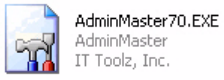
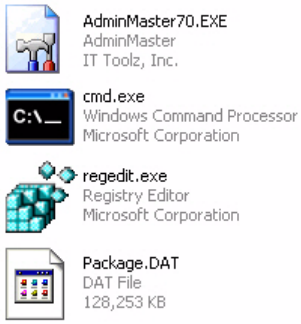
Number of Shortcuts	ThinApp Application Components	Description
1 shortcut	`ProductName.exe` 	The ThinApp application consists of a single executable (`.exe`) file: • **Launching the application**—This executable file is used to launch the ThinApp application. • **Location of application data**—This executable file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

Table 11-23 • Components of a ThinApp Application

Number of Shortcuts	ThinApp Application Components	Description

More than 1 shortcut	`ProductName.exe`
	`FeatureName.exe`
	`Package.DAT`
	
		The ThinApp application consists of two or more executable files and a `Package.DAT` file:
		• **Launching the application**—Each of the executables is used to launch the ThinApp application or a specific feature of the ThinApp application.
		• **Location of application data**—The `Package.DAT` file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the application to run.

1 shortcut with diagnostic tools	`ProductName.exe`
	`cmd.exe`
	`regedit.exe`
	`Package.DAT`
	 
		The ThinApp application consists of three executable files and a `Package.DAT` file:
		• **Launching the application**—The package executable is used to launch the ThinApp application.
		• **Launching the diagnostic tools**—The `cmd.exe` and regedit.exe executables are used to launch the Command Prompt and Registry Editor diagnostic tools.
		• **Location of application data**—The `Package.DAT` file contains all of the files, registry keys, DLLs, ThinApp components, and third party libraries that are required for the ThinApp application and the diagnostic tools to run.

ThinApp application files are saved in a directory named `ThinAppPackage`. The location of the `ThinAppPackage` directory depends upon the type of file you are editing in InstallShield:

- **InstallShield project**—The `ThinAppPackage` directory will be located in a subdirectory of the directory that contains this InstallShield project file, such as:

  `C:\InstallShield 2009 Projects\ProductName\ConfigurationName\ReleaseName\ThinAppPackage`

- **Windows Installer package**—The `ThinAppPackage` directory will be located in the same directory as the Windows Installer file, such as:

  `C:\FolderContainingMSI\ThinAppPackage`

Caution • *Modifying these files directly is **not recommended**. To make any modifications, use the InstallShield ThinApp Assistant.*

Intermediate Data Files: Interm Directory

When a ThinApp application is built, files that support the ThinApp application build process are extracted out of the Windows Installer package and saved in a subdirectory of the `ThinAppPackage` directory named the `Interm` directory.

Figure 11-26: Interm Subdirectory of the ThinAppPackage Directory

The data in this directory is then compiled into ThinApp application as part of the build process. The data in the `Interm` directory *does not* need to be distributed with the ThinApp application.

Supported InstallShield Project Types

The **VMware ThinApp** tab is available when one of the following InstallShield project types is open:

- Basic MSI Project
- MSI Database (Direct Edit Mode)
- Transform (Direct MST Mode)

How Transforms are Included in a ThinApp Application

The ThinApp Assistant supports the inclusion of transform files with Windows Installer packages in a ThinApp application.

- **How transforms are applied when building a ThinApp application**—When building a ThinApp application, transforms that you have specified are automatically applied to the base Windows Installer (`.msi`) package to create a temporary package, and then the ThinApp application is generated from that temporary package.
- **Creating a new transform**—You can create a new transform in InstallShield, and then build a ThinApp application from that transform file. When you create a new transform file in InstallShield, you specify the root `.msi` file in the **Open Transform** wizard. The steps you take to generate a ThinApp application after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.
- **Converting a Windows Installer package with existing transforms**—If you have a Windows Installer package and one or more existing transform files, and you want to include these transforms in the ThinApp application, you need to open one of the *transforms* in InstallShield (rather than the `.msi` file). The **Open Transform** wizard will open, and you will be prompted to specify the root `.msi` file and which of the existing `.mst` files you want to include. The steps you take to generate a ThinApp application after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.

Caution • *All of the transforms that you add to a ThinApp application must be located in the same folder as the Windows Installer .msi package so that they can be accessed when the ThinApp application is built.*

About Sandboxes

A ThinApp application runs in a *sandbox*, a virtual operating system—consisting of a virtual file system and a virtual registry—which is embedded with each ThinApp application. Running an application in a sandbox protects the local operating system from installation modifications that could affect stability or security. In a sandbox, system resources (such as files and registry keys) are redirected from the physical operating system files to the sandbox.

Many applications fail to run if the user does not have administrative rights because they expect to be able to write to global locations like `HKEY_LOCAL_MACHINE` and `C:\Program Files`. Using sandbox technology makes applications believe they have the ability to make global changes when they are actually writing to user and application-specific locations, and allows applications that require administrative rights to run without additional privileges. This feature allows ThinApp applications to run in security-restricted environments such as Windows Vista and Terminal Server.

What is a Sandbox Cache?

When you run a ThinApp application, additional files or registry keys may be produced. Depending upon the isolation options, some of this run time data will need to be stored locally in a sandbox cache, a local per-user directory.

When the ThinApp application is built, the *local Sandbox cache* is created in the following location, using the **Sandbox Name** that was entered on the **General Settings** page.

`c:\Documents & Settings\`*`USER_NAME`*`\Application Data\ThinApp\`*`SANDBOX_NAME`*

If the user's `Application Data` directory is stored on a network share, the ThinApp application's settings will automatically migrate when the same user logs in on another machine. You can also choose to create the sandbox cache in an external storage device such as a USB flash drive.

Using the ThinApp Assistant to Create a ThinApp Application

The steps you need to take to create a ThinApp application are the following:

Table 11-24 • Steps to Take to Create a ThinApp Application Using the ThinApp Assistant

Step #	Description
Step 1	Specifying ThinApp General Settings
Step 2	Managing Files and Folders in a ThinApp Application
Step 3	Setting ThinApp Isolation Options
Step 4	Modifying Shortcuts to the ThinApp Application's Executable Files
Step 5	Modifying ThinApp Application Registry Settings
Step 6	Modifying Build Options

Table 11-24 • Steps to Take to Create a ThinApp Application Using the ThinApp Assistant

Step #	Description
Step 7	Building a ThinApp Application

Specifying ThinApp General Settings

When creating a ThinApp application, you can, optionally, specify sandbox and Active Directory settings. You can also specify whether to include diagnostic tools with the ThinApp application. The following tasks are performed on the **General Settings** page of the **ThinApp Assistant**:

- Specifying Sandbox Information
- Specifying Control Access via Active Directory
- Prerequisites for Building a ThinApp Application
- Including Diagnostic Tools With a ThinApp Application

Specifying Sandbox Information

In this step, you have the option of entering a name for the Sandbox cache that is created when the ThinApp application is built.

Note • *For information on sandboxes and sandbox caches, see About Sandboxes.*

Task: ***To specify sandbox information:***

1. In the **ThinApp Assistant**, open the **General Settings** page.
2. When a ThinApp application is built, a Sandbox cache is created in the following location:

   `c:\Documents & Settings\`*`USER_NAME`*`\Application Data\ThinApp\`*`SANDBOX_NAME`*

   By default, AdminStudio names the Sandbox by assigning it a unique GUID. However, if you want to override this default Sandbox name, you may (optionally) enter a new name in the **Sandbox Name** field.

   Note • *The* ***Sandbox Name*** *you enter here is also recorded in the* `Package.ini` *file.*

3. If you want changes for Network mapped drives to be saved in the sandbox, then select the **Mapped Network Drive Changes go to Sandbox** option.
4. If you want changes for removable disks to be saved in the sandbox, then select the **Removable Disk Changes go to Sandbox** option.

5. If you want to delete the sandbox content when the ThinApp application exits, then select the **Reset Sandbox on Exit** option.

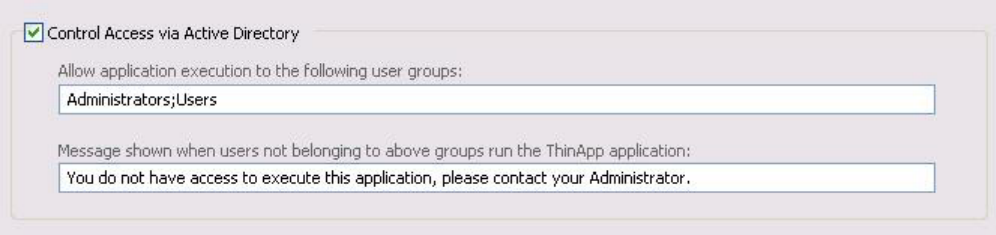
Specifying Control Access via Active Directory

You can control the access of users to a ThinApp application by specifying Active Directory groups on the **General Settings** page. At build-time, ThinApp assigns a unique GUID-like number to uniquely identify each Active Directory Group that you identify. Members of those groups will have access to the ThinApp application. For more detailed information about how Active Directory permissions are assigned, see About Controlling Access to ThinApp Applications.

To specify control access via Active Directory on the **General Settings** page, perform the following steps:

Task: ***To specify control access via Active Directory:***

1. In the **ThinApp Assistant**, open the **General Settings** page.
2. Select the **Control Access via Active Directory** option. The fields below are enabled.



3. In the **Allow application execution to the following user groups** field, enter the names of all of the Active Directory groups that you want to have permission to run this ThinApp application, separated by semi-colons, such as:

   ```
   GroupOne;GroupTwo;GroupThree
   ```

4. In the **Message shown when users not belonging to above groups run the ThinApp application**, enter the message that will be displayed when users that do not belong to the specified groups attempt to launch a ThinApp application.

Caution • *If you do not select the **Control Access via Active Directory** option, anyone who has access to a directory containing a ThinApp application will be able to run the application.*

About Controlling Access to ThinApp Applications

Note the following about controlling access to ThinApp applications via Active Directory:

- **You must be connected**—You must be connected to your Active Directory domain when you build the ThinApp application.

- **Groups must exist**—The Active Directory groups that you specify must exist when the ThinApp application is built.
- **If you delete a group and then recreate it, you must rebuild**—If you delete a group and recreate it, you will need to rebuild the ThinApp application in order to authenticate against the "new" group.
- **Offline users can authenticate using cached credentials**—When users are offline, they can authenticate using cached credentials. Assuming that the user can log into their laptop, ThinApp Active Directory authentication will still work.
- **Sometimes you may need to update credentials manually**—Cached credentials may not refresh on clients until the next Active Directory refresh cycle. To manually refresh the cached group policy credentials, you can use the `gpupdate` command. Sometimes the user may need to log-off before the credentials are recached.
- **"Administrators" and "Everyone" Groups use same credentials**—Special groups like Administrators and Everyone have the same SID on every Active Directory domain and Workgroup. Other groups you create will have a domain-specific SID, meaning a user cannot create their own local group with the same name to bypass authentication.

Prerequisites for Building a ThinApp Application

AdminStudio will convert the package installation into a format compatible with ThinApp. However, the ThinApp build process requires the availability of certain ThinApp tools.

As a prerequisite to building a ThinApp application from AdminStudio, you must have installed ThinApp and accepted any and all license agreements.

***Note** • For more information, see ThinApp on the VMware Web site.*

Including Diagnostic Tools With a ThinApp Application

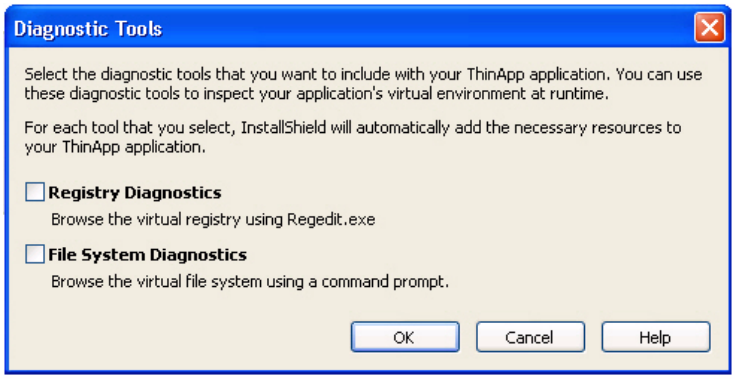
On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **General Settings** page, you can choose to include the Registry Editor and the Windows Command Prompt diagnostic tools with your ThinApp application.

If you include diagnostic tools with your ThinApp application, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running a ThinApp application and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

***Caution** • If you choose to include these diagnostic tools, the versions of `regedit.exe` and `cmd.exe` that are part of the operating system on the build machine are added to the ThinApp application. However, these tools may not be compatible with other operating systems.*

Task: ***To include diagnostic tools with a ThinApp application:***

1. In the **ThinApp Assistant**, open the **General Settings** page.
2. In the **More Options** list, click **Diagnostic Tools**. The **Diagnostic Tools** dialog box opens.



3. If you want to include the Registry Editor (regedit.exe) with your ThinApp application so that you can browse the registry at runtime from within the virtual environment, select the **Registry Diagnostics** option.
4. If you want to include the Windows Command Prompt application with your ThinApp application so that you can browse the virtual file system at runtime from within the virtual environment, select the **File System Diagnostics** option.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the ThinApp application.

When the user runs this ThinApp application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

Managing Files and Folders in a ThinApp Application

The next step in creating a ThinApp application is to view existing files and folders, add and delete files and folders, and override the default isolation options for folders.

The following tasks are performed on the **Files & Folders** page.

- Adding, Deleting, and Moving Files and Folders in an App-V Application
- Controlling the Display of Predefined Folders

Adding, Deleting, and Moving Files and Folders in a ThinApp Application

The directories in the destination tree on the **Files & Folders** page of the ThinApp Assistant represent how your application will look when it is installed on to your customer's machine.

On the **Files & Folders** page, you can view all of the files and folders that are currently in your ThinApp application, add new files and folders to include in the ThinApp application, and delete files and folders from the ThinApp application.

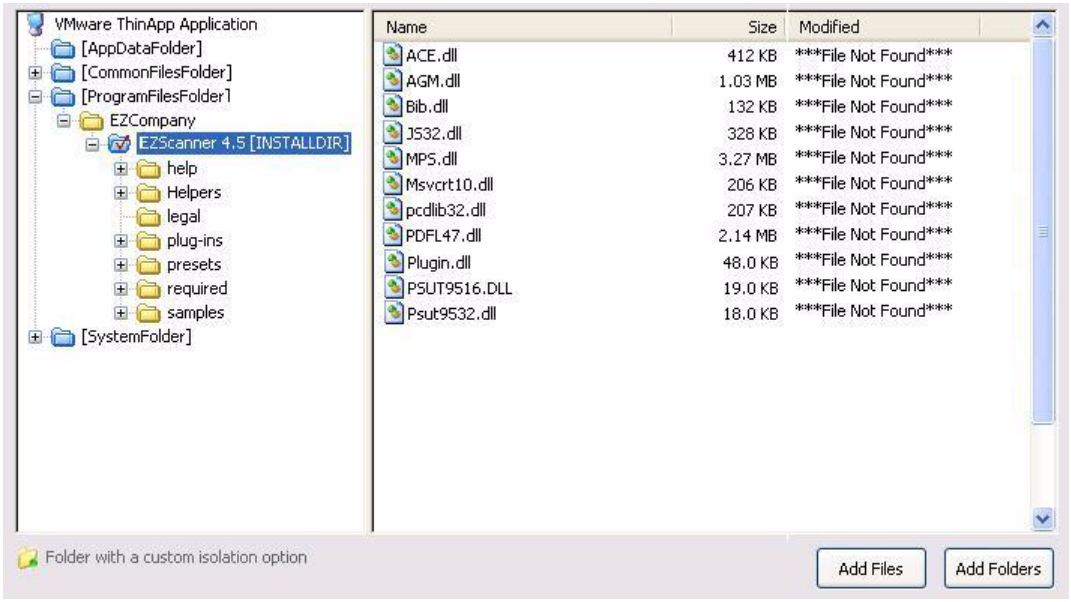
- Adding Files to a ThinApp Application
- Adding an Existing Folder (and its Contents) to a ThinApp Application
- Creating a New Folder
- Moving Files and Folders
- Deleting Files and Folders

Adding Files to a ThinApp Application

To add files to a ThinApp application, perform the following steps:

Task: ***To add a files to a ThinApp application:***

1. In the **ThinApp Assistant**, open the **Files & Folders** page. The files and folders are listed in the **VMware ThinApp Application** tree, organized by installation directory.



Folders are listed in the column on the left, and all of the files in the selected folder are listed on the right. Blue folders are the supported MSI standard folders. The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

2. Browse through the folder tree to find the folder that you would like to add files to.
3. Select the folder and click the **Add Files** button. The **Open** dialog box opens.
4. Select the file or files that you want to add and click **Open**. The files you selected are now listed.

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Adding a File by Dragging and Dropping Files From Your System

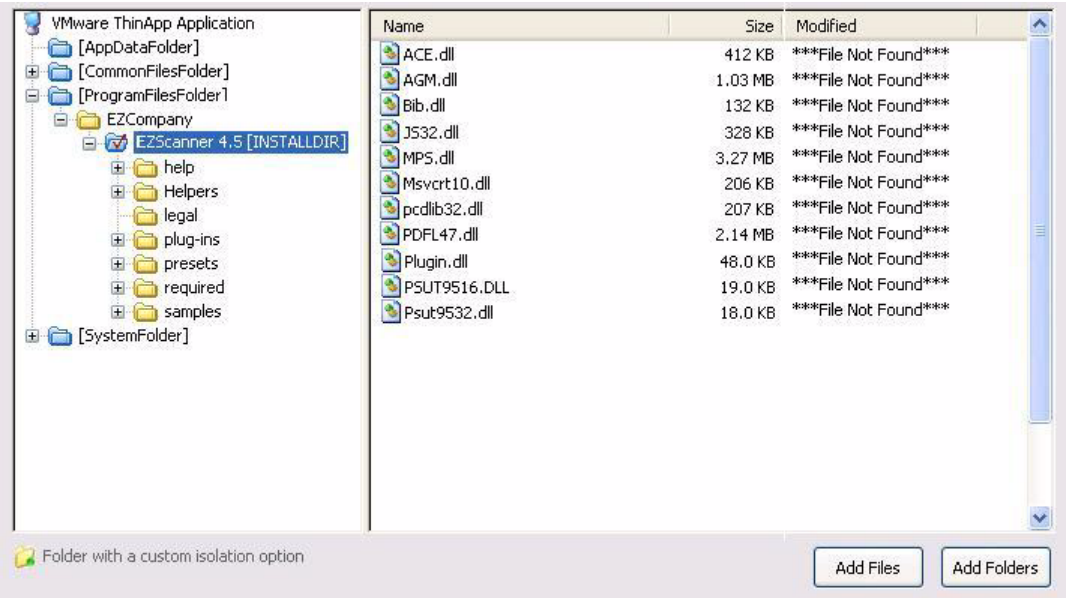
You can also add files or folders to your ThinApp application on the **Files & Folders** page by dragging them from a directory on your computer to the desired location in the tree.

Adding an Existing Folder (and its Contents) to a ThinApp Application

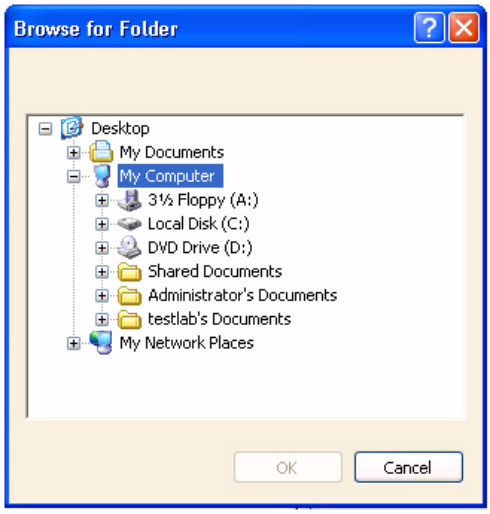
To add an existing folder and all of the files and subfolders within it to a ThinApp application, perform the following steps:

Task: ***To add an existing folder to a ThinApp application:***

1. In the **ThinApp Assistant**, open the **Files & Folders** page. The files and folders are listed in the **ThinApp Application** tree, organized by installation directory.

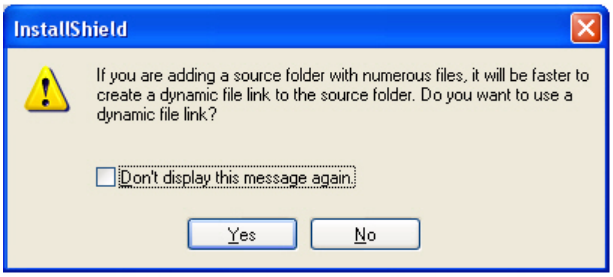


2. Browse through the folder tree to find the folder that you would like to add a folder into.
3. Select the folder and click the **Add Folders** button.The **Browse for Folder** dialog box opens, listing all of the directories available to your computer.



4. Select a folder and click **OK**.

   If you are editing an InstallShield project (not a Windows Installer package), you are prompted to choose whether you want to create a dynamic file link to the source folder.



5. Indicate whether you want to create a dynamic file link by selecting one of the following:

   - **No**—For more flexibility with ThinApp options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
   - **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

   The folder that you selected is now listed, along with of the files and folders within it.

Creating a New Folder

You can create a new, empty folder by selecting an existing folder in the tree and selecting **New Folder** from the context menu.

Task: ***To create a new folder:***

1. Right-click on a folder in the **VMware ThinApp Application** tree and select **New Folder**. A new folder is created as a subfolder of the selected folder:

2. Enter a name for the new folder.

Moving Files and Folders

To change the folder's location in the ThinApp application folder tree structure, perform the following steps:

Task: ***To move a file or folder:***

1. Select the file or folder that you want to move.
2. With the mouse button down, drag the file or folder to the new location.
3. Release the mouse button.

Deleting Files and Folders

To delete a file or a folder (and all of its contents) from a ThinApp application, perform the following steps:

Task: ***To delete a file or folder:***

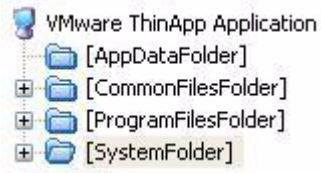
1. Select the file or folder in the **VMware ThinApp Application** tree that you want to delete.
2. Select **Delete** from the context menu. You are prompted to confirm the deletion.
3. Click **Yes**. The selected file or folder is deleted.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains from the entire Project, not just from the ThinApp application.*

Note • *You cannot delete predefined folders. You can only turn off the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Controlling the Display of Predefined Folders

On the **Files & Folders** page, the **VMware ThinApp Application** tree initially displays the more commonly used predefined folders, such as `[ProgramFilesFolder]` and `[CommonFilesFolder]`.

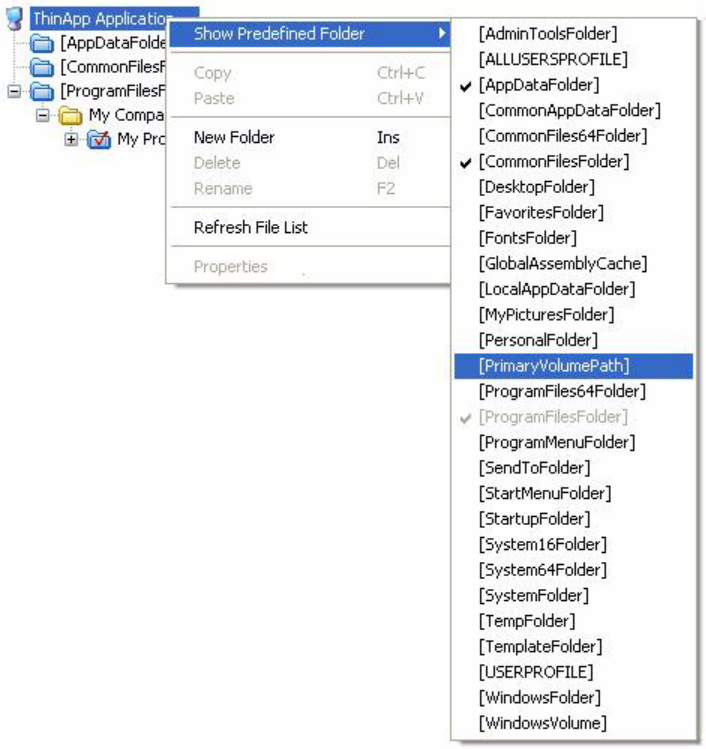


These predefined folders are dynamic, meaning that they do not use hard-coded paths. The value for each destination folder is obtained from the operating system of the target machine.

You can control which predefined folders are listed in this tree.

Task: ***To change which predefined folders are listed:***

1. In the **VMware ThinApp Application** tree, select the **ThinApp Application** node (or any of the files or folders that are listed, point to **Show Predefined Folder**. A list of predefined folders opens.



   Those folders that are already displayed are preceded by a check mark, and those that are not displayed do not have a check mark.

2. To add a folder to the tree listing, select a folder that is not currently listed in the tree.

Note • *These predefined folders are always added to the root of the* ***VMware ThinApp Application*** *tree, no matter what file or folder you had selected when you selected it from the Predefined Folders list.*

3. To remove a folder from the tree listing, select that folder name in this list (which is preceded by a check mark.

Note • *You cannot turn off the display of the* `[ProgramFilesFolder]`.

Setting ThinApp Isolation Options

ThinApp uses a sandbox virtual environment to control application compatibility and accessibility. The isolation option that is assigned to a folder or registry key specifies how the virtual environment will provide access to system resources requested by the application. You can use isolation options to control what an application can read and write on the local machine.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, in the ThinApp Assistant, you can override the default settings for selected folders or registry keys to exert control over application interactions with client operating system resources.

You set isolation options on the **Isolation Options** dialog box, which is open by selecting a folder or registry key and then selecting **Isolation Options** from the context menu.

Information about setting isolation options is presented in the following topics:

- Overview of ThinApp Isolation Options
- Setting Isolation Options for Folders
- Inheritance of Isolation Options from Folders to Files

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings.*

Overview of ThinApp Isolation Options

ThinApp uses virtual environments to control application compatibility and accessibility. The *isolation option* that is assigned to a folder or registry key specifies how the virtual environment will provide access to system resources requested by the application.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, in the ThinApp Assistant, you can override the default settings for selected folders or registry keys to exert control over application interactions with client operating system resources.

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings. Select the* ***Default*** *isolation option unless you require specific custom handling.*

You set isolation options on the **Isolation Options** dialog box, which is opened by selecting **Isolation Options** on the context menu when you have a folder selected on the **Files & Folders** page or a registry key selected on the **Registry** page.

Information about isolation options is presented in the following sections:

- Available ThinApp Isolation Options
- ThinApp Isolation Option Use Scenarios
- ThinApp Assistant Default Isolation Options

Available ThinApp Isolation Options

On the **Isolation Options** dialog box, you can choose one of the following isolation options:

Table 11-25 • ThinApp Isolation Options

Option	Visibility of System Elements	Modifications to Virtual Elements	Modifications to System Elements	New Elements	If System and Virtual Element at Same Location
Default	*As defined internally by the ThinApp Assistant*				
Write Copy	Visible	Sandbox	Sandbox	Created in Sandbox	Sees Virtual Element
Merged	Visible	Sandbox	System	Created in System	Sees Virtual Element
Full	Not Visible	Sandbox	N/A (System elements cannot be modified)	Created in Sandbox	N/A (System elements cannot be read)

ThinApp Isolation Option Use Scenarios

The following table describes scenarios where you would use each isolation option:

Table 11-26 • Use Scenarios for ThinApp Isolation Options

Option	Use Scenario
Write Copy	You would use **Write Copy** isolation when: • Application was not designed or tested for multi-user environments and expects it can modify files and keys without impacting other users. • Application expects write permission to Global locations and was not designed for locked-down desktop environments found in corporate environments or Windows Vista. With **Write Copy** isolation, ThinApp makes copies of registry keys and files written by the application and performs all of the modifications in a user-specific sandbox. With this type of isolation, the ThinApp applications believe that they have global write permissions, while they really only modify the sandbox directory.
Merged	You would use **Merged** isolation when the ThinApp application needs write access to user-specific storage areas, like the Desktop and My Documents.
Full	You would use **Full** isolation when a ThinApp application needs to run on a machine where earlier or later versions of the same application are either installed or were not uninstalled correctly. For directories and registry keys that have **Full** isolation, the ThinApp application will not be aware of any host computer file that might exist, and it sees only virtual files and registry keys at fully isolated locations.

ThinApp Assistant Default Isolation Options

If you do not set any isolation options on a folder or registry key in the ThinApp Assistant, the following default isolation options are applied:

Table 11-27 • ThinApp Assistant Default Isolation Options

Isolation Option	Condition
Write Copy Isolation	All other directories and subkeys associated with the product are assigned **Write Copy** isolation.
Merged Isolation	User-specific storage areas like the Desktop and My Documents, are set to Merged Isolation so that application has direct Write access to these locations

Note • *Network shares are not affected by isolation modes. Read and write operations to network shares occur unchanged by ThinApp.*

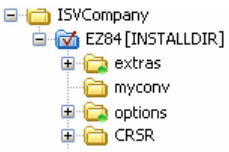
Note • *These default isolation options are built into the ThinApp Assistant.*

Setting Isolation Options for Folders

To override a folder's default isolation options, perform the following steps:

Task: ***To set an isolation option on a folder.***

1. Open the **Files & Folders** page.
2. Browse through the folder tree to find the folder that you would like to modify.
3. Select the folder and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.
4. Select one of the following options:
   - Default
   - Write Copy
   - Merged
   - Full
5. Click **OK**. Folders that have an isolation setting other than default are marked with a special icon:



Inheritance of Isolation Options from Folders to Files

Isolation options for files and subfolders are always inherited. The ThinApp virtual environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for `C:\Windows` and one for `C:\Windows\System32`. When the application requests `C:\Windows\System32\Notepad.exe`, then the `C:\Windows\System32` isolation rule will be applied because `C:\Windows\System32` is a more specific reference to `C:\Windows\System32\Notepad.exe` than is `C:\Windows`.

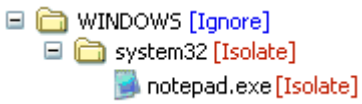


Figure 11-27: Example of Inheritance of Isolation Options from Folders to Files

Modifying Shortcuts to the ThinApp Application's Executable Files

You define application shortcuts to enable users to launch a ThinApp application from within the virtual environment.

By default, the **ThinApp Assistant** creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package). These shortcuts are listed in a checklist on the **Applications** page.

When you select each shortcut, details about it are displayed:

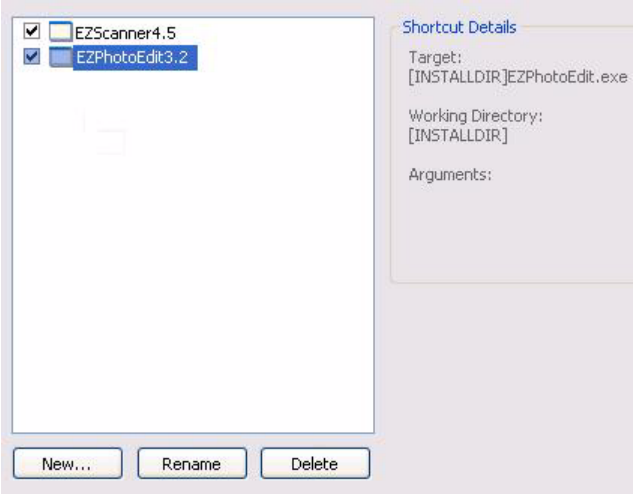


Figure 11-28: List of Shortcuts for an Application

Caution • *You must define at least one shortcut to enable users to launch the application from the isolation environment.*

On the **Applications** page, you can create, delete, include, exclude, or rename ThinApp application executables, which are derived from the shortcuts in its Windows Installer package.

- ThinApp Applications and the Virtual Environment
- Compressing a ThinApp Application
- ThinApp Shortcut Requirements
- Creating a New ThinApp Application
- Including an Existing ThinApp Application
- Excluding or Deleting an Existing ThinApp Application

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

ThinApp Applications and the Virtual Environment

On the **Applications** page of the ThinApp Assistant, you define application shortcuts to enable users to launch a ThinApp application from within the virtual environment. By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project.

To deploy a ThinApp application—on a local drive or a network share—systems administrators just need to give users access to the ThinApp application.

Compressing a ThinApp Application

A ThinApp application consists of either:

- **One executable file (.exe)**—This file is used to both launch the ThinApp application and also contain all of the data that is required for the application to run. In this scenario, this executable file would be a large file.

  or

- **Several executable files (.exe) and a Package.DAT file**—Each of the executables is used to launch the ThinApp application or a specific feature of the ThinApp application and `Package.DAT` contains all of the data that is required for the application to run. In this scenario. `Package.DAT` would be a large file.

Each time a user launches a ThinApp application, its data (from either the executable file or from `Package.DAT`) is read into the computer's memory. To reduce the application size, you can select a **Compression Type** on the **Build Options** page to compress all of the data.

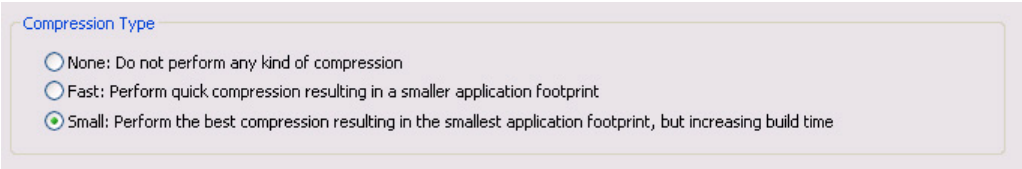


Figure 11-29: Compression Types on the Build Options Page

The following **Compression Types** are available:

Table 11-28 • ThinApp Compression Types

Type	Description
None	Do not perform any type of compression.
Fast	Perform quick compression resulting in a smaller application footprint.
Small	Perform the best compression resulting in the smallest application footprint, but increasing build time.

Application startup time is most effected by compression options used:

- **No compression**—Without compression enabled, startup speeds are comparable to normal application startup times.

- **Fast compression**—With fast compression options enabled, applications may startup faster than normal when the disk cache is empty and slightly slower than normal when the disk cache has been pre-filled, depending on processor speed and disk speeds.

You may also want to compress a ThinApp application to make it easier to distribute it throughout your organization.

When you perform compressed builds, large temporary files are saved in a cache location. To delete all of these temporary files, select the **Clear the VMware ThinApp Cache** option in the **More Options** list on the **Build Options** page

ThinApp Shortcut Requirements

For each ThinApp application, you are required to define **at least one** shortcut. You define application shortcuts to enable users to launch a ThinApp application from within the virtual environment. By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package).

If you build a ThinApp application that does not contain any shortcuts, users will not be able to launch the application.

Creating a New ThinApp Application

On the **Applications** page of the ThinApp Assistant, you specify the executables that you want to create ThinApp applications for.

Task: ***To create a new ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project (or Windows Installer package) are listed:
   - Those that are currently included in the ThinApp application are selected.
   - Those that are currently excluded from the ThinApp application are not selected.
2. Click **New**. The **Browse for a Shortcut Target File** dialog box opens and prompts you to select a file within this ThinApp application.
3. Select the file that you want to create a shortcut to.
4. Click **Open**. A new shortcut is listed, and it is named the same name as the selected file.
5. To include this shortcut in the ThinApp application, make sure that its check box is selected.

Including an Existing ThinApp Application

If you want to include a previously excluded shortcut in a ThinApp application, perform the following steps:

Task: ***To include an existing ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To include a previously excluded shortcut, select the shortcut and select the check box.

Excluding or Deleting an Existing ThinApp Application

By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package). These shortcuts are listed in a checklist on the **Applications** page.

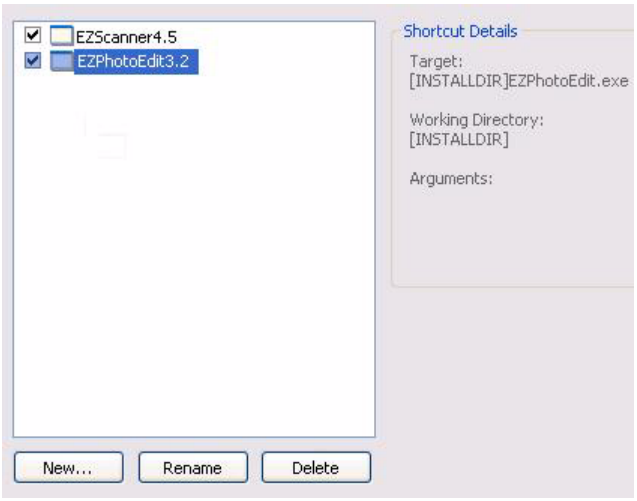


Figure 11-30: Initial List of Shortcuts for an Application

To prevent the shortcut from being created in the ThinApp application, you can choose to either delete or exclude it.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project. See Excluding a ThinApp Application.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the ThinApp application and the InstallShield project. See Deleting a ThinApp Application.

Caution • *If you delete a shortcut on the* ***Applications*** *page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

If you have any unnecessary shortcuts in your project, you can simply exclude them from the ThinApp application by unchecking them in the shortcuts list. If you like to permanently remove a shortcut, you can delete it from the shortcut list.

Excluding a ThinApp Application

If you want to exclude one of these shortcuts from being created in the ThinApp application, perform the following steps:

Task: ***To exclude a ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To exclude a shortcut, select the shortcut and clear the check box.

Note • *When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project.*

Deleting a ThinApp Application

To delete a ThinApp application, perform the following steps.

Task: ***To delete a ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
2. Select the shortcut and click **Delete**.

Caution • *If you delete a shortcut on the **Applications** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

Excluding vs. Deleting ThinApp Application Shortcuts

To prevent a shortcut from being created in the ThinApp application, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a ThinApp Application.
- **Deleting a shortcut**—When you delete a shortcut, it is removed from both the ThinApp application and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a ThinApp Application.

Renaming a ThinApp Application

To rename a ThinApp application, perform the following steps:

Task: ***To rename a ThinApp application:***

1. Open the **Applications** page. All of the shortcuts that exist in the project are listed.
2. Select the shortcut that you want to rename and click **Rename**. A box appears around the shortcut name, and the shortcut name becomes an editable field.
3. Enter a new name for the shortcut.

Modifying ThinApp Application Registry Settings

Using the **ThinApp Assistant**, you can view existing registry keys, values, and data, and add or delete registry items in your ThinApp application.

You can also override the default isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to system resources requested by the application.

Information about modifying registry settings on the **Registry** page includes the following topics:

- About the Windows Registry
- Adding or Deleting Registry Keys and Values
- Setting ThinApp Isolation Options on Registry Keys

About the Windows Registry

The Windows registry is a system-wide database that contains configuration information used by applications and the operating system. The registry stores all kinds of information, including the following:

- Application information such as company name, product name, and version number
- Path information that enables your application to run
- Uninstallation information that enables end users to uninstall the application easily without interfering with other applications on the system
- System-wide file associations for documents created by an application
- License information
- Default settings for application options such as window positions

Keys, Value Names, and Values

The registry consists of a set of keys arranged hierarchically under the My Computer explorer. Just under My Computer are several root keys. An installation can add keys and values to any root key of the registry. The root keys that are typically affected by installations are:

- HKEY_LOCAL_MACHINE
- HKEY_USERS
- HKEY_CURRENT_USER
- HKEY_CLASSES_ROOT

A key is a named location in the registry. A key can contain a subkey, a value name and value pair, and a default (unnamed) value. A value name and value pair is a two-part data structure under a key. The value name identifies a value for storage under a key, and the value is the actual data associated with a value name. When a value name is unspecified for a value, that value is the default value for that key. Each key can have only one default (unnamed) value.

Note that the terms key and subkey are relative. In the registry, a key that is below another key can be referred to as a subkey or as a key, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding or Deleting Registry Keys and Values

Editing the registry on the **Registry** page is performed much like it is performed on the InstallShield **Registry View**. See Editing the Registry.

Setting ThinApp Isolation Options on Registry Keys

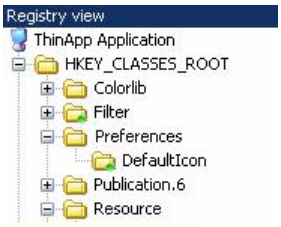
To override a registry key's default isolation options (which are built into the ThinApp Assistant), perform the following steps:

Task: ***To set an isolation option on a registry key:***

1. Open the **Registry** page.
2. Browse through the registry tree to find the key that you would like to modify.
3. Select the folder or key and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.

   Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

4. Select one of the following options:
   - Default
   - Write Copy
   - Merged
   - Full
5. Click **OK**. Registry keys that have an isolation setting other than default are marked with a special icon:



Tip • *To import an existing registry (`.reg`) file, click the **Import a .reg file** option on the **More Options** list to open the Registry Import Wizard.*

Inheritance of ThinApp Isolation Options in the Registry

Isolation options for registry keys are always inherited. The ThinApp virtual environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for the `Microsoft` registry key and one for `Microsoft\Windows` registry key. When the application requests `Microsoft\Windows\CurrentVersion`, then the `Microsoft\Windows` isolation rule will be applied because `Microsoft\Windows` is a more specific reference to `Microsoft\Windows\CurrentVersion` than is `Microsoft`.

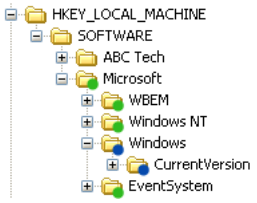


Figure 11-31: Example of Inheritance of Isolation Options from Folders to Files

Modifying Build Options

On the **Build Options** page, you choose which releases of this InstallShield project you want to build a ThinApp application for when the project is built, specify the type of compression, disable the ThinApp Log Monitor tracing capabilities, and specify whether you want to include additional Windows Installer packages in the virtual package.

Also, if you are editing a Windows Installer package in Direct Edit mode (or Direct MST mode), you need to select the **Build ThinApp Application** option on the **Build Options** page before you will be able to build a ThinApp application for that Windows Installer package.

- Selecting Releases to Build
- Enabling ThinApp Application Building When in Direct Edit Mode
- Including Additional Windows Installer Packages in a ThinApp Application
- Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application
- Setting ThinApp Log Monitor Tracing Options

- Setting AppLink Options
- Setting AppSync Options

Important • *You must create at least one Release (on the **Releases** view of the Installation Designer) before you will be able to select a Release on the **Build Options** page.*

Selecting Releases to Build

You select the releases that you want to build a ThinApp application for on the **Releases** tree of the **Build Options** page.

Important • *You cannot create or edit a release in the ThinApp Assistant. If no releases exist, you can simply click the **Build** toolbar button to create a new release or open the **Releases** view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a ThinApp application. For more information, see Creating and Building Releases.*

*If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Options** page is not displayed.*

Task: ***To select releases to build:***

1. Open the **Build Options** page.
2. Select the releases in the **Releases** tree that you want to build a ThinApp application for.

Important • *When you select a release on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build a ThinApp application for that release. However, the releases that are selected on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Options** page:*

- ***Active release selected**—A Windows Installer package and a ThinApp application would be built.*
- ***Active release not selected**—Only a Windows Installer package would be built.*

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

Enabling ThinApp Application Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **ThinApp Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a ThinApp application for this Windows Installer package. To do this, perform the following steps:

Task: ***To enable ThinApp application building when in Direct Edit Mode:***

1. Open a Windows Installer package or a transform file in InstallShield. It will be opened in Direct Edit Mode or Direct MST Mode, and the Build function (**Build** on the **Build** menu and the **Build** toolbar button) will be disabled.
2. Open the **Build Options** page of the ThinApp Assistant.
3. Select the **Build ThinApp application** option. After you select this option, the **Build ThinApp application** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Including Additional Windows Installer Packages in a ThinApp Application

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the ThinApp Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in a ThinApp application, set the **Would you like to include additional MSI files in the virtual package?** option on the **Build Options** page to **Yes**, and then select the packages that you want to add.

Task: ***To include additional Windows installer packages in a ThinApp application:***

1. Open the **Build Options** page.
2. Set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**.
3. Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
   - The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
   - Use the Delete button () to delete a package from the list.

Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application

You can choose to build a Windows Installer package to assist in the distribution of a ThinApp application. This simplifies the deployment of a ThinApp application by enabling you to use enterprise distribution tools such as Microsoft System Center Configuration Manager or Novell ZENworks Configuration Management.

To build a Windows Installer file with your ThinApp application, select the **Generate a Windows Installer (MSI) file as part of the build output** option on the **Build Options** page of the ThinApp Assistant. By default, this option is not selected.

This Windows Installer file can be run to properly install the ThinApp application on an end-user's desktop. A ThinApp application installed using a Windows Installer package can be uninstalled using **Add or Remove Programs** in the Control Panel.

Setting ThinApp Log Monitor Tracing Options

ThinApp Log Monitor is an application in the ThinApp Suite that allows you to record detailed information about any application's execution history for later review. The following events are recorded:

- **API calls**—Win32 API calls with parameter and result information made by applications running in the ThinApp virtual operating system
- **Errors**—A list of potential errors, exceptions, and security events within the application
- **Loaded DLLs**—A list of all DLLs loaded by the application and address ranges

Log Monitor is launched by selecting a shortcut in the ThinApp Suite group on the Windows Start menu.

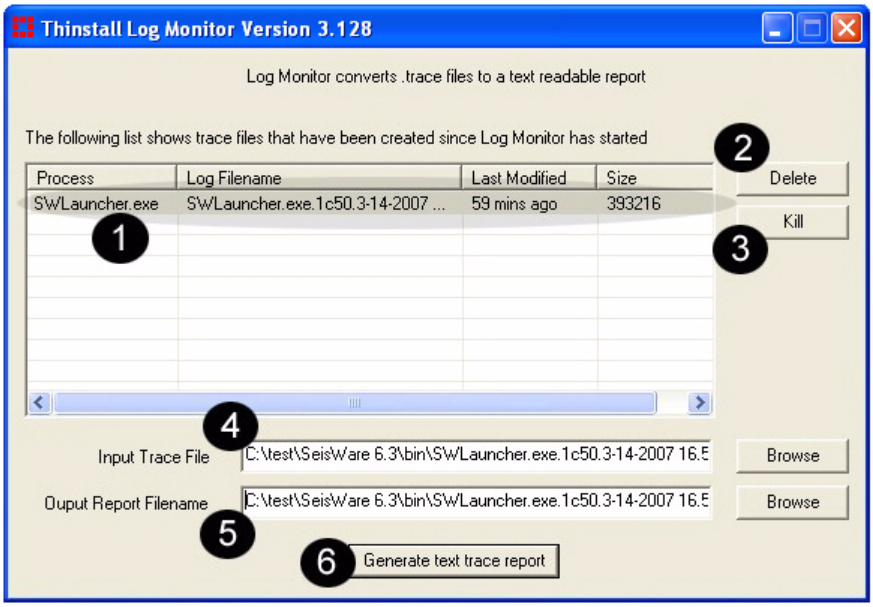


Figure 11-32: ThinApp Log Monitor

Log Monitor displays the following information:

Table 11-29 • ThinApp Log Monitor Interface

#	Name	Description
1	**Process List**	Any new ThinApp process which has been started after Log Monitor begins will be listed. If you click on one of the processes, the **Input Trace File** and **Output Report Filename** fields are automatically populated. ***Note*** • *If the application was built with the **Disable Log Monitor Tracing** option on the **Build Options** page selected, it will not be listed.*
2	**Delete**	Click to delete trace files for the selected processes in the Process List.
3	**Kill**	Click to kill currently running process that is selected in the Process List. You would do this to stop a process from logging additional entries once an error condition has been reached.
4	**Input Trace File**	Click Browse to manually browse for a trace file to convert.
5	**Output Report File**	The file listed in this field is generated when you click **Generate text trace report**. This report should be viewed with a text editor that supports UNIX-style line breaks such as Wordpad or Word (not Notepad).

Disabling Log Monitor Tracing

If you do not want to allow ThinApp Log Monitor tracing in a ThinApp application, select the **Disable Log Monitor Tracing** option on the **Build Options** page.

Task: ***To disabling the ThinApp Log Monitor tracing capabilities:***

1. Open the **Build Options** page.
2. Select the **Disable Log Monitor Tracing** option.
3. Build the ThinApp application.

Setting AppLink Options

Note • *The AppLink Settings feature requires ThinApp 4.x. If you are using Thinstall 3.x, any AppLink settings that you define will be ignored.*

The AppLink (Application Link) feature enables you to configure relationships between ThinApp applications that work together. You can set AppLink settings for the current ThinApp application on the **AppLink Settings** dialog box, which is opened by clicking the **AppLink Settings** option in the **More Options** menu of the ThinApp Assistant **Build Options** page.

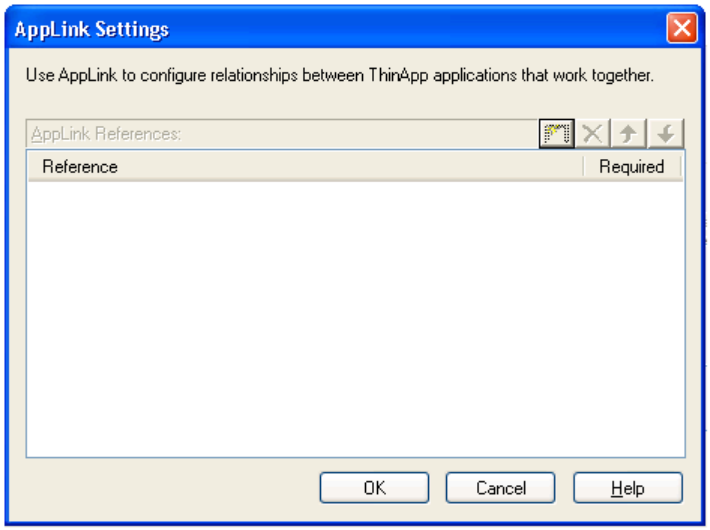
You can use the AppLink feature to perform the following tasks:

- **Linking runtime components to applications**—You can link runtime components to the applications that use them. For example, you can link a package containing the Java runtime environment (JRE) or ODBC drivers to a package containing a browser application.
- **Linking add-ons and plug-ins to applications**—You can link add-ons and plug-ins to applications. For example, Microsoft Office add-ons can be linked to applications or Adobe Photoshop plug-ins can be linked to a package containing Photoshop.
- **Linking packaged applications to service packs**—You can link packaged applications to service packs. By using AppLink, you can upgrade or roll back your service packs by changing the service pack that you capture and link to its parent application.

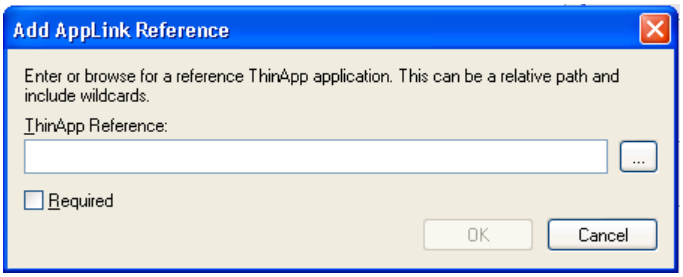
To set AppLink options for a ThinApp application, perform the following steps.

Task: ***To configure AppLink settings for your ThinApp application:***

1. On the **Build Settings** page of the ThinApp Assistant, click the **AppLink Settings** option in the **More Options** menu. The **AppLink Settings** dialog box opens.



2. Click the Browse button to open the **Add AppLink Reference** dialog box.



3. In the **ThinApp Reference** box, enter the relative (runtime) path to the existing ThinApp application that you want to link to.

   - If you want to add multiple applications, repeat the procedure as necessary.
   - You can also use wild cards. See Security and Authorization
   - The order in which packages are imported can be changed by selecting a package and clicking the up and down arrows. See Collisions and Order of Import for more information on order.

   Note • *Required and Optional links are listed on the AppLink Settings dialog box together and the order can be changed using the up and down arrows. However, at runtime, all of the applications in the Required category are read first, before those in the Optional category, even if applications in the Optional category were originally higher in the list. When the AppLink Settings dialog box is reopened, the AppLink References will be grouped by category rather than be in the order that was arranged prior to closing the dialog box. In other words, the category order (Required and Optional) overrides the order set by the user.*

   - To delete a package you have added, select the package and click the Delete ( ) button.

Important • *When linking to a ThinApp application that has only one shortcut, select its .EXE file. When linking to a ThinApp application that has more than one shortcut, select either its Package.DAT file (if the ThinApp application was built with AdminStudio) or its primary executable file (if the ThinApp application was built with ThinApp).*

Important • *On the* ***Add AppLink Reference*** *dialog box, if you click Browse and browse for a ThinApp application, the absolute path to that application is entered, such as* `C:\Program Files\AppName\filename.exe`. *In that case, the parent ThinApp application needs that linked application to be found at the specified absolute path location at runtime, which is unlikely. Therefore, it is recommended that you enter a relative path name.*

4. If you want this package to be required, select the **Required** option. If a required package is missing from the virtual package, it will fail to run. Note the following about required packages:

   - If any specified package fails to import, an error message will be displayed and the parent executable file will exit.

- If a wildcard pattern is used to specify a package, no error message is displayed if no files match the wildcard pattern. Therefore, if a wildcard pattern is used to specify a package, the reference is always optional.
- To continue even if load errors occur, make the package references optional instead.

5. Click **OK** to return to the **AppLink References** dialog box. The item you selected is now listed in the **AppLink References** list.
6. Click **OK** to return to the **Build Options** page.

Setting AppSync Options

***Note** • The AppSync Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppSync settings that you define will be ignored.*

AppSync (Application Sync) enables you to automatically keep deployed ThinApp applications up to date. When an application starts up, AppSync can query a Web server to see if an updated version of the package is available. If an update is available, the differences between the existing package and the new package will be downloaded and used to construct an updated version of the package. The updated package will be used for future deployments.

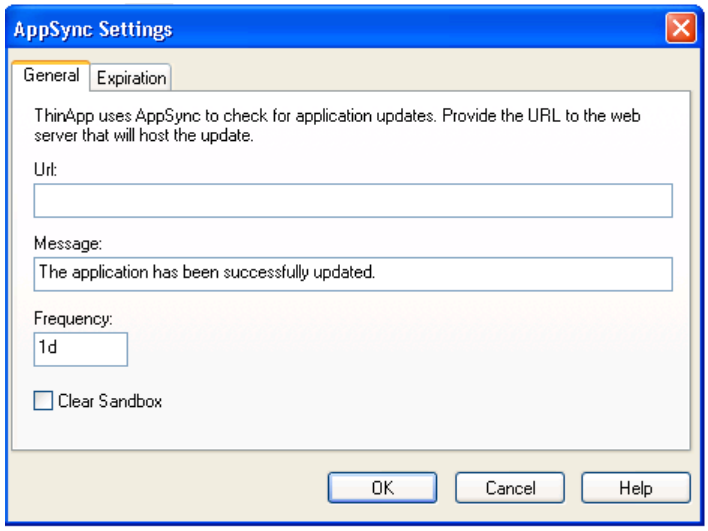
You can use the AppSync feature to perform the following tasks:

- **Distribute runtime components separately**—You can use AppSync to distribute runtime components separately from the applications that use them. For example, the Java Runtime Environment (JRE) or ODBC drivers.
- **Apply layered service packs to applications**—You can use AppSync to apply layered service packs to your applications. Application Sync enables you to distribute service packs and roll back to previous versions, if necessary.

On the AppSync Settings dialog box, you specify the location of the update, the message displayed to the user, and the expiration settings. You set AppSync settings for the current ThinApp application on the **Build Options** page of the ThinApp Assistant. To configure AppSync settings for a ThinApp application, perform the following steps:

Task: ***To configure AppSync settings for your ThinApp application,***

1. On the **Build Settings** page, click the **AppSync Settings** option in the **More Options** menu. The **General** tab of the **AppSync Settings** dialog box opens.



2. In the **Url** field, specify the location of the Web server that hosts application updates. When entering the URL, consider the following:

   - **Supports HTTP and HTTPS**—Application Sync works over both the HTTP (unsecure) and HTTPS (secure) protocol.
   - **Can include login information**—You can include a user name and password in the URL that will be used for basic authentication. The standard Windows/Internet Explorer proxy setting is respected. For example:

     ```
     https://www.example.com/some/path/PackageName.exe
     ```

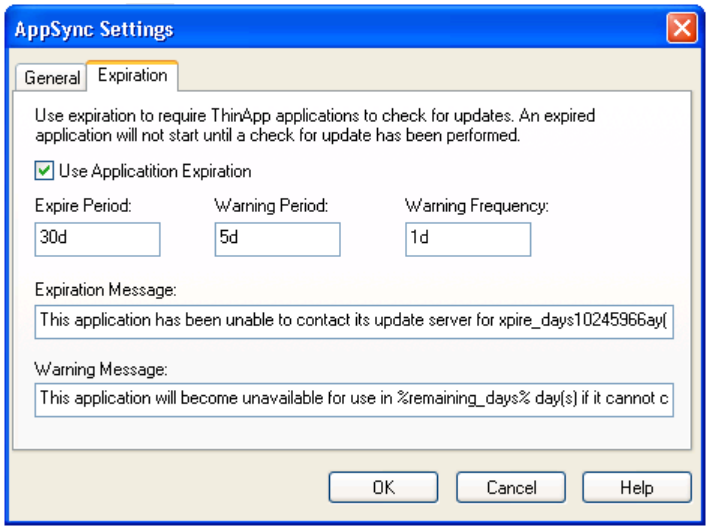
3. In the **Message** field, enter the information you want to display to the user when the ThinApp application is updated. By default, the following is entered:

   ```
   The application has been successfully updated.
   ```

4. By default, a package will connect to the Web server once per day to see if an updated version is available. You can set the frequency by modifying the **Frequency** setting. For example, to set the **Frequency** to 2 days, enter `2d`. For 2 weeks, enter `2w`, etc.

5. If you want to automatically clear the sandbox after an update, select the **Clear Sandbox** option. By default, this option is not selected.

6. Click **Expiration** to open the **Expiration** tab. On this tab, you can specify that a ThinApp application is required to check for updates at a defined frequency. If the ThinApp application fails to successfully check for updates within that defined frequency, it will fail to run.



7. To require that an application has to check for updates at a specified frequency, select the **Use Application Expiration** option

8. In the **Expire Period** box, enter the update frequency in minutes (`m`), hours (`h`), or days (`d`). For example:

   - To set the period to 30 days, enter `30d`.
   - To set the period to 72 hours, enter `72h`.
   - If you do not want the package to expire, clear the **Use Application Expiration** check box.

   If the Web server cannot be reached, meaning that the update fails, the package will continue to work until the **Expire Period** is reached. The default setting is 30 days.

9. In the **Warning Period** box, enter the amount of time prior to expiration that the user is first warned. For example, to set the period at 5 days, enter `5d`.

10. In the **Warning Frequency** box , enter the frequency that a warning message will be displayed to the user before the package expires. With the default of one day, the warning message will be displayed once per day only. To configure the warning to pop up on every application launch, enter `0`. To configure it to pop up every 4 days, enter `4d`.

    Note the following about warning frequency:

    - After the warning period has started, the Web server will be checked on every launch of an application, overriding any previous setting.
    - As long as a package has not expired, this parameter checks for new versions and downloads will occur in the background. The user can continue to use the old version.
    - If the application is terminated by the user before the download is complete, the download will resume when a virtual application is launched again. After the download completes, the new version will be activated on the next launch.
    - When the package has expired, the version check and download will happen in the foreground. A progress bar will be shown during the download phase.

11. Before the expiration limit has been reached and a ThinApp application is started, it will try to connect to the Web server and check for a new version. If the connection fails, a message box will be shown. The default message is:

    ```
    This application will become unavailable for use in Warning_Period days if it cannot contact its
        update server. Check your network connection to ensure uninterrupted service
    ```

12. After the expiration limit has been reached and a ThinApp application is started, it will try to connect to the Web server and check for a new version. If the connection fails, the message entered in the **Expiration Message** box will be shown and execution will be terminated. The default message is:

    ```
    This application has been unable to contact its update server for Expire_Period days, so it is
        unavailable for use. Check your network connection and try again.
    ```

Note • *If you use AppSync, VMware recommends that you disable automatic application updates that are configured in your virtual application. Conflicts might occur between the linked packages and the software that is automatically updated. If an automatic update feature updates an application, it stores the updates in the sandbox. If AppSync then updates the application to a different version, the updates stored in the sandbox take precedence over the files contained in the version that AppSync created. The order of precedence for the update files are those in the sandbox, then the virtual operating system, and then the physical machine.*

Building a ThinApp Application

The method for building a ThinApp application depends upon what file you have open—an InstallShield project or a Windows Installer package.

- Building a ThinApp Application for an InstallShield Project
- Building a ThinApp Application for a Windows Installer Package

Building a ThinApp Application for an InstallShield Project

To build a ThinApp application for an InstallShield project, perform the following steps:

Task: ***To build a ThinApp application for an InstallShield project:***

1. Open the InstallShield project in InstallShield.
2. On the **Releases** view of the Installation Designer, make sure that at least one release has been created, and select the release that you want to build.

Important • *You cannot create or edit a release in the ThinApp Assistant. If no releases exist, or if you want to create a new release, open the* ***Releases*** *view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a ThinApp application. For more information, see Creating and Building Releases.*

3. Open the **Build Options** page of the ThinApp Assistant.
4. In the **Releases** tree, select the same release that is selected on the **Releases** view of the InstallShield Installation Designer. This is the release that you will build a ThinApp application for.

   Important • *When you select a release on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build a ThinApp application for that release. However, the releases that are selected on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what was selected on the **Build Options** page:*

   - ***Active release selected**—A Windows Installer package and a ThinApp application would be built.*
   - ***Active release not selected**—Only a Windows Installer package would be built.*

   To build more than one release at a time, perform a batch build. See Performing Batch Builds.
5. Click the **Build** toolbar button (or select **Build Release** on the **Build** menu) to start building the active release.

   The output of the build will be a Windows Installer package and a ThinApp application. For information on the files included in a ThinApp application, see Components of a ThinApp Application.

Building a ThinApp Application for a Windows Installer Package

To build a ThinApp application for a Windows Installer package, perform the following steps:

Task: ***To build a ThinApp application for a Windows Installer package:***

1. Do one of the following to open a Windows Installer package:
   - On the **File** menu, select **Open** and select a Windows Installer package (`.msi`).
   - On the **File** menu, select **Open** and select a transform file (`.mst`). The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
   - On the **File** menu, select **New** to open the **New Project** dialog box. Select **Transform** and click **OK**. The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
2. Use the Installation Designer to make any desired edits to the Windows Installer package or Transform file, and use the ThinApp Assistant to set ThinApp application options.
3. Save the edits to the Windows Installer package or transform file by selecting **Save** on the **File** menu.
4. On the **Build Options** page of the ThinApp Assistant, select the **Build ThinApp application** option. The **Build Virtual Package button is enabled.**
5. Click the **Build Virtual Package button** to start building the ThinApp application.

The output of the build will be a ThinApp application. For information on the files includes in a ThinApp application, see Components of a ThinApp Application.

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

ThinApp Assistant Reference

Reference information about the ThinApp Assistant is organized into the following sections:

- Pages
- Dialog Boxes
- Building ThinApp Applications Using the Command Line
- ThinApp Application Conversion Error and Warning Messages
- Application Features Requiring Pre- or Post-Conversion Actions

Pages

The ThinApp Assistant is comprised of the following pages:

- ThinApp Assistant Home Page
- General Settings Page
- Files & Folders Page
- Applications Page
- Registry Page
- Build Options Page

ThinApp Assistant Home Page

The ThinApp Assistant Home page displays a diagram that illustrates the process of creating a ThinApp application.

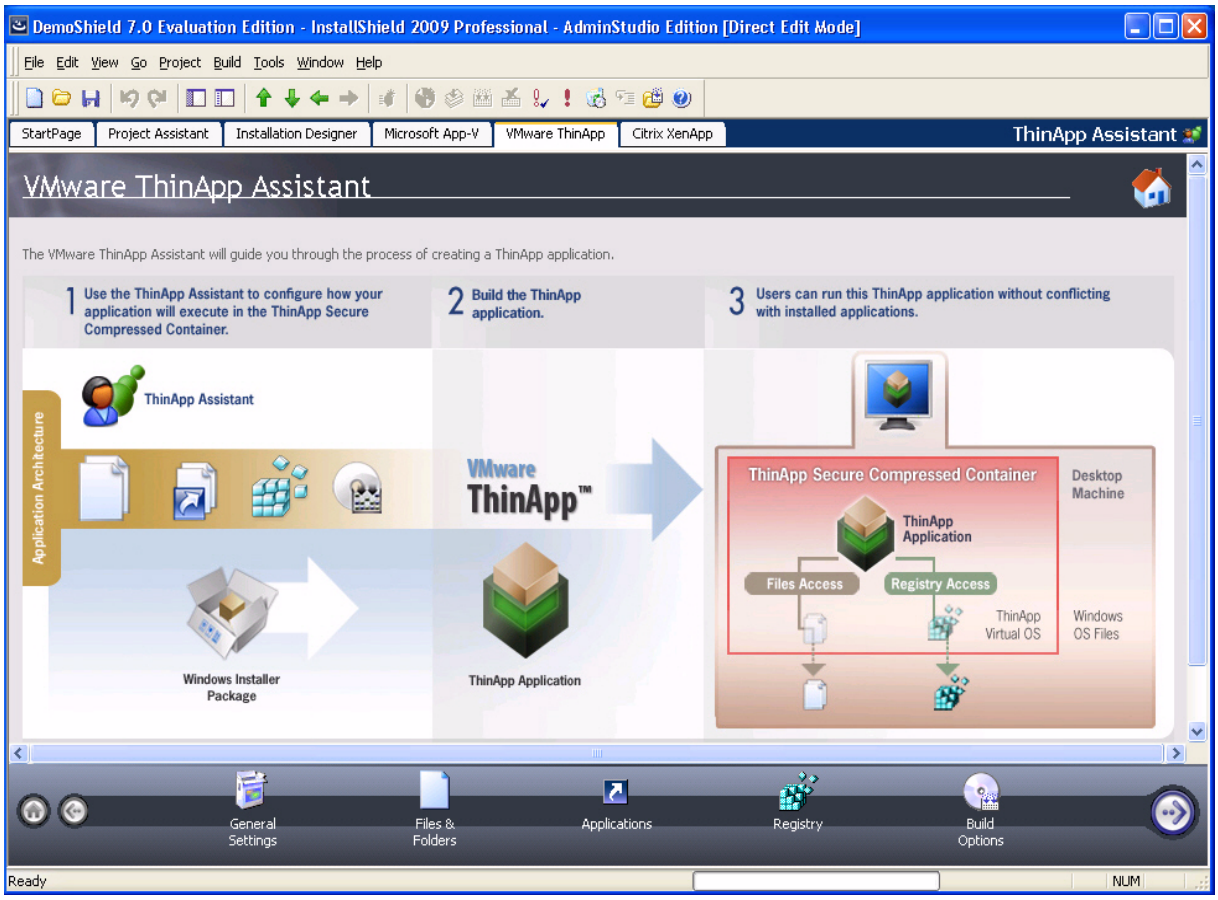


Figure 11-33: ThinApp Assistant Home Page

Click the following icons in the navigation bar at the bottom of the page to navigate through the ThinApp Assistant interface:

Table 11-30 • Navigation Bar Icons

Icon	Destination
	General Settings Page
	Files & Folders Page
	Applications Page
	Registry Page

Table 11-30 • Navigation Bar Icons

Icon	Destination
	Build Options Page
	Go to next page.
	Jump back to previous page.
	ThinApp Assistant Home Page

General Settings Page

On the **General Settings** page in the **ThinApp Assistant**, you specify Sandbox options, including options to control access to the ThinApp application using Active Directory.

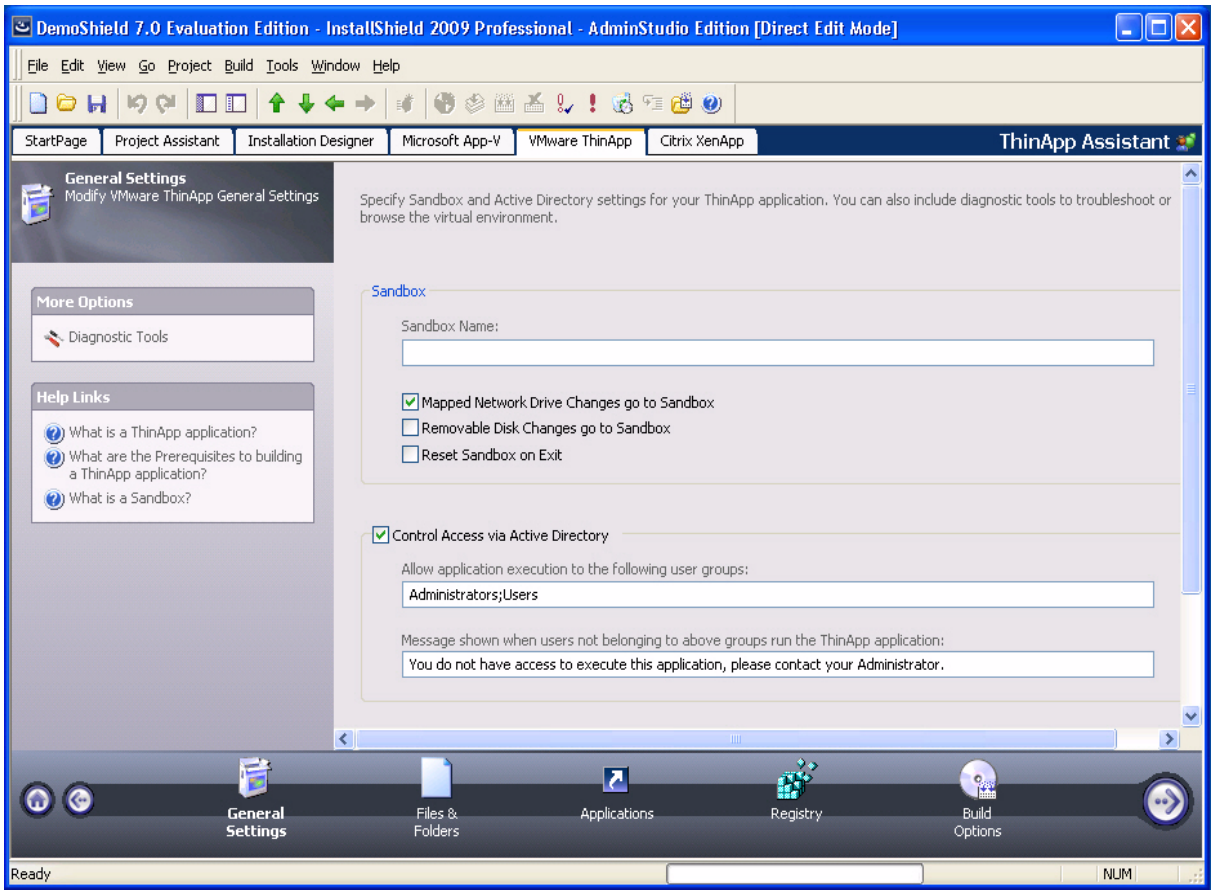


Figure 11-34: ThinApp Assistant General Settings Page

The **General Settings** page includes the following options:

Table 11-31 • General Settings Page

Option	Description
Sandbox Name	When a ThinApp application is built, a Sandbox cache is created in the following location: `c:\Documents & Settings\USER_NAME\Application Data\ThinApp\SANDBOX_NAME` By default, AdminStudio names the Sandbox by assigning it a unique GUID. However, if you want to override this default Sandbox name, you may (optionally) enter a new name in the **Sandbox Name** field.
Mapped Network Drive Changes go to Sandbox	Enable this option if you want changes for Network mapped drives to be saved in the sandbox. By default, users can read and write normally to network mapped drives.

Table 11-31 • General Settings Page (cont.)

Option	Description
Removable Disk Changes go to Sandbox	Enable this option if you want changes for removable disks to be saved in the sandbox. By default users can read and write normally to removable disks.
Reset Sandbox on Exit	Select this option to delete the sandbox content when the application exits. This resets the ThinApp application to its original captured state.
Control Access via Active Directory	If you want to control the access of users to a ThinApp application by specifying Active Directory groups, select this option and enter the names of those groups. At build-time, ThinApp would then assign a unique GUID-like number to uniquely identify each Active Directory Group that you have identified. Members of those groups will have access to the ThinApp application. • **Allow application execution to the following user groups**—Enter the names of all of the Active Directory groups that you want to have permission to run this ThinApp application, separated by semi-colons, such as: `GroupOne;GroupTwo;GroupThree` • **Message shown when users not belonging to above groups run the ThinApp application**—Enter the message that will be displayed when users that do not belong to the specified groups attempt to launch a ThinApp application. ***Caution*** • *If you do not select the **Control Access via Active Directory** option, anyone who has access to a directory containing a ThinApp application will be able to run the application.* ***Note*** • *For more information, see About Controlling Access to ThinApp Applications.*

For testing purposes, you can also choose to include diagnostic tools in your ThinApp application by selecting the **Diagnostic Tools** link in the **More Options** list. For more information, see ThinApp Diagnostic Tools Dialog Box.

Files & Folders Page

On the **Files & Folders** page of the ThinApp Assistant, you can perform the following tasks:

- View Files and Folders
- Add Files and Folders
- Delete Files and Folders
- Set Isolation Options

- Modifying the Display of Predefined Folders

View Files and Folders

On the **Files & Folders** page, you can view all of the files and folders that are currently in your ThinApp application.

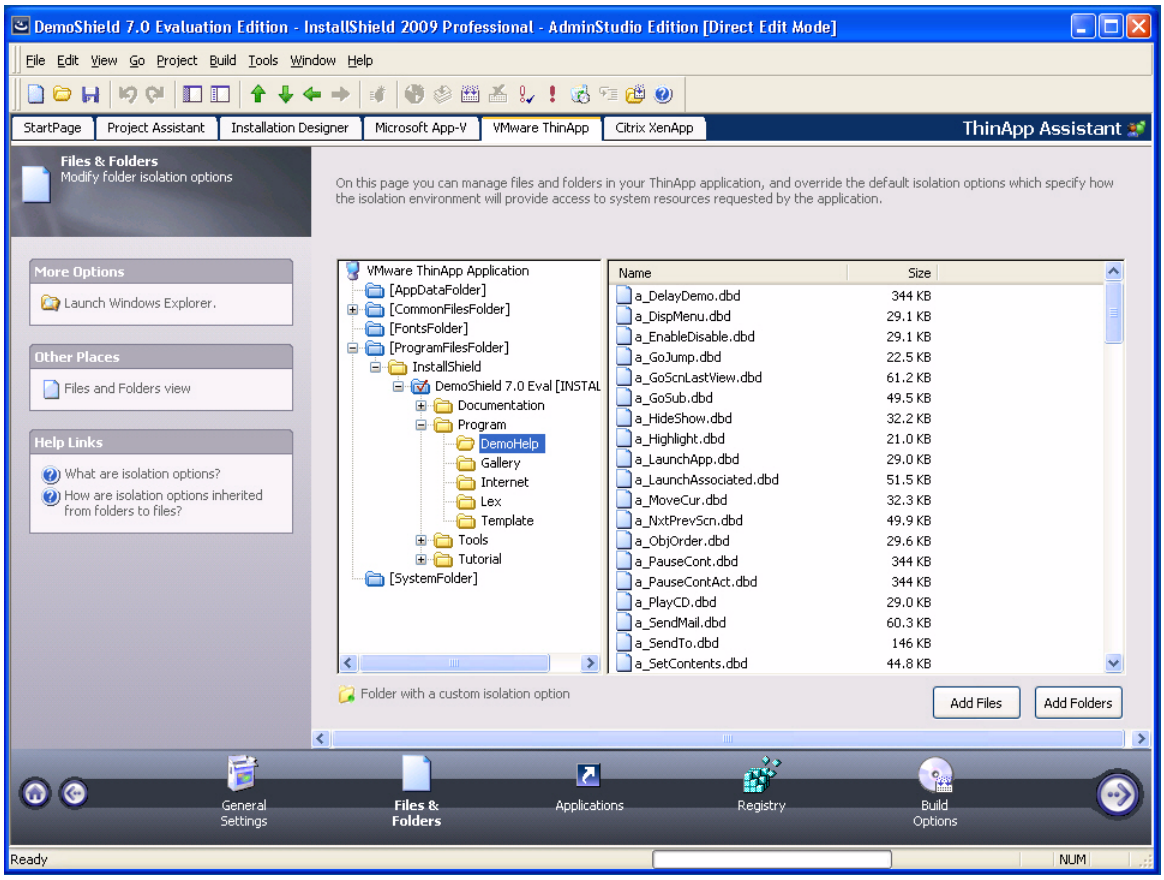


Figure 11-35: ThinApp Assistant Files & Folders Page

Folders are listed in the **VMware ThinApp Application** tree on the left, and all of the files in the selected folder are listed on the right.

- The directories in the tree represent how your application will be organized within its secure compressed container.
- Blue folders are the supported MSI standard folders.
- The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

Add Files and Folders

On the **Files & Folders** page, you can use the **Add Files** and **Add Folders** buttons to add new files and folders to include in the ThinApp application. See Adding, Deleting, and Moving Files and Folders in a ThinApp Application.

If you are editing an InstallShield project (not a Windows Installer package), and you are adding a folder to this ThinApp application, you are prompted to choose whether you want to create a dynamic file link to the source folder.

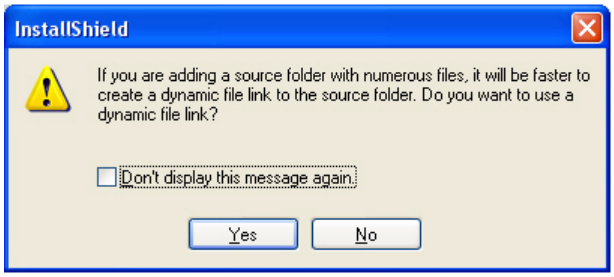


Figure 11-36: Dynamic File Link Option Dialog Box

Indicate whether you want to create a dynamic file link by selecting one of the following:

- **No**—For more flexibility with ThinApp options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
- **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

Delete Files and Folders

You can delete files and folders from the ThinApp application by selecting the file or folder you want to delete, and selecting **Delete** from the context menu. For more information, see Deleting Files and Folders.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains.*

Note • *You cannot delete predefined folders. You can only turn of the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Set Isolation Options

ThinApp uses a sandbox virtual environment to control application compatibility and accessibility. The isolation option that is assigned to a folder or registry key specifies how the virtual environment will provide access to system resources requested by the application.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options by selecting a file or folder and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting ThinApp Isolation Options.

Modifying the Display of Predefined Folders

You can specify which of the Windows Installer predefined folders are listed in the **VMware ThinApp Application** tree. See Controlling the Display of Predefined Folders.

Applications Page

You define shortcuts to enable users to launch a ThinApp application from within the sandbox virtual environment.

By default, the **ThinApp Assistant** creates ThinApp applications for all of the executable shortcut that exist in your project. The project's shortcuts are listed in a checklist on the **Applications** page.

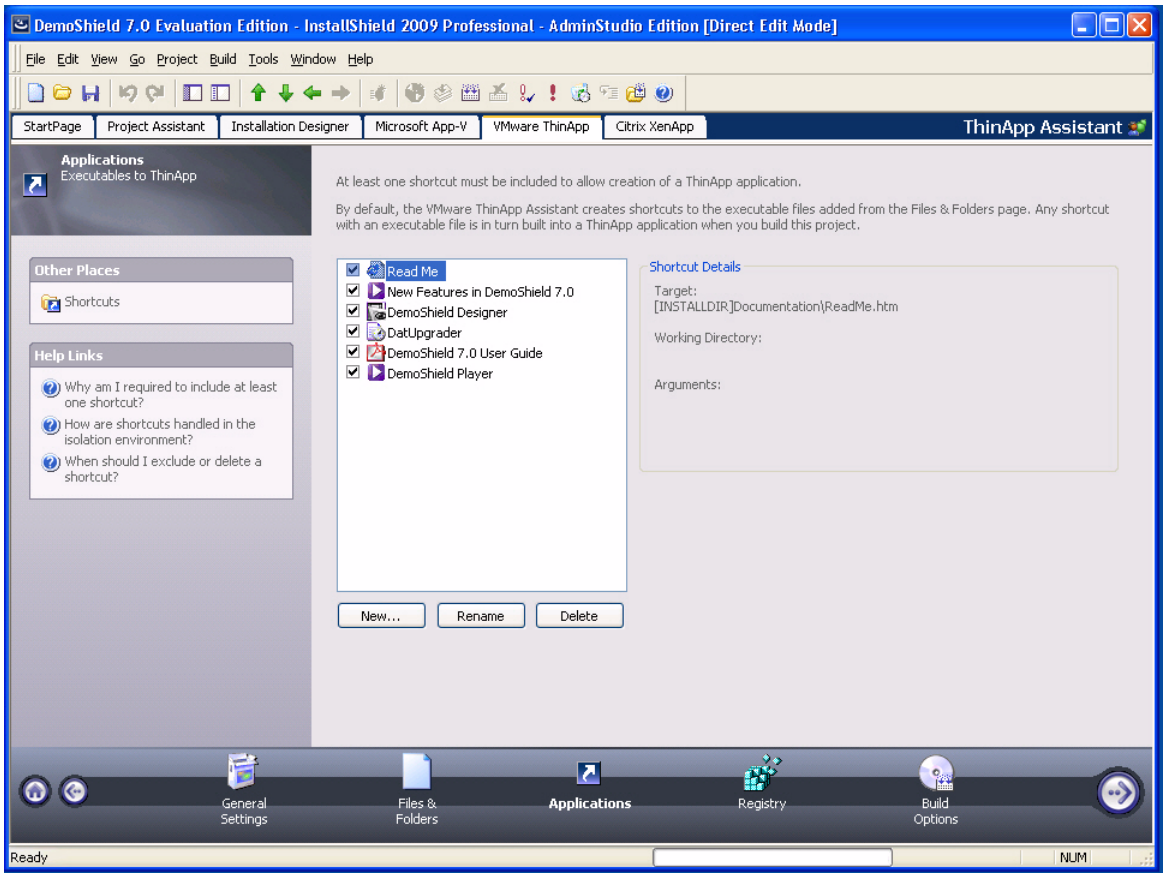


Figure 11-37: ThinApp Assistant Applications Page

Shortcut Requirements

For each ThinApp application, you are required to define **at least one** shortcut. You define application shortcuts to enable users to launch a ThinApp application from within the virtual environment. By default, the ThinApp Assistant creates ThinApp applications for all of the executable shortcuts that exist in your project (or Windows Installer package).

If you build a ThinApp application that does not contain any shortcuts, users will not be able to launch the application.

Difference Between Deleting and Excluding a Shortcut

To prevent a shortcut from being created in the ThinApp application, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the ThinApp application, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a ThinApp Application.
- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the ThinApp application and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a ThinApp Application.

Managing Shortcuts

On the **Applications** page, you can create, delete, include, exclude, or rename a ThinApp application. For step-by-step instructions, see the following topics:

- Creating a New ThinApp Application
- Including an Existing ThinApp Application
- Excluding or Deleting an Existing ThinApp Application
- Renaming a ThinApp Application

Registry Page

On the **Registry** page, you can view existing registry keys, values, and data, and add or delete registry items. You can also override the default isolation options for a registry key. Isolation options specify how the virtual environment will provide access to system resources requested by the application.

The default settings for isolation options are built into the ThinApp Assistant, and those defaults are adequate for most environments. However, you can override the default settings for selected registry keys to exert control over application interactions with client operating system resources.

You set isolation options by selecting a registry key and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting ThinApp Isolation Options.

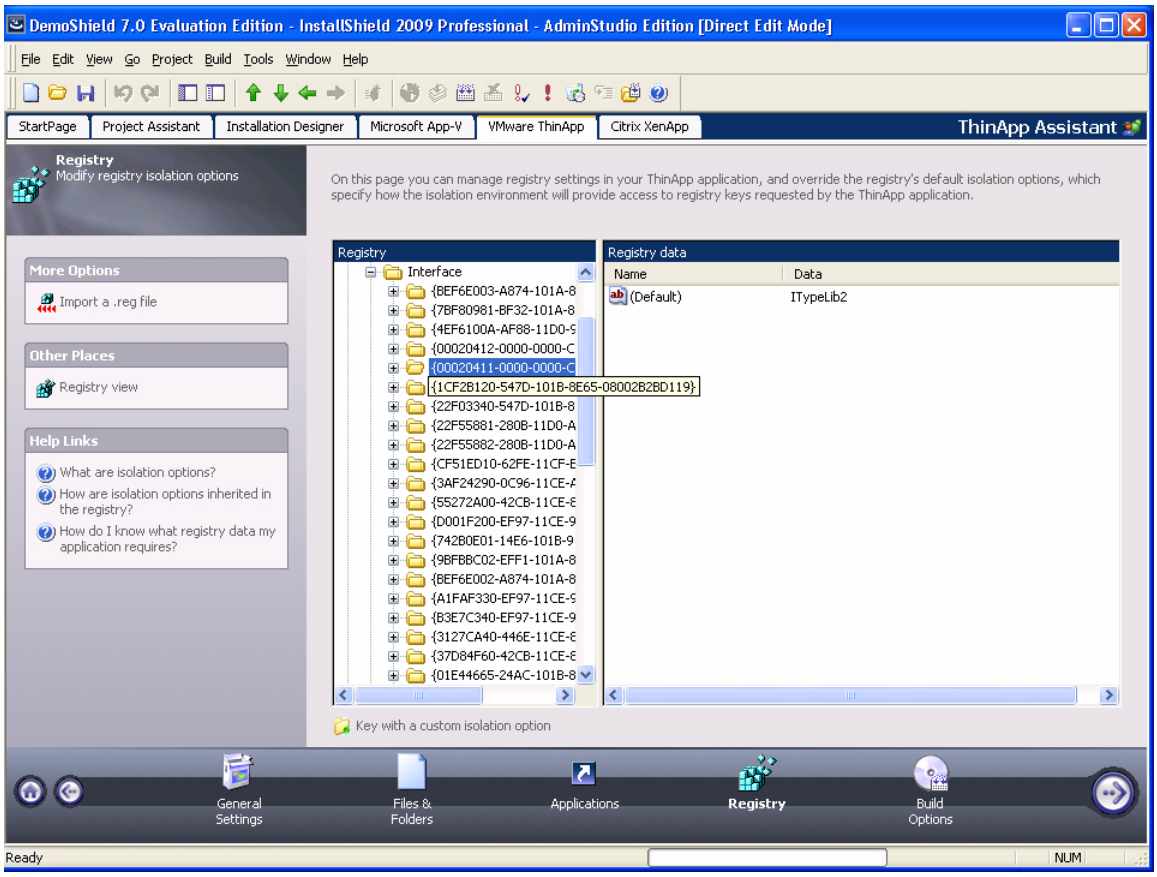


Figure 11-38: ThinApp Assistant Registry Page

Registry items that are listed on this page will be included in the ThinApp application, and those that you delete will not. By default, all new registry keys are isolated.

Tip • *To import an existing registry (.reg) file, click the* ***Import a .reg file*** *option on the* ***More Options*** *list to open the Registry Import Wizard.*

Note • *You cannot set isolation options on root registry keys.*

Editing the registry on the **Registry** page is performed much like it is performed on the InstallShield **Registry View**. See Editing the Registry.

For information on how to override a registry key's default isolation options, see Setting App-V Application Registry Isolation Options.

Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

Build Options Page

On the **Build Options** page, you can perform the following tasks:

- Specifying Build Options
- Including Additional Windows Installer Packages in a ThinApp Application
- Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application
- Selecting Releases to Build
- Enabling ThinApp Application Building When in Direct Edit Mode
- Clearing the ThinApp Cache
- Opening the ThinApp Application Folder
- Building a ThinApp Application
- Supporting AppSync and AppLink

The options on the Build Options page vary depending upon whether you are editing an InstallShield project or a Windows Installer package:

InstallShield Project

When you open an InstallShield project in InstallShield:

- The **Build Options** page includes a releases tree, and you select the release that you want to build.
- To build the ThinApp application, you click the **Build** button on the toolbar.

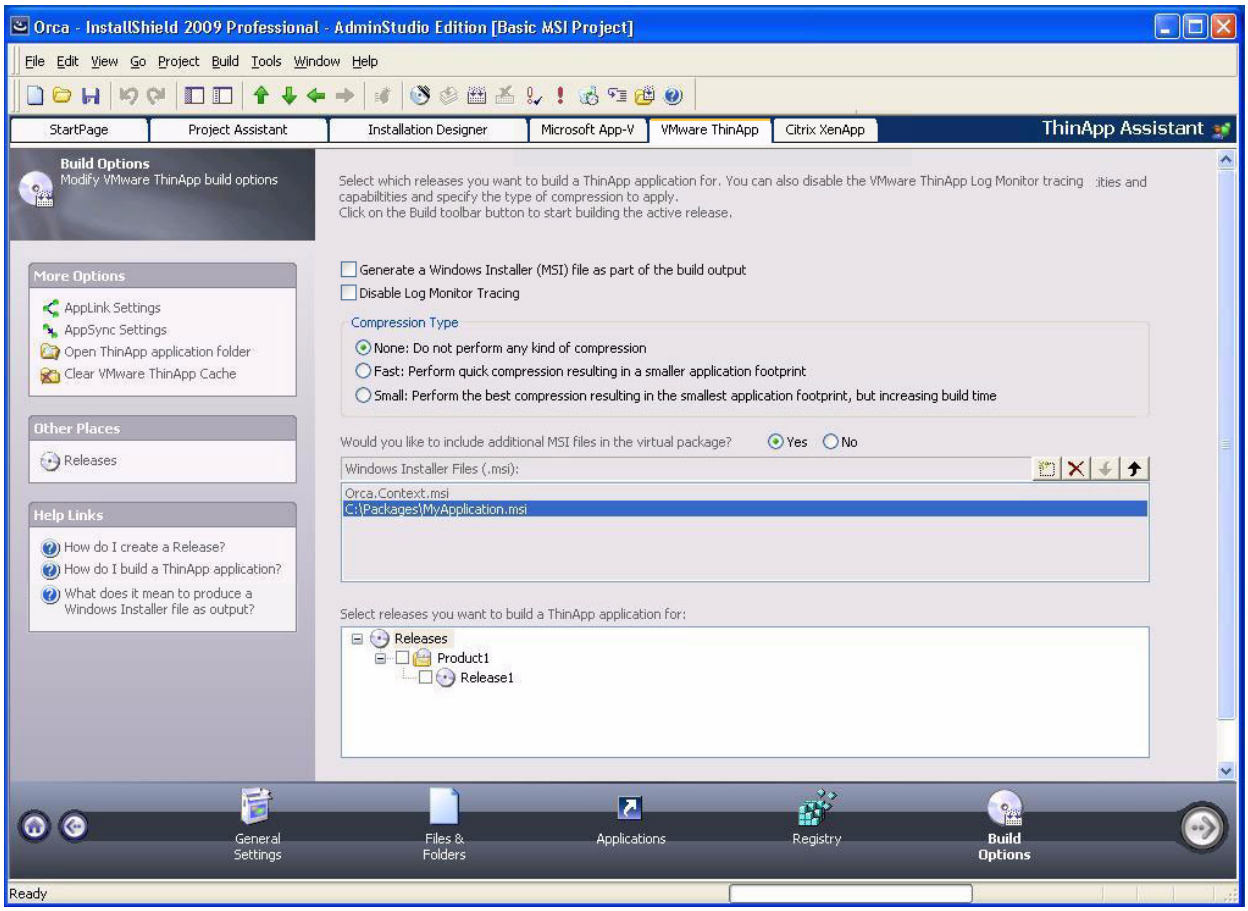


Figure 11-39: Build Settings Page, When in Basic MSI Project Mode

Windows Installer Package [Direct Edit Mode]

When you open a Windows Installer package in InstallShield:

- Because you do not have to select a release for a Windows Installer package, there is no releases tree.
- Because a Windows Installer package has already been built, InstallShield's standard build functionality is disabled. To build the ThinApp application, select the **Build ThinApp application** option and click the **Build Virtual Package** button.

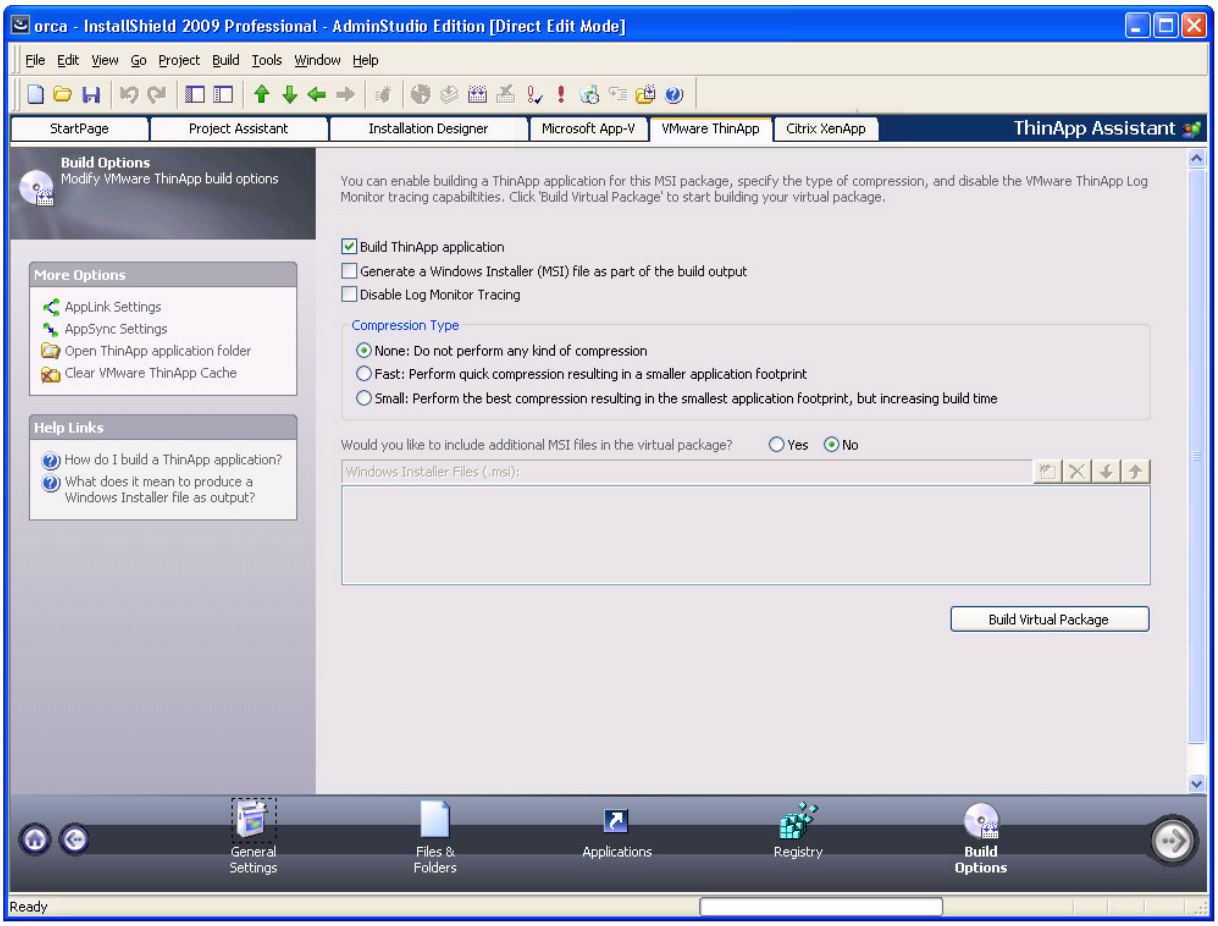


Figure 11-40: Build Settings Page, When in Direct Edit Mode

Specifying Build Options

On the **Build Options** page, you can specify the following options:

Table 11-32 • ThinApp Application Build Options

Option	Description
Build ThinApp Application	(Direct Edit/Direct MST Modes Only) When you directly edit a Windows Installer package, it is not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled. Select the **Build ThinApp Application** option to enable the Build function. When this option is selected, the **Build Virtual Package** button is enabled. For more information, see Enabling ThinApp Application Building When in Direct Edit Mode.

Table 11-32 • ThinApp Application Build Options

Option	Description
Build Virtual Package	(Direct Edit/Direct MST Modes Only) When you directly edit a Windows Installer package, if you select the **Build ThinApp Application** option, this button is enabled. Click it to build the ThinApp application. ***Note*** • *This button will also be enabled if the **Build Citrix profile** option is selected on the **Build Settings** page of the Citrix Assistant. In this scenario, if you click this button without also selecting the **Build ThinApp application** option on this page, the ThinApp application will not be built.*
Generate a Windows Installer (MSI) file as part of the build output	You can choose to build a Windows Installer package with your ThinApp application. This enables you to use enterprise distribution tools such as Microsoft System Center Configuration Manager or Novell ZENworks Configuration Management to distribute your ThinApp application. To build a Windows Installer file with your ThinApp application, select this option. By default, this option is not selected. For more information, see Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application.
Disable Log Monitor Tracing	Select this option if you do not want to allow ThinApp Log Monitor tracing for a ThinApp application. ThinApp Log Monitor is an application in the ThinApp Suite that allows you to record detailed information about any application's execution history for later review. For more information, see Setting ThinApp Log Monitor Tracing Options.
Compression Type	Select one of the following options to specify the ThinApp application's compression type: • **None**: Do not perform any type of compression • **Fast**: Perform quick compression resulting in a smaller application footprint • **Small**: Perform the best compression resulting in the smallest application footprint, but increasing build time. ***Note*** • *For more information, see Compressing a ThinApp Application.*

Table 11-32 • ThinApp Application Build Options

Option	Description
Would you like to include additional MSI files in the virtual package?	Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. To include additional Windows Installer packages in a ThinApp application, set this option to **Yes**, and then select the packages that you want to add. ***Note*** • *For more information, see Including Additional Windows Installer Packages in a ThinApp Application.*

Including Additional Windows Installer Packages in a ThinApp Application

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the ThinApp Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in a ThinApp application, set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**, and then select the packages that you want to add.

- Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
- The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
- Use the Delete button ( ) to delete a package from the list.

Building a Windows Installer Package to Assist in the Distribution of a ThinApp Application

You can choose to build a Windows Installer package to assist in the distribution of a ThinApp application by selecting the **Generate a Windows Installer (MSI) file as part of the build output** option on the **Build Options** page. By default, this option is not selected.

The Windows Installer file can be run to properly install the ThinApp application on an end-user's desktop. This simplifies the deployment of a ThinApp application by enabling you to use enterprise distribution tools such as Microsoft System Center Configuration Manager or Novell ZENworks Configuration Management.

A ThinApp application installed using a Windows Installer package can be uninstalled using **Add or Remove Programs** in the Control Panel.

Selecting Releases to Build

You select the releases that you want to build a ThinApp application for on the **Releases** tree of the **Build Options** page. By selecting a release, you are specifying that whenever that particular release is built, a ThinApp application will also be built.

Note • *If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the* ***Releases*** *tree on the* ***Build Options*** *page is not displayed.*

About Building Releases

When you select a release on the Releases tree on the **Build Options** page, you are specifying that whenever you build that particular release, you want to also build a ThinApp application for that release. However, the releases that are selected on the **Build Options** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Options** page:

- **Active release selected**—A Windows Installer package and a ThinApp application would be built.
- **Active release not selected**—Only a Windows Installer package would be built.

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

About Creating Releases

You cannot create or edit a release in the ThinApp Assistant. If no releases exist, you can simply click the **Build** toolbar button to create a new release or open the **Releases** view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a ThinApp application. For more information, see *Creating and Building Releases.*

If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Options** page is not displayed.

Enabling ThinApp Application Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **ThinApp Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a ThinApp application for this Windows Installer package. To enable the **Build** button to build just the ThinApp application, select the **Build ThinApp application** option on the **Build Options** page.

After you select this option, the **Build ThinApp application** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Clearing the ThinApp Cache

When you perform compressed builds, large temporary files are saved in a cache location. To delete all of these temporary files, select the **Clear the VMware ThinApp Cache** option in the **More Options** list on the **Build Options** page

Opening the ThinApp Application Folder

To quickly open the folder containing the ThinApp application files that were generated when this InstallShield project or Windows Installer package was built, click **Open ThinApp application folder** in the **More Options** menu.

Building a ThinApp Application

The method for building a ThinApp application depends upon what file you have open—an InstallShield project or a Windows Installer package. For detailed instructions, see one of the following topics:

- Building a ThinApp Application for an InstallShield Project
- Building a ThinApp Application for a Windows Installer Package

Supporting AppSync and AppLink

To configure AppSync and AppLink settings for your ThinApp application, click the **AppSync Settings** or **AppLink Settings** option in the **More Options** menu. For more information, see the AppSync Settings Dialog Box or the AppLink Settings Dialog Box.

Dialog Boxes

The ThinApp Assistant includes the following dialog boxes:

- ThinApp Diagnostic Tools Dialog Box
- Folder Isolation Options Dialog Box
- Registry Isolation Options Dialog Box
- AppSync Settings Dialog Box
- AppLink Settings Dialog Box
- Add AppLink Reference Dialog Box

ThinApp Diagnostic Tools Dialog Box

On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **General Settings** page, you can choose to include the Windows Command Prompt and Registry Editor diagnostic tools with your ThinApp application.

If you include diagnostic tools with your ThinApp application, you will be able to look at the registry or file system for the application while it is running in its virtual environment. For example, if you were running a ThinApp application and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Caution • *If you choose to include these diagnostic tools, the versions of* `regedit.exe` *and* `cmd.exe` *that are part of the operating system on the build machine are added to the ThinApp application. However, these tools may not be compatible with other operating systems.*

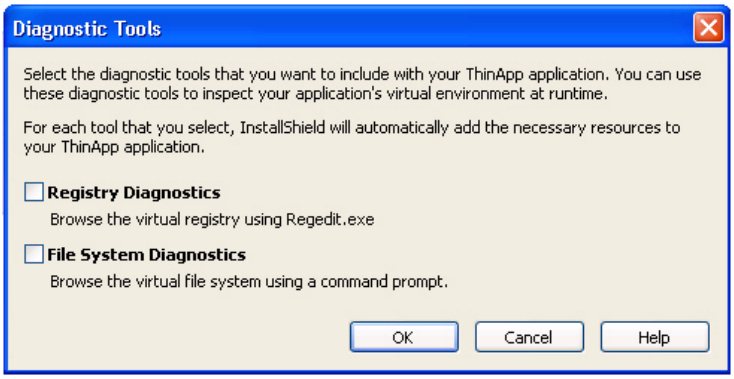


Figure 11-41: Diagnostic Tools Dialog Box

You can use these diagnostic tools to inspect your application's virtual environment at runtime. You have the following options:

Table 11-33 • Diagnostic Tools Dialog Box Options

Option	Description
Registry Diagnostics	Select this option if you want to include `regedit.exe` with your ThinApp application so that you can browse the registry.
File System Diagnostics	Select this option if you want to be able to browse the ThinApp application's virtual environment file system using a command prompt.

Launching the Diagnostic Tools Within the Virtual Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the ThinApp application.

When the user runs this ThinApp application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's virtual environment.

Folder Isolation Options Dialog Box

On the **Folder Isolation Options** dialog box, you can override the default isolation options for the selected folder.

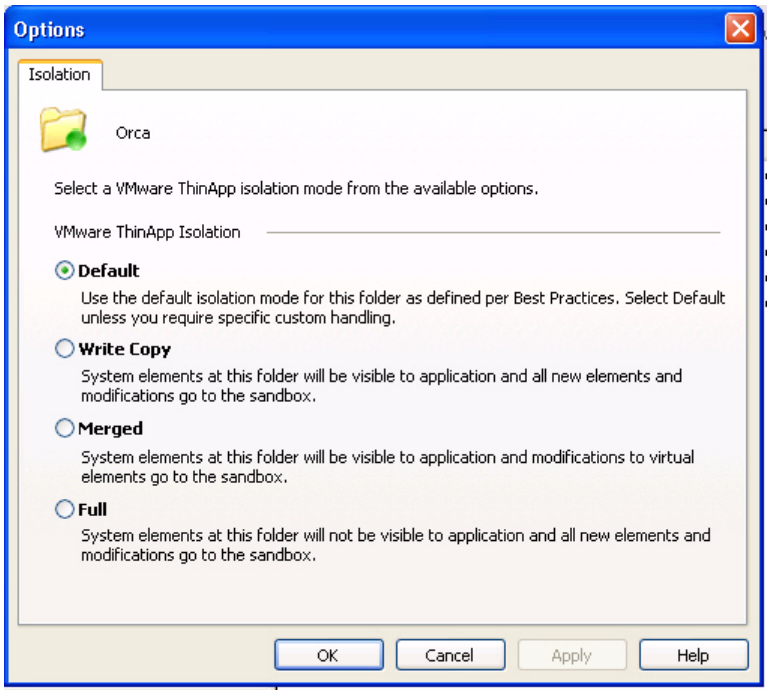


Figure 11-42: Folder Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings.*

The **Folder Isolation Options** dialog box includes the following options:

Table 11-34 • ThinApp Isolation Options

Option	Visibility of System Elements	Modifications to Virtual Elements	Modifications to System Elements	New Elements	If System and Virtual Element at Same Location
Default	*As defined internally by the ThinApp Assistant*				
Write Copy	Visible	Sandbox	Sandbox	Created in Sandbox	Sees Virtual Element
Merged	Visible	Sandbox	System	Created in System	Sees Virtual Element
Full	Not Visible	Sandbox	N/A (System elements cannot be modified)	Created in Sandbox	N/A (System elements cannot be read)

ThinApp Isolation Option Use Scenarios

The following table describes scenarios where you would use each isolation option:

Table 11-35 • Use Scenarios for ThinApp Isolation Options

Option	Use Scenario
Write Copy	You would use **Write Copy** isolation when: • Application was not designed or tested for multi-user environments and expects it can modify files and keys without impacting other users. • Application expects write permission to Global locations and was not designed for locked-down desktop environments found in corporate environments or Windows Vista. With **Write Copy** isolation, ThinApp makes copies of registry keys and files written by the application and performs all of the modifications in a user-specific sandbox. With this type of isolation, the ThinApp applications believe that they have global write permissions, while they really only modify the sandbox directory.
Merged	You would use **Merged** isolation when the ThinApp application needs write access to user-specific storage areas, like the Desktop and My Documents.
Full	You would use **Full** isolation when a ThinApp application needs to run on a machine where earlier or later versions of the same application are either installed or were not uninstalled correctly. For directories and registry keys that have **Full** isolation, the ThinApp application will not be aware of any host computer file that might exist, and it sees only virtual files and registry keys at fully isolated locations.

Registry Isolation Options Dialog Box

On the **Registry Isolation Options** dialog box, you can override the default isolation options for the selected registry key.

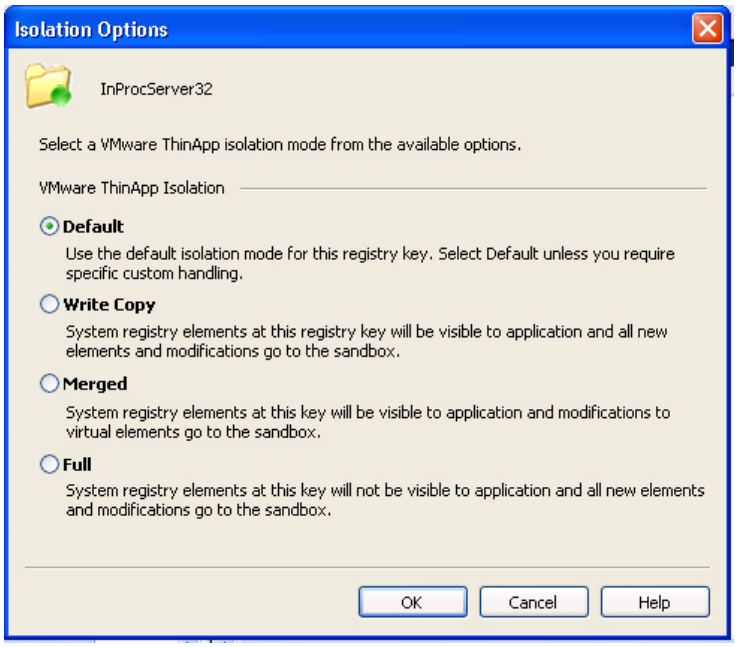


Figure 11-43: Registry Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects, ThinApp, and registry settings.*

The **Registry Isolation Options** dialog box includes the following options:

Table 11-36 • ThinApp Isolation Options

Option	Visibility of System Elements	Modifications to Virtual Elements	Modifications to System Elements	New Elements	If System and Virtual Element at Same Location
Default	*As defined internally by the ThinApp Assistant*				
Write Copy	Visible	Sandbox	Sandbox	Created in Sandbox	Sees Virtual Element
Merged	Visible	Sandbox	System	Created in System	Sees Virtual Element
Full	Not Visible	Sandbox	N/A (System elements cannot be modified)	Created in Sandbox	N/A (System elements cannot be read)

ThinApp Isolation Option Use Scenarios

The following table describes scenarios where you would use each isolation option:

Table 11-37 • Use Scenarios for ThinApp Isolation Options

Option	Use Scenario
Write Copy	You would use **Write Copy** isolation when: • Application was not designed or tested for multi-user environments and expects it can modify files and keys without impacting other users. • Application expects write permission to Global locations and was not designed for locked-down desktop environments found in corporate environments or Windows Vista. With **Write Copy** isolation, ThinApp makes copies of registry keys and files written by the application and performs all of the modifications in a user-specific sandbox. With this type of isolation, the ThinApp applications believe that they have global write permissions, while they really only modify the sandbox directory.
Merged	You would use **Merged** isolation when the ThinApp application needs write access to user-specific storage areas, like the Desktop and My Documents.
Full	You would use **Full** isolation when a ThinApp application needs to run on a machine where earlier or later versions of the same application are either installed or were not uninstalled correctly. For directories and registry keys that have **Full** isolation, the ThinApp application will not be aware of any host computer file that might exist, and it sees only virtual files and registry keys at fully isolated locations.

AppLink Settings Dialog Box

Note • *The AppLink Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppLink settings that you define will be ignored.*

The AppLink (Application Link) feature enables you to configure relationships between ThinApp applications that work together. You can set AppLink settings for the current ThinApp application on the **AppLink Settings** dialog box, which is opened by clicking the **AppLink Settings** option in the **More Options** menu of the ThinApp Assistant **Build Options** page.

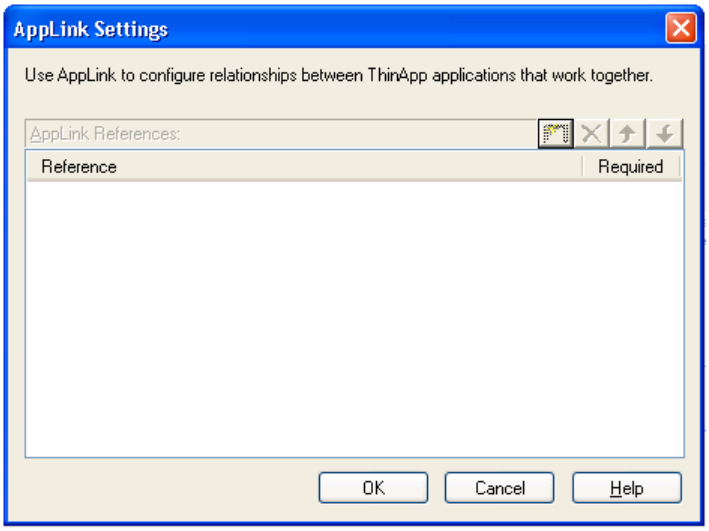


Figure 11-44: AppLink Settings Dialog Box

You can use the AppLink feature to perform the following tasks:

- **Linking runtime components to applications**—You can link runtime components to the applications that use them. For example, you can link a package containing the Java runtime environment (JRE) or ODBC drivers to a package containing a browser application.
- **Linking add-ons and plug-ins to applications**—You can link add-ons and plug-ins to applications. For example, Microsoft Office add-ons can be linked to applications or Adobe Photoshop plug-ins can be linked to a package containing Photoshop.
- **Linking packaged applications to service packs**—You can link packaged applications to service packs. By using AppLink, you can upgrade or roll back your service packs by changing the service pack that you capture and link to its parent application.

The **AppLink Settings** dialog box has the following options:

Table 11-38 • AppLink Settings Dialog Box

Option	Description
AppLink References	List of ThinApp applications that are linked to the open ThinApp application. The following information is listed: • **Reference**—List of linked ThinApp applications, including the application location and name. • **Required**—If **Yes** is listed in this column, the linked application must be available in order for the ThinApp application to run. If the linked application cannot be found, the ThinApp application will fail to run. See Required and Optional Linked Applications for more information.

Table 11-38 • AppLink Settings Dialog Box

Option	Description
Browse Button	Click the Browse button to open the **Add AppLink Reference** dialog box, where you can add a linked application to the **AppLink Reference** list. For more information, see Add AppLink Reference Dialog Box.
Up and Down Arrows	ThinApp uses a "last import wins" policy to determine what happens when two packages are imported that have the same files or registry keys. Therefore, you can use the Up and Down arrows to order the list of linked applications. See Collisions and Order of Import for more information. ***Note*** • *Initially, the Required and Optional linked applications are listed on this dialog box together, and you can change the order of these applications using the Up and Down arrows. However, at runtime, the linked applications in the Required category are read first, before those in the Optional category, even though an Optional application might have been listed before a Required application in the AppLink References list. Also, each time the* ***AppLink Settings*** *dialog box is reopened, the Required linked applications will be grouped at the top of the list, before all Optional applications.*

Required and Optional Linked Applications

When an application is linked to a ThinApp application, it can be designated to be either Required or Optional:

Required Applications

If a package is required, it has a mark in the **Required** column. If this package is missing from the virtual package, it will fail to run.

- If any specified package fails to import, an error message will be displayed and the parent executable file will exit.
- If a wildcard pattern is used to specify a package, no error message is displayed if no files match the wildcard pattern. Therefore, if a wildcard pattern is used to specify a package, the reference is always optional.
- To continue even if load errors occur, make the package references optional instead.

Optional Applications

If a package does not have a mark in the **Required** column, it is optional. An optional package operates the same as a required package except that if an import fails to load, the error is ignored and the main application will start executing.

Collisions and Order of Import

ThinApp uses a "last import wins" policy to determine what happens when two packages are imported that have the same files or registry keys.

For example, if `PackageA.exe` has `c:\myinfo.txt` in its virtual file system and `PackageB.exe` also has `c:\myinfo.txt` in its virtual file system, ThinApp will determine what happens based on which package is imported last.

- **Package order in the AppLink References list**—If `PackageA.exe` is listed before `PackageB.exe` on the AppLink References list, `PackageB.exe`'s copy of `c:\myinfo.txt` will be used. But if `PackageB.exe` is listed before `PackageA.exe` on the AppLink References list, `PackageA.exe`'s copy of `c:\myinfo.txt` will be used.
- **Wild cards**—When wild cards are used, alphabetical order is used to load packages, so if you enter `Package*.exe` in the AppLink References list, `PackageB.exe` will be loaded last (after `PackageA.exe`), so its copy of `c:\myinfo.txt` will be used.
- **VB scripts**—If two or more packages include VB scripts, the order of execution for the VB Scripts will be alphabetical order by the name of the package. If two packages contain a VB script with the same name, the "last import wins" policy will be used to execute only the version of the VB script from the last imported package containing a script with that name.

Caution • *Because VB Script name collisions could cause scripts from other packages not to be executed, it is important to use unique name for VB Script filenames.*

Security and Authorization

The user running the ThinApp application must be a member of all `PermittedGroups` sections for all of the linked (imported) ThinApp applications. If this is not the case, an `Access Denied` message will be displayed and the main ThinApp application will fail to load.

The following are limitations of the AppLink feature:

- ThinApp supports importing up to 250 packages at a time, and each package may be any arbitrary size.
- Packages that have been updated via AppSync will not have updates visible to the parent executable.
- Sandbox changes from packages being imported will not be visible to the parent executable.

Add AppLink Reference Dialog Box

Note • *The AppLink Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppLink settings that you define will be ignored.*

The AppLink (Application Link) feature enables you to configure relationships between ThinApp applications that work together. On the **Add AppLink Reference** dialog box, which is opened by clicking the Browse button on the **AppLink Settings** dialog box, you specify the name and location of a ThinApp application and indicate whether that application is Required or Optional.

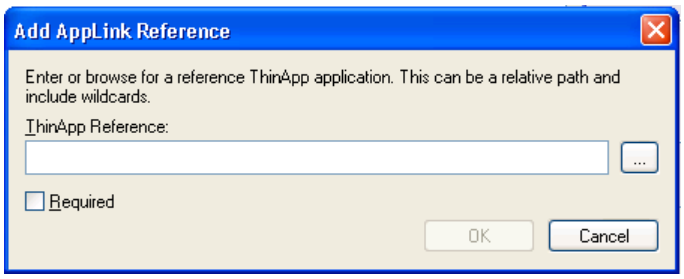


Figure 11-45: Add AppLink Reference Dialog Box

In the **ThinApp Reference** box, enter the relative (runtime) path to the existing ThinApp application that you want to link to. For more information on how to specify a ThinApp Reference, see the following:

- Enter a Relative Path
- Path Name Format
- Which ThinApp File Should Be Specified in an AppLink Reference?
- Required vs. Optional
- Examples of AppLink References

Enter a Relative Path

On the **Add AppLink Reference** dialog box, if you click Browse and browse for a ThinApp application, the absolute path to that application is entered, such as `C:\Program Files\AppName\filename.exe`. In that case, the main ThinApp application needs that linked application to be found at the specified absolute path location at runtime, which is unlikely. Therefore, it is recommended that you enter a relative path name.

Path Name Format

AppLink supports both URL and UNC path names.

Which ThinApp File Should Be Specified in an AppLink Reference?

If a ThinApp application has only one shortcut, it consists of a single executable. Therefore, you would obviously specify that executable file when creating an AppLink Reference.

However, when a ThinApp application has more than one shortcut, the ThinApp file that you specify in an AppLink Reference depends upon what tool you used to build the ThinApp application:

Table 11-39 • File to Specify in an AppLink Reference

Tool Used to Build ThinApp Application	# of Shortcuts	ThinApp Application File to Specify
AdminStudio or ThinApp	Only one	Specify the executable file (`.EXE`).

Table 11-39 • File to Specify in an AppLink Reference

Tool Used to Build ThinApp Application	# of Shortcuts	ThinApp Application File to Specify
AdminStudio	More than one	When built with AdminStudio, a ThinApp application that has more than one shortcut consists of two or more executable files and a `Package.DAT` file (as described in Components of an App-V Package). In this situation, specify the `Package.DAT` file.
ThinApp	More than one	When built with ThinApp, a ThinApp application that has more than one shortcut consists of multiple executable files, with one primary executable. In this situation, specify the primary executable file (`.EXE`).

Required vs. Optional

If you want this package to be required, select the **Required** option. If a required package is missing from the virtual package, it will fail to run. Note the following about required packages:

- If any specified package fails to import, an error message will be displayed and the parent executable file will exit.
- If a wildcard pattern is used to specify a package, no error message is displayed if no files match the wildcard pattern. Therefore, if a wildcard pattern is used to specify a package, the reference is always optional.
- To continue even if load errors occur, make the package references optional instead.

Examples of AppLink References

The following are examples of how packages can be added to the **AppLink References** list:

Table 11-40 • AppLink References Examples

Example	Description
`Plugin.exe`	This will import a single package located in the same directory as the parent executable.
`plugins\Plugin.exe`	This will import a single package located in the `plugins` subdirectory of the parent executable.

Table 11-40 • AppLink References Examples

Example	Description
`plugins\*.exe`	This will import all executables located in the `plugins` directory. ***Important*** • *If any executable fails to import because it is not a proper ThinApp package or because of a security issue, the parent executable will fail to load.*
`n:\plugins\*.exe`	This will import all EXEs located at the absolute path `n:\plugins`.
`%PLUGINS%\*.exe`	This expands the environment variable, `PLUGINS`, and imports all executables found at this location.
`plugin1.exe;plugin2.exe;plugins\*.exe`	This loads two specified plugins and a list of executables found in the `plugins` subdirectory.

AppSync Settings Dialog Box

Note • *The AppSync Settings feature requires ThinApp 4.x If you are using Thinstall 3.x, any AppSync settings that you define will be ignored.*

AppSync (Application Sync) enables you to automatically keep deployed virtual applications up to date. When an application starts up, AppSync can query a Web server to see if an updated version of the package is available. If an update is available, the differences between the existing package and the new package will be downloaded and used to construct an updated version of the package. The updated package will be used for future deployments.

You can use the AppSync feature to perform the following tasks:

- **Distribute runtime components separately**—You can use AppSync to distribute runtime components separately from the applications that use them. For example, the Java Runtime Environment (JRE) or ODBC drivers.
- **Apply layered service packs to applications**—You can use AppSync to apply layered service packs to your applications. Application Sync enables you to distribute service packs and roll back to previous versions, if necessary.

On the **AppSync Settings** dialog box, which is opened by clicking **AppSync Settings** on the M**ore Options** menu of the **Build Options** page, you can configure AppSync settings for your ThinApp application.

On the **Expiration** tab, you can specify that the ThinApp application is required to check for updates at a defined frequency. If the ThinApp application fails to successfully check for updates within that defined frequency, it will fail to run. Note that the update does not expire, the ThinApp application expires, and cannot be used until it is updated. successfully.

The **AppSync Settings** dialog box includes two tabs:

- General Tab
- Expiration Tab

General Tab

On the **General** tab, you specify the location of the Web server that hosts application updates.

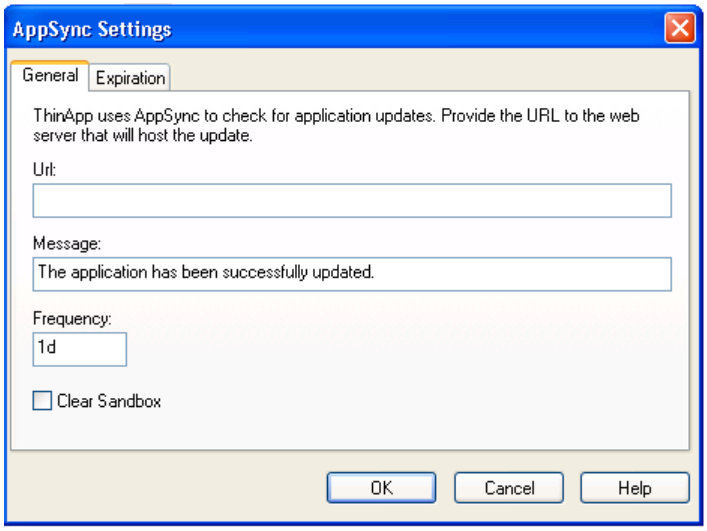


Figure 11-46: General Tab of the AppSync Settings Dialog Box

The following options are included:

Table 11-41 • General Tab of the AppSync Settings Dialog Box

Option	Description
Url	URL of the Web server where updates are stored. Application Sync works over both the HTTP (unsecure) and HTTPS (secure) protocol. Part of HTTPS is that the identity of the Web server is checked. You can include a user name and password in the URL that will be used for basic authentication. The standard Windows/Internet Explorer proxy setting is respected. For example: `https://example.com/some/path/PackageName.exe`
Message	When an updated package is first launched, an information message can be shown. For example: `Your application has been updated.`
Frequency	By default, a package will connect to the Web server once per day to see if an updated version is available. You can set the frequency by modifying this setting. For example, to set the **Frequency** to 2 days, enter `2d`. For 2 weeks, enter `2w`, etc.

Table 11-41 • General Tab of the AppSync Settings Dialog Box

Option	Description
Clear Sandbox	Gives you the option to clear the sandbox after an update. By default, the sandbox is not cleared. Select this option to clear the sandbox.

Expiration Tab

On the **Expiration** tab, you can specify that a ThinApp application is required to check for updates at a defined frequency. If the ThinApp application fails to successfully check for updates within that defined frequency, it will fail to run.

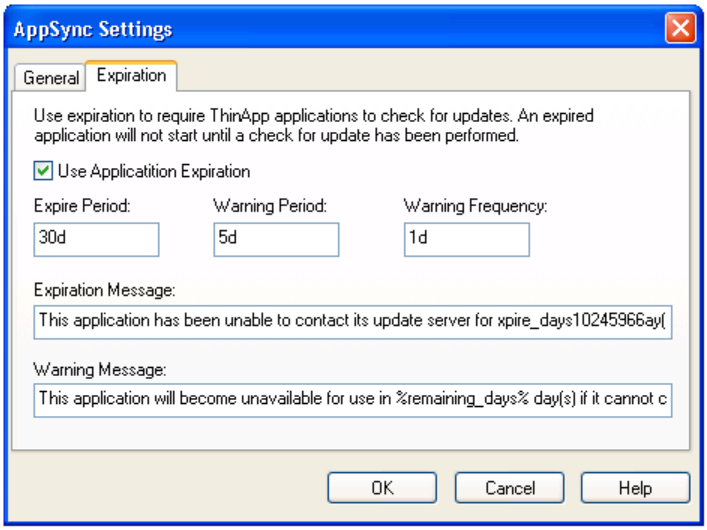


Figure 11-47: Expiration Tab of the AppSync Settings Dialog Box

The following options are included:

Table 11-42 • Expiration Tab of the AppSync Settings Dialog Box

Option	Description
Use Application Expiration	Select this option to require that an application has to check for updates at a specified frequency.
Expire Period	Sets the update frequency in minutes (m), hours (h), or days (d). If the Web server cannot be reached, the package will continue to work until the Expire Period is reached. This default setting is 30 days but you can change that setting by modifying this setting. For example: • To set the period to 30 days, enter **30d** • If you do not want the package to expire, clear the **Use Application Expiration** check box.

Table 11-42 • Expiration Tab of the AppSync Settings Dialog Box

Option	Description
Warning Period	Sets the start of the warning period before a package expires. For example, to set the period at 5 days, enter `5d`.
Warning Frequency	Sets the frequency of warnings before the package expires. With the default of one day, the warning message will be displayed once per day only. To configure the warning to pop up on every application launch, enter `0`. To configure it to pop up every 4 days, enter `4d`. After the warning period has started, the Web server will be checked on every launch of an application, overriding any previous setting. As long as a package has not expired, this parameter checks for new versions and downloads will occur in the background. The user can continue to use the old version. If the application is terminated by the user before the download is complete, the download will resume when a virtual application is launched again. After the download completes, the new version will be activated on the next launch. When the package has expired, the version check and download will happen in the foreground. A progress bar will be shown during the download phase.
Expiration Message	After the expiration limit has been reached and a virtual application is started, it will try to connect to the Web server and check for a new version. If the connection fails, a message box will be shown and execution will be terminated. The default message is shown in the example below. For example: `This application has been unable to contact its update server for` *`Expire_Period`* `days, so it is unavailable for use. Check your network connection and try again.`
Warning Message	If the connection to the Web server fails, a message box will be shown. The default message is: `This application will become unavailable for use in` *`Warning_Period`* `days if it cannot contact its update server. Check your network connection to ensure uninterrupted service`

Note • *If you use AppSync, VMware recommends that you disable automatic application updates that are configured in your virtual application. Conflicts might occur between the linked packages and the software that is automatically updated. If an automatic update feature updates an application, it stores the updates in the sandbox. If AppSync then updates the application to a different version, the updates stored in the sandbox take precedence over the files contained in the version that AppSync created. The order of precedence for the update files are those in the sandbox, then the virtual operating system, and then the physical machine.*

Building ThinApp Applications Using the Command Line

When you configure a ThinApp application in an InstallShield project and then build that project (using either the user interface or the command line), both the Windows Installer package and the ThinApp application are built. When you use the standard InstallShield command line build, you do not need to add any additional command line parameters. All of the ThinApp application settings are saved within the InstallShield project.

ThinApp Application Conversion Error and Warning Messages

For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see *Virtualization Conversion Errors and Warnings.*

Application Features Requiring Pre- or Post-Conversion Actions

Some application features are ignored when creating a ThinApp application. Therefore, some additional pre- or post-conversion actions must be taken in order for the ThinApp application to be created properly.

One action you could take to try to include ignored features in an ThinApp application is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to a ThinApp application.

For a list of ignored features, see Application Features Requiring Pre- or Post-Conversion Actions.

ThinApp Not Found

To create a ThinApp application, you are required to have both AdminStudio and ThinApp installed on the same machine. If a user attempts to create a ThinApp application without this ThinApp component, a message is displayed and the build is unsuccessful.

To purchase the ThinApp, visit the VMware Web site:

`http://thinstall.com/products/virtualization_suite.php`

ThinApp Application Configuration File: package.ini

ThinApp application configuration options that you set in the ThinApp Assistant interface are recorded in the `package.ini` file that is generated when the ThinApp application is built.

A `package.ini` contains the following groups of options:

- [BuildOptions]
- [Compression]
- [Isolation]
- [MainApp.exe]
- [Test.exe]

Note • *For the latest information on the ThinApp application configuration file, package.ini, consult your ThinApp documentation.*

[BuildOptions]

The `[BuildOptions]` section of the `package.ini` file specifies Global options which will be inherited by each child executable file. The following options are included:

Table 11-43 • [BuildOptions] Section of package.ini

Option	Description

SandboxName	When a ThinApp application is built, a Sandbox cache is created in the following location:
	`c:\Documents & Settings\USER_NAME\Application Data\ThinApp\SandboxName`
	The `SandboxName` entry in the `package.ini` file is used to name the directory where sandbox files are stored at runtime.
	`SandboxName=MyApplicationV3`
	By default, AdminStudio names the Sandbox by assigning it a unique GUID. However, if you want to override this default Sandbox name, you may (optionally) enter a new name in the package.ini file using the `SandboxName` option.
	If no Sandbox Name is entered, a unique GUID is used, such as:
	`SandboxName={2BDBE10A-9E53-4B5E-811D-DF8019D0B13C}`
	
	Note • *This option corresponds to the **Sandbox Name** field on the **General Information** page.*

InventoryName	Used by desktop management systems to identify packages for usage reporting purposes. If you do not use a desktop management system or license-controlled system, this value has no effect `InventoryName=MainApp v1.0`

Table 11-43 • [BuildOptions] Section of package.ini (cont.)

Option	Description
SandboxNetworkDrives	Enable this option if you want changes to data on Network-mapped drives to go into the sandbox. By default, the ThinApp application can read and write to network mapped drives with no changes. The value for **SandboxNetworkDrives** is set to either 0 (off) or 1 (on). `SandboxNetworkDrives=0` ***Note*** • *This option corresponds to the **Mapped Network Drive Changes go to Sandbox** option on the **General Settings** page.*
SandboxRemovableDisk	Enable this option if you want changes to data on Removable disk (floppy/flash) to go into the sandbox. By default the application can read and write to removable disk with no changes. The value for **SandboxRemovableDisk** can be set to either 0 (off) or 1 (on). `SandboxRemovableDisk=0` ***Note*** • *This option corresponds to the **Removable Disk Changes go to Sandbox** option on the **General Settings** page.*
RemoveSandboxOnExit	Enable this option if you want to delete the sandbox when the ThinApp application exits. This resets the application to its original captured state If the application spawns child processes, the clean up will be postponed until all have quit. The value for **RemoveSandboxOnExit** can be set to either 0 (off) or 1 (on). `RemoveSandboxOnExit=0` ***Note*** • *This option corresponds to the **Reset Sandbox on Exit** option on the **General Settings** page.*

Table 11-43 • [BuildOptions] Section of package.ini (cont.)

Option	Description
ExternalCOMObjects	This option allows you to specify that you want specific COM objects to be executed on the system instead of in the virtual environment. This option only applies to out-of-process COM objects (LocalServer32) and Services-based COM objects. To specify multiple objects, put a semicolon after each entry. Objects should always be specified in CLSID format The following class ID specifies the class ID for Microsoft Word: `ExternalCOMObjects={000209FF-0000-0000-C000-000000000046};{000209FF-0000-0000-C000-000000000047}` ***Caution*** • *This option is for advanced users.*

VirtualizeExternalOutOfProcessCOM	Enable this option if you want all out-of-process COM objects to be loaded outside of the virtual environment . By doing this, the application may indirectly modify the machine—for example, the MSI installer service COM object could be modified.
	
	`VirtualizeExternalOutOfProcessCOM=0`
	
	The value for this option can be set to either:
	
	• **0**—Inside virtual environment
	• **1**—Outside the virtual environment
	
	`The default is to create all out-of-process COM objects inside the virtual environment.`

PermittedGroups	Using this option, you can specify the Active Directory groups which are allowed to use this ThinApp application. `PermittedGroups=Group1;Group2;Group3` ***Note*** • *This option corresponds to the* ***Allow application execution to the following user groups*** *option on the* ***General Settings*** *page.*

Table 11-43 • [BuildOptions] Section of package.ini (cont.)

Option	Description
AccessDeniedMsg	Use this option to customize the message the user sees if they do not have permission to execute a ThinApp application. `AccessDeniedMsg=You do not have access to execute this application, please contact your Administrator` ***Note*** • *This option corresponds to the **Message shown when users not belonging to above groups run the ThinApp application** field on the **General Settings** page.*

ChildProcessEnvironmentDefault and **ChildProcessEnvironmentExceptions**	Executables located in the virtual file system are always executed within the virtual environment. Executables located in the physical file system can be executed inside or outside the virtual environment.
	The default is determined by the **ChildProcessEnvironmentDefault** option, which can be set to **Virtual** or **External**. If this option is not present, the default is the **Virtual** environment.
	It is possible to override the default for specific applications by specifying a list of applications, separated by semicolons, using the **ChildProcessEnvironmentExceptions** option. If a complete path is specified, the full name of the executable is used for the comparison; otherwise, only the file name is used.
	For example:
	`ChildProcessEnvironmentDefault=Virtual`
	`ChildProcessEnvironmentExceptions=exec.exe;c:\path\file.exe`
	In this example, `c:\exec.exe`, `c:\Windows\exec.exe` and `c:\path\file.exe` would be executed externally.

AutoShutdownServices	Use this option to specify if virtualized services keep on running when the last non-service process exits. Permitted values are: • **0**—Keep on running. • **1**—Stop virtualized services (Default). `AutoShutdownServices=1`

Table 11-43 • [BuildOptions] Section of package.ini (cont.)

Option	Description
NetRelaunch	Under some conditions, Norton AntiVirus will try to perform a complete scan of an executable. This scan can have a big impact on launch times for large executable files located on network shares. Norton AntiVirus decides to perform a complete scan under these conditions: • **If the executable is launched from a network share or removable disk.** It skips the scan when the executable is located on the hard drive). • **When the executable makes its first network connection.** It does not scan the executable if the executable does not make any network connections. Because a large number of desktops have Norton AntiVirus installed, ThinApp automatically compensates for this by allowing applications to launch from a network share without incurring the lengthy scan times. It does so by creating a small stub executable in the user's sandbox which is then relaunched. Because the small executable can be scanned quickly, it will load the remainder of the application data from the original source location. You can disable ThinApp default behavior by adding the `NetRelaunch=1` option to disable full file scans. `NetRelaunch=1`

[Compression]

The `[Compression]` options specify the default compression options to use when building the ThinApp application.

Table 11-44 • [Compression] Section of package.ini

Option	Description
CompressionType	To reduce the application startup time, you can specify the CompressionType option to compress the ThinApp application. `CompressionType=Fast` Specify one of the following options: • **None**: Do not perform any type of compression • **Fast**: Perform quick compression resulting in a smaller application footprint • **Small**: Perform the best compression resulting in the smallest application footprint, but increasing build time. ***Note*** • *This option corresponds to the **Compression Type** options on the **Build Options** page.*

[Isolation]

The `[Isolation]` options specify the isolation options to use for folders and registry keys when building the ThinApp application.

Table 11-45 • [Isolation] Section of package.ini

Option	Description
DirectoryIsolationMode	This option specifies the default isolation options to use for folders when building this project. `DirectoryIsolationMode=WriteCopy|Merged` This option has the following possible values: • **WriteCopy**—System elements are visible, modifications to both virtual and system elements are made in the sandbox, new elements are created in the sandbox, and if a system element and a virtual element are at the same location, the application sees the virtual element. • **Merged**—System elements are visible, modifications to virtual elements are made in the sandbox, modifications to system elements are made on the system, new elements are created on the system, and if a system element and a virtual element are at the same location, the application sees the virtual element.

Table 11-45 • [Isolation] Section of package.ini (cont.)

Option	Description
RegistryIsolationMode	This option specifies the default isolation options to use for registry keys when building this project. `RegistryIsolationMode=WriteCopy|Merged` This option has the following possible values: • **WriteCopy**—System elements are visible, modifications to both virtual and system elements are made in the sandbox, new elements are created in the sandbox, and if a system element and a virtual element are at the same location, the application sees the virtual element. • **Merged**—System elements are visible, modifications to virtual elements are made in the sandbox, modifications to system elements are made on the system, new elements are created on the system, and if a system element and a virtual element are at the same location, the application sees the virtual element.

[MainApp.exe]

The `[MainApp.exe]` section specifies the source executable, the name of the file that contains read-only registry data to be bound, whether to perform logging, and the icon to use for the executable.

Table 11-46 • [MainApp.exe] Section of package.ini

Option	Description
Source	This option specifies the `.exe` which will be run to launch the ThinApp application. `Source=%ProgramFiles%\Test\MainApp.exe` This option also specifies the icon that will be used, if an icon is not explicitly specified using the Icon option.
ReadOnlyData	This option specifies the name of the file that contains read-only registry data to be bound. If the read-only registry also has an associated file-data, the file-data file should be in the same directory with the appended extension `TestMain.exe.ro.thfd`. `ReadOnlyData=bin\MainApp.exe.ro.tvr`

Table 11-46 • [MainApp.exe] Section of package.ini (cont.)

Option	Description
DisableTracing	This optional setting will disable logging/tracing capabilities for this application when Log Monitor is running. Possible values are 1 (logging is disabled) or 0 (logging is enabled). `DisableTracing=1` ***Note*** • *This option corresponds to the* ***Disable Log Monitor Tracing*** *option on the* ***Build Options*** *page.*

Icon	By default the icon is used from the executable identified in the Source option. You can change this to specify one of the following:
	`Icon=SomeOtherEXE.exe`
	`Icon=NULL`
	`Icon=SomeOtherIco.ico`

Table 11-46 • [MainApp.exe] Section of package.ini (cont.)

Option	Description

RetainAllIcons	By default, each application retains the main Group Icon from its Source executable and the individual icon resource pointed to by the Group Icon. Tlink will strip out extra icons that cannot be used directly by the system shell. However, you can force these extra icons to be included in the ThinApp executable by using the `RetainAllIcon=1` option. For example:
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`RetainAllIcons=1`
	Instead of using the Source option to identify your application icon, you can also use:
	1. **The value NULL.** In this case, the application will not have an icon and Windows will use the default application icon.
	`[myapp.exe}`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=NULL`
	2. **The path to another .exe file.** In this case, Tlink will load the icons from a different .exe file. If a full path is not specified, the path is relative to the project directory.
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=%ProgramFilesDir%\myapp\app2.exe`
	Executable files can contain multiple icon sets. You can optionally specify which set to use by appending ",1" ",2" to the end of the Icon path name like this:
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=%ProgramFilesDir%\myapp\app2.exe,1`
	3. **The path to an .ico icon file.** In this case, Tlink will load the icons from the specified .ico file. If a full path is not specified, the path is relative to the project directory.
	`[myapp.exe]`
	`Source=%ProgramFilesDir%\myapp\app.exe`
	`Icon=%ProgramFilesDir%\myapp\myicon.ico`

[Test.exe]

The `[MainApp.exe]` section specifies the Shortcut and WorkingDirectory options.

Table 11-47 • [Test.exe] Section of package.ini

Option	Description
Shortcut	The **Shortcut** option specifies whether the `.exe` that is generated will contain any registry or file data. This information will be loaded from the `.exe` referenced by the **Shortcut** option. Shortcut applications can specify `WorkingDirectory` and `CommandLine` `Shortcut=MainApp.exe`
WorkingDirectory	The **WorkingDirectory** option specifies where the ThinApp application will start. If this option is not specified, the Current Working Directory will be inherited from the parent process `WorkingDirectory=%ProgramFiles%\Test`

Creating Citrix Profiles

You can use the Citrix Assistant to help you author a Citrix profile for an application. The Citrix profile can then be deployed on a Citrix XenApp. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine. Using the Citrix Assistant, you can configure a Citrix profile's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options.

Information about creating Citrix profiles using the InstallShield Citrix Assistant is organized into the following sections:

- Overview of the Citrix Assistant
- Using the Citrix Assistant to Create a Citrix Profile
- Citrix Assistant Reference

Overview of the Citrix Assistant

You can use the Citrix Assistant to help you author a Citrix profile for an application. The Citrix profile can then be deployed on a Citrix XenApp. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine. Using the Citrix Assistant, you can configure an application's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options.

The process for authoring a Citrix profile using the Citrix Assistant is as follows:

Table 11-48 • Steps to Convert a Windows Installer Package to a Citrix Profile

Step	Go To:	Actions
Getting Started	**InstallShield Start Page**	Create or open one of the following project types: • Basic MSI • MSI Database (Direct Edit Mode) • Transform (Direct MST Mode)
	Citrix Assistant Home Page	Click on the **Citrix XenApp** tab to open the **Citrix Assistant Home** page
Specifying Citrix Profile Information	**Profile Information Page** 	Specify the name and version of the Citrix profile, whether this package can run executables that are not included with the Citrix profile, and whether to include diagnostic tools with the Citrix profile.
Specifying Operating System and Language Requirements	**Profile Requirements Page** 	Specify the operating systems and language requirements that client workstations must meet in order for this application to operate properly. You can also specify pre-launch and post-exit scripts to execute.
Managing Files and Folders in a Citrix Profile	**Profile Files Page** 	View existing files and folders, add and delete files.
Setting Isolation Options	**Profile Files Page** 	Override the Citrix default isolation options for selected folders and files. Isolation options specify how the virtual environment will provide access to files and folders requested by the Citrix profile.
Modifying Profile Shortcut Settings	**Profile Shortcuts Page** 	Create, delete, include, exclude, or rename a Citrix profile's executables, which are derived from the shortcuts in its Windows Installer package.
Modifying Profile Registry Settings	**Profile Registry Page** 	Add, delete, or modify the registry settings in your Citrix profile, and override the Citrix default isolation options for selected registry keys. Isolation options specify how the virtual environment will provide access to registry keys requested by the Citrix profile.

Table 11-48 • Steps to Convert a Windows Installer Package to a Citrix Profile

Step	Go To:	Actions
Modifying Build Settings	**Build Settings Page**	Choose whether to digitally sign the Citrix profile and select the releases that you want to build. Also, when you have a Windows Installer package open in Direct Edit mode, you can enable the **Build Release** option on the **Build** menu by making a selection on this page.
Building a Citrix Profile	**Build on the Toolbar** OR **Build Citrix Profile (F7) on the Build Menu**	Click **Build** to build the active Release and create a Citrix profile. Also, when you have a Windows Installer package open in Direct Edit mode, you can enable the **Build Release** option on the **Build** menu by selecting the **Build Citrix Profile** option on this page.

Information about the Citrix XenApp and Citrix profiles is presented in the following topics:

- About Citrix XenApp
- About the Citrix Assistant
- About Citrix Profiles
- Benefits of Deploying Citrix Profiles
- Supported InstallShield Project Types
- How Transforms are Included in a Citrix Profile

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

About Citrix XenApp

Citrix XenApp is an application delivery system for Windows applications that offers both application virtualization and application streaming. Applications are centralized on the Citrix XenApp, and then those applications are deployed to users throughout the enterprise. These deployed applications run within isolation environments that prevent them from interfering with other software running on the same machine.

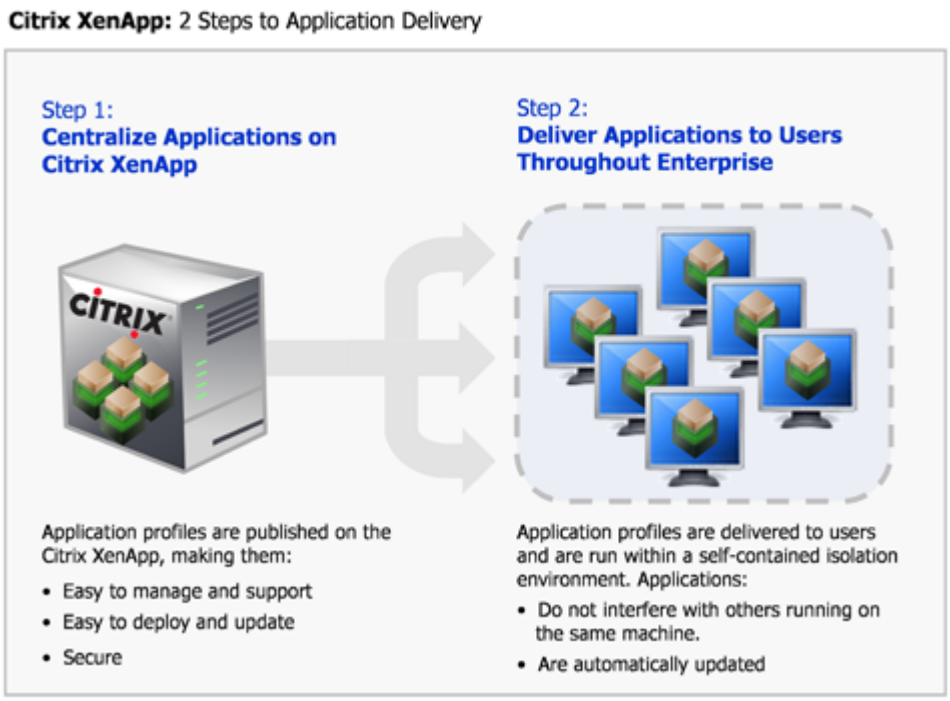


Figure 11-48: Citrix XenApp: Two Steps to Application Delivery

When applications are deployed on a Citrix XenApp, users can run those applications in an isolation environment, without installing, while connected or offline. Applications behave just like they were installed locally, but without any of the problems of installation, such as interfering with other applications on the same device. Files are saved locally and individual settings are preserved. Every time the application is run, it checks for errors or updates and they are delivered automatically.

Note • *For more information, see Benefits of Deploying Citrix Profiles.*

About the Citrix Assistant

You can use the Citrix Assistant to prepare a Windows Installer package for deployment on Citrix XenApp by converting it to a Citrix profile. During this process, you:

- **Profile Information**—Specify profile information.
- **OS and Language Requirements**—Specify the operating system and language requirements for the application.
- **Files, Folders, Shortcuts, Registry Settings**—Specify files, folders, shortcuts, and registry settings included in application.
- **Isolation Options**—Define a set of options for running the application in isolation on the user desktop.
- **Build**—Specify build settings and build a Citrix profile.

The following diagram illustrates the Citrix profile creation process:

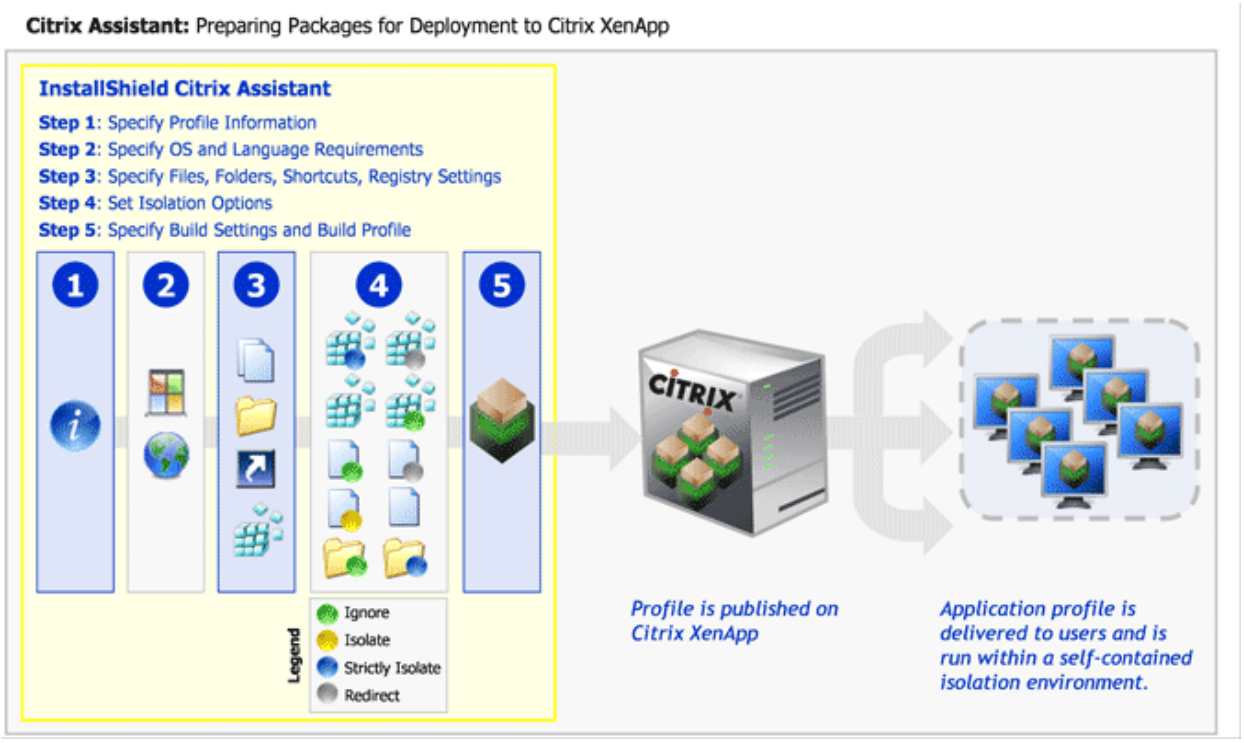


Figure 11-49: Preparing Packages for Deployment on Citrix XenApp

Note • *You can also convert a Windows Installer package to a virtual application using Automated Application Converter. See Performing Automated Repackaging and Virtualization Using the Automated Application Converter.*

About Citrix Profiles

When you use the Citrix Assistant to prepare a Windows Installer package for deployment on the Citrix XenApp, the resources you generate are called *profiles*. A profile consists of the following files and directories:

Table 11-49 • Components of an Citrix Profile

Component	Name	Description
Profile Manifest File	`myapp.profile`	An XML file that defines the profile.
CAB File	`[alphanumeric_string].cab`	Compressed cabinet file that provides the isolation environment contents for the application.
Hashes File	`Hashes.txt`	Hash key file for digital signatures and signing profiles.

Table 11-49 • Components of an Citrix Profile

Component	Name	Description
Icons File	`Icons.bin`	Icons repository.
Scripts Folder	`Scripts`	Folder containing any pre- launch or post-exit scripts that you have chosen to include.

Caution • *Modifying these files directly is **not recommended**. To make any modifications, use the Citrix Assistant.*

These files are saved in a directory named `CitrixProfile`. The location of the `CitrixProfile` directory depends upon the type of file you are editing in InstallShield:

- **InstallShield project**—The `CitrixProfile` directory will be located in a subdirectory of the directory that contains this InstallShield project file, such as:

  `C:\InstallShield 2008 Projects\ProductName\ConfigurationName\ReleaseName\CitrixProfile`

- **Windows Installer package**—The `CitrixProfile` directory will be located in the same directory as the Windows Installer file, such as:

  `C:\FolderContainingMSI\CitrixProfile\ProductName`

The contents of the application profile are published on the Citrix XenApp.

A profile can contain a single application or suite of applications. For example, you can profile Microsoft Word by itself, or you can profile the entire Microsoft Office suite in a single profile.

Benefits of Deploying Citrix Profiles

Converting a Windows Installer package to a Citrix profile and deploying it on a Citrix XenApp offers the following benefits:

- Reduces Application Conflicts
- Enables Rapid, Low Cost Application Deployment
- Enables Automatic Software Updates
- Centralized Application Management Provides Controlled Access and Security
- Enables User-Based Application Access Rather Than Machine-Based Access

Reduces Application Conflicts

Traditionally to deploy an application throughout an enterprise, the application was installed on each user's desktop. Therefore, prior to installation, each application had to be tested for conflicts against each target desktop image (operating system with existing applications). After resolving conflicts that were found during testing, each application then had to be installed on each desktop. This process was very time consuming not only during initial installation, but also when applying patches or upgrading.

Citrix profiles run within isolation environments, which separate the interaction between an application and the underlying operating system's resources in order to prevent the applications from interfering with others running on the same machine. Because applications do not interact, the need to perform any conflict analysis and regression testing prior to deployment is eliminated. This not only results in rapid application deployment, but it also reduces the total cost of application delivery, due to decreased labor by IT.

Also, because users running applications in an isolation environment encounter no conflicts with other applications, user calls to the help desk are decreased.

Enables Rapid, Low Cost Application Deployment

Deploying Citrix profiles on Citrix XenApp simplifies the deployment of new applications, updates and patch deployment, regardless of the diversity of the access devices, software languages, computing architectures, and networks that are involved.

- **Only a single instance of the application is installed**—Instead of deploying, managing, updating and securing a vast array of heterogeneous client software on each individual user's access device, a single instance of the application is installed on the Citrix XenApp. The IT department only has to test for one environment, and deploy and update in one place. This reduces the cost of application installation and support. Also, you can deploy a Citrix profile once on a Citrix XenApp and replicate it to other Citrix XenApps within the existing enterprise infrastructure.
- **Prevents application-specific server silos**—Deploying applications on Citrix XenApp prevents the build-up of application-specific server silos because you can safely install and reliably run multiple application versions and incompatible applications on the same server.
- **Enables you to quickly install and update software throughout your enterprise**—Because you can manage the delivery of all of your Windows-based applications from one centralized location, there is no need to go from desktop to desktop, travel from office to office, or wait for laptops to return to headquarters in order to install or update software. With Citrix XenApp, you can deliver applications and updates instantly anywhere, any time—to offshore employees, outsourcers, new branch offices, new mergers and acquisitions, and mobile workforces.

Enables Automatic Software Updates

When an upgrade or patch needs to be deployed, you would only need to update the Citrix profile on the Citrix XenApp, which will then automatically update all of the instances of that Citrix profile throughout the enterprise. This means that users always have the latest application updates and patches, automatically.

Centralized Application Management Provides Controlled Access and Security

With Citrix XenApp, you can centralize applications and data in secure data centers, which increases data security and ensures fast, reliable performance. Centralized application management using Citrix XenApp provides the following benefits:

- **Enhances security**—Enables you to control, protect, and retain intellectual property centrally to reduce the chance for data loss and theft. Citrix XenApp helps you prevent data from leaving the data center without your explicit permission, which supports regulatory compliance and security objectives. You can provide authorized access to appropriate users—such as employees, customers, and partners—while verifying the ongoing security of the environment.

- **Can provide managed access to applications to users outside of your organization**—You can standardize the use of applications, without having to standardize the machines that the applications use. This enables you to provide managed access to applications from computers that are not your own corporate assets, such as from contractor or consultant computers.
- **Monitors application usage and performance**—Citrix XenApp gives you end-to-end visibility into application usage and performance. It gives IT administrators the power to understand who is using what, how often, and to what extent. They can observe, monitor, measure, audit, report and archive all the dimensions of information flow throughout the computing environment. This enables informed decisions regarding application consolidation and retirement, capacity planning, service level agreements and departmental charge-back
- **Enables identity-driven access**—Citrix XenApp enables you to provide identity-driven access tailored to any user environment. It automatically analyzes the user's permissions and then delivers the appropriate level of access to applications without compromising security. Depending on who and where users are and what device and network they're using, they may be granted different levels of access. You can also easily "decommission" applications by simply turning off a user's permission to it.

Enables User-Based Application Access Rather Than Machine-Based Access

Users can access their applications anywhere on the network, regardless of where they are or what device they are using.

Supported InstallShield Project Types

The **Citrix XenApp** tab is available when one of the following InstallShield project types is open:

- Basic MSI Project
- MSI Database (Direct Edit Mode)
- Transform (Direct MST Mode)

How Transforms are Included in a Citrix Profile

The Citrix Assistant supports the inclusion of transform files with Windows Installer packages in a Citrix profile.

- **How transforms are applied during profile generation**—When building a Citrix profile, transforms that you have specified are automatically applied to the base Windows Installer (`.msi`) package to create a temporary package, and then the Citrix profile is generated from that temporary package.
- **Creating a new transform**—You can create a new transform in InstallShield, and then build a profile from that transform file. When you create a new transform file in InstallShield, you specify the root `.msi` file in the **Open Transform** wizard. The steps you take to generate a profile after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.
- **Converting a Windows Installer package with existing transforms**—If you have a Windows Installer package and one or more existing transform files, and you want to include these transforms in the Citrix profile, you need to open one of the *transforms* in InstallShield (rather than the `.msi` file). The **Open Transform** wizard will open, and you will be prompted to specify the root `.msi` file and which of the existing

`.mst` files you want to include. The steps you take to generate a profile after using that wizard are exactly the same as if you were editing a Windows Installer package in Direct Edit mode.

Caution • *All of the transforms that you add to a Citrix profile must be located in the same folder as the Windows Installer .msi package so that they can be accessed when the profile is built.*

Using the Citrix Assistant to Create a Citrix Profile

The steps you need to take to create a Citrix profile are the following:

Table 11-50 • Steps to Take to Create a Citrix Profile Using the Citrix Assistant

Step #	Description
Step 1	Specifying Citrix Profile Information
Step 2	Specifying Operating System and Language Requirements
Step 3	Managing Files and Folders in a Citrix Profile
Step 4	Setting Isolation Options
Step 5	Modifying Profile Shortcut Settings
Step 6	Modifying Profile Registry Settings
Step 7	Modifying Build Settings
Step 8	Building a Citrix Profile

Specifying Citrix Profile Information

When creating a Citrix profile, you need to specify the **Name**, **Description**, and **Version** of the Citrix profile. You also need to specify whether this package can run executables that are not included with the Citrix profile, and whether to include diagnostic tools with the Citrix profile. The following tasks are performed on the **Profile Information** page of the **Citrix Assistant**:

- Specifying the Profile Name, Description, and Version
- Specifying Whether Users Should Be Able to Update Applications
- Including Diagnostic Tools With a Citrix Profile

Specifying the Profile Name, Description, and Version

On the **Profile Information** page of the Citrix Assistant, you name the Citrix profile, and provide a description and version number.

Task: ***To specify the Citrix profile name, description, and version:***

1. In the **Citrix Assistant**, open the **Profile Information** page.
2. In the **Name** field, enter a name for this Citrix profile. The name you enter here determines the file name of the generated Citrix profile.

   Tip • *Do not include the version number in the profile name.*
3. In the **Description** field, enter a brief explanation of the purpose of this package. This information is stored as package metadata.
4. In the **Version** field, enter the version number of this Citrix profile. This information is stored as package metadata.
5. On the **File** menu, click **Save** to save your changes.

Specifying Whether Users Should Be Able to Update Applications

The **Profile Information** page is where you specify whether users can update applications.

Task: ***To specify whether users should be able to update applications:***

1. In the Citrix Assistant, open the **Profile Information** page.
2. Select or clear the **Enable User Updates (Allow profiled application to update itself—Not recommended)** check box:

   - To allow the profiled application to download and install vendor-supplied updates over the Internet, select this check box. The updates are stored within the user profile root location for the specific user.
   - To ensure that all executable files from the profile are launched from the installation root location, and not from the user profile location, clear this check box. When this check box is cleared, the system prevents code from being run if it is not streamed from the server. Clearing this check box enables you to control updates through the profiler.

   This check box is cleared by default.
3. On the **File** menu, click **Save** to save your changes.

Including Diagnostic Tools With a Citrix Profile

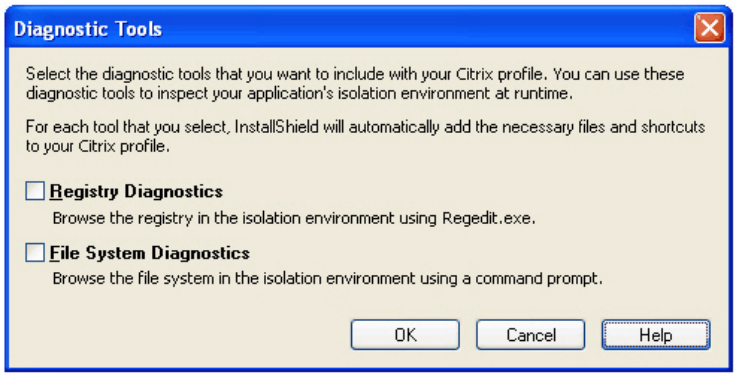
On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **Profile Information** page, you can choose to include the Registry Editor and the Windows Command Prompt diagnostic tools with your Citrix profile.

If you include diagnostic tools with your Citrix profile, you will be able to look at the registry or file system for the application while it is running in its isolation environment. For example, if you were running a Citrix profile and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Caution • *If you choose to include these diagnostic tools, the versions of `regedit.exe` and `cmd.exe` that are part of the operating system on the build machine are added to the Citrix profile. However, these tools may not be compatible with other operating systems.*

Task: ***To include diagnostic tools with a Citrix profile:***

1. In the **Citrix Assistant**, open the **Profile Information** page.
2. In the **More Options** list, click **Diagnostic Tools**. The **Diagnostic Tools** dialog box opens.



3. If you want to include the Registry Editor (regedit.exe) with your Citrix profile so that you can browse the profile registry at runtime from within the isolation environment, select the **Registry Diagnostics** option.
4. If you want to include the Windows Command Prompt application with your Citrix profile so that you can browse the virtual file system at runtime from within the isolation environment, select the **File System Diagnostics** option.

Launching the Diagnostic Tools Within the Isolation Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the profile.

When the user runs this Citrix profile application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's Citrix isolation environment.

Specifying Operating System and Language Requirements

The next step in creating a Citrix profile is to open the **Profile Requirements** page and specify the operating system and language requirements that client workstations need in order to run the application locally.

Some applications can run on multiple operating systems and languages, while others, such as custom applications, might be able to run only on a particular operating system or language. When creating a profile, you need to customize it for the supported operating systems and languages .

Information about specifying operating system and language requirements includes the following topics:

- Setting Operating System Requirements and Service Pack Levels
- Setting Language Requirements
- How Requirements are Applied at Runtime
- Adding Pre-Launch and Post-Exit Scripts

Setting Operating System Requirements and Service Pack Levels

To specify the operating system and service pack level requirements for your application, perform the following steps.

Task: ***To specify operating system requirement and service pack levels:***

1. In the **Citrix Assistant**, open the **Profile Requirements** page.
2. For the **Does your Citrix profile have any specific operating system requirements?** option, select one of the following:
   - **No**—Select this option if this application will run on all of the listed operating systems (which are the operating systems that the Citrix client supports). When this option is selected, the operating system check boxes are locked and cannot be changed.
   - **Yes**—Select this option if the application does not support one of the listed operating systems. When you select this option, the check boxes are unlocked and you can clear the selection of the unsupported operating systems.
3. If you set the previous option to **Yes**, do the following:
   a. Select the operating systems that this application supports, and clear those that this application does not support.

b. For each of the selected operating systems, double-click on it and select **Service Packs Requirement** from the context menu to open the **Service Packs Requirements** dialog box, and choose one of the following options:

- **No Service Pack Requirement**—This application supports all versions of this operating system, regardless of the number of Service Packs installed.
- **No Service Pack Allowed**—This application only supports the initial release of this operating system; if any Service Packs are installed, this application will not run properly.
- **Exact Service Pack Level**—This application requires the installation of a specific Service Pack on this operating system in order to run properly. Enter the required Service Pack Level in the box.
- **At Least Service Pack Level**—To run properly, this application requires that this operating system have at least the specified Service Pack (or higher) installed. Enter the minimum required Service Pack Level in the box.
- **At Most Service Pack Level**—To run properly, this application requires that this operating system have at most the specified Service Pack (or lower) installed. Enter the maximum required Service Pack Level in the box.
- **Range of Service Pack Levels**—To run properly, this application requires that this operating system have a specified range of Service Packs installed. If you select this option, specify the **Minimum Level** and **Maximum Level** in the boxes.

Setting Language Requirements

To specify language requirements for your application, perform the following steps.

Task: ***To specify operating system requirement and service pack levels:***

1. In the **Citrix Assistant**, open the **Profile Requirements** page.
2. For the **Does your Citrix profile have any specific language requirements?** option, select one of the following:
   - **No**—Select this option if this application will run on all of the listed languages (which are the languages that the Citrix client supports). When this option is selected, the language check boxes are locked and cannot be changed.
   - **Yes**—Select this option if the application does not support one of the listed languages. When you select this option, the check boxes are unlocked and only **English** is selected by default.
3. If you selected **Yes** in the previous step, select only those languages that this application supports.

How Requirements are Applied at Runtime

The requirements you specify on the **Profile Requirements** page determine how, or if, a user has access to the application.

When a user attempts to run an application, the Citrix XenApp checks to see whether that user's workstation meets the profile's specified requirements. Then, depending upon the **Application Type** assigned to that profile when it was published on the server, the user is:

- granted access to run the application locally, *or*
- granted access to run the application from the server, *or*
- denied access to the application.

The user access scenarios are presented in the following table:

Table 11-51 • Citrix XenApp User Access Scenarios

Application Type	User Access to Application
Accessed from a server	User runs the application on the Citrix XenApp, using shared server resources.
Streamed if possible, otherwise from a server	User access depends upon whether their workstation meets the profile's specified requirements: • **Meets requirements**—The profile is streamed (copied) to the user's workstation, and the user runs the application locally (from within its isolation environment). • **Does not meet requirements**—User runs the application on the Citrix XenApp, using shared server resources.
Streamed to client	User access depends upon whether their workstation meets the profile's specified requirements: • **Meets requirements**—The profile is streamed (copied) to the user's workstation, and the user runs the application locally (from within its isolation environment). • **Does not meet requirements**—User cannot access the application.

Caution • *If an application has specific operating system or language requirements and you fail to specify them correctly when creating the profile, users who do not meet those requirements will be given access to run applications locally and they will probably encounter application errors.*

Adding Pre-Launch and Post-Exit Scripts

You can choose to include scripts with your profile that must execute either before profile launch or after profile exit in order for your application to run properly. On the **Script Execution** dialog box, which is opened by clicking **Script Execution** in the **More Options** list on the **Profile Requirements** page, you can view and manage all of the **Before Profile Launch** and **After Profile Exit** script files you are including with your Citrix profile.

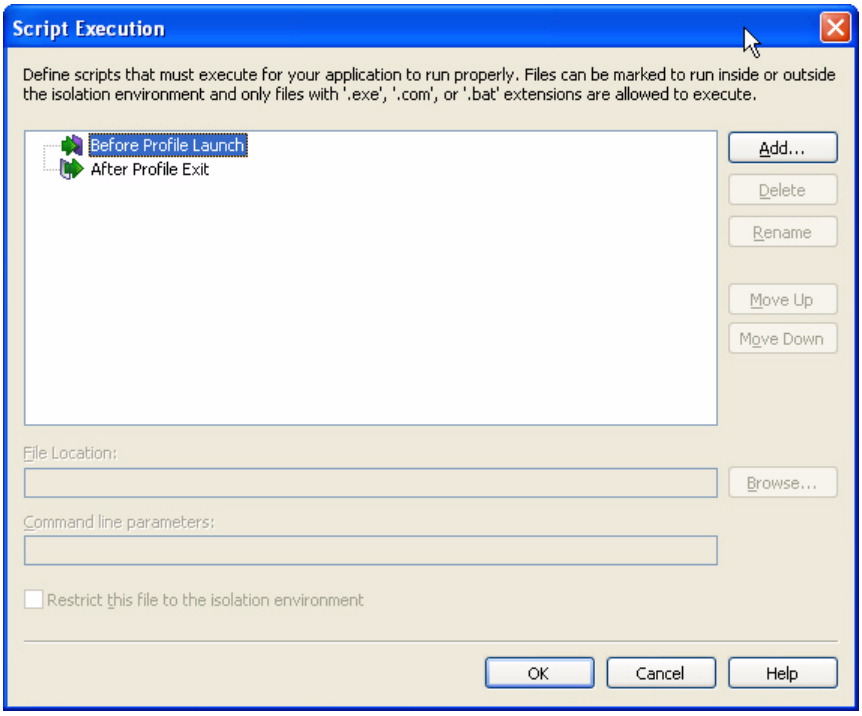
- Files can be marked to run inside or outside of the isolation environment.

- Only files with `.exe`, `.cmd`, `.com`, or `.bat` extensions are allowed to execute.

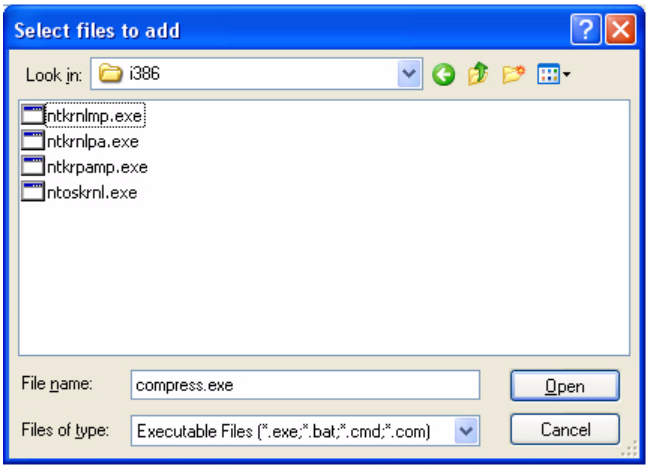
To add a script to your Citrix profile, perform the following steps.

Task: ***To add a before launch or after exit script to your Citrix profile:***

1. Open the **Profile Requirements** page of the Citrix Assistant.
2. In the **More Options** list, click **Script Execution**. The **Script Execution** dialog box opens.



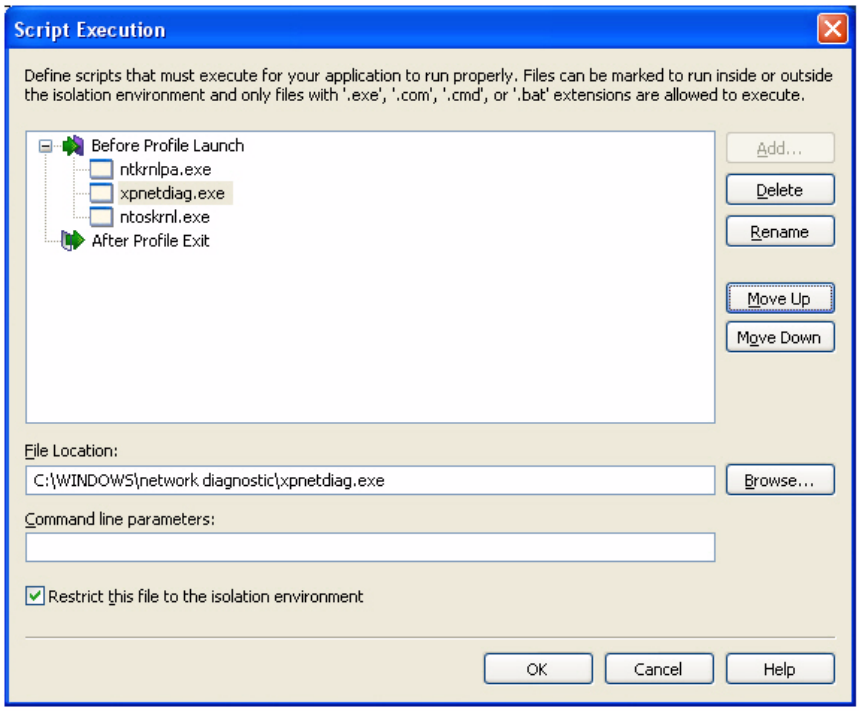
3. Select the **Before Profile Launch** or **After Profile Exit** node in the tree.
4. Click **Add...** The **Select Files to Add** dialog box opens.



5. Select the script file(s) (`.exe`, `.bat`, `.cmd`, or `.com`) that you want to add, and click **Open**. The file is added to the Script Execution tree on under the appropriate node.

Tip • *Use the Shift key to select multiple contiguous files, and use the Ctrl key to select multiple non-contiguous files.*

6. Select a script in the tree. Several new fields and options are enabled.



You can now perform any of the following tasks:

- **Rename the file's display name**—To rename the script file's display name, click the **Rename** button and enter a new name. The name that is displayed on this dialog box to identify the script is changed, but the original name of the script file is not changed.
- **Select a different script**—To select a different script, click the Browse button and select a different script file (`.exe`, `.bat`, `.cmd`, or `.com`).
- **Reorder scripts**—If multiple scripts are listed under a node, you can use the **Move Up** and **Move Down** buttons to change the order that the scripts will be run. You can also reorder the scripts using the Ctrl+Shift+Up Arrow and Ctrl+Shift+Down Arrow keys.
- **Restrict script to isolation environment**—If you want this script to only be able to run within the Citrix profile's isolation environment, select the **Restrict this file to the isolation environment** option.
- **Add command line parameters**—To add command line parameters to run along with the script, enter them in the **Command line parameters** box.
- **Delete a script**—To delete a script from the profile, select it and click the **Delete** button.

7. When you have set all desired options for the script, click **OK**.

Managing Files and Folders in a Citrix Profile

The next step in creating a Citrix profile is to view existing files and folders, add and delete files and folders, and override the default isolation options for folders and files.

The following tasks are performed on the **Profile Files** page.

- Managing Files and Folders in a Citrix Profile
- Controlling the Display of Predefined Folders
- Overview of Citrix Isolation Options

Managing Files and Folders in a Citrix Profile

The directories in the destination tree on the **Profile Files** page of the Citrix Assistant represent how your application will be organized within its isolation environment.

On the **Profile Files** page, you can view all of the files and folders that are currently in your Citrix profile, add new files and folders to include in the Citrix profile, and delete files and folders from the Citrix profile.

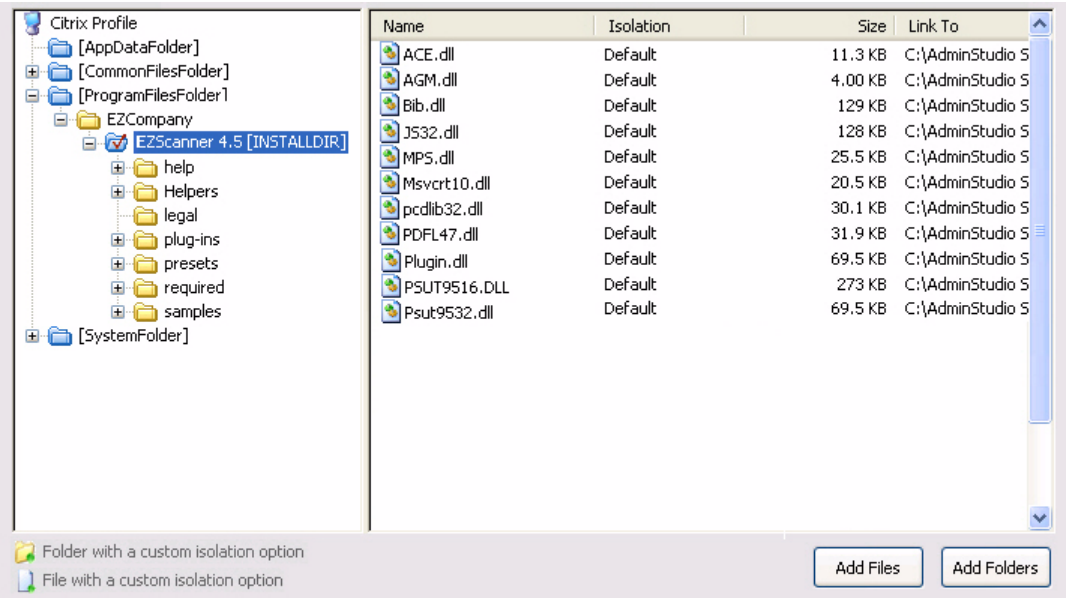
- Adding Files to a Citrix Profile
- Adding an Existing Folder (and its Contents) to a Citrix Profile
- Creating a New Folder
- Moving Files and Folders
- Deleting Files and Folders

Adding Files to a Citrix Profile

To add files to a Citrix profile, perform the following steps:

Task: ***To add a files to a Citrix profile:***

1. In the **Citrix Assistant**, open the **Profile Files** page. The files and folders are listed in the **Citrix Profile** tree, organized by installation directory.



   Folders are listed in the column on the left, and all of the files in the selected folder are listed on the right. Blue folders are the supported MSI standard folders. The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

2. Browse through the folder tree to find the folder that you would like to add files to.
3. Select the folder and click the **Add Files** button. The **Open** dialog box opens.
4. Select the file or files that you want to add and click **Open**. The files you selected are now listed.

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Adding a File by Dragging and Dropping Files From Your System

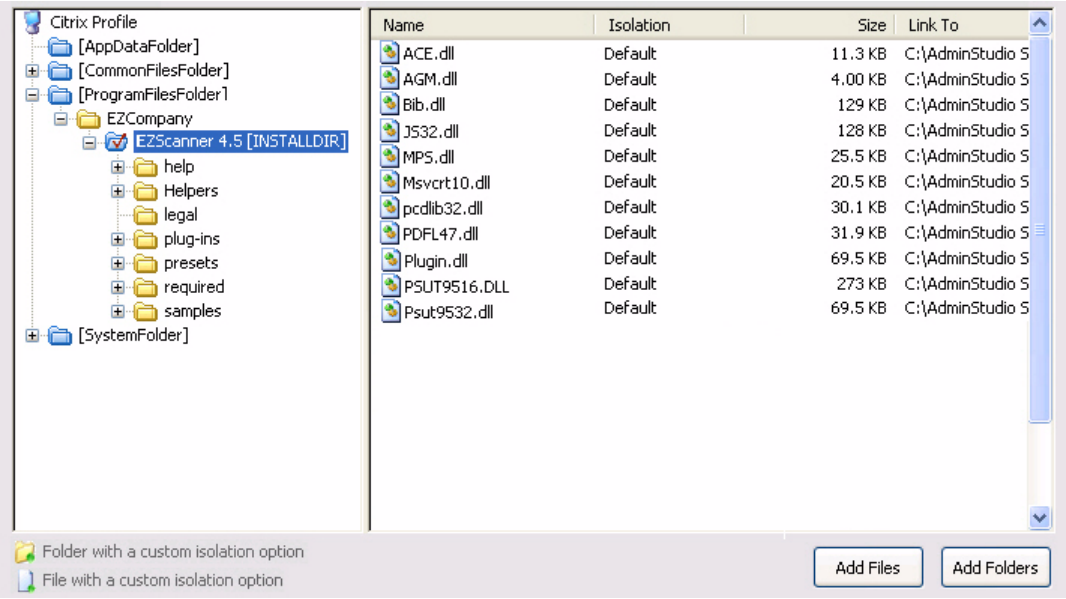
You can also add files or folders to your Citrix profile on the **Profile Files** page by dragging them from a directory on your computer to the desired location in the tree.

Adding an Existing Folder (and its Contents) to a Citrix Profile

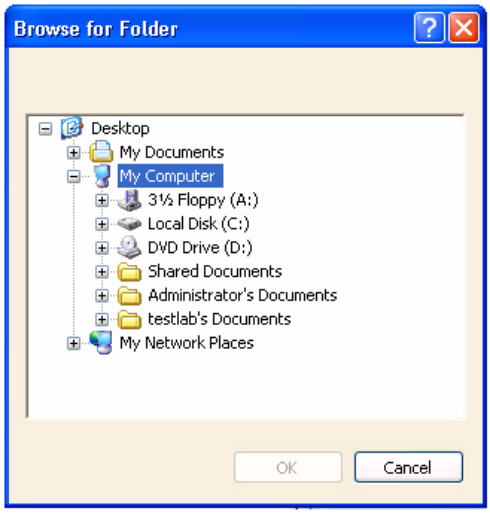
To add an existing folder and all of the files and subfolders within it to a Citrix profile, perform the following steps:

Task: ***To add an existing folder to a Citrix profile:***

1. In the **Citrix Assistant**, open the **Profile Files** page. The files and folders are listed in the **Citrix Profile** tree, organized by installation directory.

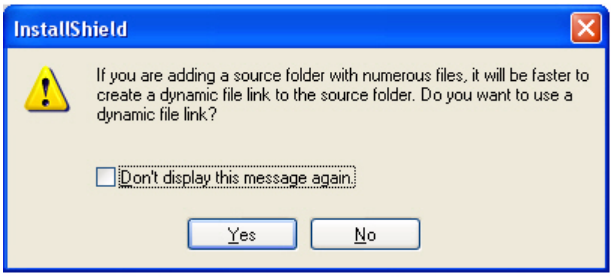


2. Browse through the folder tree to find the folder that you would like to add a folder into.
3. Select the folder and click the **Add Folders** button.The **Browse for Folder** dialog box opens, listing all of the directories available to your computer.



4. Select a folder and click **OK**.

   If you are editing an InstallShield project (not a Windows Installer package), you are prompted to choose whether you want to create a dynamic file link to the source folder.



5. Indicate whether you want to create a dynamic file link by selecting one of the following:

   - **No**—For more flexibility with Citrix options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
   - **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

   The folder that you selected is now listed, along with of the files and folders within it.

Creating a New Folder

You can create a new, empty folder by selecting an existing folder in the tree and selecting **New Folder** from the context menu.

Task: ***To create a new folder:***

1. Right-click on a folder in the **Citrix Profile** tree and select **New Folder**. A new folder is created as a subfolder of the selected folder:

2. Enter a name for the new folder.

Moving Files and Folders

To change the folder's location in the Citrix Profile folder tree structure, perform the following steps:

Task: ***To move a file or folder:***

1. Select the file or folder that you want to move.
2. With the mouse button down, drag the file or folder to the new location.
3. Release the mouse button.

Deleting Files and Folders

To delete a file or a folder (and all of its contents) from a Citrix profile, perform the following steps:

Task: ***To delete a file or folder:***

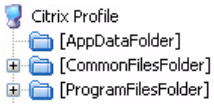
1. Select the file or folder in the **Citrix Profile** tree that you want to delete.
2. Select **Delete** from the context menu. You are prompted to confirm the deletion.
3. Click **Yes**. The selected file or folder is deleted.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains from the entire Project, not just from the Citrix profile.*

Note • *You cannot delete predefined folders. You can only turn off the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Controlling the Display of Predefined Folders

On the **Profile Files** page, the **Citrix Profile** tree initially displays the more commonly used predefined folders, such as `[ProgramFilesFolder]` and `[CommonFilesFolder]`.

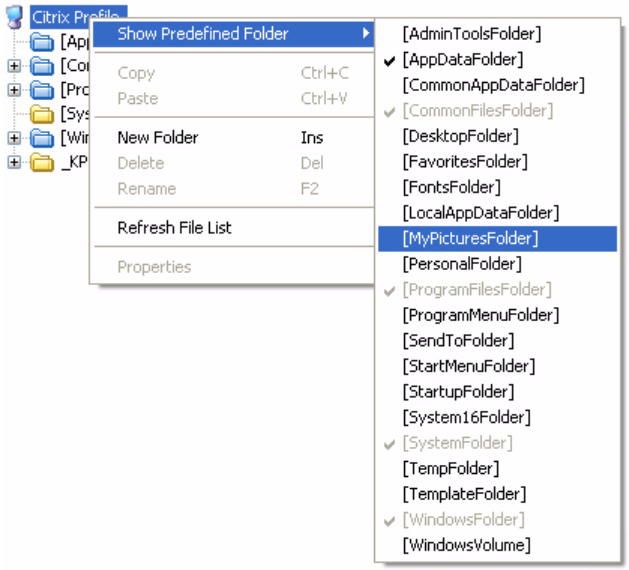


These predefined folders are dynamic, meaning that they do not use hard-coded paths. The value for each destination folder is obtained from the operating system of the target machine.

You can control which predefined folders are listed in this tree.

Task: ***To change which predefined folders are listed:***

1. In the **Citrix Profile** tree, select the **Citrix Profile** node (or any of the files or folders that are listed, point to **Show Predefined Folder**. A list of predefined folders opens.



   Those folders that are already displayed are preceded by a check mark, and those that are not displayed do not have a check mark.

2. To add a folder to the tree listing, select a folder that is not currently listed in the tree.

   Note • *These predefined folders are always added to the root of the Citrix Profile tree, no matter what file or folder you had selected when you selected it from the Predefined Folders list.*

3. To remove a folder from the tree listing, select that folder name in this list (which is preceded by a check mark.

   Note • *You cannot turn off the display of the* `[ProgramFilesFolder]`.

Setting Isolation Options

The Citrix XenApp uses isolation environments to control application compatibility and accessibility. The isolation option that is assigned to a file, folder or registry key specifies how the isolation environment will provide access to system resources requested by the application.

The default settings for isolation options are set on the Citrix XenApp, and those defaults are adequate for most environments. However, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options on the **Isolation Options** dialog box, which is open by selecting a file or folder and then selecting **Isolation Options** from the context menu.

Information about setting isolation options is presented in the following topics:

- Overview of Citrix Isolation Options
- Setting Isolation Options for Folders and Files
- Inheritance of Isolation Options from Folders to Files

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

Overview of Citrix Isolation Options

The Citrix XenApp uses isolation environments to control application compatibility and accessibility. The isolation option that is assigned to a file, folder or registry key specifies how the isolation environment will provide access to system resources requested by the application.

The default settings for isolation environments are set on the Citrix XenApp, and those defaults are adequate for most environments. However, in the Citrix Assistant, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options on the **Isolation Options** dialog box, which is opened by selecting **Isolation Options** on the context menu when you have a file or folder selected on the **Profile Files** page or a registry key selected on the **Profile Registry** page.

On the **Isolation Options** dialog box, you can choose one of the following isolation options:

Table 11-52 • Isolation Options

Option	Description
Default	Select this option if you want the default isolation option for this file/folder/registry key as defined on the Citrix XenApp to be applied to this selection. This is the default selection for all files, folders, and registry keys. ***Caution*** • *You should select this option unless you require specific custom handling.*

Table 11-52 • Isolation Options (cont.)

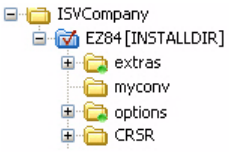
Option	Description
Ignore	Choose the **Ignore** option to direct the isolation environment to *always* use the copy of this selected file/folder/registry key that is on the system, not the one inside the isolation environment. Choosing this option gives the isolation environment direct access to the same location on the system that a non-isolated version of this application would have. By assigning the **Ignore** isolation option, you are creating a "hole" in the isolation environment to allow an application to write to the underlying system. For example, you would select **Ignore** in the following situations: • If an application creates a directory for per-user data that is stored in a non-standard location. • If the workstation has extra drive volumes and an installer writes to those drives while installing into a target. • If your file share volume is on your packaging workstation. • When the Citrix profile needs to share data with an application outside the isolation environment, such as when users print to a network printer.
Isolate	Choose the **Isolate** option to direct the isolation environment to first try to find the copy of this file/folder/registry key that is inside the isolation environment. If the item is not found there, then the isolation environment will use the copy of this file/folder/registry key that is on the system. Selecting **Isolate** ensures that the isolation environment is not given direct write access to the specified system resource.
Strictly Isolate	Choose the **Strictly Isolate** option to direct the isolation environment to always use the copy of this file/folder/registry key that is in the isolation environment, not on the system. This is useful when running two versions of an application on the same machine.
Redirect	Choose the **Redirect** option to redirect a request by the isolation environment for a file/folder/registry key to a specified location on the system (without first searching the user profile root and installation root locations). When selecting this option, you also need to select the location that the isolation environment should redirect to: • **Source**—Lists the name of the selected item (filename, folder name, registry key). • **Destination**—*[Files and folders only]* Click the Browse [...] button and select the file or folder on the system that you want to redirect to. • **Destination Root**—*[Registry keys only]* Select the registry root of the registry key on the system that you want to redirect to. • **Destination Key**—*[Registry keys only]* Select the registry key on the system that you want to redirect to.

Setting Isolation Options for Folders and Files

To override a file or folder's default isolation options set on the Citrix XenApp, perform the following steps:

Task: ***To set an isolation option on a folder or file.***

1. Open the **Profile Files** page.
2. Browse through the folder tree to find the file or folder that you would like to modify.
3. Select the file or folder and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.
4. Select one of the following options:
   - Default
   - Ignore
   - Isolate
   - Strictly Isolate
   - Redirect
5. Click **OK**. Files and folders that have an isolation setting other than default are marked with a special icon:



Inheritance of Isolation Options from Folders to Files

Isolation options for files, folders and registry keys are always inherited. The Citrix isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for `C:\Windows` and one for `C:\Windows\System32`. When the application requests `C:\Windows\System32\Notepad.exe`, then the `C:\Windows\System32` isolation rule will be applied because `C:\Windows\System32` is a more specific reference to `C:\Windows\System32\Notepad.exe` than is `C:\Windows`.

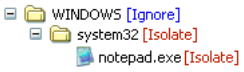


Figure 11-50: Example of Inheritance of Isolation Options from Folders to Files

Modifying Profile Shortcut Settings

You define profile shortcuts to enable users to launch a Citrix profile from within the isolation environment.

By default, the **Citrix Assistant** creates shortcuts to all of the executable (.exe) files that were added to the profile on the **Profile Files** page. These shortcuts are listed in a checklist on the **Profile Shortcuts** page.

Tip • *Citrix currently only supports 16 color icons for shortcuts. Therefore, if you specify an* ***Icon File*** *on the* ***Shortcuts*** *view of the Installation Designer, be sure to select an icon that includes only 16 colors.*

When you select each shortcut, details about it are displayed:

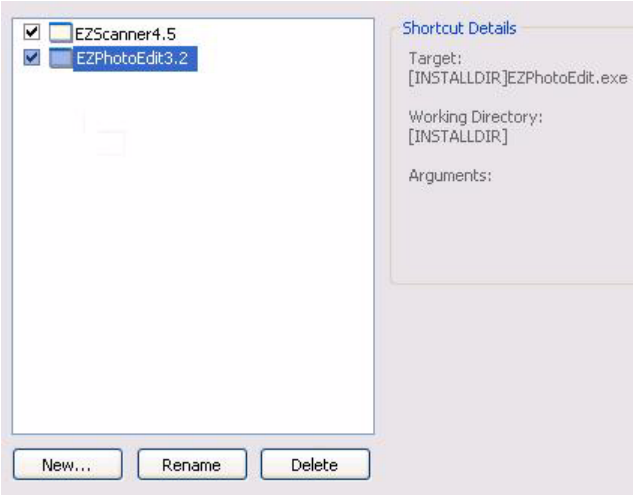


Figure 11-51: Initial List of Shortcuts for an Application

Caution • *You must define at least one shortcut to enable users to launch the application from the isolation environment.*

On the **Profile Shortcuts** page, you can create, delete, include, exclude, or rename a profile's shortcuts.

- Shortcuts and the Isolation Environment
- Shortcut Requirements
- Creating a New Profile Shortcut
- Including an Existing Profile Shortcut
- Excluding vs. Deleting a Profile Shortcut
- Conditions When a Shortcut Should be Excluded or Deleted

Shortcuts and the Isolation Environment

When a profile is published on the Citrix XenApp, the administrator has the option of placing available shortcuts on the client's desktop, client's Start menu, or only in the Citrix Program Neighborhood Agent applications list.

Shortcut presentation is specified in the **Application shortcut placement** area of the **Shortcut presentation** view of the Citrix Access Management Console **Application Properties** dialog box.

In the **Application shortcut placement** area, you have the following options:

Table 11-53 • Shortcut Presentation Options

Option	Description

Add to the client's Start menu	Select this option to create a shortcut to this application in the user's local **Start** menu. A Client Application folder appears in the first pane of the **Start** menu:
	`Start`
	`  MyApplicationFolder`
	`  ApplicationName`
	When you select this option, the **Place under Programs folder** and **Start menu folder** fields are enabled.

• **Place under Programs folder** (Program Neighborhood Agent Only)
• **Start menu folder** (Program Neighborhood Agent Only)
	Select the **Place under Programs folder** option to create a shortcut to this application under the **Programs** folder of the user's local **Start** menu.
	• If you leave the **Start menu folder** field blank, the shortcut is created in root folder of the **Programs** menu.
	`Start`
	`  Programs`
	`  MyApplicationFolder`
	`  ApplicationName`
	• If you specify a folder structure in the **Start menu folder** field, the shortcut is created in that folder structure within the local **Programs** folder, with each folder name separated with a backslash. For example, if you entered the following in the **Start menu folder** field:
	`MyApplicationFolder/ApplicationTools`
	Then, the shortcut would be created in the following folder structure:
	`Start`
	`  Programs`
	`  MyApplicationFolder`
	`  ApplicationTools`
	`  ApplicationName`

Add shortcut to the client's desktop	Select this option to create a shortcut to this application on the user's local desktop.

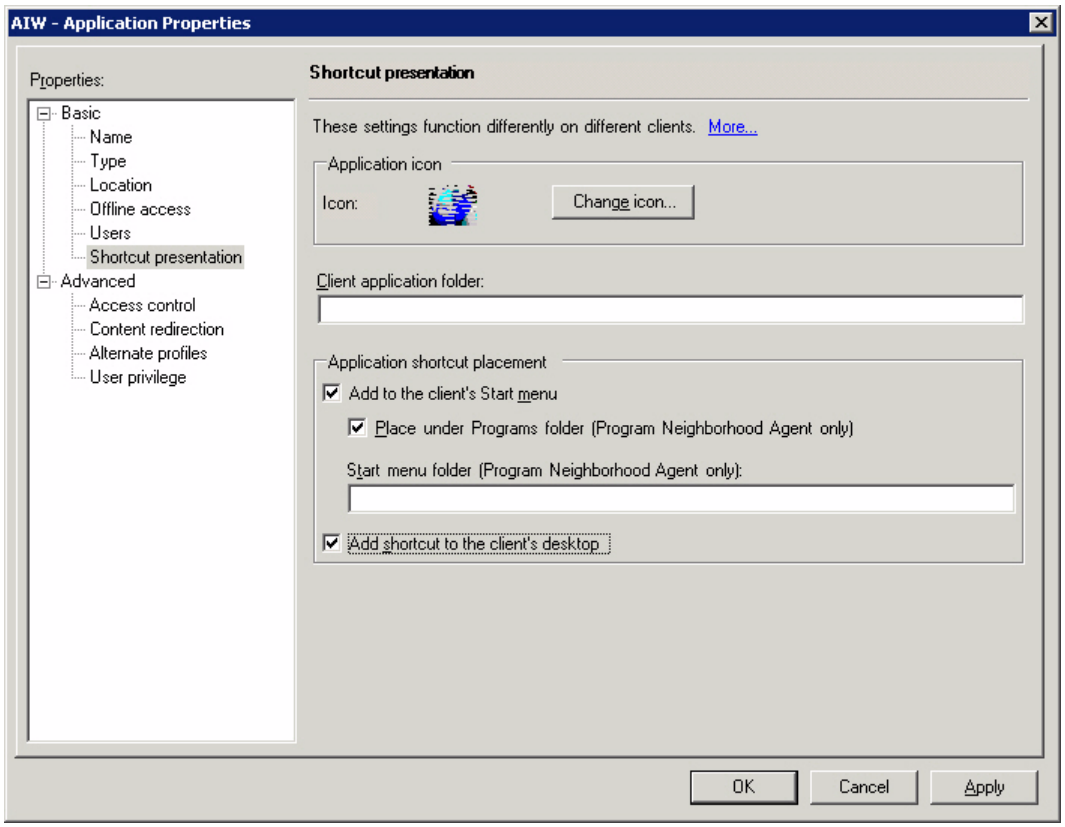


Figure 11-52: Shortcut Presentation View of the Citrix Presentation Server Application Properties Dialog Box

Shortcut Requirements

For each Citrix profile, you are required to define **at least one** shortcut. Profile shortcuts enable users to access the isolation environment and launch the application. If you build a Citrix profile that does not contain any shortcuts, users will not be able to launch the application.

Creating a New Profile Shortcut

On the **Profile Shortcuts** page, you can add a new shortcut to a file within the Citrix profile.

Task: ***To create a new shortcut:***

1. Open the **Profile Shortcuts** page. All of the shortcuts are listed:
   - Those that are currently included in the profile are selected.
   - Those that are currently excluded from the profile are not selected.
2. Click **New**. The **Browse for a Shortcut Target File** dialog box opens and prompts you to select a file within this profile.

3. Select the file that you want to create a shortcut to.
4. Click **Open**. A new shortcut is listed, and it is named the same name as the selected file.
5. To include this shortcut in the Citrix profile, make sure that its check box is selected.

Including an Existing Profile Shortcut

If you want to include a previously excluded shortcut in a Citrix profile, perform the following steps:

Task: ***To include an existing profile shortcut:***

1. Open the **Profile Shortcuts** page. All of the executable (`.exe`) files that were added on the **Profile Files** page are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To include a previously excluded shortcut, select the shortcut and select the check box.

Excluding vs. Deleting a Profile Shortcut

By default, the **Citrix Assistant** creates shortcuts to all of the executable (`.exe`) files that were added on the **Profile Files** page, and lists them in a checklist on the **Profile Shortcuts** page.

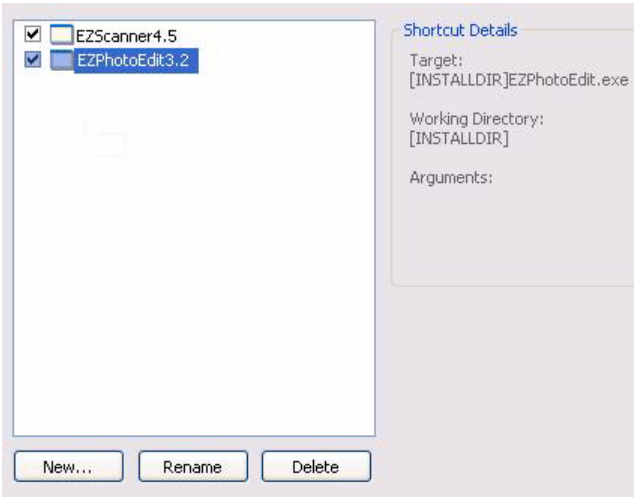


Figure 11-53: Initial List of Shortcuts for an Application

To prevent the shortcut from being created in the Citrix profile, you can choose to either delete or exclude it.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project. See Excluding a Profile Shortcut.

- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the Citrix profile and the InstallShield project. See Deleting a Shortcut.

If you have any unnecessary shortcuts in your project, you can simply exclude them from the Profile by unchecking them in the shortcuts list. If you like to permanently remove a shortcut, you can delete it from the shortcut list.

Excluding a Profile Shortcut

If you want to exclude one of these shortcuts from being created in the Citrix profile, perform the following steps:

Task: ***To exclude a shortcut:***

1. Open the **Profile Shortcuts** page. All of the executable (`.exe`) files that were added on the **Profile Files** page are listed.
   - Those that are currently included are selected.
   - Those that are currently excluded are not selected.
2. To exclude a shortcut, select the shortcut and clear the check box.

Note • *When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project.*

Deleting a Shortcut

To delete a shortcut, perform the following steps.

Task: ***To delete a shortcut:***

1. Open the **Profile Shortcuts** page. All of the shortcuts are listed.
2. Select the shortcut and click **Delete**.

Note • *If you delete a shortcut on the **Profile Shortcuts** page, the shortcut is also deleted from the InstallShield project, and, subsequently, from the Windows Installer package.*

Conditions When a Shortcut Should be Excluded or Deleted

To prevent a shortcut from being created in the Citrix profile, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a Profile Shortcut.

- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the Citrix profile and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a Shortcut.

Renaming a Shortcut

To rename a shortcut, perform the following steps:

Task: ***To add or delete a shortcut:***

1. Open the **Profile Shortcuts** page. All of the executable (`.exe`) files that were added on the **Profile Files** page are listed.
2. Select the shortcut that you want to rename and click **Rename**. A box appears around the shortcut name, and the shortcut name becomes an editable field.
3. Enter a new name for the shortcut.

Modifying Profile Registry Settings

Using the **Citrix Assistant**, you can add, delete, or modify the registry settings in your Citrix profile.

You can also override the Citrix default isolation options for selected registry keys. Isolation options specify how the isolation environment will provide access to system resources requested by the application.

Information about modifying profile registry settings on the **Profile Registry** page includes the following topics:

- About the Windows Registry
- Adding or Deleting Registry Keys and Values
- Setting Registry Isolation Options

About the Windows Registry

The Windows registry is a system-wide database that contains configuration information used by applications and the operating system. The registry stores all kinds of information, including the following:

- Application information such as company name, product name, and version number
- Path information that enables your application to run
- Uninstallation information that enables end users to uninstall the application easily without interfering with other applications on the system
- System-wide file associations for documents created by an application
- License information
- Default settings for application options such as window positions

Keys, Value Names, and Values

The registry consists of a set of keys arranged hierarchically under the My Computer explorer. Just under My Computer are several root keys. An installation can add keys and values to any root key of the registry. The root keys that are typically affected by installations are:

- HKEY_LOCAL_MACHINE
- HKEY_USERS
- HKEY_CURRENT_USER
- HKEY_CLASSES_ROOT

A key is a named location in the registry. A key can contain a subkey, a value name and value pair, and a default (unnamed) value. A value name and value pair is a two-part data structure under a key. The value name identifies a value for storage under a key, and the value is the actual data associated with a value name. When a value name is unspecified for a value, that value is the default value for that key. Each key can have only one default (unnamed) value.

Note that the terms key and subkey are relative. In the registry, a key that is below another key can be referred to as a subkey or as a key, depending on how you want to refer to it relative to another key in the registry hierarchy.

Adding or Deleting Registry Keys and Values

Editing the registry on the **Profile Registry** page is performed much like it is performed on the InstallShield **Registry View**. See Editing the Registry.

Setting Registry Isolation Options

To override a registry key's default isolation options set on the Citrix XenApp, perform the following steps:

Task: ***To set an isolation option on a registry key:***

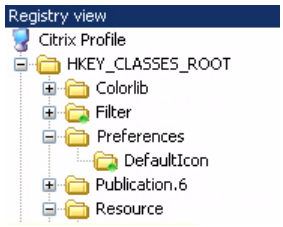
1. Open the **Profile Files** page.
2. Browse through the registry tree to find the key that you would like to modify.
3. Select the folder or key and click **Isolation Options** on the context menu. The **Isolation Options** dialog box opens.

   Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

4. Select one of the following options:
   - Default
   - Ignore

- Isolate
- Strictly Isolate
- Redirect

5. Click **OK**. Registry keys that have an isolation setting other than default are marked with a special icon:



Tip • *To import an existing registry (.reg) file, click the* ***Import a .reg file*** *option on the* ***More Options*** *list to open the Registry Import Wizard.*

Inheritance of Isolation Options in the Registry

Isolation options for registry keys are always inherited. The Citrix isolation environment will apply the most specific reference to that resource.

For example, suppose you have an isolation option for the `Microsoft` registry key and one for `Microsoft\Windows` registry key. When the application requests `Microsoft\Windows\CurrentVersion`, then the `Microsoft\Windows` isolation rule will be applied because `Microsoft\Windows` is a more specific reference to `Microsoft\Windows\CurrentVersion` than is `Microsoft`.

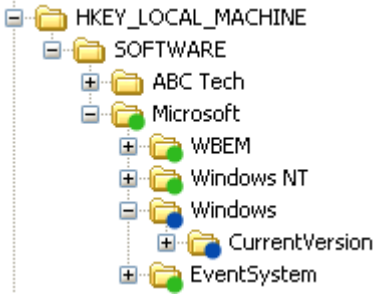


Figure 11-54: Example of Inheritance of Isolation Options from Folders to Files

Modifying Build Settings

On the **Build Settings** page, you choose which releases of this InstallShield project you want to build a Citrix profile for when the project is built, specify whether you want to digitally sign the Citrix profile, and specify whether you want to include additional Windows Installer packages in the Citrix profile.

Also, if you are editing a Windows Installer package in Direct Edit mode (or Direct MST mode), you need to select the **Build Citrix Profile** option on the **Build Settings** page before you will be able to build a Citrix profile for that Windows Installer package.

- Selecting Releases to Build
- Digitally Signing a Citrix Profile
- Including Additional Windows Installer Packages in a Citrix Profile
- Enabling Citrix Profile Building When in Direct Edit Mode

Important *• You must create at least one Release (on the **Releases** view of the Installation Designer) before you will be able to select a Release on the **Build Settings** page.*

Selecting Releases to Build

You select the releases that you want to build a Citrix profile for on the **Releases** tree of the **Build Settings** page.

Important *• You cannot create or edit a release in the Citrix Assistant. If no releases exist, you can simply click the **Build** toolbar button to create a new release or open the **Releases** view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a Citrix profile. For more information, see Creating and Building Releases.*

*If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Settings** page is not displayed.*

Task: ***To select releases to build:***

1. Open the **Build Settings** page.
2. Select the releases in the **Releases** tree that you want to build a Citrix profile for.

   Important *• When you select a release on the **Build Settings** page, you are specifying that whenever you build that particular release, you want to also build a Citrix profile for that release. However, the releases that are selected on the **Build Settings** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Settings** page:*

   - ***Active release selected****—A Windows Installer package and a Citrix profile would be built.*
   - ***Active release not selected****—Only a Windows Installer package would be built.*

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

Digitally Signing a Citrix Profile

You can digitally sign your Citrix profile to assure end users that neither your installation nor the code within your application has been tampered with or altered since publication. When you digitally sign your application, end users are presented with a digital certificate when they run your installation.

Task: ***To digitally sign a Citrix profile:***

1. Open the **Build Settings** page.
2. Select the **Digitally sign Citrix profile** option. The **Personal Information Exchange file (.pfx)** field is enabled. A `.pfx` file is a standard file format for digital certificates.
3. Click **Browse** and select the `.pfx` file that you want to use to digitally sign this Citrix profile.

Including Additional Windows Installer Packages in a Citrix Profile

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual application, you can also use the Citrix Assistant to convert an application suite of multiple Windows Installer packages into one virtual application.

To include additional Windows Installer packages in a Citrix profile, set the **Would you like to include additional MSI files in the virtual package?** option on the **Build Settings** page to **Yes**, and then select the packages that you want to add.

Task: ***To include additional Windows installer packages in a Citrix profile:***

1. Open the **Build Settings** page.
2. Set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**.
3. Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
   - The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
   - Use the Delete button ( ) to delete a package from the list.

Enabling Citrix Profile Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **Citrix Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a Citrix profile for this Windows Installer package. To do this, perform the following steps:

Task: ***To enable Citrix profile building when in Direct Edit Mode:***

1. Open a Windows Installer package or a transform file in InstallShield. It will be opened in Direct Edit Mode or Direct MST Mode, and the Build function (**Build** on the **Build** menu and the **Build** toolbar button) will be disabled.
2. Open the **Build Settings** page of the Citrix Assistant.
3. Select the **Build Citrix Profile** option. After you select this option, the **Build Citrix Profile** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Building a Citrix Profile

The method for building a Citrix profile depends upon what file you have open—an InstallShield project or a Windows Installer package.

- Building a Citrix Profile for an InstallShield Project
- Building a Citrix Profile for a Windows Installer Package

Building a Citrix Profile for an InstallShield Project

To build a Citrix profile for an InstallShield project, perform the following steps:

Task: ***To build a Citrix profile for an InstallShield project:***

1. Open the InstallShield project in InstallShield.
2. On the **Releases** view of the Installation Designer, make sure that at least one release has been created, and select the release that you want to build.

   Important • *You cannot create or edit a release in the Citrix Assistant. If no releases exist, or if you want to create a new release, open the* ***Releases*** *view of the InstallShield Installation Designer. You must create at least one release before you will be able to build a Citrix profile. For more information, see Creating and Building Releases.*
3. Open the **Build Settings** page of the Citrix Assistant.

4. In the **Releases** tree, select the same release that is selected on the **Releases** view of the InstallShield Installation Designer. This is the release that you will build a Citrix profile for.

   Important • *When you select a release on the **Build Settings** page, you are specifying that whenever you build that particular release, you want to also build a Citrix profile for that release. However, the releases that are selected on the **Build Settings** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what was selected on the **Build Settings** page:*

   - ***Active release selected**—A Windows Installer package and a Citrix profile would be built.*
   - ***Active release not selected**—Only a Windows Installer package would be built.*

   To build more than one release at a time, perform a batch build. See Performing Batch Builds.

5. Click the **Build** toolbar button (or select **Build Release** on the **Build** menu) to start building the active release.

   The output of the build will be a Windows Installer package and a Citrix profile. For a description of the files that comprise a Citrix profile, see About Citrix Profiles.

Building a Citrix Profile for a Windows Installer Package

To build a Citrix profile for a Windows Installer package, perform the following steps:

Task: ***To build a Citrix profile for a Windows Installer package:***

1. Do one of the following to open a Windows Installer package:
   - On the **File** menu, select **Open** and select a Windows Installer package (`.msi`).
   - On the **File** menu, select **Open** and select a transform file (`.mst`). The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
   - On the **File** menu, select **New** to open the **New Project** dialog box. Select **Transform** and click **OK**. The **Open Transform Wizard** opens and you are prompted to identify the transform file's associated Windows Installer package.
2. Use the Installation Designer to make any desired edits to the Windows Installer package or Transform file, and use the Citrix Assistant to set Citrix profile options.
3. On the **Build Settings** page of the Citrix Assistant, select the **Build Citrix profile** option.
4. Save the edits to the Windows Installer package or transform file by selecting **Save** on the **File** menu.
5. Click the **Build** toolbar button (or select **Build Citrix Profile** on the **Build** menu) to start building the Citrix profile.

   The output of the build will be a Windows Installer package and a Citrix profile. For a description of the files that comprise a Citrix profile, see About Citrix Profiles.

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

Citrix Assistant Reference

Reference information about the Citrix Assistant is organized into the following sections:

- Pages
- Dialog Boxes
- Building Citrix Profiles Using the Command Line
- Citrix Profile Conversion Error and Warning Messages
- Application Features Requiring Pre- or Post-Conversion Actions

Pages

The Citrix Assistant is comprised of the following pages:

- Home Page
- Profile Information Page
- Profile Requirements Page
- Profile Files Page
- Profile Shortcuts Page
- Profile Registry Page
- Build Settings Page

Home Page

The Citrix Assistant Home page displays a diagram that illustrates the process of creating a Citrix profile for deployment on Citrix XenApp.

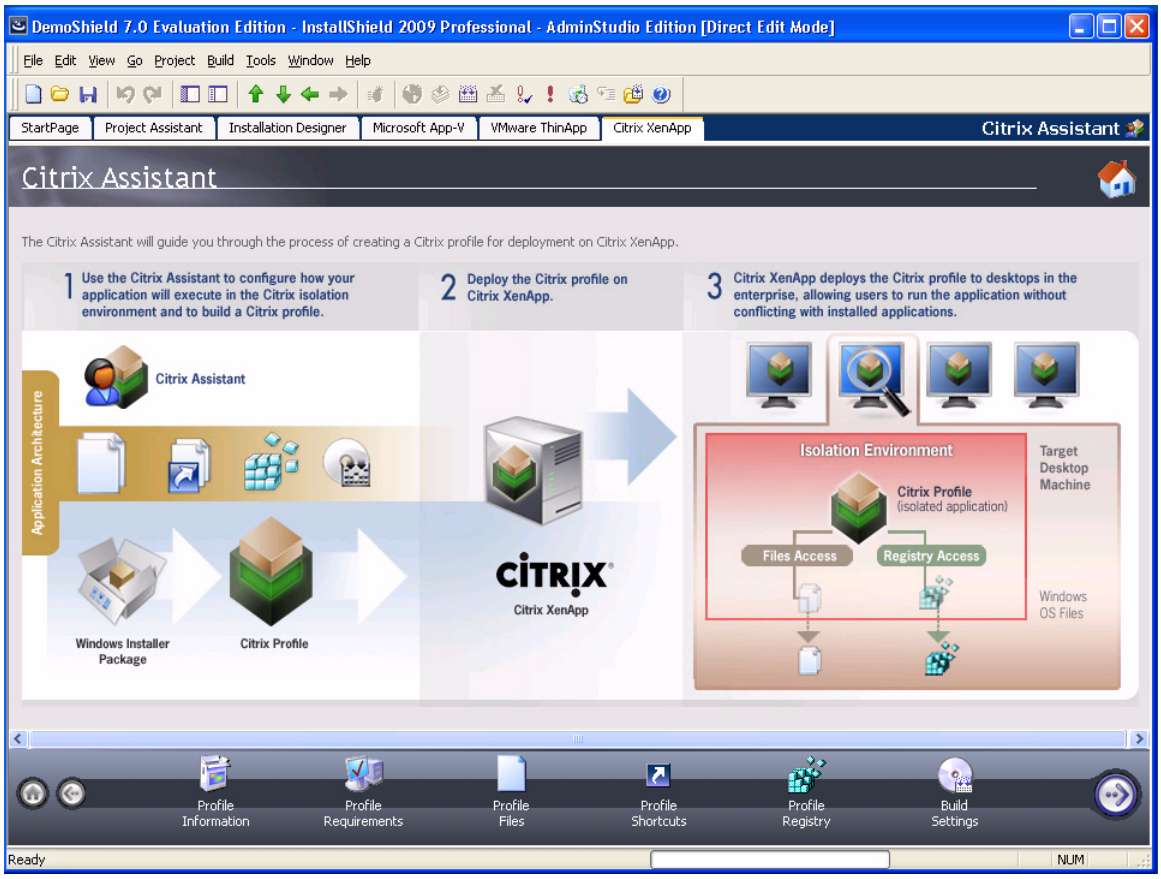


Figure 11-55: Citrix Assistant Home Page

Click the following icons in the navigation bar at the bottom of the page to navigate through the Citrix Assistant interface:

Table 11-54 • Navigation Bar Icons

Icon	Destination
	Profile Information Page
	Profile Requirements Page
	Profile Files Page
	Profile Shortcuts Page

Table 11-54 • Navigation Bar Icons

Icon	Destination
	Profile Registry Page
	Build Settings Page
	Go to next page.
	Jump back to previous page.
	Home Page

Profile Information Page

On the **Profile Information** page in the Citrix Assistant, you specify the **Name**, **Description**, and **Version** of the Citrix profile you are creating. This page is also where you specify whether users can update applications.

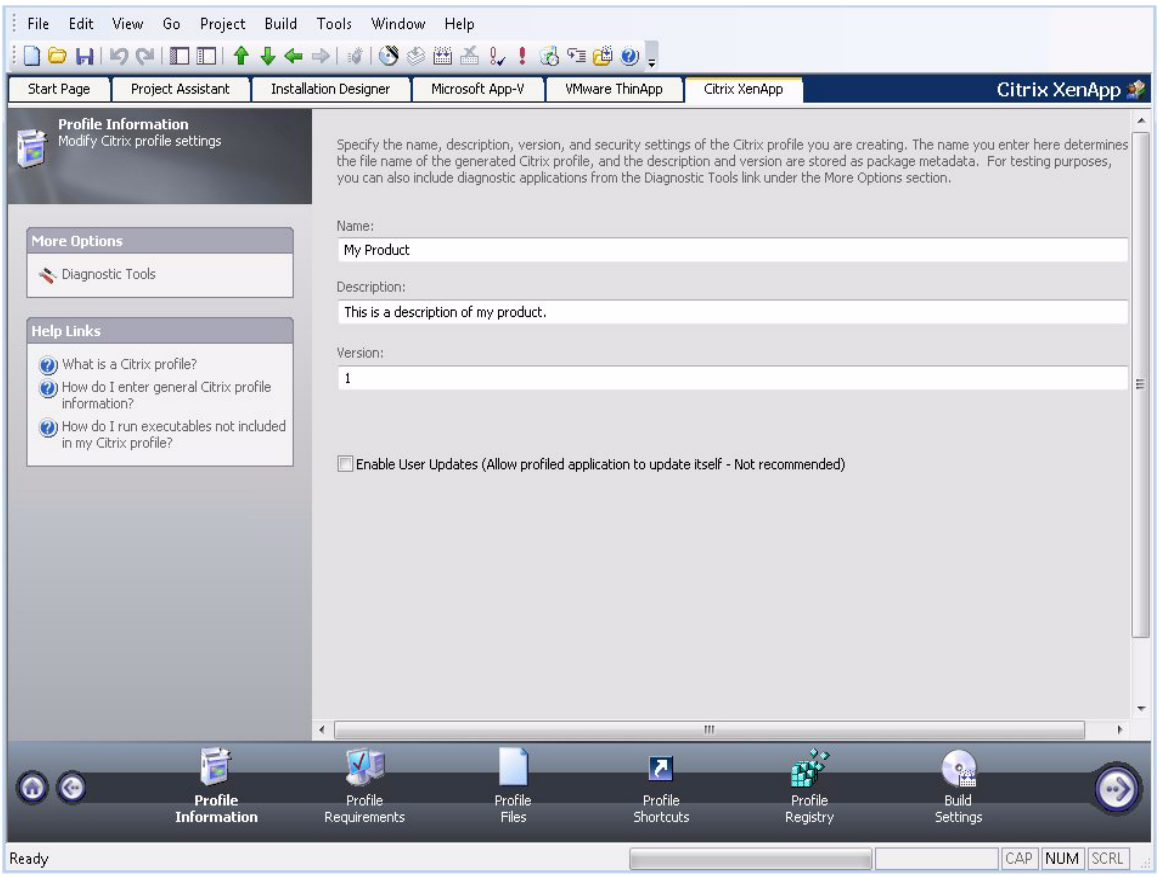


Figure 11-56: Citrix Assistant Profile Information Page

The Profile Information page includes the following options:

Table 11-55 • Profile Information Page

Option	Description
Name	Enter a name for this Citrix profile. The name you enter here determines the file name of the generated Citrix profile. ***Note*** • *Do not include the version number in the package name.*
Description	Briefly describe this package. This information is stored with the package as metadata.
Version	Enter the version number of this Citrix profile. This information is stored as package metadata.

Table 11-55 • Profile Information Page (cont.)

Option	Description
Enable User Updates (Allow profiled application to update itself—Not recommended)	To allow the profiled application to download and install vendor-supplied updates over the Internet, select this check box. The updates are stored within the user profile root location for the specific user. To ensure that all executable files from the profile are launched from the installation root location, and not from the user profile location, clear this check box. When this check box is cleared, the system prevents code from being run if it is not streamed from the server. Clearing this check box enables you to control updates through the profiler. This check box is cleared by default.

For testing purposes, you can also choose to include diagnostic tools in your Citrix profile by selecting the **Diagnostic Tools** link in the **More Options** list. For more information, see Diagnostic Tools Dialog Box.

Profile Requirements Page

On the **Profile Requirements** page of the Citrix Assistant, you can select the operating systems that client workstations must be running in order for your application to operate properly. By default, all operating systems that are supported by the Citrix client are selected.

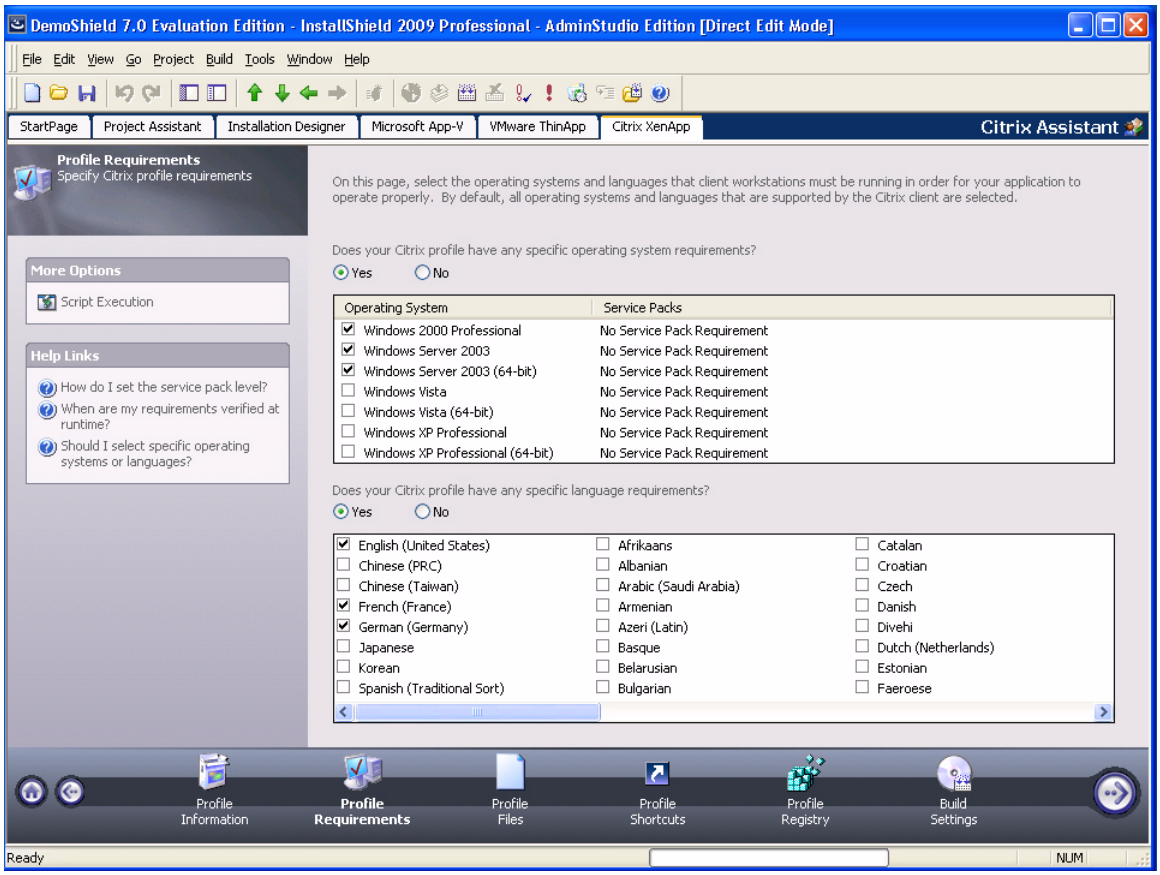


Figure 11-57: Citrix Assistant Profile Requirements Page

The **Profile Requirements** page has the following options:

Table 11-56 • Profile Requirements Page Options

Option	Description
Does your Citrix profile have any specific operating system requirements?	Select one of the following: • **Yes**—Select this option if the application does not support one of the listed operating systems. When you select this option, the check boxes are unlocked and you can clear the selection of the unsupported operating systems. • **No**—Select this option if this application will run on all of the listed operating systems (which are the operating systems that the Citrix client supports). When this option is selected, the operating system check boxes are locked and cannot be changed.

Table 11-56 • Profile Requirements Page Options

Option	Description
Operating System / Service Packs List	If you set the **Does your Citrix profile have any specific operating system requirements?** option to **Yes**, this list becomes enabled. To specify operating system requirements, first select the operating systems that this application supports, and clear those that this application does not support. Then, for each of the selected operating systems, right-click on it and select **Service Packs Requirement** from the context menu to open the **Service Packs Requirements** dialog box, and choose one of the following options: • **Not Required**—This application supports all versions of this operating system, regardless of the number of Service Packs installed. • **No Service Packs Allowed**—This application only supports the initial release of this operating system; if any Service Packs are installed, this application will not run properly. • **Exact Service Pack Level**—This application requires the installation of a specific Service Pack on this operating system in order to run properly. Enter the required Service Pack Level in the box. • **At least Service Pack Level**—To run properly, this application requires that this operating system have at least the specified Service Pack (or higher) installed. Enter the minimum required Service Pack Level in the box. • **At most Service Pack Level**—To run properly, this application requires that this operating system have at most the specified Service Pack (or lower) installed. Enter the maximum required Service Pack Level in the box. • **Range of Service Pack Levels**—To run properly, this application requires that this operating system have a specified range of Service Packs installed. If you select this option, specify the Minimum Level and Maximum Level in the boxes.
Does your Citrix profile have any specific language requirements?	Select one of the following: • **No**—Select this option if this application will run on all of the listed languages (which are the languages that the Citrix client supports). When this option is selected, the language check boxes are locked and cannot be changed. • **Yes**—Select this option if the application does not support one of the listed languages. When you select this option, the check boxes are unlocked and only **English** is selected by default.

Table 11-56 • Profile Requirements Page Options

Option	Description
Language List	If you set the **Does your Citrix profile have any specific language requirements?** option to **Yes**, this list becomes enabled. Select only those languages that this application supports.

Profile Files Page

On the **Profile Files** page of the Citrix Assistant, you can perform the following tasks:

- View Files and Folders
- Add Files and Folders
- Delete Files and Folders
- Set Isolation Options
- Modifying the Display of Predefined Folders

View Files and Folders

On the **Profile Files** page, you can view all of the files and folders that are currently in your Citrix profile.

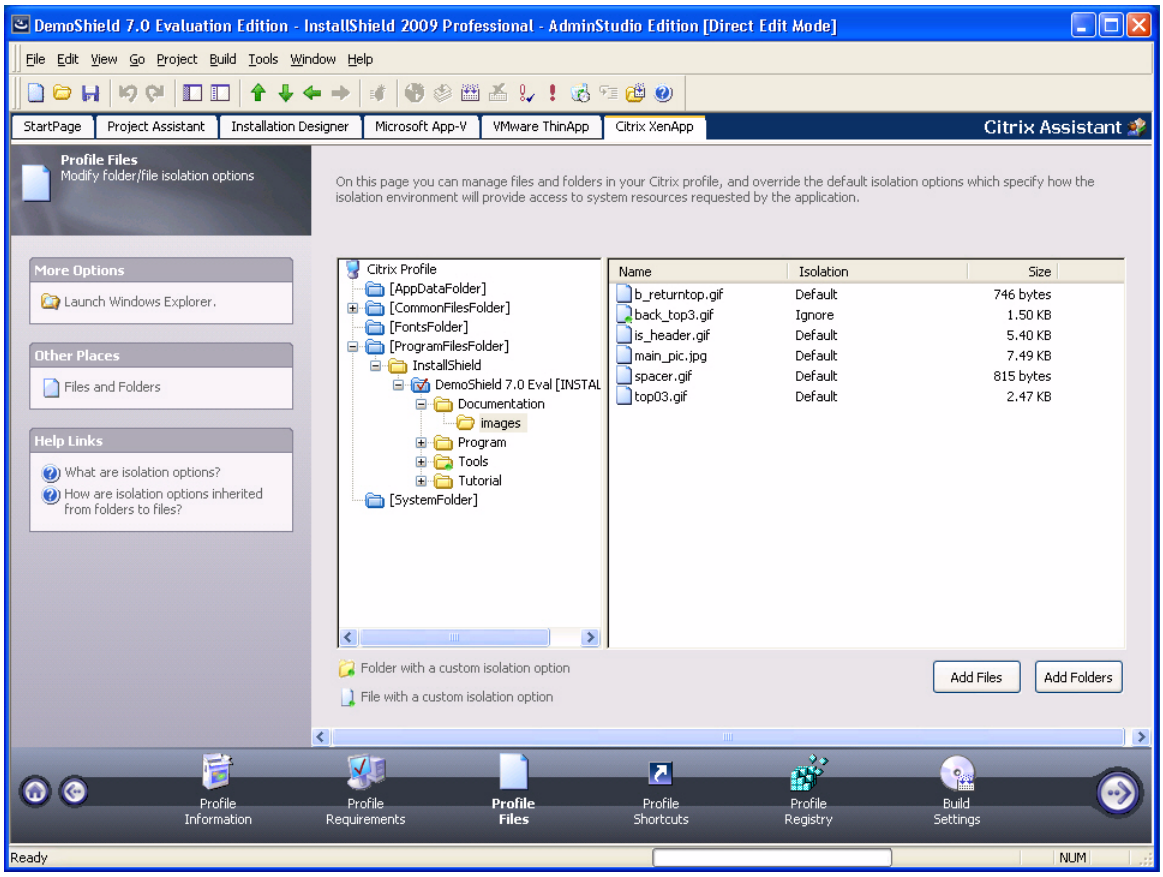


Figure 11-58: Citrix Assistant Profile Files Page

Folders are listed in the Citrix Profile tree on the left, and all of the files in the selected folder are listed on the right.

- The directories in the tree represent how your application will look when it is installed on to your customer's machine.
- Blue folders are the supported MSI standard folders.
- The folder with the check mark is `INSTALLDIR`, which represents the main product installation directory.

Add Files and Folders

On the **Profile Files** page, you can use the **Add Files** and **Add Folders** buttons to add new files and folders to include in the Citrix profile. See Managing Files and Folders in a Citrix Profile.

If you are editing an InstallShield project (not a Windows Installer package), and you are adding a folder to this profile, you are prompted to choose whether you want to create a dynamic file link to the source folder.

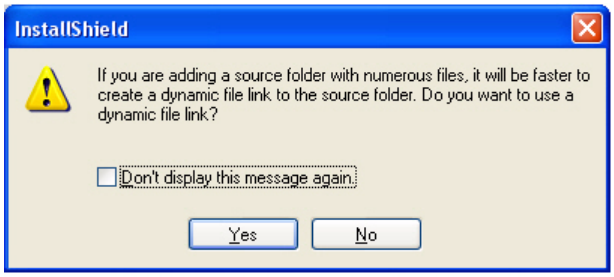


Figure 11-59: Dynamic File Link Option Dialog Box

Indicate whether you want to create a dynamic file link by selecting one of the following:

- **No**—For more flexibility with Citrix options, it is recommended that you select **No** to indicate that you *do not* want to use a dynamic file link, because you would then not be able to customize isolation options for any of the items in this folder.
- **Yes**—If you wish to use the default isolation options for all the files and folders under this folder, then select the dynamic file link option by clicking **Yes**. The **Dynamic File Link Settings** dialog box would then open, prompting you to specify the source folder for your dynamic link, and to set options regarding which files and folders to include in the dynamic link. See Dynamic File Link Settings Dialog Box.

Delete Files and Folders

You can delete files and folders from the Citrix profile by selecting the file or folder you want to delete, and selecting **Delete** from the context menu. For more information, see Deleting Files and Folders.

Caution • *If you choose to delete a folder, you are also deleting all of the files and subfolders that the folder contains.*

Note • *You cannot delete predefined folders. You can only turn of the display of those folders. For more information, see Controlling the Display of Predefined Folders.*

Tip • *To select multiple files, use the Shift key (for contiguous files) or the Ctrl key (for non-contiguous files).*

Set Isolation Options

The Citrix XenApp uses isolation environments to control application compatibility and accessibility. The isolation option that is assigned to a file, folder or registry key specifies how the isolation environment will provide access to system resources requested by the application.

The default settings for isolation environments are set on the Citrix XenApp, and those defaults are adequate for most environments. However, in the Citrix Assistant, you can override the default settings for selected files, folders, or registry keys to exert control over application interactions with client operating system resources.

You set isolation options by selecting a file or folder and then selecting **Isolation Options** from the context menu. For an overview of the available isolation options, and for instructions on how to set them, see Setting Isolation Options.

Modifying the Display of Predefined Folders

You can specify which of the Windows Installer predefined folders are listed in the **Citrix Profile** tree. See Controlling the Display of Predefined Folders.

Profile Shortcuts Page

You define profile shortcuts to enable users to launch a Citrix profile from within the isolation environment.

By default, the **Citrix Assistant** creates shortcuts to all of the executable (`.exe`) files that were added to the profile. These shortcuts are listed in a checklist on the **Profile Shortcuts** page.

Note • *Only shortcuts to executables are included in the profile. The Citrix Administrator chooses which of these shortcuts will be available to their users. When publishing, the Citrix Administrator chooses where to place the shortcut for their users to see.*

Tip • *Citrix currently only supports 16 color icons for shortcuts. Therefore, if you specify an* ***Icon File*** *on the* ***Shortcuts*** *view of the Installation Designer, be sure to select an icon that includes only 16 colors.*

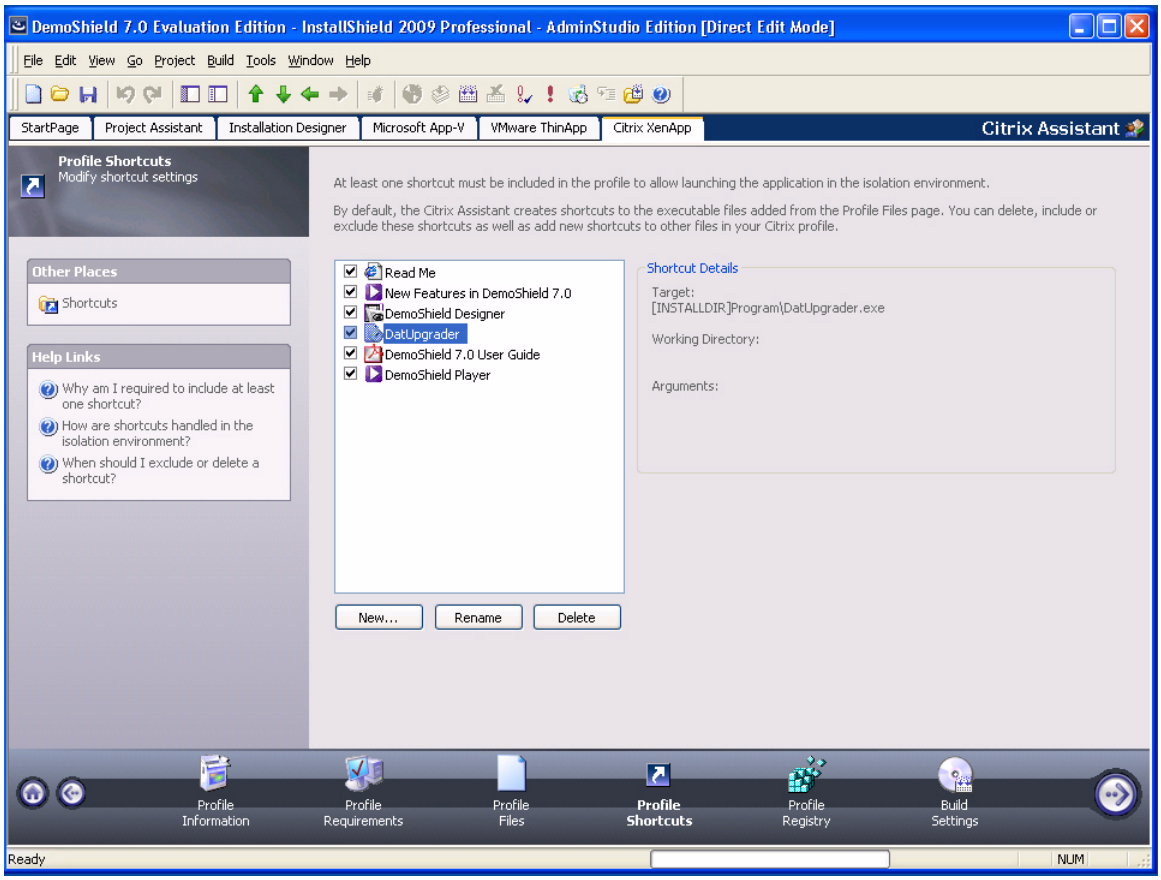


Figure 11-60: Profile Shortcuts Page

Shortcut Requirements

For each Citrix profile, you are required to define **at least one** shortcut. Profile shortcuts enable users to access the isolation environment and launch the application. If you build a Citrix profile that does not contain any shortcuts, users will not be able to launch the application.

Difference Between Deleting and Excluding a Profile Shortcut

To prevent a shortcut from being created in the Citrix profile, you can choose to either delete or exclude it, depending upon whether you want it to remain in the InstallShield project.

- **Excluding a shortcut**—When you exclude a shortcut, it will not be created in the Citrix profile, but it will remain in the InstallShield project. This means that the shortcut would be included in the Windows Installer package that is built from this InstallShield project. See Excluding a Profile Shortcut.
- **Deleting a Shortcut**—When you delete a shortcut, it is removed from both the Citrix profile and the InstallShield project. This means that the shortcut would also be deleted from the Windows Installer package that is built from this InstallShield project. See Deleting a Shortcut.

Managing Shortcuts

On the **Profile Shortcuts** page, you can create, delete, include, exclude, or rename a profile's shortcuts. For step-by-step instructions, see the following topics:

- Creating a New Profile Shortcut
- Including an Existing Profile Shortcut
- Excluding vs. Deleting a Profile Shortcut
- Conditions When a Shortcut Should be Excluded or Deleted
- Renaming a Shortcut

Profile Registry Page

On the **Profile Registry** page, you can view existing registry items, and add or delete registry items. You can also override the Citrix default isolation options for a registry key. Isolation options specify how the isolation environment will provide access to system resources requested by the application.

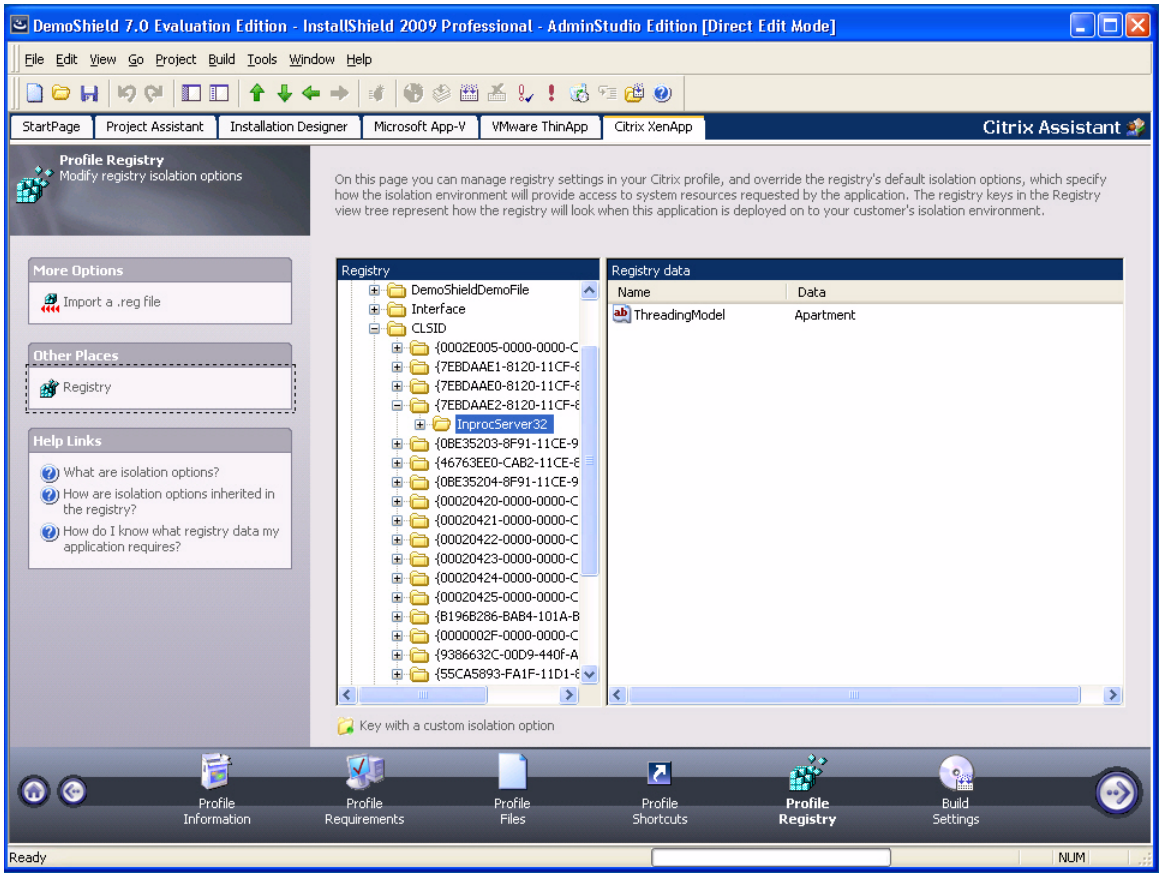


Figure 11-61: Citrix Assistant Profile Registry Page

Registry items that are listed on this page will be included in the Citrix profile, and those that you delete will not. By default, all new registry keys are isolated.

Tip • *To import an existing registry (.reg) file, click the* ***Import a .reg file*** *option on the* ***More Options*** *list to open the Registry Import Wizard.*

Note • *You cannot set isolation options on root registry keys.*

Editing the registry on the **Profile Registry** page is performed much like it is performed on the InstallShield **Registry View**. See Editing the Registry.

For information on how to override a registry key's default isolation options, see Setting Registry Isolation Options.

Important • *While you cannot explicitly set an isolation option on a registry value, registry values are subject to the isolation options of their keys.*

Build Settings Page

On the **Build Settings** page, you can perform the following tasks:

- Selecting Releases to Build
- Digitally Signing a Citrix Profile
- Including Additional Windows Installer Packages in a Citrix Profile
- Enabling Citrix Profile Building When in Direct Edit Mode
- Building a Citrix Profile

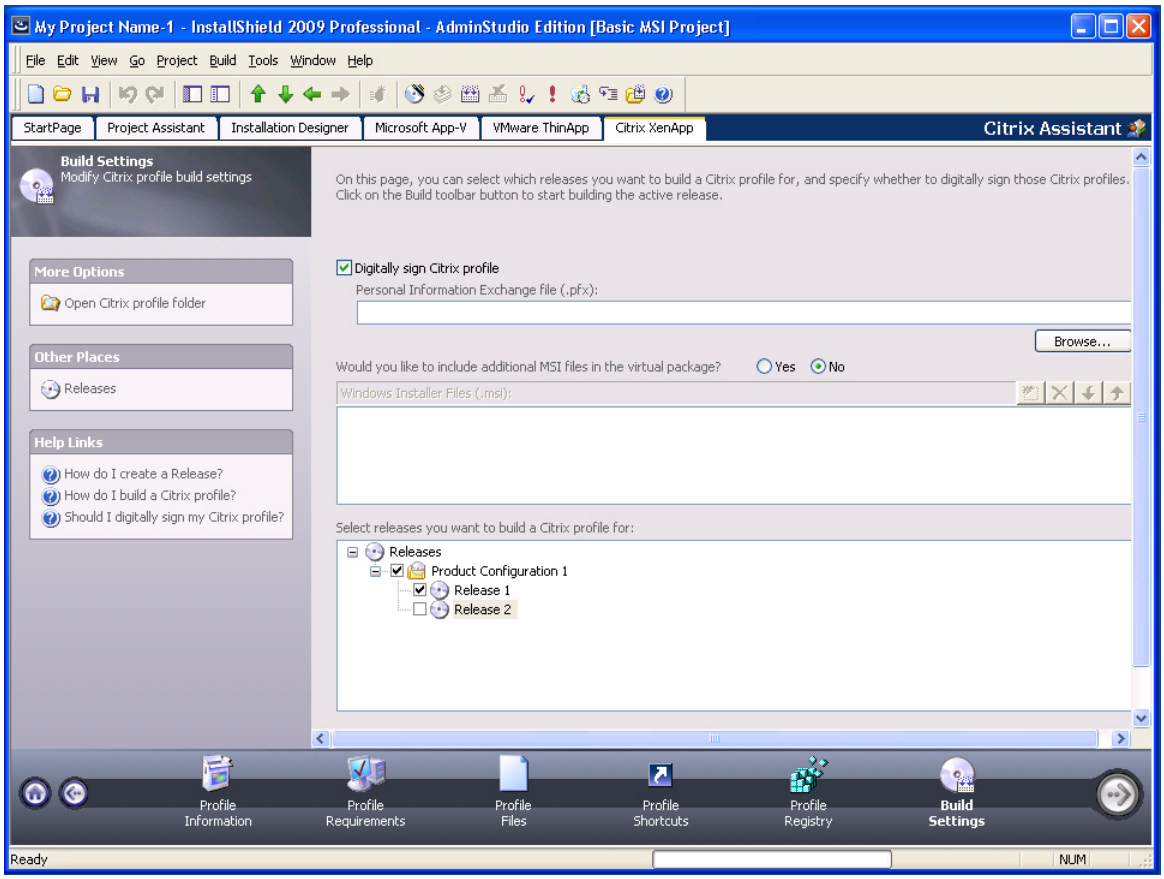


Figure 11-62: Citrix Assistant Build Settings Page

Selecting Releases to Build

You select the releases that you want to build a Citrix profile for on the **Releases** tree of the **Build Settings** page. By selecting a release, you are specifying that whenever that particular release is built, a Citrix profile will also be built.

Note • *If you are editing a Windows Installer package (Direct Edit Mode) or transform file (Direct MST Mode), the **Releases** tree on the **Build Settings** page is not displayed.*

About Building Releases

When you select a release on the Releases tree on the **Build Settings** page, you are specifying that whenever you build that particular release, you want to also build a Citrix profile for that release. However, the releases that are selected on the **Build Settings** page have no bearing upon which release is built when you click the **Build** button on the toolbar. When you initiate a build by clicking the **Build** button, a build is initiated for the **active** release—the release that was most recently selected on the Installation Designer **Releases** view. The output of that build would depend upon what releases were selected on the **Build Settings** page:

- **Active release selected**—A Windows Installer package and a Citrix profile would be built.

- **Active release not selected**—Only a Windows Installer package would be built.

Note • *To build more than one release at a time, perform a batch build. See Performing Batch Builds.*

About Creating Releases

You create and edit releases on the **Releases** view of the InstallShield Installation Designer. You cannot create or edit a release in the Citrix Assistant.

If no releases exist, or if you want to create a new release, open the **Releases** view of the Installation Designer. You must create at least one release before you will be able to build a Citrix profile. For more information, see *Creating and Building Releases.*

Digitally Signing a Citrix Profile

You can digitally sign your Citrix profile to assure end users that neither your installation nor the code within your application has been tampered with or altered since publication. When you digitally sign your application, end users are presented with a digital certificate when they run your installation.

To digitally sign a Citrix profile, select the **Digitally sign Citrix profile** option on the **Build Settings** page. When this option is selected, the **Personal Information Exchange file (.pfx)** field is enabled. A `.pfx` file is a standard file format for digital certificates. You then click **Browse** and select the `.pfx` file that you want to use to digitally sign this Citrix profile.

Including Additional Windows Installer Packages in a Citrix Profile

Sometimes a primary Windows Installer package uses other Windows Installer packages indirectly, such as driver files, client components, etc. In addition to being able to convert a single Windows Installer package to a virtual package, you can also use the Citrix Assistant to convert an application suite of multiple Windows Installer packages into one virtual package.

To include additional Windows Installer packages in a Citrix profile, set the **Would you like to include additional MSI files in the virtual package?** option to **Yes**, and then select the packages that you want to add.

- Click the New button ( ) and select the Windows Installer packages that you want to add. After each file is selected, it will be listed in the **Windows Installer Files (.msi)** list.
- The order of the packages can be changed by selecting a package in the list and clicking the Move Up ( ) and Move Down ( ) buttons.
- Use the Delete button ( ) to delete a package from the list.

Enabling Citrix Profile Building When in Direct Edit Mode

When you are editing a Windows Installer (`.msi`) package or a transform (`.mst`) file in the **Citrix Assistant**, you are in Direct Edit Mode or Direct MST Mode. Because you are directly editing a Windows Installer package, you save your changes by selecting **Save** on the **File** menu. It not necessary to build the package, because it is already built. Therefore, InstallShield's **Build** function is disabled.

However, you do need to run the build process to build a Citrix profile for this Windows Installer package. To enable the **Build** button to build the Citrix profile, select the **Build Citrix Profile** option on the **Build Settings** page.

After you select this option, the **Build Citrix Profile** selection on the **Build** menu becomes enabled, as does the **Build** toolbar button.

Building a Citrix Profile

The method for building a Citrix profile depends upon what file you have open—an InstallShield project or a Windows Installer package. For detailed instructions, see one of the following topics:

- Building a Citrix Profile for an InstallShield Project
- Building a Citrix Profile for a Windows Installer Package

Dialog Boxes

The Citrix Assistant includes the following dialog boxes:

- Script Execution Dialog Box
- Diagnostic Tools Dialog Box
- File Isolation Options Dialog Box
- Folder Isolation Options Dialog Box
- Registry Isolation Options Dialog Box
- Service Packs Requirement Dialog Box

Script Execution Dialog Box

On the **Script Execution** dialog box, which is opened by clicking **Script Execution** in the **More Options** list on the **Profile Requirements** page, you can choose to include scripts that must execute for your application to run properly. From this dialog box, you can view and manage all of the **Before Profile Launch** and **After Profile Exit** script files you are including with your Citrix profile.

- Files can be marked to run inside or outside of the isolation environment.
- Only files with `.exe`, `.com`, `.cmd`, or `.bat` extensions are allowed to execute.

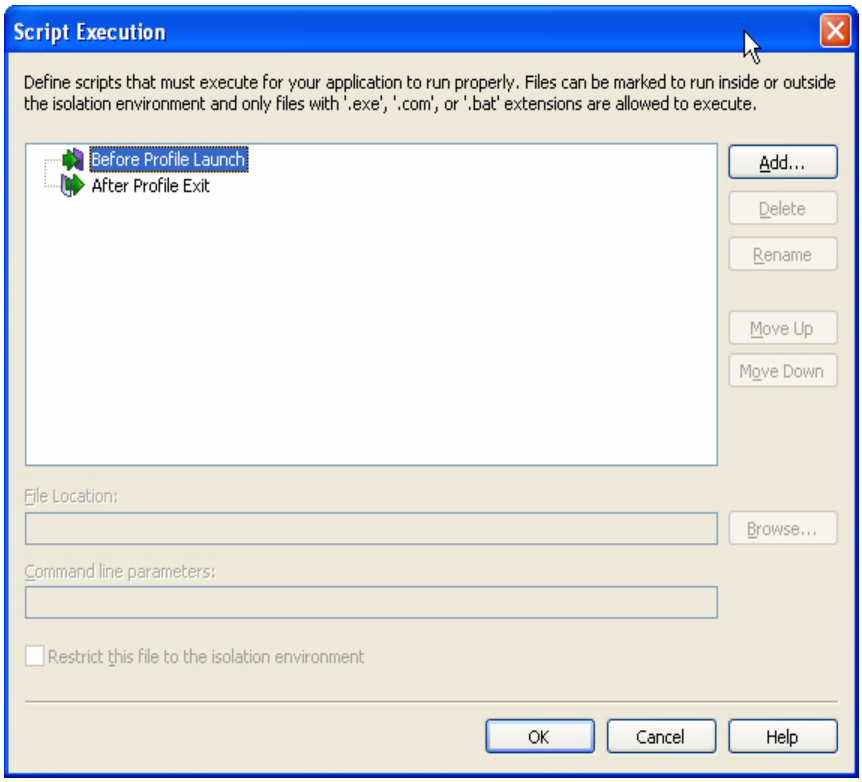


Figure 11-63: Custom Execution Dialog Box

For step-by-step instructions on using this dialog box, see Adding Pre-Launch and Post-Exit Scripts.

Diagnostic Tools Dialog Box

On the **Diagnostic Tools** dialog box, which is opened by selecting **Diagnostic Tools** in the **More Options** list on the **Profile Information** page, you can choose to include the Windows Command Prompt and Registry Editor diagnostic tools with your Citrix profile.

If you include diagnostic tools with your Citrix profile, you will be able to look at the registry or file system for the application while it is running in its isolation environment. For example, if you were running a Citrix profile and got an error message stating that the application cannot load a DLL, you could use these diagnostic tools to troubleshoot the problem.

Caution • *If you choose to include these diagnostic tools, the versions of* `regedit.exe` *and* `cmd.exe` *that are part of the operating system on the build machine are added to the Citrix profile. However, these tools may not be compatible with other operating systems.*

Figure 11-64: Diagnostic Tools Dialog Box

You can use these diagnostic tools to inspect your application's isolation environment at runtime. You have the following options:

Table 11-57 • Diagnostic Tools Dialog Box Options

Option	Description
Registry Diagnostics	Select this option if you want to include `regedit.exe` with your Citrix profile so that you can browse the profile registry.
File System Diagnostics	Select this option if you want to be able to browse the Citrix profile's isolation environment file system using a command prompt.

Launching the Diagnostic Tools Within the Isolation Environment

If you selected the **Registry Diagnostics** or **File System Diagnostics** options on the **Diagnostic Tools** dialog box, shortcuts to those tools are automatically added to the profile.

When the user runs this Citrix profile application, two additional shortcuts will be available in the application's shortcut folder: The names of these shortcuts will reflect the application name, such as:

```
[ProductName] Registry
[ProductName] File System
```

When the user launches one of these shortcuts, that diagnostic tool is launched inside the context of the application's Citrix isolation environment.

File Isolation Options Dialog Box

On the **File Isolation Options** dialog box, you can override the default Citrix isolation option for the selected file.

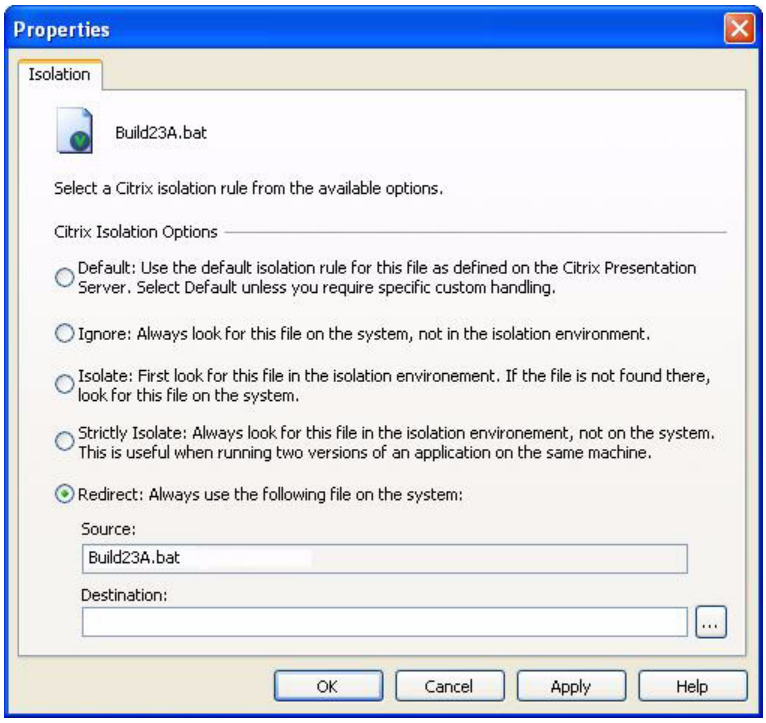


Figure 11-65: File Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

The File Isolation Options dialog box includes the following options;

Table 11-58 • File Isolation Options

Option	Description
Default	Use the default isolation option for this file as defined on the Citrix XenApp. ***Note*** • *Select this option unless you require specific custom handling.*
Ignore	Always look for this file on the system, not in the isolation environment.
Isolate	First look for this file in the isolation environment. If the file is not found there, look for this file on the system.

Table 11-58 • File Isolation Options

Option	Description
Strictly Isolate	Always look for this file in the isolation environment, not on the system. ***Note*** • *This is useful when running two versions of an application on the same machine.*
Redirect	Always use the following file on the system, not the one in the isolation environment. If you select this option, also specify the following: • **Source**—Name of the selected file. • **Destination**—Select the file on the system that you want the application to use instead of the selected file.

Folder Isolation Options Dialog Box

On the **Folder Isolation Options** dialog box, you can override the default Citrix isolation option for the selected folder.

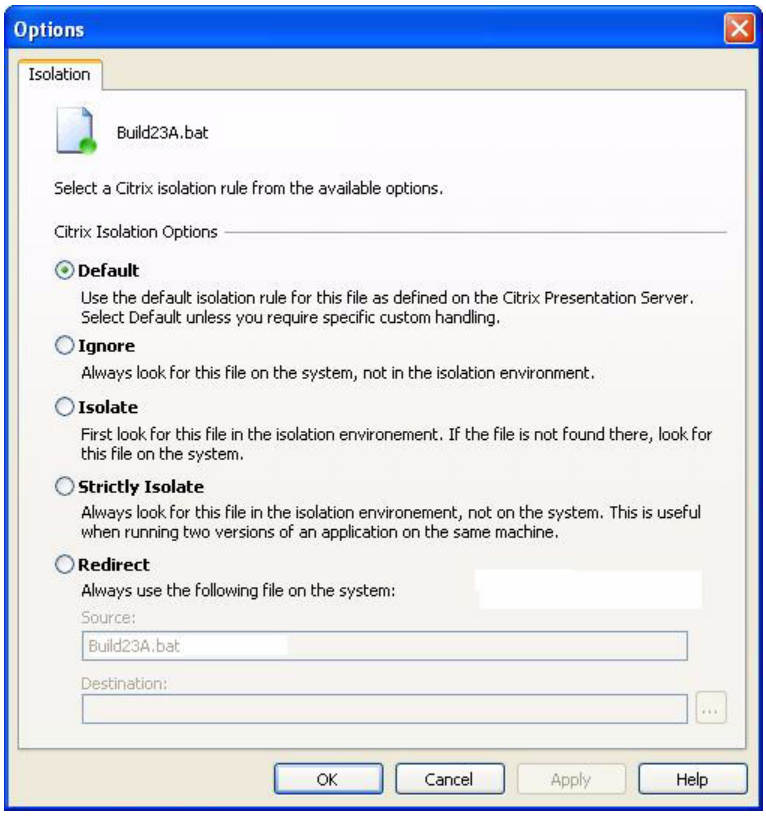


Figure 11-66: Folder Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

The **Folder Isolation Options** dialog box includes the following options;

Table 11-59 • Folder Isolation Options

Option	Description
Default	Use the default isolation option for this file as defined on the Citrix XenApp. ***Note*** • *Select this option unless you require specific custom handling.*
Ignore	Always look for this file on the system, not in the isolation environment.
Isolate	Look for this folder in both the isolation environment and on the system. If the folder exists in both places, list both in the search results.
Strictly Isolate	Always look for this folder in the isolation environment, not on the system. ***Note*** • *This is useful when running two versions of an application on the same machine.*
Redirect	Always look in the following folder on the system, not in the one in the isolation environment. If you select this option, also specify the following: • **Source**—Name of the selected folder. • **Destination**—The directory on the system where you want the application to look instead of looking in the selected folder in the isolation environment.

Registry Isolation Options Dialog Box

On the **Registry Isolation Options** dialog box, you can override the default Citrix isolation option for the selected registry key.

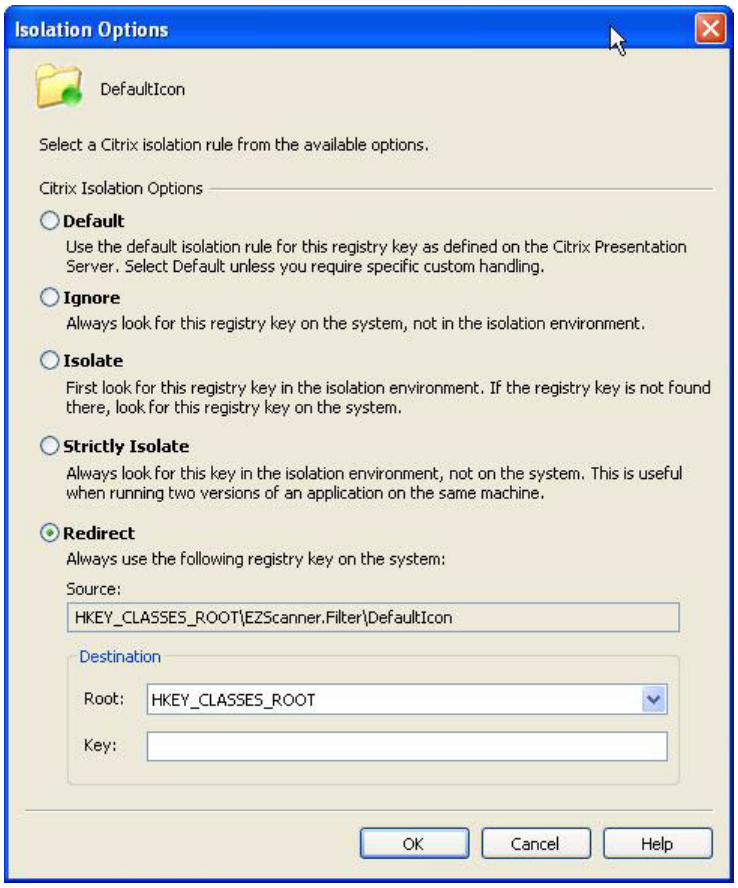


Figure 11-67: Registry Isolation Options Dialog Box

Caution • *Modify isolation options only if you have advanced knowledge of Microsoft operating system objects and registry settings.*

The Registry Isolation Options dialog box includes the following options;

Table 11-60 • Registry Isolation Options

Option	Description
Default	Use the default isolation option for this registry key as defined on the Citrix XenApp. ***Note*** • *Select this option unless you require specific custom handling.*
Ignore	Always look for this registry key on the system, not in the isolation environment.
Isolate	Look for this registry key in both the isolation environment and on the system. If the registry key exists in both places, list both in the search results.

Table 11-60 • Registry Isolation Options (cont.)

Option	Description
Strictly Isolate	Always look for this registry key in the isolation environment, not on the system. ***Note*** • *This is useful when running two versions of an application on the same machine.*
Redirect	Always use the following registry key on the system. If you select this option, also specify the following: • **Source**—Lists the name of the selected registry key. • **Destination Root**—Select the registry root of the registry key on the system that you want to redirect to. • **Destination Key**—Select the registry key on the system that you want to redirect to.

Service Packs Requirement Dialog Box

The **Service Packs Requirement** dialog box is opened by selecting an operating system on the **Profile Requirements** page of the Citrix Assistant and selecting **Service Packs Requirement** from the context menu.

On this dialog box, you can specify which, if any, Service Packs are required for the application to run on the selected operating system

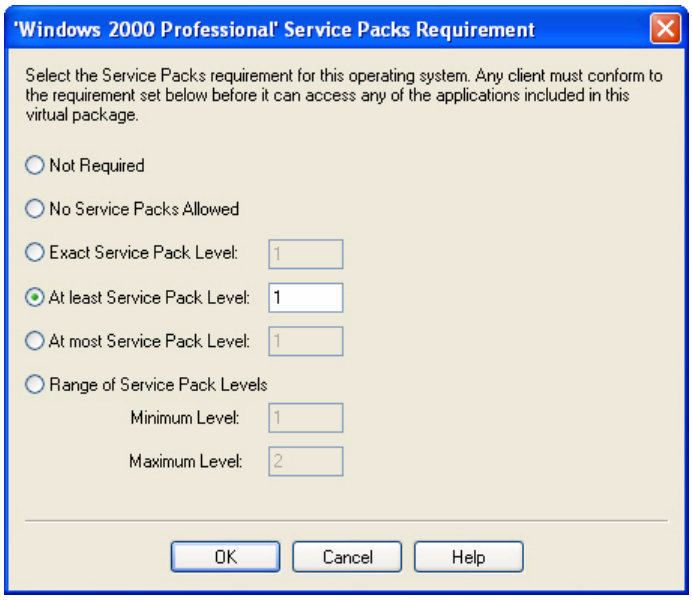


Figure 11-68: Service Packs Requirements Dialog Box

The **Service Packs Requirement** dialog box includes the following options:

Table 11-61 • Service Packs Requirement Options

Option	Description
No Service Pack Requirement	Select this option if this application supports all versions of this operating system, regardless of the number of Service Packs installed.
No Service Pack Allowed	Select this option if this application only supports the initial release of this operating system; if any Service Packs are installed, this application will not run properly.
Exact Service Pack Level	Select this option if this application requires the installation of a specific Service Pack on this operating system in order to run properly. Enter the required Service Pack Level in the box.
At Least Service Pack Level	Select this option if, to run properly, this application requires that this operating system have at least the specified Service Pack (or higher) installed. Enter the minimum required Service Pack Level in the box.
At Most Service Pack Level	Select this option if, to run properly, this application requires that this operating system have at most the specified Service Pack (or lower) installed. Enter the maximum required Service Pack Level in the box.
Range of Service Pack Levels	Select this option if, to run properly, this application requires that this operating system have a specified range of Service Packs installed. If you select this option, specify the **Minimum Level** and **Maximum Level** in the boxes.

Building Citrix Profiles Using the Command Line

When you configure a Citrix profile in an InstallShield project and then build that project (using either the user interface or the command line), both the Windows Installer package and the Citrix profile are built. When you use the standard InstallShield command line build, you do not need to add any additional command line parameters. All of the Citrix profile settings are saved within the InstallShield project.

Citrix Profile Conversion Error and Warning Messages

Note • *For troubleshooting information about resolving errors and warnings that you may encounter when you are building a virtual application, see Virtualization Conversion Errors and Warnings.*

Application Features Requiring Pre- or Post-Conversion Actions

Some application features are ignored when creating a Citrix profile. Therefore, some additional pre- or post-conversion actions must be taken in order for the application profile to run on Citrix XenApp.

One action you could take to try to include ignored features in a Citrix profile is to first repackage the application using the Repackaging Wizard, and then convert the repackaged application to a Citrix profile.

For a list of ignored features, see Application Features Requiring Pre- or Post-Conversion Actions.

12

Customizing and Authoring Installations Using InstallShield

Edition • *InstallShield Editor is included with AdminStudio Standard, Professional, and Enterprise Editions.*

InstallShield Editor provides the most comprehensive and flexible setup-creation technology available for the Windows Installer. With the latest InstallShield installation development environment (Interface) you can create setup packages that utilize Windows Installer technology, while harnessing the flexibility provided by InstallScript, InstallShield's development language.

Administrators can also take advantage of InstallShield Editor to customize repackaged legacy setups, further enhancing them prior to deploying them in production environments. For AdminStudio Professional and Enterprise Editions, InstallShield Editor contains integrated Application Manager and ConflictSolver functionality, allowing you to perform conflict identification before leaving InstallShield Editor.

InstallShield Editor's documentation is divided into the following main areas:

Table 12-1 • InstallShield Editor Documentation

Section	Description
InstallShield Editor Integration with Application Manager and the Software Repository	Explains how to use the Application Manager submenu on the InstallShield Editor File menu to access packages stored in an AdminStudio Application Catalog Software Repository.
Application Manager Tab on InstallShield Editor Options Dialog Box	Describes the additional tab that appears on the InstallShield Editor Options dialog box when InstallShield Editor is invoked from ConflictSolver: the ConflictSolver tab.
Microsoft App-V, VMware ThinApp, and Citrix XenApp Virtualization Support	Provides an overview of how Repackager and InstallShield Editor provide support for the conversion of Windows Installer packages to virtual applications.

Table 12-1 • InstallShield Editor Documentation (cont.)

Section	Description
InstallShield Editor Help Library	Explains how to use InstallShield Editor to take advantage of its features to build Windows Installer packages.

AdminStudio-Specific Functionality in InstallShield Editor

When running InstallShield Editor from AdminStudio, some AdminStudio-specific functionality is enabled:

Table 12-2 • AdminStudio Functionality in InstallShield Editor

Functionality	Description
InstallShield Editor Integration with Application Manager and the Software Repository	If you have the AdminStudio Enterprise Edition, which includes the Software Repository feature, you can perform the following tasks from the InstallShield Editor interface: • Getting a Copy of a Software Repository Package • Opening and Editing a Software Repository Package • Adding a Package to the Software Repository via the InstallShield Editor Build Process • Using the Application Manager Context Menu
Repackager Integration	You can launch Repackager from the InstallShield Editor interface for conversion of the following: • Novell ZENworks projects (`.axt/.aot`) • Microsoft SMS Installer projects (`.ipf`) • WinINSTALL exported text files (`.txt`) • Wise Installation files (`.wse`) • Legacy Repackager projects (`.inc`) When you attempt to open any of these file types in InstallShield Editor, you can launch Repackager to perform the conversion.
Application Manager options in the **Options** dialog box AdminStudio Professional and Enterprise Editions	The InstallShield Editor **Options** dialog box has an additional tab for **Application Manager** and ConflictSolver configuration. See Application Manager Tab on InstallShield Editor Options Dialog Box

Table 12-2 • AdminStudio Functionality in InstallShield Editor (cont.)

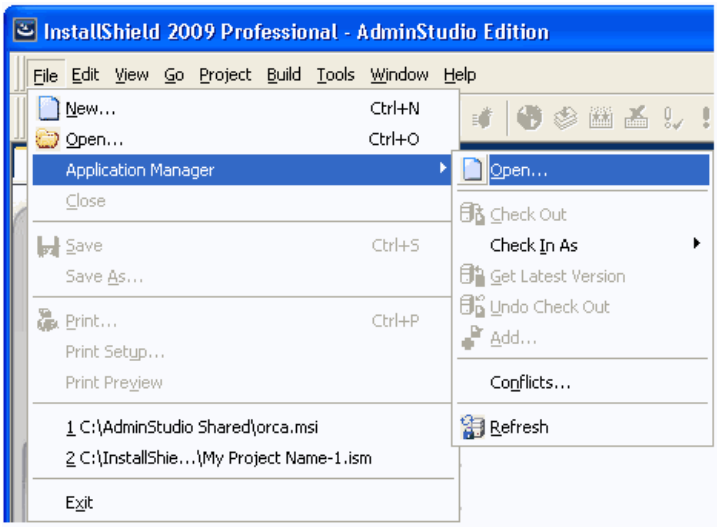
Functionality	Description
Importing packages and merge modules into the AdminStudio Application Catalog from InstallShield Editor AdminStudio Professional and Enterprise Editions	Open the **File** menu, point to **Application Manager**, and click **Add** to open the **Import Wizard**. Using this Wizard, you can import Windows Installer packages and merge modules into the AdminStudio Application Catalog.
Checking for conflicts from InstallShield Editor AdminStudio Professional and Enterprise Editions	Open the **File** menu, point to **Application Manager**, and click **Conflicts** to open the **Conflict Wizard**. Using this Wizard, you can check for package conflicts before leaving InstallShield Editor.

Note • *AdminStudio Role permissions apply to Application Manager and ConflictSolver functionality in InstallShield Editor. If you are not assigned to a Role with sufficient permissions, you may not be able to access some of these features.*

InstallShield Editor Integration with Application Manager and the Software Repository

Edition • *The Software Repository feature is available in AdminStudio Enterprise Edition.*

By making selections from the **Application Manager** submenu on the InstallShield Editor **File** menu, you can access packages stored in a AdminStudio Application Catalog Software Repository.



InstallShield Editor can perform the following tasks using packages in the Software Repository:

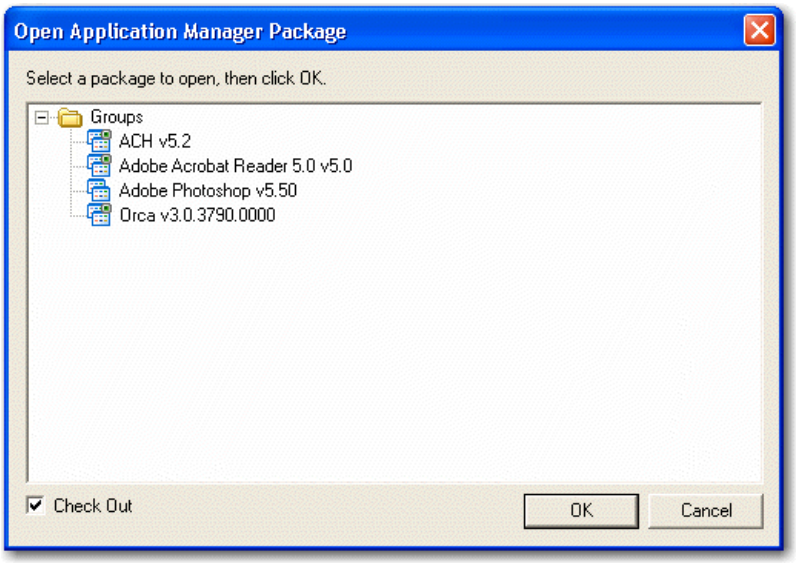
- Getting a Copy of a Software Repository Package
- Opening and Editing a Software Repository Package
- Adding a Package to the Software Repository via the InstallShield Editor Build Process
- Using the Application Manager Context Menu

Note • *For additional information, see Using the Software Repository.*

Getting a Copy of a Software Repository Package

Task: ***To copy a package from the Software Repository and open it in InstallShield Editor:***

1. In AdminStudio, connect to an Application Catalog that has the Software Repository enabled.
2. Launch InstallShield Editor.
3. On the **File** menu, point to **Application Manager** and click **Open**. The **Open Application Manager Catalog** dialog box opens, listing the package tree in the AdminStudio Application Catalog:



In Application Manager, packages that are managed within the Software Repository have a different icon than those that are not:

Table 12-3 •

Icon	Description
	Package is not managed within the Software Repository.
	Package is managed within the Software Repository.
	Package is managed within the Software Repository and is checked out.
	Merge Module is managed within the Software Repository.

4. Clear the **Check Out** option.

5. Select a package that is stored in the Software Repository and click **OK**. A copy of the selected package is saved onto your local directory. When the copy is complete, the selected package is opened in **Direct MST Mode** and the **Installation Information View** is open.

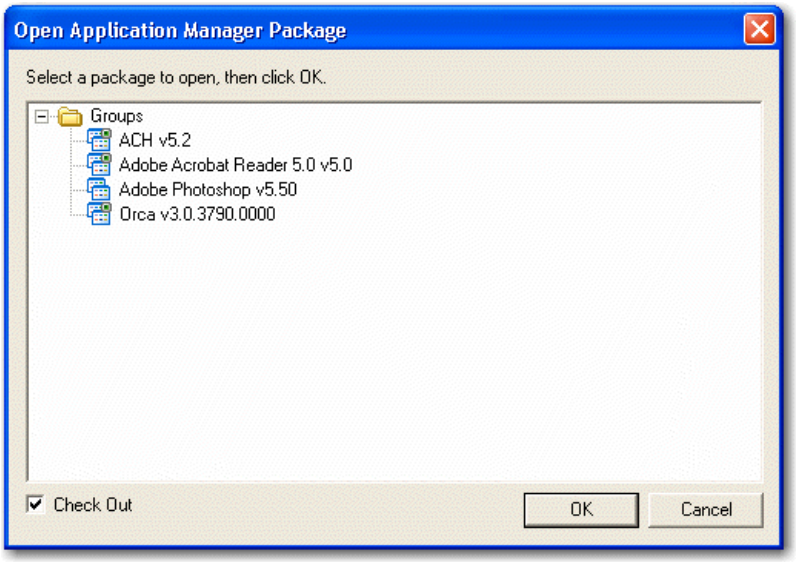
   You can now customize this file by creating a transform and save your changes onto your local directory. The original package in the Software Repository is not modified.

Opening and Editing a Software Repository Package

Edition • *The Software Repository feature is available in AdminStudio Enterprise Edition.*

Task: ***To open and edit Software Repository package in InstallShield Editor:***

1. In AdminStudio, connect to an Application Catalog that has the Software Repository enabled.
2. Launch InstallShield Editor.
3. On the **File** menu, point to **Application Manager** and click **Open**. The **Open Application Manager Catalog** dialog box opens, listing the package tree in the AdminStudio Application Catalog:



In Application Manager, packages that are managed within the Software Repository have a different icon than those that are not:

Icon	Description
	Package is not managed within the Software Repository.

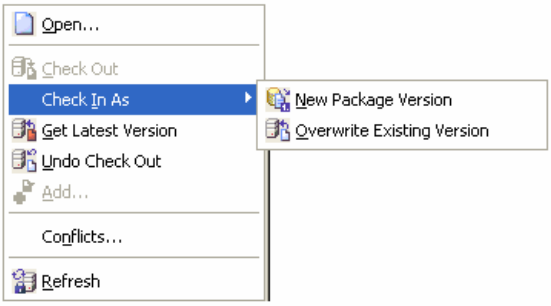
Icon	Description
	Package is managed within the Software Repository.
	Package is managed within the Software Repository and is checked out.
	Merge Module is managed within the Software Repository.

4. Select the **Check Out** option.
5. Select a package that is stored in the Software Repository and click **OK**. A copy of the selected package is saved onto your local directory.

   When the copy is complete, the selected package is opened in **Direct MST Mode** and the **Installation Information View** is open.

   The package is now in Checked Out status in Application Manager, meaning that no one else can check it out until you check it back in. If you were to open Application Manager and select this package, you would see the Checked Out icon in the Product View:

6. Customize this package using InstallShield Editor.
7. When you have finished editing this package, on the File menu, point to Application Manager and click **Check in As**. You are now given two options: **New Package Version** or **Overwrite Existing Version**.



8. Select one of those options to save the edited package in the Application Catalog.

Adding a Package to the Software Repository via the InstallShield Editor Build Process

Task: ***To add a package to the Software Repository via the InstallShield Editor Build process:***

1. In AdminStudio, connect to an Application Catalog that has the Software Repository enabled.
2. Launch InstallShield Editor.
3. Open an InstallShield project (`.ism`).
4. Open the **Releases View**.
5. Set the **Publish to Software Repository** property to **Yes**.
6. Select the **AdminStudio Group** that you want the imported package to belong to.
7. Select one of the following **Duplicate Package** options:
   - **New Package Version**
   - **Overwrite Existing Version**
   - **New Package History Version**
   - **Ignore if Exists**
8. Build the setup.

   Upon completion of the build, the MSI package is published to the Software Repository. The progress messages are displayed in the Output window.

   Note • *If the setup is compressed, an administrative image must be created before the package can be published to the Software Repository.*

Using the Application Manager Context Menu

Instead of opening packages from the **Open Application Manager Catalog** dialog box in InstallShield Editor, you can select a Software Repository package in the Application Manager tree and select **Edit with InstallShield Editor** from the context menu. The package is checked out to you and opened in the Direct MST Mode of InstallShield Editor.

Application Manager Tab on InstallShield Editor Options Dialog Box

The InstallShield Editor **Options** dialog box has a tab named **Application Manager**.

From this tab, you can specify the Application Catalog database to which you want to connect when using integrated Application Manager functionality in InstallShield Editor. If you click the folder icon, the **Connect Application Catalog** dialog box opens, from which you can select the Application Catalog.

If you click **Advanced**, the Options Dialog Box opens.

Microsoft App-V, VMware ThinApp, and Citrix XenApp Virtualization Support

Edition • *The Automated Application Converter, Microsoft App-V Assistant, ThinApp Assistant, and Citrix Assistant are included in the AdminStudio Virtualization Pack.*

Important • *AdminStudio ThinApp support requires a separate purchase of VMware ThinApp™.*

Both AdminStudio and InstallShield Editor provide support for the conversion of Windows Installer packages to virtual packages:

- **Automated Application Converter**—You can use the Automated Application Converter to convert a Windows Installer package to a Microsoft App-V application, ThinApp application, or Citrix profile. Automated Application Converter can examine a group of selected setups and perform automated virtualization of those that can be cleanly virtualized. For those setups that cannot be cleanly virtualized (due to custom actions, etc.), Automated Application Converter can perform automated repackaging of those setups and then perform automated virtualization of those repackaged MSIs.
- **Repackager Interface**—By selecting an option on the Repackaged Output view, you can simultaneously build an InstallShield Editor project, a Windows Installer package, a Microsoft App-V application, a ThinApp application, and a Citrix profile from a Repackager project.
- **Microsoft App-V Assistant**—Using the Microsoft App-V Assistant, you can convert a Windows Installer package or an InstallShield project to a customized App-V application. You can modify a Microsoft App-V application's operating system requirements, files, folders, shortcuts, registry settings, isolation options, and build options.
- **ThinApp Assistant**—Using the VMware ThinApp Assistant, you can convert a Windows Installer package or an InstallShield project to a customized ThinApp application. You can configure a ThinApp application's files, folders, shortcuts, registry settings, isolation options, and build options.

- **Citrix Assistant**—Using the Citrix Assistant, you can convert a Windows Installer package or an InstallShield project to a customized Citrix profile. You can modify a Citrix profile's operating system and language requirements, files, folders, shortcuts, registry settings, script execution, isolation options, and build options.

For more information on the capabilities of these features, see Getting Started With Application Virtualization.

Differences Between InstallShield Editor and InstallShield Professional Edition

The InstallShield Editor that is included with AdminStudio 10.0 SP2 is based upon InstallShield Professional Edition, but it has a slightly different feature set. Those differences are explained here.

Default Project Types

In InstallShield Editor, all non-Windows Installer-based project types are disabled by default. You can enable these additional project types, such as InstallScript, on the InstallShield Editor **Options** dialog box, which is opened by clicking **Options** on the **Tools** menu.

Multilingual Runtime Language Support

InstallShield Editor includes InstallShield Premier Edition's multilingual runtime language support, which enables you to create a single installation that displays end-user text in multiple languages. If an installation will contain more than one language, you can specify whether to prompt the end user to select the run-time language, or to automatically display the language of the target system's operating system.

Using a Network Repository to Share Project Elements

A repository is a collection of common elements that can be shared and reused in different installation projects, enabling you to ensure consistency. InstallShield Editor includes not only InstallShield Professional Edition's *local* repository support, but also InstallShield Premier Edition's *network* repository support, which fosters collaboration among installation authors.

InstallShield Editor Help Library

Edition • *InstallShield Editor is included with AdminStudio Standard, Professional, and Enterprise Editions.*

The InstallShield Editor Help Library gives you unified access to InstallShield Editor Help. This library's help topics contain information that assist you in finding answers with InstallShield Editor.

Open the InstallShield Editor Help Library to see a listing of the Help Library's contents.

Note • *You can also download the InstallShield documentation PDFs from the InstallShield Documentation Center.*

13

Customizing Installations Using Tuner

Using Tuner, you can add to, modify, or remove information from a Windows Installer package. This involves creating a transform file, where all the modifications are stored. When you install the package and transform together, your modifications are reflected in the installation.

Tuner allows you to configure the initial state of features, add or remove files from an installation, edit registry entries, configure setup properties, set Add/Remove Programs options, and configure servers for application resiliency. You can also validate Windows Installer packages and transform files to ensure they conform to Microsoft guidelines.

Tuner user documentation is presented in the following sections:

Table 13-1 • Tuner User Documentation

Section	Description
Working with Transforms	Explains how to create a transform file to customize a Windows Installer-based installation.
Validation	Explains how to compare a Windows Installer-based installation to a known set of guidelines (an evaluation file) to ensure it has been created to those guidelines.
Setup Organization	Explains how to modify two main parts of the installation that your end users will see: the default path and default company name, and the actual features that can, will, or will not be installed.
Configuring Package Content	Explains how to use a transform file to manipulate the original package contents, including files and folders, registry entries, shortcuts, INI files, ODBC resources, and NT services.
Working with Dialogs	Explains how to disable particular panels that appear during the installation, administrative, patch, or maintenance sequences.

Table 13-1 • Tuner User Documentation (cont.)

Section	Description
Configuring Additional Server Locations	Explains how to configure additional server locations.
Changing Add/Remove Program Settings	Explains how to configure the Windows setup to give the user the option of removing, repairing, or changing the installation with the click of a button.
Customizing Setup Properties	Explains how to edit property values in the properties table (the underlying structure of Windows Installer packages). You can also add your own custom properties.
Preparing Packages for Distribution	Explains how to postvalidate your transform and base Windows Installer package, and how to package the transform and base package for distribution.
Directly Editing Packages	Explains how to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.
Tuner Reference	Information on Tuner Views and dialog boxes.

When to Use Tuner vs. InstallShield Editor

Most system administrators use InstallShield Editor to import repackaged setups and convert them into Windows Installer packages. InstallShield Editor is also ideal for making changes to the package that you want reflected in all deployments of the package. However, it is recommended that you use Tuner to create transforms for changes that you only want to affect a particular deployment, rather than every installation.

Working with Transforms

The Microsoft-designated term transform refers to a specific file type used to customize a Windows Installer-based installation. A transform contains all modification information, such as whether features are installed, how they are installed, which files, shortcuts, and registry entries are included, and Windows 2000 and XP Add/Remove Programs information. Transform files use an .mst extension.

For example, you may need to customize an installation for different departments in your company. Typical business productivity suites come with a spreadsheet program, a word processor, and a presentation tool. Your accounting department may only need the spreadsheet and the presentation programs. On the other hand, the writing department may need only the word processor and the spreadsheet. A third department may need the entire suite of applications. Instead of manually setting up every person in the company, you can take the original setup of the entire suite, and create a customization project in the form of a transform to meet the needs of each department. A transform would need to be created for every configuration that you plan to use.

Once you have created a transform, you can apply it at runtime, depending on whose machine the application is being installed on. For example, in the accounting department, the transform limits the installation to include only the spreadsheet and presentation programs.

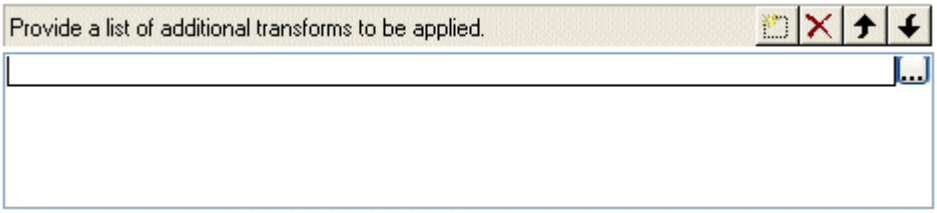
In some cases, it may be necessary to have multiple transform files for an installation. For example, a vendor may use a transform file for language-specific information. When you want to customize that MSI file, you need to include the preexisting transform so your modifications affect the entire existing package.

Creating New Transform Files

Task: ***To create a new transform file:***

1. Launch **Tuner** from the AdminStudio interface. The **Tuner Start Page** opens.
2. Click **Create a new transform** on the left side of the view or select **New** from the **File** menu. The right side of the view changes to display the fields necessary to create a new transform.
3. In the **Select an MSI File** field of the **Base Windows Installer Package** area, enter the name and location of the Windows Installer package that you are customizing, or click **Browse** to locate it.
4. If there are transforms already associated with the Windows Installer package, (for example, previous customizations or transforms containing language-specific information), go to the **Provide a list of additional transforms to be applied** area and click the New button:

   A new entry appears in the list.

   When an entry appears in the list, click the Browse button (...) to the right of it and locate the transform. If multiple transforms are associated with this package, use the Move Up and Move Down buttons to specify the order in which the transforms are applied.

   Caution • *When using multiple transforms, keep in mind that the order in which they are applied is critical. For example, if you create a transform for a Windows Installer package that creates a new value for a property, and then create a second transform that changes the value created in the first transform, everything works correctly. However, if you apply the second transform first, that transform is attempting to modify the property's value, instead of creating it. That will result in an error.*

   One simple example of where this may be a problem is with the default company name. If the value is not set by default, and you set it in using the first transform, a new value for the property is created. If you create a second transform that modifies the combined original package and first transform, and the second transform changes the default company name, it is only changing the property. However, if you try to apply the second transform

without the first one, Windows Installer interprets this as trying to change a null value to another value, which will result in an error.

5. By default, the transform will be created in the same directory as the Windows Installer package, and named the same as the base package with an `.mst` extension. However, if you want to change the name and/or location of the transform, you can also do so in the **Windows Installer Transforms** area. Click **Browse** to open the **Save Customization File** dialog box.

   Navigate to the directory in which you want to store the transform file you are creating. Provide the name of the transform with an .mst extension (for example, `MyTransform.mst`) and click **Save**. The dialog box closes and the path and file name appear in the edit field.

6. If you want to create a **Response Transform**, check the appropriate box. If you are using a response transform, you can specify additional command-line properties (in property name/value pairs separated by semicolons) to pass to the response transform. These must be PUBLIC properties, and only control how the dialogs are displayed during creation of the response transform. They are not persisted outside of the UI sequence during creation. For example, you can pass the property/value pair `ARPHELPTELEPHONE=1-111-111-1111` to set the value of the **Help Telephone** field of **Add/Remove Programs**. See Using Response Transforms for more information.

   You might pass a property/value pair during response transform creation to display all dialogs during an installation that may not be displayed based on your system configuration (for example, to show Windows 9x-only dialogs on a Windows NT platform). You can then make appropriate responses and have them included in your transform.

7. Ensure all the information entered is correct, and click **Create**.

   - **If you are creating a Standard Transform**, the transform file is opened in the Tuner interface, displaying the Package Validation View.
   - **If you are creating a Response Transform**, a simulated installation of the selected application begins. Step through the installation, making changes as necessary. When you reach the end of the installation sequence and click Install, the installation will exit and the Tuner interface will open your transform, displaying the Package Validation View. Your transform contains all of the changes you made during the simulated installation.

Note • *You can access information about the original MSI file and associated transforms by selecting Properties from the File menu.*

Opening Existing Transforms

On the Tuner Start Page, when you click on Open an existing transform file, the pane on the right of the interface changes. You can then specify the name and location of the base Windows Installer package, any associated transforms, and the name and location of the transform file.

Note • *Generally, you will only use this option when opening existing transforms that were created by a product other than Tuner, or created by someone other than yourself. Transforms you create using Tuner are more easily accessed through using the Open a recent transform file selection.*

Opening Recently Accessed Transforms

On the Tuner Start Page, when you click on Open a recent transform file, the pane to the right changes to a list containing your most recently accessed transforms.

From this View, you can perform the following tasks:

- To open a transform file, select a transform file and click Open.
- To view information on the transform file, select a transform file and click Properties. The Properties dialog box opens, listing details about the base MSI package and associated transforms.
- To specify the view that will appear when Tuner is started, select one of the following options:
  - Reload the last project saved when restarting Tuner
  - Make this my default Tuner Start Page Screen
  - Make Welcome my default Tuner Start Page Screen

Creating Generic Transforms

Most transforms are tied to a specific product code, meaning they can only be applied to a specific version of a product. Generic transforms, however, do not have that limitation. They can be created to apply to multiple versions of a Windows Installer package (for example, Office XP and Office 2000), or to any Windows Installer package.

Task: ***To create a generic transform:***

1. Create a transform in Tuner as you would do normally.
2. With the transform project open, select Transform Summary Information from the Project menu to access the **Transform Summary** dialog box.
3. When the **Transform Summary** dialog box appears, change the validation options to reflect how you want the transform applied. If you want to create a completely generic transform, deselect all validation options.

Tip • *One use for generic transforms is to enforce standard Add/Remove Programs information on every package installed in your environment. The same transform can be used to set all relevant properties.*

Using Response Transforms

There are two ways you can create a transform file in Tuner:

- The first and the most common way is to begin by creating an empty transform and then making customization by navigating to different views in Tuner.
- The second way is by running the installation and then customizing various options available in each setup panel. The installation is only simulated and no changes take place on the user system. Tuner saves all the changes that the user has made on each panel of the setup in the transform. This type of transform is called a response transform.

Response transforms, much like installation response files, allow you to run an existing Windows Installer-based installation and capture your configurations. Unlike a response file, these changes are used as a starting point for your new transform.

For example, if you use Tuner to create a response transform, you might select certain features you want installed, the location of the installation, and company information. When the Tuner interface is opened, these values will already be set for your new transform file. You can then make further customizations as necessary.

You might want to create a response transform for an installation, and then fill in your company name as the default name, and a specific directory for installation which is different from the one suggested by the manufacturer. Further, you may want to configure a specific feature, such as clip art, to not be installed. By using the familiar installation user interface, you can quickly make your basic customizations before using the Tuner environment to refine the transform.

Viewing Transform Properties

To view properties of the transform you are currently creating or editing, select Properties from the File menu. To view the properties of a project from the Open a Recent Transform View, select the transform file and click Properties or selecting Properties from the context menu.

The resulting Properties dialog box provides information about the transform, including the name and location of the base Windows Installer file, and any additional transforms that are associated with this transform and MSI file.

Validation

What is Validation?

Validation is the process of comparing a Windows Installer-based installation to a known set of guidelines (an evaluation file) to ensure it has been created to those guidelines. Tuner can perform two types of validations: prevalidation and postvalidation.

- **Prevalidation** compares only the base Windows Installer package to an evaluation file. This ensures that, when starting a customization project, the initial file was created using the guidelines in that evaluation file. If it does not pass prevalidation, then the installation *may* work fine, but it may not be able to use all Windows Installer features.

- **Postvalidation** compares the base Windows Installer package and the changes made in a customization project against an evaluation file. In this case, the combination of the initial file and the subsequent modifications of the transform can produce different results than a comparison of just the base Windows Installer package. If the initial file was valid and postvalidation fails, the problems exist in the customization project. In some situations, advanced users may be able to use a transform file to make an initially invalid MSI file valid in conjunction with the transform.

What Do You Validate Against?

Tuner provides two files that you can validate against: the Windows 2000 Logo Program Suite and the Full MSI Validation Suite.

- The Windows 2000 Logo Program Suite is a subset of the Full MSI Validation Suite, and is used to certify that the installation meets the Microsoft standards for the Windows logo.
- The Full MSI Validation Suite is used to ensure that the installation meets all MSI standards.

Validation Procedure

Follow this procedure when validating an installation:

- In practice, you should prevalidate the base MSI file to ensure compliance to MSI standards before you begin creating a customization project.
- After you have finished your project, postvalidate it to make sure it is still compliant.
- If the base installation was compliant and the postvalidation fails, go back to the changes you made to determine what caused the validation problems.

For full details, consult the Windows Installer Help.

Prevalidating Windows Installer Packages

Prevalidation compares only the base Windows Installer package to an evaluation file. This ensures that, when starting a customization project, the initial file was created using the guidelines in that evaluation file.

Performing a Prevalidation

Task: ***To prevalidate a Windows Installer package:***

1. After creating a new transform file and specifying the base Windows Installer package, select the Prevalidation view from the checklist. The **Prevalidation View** appears, listing the name of the base Windows Installer Package.
2. Specify or browse to the Evaluation File you want to use.
3. If you want to run specific Internal Consistency Evaluators (ICEs), specify them in the ICEs to Run text box, separating them by semicolons if there are more than one (for example, ICE07;ICE13;ICE72). Otherwise, all ICEs are used.

4. Specify the result level by checking the Show “INFO” messages, Show “WARNING” messages, and/or Show “ERROR” messages check boxes. It is highly recommended that you check at least the Error check box so you are certain you are not suppressing results that occur in invalid packages.
5. Click the MSI Validation button ( ) on the toolbar, or click the Start button in the view.

Viewing the Prevalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Prevalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning ( ), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row’s associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the **Direct Editor** to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Directly Editing Packages.

Note • *If no errors appear in the results (providing you are displaying errors), then the package is valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

Handling Invalid Windows Installer Packages

Ideally, all Windows Installer packages will pass validation. Realistically, many will fail (generating errors). When a package fails validation, it means the package was not built to Microsoft’s specifications. It does not mean the installation does not work. However, there are a few things you can do when your package has validation errors:

Table 13-2 • Methods to Resolve Validation Errors

Solution	Explanation
Use Tuner to correct validation errors.	This involves opening the base package using Tuner and creating a transform file which contains your corrections.
Contact the installation vendor.	The company that created the installation (usually the same company that created the software) may be able to resolve the validation issues and provide you with a valid setup. Be sure to provide the validation report to vendors so they know where to focus.
Reconsider using the application.	Although it might be an extreme reaction to an invalid package, there may be compelling reasons not to use an installation not built to Microsoft guidelines.

Table 13-2 • Methods to Resolve Validation Errors (cont.)

Solution	Explanation
Ignore the problems and install anyway.	This is probably the most likely scenario. The invalid installation may not be worth trying to fix, or even have errors that you are concerned about. You could proceed and just use the installation as it is. From a practical standpoint, this may be your best option.

Note • *Most packages will also generate Warnings during validation. These can occur in valid packages, and many cannot be removed. Although the presence of Warnings does not make a package invalid, it is generally a good practice to eliminate Warnings (if possible).*

Postvalidating Transforms

Task: ***To postvalidate a Windows Installer package and the transform you are creating:***

1. Select the Postvalidation view from the checklist. The Postvalidation view appears, listing the name of the base Windows Installer Package.
2. Specify or browse to the Evaluation File you want to use.
3. If you want to run specific Internal Consistency Evaluators (ICEs), specify them in the ICEs to Run text box, separating them by semicolons if there are more than one (for example, ICE07;ICE13;ICE72). Otherwise, all ICEs are used.
4. Specify the result level by checking the Show “INFO” messages, Show “WARNING” messages, and/or Show “ERROR” messages check boxes. It is highly recommended that you check at least the Error check box so you are certain you are not suppressing results that occur in invalid packages.
5. Click the Transform Validation button ( ) on the toolbar, or click the Start button in the view.

Viewing the Postvalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning (), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row’s associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Directly Editing Packages.

Note • *If no errors appear in the results (providing you are displaying errors), then the package and transform are valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

Tip • *It is possible for a package that passed the prevalidation to fail the postvalidation. Remember changes made in the Setup Properties can affect your installation. If your package fails postvalidation, check all changes made in the Setup Properties for accuracy. To identify the original Setup Properties, you can create a new transform file that can be deleted at any time. Changes made using the Direct Editor can also affect your installation's functionality.*

Evaluation Files and Internal Consistency Evaluators

When you prevalidate the base Windows Installer package, or postvalidate the package and your transform, Tuner runs several Internal Consistency Evaluators (ICEs) contained in the specified evaluation file. If the base package or your transform and package does not pass one of these ICEs, Tuner reports the failure. If the problem is in the base package, you can contact the software vendor to report the problem.

Setup Organization

The Organization View allows you to modify two main parts of the installation that your end users will see: the default path and default company name, and the actual features that can, will, or will not be installed.

Topics in this section include the following:

- Changing a Feature's Visibility
- Setting the Initial State of a Feature
- Editing a Feature's Description
- Setting the Default Destination
- Setting the Default Organization
- Changing the Destination Variable
- Preventing Features from Displaying During Custom Installation
- Setting Feature Properties
- Using Feature Advertisement

Changing a Feature's Visibility

Task: ***To change the visibility of a feature:***

1. Under Organization in the checklist, select the Features View. This project's Features are listed in the second column.
2. Select the feature that you would like to change the visibility on. The Properties for that feature are listed.
3. Click in the Visible property in the Feature Properties grid and make a selection from the drop-down menu. Your options are:

Option	Description
Not Visible	The feature will not show up in the custom setup dialog box during installation.
Visible and Expanded	The feature will be displayed with all its subfeatures visible in the custom setup dialog box during installation.
Visible and Collapsed	The feature will be displayed in a collapsed state in the custom setup dialog box during installation.

Setting the Initial State of a Feature

Task: ***To set the Initial State of a feature:***

1. Under Organization in the checklist, select the Features view. This project's Features are listed in the second column.
2. Select the feature that you would like to change the initial state of. The Properties for that feature are listed.
3. Click in the Initial State property in the Feature Properties grid and choose one of the selections from the drop-down menu. Your selections are:
   - The feature is not installed: By default, the feature will not be installed during setup.
   - The feature is installed on the local drive: By default, the feature will be installed on the local drive during setup.
   - The feature is run from source, CD, or the network: By default, the feature will be run from the source, whether it be from the installation CD or from the network.
   - The feature is advertised: By default, the feature will be advertised, but not installed. Essentially, this is an on-demand option; a shortcut will be created during setup, and if the shortcut is clicked, the feature will then be installed from the source. This ensures features that may be unnecessary are not installed until they are needed, if ever. For more information, see Using Feature Advertisement.

Note • *The initial default settings run the Setup in a quiet mode.*

Editing a Feature's Description

Task: ***To edit a feature's description:***

1. Under Organization in the checklist, select the Features view. This project's Features are listed in the second column.
2. Select the feature that you would like to change the description of. The Properties for that feature are listed.
3. Click in the Description property in the Feature Properties grid.
4. Enter the new feature description in the Description value cell.

Setting the Default Destination

Task: ***To specify the Default Destination Path for an installation:***

1. Under Organization, select the Product Properties view from the checklist. The Product Properties view appears.
2. Click in the Default Destination Path property in the Product Properties grid.
3. Provide the path that you want to use as the Default Destination Path.

Caution • *Consult the*Product Properties View *help topic for important information about the Default Destination Variable and how it can be affected by changing this value.*

Setting the Default Organization

Task: ***To specify the default organization for the installation:***

1. Under Organization, select the Product Properties view from the checklist. The Product Properties View appears.
2. Click in the Company Name property in the Product Properties grid.

3. Enter the name you want to use as the default organization name. The organization name can be a maximum of 30 characters in length.

Changing the Destination Variable

Task: ***To specify the Destination Variable that holds the Default Destination Path:***

1. Under Organization, select the Product Properties view from the checklist. The Product Properties view appears.
2. Click in the Default Destination Variable property in the Product Properties grid.
3. Select the Default Destination Variable you want to use from the drop-down menu.

Caution • *Consult the Product Properties View help topic for important information about this variable.*

Preventing Features from Displaying During Custom Installation

Task: ***To prevent a feature from being displayed to your end users during a custom installation:***

1. Under Organization in the checklist, select the Features View. This project's Features are listed in the second column.
2. Select the feature that you would like to hide. The Properties grid for that feature appears.
3. Click in the Visible property in the Properties grid and select Not Visible from the drop-down menu.

The feature selected will not be visible during custom installation. Depending on the feature's Initial State, the feature may or may not be installed on the end user's system.

Setting Feature Properties

Task: ***To set Feature Properties:***

1. Select Features under Organization in the checklist.
2. Select the Feature that you want to edit. The Feature Properties view appears.
3. In the Description text box, enter a description that will be displayed when a feature is clicked in the Custom Setup dialog box

4. From the Visible drop down list, select an option to specify how the feature is presented to the end user in the Custom Setup dialog. The following options are available:

   - **Visible and Collapsed:** The feature will be displayed in the Custom Setup dialog with its subfeatures collapsed by default.
   - **Visible and Expanded:** The feature will be displayed in the Custom Setup dialog with its subfeatures expanded by default.
   - **Not Visible:** The feature will not be displayed to the end user in the Custom Setup dialog.

   Although an end user obviously cannot select or deselect an invisible feature, this property does not have any direct bearing on whether a feature is installed. In other words, a feature is not automatically installed if it is invisible; it just cannot be deselected if it would otherwise be installed, or selected if it should not be installed.

5. From the Initial State list, select an option to determine how (or if) the feature is installed during installation:

   - **The feature is not installed (INSTALLSTATE_ABSENT):** The feature will not be installed during setup.
   - **The feature is installed on the local drive (INSTALLSTATE_LOCAL):** The feature will be installed on the local drive during setup.
   - **The feature is run from source, CD, or network (INSTALLSTATE_SOURCE):** The feature will be run from the source, whether it is from the installation CD or from the network.
   - **The feature is advertised (INSTALLSTATE_ADVERTISED):** The feature will be advertised, but not installed. Essentially, this is an on-demand option; a shortcut will be created during setup, and if the shortcut is clicked, the feature will then be installed from the source. This ensures features that may be unnecessary are not installed until they are needed, if ever. For more information, see Using Feature Advertisement.

Using Feature Advertisement

Windows Installer supports many features of Windows 2000 and later platforms, including feature advertisement. This convenience enables any product feature to be in one of four installation states:

- The feature is not installed
- The feature is installed on the local drive
- The feature is run from source, CD, or the network
- The feature is advertised

When features are advertised, they are not actually installed on the local system. However, they appear to be, in that the appropriate shortcuts to launch the feature are present. The first time a user attempts to use a feature that is advertised, it is installed on the computer.

Feature advertisement is fully supported only on the Windows XP, Windows 2000, and Windows Me operating systems, though an updated shell is available for Windows 95, Windows 98, and Windows NT 4.0. To activate the shell for any of these platforms, Microsoft's Internet Explorer 4.01 Service Pack 1 must be installed and Active Desktop enabled.

Note • *For instructions on how to specify feature advertisement and the other installation states of a feature, see Setting the Initial State of a Feature.*

Configuring Package Content

Many modifications you can make in a transform file involve manipulating the original package contents. This includes the following:

- Files and Folders
- Registry Entries
- Shortcuts
- INI Files
- ODBC Resources
- NT Services

Files and Folders

From the Files and Folders view, you can perform all file operations in Tuner. This includes viewing files in the source MSI package, adding new files, preventing files from being installed, and removing added files.

Topics in this section include the following:

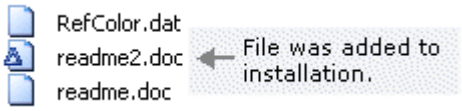
- Adding Files
- Displaying Files from the Base Windows Installer Package
- Preventing Installation of Files from the MSI
- Removing Added Files
- Storing Added Files

Adding Files

Task: ***To add files to an installation:***

1. Select the Files and Folders view from the checklist. The Files and Folders View appears.
2. Navigate to the location in the Source computer's directory tree that contains the file you want to add.
3. Select the file you want to add from the Source computer's files pane.

4. Drag the file to the appropriate folder in the Destination computer's folders tree. The file then appears in the Destination computer's files pane, with an icon indicating that it is an added file.



Displaying Files from the Base Windows Installer Package

Task: ***To display files from the base Windows Installer package in the Files and Folders view:***

1. Select Options from the Tools menu to display the Options dialog box.
2. Select the View Settings tab. The Option Dialog's View Settings pane opens.
3. Select the Display files from the original MSI package in addition to files added in the transform check box.
4. Click OK.

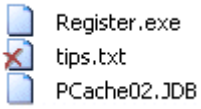
   When you return to the Files and Folders view, all files from the base Windows Installer package are displayed as well as the files added in the transform. By default, this option is enabled.

Preventing Installation of Files from the MSI

Task: ***To prevent files from the base Windows Installer package from being installed:***

1. Select Files and Folders from the checklist. The Files and Folders View opens.
2. Navigate to the folder which contains the file in the Destination computer's folders tree that you want to remove from the installation.
3. In the Destination computer's files pane, select the file you want to prevent from being installed and select Remove from the context menu.

   The actual file within the Windows Installer package is not deleted—only the entry in the File table is removed. The icon for the file changes to a computer with a red "X" over it.



   If you remove a file, and later want to restore it to the installation, simply right-click the file again and select Restore.

Note • *Key files in the base Windows Installer package are denoted with a key icon, and cannot be marked for deletion. Additionally, files contained within cabinet (.CAB) files are not displayed, and therefore cannot be marked for deletion.*

Removing Added Files

Task: ***To remove files you have added to an installation:***

1. Select Files and Folders from the checklist. The Files and Folders View opens.
2. In the Destination computer's folders tree, navigate to the folder containing the added file that you want to remove.
3. Select the file you want to remove from the Destination computer's files pane.
4. Select Remove from the context menu *or* press the Delete key.

Storing Added Files

When you add files to a transform, Tuner stores them in a CAB file with the same name as your transform. The added files are placed in the CAB and Tuner no longer maintains a reference to the original file location on the source computer. If you add additional files after saving the transform, the contents of the CAB file are extracted and recompressed along with the new files.

Because this mechanism relies on the presence of the CAB file, this file must be stored in the same location as the transform. If you move, modify, or delete the CAB file, Tuner will no longer be able to include the added files in the transform. You must then delete the files from the Files and Folders View and re-add them from their original locations, or locate the original CAB and place it back in the same folder as the transform.

Also, because the contents of the CAB file are uncompressed and recompressed when you add subsequent files to the installation, you must have sufficient disk space for this extraction when you save the transform.

Registry Entries

You can use the Registry view to create keys and values similar to how you use the Windows Registry Editor, or you can copy or drag and drop existing keys and values from the Source view.

Further, you can use this view to import an existing REG file using the Registry Import Wizard. You can also modify or delete registry keys that are part of the base installation. If you add new registry keys, they will always be installed.

Topics in this section include the following:

- Creating a Registry Key
- Creating a Registry Value
- Importing REG Files
- Removing Registry Information

Creating a Registry Key

Task: ***To specify a registry key that will be created on the target system:***

1. Select Registry from the checklist. The Registry View opens.
2. In the Destination Computer Registry View, select the key to which you want to add a value. All existing values for that key in the Destination Computer Registry Data pane are displayed.
3. To create a subkey, select a registry hive (such as HKEY_CURRENT_USER) or an existing key, point to New on the context menu, and select Key.

   A new key is created with the name "NewKey*n*" (where *n* is a successive number).
4. Enter a meaningful name now to rename the key. If you want to change the name later, right-click on the key and select Rename.
5. Right-click on the new key and select whether the key is to be created on installation, deleted on uninstallation, or both.

   Choose one of the following:

Option	Description
Create Key at Install	Creates the new registry key during installation if the key does not exist on the target machine.
Delete Key at Uninstall	Deletes the registry key during uninstallation, regardless of whether they key existed prior to the MSI's installation. This means that the key, and all its contents and sub-keys, will be removed regardless of whether other software information that is unrelated to this MSI exists. This can have a severe impact on other programs; only select this option if you are sure that the only software affected is the base MSI.
Both Create and Delete	Both of the above scenarios will occur.

   Your new key is created with an empty default string value. To modify the value name and data, see Creating a Registry Value.
6. To remove the key, right-click on it and select Delete.

Creating a Registry Value

Adding a New Value Name

Task: ***To add a new value name:***

1. Select Registry from the checklist. The Registry View opens.
2. Select the key to which you want to add a value from the Destination Computer Registry view. Existing values for the key are displayed in the Destination Computer Registry Data view.
3. Right-click in the list of values and select New String Value, New Binary Value, or New DWORD Value, depending on the type of data you want to register.

   A new empty value name is created with the name “New Value #n” (where *n* is a successive number).
4. Enter a meaningful name now to rename the value. To rename the value later, right-click on the value name and select Rename.

Note • *When creating binary values, Tuner automatically converts whatever input you provide into a binary value.*

Modifying the Value Data

Each new key has an empty default string value.

Task: ***To modify this or any value data:***

1. Right-click on a value name and select Modify. The Edit Data dialog box opens.
2. In the Value data text box, enter a new value or edit the existing value.
3. Click OK.

Importing REG Files

Tuner allows you to import any existing REG files that you may have created previously. To import a REG file you need to launch the Registry Import Wizard.

Task: ***To import a REG file:***

1. Select Registry from the checklist. The Registry View opens.
2. Right-click on a registry hive in the Destination Computer Registry View and select Import REG File. The Welcome panel of the Import REG File Wizard appears.
3. On the Welcome panel, click Next. The Import Registry File panel appears.

4. In the Registry File text box, either type the location of the registry file or browse to it, and click Next. The Import Conflict Options panel appears.
5. Select how you would like to handle duplicate registry data during the import. You have two options:

Option	Description
Overwrite the registry data	If any conflicts exist, the old registry keys will be overwritten by the new keys.
Do not overwrite the registry data	If duplicate keys are encountered, keep the existing keys.

6. After you have selected the method, click Import to continue. The Finishing Registry Import panel appears.
7. After the registry file has been scanned, click Finish to insert all entries from the REG file into the Destination Computer Registry view. You can then modify the entries.

Removing Registry Information

Task: ***To remove registry information:***

1. Select Registry from the checklist. The Registry View opens.
2. Navigate to the registry entry that you want to remove in the Destination Computer Registry View.
3. If you want to remove a value from a specific key, right-click on the value in the Destination Computer Registry Data pane and select Delete.
4. If you want to remove an entire key, right-click on the key in the Destination Computer Registry View pane and select Delete or press the Delete key.

Shortcuts

The Shortcuts view offers an integrated, visual method for adding shortcuts and program folders to the installation. Existing shortcuts can also be modified or removed.

Shortcuts can be placed in:

- folders already defined by the installation,
- standard folders that are predefined by the Windows Installer such as the Fonts folder, or
- new folders which you can create.

Each shortcut has several properties that specify the target program, hot key combination, icon, and other information necessary to launch the application. When you create a new shortcut, it will always be installed.

Note • *Shortcuts created in the transform are denoted by* *and shortcuts from the base Windows Installer package are denoted by* .

Topics in this section include the following:

- Creating Shortcuts
- Changing a Shortcut's Icon
- Change a Shortcut's Location
- Changing a Shortcut's Target
- Creating a Hot Key
- Removing Shortcuts
- Determining the Path of Changed Shortcuts

Creating Shortcuts

Task: ***To create a shortcut:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder in which you want to put the shortcut.
3. Right-click on the folder and select New Shortcut.
4. Provide a name for the shortcut.
5. Enter properties for the shortcut in the Properties Grid.

Changing a Shortcut's Icon

Task: ***To change the icon used for a shortcut:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder containing the shortcut you want to edit.
3. Select the Icon property from the Properties Grid.
4. Click the Change Icon button in the pane below the grid. The Change Icon dialog box opens.

5. Select one of the displayed icons or browse to the file that contains the icon you want to use for the shortcut.
6. After you have selected the appropriate icon, click OK. The new icon is now displayed to the left of the Change Icon button.

Change a Shortcut's Location

Task: ***To change a shortcut's location:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder containing the shortcut you want to move.
3. Select the shortcut and drag it to another folder in the Shortcuts tree.

Changing a Shortcut's Target

Task: ***To change a shortcut's target:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. In the Shortcuts folder tree, navigate to the folder containing the appropriate shortcut.
3. Select the Target property from the Properties Grid.
4. Select the appropriate Target from the list. The Target Type that is selected affects what you should enter in the Target property field:
   - **File from MSI Package & File from File System:** Provide the full path to the application or batch file.
   - **Destination Folder:** Select a folder name from the drop-down list. The list includes available folders on the target system, from the MSI package, and from the transform.
   - **Advertised Shortcut:** Enter the feature name. You can determine the name of the feature by going to the Direct Editor and selecting the Feature table. The list of features that you can target is listed in the Features column of the table.

Creating a Hot Key

A Hot Key is a combination of keys used to launch a shortcut instead of using the mouse.

Task: ***To create a hot key:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. Select the shortcut to which you want to add the Hot Key.

3. Click on the Hot Key field in the properties grid. The Hot Key dialog box opens.
4. Press the keys on the keyboard that you want to use for the shortcut. The shortcut appears in the dialog.
5. If the shortcut is correct, click OK. The dialog box closes and the shortcut's converted ASCII value appears as the value for the Hot Key.

***Note** • These four fields in the Shortcuts Property Box are required for creating a Hot Key: Icon, Target, Run, and Hot Key. When you are creating the Hot Key Combination, DO NOT use a keyboard combination already adopted by Microsoft (such as Ctrl+V, which is used for Paste). Otherwise, the shortcut will not work.*

Removing Shortcuts

Task: ***To remove a shortcut:***

1. Select Shortcuts from the checklist. The Shortcuts View opens.
2. Use the Shortcuts Tree to navigate to the shortcut you want to delete.
3. Right-click on the shortcut and select Delete to remove it.

Determining the Path of Changed Shortcuts

Task: ***To determine the actual path of a changed shortcut:***

1. Go to the Direct Editor and select the Directory table.
2. Find the shortcut Target directory, such as INSTALLDIR.
3. In that Directory row, find the value in the Directory_Parent column. In this example, the value is DIR26.
4. Look for a row in the Directory column that does not have a Directory_Parent entry. The directory in the row that has no value in the Directory_Parent column is the root directory. In this example, the root directory is TARGETDIR.

INI Files

Initialization (INI) files serve as a repository in which you can store and retrieve information between uses of your application. Typically INI Files contain key name-value pairs representing run-time options for applications. Some .ini files, such as Boot.ini and Wininit.ini, are used by the operating system.

INI files are divided into sections, each section containing keywords. Sections are divided by the square brackets surrounding them—[SectionName], for example. INI file keywords are the lowest level of organization in an .ini file. These keywords store data that must persist between uses of an application.

The INI Files View provides a graphical way for users to add, modify, or delete the contents of the IniFile Table. It displays the contents of the IniFile table from the source Windows Installer package and the transform.

Topics in this section include the following:

- Adding INI Files
- Importing Existing INI Files
- Adding Sections to INI Files
- Adding New Keys to INI File Sections
- Modifying INI File Keys, Values, and Actions
- Removing INI Files
- Removing Sections from INI Files
- Removing INI File Section Keys

Adding INI Files

Task: ***To add an INI file to your transform file:***

1. Select INI Files from the checklist. The INI Files View opens.
2. Right-click on the appropriate destination folder in the INI File tree and select New IniFile.
3. To rename the new INI file, select IniFile#.ini and select Rename from the context menu.
4. To rename the new INI section, select NewSection#.ini and select Rename from the context menu.
5. With the new INI section selected, enter a Key name, Value, and Action for the default INI key value. See Modifying INI File Keys, Values, and Actions.
6. Add additional Sections and Keys, as described in Adding Sections to INI Files and Adding New Keys to INI File Sections.

Importing Existing INI Files

Task: ***To import an existing INI file:***

1. Select INI Files from the checklist. The INI Files View opens.
2. Right-click on the appropriate destination folder in the INI File tree and select Import INI File from the context menu. The Welcome Panel of the Import INI File Wizard opens.
3. Click Next. The Import INI File Panel opens.
4. Enter or browse to the INI file you want to import. Click Next. The Import Conflict Options Panel opens.

5. Select how you want to handle duplicate keys and values. Click Import.
6. Once the INI file has been imported, click Finish.

The imported INI file appears under the selected destination folder. You can then make further adjustments to it as needed.

Adding Sections to INI Files

Task: ***To add a section to an INI file:***

1. Select the INI Files View from the checklist.
2. Right-click on the appropriate INI File in the INI File tree and select New Section. A new INI section, named NewSection1, is created under the selected INI file, complete with a default Key name, Value, and Action.
3. To rename the new section, select NewSection1 and select Rename from the context menu.
4. With the new INI section selected, enter a Key name, Value, and Action for the default INI key value. See Modifying INI File Keys, Values, and Actions.
5. Add additional Keys to this new INI file section as necessary. See Adding New Keys to INI File Sections.

Adding New Keys to INI File Sections

Task: ***To add a new INI file key:***

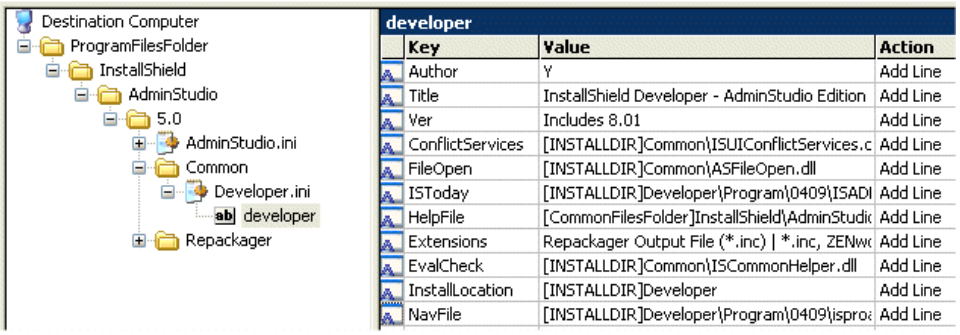
1. Select the INI Files View from the checklist.
2. Expand the listing of the INI file that you would like to edit so that all of its sections are displayed.
3. Select the INI file section that you would like to edit. That section's defined keys are listed on the right.
4. Right-click in the key listing and select Add from the menu. A new key is added to the key listing with a default Key name, Value and Action.
5. Edit the Key name, Value, and Action for this new Key. See Modifying INI File Keys, Values, and Actions.

Modifying INI File Keys, Values, and Actions

INI files contain key name-value pairs representing run-time options for applications. To define and modify the key names, values, and /or actions, perform the following steps.

Task: ***To modify keys, values, and/or actions in INI files:***

1. Select the INI Files View from the checklist.
2. In the INI File tree, select the INI file that you want to edit, and expand the listing so that you can select the appropriate section. That section's keys, values, and actions are listed:



3. Modify the Key name, Value, and Action for each key, as necessary.

Option	Description
Key	The name of the key. This should be entered in the exact way you want it to appear in the target INI file.
Value	The key's value. Windows Installer properties can be used in your keyword's value. To do this, surround the property with square brackets—[INSTALLDIR], for example. For a comprehensive list of Windows Installer properties, refer to the Property Reference topic in the Windows Installer help.
Action	The action the key performs. Select this from the list in the property sheet. The available options are: • **Add Tag:** Creates a new entry or appends a new comma-separated value to an existing entry. • **Create Line:** Creates a .ini entry only if the entry does not already exist. • **Add Line:** Creates or updates a .ini entry.

Removing INI Files

Task: ***To remove an INI file:***

1. Select the INI File view from the checklist.
2. From the INI File tree, right-click on the INI file you want to delete and select Remove.

Removing Sections from INI Files

Task: ***To remove a section from INI file:***

1. Select the INI File view from the checklist.
2. From the INI File tree, right-click on the section you want to delete and select Remove.

Removing INI File Section Keys

Task: ***To remove an INI file section key:***

1. Select the INI Files view from the checklist.
2. Select the INI File section that contains the key you want to delete from the INI Files tree.
3. Right-click the key you want to remove and select Delete.

ODBC Resources

Open Database Connectivity (ODBC) Resources are ones that involve interaction with databases. Tuner allows you to view existing ODBC Data Sources, ODBC Drivers, and ODBC Translators.

Topics in this section include the following:

- Adding New Data Sources
- Adding New ODBC Data Source Attributes
- Adding New ODBC Driver Attributes
- Editing ODBC Data Source Attributes
- Editing ODBC Driver Attributes
- Removing Existing ODBC Data Sources
- Removing ODBC Driver Attributes
- Removing ODBC Data Source Attributes

Adding New Data Sources

Task: ***To add a new ODBC Data Source:***

1. Select ODBC Resources from the checklist. The ODBC Resources View opens.
2. Right-click either ODBC Data Sources group or one of its children groups from the ODBC Resources tree and select New Data Source from the context menu. The ODBC Data Source dialog box opens.
3. Select the required data source and click OK.

Caution • *If you are adding an ODBC Data Source that does not exist on your computer, type the name of the Data Source into the ODBC Data Source dialog. Keep in mind that adding a data source to a Windows Installer package that does not contain the corresponding driver may render the package useless.*

Adding New ODBC Data Source Attributes

Task: ***To add a new ODBC data source attribute:***

1. Select ODBC Resources from the checklist. The ODBC Resources View opens.
2. Select the ODBC Data Source to which you want to add a new attribute from the ODBC Resources tree. The property grid for the selected ODBC Data source opens.
3. Right-click in the property grid and select Add. A new attribute is listed, with the default values of ATTRIBUTE and NULL_VALUE.
4. Enter information for the new attribute.

Adding New ODBC Driver Attributes

Task: ***To add a new ODBC driver attribute:***

1. Select ODBC Resources from the checklist. The ODBC Resources View opens.
2. Select the ODBC driver to which you want to add a new attribute from the ODBC Resources tree. The property grid for the selected ODBC driver appears.
3. Right-click in the property grid and select Add from the context menu. A new attribute is listed, with the default values of ATTRIBUTE and NULL_VALUE.
4. Enter information for the new attribute.

Editing ODBC Data Source Attributes

Task: ***To edit an ODBC data source attribute:***

1. On the ODBC Resources View, select the ODBC data source that contains the attribute you want to modify from the ODBC Resources tree. The property grid for that data source appears.
2. In the properties grid, edit the appropriate attribute.

Editing ODBC Driver Attributes

Task: ***To edit an ODBC driver attribute:***

1. On the ODBC Resources View, select the ODBC driver that contains the attribute you want to modify from the ODBC Resources tree. The property grid for that ODBC driver appears.
2. In the properties sheet, edit the appropriate attribute.

Removing Existing ODBC Data Sources

Task: ***To remove an existing ODBC Data Source:***

1. Select the ODBC Resources view from the checklist.
2. Right-click on the ODBC data source you want to remove from the ODBC Resources tree and select Delete.

Removing ODBC Driver Attributes

Task: ***To remove an ODBC driver attribute:***

1. Select the ODBC Resources view from the checklist.
2. Select the ODBC driver that contains the attribute you want to delete from the ODBC Resources tree.
3. In the properties sheet, right-click on the attribute you want to remove and select Delete.

Removing ODBC Data Source Attributes

Task: ***To remove an ODBC data source attribute:***

1. Select the ODBC Resources view from the checklist.
2. Select the ODBC data source that contains the attribute you want to delete from the ODBC Resources tree.
3. In the properties sheet, right-click on the attribute you want to remove and select Delete.

NT Services

The NT Services view provides a way to change parameters for NT Services included in the base Windows Installer package. Topics in this section include the following:

- Setting NT Service Arguments
- Setting NT Service Dependencies
- Setting the NT Service Description
- Setting the NT Service Display Name
- Setting the NT Service Error Control Level
- Setting the NT Service Load Order Group
- Setting the NT Service Overall Install Result
- Setting the NT Service Start Type
- Setting NT Service Start Name and Password
- Setting the NT Service Type

Setting NT Service Arguments

Task: ***To set NT service arguments:***

1. Select the NT Services from the checklist. The NT Services View opens.
2. Double-click the current value in the properties grid and modify it as needed.

Setting NT Service Dependencies

Task: ***To set NT service dependencies:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Dependencies value in the properties grid and add the names of services or load ordering groups that must be started prior to this service.

Note • *If the dependency is on a load ordering group, the service can start if at least one member of the load ordering group is running after an attempt is made to start all load ordering group members.*

Setting the NT Service Description

Task: ***To set the NT service description:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Description value in the properties grid and modify it as needed.

Setting the NT Service Display Name

Task: ***To set the NT service display name:***

1. Select the NT Services view from the checklist.
2. Double-click the current Display Name value and modify it. The display name can be up to 256 characters in length.

Setting the NT Service Error Control Level

Task: ***To set the NT service error control level:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Error Control value in the properties grid to access the pop-up menu. The possible values are as follows:

Value	Description
Ignore Error	Logs the error and continues with service startup.
Normal Error	Logs the error, displays an error message, and continues with service startup.
Critical Error	Logs the error (if possible) and restarts the system with the last configuration known to be good. If the last-known-good configuration is the one that caused the error, fail the startup.

Setting the NT Service Load Order Group

Task: ***To set the NT service load order group:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Load Order Group value in the properties grid and modify it as needed. If this service does not belong to a group, leave this value blank.

Setting the NT Service Overall Install Result

Task: ***To set the NT service overall install result:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Overall Install value in the properties grid to access the drop-down menu.
3. Select either Continue overall install if service fails to install or Fail overall install if service fails to install as this property's value.

Setting the NT Service Start Type

Task: ***To set the NT service start type:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Start Type value in the properties grid to access the drop-down menu.
3. Select the desired Start Type from the following possible values:

Option	Description
Automatic	The service starts during system startup.
Manual	The service is only started when the service control manager calls the StartService function.
Disabled	The service is not started.
Start at Boot Time	The driver is started by the operating system loader. (Device driver only)
Started by the System	The driver is started by calling the IoInitSystem function. (Device driver only)

Setting NT Service Start Name and Password

Task: ***To set the NT service start name and password:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Start Name value in the properties grid and provide the name under which this service will run.
3. Click the current Password value in the properties grid and provide the password associated with the Start Name.

Setting the NT Service Type

Task: ***To set the NT service type:***

1. Select NT Services from the checklist. The NT Services View opens.
2. Double-click the current Service Type value in the properties grid to access the drop-down menu.
3. Select the desired Start Type from the following possible values:

- Service that Runs in its Own Process
- Service that Shares a Process with Others

4. Optional: click the current Interact with Desktop value in the properties grid to access the drop-down menu and specify whether the service needs to interact with the desktop.

Working with Dialogs

When customizing the Windows Installer package, you may want to disable particular panels that appear during the installation, administrative, patch, or maintenance sequences. You can do so from the Dialogs view. This view contains a list of each of the four installation modes (installation, administrative, maintenance, and patch), with the associated dialogs that appear as part of the UI sequence during the selected mode. You can enable or disable these dialogs by either the check box to the left of the dialog name, or by using the Show and Hide buttons.

Topics in this section include the following:

- Hiding Dialogs During UI Sequences
- Restoring Dialog Sequences
- Suppressing the License Agreement Dialog Box
- Disabling Custom Setups
- Editing Dialog Properties
- Dialogs View vs. Command-Line Options
- Dialog Suppression Issues

Hiding Dialogs During UI Sequences

Task: ***To hide a dialog box during a UI sequence:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the dialog you want to hide.
3. From the Dialogs list, clear the check mark next to the dialog you want to hide. Alternatively, select the dialog from the list and either click Hide or press the Space Bar.
4. If necessary set the properties for the dialog you are removing to preserve the UI sequence integrity. The Dialog Properties dialog box automatically appears if it is necessary to edit properties.

Restoring Dialog Sequences

Task: ***To restore dialogs in a UI sequence:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the dialog you want to restore.
3. From the Dialogs list, check the box next to the dialog you want to restore. Alternatively, select the dialog from the list and either click Show or press the Space Bar.

Suppressing the License Agreement Dialog Box

Using the Dialogs view, it is possible to suppress the license acceptance dialog. This involves both turning off its display, and providing the value that the setup will interpret as acceptance of the agreement.

Note • *This procedure assumes the original Windows Installer setup was created using InstallShield Editor. If another setup authoring application was used, the names of dialogs and properties may not be the same. The same general procedure still applies.*

Task: ***To use a transform to suppress display of the license acceptance dialog:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the License Agreement dialog. Typically, this is only in the Installation Sequence.
3. Select the **LicenseAgreement** dialog from the Dialogs list and click Hide.
4. When the Dialog Properties dialog box appears, change the AgreeToLicense value to Yes.
5. Click OK to dismiss the dialog.

When an end user runs the installation using this transform, the License Agreement dialog will not appear.

Disabling Custom Setups

You can use the Dialogs view to prevent users from performing custom setups. However, this requires not only the elimination of the Custom Setup panel during installation, but also the Setup Types panel. Additionally, you must ensure you have configured the features you want installed, as your end user will have no way to override them.

***Note** • This procedure assumes the original Windows Installer setup was created using InstallShield Editor. If another setup authoring application was used, the names of dialogs and properties may not be the same. The same general procedure still applies.*

Task: ***To use a transform to disable a custom setup:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the SetupType dialog. Typically, this is only in the Installation Sequence.
3. Select the SetupType dialog from the Dialogs list and click Hide.
4. When the Dialog Properties dialog box appears, change the ADDLOCAL value to ALL.
5. Click OK to dismiss the dialog.
6. Select the CustomSetup dialog from the Dialogs list and click Hide.
7. When the Dialog Properties dialog box opens, change the _BrowseProperty value to INSTALLDIR.
8. Click OK to dismiss the dialog.

When an end user runs the installation using this transform, the user will not have the option to perform a custom setup.

Editing Dialog Properties

Task: ***To edit properties for a UI sequence dialog:***

1. Select Dialogs from the checklist. The Dialogs View opens.
2. Using the Installation Mode list, select the UI sequence containing the dialog box containing properties you want to edit.
3. From the Dialogs list, select the appropriate dialog.
4. Click Properties. The Dialog Properties dialog box opens.

   ***Note** • The Properties button is only enabled when you have selected a dialog box containing editable properties.*
5. From the Dialog Properties dialog box, double-click the value cell for the property you want to edit.
6. Change the property value as necessary, and click OK.

Dialogs View vs. Command-Line Options

Generally, you should use the Dialogs View when you are still planning on displaying some panels during UI sequences. Typically, you may want to remove the License Agreement panel or the ability for a user to perform a custom setup, and these can both be accomplished easily from the Dialogs View.

However, consider using the Windows Installer command-line options (particularly `/qn`) when you want to eliminate the user interface entirely.

Dialog Suppression Issues

When suppressing dialog box display, it is important to consider some implications of your actions. Particularly, when removing a dialog from a user interface sequence, there may be properties normally set by that dialog. For example, the LicenseAgreement dialog has a radio button which can set a property to Yes or No, depending on whether you agree to the terms in the license. The value of this property also determines whether the installation should continue. Therefore, if you remove the LicenseAgreement dialog from a sequence, you need to use the Dialog Properties dialog box to set the value of this property so the installation can continue.

Beyond setting necessary properties, you also should consider how features are displayed. For example, you may want to disable custom setups via the transform file. However, you must ensure each feature you want installed is configured to be installed; your end users will have no way to override the choices by performing a custom setup.

Configuring Additional Server Locations

If you install from a network server, and if you install features to run from the server or that will be advertised for installation on their first use, the applications may need access to the server sometime after the initial installation. The applications may also require access to the server if a file is deleted or becomes corrupt, as the application can copy the problematic file(s) automatically from the server.

Topics in this section include the following:

- Adding Additional Server Locations
- Modifying Server Locations
- Removing Server Locations
- Reordering Server Locations

Adding Additional Server Locations

Task: ***To add an additional server location:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Click the New button ( ) in the Addition Server Location Paths window.
3. Enter or click the Browse (...) button and browse to the server location.

Note • *The validity of the server location is determined when the installation needs to access the server remotely. In other words, if a server is not available, or if you added an invalid server, the entry will be ignored if the resource is needed.*

Modifying Server Locations

Task: ***To modify an additional server location entry:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Select the server entry from the Addition Server Location Paths window and either edit the entry or use the Browse (...) button to browse to desired location.

Removing Server Locations

Task: ***To remove an additional server location:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Select the server entry from the Addition Server Location Paths window.
3. Click the Remove button ( ).

Reordering Server Locations

Task: ***To change the order in which additional server locations are accessed:***

1. Select Server Locations from the checklist. The Server Locations View opens.
2. Select a server entry from the Addition Server Location Paths window.
3. Depending on whether you want to promote the server location or demote it, click the up and down arrow buttons at the top right of the view ( ). You can also use the Alt+Up Arrow and Alt+Down Arrow keyboard shortcuts.
4. Repeat with other server location entries as necessary.

Changing Add/Remove Program Settings

The Windows 2000 and Windows XP Add/Remove Programs in Control Panel differs from the previous Windows operating systems in many ways. Depending on how the Windows Installer setup is configured, the user has the option of removing, repairing, or changing the installation with the click of a button.

Windows 2000 and XP users are also able to access additional information in Add/Remove Programs in Control Panel not available on previous platforms. With this information, it is easier for your end users to find technical support links and phone numbers, product update information, and information about your company.

Topics in this section include the following:

- Changing Add/Remove Programs Properties
- Disabling the Modify, Remove, or Repair Buttons

Changing Add/Remove Programs Properties

Task: ***To change properties in the Windows 2000 or XP Add/Remove Programs:***

1. Select Add/Remove Programs from the checklist. The Add/Remove Programs View opens.
2. Double-click the value for the property you want to change.
3. Either enter the information into the properties grid, or use the drop-down menu to select a value.

Disabling the Modify, Remove, or Repair Buttons

Task: ***To disable the Modify, Remove, or Repair buttons in Windows 2000 or XP's Add/Remove Programs in Control Panel:***

1. Select Add/Remove Programs from the checklist. The Add/Remove Programs View opens.
2. Double-click the appropriate Disable Modify/Remove/Repair Button property from the properties grid.
3. Use the drop-down menu to change the value to No.

Customizing Setup Properties

Even though Tuner provides you views to customize many areas of the Windows Installer package, it may be necessary to edit property values that are not available elsewhere. The Setup Properties view exposes the entries in the properties table (the underlying structure of Windows Installer packages). You can also add your own custom properties here.

Topics in this section include the following:

- Adding Custom Setup Properties
- Adding and Editing Comments
- Removing Custom Setup Properties
- Modifying Setup Properties

Adding Custom Setup Properties

Task: ***To add a new setup property:***

1. Select Setup Properties from the checklist. The Setup Properties View opens.
2. Right-click in the properties grid and select Add. A New property is added to the bottom of the list with the a Property Name of NEW_PROPERTY and a Value of NULL_VALUE.
3. Provide a new name for your property. If you want to change it later, click on the property name to edit it.
4. Enter the property's value.

Adding and Editing Comments

Tuner supports the ability to add comments to each property available in the Setup Properties view. This provides a way to clarify what specific properties do, and to enter any important information that you may need later. The original software vendor may have used InstallShield Editor to include comments in the original Windows Installer package.

Task: ***To add or edit a comment for a property:***

1. Select the Setup Properties view from the checklist.
2. Double-click in the comment column for the property to which you want to add or edit the comment.
3. Add or edit the comment as appropriate.

Removing Custom Setup Properties

Task: ***To remove a custom setup property:***

1. Select Setup Properties from the checklist. The Setup Properties View opens.
2. Right-click on the property you want to remove and select Delete.
3. Confirm the deletion.

Modifying Setup Properties

Task: ***To modify the property value of a setup:***

1. Select Setup Properties from the checklist. The Setup Properties View opens.
2. Select the property that you want to modify.
3. Double-click the property's value and edit it in the grid.

Make sure you are entering valid values when you modify properties, otherwise validation or installation errors may result.

Preparing Packages for Distribution

The final step in creating a customization involves two parts. First, you should postvalidate your transform and base Windows Installer package. This ensures that you have not introduced any errors into the installation, and may help you verify that you have corrected errors that existed in the base package. Secondly, you need to actually package the transform and base package for distribution. These tasks are accomplished using the Postvalidation view and Package view, respectively.

Topics in this section include the following:

- Copying the Installation to a Network Location
- Copying the Installation to an FTP Server
- Creating a Package Definition File (PDF)
- Creating an SMS File
- Instructing SMS to Create a Management Information Format File at Deployment Time
- Deploying Windows Installer Setup Packages with Systems Management Server 2.0
- Creating a Setup.exe File for the Package and Transform
- Additional Setup.ini Parameters

Copying the Installation to a Network Location

Task: ***To copy your installation to a Network location during packaging:***

1. From the checklist, select Package, and then select Location from the second column. The Location View of the Package View opens.
2. Select the Network Location option button.
3. Specify the network location, or click Browse to locate it.

   When you create your package, the appropriate files will be copied to the network location you specified.

Note • *If the transforms are copied to the same location as the original MSI, only the transform, setup.exe, setup.ini, and Windows Installer engines are copied.*

Copying the Installation to an FTP Server

Task: ***To copy your installation to an FTP server during packaging:***

1. From the checklist, select Package, and then select Location from the second column. The Location View of the Package View opens.
2. Select the FTP Server option button.
3. Specify the FTP server name (FTP Location), the UserName, and Password for the FTP server.

   When you create your package, the appropriate files will be copied to the FTP server you specified.

Creating a Package Definition File (PDF)

Task: ***To create a Package Definition File (PDF):***

1. From the checklist, select Package, and then select SMS from the second column. The SMS View of the Package View opens.
2. Select the Create Package Definition File check box.

   When you create your package, the resulting file has a .PDF extension. Here is a sample ORCA.PDF file:

```
[PDF]
Version=1.0

[Package Definition]
Product=Orca
Version=1.20.1827.1
Comment=Microsoft
SetupVariations=Typical, Automated

[Typical Setup]
CommandName = Typical Installation
CommandLine = msiexec /i Orca.msi
UserInputRequired = TRUE
SynchronousSystemExitRequired = TRUE
SupportedPlatforms = Win 9x, Win NT (i386)

[Automated Setup]
CommandName = Automated Installation
CommandLine = msiexec /i /q Orca.msi
UserInputRequired = FALSE
SynchronousSystemExitRequired = TRUE
SupportedPlatforms = Win 9x, Win NT (i386)

[Setup Package for Inventory]
InventoryThisPackage=FALSE
```

Creating an SMS File

Task: ***To create a SMS file for SMS 2.0 or later:***

1. From the checklist, select Package, and then select SMS from the second column. The SMS View of the Package View opens.
2. Select the Create SMS file check box.

   When you create your package, the resulting file has an .SMS extension. Here is a sample ORCA.SMS file:

```
[PDF]
Version=2.0

[Package Definition]
MIFFilename=Sample.MIF
Name=Orca
Publisher=Microsoft
Version=1.20.1827.1
Language=English
Programs=Typical, Automated, Test

[Typical]
Name = Typical
CommandLine = msiexec /i Orca.msi
UserInputRequired = TRUE
UninstallKey={8FC71000-88A0-4B41-82B8-8905D4AA904C}
AfterRunning=ProgramRestart
SupportedClients = Win 9x, Win NT (i386)

[Automated]
Name = Automated
CommandLine = msiexec /i /q Orca.msi
UserInputRequired = FALSE
UninstallKey={8FC71000-88A0-4B41-82B8-8905D4AA904C}
AfterRunning=ProgramRestart
SupportedClients = Win 9x, Win NT (i386)

[Test]
Name = Test
CommandLine = msiexec /i Orca.msi EXECUTEMODE=None
UserInputRequired = FALSE
AfterRunning=ProgramRestart
SupportedClients = Win 9x, Win NT (i386)
```

Instructing SMS to Create a Management Information Format File at Deployment Time

Task: ***To instruct SMS to create a Management Information Format (MIF) file at deployment time:***

1. From the checklist, select Package, and then select SMS from the second column. The SMS View of the Package View opens.
2. Select the Create SMS file check box.
3. Provide the name of the application you are installing in the Install MIF Filename field.
4. Provide the name of the application to be uninstalled in the Uninstall MIF Filename field.
5. Enter the serial number for the product in the Serial Number field.

The resulting file has a .MIF extension.

Deploying Windows Installer Setup Packages with Systems Management Server 2.0

Task: ***To deploy Windows Installer setup packages with SMS 2.0:***

Perform the steps detailed in the Microsoft White Paper: Deploying Windows Installer Setup Packages with Systems Management Server 2.0.

Creating a Setup.exe File for the Package and Transform

Task: ***To create a Setup.exe file to begin the installation of your base package and transform:***

1. From the checklist, select Package, and then select Setup from the second column. The Setup View of the Package View opens.
2. Select the Create Installation Launcher (Setup.exe) check box.
3. Specify whether you want to include the Windows 95/98 or Windows NT MSI engines.
4. Provide any command-line arguments for your installation.
5. Save your transform. The Setup.exe file is stored in the directory specified in the Location panel of the Packaging Wizard, or in the Location view within the Package view.

Additional Setup.ini Parameters

You can modify the Setup.ini file generated by Tuner for added functionality. The following two parameters can be included in the [Startup] section of the Setup.ini:

Table 13-3 • Setup.ini File [Startup] Section Parameters

Parameter	Description
SuppressWin2k	If you add this line and set its value to “Y” or “y” (SuppressWin2k=Y or SuppressWin2k=y), Setup.exe will not display a message on the target system stating that an older version of Windows Installer is installed.
SuppressReboot	If you add this line and set its value to “Y” or “y” (SuppressReboot=Y or SuppressReboot=y), Setup.exe will delay the reboot typically required by the installation of a newer version of Windows Installer on the target machine. If the application’s setup requires a reboot, this will occur normally.

Directly Editing Packages

Windows Installer packages are relational databases consisting of dozens of interrelated tables. These tables reflect the application’s features, components, relationship between features and components, registry information, and user interface.

The Direct Editor allows you to edit values in the MSI tables of the base Windows Installer package and store them in your transform. As you change values elsewhere in your transform, those changes are reflected in the Direct Editor, and vice versa. The complete list of MSI tables contained in the installation package is displayed in the left pane. When you select a table, the contents are displayed in the right pane.

Resizing Table Columns in the Direct Editor

When you initially open the Direct Editor, the selected table’s columns are listed in a compact format so that the maximum number of columns are displayed.

To automatically resize a column so that its width matches that of its longest entry, double-click on the column heading. This new column width setting is automatically saved and will be implemented the next time you view this table column in the Direct Editor.

Sorting Table Columns in the Direct Editor

To sort a table column, click the column heading once. The order will toggle between ascending and descending.

Adding a New Record Using the Direct Editor

Task: ***To add a new record in the Direct Editor:***

1. Select Direct Editor from the checklist. The Direct Editor opens.
2. Select the table you want to add a row to from the table tree.
3. Click on the last row in the table's grid, or click anywhere in the table and press the Insert key.

   A new record is added to the grid using a unique key for the record (use `ColumnName<n>` as the template). You can then modify the record.

Caution • *To ensure files are deployed on the target system, add files in the Files and Folders View, rather than to the file table. Files added to the File table are not physically added to the transform.*

Finding and Replacing Using the Direct Editor

The Direct Editor supports find and replace throughout all tables. The following commands are available:

Table 13-4 • Direct Editor Commands

Command	Description
Find (Ctrl+F)	Displays a standard Find dialog box and allows you to search for a string in all tables.
Find Next (F3)	Allows you to search for the next occurrence of a given string in all tables.
Replace (Ctrl+H)	Displays a standard Search and Replace dialog box and allows you to search for data in all the MSI tables and gives you the option to replace the data.

Launching the Direct Editor from the Validation Tab

When performing a Prevalidation or Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error (), a Warning (), or an Informational Message (), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red. This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.

Resizing Table Columns in the Direct Editor

When you initially open the Direct Editor, the selected table's columns are listed in a compact format so that the maximum number of columns are displayed.

To automatically resize a column so that its width matches that of its longest entry, double-click on the column heading. This new column width setting is automatically saved and will be implemented the next time you view this table column in the Direct Editor.

Sorting Table Columns in the Direct Editor

To sort a table column, click the column heading once. The order will toggle between ascending and descending.

Tuner Reference

Reference information for Tuner is divided into the following sections:

Table 13-5 • Tuner Reference Sections

Section	Description
User Interface Reference	Contains general information about the Tuner user interface, including menus, toolbars, keyboard shortcuts, windows, and dialog boxes.
Tuner Views	Contains detailed reference information about each Tuner view.
Import INI File Wizard	Provides reference information for the Import INI File Wizard. This is the same information accessible by clicking Help in the Wizard.
Import REG File Wizard	Provides reference information for the Import REG File Wizard. This is the same information accessible by clicking Help in the Wizard.
Packaging Wizard	Contains panel-by-panel reference information for the Tuner Packaging Wizard. Also includes information about additional Setup.ini options.

User Interface Reference

This book describes the user interface components, such as menu items, the toolbar, views, and dialog boxes you will encounter throughout Tuner.

- Menus and Toolbar
- View Bar
- Checklist
- Output Window
- Customize Dialog Box

- Properties Dialog Box

These topics are the same detailed documentation that is displayed when you press the F1 key or click the Help button while working in a dialog.

Menus and Toolbar

The Tuner user interface has several menus, each of which contain different commands. The functionality of each command is described below. Additionally, the Toolbar provides quick access to some of the frequently used commands; the corresponding toolbar buttons are listed with the appropriate commands.

Table 13-6 • Tuner Menus and Toolbar

Menu	Command	Shortcut	Toolbar Button	Description
File	**New**	Ctrl+N		Takes you to the Create a New Transform File area of the Tuner Start Page view.
File	**Open**	Ctrl+O		Takes you to the Open a Recent Transform File area of the Tuner Start Page view.
File	**Close**			Closes the currently open transform file.
File	**Save**	Ctrl+S		Saves the current transform file.
File	**Save As**			Prompts you to name the transform file you are saving.
File	**Properties**			Displays properties for the current transform, including the name and location of the base Windows Installer package.
File	**[1], [2], [3], or [4]**			Allows you to select one of the four most recently accessed transforms.
File	**Exit**			Exits Tuner.
Edit	**Undo**	Ctrl+Z		Undoes the last action.
Edit	**Cut**	Ctrl+X		Removes the selected text to the clipboard.
Edit	**Copy**	Ctrl+C		Copies the selected text to the clipboard.
Edit	**Paste**	Ctrl+V		Pastes the contents of the clipboard to the current cursor location.
View	**Output Window**			Toggles the Output window.

Table 13-6 • Tuner Menus and Toolbar (cont.)

Menu	Command	Shortcut	Toolbar Button	Description
View	**Check List**			Toggles the checklist.
View	**View Bar**			Toggles the View Bar.
View	**Header Bar**			Toggles the Header Bar.
View	**Toolbar**			Toggles the Toolbar.
View	**Status Bar**			Toggles the Status Bar.
Go	**Previous View**			Takes you to the previous view in the checklist.
Go	**Next View**			Takes you to the next view in the checklist.
Go	**Back**			Moves you to the last view.
Go	**Forward**			Moves you to the next view.
Go	**Start Page**			Takes you to the Tuner Start Page.
Go	**Help**			Takes you to the Help view.
Go	**Package Validation**			Takes you to the Prevalidation view.
Go	**Organization**			Takes you to the Product Properties and Features views.
Go	**System Configuration**			Takes you to the Files, Registry, Shortcuts, INI Files, NT Services, and ODBC Resources views.
Go	**Application Configuration**			Takes you to the Server Locations, Setup Properties, Dialogs, and Add/Remove Programs views.
Go	**Package Preparation**			Takes you to the Postvalidation and Package views.
Go	**Additional Tools**			Takes you to the Direct Editor view.
Project	**MSI Validation**			Runs prevalidation on the MSI file.

Table 13-6 • Tuner Menus and Toolbar (cont.)

Menu	Command	Shortcut	Toolbar Button	Description
Project	**Transform Validation**			Runs postvalidation on the MST file.
Project	**Transform Summary Information**			Launches the Transform Summary dialog.
Project	**Test**	Ctrl+T		Allows you to test your custom installation without actually installing.
Project	**Run**	Ctrl+F5		Performs the actual installation of the Windows Installer package and your transform.
Project	**Package**			Packages the transform and Windows Installer package based on the current packaging settings.
Project	**Packaging Wizard**	Ctrl+F7		Launches the Packaging Wizard.
Project	**Stop**			Halts in-progress validation or packaging.
Tools	**Customize**			Allows you to customize toolbars and menus.
Tools	**Options**			Allows you to specify the default locations for MSI and MST files.
Help	**Help Library**			Launches the online help (which you are currently viewing).
Help	**MSI Help**			Brings up the MSI online help.
Help	**ReadMe**			Displays the AdminStudio ReadMe file.
Help	**Support Central**			Connects to the AdminStudio Support Web site.
Help	**About Tuner**			Displays information about Tuner, including the version and copyright notice.

View Bar

The View Bar, when visible, is located at the far left of the user interface. It provides quick shortcuts to important areas of Tuner, and can be toggled on and off using the View Bar command under the View menu, or from the corresponding toolbar button.

There are three different View Bars available:

InstallShield

This View Bar gives you quick access to the Tuner Start Page and the Help view.

Checklist Steps

This View Bar provides access to each of the checklist steps, which include Package Validation, Organization, System Configuration, Application Configuration, Package Preparation, and Additional Tools views.

Views

This View Bar gives you quick access to each view in Tuner.

Checklist

The checklist is a graphical tree that shows you all of the views available in Tuner, as well as their association with other views. When you select a view, it appears in the pane to the right of the checklist; you can then customize the part of the Windows Installer package pertaining to that view. The Customization Steps Checklist is a subset of the entire checklist.

The checklist can be toggled from the View menu and from the Toolbar.

Customization Steps Checklist

To assist you with your customization, Tuner provides you with a set of steps that cover all parts of the MSI file that can be customized in your transform project. You do not have to follow these steps in sequential order, or even complete all the steps. Below is a brief description of each step, and how they fit into the customization workflow.

Table 13-7 • Customization Steps

Step	Description
Prevalidation View	This step allows you to pre-check the base Windows Installer package to ensure it is valid according to MSI standards before you take the time to create the transform file. If it is invalid, there may be unexpected results with your installation. Tuner allows you to copy the results of the prevalidation to the clipboard, where you can paste them into a message to the application's vendor. This is one of the most important steps in the Tuner workflow. Without it, all of your work might be wasted.

Table 13-7 • Customization Steps (cont.)

Step	Description
Organization View	This allows you to specify both the default destination that the installation will suggest, as well as the suggested default company name. You also can modify some properties of individual features in the product.
System Configuration View	This is where Tuner shows you its true customization power. You can add files and registry entries to the installation, allowing you to add company-specific files, templates, etc. to the installation. For example, if you have a set of word processing templates your company uses, you can include them you can modify or remove existing shortcuts, or add your own as necessary.
Application Configuration View	This step allows you to change functionality for Add/Remove Programs in Control Panel for Windows 2000 and later. You can also see the full spectrum of MSI properties as they exist in the base Windows Installer package, and add, modify, or remove them as necessary. Additionally, you can configure source resiliency through the Server Locations view, and customize user interface sequences through the Dialogs view.
Package Preparation View	This step allows you to postvalidate your project, verifying the original Windows Installer package and transform file, meets MSI-validation standards. You also can select the distribution options, including ones for network, FTP, single-executable, and SMS distributions.

Output Window

When you prevalidate the base Windows Installer package, postvalidate the package and your transform, or package the transform and base package, the Output Window appears at the bottom of the interface. It consists of three tabs:

- **Output**—Lists the Errors (), Warnings (), and Informational Messages () that are generated during prevalidation, postvalidation, and packaging.
- **Validation**—Upon completion of the pre- or postvalidation, this tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error (), a Warning (), or an Informational Message (), the name of the ICE that generated it, and a brief description of what caused it to occur.

  If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red. You can then use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.
- **Packaging**—Displays packaging/distribution information, and displays a summary of the files copied. You can copy these results to the clipboard by right-clicking and selecting Copy.

The Output Window can be toggled from the View menu.

Customize Dialog Box

The Customize Dialog box allows you to customize which toolbars are available in the Tuner user interface, as well as the buttons that are available on the toolbars. The dialog box consists of two tab panels:

Toolbars

From the Toolbars panel, you can select viewing properties for all toolbars, such as whether tooltips are displayed, the style of the toolbar, and the size of the buttons. You can also create your own custom toolbar, onto which you can place buttons found in the Command tab panel.

Command

The Command panel allows you to customize toolbars and the menu bar. Simply drag the command or menu you want to add to the existing toolbar; it appears where you place it. To remove a command or menu, select it and drag it off the toolbar.

Properties Dialog Box

This dialog box displays properties of the transform you are currently creating or editing—including the name and location of the base Windows Installer file, and any additional transforms that are associated with this transform and MSI file.

Options Dialog Box

The Options dialog box has two tabs: File Locations and View Settings.

File Locations

Within the File Locations tab, you can specify the default location of your source MSI files. This is reflected in the New Transform and Open Existing Transform Views of the Tuner Start Page.

View Settings

From the View Settings tab, you can select whether you want to display files from the base Windows Installer package (MSI) in the Files and Folders view in addition to files added in the transform.

Transform Summary Dialog Box

The Transform Summary dialog box, available from the Project menu, allows you to configure how to handle specific errors when the transform is applied. Additionally, you can configure how the Windows Installer Service verifies whether the transform can be applied against a given package.

Suppression Options

Options in this section allow you to configure whether installations with this transform applied will continue or fail if certain errors are encountered. You can configure the following options:

Table 13-8 • Suppression Options

Option	Description
Add Existing Row	Suppresses errors resulting from adding rows that already exist.
Delete Missing Row	Suppresses errors resulting from deleting rows that do not exist.
Add Existing Table	Suppresses errors resulting from adding existing tables.
Delete Missing Table	Suppresses errors resulting from deleting tables that do not exist.
Modify Missing Row	Suppresses errors resulting from updating rows that do not exist.
Change Code Page	Suppresses errors resulting from mismatched code pages.

Validation Options

Options in this section allow you to specify how the Windows Installer Service verifies the transform can be applied to a Windows Installer package. You can configure the following options:

Table 13-9 • Validation Options

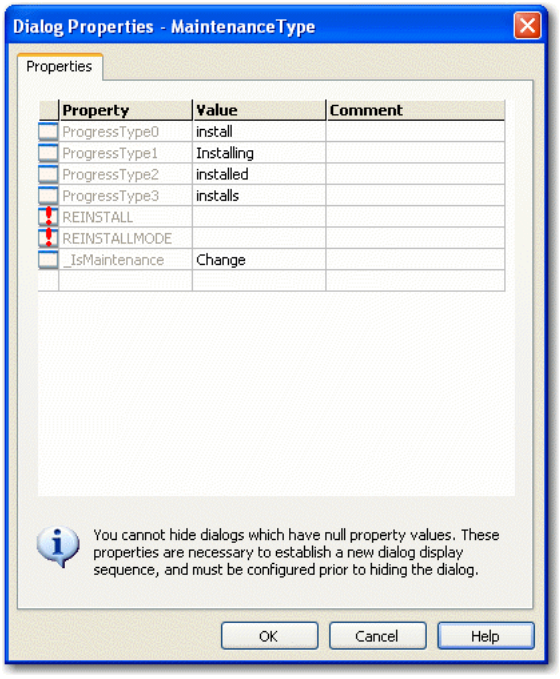
Option	Description
Same Language	If selected, the package against which the transform is applied must be the same language as the package used to create the transform.
Same Product Code	If selected, the product code for the package against which the transform is to be applied must be the same as the package product code of the package used to create the transform. If not selected, you can create a generic transform that can be applied against multiple Windows Installer packages.
Same Upgrade Code	If selected, the upgrade code of the package against which the transform is applied must be the same as the upgrade code of the package used to create the transform.
Product Version is Lower	If selected, the product version must be less than the version of the package used to create the transform. This can be combined with the Product Version is Equal option to create a “less than or equal to” comparison.
Product Version is Equal	If selected, the product version must be equal to the version of the package used to create the transform. This can also be combined with either the Product Version is Lower option or the Product Version is Higher option, creating a “less than or equal to” or “greater than or equal to” comparison.

Table 13-9 • Validation Options (cont.)

Option	Description
Product Version is Higher	If selected, the product version must be greater than the version of the package used to create the transform. This can be combined with the Product Version is Equal option to create a “greater than or equal to” comparison.
Version Checking	When using product version comparisons, you must indicate to what degree you want the comparison made. You can compare only the major versions, the major and minor versions, or the major, minor, and update versions. You can also select None to clear version checking.

Dialog Properties Dialog Box

You can use the Dialog Properties dialog box to view or change properties associated with UI sequence dialogs. The Dialog Properties dialog box is accessible by either selecting the appropriate dialog in the sequence and clicking Properties, or if you attempt to hide a dialog which has properties that must be configured.



Note • *You must ascertain the purpose of each property from the Windows Installer package, as these properties are usually custom in nature. This is especially true for properties that must be set prior to hiding a dialog from the UI sequence.*

Null Properties

You cannot hide UI sequence dialogs until you provide values for all currently null properties. These values are necessary to establish a new dialog box display sequence. Each property which must be configured prior to hiding the dialog in the sequence is marked with . Either configure the null properties, or click Cancel to return to the Dialogs view. Typically, configuring a null property involves clicking in the property's value field and selecting the value from the drop-down list.

Once you have provided a value for a null property, or if the property does not require configuration, it is denoted with .

Tuner Views

The following views are available in Tuner:

Table 13-10 • Tuner Views

Views	Subviews
Tuner Start Page	• Welcome to Tuner View • Create a New Transform View • Open a Recent Transform View • Open an Existing Transform View
Help View	
Package Validation View	• Prevalidation View
Organization View	• Product Properties View • Features View
System Configuration View	• Files and Folders View • Registry View • Shortcuts View • INI Files View • ODBC Resources View • NT Services View
Application Configuration View	• Server Locations View • Setup Properties View • Dialogs View • Add/Remove Programs View
Package Preparation View	• Postvalidation View • Package View

Table 13-10 • Tuner Views (cont.)

Views	Subviews
Additional Tools View	• Direct Editor

Tuner Start Page

From the Tuner Start Page, you can create a new transform project or open an existing one.

Select one of the following links for more information:

Table 13-11 • Tuner Start Page Subviews

View	Description
Welcome to Tuner View	General information about Tuner.
Create a New Transform View	Create a new customization, either starting with a blank transform, or by using a Response Transform based on selections from a custom installation.
Open a Recent Transform View	Select a transform previously created with Tuner.
Open an Existing Transform View	Select a transform created with a tool other than Tuner. You need to provide additional information, such as the base Windows Installer package for this transform.

Welcome to Tuner View

The Welcome view provides you with links to information on the Windows Installer service, Microsoft's integrated method of handling installations, as well as information about Tuner in the form of checklist steps. These steps cover the customization capabilities of Tuner.

Create a New Transform View

This view is displayed when you click **Create a new transform** or select **New** from the **File** menu.

This view contains the following options:

Base Windows Installer Package Subview

Table 13-12 • Base Windows Installer Package Subview

Field	Description
Select an MSI file	Enter the name and location of the Windows Installer package that you are customizing, or click Browse to locate it.
Provide a list of additional transforms to be applied	If there are transforms already associated with the Windows Installer package, (for example, previous customizations or transforms containing language-specific information), click the New button ( ). When an entry appears in the list, click the Browse button (...) to the right of it and locate the transform. If multiple transforms are associated with this package, use the Move Up ( ) and Move Down ( ) buttons to specify the order in which the transforms are applied. ***Caution*** • *When using multiple transforms, keep in mind that the order in which they are applied is critical. For example, if you create a transform for a Windows Installer package that creates a new value for a property, and then create a second transform that changes the value created in the first transform, everything works correctly. However, if you apply the second transform first, that transform is attempting to modify the property's value, instead of creating it. That will result in an error.* *One simple example of where this may be a problem is with the default company name. If the value is not set by default, and you set it in using the first transform, a new value for the property is created. If you create a second transform that modifies the combined original package and first transform, and the second transform changes the default company name, it is only changing the property. However, if you try to apply the second transform without the first one, Windows Installer interprets this as trying to change a null value to another value, which will result in an error.*

Windows Installer Transforms Subview

Table 13-13 • Windows Installer Transforms Subview

Field	Description
Provide a new project name and location (or accept the default) and click Create to create a new Customization project	Provide a new project name and location. By default, the transform will be created in the same directory as the Windows Installer package, and named the same as the base package with an .mst extension. If you want to change the name and/or location of the transform, click Browse to open the Save Customization File dialog. Navigate to the directory in which you want to store the transform file you are creating. Provide the name of the transform with an .mst extension (for example, MyTransform.mst) and click Save. The dialog box closes and the path and file name appear in the edit field.

Table 13-13 • Windows Installer Transforms Subview

Field	Description
Response Transform	If you want to create a Response Transform, select this check box. If creating a Response Transform, step through the installation, making changes as necessary. When you reach the end of the installation sequence and click Install, the installation will exit and the Tuner interface will open your transform, which contains all of the changes you made during the simulated installation.
Command line properties	If you are using a response transform, you can specify additional command-line properties (in property name/value pairs separated by semicolons) to pass to the response transform. These must be PUBLIC properties, and only control how the dialogs are displayed during creation of the response transform. They are not persisted outside of the UI sequence during creation. For example, you can pass the property/value pair ARPHELPTELEPHONE=1-111-111-1111 to set the value of the Help Telephone field of Add/Remove Programs. You might pass a property/value pair during response transform creation to display all dialogs during an installation that may not be displayed based on your system configuration (for example, to show Windows 9x-only dialogs on a Windows NT platform). You can then make appropriate responses and have them included in your transform.

Note • *You can access information about the original MSI file and associated transforms by selecting Properties from the File menu.*

Open a Recent Transform View

This view, a list containing your most recently accessed transforms, is displayed when you click on Open a recent transform. Select a transform and click Open to open it, or select Properties from the context menu to view information about it (including details about the base MSI package and associated transforms).

You can also select one of the options at the bottom of the view to determine the view that is opened when you start up Tuner:

- load the last accessed transform when opening Tuner,
- make this recent list the default Tuner Start Page screen, or
- make the Welcome screen the default Tuner Start Page screen

Open an Existing Transform View

This view is displayed when you click on Open an existing transform or select File | Open. On this view, you can specify the name and location of the base Windows Installer package, any associated transforms, and the name and location of the transform file.

Generally, you will only use this option when opening existing transforms that were created by a product other than Tuner, or created by someone other than yourself. Transforms you create using Tuner are more easily accessed through using the Open a recent transform selection.

Help View

The Help view provides you instant access to this online help library. You can also access Microsoft's comprehensive Windows Installer reference library.

Package Validation View

The first recommended step in creating a transform is to perform validation on the base Windows Installer package. This helps you identify potential problems that you may or may not want to correct using Tuner.

Package validation is performed in the Prevalidation View. To continue, click Prevalidation under Package Validation in the checklist.

Prevalidation View

The Prevalidation view provides you a way to ensure the base Windows Installer package for your transform is valid. If it fails the validation test, then unexpected (and unwanted) results can occur during installation.

To begin the prevalidation process, select the evaluation file that you want to use for package validation (or click Browse to locate it), and click Start. By default, the file is evaluated using the full logo-compliant validation file, and all internal consistency evaluators (ICEs) are checked. If you just want to test specific ICEs, after you select the evaluation file, specify the ICE names, and separate them by semicolons if there is more than one.

You can toggle the information level of the displayed results by checking the Show Info Messages, Show Error Messages, and Show Warning Messages check boxes. If any errors are present, the Windows Installer Package is invalid. Warning messages highlight potential problems, but will not cause validation to fail. Informational messages display ongoing information during the validation process.

Viewing the Prevalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Prevalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error ( ), a Warning ( ), or an Informational Message ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Direct Editor.

***Note** • If no errors appear in the results (providing you are displaying errors), then the package is valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

Organization View

The Organization view allows you to modify two main parts of the installation that your end users will see: the default path and default company name, and the actual features that can, will, or will not be installed.

Each subview of the Organization View is described below:

Table 13-14 • Organization View Subviews

Views	Description
Product Properties View	When a user runs a custom installation of a Windows Installer package, the Custom Setup dialog box provides a default installation path and a default organization name. The Product Properties view provides a mechanism for changing these defaults.
Features View	Features are the building blocks of the installation. They represent distinct pieces of functionality to end users, such as program files, help files, or clip art. You can modify which features and subfeatures are installed, and how they are installed, in the Features view.

Product Properties View

This view gives you a way to specify the default path on the user's computer into which the application will be installed. You can also specify the default organization name (i.e., your company's name).

The following properties are associated with this view:

Default Destination Variable

This is the name of the variable that stores the Default Destination Path. If you change this variable, you could create errors during postvalidation. Click on the variable's value to display a combo box that allows you to select a variable.

Generally, the variable used will be INSTALLDIR or INSTALLLOCATION (both author-created variables). However, another variable can appear as the value: TARGETDIR (a Windows Installer variable). If TARGETDIR is suggested, it is strongly recommended you contact the vendor who created the original MSI and ask what was used for the Default Destination Variable. While it is possible that it was TARGETDIR, it is also possible another variable was used and Tuner cannot identify this non-standard variable.

If the incorrect variable is set here, and/or if the Default Destination Path is changed, the installation may not function properly. If that happens, you can reset the information in this view by clicking on the Reset button when the Default Destination Variable is selected.

More information can be found in the Windows Installer Help.

Default Destination Path

This location, stored within the Default Destination Variable, is the path where the application will be installed on the target machine, unless overridden during installation from the Custom Setup dialog. Click on the path's value to display a combo box of possible paths, or edit the path in the value field. It can be a hard-coded path, or it can be a Windows Installer folder property . Further levels can be separated with a backslash—for example, `ProgramFilesFolder\MyApp\Bin`.

To comply with Windows logo requirements, the application must default to a subfolder of ProgramFilesFolder, which can vary depending on the system's locale and user settings. If ProgramFilesFolder\ProductName, is specified as the default value for the Destination Folder property, then this product's files will always be installed to the logo-compliant location.

Company Name

This is the name the installation suggests for your organization during setup. If it is not set, the installer will automatically set it during installation using values from the registry. Once the value has been entered for the name of the organization, the COMPANYNAME property can be seen in the Setup Properties View.

Features View

This view allows you to change Feature properties to best suit your situation.

Features are the building blocks of an installation from an end user's perspective. They represent a specific capability of the product, such as the help files or a part of a product suite that can be installed or uninstalled based on the end user's selections. Features can be composed of subfeatures, which in turn can be composed of further subfeatures. Depending on the visibility of the "parent" feature, end users can select which portions of a feature to install in the Custom Setup dialog.

Each feature and subfeature has properties that can be modified within Tuner. These include a description of each feature (as it appears in the Custom Setup dialog box), its visibility, and its initial state . Tuner allows you to change these feature properties to best suit your situation. For example, you may want to prevent a specific application within a suite from being installed in a particular transform file. By changing its initial state and visibility, you can prevent your end user from ever seeing this feature during installation.

The Features View contains the following options:

Table 13-15 • Features View Options

Options	Description
Description	This description will be displayed when this feature is clicked in the Custom Setup dialog box during installation.
Visible	Specifies how the feature will be presented to the end user during installation in the Custom Setup dialog. The following options are available: • **Visible and Collapsed**—This feature will be displayed in the Custom Setup dialog with its subfeatures collapsed by default. • **Visible and Expanded**—This feature will be displayed in the Custom Setup dialog with its subfeatures expanded by default. • **Not Visible**—This feature will not be displayed to the end user in the Custom Setup dialog. Although an end user obviously cannot select or deselect an invisible feature, this property does not affect whether a feature is installed. In other words, a feature is not automatically installed if it is invisible; it just cannot be deselected if it would otherwise be installed, or selected if it should not be installed.
Initial State	Provide the initial state for the feature. The end user can override this from the Custom Setup dialog. Your options are: • **The feature is not installed**—By default, the feature will not be installed during setup. • **The feature is installed on the local drive**—By default, the feature will be installed on the local drive during setup. • The feature is run from source, CD, or network—By default, the feature will be run from the source, whether it be from the installation CD or from the network. • **The feature is advertised**—By default, the feature will be advertised, but not installed. Essentially, this is an on-demand option; a shortcut will be created during setup, and if the shortcut is clicked, the feature will then be installed from the source. This ensures features that may be unnecessary are not installed until they are needed, if ever.

System Configuration View

The System Configuration view provides you with the ability to add additional files to a Windows Installer installation package, as well as add, remove, or modify shortcuts and registry information. Ultimately, this allows you to customize the installation to your needs, such as including company-specific templates in the correct folder during installation.

Files and Folders View

The Files and Folders View consists of four panes, representing the Source and Destination views.

Source Computer View

The Source view, located at the top, displays the folder and file structure on the user's computer.

Destination Computer View

The Destination computer's folders pane represents the folders on the target machine for the installation. The folders initially displayed for the target machine are ones used commonly in installations.

The Destination computer's files pane displays the files that are part of the installation. Initially, only files contained in the base Windows Installer package are displayed. When you add files to the package (into the transform), these also appear in this pane.

Note • *Tuner cannot display files contained within compressed files.*

A key file is a file that the Windows Installer uses to detect a component's presence. If the key file is in its proper location, the installer assumes that the entire component is installed correctly. Each component can have a key file, represented in the Files and Folders View by a key icon (). The key files were set by the setup author, and cannot be modified using Tuner.

Destination Computer View Tasks

The following tasks are performed in the Destination Computer View:

- **Defining a New Folder**—To define a new folder, select either the Destination Computer or a predefined folder from the Destination Computer's Folders pane and select Add from the context menu.
- **Adding Files to an Installation**—To add files to an installation, simply locate them within the Source view, and either drag and drop them to the appropriate destination folder, or use the copy and paste commands.
- **Removing Folders or Files**—To remove folders and files you have already added to the installation, select the file or folder from the Destination view and select Delete from the context menu. Predefined folders are required for installation and cannot be removed. If you add files to an installation, they are always installed.
- **Removing Files from the Base Windows Installer Package**—Files from the base Windows Installer package can be removed during installation, except for key files. To remove a non-key file from the MSI, right-click on the file and select Remove from the context menu. The file is marked with an icon indicating it is not to be installed during installation.

Registry View

Similar to the Files and Folders View, the Registry view consists of four panes representing the Source and Destination views.

Source Computer View

The Source view, located at the top, displays the registry entries on the administrator's computer.

Destination Computer View

The Destination view represents the registry entries on the target machine for the installation. By default, the Destination view contains the following read-only registry hives: HKEY_CLASSES_ROOT, HKEY_CURRENT_USER, HKEY_LOCAL_MACHINE, HKEY_USERS, and HKEY_USER_SELECTABLE.

You can use the Registry view to:

- Create registry keys and values. You can do this using the same procedure you use to create registry keys and values in the Windows Registry Editor, or you can copy or drag and drop existing keys and values from the Source view to the Destination view.
- Import an existing REG file using the Import REG File Wizard.
- Modify or delete registry keys that are part of the base installation. If you add new registry keys, they will always be installed.

Setting Registry Key Options

You must be cautious when creating new registry keys so adverse results do not occur. You must specify whether the key is to be created at install, removed at uninstall, or both. These mutually exclusive options are available by right-clicking on a key.

Table 13-16 • Registry Key Options

Option	Description
Create Key at Install	Creates the new registry key during installation if the key does not exist on the target machine.
Delete Key at Uninstall	Deletes the registry key during uninstallation, regardless of whether the key existed prior to the MSI's installation. This means that the key, and all its contents and sub-keys, will be removed regardless of whether other software information that is unrelated to this MSI exists. This can have a severe impact on other programs; only select this option if you are sure that the only software affected is the base MSI.
Both Create and Delete	Both of the above scenarios will occur.

Shortcuts View

The Shortcuts view offers an integrated, visual method for adding shortcuts and program folders to the installation. Existing shortcuts can also be modified or removed. For information on how to use the Shortcuts view to create, edit, or remove shortcuts, refer to one of the following topics:

- **Shortcut Properties**—Explains the Properties that you set when creating a shortcut.

- **Shortcut Locations**—Explains how to specify the location of a shortcut, either in a predefined folder or a folder that you create.
- **Shortcut Targets**—Explains how to specify the path to the shortcut's target application, batch file, folder, or feature.

Note • *Shortcuts created in the transform are denoted by* *and shortcuts from the base Windows Installer package are denoted by* .

Shortcuts View/Shortcut Properties

When creating or editing a shortcut, you specify properties on the Shortcut Properties View.

The Shortcut Properties are described in the following table:

Table 13-17 • Shortcut Properties

Option	Description
Icon	Displays the name of the icon for this shortcut. • **To change the icon**, click the Change Icon button. On the Change Icon dialog box, click Browse to select an EXE file or DLL. All of the icons contained in that EXE or DLL file are displayed. Select the appropriate icon and click OK. On the Shortcut Properties view, the name of the icon now appears in the icon value cell and a picture of the icon appears next to the Change Icon button. • **To remove the icon**, click Clear Icon.
Description	A brief description of the shortcut. The text in this field will appear when users hover the mouse pointer over the shortcut in Windows 2000 or later.
Arguments	Any command-line arguments for the shortcut.
Target Type	The destination folder, or a file from the MSI or transform. This property can be set to one of the following values: • **File from MSI Package**—Used if the shortcut is to a file that is part of the base Windows Installer package. • **File from System**—Used if the shortcut is to a file that already exists on the target system. It may also be from the transform file. • **Destination Folder**—Used if the shortcut points to a folder rather than a file. The folder can be on the target system, from the MSI package, or from the transform. • **Advertised shortcut**—Used if you want this to be a shortcut to an "advertised" feature. Advertised features are not installed immediately during the setup process. Instead, they are installed when requested. The shortcut makes it appear that the feature is already installed, although it is not installed until the end user requests it.

Table 13-17 • Shortcut Properties (cont.)

Option	Description
Target	Path and file name for this shortcut's target. There are many potential types of targets, including applications, files, folders, printers, and computers on a network. Instead of hard-coding a path, you can use a Windows Installer folder property in square brackets—for example, *[INSTALLDIR]\MyApp.exe*. You can also target Windows Installer features, which you can use for feature advertisement. To specify a feature as a shortcut target, enter the name of the feature in the Target field. You can determine the name of the feature by going to the MSI Table Editor and selecting the Feature table. The list of features you can target with your shortcut is listed in the Features column of the table.
Run	Specifies how the item is displayed when the shortcut is double-clicked. You can select from the following options: • **Normal Window**—Launches the program in a normal sized window. • **Maximized Window**—Launches the program in full-screen view. • **Minimized Window**—Launches the program in a minimized window, visible only on the taskbar.
Working Directory	Default directory for the Save As and Open dialogs. If you are modifying an existing shortcut, or creating a new one, you can select a Windows Installer folder property from the list instead of hard-coding a path. Separate further levels with a backslash—for example, `<ProgramFilesFolders>\MyApp\Bin`.
Hot Key	The decimal value of the hot key combination for this shortcut. The Hot Key feature allows end users to launch a shortcut by pressing a combination of keys, rather than using the mouse. When you click in the Hot Key field to create or modify a shortcut, the Hot Key dialog box opens. While the Hot Key dialog box is open, press the desired hot key combination; those keystrokes are recorded. When you click OK on the Hot Key dialog box, Tuner automatically converts the keystrokes into a decimal value and enters that value in the Hot Key field. ***Caution*** • *Microsoft recommends that you do not set this value, as it may conflict with existing hot key combinations on the target machine.*

Shortcuts View/Shortcut Target

There are four shortcut Target Types you can add to a transform:

Table 13-18 • Shortcut Target Types

Target Type	Description
File from MSI Package	Used if the shortcut's target is a file that is part of the base Windows Installer package.
File from File System	Used if the shortcut's target is a file already existing on the target system.
Advertised Shortcut	Used if you want this to be a shortcut to an "advertised" feature. Advertised features are not installed immediately during the setup process. Instead, they are installed when requested. The shortcut makes it appear that the feature is already installed, although it is not installed until the end user requests it.
Destination Folder	Used if the shortcut points to a folder rather than a file.

The Target Type that is selected affects what you should enter in the Target property field:

File from MSI Package & File from File System

Under most circumstances, shortcut targets are applications or batch files. You simply provide the full path to the application or batch file in the Target property. However, after entering the target, if you leave the Shortcuts view and then return to it, you find that the target has changed. For example, you might have entered `C:\Temp\MyFolder\mytarget.exe` as the target, but it now reads `[MyFolder]mytarget.exe`. What has happened is that the path has been replaced based on entries made to the Directory table. For more information, see Determining the Path of Changed Shortcuts.

By stringing together the directories you just located in the Directory table, you can determine the path represented by `[MyFolder]` in the shortcut target. If you use the drop-down list in the Target property, you can determine the absolute values of these other directories in the same fashion.

Destination Folder

To point the shortcut to a folder rather than a file, select Destination Folder in the Target Type property, and then select a folder name from the Target property drop-down list. The Target list includes available folders on the target system, from the MSI package, and from the transform.

Advertised Shortcut

You can also target Windows Installer features, which you can use for feature advertisement. To specify a feature as a shortcut target, simply enter the feature name in the Target field. You can determine the name of the feature by going to the Direct Editor and selecting the Feature table. The list of features that you can target is listed in the Features column of the table.

Shortcuts View/Shortcut Locations

When you first navigate to the Shortcuts view, you see a set of predefined folders with existing shortcuts (if the base MSI package had shortcuts defined). You can modify or remove these shortcuts according to your needs.

If you need to create your own shortcut, you can place it in a new folder you define or in a predefined folder. Additional predefined folders that are not displayed can be accessed by right-clicking on the uppermost item in the Shortcuts explorer and selecting Show Folder.

Under the Show Folder submenu is a list of the additional predefined folders supported in the Shortcuts view. Select the folder where you want your shortcut created to have it displayed in the Shortcuts explorer. The predefined shortcut destinations are described below.

Alternately, you can create your own folder in which to place shortcuts by right-clicking either a folder or the top level Shortcuts item and selecting New Folder. To remove a folder that you have added, right-click on it and select Delete.

The following predefined folders are available for shortcuts:

Table 13-19 • Predefined Shortcut Folders

Predefined Folder Name	Description
AppDataFolder	The current user's Application Data folder.
CommonFilesFolder	The Common Files folder for the current user.
DesktopFolder	The user's desktop. Although placing a shortcut on the desktop makes it easily visible, it can also be distracting to users, so it should be used sparingly.
FavoritesFolder	he Favorites folder for the current user.
FontsFolder	References the target machine's Fonts folder.
INSTALLDIR	The installation's default destination folder.
ProgramFilesFolder	References the target machine's Program Files folder.
ProgramMenuFolder	The Program menu for the current user.
SendToFolder	The user's Send To folder, which is accessible when you right-click on files. Shortcuts are placed here so users can have quick access to the target program from many file types.
StartMenuFolder	The Start menu folder for the current user.
StartupFolder	The current user's Startup folder. Shortcuts placed here automatically launch their targets whenever Windows is started.
SystemFolder	The target machine's System folder.

Table 13-19 • Predefined Shortcut Folders (cont.)

Predefined Folder Name	Description
TempFolder	References the target machine's Temp folder (usually C:\Temp).
TemplateFolder	The current user's Template folder.
WindowsFolder	The target machine's Windows folder.

INI Files View

Initialization (INI) files serve as a database in which you can store and retrieve information between uses of your application. Typically, INI Files contain key name-value pairs representing run-time options for applications. Some INI files, such as Boot.ini and Wininit.ini, are used by the operating system.

The INI Files view provides a graphical way for users to add, modify, or delete the contents of the IniFile Table.

- The INI Files view displays the contents of the IniFile table from the source Windows Installer package and the transform .
- The view itself consists of three panes: the leftmost a tree of predefined folders from the Windows Installer package and user-defined folders from the transform.
- The top-right section displays the keys and values in the selected IniFile section. Windows Installer and transform values are distinguished by different icons in this pane.
- The lower-right pane provides information about the selected key.

Editing INI Files in Tuner

To edit an INI File in Tuner, simply expand the appropriate IniFile node in the tree. Then select the appropriate section, which appears in the upper right pane. You can then edit the keys and values appropriately. You can also insert new keys and values by right-clicking in the key and value pane and selecting Add. If you want to add a new section to an INI File, right-click on the INI File in the tree and select New Section. You can also delete an INI File, a section of an INI File, or a key by right-clicking the appropriate node or property sheet entry and selecting Remove.

For detailed instructions on performing these tasks, click on one of the following topics:

- Adding INI Files
- Adding New Keys to INI File Sections
- Adding Sections to INI Files
- Importing Existing INI Files
- Modifying INI File Keys, Values, and Actions
- Removing INI Files
- Removing INI File Section Keys

- Removing Sections from INI Files
- System Configuration View

ODBC Resources View

Open Database Connectivity (ODBC) Resources are ones that involve interaction with databases. Tuner allows you to view existing ODBC Data Sources, ODBC Drivers, and ODBC Translators.

The left pane of the ODBC Resources view contains a tree with the three root nodes: ODBC Data Sources, ODBC Drivers, and ODBC Translators. When any of these are expanded, individual Data Sources, Drivers, and Translators contained in the Windows Installer package are displayed. When selected, each of these individual nodes displays information in a property grid displayed in the upper right pane.

There are three different types of ODBC Resources available for viewing and/or modification through the ODBC Resources view:

Table 13-20 • ODBC Resource Types

Resource Type	Description
ODBC Data Sources	The source of the data (database type) and information on how to connect to that database. Common database types include Microsoft SQL Server, Microsoft Access, and Visual FoxPro. Connection information may include the name of the database, where the server that hosts it is located, and logon/password information. You can add new ODBC Data Sources from the ones existing on your computer, or delete ones you add or existing ones from the MSI. You can also add, edit, and delete ODBC Data Source attributes. If your machine does not have the ODBC Data Source that is needed by the package, you can type it into the ODBC Data Source dialog. See ODBC Resources and Adding New Data Sources for more information.
ODBC Drivers	These are libraries that implement functions involving ODBC. Each database type has its own ODBC driver. You can add only those Data Sources for which ODBC Drivers exist in the MSI package. You can add, edit, or delete new attributes for ODBC Drivers, and you can edit or delete all attributes except for File, Setup File, and Feature. See Adding New ODBC Driver Attributes and Editing ODBC Driver Attributes for more information.
ODBC Translators	These translate one form of raw data into another form that can be used with a specific database type. For example, an ODBC translator may convert from one code package to another. You can only view the contents of an ODBC Translator and cannot add, delete or modify them.

Note • *Only ODBC Data Source attributes are editable; ODBC Drivers and ODBC Translators are provided in read-only form.*

NT Services View

The NT Services view provides a way to change parameters for NT Services included in the base Windows Installer package.

Note • *NT services cannot be added to a setup using Tuner. You can only make modifications to services in the base Windows Installer package.*

The following options can be modified:

Table 13-21 • NT Services View Options

Option	Description
Name	This property contains the name of the service to install. This property may have the same value as the Display Name, but is used by the installer in a different way.
Display Name	The name of the service as it appears in user interfaces (such as the name used under the NT Services control panel). This string can be a maximum of 256 characters in length. It may be the same as the Name property.
Service Type	There are two service types available: • Service that runs in its own process • Service that shares a process with others
Interact with Desktop	Although uncommon, some services need to interact with the desktop to display message or dialog boxes for the user. If this service requires this functionality, this property's value is set to Yes.

Table 13-21 • NT Services View Options (cont.)

Option	Description
Start Type	The value in this property dictates when the service is started. The possible values are: • **Automatic**—The service starts during system startup. • **Manual**—The service starts when the service control manager calls the StartService function. • **Disabled**—The service cannot be started. There are two additional values, available only for driver services: • **Start at boot time**—The device driver is started by the operating system loader. • **Started by the system**—The device driver is started by calling the IoInitSystem function.
Error Control	This property specifies what action is taken by the startup program should the service fail to start properly during startup. The available values are: • **Ignore Error**—Logs the error and continues startup. • **Normal Error**—Logs the error, displays a message box informing the user of the problem, and continues startup. • **Critical Error**—Logs the error, if possible, and restarts the system with the last-known-good configuration. If the last-known-good configuration caused the failure, the startup operation fails.
Overall Install	This property's value specifies how the installation handles a situation when this service cannot be installed for some reason. There are two possible resolutions: • Continue overall install if service fails to install • Fail overall install if service fails to install
Load Order Group	The value of this property is a string that names the load ordering group of which this service is a member. If the service does not belong to a load order group, this value should be either an empty string or NULL.
Dependencies	A list of names of services or load ordering groups that the system must start prior to starting this service.
Start Name	The name under which the service is logged on. Leaving this field blank causes the service to be installed for the LocalSystem account. See the Windows Installer help topic ServiceInstall Table for information on the format for the StartName value.
Password	The password associated with the start name. Most services will have a blank value for this property.

Table 13-21 • NT Services View Options (cont.)

Option	Description
Arguments	This property contains any command-line arguments or properties required to run the service.
Description	This property contains a localizable description of the service. It is typically set by the setup author.
Feature	This read-only property contains the name of the feature with which this service is associated.

Application Configuration View

Application Configuration in Tuner involves adding or modifying properties that affect your setup as well as specifying properties for Add/Remove Programs in Control Panel for Windows 2000 and XP. You can also configure source resiliency using the Server Locations view, and customize user interface sequences from the Dialogs view.

The Windows 2000 and Add/Remove Programs in Control Panel differs from the previous Windows operating systems in many ways. Depending on how the Windows Installer setup is configured, the user has the option of removing, repairing, or changing the installation with the click of a button. Windows 2000 users are also be able to access additional information in Add/Remove Programs not available on previous platforms. With this information, it is easier for your end users to find technical support links, phone numbers, product update information, and information about your company.

Add/Remove Programs functionality can also be disabled to limit the number of end users who have access this feature.

Server Locations View

If you install from a network server, and if you install features to run from the server or that will be advertised for installation on their first use, the applications may need access to the server sometime after the initial installation. The applications may also require access to the server if a file is deleted or becomes corrupt, as the application can copy the problematic file(s) automatically from the server.

To ensure that users always have access to an available network server for these circumstances, you can copy the administrative installation to one or more additional servers, and then specify those servers from within this view. If the primary server should become unavailable, the Windows Installer will attempt to connect to the other servers specified here, in the order they are specified. If no server is found, the Windows Installer will prompt the user to specify the location of the server.

Setup Properties View

Even though Tuner provides you views to customize many areas of the Windows Installer package, it may be necessary to edit property values that are not available elsewhere. The Setup Properties view exposes the entries in the properties table (the underlying structure of Windows Installer packages). You can also add your own custom properties here.

Properties exist in two formats: Private and Public. Private properties are set by the software vendor or by the Windows Installer during installation and cannot be altered. Private properties are always lowercase, and appear in Tuner in grayed out text. Public properties, which are always in capital letters, can also be set by the software vendor, but can be edited. Tuner also allows the addition of Public properties to the transform. The Public properties that you create can be edited or removed as necessary, whereas preexisting Public properties can only be edited.

***Caution** • Before you begin changing properties in the Setup Properties view, ensure you know exactly what you are doing. The changes you make may cause validation errors, installation errors, or other unforeseen problems.*

Dialogs View

When customizing the Windows Installer package, you may want to disable particular panels that appear during the installation, administrative, patch, or maintenance sequences. You can do so from the Dialogs view.

This view contains a list of each of the four installation modes (installation, administrative, maintenance, and patch), with the associated dialogs that appear as part of the UI sequence during the selected mode. You can enable or disable these dialogs by either the check box to the left of the dialog name, or by using the Show and Hide buttons.

If you hide a dialog that appears in more than one sequence, the dialog is hidden during all sequences.

***Note** • During each installation mode, Windows Installer displays a Wizard containing a sequence of panels. However, the underlying Windows Installer technology actually uses a series of dialogs displayed in sequence. During runtime, they are referred to as panels (as with other Wizards); at design time, they are individual dialogs that can be enabled or disabled as necessary.*

Add/Remove Programs View

The Windows 2000 and Windows XP Add/Remove Programs in Control Panel differs from the previous Windows operating systems in many ways. Depending on how the Windows Installer setup is configured, the user has the option of removing, repairing, or changing the installation with the click of a button.

Windows 2000 and XP users are also able to access additional information in Add/Remove Programs in Control Panel not available on previous platforms. With this information, it is easier for your end users to find technical support links and phone numbers, product update information, and information about your company.

You can set the following options from the Add/Remove Programs view:

Table 13-22 • Add/Remove Programs View Options

Option	Description
Publisher URL	Contains a URL for the publisher's home page. Corresponds to the ARPURLINFOABOUT property in the Setup Properties view.
Product Info and Update URL	Contains a URL that links to update information for the application. Corresponds to the ARPURLUPDATEINFO property in the Setup Properties view.
Help URL	Contains the Internet address for technical support. Product maintenance applets display this value. Corresponds to the ARPHELPLINK property in the Setup Properties view.
Help Telephone	Contains the telephone number that users can call for assistance with the product. Corresponds to the ARPHELPTELEPHONE property in the Setup Properties view.
Contact Person	Contains a the name of the person to contact for help or information about the product. Corresponds to the ARPCONTACT property in the Setup Properties view.
Comments	Contains additional information that is provided for the user. Corresponds to the ARPCOMMENTS property in the Setup Properties view.
Disable Modify Button	Provides a way to prevent users from running the application setup to modify the product's installation. Corresponds to the ARPNOMODIFY property in the Setup Properties view.
Disable Remove Button	Provides a way to prevent users from running the application setup to remove (uninstall) the product from the user's computer. Corresponds to the ARPNOREMOVE property in the Setup Properties view.
Disable Repair Button	Provides a way to prevent users from running the application setup to repair missing or corrupt product files. Corresponds to the ARPNOREPAIR property in the Setup Properties view.

Package Preparation View

The final step in creating a customization involves two parts. First, you should postvalidate your transform and base Windows Installer package. This ensures that you have not introduced any errors into the installation, and may help you verify that you have corrected errors that existed in the base package. Secondly, you need to actually package the transform and base package for distribution.

These steps are carried out in the following Package Preparation View subviews:

Table 13-23 • Package Preparation View Subviews

Views	Description
Postvalidation View	Allows you to ensure that your Windows Installer package is valid. The difference is that it also checks your newly created transform to make sure it is valid in relation to the base package.
Package View	When you have finished with your transform and postvalidation, the last step is to prepare the overall package so you can distribute it to your users (using a third-party tool such as Microsoft SMS).

Postvalidation View

Much like the Prevalidation view, the Postvalidation View allows you to ensure that your Windows Installer package is valid. The difference is that it also checks your newly created transform to ensure it is valid in relation to the base package. You can run the same internal consistency evaluators in the evaluation file, and receive the report back on the overall package and transform validity. By default, all ICEs are checked for the specified evaluation file.

You can select the information level of the displayed results by checking the Show Info Messages, Show Error Messages, and Show Warning Messages check boxes. If any errors are present, the Windows Installer package is invalid. Warning messages highlight potential problems, but will not cause validation to fail. Informational messages display ongoing information during the validation process.

If you started off with a valid Windows Installer package, yet postvalidation fails, it is likely your problems relate to changes you made in the transform. Make sure you look at the Evaluation Files and Internal Consistency Evaluators topic to see what each ICE message means. You can also consult the online MSI Help reference, available from the Help Menu, for more details.

It is also possible to start off with an invalid base package, but the postvalidation does not have any errors. If this happens, the properties you changed in your transform can bring the overall package and transform up to a valid package.

Output of the postvalidation appears in the Output and Validation tabs of the Output Window.

Viewing the Postvalidation Results

As each ICE is run, Errors, Warnings, and Info messages are generated, and are listed in the Output tab at the bottom of the interface.

Upon completion of the Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error (), a Warning (), or an Informational Message (), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform. For more information, see Direct Editor.

***Note** • If no errors appear in the results (providing you are displaying errors), then the package and transform are valid against the specific ICEs you specified, or against the entire evaluation file (if no ICEs were selected).*

***Tip** • It is possible for a package that passed the prevalidation to fail the postvalidation. Remember changes made in the Setup Properties can affect your installation. If your package fails postvalidation, check all changes made in the Setup Properties for accuracy. To identify the original Setup Properties, you can create a new transform file that can be deleted at any time. Changes made using the Direct Editor can also affect your installation's functionality.*

Package View

When you have finished with your transform and postvalidation, the last step is to prepare the overall package so you can distribute it to your users (using a third-party tool such as Microsoft SMS). Tuner provides several different packaging options that can be used individually or in conjunction with one another. Note that Tuner does not actually perform the distribution; rather, it gathers the necessary files together, and copies them to a location you specify. You can then use some of the standard software distribution tools to roll out your customized package.

The three subviews contained within the Package view are:

Table 13-24 • Package View Subviews

View	Description
Package View/Location View	Place the transform and base Windows Installer file on either an FTP server or network location.
Package View/Setup View	Package the transform and base Windows Installer file with an executable launcher (Setup.exe) to begin the installation. You have the option of including the MSI Engine for the appropriate platform to ensure Windows Installer functionality. Setup.exe uses information contained in setup.ini to determine the package, associated transform, and any command-line parameters.
Package View/SMS View	Package and prepare the transform and Windows Installer file for distribution using Microsoft SMS. Tuner can create a PDF file and/or an SMS file for your package and transform.

***Note** • These three subviews are also available through the Packaging Wizard, which can be accessed from the Project menu by selecting Package.*

Package View/Location View

It is from the Location View that you specify information regarding where to place the transform and initial Windows Installer file. You have the option to copy files to a network location, including performing an administrative installation, or copy files to an FTP server.

Package View/Setup View

In the Setup view, you can set options that are used to create a Setup.exe file to launch both your transform and Windows Installer package. You have the option to include the MSI engine for Windows 95/98 and NT, as well as include any command-line arguments for the installation.

Package View/SMS View

In many cases, you may want to perform distribution of the transform and Windows Installer package using Microsoft SMS. Tuner provides a way to create both a Package Definition File (PDF) or an SMS file in the SMS View.

Select the file type(s) you want to create. If you create an SMS file, Tuner can instruct SMS to create a Management Information Format (MIF) file when SMS deploys the package and transform. If you want to do this, provide the Install MIF Filename, Uninstall MIF Filename, and serial number.

For more information about Microsoft SMS, consult the SMS documentation.

Additional Tools View

Tuner includes an extremely flexible additional tool: the Direct Editor. Using this tool, you can directly edit the Windows Installer tables that make up the Windows Installer package. This provides you with extremely granular control over the transform you are creating.

Direct Editor

Windows Installer packages are relational databases consisting of dozens of interrelated tables. These tables reflect the application's features, components, relationship between features and components, registry information, and user interface.

The Direct Editor allows you to edit values in the MSI tables of the base Windows Installer package and store them in your transform. As you change values elsewhere in your transform, those changes are reflected in the Direct Editor, and vice versa. The complete list of MSI tables contained in the installation package is displayed in the left pane. When you select a table, the contents are displayed in the right pane.

Working Directly with MSI Tables

The Tuner Direct Editor provides the ability to work directly with MSI tables. This includes the ability to edit the contents, as well as find and replace values.

Tip • *When viewing or editing specific tables, pressing F1 launches the Microsoft Windows Installer help system to the appropriate table, if it is a standard Windows Installer table. When F1 is pressed while viewing a non-standard table, the Windows Installer help system launches to its default topic. Consult the software vendor for information about custom tables.*

Table Functionality

The following functionality is available for tables:

Table 13-25 • Direct Editor Table Functionality

Function	Keyboard Shortcut	Description
Add Records	Insert	Adds a new record to the table.
Delete Records	Del	Deletes the selected record after user confirmation. Referential integrity is not maintained.
Cut Row(s)	Ctrl+X	Enables users to cut single or multiple rows or cells in the grid to the clipboard.
Copy Row(s)	Ctrl+C	Copies the selected cell or row in the grid to the clipboard.
Paste Row(s)	Ctrl+V	Pastes the contents of the clipboard into a given cell or row(s).

Editing Tables by Launching the Direct Editor from the Validation Tab

Upon completion of a Pre- or Postvalidation, the Validation tab is automatically selected, and all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error (), a Warning (), or an Informational Message (), the name of the ICE that generated it, and a brief description of what caused it to occur.

If a row is grayed out, it indicates that the table cannot be edited in the Direct Editor (perhaps because it is in an external package). If a row is active, you can double-click on it to open that row's associated table. The Direct Editor is launched and the table and/or table cells that are causing the problem are highlighted in red.

This feature makes it very easy for you to use the Direct Editor to edit values in the MSI tables of the base Windows Installer package and store them in your transform.

Resizing Table Columns in the Direct Editor

When you initially open the Direct Editor, the selected table's columns are listed in a compact format so that the maximum number of columns are displayed.

To automatically resize a column so that its width matches that of its longest entry, double-click on the column heading. This new column width setting is automatically saved and will be implemented the next time you view this table column in the Direct Editor.

Sorting Table Columns in the Direct Editor

To sort a table column, click the column heading once. The order will toggle between ascending and descending.

Import INI File Wizard

Tuner allows you to import any existing INI files that you may have created previously. To import a INI file, you need to launch the Import INI File Wizard.

The Wizard consists of the following panels:

- Welcome Panel
- Import INI File Panel
- Import Conflict Options Panel
- Finishing INI File Import Panel

Within the INI Files View, right-click on a folder under the Destination Computer node (or right-click on the Destination Computer node to add a folder first), and then select Import INI File. The Wizard that appears prompts you for the location of the INI file, as well as what to do when there are conflicts arising from duplicate values. When import occurs, Tuner merges the contents of the INI file with existing INI file data.

Welcome Panel

The Import INI File Wizard allows you to import data contained in an INI file into your transform.

Import INI File Panel

From this panel, you need to specify the name of the INI file (.ini) you want to import into your transform. Alternatively, click Browse and navigate to it.

Import Conflict Options Panel

The Import Conflict Options panel allows you to specify how you want to handle duplicate INI file data.

Select one of the following options for the Wizard to use to determine how to handle these conflicts:

Table 13-26 • Import Conflict Options

Option	Description
Overwrite the data in the IniFile table	If conflicts exist, the Wizard will overwrite the INI file keys and values with any duplicate keys from the registry file (.reg).
Do not overwrite the data in the IniFile table	If duplicate keys and values are found, the Wizard will retain the existing INI file data and not overwrite it.

Click Import to import the .ini file. When the file has been imported, the Finishing INI File Import Panel is displayed.

Finishing INI File Import Panel

This panel appears following import of the .ini file. Click Finish to exit the Wizard and return to the INI Files View.

Import REG File Wizard

Tuner allows you to import any existing REG files that you may have created previously. To import a REG file, you need to launch the Import REG File Wizard.

The Wizard consists of the following panels:

- Welcome Panel
- Import Registry File Panel
- Import Conflict Options Panel
- Finishing Registry Import Panel

To launch the Import REG File Wizard, go to the Registry View, right-click on one of the registry hives or on a registry key you have added, and select Import REG File from the context menu.

The Wizard that appears prompting you for the location of the registry file, as well as what to do when there are conflicts arising from duplicate keys. When import occurs, Tuner merges the contents of the REG file with existing registry data.

Welcome Panel

The Import REG File Wizard allows you to add registry data contained in a registry file (.reg) into your transform.

Import Registry File Panel

From this panel, you need to specify the name of the registry file (.reg) you want to import into your transform. Alternately, click Browse and navigate to it.

Import Conflict Options Panel

The Import Conflict Options panel allows you to specify how you want to handle duplicate registry keys and values.

Select one of the following options for the Wizard to use to determine how to handle these conflicts:

Table 13-27 • Import Conflict Options

Option	Description
Overwrite the registry data	If conflicts exist, the Wizard will overwrite the registry keys and values with any duplicate keys from the registry file (.reg).
Do not overwrite the registry data	If duplicate keys and values are found, the Wizard will retain the existing registry data and not overwrite it.

Click Import to import the .reg file. When the file has been imported, the Finishing Registry Import Panel opens.

Finishing Registry Import Panel

This panel appears following import of the .reg file. Click Finish to exit the Wizard and return to the Registry View.

Packaging Wizard

The Packaging Wizard provides a way to step through the packaging process for the transform and Windows Installer package. Please note that this packaging merely places the installation on a network location or FTP server, creates a Setup.exe file, and/or creates files for SMS distribution. It does not actually distribute the installation to client machines. To invoke the Packaging Wizard, select Packaging Wizard from the Project menu, or select the Packaging Wizard button ( ) from the toolbar.

The Packaging Wizard consists of the following four panels:

- Location Panel
- Setup.exe Panel
- SMS Panel
- Packaging Summary Panel

Location Panel

The first panel of the Packaging Wizard allows you to specify the location to store the installation files (including transforms). If you select Network Location, you can specify or browse to the directory location.

Alternately, you can copy the installation files to an FTP server. If you select this option, you must specify the FTP location (URL), the user name under which to log in, and the password.

Setup.exe Panel

This Packaging Wizard panel allows you to include a setup.exe launcher for your package and transform, and include the appropriate MSI engine for Windows 9x or NT. You can also specify command-line arguments for the Windows Installer.

SMS Panel

In many cases, you may want to perform distribution of the transform and Windows Installer package using Microsoft SMS. Tuner provides a way to create both a Package Definition File (PDF) or an SMS file in the SMS View. Select the file type(s) you want to create. If you create an SMS file, Tuner can instruct SMS to create a Management Information Format (MIF) file when SMS deploys the package and transform. If you want to do this, provide the Install MIF Filename, Uninstall MIF Filename, and serial number.

Packaging Summary Panel

The Package Summary panel informs you of the packaging options selected in the three previous panels. If you need to make changes, use the Back button to return to the previous panel. The Cancel button aborts the packaging operation. If you are satisfied with the selected options, click Finish to copy the installation files to the specified location and/or create Setup.exe and SMS files.

Part 4

Testing Windows Installer Packages

This part of the AdminStudio 10.0 SP2 User Guide includes the following chapters:

- Identifying and Resolving Application Conflicts Using ConflictSolver
- Testing and Fixing Application Compatibility Using Compatibility Solver
- Identifying and Resolving Package Errors Using PackageExpert
- Analyzing the Impact of Installing Microsoft Operating System Patches
- Isolating Applications Using Application Isolation Wizard
- Ensuring Package Quality Using QualityMonitor

14

Identifying and Resolving Application Conflicts Using ConflictSolver

Edition • *ConflictSolver is included with AdminStudio Professional and Enterprise Editions.*

Although a Windows Installer package, merge module, or App-V package may be built to guidelines put forth by Microsoft, it is possible that the interaction between packages, or between a package and the base operating system, may cause unwanted results in your production environment. ConflictSolver enables you to identify these conflicts before you deploy packages, and resolve the problems before they affect your end users.

ConflictSolver allows you to check for a variety of conflict types, including components, files, registry entries, shortcuts, INI files, ODBC resources, NT services, file extensions, product codes, and merge module conflicts. In many cases, ConflictSolver can resolve the issues automatically. You can also create your own custom rules to ensure packages conform to your internal standards and practices.

The ConflictSolver user documentation is organized into the following areas:

Table 14-1 • ConflictSolver User Documentation

Section	Description
About ConflictSolver	Compares ConflictSolver and Application Manager, describes ConflictSolver views, explains how to connect to Application Catalogs from ConflictSolver, and describes the difference between Conflict ACE Rules and Best Practice ACE Rules.
Best Practices for Conflict Detection	Explains the “best practices” you should take to ensure optimal performance of ConflictSolver during package import and conflict detection.
Validating Packages	Optimally, packages should pass ICE validation. Topics in this section instruct you how to perform validation, and what to do if it fails.

Table 14-1 • ConflictSolver User Documentation (cont.)

Section	Description
Identifying and Resolving Conflicts for Windows Installer Packages	ConflictSolver's core purpose is to help you identify and resolve application conflicts. Topics in this section show you how.
Generating Reports	You can generate both pre-defined and user-defined reports using ConflictSolver.
Application Conflict Evaluators (ACEs)	ConflictSolver includes many rules that are used for identifying package conflicts. This section covers each rule, and also discusses how to create your own rules.
Conflict Application Resolution Definitions (CARDs)	Conflict Application Resolution Definitions (CARDs) are the rules used to fix conflicts identified by Application Conflict Evaluators (ACEs) when conflict detection is performed.
ConflictSolver Reference	Every view, dialog box, and Wizard available in ConflictSolver is discussed in this section.

About ConflictSolver

This section includes the following topics:

- Application Manager and ConflictSolver
- Types of ACE Rules: Best Practice and Conflict
- Connecting to Existing Application Catalogs

Application Manager and ConflictSolver

In previous releases, ConflictSolver was the only tool that provided a comprehensive view of the Application Catalog. Consequently, ConflictSolver was used to perform functions that are not related to conflict detection or resolution, such as importing and storing packages in a database, and organizing those packages.

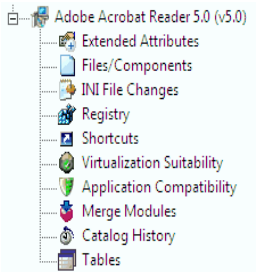
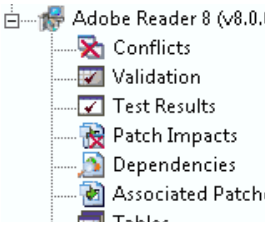
To reduce the complexity of ConflictSolver, the functionality that was being provided by ConflictSolver has been divided into two tools: ConflictSolver and Application Manager. This section compares these tools and includes the following topics:

- Functionality Comparison
- Accessing the Tools
- Displaying ConflictSolver Options in Application Manager
- Creating and Connecting to Application Catalogs

Functionality Comparison

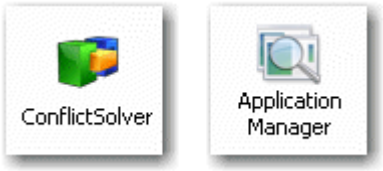
This table compares the functionality included in Application Manager and ConflictSolver.

Table 14-2 • Comparison of ConflictSolver and Application Manager Functionality

View	Major Functionality	Package Sub-Nodes
Application Manager	Use to perform the following tasks • **Import packages**—Import packages into the Application Catalog. • **Manage packages**—Organize packages into Groups, rename packages, and delete packages. • **Merge Wizard**—Merge data from a source Application Catalog into the currently open Application Catalog. • **Package Auto Import**—Set up the automatic import of packages. • **Replication**—Makes it possible to distribute an Application Catalog database where it is needed across a large enterprise. • **Software Repository**—Import packages into and check in and check out packages from the Software Repository. • **Associate a package with a Workflow Manager package** • **Launch applications**—Launch Predeployment Test Preparation Wizard, InstallShield Editor, and Distribution Wizard. • **Standard Reports**—View Package, File, and Registry reports. • **Application Readiness Dashboard**—Includes Deployment Type Breakdown, Error Category `Breakdown`, Virtualization Readiness, Package Quality, and Package Conflicts reports.	
ConflictSolver	Use ConflictSolver to perform the following tasks: • **Conflict Analysis and Resolution**—Check one or more packages for conflicts against others in the Application Catalog, and resolve those conflicts. • **Validation**—Validate Windows Installer package against Microsoft ICEs (Internal Consistency Evaluators), • **Dependency Scan**—Generate a list of all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog. • **Patch Impact Analysis Wizard**—Launch Patch Impact Analysis Wizard for selected package. • **Reports**—View Package, File, and Registry reports. For information on using ConflictSolver, see Identifying and Resolving Application Conflicts Using ConflictSolver.	

Accessing the Tools

Both Application Manager and ConflictSolver can be opened from the Start Page, the Tools Gallery, or from a menu shortcut.



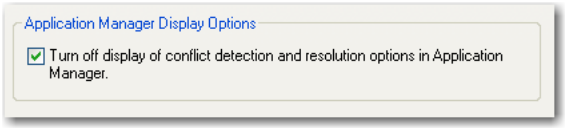
You can toggle between Application Manager, ConflictSolver, and ConflictSolver Process Assistant by clicking a button in the toolbar:

When the ConflictSolver or Application Manager is active, **ConflictSolver** or **Application Manager** is listed in the Title bar and also in the top right corner of the interface.

Displaying ConflictSolver Options in Application Manager

By default, all functionality for interacting with the Application Catalog, including conflict detection and resolution, is available in Application Manager.

To turn off the display of conflict detection and resolution options in Application Manager, select the checkbox in the **Application Manager Display Options** area of the **General** tab of the ConflictSolver and Application Manager Options Dialog Box.



Creating and Connecting to Application Catalogs

You can connect to Application Catalogs from both Application Manager and ConflictSolver as described in Connecting to an Existing Application Catalog. However, you use the AdminStudio interface to create new application catalogs. See Creating New Application Catalogs.

Types of ACE Rules: Best Practice and Conflict

When performing conflict analysis with ConflictSolver, you will be using two different types of ACE rules:

- Best Practice ACE Rules
- Conflict ACE Rules

Best Practice ACE Rules

Best Practice ACEs are rules that internally perform checks against the structure of a Windows Installer or App-V package to enforce best practices. Best Practice ACEs are similar to Microsoft ICEs. Both Microsoft ICEs (custom actions written by Microsoft used during validation) and Best Practice ACEs are used to determine if a package is built according to standards.

When testing a package against a Best Practice ACE Rule, there is no target package involved. The source package is tested to see if it meets the standards defined in the ACEs.

Conflict ACE Rules

Conflict ACE Rules are used to detect conflicts between two packages. You can use Conflict ACE Rules to detect conflicts between one or more source packages (Windows Installer and App-V) and one or more target packages in the Application Catalog. Conflict evaluation is done for each source package against each target package.

Note • *Multiple source package selection is only supported for internal packages (packages which have already been imported into the Application Catalog).*

Check Conflicts Across All Source and Target Packages

In order to provide a more meaningful conflict analysis, you can choose to have ConflictSolver perform conflict analysis of each source package against every other source package and each target package against every other target package. This option is enabled by selecting the **Check conflicts across all source and target packages** option on the **Rules** tab of the ConflictSolver **Options** dialog box. By default, this option is not selected.

Connecting to Existing Application Catalogs

From ConflictSolver, you can choose to connect to a standalone Application Catalog or the AdminStudio Enterprise Server Application Catalog.

For more information, see Connecting to an Existing Application Catalog.

Best Practices for Conflict Detection

When using ConflictSolver, the most database-intensive (and therefore most time consuming) tasks that you perform are importing packages and performing conflict detection. The time needed to complete these tasks depends upon how you have configured ConflictSolver and the steps that you follow. To ensure optimal performance of ConflictSolver during package import and conflict detection, we recommend that you incorporate the following best practices:

- **Plan Your Import**—If you are not using user-defined ACE rules, specify that you do not want to import binary data and custom tables. Also, specify that you do not want to check for validation errors or conflicts during import, and use the bulk import command line process. See Plan Your Import.
- **Plan Your Conflict Detection**—Perform conflict detection on internal packages only, and limit the scope of conflict detection by choosing to run only the ACEs that are relevant to your organization. See Plan Your Conflict Detection.
- **Perform Conflict Detection**—Follow this sequence of steps to quickly and efficiently perform conflict detection. See Perform Conflict Detection.

Plan Your Import

Planning your import consists of the following steps:

- Check and Configure Your Import Options
- Use the Command Line / Bulk Import Option of ConflictSolver

Check and Configure Your Import Options

Before initiating an import, you should review the following options on the **Import** tab of the ConflictSolver **Options** dialog box:

- **Import Binary Data & Import Custom Tables**—If you do not expect to run user-defined ACE rules, then you do not need to import either binary data or custom tables.
- **Check for Validation Errors and Check for Conflicts**—Checking packages for validation at the time of import is often unnecessary. Also, conflict detection at the time of package importation will become an exponentially slow operation as the size of your Application Catalog increases, since this option causes the package to be conflict checked against all existing packages in the Application Catalog.
- **Check for Windows Integrity**—Select the **PackageExpert** option on the **Import** tab if you want to run Windows quality validation tests and Windows Installer validation tests on newly imported packages. See Identifying and Resolving Package Errors Using PackageExpert.

Generally, unless it supports some specific business case, deselect the five import options on the **Import** tab of the ConflictSolver **Options** dialog box.

Use the Command Line / Bulk Import Option of ConflictSolver

ConflictSolver supports a bulk import process, which allows you to import multiple installer packages (such as MSI, SFT, MSM, MST, MSP, OSC and NCP files) at once without any user intervention. You can set up these files and have them run overnight or on an unattended workstation. By doing this, you minimize the need for someone to monitor the import process, freeing that person up to perform other tasks.

Also, if you use Bulk Import to import packages before they have been conflict checked, ConflictSolver performance is improved because it is quicker to perform conflict detection between internal packages than between internal packages and an external package.

Performing a Bulk Import

To perform a bulk import, use the following command line:

```
iscmide.exe -I -C"c:\mypackages\myconfig.ini"
```

If you are using a configuration file, the only necessary parameters to pass at the command line are `-I` and `-C"configuration_file_name.ini"`. You can include all other parameters inside the INI file. The following INI file is used to import packages into an SQL Server Application Catalog for a named user:

```
[General]
DatabaseType=SQL
LogFile=c:\temp\importlog.txt
PackageFile=4
MSMFile=1

[SQL]
Server=ConflictSolverSQL2K
UserID=Admin
Password=mypassword
Database=AdminStudio10

[PackageFile-1]
File=\\server\Data1.msi
Transform1=\\server\Data1a.mst
Transform2=\\server\Data1b.mst
Group=OfficeApps

[PackageFile-2]
File=\\server\ABCApplication.sft

[PackageFile-3]
File=\\server\Data2.msi
Group=OfficeApps\Secondary

[PackageFile-4]
File=\\server\Data3.msi

[MSMFile-1]
File=\\server\CrystalReports.msm
```

Important • *In previous releases, there was a `[MSIFile-n]` section in the `.ini` file. In AdminStudio 10.0, this has been changed to `[PackageFile-n]` to support the bulk import of both Windows Installer (`.msi`) and Microsoft App-V (`.sft`) packages. When updating your scripts, you should convert `[MSIFile-n]` entries to `[PackageFile-n]` entries.*

For more information on command line options, see Application Manager Command-Line Functionality and Application Manager Configuration File.

Organizing Your Bulk Import

If a package has not been conflict checked, import it into a isolated (working) group, check it for conflicts, and only move it into another group when it is free of conflicts. For purposes of conflict detection, you would not use any of the other packages in this working group as one of the target packages because they are still in the process of being analyzed for conflicts against the rest of the Application Catalog (potentially by other users).

Plan Your Conflict Detection

Planning your conflict detection consists of the following steps:

- Perform Conflict Detection on Internal Application Catalog Packages
- Check and Configure Your Conflict Options

Perform Conflict Detection on Internal Application Catalog Packages

Checking for conflicts against external packages is much slower than checking against a package already imported into the Application Catalog. Therefore, we recommend that you import all packages that you want to conflict check into the Application Catalog and select the **Internal Application Catalog package** option on the **Source Type** panel of the **Conflict Wizard**.

Check and Configure Your Conflict Options

ConflictSolver includes over 50 Application Conflict Evaluators (ACEs)—rules that allow ConflictSolver to identify conflicts between packages and between packages and operating system images.

Before beginning conflict detection, review the purpose of each of the ACEs. Then, on the **Rules** tab of the ConflictSolver **Options** dialog box, select only those conflicts that are appropriate for the packages you are checking. By limiting the scope of conflict detection, you will be improving the performance of the conflict detection process.

For a description of all ConflictSolver ACE rules, see Application Conflict Evaluators (ACEs).

Perform Conflict Detection

To ensure optimal performance of ConflictSolver during conflict detection, perform these steps in the following sequence:

- Pre-Conflict Detection Planning

- Conflict Detection Processing
- Post-Conflict Detection Evaluation

Pre-Conflict Detection Planning

Task: ***To perform pre-conflict detection planning***

1. Create a new SQL Server Application Catalog database using the **Create** command on the AdminStudio Interface **Catalog** menu. See Creating New Application Catalogs.

   Note • *If you have AdminStudio Enterprise Edition, you can also connect to the AdminStudio Enterprise Server Application Catalog. See Connecting AdminStudio Client Tools to the AdminStudio Enterprise Server Application Catalog.*

2. Use the **OS Snapshot Wizard** to generate snapshots for the desktop(s) used by your company. See Capturing an OS Snapshot.
3. Use ConflictSolver to create the necessary groups in the database to match how you will distribute these packages. See Adding Groups.
4. Import all OS Snapshot(s) into ConflictSolver using the bulk import process, and copy these snapshots into their appropriate groups. See Application Manager Command-Line Functionality.
5. If you intend to do conflict detection with ACE12 (meaning that you have access/rights to manipulate the MSI package in Tuner or InstallShield Editor to correct any conflicts that are found), then pre-import all Merge Modules that are used by your packages, ideally using the bulk import process. If you are not performing conflict detection on merge modules, you do not need to import merge modules.
6. Find out how these packages will be distributed and used in your enterprise, and then decide which ACEs you need to use during conflict detection.

Conflict Detection Processing

Task: ***To perform conflict detection processing***

1. Perform validation. Validation errors may not be serious, but you should be aware of them. Because the results of the ConflictSolver **Validation Wizard** are not persisted in the database, you can use another product, like Microsoft Orca, to do your validation, if desired. See Performing Validation.
2. Use the bulk import process to import your packages into a temporary working group.
3. Select your most critical, important application (perhaps for each group), and run validation and the source-only ACEs (ACE04, ACE05, ACE06) on it. If the package passes evaluation, then move it from the working group into its appropriate group. This package will represent the baseline against which all subsequent packages will be checked against.

4. Select your next package and run conflict detection with the ACE rules that you have identified as important against the group that this package will be assigned to. Evaluate the results and move the package into its appropriate group. Repeat the process. See Identifying and Resolving Conflicts for Windows Installer Packages.
5. You should always be evaluating one package against a group of already approved packages for conflicts. Any resolutions that you perform will be against the package you are evaluating.

Post-Conflict Detection Evaluation

Once you are done performing conflict detection with all of your packages, you should have groups of packages which can co-exist with each other.

Scanning for Dependencies

Scanning for dependencies generates a list of all of a Windows Installer package's files that have dependencies with files used by other products or operating systems in the Application Catalog.

If dependency information has not already been generated by the Patch Impact Analysis Wizard, you can generate dependency information for a Windows Installer package by selecting it and selecting **Scan for Dependencies** on the context menu.

Note • *For Microsoft App-V packages, a dependency scan is automatically performed during import into the Application Catalog, and the results are displayed on the App-V package's* ***Dependencies View****.*

Task: ***To scan a package for dependencies:***

1. If the package that you want to scan is stored in the Software Repository, open Application Manager, select the package node, and select **Check out** from the context menu. The package is checked out.
2. Open ConflictSolver, select the package node, and select **Scan for Dependencies** from the context menu.
3. Select the dependencies node to open the Dependencies View, which lists all of the selected package's files that have dependencies with files used by other products or operating systems in the Application Catalog. The following information is displayed:

Option	Description
Files With Dependencies	Make selections from this list to further filter the output listing, or select (All) to display all dependencies.
File	Name of a file included in this product that is dependent upon a file used by another product or operating system in the Application Catalog.
Dependent File	Name of the file that this product's file is dependent upon.

Option	Description
Size	Size of the dependent file. This field is populated for file dependencies identified from the MSI package itself. For dependencies that were derived from the static scanning process, this field will be undefined.
Version	Version of the dependent file. For dependencies that were derived from the static scanning process, this field will be undefined.
Language	Language of the dependent file. For dependencies that were derived from the static scanning process, this field will be undefined.

Note • *Regarding scanning for dependencies, note the following:*

- *If the **Only Display View Nodes With Data** option on the General tab of the ConflictSolver and Application Manager **Options** dialog box is selected, if you scan for dependencies and no dependencies are found, the **Dependencies** node will still not be displayed.*
- *Dependency information is also generated by the Patch Impact Analysis Wizard.*
- *If you want to perform validation, scan for dependencies, or conflict resolution on a package that is stored in the Software Repository, you must check out the package before you begin (because these actions may result in data in the package being changed).*

Validating Packages

An important first step in conflict identification and resolution is to validate the original Windows Installer package. Validation compares the package against Microsoft ICEs (Internal Consistency Evaluators), custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards. ICE rules return the following types of messages:

- **Errors**—Database authoring that causes incorrect behavior. An example of an error would be duplicate component GUIDs, which would cause the Installer to incorrectly register components.
- **Warnings**—Database authoring that can cause incorrect behavior. Warnings can also report unexpected side-effects of database authoring. An example of a warning would be entering the same property name in two conditions that differ only by the case of letters in the name. Because the Installer is case-sensitive, the Installer treats these as different properties.
- **Informational**—Information from the ICE that do not indicate a problem with the database. Often they are information about the ICE itself, such as a brief description. They can also provide progress information as the ICE runs.

Note • *For a full listing and description of all Microsoft ICEs, see ICE Reference in the Microsoft MSDN Library.*

Ideally, all packages that you import into ConflictSolver will pass validation, but realistically that will not be the case. ConflictSolver allows you to import any Windows Installer package into ConflictSolver, regardless of whether it has passed validation or not. However, should you import a Windows Installer package that does not pass validation, potential issues may arise due to the package not having been authored to Microsoft standards.

You can choose to validate an MSI package either before import, during import, or after import. If you choose to validate the package after it has been imported into the Application Catalog, those validation results are persisted, are listed on the **Validation** view, and are included in the Package Report (see Generating Package Reports).

Tasks associated with validation include:

- Performing Validation
  - Validating Before Import
  - Validating During Import
  - Validating After Import
- Changing the Default Validation File
- Excluding Specific ICEs from Execution During Validation
- Handling Invalid Windows Installer Packages
- About ICE43, ICE50, and ICE57 Validation Rules for Shortcuts

Performing Validation

You can choose to validate an MSI package before it is imported into the Application Catalog, during import, or after it has been imported:

- **Before Import**—You can use the Validation Wizard to validate an MSI package before it has been imported into the Application Catalog.
- **During Import**—If the **Check for Validation Errors** option is selected on the Import Tab of the **Options** dialog box, validation will automatically be performed when you use the Import Wizard to import an MSI package. Behind the scenes, the Validation Wizard checks the package for validation errors, and if the package fails validation, it will not be imported into the Application Catalog.
- **After Import**—After an MSI package has been imported into the Application Catalog, you can use the Validate Package function to perform validation.

Whether you perform validation before, during, or after import, validation results are displayed in the **Validate** tab of the Output Window. However, if you perform validation after an MSI package has been imported into the Application Catalog, those validation results are persisted, and can be viewed in the Validation View and are included in the Package Report.

Validating Before Import

Before importing a package into the Application Catalog, you may want to perform validation to ensure that the package is built according to Windows Installer standards.

Task: ***To validate a Windows Installer package prior to import:***

1. Select **Validation Wizard** from the **Conflicts** menu. The **MSI Source Information Panel** of the **Validation Wizard** opens.
2. Specify or browse to the Windows Installer package you want to validate.
3. If necessary, click the New button ( ) and add any associated transforms to include in the validation. For multiple transforms, you can move the transforms up and down to change the order in which they are applied to the base package (which can affect validation).
4. Click **Next**. The **Summary Panel** opens.
5. Review your options in the Summary Panel. Click **Finish** to begin validation.

   As validation occurs, messages are listed in the **Output** tab of the **Output Window**. When validation is complete, output appears in the **Validate** tab of the **Output Window** in table format. Each table row lists an icon to indicate whether it is an Error ( ) or a Warning ( ), the name of the ICE that generated it, and a brief description of what caused it to occur.
6. To view or print out a list of these validation errors for review, click in the **Output** tab of the **Output Window**, select **Select All** from the context menu, and then copy and paste the text into another application.

Tip • *Should you find any validation errors, you may be able to correct them using Tuner or InstallShield Editor.*

Note • *If you use the Validate Package function to validate a package after you have imported it into the Application Catalog (instead of performing validation before it is imported), you can view validation results in the Validation View and include validation results in the Package Report. See Validating an Imported Package and Generating Package Reports.*

Validating During Import

While you are importing a package into the Application Catalog using the Import Wizard, you can choose to also perform validation to ensure that the package is built according to Windows Installer standards. To do this, you would select the **Check for Validation Errors** option on the **Import** tab of the **Options** dialog box.

From this point, when you import a package using the Import Wizard, validation will automatically be performed. Behind the scenes, the Validation Wizard checks the package for validation errors and reports those results in the **Validation** tab of the Output Window. If the package fails validation (by generating an Error), the package will not be imported into the Application Catalog.

Tip • *Should you find any validation errors, you may be able to correct them using Tuner or InstallShield Editor.*

Task: ***To validate a package during import:***

1. Select **Options** from the **Tools** menu. The **Options** dialog box opens.
2. Open the **Import** tab.
3. Select the **Check for Validation Errors** check box.
4. Click **OK**.

Note • *If you validate a package during import, and want to view or print out a list of validation errors for review, click in the Output tab of the Output Window, select Select All from the context menu, and then copy and paste the text into another application. However, if you use the Validate Package function to validate a package after you have imported it into the Application Catalog (instead of performing validation during import), you can view validation results in the Validation View and include validation results in the Package Report. See Validating an Imported Package and Generating Package Reports.*

Validating After Import

Before performing conflict detection on an imported Windows Installer package, it is a good idea to perform validation by using the **Validate Package** function available on the Product View to ensure that the package is built according to Windows Installer standards.

The Validate Package function includes the following tasks and options:

- Validating an Imported Package
- Viewing ICE Error Information
- Suppressing an ICE Error
- Deleting an ICE Error

Validating an Imported Package

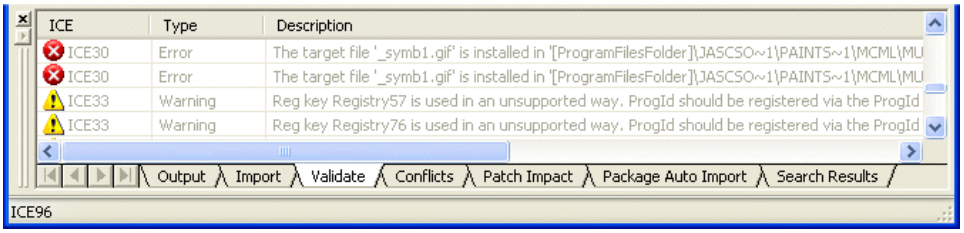
You can validate an imported package by using the Validate Package function on the Product View.

Task: ***To validate an imported Windows Installer package:***

1. Launch ConflictSolver.
2. On the **Product View**, select the Windows Installer package that you want to validate and select **Validate Package** from the context menu.

   Validation can only be performed on Windows Installer .msi packages. If you select another type of package in the Application Catalog, **Validate Package** will not be available on the context menu for that package.

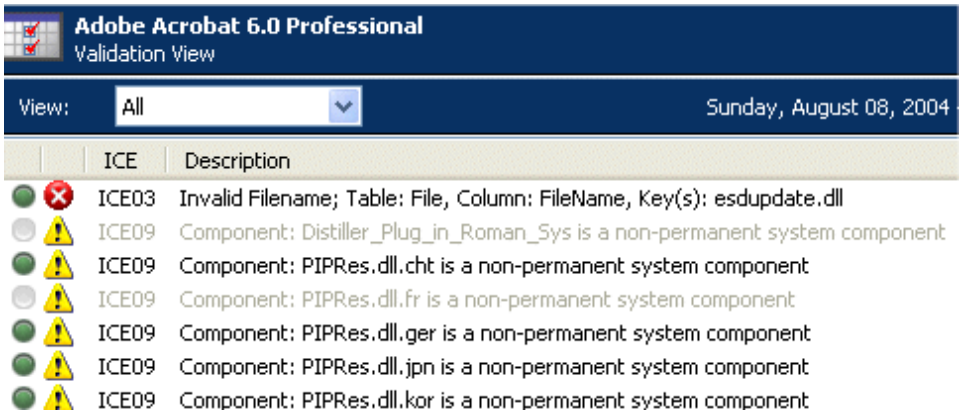
As validation occurs, messages are listed in the **Output** tab of the **Output** window. When validation is complete, output appears in the **Validate** tab of the **Output** window in table format. Each table row lists an icon to indicate whether it is an Error (⊗) or a Warning (⚠), the name of the ICE that generated it, and a brief description of what caused it to occur.



3. To view or print out a list of these validation errors for review, you have several options:
   - Click in the **Output** tab of the **Output Window**, select **Select All** from the context menu, and then copy and paste the text into another application.
   - Select the package and then select **Package** on the **Reports** menu or select **Package** under **Reports** on the context menu. See Generating Package Reports for more information.

Viewing ICE Error Information

After an imported package has been validated, detailed information on all of the ICE Errors and Warnings that were found can be viewed on the **Validation View**. Each ICE error that was found during validation is listed, with an icon to indicate whether it is an Error (⊗) or a Warning (⚠), the name of the ICE that generated it, and a brief Description of what caused it to occur.



You can also view the full text of the description, add explanatory text, and suppress or enable each ICE error on the Properties dialog box.

Task: ***To view ICE error information:***

1. Perform the steps listed in Validating an Imported Package.
2. On the **Product View**, expand the tree for the validated Windows Installer package and select the **Validation** node. The **Validation View** opens.
3. To easily view the full text of the ICE error description, select an ICE Error or Warning number and then select **Properties** from the context menu. The **Properties** dialog box opens listing the ICE Error Number and its full **Description**.
4. To add a comment to this ICE error, add text in the **Explanation** text box.
5. To set the **State** of an ICE, select **Suppressed** or **Enabled**. See Suppressing an ICE Error for more information.
6. Click **OK** to close the **Properties** dialog box.

Note • *The Validation View displays the ICE error information that was generated by using the Validate Package function available on the Product View for packages that have already been imported into the Application Catalog. Validation results generated for a package before it was imported into the Application Catalog or during import are not persisted and are not displayed on the Validation View. See Validating After Import*

Suppressing an ICE Error

If you do not want ConflictSolver to check for a particular ICE error during subsequent validations of a package, you can choose to suppress it. When you suppress an ICE error, it is listed in gray on the Validation View and is not executed in subsequent validations. You might choose to suppress an ICE error that is a known issue at your organization which does not need additional corrections.

Suppressed ICE errors are still listed in the Package Report for that package. To remove an ICE error from the Package Report, you need to Delete it rather than Suppress it. See Deleting an ICE Error and Generating Package Reports.

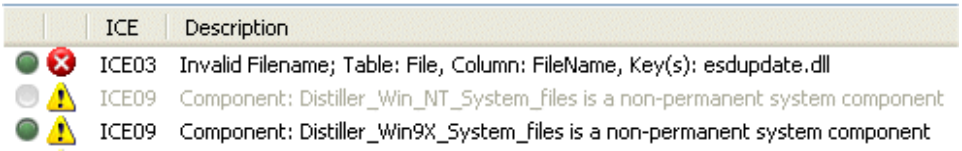
Task: ***To suppress an ICE error:***

1. Perform the steps listed in Validating an Imported Package.
2. On the **Product View**, expand the tree for the validated Windows Installer package and select the **Validation** node. The **Validation View** opens.
3. Select the ICE error that you want to suppress and then select **Properties** from the context menu. The **Properties** dialog box opens.
4. Under **State**, select **Suppressed**.

Note • *You can also choose to enable or suppress an ICE error by selecting the ICE error on the* **Validation View** *and then selecting* **Enable** *or* **Suppress** *from the context menu. When using the context menu to set an ICE error's state, you can perform the operation on multiple ICE errors at once. Use the Shift key to select multiple contiguous items in the list, and use the Ctrl key to select multiple non-contiguous items.*

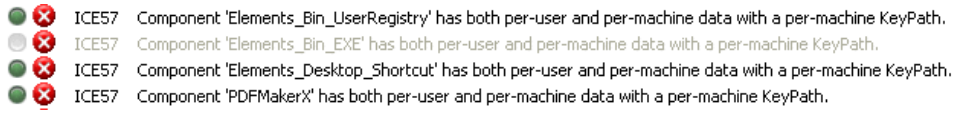
5. You could choose to document the reason that you are suppressing this ICE error in the **Explanation** text box.
6. Click **OK** to close the **Properties** dialog box.

   On the **Validation View**, the ICE error that you just suppressed is now listed in gray:



Suppressing a ICE Error vs. an ICE Rule

When you suppress an ICE error, you are just suppressing that particular error; you are not suppressing subsequent checks for an entire ICE rule during subsequent validations of that package. For example, if you suppress one ICE57 error, all of the other ICE57 errors are not suppressed:



To suppress all ICE errors for an ICE rule, do one of the following:

- **To suppress an ICE rule for one package**—Perform the steps listed in Validating an Imported Package, and then go to the Validation View and suppress all ICE errors found for that ICE rule.
- **To suppress an ICE rule for all packages in the Application Catalog**—Enter the ICE rule number on the **Validate** tab of the **Options** dialog box. See Excluding Specific ICEs from Execution During Validation.

Deleting an ICE Error

You can permanently delete an ICE error from subsequent execution and reporting for that package. ICE errors that are deleted are no longer listed in the Validation View and will not be listed in the Package Report.

Task: ***To delete an ICE error:***

1. Perform the steps listed in Validating an Imported Package.
2. On the **Product View**, expand the tree for the validated Windows Installer package and select the **Validation** node. The **Validation View** opens, listing all ICE errors for the selected package.

3. Select the ICE number of the specific ICE error(s) you want to delete and select **Delete** from the context menu. You can delete multiple ICE errors at once. Use the **Shift** key to select multiple contiguous items in the list, and use the **Ctrl** key to select multiple non-contiguous items.

   - If you are deleting an ICE error that is currently Suppressed, you will be asked to confirm the deletion.
   - If you are deleting an **Enabled** ICE error, no confirmation message will appear.

   The deleted ICE error will no longer be listed in the **Validation View** or on the Package Report for that package. You have permanently suppressed the selected ICE error, and ConflictSolver will not execute that ICE error during subsequent validations of this package.

Changing the Default Validation File

Validation involves comparing a Windows Installer package against a known criteria to identify deviations from those rules. By default, ConflictSolver compares packages against the Full MSI Validation Suite. This suite contains a comprehensive set of Internal Consistency Evaluators (ICEs)—guidelines created by Microsoft to ensure an installation package works correctly with the Windows Installer engine. In the overwhelming majority of cases, this is the file you will want ConflictSolver to use. However, there may be times you want to compare your base package against a different validation file, depending on your needs.

Task: ***To specify a different validation file to use:***

1. Select **Options** from the **Tools** menu. The **Options** dialog box opens.
2. Open the **Validate** tab.
3. In the **MSI Input files** and **MSM Input files** areas, click the Browse button (…) next to the **Cub File** field to locate the validation file (`.cub`) that you would like to use.
4. To permanently exclude the execution of specific ICEs, list them in the **ICE Rules** field, separated by semicolons (;). Otherwise, all rules will be used. See Excluding Specific ICEs from Execution During Validation for more information.

Excluding Specific ICEs from Execution During Validation

You can permanently exclude the execution of specific ICEs that are listed in the selected `.cub` file.

Task: ***To exclude the execution of specific ICEs during validation:***

1. Select **Options** from the **Tools** menu. The **Options** dialog box opens.
2. Open the **Validate** tab.
3. In the **MSI Input files** and **MSM Input files** areas, list the ICEs that you want to exclude in the **ICE Rules** field, separated by semicolons (;), such as:

```
ICE07; ICE09; ICE22
```

All of the ICEs listed in the specified `.cub` files will be executed during validation except for the ICEs entered here.

Handling Invalid Windows Installer Packages

Ideally, all Windows Installer packages will pass validation. Realistically, many will fail. When a package fails validation, it means the package was not built to Microsoft's specifications. It does not mean the installation does not work. However, there are a few things you can do when your package has validation errors:

Table 14-3 • Validation Error Solutions

Solution	Explanation
Use Tuner to correct validation errors.	This involves opening the base package using Tuner and creating a transform file which contains your corrections. The same validation tool is available in Tuner.
Contact the installation vendor.	The company that created the installation (usually the same company that created the software) may be able to resolve the validation issues and provide you with a valid setup. Be sure to provide the validation report to vendors so they know where to focus.
Reconsider using the application.	Although it might be an extreme reaction to an invalid package, there may be compelling reasons not to use an installation not built to Microsoft guidelines.
Ignore the problems and import the package into ConflictSolver anyway.	This is probably the most likely scenario. The invalid installation may not be worth trying to fix, or even have errors that you are concerned about. ConflictSolver allows you to import an installation package regardless of its validity; just know that there could be problems in the future in doing so. From a practical standpoint, this may be your best option.

About ICE43, ICE50, and ICE57 Validation Rules for Shortcuts

Each entry in the **CreateLink** section of the `.inc` file is converted into an entry in the Shortcut table. The exact properties of the shortcut depends upon the information in the **CreateLink** line as well as the nature of the target file itself.

ICE 43, ICE 50, and ICE 57 are the most common validation rules for shortcuts.

Shortcut Types

The primary distinction between shortcuts is advertised vs. non-advertised. Here are two reasons why it is preferable to create advertised shortcuts:

- Advertised shortcuts are triggers for MSI's self-repair mechanism.
- Non-advertised shortcuts are intended for a per-user context only:
  - The target file must be a file installed in a user-specific directory.
  - The keypath of the component containing the target file must be a user-specific registry value.

Conversion from CreateLink entries to Shortcut Table Entries

The .inc converter will always try to create advertised shortcuts for every CreateLink line found in the .inc file. However, not every CreateLink line can be converted into an advertised shortcut.

In order to create an advertised shortcut, the information in the CreateLink line must meet all of the following requirements:

- The target file must be the keypath of its component. This means that the target file must be listed in the .inc file list. The converter will create a new component for a non-PE (portable executable) target files, so that it is guaranteed to become the keypath of the component. (Normally, for each target directory, non-PE files are grouped together into one component).
- The target file must contain an icon.

In general, this means that as long as the target file contains an icon, the converter will be able to create an advertised shortcut for it. However, whenever a shortcut cannot be advertised, the converter does the following:

- It creates a "catch all" component (if not yet created) named `ShortcutsComponent`. It also creates an HKCU registry entry in the Registry table, and that entry is used as the keypath for the `ShortcutsComponent`. This is done to avoid ICE43.
- A new shortcut entry is created in the Shortcut table, associated with the `ShortcutsComponent`.

Identifying and Resolving Conflicts for Windows Installer Packages

Conflict identification occurs if you import a package into ConflictSolver using the Import Wizard, providing the **Check for Conflicts** option has been selected in the **Import** tab of the **Options** dialog box. ConflictSolver checks for conflicts using the default rules specified in the **Rules** tab of the **Options** dialog box.

You can also identify conflicts using the Conflict Wizard. From the Source Type Panel, you can select whether you want to check an external Windows Installer package against packages in the Application Catalog, or check one or more packages already in the Application Catalog against others in the Application Catalog.

The following topics explain how to perform conflict identification and resolution:

Table 14-4 • Conflict Identification and Resolution Tasks

Section	Topics
Best Practices for Conflict Detection	• Plan Your Import • Plan Your Conflict Detection • Perform Conflict Detection
Setting Options	• Changing Default Conflict Types Checked • Changing Resolution Options
Checking for Conflicts	• Checking for Conflicts Using the Conflict Wizard • Checking for Conflicts During Import • Viewing Identified Conflicts in the Conflicts View • Running Conflict Identification Again • Deleting Persisted Conflict Data • Conflict Persistence
Resolving Conflicts	• Conflict Resolution Process • Automatically Resolving Conflicts • Manually Resolving Conflicts • Resolving Conflicts Directly in Windows Installer Packages • Using Transforms for Conflict Resolution • Reimporting Packages after Successful Conflict Resolution

Using Multiple Source Packages in an Analysis

When performing conflict analysis between packages in the Application Catalog, you may choose to select either one or multiple source packages and multiple target packages. Conflict analysis is then performed for each source package against each target package.

Also, in order to provide a more efficient conflict analysis, you can choose to have ConflictSolver perform conflict analysis of each source package against every other source package and each target package against every other target package. This option is enabled by selecting the **Check conflicts across all source and target packages** option on the **Rules** tab of the **Options** dialog box. By default, this option is not selected.

Setting Options

You can set the following ConflictSolver conflict identification and resolution options:

- **Changing Default Conflict Types Checked**—Select the ACE rules that will be selected by default during conflict detection.

- **Changing Resolution Options**—Specify whether you want to save resolutions in the original MSI package or in a transform file.

Changing Default Conflict Types Checked

In the **Enabled Predefined and Custom Rules** area of the **Rules** tab of the **Options** dialog box, you can select the default set of ACEs that will be included in conflict analysis.

ACEs associated with unselected boxes will not be performed by default during conflict identification. However, you can override these settings in the **Summary** panel of the **Conflict Wizard**.

Task: ***To change default conflict types checked:***

1. Select **Options** from the **Tools** menu. The **Options** dialog box opens.
2. Open the **Rules** tab.
3. In the **Enabled Predefined and Custom Rules** area, select the **Show best practice rules** option. The **Best Practices** ACE rules tree is displayed.
4. Expand the tree and select the Best Practice ACE rule conflicts you want to check for from the list.
5. Select the **Show conflict detection rules** option. The **Conflict Types** ACE rules tree is displayed.
6. Expand the tree and select the **Conflict Types** ACE rule conflicts you want to check for from the list.
7. When performing conflict analysis using multiple source packages and one or more target packages, ConflictSolver will evaluate each source package against each target package. However, if you want ConflictSolver to also perform conflict analysis of each source package against every other source package and each target package against every other target package, select the **Check conflicts across all source and target packages** option.
8. Click **OK**.

The ACE rules associated with each conflict type are listed in the following table:

Table 14-5 • Best Practice ACE Rules

ACE Type	Conflict Type	Associated Rules
Best Practice ACEs	**Components**	ACE04, ACE05, ACE06,
	Merge Module Integrity	ACE26, ACE36
	Recommended Rules	ACE25, ACE27, ACE28, ACE29, ACE31, ACE32, ACE33, ACE34, ACE35
	Windows Terminal Server Validation	WTS01, WTS02, WTS03, WTS04, WTS05, WTS06, WTS07, WTS08, WTS09

Table 14-5 • Best Practice ACE Rules

ACE Type	Conflict Type	Associated Rules
Conflict ACEs	**Component**	ACE02, ACE09, ACE30
	File Extensions	ACE17
	Files	ACE03, ACE07, ACE08, ACE12, ACE23
	INI Files	ACE14, ACE21, ACE22
	NT Services	ACE16
	ODBC Resources	ACE15
	Product Properties	ACE18, ACE19, ACE20
	Registry	ACE10, ACE24
	Shortcuts	ACE13

Changing Resolution Options

You can change resolution options on a product-by-product basis.

Task: ***To change resolution options:***

1. From the **Product View**, select the product for which you want to change resolution options.
2. Click on the plus sign to expand the tree listing and select the **Conflicts** node. The **Conflicts** view opens.
3. Ensure the **Summary** view is selected from the **Conflict Information** list.
4. In the **Resolution Options** area, click **Edit**. The **Resolution Options** dialog box opens.
5. Select whether you want to save resolutions in the **original MSI package** or in a **transform file**. If you select a transform file, browse to or specify it.
6. Click **OK**.

Any resolutions you make are stored in the specified transform (if selected) or in the original package. By selecting the **Perform resolutions against a transform file** option in the **Resolution** tab of the **Options** dialog box, you can set the default behavior to store resolutions in a transform.

Note • *You can only modify resolution options if persisted conflict data exists for a product.*

Checking for Conflicts Across All Source and Target Packages

When performing conflict analysis using multiple source packages and one or more target packages, ConflictSolver will evaluate each source package against each target package.

However, if you want ConflictSolver to also perform conflict analysis of each source package against every other source package and each target package against every other target package, select the **Check conflicts across all source and target packages** option on the Rules Tab of the **Options** dialog box.

***Note** • If performing conflict analysis with multiple source packages, and if this option is selected, both ACE21 and ACE22 are automatically evaluated when either one is selected. If only one of them is evaluated, then there is the possibility that conflicts will not be detected.*

Checking for Conflicts

The following topics are included in this section:

- Checking for Conflicts Using the Conflict Wizard
- Checking for Conflicts During Import
- Viewing Identified Conflicts in the Conflicts View
- Running Conflict Identification Again
- Deleting Persisted Conflict Data
- Conflict Persistence

Checking for Conflicts Using the Conflict Wizard

The Conflict Wizard can be used to identify conflicts between an external Windows Installer package and ones already in the Application Catalog, or between one or more packages in the Application Catalog.

Identifying Conflicts with an External Package

The ConflictSolver Wizard allows you to run conflict checks against external MSI packages and packages imported into the ConflictSolver Application Catalog Database. ConflictSolver supports external package conflict checking for User Defined Custom/Source Only Packages ACEs. The Source package can be selected from the ConflictSolver Application Catalog Database or from an external MSI package. See Creating a Custom/Source Only Packages ACE for more information.

However, ConflictSolver does not support external package conflict checking for Custom/Source and Target Packages ACEs. Both the Source and Target Packages must be selected from the ConflictSolver Application Catalog Database.

Task: ***To identify conflicts between an external Windows Installer package and packages already in the Application Catalog:***

1. Select **Conflict Wizard** from the **Conflicts** menu. The **Welcome Panel** opens.
2. Click **Next**. The **Source Type Panel** opens.
3. Select **Windows Installer** package. Click **Next**. The **MSI Source Information Panel** opens.
4. Click **Browse** to locate the MSI file you want to check against packages in the Application Catalog.
5. Should that package require transforms, click the New button ( ) above the **Transforms** area. Use the Browse button (...) to locate the transform. If the package requires multiple transforms, you can repeat the procedure as necessary. The order in which transforms are applied can be changed by selecting a transform and clicking the Move Up ( ) and Move Down ( ) buttons. If you need to delete a transform you have added, select the transform and click the Delete button ( ).
6. Click **Next**. The **Target Information Panel** opens.
7. Select the packages in the Application Catalog you want to compare the selected package against. You can also select entire groups, which will compare the package against all packages and OS Snapshots in that group.
8. Click **Next**. The **Conflict Rules Panel** opens.
9. Select the types of conflicts you want to identify and click **Next**. The **Summary Panel** opens.
10. Review the options selected in the Summary Panel and click **Finish** to begin the conflict identification process. Progress appears in the **Output** tab of the **Output Window**.

    When the conflict analysis is complete, all of the Errors and Warnings that were generated are listed in a table format in the **Conflicts** tab of the **Output Window**.

    If a row is active, you can double-click on it to view that row's associated table. The **Tables View** is launched and the table and/or table cells that are causing the problem are highlighted in red. If a row is grayed out, it indicates that the table cannot be viewed in the ConflictSolver (perhaps because it is in an external package). For information on resolving conflicts, see Resolving Conflicts.

Identifying Conflicts Between Existing Packages

To identify conflicts between packages in the Application Catalog, perform the following steps.

Task: ***To identify conflicts between packages already in the Application Catalog:***

1. Select **Conflict Wizard** from the **Conflicts** menu. The **Welcome Panel** opens.
2. Click **Next**. The **Source Type Panel** opens.
3. Select **Internal Application Catalog package** and click **Next**. The **Source Package Panel** opens.
4. Select the package(s) you want to compare against other package(s) in the Application Catalog.

When performing conflict analysis using multiple source packages and one or more target packages, ConflictSolver will evaluate each source package against each target package. However, if you want ConflictSolver to also perform conflict analysis of each source package against every other source package and each target package against every other target package, select the **Check conflicts across all source and target packages** option on the **Rules** tab of the **Options** dialog box.

5. Click **Next**. The **Choose Action** panel opens.
6. To include Best Practice ACE rules in this conflict analysis, select the **Evaluate source package against best practice rules** option.
7. To include Conflict ACE rules in this conflict analysis, select the **Detect conflicts between source package and other packages** option.
8. To display the **Conflict Resolution** dialog box after conflict detection, select the **Run Resolution** option.

   From the **Conflict Resolution** dialog box, you can access information pertaining to the last execution of the Conflict Wizard. This persisted conflict data allows you to view when the last execution was performed, the products that it was run against, the ACE rules used, and conflicts discovered. The dialog box also provides a list of updated, deleted, or added products that may necessitate performing conflict identification again.
9. To generate a conflict detection analysis report containing a summary of the conflicts that were detected, and a detailed description of each identified conflict including how this conflict will be resolved, select the **Generate Reports** option.
10. Click **Next**. If you selected the **Evaluate source package against best practice rules** option on the **Choose Action Panel**, the The **Best Practice Rules Panel** opens.
11. Select the Best Practice Rules that you want to evaluate in conflict analysis. By default, the Best Practice ACEs that are selected on the **Rules** tab of the **Options** dialog box will be selected.
12. Click **Next**. If you selected the **Detect conflicts between source package and other packages** option on the **Choose Action Panel**, the **Conflict Rules Panel** opens.
13. Select the Conflict Rules that you want to evaluate in conflict analysis. By default, the Conflict Rules ACEs that are selected on the **Rules** tab of the **Options** dialog box will be selected.
14. Click **Next**. If you selected any Conflict Rules to evaluate, the **Target Information Panel** opens.
15. Select all the packages you want to compare the source package(s) against. (The Target Information Panel excludes all packages that you selected on the Source Package Panel.) Click **Next**. The **Summary Panel** opens.
16. Review the options selected in the **Summary Panel** and click **Finish** to begin the conflict identification process. Progress appears in the **Output** tab of the **Output Window**.

    When the conflict analysis is complete, all of the Errors and Warnings that were generated are listed in a table format in the **Conflicts** tab of the **Output Window**.

    If a row is active, you can double-click on it to view that row's associated table. The **Tables View** is launched and the table and/or table cells that are causing the problem are highlighted in red. If a row is grayed out, it indicates that the table cannot be viewed in ConflictSolver (perhaps because it is in an external package). For information on resolving conflicts, see Resolving Conflicts.

17. If you selected the **Run Resolution** option on the **Choose Action panel**, the **Conflict Resolution** dialog box opens, listing the total number of conflicts that were identified and the number of products and rules that were checked. See Resolving Conflicts for more information.

Checking for Conflicts During Import

Task: ***To check for conflicts during import:***

1. Select **Options** from the **Tools** menu. The **Options** dialog box opens.
2. Open the **Import Tab**.
3. Select the **Check for Conflicts** check box.
4. Open the **Conflicts** tab.
5. Select the conflict types you want checked by default when importing packages.
6. Click **OK**.

From this point, when you import a package into the Application Catalog, the package will automatically be checked for conflicts against all packages already in ConflictSolver using the default ACE rules selected in the **Conflicts** tab of the **Options** dialog box.

Depending on how many packages already in ConflictSolver, and depending on the number of ACE rules selected by default, this can take a significant amount of time.

Viewing Identified Conflicts in the Conflicts View

Task: ***To view identified conflicts:***

1. From the **Product View**, select the product for which you want to view identified conflicts.
2. Select the Conflicts icon () to view the **Conflicts View** for the product.

   The Summary view displays an overall summary of the conflicts identified, including the products this product was checked against for conflicts, the ACE rules used, and resolution options.

   You can change the Conflict Information list at the top of the view to see individual conflict categories and perform resolution. See Resolving Conflicts for more information.

Suppressing ACE Errors

If you do not want ConflictSolver to check for a particular ACE error or warning during subsequent analyses of a package, you can choose to suppress it by selecting it and clicking the **Suppress** button.

Suppressed ACE errors are listed in gray, and are preceded by a white icon, while enabled ACE errors are preceded by a green icon:

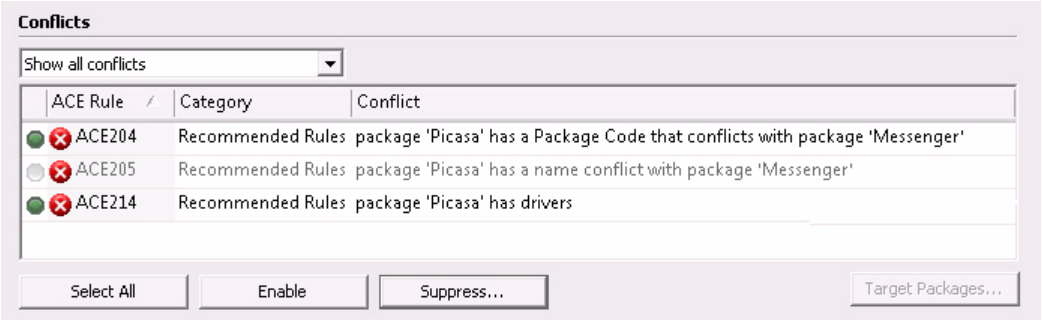


Figure 14-1: Suppressing an ACE Error

Note • *This function is similar to the **Enable** and **Suppress** right-click menu options in the **Validation** View for Windows Installer packages.*

Consolidated Conflicts Listing

When evaluating conflicts between a source package and multiple destination packages, the same conflict may be detected between the source package and more than one destination package. In the **Conflicts View** listing, these conflicts are consolidated and displayed as single entry whenever possible.

To see a list of all of the target packages that detected the same conflict, click the **Target Packages** button to view a list of the package on a pop-up dialog box.

Note • *You can also view conflicts generated from the comparison of an external package with those packages already in ConflictSolver by selecting* **Resolve External Package Conflicts** *from the* **Conflicts** *menu. The resulting* **Conflict Resolution** *dialog box displays conflict results from the last conflict execution for an external package.*

Running Conflict Identification Again

At times, it may be necessary to run conflict identification again on a package that you have previously checked for conflicts. This could be because other packages you previously compared this package against have changed. It may be because you have added additional packages to ConflictSolver.

There are two ways to accomplish this.

- If you want to run conflict identification again using the same options as the last time you ran conflict identification on the package, you can do so by clicking the Run Again button from the package's Conflicts View.
- In all other cases, use the Conflict Wizard to perform conflict identification.

Note • *When you run conflict identification for a package on which you have previously performed conflict identification, the persisted conflict data is deleted and replaced with the new conflict data.*

Deleting Persisted Conflict Data

Task: ***To delete persisted conflict data from a package:***

1. Right-click on the package in the **Product View**, point to **Delete**, and select **Persisted Conflict Information**.
2. Confirm the deletion.

Conflict Persistence

ConflictSolver automatically persists conflict information for packages in ConflictSolver, and for the last execution of the Conflict Wizard between an external package and packages already in ConflictSolver. This enables you to run conflict identification between packages without having to perform the resolution immediately.

Your workflow may, for example, dictate that you import several packages into ConflictSolver first (placing them in some sort of staging group) and then perform conflict identification on them all at once. Resolution of identified conflicts may be much later in the workflow. ConflictSolver assists you by retaining the conflict information until you are ready to work with it.

Each package in ConflictSolver includes a Conflicts View, where this persisted conflict data can be accessed. You can see when conflict identification was performed, between which packages, using which ACE rules, and the result. You can also see which packages have been updated, added to, or deleted from ConflictSolver since conflict identification was performed.

Resolving Conflicts

The following topics are included in this section:

- Conflict Resolution Process
- Automatically Resolving Conflicts
- Manually Resolving Conflicts
- Resolving Conflicts Directly in Windows Installer Packages
- Using Transforms for Conflict Resolution
- Reimporting Packages after Successful Conflict Resolution

Note • *Automatic conflict resolution is supported for Windows Installer packages only (not for App-V packages). See Testing Microsoft App-V Packages.*

Caution • *If you want to perform validation, scan for dependencies, or conflict resolution on a package that is stored in the Software Repository, you must check out the package before you begin (because these actions may result in data in the package being changed).*

Conflict Resolution Process

After conflict analysis is performed, either by using the Conflict Wizard or by checking for conflicts during import, all of the Errors and Warnings that were generated are listed in the Conflicts tab of the Output Window in a table format. You can resolve these conflicts by clicking Resolve or Resolve All on the Conflicts View. However, the Conflict Resolution Process differs depending upon options you select on the Resolution Tab of the Options dialog box.

Review and Approve Conflicts

The selection you make for the **Preview and approve all resolutions** option on the **Resolution** tab of the **Options** dialog box determines if you will be prompted to review resolutions before they are executed.

- If the **Preview** and **Approve all resolutions** option is selected, the **Resolution Details** dialog box appears when you click **Resolve** or **Resolve All** on the Conflicts View.

  This dialog box lists the proposed CARD resolutions that will be made to the package. Only the CARD resolutions that are selected when you click **Approve** will be executed.

- If the **Preview and approve all resolutions** option is not selected, the **Resolution Details** dialog box will not appear, and all selected conflict resolutions will be executed.

- If the **Resolution Details** dialog box opens but you no longer want it to, select the **Don't show this dialog again** check box. This is equivalent to clearing the **Preview and approve all resolutions** option on the **Resolution** tab of the **Options** dialog box.

User Input Option

If a CARD can be resolved in multiple ways, and if the Maximize user flexibility in selecting resolutions option is selected on the **Resolution** tab of the **Options** dialog box, the **CARD Resolution Options** dialog box appears, prompting you to make a selection to determine how conflict resolution will proceed. The following CARD can be resolved in multiple ways:

- **CARD06**—ACE06 accumulates a list of potential files to be used for the KeyPath. The CARD Resolution Options dialog box appears, prompting you to choose one file from the list of potential files. See CARD06 for more information.

Conflict Resolution and the Software Repository

If a package is stored in the Software Repository, resolution and re-import behaviors (behaviors that modify the Application Catalog) are dependent upon a package's checked-in or checked-out status.

Resolution Behaviors

Because conflict resolution requires that the package be modified, if a package is stored in the Software Repository and is currently checked-out by another user, it cannot be modified, and therefore cannot be resolved. This resolution behavior is detailed in the following table.

Table 14-6 • Job Manager Resolution Behavior Regarding the Software Repository

Package State	Resolution Behavior
Package in Software Repository / Checked Out	Job Manager ignores the resolution process for these packages. The packages are already being edited by someone else.
Package in Software Repository / Not Checked Out	The package will be checked out by the virtual user **Job Manager Engine** prior to the normal operation of the resolution process. If the resolution process results in a no-operation or the user does not elect to re-import the package, then the check-out operation will be cancelled.
Package Not in Software Repository	The resolution process will operate normally.

Re-Import Behaviors

When defining a Resolution Job Step, you have to indicate the reimport option for resolved packages. When defining a **Resolution** Job Step, you have the choice of three re-import options for resolved packages: re-import as a new version, re-import by overwriting the existing version, or do not re-import.

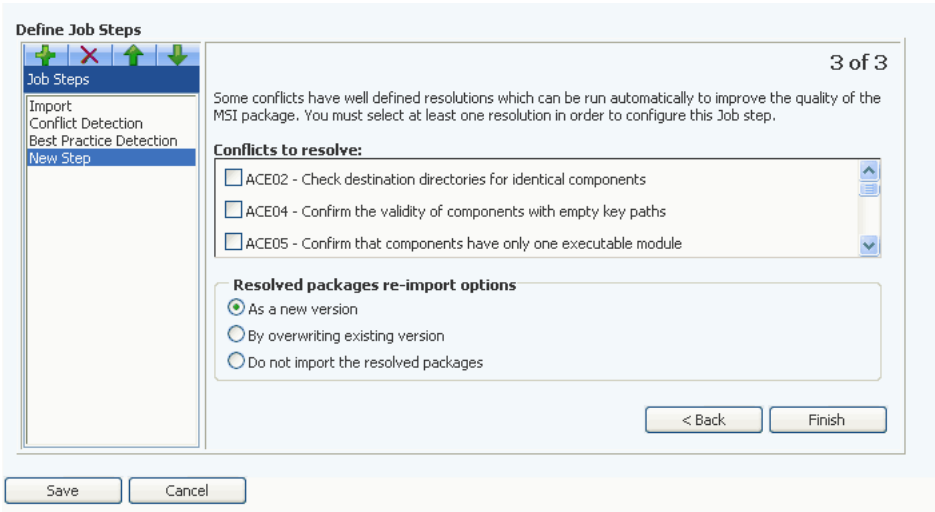


Figure 14-2: Reimport Options in Job Manager

The choice you make depends upon whether the package is stored in the Software Repository:

Table 14-7 • Resolved Package Re-Import Options

Resolved Packages Re-import Options	Package in Software Repository	Package Not in Software Repository
As a new version	Package will be re-imported and checked-in as a new version.	N/A Package cannot be imported as a new version unless it is in the Software Repository.
By overwriting existing version	Package will be re-imported and checked-in, replacing the existing version.	Package will be re-imported, replacing the original package.
Do not import the resolved packages	Package will not be re-imported.	Package will not be re-imported.

Automatically Resolving Conflicts

If the Conflict Wizard identifies conflicts, ConflictSolver may be able to resolve some of them automatically (ACE02, ACE04, ACE05, ACE06, ACE07, ACE15, ACE18, ACE19, and ACE20). You first need to view conflicts identified, and then select to resolve them.

Task: ***To resolve an automatically resolvable conflict:***

1. From the **Product View**, select the product for which you want to view identified conflicts.
2. Select the **Conflicts View** for the product.
3. Select the category containing the automatically resolvable conflict from the **Conflict Information** list.
4. Expand the ACE containing conflicts that can be resolved.

   The icon displayed next to the conflict indicates its status:

Status	Icon
Resolvable	
Resolved	
Unresolvable	

5. Select the conflict you want to resolve and click **Resolve**. Alternatively, click **Resolve All** to resolve all resolvable conflicts for the selected conflict category.
6. For CARDs that can be resolved in multiple ways, the **CARD Resolution Options** dialog box appears, prompting you to make a selection to determine how conflict resolution will proceed. Make a selection and click **OK** to continue.

   Note • *CARDs can be resolved in multiple ways if the* **Maximize user flexibility in selecting resolutions** *option is selected on the* **Resolution** *tab of the* **Options** *dialog box.*

7. If the **Preview and approve all resolutions** option on the **Resolution** tab of the **Options** dialog box is selected, the **Resolution Details** dialog box will appear, listing the proposed CARD resolutions that will be made to the package.

   Select the CARD resolutions that you want to execute, and click **Approve** to execute them.

   If the **Preview and approve all resolutions** option is not selected, the **Resolution Details** dialog box will not appear, and all selected conflict resolutions will be executed.

Resolutions are made to either the original Windows Installer package or in a transform (if the Perform resolutions against a transform file option is selected on the Resolution tab of the Options dialog box).

Manually Resolving Conflicts

Due to their complexity, some conflicts require manual resolution using Tuner or InstallShield Editor.

For information on manual solutions, consult the Application Conflict Evaluators (ACEs) topic or individual ACE topics. These topics can provide valuable information regarding what is causing the conflict, so you can use Tuner or InstallShield Editor to solve it.

Once you have resolved detected conflicts manually, you can reimport the package into ConflictSolver.

Resolving Conflicts Directly in Windows Installer Packages

ConflictSolver provides two methods for implementing conflict resolutions: directly in the Windows Installer package, or through the use of a transform.

Task: ***To configure ConflictSolver to resolve conflicts directly in the original Windows Installer package:***

1. Select **Options** from the **Tools** menu to open the **Options** dialog box.
2. On the **Resolution** tab, clear the **Perform resolutions against a transform file** check box.
3. Click **OK** to close the **Options** dialog box.

With this setting disabled, all conflict resolutions are by default performed directly on the base Windows Installer package. In either case, you can override this setting on a package-by-package basis by clicking **Edit** in the **Summary** section of the **Conflicts View** and editing the resolution options on the **Resolution Options** dialog box.

Using Transforms for Conflict Resolution

ConflictSolver provides two methods for implementing conflict resolutions: directly in the Windows Installer package, or through the use of a transform.

Task: ***To configure ConflictSolver to use transforms for conflict resolutions:***

1. Click **Options** on the **Tools** menu. The **Options** dialog box opens.
2. Open the **Resolution Tab** and select the **Perform resolutions against a transform file** check box.
3. Click **OK** to close the **Options** dialog box.

With this setting enabled, all resolutions will by default be placed within a transform, rather than being made in the Windows Installer package. In either case, you can override this setting on a package-by-package basis by clicking Edit in the Summary section of the **Conflicts View** and editing the resolution options on the **Resolution Options** dialog box.

Reimporting Packages after Successful Conflict Resolution

Following resolution of conflicts, you should reimport the package into ConflictSolver. If you have performed automatic resolutions, you can do so by clicking the **Do you want to reimport this package?** link in the **Product View** for that package. The package (and the resolution transform if used) are reimported into ConflictSolver.

However, if you have performed manual resolutions to the package via a transform, you need to delete the package from ConflictSolver and then reimport the package with its externally created transform.

After reimporting a package, it is generally a good practice to run conflict identification again.

Testing Microsoft App-V Packages

You can use ConflictSolver to test imported App-V packages against best practice rules, and to identify conflicts between App-V packages and between App-V packages and Windows Installer packages.

- Performing Best Practice Analysis of App-V Packages
- Performing Conflict Analysis of App-V Packages

Performing Best Practice Analysis of App-V Packages

Best Practice ACEs are rules that internally perform checks against the structure of an App-V package to enforce best practices. Best Practice ACEs are similar to Microsoft ICEs. Both Microsoft ICEs (custom actions written by Microsoft used during validation) and Best Practice ACEs are used to determine if a package is built according to standards.

When testing a package against a Best Practice ACE Rule, there is no target package involved. The source package is tested to see if it meets the standards defined in the ACEs.

Task: ***To perform Best Practice analysis of App-V packages:***

1. Select **Conflict Wizard** from the **Conflicts** menu. The **Welcome Panel** opens.
2. Click **Next**. The **Source Type Panel** opens.
3. Select **Internal Application Catalog package** and click **Next**. The **Source Package Panel** opens.
4. Select the App-V packages that you want to test and click **Next**. The **Choose Action** panel opens.

   Tip • *If you want to test a single App-V package, instead of launching the Conflict Wizard from the* ***Conflicts*** *menu, you could select the App-V package in the tree and then select* ***Conflict Wizard*** *from the context menu. The* ***Choose Action*** *panel of the Conflict Wizard would open with that App-V package already selected.*

5. Under **Rules** select **Evaluate source package against best practice rules**.

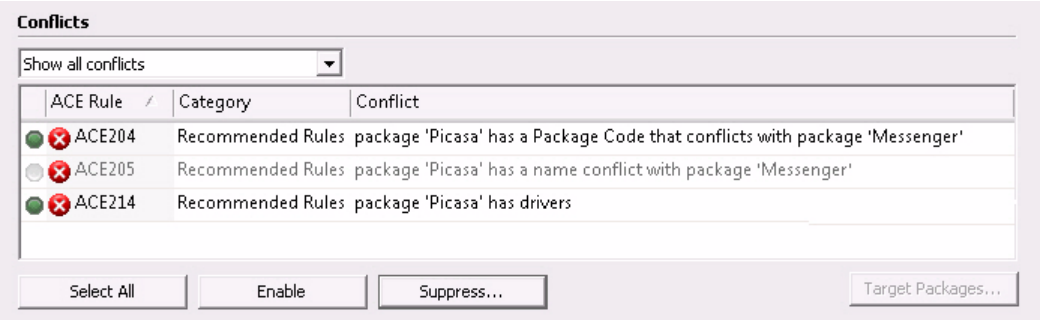
Tip • *If you wanted to perform both best practice and conflict analysis at the same time, you could select both options under* ***Rules****.*

6. Click **Next**. The **Best Practice Rules** panel opens.
7. Select the App-V **Recommended Rules** that you want to use. The following best practice rules are available:

ACE	Checks for ...
ACE201	Hardcoded targets in a shortcut table
ACE202	Hardcoded arguments in a shortcut table
ACE203	Hardcoded working directories in a shortcut table
ACE208	No shortcuts
ACE209	Shell extensions
ACE210	ClickOnce
ACE211	DLL surrogates
ACE212	Boot services
ACE213	OS integrated files
ACE214	Drivers
ACE216	Long .SFT filenames

8. Click **Next**. The **Summary** panels opens.
9. Review the options selected in the **Summary Panel** and click **Finish** to begin the testing process. Progress appears in the **Output** tab of the **Output Window**. When the testing is complete, all of the errors and warnings that were generated are listed in a table format in the **Conflicts** tab of the **Output Window**.
10. To view a detailed list of the errors and warnings that were generated, expand the **App-V** package node in the tree and select the **Conflicts** node to open the **Conflicts** view. The results are listed with icons to indicate whether it is an Error ( ) or a Warning (), the name of the ACE that generated it, and a brief description of what caused it to occur. You can sort by any column by clicking the appropriate column title.
11. If you do not want ConflictSolver to check for a particular ACE error or warning during subsequent analyses of a package, you can choose to suppress it by selecting it and clicking the **Suppress** button.

    Suppressed ACE errors are listed in gray, and are preceded by a white icon, while enabled ACE errors are preceded by a green icon:



12. If you want to fix any of the detected warnings or errors, open the package in the Virtual Package Editor.

Performing Conflict Analysis of App-V Packages

Conflict ACE rules are used to detect conflicts between an App-V source package and other App-V target packages or between an App-V source package and Windows Installer target packages.

Task: ***To perform conflict analysis of App-V packages:***

1. Select **Conflict Wizard** from the **Conflicts** menu. The **Welcome Panel** opens.
2. Click **Next**. The **Source Type Panel** opens.
3. Select **Internal Application Catalog package** and click **Next**. The **Source Package Panel** opens.
4. Select the App-V packages that you want to test and click **Next**. The **Choose Action** panel opens.

   Tip • *If you want to test a single App-V package, instead of launching the Conflict Wizard from the* ***Conflicts*** *menu, you could select the App-V package in the tree and then select* ***Conflict Wizard*** *from the context menu. The* ***Choose Action*** *panel of the Conflict Wizard would open with that App-V package already selected.*

5. Under **Rules** select **Detect conflicts between source package and other packages**.

   Tip • *If you wanted to perform both best practice and conflict analysis at the same time, you could select both options under* ***Rules****.*

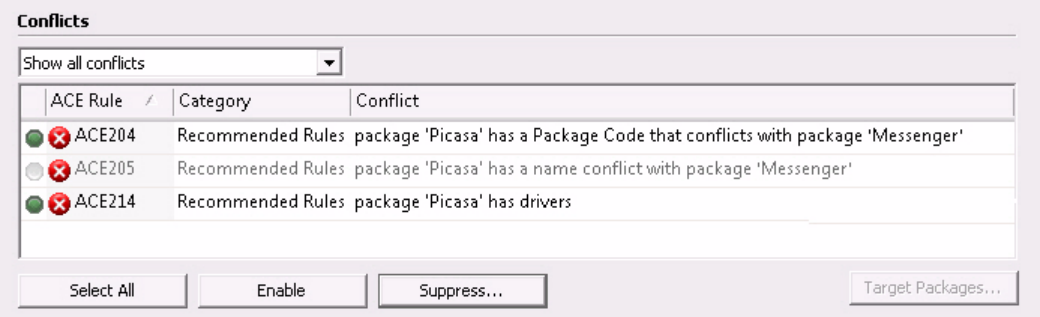
6. Click **Next**. The **Conflict Rules** panel opens.
7. Select the App-V **Recommended Rules** that you want to use. The following conflict rules are available:

ACE	Checks for ...
ACE200	Shortcut location conflicts
ACE204	Package Code conflicts
ACE205	Package Name conflicts
ACE206	File extension and PROGID conflicts
ACE207	App-V conflicts in root folder paths
ACE215	App-V shortcut name and version conflicts

8. Click **Next**. The **Target Information** panel opens.
9. Select all the App-V and Windows Installer packages you want to compare the source package against and click **Next**. The **Summary Panel** opens.

10. Review the options selected in the **Summary Panel** and click **Finish** to begin the conflict analysis. Progress appears in the **Output** tab of the **Output Window**. When the conflict analysis is complete, all of the errors and warnings that were generated are listed in a table format in the **Conflicts** tab of the **Output Window**.
11. To view a detailed list of the errors and warnings that were generated, expand the **App-V** package node in the tree and select the **Conflicts** node to open the **Conflicts** view. The results are listed with icons to indicate whether it is an Error (⊗) or a Warning (⚠), the name of the ACE that generated it, and a brief description of what caused it to occur. You can sort by any column by clicking the appropriate column title.
12. If you do not want ConflictSolver to check for a particular ACE error or warning during subsequent analyses of a package, you can choose to suppress it by selecting it and clicking the **Suppress** button.

    Suppressed ACE errors are listed in gray, and are preceded by a white icon, while enabled ACE errors are preceded by a green icon:



13. If you want to fix any of the detected warnings or errors, open the package in the Virtual Package Editor.

Generating Reports

Pre-Defined Reports

ConflictSolver includes several built-in reports to provide information on packages and conflicts identified, all accessible from the **Reports** menu. The included reports are:

Table 14-8 • Pre-Defined Reports List

Reports	Description
Package	Provides a report including the following information: • **Product Information**—Lists all information that appears on the Product View for a package. • **Conflict Information**—Lists all conflicts found during the most recent Conflict Wizard execution, or the most recent Import (if conflict identification was performed). • **Validation Information**—Lists all validation errors and warnings found during the most recent validation execution for that package. ***Note*** • *Validation information is persisted only for validations performed after the package was imported into the Application Catalog; validation results for validation performed during import are not persisted.* • **Extended Attributes**—Lists user-defined information that appears on the Extended Attributes View for a package.
File	The following information is displayed: • **Product Information**—Lists all information that appears on the Product View for a package. • **Files**—Lists all of the files that are included with the package. These files are also listed on the Files/Components View for that package.
Registry	The following information is displayed: • **Product Information**—Lists all information that appears on the Product View for a package. • **Registry**—Lists any registry entries created or changed by the package. These registry entries are also listed on the Registry View for that package.

Table 14-8 • Pre-Defined Reports List

Reports	Description
Crystal Reports	The following reports in Crystal Report format are available: • **Crystal Reports Conflicts**—Provides a report of all conflicts found during the most recent Conflict Wizard execution, or the most recent Import (if conflict identification was performed). This report is also available on a per-product basis by right-clicking on a product in the Product View. • **Crystal Reports Files**—Provides a comprehensive listing of all files in ConflictSolver, sorted on a per package and per component basis. This report is also available on a per-product basis by right-clicking on a product in the Product View. • **Crystal Reports Registry**—Provides a comprehensive listing of all registry entries in ConflictSolver, sorted on a per package and per component basis. This report is also available on a per-product basis by right-clicking on a product in the Product View. ***Note*** • *The three Crystal Reports selections will be listed on the* **Report** *menu if the* **Display RPT Reports** *option on the* **General** *tab of the ConflictSolver* **Options** *dialog box is selected. To suppress the display of the Crystal Reports selections, clear this option.*

User-Defined Reports

ConflictSolver supports custom user-defined reports; however, the responsibility of creating them is up to individual users. Reports must be in Crystal Reports format (.rpt) and placed in the folder specified in the General tab of the **Options** dialog box, available from the Tools menu. These reports appear under the Reports menu under Crystal Reports. See Creating Custom Reports for detailed instructions.

Report Center Reports

In AdminStudio Enterprise Edition, you can view reports using the AdminStudio Enterprise Server Report Center. Available reports include:

- Catalog Summary Report
- Package Reports
- Package Contents Reports
- Consolidated Package Information
- Group Summary Report

For more information, see Generating and Viewing Reports in Report Center.

Generating Package Reports

A Package Report lists general package information, extended attributes for that package, and any validation and conflict information that exists.

Task: ***To generate a Package Report:***

1. In the **Products View**, select the root node of the product that you want to generate a Package Report for.
2. Click **Package** on the **Reports** menu.

   or

   Click **Package** under **Reports** on the context menu.

   You are prompted to select a location where you want to save the report.
3. Select a location and click **Save**. The Package Report opens in a browser window displaying the following information:
   - **Product Information**—Lists all information that appears on the **Product View** for a package.
   - **Conflict Information**—Lists all conflicts found during the most recent **Conflict Wizard** execution, or the most recent import (if conflict identification was performed). These conflicts are also listed on the **Conflicts View** for that package.
   - **Validation Information**—Lists all validation errors and warnings found during the most recent validation execution for that package. These errors and warnings are also listed on the **Validation View** for that package.
   - **Extended Attributes**—Lists all user-defined information that appears on the **Extended Attributes View** for a package.
4. Click the section title to expand or contract that section of the report.
5. To print a report, click the **Print Page** link at the top of the report.

Generating File and Registry Reports

The Files Report lists of all of the files included with the package, and the Registry Report lists any registry entries created or changed by the package.

Task: ***To generate a Files or Registry report:***

1. In the **Products View**, select the root node of the product that you want to generate a report for.
2. Click **File** or **Registry** on the **Reports** menu.

   or

   Click **File** or **Registry** under **Reports** on the context menu.

   You are prompted to select a location where you want to save the report.
3. Select a location and click **Save**. The report opens in a browser window.

   For the Files Report, the following information is displayed:

- **Product Information**—Lists all information that appears on the Product View for a package.
- **Files**—Lists all of the files that are included with the package. These files are also listed on the **Files/ Components View** for that package.

For the Registry Report, the following information is displayed:

- **Product Information**—Lists all information that appears on the **Product View** for a package.
- **Registry**—Lists any registry entries created or changed by the package. These registry entries are also listed on the **Registry View** for that package.

4. Click the section title to expand or contract that section of the report.
5. To print a report, click the **Print Page** link at the top of the report.

Generating Reports in Crystal Reports Format

Three Crystal Reports are shipped with AdminStudio:

- **Conflicts**—Provides a report of all conflicts found during the most recent Conflict Wizard execution, or the most recent Import (if conflict identification was performed). This report is also available on a per-product basis by right-clicking on a product in the Product View.
- **Files**—Provides a comprehensive listing of all files in current Application Catalog, sorted on a per package and per component basis. This report is also available on a per-product basis by right-clicking on a product in the Product View.
- **Registry**—Provides a comprehensive listing of all registry entries in the current Application Catalog, sorted on a per package and per component basis. This report is also available on a per-product basis by right-clicking on a product in the Product View.

Task: ***To generate a Crystal Reports report:***

1. In the **Products View**, select the root node of the product that you want to generate a report for.
2. Select **Crystal Reports Conflicts**, **Crystal Reports Files**, or **Crystal Reports Registry** from the submenu.

   The three Crystal Reports selections will be listed on the **Report** menu if the **Display RPT Reports** option on the **General** tab of the ConflictSolver **Options** dialog box is selected.
3. To print a report, click the **Print Page** link at the top of the report.

Creating Custom Reports

ConflictSolver stores imported packages in a relational database. Crystal Reports can be used to quickly query this database and generate reports.

The following procedure explains how to create a custom report for ConflictSolver using AdminStudio and Crystal Reports 8. The resulting report allows you to search ConflictSolver and generate reports based on the presence of a specified INI file. You must have Crystal Reports 8 installed for this procedure.

This procedure contains the following steps:

- Step 1: Creating an ODBC File DSN
- Step 2: Creating the Base Report
- Step 3: Configuring Basic Settings
- Step 4: Creating Page Column Headings
- Step 5: Adding Database Fields
- Step 6: Adding Parameters
- Step 7: Creating Conditions for Filtering
- Step 8: Modifying the Footer
- Step 9: Using the Report in ConflictSolver

Step 1: Creating an ODBC File DSN

Task: ***To create an ODBC File DSN:***

1. From the **Windows Start** menu, point to **Programs**, then **Administrative Tools**, then click on **Data Sources (ODBC)**. This starts the Microsoft ODBC Data Source Administrator.
2. Click on the **File DSN** tab and click **Add**.
3. Select **Microsoft Access Driver (*.mdb)** from the drivers list and click **Next**.
4. Name the data source as `CSIniFileSearch` and click **Next**.
5. Click **Finish**.
6. Click **Select** and browse and select the ConflictSolver sample Access Application Catalog.
7. Click **OK**.
8. Click **OK** again.

Step 2: Creating the Base Report

Task: ***To create the base report:***

1. Start Crystal Reports.
2. Select **Blank Report** either from the **Welcome** dialog box or from the **New Report** dialog box.
3. Expand **CSIniFileSearch.dsn** under **ODBC** from the **Data Explorer** dialog box.
4. Select **IniFile** from the list of tables under **CSIniFileSearch.dsn** and click **Add**.
5. Select **Component** from the list of tables under **CSIniFileSearch.dsn** and click **Add**.
6. Select **Package** from the list of tables under `CSIniFileSearch.dsn` and click **Add**.
7. Click **Close**.
8. Crystal Reports will use its smart linking feature to link the selected tables. Delete the link between **IniFile** and **Package**.
9. Create a link between **Component** and **Package** table by dragging the **PackageCode** field from the **Component** table and dropping into the **PackageCode** field in the **Package** table.
10. Click **OK** on the **Visual Linking Expert** dialog box.
11. From the **Field Explorer** dialog box, click **Close**. You should now see a blank report in the design mode with header, details and footer sections.

Step 3: Configuring Basic Settings

Task: ***To configure basic settings:***

1. From the **File** menu, select **Printer Setup** and change the orientation from **Portrait** to **Landscape**.
2. From the **File** menu, select **Summary Info**.
3. Type `IniFile Search Report` in the **Title** text box, and click OK.
4. From the Insert menu, select **Special Field**.
5. Scroll down and select **Report Title** from the list and click on the **Insert to Report** toolbar button.
6. Click **Close**.

Table Name Prefixes for SQL Server Databases

If you are building a report from an SQL Server database, the names of the added tables may be prefixed by `[database_name].[user_name]`.

Delete these prefixes and just keep the table name, so that your report works even if you connect to other Application Catalogs.

Task: ***To delete prefixes:***

1. Click on **Database | Set Location**.
2. Click on each table.
3. Look at the Table text box name under the **Location** frame and make sure that the table contains just the table names.

Step 4: Creating Page Column Headings

Task: ***To create page column headings:***

1. Select **Text Object** from the Insert menu and drop the mouse on the left side inside the **Page Header** section.
2. Type `Product` in the text object.
3. Click on the above text object, press Ctrl+C to copy it, press Ctrl+V and drop the mouse adjacent to the first text object. Change the text from `Product` to `Component`.
4. Repeat the above step and add the columns named File Name, Path, INI Section, INI Key, INI Key Value, and Action.
5. If desired, you can right-click on the text objects and select **Change Font** to update the font.

Step 5: Adding Database Fields

Task: ***To add database fields:***

1. Select **Database Fields** from the **Insert** menu.
2. Expand the **Package** table node, press **Enter** on the **ProductName** field, and drop the mouse in the **Details** section under the **Product** page heading.
3. Expand the **Component** table, select **Component** field, press **Enter**, and drop the mouse in the **Details** section under the **Component** page heading.
4. Select the **IniFile** table and drop **FileName**, **FullPath**, **IniSection**, **IniKey**, **IniValue**, and **IniAction** fields.
5. Select all the fields in the detail section
6. Right-click on the fields and select **Format Object**.
7. Check the **Can Grow** box.

8. Save the report file under the following folder:

   AdminStudio Installation Directory\ConflictSolver\ Reports\Standard\MS

Step 6: Adding Parameters

Standard ConflictSolver reports require at least two parameters to be defined in the report file.

Task: ***To add parameters:***

1. From the **Insert** menu, select **Parameter** Field.
2. Click on **New** from the toolbar or press Ctrl+N.
3. Type the parameter name as `ProductName` and leave the type to default value of `String`.
4. Repeat this step and add one more parameter named **ProductVersion** of type `String`.
5. Click **Close**.

ConflictSolver automatically passes values for these parameters. In ConflictSolver when the report is selected from the main menu item, these parameters are passed as *; and when the report is selected by right clicking on the package, ConflictSolver passes the ProductName and ProductVersion for the selected package.

For this report, a prompt is needed so you can enter an INI file name, and based on that, show the filtered record. This requires another parameter.

Task: ***To add the third parameter:***

1. From the Insert menu, select **Parameter Field**.
2. Type the parameter name as `INIFileName`.
3. Set the type to **String**.
4. Type the following into the **Prompting Text** text box:

   ```
   Enter the INI File Name. Use * as a wildcard character.
   ```

5. Click **OK**.
6. Click **Close** to dismiss the **Field Explorer** dialog box.

Step 7: Creating Conditions for Filtering

Task: ***To create the condition to filter the records based on the input parameters (ProductName, ProductVersion, and INIFileName):***

1. From the **Report** menu, point to **Edit Selection Formula** and select **Record**.
2. Type the following text in the **Filter Criteria** text box:

```
{IniFile.FileName} like {?INIFileName}andif {?ProductName} = '*' then TRUEelse
    {Package.ProductName} = {?ProductName}andif {?ProductVersion} = '*' then TRUEelse
    {Package.ProductVersion} = {?ProductVersion}
```

   Crystal Reports uses the above selection formula while sending the SQL query to the back end database server. The above formula makes the WHERE clause in the SELECT database query, which is internally sent by Crystal Reports while viewing the report. The above formula also filters the Application Catalog records based on the IniFile.FileName, and if ProductName/ProductVersion are passed they are used, else only the IniFile.FileName is used in the selection criteria.

3. Click **Save** from the toolbar.

Step 8: Modifying the Footer

Task: ***To add the date/time and page number to the report footer:***

1. From the **Insert** menu, select **Special Field**.
2. Select **Page N of M** and drop it on the report within the **Report Footer** section.
3. Repeat for **Print Date** and **Print Time**.
4. Save the report.
5. Close Crystal Reports and start ConflictSolver.

Step 9: Using the Report in ConflictSolver

The Reports menu item in ConflictSolver is dynamically generated based on the .rpt report files present in the Reports folder. Start ConflictSolver, open the sample Application Catalog, click on Options from the Tools menu, and ensure that the above report is saved in the Reports folder as specified in the General tab.

If you select INIFile Search Report from the Reports menu, you can access the report you just created.

Task: ***To use the report:***

1. From the **Reports** menu, point to **Standard** and then select **INIFile Search Report**.
2. Type `system.ini` in the text box and click **OK**. A report is generated. You can use `*` as the wildcard character while typing the INI file name.

   Open some other Application Catalog for ConflictSolver, run the report, type the name of the file and verify the results by looking at the report. You can also right-click on the package in ConflictSolver and generate the IniFile Search report only for the selected package.

Application Conflict Evaluators (ACEs)

Application Conflict Evaluators (ACEs) are the rules that allow ConflictSolver to identify conflicts between:

- Windows Installer packages
- Windows Installer packages and operating system images (OS Snapshots)
- App-V packages
- App-V packages and Windows Installer packages

ConflictSolver can automatically solve some of the conflicts spelled out in these rules. When an automatic remedy is available, the associated rule used to fix the conflict is spelled out in a Conflict Application Resolution Definition (CARD).

This table lists each of the Best Practice ACE rules, and is grouped by conflict type:

Table 14-9 • Best Practice ACE Rules

Conflict Type	Associated Rules
Components	ACE04, ACE05, ACE06,
Merge Module Integrity	ACE26, ACE36
Recommended Rules	ACE25, ACE27, ACE28, ACE29, ACE31, ACE32, ACE33, ACE34, ACE35
App-V Recommended Rules	ACE201, ACE202, ACE203, ACE208, ACE209, ACE210, ACE211, ACE212, ACE213, ACE214, ACE216
Windows Terminal Server Compatibility	WTS01, WTS02, WTS03, WTS04, WTS05, WTS06, WTS07, WTS08, WTS09

This table lists each of the Conflict ACE rules, and is grouped by conflict type.

Table 14-10 • Conflict ACE Rules

Conflict Type	Associated Rules
Component	ACE02, ACE09, ACE30
File Extensions	ACE17
Files	ACE03, ACE07, ACE08, ACE12, ACE23
INI Files	ACE14, ACE21, ACE22
NT Services	ACE16
ODBC Resources	ACE15
Product Properties	ACE18, ACE19, ACE20
Registry	ACE10, ACE24
Shortcuts	ACE13
App-V Recommended Rules	ACE200, ACE204, ACE205, ACE206, ACE207, ACE215

User-Defined ACEs

In addition to these ACEs included with ConflictSolver, you can also create three types of user-defined ACEs to use when detecting conflicts: Source Only, Source and Target, and DLL-Based. See User-Defined ACEs for more information.

ACE Index

The following list provides links to each individual ACE. These ACEs are used to identify potential conflicts between installation packages.

Table 14-11 • ACE Index

ACE Rule	Rule Type	Conflict Type	Brief Description of ACE
ACE02	**Conflict**	Component	Checks to see if components in different packages that have matching ComponentIds also have identical destination paths.
ACE03	**Conflict**	Files	Checks to see if components in different packages that have matching ComponentIds also contain the same files.

Table 14-11 • ACE Index (cont.)

ACE Rule	Rule Type	Conflict Type	Brief Description of ACE
ACE04	**Best Practice**	Component	Checks to see if components that have NULL KeyPath values are valid by checking for an entry for the component in the CreateFolder table.
ACE05	**Best Practice**	Component	Checks for the existence of more than one executable (EXE, DLL, OCX, HLP, CHM, TLB, SYS, DRV) per component in a Windows Installer package.
ACE06	**Best Practice**	Component	Checks whether the executable module (EXE, DLL, OCX, HLP, CHM, TLB, SYS, or DRV) within the component is the key file.
ACE07	**Conflict**	Files	Checks for the existence of the same file in components with different ComponentIds.
ACE08	**Conflict**	Files	Identifies components with identical ComponentIds, and checks those components to see if the versions of the files in each component match.
ACE09	**Conflict**	Component	Checks to see if merge modules with identical ComponentIds are identical.
ACE10	**Conflict**	Registry	Checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds.
ACE12	**Conflict**	Files	Checks for components that contain files that can be replaced by one of the merge modules that you have imported into the Application Catalog database. (Before running ACE12, you should import all merge modules you are likely to use at your organization.)
ACE13	**Conflict**	Shortcuts	Checks for the existence of the same shortcut within different packages in components with different ComponentIds.
ACE14	**Conflict**	INI Files	Checks for the existence of components with different ComponentIds that modify the same INI file entry, such as the [File Name/Section/Key/Value] entry.
ACE15	**Conflict**	ODBC Resources	Checks for the existence of identical ODBC entries in components with different ComponentIds.
ACE16	**Conflict**	NT Services	Checks for the existence of identical NT Services in components with different ComponentIds.

Table 14-11 • ACE Index (cont.)

ACE Rule	Rule Type	Conflict Type	Brief Description of ACE
ACE17	**Conflict**	File Extensions	Checks for identical file Extension/Verb combinations in components with different ComponentIds.
ACE18	**Conflict**	Product Codes	Checks the Package Code to see if it is unique.
ACE19	**Conflict**	Product Codes	Checks the Product Code to see if it is unique.
ACE20	**Conflict**	Product Codes	Checks the Upgrade Code to see if it is unique.
ACE21	**Conflict**	INI Files	Checks entries in the IniFile table to see if they conflict with similar entries in the File table. The IniFile and File tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated.
ACE22	**Conflict**	INI Files	Checks file name/target directory pairs in the File table to see if they conflict with similar entries in the IniFile table. The IniFile and File tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated.
ACE23	**Conflict**	Files	Identifies file duplication between source and target packages. ACE23 checks to see if files with the same name and destination directory have the same size, version, and language when comparing a source package against a target package. If a file with the same name and destination directory is found in both the source and target packages, but the file has a different size, version, or language, ACE23 fails.
ACE24	**Conflict**	Registry	Checks to see if registry entries with the same registry hive, key, and value name have the same data type and value. If a registry entry with the same registry hive, key, and value name is found in both the source and target package, but the registry entry has a different data type or value, ACE24 fails.
ACE25	**Best Practice**	Recommended Rules	A check of the CustomAction table to identify any hard coded paths.
ACE26	**Best Practice**	Merge Module Integrity	A check of the Merge Modules in a package to confirm that they also exist in the Application Catalog.
ACE27	**Best Practice**	Recommended Rules	Checks to see if data exists in the DuplicateFile MSI table that is not being executed with an associated DuplicateFiles action.

Table 14-11 • ACE Index (cont.)

ACE Rule	Rule Type	Conflict Type	Brief Description of ACE
ACE28	**Best Practice**	Recommended Rules	A check of the Environment table to identify any hard coded paths.
ACE29	**Best Practice**	Recommended Rules	A check of the IniFile table to identify any hard coded paths.
ACE30	**Conflict**	Components	A rule to identify components with different ComponentIDs installing the same key file to the same directory. (A check for KeyPath conflicts across components.)
ACE31	**Best Practice**	Recommended Rules	Checks to see if data exists in the MoveFile MSI table that is not being executed with an associated MoveFiles action.
ACE32	**Best Practice**	Recommended Rules	A check of the Registry table to identify any hard coded paths.
ACE33	**Best Practice**	Recommended Rules	Checks to see if data exists in the RemoveFile MSI table that is not being executed with an associated RemoveFiles action.
ACE34	**Best Practice**	Recommended Rules	Checks to see if data exists in the RemoveIniFile MSI table that is not being executed with an associated RemoveIniFiles action.
ACE35	**Best Practice**	Recommended Rules	Checks to see if data exists in the RemoveRegistry MSI table that is not being executed with an associated RemoveRegistryValues action.
ACE36	**Best Practice**	Merge Module Integrity	A check of the required Merge Modules in a package to confirm that they also exist in the Application Catalog.
ACE200	**Conflicts** / **App-V**	App-V Recommended Rules	Checks to see if two or more packages contain a shortcut with the same display name and location.
ACE201	**Best Practices** / **App-V**	App-V Recommended Rules	Checks to see if a target in the package has a hard-coded path, such as C:\...\, which may not be present in a virtual environment.
ACE202	**Best Practices** / **App-V**	App-V Recommended Rules	Checks to see if a command-line argument for a target in the package includes a hard-coded path, such as C:\...\, which may not be present in a virtual environment.

Table 14-11 • ACE Index (cont.)

ACE Rule	Rule Type	Conflict Type	Brief Description of ACE
ACE203	**Best Practices** / **App-V**	App-V Recommended Rules	Checks to see if a working directory for a target in the package includes a hard-coded path, such as C:\...\, which may not be present in a virtual environment.
ACE204	**Conflicts** / **App-V**	App-V Recommended Rules	Checks to see if two or more packages have the same package GUID.
ACE205	**Conflicts** / **App-V**	App-V Recommended Rules	Checks to see if two or more packages have the same name.
ACE206	**Conflicts** / **App-V**	App-V Recommended Rules	Checks to see if two or more packages have support for the same file extension or progid.
ACE207	**Conflicts** / **App-V**	App-V Recommended Rules	Checks to see if two or more packages have the same long or short name for the root folder.
ACE208	**Best Practices** / **App-V**	App-V Recommended Rules	Checks for App-V packages that do not contain any shortcuts.
ACE209	**Best Practices** / **App-V**	App-V Recommended Rules	Checks App-V packages for shell extensions.
ACE210	**Best Practices** / **App-V**	App-V Recommended Rules	Checks App-V packages for ClickOnce.
ACE211	**Best Practices** / **App-V**	App-V Recommended Rules	Checks App-V packages for DLL surrogates.
ACE212	**Best Practices** / **App-V**	App-V Recommended Rules	Checks App-V packages for boot services.
ACE213	**Best Practices** / **App-V**	App-V Recommended Rules	Checks App-V packages for OS integrated files.

Table 14-11 • ACE Index (cont.)

ACE Rule	Rule Type	Conflict Type	Brief Description of ACE
ACE214	**Best Practices** / **App-V**	App-V Recommended Rules	Checks App-V packages for drivers.
ACE215	**Conflicts** / **App-V**	App-V Recommended Rules	Checks to see if an App-V package contains a shortcut that uses the same name and version as one in another package.
ACE216	**Best Practices** / **App-V**	App-V Recommended Rules	Checks to see if an App-V package's `.sft` file name is over 56 characters in length.
WTS01	**Best Practice**	Windows Terminal Server Compatibility	Checks to see if the ALLUSERS property is defined.
WTS02	**Best Practice**	Windows Terminal Server Compatibility	Checks for any user specific registry entries.
WTS03	**Best Practice**	Windows Terminal Server Compatibility	Checks for any per-user component destination.
WTS04	**Best Practice**	Windows Terminal Server Compatibility	Checks for any per-user ODBC data source.
WTS05	**Best Practice**	Windows Terminal Server Compatibility	Checks for any per-user environment settings.
WTS06	**Best Practice**	Windows Terminal Server Compatibility	Checks a package to make sure that all of the executables that it contains have the TSAWARE flag enabled.
WTS07	**Best Practice**	Windows Terminal Server Compatibility	Checks the LaunchCondition table of a package for instances of `TerminalServer` or `RemoteAdminTS` conditions.
WTS08	**Best Practice**	Windows Terminal Server Compatibility	Checks a package for 16-bit applications; 16-bit applications will not run on Windows 2008 R2 because it is a 64-bit platform.

Table 14-11 • ACE Index (cont.)

ACE Rule	Rule Type	Conflict Type	Brief Description of ACE
WTS09	**Best Practice**	Windows Terminal Server Compatibility	Checks a package for applications manifested to run as Administrator.

ACE02

ACE02 checks to see if components in different packages that have matching ComponentIds also have identical destination paths. The ACE02 Error String is displayed in the following format:

```
The destination [PATH1] for the component [COMPONENT1] in the package [PACKAGE1] conflicts with the
    destination for the component [COMPONENT1] in the package [PACKAGE2]. The correct destination
    should be [PATH2].
```

Summary

Table 14-12 • ACE02 Summary

Topic	Description
Conflict Type:	Conflict Rule / Components
Description:	Checks to see if components in different packages that have matching ComponentIds also have identical destination paths.
Result:	If components with the same ComponentId have different destination paths, ACE02 fails.
Resolution Type:	Automatic (CARD02)
Resolution:	The destination path of the component in the Source package is automatically set to match that of the component in the Target package.

ACE03

ACE03 checks to see if components in different packages that have matching ComponentIds also contain the same files. The ACE03 Error String is displayed in the following format:

```
The file(s) [FILENAME] in the component [COMPONENT1] in the package [PACKAGE1] are either new to or
    missing from the component [COMPONENT1] in the package [PACKAGE2].
```

Summary

Table 14-13 • ACE03 Summary

Topic	Description
Conflict Type:	Conflict Rule / Files
Description:	Checks to see if components in different packages that have matching ComponentIds also contain the same files.
Result:	If components with the same ComponentId do not contain the same files (either files are missing or they are different versions), ACE03 fails.
Resolution Type:	Manual
Resolution:	Use InstallShield Editor to create a Windows Installer Transform (.mst) file for the Source package MSI so that the component in the Source package contains the same files as the component in the Target package.

Cause

The ComponentIds for components in two different MSI packages are identical but either files are missing or different in those components.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the MSI package in InstallShield Editor.
2. Add or modify the components as necessary so that the file(s) is present in both components and the versions match in both MSI packages. To accomplish this:
   a. Open the **Components View** and find the component that needs to be modified.
   b. Expand the list under the component and select **Files**. A list of the files included with that component is displayed.
   c. To delete a file that is not present in the other component, select the file, right-click and choose **Delete** from the menu.
   d. If you need to add a new file to match the other component, right click anywhere in the **Files** list, choose **Add** from the menu, and add the correct file.
3. Select **Save As** from the **File** menu and save the changes as a Windows Installer Transform (.mst) file.
4. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the Conflict Wizard to check it against ACE03 again.

ACE04

ACE04 checks to see if components that have NULL KeyPath values are valid by checking for an entry for the component in the CreateFolder table. The ACE04 Error String is displayed in the following format:

```
The component [COMPONENT1] in the package [PACKAGE1] does not have a key file. Additionally, the
    directory/component pair is not listed in the CreateFolder table.
```

Summary

Table 14-14 • ACE04 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Components
Description:	Checks to see if components that have NULL KeyPath values are valid by checking for an entry for the component in the CreateFolder table.
Result:	If a component does not have a key file, and if the directory/component pair is not listed in the CreateFolder table, ACE04 fails.
Resolution Type:	Automatic (CARD04)
Resolution:	A CreateFolder entry is created for the component.

ACE05

ACE05 checks for the existence of more than one executable (EXE, DLL, OCX, HLP, CHM, TLB, SYS, DRV) per component in a Windows Installer package. The ACE05 Error String is displayed in the following format:

```
The component [COMPONENT1] in the package [PACKAGE1] has more than one executable module (*.chm,
    *.dll, *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb). The file [EXECUTABLE_MODULE_FILENAME] must
    be included as a separate component.
```

Summary

Table 14-15 • ACE05 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Components
Description:	Checks for the existence of more than one executable (EXE, DLL, OCX, HLP, CHM, TLB, SYS, DRV) per component in a Windows Installer package.
Result:	If more than one executable exists in a component, ACE05 fails.
Resolution Type:	Automatic (CARD05)

Table 14-15 • ACE05 Summary

Topic	Description
Resolution:	Modifies the component so that only one EXE or DLL exists, and it adds new components for remaining EXE, DLL, OCX, HLP, CHM, TLB, SYS, and DRV files.

ACE06

ACE06 checks whether the executable module (EXE, DLL, OCX, HLP, CHM, TLB, SYS, or DRV) within the component is the key file. The ACE06 Error String is displayed in the following format:

```
The component [COMPONENT1] in the package [PACKAGE1] does not have an executable module (*.chm,
    *.dll, *.drv, *.exe, *.hlp, *.ocx, *.sys, or *.tlb) as the key file. The current key file [NON-
    EXECUTABLE_FILENAME] can be replaced with [EXECUTABLE_MODULE_FILENAME].
```

Summary

Table 14-16 • ACE06 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Components
Description:	Checks whether the executable module (EXE, DLL, OCX, HLP, CHM, TLB, SYS, or DRV) within the component is the key file.
Result:	If the executable module is not the key file, ACE06 fails.
Resolution Type:	Automatic (CARD06)
Resolution:	The executable module is automatically made the key file.

ACE07

ACE07 checks for the existence of the same file in components with different ComponentIds. ACE07 reports four types of errors, depending on the various combinations of source and target files and whether or not they originated in a merge module:

Table 14-17 • Four Types of ACE07 Errors

Source File	Target File	Severity	CARD-Enabled	Conflict View Message	Output Message
Not Merge Module	**Merge Module**	Warning	No	The file [FILENAME] in the component [COMPONENTNAME] is identical to the merge module installed file in the component [COMPONENTNAME]. Confirm this error by running this package against ACE12.	The file [FILENAME] in the component [COMPONENTNAME] in the package [PACKAGENAME] is identical to the merge module installed file in the component [COMPONENTNAME] in the package [PACKAGENAME]. Confirm this error by running this package against ACE12.
Not Merge Module	**Not Merge Module**	Error	Yes	The file [FILENAME] in the component [COMPONENTNAME] is identical to a file in the component [COMPONENTNAME], but the components have different GUIDs.	The file [FILENAME] is identical in both the component [COMPONENTNAME] in the package [PACKAGENAME] and the component [COMPONENTNAME] in the package [PACKAGENAME], but the components have different GUIDs.

Table 14-17 • Four Types of ACE07 Errors

Source File	Target File	Severity	CARD-Enabled	Conflict View Message	Output Message
Merge Module	**Merge Module**	Error	No	The file [FILENAME] in the merge module installed component [COMPONENTNAME] is identical to a file in another merge module installed component [COMPONENTNAME], but the components have different GUIDs. Run ACE12 to determine which merge module is most appropriate to use.	The file [FILENAME] is identical and installed by merge modules in both the component [COMPONENTNAME] in the package [PACKAGENAME] and the component [COMPONENTNAME] in the package [PACKAGENAME], but the components have different GUIDs. Run ACE12 to determine which merge module is most appropriate to use.
Merge Module	**Not Merge Module**	Error	No	The file [FILENAME] in the merge module installed component [COMPONENTNAME] is identical to the file in the component [COMPONENTNAME]. Confirm this error by running the [PACKAGENAME] package against ACE12.	The file [FILENAME] in the merge module installed component [COMPONENTNAME] in the package %s is identical to the file in the component [COMPONENTNAME] in the package [PACKAGENAME]. Confirm this error by running the [PACKAGENAME] package against the ACE12.

Summary

Table 14-18 • ACE07 Summary

Topic	Description
Conflict Type:	Conflict Rule / Files
Description:	Checks for the existence of the same file in components with different ComponentIds.
Result:	ACE07 reports four types of errors, depending on the various combinations of source and target files and whether or not they originated in a merge module. (See table above.)
Resolution Type:	Automatic (CARD07) only if both the source and target files originated in a Merge Module. For all other errors, confirm the error by running the package against ACE12 and then, based upon the results, decide how to proceed.

Table 14-18 • ACE07 Summary

Topic	Description
Resolution:	The Source package ComponentId is changed to match the Target package Component Id.

ACE08

ACE08 identifies components with identical ComponentIds, and checks those components to see if the versions of the files in each component match. The ACE08 Warning String is displayed in the following format:

```
The file [FILENAME], version [VERSION_NUMBER1], in the component [COMPONENT1] conflicts with the
    same file, version [VERSION_NUMBER2], in the component [COMPONENT2].
```

Note • *ACE08 is a warning, not an error.*

Summary

Table 14-19 • ACE08 Summary

Topic	Description
Conflict Type:	Conflict Rule / Files
Description:	Identifies components with identical ComponentIds, and checks those components to see if the versions of the files in each component match.
Result:	If components with identical ComponentIds contain files with different versions, ACE08 fails.
Resolution Type:	Manual
Resolution:	Change the file versions to match those of the Target component, or change the file versions in the Source component.

Cause

Two components with identical ComponentIds in different packages have the same file with different versions.

Resolution

This conflict requires a manual resolution. Follow these steps:

1. Open the MSI package in InstallShield Editor.
2. Once the project is open, there are two options that can be used to resolve the issue:

   Option #1: Change the Component Code. To accomplish this:

a. Open the **Components View** and select the component that needs to be modified.

b. In the Component's **Property** grid, change the value in the **Component Code** field.

Option #2: Replace the files so that the versions match in both MSI packages. To accomplish this:

a. Open the **Components View** and find the component that needs to be modified.

a. Expand the listing under the component, and select **Files**. The list of files included with that component is displayed.

b. To delete a file with the wrong version, select the file, right-click and choose **Delete** from the menu.

c. To add the correct version of a file, right click anywhere in the **Files** list, choose **Add** from the menu, and add the new file with the correct version.

3. Select **Save As** on the **File** menu and save the changes as a Windows Installer Transform (.mst) file.

4. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE08 again.

ACE09

ACE09 checks to see if merge modules with identical ComponentIds are identical. The ACE09 Error String is displayed in the following format:

```
The merge module, [MERGE_MODULE_NAME], version [VERSION_NUMBER1], conflicts with the same merge
    module, version [VERSION_NUMBER2], in another package.
```

Note • *ACE09 is a warning, not an error.*

Summary

Table 14-20 • ACE09 Summary

Topic	Description
Conflict Type:	Conflict Rule / Components
Description:	Checks to see if merge modules with identical ComponentIds are identical.
Result:	If merge modules with identical ComponentIds are different, a warning is generated.
Resolution Type:	Manual
Resolution:	Obtain the latest version of the merge module and rebuild the MSI.

Cause

Two merge modules with identical ComponentIds but different versions are found in different packages.

Resolution

Task: ***To perform a manual resolution:***

1. Obtain the latest version of the merge module displayed in the ACE message. The merge module may be available for download at the vendor's web site.
2. Open the MSI package in **InstallShield Editor**.
3. Place the new merge module in one of the Merge Module Locations that are specified on the **Merge Modules** tab of the ConflictSolver **Options** dialog box. This allows the merge module to be displayed in the **Redistributables** view of the InstallShield Editor.
4. Navigate to the **Redistributables View** in the InstallShield Editor project. Deselect the old merge module causing the conflict and select the new merge module in its place.
5. Select **Save As** on the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
6. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE09 again.

ACE10

ACE10 checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds. The ACE10 Error String is displayed in the following format:

```
The registry entry [REGISTRY_ENTRY] [REGISTRY_KEY] in the component [COMPONENT1] in the package
    [PACKAGE1] conflicts with the same registry entry in the component [COMPONENT1] in the package
    [PACKAGE2].
```

Note • *ACE10 is a warning, not an error.*

Summary

Table 14-21 • ACE10 Summary

Topic	Description
Conflict Type:	Conflict Rule / Registry
Description:	Checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds.
Result:	If the same Root/Key/Name registry combination is in more than one component, a warning is generated.
Resolution Type:	Manual
Resolution:	Change the ComponentId of the Source component to match that of the Target component.

Cause

The same Root/Key/Name registry combination is found in more than one component.

Conditions When an ACE10 Error Can Be Ignored

ACE10 uses data from the MSI Registry table to check for identical registry Root/Key/Name combinations in different components. However, there may be situations in which ACE10 will report an error unnecessarily. MSI supports a grammar for the Registry table in which if the Value field is preceded or terminated by the sequence tilde '[~]', then the registry value will be appended or prepended, respectively, to the existing registry Value. This sort of operation may be perfectly acceptable if the applications in question are modifying a common registry key in a manner consistent with its purpose.

You will need to decide individually if an ACE10 error is valid, but consider checking the Registry Value field, as its contents may prove useful in helping you decide.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the MSI package in InstallShield Editor.
2. Once the project is open, navigate to the **Components View** and find the component that needs to be modified by referring to the component name displayed in the error message.
3. In the Component's **Property** grid, change the value in the **Component Code** field to match the Component Code (ComponentId) of the component in the other project.

   If it is unclear what the value should be changed to, follow these steps:

a. Open up the package that will not be modified in Direct Edit mode by selecting Direct Editor under **Additional Tools**. A list of the package tables is displayed.

b. Select the **Component** table and search for the **Component Name** displayed in the conflict.

c. Once the entry is found, take note of the value displayed in the **ComponentId** column. This is the Component Code that should be used for the component in the other package.

4. Select **Save As** from the **File** menu and save the changes as a Windows Installer Transform (.mst) file.
5. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE10 again.

ACE12

ACE12 checks for components that contain files that can be replaced by one of the merge modules that you have imported into the Application Catalog database. (Before running ACE12, you should import all merge modules you are likely to use at your organization into the Application Catalog database.)

ACE12 reports an Error if the file in question originated from a merge module and the merge module was found in the Application Catalog. If the merge module was not found in the Application Catalog, ACE12 reports a Warning:

Table 14-22 • ACE12 Output Summary

Package File	Severity	Output
Not from Merge Module	Error	The file [FILENAME] in the component [COMPONENT1] in the package [PACKAGE1] should be replaced with the merge module [MERGE_MODULE_NAME].
From Merge Module	Warning	The [FILENAME] file originating from the [ModuleID] Merge Module in package [PACKAGE1] is a candidate to be replaced with the [MERGE_MODULE_NAME]. However, the [ModuleID] Merge Module is not in the Application Catalog which makes proper evaluation impossible.

Summary

Table 14-23 • ACE12 Summary

Topic	Description
Conflict Type:	Conflict Rule / Files
Description:	Checks for components that contain files that can be replaced by one of the merge modules that you have imported into the Application Catalog database. (Before running ACE12, you should import all merge modules you are likely to use at your organization.)

Table 14-23 • ACE12 Summary

Topic	Description
Result:	ACE12 reports a Warning if the file in question originated from a merge module and the merge module was found in the Application Catalog. If the merge module was not found in the Application Catalog, ACE12 reports an Error.
Resolution Type:	Manual
Resolution:	Rebuild the package using the merge module rather than the duplicate files.

Cause

A component within one package contains a file that is found in a merge module of another package. Using merge modules is always preferable as a way to install files in a consistent fashion.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the MSI package in InstallShield Editor.
2. Once the project is open, navigate to the **Components View**, and find the component that needs to be modified by referring to the component name displayed in the error message.
3. Expand the list under the component and select **Files**. A list of the files included with that component is displayed.
4. Select the file that was displayed in the error message, right-click, and select **Delete** from the menu.
5. Take note of the feature that the component is associated with and then navigate to the **Redistributables View**.
6. In the **Redistributables View**, select the appropriate merge module.
7. After selecting the merge module, move to the **Conditional Installation View** on the right side and select the feature that was mentioned earlier.
8. Rebuild the package.
9. Open ConflictSolver and reimport this package into your Application Catalog, and then use the Conflict Wizard to check it against ACE12 again.

ACE13

ACE13 checks for the existence of the same shortcut within different packages in components with different ComponentIds. The ACE13 Error String is displayed in the following format:

```
The shortcut [SHORTCUT_NAME] in the component [COMPONENT1] in the package [PACKAGE1] conflicts with
    the same shortcut in the component [COMPONENT2] in the package [PACKAGE2].
```

Summary

Table 14-24 • ACE13 Summary

Topic	Description
Conflict Type:	Conflict Rule / Shortcuts
Description:	Checks for the existence of the same shortcut within different packages in components with different ComponentIds.
Result:	If the same shortcut exists in more than one component, ACE13 fails.
Resolution Type:	Manual
Resolution:	Change the ComponentId of the Source component to match the Component Id of the Target component.

Cause

The same shortcut is found in components with different ComponentIDs in two different packages.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the MSI package in InstallShield Editor.
2. Once the project is open, navigate to the **Components View**, and find the component that needs to be modified by referring to the component name displayed in the error message.
3. In the Component's **Property** grid, change the value in the **Component Code** field to match the ComponentId of the component in the other project.

   If it is unclear what the value should be changed to, follow these steps:

   a. Open up the package that will not be modified in Direct Edit mode by selecting **Open** on the **File** menu, and then selecting **Direct Editor** under **Additional Tools**. A list of the package tables is displayed.

   b. Select the **Component** table and search for the **Component Name** displayed in the conflict.

c. Once the entry is found, take note of the value displayed under the **ComponentId** column. This is the **Component Code** that should be used for the component in the other package.

4. Select **Save As** on the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
5. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE13 again.

ACE14

ACE14 checks for the existence of components with different ComponentIds that modify the same INI file entry, such as the [File Name/Section/Key/Value] entry. If this condition is found, ACE14 returns the following Warning:

```
The INI file entry [INI_FILE_ENTRY] in the file [INI_FILE_NAME] in the component [COMPONENT1] in the
    package [PACKAGE1] conflicts with the same INI file entry in the component [COMPONENT1] in the
    package [PACKAGE2].
```

Summary

Table 14-25 • ACE14 Summary

Topic	Description
Conflict Type:	Conflict Rule / INI Files
Description:	Checks for the existence of components with different ComponentIds that modify the same INI file entry, such as [File Name/Section/Key/Value].
Result:	If the same INI file entry is modified by different components, ACE14 fails.
Resolution Type:	Manual using the instructions under Resolution below
Resolution:	Change the ComponentId of the Source component to match the ComponentId of the Target component.

Cause

The same INI file is found in components with different ComponentIds in two different packages.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the MSI package in InstallShield Editor.
2. Once the project is open, navigate to the **Components View**, and find the component that needs to be modified by referring to the component name displayed in the Warning.

3. In the Component's **Property** grid, change the value in the **Component Code** field to match the Component Code (ComponentId) of the component in the other project.

   If it is unclear what the value should be changed to, follow these steps:

   a. Open up the package that will not be modified in Direct Edit mode by selecting **Open** on the **File** menu, and then selecting **Direct Editor** under **Additional Tools**. A list of the package tables is displayed.

   b. Select the **Component** table and search for the **Component Name** displayed in the conflict.

   c. Once the entry is found, take note of the value displayed under the **ComponentId** column. This is the Component Code that should be used for the component in the other package.

4. Select **Save As** on the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.

5. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE14 again.

ACE15

ACE15 checks for the existence of identical ODBC entries in components with different ComponentIds. The ACE15 Error String is displayed in the following format:

```
The ODBC entry [ODBC_ENTRY] in the component [COMPONENT1] in the package [PACKAGE1] conflicts with
    the same ODBC entry in the component [COMPONENT1] in the package [PACKAGE2].
```

Summary

Table 14-26 • ACE15 Summary

Topic	Description
Conflict Type:	Conflict Rule / ODBC Resources
Description:	Checks for the existence of identical ODBC entries in components with different ComponentIds.
Result:	If identical ODBC entries exist in components with different ComponentIds, ACE15 fails.
Resolution Type:	Automatic (CARD15)
Resolution:	ConflictSolver changes the Source ComponentId to match the Target ComponentId.

ACE16

ACE16 checks for the existence of identical NT Services in components with different ComponentIds. The ACE16 Error String is displayed in the following format:

```
The NT service [SERVICE1] in the component [COMPONENT1] in the package [PACKAGE1] conflicts with the
    same NT Service in the component [COMPONENT1] in the package [PACKAGE2].
```

Summary

Table 14-27 • ACE16 Summary

Topic	Description
Conflict Type:	Conflict Rule / NT Services
Description:	Checks for the existence of identical NT Services in components with different ComponentIds.
Result:	If identical NT Services exist within different components, ACE16 fails.
Resolution Type:	Manual
Resolution:	Change the ComponentId of the Source component to match the ComponentId of the Target component.

Cause

The same NT Service is found in components with different ComponentIDs in two different packages.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the package in InstallShield Editor.
2. Once the project is open, navigate to the **Components View**, and find the component that needs to be modified by referring to the component name displayed in the error message.
3. In the Component's **Property** grid, change the value in the **Component Code** field to match the Component Code (ComponentId) of the component in the other project.

   If it is unclear what the Component Code should be changed to, follow these steps:

   a. Open up the package that will not be modified in Direct Edit mode by selecting **Open** on the **File** menu, and then selecting **Direct Editor** under **Additional Tools**. A list of the package tables is displayed.

   b. Select the **Component** table and search for the **Component Name** displayed in the conflict.

c. Once the entry is found, take note of the value displayed in the **ComponentId** column. This is the Component Code that should be used for the component in the other package.

4. Select **Save As** from the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
5. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE16 again.

ACE17

ACE17 checks for identical file Extension/Verb combinations in components with different ComponentIds. The ACE17 Error String is displayed in the following format:

```
The verb [VERB_NAME] in extension [EXTENSION_NAME] in the component [COMPONENT1] in the package
    [PACKAGE1] conflicts with the same verb & extension in the component [COMPONENT1] in the package
    [PACKAGE2].
```

Summary

Table 14-28 • ACE17 Summary

Topic	Description
Conflict Type:	Conflict Rule / File Extensions
Description:	Checks for identical file Extension/Verb combinations in components with different ComponentIds.
Result:	If identical file Extension/Verb combinations exist in components with different ComponentIds, ACE17 fails.
Resolution Type:	Manual
Resolution:	Change the ComponentId of the Source component to match the ComponentId of the Target component.

Cause

The same Extension/Verb combination is found in components with different ComponentIds in two different packages.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the transform file or MSI package in InstallShield Editor.
2. Once the project is open, navigate to the **Components View**, and find the component that needs to be modified by referring to the component name displayed in the error message.
3. In the Component's **Property** grid, change the value in the **Component Code** field to match the Component Code (ComponentId) of the component in the other project.

   If it is unclear what the **Component Code** should be changed to, follow these steps:

   a. Open up the package that will not be modified in Direct Edit mode by selecting **Open** on the **File** menu, and then selecting **Direct Editor** under **Additional Tools**. A list of the package tables is displayed.

   b. Select the **Component** table and search for the Component Name displayed in the conflict.

   c. Once the entry is found, take note of the value displayed in the **ComponentId** column. This is the **Component Code** that should be used for the component in the other package.
4. Select **Save As** from the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
5. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE17 again.

ACE18

ACE18 checks the Package Code to see if it is unique. The ACE18 Error String is displayed in the following format:

```
The package code [PACKAGE_CODE] in the package [PACKAGE1] is the same as the package code in the
    package [PACKAGE2].
```

Summary

Table 14-29 • ACE18 Summary

Topic	Description
Conflict Type:	Conflict Rule / Product Properties
Description:	Checks the Package Code to see if it is unique.
Result:	If the Package Code is identical to any other Package Code in ConflictSolver, ACE18 fails.
Resolution Type:	Automatic (CARD18)
Resolution:	A new, unique Package Code is automatically generated to replace the duplicate code.

***Note** • ACE18 can only be resolved automatically when applying changes directly to the original Windows Installer package. It cannot be used when resolving using a transform.*

ACE19

ACE19 checks the Product Code to see if it is unique. The ACE19 Error String is displayed in the following format:

```
The product code [PRODUCT_CODE] in the package [PACKAGE1] is the same as the product code in the
    package [PACKAGE2].
```

Summary

Table 14-30 • ACE19 Summary

Topic	Description
Conflict Type:	Conflict Rule / Product Properties
Description:	Checks the Product Code to see if it is unique.
Result:	If the Product Code is identical to any other Product Code in ConflictSolver, ACE19 fails.
Resolution Type:	Automatic (CARD19)
Resolution:	A new, unique Product Code is automatically generated to replace the duplicate code.

ACE20

ACE20 checks the Upgrade Code to see if it is unique. The ACE20 Error String is displayed in the following format:

```
The upgrade code [UPGRADE_CODE] in the package [PACKAGE1] is the same as the upgrade code in the
    package [PACKAGE2].
```

Summary

Table 14-31 • ACE20 Summary

Topic	Description
Conflict Type:	Conflict Rule / Product Properties
Description:	Checks the Upgrade Code to see if it is unique.
Result:	If the Upgrade Code is not unique, ACE20 fails.
Resolution Type:	Automatic (CARD20)

Table 14-31 • ACE20 Summary

Topic	Description
Resolution:	A new, unique Upgrade Code is automatically generated to replace the duplicate code.

ACE21

ACE21 checks entries in the `IniFile` table to see if they conflict with similar entries in the `File` table. The `IniFile` and `File` tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated. The ACE21 Error String is displayed in the following format:

```
The INI file [INI_FILENAME] installed using the IniFile table to the destination [PATH_NAME] in the
    package [PACKAGE1.MSI] conflicts with the same file in the File table in the package [PACKAGE2].
```

Note • *If performing conflict analysis with multiple source packages, and if the* **Check conflicts across all source and target packages** *option on the* **Rules** *tab of the* **Options** *dialog box is selected, both ACE21 and ACE22 are automatically evaluated when either one is selected. If only one of them is evaluated, then there is the possibility that conflicts will not be detected.*

Summary

Table 14-32 • ACE21 Summary

Topic	Description
Conflict Type:	Conflict Rule / INI Files
Description:	Checks entries in the IniFile table to see if they conflict with similar entries in the File table. The IniFile and File tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated.
Result:	If an entry in the IniFile table of the first MSI package duplicates a file name in a component of the File table of the second MSI package, and both are set to be installed to the same destination, ACE21 fails.
Resolution Type:	Manual
Resolution:	Ensure identically named INI files with identical destinations are identical in both the File and IniFile tables. This may involve adding or deleting sections or changing values in the INI files.

Cause

An entry in the IniFile table of the first MSI package duplicates a file name in a component in the File table of the second MSI package and both files are set to be installed to the same destination.

Resolution

This conflict requires a manual resolution.

Task: ***To perform a manual resolution:***

1. In InstallShield Editor, open a transform file or MSI package that has an entry in the IniFile table.
2. Once the project is open, navigate to the **INI Files Changes View**, and find the INI file entry causing the conflict by referring to the INI file name displayed in the error message.
3. Analyze the sections, keywords, and values in this INI file and compare them to the INI file in the other MSI package. Edit the sections, keywords, and values in this INI file so that they match those of the second INI file. This may require adding and/or deleting data from the **INI File Changes View**.

   In order to retrieve the INI file from the other MSI package, you may need to extract it from the MSI file itself or a cabinet file if the files are not uncompressed on the source media.
4. Select **Save As** from the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
5. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the Conflict Wizard to check it against ACE21 again.

ACE22

ACE22 checks file name/target directory pairs in the File table to see if they conflict with similar entries in the `IniFile` table. The `IniFile` and File tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated. The ACE22 Error String is displayed in the following format:

```
The INI file [INI_FILENAME] installed using the File table to the destination [PATH_NAME] in the
    package [PACKAGE3.MSI] conflicts with the same file in the IniFile table in the package
    [PACKAGE4].
```

Note • *If performing conflict analysis with multiple source packages, and if the* **Check conflicts across all source and target packages** *option on the* **Rules** *tab of the* **Options** *dialog box is selected, both ACE21 and ACE22 are automatically evaluated when either one is selected. If only one of them is evaluated, then there is the possibility that conflicts will not be detected.*

Summary

Table 14-33 • ACE22 Summary

Topic	Description
Conflict Type:	Conflict Rule / INI Files

Table 14-33 • ACE22 Summary

Topic	Description
Description:	Checks file name/target directory pairs in the File table to see if they conflict with similar entries in the IniFile table. The IniFile and File tables can change the same physical file. As a result, this ACE identifies these duplicate file changes so that they can be evaluated.
Result:	If a file name/target directory pair in a component in the File table is also listed as an IniFile table entry, ACE22 fails.
Resolution Type:	Manual
Resolution:	Ensure identically named INI files with identical destinations are identical in both the File and IniFile tables. This may involve adding or deleting sections or changing values in the INI files.

Cause

An INI file in a component of the first MSI package duplicates an entry in the IniFile table of the second MSI package and both files are set to be installed to the same destination.

Resolution

This conflict requires a manual resolution.

Task: ***To perform a manual resolution:***

1. In InstallShield Editor, open a transform file or MSI package that has an entry in the IniFile table.
2. Once the project is open, navigate to the **INI Files Changes View**, and find the INI file entry causing the conflict by referring to the INI file name displayed in the error message.
3. Analyze the sections, keywords, and values in this INI file and compare them to the INI file in the other MSI package. Edit the sections, keywords, and values in this INI file so that they match those of the second INI file. This may require adding and/or deleting data from the **INI File Changes View**.

   In order to retrieve the INI file from the other MSI package, you may need to extract it from the MSI file itself or a cabinet file if the files are not uncompressed on the source media.
4. Select **Save As** from the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
5. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE22 again.

ACE23

ACE23 identifies file duplication between source and target packages. ACE23 checks to see if files with the same name and destination directory have the same size, version, and language when comparing a source package against a target package. If a file with the same name and destination directory is found in both the source and target packages, but the file has a different size, version, or language, ACE23 fails.

The ACE23 Error String is displayed in the following format:

```
The file [EXECUTABLE_FILENAME, SIZE, VERSION, LANGUAGE] installed to location [PATHNAME] by
    component [COMPONENT1] in Package [PACKAGE1] conflicts with the same file in component
    [COMPONENT2] in Package [PACKAGE2].
```

Note • *ACE23 can be used in conflict analysis when MSI Package, OS Snapshot File, or Marimba NCP File data types are selected as either source or target packages. However, if you specify only Marimba NCP or OS Snapshot files as your source packages, ACE23 and ACE24 will be the only ACEs that you can select on the Summary Panel of the Conflict Wizard. All other ACEs will be unavailable.*

Summary

Table 14-34 • ACE23 Summary

Topic	Description
Conflict Type:	Conflict Rule / Files
Description:	Checks to see if files with the same name and destination directory have the same size, version, and language.
Result:	If a file with the same name and destination directory is found in both the source and target package (and, in the case of an MSI package comparison, the packages have different ComponentId values), but the file has a different size, version, or language, ACE23 fails. If the Source and Target packages are MSIs and have the same Component Id value, no error is reported.
Resolution Type:	Manual
Resolution:	Investigate and decide which file has precedence. If the source file has precedence over the target file, remove the target file.

Cause

A file with the same name and destination directory is found in both the source and target package, but the file has a different size, version, or language.

Resolution

This conflict requires a manual resolution. Use ONE of the following solutions:

Task: ***Replace the file in the source package with the same file found in the target package***

1. Retrieve a copy of the file from the operating system in which the target package was taken.
2. Open a transform file or MSI package in InstallShield Editor.
3. Once the project is open, navigate to the **Files and Folders** view, and find the file causing the conflict. Replace that file with the one retrieved from the operating system.
4. Select **Save As** from the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
5. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the Conflict Wizard to check it against ACE23 again.

Task: ***Change the destination of the component that installs the file in the MSI package***

1. Open a transform file or MSI package in InstallShield Editor.
2. Once the project is open, navigate to the **Components** view, and select the component that contains the file causing the conflict.

   In order to quickly find the component name, go to the File Table via the Direct Editor view and search for the file name. Then, check the component column for the component name.
3. Select the **Destination** field of the component from the **Property** grid and change it to a new destination.

   Changing the destination may cause the application to break. Before changing the destination, verify that the application will still work using the new destination.
4. Select **Save As** from the **File** menu and save the changes as a **Windows Installer Transform** (**.mst**) file.
5. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE23 again.

ACE24

ACE 24 checks to see if registry entries with the same registry hive, key, and value name have the same data type and value. If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target package, but the registry entry has a different data type or value, ACE24 fails.

The ACE24 Error String is displayed in the following format:

```
The registry entry [REGISTRY_ENTRY] in [PACKAGE1] conflicts with the same registry entry in the
    [COMPONENT] in [PACKAGE2].
```

Note • *ACE24 can be used in conflict analysis when MSI Package, OS Snapshot File, or Marimba NCP File data types are selected as either source or target packages. However, if you specify only Marimba NCP Files as your source packages, ACE23 and ACE24 will be the only ACEs that you can select on the Conflict Rules Panel of the Conflict Wizard. All other ACEs will be unavailable.*

Summary

Table 14-35 • ACE24 Summary

Topic	Description
Conflict Type:	Conflict Rule / Registry
Description:	Checks to see if registry entries with the same registry hive, key, and value name have the same data type and value.
Result:	If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target packages, but the registry entry has a different data type or value, ACE24 fails.
Resolution Type:	Manual
Resolution:	Investigate and decide which registry entry has precedence. If the source registry entry has precedence over the target registry entry, remove the target registry entry.

Cause

A registry entry with the same registry hive, key, and value name in a package is found in both the source and target package, but the registry entry has a different data type or value.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open a transform file or MSI package in InstallShield Editor.
2. Once the project is open, navigate to the **Registry** view, and find the registry value causing the conflict. Replace that value with the same registry value and data type found in the target package.
3. Select **Save As** from the **File** menu and save the changes as a **Windows Installer Transform (.mst)** file.
4. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the **Conflict Wizard** to check it against ACE24 again.

ACE25

ACE25 checks the entries in the `CustomAction` table to identify any hard coded paths. The ACE25 Warning String is displayed in the following format:

```
The [CUSTOM_ACTION_NAME] custom action has a hard coded directory path of [TARGET_PATH] in its Target
    field.
```

Summary

Table 14-36 • ACE25 Summary

Topic	Description
Conflict Type:	Best Practices / Recommended Rules
Description:	Checks the entries in the package's CustomAction table to identify any hard coded paths in the Target field.
Result:	If a package has a hard coded path in the Target field of the CustomAction table, a warning is generated.
Resolution Type:	Manual
Resolution:	Open the package file in InstallShield Editor and change the hard coded paths to relative paths in the CustomAction table.

Resolution

Open the package file in InstallShield Editor, and in the Target field of the CustomAction table, change any hard coded paths to relative paths.

ACE26

ACE26 checks the Merge Modules in a package to confirm that they also exist in the Application Catalog.The ACE26 Warning String is displayed in the following format:

```
The [MERGE_MODULE_NAME], version [VERSION] Merge Module is included with this package and yet not
    imported into the Application Catalog. It is recommended that all Merge Modules be imported into
    the Application Catalog.
```

Note • *ACE26 and ACE36, optional Best Practice ACEs, both check for conflicts with Merge Modules. ACE26 checks merge modules listed in the* ***ModuleSignature*** *table while ACE36 checks the* ***ModuleDependency*** *table.*

These ACEs are provided to encourage you to import Merge Modules into the Application Catalog and, by doing so, improve the effectiveness of ACE12, which checks for components that contain files that could be replaced by one of the imported Merge Modules.

Summary

Table 14-37 • ACE26 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Merge Module Integrity
Description:	Checks the Merge Modules in a package to confirm that they have been imported into the Application Catalog.
Result:	If a package refers to a Merge Module that does not exist in the Application Catalog, a warning is generated.
Resolution Type:	Manual
Resolution:	Import the missing Merge Module into the Application Catalog.

Resolution

Open the package file in Application Manager and import the identified Merge Module into the Application Catalog.

ACE27

The DuplicateFileData ACE checks to see if data exists in the DuplicateFile MSI table that is not being executed with an associated DuplicateFiles action.

```
The package contains data ([FileKey]) in the 'DuplicateFile' table but not the necessary actions to
    use this data. You should consider whether a 'DuplicateFiles' or 'RemoveDuplicateFiles' action is
    needed for your 'InstallExecuteSequence' table.
```

Summary

Table 14-38 • ACE27 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks to see if data exists in the DuplicateFile MSI table that is not being executed with an associated DuplicateFiles action.
Result:	If a package contains data in the DuplicateFile table but not the necessary actions to use this data, a warning is generated.
Resolution Type:	Manual

Table 14-38 • ACE27 Summary

Topic	Description
Resolution:	Consider whether a DuplicateFiles or RemoveDuplicateFiles action is needed for your InstallExecuteSequence table. If so, open the MSI file in InstallShield and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the DuplicateFile table.

Resolution

Open the MSI file in InstallShield and add a DuplicateFiles or RemoveDuplicateFiles action to the InstallExecuteSequence table, or remove the unused data from the DuplicateFile table.

ACE28

ACE28 checks the entries of the Environment table to identify any hard coded paths. The ACE28 Warning String is displayed in the following format:

```
The [ENVIRONMENT_TABLE_NAME] Environment table entry has a hard coded directory path of
    [DIRECTORY_PATH] in its Value field.
```

Summary

Table 14-39 • ACE28 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks the Value field of the package's Environment table to identify any hard coded paths.
Result:	If any hard coded paths exist in the Value field of a package's Environment table, a warning is generated.
Resolution Type:	Manual
Resolution:	Open the package file in InstallShield Editor and change any hard coded paths in the Value field of the Environment table to relative paths.

Resolution

Open the package file in InstallShield Editor and change any hard coded paths in the Value field of the Environment table to relative paths.

ACE29

ACE29 checks the entries in the `IniFile` table to identify any hard coded paths. The ACE29 Warning String is displayed in the following format:

```
The [PATH_NAME]\[INI_FILE_NAME] INI file has a hard coded directory path of [HARD_CODED_PATH] in its
    Value field.
```

Summary

Table 14-40 • ACE29 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks the Value field of a package's IniFile table to identify any hard coded paths.
Result:	If the Value field of a package's IniFile table has any hard coded paths, a warning is generated.
Resolution Type:	Manual
Resolution:	Open the package file in InstallShield Editor and change any hard coded paths in the Value field of the IniFile table to relative paths.

Resolution

Open the package file in InstallShield Editor and change any hard coded paths in the Value field of the IniFile table to relative paths.

ACE30

The ACE30 is a check for KeyPath conflicts across components. It identifies components with different ComponentIDs installing the same key file to the same directory. The ACE30 Error String is displayed in the following format:

```
The [SOURCE_COMPONENT_NAME] component in [SOURCE_PRODUCT_NAME] is installing the
    [SOURCE_PATH_NAME]\[SOURCE_KEYPATH_FILE] file which is also being installed by
    [TARGET_PRODUCT_NAME] with a different ComponentId.
```

More precisely, ACE30 checks to determine if different components are installing identically-named key files to the same directory.

Note • *The name of the target product is not displayed when viewing this ACE Error String in the Conflicts View.*

The ACE executes the following query to check to see if the same key file is being installed with different ComponentIds:

```
Source.ComponentId <> Target.ComponentId AND Source.csFullPath = Target.csFullPath AND
    Source.KeyPath = Target.KeyPath
```

Summary

Table 14-41 • ACE30 Summary

Topic	Description
Conflict Type:	Conflict Rule / Components
Description:	Checks for KeyPath conflicts across components. It identifies components with different ComponentIDs installing the same key file to the same directory.
Result:	If components with different ComponentIDs installing the same key file to the same directory is identified, a warning is generated.
Resolution Type:	Manual
Result:	If a product component is installing a KeyPath file which is also being installed by a product with a different Component Id, the KeyPath ACE fails.

ACE31

The MoveFileData ACE checks to see if data exists in the MoveFile MSI table that is not being executed with an associated MoveFiles action.

```
The package contains data ([FileKey]) in the 'MoveFile' table but not the necessary actions to use
    this data. You should consider whether a 'MoveFiles' action is needed for your
    'InstallExecuteSequence' table.
```

Summary

Table 14-42 • ACE31 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks to see if data exists in the MoveFile MSI table that is not being executed with an associated MoveFiles action.
Result:	If a package contains data in the MoveFile table but not the necessary actions to use this data, a warning is generated.
Resolution Type:	Manual

Table 14-42 • ACE31 Summary

Topic	Description
Resolution:	Consider whether a MoveFiles action is needed for your InstallExecuteSequence table. If so, open the MSI file in InstallShield and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the MoveFile table.

Resolution

Open the MSI file in InstallShield and add a MoveFiles action to the InstallExecuteSequence table, or remove the unused data from the MoveFile table.

ACE32

ACE32 checks the entries of the Registry table to identify any hard coded paths. The ACE32 Warning String is displayed in the following format:

```
The [REGISTRY_TABLE_ENTRY_NAME] Registry table entry has a hard coded directory path of
    [DIRECTORY_PATH_NAME] in its Value field.
```

Summary

Table 14-43 • ACE32 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks entries in the Registry table to identify any hard coded paths in its Value field.
Result:	If there are any hard coded paths in the Value field of the package's Registry table, a warning is generated.
Resolution Type:	Manual
Resolution:	Open the package file in InstallShield Editor and change any hard coded paths in the Value field of the Registry table to relative paths.

Resolution

Open the package file in InstallShield Editor and change any hard coded paths in the Value field of the Registry table to relative paths.

ACE33

The RemoveFileData ACE checks to see if data exists in the RemoveFile MSI table that is not being executed with an associated RemoveFiles action.

```
The package contains data ([FileKey]) in the 'RemoveFile' table but not the necessary actions to use
    this data. You should consider whether a 'RemoveFiles' action is needed for your
    'InstallExecuteSequence' table.
```

Summary

Table 14-44 • ACE33 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks to see if data exists in the RemoveFile MSI table that is not being executed with an associated RemoveFiles action.
Result:	If a package contains data in the RemoveFile table but not the necessary actions to use this data, a warning is generated.
Resolution Type:	Manual
Resolution:	Consider whether a RemoveFiles action is needed for your InstallExecuteSequence table. If so, open the MSI file in InstallShield and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the RemoveFile table.

Resolution

Open the MSI file in InstallShield and add a RemoveFiles action to the InstallExecuteSequence table, or remove the unused data from the RemoveFile table.

ACE34

The RemoveIniFileData ACE checks to see if data exists in the RemoveIniFile MSI table that is not being executed with an associated RemoveIniFiles action.

```
The package contains data ([RemoveIniFile]) in the 'RemoveIniFile' table but not the necessary
    actions to use this data. You should consider whether a 'RemoveIniFiles' action is needed for
    your 'InstallExecuteSequence' table.
```

Summary

Table 14-45 • ACE34 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks to see if data exists in the RemoveIniFile MSI table that is not being executed with an associated RemoveIniFiles action.
Result:	If a package contains data in the RemoveIniFile table but not the necessary actions to use this data, a warning is generated.
Resolution Type:	Manual
Resolution:	Consider whether a RemoveIniFiles action is needed for your InstallExecuteSequence table. If so, open the MSI file in InstallShield and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the RemoveIniFile table.

Resolution

Open the MSI file in InstallShield and add a RemoveIniFiles action to the InstallExecuteSequence table, or remove the unused data from the RemoveIniFile table.

ACE35

The RemoveRegistryData ACE checks to see if data exists in the RemoveRegistry MSI table that is not being executed with an associated RemoveRegistryValues action.

```
The package contains data ([RemoveRegistry]) in the 'RemoveRegistry' table but not the necessary
    actions to use this data. You should consider whether a 'RemoveRegistryValues' action is needed
    for your 'InstallExecuteSequence' table.
```

Summary

Table 14-46 • ACE35 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Recommended Rules
Description:	Checks to see if data exists in the RemoveRegistry MSI table that is not being executed with an associated RemoveRegistryValues action.
Result:	If a package contains data in the RemoveRegistry table but not the necessary actions to use this data, a warning is generated.
Resolution Type:	Manual

Table 14-46 • ACE35 Summary

Topic	Description
Resolution:	Consider whether a RemoveRegistryValues action is needed for your InstallExecuteSequence table. If so, open the MSI file in InstallShield and add the appropriate action to the InstallExecuteSequence table, or remove the unused data from the RemoveRegistry table.

Resolution

Open the MSI file in InstallShield and add a RemoveRegistryValues action to the InstallExecuteSequence table, or remove the unused data from the RemoveRegistry table.

ACE36

ACE36 checks the required Merge Modules in a package to confirm that they also exist in the Application Catalog. The ACE36 Warning String is displayed in the following format:

```
The required [MERGE_MODULE_NAME], version [REQUIRED_VERSION] Merge Module is included with this
    package and yet not imported into the Application Catalog. It is recommended that all Merge
    Modules be imported into the Application Catalog.
```

Note • *ACE26 and ACE36, optional Best Practice ACEs, both check for conflicts with Merge Modules. ACE26 checks merge modules listed in the* ***ModuleSignature*** *table while ACE36 checks the* ***ModuleDependency*** *table.*

These ACEs are provided to encourage you to import Merge Modules into the Application Catalog and, by doing so, improve the effectiveness of ACE12, which checks for components that contain files that could be replaced by one of the imported Merge Modules.

Summary

Table 14-47 • ACE36 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Merge Module Integrity
Description:	Checks to see if a package's required Merge Modules have been imported into the Application Catalog.
Result:	If any of a package's required Merge Modules do not exist in the Application Catalog, a warning is generated.
Resolution Type:	Manual
Resolution:	Import the package's missing Merge Modules into the Application Catalog.

Resolution

Import the package's missing Merge Modules into the Application Catalog.

ACE200

ACE200 checks to see if two or more packages contain a shortcut with the same display name and location. It can be run against an App-V source package and against either an App-V or MSI target package. ACE200 identifies the shortcut name and location for the source package, then compares it to the target package(s) name and location.

The ACE200 Error string is displayed in the following format:

```
Shortcut [SHORTCUT_NAME] in package [PACKAGE_NAME] has a target that conflicts with shortcut
    [SHORTCUT_NAME] in package [PACKAGE_NAME]
```

Summary

Table 14-48 • ACE200 Summary

Topic	Description
Conflict Type:	Conflict Rule / App-V Recommended Rules
Description:	Checks to see if two or more packages contain a shortcut with the same display name and location.
Result:	If two or more packages contain a shortcut with the same display name and location, an error is generated.
Resolution Type:	Manual using the Virtual Package Editor
Resolution:	To resolve this ACE in an App-V package, perform the following steps: ***To resolve ACE200:*** 1. Open the App-V package in the Virtual Package Editor. 2. Open the **Shortcuts** view, and do one of the following: Select the shortcut, and then modify the value in the **Display Name** setting or the **Location** setting. Remove the shortcut from the App-V package.

Resolution

Open the App-V package in the Virtual Package Editor, open the **Shortcuts** view, and do one of the following:

- Select the shortcut, and then modify the value in the **Display Name** setting or the **Location** setting.
- Remove the shortcut from the App-V package.

ACE201

ACE201 checks to see if a target in the package has a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.

The ACE201 Warning string is displayed in the following format:

```
Package [PACKAGE_NAME] has a shortcut named [NAME] with a hardcoded Target of [TARGET].
```

Summary

Table 14-49 • ACE201 Summary

Topic	Description
Conflict Type:	Best Practice Rule / App-V Recommended Rules
Description:	Checks to see if a target in the package has a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.
Result:	If a target in the package has a hard-coded path, a warning is generated.
Resolution Type:	Manual
Resolution:	To resolve this ACE in an App-V package, change the path of the target to use a variable instead of a hard-coded path, perform the following steps: ***To resolve ACE201:*** 1. Open the App-V package in Virtual Package Editor. 2. In the View List under **Application Data**, click **Shortcuts**. 3. In the **Targets** explorer, select the target that contains the hard-coded path. 4. In the **Target** setting, replace the existing path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`—instead of the hard-coded path. ***Note*** • *If there is no appropriate `CSIDL` or `SFT` constant, you may need to use a hard-coded path that starts with a drive letter.*

Resolution

To resolve this ACE in an App-V package, change the path of the target to use a variable instead of a hard-coded path by performing the following steps:

Task: ***To resolve ACE201:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Application Data**, click **Shortcuts**.
3. In the **Targets** explorer, select the target that contains the hard-coded path.
4. In the **Target** setting, replace the existing path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`—instead of the hard-coded path.

Note • *If there is no appropriate `CSIDL` or `SFT` constant, you may need to use a hard-coded path that starts with a drive letter.*

ACE202

ACE202 checks to see if a command-line argument for a target in the package includes a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.

The ACE202 Warning string is displayed in the following format:

```
Package [PACKAGE_NAME] has a shortcut named [NAME] with a hardcoded argument of [ARGUMENTS].
```

Summary

Table 14-50 • ACE202 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks to see if a command-line argument for a target in the package includes a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.
Result:	If a command-line argument for a target in the package includes a hard-coded path, a warning is generated.
Resolution Type:	Manual
Resolution:	To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path by performing the following steps: ***To resolve ACE202:*** 1. Open the App-V package in the Virtual Package Editor. 2. In the View List under **Application Data**, click **Shortcuts**. 3. In the **Targets** explorer, select the target that contains the hard-coded path. 4. In the **Arguments** setting, replace the existing path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`—instead of the hard-coded path. ***Note*** • *If there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.*

Resolution

To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path by performing the following steps:

Task: ***To resolve ACE202:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Application Data**, click **Shortcuts**.
3. In the **Targets** explorer, select the target that contains the hard-coded path.
4. In the **Arguments** setting, replace the existing path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`—instead of the hard-coded path.

Note • *If there is no appropriate* `CSIDL` *or* `SFT` *constant, you may need to use a hard-coded path that starts with a drive letter.*

ACE203

ACE203 checks to see if a working directory for a target in the package includes a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.

The ACE203 Warning string is displayed in the following format:

```
Package [PACKAGE_NAME] has a shortcut named [NAME] with a hardcoded working directory of
    [DIRECTORY_NAME].
```

Summary

Table 14-51 • ACE203 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks to see if a working directory for a target in the package includes a hard-coded path, such as `C:\...\`, which may not be present in a virtual environment.
Result:	If a working directory for a target in the package includes a hard-coded path, a warning is generated.
Resolution Type:	Manual
Resolution:	To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path by performing the following steps: ***To resolve ACE203:*** 1. Open the App-V package in the Virtual Package Editor. 2. In the View List under **Application Data**, click **Shortcuts**. 3. In the **Targets** explorer, select the target that contains the hard-coded path. 4. In the **Working Directory** setting, replace the existing path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`—instead of the hard-coded path. ***Note*** • *If there is no appropriate CSIDL or SFT constant, you may need to use a hard-coded path that starts with a drive letter.*

Resolution

To resolve this ACE in an App-V package, change the path to use a variable instead of a hard-coded path by performing the following steps:

Task: ***To resolve ACE203:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Application Data**, click **Shortcuts**.
3. In the **Targets** explorer, select the target that contains the hard-coded path.
4. In the **Working Directory** setting, replace the existing path with a path that uses a `CSIDL` constant or an `SFT` constant—such as `CSIDL_APPDATA` or `SFT_PROGRAM_FILES_X64`—instead of the hard-coded path.

Note • *If there is no appropriate `CSIDL` or `SFT` constant, you may need to use a hard-coded path that starts with a drive letter.*

ACE204

ACE204 checks to see if two or more packages have the same package GUID. If two packages have the same package GUID, they cannot be deployed simultaneously as separate packages. ACE204 can be run against an App-V source package and against only an App-V target package.

The ACE204 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has a Package GUID that conflicts with package [PACKAGE_NAME]
```

Summary

Table 14-52 • ACE204 Summary

Topic	Description
Conflict Type:	Conflict / App-V Recommended Rules
Description:	Checks to see if two or more packages have the same package GUID.
Result:	If two or more packages have the same package GUID, an error is generated.
Resolution Type:	Manual
Resolution:	If you are creating an update package that can upgrade earlier versions of the virtual package, the package GUID should stay the same. If you are creating a new package that can be deployed simultaneously as another package, the package GUID in one of the packages must be changed. To change the package GUID, open the App-V package in the Virtual Package Editor and save the package as a new package.

Resolution

If you are creating an update package that can upgrade earlier versions of the virtual package, the package GUID should stay the same.

If you are creating a new package that can be deployed simultaneously as another package, the package GUID in one of the packages must be changed. To change the package GUID, open the App-V package in the Virtual Package Editor and save the package as a new package.

ACE205

ACE205 checks to see if two or more packages have the same name. This is not advisable from a best practice perspective, and it may cause some issues if you try to simultaneously deploy the App-V packages. ACE205 can be run against an App-V source package and against either an App-V or MSI target package.

The ACE205 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has a name conflict with package [PACKAGE_NAME].
```

Summary

Table 14-53 • ACE205 Summary

Topic	Description
Conflict Type:	Conflict / App-V Recommended Rules
Description:	Checks to see if two or more packages have the same name.
Result:	If two or more packages have the same name, an error is generated.
Resolution Type:	Manual
Resolution:	To resolve this ACE in an App-V package, perform the following steps: ***To resolve ACE205:*** 1. Open the App-V package in the Virtual Package Editor. 2. In the View List under **Package Information**, click **General Information**. 3. In the **Name** setting, replace the duplicate name with a unique name.

Resolution

To resolve this ACE in an App-V package, perform the following steps:

Task: ***To resolve ACE205:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Package Information**, click **General Information**.
3. In the **Name** setting, replace the duplicate name with a unique name.

ACE206

ACE206 checks to see if two or more packages have support for the same file extension or progid. A file extension can be registered with only one application at a time. ACE206 can be run against an App-V source package and against either an App-V or MSI target package.

The ACE206 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has a conflicting file extension [FILE_EXTENSION] or progid [PROGID] with
    package [PACKAGE_NAME]
```

Summary

Table 14-54 • ACE206 Summary

Topic	Description
Conflict Type:	Conflict / App-V Recommended Rules
Description:	Checks to see if two or more packages have support for the same file extension or progid.
Result:	If two or more packages have support for the same file extension or progid, an error is generated.
Resolution Type:	Manual
Resolution:	To resolve ACE206, you may need to decide which package should contain the file extension association and which should not. Then you can use the Virtual Package Editor to remove the appropriate file extension.

Resolution

To resolve ACE206, you may need to decide which package should contain the file extension association and which should not. Then you can use the Virtual Package Editor to remove the appropriate file extension.

ACE207

ACE207 checks to see if two or more packages have the same long or short name for the root folder. These names must be unique because two packages with the same root folder name cannot be deployed simultaneously. ACE207 can be run against an App-V source package and against only an App-V target package.

The ACE207 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has a conflicting root Directory [DIRECTORY_NAME] with package [PACKAGE_NAME]
```

Summary

Table 14-55 • ACE207 Summary

Topic	Description
Conflict Type:	Conflict / App-V Recommended Rules
Description:	Checks to see if two or more packages have the same long or short name for the root folder.
Result:	If two or more packages have the same long or short name for the root folder, an error is generated.
Resolution Type:	Manual
Resolution:	To resolve ACE207, perform the following steps: ***To resolve ACE207:*** 1. Open the App-V package in the Virtual Package Editor. 2. In the View List under **Package Information**, click **General Information**. 3. In the **Root Folder Name** setting, replace the duplicate folder name with a unique folder name. ***Note*** • *Instances of the old package's root folder name may still exist in location-related configuration data, such as in registry entries, .ini files, or XML files in the App-V package. The root folder name is not updated in those areas automatically if you change the root folder name in the **General Information** view.* *Therefore, if you know that the old package contains configuration data, you may need to identify where it is. Then you can use the Virtual Package Editor to update the root folder name as necessary. For example, you may want to use the Virtual Package Editor to extract a configuration file from the package. Next, you can update the root folder name in the file. In the Virtual Package Editor, you would then delete the old file from the App-V package, and add the updated file.*

Resolution

To resolve ACE207, perform the following steps:

Task: ***To resolve ACE207:***

1. Open the App-V package in the Virtual Package Editor.
2. In the View List under **Package Information**, click **General Information**.
3. In the **Root Folder Name** setting, replace the duplicate folder name with a unique folder name.

Note *• Instances of the old package's root folder name may still exist in location-related configuration data, such as in registry entries, .ini files, or XML files in the App-V package. The root folder name is not updated in those areas automatically if you change the root folder name in the* ***General Information*** *view.*

Therefore, if you know that the old package contains configuration data, you may need to identify where it is. Then you can use the Virtual Package Editor to update the root folder name as necessary. For example, you may want to use the Virtual Package Editor to extract a configuration file from the package. Next, you can update the root folder name in the file. In the Virtual Package Editor, you would then delete the old file from the App-V package, and add the updated file.

ACE208

ACE208 checks to see if an App-V package does not contain any shortcuts.

You can ignore this ACE if this package is intended to be used as a dependency by a different App-V package through Dynamic Suite Composition. In this case, you need to edit the other App-V package in the Virtual Package Editor and select this App-V package as a dependency in the **Dependencies** view.

If this package is intended to be used as a plug-in, you need to create a different kind of shortcut. For normal applications you target the application's executable; for plug-ins, you target the application which it plugs into. Some common examples include Office and Internet Explorer.

The ACE208 Warning string is displayed in the following format:

```
Package [PACKAGE_NAME] has no shortcuts.
```

Summary

Table 14-56 • ACE208 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks to see if an App-V package does not contain any shortcuts.
Result:	If an App-V package does not contain any shortcuts, a warning is generated.
Resolution Type:	Manual
Resolution:	You can ignore this ACE if this package is intended to be used as a dependency by a different App-V package through Dynamic Suite Composition. In this case, you need to edit the other App-V package in the Virtual Package Editor and select this App-V package as a dependency in the **Dependencies** view. If this package is intended to be used as a plug-in, you need to create a different kind of shortcut. For normal applications you target the application's executable; for plug-ins, you target the application which it plugs into. Some common examples include Office and Internet Explorer. If end users need to be able to launch this App-V package independently, consider opening the package in the Virtual Package Editor and adding a target to the App-V package if necessary, and then adding a shortcut to the target.

Resolution

You can ignore this ACE if this package is intended to be used as a dependency by a different App-V package through Dynamic Suite Composition. In this case, you need to edit the other App-V package in the Virtual Package Editor and select this App-V package as a dependency in the **Dependencies** view.

If this package is intended to be used as a plug-in, you need to create a different kind of shortcut. For normal applications you target the application's executable; for plug-ins, you target the application which it plugs into. Some common examples include Office and Internet Explorer.

If end users need to be able to launch this App-V package independently, consider opening the package in the Virtual Package Editor and adding a target to the App-V package if necessary (using the **Shortcuts** view), and then adding a shortcut to the target.

ACE209

ACE209 checks App-V packages for shell extensions.

The ACE209 Warning string is displayed in the following format:

```
Package [PACKAGE_NAME] has shell extensions.
```

Summary

Table 14-57 • ACE209 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks App-V packages for shell extensions.
Result:	If an App-V package has a shell extension, a warning is generated.
Resolution Type:	Manual
Resolution:	You need to assess how important the shell extension is to the application so that you can determine if it matters if that shell extension will not function. You then need to do one of the following: • **If the shell extension is unimportant**, it is probably safe to deploy the package with slightly reduced functionality. • **If the shell extension is important**, this package will not function well and should not be deployed.

Resolution

You need to assess how important the shell extension is to the application so that you can determine if it matters if that shell extension will not function. You then need to do one of the following:

- **If the shell extension is unimportant**, it is probably safe to deploy the package with slightly reduced functionality.
- **If the shell extension is important**, this package will not function well and should not be deployed.

ACE210

ACE210 checks App-V packages for ClickOnce.

The ACE210 Warning string is displayed in the following format:

```
Package [PACKAGE_NAME] has ClickOnce.
```

Summary

Table 14-58 • ACE210 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks App-V packages for ClickOnce.
Result:	If an App-V package has a ClickOnce, a warning is generated.
Resolution Type:	Manual
Resolution:	You need to assess how important the ClickOnce is to the application so that you can determine if it matters if that the ClickOnce will not function. You then need to do one of the following: • **If the ClickOnce is unimportant**, it is probably safe to deploy the package with slightly reduced functionality. • **If the ClickOnce is important**, this package will not function well and should not be deployed.

Resolution

You need to assess how important the ClickOnce is to the application so that you can determine if it matters if that the ClickOnce will not function. You then need to do one of the following:

- **If the ClickOnce is unimportant**, it is probably safe to deploy the package with slightly reduced functionality.
- **If the ClickOnce is important**, this package will not function well and should not be deployed

ACE211

ACE211 checks App-V packages for DLL surrogates.

The ACE211 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has dll surrogates.
```

Summary

Table 14-59 • ACE211 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks App-V packages for DLL surrogates.
Result:	If an App-V package has a DLL surrogate, an error is generated.
Resolution Type:	Manual
Resolution:	You need to assess how important the DLL surrogate is to the application so that you can determine if it matters if that DLL surrogate will not function. You then need to do one of the following: • **If the DLL surrogate is unimportant**, it is probably safe to deploy the package with slightly reduced functionality. • **If the DLL surrogate is important**, this package will not function well and should not be deployed.

Resolution

You need to assess how important the DLL surrogate is to the application so that you can determine if it matters if that DLL surrogate will not function. You then need to do one of the following:

- **If the DLL surrogate is unimportant**, it is probably safe to deploy the package with slightly reduced functionality.
- **If the DLL surrogate is important**, this package will not function well and should not be deployed.

ACE212

ACE212 checks App-V packages for boot services.

The ACE212 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has boot services.
```

Summary

Table 14-60 • ACE212 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks App-V packages for boot services.
Result:	If an App-V package has boot services, an error is generated.
Resolution Type:	Manual
Resolution:	You need to assess how important the boot service is to the application so that you can determine if it matters if that boot service will not function. You then need to do one of the following: • **If the boot service is unimportant**, it is probably safe to deploy the package with slightly reduced functionality. • **If the boot service is important but separable**, install the boot service on the main machine, and then a modified virtual package can then be deployed successfully. • **If the boot service is important but not separable**, this package will not function well and should not be deployed.

Resolution

Separate the boot services from the rest of the application (if possible) so that the boot services can be installed on the main machine. Then, virtualize the rest of the application.

ACE213

ACE213 checks App-V packages for OS integrated files.

The ACE213 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has OS integrated files
```

Summary

Table 14-61 • ACE213 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks App-V packages for OS integrated files.
Result:	If an App-V package has an OS integrated file, an error is generated.
Resolution Type:	None
Resolution:	It is likely that the OS integrated file is central to the virtual package. Therefore it is recommended that this App-V package not be deployed.

Resolution

It is likely that the OS integrated file is central to the virtual package. Therefore it is recommended that this App-V package not be deployed.

ACE214

ACE214 checks App-V packages for drivers.

The ACE214 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has drivers
```

Summary

Table 14-62 • ACE214 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks App-V packages for drivers.
Result:	If an App-V package has a driver, an error is generated.
Resolution Type:	Manual
Resolution:	Separate the drivers from the rest of the application so that the drivers can be installed on the main machine. Then, virtualize the rest of the application.

Resolution

Separate the drivers from the rest of the application so that the drivers can be installed on the main machine. Then, virtualize the rest of the application.

ACE215

ACE215 indicates that an App-V package contains a shortcut (App-V application) that uses the same name and version as one in another package. The combination of the name and version should be unique for shortcuts in different packages, since only one application is published and available at any given time. ACE215 can be run against an App-V source package and against only an App-V target package.

The ACE215 Error string is displayed in the following format:

```
Shortcut [SHORTCUT_NAME] in package [PACKAGE_NAME] has a name and version that conflicts with
    shortcut [SHORTCUT_NAME] in package [PACKAGE_NAME]
```

Summary

Table 14-63 • ACE215 Summary

Topic	Description
Conflict Type:	Conflict / App-V Recommended Rules
Description:	Checks to see if an App-V package contains a shortcut (App-V application) that uses the same name and version as one in another package.
Result:	If an App-V package contains a shortcut (App-V application) that uses the same name and version as one in another package, an error is generated.
Resolution Type:	Manual
Resolution:	To resolve this ACE in an App-V package, perform the following steps: ***To resolve ACE215:*** 1. Open the App-V package in the Virtual Package Editor. 2. Open the **Shortcuts** view, and do one of the following: Select the target that contains the shortcut, and then modify the value in the **Name** setting or the **Target Version** setting. Remove the shortcut from the App-V package.

Resolution

To resolve this ACE in an App-V package, perform the following steps:

Task: ***To resolve ACE215:***

1. Open the App-V package in the Virtual Package Editor.
2. Open the **Shortcuts** view, and do one of the following:
   - Select the target that contains the shortcut, and then modify the value in the **Name** setting or the **Target Version** setting.
   - Remove the shortcut from the App-V package.

ACE216

ACE216 indicates that an App-V package's `.sft` file name that is over 56 characters in length.

The ACE216 Error string is displayed in the following format:

```
Package [PACKAGE_NAME] has a SFT filename that is too long (> 56 characters).
```

Summary

Table 14-64 • ACE216 Summary

Topic	Description
Conflict Type:	Best Practice / App-V Recommended Rules
Description:	Checks to see if an App-V package's `.sft` file name is over 56 characters in length.
Result:	If an App-V package's `.sft` filename is over 56 characters in length, an error is generated.
Resolution Type:	Manual
Resolution:	To resolve this ACE in an App-V package, rename the `.sft` file to give it a name that contains less than 56 characters.

Resolution

To resolve this ACE in an App-V package, rename the `.sft` file to give it a name that contains less than 56 characters.

WTS01

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

Note • *If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS01 checks to see if the ALLUSERS property is defined. The WTS01 warning string is displayed in the following format:

```
The product [PRODUCT_NAME] ('[VERSION]') is currently configured to be installed as per-user. This
    value affects deployment of this package in a terminal server environment.
```

Summary

Table 14-65 • WTS01 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS01 checks to see if the ALLUSERS property is defined.
Result:	If the ALLUSERS property is defined, a warning is generated.
Resolution Type:	Automatic (WTSFIX01)
Resolution:	The ALLUSERS property is set to 1.

WTS02

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

Note • *If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS02 checks for any registry entries installed to per-user specific locations. The WTS02 warning string is displayed in the following format:

```
The component [COMPONENT_NAME] in package [PACKAGE_NAME]('[VERSION]') contains a per-user registry
    key [KEY_NAME] with KeyPath [KEYPATH_VALUE]. This value affects deployment of this package in a
    terminal server environment.
```

Summary

Table 14-66 • WTS02 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS02 checks for any registry entries installed to per-user specific locations.
Result:	If a registry entry is installed to a per-user specific location, a warning is generated.
Resolution Type:	Automatic (WTSFIX02)
Resolution:	Clears the KEYPATH entry for these resources to ensure that Windows Installer repair mode will not be invoked.

WTS03

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

Note • *If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS03 checks for any file entries installed to per-user specific locations. The WTS03 warning string is displayed in the following format:

```
The component '[COMPONENT_NAME]' in package '[PACKAGE_NAME]'('[VERSION]') has a per-user destination
    '[PATH_NAME]'. This value affects deployment of this package in a terminal server environment.
```

Summary

Table 14-67 • WTS03 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS03 checks for any file entries installed to per-user specific locations.
Result:	If any file entries are installed to per-user specific locations, a warning is generated.
Resolution Type:	Automatic (WTSFIX03)

Table 14-67 • WTS03 Summary

Topic	Description
Resolution:	Clears the KEYPATH entry for these resources to ensure that Windows Installer repair mode will not be invoked.

WTS04

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

Note • *If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS04 checks for any ODBC data source entries installed to per-user specific locations. The WTS04 warning string is displayed in the following format:

```
The component '[COMPONENT_NAME]' in package '[PACKAGE_NAME]'('[VERSION]') has a per-user ODBC Data
    Source '[NAME]'('[VALUE]'). This value affects deployment of this package in a terminal server
    environment.
```

Summary

Table 14-68 • WTS01 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS04 checks for any ODBC data source entries installed to per-user specific locations.
Result:	If an ODBC data source entry is installed to a per-user specific location, a warning is generated.
Resolution Type:	Automatic (WTSFIX04)
Resolution:	Clears the KEYPATH entry for these resources to ensure that Windows Installer repair mode will not be invoked.

WTS05

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

***Note** • If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS05 checks for any environment strings installed to per-user specific locations. The WTS05 warning string is displayed in the following format:

```
The component '[COMPONENT]' in package '%s'('%s') has a per-user Environment Setting
    '[NAME]'('[VALUE]'). This value affects deployment of this package in a terminal server
    environment.
```

Summary

Table 14-69 • WTS05 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS05 checks for any environment strings installed to per-user specific locations.
Result:	If an environment string is installed to a per-user specific locations, a warning is generated.
Resolution Type:	Manual
Resolution:	Duplicate the per-user Environment table entries and conditionalize the installation of these new Environment table entries to only be installed if the ALLUSERS property is set to 1.

WTS06

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

***Note** • If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS06 checks a package to make sure that all of the executables that it contains have the TSAWARE flag enabled. The WTS06 error string is displayed in the following format:

```
The binary '[EXECUTABLE_FILE_NAME]' in package '[PACKAGE_NAME]'('PACKAGE_NAME') does not have the
    TSAWARE flag set. This affects the deployment of this package in a terminal server environment.
```

Summary

Table 14-70 • WTS06 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS06 checks packages to determine if they contain an executable that does not have the TSAWARE flag enabled.
Result:	If a package contains an executable that does not have the TSAWARE flag enabled, an error is generated.
Resolution Type:	Manual
Resolution:	To resolve this error, remove from the package the executable that does not have the TSAWARE flag enabled.

WTS07

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

Note • *If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS07 checks the LaunchCondition table of a package for instances of `TerminalServer` or `RemoteAdminTS` conditions. The WTS07 warning/error string is displayed in the following format:

```
The InstallExecuteSequence/InstallUISequence table has a Type 19 CustomAction [ACTION] with the
    condition [CONDITION]. This may affect the deployment of this package in a terminal server
    environment.
```

OR

```
The LaunchCondition table has the condition [CONDITION]. This may affect the deployment of this
    package in a terminal server environment.
```

Summary

Table 14-71 • WTS07 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS07 checks a package for instances of `TerminalServer/RemoteAdminTS` conditions.
Result:	If package includes an instance of a `TerminalServer/RemoteAdminTS` condition, a warning/error is generated.
Resolution Type:	Manual
Resolution:	This issue cannot be resolved unless it is possible to delete the `TerminalServer/RemoteAdminTS` condition from the package. Contact the software vendor to determine the ramifications of removing the condition.

WTS08

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

Note • *If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS08 checks a package for 16-bit applications; 16-bit applications will not run on Windows 2008 R2 because it is a 64-bit platform. The WTS08 error string is displayed in the following format:

```
The binary '[FILE_NAME]' in package '[PACKAGE_NAME]'('[PACKAGE_NAME]') is 16-bit. This affects the
    deployment of this package in a terminal server environment.
```

Summary

Table 14-72 • WTS08 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS08 checks a package for 16-bit binary files.
Result:	If a package includes a 16-bit binary file (`.exe`, `.dll`, or `.ocx`) , an error is generated.
Resolution Type:	Manual
Resolution:	To resolve this error, remove the specified 16-bit binary file from the Windows Installer package.

WTS09

This ACE examines a Windows Installer package for compatibility with Windows Terminal Server. Windows Terminal Server generally requires installation components to be installed for all users, rather than on a per-user basis.

Note • *If your package is not targeting the Windows Terminal Server environment, then you do not need to run this ACE.*

WTS09 checks a package for applications manifested to run as Administrator. These packages will typically not be good candidates to run in a per user environment because users of Terminal Services typically do not have Admin privileges. The WTS09 error string is displayed in the following format:

```
The binary '[FILE_NAME]' in package '[PACKAGE_NAME]'('[PACKAGE_NAME]') is manifested to run as
    Administrator. This affects the deployment of this package in a terminal server environment.
```

Summary

Table 14-73 • WTS09 Summary

Topic	Description
Conflict Type:	Best Practice Rule / Windows Terminal Server Compatibility
Description:	WTS09 checks a package for binary files manifested to run as Administrator.

Table 14-73 • WTS09 Summary

Topic	Description
Result:	If a package includes a binary file (`.exe`, `.dll`, or `.ocx`) that is manifested to run as Administrator, an error is generated.
Resolution Type:	Manual
Resolution:	The only way to resolve this error is to remove the specified binary file that is manifested to run as Administrator from the Windows Installer package.

User-Defined ACEs

In addition to business rules (ACEs) shipped with ConflictSolver, you can also create your own company-specific business rules to use when detecting conflicts. For example, your organization may want to identify (and change) any VBScript custom actions that have a hard-coded drive letter, any applications that create desktop or uninstall shortcuts, any applications that have Startup registry entries, or any applications that place files in the system directory. You can create custom rules to identify these (and many more company-specific situations) using ConflictSolver.

Creation of user-defined ACEs is done using the Rules Wizard.

Types of User-Defined ACEs

In addition to the ACEs included with ConflictSolver, you can also create your own user-defined ACEs to use when detecting conflicts. You can create three types of ACEs:

Custom - Source Only Packages ACEs

Custom - Source Only Packages ACEs allow you to quickly test any column or any value of a table to support your business logic. For example, you could use a user-defined ACE to identify packages that create a desktop icon. To define a Source Only Packages ACE, you must define an SQL “Where” clause.

For an example of this type of user-defined ACE, see Creating a Custom/Source Only Packages ACE.

Note • *ConflictSolver supports external package conflict checking for Custom - Source Only Packages ACEs. The Source package can be selected from the Application Catalog Database or from an external MSI package.*

Custom - Source and Target Packages ACEs

Custom - Source and Target Packages ACEs allow you to compare columns or values of Source package tables (new packages that you want to install onto a user’s system) to columns or values of Target package tables (packages already installed on a user’s system).

For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the .ini file in the System directory of the Target system.

To define a Source and Target Packages ACE, you must define an SQL “Where” clause, and specify a Join Column—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages.

For an example of this type of user-defined ACE, see Creating a Custom/Source and Target Packages ACE.

***Note** • ConflictSolver does not support external package conflict checking for Custom - Source and Target Packages ACEs. Both the Source and Target Packages must be selected from the Application Catalog Database.*

DLL - User Provided DLL Based ACEs

DLL - User Provided DLL Based ACEs allow you to run more complex tests—testing many tables in any combination. For example, you could use a DLL-Based ACE to confirm that a source product language is the same as all target product languages. To define a DLL-Based ACE, you use SQL and various programming languages to construct a Windows DLL. With DLL-Based ACEs, you can use a Conflict Application Resolution Definition (CARD) to fix the conflict.

For an example of this type of user-defined ACE, see Creating a User Provided DLL-Based ACE.

Where You Create User-Defined ACEs

When creating ACEs, you need to provide basic information for display in the Rules viewer, Conflicts tab of the Options dialog box, and in the Output Window. You must associate a table with the ACE. You also must categorize the ACE (either using existing ACE categories or by creating your own).

Creation of User-Defined ACEs is done using the Rules Wizard. You can view and edit ACEs by launching the Rules Viewer Dialog Box and opening an ACE in the ACE Rule Properties Dialog Box.

Creating User-Defined ACEs

The section describes how to create the three types of user-defined ACEs:

- Creating a Custom/Source Only Packages ACE
- Creating a Custom/Source and Target Packages ACE
- Creating a User Provided DLL-Based ACE

***Note** • You can create user-defined ACEs for both Windows Installer and App-V packages.*

Creating a Custom/Source Only Packages ACE

You can create a user-defined ACE to apply to Source Only Packages. One common task you may want to create a Source Only Packages ACE to handle is to identify packages which create a desktop icon.

ConflictSolver supports external package conflict checking for Custom - Source Only Packages ACEs. The Source package can be selected from the Application Catalog database or from an external package.

Task: ***To create a Source Only Packages ACE that identifies desktop icon creation:***

1. Launch ConflictSolver.
2. From the **Conflicts** menu, select **Rules Viewer**. The **Rules Viewer** dialog box opens.
3. Click **New** to launch the **Rules Wizard**. The **Welcome Panel** opens.
4. Click **Next**. The **General Information Panel** opens.
5. Input the following values:

Option	Value
Name	MYACE
Associated Table	csmsiShortcut
Package Type	MSI
Brief Description	MYACE - Find desktop icons
Description	Locates package that create desktop icons.
Information URL	http://www.yourcompany.com/MYACE.htm

6. Click **Next**. The **Additional Information Panel** opens.
7. Input the following values:

Option	Value
Category	Shortcuts
Rule Type	Custom - Source Only Packages

8. Click **Next**. The **Custom Options Panel** opens.

9. Input the following values:

Option	Value
Error String	`Desktop icons [Name] are not allowed.`
Display String	`A desktop icon called [Name] is created.`
Severity	`Warning`
Report 'No' Results	`Deselected`

In the example above, `[Name]` is a token. Tokens allow you to insert values at run-time from the internal Application Catalog Database or from an external MSI package into the Error or Display string. To use token replacement in a string, click the arrow to the right of the Error String and Display String text boxes and select a column name from the list. The column name is then inserted into the string in the following format: `[ColumnName]`.

The Token list is provided for your convenience; if you prefer, you can type the tokens directly in the text boxes. You could also use the `[ProductName]` pseudo-token to insert the name of the package in a message, even though ProductName is not a table column name.

Note • *For more information, see Token Grammar*

10. Click Next. The **Where Clause Panel** opens.
11. In the **Where Clause Panel**, click Build Expression. The **Expression Builder** dialog box opens.
12. In the **Expression Builder**, input the following values:

Option	Value
Table Columns	`[Directory_]`
Comparison Operator	`= (Equal To)`
String Constant	`DesktopFolder`

13. Click OK to close the **Expression Builder** and return to the **Where Clause Panel**. The expression that you just built is now displayed in the Where Clause text box:

```
[Directory_] = 'DesktopFolder'
```

When you are constructing simple expressions, it is helpful to use the Expression Builder dialog box, but you are not limited to the formatting options that the Expression Builder provides to you. If you know how to write Where clauses in SQL, you can use significantly more powerful expressions by entering them directly in the Where Clause text box on the Where Clause Panel or on the Where Clause Tab of the ACE Rule Properties Dialog Box.

14. Click **Test** to validate the expression.

15. Click **Next**. The **Summary Panel** opens.
16. On the **Summary Panel**, review the summary of your new ACE and click **Finish**.
17. Click **Close** to exit the **Rules Viewer**. This new user-created ACE is now available for use in subsequent conflict identification.

Creating a Custom/Source and Target Packages ACE

You can create a custom ACE to apply to Source and Target Packages. For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the `.ini` file in the `System` directory of the Target system.

Caution • *ConflictSolver does not support external package conflict checking for Custom - Source and Target Packages ACEs. Both the Source and Target Packages must be selected from the ConflictSolver Application Catalog Database. If you attempt to run a conflict check on an external MSI package using a Source and Target Packages ACE, that custom ACE will not be executed.*

Task: ***To create a Source and Target Packages ACE that identifies .ini file conflicts:***

1. Launch ConflictSolver.
2. From the **Conflicts** menu, select **Rules Viewer**. The **Rules Viewer** dialog box opens.
3. Click New to launch the **Rules Wizard**. The **Welcome Panel** opens.
4. Click **Next**. The **General Information Panel** opens.
5. Input the following values:

Option	Value
Name	`INICheck`
Associated Table	`csmsiIniFile`
Package Type	`MSI`
Brief Description	`INICheck - Identifies conflicts found in an .ini file.`
Description	`Determines if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the .ini file in the Target system's System directory.`
Information URL	`http://www.yourcompany.com/INICheck.htm`

6. Click **Next**. The **Additional Information Panel** opens.

7. Input the following values:

Option	Value
Category	`INI Files`
Rule Type	`Custom - Source and Target Package`

8. Click **Next**. The **Custom Options Panel** opens.

9. Input the following values:

Option	Value
Error String	`The INI file called [Source.FileName] in the directory [Source.csFullPath] writes to the [Source.Section] section which is also written by the target package, [Target.ProductName].`
Display String	`The INI file called [Source.FileName] in the directory [Source.csFullPath] writes to the [Source.Section] section.`
Severity	`Warning`
Report 'No' Results	Deselected

In the example above, `[Source.Filename]`, `[Source.csFullPath]` and `[Source.Section]` are tokens. Tokens allow you to insert values at runtime from the internal Conflict Solver Application Catalog Database into the Error or Display string. To use token replacement in a string, click the arrow to the right of the Error String or Display String text boxes and select a column name from the list. The column name is then inserted into the string in the format of `[Source.ColumnName]` or `[Target.ColumnName]`, with the prefix identifying whether the column is in the Source or Target package.

***Note** • If no prefix is used, ConflictSolver assumes the "`Source.`" prefix.*

You can also use the `[Target.ProductName]` and `[Source.ProductName]` pseudo-tokens to insert the name of the Source or Target package in a message, even though ProductName is not a table column name.

***Note** • The Token list is provided for your convenience; if you prefer, you can type the variables directly in the text boxes. For more information, see Token Grammar.*

10. Click **Next**. The **Where Clause Panel** opens.

11. Click **Build Expression**. The **Expression Builder** dialog box opens.

12. Input the following values:

Option	Value
Table Columns	`[Source].[csFullPath]`
Comparison Operator	`= (Equal To)`
String Constant	`SystemFolder`

13. Click **OK** to close the **Expression Builder** and return to the **Where Clause Panel**. The expression that you just built is now displayed in the Where Clause text box:

```
[Source].[csFullPath] = 'SystemFolder'
```

Note • *When you are constructing simple expressions, it is helpful to use the Expression Builder dialog box, but you are not limited to the formatting options that the Expression Builder provides to you. If you know how to write Where clauses in SQL, you can use significantly more powerful expressions by entering them directly in the* **Where Clause** *text box on the* **Where Clause Panel** *or on the* **Where Clause Tab** *of the ACE Rule Properties Dialog Box.*

14. Click Build Expression again to open the **Expression Builder**.

15. Input the following values:

Option	Value
Table Columns	`[Source].[Section]`
Comparison Operator	`= (Equal To)`
String Constant	`[Target].[Section]`
Expression Operator	`AND`

Note • *When using the Expression Builder dialog box to create a Source and Target Packages custom ACE to compare the value of a column in the source table to the value of a column in the target table, you can select the first table column name from the Table Columns list. However, you have to manually enter the second table column name in the Constant text box. When doing so, enter the table column name using the same syntax that is used in the Table Columns list: [Source].[ColumnName] or [Target].[ColumnName].*

16. Click **OK** to close the **Expression Builder** and return to the **Where Clause Panel**. The expression that you just built is now displayed in the **Where Clause** text box, added to the end of the first expression you built:

```
[Source].[csFullPath] = 'SystemFolder' AND
[Source].[Section] = '[Target].[Section]'
```

17. Select **csFullPath** from the **Join Column** list.

    The **Join Column** is a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages.

18. Click **Test** to validate the expression. A message appears stating that the query executed properly.
19. Click **Next**. The **Summary Panel** opens.
20. Review the summary of your new ACE and click **Finish**.
21. Click **Close** to exit the **Rules Viewer**. This new Custom ACE is now available for use in subsequent conflict identification.

Creating a User Provided DLL-Based ACE

To demonstrate how to create a DLL-Based ACE, an ACE SDK was installed with AdminStudio, in the following directory:

AdminStudio Installation Directory\ConflictSolver\ACESDK

Included in this directory is a VC++ 6.0 DLL-based project which is a fully formed example ACE. This example includes utility functions that allow you to integrate your own ACE execution within our conflict persistence model.

Specifying the Visual Studio C++ Type Library File Path

ConflictSolver provides a data structure to each ACE constructed using the ACE SDK. This data structure includes an ADO Connection interface which is the means by which you can execute queries against the Application Catalog.

In order to build the ACE SDK, you will need to make sure that Visual Studio can locate the needed files for ADO.

The *first time* you create a DLL-based ACE, you need to open Visual Studio C++ and specify the path for the type library file (`msado15.dll`) by performing the following steps:

Task: ***To specify the Visual Studio C++ type library file path:***

1. Launch Visual Studio C++ 6.0.
2. Select **Options** from the **Tools** menu. The **Options** dialog box opens.
3. On the **Directories** tab, set the **Platform** field to **Win32** and the **Show directories for** field to **Library files**.
4. Under **Directories**, specify the location of the needed files for an ADO Connection interface. The directory specified below is a likely common directory for storing these files:

   `C:\Program Files\Common Files\System\ADO`

5. Click **OK** to exit the **Options** dialog box.

Caution • *If you fail to specify the correct Library files path, you will encounter the following error when building an ACESDK project with Visual Studio C++:*

```
fatal error C1083: Cannot open type library file: 'msado15.dll'; No such file or directory
```

Creating a DLL-Based ACE

To learn how to create a DLL-Based ACE, use the ACE SDK files to perform the following steps:

Task: ***To create a DLL-Based ACE:***

1. Launch Windows Explorer and navigate to the following directory:

   AdminStudio Installation Directory\ConflictSolver\ACESDK

2. Copy this folder and its contents and store it in a convenient location.
3. Launch VC++ 6.0 and open the ACESDK.dsp project file within that newly created folder.
4. Review the code and make any desired changes.
5. Still in VC++ 6.0, build the ACESDK.dsp project to create a new .DLL file.
6. Launch ConflictSolver.
7. From the **Conflicts** menu, select **Rules Viewer**. The **Rules Viewer** dialog box opens.
8. Click **New** to launch the **Rules Wizard**. The Rules Wizard **Welcome Panel** opens.
9. Click **Next**. The **General Information Panel** opens.
10. Input the following values:

Option	Value
Name	ACELanguage
Associated Table	csmsiProperty (This is the ConflictSolver table associated with this example ACE.)
Package Type	MSI
Brief Description	ACELanguage - Check product language consistency.
Description	Confirm that source product language is the same as all target product languages.
Information URL	http://www.yourcompany.com/ACELanguage.htm

11. Click **Next**. The **Additional Information Panel** opens.

12. Input the following values:

Option	Value
Category	Type in a new category name: `Product Language`
Rule Type	`DLL - User Provided DLL`

13. Click **Next**. The **DLL-Based ACEs Panel** opens.
14. Input the following values:

Option	Value
ACE/CARD DLL File	Click Browse and select the DLL that you built in Step 5 above.
ACE Function Name	`ExampleACE` (as designed in the sample)
CARD Function Name	`ExampleCARD` (as designed in the sample)

15. Click **Test** next to **ACE Function Name** or **CARD Function Name** to validate that the exported function does exist.
16. Click **Next**. The **Summary Panel** opens.
17. On the **Summary Panel**, review the summary of your new ACE and click **Finish**.
18. Click **Close** to exit the **Rules Viewer**. This new DLL-Based ACE is now available for use in subsequent conflict identification.

Editing User-Defined ACEs

Task: ***To edit a user-defined ACE:***

1. From the **Conflicts** menu, select **Rules Viewer**. The **Rules Viewer** dialog box opens.
2. Select the user-defined ACE you want to modify and click **Edit**. The **ACE Rule Properties** dialog box opens.
3. In the **ACE Rule Properties** dialog box, edit the options as necessary.
4. Click **OK** to close the dialog box.
5. Click **Close** to exit the **Rules Viewer** dialog box.

The modified ACE is available for use in subsequent conflict identification.

Deleting User-Defined ACEs

Task: ***To delete a user-defined ACE:***

1. From the **Conflicts** menu, select **Rules Viewer**. The **Rules Viewer** dialog box opens.
2. Select the ACE you want to remove and click **Delete**.
3. Confirm the deletion.

The user-defined ACE is removed from the available rules. If this ACE was the only one in a user-defined ACE category, the category will be removed when you close the Rules Viewer dialog box.

Viewing ACE Metrics

When ACEs are run, ConflictSolver generates metrics and logs them in the `AceLog.txt` file, located in the following directory:

```
AdminStudio Shared\ConflictSolver\AceLog.txt
```

The following is an example of the beginning of an AceLog.txt file:

```
11/06/03 09:26:35 ACE/CARD Execution started

-------------------------------------------
ACE03
-------------------------------------------
SELECT DISTINCT csmsiComponent.Component,
     csmsiComponent.ComponentId,csmsiComponent.RowID
     FROM csmsiComponent WHERE csmsiComponent.PkgRowID_ = 5

Records returned =>(124)
Query Time =>(0.23 seconds)
```

```
SELECT csmsiFile.FileName FROM csmsiFile WHERE
     csmsiFile.Component_ = 'DeletedLinks.10' AND
     csmsiFile.PkgRowID_ = 5

ORDER BY csmsiFile.FileName

Records returned =>(0)

Query Time =>(0.07 seconds)

Time taken to execute (ACE03) : 1.06 seconds

---------------------------------------------
ACE07
---------------------------------------------
SELECT [fs].[RowID], [ft].[RowID], [ft].[PkgRowID_],
     [ft].[FileName], [cs].[Component], [ct].[Component],
     [cs].[ComponentId], [ct].[ComponentId], [fs].[Version],
     [fs].[FileSize], [fs].[Language] FROM  (([csmsiFile] AS [fs]
     INNER JOIN  [csmsiFile] AS [ft] ON ( [fs].[FileName] =
     [ft].[FileName] AND [fs].[Version] = [ft].[Version] AND
     [fs].[FileSize] = [ft].[FileSize] AND [fs].[Language] =
     [ft].[Language]) ) INNER JOIN [csmsiComponent] AS [cs] ON
     [fs].[PkgRowID_] = [cs].[PkgRowID_] AND [cs].[Component] =
     [fs].[Component_]) INNER JOIN [csmsiComponent] AS [ct] ON
     [ft].[PkgRowID_] = [ct].[PkgRowID_] AND [cs].[ComponentId] <>
     [ct].[ComponentId] AND [cs].[csFullPath] = [ct].[csFullPath] AND
     [ct].[Component] = [ft].[Component_] WHERE  [fs].[PkgRowID_] = 5
     AND  [ft].[PkgRowID_] IN (4)

Records returned =>(0)
Query Time =>(0.10 seconds)

Time taken to execute (ACE07) : 0.74 seconds

---------------------------------------------
ACE08
---------------------------------------------

SELECT [cs].[Component] AS [SrcComponent], [fs].[RowID] AS [SrcRowID],
     [ft].[RowID] AS [TargetRowID], [ft].[PkgRowID_] AS
     [TargetPkgRowID], [fs].[Version] AS [SrcVersion],
     [fs].[FileName], [cs].[csFullPath], [ct].[Component] AS
     [TargetComponent], [ft].[Version] AS [TargetVersion] FROM
     (([csmsiFile] AS [fs] INNER JOIN [csmsiFile] AS [ft] ON
     [fs].[FileName] = [ft].[FileName]) INNER JOIN [csmsiComponent]
     AS [cs] ON [cs].[PkgRowID_] = [fs].[PkgRowID_] AND
     [cs].[Component] = [fs].[Component_]) INNER JOIN
     [csmsiComponent] AS [ct] ON [cs].[ComponentId] =
     [ct].[ComponentId] AND  [ct].[PkgRowID_] =
     [ft].[PkgRowID_] AND [ct].[Component] = [ft].[Component_] AND
     [cs].[csFullPath] = [ct].[csFullPath] WHERE
     [fs].[PkgRowID_] = 5 AND [ft].[PkgRowID_] IN (4) AND
     ([fs].[Version] <> [ft].[Version] OR ([fs].[Version] IS
     NULL AND [ft].[Version] IS NOT NULL) OR ([ft].[Version]
```

```
     IS NULL AND [fs].[Version] IS NOT NULL))

Records returned =>(0)
Query Time =>(0.09 seconds)

Time taken to execute (ACE08) : 0.73 seconds
```

Location of ACE Files

ACE information is stored in three files that are installed with ConflictSolver.

Table 14-74 • ACE File Names and Locations

Type	File Name	Installation Location
Standard ACEs	`isconflict.ace`	AdminStudio installation directory in the following folder: `ConflictSolver/Support`
Merge Module ACEs	`ismmconflict.ace`	AdminStudio installation directory in the following folder: `ConflictSolver/Support`
Custom ACE File	`CustomConflictFile.ace`	AdminStudio Shared directory

ConflictSolver requires that the Standard ACE and Merge Module ACE files must remain in their installed location. However, the location of the Custom ACE File can be changed by opening the **Rules** tab of the **Options** dialog box, and editing the **Custom ACE Rule File** field.

Conflict Application Resolution Definitions (CARDs)

Conflict Application Resolution Definitions (CARDs) are the rules used to fix conflicts identified by Application Conflict Evaluators (ACEs) when conflict detection is performed. Before a CARD is run, you are given the opportunity to review how the CARD will resolve the conflict and decide whether or not to perform the resolution. See Conflict Resolution Process for more information.

The following links, grouped by conflict type, explain each CARD.

Table 14-75 • CARD Categories

Conflict Type	Associated Resolutions
Component	CARD02, CARD04, CARD05, CARD06

Table 14-75 • CARD Categories

Conflict Type	Associated Resolutions
Files	CARD07
ODBC Resources	CARD15
Product Properties	CARD18, CARD19, CARD20

CARD Index

The following list provides links to each individual Conflict Application Resolution Definition (CARD) and its associated ACE (Application Conflict Evaluators). These CARDs are used to resolve conflicts between installation packages that were identified by the CARD's corresponding ACE.

Conflicts found by ACEs that do not have associated CARDs must be resolved manually. See ACE Index for more information.

Table 14-76 • CARD Index

CARD	Brief Description of ACE	Action Taken by CARD
CARD02 for ACE02	Checks to see if components in different packages that have matching ComponentIds also have identical destination paths.	The destination path of the component in the Source package is automatically set to match that of the component in the Target package.
CARD04 for ACE04	Checks to see if components that have NULL KeyPath values are valid by checking for an entry for the component in the CreateFolder table.	A CreateFolder entry is created for the component.
CARD05 for ACE05	Checks for the existence of more than one executable (EXE, DLL, OCX, HLP, CHM, TLB, SYS, DRV) per component in a Windows Installer package.	Modifies the component so that only one EXE or DLL exists, and it adds new components for remaining EXE, DLL, OCX, HLP, CHM, TLB, SYS, and DRV files.
CARD06 for ACE06	Checks whether the executable module (EXE, DLL, OCX, HLP, CHM, TLB, SYS, or DRV) within the component is the key file.	The executable module is automatically made the key file.
CARD07 for ACE07	Checks for the existence of the same file in components with different ComponentIds.	The Source package ComponentId is changed to match the Target package Component Id.

Table 14-76 • CARD Index

CARD	Brief Description of ACE	Action Taken by CARD
CARD15 for ACE15	Checks for the existence of identical ODBC entries in components with different ComponentIds.	ConflictSolver changes the Source ComponentId to match that of the Target ComponentId.
CARD18 for ACE18	Checks the Package Code to see if it is unique.	A new, unique Package Code is automatically generated to replace the duplicate code.
CARD19 for ACE19	Checks the Product Code to see if it is unique.	A new, unique Product Code is automatically generated to replace the duplicate code.
CARD20 for ACE20	Checks the Upgrade Code to see if it is unique.	A new, unique Upgrade Code is automatically generated to replace the duplicate code.

Note • *Before a CARD is run, you are given the opportunity to review how the CARD will resolve the conflict and decide whether or not to perform the resolution. See Conflict Resolution Process for more information*

CARD02

Conflict

CARD02 resolves ACE02, which checks to see if components in different packages that have matching ComponentIds also have identical destination paths. If components with the same ComponentId have different destination paths, ACE02 fails.

Resolution

CARD02 automatically sets the destination path of the component in the Source package to match that of the component in the Target package. To do this, CARD02 runs the following query and replaces the Directory_ column with a run-time translation of the targets component (`csFullPath`) path:

```
MsiDBUtils::GetDirectoryTargetPathKey.
SELECT `Directory_` FROM `Component`
     WHERE ComponentId='Source ComponentId'
```

CARD04

Conflict

CARD04 resolves ACE04, which checks to see if components that have NULL KeyPath values are valid by checking for an entry for the component in the CreateFolder table. If a component does not have a key file, and if the directory/component pair is not listed in the CreateFolder table, ACE04 fails.

Resolution

CARD04 creates a CreateFolder entry for the component by executing the following query:

```
INSERT INTO CreateFolder ( `Directory_`, `Component_` )
     VALUES ('Source Directory','Source Component')
```

CARD05

Conflict

CARD05 resolves ACE05, which checks for the existence of more than one executable (EXE, DLL, OCX, HLP, CHM, TLB, SYS, DRV) per component in a Windows Installer package. If more than one executable exists in a component, ACE05 fails.

Resolution

CARD05 modifies the component so that only one EXE or DLL exists, and it adds new components for remaining EXE, DLL, OCX, HLP, CHM, TLB, SYS, and DRV files. To do this, CARD05 generates a new component name and ComponentId and inserts a record in the Component table and FeatureComponents table. The relevant entry in the File table would then be updated to effectively move the file into the new component.

CARD06

Conflict

CARD06 resolves ACE06, which checks whether the executable module (EXE, DLL, OCX, HLP, CHM, TLB, SYS, or DRV) within the component is the key file. If the executable module is not the key file, ACE06 fails.

Resolution

CARD06 automatically makes the executable module the key file. To do this, CARD06 runs the following query and replaces the KeyPath with an entry from the File Table - File Column which is associated with this component and of the correct type.

```
SELECT `KeyPath` FROM `Component` WHERE `
     ComponentId` = 'Source ComponentId'
```

CARD07

Conflict

CARD07 resolves ACE07, which checks for the existence of the same file in components with different ComponentIds. If the same file exists in components with different ComponentIds, ACE07 fails.

Resolution

CARD07 changes the Source package ComponentId to match the Target package Component Id. To do this, CARD07 runs the following query against the source package and then updates the ComponentId with the ComponentId from the target package:

```
SELECT `ComponentId` FROM `Component` WHERE
     Component` = 'Source Package ComponentId'
```

CARD15

Conflict

CARD15 resolves ACE15, which checks for the existence of identical ODBC entries in components with different ComponentIds. If identical ODBC entries exist in components with different ComponentIds, ACE15 fails.

Resolution

CARD15 changes the Source ComponentId to match that of the Target ComponentId. To do this, CARD15 runs the following query against the source package and then updates the ComponentId with the ComponentId from the target package:

```
SELECT `ComponentId` FROM
Component` WHERE `Component` = 'Source Package ComponentId'
```

CARD18

Conflict

CARD18 resolves ACE18, which checks the Package Code to see if it is unique. If the Package Code is identical to any other Package Code in ConflictSolver, ACE18 fails.

Resolution

CARD18 updates the PackageCode by inserting a newly generated GUID into the MSI package using MsiSummaryInfoSetProperty.

CARD19

Conflict

CARD19 resolves ACE19, which checks the Product Code to see if it is unique. If the Product Code is identical to any other Product Code in ConflictSolver, ACE19 fails.

Resolution

CARD19 automatically generates a new, unique Product Code to replace the duplicate code. To do this, CARD19 runs the following query.

```
UPDATE Property SET `Value` = 'Newly generated GUID'
     WHERE `Property` = 'ProductCode'
```

CARD20

Conflict

CARD20 resolves ACE20, which checks the Upgrade Code to see if it is unique. If the Upgrade Code is not unique, ACE20 fails.

Resolution

CARD20 generates a new, unique Upgrade Code to replace the duplicate code. To do this, CARD20 runs the following query:

```
UPDATE Property SET `Value` = 'Newly generated GUID'
     WHERE `Property` = 'UpgradeCode'
```

ConflictSolver Reference

ConflictSolver Reference includes the same topics displayed when you click a help button from a Wizard or dialog box in the ConflictSolver interface. Reference information for ConflictSolver is organized into the following areas:

Table 14-77 • ConflictSolver Reference Organization

Section	Description
ConflictSolver Interface	This section contains information about the main ConflictSolver interface, including menus and the toolbar.
Views	Each view available in ConflictSolver is covered in this section.
Dialog Boxes	Specific help for each dialog box in ConflictSolver is covered in this section.
Wizards	This section contains a panel-by-panel reference for each Wizard in ConflictSolver.

ConflictSolver Interface

The ConflictSolver user interface consists of three areas: the ConflictSolver Navigation window, the Details pane, and the Output Window. Both the ConflictSolver Navigation window and the Output Window are dockable.

- **Navigation window**—The ConflictSolver Navigation window consists of two tabs: the Product View and the Merge Modules View.
- **Details pane**—When you select different items in the Product View or Merge Modules View, the Details pane displays corresponding information about that item.
- **Output window**—The Output Window consists of tabs where output is displayed during different ConflictSolver processes.

ConflictSolver also supports many context menus in addition to the menu bar and toolbar functionality.

Menus and Toolbar

The following commands and toolbar buttons are available in ConflictSolver. Additionally, several commands can be accessed through Context Menus.

Table 14-78 • Menus & Toolbars

Menu	Command	Shortcut	Button	Description
Catalog	**New**			Opens the Application Catalog Wizard, which you can use to create a new Application Catalog.
Catalog	**Connect**	Ctrl+O		Displays the Connect Application Catalog dialog box, where you can open an existing Application Catalog.
Catalog	**Refresh**	Ctrl+R		Refreshes the current Product and Merge Module views. This is particularly useful if multiple people are working on the same Application Catalog.
Catalog	**Disconnect**			Closes the currently open Application Catalog.
Catalog	**Properties**			View Application Catalog properties.
Catalog	**Change AES Password**			Change the password of the current user to log in to the AdminStudio Enterprise Server.
Catalog	**Log Out**			Log out of AdminStudio Enterprise Server.
Catalog	**Exit**			Closes ConflictSolver.
Edit	**Find**	Ctrl+F		Use to search for data in various tables in the Application Catalog.

Table 14-78 • Menus & Toolbars (cont.)

Menu	Command	Shortcut	Button	Description
View	**Toolbar**	Ctrl+0		Toggles the Toolbar.
View	**Status Bar**	Ctrl+1		Toggles the Status Bar.
View	**Output Window**	Ctrl+2		Toggles the Output Window.
View	**Application Manager**	Ctrl+3		Select to switch to Application Manager.
View	**ConflictSolver**	Ctrl+4		Select to switch to ConflictSolver.
Conflicts	**Validation Wizard**	F4		Launches the Validation Wizard, which you can use to select any Windows Installer package for validation. The output is displayed in the Output Window.
Conflicts	**Conflict Wizard**	F5		Launches the Conflict Wizard, through which you can open a Windows Installer package and start comparing it against packages in ConflictSolver for conflicts. The output is displayed in the Output Window.
Conflicts	**Patch Impact Analysis Wizard**	F7		Launches the Patch Impact Analysis Wizard, which you can use to obtain the Windows operating system and Microsoft application patches you need and thoroughly test the impact they will have on your environment.
Conflicts	**Stop**			Stops the validation or conflict search.
Conflicts	**Resolve External Package Conflicts**	F6		Displays the Conflict Resolution dialog box, through which you resolve conflicts discovered by ConflictSolver between an external package and packages already in ConflictSolver.
Conflicts	**Rules Viewer**			Displays the Rules Viewer dialog box, from which you can create new ACE rules.
Reports	**Package**			Creates the Package Report.
Reports	**File**			Creates the Files Report.

Table 14-78 • Menus & Toolbars (cont.)

Menu	Command	Shortcut	Button	Description
Reports	**Registry**			Creates the Registry Report.
Reports	**Crystal Reports Conflicts**			Creates the pre-defined Crystal Reports Conflicts Report.
Reports	**Crystal Reports Files**			Creates the pre-defined Crystal Reports Files Report.
Reports	**Crystal Reports Registry**			Creates the pre-defined Crystal Reports Registry Report.
Tools	**Customize**	Alt+M		Opens the Customize dialog box, where you can customize toolbar settings.
Tools	**Options**	Alt+T		Opens the Options dialog box, where you can change various ConflictSolver settings including default conflicts to identify, the default location for Windows Installer packages, the default Application Catalog, report locations, the default validation file, and duplicate package identification options.
Help	**Contents**			Launches the online Help Library and displays the Contents tab.
Help	**Index**			Launches the online Help Library and displays the Index tab.
Help	**Search**			Launches the online Help Library and displays the Search tab.
Help	**Support Central**			Connects to the AdminStudio Support Web site.
Help	**Web Community**			Connects to the Web Community.
Help	**ReadMe**			Displays the AdminStudio Release Notes.
Help	**Feedback**			Connects to an online form, through which you can provide feedback about ConflictSolver or AdminStudio.

Table 14-78 • Menus & Toolbars (cont.)

Menu	Command	Shortcut	Button	Description
Help	**About ConflictSolver**			Displays the About dialog box with version information for ConflictSolver.
	Help Library			Launches the online Help Library.

Product View Icons

The following icons are used on the ConflictSolver and Application Manager Product Views:

Table 14-79 • ConflictSolver and Application Manager Product View Icons

Name	Icon	Description
Application Catalog		The Application Catalog that you are connected to. When you select it, one of the following views opens: • Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards • Application Catalog View (Professional Edition)
Group		A group, which is used to organize your data.

Table 14-79 • ConflictSolver and Application Manager Product View Icons (cont.)

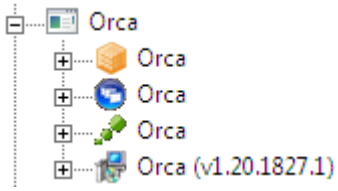
Name	Icon	Description
Product		An application that has been imported into the Application Catalog database. The application can include multiple deployment types—such as a Windows Installer package, a Microsoft App-V package, a VMware ThinApp package, etc. Each deployment type is indicated by a separate subnode of this application node. In the following example, the application includes a Microsoft App-V package, a Citrix XenApp package, a VMware ThinApp package, and a Windows Installer package. 
		Microsoft App-V package that has been imported into the Application Catalog.
		Citrix XenApp package that has been imported into the Application Catalog.
		VMware ThinApp package that has been imported into the Application Catalog.
		An “X” overlay on an icon indicates that that the package has an unresolved conflict, or that one of an application’s deployment types has an unresolved conflict.

Table 14-79 • ConflictSolver and Application Manager Product View Icons (cont.)

Name	Icon	Description
Product (continued)		An "!" overlay on an icon indicates that the package has an unresolved question with its associated files (or that one of an application's deployment types has an unresolved question with its associated files). One or more files, transforms, or patch files associated with a package is either missing from the original import directory or its last modified date does not match the last modified date stored in the Application Catalog. When a package icon with this overlay is selected, a message appears in the view identifying the files in question and prompting you to take action to resolve the problem.
		An icon that contains both an "!" and an "X" overlay icon indicates that it is a package with unresolved conflicts that also has an unresolved question with its associated files.
		This overlay icon indicates that the package is managed within the Software Repository. This overlay icon is only displayed in Application Manager.
		This overlay icon indicates that the package is managed within the Software Repository and is checked out. This overlay icon is only displayed in Application Manager.
		This overlay icon indicates that the package is linked to a Remote Application Catalog. See Automatically Importing Packages.
		These overlay icons indicate that the package is linked in a local Application Catalog that has been updated on the Remote Server and needs to be refreshed in order to update its data with the data on the Remote Server. See Manually Updating a Linked Package.
Transforms		A transform.
		A transform that has an unresolved question. It is either missing from the original import directory or its last modified date does not match the last modified date stored in the Application Catalog. When this icon is selected, a message appears in the Application Manager Product View prompting you to take action to resolve the problem.
OS Snapshot		An OS snapshot, which is a file representing a particular computer system's contents.
Conflicts		Click to access the package's Conflicts View, where you can access information pertaining to the last execution of the Conflict Wizard.

Table 14-79 • ConflictSolver and Application Manager Product View Icons (cont.)

Name	Icon	Description
Validation		Click to access the package's Validation View, which lists all ICE Errors and Warnings that were generated the last time the selected package was validated.
Patch Impacts		Click to access the package's Patch Impacts Analysis View, which lists patches for which there is patch impact data persisted against the product, and identifies the patch that caused the impact.
Dependencies		Click to display the Dependencies associated with the package.
Extended Attributes		Click to display the optional Extended Attributes associated with the package. See Extended Attributes View for more information.
Files, Files/ Components, Files/ Directories		Click to display the Files, Files/Components, or Files/Directories view for this package.
		Click to display the Files, Files/Components, or Files/Directories view for this package. The "!" overlay icon indicates that a file is missing from the original import directory. When this icon is selected, a message appears in the view describing the missing files and prompting you to locate the missing files.
INI File Changes		Click to display the INI File Changes View, listing any INI file changes made by the product.
Merge Modules		Click to display any merge modules included the package.
Associated Patches		Click to display the product's Associated Patches View, which displays a list of imported patches that, if installed, would update the selected package.
Registry		Click to display any registry entries created or changed by the package.
Shortcuts		Click to display any shortcuts created by the package.
Tables		Click to view the data for a given package in the Application Catalog.
Catalog History		Click to view a list of logged events for the selected package.
Virtualization Suitability		Click to view the results of a Virtualization Readiness analysis of that package. See Virtualization Suitability View.
Application Compatibility		Click to view Compatibility Solver test results for a selected Windows Installer package. See Application Compatibility View.

Table 14-79 • ConflictSolver and Application Manager Product View Icons (cont.)

Name	Icon	Description
App-V History		Click to open a view which lists an entry for each time this App-V package has been saved.
File Type Associations		Click to view a list of the file type associations for this App-V package.
Environment Variables		Click to display the environment variables used in this App-V package.

Context Menus

ConflictSolver includes several context menus which can be accessed by right-clicking on nodes within the Product View. These menus provide specific functionality in relation to what is clicked.

- Groups
- MSI Package
- Other Setup Types
- Marimba NCP Files
- OS Snapshot
- Patches View on the Patches Tab

Groups

When you right-click on a group in the Product View, the following commands are available through the context menu:

Table 14-80 • ConflictSolver Group Context Menu Commands

Command	Description
Conflict Wizard	Launches the Conflict Wizard directly to the Target Information Panel.
Patch Impact Analysis Wizard	Launches the Patch Impact Analysis Wizard, which you can use to identify conflicts between Microsoft patches and packages and OS Snapshots in your Application Catalog.
Scan for Dependencies	If dependency information has not already been generated by the Patch Impact Analysis Wizard, select this option to generate dependency information for all MSI packages in this group.

MSI Package

When you right-click on an MSI package in the Product View, the following commands are available through the context menu:

Table 14-81 • MSI Package Context Menu Commands

Command	Description
Validate Package	Select to validate the selected package.
Test Package	Select to test the selected package using PackageExpert. See Testing Packages From ConflictSolver for more information.
Conflict Wizard	Launches the Conflict Wizard directly to the Target Information Panel.
Patch Impact Analysis Wizard	Launches the Patch Impact Analysis Wizard, which you can use to identify conflicts between Microsoft patches and packages and OS Snapshots in your Application Catalog.
Scan for Dependencies	If dependency information has not already been generated by the Patch Impact Analysis Wizard, select this option to generate dependency information for an MSI package.
Associate with Workflow Manager Application	Launches the Associate with Workflow Manager Application dialog box, from which you can pick a package in Workflow Manager with which to associate the extended attribute data for the selected product.
Reports	Select one of the following: • **Package**—Creates the Package Report • **File**—Creates the Files Report. • **Registry**—Creates the Registry Report.

Other Setup Types

No context menu is available for an Other Setup Types package (a package containing non-MSI based setup files) in the Product View.

Marimba NCP Files

When you click on an Marimba NCP File package in the Product View, the following commands are available through the context menu:

Table 14-82 • Marimba NCP Files Context Menus

Command	Description
Conflict Wizard	Launches the Conflict Wizard directly to the Target Information Panel.

OS Snapshot

When you right-click on an OS Snapshot in the Product View, the following commands are available through the context menu:

Table 14-83 • OS Snapshot Context Menu Commands

Command	Description
Conflict Wizard	Launches the Conflict Wizard directly to the Target Information Panel.
Patch Impact Analysis Wizard	Launches the Patch Impact Analysis Wizard, which you can use to identify conflicts between Microsoft patches and packages and OS Snapshots in your Application Catalog.

Patches View on the Patches Tab

When you click on an individual patch in the Patch View, the following commands are available through the context menu:

Table 14-84 • Patches View Context Menus

Command	Description
Patch Impact Analysis Wizard	Launches the Patch Impact Analysis Wizard, which you can use to identify conflicts between Microsoft patches and packages and OS Snapshots in your Application Catalog.
Generate Report	Select to generate a Patch Report for that patch.

Output Window

When validation is performed and when the Import, Package Auto Import, Patch Impact Analysis, and Conflict Wizards are run, the output messages and results of those wizards are displayed in the various tabs of the Output Window.

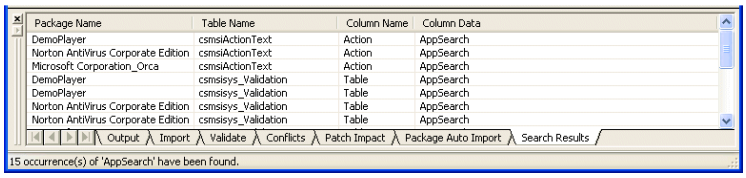
Table 14-85 • Output Window Context Menus

Tab	Description
Output Tab	When the Validation Wizard or Conflict Wizard or Patch Impact Analysis Wizard is run, or when validation is performed during import or by using the Validate Package function, messages about the validation or conflict identification processes are displayed in this tab. When the process is complete, the Errors, Warnings, and Info messages (for Validation only) that were generated are listed in a table format in the Validate or Conflicts tab. Identified patch conflicts are listed in table format in the Patch Impact tab.

Table 14-85 • Output Window Context Menus

Tab	Description
Import Tab	When the Import Wizard is run to import a package into the Application Catalog, messages about the Import process are displayed in this tab.
Validate Tab	After you perform validation, all of the Errors, Warnings, and Info messages that were generated are listed in table format. Each table row lists an icon to indicate whether it is an Error (), Warning (), or Informational message (), the name of the ICE that generated it, and a brief description of what caused it to occur. To sort this list, click on the column heading of the column you want to sort by. ***Note*** • *If you used the Validate Package function to validate an MSI package that has been imported into the Application Catalog, these results are persisted and can be viewed on the Validation View. See Viewing ICE Error Information.*
Conflicts Tab	After the Conflict Wizard is run to identify conflicts between a Windows Installer package and packages already cataloged in ConflictSolver, or if conflicts are checked during import, all of the Errors and Warnings that were generated are listed in a table format. Each table row lists an icon to indicate whether it is an Error () or a Warning (), the name of the ACE that generated it, and a brief description of what caused it to occur. To sort this list, click on the column heading of the column you want to sort by. If a row is active, you can double-click on it to view that row's associated table. The Tables View is launched and the table and/or table cells that are causing the problem are highlighted. If a row is grayed out, it indicates that the table cannot be viewed in ConflictSolver (perhaps because it is in an external package). To resolve conflicts, see Resolving Conflicts.

Table 14-85 • Output Window Context Menus

Tab	Description
Patch Impact Tab	After the Patch Impact Analysis Wizard is run to identify conflicts between a Microsoft patch and a Windows Installer package or OS Snapshot, all of the impacts that were generated are listed in a table format. Each table row lists the following information: • **Patch**—Name of the patch that caused the conflict. • **Product**—Name of the impacted product. • **Type**—The type of impact. • **Description**—Description of the conflict that was found between the patch and the product. The following is an example of a conflict description: `Package 'WindowsXP v5.00' uses version '6.0.9589' of file 'SystemFolder\expsrv.dll' but Patch 'Windows2000-KB837001-x86-ENU' uses version '6.0.72.9589' of the same file` If you double-click on a row in this window, the conflicting file on the target product that is being impacted by the patch will be displayed and highlighted in the Tables View. For further information on the impacts generated by the Patch Impact Analysis, you may want to generate a Patch Report, or view the Product tab's Patch Impacts Analysis View, Dependencies View, and Associated Patches View. See Analyzing the Impact of Installing a Microsoft Operating System Patch for more information.
Package Auto Import Tab	When the Package Auto Import Wizard is run to automatically import packages in a Remote Application Catalog or Network Directory, messages about the import process are displayed in this tab.
Search Results Tab	When Find is used to search for data in Application Catalog tables, the data that is found is displayed in this tab, in the following format:  If you double click on this data, ConflictSolver will navigate to the appropriate record in the Tables View, and that record will be highlighted.

Note • *All Output Window tabs are shown in both the Application Manager View and the ConflictSolver View.*

Views

The following views are associated with the Products tab in ConflictSolver:

- Product View
- OS Snapshot View
- Other Setup Types View
- NCP Views
- App-V Package View
- Patches Tab Views

Product View

When you select a Windows Installer Package (.msi) in the Product View, details about that product are displayed in the right pane of the user interface.

The following information is included:

Table 14-86 • Product View Information

Field	Description
Manufacturer	The manufacturer of the package.
Subscribed	If this package is associated with a subscribed database, the name of that database is listed here. If it is not, the following statement appears: `Not associated with any subscribed database.`
Conflict Results	If any persisted conflict data exists, this is a hyperlink to the package's Conflicts View.
Package Code	The globally unique identifier (GUID) for the setup package.
Product Code	A string that uniquely identifies the product.
Upgrade Code	A string used to upgrade the application. The upgrade code for a package groups that package into a specific product family.
Language	The language of this package.
File	This can be either a hard-coded path or a UNC path.
Imported On	The date and time the package was imported.
Original File	For patches, this is the original MSI package against which patch packages were applied to get to the final imported MSI package.

Table 14-86 • Product View Information (cont.)

Field	Description
Transforms	This can be either a hard-coded path or a UNC path.
Patches	The patch files that were used to get to the final imported MSI package.
Groups	Any groups to which the package belongs.
Description	You can edit this with additional information about the product.

Note • *If the package and/or transforms are no longer in their original import directory, you can locate the file(s) from the provided hyperlink. You are also informed if the last modified date for the package in ConflictSolver does not match the last modified date of the package in its external location. You are given the opportunity to reimport the package to keep it synchronized in ConflictSolver.*

If you click on the plus sign to expand a package in the Product View, a node is listed for each available constituent view. For example, for a Windows Installer package, the following nodes are listed:

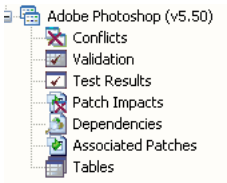


Figure 14-3: Product View Nodes

When you select one of these nodes, a constituent view opens in the right pane:

- Conflicts View
- Validation View
- Test Results View
- Patch Impact View (Products Tab)
- Dependencies View
- Associated Patches View
- Tables View

Note • *The* **Only Display View Nodes With Data** *option on the* **General** *tab of the ConflictSolver* **Options** *dialog box controls whether product nodes (constituent views) appear if no data is contained in that view. If you select the option, products containing views without data will not display those views. For example, if a product has no Dependencies, then the Dependencies node is not displayed for that product.*

Conflicts View

Each Windows Installer and App-V package contains a **Conflicts** view, where you can access information pertaining to the last execution of the Conflict Wizard. This persisted conflict data allows you to view when the last execution was performed, the products that it was run against, the ACE rules used and conflicts discovered. The view also provides a list of updated, deleted, or added products that may necessitate performing conflict identification again.

Summary Information

The main area of this view displays summary information for the last conflict identification execution. This information includes:

Table 14-87 • Conflicts View Summary Information

Item	Description
[Name of Package] Conflict View	The name of the product on which conflict identification was performed is listed above **Conflict View** in the title bar.
[Date and Time]	The date and time of the last conflict identification.
Resolution Options	Lists the currently selected resolution option. Click **Edit** to open the **Resolution Options** dialog box, where you can choose whether to save resolutions in the original Windows Installer package or in a new transform file. ***Note*** • *Displayed for Windows Installer packages only.*
Total number of conflicts	The number of conflicts detected. ***Note*** • *Click this link to automatically select **Show all conflicts** from the **Conflicts** filter list.*
Suppressed conflicts	The number of detected conflicts which have been suppressed. ***Note*** • *Click this link to automatically select **Show all conflicts** from the **Conflicts** filter list.*

Table 14-87 • Conflicts View Summary Information (cont.)

Item	Description
Automatically resolvable conflicts	The number of detected conflicts which can be resolved automatically. ***Note*** • *Displayed for Windows Installer packages only.* ***Note*** • *Click this link to automatically select **Show automatically resolvable conflicts** from the **Conflicts** filter list.*
Manually resolvable conflicts	The number of conflicts which can be resolved manually. ***Note*** • *Displayed for Windows Installer packages only.* ***Note*** • *Click this link to automatically select **Show manually resolvable conflicts** from the **Conflicts** filter list.*
Resolved conflicts	The number of detected conflicts which have been resolved. ***Note*** • *Displayed for Windows Installer packages only.* ***Note*** • *Click this link to automatically select **Show resolved conflicts** from the **Conflicts** filter list.*
ACE rules evaluated	Click this link to open the **ACE Rules** tab of the Conflict Information Dialog Box with a list of the ACE rules used during conflict identification.
Packages evaluated	Click this link to open the **Products** tab of the Conflict Information Dialog Box with a list of the packages that were included in conflict identification.

Also, the following information may also be listed:

- The number of checked packages that have changed since conflict identification.
- The number of checked packages that have been deleted from ConflictSolver since conflict identification.
- The number of packages that have been imported into ConflictSolver since conflict identification.

Links

If you click on any of the conflict links, the list will be filtered by that type of conflict. For example, if you click on the **Automatically resolvable conflicts** link, only automatically resolvable conflicts will appear in the list.

If you click on the **ACE rules evaluated** or **Packages evaluated** link, the Conflict Information Dialog Box opens, displaying information on the selected link. For example, if you click on the **ACE rules evaluated** link, the **Conflict Information** dialog box opens with the **ACE Rules** tab selected.

Conflicts List

A list of the conflicts found during conflict analysis is displayed. You can filter this list by making selections from the list, such as **Show all conflicts**, **Show automatically resolvable conflicts**, etc.

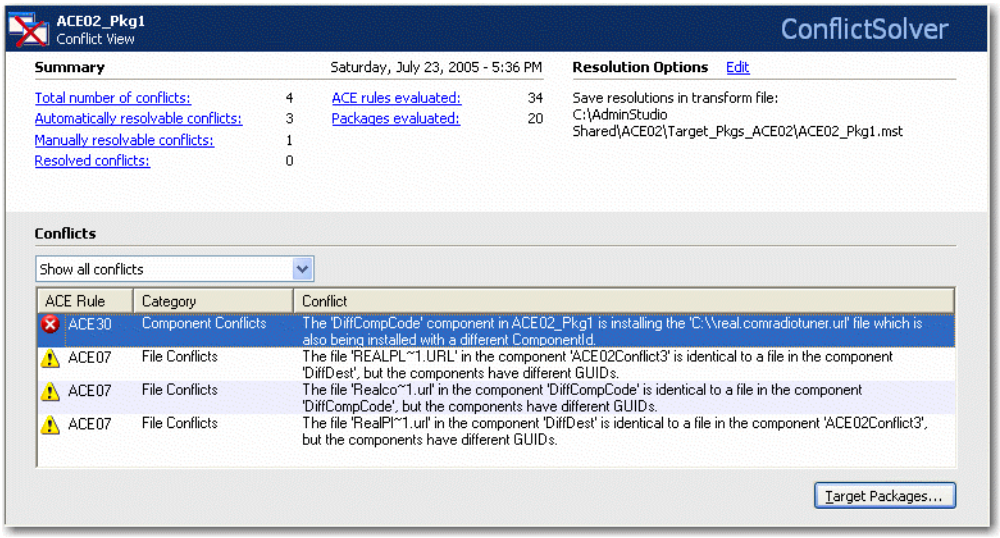


Figure 14-4: Conflict View

Note • *Automatic conflict resolution is supported for Windows Installer packages, but not for App-V packages.*

The icon displayed next to each conflict indicates its status:

Table 14-88 • Conflict Status Icons

Icon	Status
	Resolvable
	Resolved
	Unresolvable

For each conflict, the ACE Rule number, the ACE category, and a description of the conflict is listed.

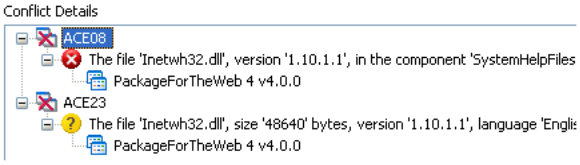
When evaluating conflicts between one source package and multiple destination packages, the same conflict may be detected between the source package and more than one destination package. These conflicts are consolidated and displayed as single entry. To see a list of all of the target packages that detected the same conflict, click the **Target Packages** button to view a list of the package on a pop-up dialog box.

In many cases, these errors can be resolved by clicking either **Resolve** or **Resolve All**. In cases where no automatic resolution is possible, the Unresolvable icon ( ) appears next to the conflict and the **Resolve** button is disabled.

If one of the following options on the Resolution Tab of the **Options** dialog box is selected, several additional dialog boxes could appear prompting for approval or additional information:

- If the **Preview and approve all resolutions** option is selected, the **Resolution Details** dialog box appears, listing the changes that are going to be made and prompting you to approve them.
- If the **Maximize user flexibility in selecting resolutions** option is selected, and if you are using a CARD with multiple resolution options, the **CARD Resolutions** dialog box would display a list of resolutions and would prompt you to select one.

For File Conflicts, click the plus sign next to the conflict icon to see the package that conflicts with the selected package.



Resolution Options

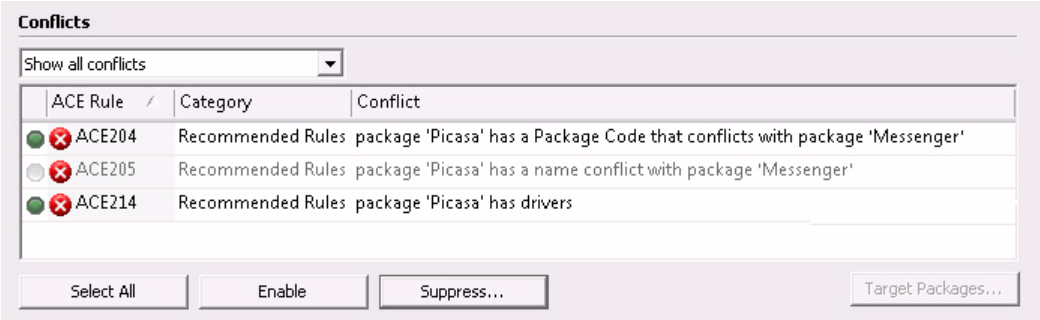
This section displays the current resolution option: either resolve in the original Windows Installer package, or using the specified transform. Click **Edit** to change the current settings on the Resolution Options Dialog Box.

Note • *Automatic conflict resolution is not supported for App-V packages. Therefore, the* ***Resolution Options*** *section is not displayed on the* ***Conflicts View*** *for an App-V package.*

Suppressing / Enabling ACE Errors

If you do not want ConflictSolver to check for a particular ACE error or warning during subsequent analyses of a package, you can choose to suppress it by selecting it and clicking the **Suppress** button.

Suppressed ACE errors are listed in gray, and are preceded by a white icon, while enabled ACE errors are preceded by a green icon:



Validation View

The Validation View, which is displayed when you select the Validation node under a Windows Installer package on the Product View, lists all ICE Errors and Warnings that were generated the last time the selected package was validated. If a package has not been validated, no ICE errors will be listed.

Note • *To validate a package that has already been imported into the Application Catalog, use the Validate Package function available on the Product View.*

ICE errors include an icon to indicate whether it is an Error (⊗) or a Warning (⚠), the name of the ICE that generated it, and a brief Description of what caused it to occur:

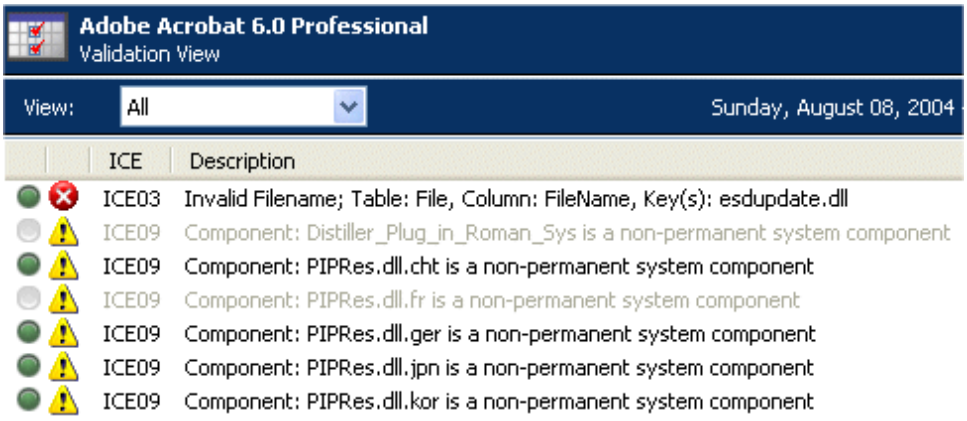


Figure 14-5: Example of ICE Error Listing

You can sort by any column by clicking the appropriate column title.

Viewing ICE Error Information

To easily view the full text of the ICE error description, select an ICE Error or Warning number and then select Properties from the context menu. The Properties Dialog Box opens listing the ICE error number and its full description. For more information, see Viewing ICE Error Information.

Suppressing an ICE Error

If you do not want ConflictSolver to check for a particular ICE error during subsequent validations of a package, you can choose to suppress an ICE error. See Suppressing an ICE Error for more information. Suppressed ICE errors are listed in gray:

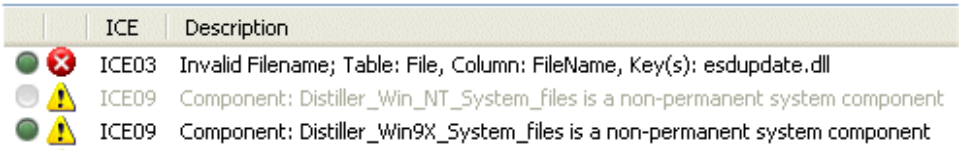


Figure 14-6: ICE Error Listing with a Suppressed ICE Error (in gray)

You can make a selection from the View list to filter the list by Enabled or Suppressed status.

Deleting an ICE Error

You can permanently delete an ICE error from subsequent execution and reporting for that package. ICE errors that are deleted are no longer listed in the Validation View and will not be listed in the Package Report. See Deleting an ICE Error for more information.

Test Results View

You can select a package in ConflictSolver and then choose to test the package using PackageExpert. Test results will be displayed in the ConflictSolver **Test Results** view.

For more information, see Testing Packages From ConflictSolver and Viewing Test Results in ConflictSolver.

Patch Impact View (Products Tab)

The **Patch Impact View** is displayed when you select the **Patch Impacts** node under a product on the **Products** tab.

Summary View

The information displayed on the **Patch Impact View** is dependent upon the selection made in the **Impact category** list. The **Summary View**, which is displayed when **Summary** is selected from the **Impact category list**, displays a list of patches for which there is patch impact data persisted against the product. The following information is displayed:

Table 14-89 • Summary View Information

Option	Description
Id	Number identifying this patch's associated Microsoft Security Bulletin.
Name	Name of the patch file.
Title	Title of the patch.

Table 14-89 • Summary View Information

Option	Description
Release Date	Date this patch was published by Microsoft.

File Impacts View

The **File Impacts View**, which is displayed when **File Impacts** is selected from the **Impact category** list and you have file impacts persisted, lists all impacts against this product or OS Snapshot and identifies the patch that caused the impact. If you double-click on one of the patches, the Patch View for that patch will open. The following information is displayed:

Table 14-90 • File Impacts View Information

Option	Description
Description	Description of the impact.
Id	Number identifying this patch's associated Microsoft Security Bulletin.
Name	Name of the patch file.
Title	Title of the patch.

Note • *The* **Patch Impact View** *on the Products tab is a corollary to the Impact Analysis View (Patches Tab).*

Dependencies View

On the **Dependencies View**, which is accessed by selecting the **Dependencies** node under a Windows Installer package in the Product View, you can view a list of all of the files of a selected package that have dependencies with files used by other packages or operating systems in the Application Catalog. This view is displayed for Windows Installer `.msi` packages in which file dependency information exists.

Dependency information is generated by the Patch Impact Analysis Wizard. You can also generate this information by selecting a Windows Installer package in the Product View and then selecting **Scan for Dependencies** from the context menu.

Note • *If the* ***Only Display View Nodes With Data*** *option on the* ***General*** *tab of the ConflictSolver* ***Options*** *dialog box is selected, if you scan for dependencies and no dependencies are found, the* ***Dependencies*** *node will still not be displayed.*

The following information is displayed:

Table 14-91 • Dependencies View Information

Option	Description
Files With Dependencies	Make selections from this list to further filter the output listing, or select (All) to display all dependencies.
File	Name of a file included in this product that is dependent upon a file used by another product or operating system in the Application Catalog.
Dependent File	Name of the file that this product's file is dependent upon.
Size	Size of the dependent file. This field is populated for file dependencies identified from the MSI package itself. For dependencies that were derived from the static scanning process, this field will be undefined.
Version	Version of the dependent file. For dependencies that were derived from the static scanning process, this field will be undefined.
Language	Language of the dependent file. For dependencies that were derived from the static scanning process, this field will be undefined.

Associated Patches View

On the Associated Patches View, you can view a list of imported patches that, if installed, would update the selected product. ConflictSolver examines the patches in the catalog and attempts to identify those patches which will impact this package.

Due to differences in the way versions are compared, it is possible that other patches that impact this package may exist. For more definitive information, open the **Patch Properties** dialog box and compare the product and OS snapshot version information of the patch against the specific product and version information.

In the Associated Patches View, the following information is displayed:

Table 14-92 • Associated Patches View Information

Option	Description
Name	Name of the patch that is associated with this product.
Information	Microsoft Security Bulletin identification number and description of the patch.

If you double-click on a patch in the Associated Patches View, the Patch View (on the Patches tab) for that patch opens, listing general information on the selected patch.

Tables View

The Tables view provides a way to view the data for a given package in the Application Catalog. Select the specific table you want to view from the Tables list at the top of the view.

Most tables are derived directly from standard MSI tables, as described in the Windows Installer SDK online help.

When building your own ACE rules to use for conflict identification, it is important to understand the data available for packages so you can construct the necessary rule.

OS Snapshot View

When you click on an OS Snapshot in the Product View, details about the snapshot appear in the right pane of the user interface.

The following information is displayed:

Table 14-93 • OS Snapshot View Information

Field	Description
Version	Version of the operating system of the OS Snapshot, such as Windows XP - 5.1.2600.
Language	The language the operating system was written for.
File	This can be either a hard-coded path or a UNC path.
Imported On	The date and time the OS Snapshot was imported.
Description	You can edit this with additional information about the OS Snapshot.

Click the plus sign next to the OS Snapshot icon to view these OS Snapshot constituent views:

- Tables View for OS Snapshots

Tables View for OS Snapshots

The Tables View for an OS Snapshot is identical to the Tables View shown when a Product is selected. See Tables View in the Product View section.

Other Setup Types View

When you click on an Other Setup Types package (a package containing non-MSI based setup files) in the Product View, details about that setup are displayed in the right pane of the user interface.

The following information is included:

Table 14-94 • Other Setup Types View Information

Field	Description
Main Directory	The original location of the non-MSI based setup files when they were imported into ConflictSolver.
Imported On	The date and time the setup was imported.
Files	A listing of the individual files making up the setup are listed, allowing you to view or delete files by making a selection from the context menu. • **To view the contents of a file**, select **View** from the context menu. You can only view files supported by applications installed on your workstation. If you attempt to view a non-supported file, no View window will appear. If you attempt to view an executable, that executable will be launched. • **To delete a file from this setup**, select **Delete** from the context menu.

If you expand an Other Setup Type, you can view its constituent view, the Tables View for Other Setup Types.

Tables View for Other Setup Types

The Tables View for an Other Setup Types package is identical to the Tables View shown when a Product is selected. See Tables View in the Product View section.

NCP Views

When you select a Marimba NCP file (.ncp) from the Product View, details about that file are displayed in the right pane of the user interface.

The following information is included:

Table 14-95 • NCP Views Information

Column	Description
Conflict Results	Date that this package was last checked for conflicts. Click the link to jump to the Conflicts View for Marimba NCP Files.
Version	Version of the software package
File	Name and path of the NCP file that was imported
Imported On	Date when the file was imported
Description	After the NCP file is imported, you can edit this with additional information.

If you expand an NCP file, you can view its constituent views:

- Conflicts View for Marimba NCP Files
- Tables View for Marimba NCP Files

Conflicts View for Marimba NCP Files

The Conflicts View for an NCP File is identical to the Conflicts View shown when a Product is selected. See Conflicts View in the Product View section.

Tables View for Marimba NCP Files

The Tables View for an NCP file is identical to the Tables View shown when a Product is selected. See Tables View in the Product View section.

App-V Package View

When you select an App-V package in the Application Catalog, the **App-V Package View** opens. The following information is displayed:

Table 14-96 • App-V Package View

Option	Description
Type	Identifies the virtual package type as **App-V**.
File	Identifies the location of this package. It can be either a hard-coded path or a UNC path. If the package is part of the Software Repository, the following statement appears: `Managed within the Software Repository`
Imported On	Date and time the virtual package was imported into the Application Catalog.
Applications	Lists the Applications that reference this App-V package.

Table 14-96 • App-V Package View

Option	Description
Associated MSI Packages	Lists the App-V package's source Windows Installer package. By associating a virtual package with the Windows Installer package which originated it, you have the convenience of being able to easily locate the virtual package's originating Windows Installer package, modify the original Windows Installer package, and then regenerate the virtual package. ***Note*** • *To quickly locate one of these listed Windows Installer packages in the Application Catalog, double-click on it.* In order to be listed here, the Windows Installer package has to have already been imported into the Application Catalog and must be associated with this virtual package. There are several ways to associate a virtual package with its source Windows Installer package: • Importing a Virtual Package During the Import of its Source Windows Installer Package • Importing a Virtual Package After the Import of its Source Windows Installer Package • Manually Associating a Virtual Package with a Windows Installer Package
Description	You can edit this field and add additional information about the virtual package.
View Package Version History	If you are connected to a Software Repository-enabled Application Catalog and you select a package that has more than one version, a **View Package Version History** link appears in the header of this view. If you click on this link, the **Package Versions** dialog box opens, listing all of the versions of the selected package.

If you expand the App-V node in the tree, you can view its constituent views:

- **Conflicts**—See App-V Conflict View.
- **Dependencies**—See App-V Dependencies View.
- **Tables**—See App-V Tables View.

App-V Conflict View

When you expand an App-V node in the Application Manager tree, you can click on **Conflicts** to display the **Conflict View** for an App-V package.

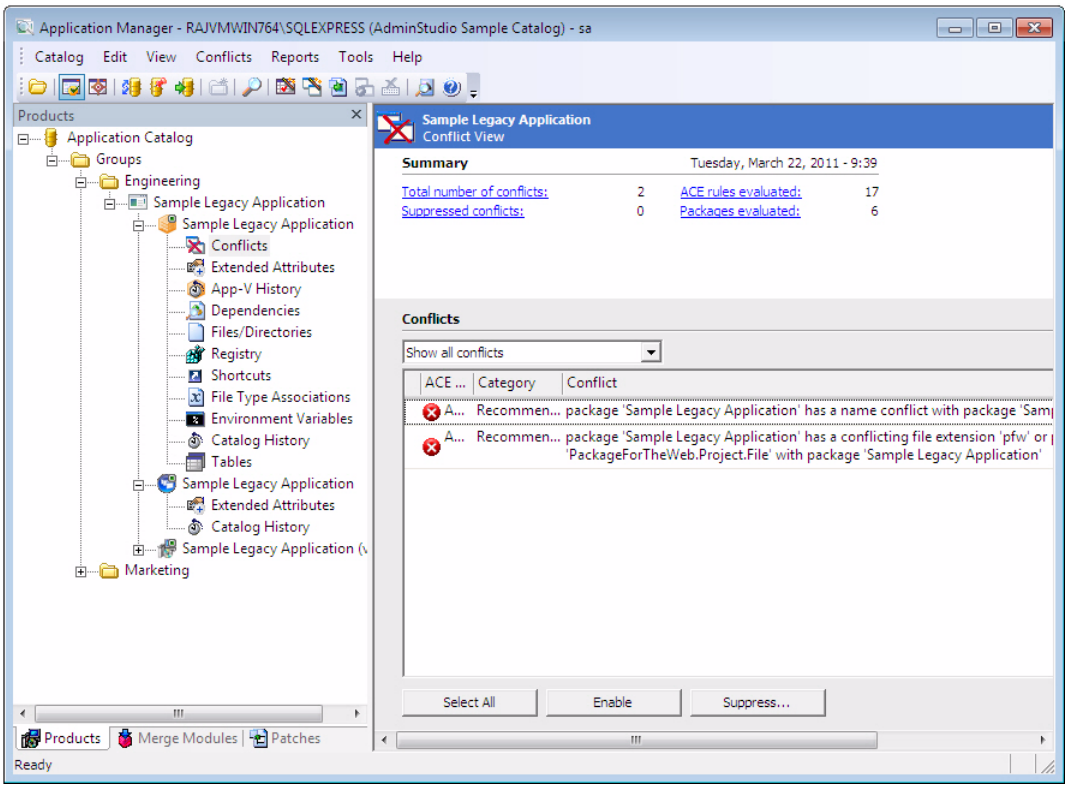


Figure 14-7: App-V Conflict View

On this view, you can access information pertaining to the last execution of the Conflict Wizard. This persisted conflict data allows you to view when the last execution was performed, the packages that it was run against, the ACE rules used and conflicts discovered. The view also provides a list of updated, deleted, or added packages that may necessitate performing conflict identification again.

The information displayed on this view is the similar to the **Conflict** view for Windows Installer packages. See Conflicts View.

App-V Dependencies View

When you expand an App-V node in the Application Manager tree, you can click on **Dependencies** to display a listing of both the applications this package is dependent on and the applications dependent upon this application.

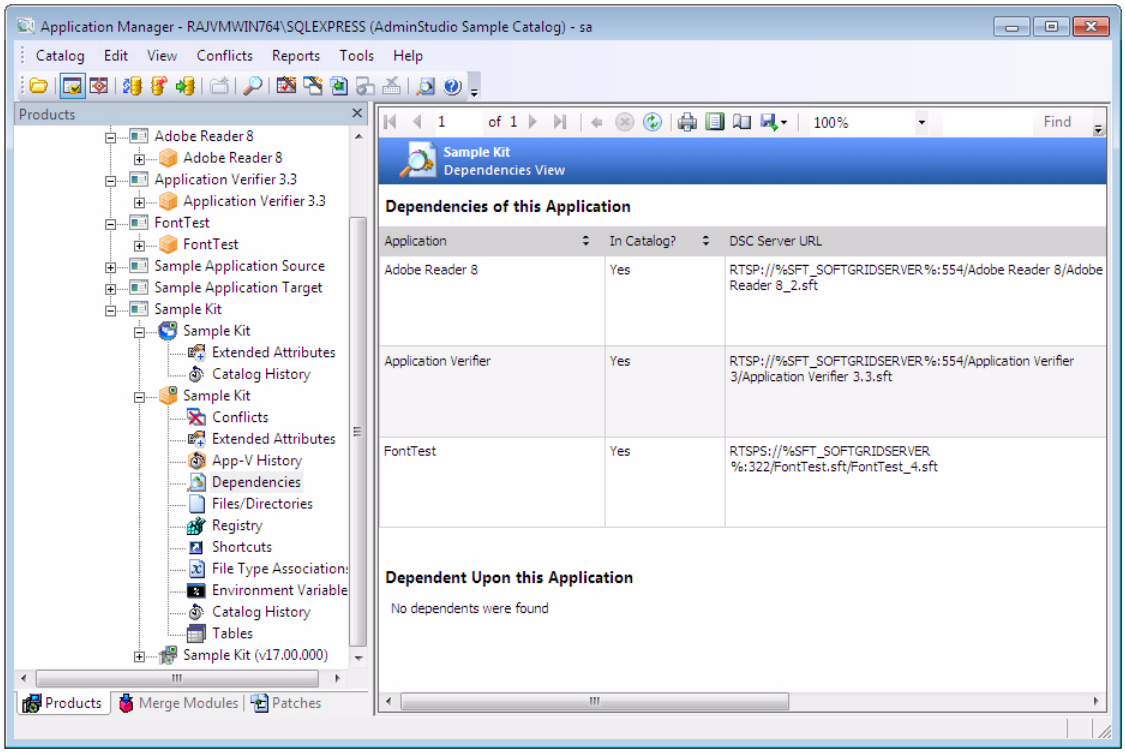


Figure 14-8: App-V Dependencies View

For each dependency, the following information is listed:

- Application
- In Catalog? (Yes / No)
- DSC Server URL
- Server URL
- Mandatory? (Yes / No)

App-V Tables View

When you expand an App-V node in the Application Manager tree, you can click on **Tables** to display the **Tables View**, which provides a way to view the data for this specific App-V package. Select the specific table you want to view from the **Tables** list at the top of the view.

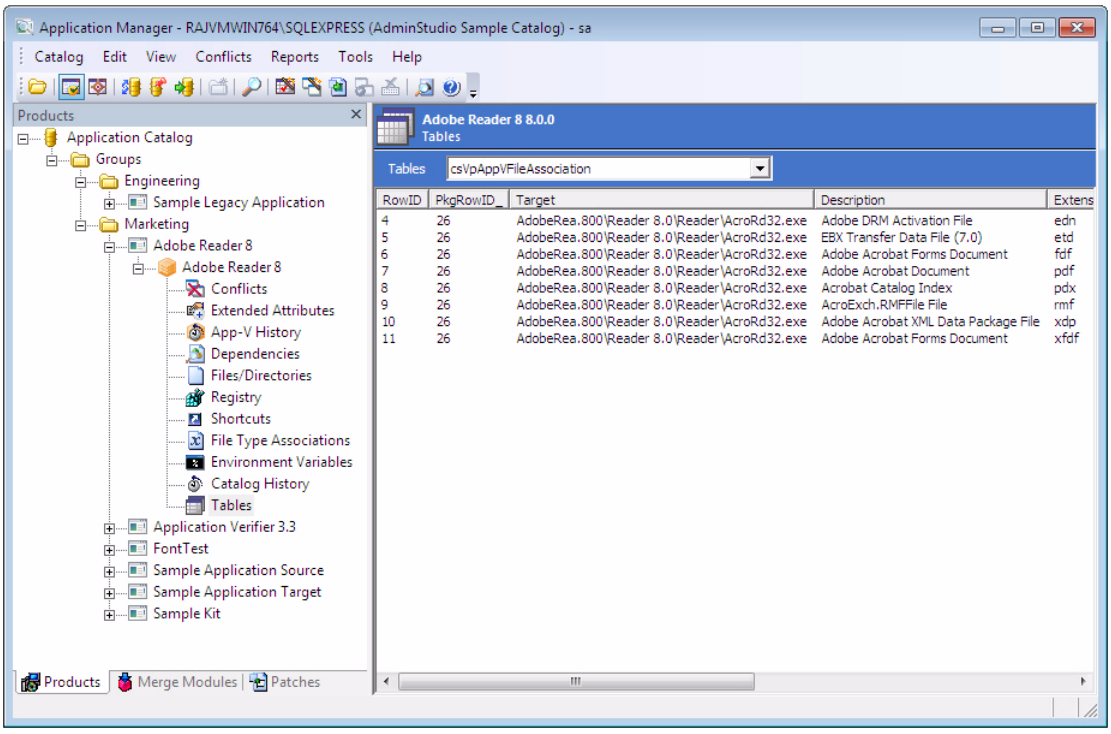


Figure 14-9: App-V Tables View

When building your own ACE rules to use for conflict identification, it is important to understand the data available for App-V packages so you can construct the necessary rule.

Patches Tab Views

The Patches tab includes the following views which provide content and impact information about Microsoft patches that have been imported into the Application Catalog.

Table 14-97 • Patches Tab Views

View	Description
Patches Group View	Opens when the root group in the **Patches** tab is selected, and lists all of the groups that have been created in the **Patches** tab.
New Patches Group View	All new patches are imported into the **New Patches** group, and this view lists all of the patches in that group.

Table 14-97 • Patches Tab Views (cont.)

View	Description
Group View of a Selected Group	Opens when a group other than the root group in the **Patches** tab is selected. For each selected group, a list of all of the patches in that group is displayed.
Patch View	Lists general content information on a selected patch.
Impact Analysis View (Patches Tab)	Opens when the **Impacts** node under a Patch on the **Patches** tab is selected. Displays a list of patches for which there is patch impact data persisted against a product. It also lists all impacts against a product, and identifies the patch that caused the impact.

Our patches tree structure would then look and operate much the same as our Group - Package tree including a RC menu. Drag-Drop support would be provided for the groups and patches in much the same way as for groups and products.

Patches Group View

The Patches Group View opens when the root group in the **Patches** tab is selected. The Patches Group View lists all groups that have been created in the Patches tab.

All new patches are imported into the **New Patches** group, and then you can organize the patches into other groups according to your business needs. See Organizing Your Application Catalog Using Groups.

The **New Patches** group is automatically created during installation. While it can be renamed, it cannot be deleted.

Context Menu Options

When the root group in the Patches Tab is selected, the following items are available on the context menu:

- **Refresh**—Refresh the patch listing to reflect the most recent modifications.
- **Import Patches**—Opens the OS Security Patch Wizard interface so that you can import patches into the Application Catalog.
- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Properties**—Open the **Group Properties** dialog box.

New Patches Group View

The New Patches Group View opens when the **New Patches** group on the Application Manager or ConflictSolver **Patches** tab is selected. All new patches are imported into the **New Patches** group. You can then organize the patches into other groups according to your business needs. See Organizing Your Application Catalog Using Groups.

The **New Patches** group is automatically created during installation. While it can be renamed, it cannot be deleted.

The New Patches Group View displays a list of all of the patches in that group, including the following information:

Table 14-98 • New Patches Group View Information

Option	Description
Name	Name of patch file.
Description	Description of the patch file.
Release Date	Date the patch was released by Microsoft.
Import Date	Date the patch was imported into the Application Catalog.

If you select a patch in this list, detailed patch properties are displayed in the area to the left of the list, including the **Title**, **Summary**, and **Release Date** of the patch.

Context Menu Options

When the **New Patches** group in the **Patches Tab** is selected, the following items are available on the context menu:

- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Properties**—Open the **Group Properties** dialog box.

Group View of a Selected Group

The Group View of a selected group opens when a group other than the root group or the **New Patches** group in the **Patches** tab is selected. For each selected group, a list of all of the patches in that group is displayed, including the following information:

Table 14-99 • Group View of a Selected Group Information

Option	Description
Name	Name of patch file.
Description	Description of the patch file.
Release Date	Date the patch was released by Microsoft.
Import Date	Date the patch was imported into the Application Catalog.

If you select a patch in this list, detailed patch properties are displayed in the area to the left of the list, including the **Title**, **Summary**, and **Release Date** of the patch.

Context Menu Options

When a group other than the root group or the **New Patches** group in the **Patches Tab** is selected, the following items are available on the context menu:

- **New Group**—Create a new group.
- **Rename**—Rename the selected group.
- **Delete**—Delete the selected group.
- **Cut**—Copy the selected group to the clipboard so that you can remove it from its current location and paste it into another location.
- **Paste**—Insert the patch or the group on the clipboard into the selected location.
- **Properties**—Open the **Group Properties** dialog box.

Patch View

The **Patch View**, which is displayed when a patch is selected on the **Patches** tab, lists general information on the selected patch. The following information is included:

Table 14-100 • Patch View Information

Option	Description
ID	Microsoft Security Bulletin ID. Click this link to view this bulletin on the Microsoft Web site.
Title	Title of patch.
Release Date	Date the patch was released by Microsoft.
KB Article	Microsoft Knowledge Base article ID. Click this link to view this article on the Microsoft Web site.
Imported On	Date patch was imported into the Application Catalog
Groups	List of all of the groups that this patch is included in.
Description	You can enter a description of the patch in this field.

In the Application Manager, you can view additional detailed patch information by selecting a patch and then selecting **Properties** from the context menu.

Context Menu Options

When a patch in the **Patches Tab** is selected, the following items are available on the context menu:

- **Rename**—Rename the selected patch.
- **Delete**—Delete the selected patch.
- **Cut**—Copy the selected patch to the clipboard so that you can remove it from its current location and paste it into another location.
- **Copy**—Copy the selected patch to the clipboard so that you can paste a copy of it into another location.
- **Paste**—Insert the patch on the clipboard into the selected location.
- **Generate Report**—Generate a Patch Impact Analysis Report for that patch. See Generating the Patch Report.
- **Properties**—Open the **Patch Properties** dialog box for that patch.

Impact Analysis View (Patches Tab)

The **Impact Analysis View** opens when you select the **Impacts** node under a patch on the ConflictSolver **Patches** tab. The information displayed on this view is dependent upon the selection made in the **Impact category** list.

Summary View

The **Summary View**, which is displayed when **Summary** is selected from the **Impact category** list, lists details regarding the last time impact analysis was performed, including:

- **Products that were checked for impacts** for the selected patch during patch impact analysis.
- **Products that have been updated** since the last impact analysis was performed.
- **Products that have been removed** from the Application Catalog since the last impact analysis was performed.
- **Products that have been added** to the Application Catalog since the last impact analysis was performed.

If one of these four categories include one or more products, a link to the Patch Impact Information Dialog Box is provided, which lists information on individual tabs for each of the categories. If a category does not include any products, its tab is not displayed.

Click **Run Again** to run patch impact analysis based upon the selections that were previously chosen in the Patch Impact Analysis Wizard. Analysis messages are listed in the **Output** tab of the **Output Window**. When analysis is complete, patch conflicts are listed on the **Patch Impact** tab in table format. See Analyzing the Impact of Installing a Microsoft Operating System Patch for more information.

File Impacts View

On the **File Impacts View**, which is displayed when **File Impacts** is selected from the **Impact category** list, all impacts against this patch are listed, and the product(s) that the patch impacted are identified. The following information is displayed:

Table 14-101 • File Impacts View Information

Option	Description
Description	Description of the impact.
Package Name	Name of the product that this patch impacted.

If you double-click on one of the listed products, that product will be opened in the Tables View with the row that is causing the problem highlighted.

Note • *The* **Impacts Analysis View** *on the* **Patches** *tab is a corollary to the Patch Impact View (Products Tab).*

Dialog Boxes

The following dialog boxes are accessible from ConflictSolver:

- About ConflictSolver Dialog Box
- ACE Rule Properties Dialog Box
- Add Ignore Table Dialog Box
- CARD Resolution Options Dialog Box
- Conflict Information Dialog Box
- Conflict Resolution Dialog Box
- Customize Dialog Box
- Expression Builder Dialog Box
- Options Dialog Box
- Patch Impact Information Dialog Box
- Resolution Details Dialog Box
- Resolution Operations Dialog Box
- Resolution Options Dialog Box
- Rules Viewer Dialog Box

About ConflictSolver Dialog Box

The About InstallShield ConflictSolver dialog box can be accessed by selecting About ConflictSolver from the Help menu. This dialog box displays information about the product, including the full version number (essential if you need technical support).

ACE Rule Properties Dialog Box

The ACE Rule Properties dialog box allows you to edit an existing user-defined ACE rule. You can display the dialog by clicking Edit on the on the Rules Viewer dialog box.

The following tabs are part of the ACE Rule Properties dialog box:

- General Information Tab
- Additional Information Tab
- Custom Options Tab
- Where Clause Tab
- DLL Information Tab

Note • *You can only edit user-defined ACE rules; you are not permitted to edit the ACE rules that were installed with ConflictSolver.*

General Information Tab

From the General Information tab, you can configure information about the new ACE rule.

This information is used primarily for display information (Name, Brief Description, Description, and Information URL).

Table 14-102 • General Information Tab Information

Option	Description
Name	The name of the ACE, used to organize the rule in ConflictSolver. This is displayed in several places, including the Output Window, the Rules Viewer dialog box, and the Conflicts tab of the Options dialog box.
Associated Table	Select the table in the Application Catalog which will be queried in the user-defined ACE. This also determines which columns are available in the Expression Builder dialog box, and which tokens are available for the Error and Display strings on the Custom Options Panel of the Rules Wizard.
Package Type	Select **MSI** or **App-V** to identify the type of package that this rule will be run on.

Table 14-102 • General Information Tab Information

Option	Description
Brief Description	Enter a brief description which will be displayed in the Rules Viewer dialog box, the Conflicts tab of the Options dialog box, and in the Output Window. This description should be clear enough so users can understand when to use this ACE.
Description	Enter a description of the ACE, which is displayed at the bottom of the Rules Viewer dialog box when the ACE is selected and in the Output Window during conflict identification when the ACE executes.
Information URL	Provide a URL to get further information for the ACE. This URL appears in the Conflict Details area of the Conflicts View after conflicts have been identified.

To continue editing ACE Properties, click the Additional Information, Custom Options, Where Clause, or DLL Information tabs. To save your edits and close this dialog box, click OK.

Additional Information Tab

From the Additional Information tab, you can edit information for categorizing the ACE in relation to other ACEs in ConflictSolver.

Table 14-103 • Additional Information Tab Information

Option	Description
Category	Either select an existing category for this new rule, or enter the name for a new category. These categories are displayed in the Conflict View and the Conflicts tab of the Options dialog box. Ideally, any user-defined ACEs should be put in their own category.

Table 14-103 • Additional Information Tab Information

Option	Description
Rule Type	The Rule Type of this ACE is displayed (read only). • **Custom - Source Only Packages ACEs** allow you to quickly test any column or any value of a table. For example, you could use a user-defined ACE to identify packages that create a desktop icon. To define a user-defined ACE, you use an SQL "Where" clause. • **Custom - Source and Target Packages ACEs** allow you to compare columns or values of Source package tables (new packages that you want to install onto a user's system) to columns or values of Target package tables (packages already installed on a user's system). For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would produce duplicate registry entries. To define a Source and Target Packages ACE, you must define an SQL "Where" clause, and specify a Join Column (a table column in the Application Catalog database that has a matching value for both the Source and Target packages). • **DLL - User Provided DLL based ACEs** allow you to run more complex tests—testing many tables in any combination. For example, you could use a DLL-Based ACE to confirm that a source product language is the same as all target product languages. To define a DLL-Based ACE, you use SQL and other programming commands. With DLL-Based ACEs, you can use a Conflict Application Resolution Definitions (CARDs) to fix the conflict.

To continue editing ACE Properties, click the General Information, Custom Options, Where Clause, or DLL Information tabs. To save your edits and close this dialog box, click OK.

Custom Options Tab

From the Custom Options tab, you can edit this ACE's display strings for the Output window and Conflict Details.

The following options are included:

Table 14-104 • Custom Options Tab Properties

Option	Description
Error String	This string appears in the Output window when a violation of this ACE rule is detected during conflict identification. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter the following in this field: `Failure in creating desktop icon` ***Note*** • *Tokens allow you to insert values at run-time from the installation package into the string, such as specifying a file name. To use token replacement in the error string, use the arrow to the right of the Error String field. For more information, see Token Grammar.*

Table 14-104 • Custom Options Tab Properties (cont.)

Option	Description
Display String	This string appears in the Conflict Details area of the Conflicts View after conflicts have been identified. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter the following in this field: `Duplicate desktop icon found` ***Note*** • *Tokens allow you to insert values at run-time from the installation package into the string, such as specifying a file name. To use token replacement in the error string, use the arrow to the right of the Display String field. For more information, see Token Grammar.*
Severity	Specify whether this ACE should be an Error or a Warning.
Report 'No' results	User-defined ACEs report conflicts based on the provided query. However, you may want to report the absence of the data if it could not be found. If you select this option, if the ACE does not return any results, it will be reported as an error (or warning), with the description and error strings as specified. If you expect a **No** result, do not use tokens in your display or error strings.

To continue editing ACE Properties, click the General Information, Additional Information, or Where Clause tabs. To save your edits and close this dialog box, click OK.

Where Clause Tab

From the Where Clause tab, you can edit the Where clause for the ACE. If you do not know how to build a Where clause, you can click the Build Expressions button to launch the **Expression Builder** dialog box. You can also click Test to validate the Where clause syntax.

If you selected Custom - Source and Target Packages when you created this ACE, you must have also selected a Join Column—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages. For example, if you wanted to evaluate records from two tables that have a installation directory of `C:\ProgramFiles`, then you would specify Directory as the Join column. To change the Join Column, select a different column name from the list.

To continue editing ACE Properties, click the General Information, Additional Information, or Custom Options tabs. To save your edits and close this dialog box, click OK.

Tip • *To improve query performance, enclose table names in square brackets ([]).*

DLL Information Tab

From the **DLL Information** tab, you can edit specific information about the ACE/CARD DLL file and the ACE and CARD Function Names that DLL-based ACEs require to operate.

The following options are included:

Table 14-105 • DLL Information Tab Information

Option	Description
ACE/CARD DLL File	Select the name of the ACE DLL that you are testing.
ACE Function Name	Enter the name that you chose to “export” for this ACE function.
CARD Function Name	Enter the name that you chose to “export” for this CARD function.
Test	Click the Test button next to the ACE Function Name or CARD Function name to validate that the exported function does exist.

To continue editing ACE Properties, click the General Information or Additional Information tabs. To save your edits and close this dialog box, click OK.

Add Ignore Table Dialog Box

This dialog box allows you to specify a custom table to ignore during import of a package into ConflictSolver. You can also provide comments about the table.

Ignored tables will not automatically be imported when using the Import Custom Tables option in the Import Tab of the Options dialog box.

CARD Resolution Options Dialog Box

Some CARDs can resolve a conflict in more than one way. For example, ACE06 checks whether the executable module within the component is the key file, and accumulates a list of potential files to be used for the key file. Any one of these files could be designated as the key file to resolve the conflict.

If a CARD can be resolved in multiple ways, and if the Maximize user flexibility in selecting resolutions option is selected on the Resolution Tab of the **Options** dialog box, the CARD Resolution Options dialog box opens, prompting you to make a selection to determine how conflict resolution will proceed.

Conflict Information Dialog Box

When you click on one of the informational links in the Summary section of either the Conflicts View or Conflict Resolution Dialog Box, the Conflict Information dialog box opens, displaying information on the selected link. For example, if you click on the ACE rule link, the Conflict Information dialog box opens with the ACE Rules tab selected.

The dialog box contains the following tabs (which are only displayed if they have data):

Table 14-106 • Conflict Information Dialog Box Tabs

Tab	Description
Products	The product(s) against which conflict identification was performed. The name, description, and any groups the products belong to are displayed.
ACE Rules	The category and description for each ACE rule used in conflict identification.
Updated Products	The name, description, and modification date of any relevant product that has been updated since the last conflict identification was performed.
Deleted Products	The name and description of any relevant product that has been removed from ConflictSolver since the last conflict identification was performed.
New Products	The name, description, and import date of any product that has been added to ConflictSolver since the last conflict identification was performed.

Note • *This dialog box is similar in function to the Patch Impact Information dialog box.*

Conflict Resolution Dialog Box

You can display the **Conflict Resolution** dialog box either by running the Conflict Wizard and selecting the **Run Resolution** option in the **Conflict Rules** panel, or by selecting **Resolve External Package Conflicts** from the **Conflicts** menu after a previous conflict identification between an external package and products in ConflictSolver.

Tip • *It is strongly recommended that you import packages into ConflictSolver prior to performing conflict identification, instead of running conflicts between an external package and products in ConflictSolver.*

From this dialog box, you can access information pertaining to the last execution of the Conflict Wizard. This persisted conflict data allows you to view when the last execution was performed, the products that it was run against, the ACE rules used, and conflicts discovered. The view also provides a list of updated, deleted, or added products that may necessitate performing conflict identification again.

Summary Information

The main area of this view displays summary information for the last conflict identification execution. This includes:

- The name of the product on which conflict identification was performed.
- The date and time of the last conflict identification.
- The number of conflicts detected.

- The number of detected conflicts which can be resolved automatically.
- The products the current product was checked against.
- The ACE rules used during conflict identification.
- The number of checked packages that have changed since conflict identification.
- The number of checked packages that have been deleted from ConflictSolver since conflict identification.
- The number of packages that have been imported into ConflictSolver since conflict identification.

Links

When you click on one of the informational links, the Conflict Information Dialog Box appears, displaying information on the selected link. For example, if you click on the ACE rule link, the Conflict Information dialog box opens with the ACE Rules tab selected.

Run Again

Click Run Again to perform conflict identification on the package again, using the specified packages (if they still exist in ConflictSolver) and ACE rules. Because packages may have been removed or modified, conflict identification may return different results.

Conflict Information

Using the pull-down list at the top of the view, you can toggle between summary information and conflict-specific information. When you select a specific conflict type, the view changes to display a list of corresponding errors in the product, grouped by ACE. The icon displayed next to the conflict indicates its status:

Table 14-107 • Conflict Status Icons

Icon	Status
	Resolvable
	Resolved
	Unresolvable

In many cases, these errors can be resolved by clicking either Resolve or Resolve All. In cases where no automatic resolution is possible, the Unresolvable icon ( ) appears next to the conflict and the Resolve button is disabled.

If one of the following options on the Resolution Tab of the Options Dialog Box is selected, several additional dialog boxes could appear prompting for approval or additional information:

- If the **Preview and approve all resolutions** option is selected, the Resolution Details Dialog Box appears, listing the changes that are going to be made and prompting you to approve them.

- If the **Maximize user flexibility in selecting resolutions** option is selected, and if you are using a CARD with multiple resolution options, a list of options from you to choose from could appear the CARD Resolution Options Dialog Box, and you would be prompted to select one.

For File Conflicts, click the plus sign next to the conflict icon to see the package that conflicts with the selected package.

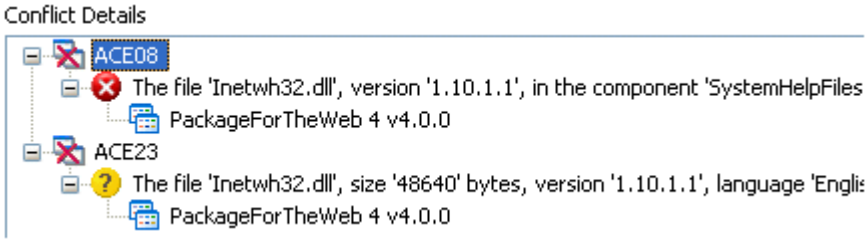


Figure 14-10: Conflicts listed in the Conflicts View

Resolution Options

This section displays the current resolution option: either resolve in the original Windows Installer package, or using the specified transform. Click Edit to change the current settings on the Resolution Options dialog box.

Customize Dialog Box

The Customize Dialog box, which is accessed by selecting Customize from the Tools menu, allows you to customize which tool bars are available in the ConflictSolver user interface, as well as the buttons that are available on the tool bars. The dialog box consists of two tab panels:

Toolbars

From the Toolbars panel, you can select viewing properties for all tool bars, such as whether teletypes are displayed, the style of the toolbar, and the size of the buttons. You can also create your own custom toolbar, onto which you can place buttons found in the Command tab.

Command

The Command panel allows you to customize tool bars and the menu bar. Drag the command or menu you want to add to the existing toolbar; it appears where you place it. To remove a command or menu, select it and drag it off the toolbar.

Properties Dialog Box

To open the Properties dialog box, select an ICE on the Validation View and select Properties from the context menu. The full Description of the ICE is displayed, along with the ICE number. You can add a comment in the Explanation text box.

Changing the State of an ICE Error

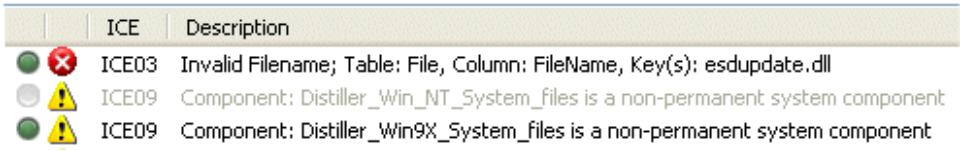
If you do not want ConflictSolver to check for a particular ICE error during subsequent validations of a package, you can choose to suppress it. You might choose to suppress an ICE error that is a known issue at your organization which does not need additional corrections. The ICE Error state is listed in the State area of the Properties dialog box:

- **Enabled**—Continue to check this ICE rule during subsequent validations of this package.
- **Disabled**—Do not check this ICE during subsequent validations of this package.

You can change the State of an ICE error by selecting **Enabled** or **Suppressed** on Properties Dialog Box, or by selecting the ICE error on the Validation View and then selecting Enable or Suppress from the context menu.

If you choose to suppress an ICE, you could document the reason for the suppression in the Explanation text box on the Properties Dialog Box.

On the Validation View, suppressed ICE errors are listed in gray:



Note • *Suppressed ICE errors are still listed in the Package Report for that package. To remove an ICE error from the Package Report, you need to Delete it rather than Suppress it. See Deleting an ICE Error for more information.*

Expression Builder Dialog Box

The Expression Builder dialog box, available by clicking Build Expression on the **Where Clause Panel** in the Rules Wizard (when creating a new user-defined ACE) or from the Where Clause Tab of the ACE Rule Properties Dialog Box (when editing a user-defined ACE), allows you to build simple Where clause expressions for ConflictSolver user-defined ACEs.

Set values for the following options:

Table 14-108 • Expression Builder Dialog Box Properties

Option	Description
Table Columns	This list is populated from the table columns in the table defined in the Rules Wizard General Information Panel. Select the table column used in this Where clause.

Table 14-108 • Expression Builder Dialog Box Properties (cont.)

Option	Description
Comparison Operator	Pick an operator to use for comparison in the Where clause. You can pick from the following: • = (Equal To) • <> (Not Equal To) • > (Greater Than) • < (Less Than) • >= (Greater Than or Equal To) • <= (Less Than or Equal To)
Constant	This constant can be a numerical value or string value. The property label will change based on the expected constant type. This value is compared against the data in the specified table. ***Note*** • *When using the Expression Builder dialog box to create a Source and Target Packages custom ACE to compare the value of a column in the source table to the value of a column in the target table, you can select the first table column name from the Table Columns list. However, you have to manually enter the second table column name in the Constant text box. When doing so, enter the table column name using the same syntax that is used in the Table Columns list: [Source].[ColumnName] or [Target].[ColumnName].* *See Creating a Custom/Source and Target Packages ACE.*
Expression Operator	If there is more than one expression in the Where clause, you can specify an operator to join the current expression to the previous expression.

Tip • *When you are constructing simple expressions, it is helpful to use the Expression Builder dialog box, but you are not limited to the formatting options that the Expression Builder provides to you. If you know how to write Where clauses in SQL, you can use significantly more powerful expressions by entering them directly in the Where Clause text box on the Where Clause Panel of the Rules Wizard or on the Where Clause Tab of the ACE Rule Properties Dialog Box.*

Options Dialog Box

The Options dialog box, accessible by selecting Options from the Tools menu, provides a way to configure how ConflictSolver handles validation and conflict identification, as well as some general settings. This dialog box is divided into the following tabs:

- Rules Tab
- Duplicate Package Tab

- Extended Attributes Tab
- General Tab
- Import Tab
- Resolution Tab
- Validate Tab
- History Tab

Rules Tab

From the **Rules** tab of the Options dialog box, you can configure the following options:

Table 14-109 • Rules Tab Properties

Option	Description
Suppress Warnings/Info	When these boxes are checked, the messages of that type will not be displayed in the **Conflicts** tab of the Output Window during conflict identification. It is strongly recommended you do not suppress warning messages so you can ensure proper conflict identification.
Check conflicts across all source and target packages	When performing conflict analysis using multiple source packages and one or more target packages, ConflictSolver will evaluate each source package against each target package. However, if you want ConflictSolver to also perform conflict analysis of each source package against every other source package and each target package against every other target package, select this option. ***Note*** • *If performing conflict analysis with multiple source packages, and if this option is selected, both ACE21 and ACE22 are automatically evaluated when either one is selected. If only one of them is evaluated, then there is the possibility that conflicts will not be detected.*

Table 14-109 • Rules Tab Properties (cont.)

Option	Description
Enabled Predefined and Custom Rules	There are two types of ACE rules: • **Conflict ACEs**—Detect conflicts between two Windows Installer packages. • **Best Practice ACEs**—Internally perform checks against the structure of a Windows Installer package. (Similar to Microsoft's ICE validation rules.) Select one of these radio buttons to toggle the ACE tree listing between Conflict ACEs and Best Practice ACEs: • **Show best practice rules** • **Show conflict detection rules** In both the Conflict ACE and Best Practice ACE trees, if you want a specific type of conflict identified by default, select the appropriate check box. ACEs associated with unselected boxes will not be performed by default during conflict identification. However, you can override these in the Summary Panel of the Conflict Wizard.
Custom ACE Rule File	The user-defined ACE file specified here is run after the pre-defined ACE rules are run. The selection of this user-defined ACE file will affect the default Conflict Types displayed on this dialog (described above), as well as those displayed on the Rules Viewer and the Conflict Wizard. By default, a file path to an initially empty user-defined ACE file is provided for you. If you have already created a user-defined ACE, specify the location of that user-defined ACE file to activate it. Only one user-defined ACE file can be active at one time. You use user-defined ACEs to extend the functionality of pre-defined ACEs with company-specific functionality. By selecting different user-defined ACE files, you can organize rules appropriate for individual users in your organization. See User-Defined ACEs for more information.

Duplicate Package Tab

When you import a package into the Application Catalog, Application Manager checks specific identifiers that are selected on the **Duplicate Package** tab to determine if that package has already been imported.

If Application Manager determines that you are attempting to import a duplicate package (based upon the selected identifiers), the **Product Name Change** dialog box opens, where you are given the opportunity to enter a name of your choosing to identify the duplicate package before the package is imported.

Note • *If Application Manager is performing a bulk import or reimport, it still identifies duplicate packages using the user-specified criterion. However, Application Manager will generate a unique displayed product name and allow the import to proceed without prompting the user.*

The identifiers you can select on the **Duplicate Package** tab are as follows:

Table 14-110 • Duplicate Package Tab Properties

Option	Description
Duplicate Package Identification Options	Select one or more of the following options to specify the identifiers that Application Manager will check to determine if a Windows Installer package has already been imported: • **PackageCode Property**—Identifier of package product was installed from. No two non-identical `.msi` files should ever have the same package code. • **ProductCode Property**—Unique identifier for the particular product release, represented as a string GUID, for example {12345678-1234-1234-1234-123456789012}. • **Product Language**—The language the installer should use for any strings in the user interface that are not authored into the database. • **Product Version**—Version of the product in string format. The format of the string is: `major.minor.build`. • **List of Transform Files**—A list of the transformations associated with this package import operation. • **[None Selected]**—If you do not select any of these five identifiers, Application Manager checks the ProductName Property identifier to determine if a package is a duplicate.
Duplicate Virtual Package Identification Options	Select one or more of the following options to specify the identifiers that Application Manager will check to determine if an App-V package has already been imported: • **PackageGUID**—Unique identifier of App-V package. • **VersionGUID**—Unique identifier of App-V package version. • **[None Selected]**—If you do not select either of these identifiers, Application Manager checks the Product Name identifier to determine if a package is a duplicate.

Table 14-110 • Duplicate Package Tab Properties

Option	Description

Duplicate Package Naming Syntax	When it identifies a duplicate package, Application Manager generates a new name for that package using the syntax specified in this field. The default syntax is:
	`[Manufacturer]_[ProductName]`
	This means that if Application Manager encountered a duplicate package, it would pre-pend the duplicate's Product Name with the Manufacturer's name and, if necessary, append the Product Name with numbers. For example, the second time that PowerPoint is imported, its name would be changed to:
	`Microsoft Corporation_PowerPoint`
	To change the default naming syntax, edit this field.
	This generated name is displayed on the **Product Name Change** dialog box, and it is the name that will be displayed in the product tree on the left side of the ConflictSolver Main View.
	Note • *Changing this "display" name does not change the ProductName Property that appears in the title bar of the ConflictSolver Product View.*

Example of Importing a "Duplicate" Package

One common reason why you might want to import the same package into your Application Catalog database more than once would be if you wanted to use Tuner to create custom installation SKUs of a common MSI package to distribute to different departments in your organization, each installation including certain features that are appropriate for the department and excluding certain features that are not appropriate. For example, if you were distributing a copy of Microsoft Office, you could add transforms to the Microsoft Office MSI package so that:

- Accounting's installation would include only Word and Excel
- Marketing's installation would include only Word and PowerPoint, and
- Development's installation would include only Word and Access.

Therefore, you might want to import the same package into your database more than once, each time with a different set of transformations. What happens when you import the package the second time depends upon the identifiers you selected on the **Duplicate Package** tab. In this example:

- If you select the **List of Transform Files** and **ProductCode** identifiers on the **Duplicate Package** tab of the **Options** dialog box, Application Manager will not identify these two packages as duplicate, even though they have the same ProductCode, because they have a different set of transformations. Therefore, the **Product Name Change** dialog box will not appear, and the package will be imported with the same display name as the first package.
- If you only select the **ProductCode** identifier on the **Duplicate Package** tab of the **Options** dialog box, Application Manager will identify the second package as a duplicate because the two packages have the same ProductCode, and the **Product Name Change** dialog box will appear.

Note • *The options that you select on the* ***Duplicate Package*** *tab of the* ***Options*** *dialog box apply globally to all packages that you attempt to import; you cannot apply different identifiers to different packages. Also, since these options are saved in the AdminStudio Shared Directory, everyone using AdminStudio at your organization who shares the same directory will share the same* ***Duplicate Package*** *options.*

Extended Attributes Tab

From the Extended Attributes tab of the Options dialog box, you can specify the name and location of the extended attribute description file (`.xml`) which specifies the extended attributes available for each package in ConflictSolver.

AdminStudio includes a default XML file for extended attributes, which is stored in the AdminStudio shared location. You can also construct your own Extended Attribute Description File. Each new Application Catalog automatically points to this file, and displays the name and location of the file in this tab.

Note • *You can also configure ConflictSolver to integrate with Workflow Manager by selecting the corresponding option.*

Tip • *If you overwrite the default XML file with your extended attributes data, all subsequent Application Catalogs created include your attributes by default.*

General Tab

From the **General** tab of the **Options** dialog box, you can configure the following options:

Table 14-111 • General Tab Properties

Properties	Description
Confirm All Drag-Drop Operations	Select this option if you want ConflictSolver to prompt you for confirmation whenever you drag and drop items.
Display Broken MSI/ MST Package Links	When selected, any broken package links will be indicated by an icon change in the Product View, and with a message in the Products view, which allows you to attempt to locate the package.
Only Display View Nodes With Data	When this option is selected, packages containing views without data will not display those views. For example, if a package has no shortcuts, then the Shortcuts view is not displayed for that package.

Table 14-111 • General Tab Properties (cont.)

Properties	Description
Display RPT Reports	By default, three Crystal Reports options are listed on the **Reports** menu and on the package context menu: • Crystal Reports Conflicts • Crystal Reports Files Report • Crystal Reports Registry To suppress the display of these **Crystal Reports** selections—so that only the standard ConflictSolver **Package**, **File**, and **Registry** selections are listed—clear this option.
RPT Reports: SQL Server	If the **Display RPT Reports** option is selected, these fields are enabled to allow you to specify the location of the Crystal Reports files. ConflictSolver searches the directory specified for Crystal Reports reports. You can directly enter the path, or use the Browse button (...) to select it. Reports must be in Crystal Report (`.rpt`) format.
Turn off display of conflict detection and resolution options in Application Manager	By default, all functionality for interacting with the Application Catalog, including conflict detection and resolution, is available in Application Manager. To turn off the display of conflict detection and resolution options in Application Manager, select this option. With this option selected, all of the conflict detection and resolution functionality performed by ConflictSolver will no longer be available in the Application Manager interface. See the Comparison of ConflictSolver and Application Manager Functionality table for more information.

Import Tab

From the Import tab, you can configure several options that affect how packages are imported into ConflictSolver. Most of these options can be overridden on a package-by-package basis; these options serve as the defaults.

You can configure the following options:

Table 14-112 • Import Tab Properties

Properties	Description
Check for Validation Errors	By default, packages will be checked for validation errors on import.
Check for Conflicts	If this option is selected, the Conflict Wizard will run immediately following the package import to perform conflict analysis on the newly imported package.

Table 14-112 • Import Tab Properties

Properties	Description
Automatically run PackageExpert test after import	If this option is selected, PackageExpert will test the package immediately following import to check the integrity of the package and automatically resolve any errors that are found.
Import Binary Data	Binary data in the package will be imported.
Import Custom Tables	By default, custom Windows Installer tables will be imported into ConflictSolver. Tables specified in the Ignore Tables list will not be imported.
Abort on Fatal Error	Select this option to instruct ConflictSolver to stop import if a fatal error occurs. Fatal errors include loss of network connection or running out of space. Non-fatal errors include duplicate packages or merge modules or data truncation.

Ignore Tables List

The Ignore Tables list displays all of the tables that will be ignored during import (not imported into ConflictSolver). You can select tables and delete them from the list, or you can add new tables to the list by clicking Add (which displays the Add Ignore Table dialog box).

Resolution Tab

On the Resolution Tab of the **Options** dialog box, you can specify options to determine how conflict resolutions are performed.

The **Resolution** Tab contains the following options:

Table 14-113 • Resolution Tab Properties

Option	Description
Perform resolutions against a transform file	If this option is selected, ConflictSolver will, by default, save conflict detection resolutions in a transform file (MST). The original Windows Installer package (MSI) remains untouched. You can override this option for an individual package by going to that package's Conflict View and clicking **Edit** and changing this option on the Resolution Options Dialog Box.

Table 14-113 • Resolution Tab Properties (cont.)

Option	Description
Preview and approve all resolutions	Select this option if you want to review and approve the suggested resolution for each conflict before ConflictSolver performs any resolutions. When the conflict detection process is completed, the Resolution Details Dialog Box is displayed, listing the conflicts that were found and the CARDs that could be used to resolve these conflicts. Before ConflictSolver will continue resolving these conflicts, each conflict resolution must be approved. See Conflict Resolution Process for more information.
Maximize user flexibility in selecting resolutions	Some CARDs can resolve a conflict in more than one way. For example, ACE06 checks whether the executable module within the component is the key file, and accumulates a list of potential files to be used for the key file. If this option is selected, this list of files is displayed on the CARD Resolution Options Dialog Box, and you are prompted to select which file to use. If this option is not selected, the CARD will perform its default resolution.

Validate Tab

From the Validate tab of the Options dialog box, you can configure the following properties:

Table 14-114 • Validate Tab Properties

Option	Description
Suppress Warnings/Info	When these boxes are checked, the messages of that type will not be displayed in the Validate tab of the Output Window during validation. It is strongly recommended you do not suppress warning messages so you can ensure proper validation of packages before importing them into ConflictSolver.

Table 14-114 • Validate Tab Properties (cont.)

Option	Description
MSI Input Files	There are two options that you can configure for MSI input files: **CUB File** The file specified in this field is file containing Internal Consistency Evaluators (ICEs) used for validation. Either enter the location of this file directly, or use the Browse button (...) to locate it. **ICE Rules** By default, ConflictSolver uses all ICEs contained in the default CUB file during validation. To permanently exclude the execution of specific ICEs, list them in this edit field, separated by semicolons (;). ***Caution*** • *The functionality of this field has changed. In previous versions of AdminStudio, ConflictSolver would use all of the ICEs in the default CUB file during validation unless you entered specific ICEs to run in the ICE Rules field. Now the ICE Rules field is used to exclude specific ICEs from being used during validation, meaning that all of the ICEs in the default CUB file will be used except those entered in the ICE Rules field.*
MSM Input Files	There are two options that you can configure for merge module input files: **CUB File** The file specified in this field is file containing Internal Consistency Evaluators (ICEs) used for validation. Either enter the location of this file directly, or use the Browse button (...) to locate it. **ICE Rules** By default, ConflictSolver uses all ICEs contained in the default CUB file during validation. To permanently exclude the execution of specific ICEs, list them in this edit field, separated by semicolons (;). ***Caution*** • *The functionality of this field has changed. In previous versions of AdminStudio, ConflictSolver would use all of the ICEs in the default CUB file during validation unless you entered specific ICEs to run in the ICE Rules field. Now the ICE Rules field is used to exclude specific ICEs from being used during validation, meaning that all of the ICEs in the default CUB file will be used except those entered in the ICE Rules field.*

History Tab

On the **History** tab, you can specify the type of package history information that you want to log and be able to view on a package's **Catalog History View**. The following options are included:

Table 14-115 • Options Dialog Box/History Tab Options

Option	Description
Maximum History Log Per Package	Enter the maximum number of History Log entries you want to store for each package. When the number of log entries exceeds this amount, the oldest entry will be deleted.
Import/Reimport	Specify whether to log and whether to display an entry each time a package is imported or reimported.
Validation	Specify whether to log and whether to display an entry each time validation is performed on a package.
Conflict Detection	Specify whether to log and whether to display an entry each time conflict detection is performed on a package.
Resolution	Specify whether to log and whether to display an entry each time a package conflict is resolved.
Extended Attribute Modification	Specify whether to log and whether to display an entry each time an extended attribute of a package is modified.
Package Description Modification	Specify whether to log and whether to display an entry each time a package description is modified.
Package Move/Copy	Specify whether to log and whether to display an entry each time a package is copied or moved from one group to another.
Patch Analysis	Specify whether to log and whether to display an entry each time patch analysis is performed on a package.

Patch Impact Information Dialog Box

When you click on one of the informational links in the **Summary** section of the Impact Analysis View (Patches Tab), the **Patch Impact Information** dialog box opens, displaying information on the selected link.

For example, if you click on the **Products** link, the **Patch Impact Information** dialog box opens with the **Products** tab selected.

The dialog box contains the following tabs (which are only displayed if they have data):

Table 14-116 • Patch Impact Information Dialog Box Tabs

Tab	Description
Products	The product(s) against which impact analysis was performed. The name, description, and any groups the products belong to are displayed.
Updated Products	The name, description, and modification date of any relevant product that has been updated since the last impact analysis was performed.
Deleted Products	The name and description of any relevant product that has been removed from ConflictSolver since the last impact analysis was performed.
New Products	The name, description, and import date of any product that has been added to ConflictSolver since the last Impact analysis was performed.

Note • *This dialog box is similar in function to the Conflict Information Dialog Box.*

Resolution Details Dialog Box

The Resolution Details dialog box, which appears when you click Resolve or Resolve All on the Conflicts View, lists the proposed CARD resolutions that will be made to the package. Select the CARD resolutions that you want to execute and click Approve.

If you want to see the precise instructions that will be executed by a CARD during the resolution process before you approve it, select a proposed CARD resolution and double-click to access the Resolution Operations dialog box.

Note • *The Resolution Details dialog box only appears if the Preview and approve all resolutions option is selected on the Resolution Tab of the Options dialog box. If this option is not selected, this dialog box does not appear. If you do not want this dialog box to appear, select the Don't show this dialog again check box on the dialog box. This is equivalent to unselecting the Preview and approve all resolutions option on the Resolution Tab of the Options dialog box.*

Resolution Operations Dialog Box

This dialog box shows the selected CARD that is being executed, and the precise instructions that will be executed by the CARD during the resolution process.

Click Close to exit this dialog box.

Resolution Options Dialog Box

From this dialog box, you can specify the resolution method.

Table 14-117 • Resolution Options Dialog Box Properties

Properties	Description
Save resolutions in original MSI package	Select this option if you want to save conflict detection resolutions in the existing Windows Installer package.
Save resolutions in a new transform file	Select this option if you want to save conflict detection resolutions in a transform file. In the text box, specify the name and location for this transform file or click Browse and select the transform file location.

This dialog box is accessible through the Conflicts View and from the Conflict Resolution dialog box by clicking Edit in the Resolution Options section.

Rules Viewer Dialog Box

The Rules Viewer dialog box, accessible by selecting Rules Viewer from the Conflicts menu, allows you to view the current categorization of ACEs used for conflict identification. More importantly, it allows you to access the Rules Wizard to include user-defined ACEs in ConflictSolver.

The primary window in the Rules Viewer dialog box displays a tree view containing each available ACE, grouped by category. If you click New, the Rules Wizard launches, allowing you to configure information for a new ACE.

When a user-defined ACE is selected on the Rules Viewer dialog box, the Edit button is enabled. When you click Edit, the ACE Rule Properties Dialog Box appears, where you can reconfigure the ACE. You can also delete user-defined ACEs by selecting them and clicking Delete.

Wizards

The following Wizards are included in ConflictSolver:

- Conflict Wizard
- Rules Wizard
- Validation Wizard

Conflict Wizard

Although a Windows Installer package or merge module may be built to guidelines put forth by Microsoft, it is possible that the interaction between packages, or between a package and the base operating system, may cause unwanted results in your production environment. You can use the Conflict Wizard to identify these conflicts before you deploy packages, and resolve the problems before they affect your end users.

The Conflict Wizard allows you to identify conflicts between a Windows Installer package and packages already cataloged in ConflictSolver. You can check for a variety of conflict types, including file, component, and registry conflicts. In many cases, ConflictSolver can resolve the issues automatically. You can also create your own custom rules to ensure packages conform to your internal standards and practices. ConflictSolver has rules to detect conflicts involving: Components, Files, Registry Entries, Shortcuts, INI Files, ODBC Resources, NT Services, File Extensions, and Product Properties.

The Conflict Wizard consists of the following panels:

- Welcome Panel
- Source Type Panel
- MSI Source Information Panel
- Source Package Panel
- Choose Action Panel
- Best Practice Rules Panel
- Conflict Rules Panel
- Target Information Panel
- Summary Panel

When run, ConflictSolver displays the output report in the Conflicts tab of the Output Window.

Welcome Panel

The first panel of the Conflict Wizard welcomes you to the Wizard.

This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 14-118 • Welcome Panel Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help about the specific Conflict Wizard panel.

Source Type Panel

From the Source Type panel, select whether you want to compare an external Windows Installer package against packages already in the ConflictSolver Application Catalog or perform a comparison between packages within the ConflictSolver Application Catalog.

- If you select Internal Application Catalog package, the next panel that appears is the Source Package panel.
- If you select Windows Installer package, the next panel that appears is the MSI Source Information Panel.

MSI Source Information Panel

From the MSI Source Information Panel, specify the Windows Installer package you want to check for conflicts against packages already in ConflictSolver.

Click Browse to locate and select the package. Should that package require transforms, click the New button ( ) above the Transforms area. Use the Browse button (...) to locate the transform. If the package requires multiple transforms, you can repeat the procedure as necessary. The order in which transforms are applied can be changed by selecting a transform and clicking the Move Up ( ) and Move Down ( ) buttons.

Delete a transform file from the Transforms list by selecting the transform and clicking the Delete button ( ).

Source Package Panel

When you select the Internal Application Catalog package option on the Conflict Wizard Source Type Panel (specifying that you want to compare source packages from within the ConflictSolver database), the Source Package panel appears.

This panel displays the packages in the same hierarchy as the Product View. For your convenience, empty groups are not displayed.

The ConflictSolver supports multiple source package checking. Therefore, select the package(s) you want to analyze for conflicts. If you select multiple source packages, ConflictSolver will conflict check each source package serially against all of the selected target packages.

***Note** • The target packages are specified on the next panel, the Target Information Panel. The Target Information panel will naturally exclude all packages that you selected on the Source Package panel.*

You can also select all packages in ConflictSolver or clear all selected packages using the Select All and Clear All buttons.

***Caution** • Since the Target Information panel excludes all packages that are selected on the Source Package panel, exercise caution when using the Select All button. If you select all of the packages on the Source Package panel, there will be no target packages listed on the Target Information Panel, and therefore you will not be able to run conflict detection.*

Choose Action Panel

On this Panel you specify whether to evaluate the selected source packages against Best Practice rules or against other packages. You can also specify whether you want ConflictSolver to automatically resolve resolvable conflicts , and whether to automatically generate a report when conflict analysis is complete.

The following options are included:

Table 14-119 • Choose Action Panel Options

Options	Description
Evaluate source package against best practice rules	Select this option to include Best Practice ACE rules in this conflict analysis.
Detect conflicts between source package and other packages	Select this option to include Conflict ACE rules in this conflict analysis.
Run Resolution	Select to display the Conflict Resolution dialog box after conflict detection. From the Conflict Resolution dialog box, you can access information pertaining to the last execution of the Conflict Wizard. This persisted conflict data allows you to view when the last execution was performed, the products that it was run against, the ACE rules used, and conflicts discovered. The dialog box also provides a list of updated, deleted, or added products that may necessitate performing conflict identification again.
Generate Reports	Select this option to generate a conflict detection analysis report containing a summary of the conflicts that were detected, and a detailed description of each identified conflict including how this conflict will be resolved.

Best Practice Rules Panel

ConflictSolver includes Best Practice rules that operate like ICEs (Internal Consistency Evaluators) to enforce Microsoft best practices. While an ICE is a custom action written by Microsoft which is used during validation to determine if an installation package is built according to Windows Installer standards, an ACE is a custom action included with ConflictSolver that is also used to enforce Windows Installer standards.

In the Best Practice Rules Panel, select the types of Best Practice Rules to evaluate in conflict analysis. By default, the Best Practice ACEs that are selected on the Rules Tab of the Options dialog box will be selected.

The following Best Practice ACE rules are available.

Table 14-120 • Best Practice ACE Rules

Conflict Type	Associated Rules
Components	ACE04, ACE05, ACE06,

Table 14-120 • Best Practice ACE Rules

Conflict Type	Associated Rules
Merge Module Integrity	ACE26, ACE36
Recommended Rules	ACE25, ACE27, ACE28, ACE29, ACE31, ACE32, ACE33, ACE34, ACE35
App-V Recommended Rules	ACE201, ACE202, ACE203, ACE208, ACE209, ACE210, ACE211, ACE212, ACE213, ACE214, ACE216
Windows Terminal Server Compatibility	WTS01, WTS02, WTS03, WTS04, WTS05

Conflict Rules Panel

In the Conflict Rules panel, select the types of conflicts to use for conflict evaluation. The following Conflict ACE rules are available:

Table 14-121 • Conflict ACE Rules

Conflict Type	Associated Rules
Component	ACE02, ACE09, ACE30
File Extensions	ACE17
Files	ACE03, ACE07, ACE08, ACE12, ACE23
INI Files	ACE14, ACE21, ACE22
NT Services	ACE16
ODBC Resources	ACE15
Product Properties	ACE18, ACE19, ACE20
Registry	ACE10, ACE24
Shortcuts	ACE13
App-V Recommended Rules	ACE200, ACE204, ACE205, ACE206, ACE207, ACE215

Note • *The default settings in the Conflict Options panel reflect the settings in the Conflicts tab of the Options dialog box (available from the Tools menu).*

Target Information Panel

In the Target Information Panel, select the individual packages or groups of packages in ConflictSolver that you want to compare the source package(s) against.

Each package selected will be compared against the packages you specified in the Source Package Panel (for internal comparisons) or MSI Source Information Panel (for comparisons with an external Windows Installer package).

***Note** • The Target Information panel excludes all packages that you selected on the Source Package panel. Empty groups are also excluded.*

You can also select all packages in ConflictSolver or clear all selected packages using the Select All and Clear All buttons.

Summary Panel

The Summary panel provides a detailed summary of the options that were selected in the previous panels of the Wizard.

Click Finish to run the Conflict Wizard using the options specified.

Rules Wizard

The Rules Wizard allows you to create user-defined ACE rules for later use by the ConflictSolver Wizard. It is accessible by clicking New on the Rules Viewer dialog box.

The following panels are part of the Rules Wizard:

- Welcome Panel
- General Information Panel
- Additional Information
- Custom Options Panel
- Where Clause Panel
- DLL-Based ACEs Panel
- Summary Panel

Welcome Panel

The Rules Wizard allows you to create user-defined ACE rules for later use in conflict identification. The first panel displayed is the Welcome panel.

Click Next to proceed to the General Information Panel.

General Information Panel

From the **General Information** panel, you can configure information about the new ACE rule. This information is used primarily for display information (Name, Brief Description, Description, and Information URL).

Also on the **General Information** panel, you specify whether you are creating a rule for a Windows Installer package (**MSI**) or a Microsoft App-V package (**AppV**).

Table 14-122 • General Information Panel Options

Option	Description
Name	The name of the ACE, used to organize the rule in ConflictSolver. This is displayed in several places, including the Output Window, the Rules Viewer dialog box, and the Conflicts tab of the Options dialog box.
Associated Table	Select the table in the Application Catalog which will be queried in the user-defined ACE. This also determines which columns are available in the Expression Builder dialog box, and which tokens are available for the Error and Display strings on the Custom Options Panel of the Rules Wizard.
Package Type	Select **MSI** or **App-V** to identify the type of package that this rule will be run on.
Brief Description	Enter a brief description which will be displayed in the Rules Viewer, the Conflicts tab of the Options dialog box, and in the Output Window. This description should be clear enough so users can understand when to use this ACE.
Description	Enter a description of the ACE, which is displayed at the bottom of the Rules Viewer dialog box when the ACE is selected and in the Output Window during conflict identification when the ACE executes.
Information URL	Provide a URL to get further information for the ACE. This URL appears in the Conflict Details area of the Conflicts View after conflicts have been identified.

Click Next to proceed to the Additional Information Panel; click Back to return to the Welcome Panel.

Additional Information

From the Additional Information Panel, you can provide information for categorizing the ACE in relation to other ACEs in ConflictSolver.

Table 14-123 • Additional Information Panel Option

Option	Description
Category	Either select an existing category for this new rule, or enter the name for a new category. These categories are displayed in the Conflict View and the Conflicts tab of the Options dialog box. Ideally, any user-defined ACEs should be put in their own category.

Table 14-123 • Additional Information Panel Option (cont.)

Option	Description
Rule Type	Specify the type of ACE you are creating: • **Custom - Source Only Packages ACEs** allow you to quickly test any column or any value of a table to support your business logic. For example, you could use a user-defined ACE to identify packages that create a desktop icon. To define a Source Only Packages ACE, you must define an SQL "Where" clause. ConflictSolver supports external package conflict checking for **Custom - Source Only Packages ACEs**. The Source package can be selected from the ConflictSolver Application Catalog Database or from an external MSI package. See Creating a Custom/Source Only Packages ACE for more information. • **Custom - Source and Target Packages ACEs** allow you to compare columns or values of Source package tables (new packages that you want to install onto a user's system) to columns or values of Target package tables (packages already installed on a user's system). For example, you could use a Source and Target Packages ACE to determine if the installation of a Source package onto a Target system would overwrite or conflict with an existing entry in the .ini file in the System directory of the Target system. To define a Source and Target Packages ACE, you must define a SQL "Where" clause, and specify a Join Column—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are selected and those rows are checked against the Source and Target Packages. ConflictSolver *does not support* external package conflict checking for **Custom - Source and Target Packages ACEs**. Both the Source and Target Packages must be selected from the ConflictSolver Application Catalog Database. See Creating a Custom/Source and Target Packages ACE for more information. • **DLL - User Provided DLL Based ACEs** allow you to run more complex tests—testing many tables in any combination. For example, you could use a DLL-Based ACE to confirm that a source product language is the same as all target product languages. To define a DLL-Based ACE, you use SQL and various programming languages to construct a Windows DLL. With DLL-Based ACEs, you can use a Conflict Application Resolution Definitions (CARDs) to fix the conflict. See Creating a User Provided DLL-Based ACE.

Click Next to proceed to the Custom Options Panel or the DLL-Based ACEs Panel; click Back to return to the General Information Panel.

Custom Options Panel

From the Custom Options panel, you can create display strings for the Output Window and Conflict Details.

Table 14-124 • Custom Options Panel Option

Option	Description
Error String	This string appears in the Output Window when a violation of this ACE rule is detected during conflict identification. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter `Failure in creating desktop icon` in this field. ***Note*** • *Tokens allow you to insert values at run-time from the internal ConflictSolver Application Catalog Database or an external MSI package into the string, such as specifying a file name. To use token replacement in the error string, click on the arrow to the right of the Error String field and pick a value from the list, or just type the values directly in the text box, in the following format:* • *Source Only Packages ACEs: [ColumnName]* • *Source and Target Packages ACEs: [Source.ColumnName] and [Target.ColumnName]* *For more information, see Token Grammar.*
Display String	This string appears in the Conflict Details area of the Conflicts View after conflicts have been identified. For example, if you were creating a user-defined ACE to identify packages that create a desktop icon, you could enter `Duplicate desktop icon found` in this field. ***Note*** • *Tokens allow you to insert values at run-time from the internal ConflictSolver Application Catalog Database or an external MSI package into the string, such as specifying a file name. To use token replacement in the error string, click on the arrow to the right of the Display String field and pick a value from the list, or just type the values directly in the text box, in the following format:* • **Source Only Packages ACEs***: [ColumnName]* • **Source and Target Packages ACEs***: [Source.ColumnName] and [Target.ColumnName]* *For more information, see Token Grammar.*
Severity	Specify whether this ACE should be an Error or a Warning.
Report 'No' results	User defined ACEs report conflicts based on the provided query. However, you may want to report the absence of the data if it could not be found. If you select this option, if the ACE does not return any results, it will be reported as an error (or warning), with the description and error strings as specified. If you expect a **No** result, do not use tokens in your display or error strings.

Click Next to proceed to the Where Clause Panel; click Back to return to the Additional Information Panel.

Token Grammar

What are Tokens?

Tokens represent data in the database that is inserted at runtime. In ConflictSolver, Tokens are used to insert values at runtime from the ConflictSolver Application Catalog Database or an external MSI package into an Error or Display String.

How to Insert Tokens

Tokens are specified on the Custom Options panel of the Rules Wizard. To use token replacement in a string, click the arrow to the right of the Error String and Display String text boxes and select a column name from the list. The column name is then inserted into the string in the following format:

- **Source Only Packages ACEs**—`[ColumnName]`
- **Source and Target Packages ACEs**—`[Source.ColumnName]` and `[Target.ColumnName]`, with the prefix identifying whether the column is in the Source or Target package. If no prefix is used, ConflictSolver assumes the “Source.” prefix.

Note *• The Token list on the Custom Options panel is provided for your convenience; if you prefer, you can type the variables directly in the text boxes. For more information see, Token Grammar.*

Caution *• While you are creating a user-defined ACE in the Rules Wizard, if you initially select a Rule Type of Custom - Source and Target Packages, and then insert tokens in the Error String and Display String fields, the “Source.” prefix will be used. But, before you finish creating this ACE, if you go back and change your Rule Type selection to Custom - Source Only Packages, the tokens that you initially entered into the Display and Error string text boxes will not automatically be updated to remove the “Source.” prefix. For ConflictSolver to correctly interpret this ACE, you need to manually go back to the Error and Display String fields and delete the “Source.” prefix.*

Using the ProductName Pseudo-tokens

You can use the pseudo-tokens of [ProductName], [Source.ProductName] and [Target.ProductName] to insert the name of the Source or Target package in an Error or Display String, even though ProductName is not a table column name.

Where Clause Panel

From the Where Clause Panel, you must define a valid Where clause for the ACE. If you do not know how to build a Where clause, you can click the Build Expressions button to launch the Expression Builder dialog box. You can also click Test to validate the Where clause syntax.

If you selected **Custom - Source and Target Packages** on the Additional Information Panel, you must also select a **Join Column**—a table column in the Application Catalog database that has a matching value for both the Source and Target packages. Rows in each of the packages that have a matching value in the Join Column are

selected and those rows are checked against the Source and Target Packages. For example, if you wanted to evaluate Source and Target packages that write files to the same directory, you might specify Directory as the Join column.

Tip • *To improve query performance, enclose table names in square brackets ([]).*

DLL-Based ACEs Panel

DLL-based ACEs require specific information about the ACE/CARD DLL file and the ACE and CARD Function Names to operate. Enter the following information:

Table 14-125 • DLL-Based ACEs Panel Options

Option	Description
ACE/CARD DLL File	Select the name of the ACE DLL that you are testing.
ACE Function Name	Enter the name that you chose to “export” for this ACE function.
CARD Function Name	Enter the name that you chose to “export” for this CARD function.
Test	Click the Test button next to the ACE Function Name or CARD Function Name to validate that the exported function does exist.

Click Next proceed to the Summary Panel; click Back to return to the Additional Information Panel.

Summary Panel

Once you have configured information for your ACE rule, the Summary panel displays information for final review.

Click Finish to accept this configuration and make the ACE available for conflict identification. Click Back to return to either the **Where Clause Panel** or the DLL-Based ACEs Panel

Validation Wizard

The Validation Wizard allows you to validate a Windows Installer package and any associated transforms prior to importing them into ConflictSolver. Validation is performed against the validation file (.cub) selected in the Validate tab of the Options dialog (available from the Tools menu).

The following panels are included in the Validation Wizard:

- MSI Source Information Panel
- Summary Panel

MSI Source Information Panel

From the MSI Source Information Panel, specify the Windows Installer package you want to check for conflicts against packages already in ConflictSolver.

Click Browse to locate and select the package. Should that package require transforms, click the New button ( ) above the Transforms area. Use the Browse button (...) to locate the transform. If the package requires multiple transforms, you can repeat the procedure as necessary. The order in which transforms are applied can be changed by selecting a transform and clicking the Move Up ( ) and Move Down ( ) buttons.

You can delete a transform from the Transform files list by selecting the transform and clicking the Delete button ( ).

Click Next to proceed to the Summary Panel.

Summary Panel

From the Summary panel, you can review the Windows Installer package and any associated transforms you will validate against the specified CUB file.

If the options are correct, click Finish to begin validation. If not, click Back to return to the MSI Source Information Panel.

15

Testing and Fixing Application Compatibility Using Compatibility Solver

Edition • *Compatibility Solver is included in the AdminStudio Application Compatibility Pack*

Compatibility Solver tests packages for application readiness on Microsoft Windows 7 (32-bit and 64-bit) and Windows Server 2008 R2 platforms, as well as compatibility with Internet Explorer 8.0. You can use Compatibility Solver's automated reporting and fixing infrastructure to prepare packages for platform migrations, ensuring application compatibility and reducing installation issues.

Compatibility Solver's Web application compatibility testing is designed to deliver fast, accurate web application compatibility information to assist decision making when considering browser and/or OS migrations. It automates the assessment of migration tasks when changing between different versions of an Internet browser, such as those encountered when upgrading from Internet Explorer version 6 to version 8.

The Compatiblity Solver Help Library is available on Flexera Software's HelpNet site:

http://helpnet.flexerasoftware.com

16

Identifying and Resolving Package Errors Using PackageExpert

Edition • *PackageExpert is included with AdminStudio Enterprise Edition.*

PackageExpert is the latest addition to AdminStudio's array of powerful testing tools. You can use PackageExpert to check the integrity of your Windows Installer packages and automatically resolve errors that are found.

PackageExpert allows you to quickly run and resolve a series of tests on Windows Installer packages to improve their overall quality and their deployment readiness. PackageExpert provides the flexibility to augment and extend the test suite to suit any particular end user requirement. PackageExpert's tests are currently organized into three major categories:

- **ICE Tests**—The ICE tests perform checks against the structure of a Windows Installer package to enforce Microsoft best practices. They are used to determine if an installation package is built according to Windows Installer standards.
- **Windows Vista Certification Tests**—The Windows Vista tests check packages for compliance with installation requirements of the Windows Vista operating system, and are intended to enable the successful deployment of packages on Vista.
- **Windows 7 Compatibility Tests**—The Windows 7 Compatibility tests check packages for compliance with installation requirements of the Windows 7 operating system, and are intended to enable the successful deployment of packages on Windows 7.

PackageExpert documentation is organized in the following sections:

Table 16-1 • PackageExpert Help Library

Section	Description
About PackageExpert	Introduces you to PackageExpert, its interface, and how it works.
Configuring PackageExpert	Explains how to specify which tests to run by default every time a package is tested.

Table 16-1 • PackageExpert Help Library

Section	Description
Testing Packages and Viewing Results	Explains how to test both standalone packages and packages in an Application Catalog. Explains how to view test results in PackageExpert and in ConflictSolver. It also explains how to customize your test results by excluding specific errors and by changing the severity of specific errors.
Resolving Errors	Explains how PackageExpert automatically resolves errors, and how you specify how you want errors to be resolved.
Adding Manual Tests	Provides instructions on how to add and run your own XML-based tests to PackageExpert.
PackageExpert Reference	Every view and dialog box available in PackageExpert is discussed in this section.

About PackageExpert

PackageExpert allows you to quickly run and resolve a series of tests on Windows Installer packages to improve their overall quality and their deployment readiness. PackageExpert provides the flexibility to augment and extend the test suite to suit any particular end user requirement. PackageExpert's tests are currently organized into three major categories:

- **ICE Tests**—The ICE tests perform checks against the structure of a Windows Installer package to enforce Microsoft best practices. They are used to determine if an installation package is built according to Windows Installer standards.
- **Windows Vista Certification Tests**—The Windows Vista tests check packages for compliance with installation requirements of the Windows Vista operating system, and are intended to enable the successful deployment of packages on Vista.
- **Windows 7 Compatibility Tests**—The Windows 7 Compatibility tests check packages for compliance with installation requirements of the Windows 7 operating system, and are intended to enable the successful deployment of packages on Windows 7.

In addition, you can add your own XML-based manual tests.

The following diagram provides an overview of how you can use PackageExpert to perform testing and resolution:

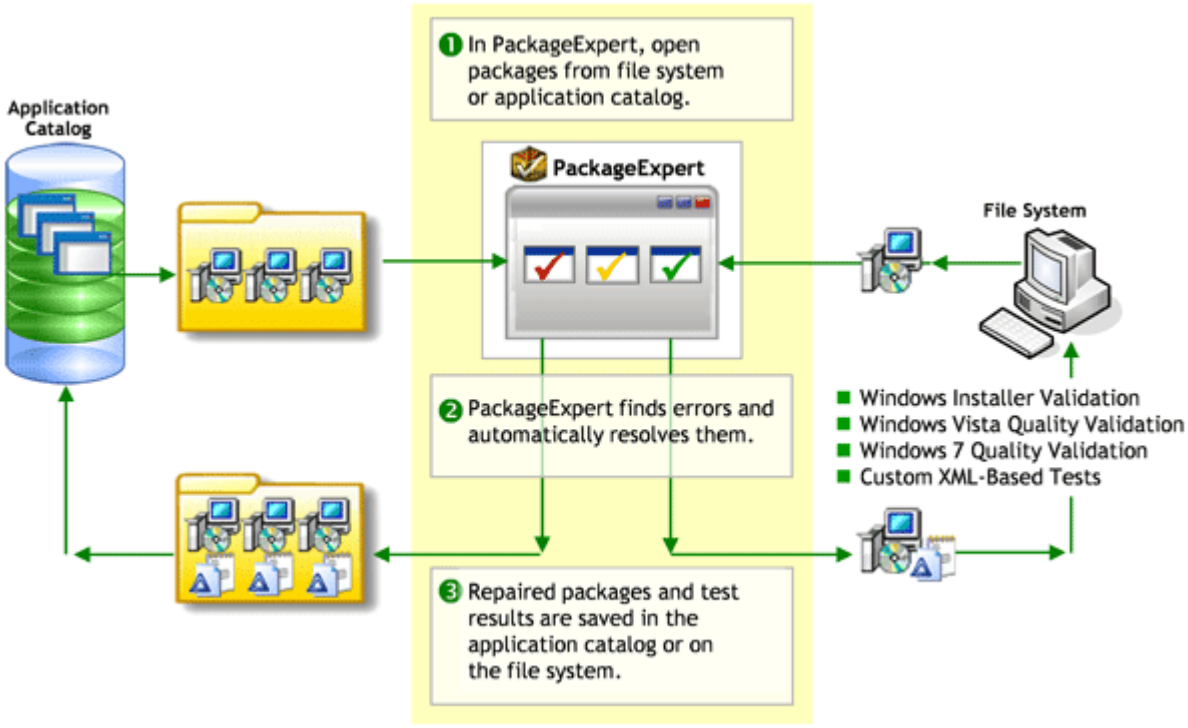


Figure 16-1: PackageExpert Overview Diagram

Using PackageExpert to perform testing is a three-step process:

- **Step 1: Select Package(s)**—Open a package from the file system or connect to an Application Catalog database.
- **Step 2: Run Tests**—Select a package or group folder and select **Test Package** from the context menu.
- **Step 3: Resolve Errors**—PackageExpert can automatically resolve some errors, while others require you to perform a manual resolution:
  - **To resolve those errors that are automatically resolvable**, select a test category, package, or message node and select **Resolve All** or **Resolve Test** from the context menu.
  - **To resolve errors requiring manual resolution**, select the message node and perform the instructions provided in the right pane under **Resolution: Manual**.

Configuring PackageExpert

You can specify which tests you want PackageExpert to run for subsequent test runs, and you can also specify whether to resolve errors directly in the original Windows Installer `.msi` file or using a transform `.mst` file.

- Setting Default Test Configuration
- Specifying the Resolution Method

Setting Default Test Configuration

On the **Configuration** view, you specify which tests you want PackageExpert to run for all subsequent test runs. These settings can be changed at any time. but any changes you make do not effect the results of packages that have already been tested.

The **Configuration** view consists of two panes:

- **Package tree**—Tests are grouped into categories and are displayed in a tree on the left pane.
- **Help pane**—When you select a test, a help topic explaining that test opens in the right pane.

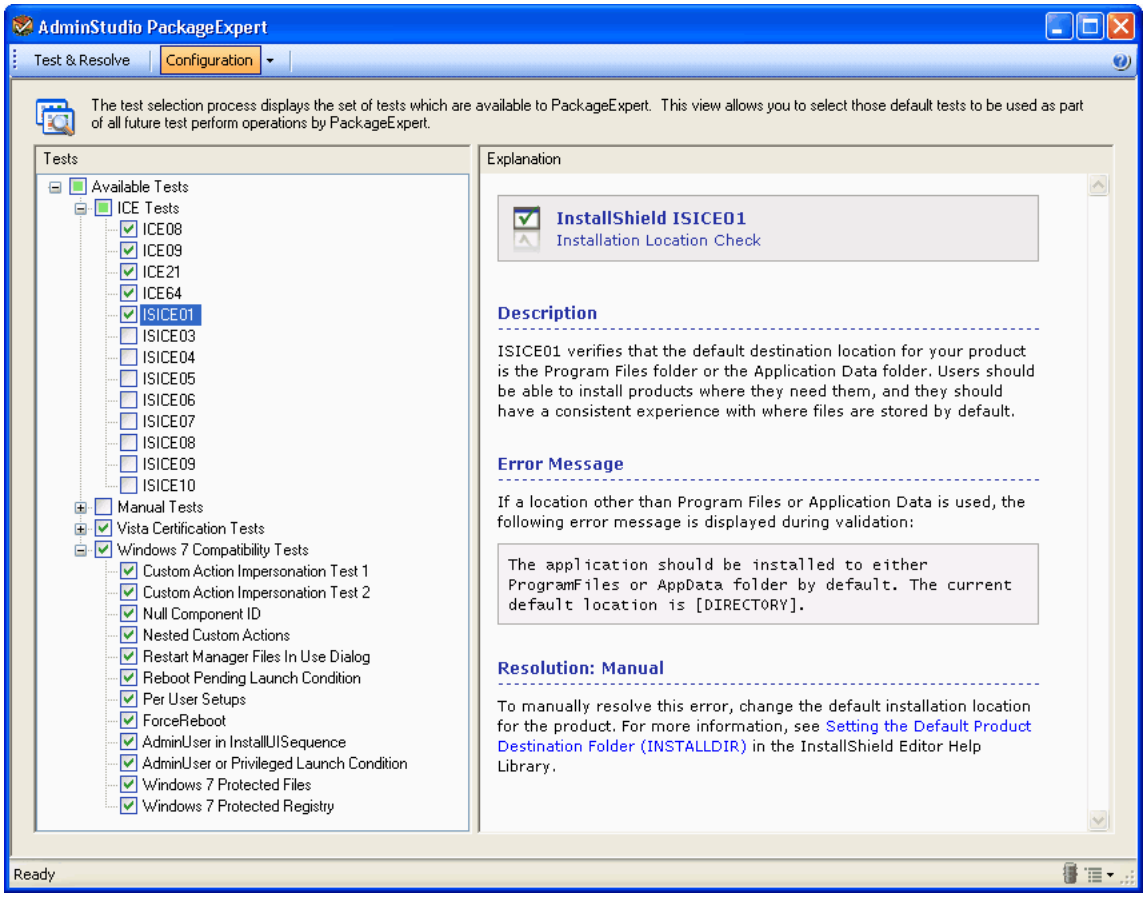


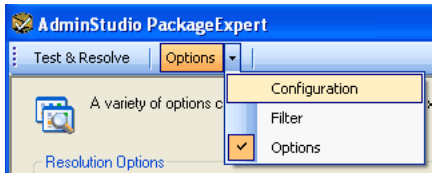
Figure 16-2: PackageExpert Configuration View

To set the default tests that will be run, perform the following steps.

Task: ***To set default tests to be run:***

1. Click **Configuration** in the PackageExpert toolbar. The **Configuration** view opens.

   Note • *The second menu item in the PackageExpert toolbar toggles between* ***Configuration****,* ***Options****, and* ***Filter****. If* ***Configuration*** *is not displayed, click the arrow next to the menu item and select* ***Configuration*** *from the list.*

2. Select the tests that you want PackageExpert to run for subsequent test runs.

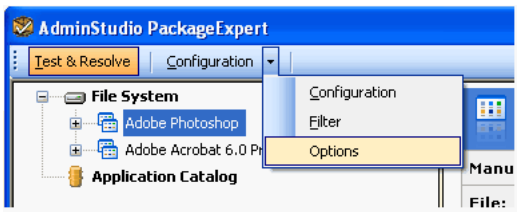
   Tip • *You can select or unselect all of the tests in a test category by selecting or unselecting the test category check box.*

Specifying the Resolution Method

On the **Options** view, you can specify whether you want to resolve errors by creating a transform (`.mst`) file or by directly editing the `.msi` package.

Task: ***To specify the resolution method:***

1. Open PackageExpert.
2. In the toolbar, click the down arrow next to **Configuration** and select **Options**. The **Options** view opens.

3. Select one of the following options to specify the resolution method you want PackageExpert to use when automatically resolving errors:
   - **Original MSI file**—Select this option to have PackageExpert directly edit the `.msi` file to resolve errors.
   - **A newly created transform (MST) file**—Select this option to have PackageExpert create a transform (`.mst`) file to resolve errors.

Testing Packages and Viewing Results

When testing with PackageExpert, you can choose to test one external Windows Installer package, or a selected group of packages in an Application Catalog. Also you can view test results in both PackageExpert and ConflictSolver, and can customize those results by excluding errors or changing the severity of errors.

Table 16-2 • Testing Packages and Viewing Results

Section	Description
Testing Packages	This section includes the following topics: • Testing External Packages • Testing Packages in an Application Catalog • Testing Packages From ConflictSolver
Viewing and Managing Test Results	This section includes the following topics: • Viewing Test Results in PackageExpert • Viewing Test Results in ConflictSolver • Publishing Test Results to the Application Catalog • Deleting Test Results • Customizing Test Results • Adding Ad-Hoc Test Results

Testing Packages

You can use PackageExpert to test packages that are stored on a local file system or in an Application Catalog. You can also select a package in ConflictSolver and use PackageExpert to test it.

- Testing External Packages
- Testing Packages in an Application Catalog
- Testing Packages From ConflictSolver

Testing External Packages

You can test an external package using PackageExpert.

- **Multiple packages can be open**—You can have multiple packages open PackageExpert at the same time. All will be listed in the **File System** package tree.

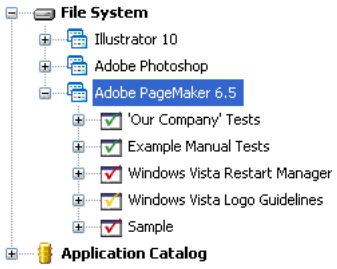


Figure 16-3: Package Tree With Multiple External Packages Open

- **Packages will remain open across sessions**—Each time you open PackageExpert, the packages that were open when you exited previously will remain open.
- **You can remove packages**—You can remove a package from the **File System** package tree by selecting it and then selecting **Remove Package** from the context menu. This action does not affect the package's underlying Windows Installer file; it is not deleted.

To test a package that is stored on a local file system, perform the following steps.

Task: ***To test an external package:***

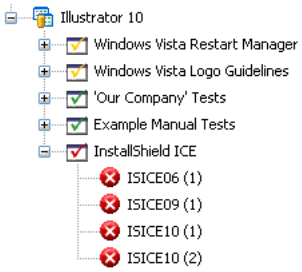
1. Select the **File System** node in the **Test & Resolve** view and select **Open Package** from the context menu. The **Select a Package** dialog box opens.
2. Click **Browse** and select the Windows Installer package that you want to test.
3. Optionally, select transform files (`.mst`) to include in the test process.

   If any transform files exist in the same directory as the `.msi` file you selected, they are automatically listed in the **Transform List**, but are not selected.

   - **To include listed transform files**—Select the transform files that you want to include in PackageExpert testing.
   - **To add additional transform files**—Click **Browse** and select a transform file to add it to the Transform list, and then select it to include it in testing.
4. Click **Test** to open the selected package and perform testing on the package.

Note • *After you have selected a package, the **Test** button becomes enabled. If you click **Test**, the tests will be run. However, if you click **Cancel**, the package will open, but the Test Category nodes will not be displayed below the package icon, and the right pane will state that none of the tests have been run.*

Messages are listed in the Output window, and for each error that is found, a message node is listed under the appropriate Test Category node:



If one test finds more than one issue in the package that produces an error, multiple message nodes, sequentially numbered, are listed under the **Test Category** node.

To view specific information about an error message, select it and view the **Error Details** in the right pane:

Error Details

Item	Details
Test Name	**Restart Manager Files In Use Dialog** / Error Number 1
Description	The 'MsiRMFilesInUse' dialog is needed to support the Restart Manager on Windows Vista.
Status	Ran
Details	Checks to see if the MsiRMFilesInUse dialog is present in the Windows Installer (.msi) package.
Help	TestHelpTopics\Restart Manager Files In Use Dialog.htm

For more information on test results, see Viewing and Managing Test Results.

Testing Packages in an Application Catalog

To open an Application Catalog and test a package or group of packages, perform the following steps:

Task: ***To test packages in an Application Catalog:***

1. On the **Test & Resolve** view, select the **Application Catalog** node. The **Connect Application Catalog** dialog box opens and you are prompted to log on to either the AdminStudio Enterprise Server database or a standalone Application Catalog database.
2. Connect to an Application Catalog as described in Connecting to an Existing Application Catalog. When you are connected, that Application Catalog's groups and packages are listed in the package tree under the **Application Catalog** node.

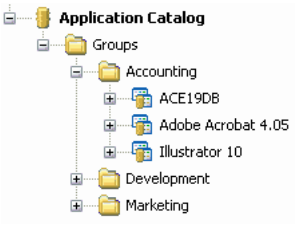
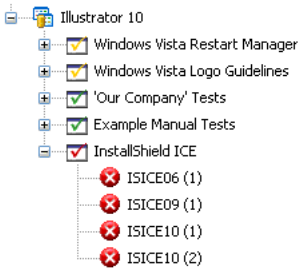


Figure 16-4: Package Tree With an Application Catalog Open

3. To test a package or group of packages in the Application Catalog, select either a package or a package group and select **Test Package** or **Test Group** from the context menu

   Messages are listed in the Output window, and for each error that is found, a message node is listed under the appropriate Test Category node:



   If one test finds more than one file or component in the package that produces an error, multiple message nodes, sequentially numbered, are listed under the **Test Category** node.

   To view specific information about an error message, select it and view the **Error Details** in the right pane:

Error Details

Item	Details
Test Name	**Restart Manager Files In Use Dialog** / Error Number 1
Description	The 'MsiRMFilesInUse' dialog is needed to support the Restart Manager on Windows Vista.
Status	Ran
Details	Checks to see if the MsiRMFilesInUse dialog is present in the Windows Installer (.msi) package.
Help	TestHelpTopics\Restart Manager Files In Use Dialog.htm

   For more information on test results, see Viewing and Managing Test Results.

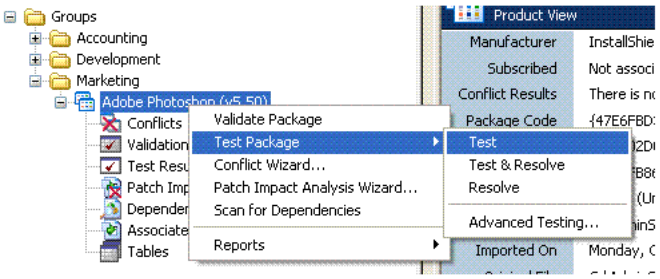
Testing Packages From ConflictSolver

You can select a package in ConflictSolver and then choose to test the package using PackageExpert. Test results will be displayed in the ConflictSolver **Test Results** view.

Note • *By default, all functionality for interacting with the Application Catalog, including conflict detection and resolution, is available in Application Manager. Unless the* ***Turn off display of conflict detection and resolution options in Application Manager*** *option on the* ***General*** *tab of the* ***Options*** *dialog box is selected, you can also perform these PackageExpert tasks using the Application Manager interface. See Application Manager and ConflictSolver for more information.*

Task: ***To test a package and resolve errors from ConflictSolver:***

1. Open the ConflictSolver **Product** view.
2. Select the package that you want to test, point to **Test**, and select one of the following options from the context menu:

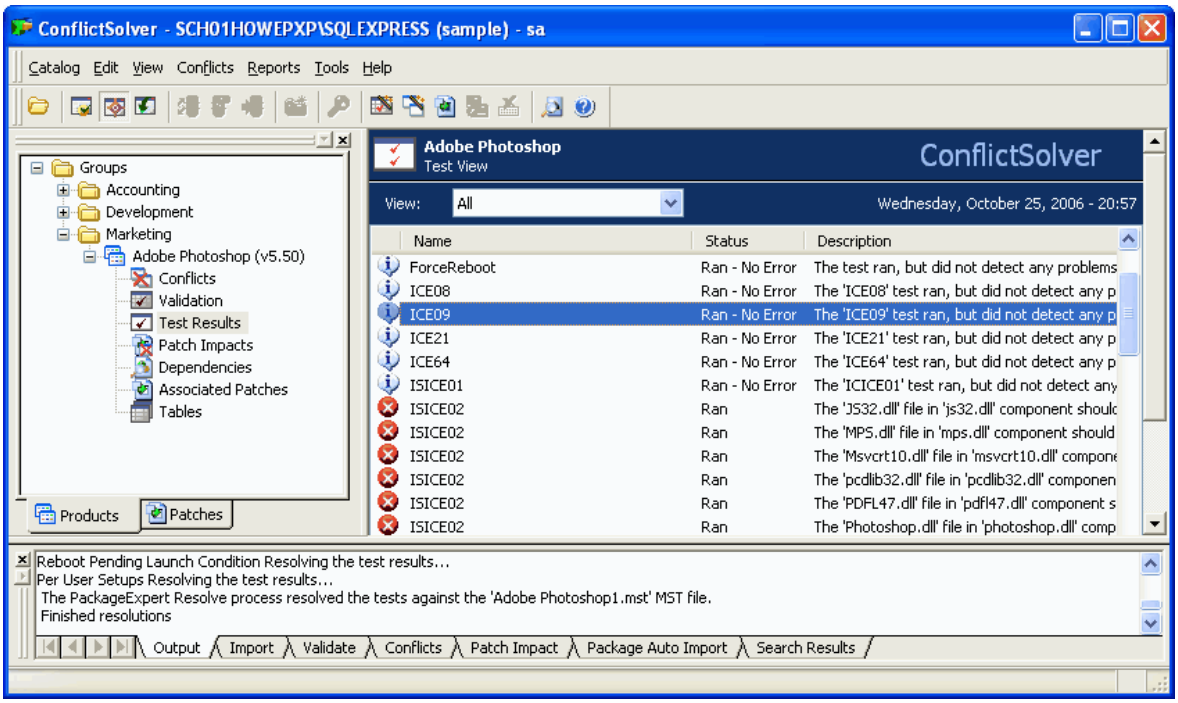


- **Test**—Select to have PackageExpert test the package but not resolve the automatically resolvable errors.
- **Test & Resolve**—Select to have PackageExpert test the package and resolve the automatically resolvable errors.
- **Resolve**—Select to have PackageExpert resolve the package's automatically resolvable errors.
- **Advanced Testing**—Select to open this package in PackageExpert and perform the testing using that interface. You will be prompted to connect to the database that you are currently connected to in ConflictSolver

If you select **Test** or **Test & Resolve**, the following message will appear in the ConflictSolver Output window, followed by other progress messages.

```
PackageExpert is testing the 'MyPackageName' package...
```

3. When testing is complete, select the **Test Result** node in the package tree to open the ConflictSolver **Test Results** view.



This list of messages can be filtered by **All**, **Ran - No Error**, **Ran**, or **Resolved** by making a selection from the **View** list.

Viewing and Managing Test Results

You can view test results in both PackageExpert and ConflictSolver. In PackageExpert, you can change the severity of a message, exclude messages from the listing, and add ad-hoc test results. For packages in an Application Catalog, you can publish a package's test results in the catalog.

- Viewing Test Results in PackageExpert
- Viewing Test Results in ConflictSolver
- Publishing Test Results to the Application Catalog
- Deleting Test Results
- Customizing Test Results
- Adding Ad-Hoc Test Results

Viewing Test Results in PackageExpert

You view both individual and summary test results in the **Test & Resolve** view. The information displayed varies depending upon what is selected in the package tree:

- Information Shown for an Error Message
- Test Result Views
- Toggling Between Local and Published Test Results

Information Shown for an Error Message

Test result messages are displayed in the right pane of the **Test & Resolve** view when a **Package**, **Test Category**, or **Message** node is selected. For each error that is found, a message node is listed under the appropriate Test Category node:

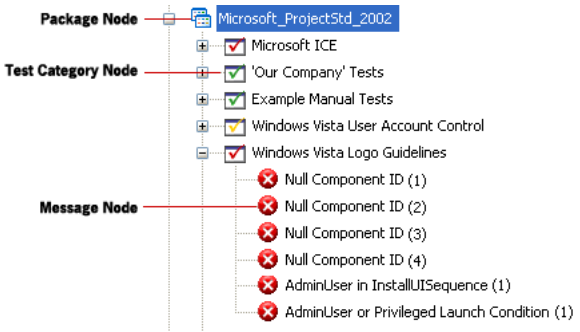


Figure 16-5: Nodes in the PackageExpert Test & Resolve View

If one test finds more than one file or component in the package that produces an error, multiple message nodes, sequentially numbered, are listed under the test category node. In the above example, the **Null Component ID** test in the **Windows Vista Logo Guidelines** test category produced four errors.

Information Shown in Summary Listings

When a **Package** or **Test Category** node is selected, a summary listing of all of the errors found in that package or test category is shown. For each error, the following information is displayed, explaining the cause of the error:

	Name	Status	Description	Help
⊗	ICE08	Ran	The 'Global_Authoring_OPCRemoveMax' component has the same ComponentId as another component and it probably shouldn't.	Info

Figure 16-6: Message Information in Summary Listing

If you click the **Info** link in the **Help** column, a help topic explaining the details of the test and how it is resolved opens.

If a test has not yet been run, a message for the test appears in the list with a **Status** of **Not Run**

	Name	Status	Description	Help
	Desktop Icons	Not run	The test has not yet been run.	Info

Information Shown When Message Node is Selected

When a message node is selected, **Error Details** for that error message are displayed in the right pane. Additional details about the test are provided, as is a link to a help topic explaining that test.

Error Details

Item	Details
Test Name	**Restart Manager Files In Use Dialog** / Error Number 1
Description	The 'MsiRMFilesInUse' dialog is needed to support the Restart Manager on Windows Vista.
Status	Ran
Details	Checks to see if the MsiRMFilesInUse dialog is present in the Windows Installer (.msi) package.
Help	TestHelpTopics\Restart Manager Files In Use Dialog.htm

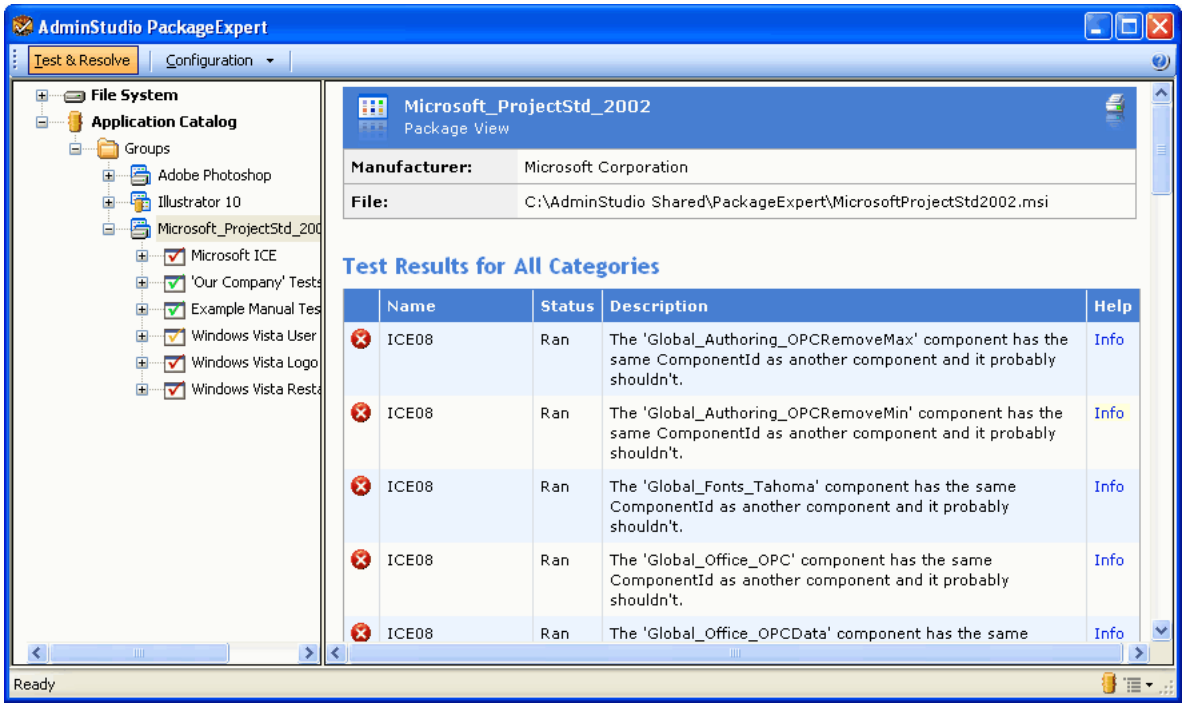
Test Result Views

The test result information displayed varies depending upon what is selected in the package tree, varying between summary information and detailed information on one error message.

- Package View
- Group View
- Application Catalog View
- Test Category View
- Message/Error View

Package View

When a package is selected in the package tree, the test results for all Test Categories are displayed, and are listed in the order of severity:



Group View

When a group in an Application Catalog is selected, a graphic illustrating the number of errors and warnings generated during testing for each package is listed:

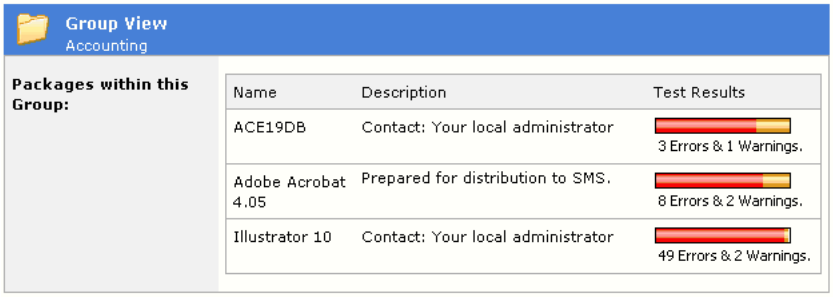


Figure 16-7: Group Summary Information

Application Catalog View

When the Application Catalog node is selected, and you are connected to an Application Catalog, two graphics illustrating the following statistics appear:

- **Percentage of Application Catalog Tested**—The percentage of the number of packages in the Application Catalog that have been tested is listed.
- **Number of Errors and Warnings**—The total number of error and warning messages generated during the testing of packages in the Application Catalog is listed.

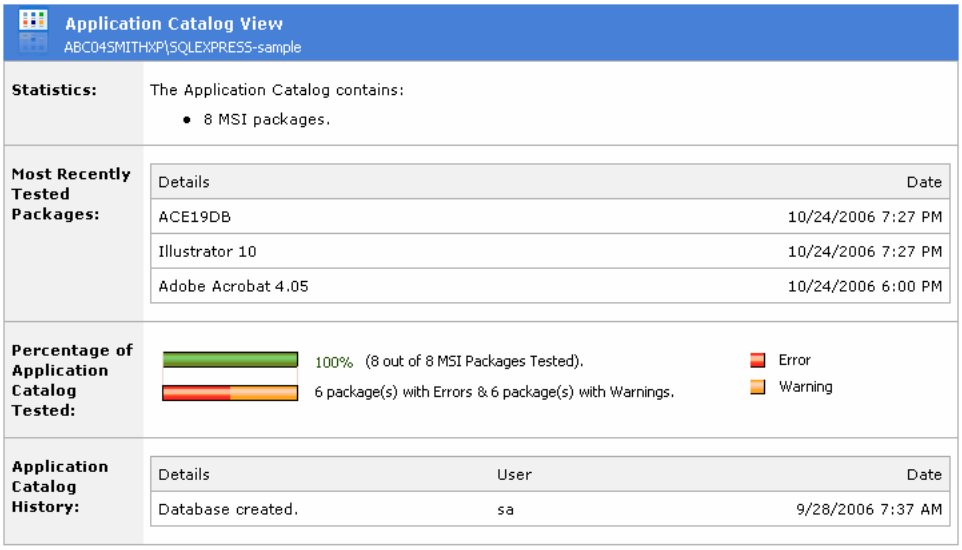


Figure 16-8: Application Catalog Summary Information

Test Category View

When a Test Category node is selected test results for all of the tests in that category are listed, and are listed in order of severity.

Test Results for Category: Windows Vista Logo Guidelines

	Name	Status	Description	Help
✖	Null Component ID	Ran	Component Global_Access_Core has a null ComponentID	Info
✖	Null Component ID	Ran	Component Global_FrontPageCore_CoreClient has a null ComponentID	Info
✖	AdminUser in InstallUISequence	Ran	The installation has actions in the InstallUISequence table that have conditions using the 'AdminUser' property.	Info
✖	AdminUser or Privileged Launch Condition	Ran	The installation has LaunchCondition entries using the 'AdminUser' or 'Privileged' properties.	Info
⚓	Nested Custom Actions	Ran - No Error	The 'Nested Custom Actions' test ran, but did not detect any problems.	Info
⚓	Per User Setups	Ran - No Error	The test ran, but did not detect any problems.	Info
⚓	ForceReboot	Ran - No Error	The test ran, but did not detect any problems.	Info

Figure 16-9: Test Category Summary Information

Message/Error View

When a message node in the package tree is selected, detailed information on that specific message is displayed in the top half of the pane—including the exact text of the selected error message, and a help topic that explains what the test is checking for and how errors are resolved is displayed in displayed in the lower half of the pane.

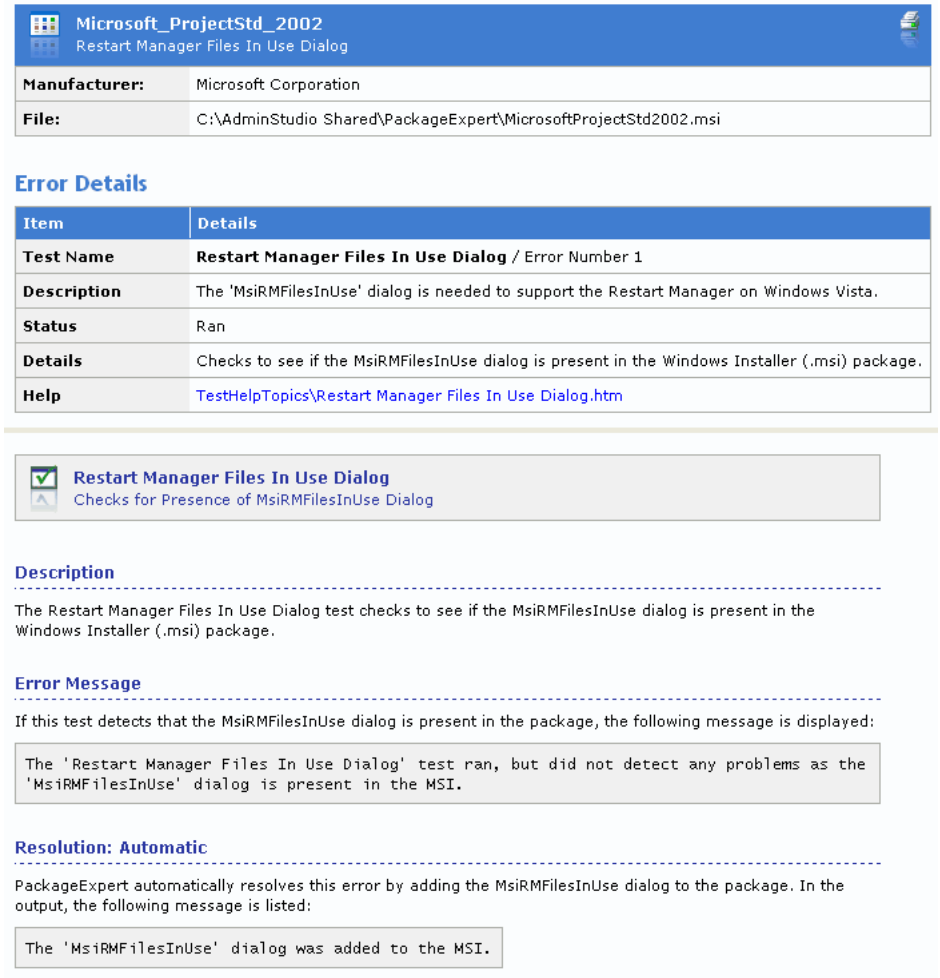
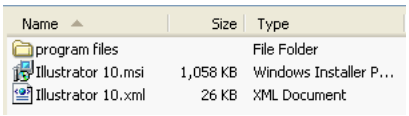


Figure 16-10: Message Node Detailed Information

Publishing Test Results to the Application Catalog

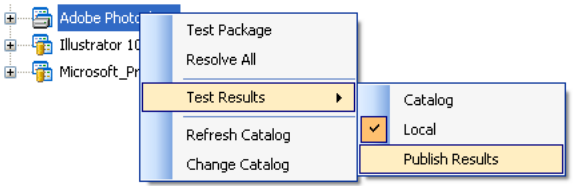
Every time you test a package in PackageExpert, the results are stored in a local `.xml` file in the same location as the `.msi` file.



When testing an Application Catalog package, you can choose to publish the package's test results in the local `.xml` file to the Application Catalog.

Task: ***To publish test results to an Application Catalog:***

1. In the Application Catalog package tree, select a package.
2. Right-click, point to **Test Results**, and click **Publish Results** on the context menu. You are prompted to confirm your selection.



3. Click **OK**. The local test results for that package are saved to the Application Catalog.

Publishing Test Results When Prompted

If you have tested an Application Catalog package, and you have not yet published its test results, when you disconnect from that Application Catalog or exit PackageExpert, the **Unpublished Packages** dialog box opens, prompting you to publish the test results:

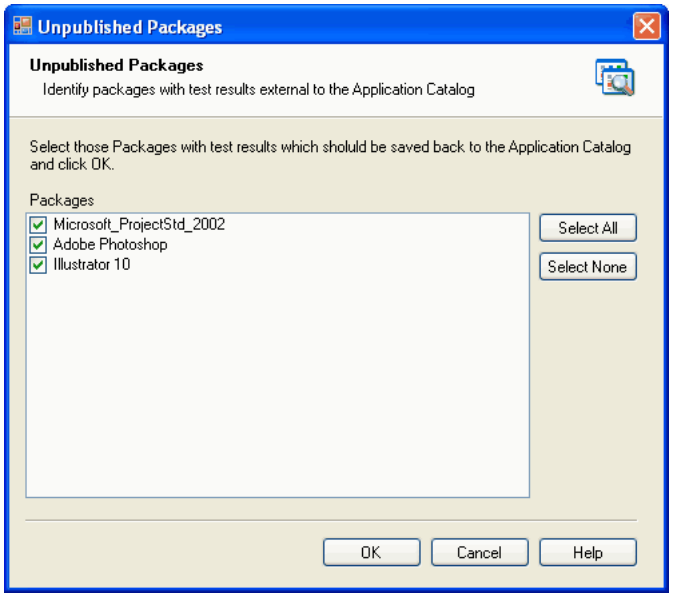


Figure 16-11: Prompt to Publish Unpublished Test Results

You are also prompted to publish test results if you disconnect from an Application Catalog and reconnect to it during the same PackageExpert session.

You can choose to publish the test results for all packages or only those packages that you select.

Toggling Between Local and Published Test Results

When viewing test results for Application Catalog packages, you can toggle between the locally saved test results (**Local**) and the results that have been published to the Application Catalog (**Catalog**).

The package icon used in the package tree varies depending upon which test results are being displayed:

Table 16-3 • Package Icons Indicating Test Result Mode

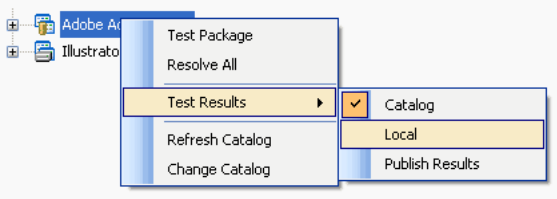
Icon	Mode	Results Being Displayed
	Local	Locally saved test results are being displayed.
	Catalog	Test results that were published to the Application Catalog are being displayed.

If PackageExpert is currently displaying the **Catalog** test results, but the **Catalog** results are out of sync with the results in the **Local** `.xml` file, an exclamation mark appears on the package icon:

To toggle the PackageExpert test result display between Local and Catalog, perform the following steps.

Task: ***To toggle the display of test results between Local and Catalog:***

1. In the Application Catalog package tree, select a package.
2. Right-click, point to **Test Results**, and click **Local** or **Catalog** on the context menu.



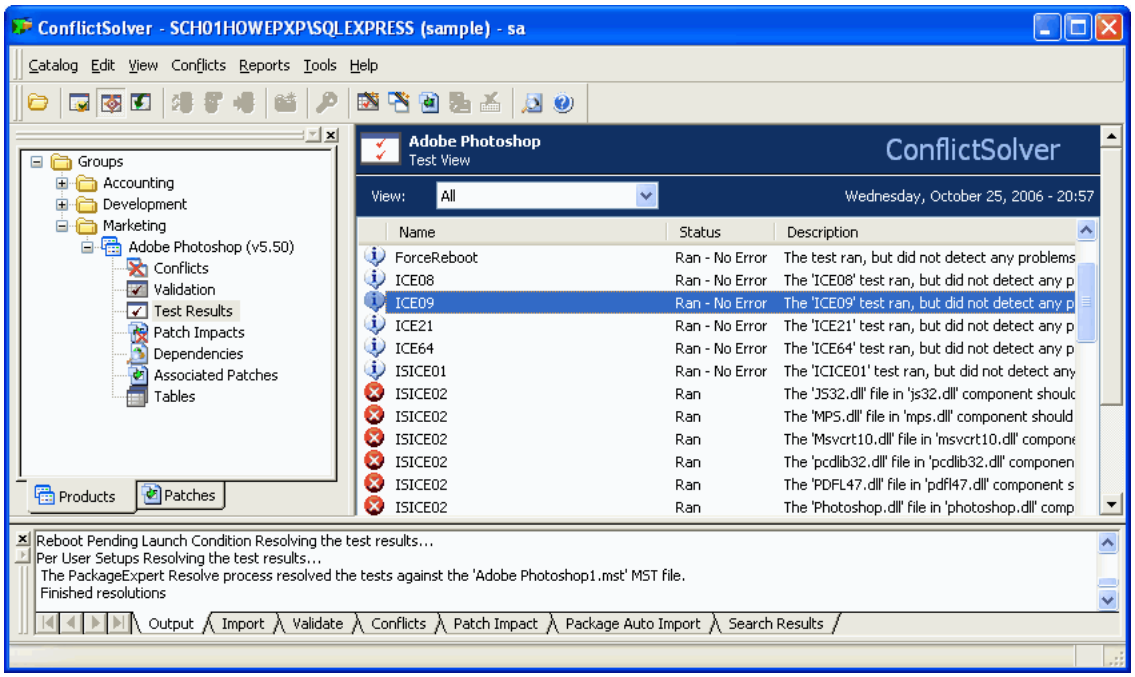
The selected test results are displayed and the package icon changes to indicate the mode you chose.

Viewing Test Results in ConflictSolver

You can test packages from the ConflictSolver interface, and view a summary of the results in the **Test Results** view. The **Test Results** view also displays a package's test results when a package is tested from the PackageExpert interface and the test results are published to the Application Catalog.

Note • *By default, all functionality for interacting with the Application Catalog, including conflict detection and resolution, is available in Application Manager. Unless the* ***Turn off display of conflict detection and resolution options in Application Manager*** *option on the* ***General*** *tab of the* ***Options*** *dialog box is selected, you can also test packages and view test results using the Application Manager interface. See Application Manager and ConflictSolver for more information.*

You open the **Test Results** view by selecting the **Test Result** node in the ConflictSolver package tree.



When the **Test Results** node is selected, any results that were previously published to the Application Catalog regarding this package are displayed.

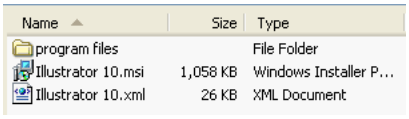
This list of messages can be filtered by **All**, **Ran - No Error**, **Ran**, or **Resolved** by making a selection from the **View** list.

Deleting Test Results

You can delete a package's local test results and, for packages in the Application Catalog, the test results published to the Application Catalog.

Deleting a Package's Local Test Results

Every time you test a package in PackageExpert, the results are stored in a local `.xml` file in the same location as the `.msi` file. If you want to delete these test results, delete the `.xml` file.



Deleting a Package's Published Test Results

You can delete PackageExpert test results that have been published to the Application Catalog using the Application Manager interface.

To delete a package's published test results, perform the following steps:

Task: ***To delete a package's published test results:***

1. Open Application Manager.
2. Select a package in the package tree, point to **Delete** and click **Persisted Test Information** from the context menu. The PackageExpert test results for that package are deleted from the Application Catalog.

Customizing Test Results

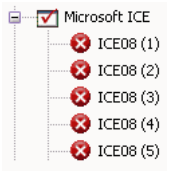
You are able to customize PackageExpert test results to accommodate unique or special circumstances at your organization by changing the severity of messages and by suppressing the display of a specific instance of an error.

- Changing the Severity of a Specific Error
- Suppressing the Display of Specific Errors

Changing the Severity of a Specific Error

Using the **Test Result Filter Editor**, you can change the severity of a specific instance of a message from its original severity to **Error**, **Warning**, or **Information**.

You are changing the severity of a *specific instance* of a message generated by a test, not the severity of all messages generated by that test. For example, suppose the ICE08 test produced five error messages:



You consider messages 1 through 4 to be valid errors. However, suppose that message 5 concerns a particular component of the package that no one at your organization is likely to use, and therefore you do not want this particular error message to have a severity of **Error**. You can change the severity to **Warning** to document the decreased importance of this particular message at your organization.

Error Details

Item	Details
Test Name	**ICE08** / Error Number 5
Description	The 'Global_RegFiles_GifRepair' was provided with a new ComponentId.
Status	Resolved
Details	Validates that the Component table contains no duplicate Component IDs (GUIDs).
Help	TestHelpTopics\Microsoft ICE_ICE08.htm

Figure 16-12: Error Details of Error Number 5 of the ISICE08 Test

Information about changing the severity of a message is presented in the following topics:

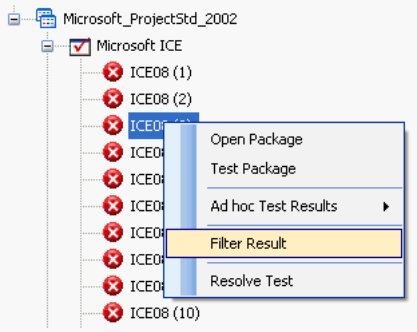
- Changing the Severity of a Message
- Editing a Result Filter
- Deleting a Result Filter

Changing the Severity of a Message

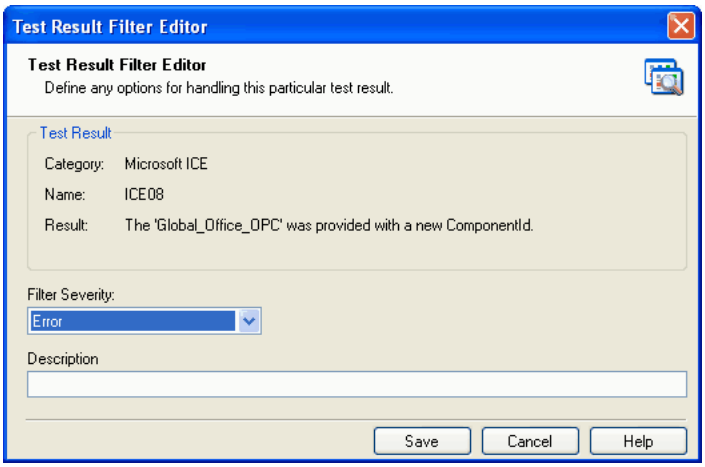
To change the severity of a message, perform the following steps:

Task: ***To change the severity of a message:***

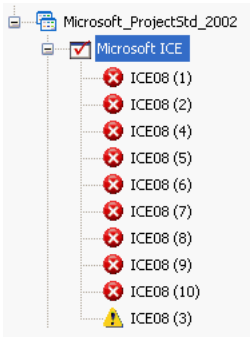
1. On the **Test & Resolve** view, select the specific message that you want to modify and select **Filter Result** from the context menu.



The **Test Result Filter Editor** dialog box opens:



2. From the **Filter Severity** list, select the severity that you want to change this message to: **Error**, **Warning**, or **Information**.
3. In the **Description** box, enter an explanation for why you changed the severity level of this message.
4. Click **Save**. This error now displays the icon of its new severity level:



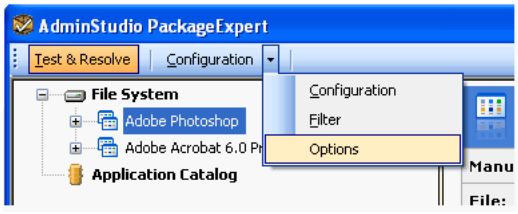
If the same problem is encountered in another package or in a subsequent test run of this package, the severity level that you defined will continue to be used.

Editing a Result Filter

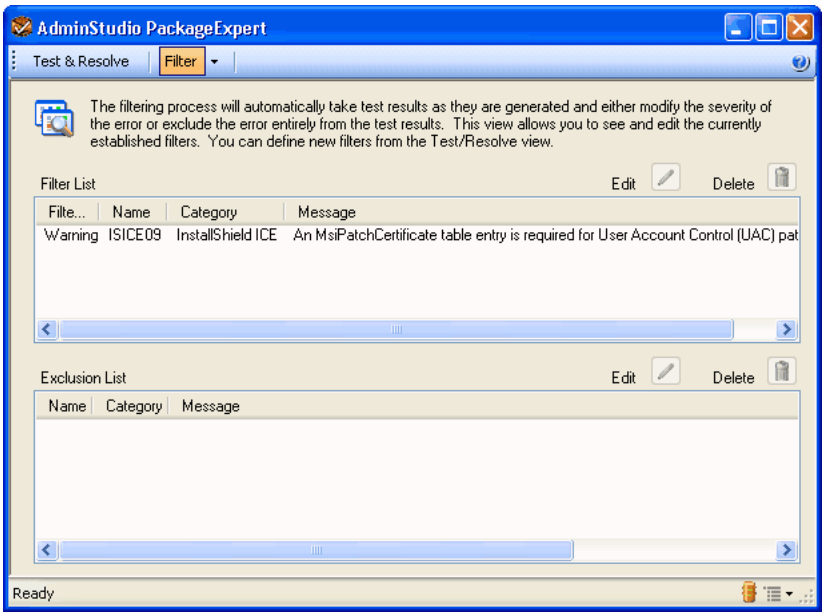
To edit an existing filter, perform the following steps.

Task: ***To edit an existing filter:***

1. In the PackageExpert toolbar, click the down arrow next to the second menu item and select **Filter**.



The **Filter** view opens, listing all of the defined severity filters in the **Filter List**:



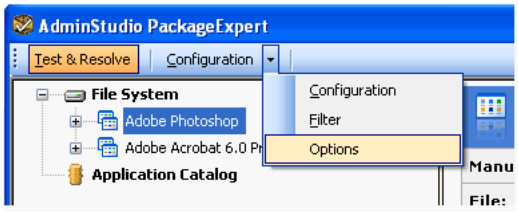
2. Select the filter that you want to edit, and click **Edit**. The **Test Result Filter Editor** dialog box opens.
3. Edit the **Filter Severity** and/or **Description** fields and click **Update** to save your edits.

Deleting a Result Filter

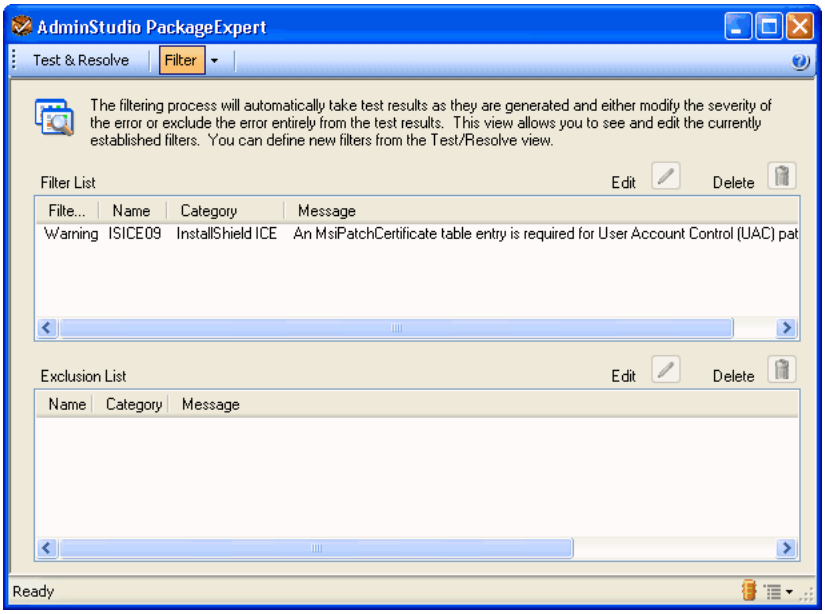
To delete a filter, so that the a message will either have its default severity level or will return to the list after being excluded, perform the following steps.

Task: ***To delete a result filter and reset to original severity:***

1. In the PackageExpert toolbar, click the down arrow next to the second menu item and select **Filter**.



The **Filter** view opens, listing all of the defined severity filters in the **Filter List**:

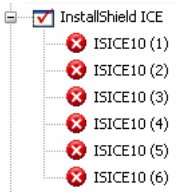


2. Select the filter that you want to delete, and click **Delete**. You are prompted to confirm the deletion.
3. Click **OK**.

Suppressing the Display of Specific Errors

Using the **Test Result Filter Editor**, you can choose to exclude the display of a *specific instance* of a message.

You are suppressing a *specific instance* of a message generated by a test, not suppressing all messages generated by that test. For example, suppose the ISICE10 test produced six error messages:



You consider messages 1, 2, 4, 5, and 6 to be valid errors. However, suppose that message 3 concerns a particular component of the package (`ISSelfRegisterCosting` in the example below) that is no longer used at your organization, and therefore you want to suppress the display of this particular error message.

Error Details

Item	Details
Test Name	**ISICE10** / Error Number 3
Description	The custom action 'ISSelfRegisterCosting' is not documented in table 'ISCustomActionReference'.
Status	Ran
Details	Verifies that each custom action in your installation is documented by validating that each entry in the CustomAction table has a corresponding ISCustomActionReference table entry.
Help	TestHelpTopics\InstallShield ICE_ISICE10.htm

Figure 16-13: Error Details of Error Number 3 of the ISICE10 Test

Information about suppressing the display of an error message is presented in the following topics:

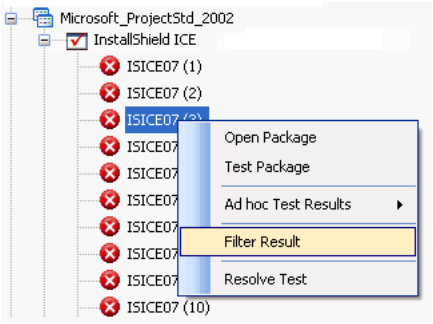
- Suppressing the Display of a Message
- Editing a Result Filter
- Deleting a Result Filter

Suppressing the Display of a Message

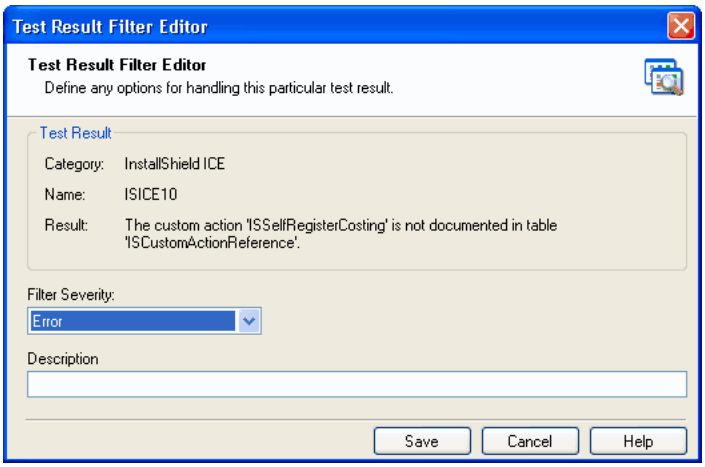
To suppress the display of a message, perform the following steps:

Task: ***To suppress the display of a message:***

1. On the **Test & Resolve** view, select the specific message that you want to suppress and select **Filter Result** from the context menu.



The **Test Result Filter Editor** dialog box opens:



2. From the **Filter Severity** list, select **Exclude**.
3. In the **Description** box, enter an explanation for why you chose to exclude this message.
4. Click **Save**. The message is no longer listed in the package tree

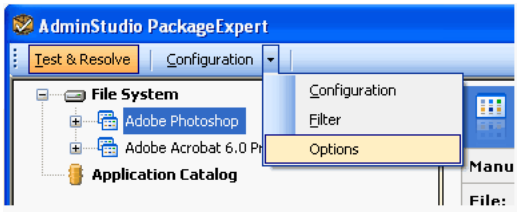
   If the same problem is encountered in another package or in a subsequent test run of this package, the error message will not be displayed.

Editing a Result Filter

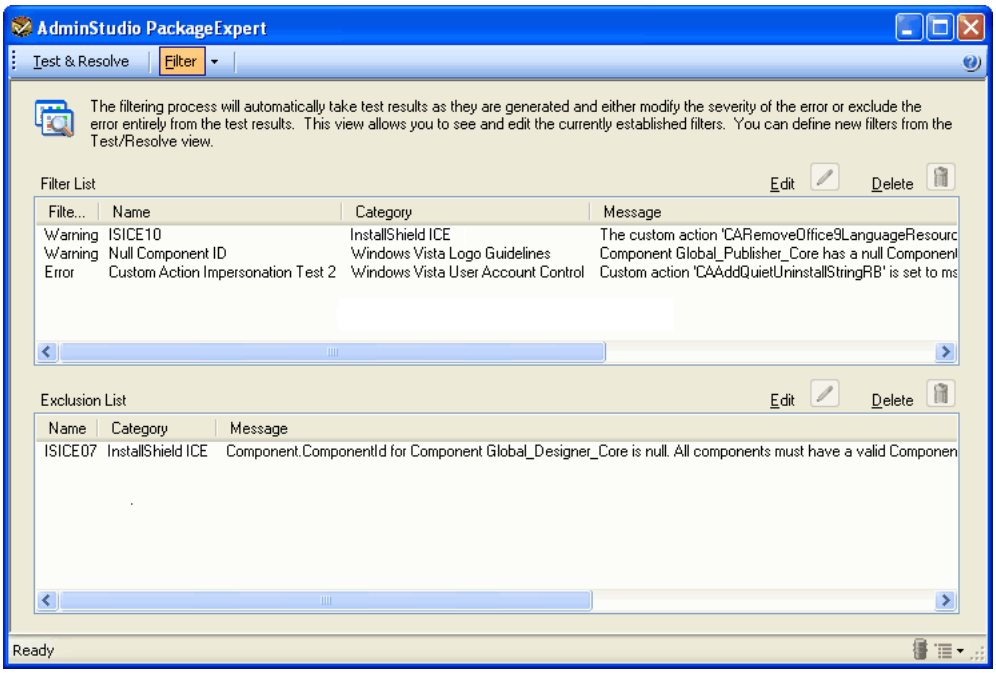
To edit an existing filter, perform the following steps.

Task: ***To edit an existing filter:***

1. In the PackageExpert toolbar, click the down arrow next to the second menu item and select **Filter**.



The **Filter** view opens, listing all of the defined Exclude filters in the **Exclusion List**:



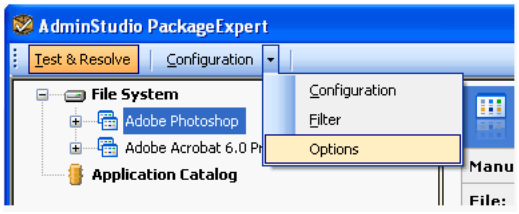
2. Select the filter that you want to edit, and click **Edit**. The **Test Result Filter Editor** dialog box opens.
3. Edit the **Filter Severity** and/or **Description** fields and click **Update** to save your edits.

Deleting a Result Filter

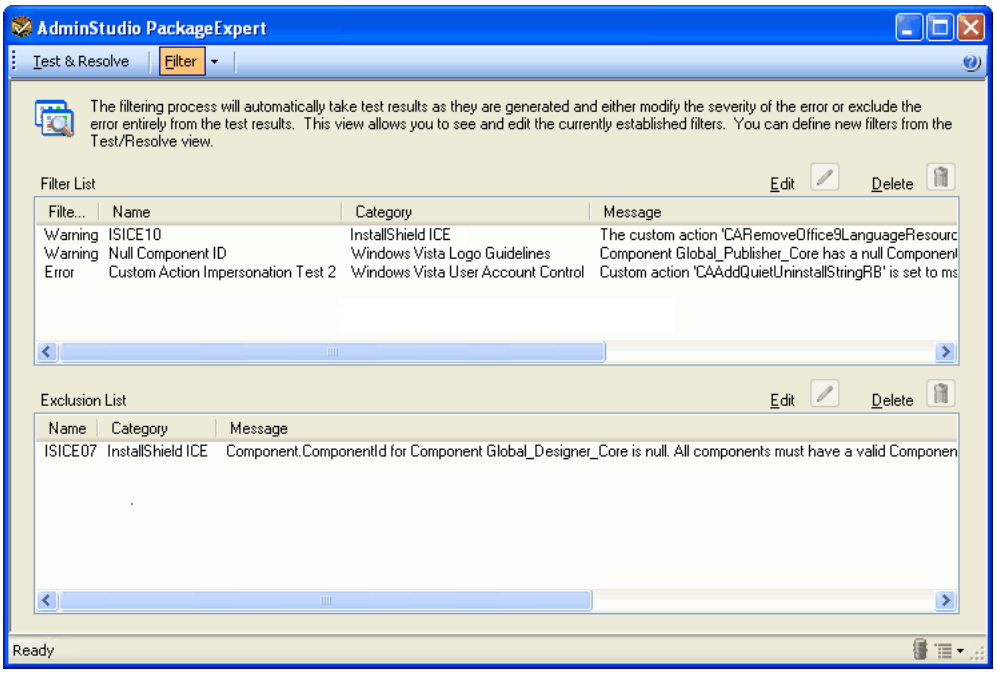
To delete a filter, so that this message will no longer be excluded, perform the following steps.

Task: ***To delete a result filter and reset to original severity:***

1. In the PackageExpert toolbar, click the down arrow next to the second menu item and select **Filter**.



The **Filter** view opens, listing all of the defined exclusion filters in the **Exclusion List**:



2. Select the filter that you want to delete, and click **Delete**. You are prompted to confirm the deletion.
3. Click **OK**.

Adding Ad-Hoc Test Results

You can manually add a test result message to a package. You might want to do this if you tested the package using QualityMonitor, InstallShield, or another tool, and you noticed something which is unexpected and which you want to document. There might be other scenarios such as an installation failure when you might want to document an error.

The following example documents an error was encountered during installation, which caused installation failure.

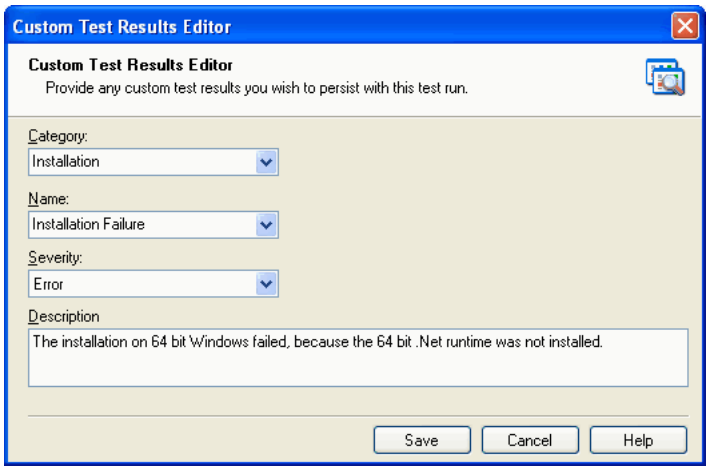


Figure 16-14: Example of a Message Added to Document Ad-Hoc Testing

When you click **Save** on the **Custom Test Results Editor** dialog box, the new message will be displayed in the package tree, and will be documented in the **Error Details** area in the right pane, as shown in the following Figure:

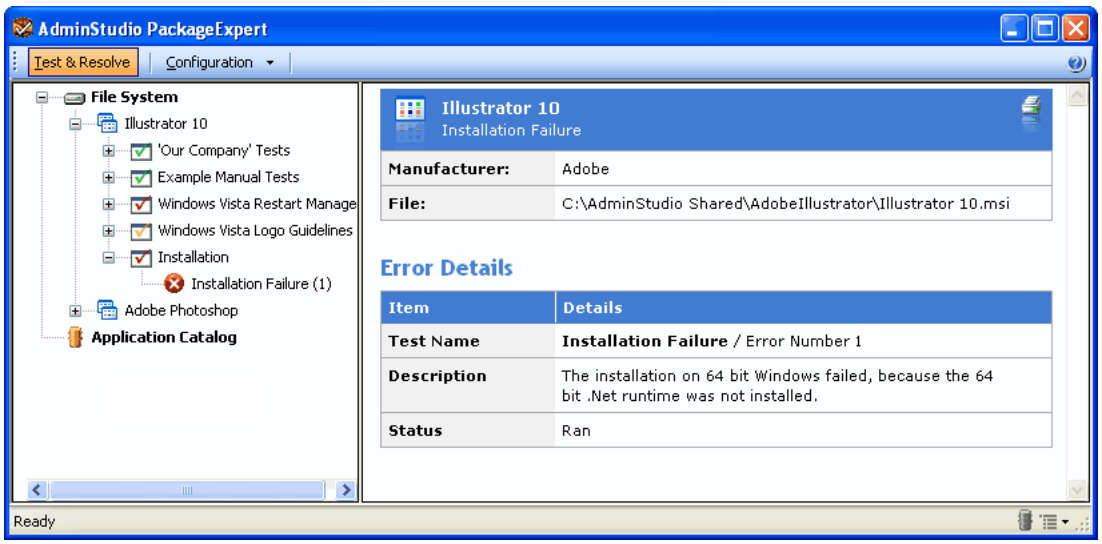


Figure 16-15: Display of a Message Added to Document Ad-Hoc Testing

When this problem is corrected, you could then run the installation on 64 bit Windows, confirm the problem is corrected, and delete or demote the severity of the test result.

The following procedures are provided:

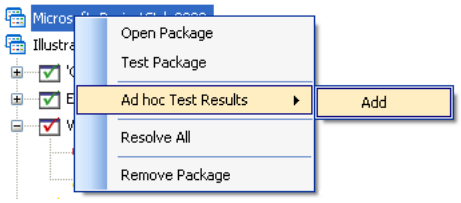
- Adding a Test Result Message
- Editing a Test Result Message
- Deleting a Test Result Message

Adding a Test Result Message

To add a test result message, perform the following steps.

Task: ***To add a test result message to a package:***

1. Select the package in the package tree that you want to add a test result message to.
2. Right-click on the package, point to **Ad hoc Test Results** and click **Add** on the context menu. The **Custom Test Results Editor** dialog box opens.



3. In the **Category** list, either enter a new name for a test category where you want to display this message, or select an existing category from the list.

Note • *Custom test results only reside in custom test result categories (test categories that you created). If a category has already been added, then it will be listed in the* ***Category*** *list. You can create a new category by entering a category name in the* ***Category*** *box.*

4. In the **Name** list, either enter a new name for the test, or—if the same test has generated more than one error—select an existing name from the list.
5. From the **Severity** list, select **Error**, **Warning**, or **Information**.
6. In the **Description** box, document the error that you discovered during ad-hoc testing. This information will be displayed under **Error Details** in the right pane when this test result node is selected.

Editing a Test Result Message

To edit the **Severity** or **Description** of an existing test result message, perform the following steps.

Task: ***To edit a test result message:***

1. Select the message node in the package tree that you want to edit.
2. Right-click on the message node, point to **Ad hoc Test Results** and click **Edit** on the context menu. The **Custom Test Results Editor** dialog box opens, with the **Severity** and **Description** fields enabled.
3. Make edits to the **Severity** and **Description** fields and click **Update**.

Deleting a Test Result Message

To delete an existing test result message, perform the following steps.

Task: ***To delete a test result message:***

1. Select the message node in the package tree that you want to delete.
2. Right-click on the message node, point to **Ad hoc Test Results** and click **Delete** on the context menu. You are prompted to confirm the deletion.
3. Click **OK**.

Resolving Errors

For some tests, PackageExpert can automatically resolve any errors that are found. PackageExpert will either modify the `.msi` package or create a transform (`.mst`) file to apply the changes, depending upon what you have specified on the **Options** view, as described in Specifying the Resolution Method.

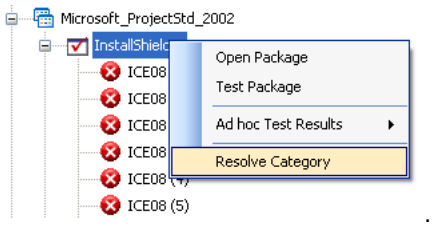
You can choose to resolve all errors for a package, all errors in a Test Category, or just one specific error.

Task: ***To automatically resolve errors:***

1. In the **Test & Resolve** view, review the messages generated during testing for a package.

   Note • *To determine whether a test is automatically resolvable, select the message node, and check to see whether* ***Resolution: Automatic*** *or* ***Resolution: Manual*** *is displayed in the help topic in the right pane.*

2. Select a package, a test category, or a message node, and select **Resolve All**, **Resolve Category**, or **Resolve Test** from the context menu



   Messages appear in the Output window informing you of each resolution. When the resolutions are complete, the name of the file that was edited is listed: either a transform file or the **.msi** package, as specified on the **Configuration** tab. For example:

```
Resolution started...
ICE08 Resolving the test results...
ICE08 Resolving the test results...
ICE08 Resolving the test results...
```

```
ICE08 Resolving the test results...
ICE09 Resolving the test results...
ICE09 Resolving the test results...
ICE09 Resolving the test results...
ICE21 Resolving the test results...
 The PackageExpert Resolve process resolved the tests against the 'MicrosoftProjectStd2002.mst'
MST file.
Finished
```

When a Warning or Error message is resolved, a green check mark is listed next to that message in the Test Results list:

Test Results for Category: Microsoft ICE

	Name	Status	Description	Help
✓	ICE08	Resolved	The 'Global_Authoring_OPCRemoveMax' was provided with a new ComponentId.	Info
✓	ICE08	Resolved	The 'Global_Authoring_OPCRemoveMin' was provided with a new ComponentId.	Info
✓	ICE08	Resolved	The 'Global_Fonts_Tahoma' was provided with a new ComponentId.	Info
✓	ICE08	Resolved	The 'Global_Office_OPC' was provided with a new ComponentId.	Info
✓	ICE08	Resolved	The 'Global_Office_OPCData' was provided with a new ComponentId.	Info

Adding Manual Tests

In PackageExpert, you can add a Manual Test to the package tree by creating an `.xml` file and placing it in the PackageExpert installation directory. By adding Manual Tests to Package Expert, you can document important tests at your organization and monitor whether these tests are performed.

- About Manual Tests
- Creating a Manual Test
- Running and Resolving a Manual Test
- Sample Manual Test

About Manual Tests

A Manual Test is an `.xml` file that contains both instructions (or links to those instructions) on exactly how to perform a Manual Test to evaluate the quality of a Windows Installer package, and instructions on how to resolve any errors that are found.

PackageExpert provides two sample Manual Tests, **Desktop Icons** and **Example**, which are displayed in the package tree:

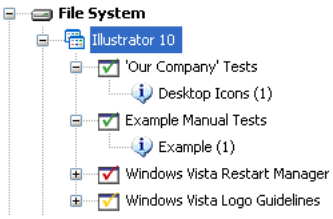


Figure 16-16: Manual Tests Listed in the Package Tree

You perform or resolve a Manual Test by selecting the test and making a selection from the context menu: **Perform Manual Test** or **Resolve Manual Test.**

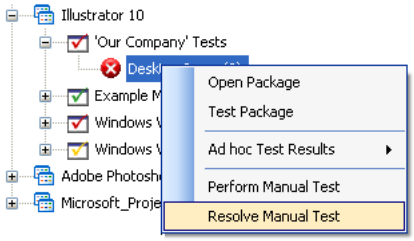


Figure 16-17: Manual Test Selections on the Context Menu

When you select **Perform Manual Test,** the instructions defined in the .xml file are listed on the **Manual Test Perform** dialog box. As you perform each step, you mark each one **Pass** or **Fail**.

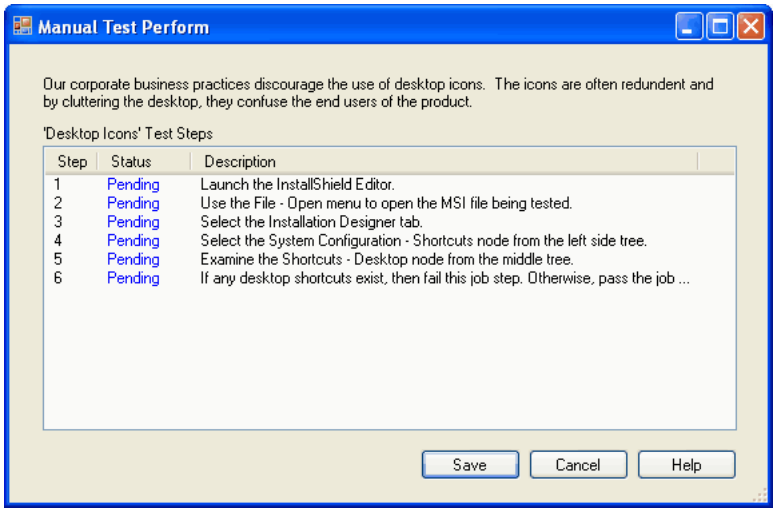


Figure 16-18: Steps to Perform a Manual Test

After you complete all of the steps on the **Manual Test Perform** dialog box, the **Resolve Manual Test** selection is enabled, and the instructions on how to resolve errors discovered by this test are listed on the **Perform Manual Test** dialog box:

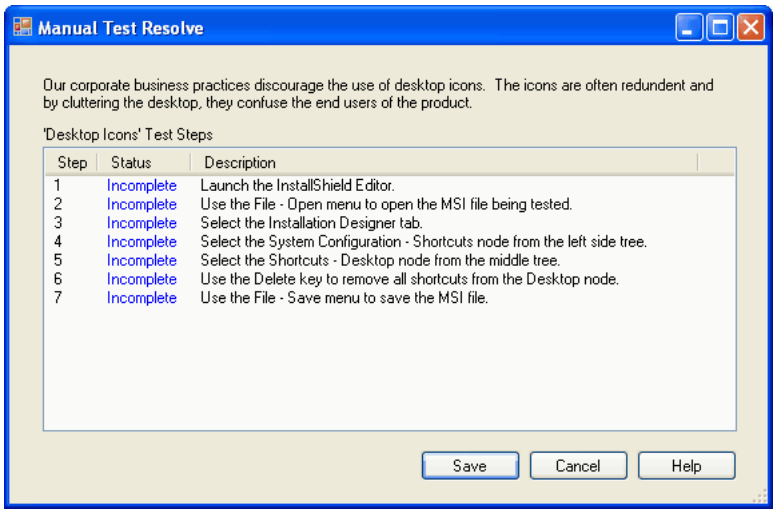


Figure 16-19: Steps to Resolve a Manual Test

To help you learn how to construct your own Manual Test, examine the sample `.xml` file provided by PackageExpert. This file, `ManualTest.xml`, can be found in the `PackageExpert` directory inside the AdminStudio installation directory.

For information on the structure of a Manual Test in an `.xml` file, see Sample Manual Test.

Creating a Manual Test

By adding Manual Tests to Package Expert, you can document important tests at your organization and monitor whether these tests are performed.

A Manual Test is an `.xml` file that contains both instructions (or links to those instructions) on exactly how to perform a Manual Test to evaluate the quality of a Windows Installer package, and instructions on how to resolve any errors that are found.

To write your own Manual Test, perform the following steps:

Task: ***To write a Manual Test:***

1. Locate and open the following file in a code editor:

   `C:\Program Files\AdminStudio\10.0 SP2\PackageExpert\manualtest.xml`

2. Locate and make a copy of a `<TestCase>` element in that XML file:

```
<TestCase name="Example" category="Example Manual Tests" class=""
testtype="Manual" selected="false">
  <TestResults>
    <TestResult state="Not run" severity="Informational" type="Manual" ordinal="1">
      The 'Example' test has not yet been run.
      <TestData></TestData></TestResult>
  </TestResults>
```

```
<TestData>
  <Details>The 'Example' test is run to detect the state of something and respond
  accordingly. You should describe why it is important.</Details>
  <KBArticle>TestHelpTopics\Example.htm</KBArticle>
  <Perform>
    <Incomplete state="Ran - Incomplete" severity="Informational">The 'Example' test has
    not been completely run.</Incomplete>
    <Fail state="Ran" severity="Informational">The 'Example' test ran and failed.</Fail>
    <Pass state="Ran" severity="Error">The 'Example' test ran successfully.</Pass>
    <Steps>
      <Step step="1" state="Pending" type="text">Step 1 - Describe the process with
      inline text.</Step>
      <Step step="2" state="Pending" type="link">Step 2 - Instructions on a web page.
      <Direct>http://www.cnn.com</Direct></Step>
      <Step step="3" state="Pending" type="file">Step 3 - Instructions referencing a
      local file.<Direct>images\info.gif</Direct></Step>
   </Steps>
  </Perform>
  <Resolve>
    <Incomplete state="Ran" severity="Warning">The 'Example' test resolution process is
    incomplete.</Incomplete>
    <Complete state="Resolved" severity="Informational">The 'Example' test resolution
    process completed successfully.</Complete>
    <Steps>
      <Step step="1" state="Incomplete" type="text">Step 1 - Describe the resolution
      process with inline text.</Step>
      <Step step="2" state="Incomplete" type="link">Step 2 - Resolution instructions on a
      web page.<Direct>http://www.msnbc.com</Direct></Step>
    </Steps>
  </Resolve>
 </TestData>
</TestCase>
```

3. Customize the copy of the `<TestCase>` element to define your Manual Test, as described in Sample Manual Test.

4. Open PackageExpert. If, in the `<TestCase>` element you set `selected="true"`, a new node will be listed in the Test Category you entered.

Running and Resolving a Manual Test

PackageExpert provides a sample Manual Test named **Desktop Icons** that you can use to learn how to run and resolve a Manual Test.

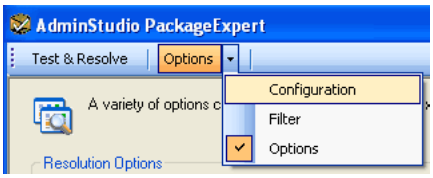
Running a Manual Test

To run the **Desktop Icons** Manual Test, perform the following steps:

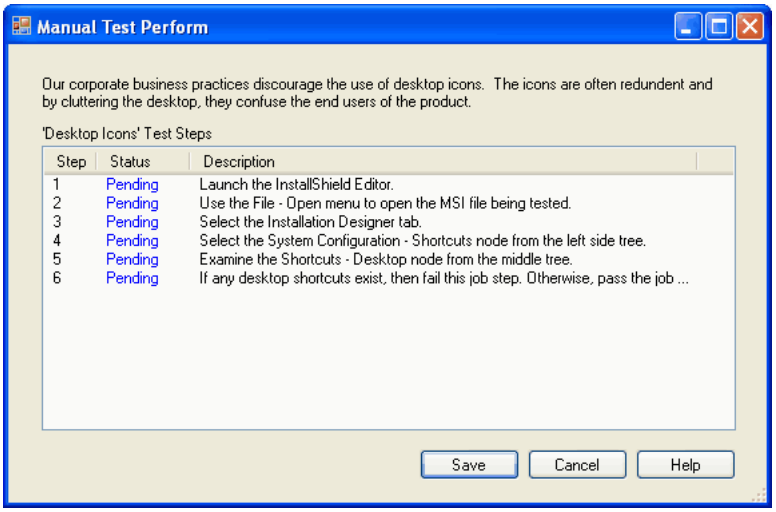
Task: ***To run this test:***

1. Click **Configuration** in the PackageExpert toolbar. The **Configuration** view opens.

   Note • *The second menu item in the PackageExpert toolbar toggles between* ***Configuration****,* ***Options****, and* ***Filter****. If* ***Configuration*** *is not displayed, click the arrow next to the menu item and select* ***Configuration*** *from the list.*

2. Select the manual test that you want to run and resolve.
3. Return to the **Test & Resolve** view and open a package in PackageExpert.
4. Select the **Desktop Icons** node and select **Perform Manual Test** from the context menu. The **Manual Test Perform** dialog box opens, listing the manual steps you need to perform to run this manual tests.

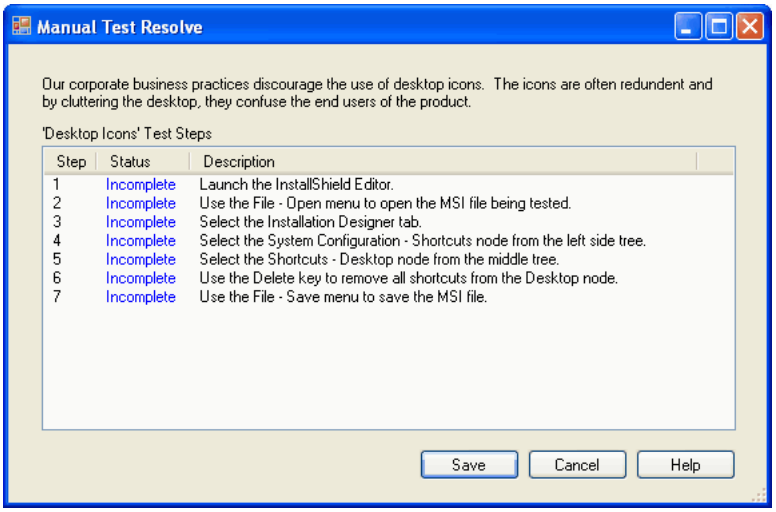
5. Perform each of the steps, and mark each as **Pass** or **Fail** after you have performed it by clicking on the step and selecting **Pass** or **Fail** from the menu.
6. Click **Save**.

Resolving a Manual Test

When performing a manual test, you mark the **Status** of each step as either **Pass**, **Fail**, or **Pending**. If you have set the status of any of the steps to **Fail**, and none of the steps are still in **Pending** status, the **Resolve Manual Test** selection on the context menu is enabled, and you can proceed with resolution.

Task: ***To resolve this test:***

1. With the package open in PackageExpert, select the **Desktop Icons** node and select **Resolve Manual Test** from the context menu. The **Manual Test Resolve** dialog box opens, listing the manual steps you need to perform to resolve this error.



2. Perform each of the steps, and mark each as **Complete** after you have performed it by clicking on the step and selecting **Complete** from the menu.
3. Click **Save**.
4. Select the **Desktop Icons** node. In the **Error Details** that are displayed, this test's **Status** is now **Resolved**, and the **Description** is `The test resolution process is complete and the desktop icons removed`.

Sample Manual Test

PackageExpert includes two sample Manual Tests.

- **Example of a Manual Test**—The **Desktop Icons** Manual Test is a functional example of a Manual Test. It examines a Windows Installer package for shortcuts destined for the Desktop and removes those shortcuts.
- **Template for a Manual Test**—The **Example** manual test is a template that you can use to create your own manual test. It is intended to illustrate the possible functionality and authoring requirements for end-user Manual Tests. Its structure is described below.

The **Example** Manual Test has the following code:

```
<TestCase name="Example" category="Example Manual Tests" class=""
testtype="Manual" selected="false">
  <TestResults>
    <TestResult state="Not run" severity="Informational" type="Manual" ordinal="1">
      The 'Example' test has not yet been run.
      <TestData></TestData></TestResult>
```

```
  </TestResults>
  <TestData>
    <Details>The 'Example' test is run to detect the state of something and respond
    accordingly. You should describe why it is important.</Details>
    <KBArticle>TestHelpTopics\Example.htm</KBArticle>
    <Perform>
      <Incomplete state="Ran - Incomplete" severity="Informational">The 'Example' test has
      not been completely run.</Incomplete>
      <Fail state="Ran" severity="Informational">The 'Example' test ran and failed.</Fail>
      <Pass state="Ran" severity="Error">The 'Example' test ran successfully.</Pass>
      <Steps>
        <Step step="1" state="Pending" type="text">Step 1 - Describe the process with
        inline text.</Step>
        <Step step="2" state="Pending" type="link">Step 2 - Instructions on a web page.
        <Direct>http://www.cnn.com</Direct></Step>
        <Step step="3" state="Pending" type="file">Step 3 - Instructions referencing a
        local file.<Direct>images\info.gif</Direct></Step>
     </Steps>
    </Perform>
    <Resolve>
      <Incomplete state="Ran" severity="Warning">The 'Example' test resolution process is
      incomplete.</Incomplete>
      <Complete state="Resolved" severity="Informational">The 'Example' test resolution
      process completed successfully.</Complete>
      <Steps>
        <Step step="1" state="Incomplete" type="text">Step 1 - Describe the resolution
        process with inline text.</Step>
        <Step step="2" state="Incomplete" type="link">Step 2 - Resolution instructions on a
        web page.<Direct>http://www.msnbc.com</Direct></Step>
      </Steps>
    </Resolve>
  </TestData>
</TestCase>
```

Each element of this code defines a part of the Manual Test:

Table 16-4 • XML Elements of a Manual Test

Element	Purpose
<TestCase>	Begins the Manual Test code and identifies the test name and category. Also, if selected="true" in this element, this test will be listed in the package tree on the **Test & Resolve** view.
<TestResults>	Lists the information that will be displayed in the **Description** field of the **Error Details** table if this test has not yet been run.
<TestData>	Marks the beginning and the end of the sections containing the instructions for performing and resolving the test.
<Details>	Lists the information that will be displayed in the **Details** field of the **Error Details** table.

Table 16-4 • XML Elements of a Manual Test

Element	Purpose
<KBArticle>	Contains a link to a help topic that provides more information on this test. This list appears in the **Help** field of the **Error Details** table.
<Perform>	Marks the beginning and the end of the instructions to perform this test.
<Incomplete state> <Pass state> <Fail state>	Identifies the states that will be displayed when you click on a step on the **Perform Manual Test** dialog box, the severity of each state, and the text that will be displayed when this state is selected.
<Steps>	Marks the beginning and the end of the steps that have to be completed to perform this test.
<Step>	Marks the beginning and the end of each step in this test.
<Resolve>	Marks the beginning and the end of the instructions to resolve this test.
<Incomplete state> <Complete state>	Identifies the states that will be displayed when you click on a step on the **Resolve Manual Test** dialog box, the severity of each state, and the text that will be displayed when this state is selected.
<Steps>	Marks the beginning and the end of the steps that have to be completed to resolve this test.
<Step>	Marks the beginning and the end of each step in this resolution.

PackageExpert Reference

PackageExpert Reference includes the same topics displayed when you click a help button from a view or dialog box in the PackageExpert interface. Reference information for PackageExpert is organized into the following area.

Table 16-5 • PackageExpert Reference

Section	Description
PackageExpert Interface	This section contains information about the various views of the PackageExpert interface and how they work together.
PackageExpert Views	Each view available in PackageExpert is covered in this section.
PackageExpert Dialog Boxes	Specific help for each dialog box in PackageExpert is covered in this section.

PackageExpert Interface

This section describes the overall PackageExpert user interface and includes the following topics:

- PackageExpert Home Page
- PackageExpert Package Tree
- PackageExpert Toolbar Menus
- PackageExpert Icons
- Status Bar

PackageExpert Home Page

If no packages are open, the initial view of the PackageExpert interface is the PackageExpert Home page. On the Home Page, the following quick links are provided:

- **Open a Package**—Open a Windows Installer package to test. See Testing External Packages.
- **Open a Catalog**—Open an Application Catalog containing packages to test. See Testing Packages in an Application Catalog.
- **Set Defaults**—Select the default tests to run each time a package is tested, and specify the resolution method you want to use. See Setting Default Test Configuration and Specifying the Resolution Method.

PackageExpert Package Tree

The PackageExpert package tree includes the following nodes:

Table 16-6 • PackageExpert Package Tree

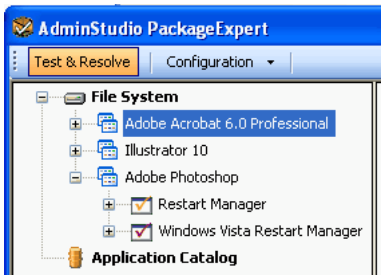
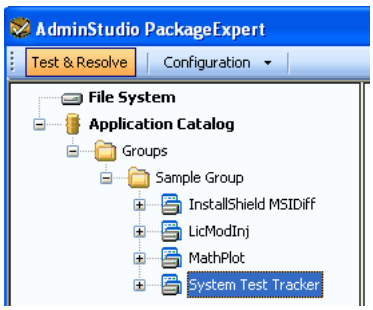
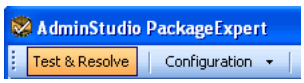
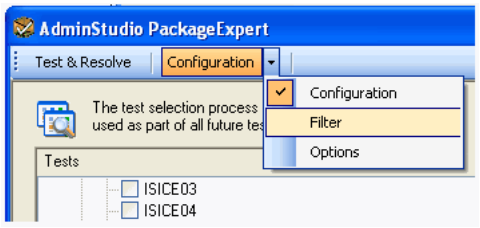
Item	Description
File System Node	Select this node and select **Open Package** from the context menu to open a standalone Windows Installer package to test. All of the packages that are currently open as listed under the File System node.  ***Note*** • *When you open a package under the **File System** node, it remains open through multiple PackageExpert sessions, until you explicitly remove it. To remove a package from the list, select it and select **Remove Package** from the context menu.*
Application Catalog Node	When you select this node, you are prompted to connect to an Application Catalog. After you have connected to an Application Catalog, the groups and packages it contains are listed in a tree format under the **Application Catalog** node: 
Test & Resolve Menu	Click this menu to return to the Test & Resolve page from a configuration page. 

Table 16-6 • PackageExpert Package Tree

Item	Description
Configuration / **Filter** / **Options Menu**	Click the arrow in the link to toggle between the following views:  • **Configuration**—Select to open the Configuration View, where you specify which tests you want to run for all subsequent test runs. • **Filter**—Select to open the Filter View, where you can edit or delete existing filters. • **Options**—Select to open the Options View where you can specify whether you want PackageExpert to resolve errors by creating a transform (`.mst`) file or by directly editing the `.msi` package.

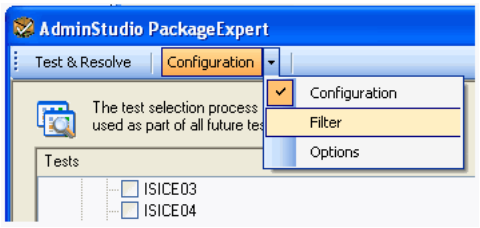
PackageExpert Toolbar Menus

The PackageExpert toolbar includes the following menu selections:

Table 16-7 • PackageExpert Package Tree

Item	Description
Test & Resolve Menu	Click this menu to return to the Test & Resolve view from another view.

Table 16-7 • PackageExpert Package Tree

Item	Description
Configuration / Filter / Options Menu	Click the arrow in the link to toggle between the following views:  • **Configuration**—Select to open the Configuration View, where you specify which tests you want to run for all subsequent test runs. • **Filter**—Select to open the Filter View, where you can edit or delete existing filters. • **Options**—Select to open the Options View where you can specify the resolution method.

PackageExpert Icons

There are two types of PackageExpert icons:

- Message Severity Icons
- Package Tree Icons

Message Severity Icons

When you select a Test Category node, all of the messages that were generated by tests in that category are listed. Each message includes an icon to indicate its severity. The severity icons indicate the following:

Table 16-8 • Severity Icons

Icon	Status	Description
	Error	Identifies a problem that would cause incorrect behavior when this package is installed or used.
	Warning	Identifies a problem that could cause incorrect behavior when this package is installed or used.
	Information	Indicates that either the test has not yet been run or the test ran and generated no errors.
	Resolved	Indicates that PackageExpert has repaired this error.

Package Tree Icons

The following icons are used in the package tree:

Table 16-9 • Package Tree Icons

Icon	Name	Description
	File System	Select this icon and select **Open Package** from the context menu to open and test a package that is stored on a local or network file system.
	Application Catalog	Select this icon to connect to an SQL Server Application Catalog database or to the AdminStudio Enterprise Server Application Catalog.
	Application Catalog Disabled	Appears in Status Bar when PackageExpert is not connected to an Application Catalog.
	Tests Category With Errors	Test results in this category include at least one message with a status of **Error**.
	Test Category With Warnings	Test results in this category includes at least one message with the status of **Warning** and no messages with the status of **Error**.
	Test Category With No Errors or Warnings	Either the tests in this category have not been run, or all messages in this category have a status of **Information** (there are no messages with the status of Error or Warning).
	Package	Represents a Windows Installer package.
	Package Displaying Catalog Test Results	This package is in an Application Catalog and PackageExpert is currently displaying the PackageExpert test results that have been published to that Application Catalog. To view a package's published test results: ***To display a package's published test results:*** 1. Select the package in the package tree. 2. Right-click, point to **Test Results** and click **Catalog** on the context menu.

Table 16-9 • Package Tree Icons

Icon	Name	Description
	Package Displaying Local Test Results	This package is in an Application Catalog and PackageExpert is currently displaying the PackageExpert test results stored in this package's local `.xml` file. ***To display a package's local test results:*** 1. Select the package in the package tree. 2. Right-click, point to **Test Results** and click **Local** on the context menu.
	Out of Sync Package	This package is in an Application Catalog and PackageExpert is currently displaying the PackageExpert test results that have been published to that Application Catalog, but the Catalog results are out of sync with the results in the local .xml file ***To publish local test results to the Application Catalog:*** 1. Select the package in the package tree. 2. Right-click, point to **Test Results** and click **Publish Results** on the context menu.
	Disabled Package	For packages in an Application Catalog, this icon indicates either that the Windows Installer `.msi` file associated with this package has been moved or deleted from its original location, or that this package is managed by the Software Repository.

Status Bar

A Status Bar appears at the bottom of all of the PackageExpert Views:

Figure 16-20: PackageExpert Status Bar

The Status Bar is comprised of the following items:

- PackageExpert Run Status
- Application Catalog Status
- Output Window Toggle

PackageExpert Run Status

PackageExpert has the following statuses:

Table 16-10 • PackageExpert Run Status

Status	Description
Ready	No tests or automated resolutions are currently being executed.
Executing automated tests on [PACKAGE_NAME]	PackageExpert is executing a test run.
Resolution started	PackageExpert is performing automated resolutions of the selected tests or test categories.

Application Catalog Status

An icon appears on the right side of the Status bar to indicate whether or not PackageExpert is currently connected to an Application Catalog:

Table 16-11 • Application Catalog Status Icons

Icon	Name	Description
	Connected to an Application Catalog	PackageExpert is connected to an Application Catalog.
	Not connected to an Application Catalog	Package Expert is not connected to an Application Catalog.

Output Window Toggle

As PackageExpert executes tests or performs resolutions, messages appear in the Output Window.

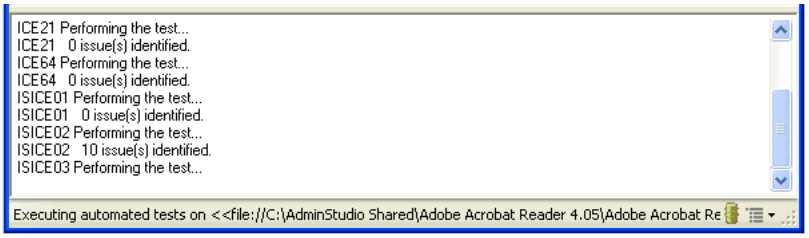
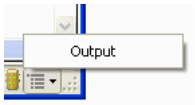


Figure 16-21: Test Execution Messages in the PackageExpert Output Window

You can toggle the display of the Output Window open or closed by clicking on the menu icon in the Status Bar and selecting **Output** from the list.



PackageExpert Views

The PackageExpert interface includes the following views:

Table 16-12 • PackageExpert Views

View	Description
Test and Resolve View	Perform testing and view test results.
Configuration View	Specify the default set of PackageExpert tests to run during subsequent test runs.
Filter View	Edit or delete existing test result filters.
Options View	Specify the resolution method that you want PackageExpert to use.

Test and Resolve View

The **Test and Resolve** view displays a package tree containing packages and groups of packages from a selected Application Catalog. The Right Pane view varies depending upon what is selected in the package tree:

- File System Node Selected
- Package Selected
- Group Selected
- Application Catalog Selected
- Test Category Selected
- Message Node Selected

File System Node Selected

When the **File System** node is selected, the **Select a Package** form is displayed in the right pane, and prompts you to select a Windows Installer package to test. See Select a Package Dialog Box for more information.

Package Selected

When a package is selected in the package tree, the following information is displayed.

Table 16-13 • Test & Resolve View / Package Selected

Item	Description
Package Name	The name of the selected package is displayed in the banner at the top of the right pane.
Manufacturer	The manufacturer of the this Windows Installer package.
File	Location of the Windows Installer package that was opened.
Test Results for All Categories	Listing of all of the messages that were generated during the last test run.
Output Window	Displays messages produced during testing and resolution.

Group Selected

When a group is selected, summary information about the Test Results of the packages in that group are displayed:

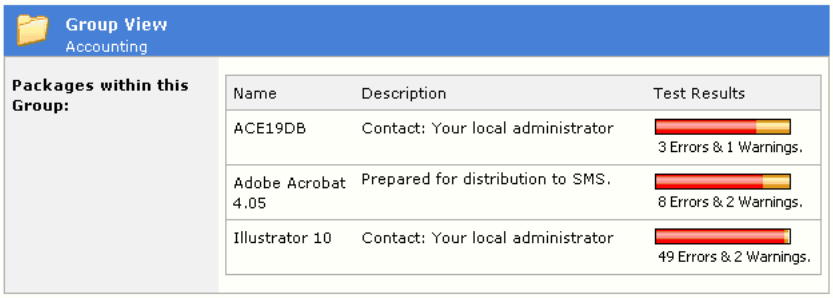


Figure 16-22: Group Summary Information

Application Catalog Selected

When the Application Catalog node is selected, and you are connected to an Application Catalog, summary information about the Test Results of the packages in the Application Catalog are displayed:

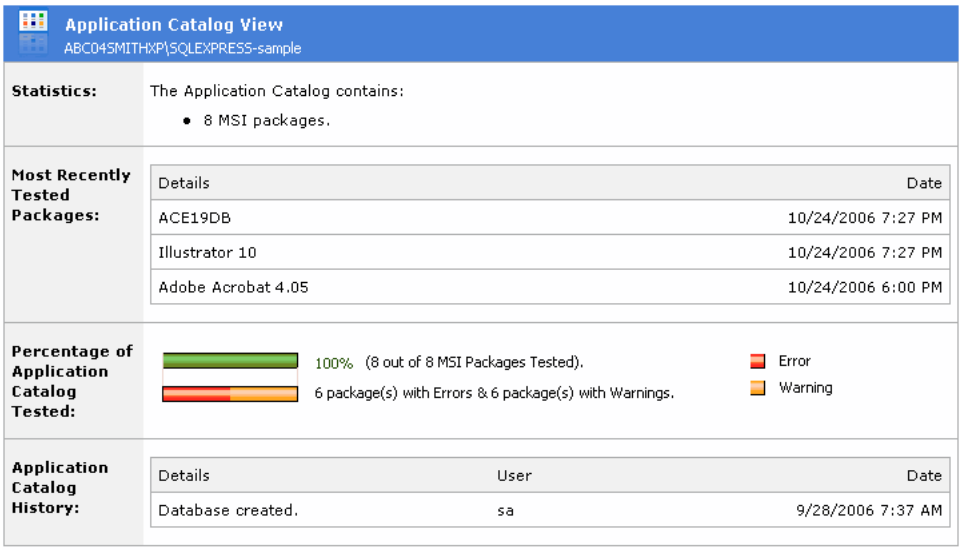


Figure 16-23: Application Catalog Summary Information

Test Category Selected

When a Test Category node is selected test results for all of the tests in that category are listed.

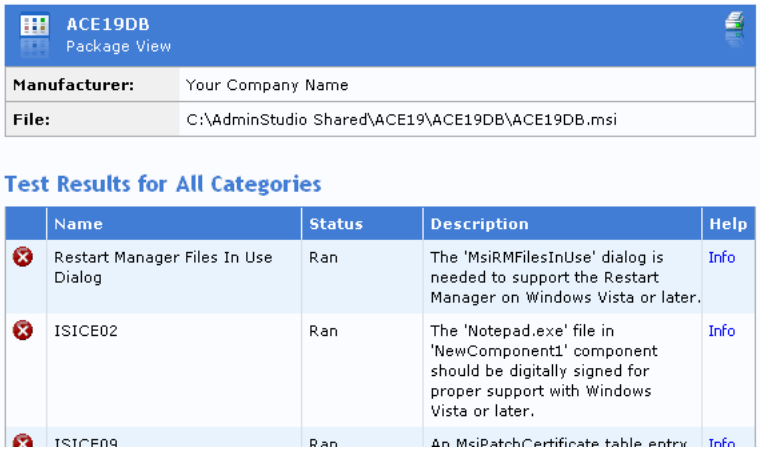


Figure 16-24: Test Category Summary Information

Message Node Selected

When a message node in the package tree is selected, detailed information on that specific message is displayed, including the exact text of the selected error message and a help topic that explains what the test is checking for and how errors are resolved.

Figure 16-25: Message Node Detailed Information

Configuration View

On the **Configuration** view, you specify which tests you want to PackageExpert to run for all subsequent test runs. These settings can be changed at any time and any changes you make do not effect the results of packages that have already been tested.

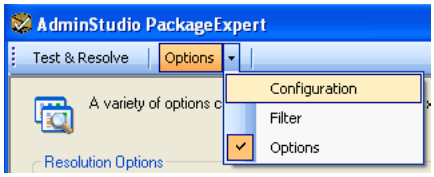
The Configuration view consists of two panes:

- **Tests tree**—Tests are grouped into categories and are displayed in a tree on the left pane. Select the tests that you want PackageExpert to run for subsequent test runs.

Tip • *You can select or unselect all of the tests in a test category by selecting or unselecting the test category check box.*

- **Help pane**—When you select a test, a help topic explaining that test opens in the right pane.

The **Configuration** view is opened by clicking **Configuration** in the PackageExpert toolbar menu. This toolbar menu toggles between **Configuration**, **Options**, and **Filter**. If **Configuration** is not displayed, click the arrow next to the menu item and select **Configuration** from the list.



Filter View

The **Filter** view lists all of the defined severity filters in the **Filter List**, all of the test result exclusion filters in the **Exclusion List**.

From the Filter view, you can **Edit** or **Delete** an existing filter.

Note • *Filters are created by selecting a test result in the **Test & Resolve** view, and selecting **Filter Result** from the context menu. The Test Result Filter Editor Dialog Box opens, where you can create a filter. See Customizing Test Results for more information.*

Editing a Result Filter

To edit an existing filter, select the filter and click the **Edit** button. The **Test Result Filter Editor** dialog box opens where you can edit the **Filter Severity** and/or **Description** fields and click **Update** to save your edits.

Deleting a Result Filter

To delete an existing filter, select the filter and click the **Delete** button.

Options View

On the **Options** view you can specify whether you want PackageExpert to resolve errors by creating a transform (`.mst`) file or by directly editing the `.msi` package.

You open the **Options** view by clicking the down arrow next to **Configuration** in the PackageExpert toolbar and selecting **Options**.

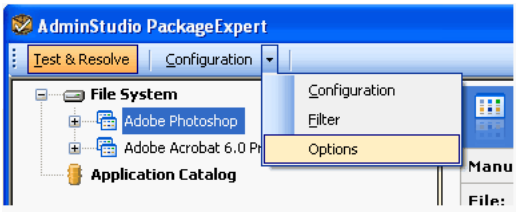


Figure 16-26: Opening the **Options** View

On the **Options** view, select one of the following options to specify the resolution method you want PackageExpert to use when automatically resolving errors:

- **Original MSI file**—Select this option to have PackageExpert directly edit the `.msi` file to resolve errors.
- **A newly created transform (MST) file**—Select this option to have PackageExpert create a transform (`.mst`) file to resolve errors.

PackageExpert Dialog Boxes

PackageExpert includes the following dialog boxes:

- Connect Application Catalog Dialog Box
- Custom Test Results Editor Dialog Box
- Manual Test Perform Dialog Box
- Manual Test Resolve Dialog Box
- Select a Package Dialog Box
- Test Result Filter Editor Dialog Box
- Unpublished Packages Dialog Box

Connect Application Catalog Dialog Box

On the **Connect Application Catalog** dialog box, which opens when you click on the Application Catalog node in the package tree, you specify the login information to connect to an Application Catalog database. This dialog box has two tabs:

- **Enterprise Server**—Select this tab to open the AdminStudio Enterprise Server Application Catalog database. See Enterprise Server Tab.
- **Standalone**—Select this tab to open an Application Catalog database other than the AdminStudio Enterprise Server Application Catalog database. See Standalone Tab.

Making this the Default Shared Application Catalog

If you select the **Make this the default shared Application Catalog** option, the Application Catalog you are opening will become the default Application Catalog (and be recorded as such in the **AdminStudio Shared** directory).

If the Application Catalog is made the default, all other AdminStudio users that use the same shared directory will automatically connect to the default Application Catalog when AdminStudio is launched. Therefore, you should only set this option if you want to affect all AdminStudio users who access that shared directory.

Note • *In the AdminStudio Enterprise Edition, only the AdminStudio Administrator or users with the Change Default Database permission will see the* ***Make this the default shared Application Catalog*** *option. This allows the AdminStudio Administrator to configure the default Application Catalog, and then subsequent installations of AdminStudio will automatically connect to the default Application Catalog if they use the same shared directory.*

Enterprise Server Tab

To connect to the AdminStudio Enterprise Server Application Catalog, you log in on the **Enterprise Server** tab of the **Connect Application Catalog** dialog box.

The **Enterprise Server** tab includes the following options:.

Table 16-14 • Connect Application Catalog Dialog Box / Enterprise Server Tab Options

Option	Description
URL	A link indicating the location of the AdminStudio Enterprise Server associated with this installation of AdminStudio. You can click on this link to open the **Select AdminStudio Enterprise Server URL** dialog box to identify the AdminStudio Enterprise Server URL that you would like to connect to.
Authentication	Select one of the following options: • Windows Authentication • AdminStudio Enterprise Server User
User Name and Password	If you chose **AdminStudio Enterprise Server User**, enter your AdminStudio Enterprise Server **User Name** and **Password** (provided by your System Administrator).

Standalone Tab

On the **Standalone** tab of the **Connect Application Catalog** dialog box, enter the information required to login to the specified Application Catalog.

The **Standalone** tab includes the following options:

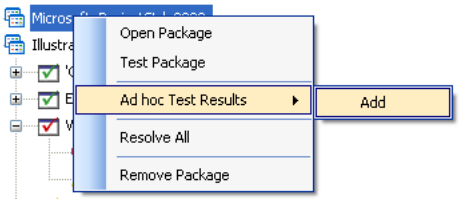
Table 16-15 • Connect Application Catalog Dialog Box / Standalone Tab Options

Option	Description
Server	Select one of the available SQL Servers on the network from this list. You can also manually enter the name of the SQL Server to which you want to connect.
Authentication	Select one of the following options: • **Windows Authentication**—Choose to use Windows network authentication (your network login ID) to log into this Application Catalog. • **Server Authentication**—Choose to use SQL Server login identification for authentication. • **Login ID and Password**—If you chose Server Authentication, enter the appropriate Login ID and Password.
Catalog	This is a list of catalogs available on the specified server. Select the name of the Application Catalog database that you want to connect to.
Test	Click this button to test whether a connection can be made to the database.
Make this the default shared Application Catalog	When this option is selected, the Application Catalog you are trying to open or create will become the default Application Catalog (and be recorded as such in the **AdminStudio Shared** directory). If the Application Catalog is made the default, all other AdminStudio users that use the same shared directory will automatically connect to the default Application Catalog when AdminStudio is launched. Therefore, you should only set this option if you want to affect all AdminStudio users who access that shared directory.

Custom Test Results Editor Dialog Box

The **Custom Test Results Editor** dialog box is used to manually add a test result message to a package. You might want to do this if you tested the package using QualityMonitor, InstallShield, or another tool, and you noticed something which is unexpected and which you want to document. Also, if a package failed during installation, you might also want to document the error.

The **Custom Test Results Editor** dialog box is opened by selecting a node in the package tree, pointing to **Ad hoc Test Results** and clicking **Add** on the context menu.



For example, suppose an error was encountered during installation, which caused installation failure, and you want to document this error.

When adding a test result on the **Custom Test Results Editor** dialog box, enter the following information:

Table 16-16 • Custom Test Results Editor Dialog Box Options

Option	Description
Category	Either enter a new name for a test category where you want to display this message, or select an existing category from the list. ***Note*** *• You can only associate an ad-hoc test result with a test category you created, not one that contains tests shipped with PackageExpert. Ad-hoc test results must always be stored in manually-created categories. If a category has already been added, then it will be listed in the* ***Category*** *list. You can create a new category by entering a category name in the* ***Category*** *box.*
Name	Either enter a new name for the test, or—if the same test has generated more than one error—select an existing name from the list
Severity	Select **Error**, **Warning**, or **Information** to characterize the severity of this error.
Description	Document the error that you discovered during ad-hoc testing. This information will be displayed under **Error Details** in the right pane when this test result node is selected.

The following dialog box displays an example of an added message:

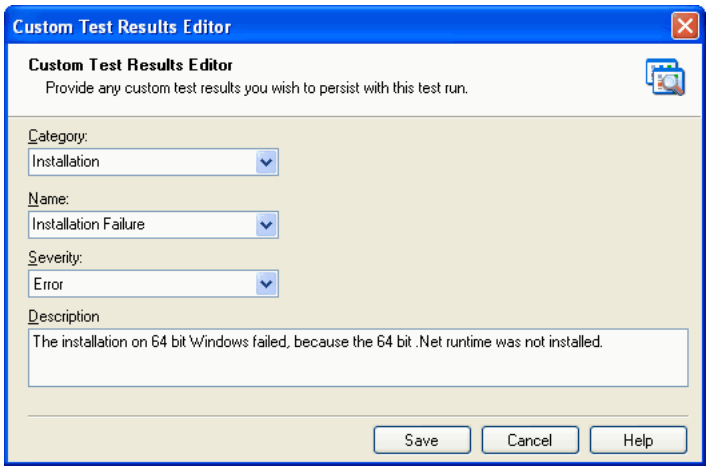


Figure 16-27: Example of a Message Added to Document Ad-Hoc Testing

When you click **Save** on the **Custom Test Results Editor** dialog box, the new message will be displayed in the package tree, and will be documented in the **Error Details** area in the right pane.

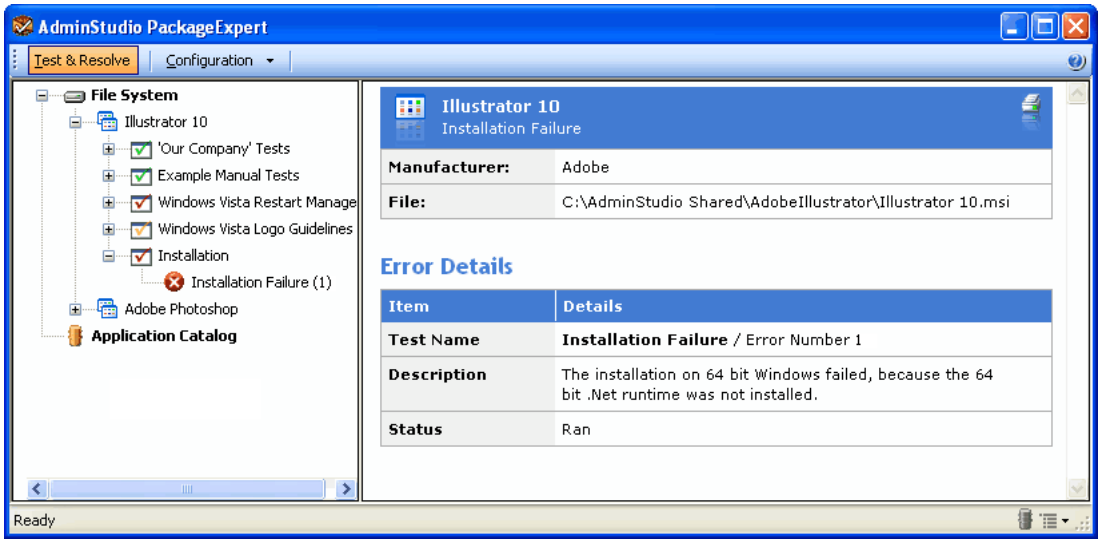


Figure 16-28: Display of a Message Added to Document Ad-Hoc Testing

If the future, when this problem has been corrected, you could delete or demote the severity of the test

Manual Test Perform Dialog Box

The **Manual Test Perform** dialog box opens when you choose to run a Manual Test that you have added to PackageExpert and contains instructions on how to perform the Manual Test.

The steps that are listed on the **Manual Test Perform** dialog box come from the XML file associated with this Manual Test. As you manually perform each listed step, you then right-click on the step and set its status to either **Pass** or **Fail**. See Adding Manual Tests for more information.

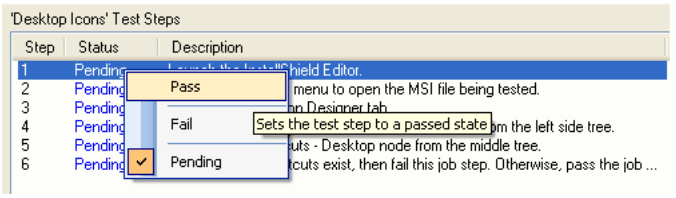


Figure 16-29: Changing the Status of a Step in a Manual Test

You open the **Manual Test Perform** dialog box by selecting a Manual Test in the package tree and selecting **Perform Manual Test** from the context menu.

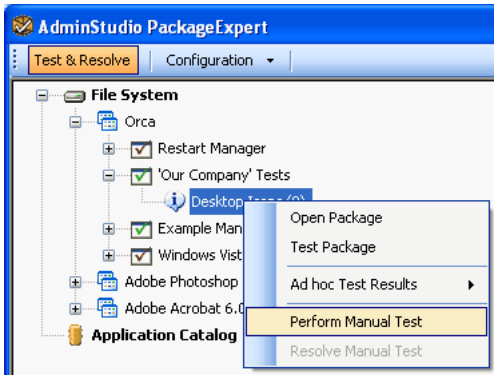
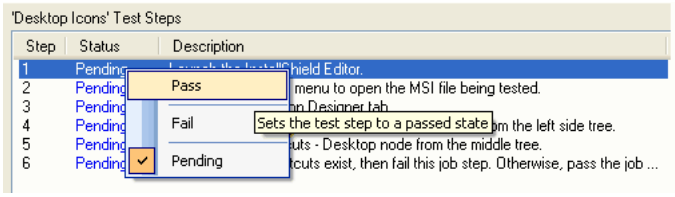


Figure 16-30: Opening the Manual Test Perform Dialog Box

Note • *In order to perform a Manual test, the Test Results mode of the package must be set to* ***Local****.*

The Manual Test Perform dialog box includes the following options:

Table 16-17 • Manual Test Perform Dialog Box

Option	Description
Step	Numbers the steps in this procedure.
Status	States whether the step has not yet been performed (**Pending**), whether it was run without errors (**Pass**), or whether was run and produced errors (**Fail**). Right-click on the listed status to change it: 
Description	Contains the instructions for performing this Manual Test.
Save	Click to save the status changes that you have made.

Manual Test Resolve Dialog Box

For Manual Tests that are added to PackageExpert, you use the **Manual Test Resolve** dialog box to document the completion of steps taken to manually resolve a test result. The **Manual Test Resolve** dialog box lists the manual steps you need to perform to resolve this error.

You open the **Manual Test Resolve** dialog box by selecting a Manual Test in the package tree, and selecting **Resolve Manual Test** from the context menu.

The steps that are listed on the **Manual Test Resolve** dialog box come from the XML file associated with this Manual Test.

The first time you open this dialog box, all of the steps are marked **Incomplete**. Perform each of the listed steps, marking each as **Complete** after you have performed them, and then click **Save**.

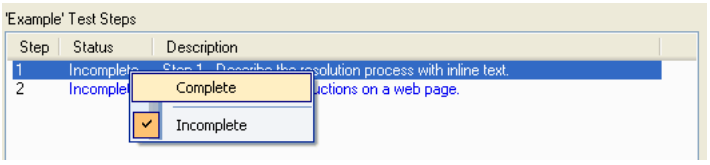
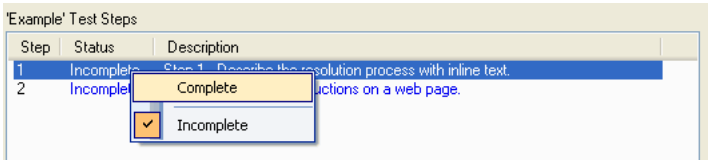


Figure 16-31: Marking Steps Complete When Resolving a Manual Test

You open the **Manual Test Resolve** dialog box by selecting a Manual Test in the package tree and selecting **Resolve Manual Test** from the context menu.

The Manual Test Resolve dialog box includes the following options:

Table 16-18 • Manual Test Resolve Dialog Box

Option	Description
Step	Numbers the steps in this resolution procedure.
Status	States whether the step has been **Completed** or is still **Incomplete**. Right-click on the listed status to change it: 
Description	Contains the instructions for performing this resolution procedure.
Save	Click to save the status changes that you have made.

Select a Package Dialog Box

If you select the **File System** node in the package tree, and select **Open Package** on the context menu, the Select a Package dialog box opens, prompting you to browse to a Windows Installer package on your file system.

The Select a Package dialog box includes the following options:

Table 16-19 • Select a Package Dialog Box Options

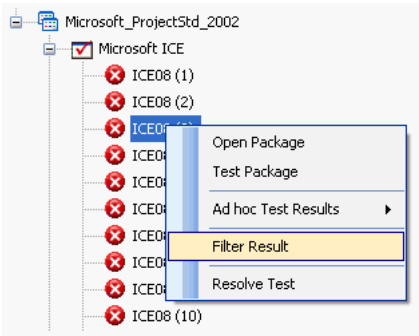
Option	Description
MSI Package	Click **Browse** and select a Windows Installer package on your local file system that you want to test.
Transform List	After you have selected an **MSI Package**, if any transform files exist in the same directory as the .msi file, they are automatically listed in the Transform List, but are not selected. • **To include listed transform files**—Select the transform files that you want to include in PackageExpert testing. • **To add additional transform files**—Click **Browse** and select a transform file to add it to the Transform list, and then select it to include it in testing.
Test	Click to open the selected package and perform testing on the package.
Cancel	Click to open the selected package without performing testing.

Test Result Filter Editor Dialog Box

You can use the **Test Result Filter Editor** dialog box to customize PackageExpert test results to accommodate unique or special circumstances at your organization. You can select a test result and change its severity or exclude it from the listings. You make these changes by making a selection from the **Filter Severity** list:

- **Change severity**—To change the severity of a specific test result from its original severity, select **Error**, **Warning**, or **Information**.
- **Exclude**—To suppress the display of a specific test result, select **Exclude**.

You open the Test Result Filter Editor dialog box by opening the **Test & Resolve** view, selecting the specific message that you want to modify and selecting **Filter Result** from the context menu.



Note • *You are filtering a specific instance of a message generated by a test, not all messages generated by that test. For example, suppose the ICE08 test produced five error messages. You consider messages 1 through 4 to be valid errors. However, message 5 concerns a particular component of the package that no one at your organization is likely to use, and therefore you do not want this particular test result to be displayed You could then change the Severity to* ***Exclude*** *to suppress the display of this test result.*

The **Test Result Filter Editor** dialog box includes the following options:

Table 16-20 • Test Result Filter Editor Dialog Box

Option	Description
Test Result Area	Lists the **Name** and **Category** of the test that produced this message, and lists the message text of the test **Result** you selected. For example, this is a message produced by the **Restart Manager Files in Use Dialog** test : `The 'MsiRMFilesInUse' dialog is needed to support the Restart Manager on Windows Vista or later.`

Table 16-20 • Test Result Filter Editor Dialog Box

Option	Description
Filter Severity	Select one of the following options to filter this message: • **Error**—Select to identify this test result as a problem that *would* cause incorrect behavior when this package is installed or used. • **Warning**—Select to identify this test result as a problem that *could* cause incorrect behavior when this package is installed or used. • **Information**—Select to indicate that this test result is not an error. • **Exclude**—Select to exclude the display of this test result. After you have set this filter, if the same problem is encountered in another package or in a subsequent test run of this package, the filter that you defined will continue to be used.
Description	Enter an explanation for why you changed the severity level or excluded this message
Save	Click to save the filter.

Unpublished Packages Dialog Box

This dialog box opens if you attempt to exit PackageExpert before you have published test results to the Application Catalog for those packages whose **Test Results** mode is set to **Catalog**.

Select those packages in the **Packages** list for which you want to save test results to the Application Catalog, and click **OK**.

17

Analyzing the Impact of Installing Microsoft Operating System Patches

Edition • *OS Security Patch Wizard is included with AdminStudio Enterprise Edition.*

OS Security Patch Wizard is a new AdminStudio tool that is used to import Microsoft operating system patch information into the Application Catalog.

You need to import Microsoft OS patch information into the Application Catalog so that you can analyze the full impact of installing these patches on user machines. Based on the analysis results, you can determine the level of testing you need to perform before distributing a Microsoft OS patch throughout your enterprise.

Information about the OS Security Patch Wizard and patch impact analysis is presented in the following sections:

Table 17-1 • OS Security Patch Wizard/Patch Impact Analysis Wizard Help Library

Section	Description
About Microsoft Operating System Patches	Explains what Microsoft operating system patches are, and why you should include them in your package testing processes.
Using OS Security Patch Wizard to Import Microsoft OS Patches	Explains how to download and import a Microsoft operating system patch into the Application Catalog. • Identifying and Downloading Microsoft Operating System Patch Files • Importing a Microsoft Operating System Security Patch Into the Application Catalog
Analyzing the Impact of Installing a Microsoft Operating System Patch	Explains how to use the ConflictSolver Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system patches and the packages and OS Snapshots in the Application Catalog.
Reference	Describes all of the panels in the OS Security Patch Wizard.

About Microsoft Operating System Patches

Each month, Microsoft releases patches to address security vulnerabilities that are discovered in Microsoft operating system software. Microsoft defines security vulnerabilities as:

"A flaw in a product that makes it infeasible – even when using the product properly—to prevent an attacker from usurping privileges on the user's system, regulating its operation, compromising data on it, or assuming ungranted trust."

Microsoft publishes a monthly Microsoft Security Bulletin Summary that lists the patches released that month. You can view these bulletins on the Microsoft TechNet Security Center Web site. The Microsoft Security Bulletin Summary for September 2007 is shown in the following figure:

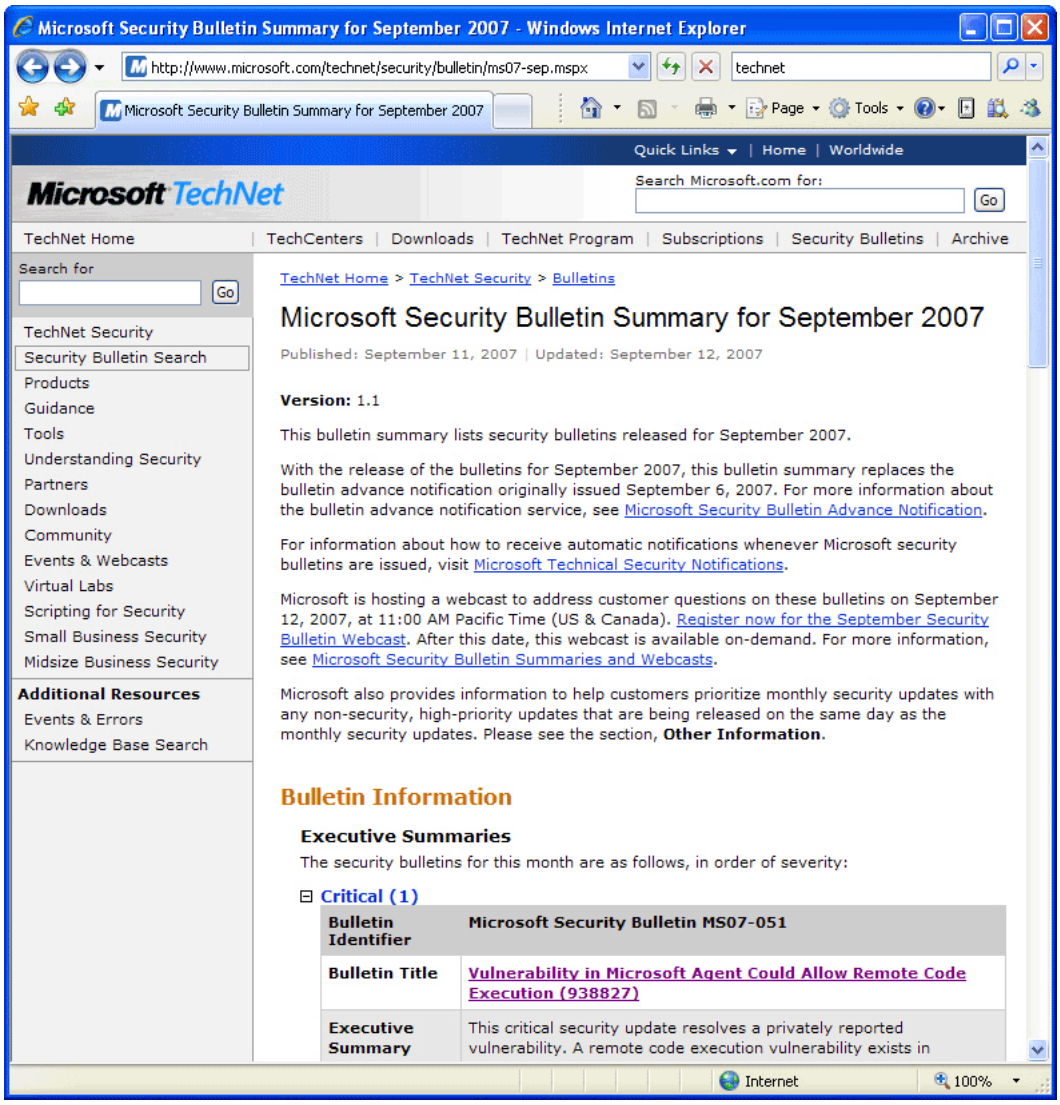


Figure 17-1: Microsoft Security Bulletin Summary for September 2007

The Security Bulletin Summary lists each patch released that month, grouped by status level, with a link to each patch's associated Security Bulletin, as shown in the following figure:

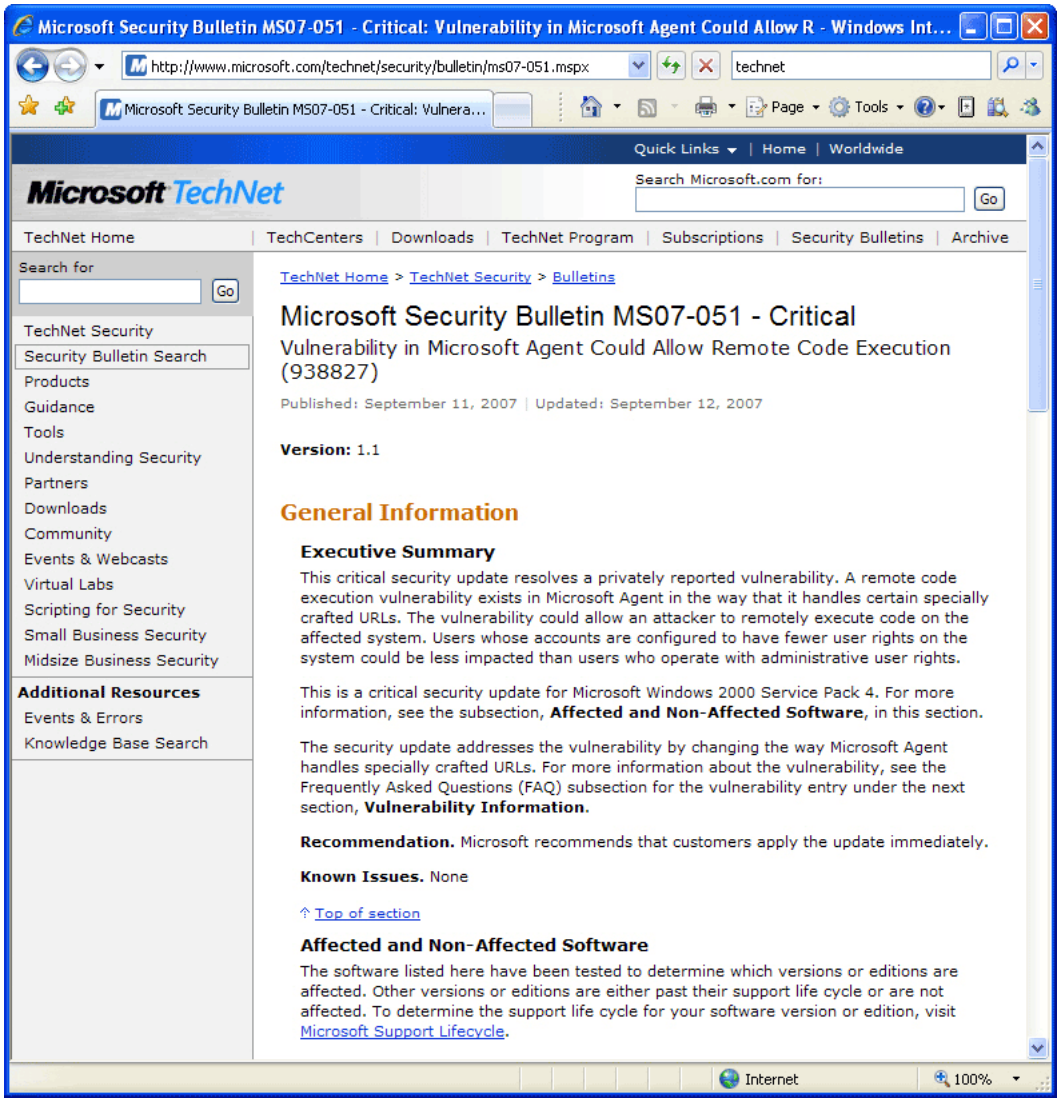


Figure 17-2: Microsoft Security Bulletin MS07-051

Security Bulletin Summaries, Security Bulletins, and patches can be accessed from the Microsoft TechNet Security Center:

`http://www.microsoft.com/technet/security/default.mspx`

Using OS Security Patch Wizard to Import Microsoft OS Patches

You can use the OS Security Patch Wizard to import Microsoft operating system patch information into the AdminStudio Application Catalog. When you select the patch to import, you are prompted to review the imported data and to add information from the patch's Security Bulletin. Information about importing Microsoft operating system patches is presented in the following sections:

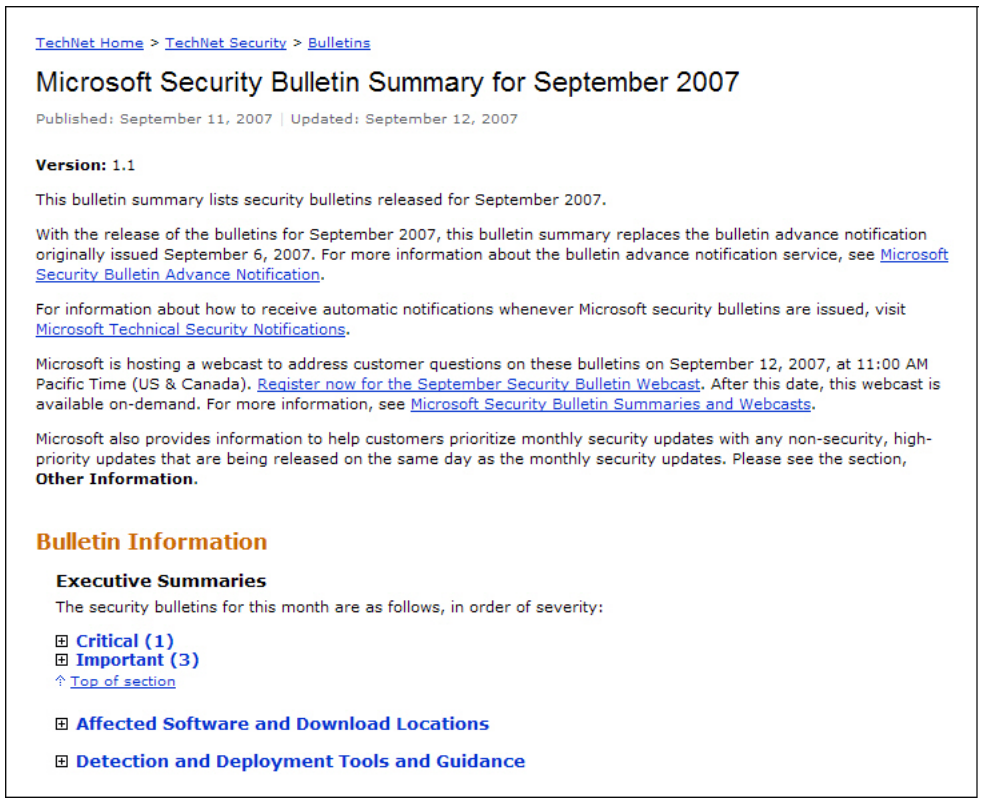
- Identifying and Downloading Microsoft Operating System Patch Files
- Importing a Microsoft Operating System Security Patch Into the Application Catalog

Identifying and Downloading Microsoft Operating System Patch Files

To identify and obtain the Microsoft operating system patch files that you want to import into the Application Catalog, perform the following steps.

Task: ***To identify and download Microsoft OS patch files:***

1. Open the Microsoft Security Bulletin Summary that lists the patch that you want to import. The following figure is of the Microsoft Security Bulletin Summary for September 2007 :



TechNet Home > TechNet Security > Bulletins

Microsoft Security Bulletin Summary for September 2007

Published: September 11, 2007 | Updated: September 12, 2007

Version: 1.1

This bulletin summary lists security bulletins released for September 2007.

With the release of the bulletins for September 2007, this bulletin summary replaces the bulletin advance notification originally issued September 6, 2007. For more information about the bulletin advance notification service, see Microsoft Security Bulletin Advance Notification.

For information about how to receive automatic notifications whenever Microsoft security bulletins are issued, visit Microsoft Technical Security Notifications.

Microsoft is hosting a webcast to address customer questions on these bulletins on September 12, 2007, at 11:00 AM Pacific Time (US & Canada). Register now for the September Security Bulletin Webcast. After this date, this webcast is available on-demand. For more information, see Microsoft Security Bulletin Summaries and Webcasts.

Microsoft also provides information to help customers prioritize monthly security updates with any non-security, high-priority updates that are being released on the same day as the monthly security updates. Please see the section, **Other Information**.

Bulletin Information

Executive Summaries

The security bulletins for this month are as follows, in order of severity:

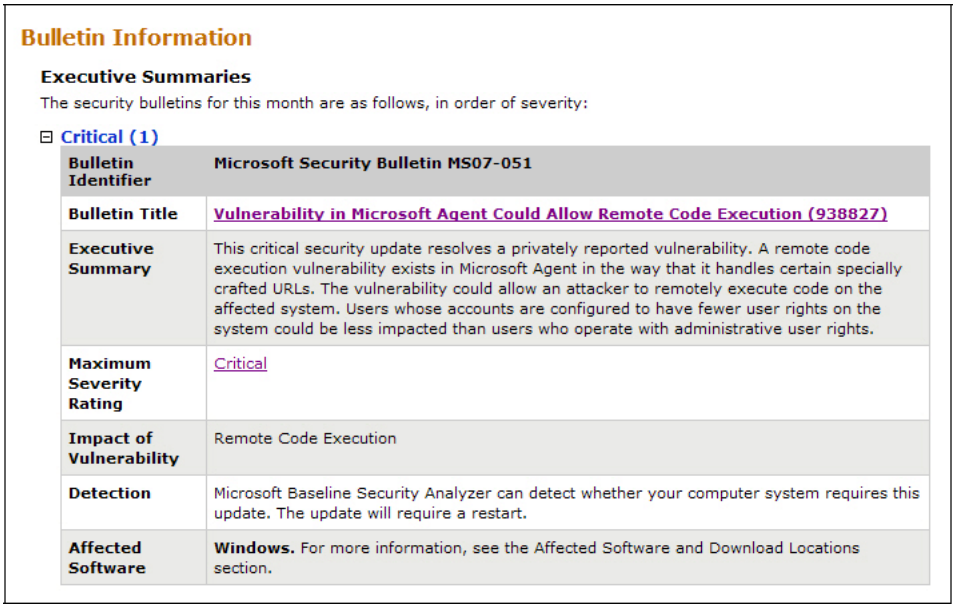
⊞ **Critical (1)**
⊞ **Important (3)**
Top of section

⊞ **Affected Software and Download Locations**

⊞ **Detection and Deployment Tools and Guidance**

In this example, three patches are listed under **Executive Summaries** in the **Bulletin Information** section: one with a status of **Critical** and two with a status of **Important**.

2. Expand the listing of the patch status that you want to review. In this example, if you expanded the **Critical** status, the following **Executive Summary** of Microsoft Security Bulletin MS07-051 is displayed:



Bulletin Information

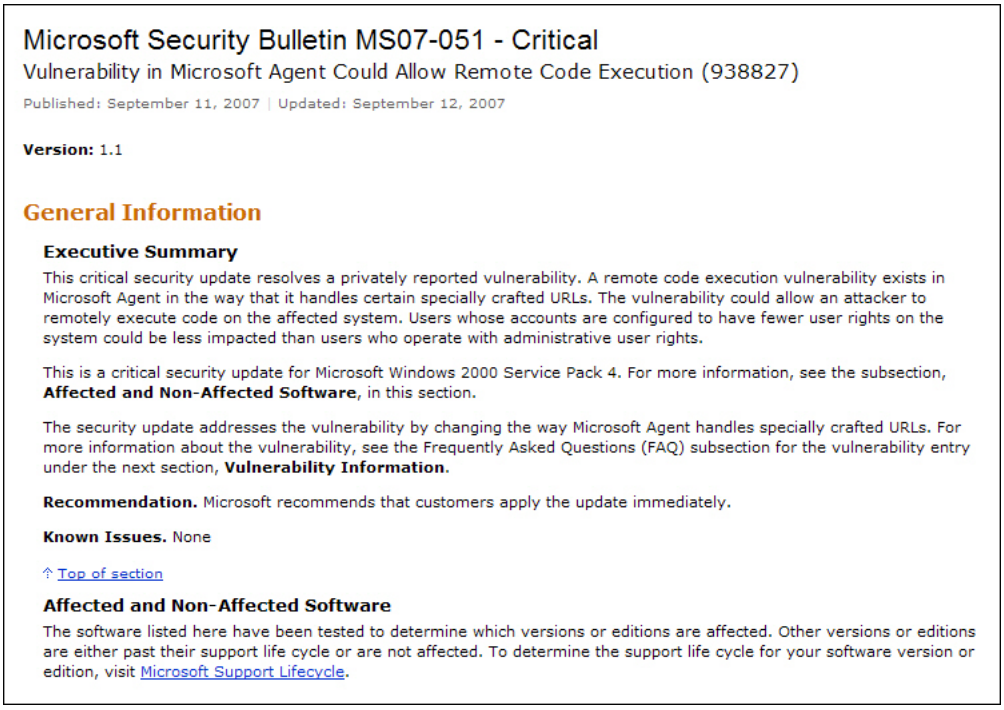
Executive Summaries

The security bulletins for this month are as follows, in order of severity:

⊟ Critical (1)

Bulletin Identifier	Microsoft Security Bulletin MS07-051
Bulletin Title	Vulnerability in Microsoft Agent Could Allow Remote Code Execution (938827)
Executive Summary	This critical security update resolves a privately reported vulnerability. A remote code execution vulnerability exists in Microsoft Agent in the way that it handles certain specially crafted URLs. The vulnerability could allow an attacker to remotely execute code on the affected system. Users whose accounts are configured to have fewer user rights on the system could be less impacted than users who operate with administrative user rights.
Maximum Severity Rating	Critical
Impact of Vulnerability	Remote Code Execution
Detection	Microsoft Baseline Security Analyzer can detect whether your computer system requires this update. The update will require a restart.
Affected Software	**Windows.** For more information, see the Affected Software and Download Locations section.

3. Click the link in the **Bulletin Title** field. The Security Bulletin for that patch opens. In this example, Security Bulletin MS07-051 opens:



Microsoft Security Bulletin MS07-051 - Critical

Vulnerability in Microsoft Agent Could Allow Remote Code Execution (938827)

Published: September 11, 2007 | Updated: September 12, 2007

Version: 1.1

General Information

Executive Summary

This critical security update resolves a privately reported vulnerability. A remote code execution vulnerability exists in Microsoft Agent in the way that it handles certain specially crafted URLs. The vulnerability could allow an attacker to remotely execute code on the affected system. Users whose accounts are configured to have fewer user rights on the system could be less impacted than users who operate with administrative user rights.

This is a critical security update for Microsoft Windows 2000 Service Pack 4. For more information, see the subsection, **Affected and Non-Affected Software**, in this section.

The security update addresses the vulnerability by changing the way Microsoft Agent handles specially crafted URLs. For more information about the vulnerability, see the Frequently Asked Questions (FAQ) subsection for the vulnerability entry under the next section, **Vulnerability Information**.

Recommendation. Microsoft recommends that customers apply the update immediately.

Known Issues. None

⇧ Top of section

Affected and Non-Affected Software

The software listed here have been tested to determine which versions or editions are affected. Other versions or editions are either past their support life cycle or are not affected. To determine the support life cycle for your software version or edition, visit Microsoft Support Lifecycle.

In the Security Bulletin, the **Affected Software** table lists the software affected by this patch, and provides a link to the download page for that specific patch:

Affected Software

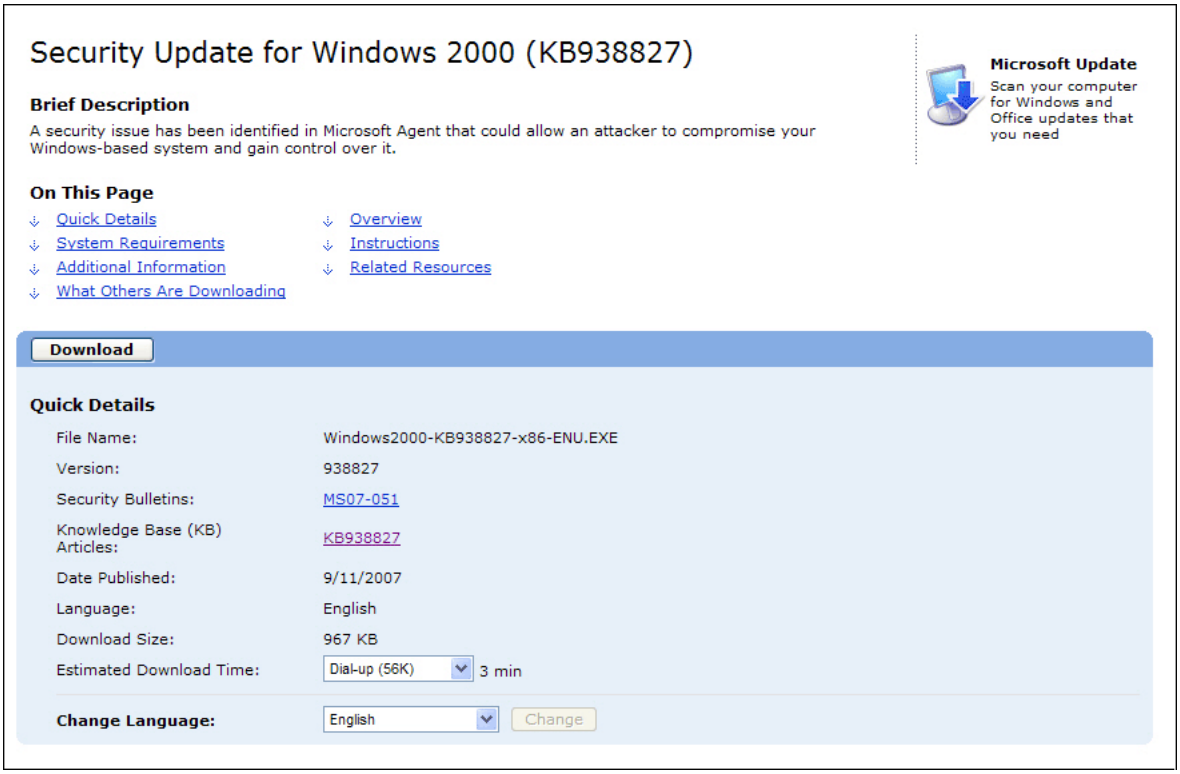
Operating System	Maximum Security Impact	Aggregate Severity Rating	Bulletins Replaced by This Update
Microsoft Windows 2000 Service Pack 4	Remote Code Execution	Critical	MS07-020

For some patches, both an **Operating System** and **Component** are listed, while for others, only an **Operating System** is listed.

Affected Software

Operating System	Component	Maximum Security Impact	Aggregate Severity Rating	Bulletins Replaced by This Update
Windows 2000 Service Pack 4	Windows Services for UNIX 3.0	Elevation of Privilege	Important	None
Windows 2000 Service Pack 4	Windows Services for UNIX 3.5	Elevation of Privilege	Important	None
Windows XP Service Pack 2	Windows Services for UNIX 3.0	Elevation of Privilege	Important	None
Windows XP Service Pack 2	Windows Services for UNIX 3.5	Elevation of Privilege	Important	None
Windows Server 2003 Service Pack 1 and Windows Server 2003 Service Pack 2	Windows Services for UNIX 3.0	Elevation of Privilege	Important	None
Windows Server 2003 Service Pack 1 and Windows Server 2003 Service Pack 2	Windows Services for UNIX 3.5	Elevation of Privilege	Important	None
Windows Server 2003 Service Pack 1 and Windows Server 2003 Service Pack 2	Subsystem for UNIX-based Applications	Elevation of Privilege	Important	None
Windows Server 2003 x64 Edition and Windows Server 2003 x64 Edition Service Pack 2	Subsystem for UNIX-based Applications	Elevation of Privilege	Important	None
Window Vista	Subsystem for UNIX-based Applications	Elevation of Privilege	Important	None
Windows Vista x64 Edition	Subsystem for UNIX-based Applications	Elevation of Privilege	Important	None

4. In the **Affected Software** table, click the link of the patch you want to import. The download page for that patch opens.



5. Click the **Download** button to download the patch. After the download is complete, the **Thank You for Downloading** page opens.

Importing a Microsoft Operating System Security Patch Into the Application Catalog

To import a Microsoft operating system security patch file into the Application Catalog, perform the following steps:

Task: ***To use OS Security Patch Wizard to import a Microsoft OS patch file into the Application Catalog:***

1. Open **OS Security Patch Wizard**. The **Welcome** panel opens.
2. Click **Next**. The **Patch File Selection** panel opens.
3. Click **Browse** and select a Microsoft patch file that you have downloaded from the Microsoft TechNet Security Center, as described in Identifying and Downloading Microsoft Operating System Patch Files.

4. Click **Next**. OS Security Patch Wizard reads and verifies the patch file and displays progress messages in the **Processing Status** area of this panel.

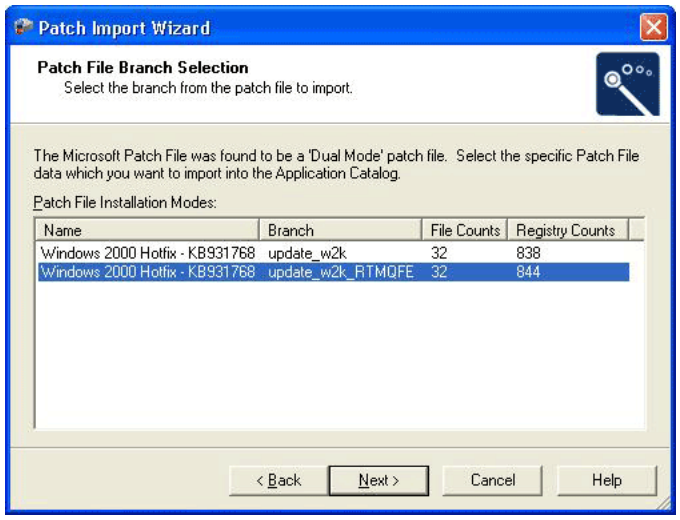
   - **Prompt to connect to the Application Catalog**—If you opened OS Security Patch Wizard from the **Start** menu, after you click **Next**, you will be prompted to provide database connection information. If you opened OS Security Patch Wizard from Application Manager, you will already be connected to the Application Catalog and will not be prompted to connect again.
   - **Invalid patch file message**—If you are attempting to import a patch file that is invalid, a message will be displayed and you will be unable to proceed with the import.

   When processing of the patch file is successfully finished, one of the following panels opens:

   - If you have downloaded a dual-mode patch, the **Patch File Branch Selection** panel opens. Continue with Step 5 below
   - If the patch you are importing is not a dual-mode patch, the **Patch Bulletin Information** panel opens. Skip to Step 7 below.

5. Windows XP software updates that are released after Service Pack 1 (SP1) may include binary files for both Windows XP-based and Windows XP-based SP1 computers. Therefore, there are multiple processing paths for installing this patch. These are called *dual-mode* patches. If you are importing a dual-mode patch, the **Patch File Branch Selection** panel opens.

   On this panel, you need to identify the branch of the patch file that you want to import into the Application Catalog. AdminStudio has preselected the branch of this patch that appears to install the latest version. You may choose to accept the default selection or make your own selection.



Note • *It is permissible to import the same dual-mode patch multiple times, selecting different branches each time. Subsequent panels will allow you to uniquely identify each patch branch. For more information on dual-mode patches and identifying their branches, see About Dual-Mode OS Patches.*

6. Click **Next**. The **Patch Bulletin Information** panel opens.

7. From the patch's associated Security Bulletin, copy and paste the **Bulletin ID** and the **Bulletin Title** associated with the patch file you are importing. You should also enter the **Bulletin Summary** and **Bulletin Release Date**.

   Important • *Security Bulletin data is used by Application Manager to integrate with the MIcrosoft Web site. Therefore, it is important that you accurately enter this data.*

   Note • *For information on how to view Microsoft Security Bulletins, see Identifying and Downloading Microsoft Operating System Patch Files.*

8. Click **Next**. The **Additional Patch Information** panel opens, displaying some data that was extracted from the patch file.

9. Review and edit the following information:

   - **Name**—By default, this field displays the name of the selected patch file, such as:

     `Windows2000-KB38827-x86-ENU.EXE`

     You should edit this name to make it easier for you to identify this patch when it is listed in the Application Catalog.

   - **Associated KB Article**—The KB article number is extracted from the patch file and automatically entered in this field.

     Important • *The Knowledge Base article identification information is used by Application Manager to integrate with the MIcrosoft Web site. Therefore, it is important that you accurately enter this data.*

   - **Effected Product**—Optionally, enter the name of the product that this patch effects. This information is typically something like Windows XP, Windows Vista, etc.

   - **Additional Comments**—Comments are extracted from the patch file and are automatically entered in this field. You can edit these comments.

10. Click **Next**. The **Patch Import Summary** panel opens, listing the selections you have made.

11. Click **Import** to initiate the patch import. When import is complete, the **Import Process Results** panel opens, listing the results of the import of the selected patch file. The results displayed on this panel, along with the summary information shown in the previous panel, are output to a log file.

    Note • *When you click **Import**, if you have not previously connected to the Application Catalog, you will be prompted to provide database connection information.*

12. To import another patch, click **< Another** to return to the beginning of the OS Security Patch Wizard.

13. When you have finished importing patches, click **Finish**.

Information About Missing Patch Files

The OS Security Patch Wizard imports a Microsoft operating system patch using the patch's INF file. An INF file is a text file that is used by Microsoft to install the patch and which contains the information necessary to install a patch, such as for a driver, including file names and locations, registry information, and version information.

An OS Security Patch can consolidate patches for different Windows Operating Systems. Any given INF file may contain information about files that do not exist or are not relevant for a particular Windows Operating System. The OS Security Patch Wizard ignores these files when importing the patch information into the Application Catalog and reports the missing files in the log file. The OS Security Patch Wizard will not import any information about the missing files into the Application Catalog.

About Dual-Mode OS Patches

The OS Security Patch Wizard imports a Microsoft operating system patch using the patch's INF file, which is a text file that contains all the information necessary to install a device (such as INI and registry changes, filenames, and locations of the source files on source media).

The **Branch** listed in this column is the name of the patch's INF file minus its extension. Because all obvious identifiable information within the INF file (like **Name**) is often identical, this Branch information is included to help differentiate the patches within a dual mode patch.

A sample branch name could be: `update_SP2GDR` or `update_SP2QFE`, where:

- **QFE** = Quality Fix Engineering updates (also known as hot fixes). These include security updates, feature packs, update roll-ups, drivers, and critical updates. They are created when an issue is identified that needs a fix in a specific and small subset of customer environments. These fixes have not been tested for the same length of time for regressions or incompatibilities as what a GDR fix typically goes through.
- **GDR** = General Distribution Release updates. These updates can be found on the Windows Update site, meaning that they are available to a wider audience. GDRs are tested to a greater degree than QFEs.

Note • *QFEs generally contain all the prior fixes from both GDR and QFE releases while GDR releases do not contain QFE fixes.*

Analyzing the Impact of Installing a Microsoft Operating System Patch

After you have imported a Microsoft operating system patch into the AdminStudio Application Catalog, you can continue to use the ConflictSolver Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system patches and the packages and OS Snapshots in the Application Catalog. This helps you determine how specific MSI packages or OS Snapshots would be affected when a Microsoft operating system patch is installed.

Important • *The patch information that you import using OS Security Patch Wizard is identical to and compatible with the patch information that you imported in previous releases using the Patch Impact Manager tool. Therefore, any procedures you previously used to analyze patch information will continue to work as they did in previous releases.*

The section is organized in the following topics:

- Performing Patch Impact Analysis
- Viewing Patch Impact Analysis Results
- Viewing Patch and Patch Impact Information in Application Manager and ConflictSolver
- Generating the Patch Report

Performing Patch Impact Analysis

You can use the Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system patches and the packages and OS Snapshots in your Application Catalog. This helps you determine how specific MSI packages or OS Snapshots would be affected when a Microsoft OS patch is installed.

Task: ***To perform patch impact analysis:***

1. On the ConflictSolver **Patches** tab, select **Patch Impact Analysis Wizard** from the **Conflicts** menu or select a patch and select **Patch Impact Analysis Wizard** from the context menu.
   - If you had a patch selected when you launched the Wizard, the **Source Patches Panel** opens, with the imported patch(es) that were selected when you launched the wizard automatically selected.
   - If you launched the Wizard by selecting **Patch Impact Analysis Wizard** from the ConflictSolver **Conflicts** menu, the **Welcome Panel** will open.
2. Click **Next**. The **OS Snapshot Panel** opens.
3. Optionally, select an OS Snapshot to include in your analysis. The selected OS Snapshot will be used to identify specific file information for any patch impacts that are discovered. Only one OS Snapshot can be selected.

   Tip • *The OS Snapshot serves as a representation of the underlying baseline system in your enterprise. If you include an OS Snapshot in your Patch Impact Analysis, then the file version information displayed in warning messages is derived from that OS Snapshot. If an OS Snapshot is not included in the analysis, the file version information will be displayed as "unknown". As such, it is recommended to include an OS Snapshot in your analysis because this allows a finer-tuned evaluation of impacts based upon file and version information.*
4. Click **Next**. The **Source Patches Panel** opens.

5. On the **Source Patches Panel**, select the patches that you want to include in your analysis.

   When searching for patches to include, you can use the **Filter by product** list to restrict the patches displayed on this panel. Also, to view more information on a patch, select the patch and click the **Patch Properties** button to open the **Patch Properties** dialog box.

6. After you have selected the patches that you want to include in your analysis, click **Next**. The **Target Products Panel** opens.

7. On the **Target Products Panel**, select the products or groups of products that you want to perform patch impact analysis on. The Patch Impact Analysis Wizard will analyze the products you select here against the patches you selected on the **Source Patches Panel** for impacts. You can use the **Select All** or **Clear All** buttons to make your selections.

8. Click **Next**. The **Summary Information Panel** opens, listing a summary of the options you selected in the Patch Impact Analysis Wizard.

9. Click **Finish** to accept these options and begin the Patch Impact Analysis. Analysis messages are listed in the **Output** tab of the ConflictSolver Output Window.

   When analysis is complete, patch conflicts are listed on the **Patch Impact** tab of the Output Window in table format.

Viewing Patch Impact Analysis Results

After Performing Patch Impact Analysis, patch conflicts are listed on the **Patch Impact** tab of the Output Window in table format.

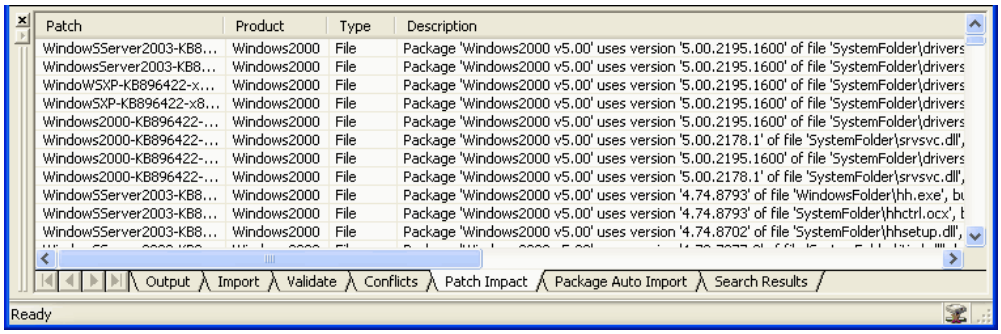


Figure 17-3: Sample Patch Impact Analysis Results

The following information is included in Patch Impact Analysis results:

Table 17-2 • Information Included in Patch Impact Analysis Results

Item	Description
Patch	Name of a Windows Installer Patch.
Product	Name of a Package or an OS Snapshot in the Application Catalog.

Table 17-2 • Information Included in Patch Impact Analysis Results

Item	Description
Type	Identifies the type of the impact as either a **File** or a **Registry** impact. ***Note*** • *Windows Installer Patches rarely impact Registry Entries, so most of the identified impacts will be identified as* ***File****.*
Description	Description of how the Windows Installer Patch impacted with the package or OS Snapshot. For example: `File 'CdrGfx.dll' in Package 'Coreldraw 12.0.0.458 v1.0' uses version 'Unknown' of file 'SystemFolder\RPCRT4.dll', but Patch 'WindowsXP-KB828741-x86-ENu.EXE' uses version '5.1.2600.1361'of the same file.` This warning message means that: • **The Windows Installer Patch**, `WindowsXP-KB828741-x86-ENu.EXE`, installs version `5.1.2600.1361` of `RPCRT4.dll`, an operating system file. and • **Corel Draw**, `Coreldraw 12.0.0.458 v1.0`, installs a file that is dependent upon the same operating system file: `RPCRT4.dll`. Therefore, this warning message means that you should evaluate Corel Draw on a system that includes this Windows Installer Patch to insure that you can safely distribute this package within your enterprise. ***Tip*** • *The OS Snapshot serves as a representation of the underlying baseline system in your enterprise. If you include an OS Snapshot in your Patch Impact Analysis, then the file version information displayed in warning messages is derived from that OS Snapshot. If an OS Snapshot is not included in the analysis, the file version information will be displayed as "unknown". As such, it is recommended to include an OS Snapshot in your analysis because this allows a finer-tuned evaluation of impacts based upon file and version information.*

Viewing Patch and Patch Impact Information in Application Manager and ConflictSolver

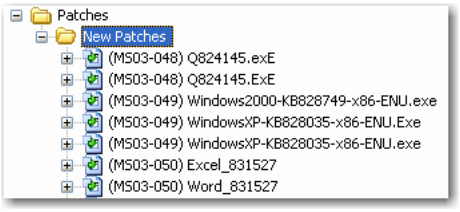
Patch content and analysis information can be viewed on the ConflictSolver and Application Manager Products and Patches tabs.

- Viewing Patch Content Information
- Viewing Dependencies
- Viewing Associated Patches
- Viewing Patch Impacts on the Products Tab
- Viewing Patch Impacts on the Patches Tab

Viewing Patch Content Information

Task: ***To view patch content information:***

1. Launch ConflictSolver or Application Manager and click the **Patches** tab. The **All Patches View** opens.
2. Expand the listing. All of the patches that have been imported into the Application Catalog are listed. Newly imported patches are listed in the **New Patches** group.



   In Application Manager, you can organize your patches into groups according to your business needs. See Organizing Your Application Catalog Using Groups.

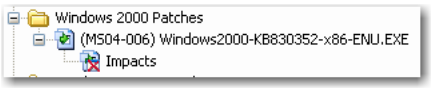
3. Select a patch. The **Patches View** opens.

   The following information on the selected patch is listed:

   - **ID**—Microsoft Security Bulletin ID. Click on the ID number link to view this bulletin on the Microsoft Web site.
   - **Title**—Title of patch
   - **Release Date**—Date the patch was released by Microsoft.
   - **KB Article**—Microsoft Knowledge Base article ID. Click on the KB Article link to view that article on the Microsoft Web site.
   - **Imported On**—Date patch was imported into the Application Catalog

- **Groups**—Listing of which groups this patch belongs to. A patch can belong to multiple groups. You can copy a patch into multiple groups.
- **Description**—You can enter a description of the patch in this field.

4. If an impact analysis was performed on a package using this patch, a **Patch Impacts** node is listed below the patch:



If you select the **Impacts** node, the **Impacts View** opens. See Viewing Patch Impacts on the Patches Tab for detailed information.

5. In Application Manager, you can view additional detailed patch information by selecting a patch and then selecting **Patch Properties** from the context menu. The **Patch Properties** dialog box opens.

The following information is listed:

- **General Tab**—View the title and a summary of a selected patch.
- **Contents Tab**—Lists all of the files and registry data contained in the selected patch.
- **Product Tab**—Lists the products that are updated by the selected patch.

Viewing Dependencies

On the ConflictSolver Dependencies View, which is accessed by selecting the **Dependencies** node under a product in the Product View, you can view a list of all of the files of a selected product that have dependencies with files used by other products or operating systems in the Application Catalog. This view is displayed for Windows Installer .msi packages in which file dependency information exists.

Dependency information is generated by the Patch Impact Analysis Wizard. You can also generate this information by selecting a product and then selecting **Scan for Dependencies** from the context menu.

Task: ***To view product dependencies:***

1. Launch ConflictSolver and click the **Products** tab. The **Group View** opens.
2. Select the product that you want to examine. The **Product View** for that product opens.
3. Expand the product to view its constituent views. If the **Dependencies** node appears, select it to display the Dependencies View.

   If the **Dependencies** node does not exist, select **Scan for Dependencies** from the context menu, and then select the **Dependencies** node.
4. Make selections from the **Files With Dependencies** list to further refine this listing, or select **(All)** to display all dependencies.
5. If you double-click on a dependency, the conflicting file on the target product will be displayed and highlighted in the Tables View.

Note • *If the **Only Display View Nodes With Data** option on the **General** tab of the ConflictSolver **Options** dialog box is selected, if you scan for dependencies and no dependencies are found, the **Dependencies** node will still not be displayed.*

Viewing Associated Patches

On the Associated Patches View, you can view a list of imported patches that, if installed, would update that product.

Task: ***To view associated patches:***

1. Launch **ConflictSolver** and click the **Products** tab. The **Group View** opens.
2. Select the product that you want to examine. The **Product View** for that product opens.
3. Expand the product to view its constituent views and select the **Associated Patches** node. The Associated Patches View opens, displaying patches associated with that product.
4. If you double-click on a patch in the **Associated Patches View**, the Patch View (on the **Patches** tab) for that patch opens, listing general information on the selected patch.

   See Viewing Patch Content Information for more information.

Viewing Patch Impacts on the Products Tab

The information listed on the Patch Impact View (Products Tab) depends upon the selection that is made in the **Impact category** list:

- When **Summary** is selected, the patches for which there is patch impact data persisted against the product are listed
- When **File Impacts** is selected, all impacts against this product are listed, and the patch that caused the impact is identified.

Task: ***To view patch impacts:***

1. Launch ConflictSolver and click the `Products` tab. The **Group View** opens.
2. Select the product that you want to examine. The **Product View** for that product opens.
3. Expand the product to view its constituent views and select the **Patch Impacts** node. The **Patch Impacts View** opens.
4. Select **Summary** from the **Impact category** list to view a list of patches for which there is patch impact data persisted against the product.
5. Select **File Impacts** from the **Impact category** list to view a list of all impacts against this product and the patch that caused the impact.

6. If you double-click on one of the patches in the list, the Patch View for that patch will open.

   If no patch impacts have been identified for this product, **File Impacts** will not be listed in the **Impact category** list.

7. To perform patch impact analysis, select the product in the product tree on the **Product View**, and then select Patch Impact Analysis Wizard from the context menu.

Note • *All patch information displayed in ConflictSolver comes from the Application Catalog (for imported patches); no information about patches from mssecure.xml file is displayed in ConflictSolver.*

Viewing Patch Impacts on the Patches Tab

The information listed on the Impact Analysis View (Patches Tab) depends upon the selection that is made in the **Impact category** list:

- When Summary is selected, details regarding the last time impact analysis was performed are listed.
- When File Impacts is selected, all impacts against this patch are listed, and the product(s) that the patch impacted are identified.

Task: ***To view patch impacts:***

1. Launch ConflictSolver and click the Patches tab. The Patches Group View opens.
2. Select the patch that you want to examine. The Patch View for that patch opens.
3. Expand the patch to view its constituent views and select the **Impacts** node. The Impact Analysis View (Patches Tab) opens.
4. Select **Summary** from the **Impact category** list to view details regarding the last time impact analysis was performed.
5. Click one of the links to open the Patch Impact Information Dialog Box.

   The **Patch Impact Information** dialog box lists the following products on individual tabs:

   - Products that were checked for impacts for the selected patch during patch impact analysis.
   - Products that have been updated since the last impact analysis was performed.
   - Products that have been removed from ConflictSolver since the last impact analysis was performed.
   - Products that have been added to the Application Catalog since the last impact analysis was performed.

6. Select **File Impacts** from the **Impact category** list to view a list of all impacts against this patch and the products that was impacted.

   If you double-click on one of the listed products, that product will be opened in the **Tables View** with the row that is causing the problem highlighted.

   If no impacts have been identified for this patch, **File Impacts** will not be listed in the Impact category list.

7. To perform patch impact analysis, select the patch in the tree, and then select **Patch Impact Analysis Wizard** from the context menu.

Note • *All patch information displayed in ConflictSolver comes from the Application Catalog (for imported patches); no information about patches from the mssecure.xml file is displayed in ConflictSolver.*

Generating the Patch Report

In Application Manager and ConflictSolver, you can generate a Patch Report which lists detailed information about each patch that has been imported into the Application Catalog, including patch impact analysis information.

The Patch Report is generated in Web Archive format (`.mht`), a single, stand-alone HTML file that can be easily viewed in a Web browser and copied and emailed throughout your organization. The report is also printer-friendly.

Task: ***To generate a Patch Report:***

1. Launch ConflictSolver or Application Manager and click on the **Patches** tab. The **Patches Group View** opens.
2. Expand the patch listing and select the patch that you want to generate a report on. The **Patch View** opens.
3. Select **Generate Report** from the context menu. The **Save Patch Report As** dialog box opens, prompting you to select a location for the `.MHT` file.
4. Confirm the report name and location and click **Save**. The report will be generated and will open in a new browser window. Click on the icons to expand or contract that section of the report. The report includes the following sections:

Icon	Name	Description
	General Information	Includes patch title, Microsoft Security Bulletin ID, URL of the Microsoft Security Bulletin, patch summary, and the date the patch was released by Microsoft.
	Products Updated	Products updated by the patch.
	Files	Files included in the patch.
	Registry Entries	Registry entries that are added or modified by the patch.
	Impacts	Products that were checked for impacts during Patch Impact Analysis, and impacts that were detected during Patch Impact Analysis.

5. To print the report with all sections expanded, click the **Print Page** icon.

Reference

This Reference section includes the same topics that are displayed when you click a help button from panel of the Patch Impact Analysis Wizard or OS Security Patch Wizard, or from the Patch Properties dialog box. Reference information is organized into the following areas:

- Patch Impact Analysis Wizard
- OS Security Patch Wizard
- Patch Properties Dialog Box

Patch Impact Analysis Wizard

You can use the Patch Impact Analysis Wizard to identify conflicts between Microsoft operating system patches and the packages and OS Snapshots in your Application Catalog. This helps you determine how specific MSI packages or OS Snapshots would be affected when a Microsoft OS patch is installed.

The Patch Impact Analysis Wizard consists of the following panels:

- Welcome Panel
- OS Snapshot Panel
- Source Patches Panel
- Target Products Panel
- Summary Information Panel

When run, the output report is displayed on the **Patch Impact** tab of the ConflictSolver Output Window.

Welcome Panel

The first panel of the Patch Impact Analysis Wizard welcomes you to the Wizard.

This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 17-3 • Patch Impact Analysis Wizard Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help about the specific Patch Impact Analysis Wizard panel.

OS Snapshot Panel

On the OS Snapshot Panel, you can optionally select an OS Snapshot to be used to identify specific file information for any patch impacts that are discovered.

The OS Snapshot serves as a representation of the underlying baseline system in your enterprise. If you include an OS Snapshot in your Patch Impact Analysis, then the file version information displayed in warning messages is derived from that OS Snapshot. If an OS Snapshot is not included in the analysis, the file version information will be displayed as “unknown”. As such, *it is recommended to include an OS Snapshot* in your analysis because this allows a finer-tuned evaluation of impacts based upon file and version information.

The following options are included:

Table 17-4 • OS Snapshot Panel Options

Option	Description
Group/OS Snapshot Tree	Select an OS Snapshot to include in your Patch Impact Analysis.

Source Patches Panel

On the Source Patches Panel, you select the patches that you want to include in your analysis.

The following options are included:

Table 17-5 • Source Patches Panel Options

Option	Description
Filter by Product	Lists all the Microsoft operating systems that have a patch that has been imported into the Application Catalog. Make a selection from this list to restrict the list of patches displayed on this panel.
Patch Listing	Patches that meet the selected filter criteria are listed. The following information is provided: • **Id**—Number identifying the Microsoft OS patch. Select the check box in this column to include this patch in your analysis. • **Name**—Name of the patch file. • **Title**—Description of the purpose of the patch file. • **Release Date**—Date that this patch was released by Microsoft.
Select All Button	Click to select all listed patches.
Clear All Button	Click to unselect any selected patches.

Table 17-5 • Source Patches Panel Options (cont.)

Option	Description
Patch Properties	Click to access the Patch Properties Dialog Box for this patch, which provides the following information: • **General Tab**—Summary information on the patch. • **Contents Tab**—Listing of the DLL files and registry entries associated with this patch. • **Products Tab**—A listing of the Product and that product's service packs that this patch is associated with.

Target Products Panel

On the Target Products Panel, select the products or groups of products that you want to perform patch impact analysis on. The Patch Impact Analysis Wizard will analyze the products you select here against the patches you selected on the **Source Patches Panel** for impacts.

The following options are included:

Table 17-6 • Target Products Panel Options

Option	Description
Group/Product Tree	A listing of all groups and products in the open Application Catalog.
Select All Button	Click to select all listed groups and products.
Clear All Button	Click to unselect all selected groups or products.

Summary Information Panel

The Summary Information Panel lists a summary of the options you selected in the Patch Impact Analysis Wizard.

Click **Finish** to begin the Patch Impact Analysis.

OS Security Patch Wizard

The OS Security Patch Wizard consists of the following panels:

- Welcome Panel
- Patch File Selection Panel
- Patch File Branch Selection Panel
- Patch Bulletin Information Panel

- Additional Patch Information Panel
- Patch Import Summary Panel
- Import Process Results Panel

Welcome Panel

OS Security Patch Wizard is a new AdminStudio tool that is used to import Microsoft operating system patch information into the Application Catalog.

Using the OS Security Patch Wizard, you can open Microsoft patch update files and incorporate the data in those files into the AdminStudio Application Catalog. Click **Next** to continue.

Patch File Selection Panel

On the **Patch File Selection** panel, select a Microsoft operating system patch file that you have downloaded from the Microsoft TechNet Security Center.

Note • *The procedure for identifying the Microsoft OS patch files that you want to import, and for downloading them from the Microsoft web site is described in Identifying and Downloading Microsoft Operating System Patch Files. Microsoft operating system patches can be downloaded from the Microsoft TechNet Security Center at:*

`http://www.microsoft.com/technet/security/default.mspx`

When you click **Next**, the OS Security Patch Wizard reads and verifies the patch file and progress messages are displayed in the **Processing Status** area of this panel.

- **Prompt to connect to the Application Catalog**—If you opened **OS Security Patch Wizard** from the **Start** menu, after you click **Next**, you will be prompted to provide database connection information. If you opened OS Security Patch Wizard from Application Manager, you will already be connected to the Application Catalog and will not be prompted to connect again.
- **Invalid patch file message**—If you are attempting to import a patch file that is invalid, a message will be displayed and you will be unable to proceed with the import.

When processing of the patch file is successfully finished, one of the following panels opens:

- If you have downloaded a dual-mode patch, the Patch File Branch Selection Panel opens.
- If the patch you are importing is not a dual-mode patch, the Patch Bulletin Information Panel opens.

Patch File Branch Selection Panel

On the **Patch File Branch Selection** panel, which only opens if you are importing a dual-mode patch, you need to identify the branch of the patch file that you want to import into the Application Catalog.

- **Dual-mode patches**—Windows XP software updates (for example, security updates, critical updates, or hot fixes) that are released after Service Pack 1 (SP1) may include binary files for both Windows XP-based and Windows XP-based SP1 computers. Therefore, there are multiple processing paths for installing this patch. These *dual-mode* update packages are designed to install post-SP1 updates on Windows XP-based computers regardless of whether they are running SP1. If you do not have SP1 installed, the dual-mode update also puts a post-SP1 version of the update on your computer, which is automatically installed if you install SP1. If you remove SP1, the pre-SP1 version of the update is restored
- **Specifying the branch to import**—On this panel, AdminStudio has preselected the branch of the patch that appears to install the latest version of this patch. You may choose to accept the default selection or make your own selection. It is permissible to import the same dual-mode patch multiple times, selecting different branches each time. Subsequent panels will allow you to uniquely identify each patch branch.

On the **Patch File Branch Selection** panel, the following information is listed for each branch of the patch:

Table 17-7 • Information Displayed for Each Branch of a Dual-Mode Patch

Property	Description
Name	Name identifying each branch of the dual-mode patch.
Branch	The OS Security Patch Wizard imports a Microsoft operating system patch using the patch's INF file, which is a text file that contains all the information necessary to install a device (such as INI and registry changes, filenames, and locations of the source files on source media). The **Branch** listed in this column is the name of the patch's INF file minus its extension. Because all obvious identifiable information within the INF file (like **Name**) is often identical, this Branch information is included to help differentiate the patches within the update. A sample branch name could be: `update_SP2GDR` or `update_SP2QFE`, where: • **QFE** = Quality Fix Engineering updates (also known as hot fixes). These include security updates, feature packs, update roll-ups, drivers, and critical updates. They are created when an issue is identified that needs a fix in a specific and small subset of customer environments. These fixes have not been tested for the same length of time for regressions or incompatibilities as what a GDR fix typically goes through. • **GDR** = General Distribution Release updates. These updates can be found on the Windows Update site, meaning that they are available to a wider audience. GDRs are tested to a greater degree than QFEs. ***Note*** • *QFEs generally contain all the prior fixes from both GDR and QFE releases while GDR releases do not contain QFE fixes.*

Table 17-7 • Information Displayed for Each Branch of a Dual-Mode Patch

Property	Description
File Count	Number of files included in the patch.
Registry Count	Number of registry entries included with the patch.

Patch Bulletin Information Panel

From the patch's associated Security Bulletin, copy and paste the **Bulletin ID** and the **Bulletin Title** associated with the patch file you are importing. You should also enter the **Bulletin Summary** and **Bulletin Release Date**.

Important • *Security Bulletin data is used by Application Manager to integrate with the Microsoft Web site. Therefore, it is important that you accurately enter this data.*

Note • *For information on how to view Microsoft Security Bulletins, see Identifying and Downloading Microsoft Operating System Patch Files.*

Additional Patch Information Panel

The **Additional Patch Information** panel displays some data that was extracted from the patch file. Review and edit the information and click **Next** to continue.

The following information is included:

Table 17-8 • Information Entered on the Additional Patch Information Panel

Property	Description
Name	By default, this field displays the name of the selected patch file, such as: `Windows2000-KB38827-x86-ENU.EXE` Edit this name to make it easier for you to identify this patch when it is listed in the Application Catalog.
Associated KB Article	The KB article number is extracted from the patch file and is automatically entered in this field. ***Important*** • *The Knowledge Base article identification information is used by Application Manager to integrate with the MIcrosoft Web site. Therefore, it is important that you accurately enter this data.*

Table 17-8 • Information Entered on the Additional Patch Information Panel

Property	Description
Effected Product	Optionally, enter the name of the product that this patch effects. This information is typically something like Windows XP, Windows Vista, etc.
Additional Comments	Comments are extracted from the patch file and automatically entered in this field. You can edit these comments.

Patch Import Summary Panel

The **Patch Import Summary** panel lists the selections you have made in the previous panels of the OS Security Patch Wizard. Click **Import** to initiate the patch import.

Import Process Results Panel

The **Import Process Results** panel lists the results of the import of the selected patch file. The results displayed on this panel, along with the summary information shown in the previous panel, are output to a log file.

To import another patch, click **< Another** to return to the beginning of the OS Security Patch Wizard.

When you have finished importing Microsoft OS patches, click **Finish**.

Patch Properties Dialog Box

You can access the **Patch Properties** dialog box from several locations:

- **Source Patches Panel of the Patch Impact Analysis Wizard**—Select a patch and then click the **Patch Properties** button.
- **Application Manager Patches Tab**—Right-click a patch and then click **Patch Properties**.

The **Patch Properties** dialog box consists of the following tabs:

Table 17-9 • Patch Properties Dialog Box Tabs

Tab	Description
General Tab	View the title and a summary of a selected patch.
Contents Tab	Lists all of the files and registry data contained in the selected patch.
Products Tab	Lists the products that are updated by the selected patch.

General Tab

The **General** tab of the **Patch Properties** dialog box lists the patch **Title** and includes a **Summary** of the purpose of the patch and the patch **Release date**. From the **General** tab, you can also click a link to go directly to the Microsoft Web site and view the Microsoft Security Bulletin and Microsoft Knowledge Base article for that patch:

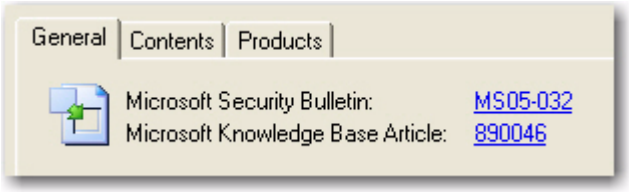


Figure 17-4: Links to Microsoft Security Bulletin and Knowledge Base Article

Contents Tab

The **Contents** tab lists all of the files and registry data contained in the selected patch. The following options are listed:

Table 17-10 • Patch Properties Dialog Box / Contents Tab Options

Option	Description
Files	This section lists all of the files included in the patch. The following information is displayed for each file: • **File**—File name. • **Directory**—Location where file will be installed. • **Version**—File version.
Registry data	The registry is a database repository for information about a computer's configuration. This section lists all of the registry data included in the patch. The following information is listed: • **Key**—Name of the registry key. • **Name**—Name of the registry value. • **Value**—Data stored for the registry value.

Products Tab

The **Products** tab lists all of the products updated by this patch, and each product's associated Service Packs. This tab includes the following options:

Table 17-11 • Products Tab Options

Option	Description
Products	All of the products updated by this patch are listed. Select a product from the list to see its associated Service Packs.
Service Packs	Listing of all of the Service Packs associated with the selected product.

18

Isolating Applications Using Application Isolation Wizard

Application isolation is one solution to component versioning conflicts. Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested.

Application Isolation Wizard user documentation is presented in the following sections:

Table 18-1 • Application Isolation Wizard User Documentation

Section	Description
About Application Isolation Wizard	Explains the reasons you would isolate applications and introduces you to the Application Isolation Wizard.
Launching the Application Isolation Wizard	Explains how to open the Application Isolation Wizard from the AdminStudio interface.
Isolation Methods	Describes the two isolation methods used by Application Isolation Wizard: Manifests and Assemblies, and Windows Installer Isolated Components.
Assemblies	Explains how Assemblies are used.
Manifests	Explains how Manifests are used.
Digital Signatures	Explains how Digital Signatures are used.
Isolating Applications	Describes how to use the Application Isolation Wizard to isolate applications.
Setting Assembly Naming Conventions	Explains assembly naming conventions.

Table 18-1 • Application Isolation Wizard User Documentation (cont.)

Section	Description
Modifying the Default Isolation Recommendations	Describes how to modify the default isolation recommendations when using Windows Installer isolated components and when using manifests for isolation.
Filtering File Listings when Manually Configuring Isolation	Explains how to filter the file listing when manually configuring application isolation using manifests, and when using Windows Installer isolated components.
Servicing Published Shared Assemblies	Explains how to service (alter) shared assemblies after publishing to update them as necessary.
Application Isolation Wizard Reference	Describes Application Isolation Wizard panels and dialog boxes.

About Application Isolation Wizard

Application isolation is one solution to component versioning conflicts, commonly known as “DLL Hell.” Isolation reduces versioning conflicts by modifying an application so it always loads the versions of components—such as DLLs—with which it was originally developed and tested. This is accomplished by providing DLLs and other shared components for specific applications, and placing information traditionally stored in the registry into other files that specify the locations of these isolated components. Application isolation provides increased stability and reliability for applications because they are unaffected by changes caused by installation and ongoing maintenance of other applications on the system.

Depending on the isolation method used in the Application Isolation Wizard, you can partially or totally isolate an application. When using assemblies and manifests to isolate applications for Windows XP systems, the assemblies can be updated following deployment without necessitating application reinstallation.

Reasons to Isolate Applications

You would want to isolate an application if:

- You want to resolve incompatibilities between different versions of shared components.
- You want to reduce the complexity of the installation by storing COM activation data in a manifest instead of the registry.
- You want to insulate the application from changes to shared components.

Tip • *Following isolation, you can use the Dynamic Dependency Scanner in InstallShield Editor to verify isolated files are loaded from a different directory.*

Reasons Not to Isolate an Application

You would not want to isolate an application if, following application isolation, you discover that the application no longer works because of an internal dependency on a component that has been moved during the isolation process.

Isolating Repackaged Setups Using Repackager

Application Isolation Wizard is a stand-alone tool which accepts a Windows Installer package as input and outputs a new, isolated Windows Installer package. You can also generate an isolated version of a repackaged setup immediately after the build step in Repackager.

If you open a Repackager project and choose the **Create an isolated version of the Windows Installer package** option on the Repackaged Output View, Repackager builds an isolated version of the Windows Installer package immediately after building the non-isolated version.

Both methods of isolating a package are performed using the same Application Isolation Wizard functionality. However, the Application Isolation Wizard provides a user interface experience that allows the user to extend the initial “dependency scanning” process for identifying file isolation candidates, while in Repackager, you specify your assembly and digital signing isolation options on the Isolation Options dialog box, and then those selections are applied to all isolated packages created by Repackager.

For more information, see Isolating Windows Installer Packages.

Launching the Application Isolation Wizard

To launch the Application Isolation Wizard, perform the following steps.

Task: ***To launch the Application Isolation Wizard:***

1. Launch AdminStudio.
2. From the Tools Gallery, click the Application Isolation Wizard icon on the left side.

The Application Isolation Wizard launches and you can immediately begin the application isolation process.

Isolation Methods

There are two isolation methods supported by the Application Isolation Wizard™: Manifests and Assemblies and Windows Installer Isolated Components.

Assemblies and Manifests

Application isolation using assemblies and manifests is the recommended isolation method for Windows XP. These assemblies and manifests provide the same end result as Windows Installer isolated components, but keep all information outside of the registry and do not require the components to be installed in the same folder as the application. This reduces the chance of errors after isolation resulting from how the application was written.

Assemblies and manifests only work under the Windows XP operating system.

Windows Installer Isolated Components

Application isolation using Windows Installer isolated components is for Windows 98 SE, Me, and 2000. It can also be used on Windows XP, but using assemblies and manifests is the better solution. The isolated component method copies shared files (typically DLLs) into an application's folder instead of a shared location. The application then uses these files instead of global ones, preventing modifications made by other applications from affecting the shared files. As a result, the application always uses the versions of these files with which it was deployed.

To instruct an application use the private files rather than shared versions, the Application Isolation Wizard populates the IsolatedComponent table with the necessary logic to use private files stored in the same folder as the application. When Windows Installer performs the setup, data from the IsolatedComponent table populates a .local file, which ultimately directs how to use the private files.

Windows Installer isolated components still require some information to be written to the registry, and also require the associated components to be in the same folder as the application. While in most cases this will still provide required isolation, depending on how the application was written, the movement of these associated components from their original locations may prevent the application from functioning correctly.

Assemblies

Assemblies are DLLs or other portable executable files that applications require to function. Under Windows XP, these can be either shared or private. Private assemblies are typically stored in the same directory as the application they support. Shared assemblies are stored in the WinSxS directory, and are digitally signed.

By creating manifests for assemblies, the Application Isolation Wizard™ allows you to create self-contained applications that can use different versions of the same DLL or other portable executable, without any version conflicts.

Shared Assemblies

Shared assemblies are assemblies available to multiple applications on a computer. Applications that require these assemblies specify their dependence within a manifest. Multiple versions of shared assemblies can be used by different applications running simultaneously.

These assemblies are stored in the WinSxS directory, and must be digitally signed for authenticity. After deployment, the version of shared assemblies can be changed, allowing for changes in dependencies.

Private Assemblies

Private assemblies are assemblies created for exclusive use by an application. They are accompanied by an assembly manifest, which contains information normally stored in the registry. Private assemblies allow you to totally isolate an application, eliminating the possibility that dependent files may be overwritten by other applications.

These assemblies are always stored in the same location as their associated executable.

Manifests

The Application Isolation Wizard™ can create two types of manifests: application manifests and assembly manifests.

Table 18-2 • Manifest Types

Manifest Type	Description
Application	Application manifests are XML files that describe an isolated application. This descriptive information includes the relationship between the application and its dependent files. Typically, the naming convention for a manifest is ApplicationName.Extension.manifest. For example, if the application was HelloWorld.exe, the manifest file is called HelloWorld.exe.manifest.
Assembly	Assembly manifests are XML files that describe an application's assemblies. This includes components such as DLLs. Information stored in the assembly manifest, such as COM registration information, ProgIDs, etc., is usually stored in the Registry. However, by making it independent from the registry, only that application can use the dependent files described in the manifest. This enables you to have multiple versions of the same DLL or other portable executable file on a system without generating compatibility conflicts. Typically, the naming convention for a manifest is AssemblyName.Extension.manifest. For example, if the component was Goodbye.dll, the manifest file is called Goodbye.dll.manifest.

Manifests as New Components

When you create manifests, the Application Isolation Wizard supports putting them into new components. If you do not select this option from the Advanced Options dialog box, the manifest will be added to the same component as the assembly.

Digital Signatures

Like conventional signatures, digital signatures identify you (or your organization) to end users. In the context of application isolation, a digital signature identifies you or your organization as the creator of shared assemblies. This ensures that the identity of a shared assembly can be verified for authenticity. Digital signatures in the Application Isolation Wizard™ require a combination of a digital certificate and a code signing technology.

Certificates

Digital certificates identify you and/or your company to end users to assure them the assembly they are about to use has not been altered. They are issued by a certification authority such as VeriSign, or created using a combination of software publishing credentials (`.spc`) and a private key (`.pvk`), both also issued by a certification authority. The certificate includes the public cryptograph key, and, when used in combination with a private key, can be used by end users to verify the authenticity of the signor.

You can create a certificate file from the constituent PVK and SPC files and import it into the Certificate Store using the PVK Digital Certificate Files Importer. You can then export the certificate (`.cer`) file for use outside of the certificate store.

Caution • *Certificate files must be 2048-bit or higher. For more information, see the article: Assembly Signing Example on the Microsoft Developer Network Web site.*

Code Signing Technologies

The Application Isolation Wizard™ supports two code signing technologies:

Table 18-3 • Supported Code Signing Technologies

Technology	Description
Credentials	Credentials consist of both Software Publishing Credentials (.spc file) and a private key (.pvk file). These two files are required in conjunction with the certificate to sign shared assemblies.
Certificate Name in the Store	Using Microsoft's Certificate Store technology, the combined software publishing credentials and private key can be placed in a repository for multiple uses. The name of the certificate is provided as opposed to the constituent files in the Credentials code signing technology.

Software Publishing Credentials

You must supply a certification authority with specific information about your company and software to obtain software publishing credentials in the form of an .spc file. Your software publishing credentials are used to generate a digital signature for your assembly.

The .spc file and .pvk (private key) file you enter in the Digital Signature tab of the Advanced Options dialog box compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Certificate Store

To perform code signing, both private key and software publishing credential information must be supplied. This must occur each time a package is signed. However, the certificate store serves as a central repository for this information, allowing you to associate the same credentials and key with multiple packages. This simplification is particularly useful when isolating applications, as typically the code signing information will be identical for all shared assemblies. Ultimately, the certificate store removes the burden of managing private key and software publishing credential information.

You can create a certificate file from the constituent PVK and SPC files and import it into the certificate store using the PVK Digital Certificate Files Importer. You can then export the certificate (`.cer`) file for use outside of the certificate store.

Private Keys

A private key (a file with the extension .pvk) is granted by a certification authority. The Application Isolation Wizard™ uses the private key you enter in the Digital Certificates tab of the Advanced Options dialog box to digitally sign your shared assembly and ensure end users of its content's authenticity.

The .spc (Software Publishing Credentials) file and .pvk file you enter in the Digital Signature tab compose the digital certificate for shared assemblies.

Contact a certification authority such as VeriSign for more information on the specifics of software publishing credentials.

Isolating Applications

To isolate applications within a Windows Installer package or a merge module, perform the following steps.

Task: ***To isolate applications within a Windows Installer package (.msi) or merge module (.msm):***

1. Launch the Application Isolation Wizard™. The **Welcome Panel** appears.
2. From the **Welcome Panel**, click **Next**. The **Windows Installer File Selection Panel** appears.
3. From the Windows Installer File Selection Panel, specify the Windows Installer package (`.msi`), Windows Installer self-extracting executable file (`setup.exe`), merge module (`.msm`) containing applications you want to isolate. Click **Next**. The Isolation Method panel appears.
4. From the **Isolation Method Panel**, select the isolation method(s) you want to use.
5. If you are using manifests, you can click Advanced to configure manifest properties and digital signature information (if required) on the Advanced Options Dialog Box.
6. Click **Next**. The **Summary Information Panel** appears.
7. From the **Summary Information Panel**, confirm the isolation selections.
8. If you want to manually configure isolation, click Modify.
   - **If you are using manifests** to isolate your application—either alone or in conjunction with Windows Installer isolated components—the **Manifest and Assembly Design** dialog box appears.
   - **If you are only using Windows Installer isolated components** to isolate the application, the **Isolated Components Design** dialog box appears. After you have completed manually configuring the isolation, click OK to return to the Summary Information Panel.
9. Click Isolate. The **Application Isolation Progress Panel** appears.

   When the Application Isolation Wizard is complete, the **Completing the Application Isolation Wizard Panel** is displayed, providing feedback on whether the Application Isolation Wizard was successful.
10. From the **Completing the Application Isolation Wizard Panel**, click **Finish**.

Setting Assembly Naming Conventions

To set the default naming convention for assemblies, perform the following steps.

Task: ***To set the default naming convention for assemblies:***

1. Launch the Application Isolation Wizard™. The **Welcome Panel** opens.
2. From the Welcome panel, click Next. The **Windows Installer File Selection Panel** opens.
3. From the Windows Installer File Selection panel, specify the Windows Installer package (.msi) or merge module (.msm) containing applications you want to isolate. Click Next. The **Isolation Method Panel** opens.
4. Select the Use manifests for isolation option.
5. Click Advanced. The Manifest Options tab of the **Advanced Options** dialog box opens.

6. Enter your Company name and Division. These two fields create the default assembly naming convention (in the form “Company.Division.Assembly” followed by a number).

Note • *To edit the Assembly Name, you can also click Modify from the Summary Information Panel later in the Wizard to open the Manifest and Assembly Design dialog box, and then click Properties to open the Application Manifest Properties dialog box, where you can edit the Assembly Name.*

7. Click OK. You are returned to the **Isolation Method Panel**.
8. Click Next. The **Summary Information Panel** opens.
9. Click Isolate to proceed with isolation using the specified naming convention.

Assemblies created during application isolation will follow the naming convention as specified.

Modifying the Default Isolation Recommendations

You can modify default isolation recommendations for the following:

- When Only Using Windows Installer Isolated Components
- When Using Manifests for Isolation

When Only Using Windows Installer Isolated Components

To modify the default isolation recommendations when only using Windows Installer isolated components, perform the following steps.

Task: ***To modify the default isolation recommendations:***

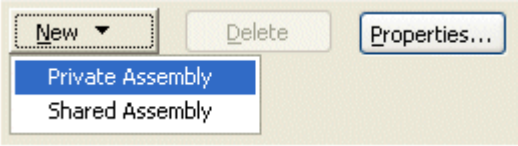
1. From the **Summary Information Panel** of the Application Isolation Wizard™, click **Modify**. The **Isolated Components Design** dialog box is displayed.
2. Select the Applications to be Isolated and then select the Files to Isolate for Selected Application. Repeat as necessary.
3. Click OK. When you return to the Summary Information Panel, verify your settings before isolating.

When Using Manifests for Isolation

To modify the default isolation recommendations when using manifests for isolation, perform the following steps.

Task: ***To modify the default isolation recommendations when using manifests for isolation:***

1. From the **Summary Information Panel** of the Application Isolation Wizard, click Modify. The **Manifest and Assembly Design** dialog box is displayed.
2. If you need to create a new assembly, select the application for which you want to create the assembly, click New, and select the assembly type: Private Assembly or Shared Assembly.



   A new assembly is listed under the selected application.
3. Select the new assembly then add or remove files in the assembly.
4. Click OK. When you return to the Summary Information panel, verify your settings before isolating.

Filtering File Listings when Manually Configuring Isolation

Task: ***To filter the file listing when manually configuring application isolation using manifests:***

1. In the **Manifest and Assembly Design** dialog box, select an assembly from the application tree.
2. Directly below the Files to be added in an assembly window, specify the files displayed from the Show filter.

Task: ***To filter the file listing when manually configuring application isolation using Windows Installer isolated components:***

1. In the **Isolated Components** dialog box, select an application from the application tree.
2. Directly below the Files to Isolate for Selected Application window, specify the files displayed from the Show filter.

Servicing Published Shared Assemblies

Shared assemblies can be serviced (altered) after publishing to update them as necessary. This is accomplished using a publisher configuration, which overrides default configurations specified in the manifest.

For an exhaustive discussion of how to service shared assemblies, refer to the article How To Build and Service Isolated Applications and Side-by-Side Assemblies for Windows XP on the Microsoft Developer Network Web site (msdn.microsoft.com).

Application Isolation Wizard Reference

The Application Isolation Wizard scans Windows Installer packages (.msi) or merge modules (.msm) and isolates applications within them. Isolation ensures that applications always use the specific shared and support files with which they were installed. This prevents the overwriting of previous versions of shared components, and ensures that other applications do not overwrite the versions of shared and support files required by your application.

The following reference topics are available for the Application Isolation Wizard:

- Welcome Panel
- Windows Installer File Selection Panel
- Isolation Method Panel
- Summary Information Panel
- Application Isolation Progress Panel
- Completing the Application Isolation Wizard Panel
- Advanced Options Dialog Box
- Manifest and Assembly Design Dialog Box
- Isolated Components Design Dialog Box
- Assembly Properties Dialog Box
- Application Manifest Properties Dialog Box

Welcome Panel

The Welcome panel is the first panel displayed when you launch the Application Isolation Wizard. It provides a general explanation of application isolation.

Click Next to proceed to the **Windows Installer File Selection Panel**.

Windows Installer File Selection Panel

Enter the full path and file name of the Windows Installer package (.msi) or merge module (.msm) that you want the Application Isolation Wizard to scan for isolation candidates. Alternately, click Browse to navigate to the file.

Click Back to return to the **Welcome Panel**; click Next to proceed to the **Isolation Method Panel**.

Isolation Method Panel

Select the application isolation method(s) you want to use for this Windows Installer package or merge module.

Make the following selections under **Manifests** and **Windows Installer Isolated Components**:

Table 18-4 • Isolation Method Panel Selection Guidelines

If Installation Will Be Deployed ...	Manifests	Windows Installer Isolated Components
Only on **Windows 2003 Server**	Select	Do Not Select
Only on **Windows XP**	Select	Do Not Select
Windows 98 SE, **Me**, and/or **Windows 2000** but not **XP**	Do Not Select	Select
Windows XP *and* **Windows 98 SE**, **Me**, and/or **Windows 2000** ***Note*** • *Your installation package size will increase, but application isolation will work on the appropriate operating systems.*	Select	Select

If you use **Manifests**, click **Advanced** to display the **Advanced Options** dialog box, from which you can configure manifest options and digital signature information.

Summary Information Panel

From this panel, review a summary of your selections prior to isolation.

For granular control over the isolation process, click Modify. If you are only using Windows Installer Isolated Components as the isolation method, the **Isolated Components Design** dialog box appears. Otherwise, the **Manifest and Assembly Design** dialog box is displayed.

Click Back to return to the **Isolation Method Panel**; click Isolate to isolate the application according to your settings. The Application Isolation Progress panel is displayed.

Application Isolation Progress Panel

During application isolation, the progress is displayed on this panel. Information about the applications, assemblies (if using manifests as the isolation method), and files is displayed above the progress bar.

Upon isolation completion (or failure), the **Completing the Application Isolation Wizard Panel** is displayed.

Completing the Application Isolation Wizard Panel

The final panel in the Application Isolation Wizard provides feedback on whether the Application Isolation Wizard was successful.

If the Application Isolation Wizard was successful, the names and locations of the original and output packages are provided. If the Wizard was not successful, this panel informs you that the selected components could not be isolated.

Advanced Options Dialog Box

The **Advanced Options** dialog box, available from the **Isolation Method** and **Ready to Isolate** panels of the Application Isolation Wizard, allows you to configure assembly types, naming conventions, and digital signature options. The **Advanced Options** dialog box presents these options on two tabs: **Manifest Options** and **Digital Signature**.

Once you have finished configuring advanced options, click OK to save your changes, or Cancel to close the dialog box without saving your modifications. When the dialog box closes, you are returned to the panel where you clicked Advanced.

Manifest Options Tab

The Manifest Options tab, available in the **Advanced Options** dialog box, allows you to configure several settings associated with manifests.

These settings include:

Table 18-5 • Manifest Options Tab Options

Option	Description
Assembly Type	This option allows you to select the type of assemblies that Application Isolation Wizard™ will create and use: • Create private side-by-side assemblies in the application folder • Create shared side-by-side assemblies in the WinSxS folder (Default) If you want to use both assembly types, you need to manually configure assemblies from the Manifest and Assembly Design Dialog Box. ***Note*** • *Manifests for shared assemblies must be digitally signed. This can be done in the Digital Signature Tab.* ***Note*** • *A 2048-bit key is required to sign a Windows XP assembly/manifest being installed to the WinSxS folder.*
Assembly Naming Conventions	Specify your company and division information to define the default naming convention that Application Isolation Wizard will use when creating assemblies during application isolation By default, assembly names are specified in the form of: `Company.Division.Assembly`
Create a new component for each assembly	Select this option if you want to create a new component for each assembly created during isolation. This check box applies to all assemblies created. Individual assemblies can be configured from the **Assembly Properties** dialog box on a per-assembly basis. ***Caution*** • *If you are creating assemblies for applications files within multiple components, this option must be selected for successful application isolation.* *If you are planning to deploy this isolated package to operating systems prior to Windows XP, always check this box.*

Digital Signature Tab

The Digital Signature tab, available in the **Advanced Options** dialog box, allows you to configure the certificate information required when using shared assemblies. This required digital signature provides an extra layer of protection, allowing you to obtain information about the company who created a global assembly.

Caution • *The Application Isolation Wizard™ uses timestamping when signing global assemblies. Consequently, you must have an Internet connection on the computer when you create a global assembly.*

You must configure the following options when signing these assemblies:

Table 18-6 • Digital Signature Tab Options

Option	Description
Certificate File	Click the Browse (…) button next to the field and navigate to the certificate file you are using to sign assemblies. A digital certificate identifies you and/or your company to end users and assures them the data they are about to receive has not been altered.
Credentials	Select this option to use credential files as the code signing technology. If you select this option, you must supply the name and location of both your software publishing credential files: **SPC File** and **PVK** File. ***Note*** • *In order to receive a software publishing credentials and a private key, you must supply a certification authority, such as such as VeriSign, with specific information about your company and software.*
SPC File	Specify the name and location of your software publishing credentials file (`.spc`).
PVK	Specify the name and location of your private key file (`.pvk`).
Certificate Name in the Store	Select this option to use an existing certificate file in the Certificate Store as the code signing technology. The Certificate Store is a central repository for certificate files. Using a Certificate Store allows you to reuse the certificate files for different purposes as necessary.

Note • *A 2048-bit key is required to sign a Windows XP assembly/manifest being installed to the WinSxS folder.*

Manifest and Assembly Design Dialog Box

If you are using manifests to isolate your application, either alone or in conjunction with Windows Installer isolated components, the Manifest and Assembly design dialog box is displayed when you click Modify from the Summary Information panel.

When you first display this dialog box, the settings the Application Isolation Wizard™ recommends for this package are displayed. By default, only executables that will be installed in the SystemFolder will be selected for isolation. You can select an application contained in the Windows Installer or merge module and create a new private or shared assembly for that application. You can then select the files to isolate for the selected application. A filter at the bottom of the dialog box allows you to restrict the file types visible.

Click Properties to display the **Application Manifest Properties** dialog box. From this dialog box, you can configure the naming convention for assemblies and manifests, and specify whether you want manifests placed into separate components.

When you have finished performing manual configuration, click OK to return to the Summary Information panel.

Isolated Components Design Dialog Box

If you are only using Windows Installer isolated components to isolate the application, this dialog box is displayed when you click Modify from the Summary Information panel.

When you first display this dialog box, the settings the Application Isolation Wizard™ recommends for this package are displayed. By default, only libraries that will be installed in the SystemFolder will be selected for isolation. You can select an application contained in the Windows Installer or merge module, and then select the files to isolate for the selected application. A filter at the bottom of the dialog box allows you to restrict the file types visible.

When you have finished performing manual configuration, click OK to return to the Summary Information panel.

Assembly Properties Dialog Box

The Assembly Properties dialog box displays information about the manifest and assembly, and can be launched from the Manifest and Assembly Design dialog by selecting an assembly and clicking Properties.

The following groups contain configurable options:

Table 18-7 • Assembly Properties Dialog Box Options

Group	Description
Manifest Details	In the Manifest Details group, you can view the file name for the manifest. It is structured in the form “Company.Division.Assembly.manifest” by default.
Assembly Identity	The Assembly Identity group contains fields for the Assembly Name and Version. When you change the assembly name, the manifest file name changes.

Table 18-7 • Assembly Properties Dialog Box Options (cont.)

Group	Description
Assembly Type	This group allows you to select whether the current assembly is private or shared. If it is shared, you must configure digital signature information in the Advanced Options dialog box.
Create new component	Select this option if you want to create a new component for this assembly. ***Caution*** • *If this assembly contains files that originate from multiple components, this option must be selected for successful application isolation.* *If you are planning to deploy this isolated package to operating systems prior to Windows XP, always check this box.* This check box applies only to this assembly. Global settings for assemblies can be configured from the **Manifest Options** tab of the **Advanced Options** dialog box.

Application Manifest Properties Dialog Box

The Application Manifest Properties dialog box displays information about the manifest and assembly, and can be launched from the Manifest and Assembly Design dialog by selecting an application and clicking Properties.

The following groups contain configurable options:

Table 18-8 • Application Manifest Properties Dialog Box Options

Options Group	Description
Manifest Details	In the Manifest Details group, you can view the file name for the manifest. It is structured in the form “Company.Division.Assembly.manifest” by default.
Assembly Identity	The Assembly Identity group contains fields for the Assembly Name and Version. When you change the assembly name, the manifest file name changes.

Command-Line Options

The Application Isolation Wizard can also be run from the command line. You can specify the following options when running the AIW.exe executable from the command line:

Table 18-9 • Application Isolation Wizard Command-Line Options

Option	Description
-?	Displays command-line help for the Application Isolation Wizard.
-version	Displays the version of AdminStudio.
-i <configuration file>	Allows you to specify a configuration file for Application Isolation Wizard settings. The default file, AIWConfig.ini, is located in <AdminStudio Directory>\Common and can be used as a model. This parameter is optional.
-p <package name>	The name and location of the package or merge module which includes applications you want to isolate. This parameter is mandatory.

Configuration Files

When using the command-line options for the Application Isolation Wizard, you can specify an INI file for configuration using the `-i` parameter. This file should take the following format:

```
[IsolationMethods]
Manifests=1
IsolatedComponents=1

[DigitalSignature]
CertificateFile=
SPCFile=
PVKFile=
CertificateName=
TimeStampAssemblies=

[Manifest]
AssemblyType=0
Company="Company"
Division="Division"
NewComponents=0
```

Each configuration corresponds to a user interface setting in the Application Isolation Wizard, as described below:

Table 18-10 • Configuration File Settings

INI File	UI Setting	Explanation
Manifests	Use manifests for isolation option on Isolation Method panel	Set this value to 1 to use manifests. Manifests only work with Windows XP.

Table 18-10 • Configuration File Settings (cont.)

INI File	UI Setting	Explanation
IsolatedComponents	Use Windows Installer isolated components for isolation on Isolation Method panel	Set this value to 1 to use Windows Installer isolated components.
CertificateFile	Certificate File field on the Digital Signature tab of the Advanced Options dialog box	Provide the name and location of the CER file.
SPCFile	SPC File field on the Digital Signature tab of the Advanced Options dialog box	Provide the name and location of the SPC file
PVKFile	PVK File field on the Digital Signature tab of the Advanced Options dialog box	Provide the name and location of the private key.
CertificateName	Certificate Name in the store field on the Digital Signature tab of the Advanced Options dialog box	Provide the name of the certificate from the certificate store.
TimeStampAssemblies	No corresponding UI setting	Set this value to 0 to disable timestamping during shared assembly creation; set it to 1 to enable timestamping. By default, the Application Isolation Wizard uses timestamping if this value is not configured.
AssemblyType	Assembly Type on the Manifest Options tab of the Advanced Options dialog box	Set this value to 0 to use private assemblies; set the value to 1 to use shared assemblies.
Company	Company field on the Manifest Options tab of the Advanced Options dialog box	Put the name of your company in quotes.
Division	Division field on the Manifest Options tab of the Advanced Options dialog box	Put the name of your division in quotes.
NewComponents	Create new component option in Manifest Options tab of Advanced Options dialog box	Set the value to 1 to create a new component for each manifest.

Manifest Examples

Following are examples of both an application manifest and an assembly manifest:

Application Manifest Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
<assemblyIdentity type="win32" name="InstallShield.Development. AppAssembly4" version="1.0.0.1"
processorArchitecture="x86" />
<description>This manifest was generated by the Application Isolation Wizard</description>
- <dependency>
   - <dependentAssembly>
      <assemblyIdentity type="win32"name="InstallShield.
         Development.LocalAssembly1" version="1.0.0.1"
         processorArchitecture="x86" />
     </dependentAssembly>
  </dependency>
  </assembly>
```

Assembly Manifest Example

```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <assembly xmlns="urn:schemas-microsoft-com:asm.v1"
     manifestVersion="1.0">
  <assemblyIdentity type="win32" name="InstallShield.Development.
     LocalAssembly1" version="1.0.0.1" processorArchitecture="x86" />
  <file name="IsCommonServices.dll">
- <comClass description="CabinetBuilder Class" clsid="
     {8D3FE200-DA96-11D3-BEE7-00105A996B4E}" progid=
     "ISHerculesCommonServices.CabinetBuilder.1"
     threadingModel="Apartment" tlbid="{2491C036-D5B0-11D3-BEE5-
     00105A996B4E}">
     <progid>ISHerculesCommonServices.CabinetBuilder</progid>
  </comClass>
- <comClass description="Cabinet Class" clsid="
     {3C35E807-C92D-11D3-BEDF-00105A996B4E}" progid=
     "ISHerculesCommonServices.CabinetExtractor.1"
     threadingModel="Apartment" tlbid="{2491C036-D5B0-11D3-BEE5-
     00105A996B4E}">
     <progid>ISHerculesCommonServices.CabinetExtractor</progid>
  </comClass>
- <comClass description="InstallShield Common Services Registry
     object" clsid="{3032B526-2C3D-11D4-AB2C-00C04F09719A}"
     progid="ISHerculesCommonServices.Registry.1" threadingModel=
     "Apartment" tlbid="{2491C036-D5B0-11D3-BEE5-00105A996B4E}">
     <progid>ISHerculesCommonServices.Registry</progid>
  </comClass>
  </file>
  </assembly>
```

19

Ensuring Package Quality Using QualityMonitor

Edition • *QualityMonitor is included with AdminStudio Professional and Enterprise Editions.*

QualityMonitor allows you to run a series of built-in tests to installed Windows Installer-based products, helping to ensure they run correctly. When failures occur, QualityMonitor can help identify where problems exist, and ultimately direct you to the solution.

QualityMonitor user documentation is presented in the following sections:

Table 19-1 • QualityMonitor User Documentation

Section	Description
About QualityMonitor	Explains the purpose and benefits of using QualityMonitor.
Creating New QualityMonitor Project Files	Explains how to create a new QualityMonitor project.
Opening Existing QualityMonitor Project Files	Explains how to open QualityMonitor project files, which have an .iqm extension.
Working with Test Cases	Describes the most common tasks you may perform when working with Test Cases.
Deployment Testing	Explains how to perform deployment tests against the installed product, ensuring that the product has been installed correctly.
Lockdown and Runtime Testing	Explains how to test an application when its target environment is restricted in some way, such as in a locked-down environment.

Table 19-1 • QualityMonitor User Documentation (cont.)

Section	Description
Using MSI Doctor to Verify Package Deployment Status	Explains how to use MSI Doctor to verify if an MSI package is installed properly. This helps prevent users from seeing an auto-repair dialog box when they run the application.
Creating Custom Test Cases	Explains how to add additional, custom Test Cases to projects—based on your business needs.
Test Reports	Explains how to create an HTML test report for the current project.
Running QualityMonitor from the Command Line	Explains how to run QualityMonitor from the command line.
QualityMonitor Reference	Provides detailed reference on each user interface element, dialog box, and view in QualityMonitor.

About QualityMonitor

Prior to deploying a Windows Installer–based application, typically you need to test it in the targeted deployment environment to ensure the application works as expected. However, it is often not feasible (or possible) to test each piece of an application's functionality, due to the complexity of the application and/or its interface. Behind the scenes, there may be dozens or hundreds of attempts to access files, registry keys, or services; errors may only become apparent in rare and isolated circumstances.

One major source of failure is when the target environment is restricted in some way, such as in a locked-down environment. In this case, there may be prohibitions on certain COM activation or registry access, which ultimately prevents an application from working correctly.

QualityMonitor allows you to run a series of built-in tests to installed Windows Installer-based products, helping to ensure they run correctly. When failures occur, QualityMonitor can help identify where problems exist, and ultimately direct you to the solution.

Creating New QualityMonitor Project Files

You can create a new QualityMonitor project by selecting the **Create new project** option on the **Welcome to QualityMonitor View**.

Task: ***To create a new QualityMonitor project file (.iqm):***

1. Launch QualityMonitor. The **Welcome to QualityMonitor View** opens.
2. Click **Open** on the **File** menu. The **Open QualityMonitor Project** dialog box opens.

3. Select the **Select an application that is installed on this machine ...** option.
4. Select an application from the available applications list.
5. Click **OK**. A new QualityMonitor project for the selected application is opened, and the **Product Information View** opens.

Opening Existing QualityMonitor Project Files

QualityMonitor's project files have an `.iqm` extension and are opened by performing the following steps.

Task: ***To open an existing QualityMonitor project file (.iqm):***

1. Launch QualityMonitor. The **Welcome to QualityMonitor View** opens.
2. Click **Open** on the **File** menu. The **Open QualityMonitor Project** dialog box opens.
3. Select the **Open QualityMonitor project (.iqm) file** option.
4. Enter or browse to the file you want to open.
5. Click **OK**.

Working with Test Cases

The primary purpose of QualityMonitor is to serve as a diagnostic tool when applications fail to function correctly in deployment environments. This is accomplished primarily through running Test Cases and individual Test Items, and evaluating the results. Topics in this section cover the most common tasks you may perform when working with Test Cases.

- Running Individual Test Items
- Running Multiple Test Items
- Adding Test Item Comments
- Adding Test Case Comments
- Viewing Test Item Details
- Clearing Test Case Results
- Manually Setting Test Case Status
- Manually Setting Test Item Status
- Filtering Test Case Data

Running Individual Test Items

You can choose to run a **Deployment Test** on an individual **Test Item**.

Task: ***To run an individual Test Item:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. In the **Test Items** list, select the **Test Item** you want to run and click the Run button.

   Depending on whether the **Test Item is** automatic or requires opening or launching files, you may need to perform some manual tasks prior to results being returned. When the test is complete, QualityMonitor displays **Passed** or **Failed** in the **Status** column of the selected **Test Item**.

Running Multiple Test Items

You can choose to run a **Deployment Test** on a multiple **Test Items** at once.

Task: ***To run multiple Test Items:***

1. Create or open a QualityMonitor project. The **QualityMonitor Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files

- Prog IDs
- Services
- Shortcuts
- Type Libraries
- ODBC Data Sources
- ODBC Drivers

3. In the **Test Items** list, select the **Test Items** you want to run. Multiple selection is supported using the Shift and Ctrl keys.
4. To run only the selected **Test Items**, click the **Run** button. To run all **Test Items**, click the **Run All** button.

   Depending on whether the **Test Item** is automatic or requires opening or launching files, you may need to perform some manual tasks prior to results being returned. When the tests are complete, QualityMonitor displays **Passed** or **Failed** in the **Status** column of the selected **Test Items**.

Adding Test Item Comments

You can choose to add comments to a **Test Item**—perhaps to document why it passed or failed, or to note an issue that needs attention.

Task: ***To add comments to an individual Test Item:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. In the **Test Items** list, right-click the **Test Item** to which you want to add comments, and select **Test Item Information** from the context menu.The **Test Item Information** dialog box appears.
4. Enter comments into the **Comments** field.

5. When finished, click **OK**.

Adding Test Case Comments

You can add comments associated with an entire Test Case.

Task: ***To add comments to a Test Case:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. In the **Comments** box at the top right of the View, enter comments. Your comments are automatically saved.

Viewing Test Item Details

When a Test Item fails, you can view details about it, including the error message associated with it. This information is displayed on the **Test Item Information** dialog box, along with the Test Item name, status, and any comments that have been entered.

Task: ***To view Test Case details:***

1. Create or open a QualityMonitor project. The **QualityMonitor Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations

- Help Files
- Prog IDs
- Services
- Shortcuts
- Type Libraries
- ODBC Data Sources
- ODBC Drivers

3. Right-click on a **Test Item** and select **Test Item Information** from the context menu. The **Test Item Information** dialog box opens, and the following details are listed:
   - **Test Item**—Name of the selected Test Item.
   - **Status**—Status of the selected Test Item: Passed, Failed, or Pending.
   - **Comments**—Any comments that were previously entered.
   - **Test Details**—If this Test Item has **Failed**, a brief explanation of the reason the Test Item failed the test is listed.
4. When finished viewing test details, click **OK** to close the dialog box.

Clearing Test Case Results

You can clear Test Case results for all Test Items from a previous Test Case execution.

Task: ***To clear Test Case results (all Test Items) from a previous Test Case execution:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources

- ODBC Drivers

3. Click the **Reset Results** button.

When you click **Reset Results**, the status of *all* Test Items is reset. To reset the status of one individual **Test Item**, right-click that **Test Item**, point to **Set Status** and select **Pending** from the context menu. See Manually Setting Test Case Status.

Manually Setting Test Case Status

Depending on your business practices and standards, you may want to override the status of a Test Case in the View List from its current state. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.

Task: ***To manually set the Test Case status:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. In the **Test Case Status** area, change the status to the desired state by selecting **Pending**, **Passed**, or **Failed**.

Manually Setting Test Item Status

You can manually set the status of an individual Test Item to Passed, Failed, or Pending.

Task: ***To manually set the status of an individual Test Item:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs
   - Services
   - Shortcuts
   - Type Libraries
   - ODBC Data Sources
   - ODBC Drivers
3. Right-click on the **Test Item**, point to **Set Status,** and select the status from the Set Status submenu: **All**, **Pending**, **Passed**, or **Failed**.

Filtering Test Case Data

On the **Product Information View**, you can choose to display only those Test Cases with a selected status: **Pending**, **Passed**, or **Failed**.

Task: ***To filter the displayed data in a Test Case:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens and a view of **Test Items** is displayed.
2. Under **Deployment Tests**, select one of the following tests to open its corresponding View:
   - Class IDs
   - File Associations
   - Help Files
   - Prog IDs

- Services
- Shortcuts
- Type Libraries
- ODBC Data Sources
- ODBC Drivers

3. From the **View these test items** list, select the filter you want to apply to the data: **All**, **Passed**, **Failed**, or **Pending**. The View is automatically updated based on the selected filter.

Deployment Testing

Deployment tests are performed against the installed product, ensuring that the product has been installed correctly and all key functionality works in the installed environment. Test Cases in this area are primarily designed to identify whether the application fails to work properly due to permission settings on the registry or individual files.

Some of the primary areas checked are:

Table 19-2 • Areas Checked During Deployment Testing

Area	Description
COM Data	Ensure all COM objects can be instantiated programatically. This includes Class IDs, Prog IDs, and Type Libraries. COM data is tested silently, returning results in the Test Case Progress area and the queue. See Checking Class IDs. Checking Prog IDs, or Checking Type Libraries.
File Associations	Ensure all file extensions have been installed and associated correctly. This involves launching a file with this extension, and determining if the correct application was used. See Checking File Associations.
Help Files	Ensure help files are installed and can be launched correctly. See Checking Help Files.
Shortcuts	Ensure each shortcut is installed and if it successfully launches the shortcut target. See Checking Shortcuts.
Type Libraries	Determines if the Type Libraries COM objects can be instantiated programatically. See Checking Type Libraries.
Manifests	Tests the manifests and assemblies used to isolate a Windows Installer package. See Checking Manifests.
ODBC Data Sources	Verify ODBC data sources. See Checking ODBC Data Sources.
ODBC Drivers	Verify ODBC drivers. See Checking ODBC Drivers

Table 19-2 • Areas Checked During Deployment Testing (cont.)

Area	Description
Services	Ensure all NT Services have been installed correctly. This is done by opening the Services Manager to determine if the Service exists on the target machine. See Checking Services.

Automatically Running All Deployment Tests Silently

You can choose to run all deployment tests silently (without prompting for user input) using either the Interface or the command line.

From the Interface

You can choose to run all deployment tests silently (without prompting for user input) by making a selection in the QualityMonitor interface.

Task: ***To run all deployment tests silently from the Interface, do one of the following:***

1. On the QualityMonitor **Product Information View**, select the **Deployment Tests** root node and then do one of the following:
   - Click the **Execute All Deployment Tests** button.
   - Select A**ll Deployment Tests** from the **Execute** menu.
   - Click the **Execute All Deployment Tests** toolbar button:

   When you select one of these options, a dialog box with a progress bar and an option to cancel will be displayed.

From the Command Line

You can also run all deployment tests silently by entering a command in the command line. See Running QualityMonitor from the Command Line for more information.

Checking Class IDs

The **Class ID** Deployment Test is run to determine if the Class ID COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

Task: ***To check Class ID functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Class IDs**. The **Class IDs View** opens.

3. Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are listed in the **Test Case Progress** area.

***Note** • When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box select the Test Item and then select **Test Item Information** from the context menu.*

Checking File Associations

The **File Associations** Deployment Test is run to determine if all file extensions have been installed and associated correctly. This involves launching a file with this extension, and determining if the correct application was used.

Task: ***To test file associations:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **File Associations** from the View List. The **File Associations View** opens.
3. Right-click on the Test Item you want to run and select **Run**.You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When the **Test Progress** dialog box opens, click **Run** to exercise the file association.
5. From the resulting **Open** dialog box, browse to a file with the appropriate extension and click **Open**. The file is launched with its associated application. Following the application launch, the **Test Result** dialog box appears.
6. Click **Yes** or **No** depending on whether the file launched with the expected program. You can also enter comments in the **Comment** field on this dialog box.

Checking Help Files

The **Help Files** Deployment Test is run to determine if the help files are installed and can be launched correctly.

Task: ***To test help file functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Help Files** from the View List. The **Help Files View** opens.

3. Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When the **Test Progress** dialog box opens, click **Run** to launch the help file. Following an attempt to launch the shortcut, the **Test Result** dialog box appears.
5. Click **Yes** or **No** depending on whether the help file launched correctly. You can also enter comments in the **Comment** field on this dialog box.

Checking Prog IDs

The **Prog IDs** Deployment Test is run to determine if the Prog ID COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

Task: ***To check Prog ID functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Prog IDs** from the View List. The **Prog IDs View** opens.
3. Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box select the Test Item and then select* ***Test Item Information*** *from the context menu.*

Checking Services

The **Services** Deployment Test is run to determine if all NT Services have been installed correctly. This is done by opening the Services Manager to determine if the Service exists on the target machine.

Task: ***To check Service functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Services** from the View List. The **Services View** opens.
3. Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box select the Test Item and then select* ***Test Item Information*** *from the context menu.*

Checking Shortcuts

The **Shortcuts** Deployment Test is run to determine if each shortcut is installed and if it successfully launches the shortcut target.

Task: ***To check shortcuts:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Shortcuts** from the View List. The **Shortcuts View** opens.
3. Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When the **Test Progress** dialog box opens, click **Run** to launch the shortcut. The **Test Result** dialog box opens.
5. Following an attempt to launch the shortcut, the **Test Result** dialog box opens. Click **Yes** or **No** depending on whether the shortcut launched correctly. You can also enter comments in the **Comment** field on this dialog box.

Checking Type Libraries

The **Type Libraries** Deployment Test is run to determine if the Type Libraries COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

Task: ***To check type library functionality:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Type Libraries** from the View List. The **Type Libraries View** opens.
3. Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box select the Test Item and then select* ***Test Item Information*** *from the context menu.*

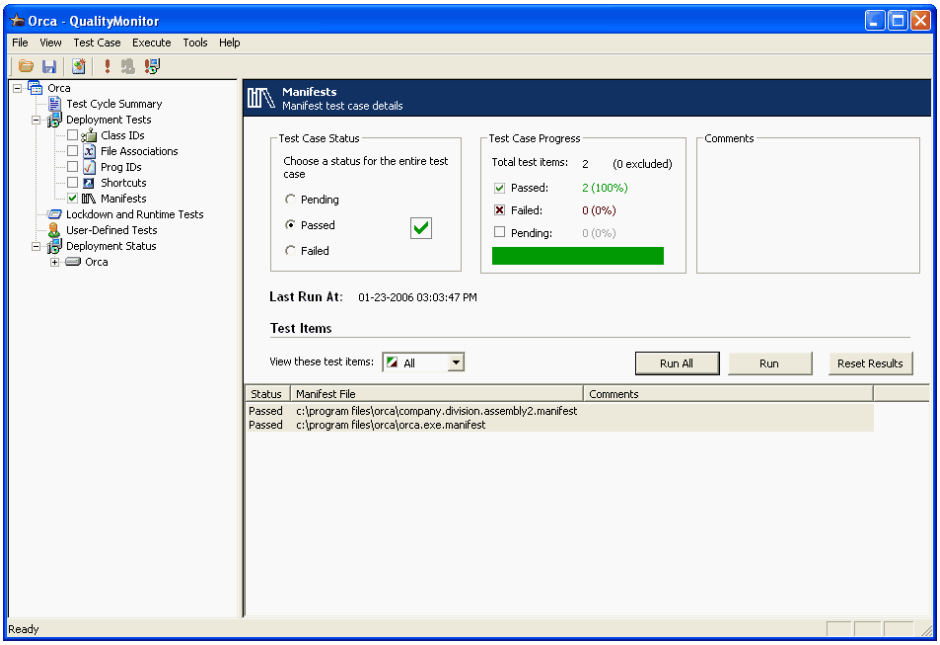
Checking Manifests

The **Manifests** Deployment Test is run to test the manifests and assemblies used to isolate a Windows Installer package.

The Manifests Deployment Test tests information from the `MsiAssembly` and `MsiAssemblyName` tables. QualityMonitor reads through the manifest/assembly files and performs the baseline Class IDs, Prog IDs, or Type Libraries testing for each entry in the files.

Task: ***To check shortcuts:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **Manifests** from the View List. The **Manifests View** opens.



3. Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.

   When testing is finished, results are recorded in the **Test Case Progress** area. Also, the **Status** of each test item (**Passed**, **Failed**, or **Pending**) is listed next to the **Manifest File** name.

***Note** • When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box select the Test Item and then select **Test Item Information** from the context menu.*

4. If desired, you can also enter comments in the **Comment** field on this dialog box.

Checking ODBC Data Sources

The **ODBC Data Sources** Deployment Test is run to verify ODBC data sources.

Task: ***To check ODBC data sources:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **ODBC Data Sources** from the View List. A list of data sources in that application appears in the lower portion of the **ODBC Data Sources View**. Only those data sources that belong to the current logged-in user are listed on the **ODBC Data Sources View**.
3. Select the Test Item you want to run and click **Run**. You can also select multiple Test Items to run, or click **Run All** to run all available Test Items.

   For certain ODBC data sources, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to take more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

***Note** • When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box select the Test Item and then select **Test Item Information** from the context menu.*

Checking ODBC Drivers

The **ODBC Drivers** Deployment Test is run to verify ODBC Drivers.

Task: ***To check ODBC drivers:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. Expand the **Deployment Tests** node and select **ODBC Drivers** from the View List. A list of drivers in that application appears in the lower portion of the **ODBC Drivers View**. Only those drivers that belong to the current logged-in user are listed.
3. Select the Test Item you want to run and click **Run**. You can also select multiple Test Items to run, or click Run All to run all available Test Items.

   For certain ODBC drivers, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to take more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.
4. When testing is finished, results are recorded in the **Test Case Progress** area.

Note • *When a Test Item is failed, you can view details about it, including the error message associated with it on the* ***Test Item Information*** *dialog box. To access this dialog box select the Test Item and then select* ***Test Item Information*** *from the context menu.*

Specifying Exclusions for Deployment Testing

When a deployment test is run on a package, some of the tests related to Class IDs, Prog IDs, and Type Library IDs fail because they refer to components which belong to the operating system rather than the software which is being tested. These errors have no impact on the integrity of the software being tested, and cause confusion among some users testing the software. Users need to be able to prevent error messages caused by files that are not affecting the performance of the software package to be listed in the test results.

To prevent these operating systems errors from being reported, you can specify a list of files to be excluded when any of the Class ID, Prog ID, or Type Library ID Deployment tests are run. You can maintain a different list for each of these three Deployment Tests.

Note • *After a Deployment Test has been run, the test results are listed in the Class IDs View, Prog IDs View, or Type Libraries View. The items included in the exclusion lists are not shown in these views, but are still stored in the QualityMonitor Project File (*`.iqm`*). When this project file is opened again in QualityMonitor, the results are checked against the exclusion list before being displayed in the Class IDs View, Prog IDs View, or Type Libraries View.*

To Add a File to the Exclusion List

On the **Exclusions** tab of the Options Dialog Box, you can manage all three exclusion lists. On the **Exclusions** tab, you can view the exclusion lists, and can add or remove entries from a list.

Using the Options Dialog Box

Task: ***To exclude Class IDs, Prog IDs, or Type Library IDs using the Options dialog box:***

1. From the QualityMonitor interface, select **Options** on the **Tools** menu. The **Options** dialog box opens.
2. Click the **Exclusions** tab. On the **Exclusions** tab, excluded items are listed for the selected **Exclusions list**.
3. From the **Exclusions list**, select the Deployment Test that you want to modify the exclusion list for: **Class ID**, **Prog ID**, or **Type Library**.
4. Click **Add**. The **Add Exclusions** dialog box opens.
5. Next to the **File Name** box, click **Browse** and select the **Application (.exe)**, **Application Extension (.dll)**, **Type Library (.tlb)**, or **ActiveX object (.oxc)** file that contains Class IDs, Prog IDs, or Type Library IDs that you want to exclude from the selected Deployment Test.

   The Class IDs, Prog IDs, or Type Library IDs that are associated with the selected file are listed, displaying the **Identifier** and a **Description** of each.
6. Select the Class IDs, Prog IDs, or Type Library IDs that you want to exclude from the Deployment Test and click **OK**.

Directly From the Results Window

After a Deployment Test has been run and the test results are listed in the Class IDs View, Prog IDs View, or Type Libraries View, you can add an item to the exclusion list directly by right-clicking on the item you want to exclude and choosing **Add to exclusion** from the context menu.

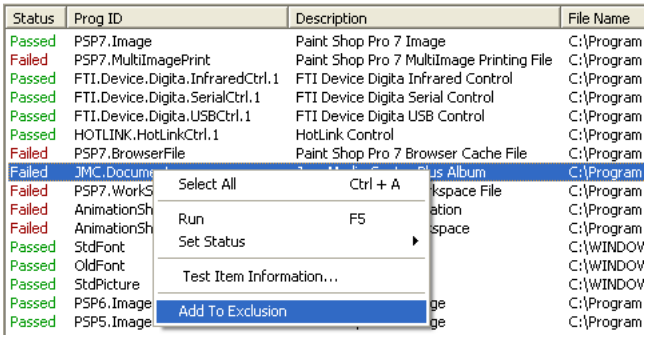


Figure 19-1: Adding a Prog ID to the Prog ID Exclusion List from the Test Results List

Deleting a Item from an Exclusion List

To delete an item from an exclusion list, perform the following steps.

Task: ***To delete a Class IDs, Prog IDs, or Type Library IDs from an exclusion list:***

1. From the QualityMonitor interface, select **Options** on the **Tools** menu. The **Options** dialog box opens.
2. Click the **Exclusions** tab. On the **Exclusions** tab, excluded items are listed for the selected **Exclusions list**.
3. From the **Exclusions list**, select the Deployment Test that you want to modify the exclusion list for: **Class ID**, **Prog ID**, or **Type Library**.
4. Select the item that you want to delete from the list and click **Remove**. The item is removed from the list.

Selecting the Default Exclusion List

On the General Tab of the **Options** dialog box, you can select the default **Exclusion file** to use to filter the test results in the **Lockdown and Runtime Tests** views. By selecting an exclusion file from a shared location, multiple people can use the same error exclusion settings.

Lockdown and Runtime Testing

Lockdown and runtime tests are available through the Lockdown and Runtime Tests View. You are provided with a list of available shortcuts in the package and all of the executables in the package. You can then launch via the shortcut or executable, and exercise functionality in the application. When you close the application, information about the executable is listed under the **Runtime Checks** node. This information, grouped into **Files**, **Registry Entries**, and **Folders** views, allows you to see failures in the application execution. These are potential issues with the application, and may or may not have any affect on the overall package integrity.

If you want to execute tests in the context of a different user (under a different user account), click **Run As** instead of **Run** to execute the test. You would then be prompted to enter a **User Name** and **Password**. For more information, see Performing Lockdown and Runtime Tests Under a Different User Account.

This section includes the following topics:

- Performing Lockdown and Runtime Tests
- Performing Lockdown and Runtime Tests Under a Different User Account
- Running Lockdown and Runtime Tests in Restricted Environments
- Performing Isolation Tests
- Filtering Results of Lockdown and Runtime Tests

Note • *Lockdown and runtime tests are only supported on Windows NT4, 2000, and XP.*

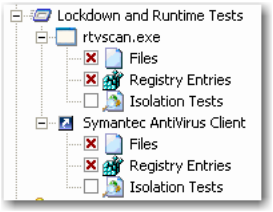
Performing Lockdown and Runtime Tests

Lockdown and Runtime Tests allow you to see failures in the application execution. These are potential issues with the application, and may or may not have any affect on the overall package integrity.

Task: ***To perform lockdown and runtime tests:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. From the View List, select **Lockdown and Runtime Tests**. The **Lockdown and Runtime Tests View** opens.
3. Select either the **Select a Shortcut** or **Select an Executable** option.
4. Select the shortcut or executable to run.
5. Click **Run**.
6. When the application launches, use the application in a normal way, performing various operations.
7. Exit the application.

   The name of the executable or shortcut is now listed as a new node under the **Lockdown and Runtime Tests** node in the View List, and a subnode is listed for any access failures for **Files**, **Folders**, or **Registry Entries**.



8. Select the **Files**, **Folders**, and **Registry Entries** nodes. The right side of the **Lockdown and Runtime Tests View** displays a list of failed **Test Items** for each node. For information on specifying which errors are listed, see Filtering Results of Lockdown and Runtime Tests.
9. To view the error message for a **Test Item**, right-click the Test Item and select **Test Item Information** on the context menu. The Test Item Information dialog box opens, listing an **Error Description** in the **Test Details** area.

Performing Lockdown and Runtime Tests Under a Different User Account

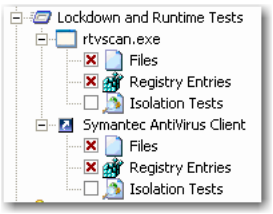
You can use the **Run As** feature to execute Lockdown and Runtime tests in the context of a different user. This allows you to validate an application in a locked-down environment without actually requiring a user to log-in with a different set of credentials. This will reduce the test cycle effort significantly.

Task: ***To perform tests under a different user account:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. From the View List, select **Lockdown and Runtime Tests**. The **Lockdown and Runtime Tests View** opens.
3. Select either the **Select a Shortcut** or **Select an Executable** option.
4. Select the shortcut or executable to run.
5. Click **Run As**.

   You are then prompted to enter a **User Name** and **Password**.
6. Enter the User Name in the format of: `DOMAINNAME\UserName`. The default value is the current **User Name**.
7. When the application launches, use the application in a normal way, performing various operations.
8. Exit the application.

   The name of the executable or shortcut is now listed as a new node under the **Lockdown and Runtime Tests** node in the View List, and a subnode is listed for any access failures for **Files**, **Folders**, or **Registry Entries**.



9. Select the **Files**, **Folders**, and **Registry Entries** nodes. The right side of the **Lockdown and Runtime Tests View** displays a list of failed **Test Items** for each node. For information on specifying which errors are listed, see Filtering Results of Lockdown and Runtime Tests.
10. To view the error message for a **Test Item**, right-click the Test Item and select **Test Item Information** on the context menu. The Test Item Information dialog box opens, listing an **Error Description** in the **Test Details** area.

Note • ***Run As*** *can also be selected using the Shift+F5 shortcut, or by selecting* ***Run As*** *from the* ***Execute*** *menu.*

Running Lockdown and Runtime Tests in Restricted Environments

When executing runtime tests in a locked-down environment, you may encounter an error such as:

```
Unable to monitor the application execution.
```

QualityMonitor needs to run under a user with Admin privileges when executing Lockdown and Runtime Tests.

To emulate lockdown environment under a restricted user, click **Run As** instead of **Run** to execute the test, and provide the **User Name** and **Password** of a locked down user. For more information, see Performing Lockdown and Runtime Tests Under a Different User Account.

Performing Isolation Tests

You can run Isolation Tests to display the location of all portable executable (PE) files (`dll/ocx/exe/tlb/olb`) that are launched from a process while performing a Lockdown and Runtime test. Viewing a listing of these portable executable file names and paths makes it easier for you to ensure that the application is fully isolated.

After you perform a Lockdown and Runtime Test for an executable (`.exe`) or a shortcut on the Lockdown and Runtime Tests View, an additional node called **Isolation Tests** is added to the tree under the executable or shortcut node.

When you select this **Isolation Tests** node, the filenames of the portable executable files and their paths are listed. By default, the status of all these items is **Pending**. To ensure that all of the executables or shortcuts in this test case are isolated, go to the **Test Case Status** area of the view, and set the status of the entire test case to either **Pending**, **Passed**, or **Failed**.

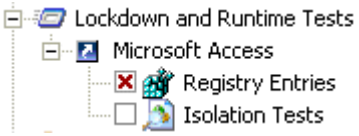
Note • *Note the following regarding Isolation testing:*

- *The Isolation Tests node will be added to the Lockdown and Runtime Tests tree only if the selected executable launches at least one portable executable file.*
- *You can specify when you would like the **Isolation Tests** node to appear in the Lockdown and Runtime Tests tree by selecting an option from the **Show Isolation Tests** list on the **General** tab of the Options Dialog Box.*
- *QualityMonitor does not support isolation testing under Windows 2000.*

Task: ***To perform isolation tests:***

1. Create or open a QualityMonitor project. The QualityMonitor **Product Information View** opens.
2. From the View List, select **Lockdown and Runtime Tests**. The **Lockdown and Runtime Tests View** opens.
3. Select either the **Select a Shortcut** or **Select an Executable** option.
4. Select the shortcut or executable to run.
5. Click **Run**.

   After a Lockdown/Runtime test is performed for an executable (`.exe`) or a shortcut, an additional node called **Isolation Tests** is added to the tree under the executable or shortcut node.



6. Select the **Isolation Tests** node. The filenames of the portable executable files and their paths is displayed in the list control of the view. By default, the status of all of the items in this test case is **Pending**.
7. To ensure that all of the executables or shortcuts in this test case are isolated, go to the **Test Case Status** area of the view, and set the status of the entire test case to either **Pending**, **Passed**, or **Failed**.

Filtering Results of Lockdown and Runtime Tests

On the Lockdown and Runtime Tests Files View, Folders View, Registry Entries View, and Isolation Tests View you can choose to filter the results that are listed.

Task: ***To filter test results:***

1. To filter the list by Test Item status, select an option from the **View these test items** list: **Passed**, **Failed**, **Pending**, or **All**.
2. To select errors to exclude from future Lockdown and Runtime result listings, perform the following steps:
   a. Click the **Set Filter** button. The Runtime Test Filters Dialog Box opens, listing all errors that were generated during this test.
   b. Select those errors that you want to exclude from future Lockdown and Runtime tests.

      These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.
3. To filter the list by one type of error that was generated, make a selection from the **Having these errors** list. This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). To see all the errors, select **Show All**.

Using MSI Doctor to Verify Package Deployment Status

You can use QualityMonitor's MSI Doctor to verify if an MSI package is installed properly. This helps prevent users from seeing an auto-repair dialog box when they run the application. Auto repair messages are displayed by applications to attempt to reinstall missing/corrupted components.

By examining an application using QualityMonitor MSI Doctor, you can quickly identify any problems by checking the status of all products and features. Using MSI Doctor, you can:

- See the status of all components

- Verify if any files are missing or if any files do not match the version or size specified in the MSI file
- See the components status segregated by features
- Configure or reinstall features
- Reinstall components

To use MSI Doctor, select the **Deployment Status** node ( ) from the QualityMonitor View List to access the Deployment Status View. Under the **Deployment Status** node, a tree view of the application's features are components are listed:

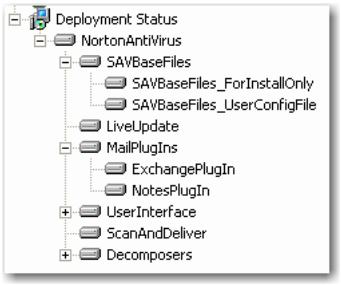


Figure 19-2: Deployment Status View: Features and Components

The following icons are displayed in the tree view and in the component list and indicate the feature or component's status:

Table 19-3 • Deployment Status View Icons

Icon	Name	Description
	Installed	Feature or Component is installed on the local machine.
	Uninstalled	Feature or Component is not installed on the local machine.
	Broken	Component is broken (a key file in the Component is missing) or Feature contains a broken Component.
	Run From Source	Feature or Component is configured to run from a source location (rather than being installed on the local machine).
	On Demand	Feature is configured to be installed when needed. Not applicable to Components.

Context Menu Functionality

The following table lists the functions available on the context menus for the **Deployment Status** icon, **Features**, and **Components**, and the dialog boxes that appear when those functions are selected:

Table 19-4 • Dialog Boxes Invoked from Deployment Status View Context Menu

Context Menu	Deployment Status Node	Feature Node	Component Node
Configure	Install or Configure Product Dialog Box	Install or Configure Feature Dialog Box	N/A
Re-install	Re-install Product/Feature Dialog Box	Re-install Product/Feature Dialog Box	Installation program is launched. (This option is only enabled if the selected Component is broken.)
Properties	Product Properties Dialog Box	Feature Properties Dialog Box	Component Properties Dialog Box

Using MSI Doctor, you can perform the following tasks:

- View Product, Feature, or Component Deployment Status Properties
- Verify Product, Feature, or Component Data
- Install or Configure Products or Features
- Reinstall Features

View Product, Feature, or Component Deployment Status Properties

To view the deployment status properties of a product, feature, or component, perform the following steps.

Task: ***To view product, feature, or component Deployment Status properties:***

1. Launch QualityMonitor and open the package that you want to view the deployment status properties of. The QualityMonitor **Product Information View** opens.
2. Select one of the following:
   - **To view Product properties**—Select the **Deployment Status** icon from the View List.
   - **To view Feature properties**—Select a **Feature** icon under the **Deployment Status** icon on the View List.
   - **To view Component properties**—Select either the **Deployment Status** icon or a **Feature** icon from the View list, and then select a **Component** from the list on the right.
3. Select **Properties** from the context menu. The **Product Properties**, **Feature Properties**, or **Component Properties** dialog box appears, displaying property information for the selected Product, Feature, or Component.
4. On the Properties dialog boxes, you can also click the **Verify Data** button to verify if the files and registry information for the selected item are installed properly. See Verify Product, Feature, or Component Data for more information.
5. Click **OK** to exit the **Properties** dialog box.

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Verify Product, Feature, or Component Data

You can verify if the files and registry information for a Product, Feature, or Component are installed properly. Verification errors are displayed on the Installed Data Dialog Box.

Task: ***To verify product, feature, or component data:***

1. Launch QualityMonitor and open the package that you want to verify the files of. The QualityMonitor **Product Information View** opens.
2. Select one of the following:

- **To verify all of the files in all of the Features in the Product**, select **Deployment Status** from the View List.
- **To verify only the files in the selected Feature**, select a **Feature** icon under **Deployment Status** on the View List.
- **To verify only the files in a specific Component of a Feature**, select either the **Deployment Status** icon in the View List or a **Feature** icon under it, and then select a component from the list on the right.

3. Select **Properties** from the context menu. The **Product Properties**, **Feature Properties**, or **Component Properties** dialog box appears, displaying property information for the Product, Feature or Component.
4. Click **Verify Data**. When you click **Verify Data**, QualityMonitor checks all of the files and registry entries included in the Product, Feature, or Component and then displays them on the **Files** and **Registry** tabs of the **Installed Data** dialog box.
   - On the **Files** tab, the following icons are used to identify verification errors:

Icon	Description
	File is missing.
	File has a different version or size than that specified in the Windows Installer package

   - On the **Registry** tab, all registry entries for the selected item are listed. Registry data is verified by checking the existence of the registry key and the value name (if one exists). (The value data is not checked.). The icon is used to indicate that a registry key or value name is incorrect or missing:
5. Click **Close** to exit the **Installed Data** dialog box, and click **OK** to exit the **Properties** dialog box.

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Install or Configure Products or Features

You can use **Configure** to install a product or feature that is not currently installed.

Task: ***To install or configure products or features:***

1. Launch QualityMonitor and open the package that you want to install or configure. The QualityMonitor **Product Information View** opens.
2. Select one of the following:
   - **To install or configure the entire Product**, select **Deployment Status** from the View List.

- **To install or configure only the selected Feature**, select a **Feature** icon under **Deployment Status** on the View List.

3. Select **Configure** from the context menu. The **Install or Configure Product** or **Install or Configure Feature** dialog box appears, prompting you to select the installation location and the installation type (on the **Install or Configure Product** dialog box only).
4. Select one of the following options to specify installation location:
   - **Default**—Files will be installed to their default location.
   - **Local**—Files will be installed on the local machine.
   - **Source**—Files will be run from the installation source.
   - **On Demand**—Files will be installed when needed.
5. (Product only) Select one of the following options to specify installation type:
   - **Minimum**—Only the essential features will be installed.
   - **Typical**—Most commonly used features will be installed.
   - **Complete**—All of the program's features will be installed.
6. Click **OK**. The Product or Feature is installed, per the options you specified.

Reinstall Features

When a Feature is broken (identified by the icon), you can fix it by re-installing the entire Product or just re-installing the broken Feature.

Task: ***To reinstall features:***

1. Launch QualityMonitor and open the package that you want to reinstall. The QualityMonitor **Product Information View** opens.
2. Select one of the following:
   - **To reinstall the entire Product**, select **Deployment Status** from the View List.
   - **To reinstall only the selected Feature**, select a **Feature** icon under **Deployment Status** on the View List.
3. Select **Re-install** from the context menu. The **Re-install Product/Feature** dialog box appears, prompting you to select a reinstall mode.
4. Select one of the following reinstall modes:
   - Repair all detected reinstall problems
   - Reinstall only if file is missing
   - Force all files to be reinstalled

- Reinstall if file is missing, or an older version exists
- Reinstall if file is missing, or an older or equal version exists
- Reinstall if existing file has different version
- Verify that required user registry entries are present
- Verify that required local machine registry entries are present
- Recreate all shortcuts

5. Click **OK**. The Product or Feature is re-installed, per the option you specified.

Reinstall Components

When a Component is broken (identified by the icon), you can fix it by re-installing the broken Component.

Task: ***To reinstall components:***

1. Launch QualityMonitor and open a package. The QualityMonitor **Product Information View** opens.
2. Select **Deployment Status** from the View List or select a **Feature** icon under **Deployment Status** on the View List. All components associated with the selected feature(s) are listed.
3. Select the component that you want to re-install and select **Re-install** from the context menu. The component is automatically reinstalled.

Creating Custom Test Cases

QualityMonitor supports adding additional, custom Test Cases to projects—based on your business needs.

You can define a template that includes information that is re-used when defining Test Cases. Then, when creating a new user-defined Test Case, you can load this template file and have the components of the new Test Case pre-populated with the information saved in the template. A single template file can load multiple user defined Test Cases.

Adding User-Defined Test Cases

Task: ***To add a user-defined case:***

1. Launch QualityMonitor and open the package that you want to create a custom Test Case for. The QualityMonitor **Product Information View** opens.
2. Right-click on the **User-Defined Tests** node and select **Add Test Case** from the context menu.

   A new Test Case appears below the **User Defined Tests** node, and you are prompted to enter a name.

3. Name the Test Case appropriately.
4. Select the new Test Case to open the **Test Case View.**
5. Under **Test Case Status**, specify the status for this Test Case: **Pending**, **Passed**, or **Failed**.
6. Click **Browse** and select an executable to associate with this Test Case, if necessary.
7. In the **Instructions** text box, enter any necessary comments.

   For example, you may want to create a custom Test Case to ensure that a specific database is updated properly after running an application.
8. Your entries are automatically saved in the new Test Case.

Creating and Using Test Case Templates

Task: ***To create and use Test Case templates:***

1. Create a new Test Case as described in Adding User-Defined Test Cases.
2. To save the Test Case as a template to re-use when creating new Test Cases, perform the following steps:
   a. Select this Test Case node in the **User Defined Tests** tree and select **Save as template** from the context menu.
   b. Specify a name and a location for the Test Case template and select **Save**. The template is saved in `.xml` format in the location you specify.
3. To add this Test Case to an existing template, perform the following steps:
   a. Select this Test Case node in the **User Defined Tests** tree and select **Add to template** from the context menu. You are prompted to select the template file that you want to add this Test Case to.
   b. Select the template that you want to add this Test Case to and select **Save**.
4. To create a new Test Case based upon a template, perform the following steps:
   a. Select the **User Defined Tests** node and select **Load template** from the context menu. You are prompted to select the template that you want to use.
   b. Select a template and click **Open**. All of the Test Cases that were saved in the template are now listed in the **User Defined Tests** tree.

You can specify that you want a specific template file automatically loaded each time a QualityMonitor project is opened. To do this, select a **Template file** on the General Tab of the **Options** dialog box, and also select the **Load Templates on Project Open** option.

Renaming User-Defined Test Cases

Task: ***To rename a user-defined Test Case:***

1. Under the **User-Defined Tests** node, select the Test Case that you want to rename, and select **Rename** from the context menu.
2. Provide a new name for the Test Case.

Test Reports

QualityMonitor allows you to create an HTML test report for the current project. This can be done by selecting **Generate Report** from the **File** menu and providing the name and location for the report. The report will then automatically open in your default browser.

Note • *If an error message occurs when generating the report, it may be because msxml4.dll or isqm.xlst is not in the same directory as the QualityMonitor executable (isqm.exe). These files must be present to create the report.*

Running QualityMonitor from the Command Line

QualityMonitor can be run from the command line by using `isqm.exe`. It can accept the following parameters:

Table 19-5 • QualityMonitor Command Line Parameters

Syntax	Options	Description

-c <MSI Product Code>		Launch QualityMonitor by opening the MSI product specified by the product code. For example:
		`isqm.exe -c {BDC62375-07B4-4CBD-9991-4C25C24F3071}`
		`isqm.exe -c {BDC62375-07B4-4CBD-9991-4C25C24F3071} -sb -f <c:\projectfile.iqm>`

	-sn	Run QualityMonitor silently without any user interaction and no progress display.
	-sb	Run QualityMonitor silently with a progress display. QualityMonitor displays the test names as they are executed and provides an option for the user to cancel.
	-r <Report File>	Generates a report file `<c:\report.htm>` with the test results. Works only when using either `-sn` or `-sb`.
	-f <Project File>	Save test results in this file. This is necessary when using either `-sn` or `-sb`. If this file does not exist, it will be created and then results will be saved.

-f <Project File>		Launch QualityMonitor with this Project File. For example:
		`isqm.exe -f c:\mydocuments\mytesting\tesresults.iqm`
		`isqm.exe -f c:\mydocuments\mytesting\tesresults.iqm -sn`

	-sn	Run QualityMonitor silently without any user interaction and no progress display.
	-sb	Run QualityMonitor silently with a progress display. QualityMonitor displays the test names as they are executed and provides an option for the user to cancel.
	-r <Report File>	Generates a report file `<c:\report.htm>` with the test results. Works only when using either `-sn` or `-sb`.
-h or -?		Help

Note • *When using any -sn or -sb command line options, you can specify the target product using the existing options /c or /f.*

- *If you use the /f option to specify the product, the input file will be modified with the test results.*
- *If you use /c to specify the target product, /f options must be used to specify the project file path which will have the test results.*
- *If both /c and /f parameters are specified, then QualityMonitor gives preference to /c and operates with the product code specified by /c.*

QualityMonitor Reference

Topics contained in this section provide detailed reference on each user interface element, dialog box, or view in QualityMonitor. This is the same documentation displayed when you click F1 from the QualityMonitor interface. Topics are organized as follows:

- Menus and Toolbar
- QualityMonitor Interface
- Dialog Boxes
- Views

Menus and Toolbar

The following table provides a description of each of QualityMonitor's menu commands and toolbar buttons:

Table 19-6 • QualityMonitor Menus and Toolbar Options

Menu	Command	Toolbar Button	Keyboard Shortcut	Description
File	**Open**		Ctrl+O	Allows you to open an existing QualityMonitor project file (.iqm) or create a new one based on an installed MSI-based application.
File	**Close**			Closes the current project.
File	**Save**		Ctrl+S	Saves the current project.
File	**Save As**			Saves the current project using the name and location you specify.

Table 19-6 • QualityMonitor Menus and Toolbar Options (cont.)

Menu	Command	Toolbar Button	Keyboard Shortcut	Description
File	**Generate Report**			Creates an HTML test report for the current project. The report will automatically open in your default browser.
File	**1,2,3,4**			Allows you to open the four most recently accessed QualityMonitor projects.
File	**Exit**			Exits QualityMonitor.
View	**Toolbar**			Toggles display of the toolbar.
View	**Status Bar**			Toggles display of the status bar.
Test Case	**Add Test Case**			Adds a custom Test Case beneath the Additional Tests view in the View List.
Execute	**All Deployment Tests**		Ctrl+F5 or Alt+E+D	Runs all the deployment tests in the current project.
Execute	**Run**		F5 or Alt+E+R	Runs the selected deployment, lockdown and runtime, or user defined test.
Execute	**Run As**		Shift+F5	Runs the selected lockdown and runtime test in the context of a different user. You are then prompted to enter a User Name and Password.
Tools	**Options**			Displays the Options dialog box.
Help	**Contents**			Launches the Help Library, displaying the Contents tab.
Help	**Index**			Launches the Help Library, displaying the Index tab.
Help	**Search**			Launches the Help Library, displaying the Search tab.
Help	**Support Central**			Accesses AdminStudio Support Central on the Web.
Help	**Web Community**			Accesses the AdminStudio Community on the InstallShield Web site.
Help	**ReadMe**			Displays the AdminStudio ReadMe file.

Table 19-6 • QualityMonitor Menus and Toolbar Options (cont.)

Menu	Command	Toolbar Button	Keyboard Shortcut	Description
Help	**Feedback**			Accesses the AdminStudio feedback form on the InstallShield Web site.
Help	**AdminStudio on the Web**			Accesses the AdminStudio Web site.
Help	**About QualityMonitor**			Displays the About dialog box.

QualityMonitor Interface

The QualityMonitor interface is divided into two main areas. The View List, which appears at the left side of the screen, provides a visual representation of the QualityMonitor project file. It provides easy access to individual views and Test Cases. The right side of the interface changes depending on the view or Test Case selected.

Dialog Boxes

The following dialog boxes are accessible from within QualityMonitor:

- About QualityMonitor Dialog Box
- Add Exclusions Dialog Box
- Component Properties Dialog Box
- Feature Properties Dialog Box
- Install or Configure Feature Dialog Box
- Install or Configure Product Dialog Box
- Installed Data Dialog Box
- Open QualityMonitor Project Dialog Box
- Options Dialog Box
- Product Properties Dialog Box
- Re-install Product/Feature Dialog Box
- Test Item Information Dialog Box
- Test Progress Dialog Box
- Test Result Dialog Box

About QualityMonitor Dialog Box

The About InstallShield QualityMonitor dialog box, available by selecting About from the Help menu, displays information about QualityMonitor and AdminStudio, including version and serial number information.

Add Exclusions Dialog Box

On the Add Exclusions dialog box, which is opened by clicking **Add** on the Exclusions Tab of the Options dialog box, you can select a Class ID, Prog ID, or Type Library ID to exclude from a Deployment Test. See Specifying Exclusions for Deployment Testing.

The following options are included:

Table 19-7 • Add Exclusions Dialog Box Options

Options	Description
File Name	Click Browse and select the **Application (.exe)**, **Application Extension (.dll)**, **Type Library (.tlb)**, or **ActiveX object (.oxc)** file that contains Class IDs, Prog IDs, or Type Library IDs that you want to exclude from the selected Deployment Test.
Identifier	Identifies the Class IDs, Prog IDs, or Type Library IDs that are associated with the selected file.
Description	Description of the Class IDs, Prog IDs, or Type Library IDs that are associated with the selected file.

Component Properties Dialog Box

The Component Properties dialog box is displayed when you select a Component in the Component list on the right side of the Deployment Status View and select Properties from the context menu.

The following information is listed:

Table 19-8 • Component Properties Dialog Box Options

Options	Description
Name	Name of the selected Component.
GUID	Number which uniquely identifies this Component.
Status	Status of Component, such as Installed on Local Drive.
Location	Location on server where Component is installed.

Table 19-8 • Component Properties Dialog Box Options (cont.)

Options	Description
Verify Data	Click to verify the following for the selected Component: • **Files**—QualityMonitor verifies the existence of files in the specified location, and compares the file size specified in the Windows Installer .msi package to that of the file on the system. Missing or modified files are identified. • **Registry**—QualityMonitor verifies the Registry data by checking the existence of the registry key and the value name (if one exists). Only the registry keys and value names are verified; the values themselves are not verified. Missing or incorrect registry keys are identified. The Files and Registry information is listed on the Installed Data Dialog Box.

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Feature Properties Dialog Box

The Feature Properties dialog box is displayed when you select a Feature under the **Deployment Status** node and then select **Properties** from the context menu.

This dialog box contains the following options:

Table 19-9 • Feature Properties Dialog Box Options

Options	Description
Name	Name of selected Feature.
Title	Title of selected Feature.
Parent	Parent Feature of this Feature (if one exists).
Description	Description of this Feature.
Last Used	Date this Feature was last used.
Usage Count	Number of times this Feature has been used.

Table 19-9 • Feature Properties Dialog Box Options (cont.)

Options	Description
Verify Data	Click to verify the following for the selected Feature: • **Files**—QualityMonitor verifies the existence of files in the specified location, and compares the file size specified in the Windows Installer .msi package to that of the file on the system. Missing or modified files are identified. • **Registry**—QualityMonitor verifies the Registry data by checking the existence of the registry key and the value name (if one exists). Only the registry keys and value names are verified; the values themselves are not verified. Missing or incorrect registry keys are identified. The Files and Registry information is listed on the Installed Data Dialog Box.

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Install or Configure Feature Dialog Box

The Install or Configure Feature dialog box is displayed when you select a Feature under the Deployment Status node and then select Configure from the context menu.

If you select an option on this dialog box and click OK, QualityMonitor will attempt to install or configure the selected Feature to the settings you specify. Select one of the following options:

- **Default**—Files will be installed to their default location.
- **Local**—Files will be installed on the local machine.
- **Source**—Files will be run from the installation source.
- **On Demand**—Files will be installed when needed.

Note • *To complete this operation, you may need the source from which the selected Feature was installed.*

Install or Configure Product Dialog Box

The **Install or Configure Product** dialog box is displayed when you select the **Deployment Status** node and then select **Configure** from the context menu. If you select an option on this dialog box and click **OK**, QualityMonitor will attempt to install or configure the Product to the settings you specify.

Installation Location

Select one of the following options:

- **Default**—Files will be installed to their default location.
- **Local**—Files will be installed on the local machine.
- **Source**—Files will be run from the installation source.
- **On Demand**—Files will be installed when needed.

Note • *To complete this operation, you may need the source from which the selected feature was installed.*

Installation Type

Select one of the following options:

- **Minimum**—Only the essential Features will be installed.
- **Typical**—Most commonly used Features will be installed.
- **Complete**—All of the program's Features will be installed.

Installed Data Dialog Box

The Installed Data dialog box appears when you are using MSI Doctor to verify package deployment status, and you perform the following steps:

Task: ***To view the Installed Data dialog box:***

1. Go to the Deployment Status View and select the **Deployment Status** node, one of the Features listed under it, or a component listed on the right, and then
2. Select **Properties** from the context menu to display the Product Properties Dialog Box, Feature Properties Dialog Box, or Component Properties Dialog Box, and then
3. Click **Verify Data**. QualityMonitor then checks all of the files and registry entries included in the selected Product, Feature, or Component and then displays them on the **Installed Data** dialog box.

Files Tab

Files are verified by checking their existence in the specified location, and comparing the file size specified in the .Windows Installer package to that of the file on the system. On the **Files** tab, the following icons are used to identify verification errors:

Table 19-10 • Verification Error Icons

Icon	Description
	File is missing.
	File has a different version or size than that specified in the Windows Installer package

If you double-click on an item listed on the **Files** tab, the Windows Explorer opens to the folder containing the selected file.

Registry Tab

The **Registry** tab lists all of the registry keys and value names in the selected Product, Feature, or Component. Registry data is verified by checking the existence of the registry key and the value name (if one exists). The icon is used to indicate that a registry key or value name is incorrect or missing.

Note • *Only the registry keys and value names are verified; the values themselves are not verified.*

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Open QualityMonitor Project Dialog Box

The Open QualityMonitor Project dialog box opens when you select to open a QualityMonitor project from the Welcome page, or when you either select **Open** from the **File** menu or click the **Open** button on the toolbar.

From this dialog box, you can either select to open an existing QualityMonitor file (and subsequently enter or browse to it), or select to create a project file based on an installed MSI-based application.

Options Dialog Box

The Options dialog box, available by selecting **Options** from the **Tools** menu, has options on two tabs: General Tab and Exclusions Tab.

General Tab

The following options are included on the **General** tab:

Table 19-11 • Options Dialog Box—General Tab Options

Options	Description
Update the test case status automatically after executing test items	Select this option if you want to automatically update test case status after executing test items. If you do not use this functionality, the View List will not automatically update after a test item has been executed.
Show Isolation Tests	Use this option to choose when you would like to Show Isolation Tests after performing a Lockdown and Runtime test. You have the following options: • **Always**—Show this view for all the executables run, irrespective of the presence of records in the IsolatedComponent table and in MsiAssembly SXS records. • **Never**—This view will not be shown irrespective of the data. • **Only if the Application is Isolated**—Show this view only if the MSI Package has either IsolatedComponent records or MSIAssembly SXS records. • **Only if the Running Operating System supports Isolation** ***Note*** • *AdminStudio always stores the information in the project but, the UI selection you make here determines whether this view will be populated.*
Template file	Select a User Defined Tests template file. If you also select the **Load Templates on Project Open** open option, all of the Test Cases in the selected **Template file** will be automatically loaded when a QualityMonitor project is open.
Exclusion file	Select an exclusion file to use to filter the test results in the **Lockdown and Runtime** views. By selecting an exclusion file from a shared location, multiple people can use the same error exclusion settings.

Exclusions Tab

When a Deployment Test is run on a package, some of the tests related to Class IDs, Prog IDs, and Type Library IDs fail because they refer to components which belong to the operating system rather than the software which is being tested. These errors have no impact on the integrity of the software being tested, and cause confusion among some users testing the software. Users need to be able to prevent error messages caused by files that are not affecting the performance of the software package to be listed in the test results.

On the **Exclusions** tab, to prevent these operating systems errors from being reported, you can specify a list of files to be excluded when any of the Class ID, Prog ID, or Type Library ID Deployment Tests are run. You can maintain a different list for each of these three Deployment Tests.

***Note** • After a Deployment Test has been run, the test results are listed in the Class IDs View, Prog IDs View, or Type Libraries View. The items included in the exclusion lists are not shown in these views, but are still stored in the QualityMonitor Project File (`.iqm`). When this project file is opened again in QualityMonitor, the results are checked against the exclusion list before being displayed in the Class IDs View, Prog IDs View, or Type Libraries View.*

The following options are included on the **Exclusions** tab:

Table 19-12 • Options Dialog Box—Exclusion Tab Options

Options	Description
Exclusion list	Select the type of Deployment Test that you want to modify the exclusion list for: • Class ID • Prog ID • Type Libraries
Listing	Listing of the Class IDs, Prog IDs, or Type Libraries that you have chosen to exclude from the selected Deployment Test. The following information is included: • **Identifier**—Number identifying the Class ID, Prog ID, or Type Library. • **File Name**—Name of file that contains the Class ID, Prog ID, or Type Library. • **Status**—The exclusion status is either **Active** or **Inactive**. Only **Active** exclusions are excluded from the test results. **Inactive** exclusions are neglected/omitted from the exclusion process. The only way to change the status from **Active** to **Inactive** (or vice versa) is to manually edit the exclusions file. • **Type**—The exclusion type is either **User** or **System**. Exclusions added by the user are of **User** type. Users can delete only **User** type exclusion entries.
Add	Click to open the Add Exclusions Dialog Box, where you can select a Class ID, Prog ID, or Type Library to exclude from Deployment Tests.
Remove	Click to delete the selected Class ID, Prog ID, or Type Library from the exclusion list.

Product Properties Dialog Box

The Product Properties dialog box is displayed when you select the **Deployment Status** node and then select **Properties** from the context menu.

This dialog box contains the following options:

Table 19-13 • Product Properties Dialog Box Options

Options	Description
Version	Product version.
Publisher	Manufacturer of Product.
Product Code	Number which uniquely identifies this Product.
Local	Directory on local machine where this MSI file is located.
Registered to	Registered user of Product.
Product ID Status	Installation status of this Product, such as “The product is installed for the current user.”
Help Link	Main help link for the Product.
Installed on	Date Product was installed.
Installed from	Location where Product was installed from.
Verify Data	Click to verify the following for the selected Product: • **Files**—QualityMonitor verifies the existence of files in the specified location, and compares the file size specified in the Windows Installer .msi package to that of the file on the system. Missing or modified files are identified. • **Registry**—QualityMonitor verifies the Registry data by checking the existence of the registry key and the value name (if one exists). Only the registry keys and value names are verified; the values themselves are not verified. Missing or incorrect registry keys are identified. The Files and Registry information is listed on the Installed Data Dialog Box.

Note • *To save all the deployment status information in the QualityMonitor project file (.iqm), select the* ***Save deployment status information when saving project*** *option on the Options Dialog Box.*

Re-install Product/Feature Dialog Box

The Re-install Product/Feature dialog box is displayed when you select a Feature under the **Deployment Status** node or you select the **Deployment Status** node and then select **Re-install** from the context menu.

If you select an option on this dialog box and click **OK**, QualityMonitor will attempt to reinstall the selected Feature(s) to the settings you specify. Select one option from the **Select Reinstall Mode** or **Additional Reinstall Modes** property:

Table 19-14 • Re-install Product/Feature Dialog Box Options

Option	Description
Select Reinstall Mode	Select one of the following options: • Repair all detected reinstall problems • Reinstall only if file is missing • Force all files to be reinstalled
Additional Reinstall Mode	Select one of the following options: • Reinstall if file is missing, or an older version exists • Reinstall if file is missing, or an older or equal version exists • Reinstall if existing file has different version • Verify that required user registry entries are present • Verify that required local machine registry entries are present • Recreate all shortcuts

Runtime Test Filters Dialog Box

The Runtime Test Filters Dialog box opens when you click the **Set Filters** button on one of the Lockdown and Runtime Test views: Files View, Folders View, Registry Entries View, or Isolation Tests View.

All of the errors that were generated for that Lockdown and Runtime Test Case are listed. If you want to exclude specific errors from future Lockdown and Runtime tests, select those errors and click **OK**.

These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file.

Test Item Information Dialog Box

The Test Item Information dialog box is displayed when you right-click on a Test Item and select **Test Item Information** from the context menu. The following information is included:

Table 19-15 • Test Item Information Dialog Box Options

Option	Description
Test Item	The file name and path of the selected Test Item.
Status	The status of the selected Test Item: **Passed**, **Failed**, or **Pending**.
Comments	Any comments that were entered to document this Test Item.
Test Details	Specific test details that can include the error message associated with the selected Test Item that was generated during testing. These error messages can help you diagnose issues with the package.

Test Progress Dialog Box

The Test Progress dialog box opens when you execute Test Items. Note the following:

- If you are performing Test Cases which have automatic execution (such as Type Libraries, Prog IDs, Services, or Class IDs), this dialog box opens briefly and automatically closes when execution is complete.
- For non-automatic Test Cases (Help Files, File Associations, and Shortcuts), this dialog box opens for each Test Item selected, allowing you to run the test and perform any necessary manual actions.
- Following execution of Test Items, the Test Result Dialog Box appears.

Test Result Dialog Box

The Test Result dialog box opens following execution of each Test Case requiring manual operations (Help Files, File Associations, and Shortcuts).

You can enter **Comments** about the execution of the functionality, and click **Yes** or **No** depending on whether the Test Item passed.

Views

The following views are available in QualityMonitor:

- Welcome to QualityMonitor View
- Product Information View
- Test Cycle Summary View

- Deployment Tests View
  - Class IDs View
  - File Associations View
  - Help Files View
  - Prog IDs View
  - Shortcuts View
  - Type Libraries View
  - ODBC Data Sources View
  - ODBC Drivers View
  - Services View
- Lockdown and Runtime Tests View
  - Runtime Execution Details View
  - Files View
  - Folders View
  - Registry Entries View
  - Isolation Tests View
- User-Defined Tests View
  - Test Case View
- Deployment Status View

Welcome to QualityMonitor View

The Welcome to QualityMonitor View is the view that is displayed before a project is created or opened. From this view you can choose to create a new QualityMonitor project, browse to an existing project, or open the most recently used project.

This view also lists the three major steps involved in using QualityMonitor to ensure package quality:

- **Open project**—Create or open a QualityMonitor project.
- **Run tests**—Run deployment tests, runtime and lockdown tests, or user-defined custom tests.
- **Analyze results**—Analyze the results of the tests.

Product Information View

The Product Information view displays information about the package you are testing in QualityMonitor.

The following data is displayed:

Table 19-16 • Product Information View Options

Option	Description
Application Name	The name of the application.
Author	The person or company who created the application.
Product Code	The package's product code.
Package Code	The package's package code.
Installed On	The date when the package was installed on the system.
Version	The package's version.

Note • *In the View List, this view is titled with the product name.*

Test Cycle Summary View

The Test Cycle Summary view provides statistics on the number of Test Cases and Test Items in the QualityMonitor project, and the ratio of cases and items passed, failed, or pending. If you add additional Test Cases, or perform runtime checking, the number of Test Cases and items will increase.

Deployment Tests View

The Deployment Tests View provides a summary of all Deployment Tests.

Deployment tests help you with up to several Test Cases to run on your Windows Installer-based application. Tests are only available if the application has the corresponding associated data (for example, if there are no shortcuts, you cannot run the Shortcuts Test Case).

QualityMonitor includes the following deployment tests:

- Class IDs View
- File Associations View
- Help Files View
- Prog IDs View

- Shortcuts View
- Type Libraries View
- ODBC Data Sources View
- ODBC Drivers View
- Services View

Automatically Running All Deployment Tests Silently

You can choose to run all deployment tests silently (without prompting for user input) using either the Interface or the command line.

From the Interface

You can choose to run all deployment tests silently (without prompting for user input) by making a selection in the QualityMonitor interface.

Task: ***To run all deployment tests silently from the Interface, do one of the following:***

1. On the QualityMonitor **Product Information View**, select the **Deployment Tests** root node and then do one of the following:
   - Click the **Execute All Deployment Tests** button.
   - Select A**ll Deployment Tests** from the **Execute** menu.
   - Click the **Execute All Deployment Tests** toolbar button:

   When you select one of these options, a dialog box with a progress bar and an option to cancel will be displayed.

From the Command Line

You can also run all deployment tests silently by entering a command in the command line. See Running QualityMonitor from the Command Line for more information.

Class IDs View

The **Class IDs** Deployment Test is run to determine if the Class ID COM objects can be instantiated programatically.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

File Associations View

The **File Associations** Deployment Test is run to determine if all file extensions have been installed and associated correctly. This involves launching a file with this extension, and determining if the correct application was used.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Help Files View

The **Help Files** Deployment Test is run to determine if the help files are installed and can be launched correctly.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Prog IDs View

The **Prog IDs** Deployment Test is run to ensure that the Prog IDs COM objects can be instantiated programatically.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Shortcuts View

The **Shortcuts** Deployment Test is run to determine if each shortcut is installed and if it successfully launches the shortcut target.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Type Libraries View

The **Type Libraries** Deployment Test is run to determine if the Type Libraries COM objects can be instantiated programatically. COM data is tested silently, returning results in the **Test Case Progress** area and the queue.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Manifests View

The **Manifests** Deployment Test is run to test the manifests and assemblies used to isolate a Windows Installer package.

The Manifests Deployment Test tests information from the `MsiAssembly` and `MsiAssemblyName` tables. QualityMonitor reads through the manifest/assembly files and performs the baseline Class IDs, Prog IDs, or Type Libraries testing for each entry in the files.

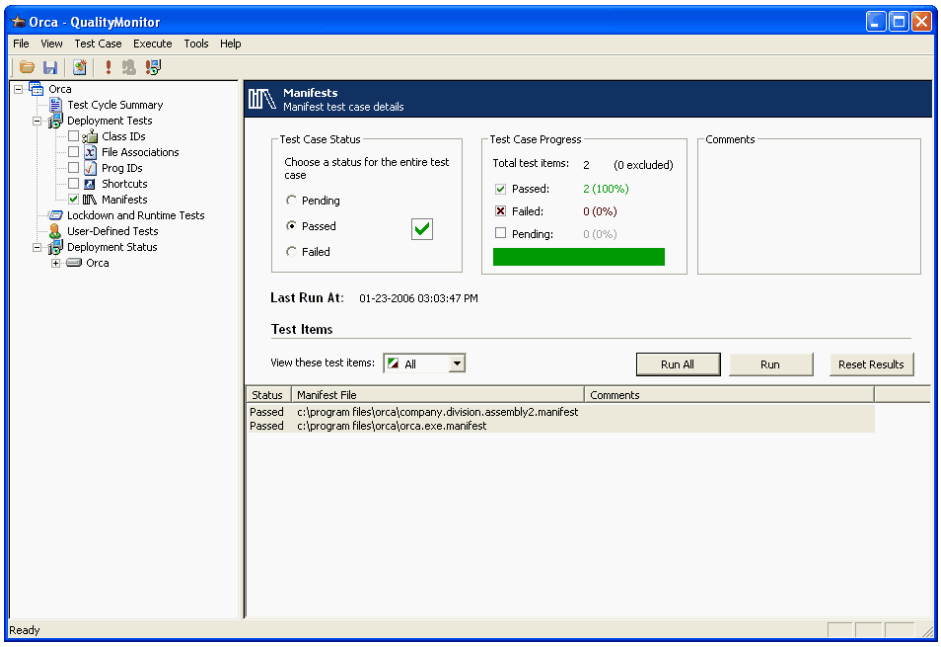


Figure 19-3: QualityMonitor Manifests View

Right-click on the **Test Item** you want to run and select **Run from the context menu**. You can also use the Shift or Ctrl keys to select multiple Test Items to run, or click **Run All** to run all available Test Items.

When testing is finished, results are recorded in the **Test Case Progress** area. Also, the **Status** of each test item (**Passed**, **Failed**, or **Pending**) is listed next to the **Manifest File** name.

When a Test Item is failed, you can view details about it, including the error message associated with it on the **Test Item Information** dialog box. To access this dialog box select the Test Item and then select **Test Item Information** from the context menu.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments. If desired, you can also enter comments in the **Comment** field on this view.

ODBC Data Sources View

The **ODBC Data Sources** Deployment Test is run to verify the ODBC data sources.

At the bottom of the view, you can see all ODBC DSNs associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

***Note** • On the ODBC Data Sources View, only those data sources that belong to the current logged-in user are listed.*

- *For certain ODBC data sources, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to prompt for more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.*

ODBC Drivers View

The **ODBC Drivers** Deployment Test is run to verify ODBC drivers.

At the bottom of the view, you can see all ODBC drivers associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

***Note** • On the ODBC Drivers View, only those drivers that belong to the current logged-in user are listed.*

- *For certain ODBC drivers, additional connection information is required for verification. When the tests are run in Full user interface mode, additional dialog boxes may be displayed during the test to take more input. However, when the tests are run in Silent user interface mode, these additional dialog boxes will not be displayed and results will be based on default information.*

Services View

The **Services** Deployment Test is run to determine if all NT Services have been installed correctly. This is done by opening the Services Manager to determine if the Service exists on the target machine.

At the bottom of the view, you can see all Test Items associated with the Test Case, and can run these items either individually or simultaneously. You can also view individual Test Item details. At the top of the view, you can see and set the status of the entire Test Case, see Test Case progress, filter Test Item data, clear the results, or add comments.

Lockdown and Runtime Tests View

From the **Lockdown and Runtime Tests View**, you can select whether you want to perform runtime checking using a shortcut or an executable in the installed package.

You can select an item in the associated list and click **Run** to launch the application. After exercising the application's functionality and closing it, additional views will appear associated with the executable. These views initially display test items that failed during application operation, and are grouped into:

- Runtime Execution Details View
- Files View
- Folders View
- Registry Entries View
- Isolation Tests View

If you want to execute tests in the context of a different user (under a different user account), click **Run As**. For more information, see Performing Lockdown and Runtime Tests Under a Different User Account.

Caution • *Lockdown and runtime checks cannot be performed on Windows 9x–based systems.*

Runtime Execution Details View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. When you select this executable or shortcut node, the Runtime Execution Details View opens, listing a summary of the execution of the Lockdown and Runtime tests.

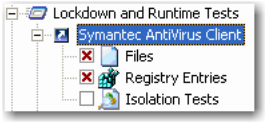


Figure 19-4: Shortcut Node under the Lockdown and Runtime Tests Node

On the Runtime Execution Details View, the following information is included:

Table 19-17 • Lockdown and Runtime Tests View / Runtime Execution Details View Options

Option	Description
Progress	The Progress area includes the following information: • **Total test cases**—Number of test cases (Files, Registry Entries, Folders, Isolation Tests for the selected executable or shortcut) that generated failures plus those test cases that have not yet been completed. • **Passed, Failed, Pending**—Percentage of total test cases that passed the test, failed the test, or have not yet been executed. If an executable or shortcut was run without any failures, the Progress area is not displayed.
Last Run At	Date and time that the last Lockdown and Runtime test was performed, and the name of the user who performed that test.

Files View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. Under this executable or shortcut node, nodes for each Test Case that generated failures or has not yet been executed are listed. If any file errors were generated during this test, then the **Files** node appears.

When the **Files** node is selected, the Files View opens and includes the following information:]

Table 19-18 • Lockdown and Runtime Tests View / Files View Options

Option	Description
Test Case Status	The Test Case Status area displays the state for the entire Test Case. • If failures were generated when this Test Case was executed, QualityMonitor sets the status to **Failed**. • If one of the Test Cases has not yet completed, QualityMonitor sets the status to **Pending**. • If no failures were generated, QualityMonitor sets the status to **Passed**. Depending on your business practices and standards, you may want to override the status of a Test Case from its current state. in this instance, you would manually select another status. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.

Table 19-18 • Lockdown and Runtime Tests View / Files View Options (cont.)

Option	Description
Test Case Progress	The Test Case Progress area includes the following information: • **Total test items**—Number of files that were tested when the shortcut or executable was run. The number of files that were excluded is also listed. • **Passed, Failed, Pending**—Percentage of total test items that passed the test, failed the test, or have not yet been executed.
Comments	Enter comments to document any special considerations or facts regarding this Test Case.
Test Items List	This lists all of the files that were executed when this Lockdown and Runtime Test was executed.
View these test items	Select one of the following to filter the file listing: **All**, **Passed**, **Failed**, or **Pending**.
Set Filter	Click on this button to open the Runtime Test Filters Dialog Box, which lists all errors that were generated during this test. You can then choose to select the errors that you want to exclude from future Lockdown and Runtime tests. These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.
Having these errors	This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). Select an item in this list to filter the file listing by one type of error that was generated. Selecting any error will show only the corresponding errors in the list. To see all the errors, select **Show All**.
Reset Results	Click to reset the status of all of the Test Items to **Pending**.

Folders View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. Under this executable or shortcut node, nodes for each Test Case that generated failures or has not yet been executed are listed. If any file errors were generated during this test, then the **Folders** node appears.

When the **Folders** node is selected, the Folders View opens and includes the following information:]

Table 19-19 • Lockdown and Runtime Tests View / Folders View Options

Option	Description
Test Case Status	The Test Case Status area displays the state for the entire Test Case. • If failures were generated when this Test Case was executed, QualityMonitor sets the status to **Failed**. • If one of the Test Cases has not yet completed, QualityMonitor sets the status to **Pending**. • If no failures were generated, QualityMonitor sets the status to **Passed**. Depending on your business practices and standards, you may want to override the status of a Test Case from its current state. in this instance, you would manually select another status. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.
Test Case Progress	The Test Case Progress area includes the following information: • **Total test items**—Number of files that were tested when the shortcut or executable was run. The number of files that were excluded is also listed. • **Passed, Failed, Pending**—Percentage of total test items that passed the test, failed the test, or have not yet been executed.
Comments	Enter comments to document any special considerations or facts regarding this Test Case.
Test Items List	This lists all of the folders that contained files that were executed when this Lockdown and Runtime Test was executed.
View these test items	Select one of the following to filter the listing: **All**, **Passed**, **Failed**, or **Pending**.
Set Filter	Click on this button to open the Runtime Test Filters Dialog Box, which lists all errors that were generated during this test. You can then choose to select the errors that you want to exclude from future Lockdown and Runtime tests. These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.
Having these errors	This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). Select an item in this list to filter the listing by one type of error that was generated. Selecting any error will show only the corresponding errors in the list. To see all the errors, select **Show All**.

Table 19-19 • Lockdown and Runtime Tests View / Folders View Options (cont.)

Option	Description
Reset Results	Click to reset the status of all of the Test Items to **Pending**.

Registry Entries View

When a Lockdown and Runtime test is run on a package executable or shortcut, a new node with the name of that executable or shortcut is listed under the **Lockdown and Runtime Test Node**. Under this executable or shortcut node, nodes for each Test Case that generated failures or has not yet been executed are listed. If any Registry Entry errors were generated during this test, then the **Registry Entries** node appears.

When the **Registry Entries** node is selected, the Registry Entries View opens and includes the following information:]

Table 19-20 • Lockdown and Runtime Tests View / Registry Entries View Options

Option	Description
Test Case Status	The Test Case Status area displays the state for the entire Test Case. • If failures were generated when this Test Case was executed, QualityMonitor sets the status to **Failed**. • If one of the Test Cases has not yet completed, QualityMonitor sets the status to **Pending**. • If no failures were generated, QualityMonitor sets the status to **Passed**. Depending on your business practices and standards, you may want to override the status of a Test Case from its current state. in this instance, you would manually select another status. In most cases, this will be setting a Test Case which QualityMonitor has marked as **Failed** (because one or more individual Test Items have failed) to **Passed**.
Test Case Progress	The Test Case Progress area includes the following information: • **Total test items**—Number of files that were tested when the shortcut or executable was run. The number of files that were excluded is also listed. • **Passed, Failed, Pending**—Percentage of total test items that passed the test, failed the test, or have not yet been executed.
Comments	Enter comments to document any special considerations or facts regarding this Test Case.
Test Items List	This lists all of the Registry Entries that were tested when this Lockdown and Runtime Test was executed.
View these test items	Select one of the following to filter the listing: **All**, **Passed**, **Failed**, or **Pending**.

Table 19-20 • Lockdown and Runtime Tests View / Registry Entries View Options (cont.)

Option	Description
Set Filter	Click on this button to open the Runtime Test Filters Dialog Box, which lists all errors that were generated during this test. You can then choose to select the errors that you want to exclude from future Lockdown and Runtime tests. These settings are stored in the default exclusion list (the **Exclusion file** selected on the General Tab of the **Options** dialog box), but no changes are made to the Project file. This filter is based on the error code associated with an error, and these error codes are stored in the default exclusion list.
Having these errors	This list includes all the unique errors that were generated when this Test Case was executed (excluding the errors that were filtered out using the **Set Filters** function). Select an item in this list to filter the listing by one type of error that was generated. Selecting any error will show only the corresponding errors in the list. To see all the errors, select **Show All**.
Reset Results	Click to reset the status of all of the Test Items to **Pending**.

Isolation Tests View

You can run Isolation Tests to display the location of all portable executable (PE) files (`dll/ocx/exe/tlb/olb`) that are launched from a process while performing a Lockdown and Runtime test. Viewing a listing of these portable executable file names and paths makes it easier for you to ensure that the application is fully isolated.

After you perform a Lockdown and Runtime Test for an executable (`.exe`) or a shortcut on the Lockdown and Runtime Tests View, an additional node called **Isolation Tests** is added to the tree under the executable or shortcut node.

When you select this **Isolation Tests** node, the filenames of the portable executable files and their paths are listed. By default, the status of all these items is **Pending**. To ensure that all of the executables or shortcuts in this test case are isolated, go to the **Test Case Status** area of the view, and set the status of the entire test case to either **Pending**, **Passed**, or **Failed**.

Note • *The Isolation Tests node will be added to the Lockdown and Runtime Tests tree only if the selected executable launches at least one portable executable file.*

- *You can specify when you would like the **Isolation Tests** node on the **General** tab of the Options Dialog Box. On the **General** tab, select an option from the **Show Isolation Tests** list .*
- *QualityMonitor does not support isolation testing under Windows 2000.*

User-Defined Tests View

As your business practices dictate, you can add additional, custom tests to the QualityMonitor project file. This is accomplished by right-clicking on the User-Defined Tests view and selecting Add Test Case.

Test Case View

When you select a user-defined Test Case under User-Defined Tests on the View List, the Test Case View opens.

This view contains the following options:

Table 19-21 • Test Case View Options

Option	Description
Test Case Status	Specify the status of the selected Test Case by selecting one of the following options: • **Pending**—Test case has not been executed. • **Passed**—Test case has been executed and has passed. • **Failed**—Test case has been executed and has failed.
Instructions	Enter any instructions to explain how to execute this Test Case.
Select an executable	Select an executable to launch when this Test Case is run.
Comments	Enter comments to document the purpose of this Test Case or to note any important issues.

Deployment Status View

The Deployment Status View lists all of the products and features in the MSI package.

Products and features in the MSI package are listed in the Deployment Status tree, with an icon indicating its status:

Table 19-22 • Deployment Status View Status Icons

Icon	Description
	installed
	not installed
	a key file is either missing or does not match the version or size of that file recorded in the MSI file

When you select the Deployment Status node, all of the components in all of the product features are listed on the right. If you select an individual feature, only those components within that feature are listed. The following information is displayed:

Table 19-23 • Deployment Status View Options

Option	Description
Component Name	Name of all components in the MSI package or selected feature.
Component Status	Status of the listed component: either installed ( ), not installed ( ), or a key file is either missing or does not match the version or size of that file recorded in the MSI file ( )
Component Location	Location of installed component.
Component ID	GUID of the component, which uniquely identifies it.

Part 5

Preparing Installations for Deployment

This part of the AdminStudio 10.0 SP2 User Guide includes the following chapters:

- Preparing Packages for Distribution Using Distribution Wizard
- Distributing Packages Using Configuration Manager Web Console
- Analyzing Installations Prior to Deployment
- License-Enabling Packages Using FLEXwrap

20

Preparing Packages for Distribution Using Distribution Wizard

The final step in the application migration process is preparing your packages for distribution. AdminStudio includes the Distribution Wizard to assist you in deploying your package (and any associated transforms) using one of the following distribution types:

Table 20-1 • Supported Distribution Types

Distribution Type	Topic
Administrative Install	Creating Administrative Installations for Packages
Altiris	Preparing for Altiris Distribution
FTP Location	Distributing Packages to FTP Servers
LANDesk	Preparing for LANDesk Distribution
ManageSoft	Preparing for ManageSoft Distribution
Marimba	Using Marimba Channel Publishing for Package Deployment
Network Location	Distributing Packages to Network Locations
Configuration Manager	Distributing to Microsoft Configuration Manager
SMS	Preparing for SMS Distribution
Configuration Manager Web Console	Distributing Packages Using the Configuration Manager Web Console
Tivoli	Preparing for Tivoli Distribution

Table 20-1 • Supported Distribution Types (cont.)

Distribution Type	Topic
ZENworks Configuration Management	Preparing for ZENworks Configuration Management 10 Distribution
ZENworks Desktop Application	Preparing for ZENworks Desktop Application Distribution
ZENworks Server Distribution	Preparing for ZENworks Server Distribution

This section also explains how to deploy a Windows installer package using an InstallShield script-based setup. See Deploying InstallScript MSI Installations.

Creating Administrative Installations for Packages

In an administrative installation, the installation software is copied to a network directory using the administrative install option provided by Windows Installer.

Task: ***To distribute your package (and any associated transforms) as an administrative installation:***

1. Launch the Distribution Wizard by either clicking on its icon in the Tools Gallery or by selecting a package in the Application Manager and selecting **Distribute Package** from the context menu.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **Administrative Install** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the Windows Installer Package (`.msi`) you want to distribute.

   If you launched the Distribution Wizard from the Application Manager by selecting a package and selecting **Distribute Package** from the context menu, the name in the **Windows Installer Package (.msi)** field is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository:

   - **Not in the Software Repository**—The full name and path of the Windows Installer file is displayed, and you can edit this entry or click **Browse** and select a different package.
   - **In the Software Repository**—Only the name of the Windows Installer file is displayed (not the full path) and this entry cannot be edited or changed.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **Administrative Install** panel opens.

7. From the **Administrative Install** panel, specify or browse to the Network Directory to which you want to distribute the package.
8. If desired, you can use short file names during the distribution by selecting the Use short file names option.
9. Click **Next**. The **Distribution Summary** panel opens.
10. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files. The **Distribution Output** panel displays progress during distribution.
11. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

Distributing Packages to FTP Servers

You can choose to distribute a package to an FTP server. If you select the **FTP Location** option on the Distribution Wizard **Distribution Type** panel, you will be required to enter the location of the FTP server, and the user name and password to connect to that server.

Task: ***To distribute your package (and any associated transforms) to an FTP server:***

1. Launch the Distribution Wizard.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **FTP Location** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **FTP Location** panel opens.
7. From the **FTP Location** panel, specify the location of the FTP server, and the user name and password to use to connect to the server. Click **Next**.
8. Review the selections you made in the **Distribution Summary** panel. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files.
9. The **Distribution Output** panel displays progress during distribution. Once the distribution finishes, click Finish to exit the Distribution Wizard.

Preparing for Altiris Distribution

AdminStudio 10.0 SP2 | Distribution Wizard

The Distribution Wizard supports the distribution of a setup along with any transforms and files via an Altiris Notification Server. A custom script file is required for Altiris distribution. The Distribution Wizard creates this custom script file using an XML template file that is provided: `AltirisTemplate.Config`.

Task: ***To prepare your package for Altiris distribution:***

1. Launch the Distribution Wizard. The **Distribution Wizard Welcome** panel opens.
2. Click **Next**. The **Distribution Type** panel opens.
3. Select **Altiris** from the **Distribution Type** list and click **Next**. The **Package Information** panel opens.
4. Click the **Browse** button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the Add button in the **Additional Transforms** area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **Altiris Integration** panel opens.
7. In the **Network Directory** field, specify or browse to the network location where you want to store the installation package. The Distribution Wizard remembers the last **Network Location** that is entered and displays it the next time this panel is accessed.

   The Distribution Wizard will copy the Windows Installer package along with any transforms and files to the UNC path specified. Also, the Distribution Wizard will use an XML template file (`AltirisTemplate.config`) to create a custom script file in this location named `<packageName>.Config`.

   Note • *You can edit AltirisTemplate.config to customize it for your organization. The file, which is installed with AdminStudio, is located in the* `Templates` *folder of the* `AdminStudio Shared` *Directory. See Altiris XML Template for more information.*

8. In the Windows Installer Command Line field, enter any additional properties that you want to pass to the Windows Installer. See the Windows Installer Property Reference for more information.
9. In the **Altiris Server Location** field, enter the `http:` address for the location of the Altiris Server. The Distribution Wizard remembers the last Altiris Server Location that is entered and displays it the next time this panel is accessed.
10. In the **User Name** and **Password** fields, enter a User Name and Password to log onto the server entered in the **Altiris Server Location** field. The Distribution Wizard remembers the last User Name that is entered and displays it the next time this panel is accessed.
11. Click **Next**. The **Distribution Summary** panel appears, listing the selections you made in the previous panels.

12. Review the information on the **Distribution Summary** panel. If you are satisfied with them, click **Next**.The D**istribution Output** panel displays progress during distribution.
13. Once the distribution finishes, click **Finish** to exit the **Distribution Wizard**.

Note • *For all distribution types, the Distribution Wizard will create a Distribution log file in the Distribution folder of the AdminStudio Shared Directory.*

Preparing for LANDesk Distribution

AdminStudio 10.0 SP2 | Distribution Wizard

With LANDesk distribution, the MSI package along with all the setup files are copied to a network location.

Task: ***To prepare your package for LANDesk distribution:***

1. Launch the Distribution Wizard.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **LANDesk** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **LANDesk Integration** panel opens.
7. In the Network Directory or URL field on the **LANDesk Integration** panel, specify the network location where you want to copy the MSI package and all of its setup files. The Network Directory could be a URL or a UNC path. This field will default to the last used path, and will provide a most recently used list.
8. Click **Next**. The **Distribution Summary** panel opens.
9. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files. The **Distribution Output** panel displays progress during distribution.
10. Once the distribution finishes, click Close to exit the Distribution Wizard.

Preparing for ManageSoft Distribution

AdminStudio 10.0 SP2 | Distribution Wizard

To convert a Windows Installer .msi package directly into a ManageSoft-wrapped .msi package for deployment to managed devices, choose the ManageSoft distribution method on the **Distribution Type** panel of the Distribution Wizard. This will also create the necessary ManageSoft .ndp file.

Caution • *To use the ManageSoft distribution method in the Distribution Wizard, your system must satisfy the requirements specified in ManageSoft 7.2 MGS.500.006 PRD as well as the AdminStudio system requirements.*

Perform the following steps to prepare for ManageSoft distribution:

Task: ***To prepare your package for ManageSoft distribution:***

1. Launch the Distribution Wizard.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **ManageSoft** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button () in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **ManageSoft Distribution Settings** panel opens.
7. Select one of the following options from the Configure MSI Package Source field:
   - **Install From Cache**—Configure the ManageSoft-wrapped package to be installed from a locally managed device cache.
   - **Install From Server**—Configure the ManageSoft-wrapped package to be installed from a Server.
   - **Install Source based upon Managed Device's Configuration**—Configure the ManageSoft-wrapped package to be installed from either a Server or device cache, depending on the package's configuration.
8. Select the Retain the ManageSoft wrapped .msi package on local managed device cache option if you do not want the Distribution Wizard to delete the ManageSoft wrapped .msi package from the local cache after installation. You might want to select this option if you want to reinstall the ManageSoft wrapped `.msi` package on another machine. This option is enabled when you select Install Source based upon Managed Device's Configuration from the **Configure MSI Package Source** list.
9. Select the **Install MSI package with elevated privileges** option if you need elevated privileges to install the ManageSoft wrapped `.msi` package due to lack of user privileges in your environment. Usually this option is not selected because elevated privileges are not necessary for most installations to complete successfully.

10. Click **Next**. The **ManageSoft Package and Environmental Settings** panel opens.
11. The package Manufacturer, ManageSoft Package Name, Application Name, and Version fields are populated based upon information saved in the selected package. Make any desired changes.
12. Under Supported Environments, select those environments where this ManageSoft-wrapped package will be deployed.
13. Click **Next**. The **Distribution Summary** panel opens.
14. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files. The **Distribution Output** panel displays progress during distribution.
15. Once the distribution finishes, click Close to exit the Distribution Wizard.

Using Marimba Channel Publishing for Package Deployment

AdminStudio 10.0 SP2 | Distribution Wizard

Systems Administrators use the Marimba Desktop suite of products to ensure that users in their network always have the most recent software on their local computer. Distribution Wizard makes it easy for you to efficiently distribute packages using Marimba.

In Marimba Channel Publishing, the System Administrator publishes an application (called a Channel) on a Marimba Transmitter web site. The Marimba Desktop Tuner component, installed on a user's local machine, communicates with the Transmitter web site to find out which Channels this user has subscribed to, and if any of those Channels have recently been updated.

Initially, the Marimba Desktop Tuner downloads the Channels that the user has subscribed to, installs them on the user's local machine. Then it periodically checks the Transmitter web site to see if any of this user's subscribed Channels have been updated. You can use the Distribution Wizard to create and publish a Channel (such as an installation setup) on the Transmitter web site so that the Marimba Transmitter can download a Channel to a local machine.

Here is a review of terms associated with Marimba distribution:

Table 20-2 • Marimba Distribution Terminology

Term	Description
Marimba Desktop Tuner	An application on your local computer that downloads Channels from the Transmitter, allowing you to run the Channel applications on your local computer.
Transmitter	The application that stores the original Channel before it is downloaded to your computer. It usually runs on another computer in your network.

Table 20-2 • Marimba Distribution Terminology (cont.)

Term	Description
Channel	Any application or file that the Marimba Desktop Tuner downloads to your computer. Channels are initially stored on the Transmitter and then the Marimba Desktop Tuner downloads the Channels so you can run them on your local computer.
Subscription	The process of getting a Channel on your computer. Once you subscribe to a Channel, the Marimba Desktop Tuner keeps track of it, making sure the information is up to date. The Marimba Desktop Tuner and the Channel are automatically set up to make sure the Channel stays up to date on the local computer by downloading updates at scheduled intervals.

Task: ***To distribute your package (and any associated transforms) using Marimba Channel publishing:***

1. Launch the Distribution Wizard.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. On the **Distribution Type** panel, select **Marimba** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. If you want to set public MSI properties in the Windows Installer package, enter those properties in the Specify Additional MSI Properties field using the following syntax:

   ```
   PROP1=value1, PROP2=value2, PROP3=value3
   ```

7. After specifying the package location and specifying any additional MSI properties, click **Next**. The **Patches** panel opens.
8. In the Patches (.msp) area:
   - If there are patches associated with the package, click the New button () and navigate to the patch you want to add.
   - Use the Up and Down arrows ( ) to set the order in which the patches are applied to the package.
   - Use the Delete button ( ) to delete a patch from the list.
9. After you have added all patches, click **Next**. The **Marimba Integration** panel opens.
10. On the **Marimba Integration** panel, do the following:
    a. **Specify the directory where you want to place the contents of the new Channel**—Specify the directory where you want to place the MSI file, or click Browse to select it. This directory will be called the Channel Directory.

Note • *Using Distribution Wizard, users can specify where they would like to copy an MSI package and all the other related files. The Distribution Wizard will then automatically find the Application Packager Channel URL and copy the files at the specified location. This location is called the Channel Directory. Users have a provision to specify any other Application Packager URL. The Channel Directory is also referred to as the Publish Directory. Distribution Wizard is required to be able to find the Application Packager Channel URL.*

b. **Include all the files that are in the same directory as the Windows Installer package**—If desired, select this option. If this option is selected, all the files and folders in the same directory as the MSI package will be copied into the Channel Directory.

c. **Specify the destination Channel URL on the transmitter**—This is an optional field. This is the URL to the Transmitter that contains the Channel. For example, the URL to a Transmitter might be http://trans.marimba.com. The URL to a Channel on that Transmitter might be http://trans.marimba.com/MyChannel. However, if you specify a value in this field, you must also specify the Channel Copier URL field on the **Advanced Marimba Options** dialog box.

11. To specify advanced Marimba options for Channel Creation and Channel Publishing, click Advanced to access the **Advanced Marimba Options** dialog box.

    a. **Specify the Application Package URL**—The Application Package URL will be used to copy the files to the Channel directory. Accept the default application package URL or enter a new one.

    b. **Specify the XML template file**—The XML template file sets properties for the MSI file during Marimba distribution. Accept the default XML template file that was shipped with AdminStudio, or enter a new one.

    c. **Specify the Channel Copier URL**—If you have specified a destination Channel URL on the transmitter, then specify the URL for the Channel Copier. Distribution Wizard uses the Channel Copier URL to publish the channel to a transmitter, where it will be available to desktops.

    d. Click **OK** to return to the **Marimba Integration** panel.

12. On the **Marimba Integration** panel, click **Next**. The **Distribution Summary** panel opens.

13. Review the selections you made in the **Distribution Summary** panel. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files. The **Distribution Output** panel displays progress during distribution.

14. Once the distribution finishes, click Close to exit the Distribution Wizard.

Distributing Packages to Network Locations

To distribute a package to a network directory, select the **Network Location** option on the Distribution Wizard **Distribution Type** panel.

Task: ***To distribute your package (and any associated transforms) to a network location:***

1. Launch the Distribution Wizard. The Distribution Wizard **Welcome** panel opens.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **Network Location** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the Browse button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **Network Location** panel opens.
7. From the **Network Location** panel, specify or browse to the Network Directory location to which you want to distribute the package, and click **Next**. The **Distribution Summary** panel opens.
8. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files. The **Distribution Output** panel displays progress during distribution.
9. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

Distributing to Microsoft Configuration Manager

To publish a Windows Installer or Microsoft App-V package from the Application Catalog to a Microsoft System Center Configuration Manager server, select **Configuration Manager** from the **Distribution Type** panel of the Distribution Wizard.

Task: ***To publish packages to Microsoft Configuration Manager:***

1. Launch the Distribution Wizard. The Distribution Wizard **Welcome** panel opens.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **Configuration Manager** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the **Browse** button and locate the Windows Installer package (`.msi`) or MIcrosoft App-V application (`.sft`) that you want to distribute.
5. Click **Next**. The **Connect to a Microsoft System Center Configuration Manager Server** panel opens.

6. Specify the connection information for a Microsoft Configuration Manager Server:

Option	Description
Server	Enter the name of the Microsoft Configuration Manager Server that you want to connect to.
Authentication	Choose one of the following options: • **Server Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server login identification to log into this server. Then enter the appropriate **User name** and **Password**. • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this Microsoft Configuration Manager Server. ***Note*** *• After you successfully connect to a Microsoft Configuration Manager Server, the next time you open this panel, those previously-entered values (except the **Password**) will pre-populate this panel.*

7. Click **Next**. The **Select Destination Folder** panel opens prompting you to select a location that the Microsoft Configuration Manager Server has access to where you want to publish the selected packages.

8. On the **Select Destination Folder** panel, enter the following information:

Property	Description
Location to Publish Packages	Enter a target path, in UNC format (`\\Server\Share`), of the location where you want to publish the selected packages. Make sure that you enter a location that the Microsoft Configuration Manager Server has access to.
Authentication	From the **Authentication** list, select one of the following options: • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this location. • **SCCM Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server authentication (your Microsoft Configuration Manager Server login ID) to log into this location. • **Server Authentication**—Choose this option if you are publishing to an alternate file server that requires credentials. Then enter the appropriate **User Name** and **Password**.

9. Click **Next**. The **Select Group** panel opens.

10. Select the **Target Group** on the Configuration Manager Server where you want to publish the package and click **Next**. The **Distribution Summary** panel opens.

11. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package. The **Distribution Output** panel displays progress during distribution.
12. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

Preparing for SMS Distribution

To prepare for SMS distribution, select **SMS** on the **Distribution Type** panel of the Distribution Wizard. The Distribution Wizard will create a Package Definition File. SMS 1.2 or earlier uses a `.pdf` file extension, and SMS 2.0 or later uses an `.sms` file extension.

Task: ***To prepare your package for SMS distribution:***

1. Launch the Distribution Wizard.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **SMS** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the Browse button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **SMS Distribution** panel opens.
7. From the **SMS Distribution** panel, configure settings as necessary and click **Next**. The **Network Location** panel opens.
8. On the **Network Location** panel, specify the network location to distribution the installation package and click **Next**. The **Distribution Summary** panel appears.
9. On the **Distribution Summary** panel, review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files, to the location specified in the **SMS Distribution** panel. The **Distribution Output** panel displays progress during distribution.
10. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

Note • *For detailed information on SMS template files and the variables they use, see SMS File Templates.*

Note • *Any associated transforms are merged into the Windows Installer package before the package is distributed to SMS.*

SMS File Templates

The following templates are used when creating SMS files: `template.pdf` (for SMS 1.2) and `template.sms` (for SMS 2.0). You can find these files in the `C:\AdminStudio Shared\Templates` folder. You can replace or modify these templates and any instances of the predefined variables (below) will be replaced.

Example

The following table displays a PDF file template in the left column, and the output of that PDF file in the right column:

Table 20-3 • Comparison of an SMS Template File and an Output File

Template	Output
[PDF] Version=1.0 [Package Definition] Product=%PRODUCT% Version=%VERSION% Comment=%MANUFACTURER% SetupVariations=Typical, Automated [Typical Setup] CommandName=Typical Installation CommandLine=msiexec /i %CMDLINE% UserInputRequired=TRUE SynchronousSystemExitRequired=TRUE SupportedPlatforms=Win 9x, Win NT (i386) [Automated Setup] CommandName=Automated Installation CommandLine=msiexec /i %CMDLINE% /q UserInputRequired=FALSE SynchronousSystemExitRequired=TRUE SupportedPlatforms=Win 9x, Win NT (i386) [Setup Package for Inventory] InventoryThisPackage=FALSE	[PDF] Version=1.0 [Package Definition] Product=Orca Version=1.50.2600.0000 Comment=Microsoft SetupVariations=Typical, Automated [Typical Setup] CommandName=Typical Installation CommandLine=msiexec /i orca.msi UserInputRequired=TRUE SynchronousSystemExitRequired=TRUE SupportedPlatforms=Win 9x, Win NT (i386) [Automated Setup] CommandName=Automated Installation CommandLine=msiexec /i orca.msi /q UserInputRequired=FALSE SynchronousSystemExitRequired=TRUE SupportedPlatforms=Win 9x, Win NT (i386) [Setup Package for Inventory] InventoryThisPackage=FALSE

Supported Variable Names

Below is a list of supported variable names and what they map to in the PDF/SMS files.

Table 20-4 • SMS Supported Variable Names

Variable Name	Value	SMS 1.2 (.PDF)	SMS 2.0 (.SMS)
%PRODUCT%	MSI ProductName	[Package Definition] Product	[Package Definition] Name

Table 20-4 • SMS Supported Variable Names (cont.)

Variable Name	Value	SMS 1.2 (.PDF)	SMS 2.0 (.SMS)
%VERSION%	MSI ProductVersion	[Package Definition] Version	[Package Definition] Version
%MANUFACTURER%	MSI Manufacturer	[Package Definition] Comment	[Package Definition] Publisher
%CMDLINE%	MSI filename + transforms	N/A	[SetupVariation] CommandLine
%MIF%	MIF filename specified in Wizard	N/A	[Package Definition] MIFFilename
%LANGUAGE%	MSI ProductLanguage converted into text	N/A	[Package Definition] Language
%PRODUCTCODE%	MSI ProductCode	N/A	[SetupVariation] UninstallKey

Note • *These variable names are case sensitive.*

Preparing for Tivoli Distribution

AdminStudio 10.0 SP2 | Distribution Wizard

To prepare your package for Tivoli distribution, perform the following steps.

Task: ***To prepare your package for Tivoli distribution:***

1. Launch the Distribution Wizard.
2. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
3. From the **Distribution Type** panel, select **Tivoli** and click **Next**. The **Package Information** panel opens.
4. On the **Package Information** panel, click the Browse button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **Tivoli Integration** panel appears.

7. Specify or browse to the Network Directory where you want to store the installation package and created SPD file. Click **Next**. The **Tivoli Settings** panel opens.
8. Configure settings as necessary and click **Next**. The **SPD Parameters** panel opens.
9. Configure each feature's install state. When finished, click **Next**. The **Distribution Summary** panel opens.
10. Review the selections you made. If you are satisfied with them, click **Next** to distribute the package, including associated transforms and files, to the location specified in the Tivoli Integration panel. The **Distribution Output** panel displays progress during distribution.
11. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

***Note** • Any associated transforms are merged into the Windows Installer package before the package is distributed to Tivoli.*

Preparing for ZENworks Configuration Management 10 Distribution

AdminStudio 10.0 SP2 | Distribution Wizard

To prepare your package for ZENworks Configuration Management 10 distribution, perform the following steps.

Task: ***To prepare your package for ZENworks Configuration Management 10 distribution:***

1. Launch the Distribution Wizard for ZENworks Configuration Management by performing the following steps:
   a. Launch the **Distribution Wizard**.
   b. On the **Welcome** panel, click **Next**. The **Distribution Type** panel opens.
   c. Select **ZENworks Configuration Management Distribution** and click **Next**.

   The **Welcome** panel of the Distribution Wizard for ZENworks Configuration Management opens.

   ***Note** • If you do not want the **Welcome** panel to be displayed each time you open this wizard, select the **Do not show the Welcome panel again** option. If this option is selected, the **Login** panel will be the first panel opened for this wizard.*
2. Click **Next** to continue. The **Login** panel opens.

3. On the **Login** panel, enter the following login information for the ZENworks Configuration Management server that you are connecting to:

Property	Description
User Name / **Password**	Enter a valid User Name and Password for the ZENworks Configuration Management server you are connecting to.

Server URL	Enter the server URL, machine name, or IP address of the ZENworks Configuration Management server using the following format:
	`http://www.servername.com` or `http://111.22.333.44`
	If you need to specify a specific port number, append the port number to the end of the URL, such as:
	`http://www.servername.com:123`
	If you are using SSL and you want a secure connection, change the `http` prefix to `https`. For example:
	`https://www.servername.com`

4. Click **Login**. The **Windows Installer Package Information** panel opens.

5. Click **Browse** next to the **Windows Installer Package File (.msi)** field and select the Windows Installer package that will be referenced by this ZENworks Configuration Management bundle.

Important *• All of the files in the selected Windows Installer file's directory and all of its subdirectories will be uploaded to the ZENworks Configuration Management Server.*

After you make your selection, if there are any transform files (`.mst`) in the same directory, they are listed in the **Windows Installer Transform Files (.mst)** area.

Note *• All of the* `.mst` *files that are in the same directory as the selected Windows Installer package are automatically listed in the* ***Windows Installer Transform Files (.mst)*** *list, even if they are not applicable to the selected package. To prevent the inclusion of non-applicable transform files, delete those transforms from the list.*

6. To include transforms with the Windows Installer package:

   - Click the New button ( ) in the **Windows Installer Transform Files (.mst)** area and select a transform file. If the package requires multiple transforms, you can repeat the procedure as necessary.
   - Select a transform and click the Delete button ( ) to delete a transform from the list.

7. If you want to customize how this package is installed, enter parameters in the **Install Parameters** field. Any actions that you enter will be performed whenever this bundle is installed.
   - The root parameter, which should not be edited or deleted, is `/i packagename.msi`.
   - These parameters are applied to `msiexec.exe` to perform the desired action.
   - By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input.

     If this operation requires user input, either remove the `/qn` parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.
   - For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.
8. If you want to customize how this package is uninstalled, enter parameters in the **Uninstall Parameters** field. Any actions that you enter will be performed whenever this bundle is uninstalled.
   - The root parameter, which should not be edited or deleted, is `/x packagename.msi`.
   - By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input.

     If this operation requires user input, either remove the `/qn` parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.
   - For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.
9. If you want to customize how this package is repaired, enter parameters in the **Repair Parameters** field. Bundles are repaired by reinstalling missing or corrupted files.
   - The root parameter, which should not be edited or deleted, is `/f packagename.msi`.
   - By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input.

     If this operation requires user input, either remove the `/qn` parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.
   - You can apply any of the following additional Repair parameters after the package name:

Parameter	Description
p	Reinstalls a file if it is missing
o	Reinstalls a file if it is missing or if an older version of the file is present on the user's system
e	Reinstalls a file if it is missing or if an equivalent or older version of the file is present on the user's system

Parameter	Description
c	Reinstalls a file if it is missing or if the stored checksum of the installed file does not match the new file's value
a	Forces a reinstall of all files
u or m	Rewrite all required user registry entries
s	Overwrites any existing shortcuts
v	Runs your application from the source and re-caches the local installation package

- For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.

10. Click **Next**. The **Bundle Creation Options** panel opens.
11. Specify whether to update an existing bundle or create a new one by selecting one of the following options:
    a. **Create a new bundle from these Windows Installer package files**—To create a new bundle to reference this Windows Installer package, select this option.
    b. **Update an existing bundle using these Windows Installer package files**—If you want to overwrite an existing bundle to contain this Windows Installer package, select this option, and then select an existing bundle from the tree:
       - **Recommended Bundles**—This group lists the bundles that contain the same Windows Installer package as the one you selected on the Windows Installer Package Information Panel.
       - **All Other Bundles**—This group lists the rest of the existing bundles on the server.
12. After making your selection, click **Next** to proceed. The **Bundle Information** panel opens.
13. Enter information to specify attributes for this bundle on ZENworks Configuration Management:

Property	Properties
Bundle Name	Enter the bundle's name as you want it to appears in ZENworks Control Center (ZCC) and the ZENworks Application Launcher (on managed devices).
Version Number	Enter the bundle's version number. If you are overwriting an existing bundle, and you enter a higher version number than the bundle's original version number, the bundle will be redeployed.
Icon	Click **Browse** and select a shortcut icon graphic (in `.ico`, `.gif`, `.jpg`,`.png`, `.bmp`, or `.exe` format) that ZENworks Application Launcher will display on managed devices. If you do not select an icon file, the standard ZENworks bundle icon will be used.

Property	Properties
Folder	From the **Folder** list, select the folder path that will be used by ZENworks Application Launcher when displaying the bundle on either the device's desktop or **Start** menu. All of the folders defined on the ZENworks server are listed. For example: • **Start Menu**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the **Start** menu, ZENworks Application Launcher creates an `Application\Accounting` folder on the root of the **Start** menu and adds the bundle to it. • **Desktop**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the desktop, ZENworks Application Launcher creates an `Applications\ Accounting` folder on the desktop and adds the bundle to it. You can place multiple bundles in a single folder by specifying the same folder path for each of the bundles.
Description	Enter a description of the bundle. This description will be displayed in ZENworks Control Center and the ZENworks Application Launcher (on managed devices).

14. After you have entered bundle information, click **Next**. The **Summary** panel opens, displaying the options you have selected for distributing this Windows Installer package on ZENworks Configuration Management.
15. Click **Publish** to complete the distribution process. The **Publishing Process** panel opens, listing the progress messages while the bundle is being published on ZENworks Configuration Management.
    - **ZENworks error messages**—Any error messages with a numeric prefix that appear on this panel are generated by ZENworks Configuration Management. To resolve these errors, contact your ZENworks Configuration Management System Administrator.
    - **Cancelling publication**—If you want to cancel the publication of the bundle on ZENworks Configuration Management, click **Cancel**.
16. When processing is complete, the **Finish** button becomes enabled. Click **Finish** to exit this wizard.

See Also
Distribution Wizard for ZENworks Configuration Management

Preparing for ZENworks Desktop Application Distribution

AdminStudio 10.0 SP2 | Distribution Wizard

The Distribution Wizard supports the distribution of a setup along with any transforms and files via ZENworks desktop application distribution. You would select this option if you want to create an MSI Distribution Object to distribute to ZENworks desktops.

Note • *If you are using ZENworks Configuration Manager Server 10, instead of using the standard Distribution Wizard, you should use the customized Distribution Wizard for ZENworks Configuration Management. See Preparing for ZENworks Configuration Management 10 Distribution for more information.*

Task: ***To prepare your package for ZENworks desktop application distribution:***

1. Launch the Distribution Wizard. The Distribution Wizard **Welcome** panel opens.
2. Click **Next**. The **Distribution Type** panel opens.
3. Select **ZENworks Desktop Application** from the **Distribution Type** list and click **Next**. The **Package Information** panel opens.
4. Click the Browse button and locate the Windows Installer Package (.msi) you want to distribute.
5. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.
6. After specifying the package location, click **Next**. The **ZENworks Login** panel opens.
7. Use this panel to login to the server location where you want to create the Distribution Object. Specify the following information:
   a. In the **Login Name** field, enter a valid Login Name for the Novell server where you want to create the Distribution Object. This is a required field. For more information, see Specifying Multiple Parameters in the Login Name and Context Fields of the ZENworks Login Panel.
   b. In the **Password** field, enter a valid password for the specified Login Name.
   c. In the **Server** field, enter the machine name or IP address of the Novell server where you want to create the Distribution Object.
   d. In the **Context** field, enter the Novell environment where you want to create the Distribution Object. Because you can choose to define the context information in the Login Name field, this field is optional. For more information, see Specifying Multiple Parameters in the Login Name and Context Fields of the ZENworks Login Panel.
   e. In the **SSL Port Number** field, specify the secure port to connect to ZENworks. The default/recommended port value is 636.

***Note** • All of these entries except the **Password** are saved and will pre-populate these fields the next time you access this panel.*

8. After you have entered the login information, click **Next**. The **ZENworks Desktop Application** panel opens.
9. Use this panel to specify the ZENworks Application Object and its properties. Specify the following information:
   a. In the Object Name field, specify the name of the Application Object to be created in the specified Novell eDirectory® tree. You can use the name of the .msi file or any other name you choose.
   b. In the Version field, specify the Application Object's version. For a new Application Object, leave this field empty (in which case 0 is used). For existing Application Objects, enter an integer to identify the version.
   c. In the **Context** field, enter the Novell environment where you want to create this Application Object. For more information, see Specifying Multiple Parameters in the Context Field of the ZENworks Server Distribution Panels.
   d. In the **Administration Package Path** field, specify the path that ConsoleOne® will use to access the .msi package. ConsoleOne uses the package file to populate information in the Application Object. This field defaults to the directory specified by the .msi file. If desired, you may edit this path.

      ***Note** • This field is used only by ConsoleOne to access the .msi package. It is not used by Novell® Application Launcher for distribution of the application. For distribution, Application Launcher uses the path defined in the SOURCE_PATH.*

   e. In the SOURCE_PATH area, specify a list of network directories that are the desired location(s) for the package source path(s). You can specify one or more source paths.

      To add a path, click the New button ( ) to open the Browse for Folder dialog box, and select the path. If you want to include multiple source paths, you can repeat the procedure as necessary. If you need to delete a path you have added, select the path and click the Delete button ().
   f. Click **Next**. The **Distribution Summary** panel appears, listing the selections you made in the previous panels.
10. Review the information, and if satisfied, click **Next**.The **Distribution Output** panel displays progress during distribution.
11. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

***Note** • For all distribution types, the Distribution Wizard will create a Distribution log file in the Distribution folder of the* `AdminStudio Shared` *directory.*

Specifying Multiple Parameters in the Login Name and Context Fields of the ZENworks Login Panel

The entries that you make in the **Login Name** and **Context** fields are concatenated to form the Novell distinguished name. If you enter `jsmith` in the **Login Name** field and `myorg` in the **Context** field, the login information that AdminStudio would pass to the server would be:

```
cn=jsmith, o=myorg
```

Specifying Multiple Parameters in the Login Name Field

Instead of making entries in both the **Login Name** and **Context** fields, you could enter both the login name and the context in the **Login Name** field (separated by either periods or commas). AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-5 • How To Specify Multiple Parameters in the Login Name Field

If you enter ...	AdminStudio assumes that ...
two unqualified parameters in the **Login** field, such as: `jsmith, myorg`	• the first entry identifies the login name (`cn`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `cn=jsmith, o=myorg`
more than two unqualified parameters in the **Login** field, such as: `jsmith, dept01, office03, myorg`	• the first entry identifies the login name (`cn`) • the last entry identifies the organization parameter (`o`) • all entries in between the first and the last entry are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `cn=jsmith, ou=dept01, ou=office03, o=myorg`

In this instance, you would leave the Context field blank.

Specifying Multiple Parameters in the Context Field

If you would like to enter multiple parameters in the Context field, you would enter the login name in the Login Name field without parameters (`jsmith`, for example), and then specify all of the context parameters in the Context field. You can enter multiple qualified parameters in any order:

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-6 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: `myorg`	The following parameter is passed to the server: `o=myorg`
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=myorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

Caution • *The Login Name is a required field; you cannot leave the Login Name field blank and specify the login name in the Context field using the* `cn` *parameter.*

Specifying Multiple Parameters in the Context Field of the ZENworks Server Distribution Panels

In the Context field, you can enter multiple qualified parameters in any order (separated by either periods or commas):

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-7 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: `myorg`	The following parameter is passed to the server: `o=myorg`
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=myorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

Note • *For all distribution types, the Distribution Wizard will create a Distribution log file in the Distribution folder of the* `AdminStudio Shared` *directory.*

Preparing for ZENworks Server Distribution

AdminStudio 10.0 SP2 | Distribution Wizard

The Distribution Wizard supports the distribution of a setup along with any transforms and files via ZENworks server distribution. You would select this option if you want to create an MSI Distribution Object to distribute to ZENworks servers.

Note • *If you are using ZENworks Configuration Manager Server 10, instead of using the standard Distribution Wizard, you should use the customized Distribution Wizard for ZENworks Configuration Management. See Preparing for ZENworks Configuration Management 10 Distribution for more information.*

Task: ***To prepare your package for ZENworks server distribution:***

1. Launch the Distribution Wizard. The Distribution Wizard **Welcome** panel opens.

   Click **Next**. The **Distribution Type** panel opens.

2. Select **ZENworks Server Distribution** from the **Distribution Type** list and click **Next**. The **Package Information** panel opens.

3. Click the Browse button and locate the Windows Installer Package (.msi) you want to distribute.

4. If there are transforms associated with the package, click the New button ( ) in the Additional Transforms area and navigate to the transform you want to add. Repeat as necessary.

5. After specifying the package location, click **Next**. The **ZENworks Login** panel opens.

6. Use this panel to login to the server location where you want to create the Distribution Object. Specify the following information:

   a. In the Login Name field, enter a valid Login Name for the Novell server where you want to create the Distribution Object. This is a required field. For more information, see Specifying Multiple Parameters in the Login Name and Context Fields of the ZENworks Login Panel.

   b. In the Password field, enter a valid password for the specified Login Name.

   c. In the Server field, enter the machine name or IP address of the Novell server where you want to create the Distribution Object.

   d. In the Context field, enter the Novell environment where you want to create the Distribution Object. Because you can choose to define the context information in the Login Name field, this field is optional. For more information, see Specifying Multiple Parameters in the Login Name and Context Fields of the ZENworks Login Panel.

   e. In the SSL Port Number field, specify the secure port to connect to ZENworks. The default/recommended port value is 636.

   All of these entries except the password are saved and will pre-populate these fields the next time you access this panel.

7. After you have entered the login information, click **Next**. The **ZENworks Server Distribution**/**Object** panel opens.

8. Use this panel to specify the ZENworks Distribution Object and its properties. Specify the following information:

   a. In the Name field, specify the Distribution Object's name.

b. In the Context field, enter the Novell environment of the Distribution Object. For more information, see Specifying Multiple Parameters in the Context Field of the ZENworks Server Distribution Panels.

9. Click **Next**. The **ZENworks Server Distribution**/**Distributor** panel opens.
10. Use this panel to specify the ZENworks Distributor Object for the Distribution Object. Specify the following information:

    a. In the Context field, enter the Novell environment where you want to create this Distributor Object. For more information, see Specifying Multiple Parameters in the Context Field of the ZENworks Server Distribution Panels.

    b. In the Distributor field, specify the Distributor Object for the Distribution Object. Click the Browse button (...) to select the Distributor from a list of Distributors within the current context (as specified on the **ZENworks Login** panel).

    c. In the Path to copy to Distributor's file system field, specify the desired path on the Distributor where you want to copy the setup files.

    d. In the Path to reference in the Distribution object field, specify the desired path as the location where you want to populate the distribution object. For example:

    Table 20-8 • Sample Operating System Distribution Object Paths

Operating System	Sample Distribution Object Path
Windows:	`c:\packages\msi_app`
Linux/Solaris:	`/packages/msi_app`
NetWare:	`data:\packages\msi_app`

11. Click **Next**. The **Distribution Summary** panel opens, listing the selections you made in the previous panels.
12. Review the information, and if satisfied, click **Next**. The **Distribution Output** panel displays progress during distribution.
13. Once the distribution finishes, click **Finish** to exit the Distribution Wizard.

Specifying Multiple Parameters in the Login Name and Context Fields of the ZENworks Login Panel

The entries that you make in the Login Name and Context fields are concatenated to form the Novell distinguished name. If you enter `jsmith` in the Login Name field and `myorg` in the Context field, the login information that AdminStudio would pass to the server would be:

```
cn=jsmith, o=myorg
```

Specifying Multiple Parameters in the Login Name Field

Instead of making entries in both the Login Name and Context fields, you could enter both the login name and the context in the Login Name field (separated by either periods or commas). AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-9 • How to Specify Multiple Parameters in the Login Name Field

If you enter ...	AdminStudio assumes that ...
two unqualified parameters in the **Login** field, such as: `jsmith, myorg`	• the first entry identifies the login name (**cn**) • the second entry identifies the organization parameter (**o**) Therefore, the following parameters are passed to the server: `cn=jsmith, o=myorg`
more than two unqualified parameters in the **Login** field, such as: `jsmith, dept01, office03, myorg`	• the first entry identifies the login name (**cn**) • the last entry identifies the organization parameter (**o**) • all entries in between the first and the last entry are assumed to be identifying organizational units (**ou**) Therefore, the following parameters are passed to the server: `cn=jsmith, ou=dept01, ou=office03, o=myorg`

In this instance, you would leave the Context field blank.

Specifying Multiple Parameters in the Context Field

If you would like to enter multiple parameters in the Context field, you would enter the login name in the Login Name field without parameters (`jsmith`, for example), and then specify all of the context parameters in the Context field. You can enter multiple qualified parameters in any order:

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-10 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: **myorg**	The following parameter is passed to the server: **o=myorg**

Table 20-10 • How to Specify Multiple Parameters in the Context Field (cont.)

If you enter ...	AdminStudio assumes that ...
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=myorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

Caution • *The Login Name is a required field; you cannot leave the Login Name field blank and specify the login name in the Context field using the* `cn` *parameter.*

Specifying Multiple Parameters in the Context Field of the ZENworks Server Distribution Panels

In the Context field, you can enter multiple qualified parameters in any order (separated by either periods or commas):

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-11 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: `myorg`	The following parameter is passed to the server: `o=myorg`

Table 20-11 • How to Specify Multiple Parameters in the Context Field (cont.)

If you enter ...	AdminStudio assumes that ...
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=myorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

Note • *For all distribution types, the Distribution Wizard will create a Distribution log file in the Distribution folder of the* `AdminStudio Shared` *directory.*

Deploying InstallScript MSI Installations

AdminStudio 10.0 SP2 | Distribution Wizard

When deploying an InstallScript MSI installation, the file `setup.exe` needs to be deployed with the InstallScript `.msi` installation file. The `setup.exe` file is required because it launches a file (`isscriptn.msi`) that installs the InstallScript engine required to run the InstallScript code. The *n* in `installscriptn.msi` indicates the version of the InstallScript engine that was used to create the InstallScript MSI installation.

If you want to deploy an InstallScript MSI installation without using `setup.exe`, such as when using Active Directory, you need to first deploy the same version of the InstallScript engine that was used to build the InstallScript MSI installation.

Installing the InstallScript Engine

Sometimes the `isscriptn.msi` file (the file that installs the InstallScript engine) is located in the same directory as the InstallScript `.msi` file. However, in some instances, the `isscriptn.msi` file is compressed within the `setup.exe` file and cannot be accessed.

If the `isscriptn.msi` file is compressed within the `setup.exe` file, you have the following options:

- **If you know which version** of the InstallScript engine was used to create your InstallScript MSI installation, you can get a copy of the InstallScript engine from the AdminStudio installation CD. All the major releases of the InstallScript engine are available in the `InstallScript_Engines` folder on the AdminStudio installation CD.

  Note • *For more information, see the Update to the Latest InstallShield Installation Engines Knowledge Base article at*

  http://consumer.installshield.com/kb.asp?id=Q108322

- **If you do not know which version** of the InstallScript engine was used to create your InstallScript MSI installation, contact the software vendor to find out the exact version.

Deploying an InstallScript MSI Installation

To deploy an InstallScript MSI installation, configure the setup and the target system in the following manner:

Task: ***To deploy an InstallScript MSI installation:***

1. Run the `isscriptn.msi` file to install the appropriate InstallScript engine on the target machine (where *n* is the version of the InstallScript engine that was used to create the application's InstallScript MSI installation).
2. Create a transform for the InstallScript MSI that includes the following changes:
   a. Add the property `ISSETUPDRIVEN` to the property table via the Direct Editor and give it a value of `1`.
   b. Add a condition to the `OnCheckSilentInstall` custom action in the `InstallExecuteSequence` via the Direct Editor that will always resolve to false or remove the custom action from the sequence.
   c. Make any additional changes in the transform, such as populating the serial number, modifying shortcuts or feature states, depending upon your organization's needs and the features and requirements of the application.
3. Next deploy the InstallScript `.msi` package.

If deploying the package via Active Directory, make sure that you set the **Installation User Interface** to **Basic** and specify any transforms that you created for the InstallScript MSI package.

Reference

The Distribution Wizard provides a straightforward way to distribute a Windows Installer package (.msi)—including any associated transforms or patches—to a network location, an FTP server, an administrative location, or using virtually any distribution system.

The Distribution Wizard is comprised of two separate wizards:

- Distribution Wizard
- Distribution Wizard for ZENworks Configuration Management

Distribution Wizard

The Distribution Wizard—which is used to distribute packages to a network location, an FTP server, an administrative location, or using virtually any distribution system—consists of the following panels:

- Welcome Panel
- Distribution Type Panel
  - Administrative Install Panel
  - Connect to a Microsoft System Center Configuration Manager Server Panel
  - Select Destination Folder
  - Select Group
  - Altiris Integration Panel
  - FTP Location Panel
  - LANDesk Integration Panel
  - ManageSoft Distribution Settings Panel
  - Marimba Panels
  - Network Location Panel
  - SMS Distribution Panel
  - Tivoli Integration Panel
  - ZENworks Login Panel
  - ZENworks Desktop Application Panel
  - ZENworks Server Distribution/Object Panel
  - ZENworks Server Distribution/Distributor Panel
- Package Information Panel
- Distribution Summary Panel
- Distribution Output Panel

Welcome Panel

The Distribution Wizard provides a straightforward way to distribute a Windows Installer package (.msi)—including any associated transforms—or a Microsoft App-V package (.sft) to a network location, an FTP server, an administrative location, a Marimba channel, via a Tivoli or SMS software package definition file, to a Microsoft Configuration Manager Server, to a ZENworks Configuration Management server, to a ZENworks desktop application, to a ZENworks server, or using Altiris, LANDesk, or ManageSoft.

Distribution Type Panel

From the **Distribution Type** panel, you can select the distribution method you want to use. You can choose one of the following distribution methods:

Table 20-12 • Distribution Wizard Distribution Types

Method	Description
Administrative Install	The installation is copied to a network directory using the administrative install option provided by Windows Installer. See Creating Administrative Installations for Packages.
Altiris	Prepare for Altiris distribution. See Preparing for Altiris Distribution.
FTP Location	Distribute to an FTP server, providing both your user name and password. See Distributing Packages to FTP Servers.
LANDesk	Distribution for LANDesk involves copying the setup files to a network location. See Preparing for LANDesk Distribution.
ManageSoft	Create a new ManageSoft project for distribution. See Preparing for ManageSoft Distribution.
Marimba	A new Marimba channel is created and files are copied to it for distribution. See Using Marimba Channel Publishing for Package Deployment.
Network Location	Distribute into a network directory. See Distributing Packages to Network Locations.
SMS	The Distribution Wizard will create a Package Definition File. SMS 1.2 or earlier uses a .pdf file extension, and SMS 2.0 or later uses an .sms file extension. See Preparing for SMS Distribution.
Configuration Manager	The Distribution Wizard will publish the selected package to Microsoft System Center Configuration Manager. ***Note*** • *To configure distribution points, connections, or advertisements, or to use a PDF file for distribution, you could also use the Configuration Manager Web Console to perform package distribution. See Distributing Packages Using Configuration Manager Web Console for more information.*
Tivoli	The Distribution Wizard copies the MSI, any MSTs, and the Tivoli SPD file to the network location specified. See Preparing for Tivoli Distribution.

Table 20-12 • Distribution Wizard Distribution Types (cont.)

Method	Description
ZENworks Configuration Management Distribution	Create an MSI distribution object to distribute on ZENworks Configuration Management. See Preparing for ZENworks Configuration Management 10 Distribution.
ZENworks Desktop Application	Create an MSI distribution object to distribute to ZENworks desktops. See Preparing for ZENworks Desktop Application Distribution.
ZENworks Server Distribution	Create an MSI distribution object to distribute to ZENworks servers. See Preparing for ZENworks Server Distribution.

Click **Next** to proceed to the associated distribution method panel.

Administrative Install Panel

This panel is displayed if you selected the Administrative Install distribution method from the **Distribution Type** panel. The installation will be copied to the network directory using the Windows Installer administrative install option.

In addition to specifying or browsing to the location where to perform the administrative installation, you can also select the Use short file names option to force the administrative installation to use the 8.3 file name convention (using the SHORTFILENAMES property).

Connect to a Microsoft System Center Configuration Manager Server Panel

On this panel, which is displayed if you selected **Configuration Manager** on the **Distribution Type** panel, you are prompted to enter connection information for your Microsoft Configuration Manager Server.

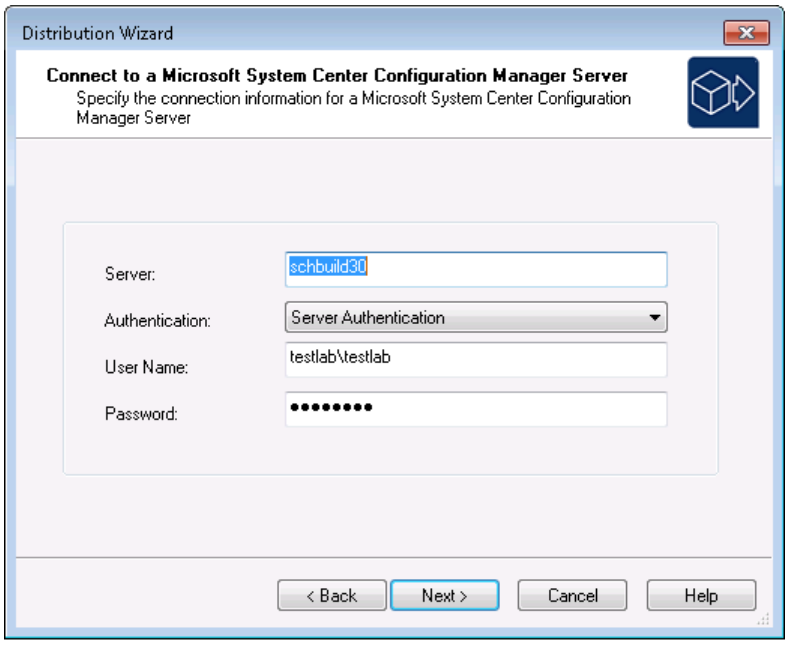


Figure 20-1: Distribution Wizard Connect to a Microsoft System Center Configuration Manager Panel

Enter the following information and click **Next** to continue.

Table 20-13 • Connect to a Microsoft System Center Configuration Manager Server Panel

Property	Description
Server	Enter the name of the Configuration Manager server that you want to connect to.
Authentication	From this list, select one of the following options: • **Windows Authentication**—Select if you want to use the credentials of the logged in user to login to the server. • **Server Authentication**—Select if you want to connect to the server using the specified **User Name** and **Password**.

Select Destination Folder

On this panel, which is displayed if you selected **Configuration Manager** on the **Distribution Type** panel, you are prompted to select a location that the Microsoft Configuration Manager Server has access to where you want to publish the selected packages.

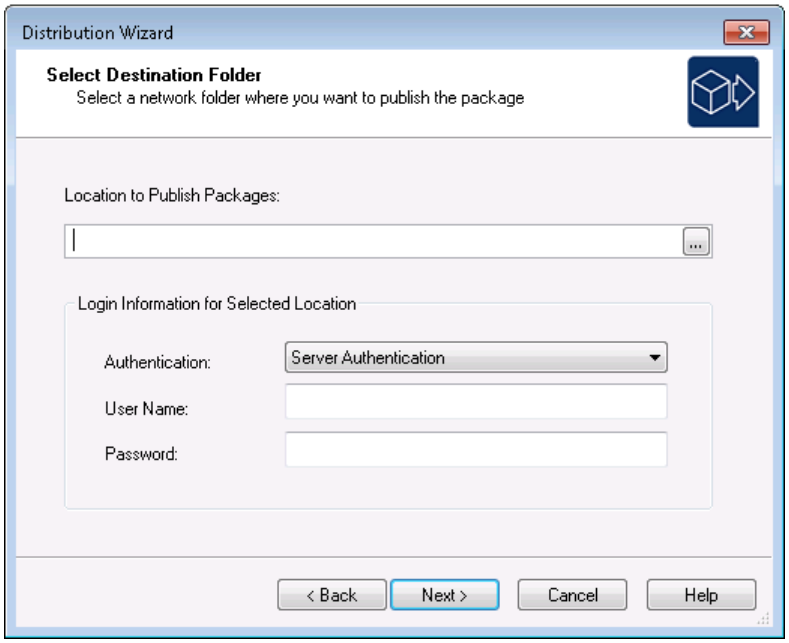


Figure 20-2: Distribution Wizard Select Destination Folder Panel

Enter the following information and click **Next** to continue.

Table 20-14 • Select Destination Folder Panel

Property	Description
Location to Publish Packages	Enter a target path, in UNC format (`\\Server\Share`), of the location where you want to publish the selected packages. Make sure that you enter a location that the Microsoft Configuration Manager Server has access to.
Authentication	From the **Authentication** list, select one of the following options: • **Windows Authentication**—Choose this option if you want to use Windows network authentication (your network login ID) to log into this location. • **SCCM Authentication**—Choose this option if you want to use Microsoft Configuration Manager Server authentication (your Microsoft Configuration Manager Server login ID) to log into this location. • **Server Authentication**—Choose this option if you are publishing to an alternate file server that requires credentials. Then enter the appropriate **User Name** and **Password**.

Select Group

On this panel, which is displayed if you selected **Configuration Manager** on the **Distribution Type** panel, select the **Target Group** on the Configuration Manager Server where you want to publish the package and click **Next** to continue.

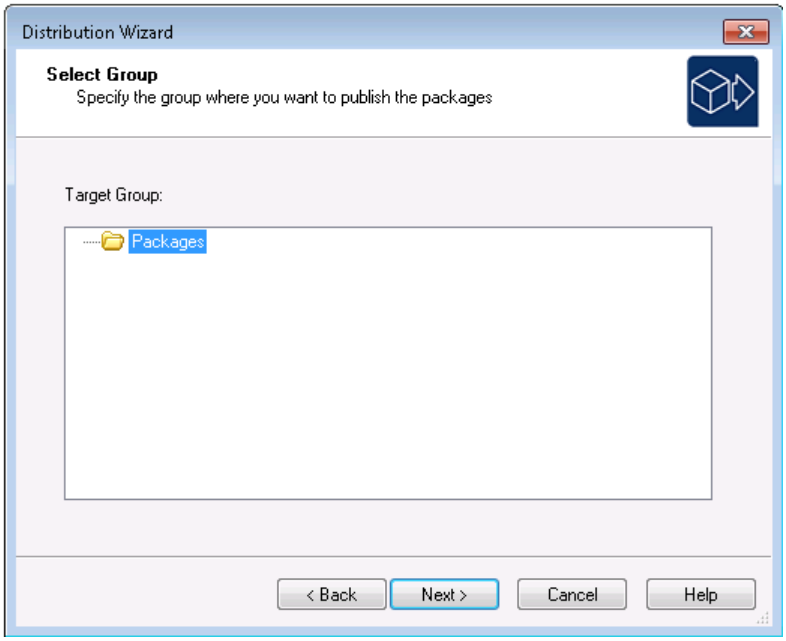


Figure 20-3: Distribution Wizard Select Group Panel

Altiris Integration Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel is displayed if you selected the Altiris distribution method from the Distribution Type panel. You would choose the Altiris method to create a package in the Altiris Notification Server.

Table 20-15 • Altiris Integration Panel

Field	Description
Network Directory	Specify or browse to the network location where you want to store the installation package. The Distribution Wizard remembers the last Network Location that is entered and displays it the next time this panel is accessed. The Distribution Wizard will copy the Windows Installer package along with any transforms and files to the UNC path specified. Also, the Distribution Wizard will use an XML template file (AltirisTemplate.config) to create a custom script file in this location named <packageName>.Config. ***Note*** *• You can edit AltirisTemplate.config to customize it for your organization. The file, which is installed with AdminStudio, is located in the Templates folder of the AdminStudio Shared Directory. See Altiris XML Template for more information.*

Table 20-15 • Altiris Integration Panel

Field	Description
Windows Installer Command Line	Enter any additional properties that you want to pass to the Windows Installer. See the Windows Installer Property Reference for more information.
Altiris Server Location	Enter the http: address for the location of the Altiris Server. The Distribution Wizard remembers the last Altiris Server Location that is entered and displays it the next time this panel is accessed.
User Name	Enter a User Name to log onto the server entered in the Altiris Server Location field. The Distribution Wizard remembers the last User Name that is entered and displays it the next time this panel is accessed.
Password	Enter the password to log onto the server entered in the Altiris Server Location field.

Click **Next** to proceed to the **Package Information** panel.

Altiris XML Template

AdminStudio 10.0 SP2 | Distribution Wizard

When using the Altiris distribution method, a custom script file is required. If you select the Altiris method on the Distribution Type Panel, the **Distribution Wizard** uses an XML Template file (`AltirisTemplate.config`) to create a custom script file named `<packageName>.Config`. The Distribution Wizard copies this configuration file along with the Windows Installer package with any transforms and files to the Network Directory specified on the Altiris Integration Panel.

You can edit `AltirisTemplate.config` to customize it for your organization. The file, which is installed with AdminStudio, is located in the Templates folder of the `AdminStudio Shared` directory.

The following variables are used in the `AltirisTemplate.Config` file:

Table 20-16 • AltirisTemplate.config Variables

Variable	Value
%DIST.ASVERSION%	AdminStudio version number
%DIST.COMMANDLINE%	Command line specified on the Altiris Integration Panel of the Distribution Wizard. See the Windows Installer Property Reference for more information.
%DIST.NETWORKLOCATION%	Network location of the MSI package as specified on the Altiris Integration Panel of the Distribution Wizard
%ProductCode%	ProductCode property from the MSI Property Table

Table 20-16 • AltirisTemplate.config Variables (cont.)

Variable	Value
%ProductName%	ProductName property from the MSI Property Table
%ProductVersion%	ProductVersion property from the MSI Property Table
%SUMMARYSTREAM.Id%	Comments property from the MSI Summary Stream. Any property with the SUMMARYSTREAM prefix will be populated based on the MSI Summary Information Stream Property as specified by the Id, in the format of: `<description>%SUMMARYSTREAM.4%</description>` Summary Stream Ids range from 1 to 19. For a complete list of Summary Information Stream Ids, see Summary Information Stream Property Set. In the example above, "4" indicates that the value of the Author property should be inserted.

Please note the following:

- Any property with the DIST prefix will be custom populated by the Distribution Wizard.
- Any other property will be populated based on the MSI Property Table.
- Typically, all variables are enclosed within '%' characters, as shown above.

FTP Location Panel

This panel is displayed when you select FTP Location as the distribution method from the **Distribution Type** panel. The installation will be upload to the FTP server specified in the **FTP Location** field. If necessary, provide a user name and password for the FTP server.

LANDesk Integration Panel

AdminStudio 10.0 SP2 | Distribution Wizard

With LANDesk distribution, the MSI package along with all the setup files are copied to a network location.

If you select LANDesk on the **Distribution Type** panel, the LANDesk Integration panel is displayed.

Specify the following options on the LANDesk Integration panel:

Table 20-17 • LANDesk Integration Panel Options

Option	Description
Network Directory or URL	Specify the network location where you want to copy the MSI package and all of its setup files. The Network Directory could be a URL or a UNC path. This field will default to the last used path, and will provide a most recently used list.

ManageSoft Distribution Settings Panel

AdminStudio 10.0 SP2 | Distribution Wizard

If you select ManageSoft on the **Distribution Type** panel, the ManageSoft Distribution Settings panel is displayed. You would choose this method to convert a Windows Installer .msi package directly into a ManageSoft-wrapped .msi package for deployment to managed devices. This will also create the necessary ManageSoft .ndp file.

Caution • *To use the ManageSoft distribution method in the Distribution Wizard, your system must satisfy the requirements specified in ManageSoft 7.2 MGS.500.006 PRD as well as the AdminStudio system requirements.*

Specify the following options on this panel:

Table 20-18 • ManageSoft Distribution Settings Panel Options

Option	Description
Configure MSI Package Source	Choose one of the following options: • **Install From Cache**—Configure the ManageSoft-wrapped package to be installed from a locally managed device cache. • **Install From Server**—Configure the ManageSoft-wrapped package to be installed from a Server. • **Install Source based upon Managed Device's Configuration**—Configure the ManageSoft-wrapped package to be installed from either a Server or device cache, depending on the package's configuration.
Retain the ManageSoft wrapped .msi package on local managed device cache	By default, the ManageSoft wrapped .msi package is deleted from the local cache after installation. Select this option of you do not want the Distribution Wizard to delete the ManageSoft wrapped .msi package from the local cache after installation. You might want to select this option if you want to reinstall the ManageSoft wrapped .msi package on another machine. This option is enabled when you select Install Source based upon Managed Device's Configuration from the Configure MSI Package Source list.

Table 20-18 • ManageSoft Distribution Settings Panel Options (cont.)

Option	Description
Install MSI package with elevated privileges	Select this option if you need elevated privileges to install the ManageSoft wrapped `.msi` package due to lack of user privileges in your environment. Usually this option is not selected because elevated privileges are not necessary for most installations to complete successfully.

ManageSoft Package and Environmental Settings Panel

AdminStudio 10.0 SP2 | Distribution Wizard

If you select ManageSoft on the **Distribution Type** panel of the Distribution Wizard, this panel opens after you have completed the **ManageSoft Distribution Settings** panel. On this panel you are prompted to specify the package settings, and select the supported environments for this package.

Specify the following options on this panel:

Table 20-19 • ManageSoft Package and Environmental Settings Panel Options

Option	Description
Manufacturer	This will be populated by default with the Manufacturer record value in the Property table from within the original .msi package. This field can be modified.
Package	This will be populated by default with the ProductName record value in the Property table from within the original .msi package. This field can be modified.
Application Name	This will be populated by default with the ProductName record value in the Property table from within the original .msi package. This field can be modified.
Version	This field will be blank by default. This field can be modified.
Supported Environments	Select those environments where this ManageSoft-wrapped package will be deployed.

Note • *All fields except Version are required to be populated in order to proceed to the next dialog. The Next button will not be enabled unless all required fields are populated.*

Marimba Panels

AdminStudio 10.0 SP2 | Distribution Wizard

The following panels are displayed when you select the Marimba distribution type from the **Distribution Type** panel:

- Patches Panel
- Marimba Integration Panel
- Advanced Marimba Options Dialog Box

Note • *For more information, see Using Marimba Channel Publishing for Package Deployment.*

Patches Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel opens when you have selected Marimba from the **Distribution Type** panel of the Distribution Wizard.

In the Patches (.msp) area:

- If there are patches associated with the package, click the New button ( ) and navigate to the patch you want to add.
- Use the Up and Down arrows ( ) to set the order in which the patches are applied to the package.
- Use the Delete button ( ) to delete a patch from the list.

Marimba Integration Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel is displayed if you selected the Marimba distribution method from the **Distribution Type** panel. The information you specify in this panel enables you to create a directory to store contents for the new Marimba channel.

This panel contains the following options:

Table 20-20 • Marimba Integration Panel Options

Option	Description
Specify the directory where you want to place the contents of the new channel	Specify or browse to the directory where you want to place the contents of the new channel.

Table 20-20 • Marimba Integration Panel Options (cont.)

Option	Description
Channel Name	This field is populated based on the .msi and .mst package name(s) selected: • **If no transforms are applied**, this field is populated with the .msi file name (base name only, with .msi removed). • **If a transform is applied**, this field is populated with the main transform name (base name only, with .mst removed). (The first transform listed in the Additional transforms field of the Package Information Panel is considered the main transform.) You can edit this field before continuing. The Channel Name will be created when the Marimba Channel is created.
Channel Description	This field is populated based on information in the .msi and .mst packages. You can edit this field before continuing.
Channel Version	This field is populated based on information in the .msi and .mst packages. You can edit this field before continuing. You can edit this field before continuing.
Include all the files that are in the same directory as the Windows Installer package	Select to include all files in the same directory as the installation package in the new directory. If this option is selected, all the files and folders in the same directory as the MSI package will be copied into the Channel Directory.
Specify the destination channel URL on the transmitter	This field defines the destination channel URL and subdirectory on the transmitter. This is an optional field. However, if you specify a value in this field, you must also specify the channel copier URL.
Advanced	Click to access the Advanced Marimba Options Dialog Box, where you can specify advanced Marimba options for Channel Publishing and Channel Creation.

Click **Next** to proceed.

Advanced Marimba Options Dialog Box

AdminStudio 10.0 SP2 | Distribution Wizard

On this dialog box, which is accessed by clicking the **Advanced** button on the **Marimba Integration** panel, you can specify advanced Marimba options for Channel Creation and Channel Publishing.

Specify the following options:

Table 20-21 • Advanced Marimba Options Dialog Box Options

Option	Description
Specify the Application Package URL	Accept the default application package URL or enter a new one.
Specify the XML template file	The XML template file sets properties for the MSI file during Marimba distribution. Accept the default XML template file that was shipped with AdminStudio, or enter a new one.
Specify the channel copier URL	If you have specified a destination channel URL on the transmitter on the **Marimba Integration** panel, then specify the URL for the channel copier in this field.

Network Location Panel

This panel is displayed when you choose the **Network Location** or **SMS** distribution methods from the **Distribution Type** panel. The installation files will be copied to the network directory you specify (or browse to) in this panel.

SMS Distribution Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel is displayed if you selected the SMS distribution method from the **Distribution Type** panel.

For SMS distribution, you create an SMS PDF (package definition file): a text file that contains all of the information necessary to create a software distribution package for an application.

Note • *For detailed information on SMS template files and the variables they use, see SMS File Templates.*

Table 20-22 • SMS Distribution Panel Options

Option	Description
Create Package Definition File (SMS 1.2 or earlier)	Select this option if you are using SMS 1.2 or earlier (which uses a .pdf file extension).
Create SMS File (SMS 2.0 or later)	Select this option if you are using SMS 2.0 or later (which uses a .sms file extension).

Table 20-22 • SMS Distribution Panel Options (cont.)

Option	Description
Install MIF Filename	If your installation will be run in a Microsoft Systems Management Server (SMS) environment, you can create a MIF (Management Information Format) file in the Windows directory to describe your application. In this text box, insert a filename for this MIF file. This is an optional step. ***Note*** • *This field is enabled when the Create SMS file (SMS 2.0 or later) option is selected.*

Tivoli Integration Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel is displayed if you selected the Tivoli distribution method from the **Distribution Type** panel. You can specify or browse to the network location where you want to store the installation package. All installation files—including the Tivoli SPD file—will be copied to this location.

Click **Next** to proceed.

Tivoli Settings Panel

AdminStudio 10.0 SP2 | Distribution Wizard

The **Tivoli Settings** panel appears when you select Tivoli on the **Distribution Type** panel. You use the **Tivoli Settings** panel to specify the directory from which you want to launch the product installation, enter a source image path to bundle the images in the software packages, and specify compression settings.

Specify the following options:

Table 20-23 • Tivoli Settings Panel Options

Option	Description
Target Image Path	Specify the directory from which you want to launch the product installation.
Redirect installation to a remote system	When this option is selected, product images are not included in the Tivoli package because they already exist in an accessible location with the installation.
Source Image Path	Specify the path of the directory where product images are stored. When distribution occurs, these product images are retrieved and distributed to the target systems. This path is only necessary for a bundled installation, where image files are distributed with the package.

Table 20-23 • Tivoli Settings Panel Options (cont.)

Option	Description
Keep Images	Select this option to retain the installation images on the target systems after installation.
Available to All Users	Select this option if the product should be made available to all target system users. If not selected, the product will only be made available to logged-on users when the installation is run.
Compression	Specify whether you want to use deflated or stored compression for all files and directories included in the Tivoli package.

SPD Parameters Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel allows you to set the install action for each feature in the package you are distributing. These settings will be recorded in the Software Package Definition File (SPDF).

The following install actions are possible when you click on the button to the left of the feature, although some may not be available depending on whether the feature supports the functionality:

Table 20-24 • SPD Parameters Panel Install Actions

Install Action	Description
Set feature to local	The feature will be installed when the setup is run.
Set feature and subfeatures to local	The feature and all its subfeatures will be installed when the setup is run.
Set feature to source	The feature will be run from the source media.
Set feature and subfeatures to source	The feature and subfeatures will be run from the source media.
Set feature to advertised	The feature will be marked as advertised when the setup is run.
Set feature and subfeatures to advertised	The feature and all its subfeatures will be marked as advertised when the setup is run.
Set feature to absent	The feature will not be installed when the setup is run.

Note • *For sample input and output of an SPD file, and a description of the variables used in an SPD file, see SPD File Details.*

SPD File Details

AdminStudio 10.0 SP2 | Distribution Wizard

A Software Package Definition File (SPDF) named template.spd will be used when creating a new SPD file (from C:\AdminStudio Shared\Templates). The .SPD file will be named after the msi file name.

Any instances of predefined variables in the file will be replaced, as described in the following table. Below the table, a Tivoli Sample Template File and a Tivoli Sample Output File are provided.

Note • *In an SPD file, variables are case sensitive.*

Table 20-25 • SPD File Variables

Variable	Value	SPD Location
%CREATION_TIME%	Current date/time	`package\creation_time`
%MODIFICATION_TIME%	Current date/time	`package\last_modification_time`
%ALLUSERS%	y or n	`package\install_msi_product\ all_users`
%REDIRECT%	y or n	`package\install_msi_product\ is_image_remote`
%KEEP_IMAGES%	y or n	`package\install_msi_product\ keep_images`
%COMPRESSION%	stored or deflated	`package\install_msi_product\ compression_method`
%IMAGE_DIR%	Image folder path	`package\install_msi_product\ image_dir`
%SOURCE_DIR%	Source folder path	`package\install_msi_product\ source_dir`
%CAPTION%	MSI file path	`package\install_msi_product\ caption`
%PACKAGE_FILE%	MSI file name	`package\install_msi_product\ package_file`
%PRODUCTCODE%	MSI ProductCode	`package\install_msi_product\ product_code`
%PRODUCTNAME%	MSI ProductName	`package\install_msi_product\ product_name`
%VERSION%	MSI ProductVersion	`package\install_msi_product\  product_version`
%FEATURES%	Name and install action for each MSI feature	`package\install_msi_product\`

Table 20-25 • SPD File Variables (cont.)

Variable	Value	SPD Location
%TRANSFORMS%	Sets TRANSFORMS property in MSI command line using transforms specified in Wizard.	package\install_msi_product\ properties

Tivoli Sample Template File

```
"TIVOLI Software Package v4.1 - SPDF"

package
   name = Noname
   title = "No title"
   version = 1.0
   web_view_mode = hidden
   undoable = o
   committable = o
   history_reset = n
   save_default_variables = n
   creation_time = %CREATION_TIME%
   last_modification_time = %MODIFICATION_TIME%
   move_removing_host = y
   no_check_source_host = y
   lenient_distribution = n
   default_operation = install
   server_mode = all
   operation_mode = not_transactional
   post_notice = n
   before_as_uid = 0
   skip_non_zero = n
   after_as_uid = 0
   no_chk_on_rm = y
   versioning_type = swd
   package_type = refresh
   stop_on_failure = y

   install_msi_product
      caption = %CAPTION%
      image_dir = %IMAGE_DIR%
      source_dir = %SOURCE_DIR%
      is_image_remote = %REDIRECT%
      keep_images = %KEEP_IMAGES%
      compression_method = %COMPRESSION%
      package_file = %PACKAGE_FILE%
      reinstall_mode = file_equal_version,user_data,machine_data,shortcut
      log_mode = disabled,fatal_exit,error
      report_log = n
      ui_level = none
```

```
      product_code = %PRODUCTCODE%
      product_name = %PRODUCTNAME%
      product_version = %VERSION%
      all_users = %ALLUSERS%
      properties = '%TRANSFORMS%'

%FEATURES%
   end
end
```

Tivoli Sample Output File

```
"TIVOLI Software Package v4.2 - SPDF"

package
   name = "MSI package"
   title = "No title"
   version = 1.0
   web_view_mode = hidden
   undoable = o
   committable = o
   history_reset = n
   save_default_variables = n
   creation_time = "2002-10-09 23:35:22"
   last_modification_time = "2002-10-09 23:46:33"
   move_removing_host = y
   no_check_source_host = y
   lenient_distribution = n
   default_operation = install
   server_mode = all
   operation_mode = not_transactional
   post_notice = n
   before_as_uid = 0
   skip_non_zero = n
   after_as_uid = 0
   no_chk_on_rm = y
   versioning_type = swd
   package_type = refresh
   stop_on_failure = y

   install_msi_product
      caption = C:\miky\adminpak.msi
      image_dir = C:\miky
      source_dir = C:\miky
      is_image_remote = n
      keep_images = n
      compression_method = stored
      package_file = adminpak.msi
      reinstall_mode = file_equal_version,user_data,machine_data,shortcut
      log_mode = disabled,fatal_exit,error
      report_log = n
      ui_level = none
      product_code = {B7298620-EAC6-11D1-8F87-0060082EA63E}
      product_name = "Windows 2000 Administration Tools"
      product_version = 5.0.0.0000
      all_users = y
```

```
      properties = 'TRANSFORMS=orca.mst'

      feature
         name = Run Time Libraries
         action = default_action
      end
      feature
         name = MSI Validation Support
         action = default_action
      end
      feature
         name = Merge Module Support
         action = default_action
      end
   end
end
```

ZENworks Login Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel is displayed if you selected the ZENworks Desktop Application or ZENworks Server Distribution method from the **Distribution Type** panel.

Use this panel to login to the server location where you want to create the Distribution Object. Enter the following options:

Table 20-26 • ZENworks Login Panel Options

Option	Description
Login Name	Enter a valid Login Name for the Novell server where you want to create the Distribution Object. This is a required field. For more information, see Specifying Multiple Parameters in the Login Name and Context Fields.
Password	Enter a valid password for the specified Login Name.
Server	Enter the machine name or IP address of the Novell server where you want to create the Distribution Object.
Context	Enter the Novell environment where you want to create the Distribution Object. Because you can choose to define the context information in the Login Name field, this field is optional. For more information, see Specifying Multiple Parameters in the Login Name and Context Fields.
SSL Port Number	Specify the secure port to connect to ZENworks. The default/recommended port value is 636.

Note • *All of these entries except the password are saved and will pre-populate these fields the next time you access this panel.*

Specifying Multiple Parameters in the Login Name and Context Fields

The entries that you make in the Login Name and Context fields are concatenated to form the Novell distinguished name. If you enter `jsmith` in the Login Name field and `myorg` in the Context field, the login information that AdminStudio would pass to the server would be:

```
cn=jsmith, o=myorg
```

Specifying Multiple Parameters in the Login Name Field

Instead of making entries in both the Login Name and Context fields, you could enter both the login name and the context in the Login Name field (separated by either periods or commas). AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-27 • How to Specify Multiple Parameters in the Login Name Field

If you enter ...	AdminStudio assumes that ...
two unqualified parameters in the **Login** field, such as: `jsmith, myorg`	• the first entry identifies the login name (`cn`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `cn=jsmith, o=myorg`
more than two unqualified parameters in the **Login** field, such as: `jsmith, dept01, office03, myorg`	• the first entry identifies the login name (`cn`) • the last entry identifies the organization parameter (`o`) • all entries in between the first and the last entry are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `cn=jsmith, ou=dept01, ou=office03, o=myorg`

In this instance, you would leave the Context field blank.

Specifying Multiple Parameters in the Context Field

If you would like to enter multiple parameters in the Context field, you would enter the login name in the Login Name field without parameters (`jsmith`, for example), and then specify all of the context parameters in the Context field. You can enter multiple qualified parameters in any order:

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-28 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: `myorg`	The following parameter is passed to the server: `o=myorg`
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=myorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

Caution • *The Login Name is a required field; you cannot leave the Login Name field blank and specify the login name in the Context field using the* `cn` *parameter.*

ZENworks Desktop Application Panel

AdminStudio 10.0 SP2 | Distribution Wizard

This panel is displayed (after you log in on the **ZENworks Login** panel) if you selected the ZENworks Desktop Application distribution method from the **Distribution Type** panel. You would select this option if you want to create an MSI Distribution Object to distribute to ZENworks desktops.

This panel includes the following options:

Table 20-29 • ZENworks Desktop Application Panel Options

Option	Description
Object Name	Specify the name of the Application Object to be created in the specified Novell eDirectory® tree. You can use the name of the .msi file or any other name you choose.
Version	Specify the Application Object's version. For a new Application Object, leave this field empty (in which case 0 is used). For existing Application Objects, enter an integer to identify the version.
Context	Enter the Novell environment where you want to create this Application Object. You can enter multiple context parameters in this field. See Specifying Multiple Parameters in the Context Field.
Administration Package Path	Use this field to specify the path that ConsoleOne® will use to access the .msi package. ConsoleOne uses the package file to populate information in the Application Object. This field defaults to the directory specified by the .msi file. If desired, you may edit this path. ***Note*** • *This field is used only by ConsoleOne to access the .msi package. It is not used by Novell® Application Launcher for distribution of the application. For distribution, Application Launcher uses the path defined in the SOURCE_PATH.*
SOURCE_PATH	Specify a list of network directories that are the desired location(s) for the package source path(s). You can specify one or more source paths. To add a path, click the Add ( ) button to open the Browse for Folder dialog box, and select the path. If you want to include multiple source paths, you can repeat the procedure as necessary. If you need to delete a path you have added, select the path and click the Delete button ( ).

Specifying Multiple Parameters in the Context Field

In the Context field, you can enter multiple qualified parameters in any order (separated by either periods or commas):

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-30 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: `myorg`	The following parameter is passed to the server: `o=myorg`
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=tcmyorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

Click **Next** to proceed.

ZENworks Server Distribution/Object Panel

AdminStudio 10.0 SP2 | Distribution Wizard

If you have selected the ZENworks Server Distribution method from the **Distribution Type** panel, this panel is displayed after you log in on the **ZENworks Login** panel. On this panel you specify the ZENworks Distribution Object and its properties.

This panel includes the following options:

Table 20-31 • ZENworks Server Distribution/Object Panel Options

Option	Description
Name	Specify the Distribution Object's name.

Table 20-31 • ZENworks Server Distribution/Object Panel Options (cont.)

Option	Description
Context	Enter the Novell environment of this Distribution Object. You can enter multiple context parameters in this field. See Specifying Multiple Parameters in the Context Field.

Specifying Multiple Parameters in the Context Field

In the Context field, you can enter multiple qualified parameters in any order (separated by either periods or commas):

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-32 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: `myorg`	The following parameter is passed to the server: `o=myorg`
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=myorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

ZENworks Server Distribution/Distributor Panel

AdminStudio 10.0 SP2 | Distribution Wizard

When you select the ZENworks Server Distribution method from the **Distribution Type** panel, this panel is displayed after the **ZENworks Server Distribution/Object** panel. On this panel, you specify the Distributor Object for the Distribution Object that you are creating.

This panel includes the following options:

Table 20-33 • ZENworks Server Distribution/Distributor Panel Options

Option	Description
Context	Enter the Novell environment where you want to create this Distributor Object. You can enter multiple context parameters in this field. See Specifying Multiple Parameters in the Context Field.
Distributor	Specify the Distributor Object for the Distribution Object that you are creating. Click the Browse button (...) to select the Distributor from a list of Distributors within the current context (as specified on the **ZENworks Login** panel).
Path to copy to Distributor's file system	Specify the desired path on the Distributor as the location where you want to copy the setup files.
Path to reference in the Distribution object	Specify the desired path as the location where you want to populate the distribution object. For example: • Windows: • c:\packages\msi_app • Linux/Solaris: • /packages/msi_app • NetWare: • data:\packages\msi_app

Specifying Multiple Parameters in the Context Field

In the Context field, you can enter multiple qualified parameters in any order (separated by either periods or commas):

```
ou=dept01, o=myorg, ou=office03
```

However, if you enter multiple unqualified parameters in the Context field, AdminStudio uses a formula to determine how to pass these parameters to the server:

Table 20-34 • How to Specify Multiple Parameters in the Context Field

If you enter ...	AdminStudio assumes that ...
one unqualified parameter in the **Context** field, such as: `myorg`	The following parameter is passed to the server: `o=myorg`
two unqualified parameters in the **Context** field, such as: `dept01, myorg`	• the first entry identifies an organizational unit (`ou`) • the second entry identifies the organization parameter (`o`) Therefore, the following parameters are passed to the server: `ou=dept01, o=myorg`
more than two unqualified parameters in the **Context** field, such as: `dept01, office03, myorg`	• the last entry identifies the organization parameter (`o`) • all previous entries are assumed to be identifying organizational units (`ou`) Therefore, the following parameters are passed to the server: `ou=dept01, ou=office03, o=myorg`

Caution • *If the above formulas do not accommodate your data, always use fully qualified naming.*

Click **Next** to proceed.

Package Information Panel

On the **Package Information** panel, you select the Windows Installer package that is ready for distribution.

Table 20-35 • Package Information Panel Options

Option	Description
Windows Installer Package (.msi)	Specify or browse to the Windows Installer package (.msi) that you want to distribute. If you launched the Distribution Wizard from the Application Manager by selecting a package and selecting **Distribute Package** from the context menu, the name in the **Windows Installer Package (.msi)** field is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository: • **Not in the Software Repository**—The full name and path of the Windows Installer file is displayed, and you can edit this entry or click **Browse** and select a different package. • **In the Software Repository**—Only the name of the Windows Installer file is displayed (not the full path) and this entry cannot be edited or changed. ***Edition*** • *The Software Repository is included in AdminStudio Enterprise Edition.*
Additional Transforms	In the Additional Transforms area: • If there are transforms associated with the package, click the New button ( ) and navigate to the transform you want to add. • Use the Up and Down arrows ( ) to set the order in which the transforms are applied to the package. • Use the Delete button ( ) to delete a transform from the list.
Specify Additional MSI Properties	[Marimba Only] Enter any public MSI properties that you want to set in the Windows Installer package. Use the following syntax: `PROP1=value1, PROP2=value2, PROP3=value3`

Click **Next** to proceed.

Distribution Summary Panel

The **Distribution Summary** panel displays a summary of all settings configured in the previous panels. When you click **Next**, the distribution begins and the **Distribution Output** panel is displayed.

Caution • *The distribution will overwrite the contents of the distribution folder.*

Distribution Output Panel

The **Distribution Output** panel displays a progress bar and status messages during distribution. When distribution completes, click **Finish** to exit the Wizard.

Note • *For all distribution types, the Distribution Wizard will create a Distribution log file in the Distribution folder of the* `AdminStudio Shared` *directory.*

Distribution Wizard for ZENworks Configuration Management

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

You can use the Distribution Wizard for ZENworks Configuration Management to distribute a Windows Installer package (`.msi`)—including any associated transforms—to ZENworks Configuration Management.

The Distribution Wizard for ZENworks Configuration Management consists of the following panels:

- Welcome Panel
- Login Panel
- Windows Installer Package Information Panel
- Bundle Creation Options Panel
- Bundle Information Panel
- Summary Panel
- Publishing Process Panel

Welcome Panel

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

You can use the Distribution Wizard for ZENworks Configuration Management to prepare a Windows Installer package (`.msi`)—including any associated transforms—for distribution on ZENworks Configuration Management.

If you do not want this panel to be displayed each time you open this wizard, select the **Do not show the Welcome panel again** option. If this option is selected, the **Login** panel will be the first panel opened for this wizard.

Click **Next** to continue.

See Also
Preparing for ZENworks Configuration Management 10 Distribution

Login Panel

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

On the **Login** panel, enter the login information for the ZENworks Configuration Management server that you want to distribute packages on, and click **Login** to proceed with the distribution process.

Enter the following information:

Table 20-36 • ZENworks Configuration Management Server Login Information

Property	Description
User Name	Enter a valid User Name for the ZENworks Configuration Management server where you want to distribute packages
Password	Enter a valid Password for the ZENworks Configuration Management server where you want to distribute packages

Server URL	Enter the server URL, machine name, or IP address of the ZENworks Configuration Management server using the following format:
	`http://www.servername.com` or `http://111.22.333.44`
	If you need to specify a specific port number, append the port number to the end of the URL, such as:
	`http://www.servername.com:123`
	If you are using SSL and you want a secure connection, change the `http` prefix to `https`. For example:
	`https://www.servername.com`

See Also
Preparing for ZENworks Configuration Management 10 Distribution

Windows Installer Package Information Panel

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

On the **Windows Installer Package Information** panel, enter the information that will be referenced by this ZENworks server bundle, and click **Next** to continue.

Enter the following information:

Table 20-37 • Windows Installer Package Information Panel Properties

Property	Description
Windows Installer Package file (.msi)	Click **Browse** and select the Windows Installer (`.msi`) package that you want to distribute.
Windows Installer Transform Files (.mst)	All of the `.mst` files that are in the same directory as the selected Windows Installer package are automatically listed in this list. To include transforms with the Windows Installer package, click the New button ( ) and select a transform. If the package requires multiple transforms, you can repeat the procedure as necessary. Use the Delete button ( ) to delete the selected transform from the list. ***Note*** • *All of the* `.mst` *files that are in the same directory as the selected Windows Installer package are automatically listed in this list even if they are not applicable to the selected package. To prevent the inclusion of non-applicable transform files, delete those transforms from the list.*
Install Parameters	You can customize how this package is installed by entering parameters in this field. These parameters are applied to `msiexec.exe` to perform the desired action. Any actions that you enter here will be performed whenever the bundle is installed. The root parameter, which should not be edited or deleted, is: `/i packagename.msi` By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input. ***Caution*** • *If this operation requires user input, either remove the* `/qn` *parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.* ***Note*** • *For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.*

Table 20-37 • Windows Installer Package Information Panel Properties (cont.)

Property	Description
Uninstall Parameters	Enter an action that will be performed whenever the bundle is uninstalled. The root parameter, which should not be edited or deleted, is: `/x packagename.msi` By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input. ***Caution*** • *If this operation requires user input, either remove the* `/qn` *parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.* ***Note*** • *For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.*

Table 20-37 • Windows Installer Package Information Panel Properties (cont.)

Property	Description
Repair Parameters	Enter an action that will be performed whenever the user chooses to repair the bundle by repairing or reinstalling missing or corrupted files. The root parameter, which should not be edited or deleted, is: `/f packagename.msi` By default, the `/qn` parameter is added to indicate that you want to perform this operation silently (with no user interface). This is the common operating behavior for installing software with ZENworks. Running an operation silently implies that it does not require any user input. ***Caution*** • *If this operation requires user input, either remove the* `/qn` *parameter, or create a response transform to preconfigure all user input. For more information, see Using Response Transforms.* You can also apply any of the following additional parameters after the package name: • **p** – Reinstalls a file if it is missing • **o** – Reinstalls a file if it is missing or if an older version of the file is present on the user's system • **e** – Reinstalls a file if it is missing or if an equivalent or older version of the file is present on the user's system • **c** – Reinstalls a file if it is missing or if the stored checksum of the installed file does not match the new file's value • **a** – Forces a reinstall of all files • **u** or **m** – Rewrite all required user registry entries • **s** – Overwrites any existing shortcuts • **v** – Runs your application from the source and re-caches the local installation package ***Note*** • *For additional parameters that can be added, see Additional Install, Uninstall, and Repair Parameters.*

Additional Install, Uninstall, and Repair Parameters

The following additional parameters can be entered in the Parameters fields.

Table 20-38 • Additional Parameters

Parameter	Description

`/j [u|m] packagename.msi`
`/j [u|m] packagename.msi /t <transform list>`
`/j [u|m] packagename.msi /g`
`/j <language ID>`
	Building with the `/j <package>` option advertises the components of your application on the end user's computer
	• **u** – Advertises components only to the current user
	• **m** – Advertises components to all users of the computer
	• **g** – Specifies language ID
	• **t** – Applies a transform to your advertised product
	Transforms allow the synchronization of applications across different languages. For example, if you upgrade the English version of your product, you could apply a transform to automatically upgrade the French version of your product.

`/L [i|w|e|a|r|u|c|m|p|v|+] <log file>`	Building with the `/L` option specifies the path to the log file. These flags indicate which information to record in the log file: • **i** – Logs status messages • **w** –Logs non-fatal warning messages • **e** – Logs any error messages • **a** – Logs the commencement of action sequences • **r** – Logs action-specific records • **u** – Logs user requests • **c** – Logs initial user interface parameters • **m** – Logs out-of-memory messages • **p** – Logs terminal settings • **v** – Logs the verbose output setting • **+** – Appends to an existing file • ***** – Is a wildcard character that allows you to log all information (excluding the verbose output setting)

Table 20-38 • Additional Parameters (cont.)

Parameter	Description
`/q [n|b|r|f]`	The `/q` option is used to set the user interface level in conjunction with the following flags: • **q** or **qn** – Creates no user interface • **qb** – Creates a basic user interface The user interface settings below display a modal dialog box at the end of installation: • **qr** – Displays a reduced user interface • **qf** – Displays a full user interface • **qn+** – Displays no user interface • **qb+** – Displays a basic user interface
`/y <filename>`	This command calls the `DllRegisterServer` entry-point function of the DLL or OCX file specified in `<filename>`.
`/z <filename>`	This command calls the `DllUnregisterServer` entry-point function of the DLL or OCX file specified in `<filename>`.
`TRANSFORMS`	Use the `TRANSFORMS` command-line parameter to specify any transforms that you would like applied to your base package. Your transform command-line call might look something like this: `msiexec /i "C:\Directory\ProductName.msi" TRANSFORMS="New Transform 1.mst"` You can separate multiple transforms with a semicolon. Because of this, it is recommended that you do not use semicolons in the name of your transform, as the Windows Installer service will interpret those incorrectly.
`Properties`	All public properties can be set or modified from the command line. Public properties are distinguished from private properties by the fact that they are in all capital letters. For example, `COMPANYNAME` is a public property. To set a property from the command line, use the following syntax: `PROPERTY=VALUE`. If you wanted to change the value of `COMPANYNAME`, you would enter: `msiexec /i "C:\Directory\ProductName.msi" COMPANYNAME="YourCompany"`

See Also
Preparing for ZENworks Configuration Management 10 Distribution

Bundle Creation Options Panel

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

On the **Bundle Creation Options** panel, specify whether you want to create a new bundle or overwrite an existing bundle. After making your selection, click **Next** to proceed.

You have the following options:

Table 20-39 • Bundle Creation Options

Option	Description
Create a new bundle from these Windows Installer package files	To create a new bundle to reference this Windows Installer package, select this option.
Update an existing bundle using these Windows Installer package files	If you want to overwrite an existing bundle to reference this Windows Installer package, select this option, and then select an existing bundle in the tree: • **Recommended Bundles**—This group lists the bundles that contain the same Windows Installer package as the one you selected on the Windows Installer Package Information Panel. • **All Other Bundles**—This group lists the rest of the existing bundles on the server.

See Also
Preparing for ZENworks Configuration Management 10 Distribution

Bundle Information Panel

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

On the **Bundle Information** panel, enter information to specify attributes for this bundle on ZENworks Configuration Management, and click **Next** to continue.

Enter the following properties:

Table 20-40 • Bundle Information Panel Properties

Property	Properties
Bundle Name	Enter the bundle's name as you want it to appears in ZENworks® Control Center (ZCC) and the ZENworks Application Launcher (on managed devices).
Version Number	Enter the bundle's version number. If you are overwriting an existing bundle, and you enter a higher version number than the bundle's original version number, the bundle will be redeployed.

Table 20-40 • Bundle Information Panel Properties (cont.)

Property	Properties
Icon	Click **Browse** and select a shortcut icon graphic (in `.ico`, `.gif`, or `.jpg` format) that ZENworks Application Launcher will display on managed devices. If you do not select an icon file, the standard ZENworks bundle icon will be used.
Folder	From the **Folder** list, select the folder path that will be used by ZENworks Application Launcher when displaying the bundle on either the device's desktop or **Start** menu. All of the folders defined on the ZENworks server are listed. For example: • **Start Menu**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the **Start** menu, ZENworks Application Launcher creates an `Application\Accounting` folder on the root of the **Start** menu and adds the bundle to it. • **Desktop**—If you specify `Applications\Accounting` as the path and choose to display the bundle on the desktop, ZENworks Application Launcher creates an `Applications\Accounting` folder on the desktop and adds the bundle to it. You can place multiple bundles in a single folder by specifying the same folder path for each of the bundles.
Description	Enter a description of the bundle. This description will be displayed in ZENworks® Control Center and the ZENworks Application Launcher (on managed devices).

See Also
Preparing for ZENworks Configuration Management 10 Distribution

Summary Panel

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

The **Summary** panel displays the options you have selected for distributing this Windows Installer package on ZENworks Configuration Management.

Click **Publish** to complete the distribution process or **Back** to change the listed options.

See Also
Preparing for ZENworks Configuration Management 10 Distribution

Publishing Process Panel

AdminStudio 10.0 SP2 | Distribution Wizard for ZENworks Configuration Management

The **Publishing Process** panel lists the progress messages while the bundle is being published on ZENworks Configuration Management.

- **ZENworks error messages**—Any error messages with a numeric prefix that appear on this panel are generated by ZENworks Configuration Management. To resolve these errors, contact your ZENworks Configuration Management System Administrator.
- **Cancelling publication**—If you want to cancel the publication of the bundle on ZENworks Configuration Management, click **Cancel**.
- **Exiting the wizard**—When processing is complete, the **Finish** button becomes enabled. Click **Finish** to exit this wizard.

See Also

Preparing for ZENworks Configuration Management 10 Distribution

21

Distributing Packages Using Configuration Manager Web Console

You can use Microsoft System Center Configuration Manager Web Console, a Web-based application, to configure package distribution options and distribute setup packages without going to Configuration Manager Server. By using the Configuration Manager Web Console, you can:

- Create and distribute packages to Configuration Manager Server from Configuration Manager Web Console.
- View the package deployment status from Configuration Manager Web Console after it is targeted to distribute using Configuration Manager.
- Choose the target audience for deployment by browsing through the Configuration Manager collections.
- Configure program options for packages and target multiple distribution points for packages.
- Refresh source files on the distribution points if source files are changed.
- Create access accounts for setup packages to secure source files.
- Delete the packages.

Information on how to use the Configuration Manager Web Console is presented in the following sections:

Table 21-1 • Configuration Manager Web Console Help Library

Section	Description
About Microsoft System Center Configuration Manager and Configuration Manager Web Console	Explains the concepts behind Microsoft System Center Configuration Manager and Configuration Manager Web Console and how to get started using Configuration Manager Web Console.

Table 21-1 • Configuration Manager Web Console Help Library

Section	Description
Distributing Packages Using the Configuration Manager Web Console	Explains how to configure package distribution options and distribute setup packages using Configuration Manager Web Console. • Configuring Distribution Settings • Distributing a New Package • Viewing the Status of a Distributed Package • Modifying the Distribution Settings of a Package • Deleting Packages
Configuration Manager Web Console Reference	Every view and dialog box available in Configuration Manager Web Console is discussed in this section.

For information on how to use the Configuration Manager Web Console, click on one the following topics:

- About Microsoft System Center Configuration Manager and Configuration Manager Web Console
- Distributing Packages Using the Configuration Manager Web Console
- Configuration Manager Web Console Reference

About Microsoft System Center Configuration Manager and Configuration Manager Web Console

AdminStudio is directly integrated with Microsoft System Center Configuration Manager via the Configuration Manager Web Console, which enables enterprises using the Configuration Manager distribution system to cut down software rollout times by greatly simplifying their distribution efforts enterprise-wide.

You can use Configuration Manager Web Console to prepare, publish, and distribute applications through your Configuration Manager system without ever touching the Configuration Manager Server console, helping you improve your application management efforts while simplifying your overall Configuration Manager management requirements. You can use Configuration Manager Web Console to perform all of your Configuration Manager administrative tasks, including publishing and advertising applications and retiring or deleting applications from circulation. Configuration Manager Web Console also allows the use of existing Configuration Manager collections to distribute your applications, which simplifies targeting.

Using the centrally managed Configuration Manager Web Console, you can better monitor any application's Configuration Manager deployment success using a single interface, simplifying the process and saving you time.

The following topics explain the concepts behind Microsoft Configuration Manager and Configuration Manager Web Console and how to get started using Configuration Manager Web Console.

- Distributing Software Through Configuration Manager Server
- Configuration Manager Web Console Overview
- Getting Started

Distributing Software Through Configuration Manager Server

Microsoft Configuration Manager is used by Systems Administrators to maintain and upgrade software for organizations with distributed networks. Configuration Manager enables you to upgrade and configure each computer from a central location or from multiple locations, and schedule software programs for distribution to specific computers.

Software distribution with Configuration Manager includes the following tasks:

Table 21-2 • Software Distribution with Microsoft Configuration Manager Server

#	Step	Description
1	**Install Client Agent Software on Client Computers**	Install the Advertised Programs Client Agent on the Client computers to enable them to receive and run programs from Configuration Manager Server (advertised programs).
2	**Prepare Site System Roles**	Prepare the following Site System Roles: • **Client Access Points**—A site system role that provides a communication point between the site server and Legacy Client computers (computers using a Windows operating system that is pre-Windows 2000). Computers contact CAPs to install and update Configuration Manager Legacy Client software. After Configuration Manager Legacy Client software has been installed on computers in a site, they contact a CAP for updated information from the Configuration Manager site server. • **Management Points**—Management points are the primary point of contact between Advanced Configuration Manager Clients and the Configuration Manager site server. • **Distribution Points**—Distribution points are site system roles that store software package files so clients can access them during the software distribution process. Clients contact distribution points to obtain source files when they run programs that are advertised to them through a Client Access point or a Management Point.

Table 21-2 • Software Distribution with Microsoft Configuration Manager Server

#	Step	Description
3	**Prepare Collections**	A collection is a customized management view of the resources in your organization. Collections serve as targets for performing software distribution on multiple resources at one time. An example of a collection is All Windows NT Workstation 4.0 Systems. The clients that need to receive the program must be members of a collection (referred to as the target collection).
4	**Create Software Distribution Packages**	Software Distribution Packages are applications packages that you are going to distribute via Configuration Manager Server. Package properties include the software name and version, the location of the package source folder, and group permissions for the distribution folder.
5	**Specify Distribution Points**	Select the Distribution Points that will receive the software.
6	**Create Programs**	Create a program for the package that is going to be distributed. Program properties specify the command line that will run on the clients, and the types of clients on which the program can run. A package can have multiple programs defined. For example, you can define a Full Installation, a Custom Installation, and a Minimum Installation for a single package.
7	**Create Advertisements**	Advertise the program that you want the clients to run. Advertising the program makes a program available to a specified target collection. The advertisement contains the name of the program, the name of the target collection, and the scheduling configuration (such as when to run the program or when will the program expire).

Configuration Manager Web Console Overview

Configuration Manager Web Console enables you to coordinate and manage the packaging, preparation, and distribution of applications across your enterprise via your Configuration Manager system without ever touching the Configuration Manager Server console. The following diagram explains the role that the Configuration Manager Web Console plays in Configuration Manager software distribution:

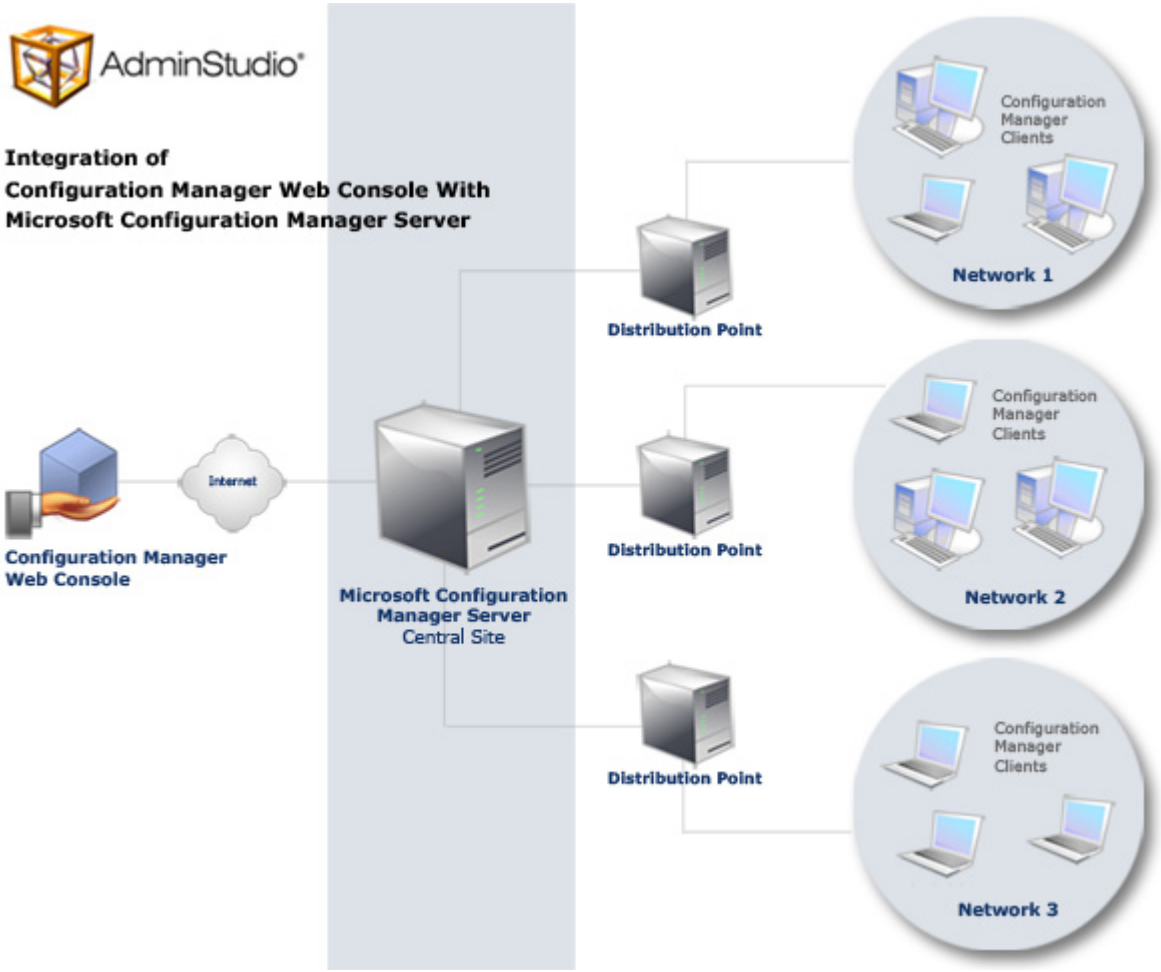


Figure 21-1: Configuration Manager Web Console and Configuration Manager Server Integration

Setting Configuration Options

In the Package Configuration area of Configuration Manager Web Console, you specify distribution configuration fields that the Configuration Manager Server utilizes in the distribution of the package. You can manually enter package and distribution configuration information and settings, or you can choose to attach a Package Definition File (`.pdf`) to a Windows Installer (`.msi`) package to pre-populate these fields. The following configuration information is specified:

- **Access Accounts**—Specify which administrators are allowed to distribute this particular package.
- **Distribution Points**—Identify which servers may act as distribution points for this particular package.

- **Programs**—Specify programs (that will run on the clients) to execute the distribution of the application, and the types of clients on which the program can run (directories, operating system versions, etc.)
- **Advertisements**—Identify the collection of users to receive this package, and specify when the package will be available for distribution and whether it is optional or mandatory.

See Setting Package Configuration Options for detailed information on setting these options.

Package Administration

The Package Administration area of Configuration Manager Web Console allows you to perform the following tasks:

- Viewing the Status of a Distributed Package
- Modifying the Distribution Settings of a Package
- Deleting Packages

Getting Started

To start using the Configuration Manager Web Console to distribute packages to Configuration Manager Server, you need to do the following:

- **Learn About Configuration Manager Web Console**—Review the About Microsoft System Center Configuration Manager and Configuration Manager Web Console section to learn how Configuration Manager Web Console is integrated with Configuration Manager Server.
- **Configure Distribution Settings**—Perform the steps in Configuring Distribution Settings to set the default Configuration Manager Server values for any new package you target for distribution.

You would then be ready to perform the steps in Distributing a New Package.

Distributing Packages Using the Configuration Manager Web Console

When you launch the Configuration Manager Web Console, the Microsoft System Center Configuration Manager Web Console Home Page opens listing three options: **Distribute a New Package**, **Package Administration**, and **Distribution Settings**. The following procedures explain how to use these options to configure package distribution options and distribute packages without using the Configuration Manager Server Console:

- Configuring Distribution Settings
- Distributing a New Package
- Viewing the Status of a Distributed Package
- Modifying the Distribution Settings of a Package
- Deleting Packages

Configuring Distribution Settings

You configure the Configuration Manager Server connection settings on the Distribution Settings Page.

Note • *The values entered on the* ***Distribution Settings*** *page will be used as the default values for any new package you target for distribution. However, you can change these values while configuring the package.*

Task: ***To configure distribution settings:***

1. Launch Configuration Manager Web Console. The **Configuration Manager Web Console** Home Page opens.
2. Click **Distribution Settings**. The **Distribution Settings** page opens.
3. In the **Site Server Name** field, enter the machine name on which Configuration Manager Server is installed. (This could be a local or remote machine.) When you make a selection in this field, the **Site Code** field is populated with a list of sites available on the Configuration Manager Server.
4. In the **Site Code** field, select the Configuration Manager site you want to use for software distribution.
5. In the **User Name** field, enter the user name of a user who has access to the Configuration Manager Server.
6. Click **Update** to save your entries
7. Click the **Home** link to return to the **Configuration Manager Web Console** Home Page.

Distributing a New Package

You can use Configuration Manager Web Console to create packages, configure package distribution options, and distribute setup packages without going to Configuration Manager Server. These package configuration settings are saved with the application in the Application Catalog, enabling you to easily redistribute the package using the same settings.

Distributing a package using Configuration Manager Web Console consists of the following tasks:

Table 21-3 • Configuration Manager Web Console Package Distribution Steps

#	Step	Description
1	Selecting a Package for Distribution	Select the package in your Application Catalog that you want to distribute, and choose whether you want to attach a Package Definition File (`.pdf`). When you use a Package Definition File, appropriate fields on the Configuration Manager Package Configuration Page are populated with the information in the package definition file you selected.

Table 21-3 • Configuration Manager Web Console Package Distribution Steps

#	Step	Description
2	Setting Package Configuration Options	In the Package Configuration area of Configuration Manager Web Console, you specify the following distribution configuration fields that the Configuration Manager Server utilizes in the distribution of the package. • Setting Package Settings Options—General information about the package, along with data source, data access, distribution settings, and reporting information. • Setting Access Accounts Options—Specify which administrators are allowed to distribute this particular package. • Setting Distribution Points Options—Specify which servers may act as distribution points for this particular package. A Distribution Point is a site system role that stores software package files so clients can access them during the software distribution process. • Setting Programs Options—Specify programs (that will run on clients) to execute the distribution of the application, and the types of clients on which the program can run (defined using directories, operating system versions, etc.). • Setting Advertisements Options—Identify the collection of users to receive this package, and specify when the package will be available for distribution and whether it is optional or mandatory.
3	Saving the Package on the Configuration Manager Server	Save the package on the Configuration Manager Server. You can then view the status of the package by following the instructions in Viewing the Status of a Distributed Package.

Selecting a Package for Distribution

To select a package for distribution from Configuration Manager Web Console using Configuration Manager Distribution, perform the following steps.

Task: ***To distribute a new package:***

1. Launch Configuration Manager Web Console. The **Configuration Manager Web Console** Home Page opens.
2. Click Distribute a New Package. The **Distribute Package** page opens, listing all of the Windows Installer (`.msi`) packages in the open Application Catalog.
3. Click the radio button next to the Application Name of the package that you want to distribute. Only one package can be selected at a time.
4. Click **Distribute**. The **PDF Selection** page opens.

5. In the **Package Configuration** area of Configuration Manager Web Console, you specify distribution configuration fields that the Configuration Manager Server utilizes in the distribution of the package. You can manually enter package and distribution configuration information and settings, or you can choose to attach a Package Definition File (`.pdf`) to a Windows Installer (`.msi`) package to pre-populate these fields.

   A Package Definition File (`.pdf`) is a text file that contains predefined programs and property settings for a package and is used to automate package creation. When you use a package definition file, appropriate fields on the Configuration Manager Package Configuration Page are populated with the information in the package definition file you selected. A package definition file typically has an `.sms` file name extension (`.pdf` in previous versions of Configuration Manager).

   If you want to distribute this application using an existing Configuration Manager Package Definition File (`.pdf` or `.sms`) (either one on the Configuration Manager Server or one on the machine where Configuration Manager Web Console is installed), select a Package Definition File by performing the following steps:

   a. Select the **Distribute using Package Definition** option if you want to distribute this application using an existing Configuration Manager Package Definition File (`.pdf` or `.sms`). When you select this check box, the **Use Existing Package Definition** option is automatically selected, and the existing Package Definitions available on the Configuration Manager Server are listed.

   b. Select the **Use Existing Package Definition** option to distribute this application using any Package Definitions available on the Configuration Manager Server. When the **Distribute using Package Definition** check box is selected, this option is automatically selected, and the existing Package Definitions available on the Configuration Manager Server are listed. Select a Package Definition File from the list.

   c. Select the **Use Package Definition from the following file** option if you want to distribute this application using a `.pdf` or `.sms` file located on the machine where Configuration Manager Console is installed. When you select this option, all of the `.pdf` or `.sms` files in the following directory are listed:

      ```
      <Configuration Manager Console Installation Directory>\SMS Web Console
          Application\Distribution Providers\SMS\PDFs
      ```

6. Select a Package Definition File from the list.
7. Click **Next**. The Configuration Manager Package Configuration Page appears, listing the information in the `.pdf` or `.sms` file (if one was selected).
8. Proceed with the steps in Setting Package Configuration Options to either enter or modify the package configuration options.

Setting Package Configuration Options

On the Configuration Manager Package Configuration Page of Configuration Manager Web Console, you specify distribution configuration fields that the Configuration Manager Server utilizes in the distribution of the package. You can manually enter package and distribution configuration information and settings, or you can choose to attach a Package Definition File (`.pdf`) to a Windows Installer package to pre-populate these fields.

Task: ***To set package configuration options:***

1. Perform the steps described in Selecting a Package for Distribution.
2. Specify configuration options on each of the following Views of the Configuration Manager Package Configuration Page:
   a. **Package Settings View**—This View opens when a package name is selected in the tree. Specify general information about the package, along with data source, data access, distribution settings, and reporting information. See Setting Package Settings Options.
   b. **Access Accounts View**—This View opens when Access Accounts is selected in the tree. Specify which administrators are allowed to distribute this particular package. See Setting Access Accounts Options.
   c. **Distribution Points View**—This View opens when Distribution Points is selected in the tree. Identify which servers may act as distribution points for this particular package. See Setting Distribution Points Options.
   d. **Programs View**—This View opens when Programs is selected in the tree. Specify programs (that will run on the clients) to execute the distribution of the application, and the types of clients on which the program can run (directories, operating system versions, etc.). See Setting Programs Options.
   e. **Advertisements View**—This View opens when Advertisements is selected in the tree. Identify the collection of users to receive this package, and specify when the package will be available for distribution and whether it is optional or mandatory. See Setting Advertisements Options.

   Caution • *Remember to click Update on each view to save your changes.*
3. When you have finished specifying configuration options, click **Next**. The **Package Summary** page opens, providing a summary of all of the options specified for the package.
4. Proceed with the steps in Saving the Package on the Configuration Manager Server.

Setting Package Settings Options

On the Package Settings View, you specify general information about the package, along with data source, data access, distribution settings, and reporting information.

Task: ***To set Package Settings configuration options:***

1. Perform the steps described in Selecting a Package for Distribution.
2. On the Configuration Manager Package Configuration Page, select the package name in the tree. The Package Settings View opens, presenting information on five tabs: **General**, **Data Source**, **Data Access**, **Distribution Settings**, and **Reporting**.

3. Enter the following information on the **General** tab.

Option	Description
Icon	To change the displayed icon, select a different icon from the list. This list is populated with the files available from: <Configuration Manager Web Console Installation Directory>\SMS Web Console Application\Distribution Providers\SMS\Icons
Name	The name of the package, up to 50 characters. This field is required.
Version	The version number of the software package, up to 32 characters.
Publisher	The name of the software publisher, up to 32 characters.
Language	The language version of the software, up to 32 characters.
Comment	Optional text about the package, such as a description. You can use up to 127 characters.

4. Click the **Data Source** tab and enter the following information:

Option	Description
This Package contains source files	This package has source files that are required to run its programs when they are advertised to clients. By default, this check box is cleared. When this check box is cleared, Configuration Manager does not use distribution points for the package.
Source Directory	The network path to the package source files.
Use a compressed copy of the source directory	Configuration Manager will create a compressed version of the source files on the site server. When you specify additional or refresh distribution points for the package, the compressed version will be decompressed and copied to the distribution points, rather than being copied directly from the original source directory. Use this option if the source files might be removed from the specified path (for example, if the source files are on a compact disc). A compressed version of the source directory is always created and used when the package is sent to distribution points in child sites. This field is available only when the This Package contains source files check box is selected.

Option	Description
Always obtain files from source directory	Configuration Manager will obtain package source files (the files needed to run the package programs, such as .exe and .dll files) from the path specified in the Source directory box. Use this option if the source files are likely to remain at the specified path, such as a share on a server. This field is available only when the This package contains source files check box is selected. By default, this option is selected. ***Note*** • *This option is unavailable if the package is a compressed version replicated from a parent site.*

5. Click the **Data Access** tab and enter the following information:

Property	Description
Access distribution folder through common Configuration Manager package share	Users will access the package through the common Configuration Manager package share (SMSPKGx$, where *x* is the drive letter) on distribution points. By default, this option is selected.
Share Distribution folder	The package will be created on distribution points with an administrator-specified share (and path name) that is entered in the Share name text box.
Disconnect users from distribution points	Notifies users before disconnecting them from distribution points when Configuration Manager updates package data. By default, this check box is cleared. • **Number of retries before disconnecting users**—The number of times that Configuration Manager tries to update the package source files before starting to disconnect users who are connected to the distribution point. The default setting is 2; you can specify 0 to 99, inclusive. • **User grace period**—The number of minutes Configuration Manager should wait after notifying users before disconnecting them from distribution points. The default setting is 5 minutes; you can specify between 0 and 59 minutes, inclusive.

6. Click the **Distribution Settings** tab and enter the following information:

Option	Description
Sending priority	The priority of this package when sent to distribution points in child sites. Packages can be sent with High, Medium, or Low priority. The default setting is Medium priority. • If a package has **High** priority, it will be sent before packages with Medium or Low priority. • If a package has **Low** priority, it will be sent after packages with higher priority settings. • If packages have **identical** priorities, they will be sent in the order in which they were created in the Configuration Manager Administrator console. Sending priority for a package is carried over in packages sent from a parent site to a child site.
Preferred sender	The sender to be used to send this package to distribution points in other sites. You can specify any sender configured for the site, or No Preference (which uses any available sender). The default setting is No Preference. ***Note*** • *To use Courier Sender to distribute packages, you must select it as the preferred sender.*

7. Click the **Reporting** tab and enter the following information:

Property	Description
Use package properties for status MIF matching	Configuration Manager will use the properties in the **General** tab for status MIF file matching. By default, this option is selected.
Use these fields for status MIF matching	Configuration Manager will use the following values specified for status MIF file matching: • **MIF file name**—The name of the MIF file that contains the package status, up to 50 characters. • **Name**—The name of the package, up to 50 characters. • **Version**—The version number of the package, up to 32 characters. • **Publisher**—The software publisher of the package, up to 32 characters.

8. When you have finished entering the Package Settings options, proceed with entering the rest of the Package Configuration options:
   - Setting Access Accounts Options
   - Setting Distribution Points Options

- Setting Programs Options
- Setting Advertisements Options

9. If you are finished entering Configuration Manager Package Configuration options, click **Next** to proceed to the **Package Summary** page and perform the steps listed in Saving the Package on the Configuration Manager Server.

Setting Access Accounts Options

On the Access Accounts View, you specify which administrators are allowed to distribute this particular package.

Task: ***To set Access Accounts configuration options:***

1. Perform the steps described in Selecting a Package for Distribution.
2. On the Configuration Manager Package Configuration Page, select **Access Accounts** in the tree. The Access Accounts View opens.
3. Use this view to create new users and groups. If you select a user's icon under Access Accounts in the tree, you can review (but not change) that user's permissions and can delete the user.

Option	Description
Domain User Listing	List of all defined users, displaying the following: • **Name**—User name using the syntax of Domain\User. • **Type**—Identifies whether this is a user or a group, and identifies whether they have Windows or Generic access. • **Permissions**—Lists the user or group's permission.
Create New Access Account	To create a new access account, select whether you want to create a Windows User Access Account or a Generic Access Account, and perform the following steps: **To create a Windows User Access Account:** 1. Enter a **User Name** using the syntax of Domain\User. 2. Under **Account Type**, select **User** or **Group**. 3. Select **this** user's permission level from the **Permissions** list. 4. Click Create. **To create a Generic Access Account:** 1. Under **Account Type**, select **Users**, **Guests** or **Administrators**. 2. Select this generic account's permission level from the **Permissions** list. 3. Click **Create**.

4. When you have finished entering the Access Accounts options, proceed with entering the rest of the Package Configuration options:
   - Setting Package Settings Options
   - Setting Access Accounts Options
   - Setting Distribution Points Options
   - Setting Programs Options
   - Setting Advertisements Options
5. If you are finished entering Configuration Manager Package Configuration options, click **Next** to proceed to the **Package Summary** page and perform the steps listed in Saving the Package on the Configuration Manager Server.

Setting Distribution Points Options

On the **Distribution Points View**, specify which servers may act as distribution points for this particular package. A Distribution Point is a site system role that stores software package files so clients can access them during the software distribution process. See Distributing Software Through Configuration Manager Server for more information.

Task: ***To set Distribution Points options:***

1. Perform the steps described in Selecting a Package for Distribution.
2. On the Configuration Manager Package Configuration Page, select Distribution Points in the tree. The Distribution Points View opens, listing targeted distribution points and displaying the following information:
   - **Name**—The name of the distribution point.
   - **Site**—The name of the site where the distribution point is located (for example, NYC - New York City).
   - **Type**—The type of distribution point (for example, Windows NT Server or NetWare Bindery).
3. To target a new distribution point(s), select a distribution point(s) from the **Select New Distribution Point** list and click **OK**. The new distribution point will now be listed under **Distribution Points** in the tree.
4. When you have finished entering the Distribution Points options, proceed with entering the rest of the Package Configuration options:
   - Setting Package Settings Options
   - Setting Access Accounts Options
   - Setting Programs Options
   - Setting Advertisements Options
5. If you are finished entering Configuration Manager Package Configuration options, click **Next** to proceed to the **Package Summary** page and perform the steps listed in Saving the Package on the Configuration Manager Server.

Setting Programs Options

On the Programs View, you specify programs (that will run on the clients) to execute the distribution of the application, and the types of clients on which the program can run (directories, operating system versions, etc.). The fields shown on the **Programs View** depends upon what is selected in the tree:

- To view a listing of existing programs and to create a new one, use the Main Programs View, which is displayed when you select **Programs** in the tree.
- To edit an existing program, select the program icon under **Programs** in the tree to access the detailed tab views.

Task: ***To set Program configuration options:***

1. Perform the steps described in Selecting a Package for Distribution.
2. On the Configuration Manager Package Configuration Page, select **Programs** in the tree. The Programs View opens.
3. To create a new program, select **Programs** in the tree, enter a name in the **Name** text box, and click **Create**. The new program will now be listed under **Programs** in the tree. To edit the program settings on the detailed tab view, select the program in the tree.
4. To edit an existing program, select the program icon under **Programs** in the tree to access the detailed tab views (**General**, **Requirements**, **Environment**, and **Advanced**) where you can edit the program settings.
5. Click the **General** tab, and enter the following information:

Option	Description
Icon	To change the displayed icon, select a different icon from the list. This list is populated with the files available from: `<Configuration Manager Web Console Installation Directory>\SMS Web Console Application\Distribution Providers\SMS\Icons`
Name	The name of the program, up to 50 characters. This field is required and must be unique within a package. This name is used when defining advertisements.
Comment	Optional text about the program, such as a description. You can use up to 127 characters.

Option	Description

Command Line	The command line for the program, up to 511 characters. This field is required. File name extensions are strongly recommended.
	
	Click **Browse** to navigate to the file. By default, the package source directory appears, if it exists and is accessible. When the program is run on a client, the command line file name will first be searched for within the package.
	
	• If the file does not exist there, Configuration Manager searches the local Windows directory.
	• If Configuration Manager cannot find the file, the program fails.
	• If the file name does not have a file name extension specified, Configuration Manager tries .pif, .com, .exe, and .bat.
	• If the file name has an extension but is not an executable, Configuration Manager tries to apply a local association. For example, if the command line is readme.gif, Configuration Manager will start the application specified on the client for opening .gif files. For example:
	
	`setup.exe /a`
	
	`command.com /c copy Jan98.dat c:\sales\Jan98.dat`
	
	Note *• On 16-bit clients, batch files MUST specify the .bat file name extension, or the command line will fail.*

Start in	The executable directory for the program, up to 127 characters. This directory can be an absolute path on the client, or a path relative to the distribution point folder that contains the package. This field is optional. For example:
	
	`c:\office97`
	`i386`

Run	The program mode. By default, programs run in **Normal** mode. You can also run a program in **Minimized**, **Maximized**, or **Hidden** mode.

Option	Description
After Running	The action that occurs after this program is completed successfully. • **No action required**—No restart or logoff is required. This is the default value. • **Program restarts computer**—The program requires a restart and performs the restart automatically. • **Configuration Manager restarts computer**—Configuration Manager restarts the computer. • **Configuration Manager logs user off**—Configuration Manager logs the current user off the computer after the program is completed. Use this option if the program requires that the user log off and then log on again before it can run. ***Note*** *• On 16-bit clients, only No action required is supported.* ***Caution*** *• Both the Configuration Manager restarts computer and the Configuration Manager logs user off actions are done forcefully after providing the user with a grace period. Applications that are running on clients when this occurs will not have an opportunity to save their states or data.*

6. Click the **Requirements** tab and enter the following information:

Option	Description
Estimated disk space	A whole number greater than or equal to zero that represents the amount of disk space the software will require when installed on the client. Also, specify the units for the value.
Maximum allowed run time	A whole number greater than zero that represents the time (in minutes) required to run the program on the client.
Notify user if the program runs 15 minutes longer than estimated	Configuration Manager will monitor the run time of the program and warn the user if the program is taking longer to run than the estimated time. By default, this check box is cleared. ***Note*** *• This feature is not supported on 16-bit clients.*
This program can run on any platform	The program can run on any platform. If this option is selected, Configuration Manager does not check the platform type when advertising the program to clients. By default, this option is selected.

Option	Description
This program can run only on specified client platforms	The processors, operating systems, and service packs on which this program will run. When this option is selected, at least one platform must also be selected. By default, no platforms are selected. Configuration Manager uses this information when evaluating which clients in a collection are to receive the advertised program. ***Note*** *• If you advertise this program to clients in Configuration Manager 1.2 sites, Configuration Manager performs a less in-depth level of platform checking than it does for Configuration Manager 2.0 clients. Platform checking for Configuration Manager 1.2 clients is limited to Windows 3.1, Windows 95, x86 Windows NT, Alpha Windows NT, and Macintosh.*
Additional Requirements	Other information for clients, up to 127 characters. This field is optional. For example, you can notify users that they should shut down all other applications before running this program.

7. Click the **Environment** tab and enter the following information:

Option	Description
Program can run	Specify whether the program requires that a user be logged on to the client computer to run: • **Only when a user is logged on**—Prevents the program from running if no user is logged on to the computer. This is the default setting. Use this option for clients that do not run a Windows NT operating system. • **Only when no user is logged on**— Prevents the program from running until the user logs off the computer. This option is valid only for Windows NT clients. This option forces the program to run under the local Administrator account on the client. If a user logs on while the installation is running, the program continues to run. • **Whether or not a user is logged on**—Enables the program to run with or without user interaction. This option is valid only for Windows NT clients. This option forces the program to run under the local Administrator account on the client.
User input required	Requires that the user interact with the program when it runs. This check box is selected when Program can run is set to Only when a user is logged on. If the program does not require user input, clear this check box.

Option	Description
Run mode	Select one of the following run modes: • **Run with user's rights** • **Run with administrative rights** To specify that the program requires administrative privileges on the computer in order to run, select Run with administrative rights. This option is available when Program can run is set to Only when a user is logged on. By default, this check box is cleared. This option is valid only for Windows NT clients. The **Run with administrative rights** option forces the program to run under the local Administrator account on the client.
Use software installation account	Run the program under an administrator-specified account (the Configuration Manager Windows NT Client Software Installation Account) on Windows NT clients. This check box is available only when Program can run is set to Only when no user is logged on or Whether or not a user is logged on. By default, this check box is cleared. This option should be used only when the program requires access to network resources other than the package files on the distribution point.
Drive mode	Select the drive mode used for network connections to the distribution points. • **Runs with UNC name**—Enables the program to run with a universal naming convention (UNC) name. This is the default setting. • **Requires drive letter**—Indicates that the program requires a drive letter to fully qualify its location, but Configuration Manager can use any available drive letter on the client. • **Requires specific drive letter**—Indicates that the program requires the drive letter you specify (A to Z) to fully qualify its location (for example, H). If the specified drive letter is already used on a client, the program will not run.
Reconnect to Distribution point at logon	The computer will reconnect to the drive when the user logs on. By default, this check box is cleared. ***Note*** • *This feature is not supported on 16-bit clients.*

8. Click the **Advanced** tab and enter the following information:

Option	Description
Run another program first	This program requires another program (in another package that is targeted for distribution using Configuration Manager Web Console or in the current package) to run before it. By default, this check box is cleared. ***Note*** *• This feature is not supported on 16-bit clients.* ***Note*** *• The program you specify to run first does not need to be advertised separately.*
Package	The package that contains the program that must run before this program.
Program	The program that must run first. ***Note*** *• If the depended-upon program fails on a client, the depending program will not run. Also, if the program that runs first has been run successfully, Configuration Manager will not automatically re-run it, even if the depending program is on a recurring schedule.*
When this program is assigned to a computer	When Program can run (in the **Environment** tab) is set to **Only when a user is logged on**, you can specify how the program is assigned to users. • **Run once for first user who logs on**—The program should run only once on the computer. This is the default setting. For example, a virus checker only needs to run once on the client. • **Run once for every user who logs on**—The program should run once for each new user who logs on. ***Note*** *• On 16-bit clients, advertised programs will run only for the first user who logs on.*

Option	Description
Remove software when it is no longer advertised	Indicates that Configuration Manager will remove this program from client computers when it is no longer advertised to those clients. This option is mainly useful for shared network applications and is available only to programs that register with Add/Remove Programs in Control Panel. To use this option, enter the name of the Uninstall Registry Key the program creates in the following registry key: `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\ CurrentVersion\Uninstall` After all advertisements for the program are revoked from the client, Configuration Manager looks up the program's uninstall command line in the `UninstallString` or `QuietUninstallString` property in this registry key. By default, this check box is cleared. An advertisement is considered revoked from a client when any of the following are true: • The client is no longer a member of the collection specified in the advertisement. • The advertisement is deleted. • The advertisement expires. • Uninstall registry key • The registry key that contains the uninstall command for this program. You can use up to 127 characters. ***Note*** • *This feature is not supported on 16-bit clients.*
Disable this program on computers where it is advertised	Temporarily disables all advertisements that contain this program. The program will be removed from the list available for users to run and will not be run via assignment until it is re-enabled. By default, this check box is cleared.

9. When you have finished entering the **Programs** options, proceed with entering the rest of the Package Configuration options:
   - Setting Package Settings Options
   - Setting Access Accounts Options
   - Setting Distribution Points Options
   - Setting Advertisements Options
10. If you are finished entering Configuration Manager Package Configuration options, click **Next** to proceed to the **Package Summary** page and perform the steps listed in Saving the Package on the Configuration Manager Server.

Setting Advertisements Options

On the Advertisements View, you identify the collection of users to receive this package, and specify when the package will be available for distribution and whether it is optional or mandatory.

Task: ***To set Advertisements configuration options:***

1. Perform the steps described in Selecting a Package for Distribution.
2. On the Configuration Manager Package Configuration Page, select **Advertisements** in the tree. The Advertisements View opens, displaying a list of existing advertisements:
   - **Name**—Name of the advertisement
   - **Program**—The program to run on the client.
   - **Collection**—The target collection of computers, users, or user groups that will receive the program.
   - **Available After**—Date after which the advertisement will be available
   - **Expires After**—Expiration date of advertisement
   - **Advertisement ID**—Number uniquely identifying the advertisement
   - **Status**—Advertisement's status
3. To create a new advertisement, enter a name in the **Name** text box and click **Create**. The new advertisement will now be listed under Advertisements in the tree.
4. To edit an existing advertisement, select it under **Advertisements** in the tree, and edit it on the detailed tab view (**General** and **Schedule**).
5. On the **General** tab, enter the following information:

Option	Description
Name	The name of the advertisement, up to 50 characters. This field is required.
Comment	Optional text about the advertisement, such as a description. You can use up to 127 characters.
Program	The program (within the current package) to advertise to clients. This field is required.
Collection	Select the existing collection that will receive the advertisement. Click Browse to open the Select a Configuration Manager Collection Dialog Box, where you can select the Configuration Manager collection that you want to use in this Advertisement definition. This field is required. ***Note*** • *Configuration Manager will not advertise a program to 16-bit clients based on user groups or user accounts, but this is supported for 32-bit clients.*

Option	Description
Include members of subcollections	Specifies that the program also be advertised to clients and users in the specified collection's subcollections. By default, this check box is selected.

6. On the **Schedule** tab, enter the following information

Option	Description
Advertisement start time	The date and time when the program is advertised and available to run on clients. By default, the current date and time is specified. **Greenwich Mean Time**—Sets the time independent of time zones. That is, if you set the time to 1:00 P.M. and select this option, the program will be advertised on all clients simultaneously—1:00 P.M. for clients in the Greenwich mean time zone, 10:00 A.M. for clients in the Eastern time zone, and 7:00 A.M. for clients in the Pacific time zone, and so on. This check box is cleared by default, which enables you to specify a relative time for advertising the program. Configuration Manager will treat the specified time as a local time. In this case, if the time is set to 1:00 P.M., the program will be advertised on clients at 1:00 P.M. in each time zone.
Mandatory assignments (Assign immediately after this event)	If you want to ensure that the program runs on all clients (users cannot choose not to run the program), select: • As soon as possible • Logon • Logoff ***Note*** *• Recurring assignments are not supported on 16-bit clients. Programs advertised to 16-bit clients run only once on those clients.*
Assignments are not mandatory over slow links	Suspends assignments for clients on a slow link. By default, this check box is selected. (This option is available only if at least one assignment is defined.)

Option	Description
Allow users to run the program independently of assignments	Enables users to run the program independent of when it is assigned. By default, this check box is cleared. (This option is available only if at least one assignment is defined.) Typically, only non-assigned programs are visible to users in the Configuration Manager client interface. Use this option to display an assigned program in the client interface. For example, to require users to upgrade their Microsoft Office software, you specify an assignment in the advertisement properties. But if you want to enable users to run the installation program before it is scheduled to run automatically, select this option so that the assigned program appears in Advertised Programs Wizard in the Control Panel on client computers.
Advertisement will expire	Advertises the program to clients only for a limited time. By default, this option is cleared. This feature is not supported on 16-bit clients; the program appears until it runs on the client. ***Note*** *• When a program expires, it no longer appears in Advertised Programs Monitor or Advertised Programs Wizard in Control Panel on client computers.To specify the cutoff date and time, click the arrows next to the date and time. The default values are six months from the current date and time.* **Greenwich Mean Time**—Set the time independent of time zones. That is, if you set the time to 1:00 P.M. and select this option, the program will expire at all computers simultaneously—1:00 P.M. for clients in the Greenwich mean time zone, 10:00 A.M. for clients in the Eastern time zone, and 7:00 A.M. for clients in the Pacific time zone, and so on. This check box is cleared by default, which enables you to specify a relative time. Configuration Manager will treat the specified time as a local time. In this case, if the time is set to 1:00 P.M., the program will expire at 1:00 P.M. on clients in each time zone.
Priority	The advertisement's priority when sent to child sites: High, Medium, or Low. The default priority is Medium.

7. When you have finished entering the **Advertisements** options, proceed with entering the rest of the Package Configuration options:
   - Setting Package Settings Options
   - Setting Access Accounts Options
   - Setting Distribution Points Options
   - Setting Programs Options
8. If you are finished entering Configuration Manager Package Configuration options, click **Next** to proceed to the **Package Summary** page and perform the steps listed in Saving the Package on the Configuration Manager Server.

Saving the Package on the Configuration Manager Server

To distribute an application from Configuration Manager Web Console using Configuration Manager Distribution, perform the following steps.

Task: ***To save the package to the Configuration Manager Server:***

1. Perform the steps described in Setting Package Configuration Options.
2. When the **Package Summary** page appears, review the summary of the all the options specified for the package.
3. To save this package on the Configuration Manager Server, click **Commit Changes to Configuration Manager**. A message will appear informing you that the package information was successfully saved to Configuration Manager.

   You could now view the status of the package by following the instructions in Viewing the Status of a Distributed Package.

Viewing the Status of a Distributed Package

Click **Package Administration** on the **Configuration Manager Web Console Home** page to see the status of a distributed package, delete a package, or modify a package's distribution settings.

Task: ***To view the status of a distributed package:***

1. Launch Configuration Manager Web Console. The **Configuration Manager Web Console** Home Page opens.
2. Click **Package Administration**. The **Package Administration** page appears, listing all packages which are targeted for distribution.
3. Select a package and click **View Status**. The **Connect to Configuration Manager** page appears.
4. Enter a valid **User Name** and **Password** for the Configuration Manager Server you want to connect to, and click **Next**. Configuration Manager Web Console retrieves the status of that package from the Configuration Manager Server, and displays that information on the Package Status Page.

Modifying the Distribution Settings of a Package

Click Package **Administration** on the **Configuration Manager Web Console** Home Page to see the status of a distributed package, delete a package, or modify a package's distribution settings.

Task: ***To modify the distribution settings of a package:***

1. Launch Configuration Manager Web Console. The **Configuration Manager Web Console** Home Page opens.
2. Click Package Administration. The **Package Administration** page appears, listing all packages which are targeted for distribution.
3. Select a package and click **Modify Settings**. The **Connect to Configuration Manager** page appears.

   If this package has not been distributed, you can change the **Site Server** and **Site Code** fields on the **Connect to Configuration Manager** page. However, if the package has been distributed, you cannot change these fields.
4. Enter a valid **User Name** and **Password** for the Configuration Manager Server you want to connect to, and click **Next**. The **General** tab of the **Package Settings View** of the **Configuration Manager Package Configuration** page opens.
5. Edit the package's distribution settings as described in Setting Package Configuration Options.

Deleting Packages

From the Configuration Manager Web Console, you can delete a package from the Application Catalog, from the Configuration Manager Server, or both.

Caution • *Deleting a distributed package from Configuration Manager will also delete its programs and any advertisements of the programs. If the package has source files, they will be removed from distribution points. If there are access accounts for this package, they will be deleted. Also, Configuration Manager Administrators' security rights to the package will be deleted.*

Caution • *When you delete a package from the Application Catalog, all package configuration information is lost, even if you do not delete it from Configuration Manager. If you delete a package from the Application Catalog but not from Configuration Manager, the package would still be available on the Configuration Manager Server, but all connection between the Application Catalog and Configuration Manager is lost. Should you want to distribute that package again in the future, you would have to re-import the application into the Application Catalog, and then re-enter package configuration information on the Configuration Manager Package Configuration Page.*

Task: ***To delete a package:***

1. Launch Configuration Manager Web Console. The **Configuration Manager Web Console** Home Page opens.
2. Click **Package Administration**. The **Package Administration** page appears, listing all packages in your Application Catalog. In the **Status** column, each package's status is either **Distributed** or **Not Distributed**.
3. Select a package and click **Delete Distribution**. The **Connect to Configuration Manager** page appears.
4. Enter a valid **User Name** and **Password** for the Configuration Manager Server you want to connect to, and click **Next**. The **Confirm Delete** page appears prompting you to confirm that you want to delete this package from the Application Catalog. If you are deleting a Distributed package, you are also prompted to confirm that you want to delete this package from Configuration Manager.
5. Select **Yes** or **No** to indicate your deletion preferences:

Package Status	Delete from Catalog?	Delete from Configuration Manager?
Not Distributed	Choose **Yes** to confirm the deletion from the Application Catalog or choose **No** to decline the deletion. If you choose **Yes**, the package will be deleted from the Application Catalog, along with all of the package configuration information that you entered on the **Configuration Manager Package Configuration** page.	N/A
Distributed	Choose **Yes** to confirm the deletion from the Application Catalog, or choose **No** to decline the deletion. If you delete a package from Configuration Manager but do not delete it from the Application Catalog, all of the package configuration information that you entered on the **Configuration Manager Package Configuration** page is saved, making it easy to distribute this package again in the future.	Choose **Yes** to confirm the deletion from Configuration Manager, or choose **No** to decline the deletion. Unless you are certain that you will never want to distribute this package again, it is recommended that if you delete a package from Configuration Manager, you do not delete it from the Application Catalog.

6. Click **OK** to confirm your selections.

Configuration Manager Web Console Reference

This section describes each of pages that you might encounter when using the Configuration Manager Web Console. The following pages and views are documented:

- Configuration Manager Package Configuration Page
  - Package Settings View
  - Access Accounts View
  - Distribution Points View
  - Programs View
  - Advertisements View
- Connect to Configuration Manager Page
- Confirm Delete Page
- Distribute New Package Page
- Distribute Package Page
- Microsoft System Center Configuration Manager Web Console Home Page
- Package Administration Page
- Package Status Page
- Package Summary Page
- PDF Selection Page
- Select a Configuration Manager Collection Dialog Box
- Distribution Settings Page

Microsoft System Center Configuration Manager Web Console Home Page

The Software Distribution page is the “home” page of Configuration Manager Web Console. From this page you can choose to distribute a new package, perform package administration, and configure distribution settings.

The following selections are available:

Table 21-4 • Configuration Manager Web Console Home Page Options

Option	Description
Distribute a New Package	Click to access the Distribute Package Page where you can select a package to distribute using distribution providers.
Package Administration	Click to access the Package Administration Page, where you can view the status of distributed applications, modify settings of distributed packages, and delete an existing distribution.
Distribution Settings	Click to access the Distribution Settings Page, where you can configure global distribution settings or configure distribution provider settings.

Connect to Configuration Manager Page

This dialog appears when you are attempting to connect to a Configuration Manager Server.

All of the settings configured on this page only apply to the selected package. The default settings (specified on the Distribution Settings Page) do not change.

Table 21-5 • Connect to Configuration Manager Page Options

Option	Description
Site Server	Enter the machine name on which Configuration Manager Server is installed. (This could be a local or remote machine.) When you make a selection in this field, the **Site Code** field is populated with a list of sites available on the Configuration Manager Server.
Site Code	Select the Configuration Manager site you want to use for software distribution.
User Name	Enter a valid user name for the Configuration Manager Server you want to connect to, such as: `Company\UserName`.
Password	Enter the appropriate password.
Next	Click to proceed with the connection.

Distribute Package Page

The Windows Installer `.msi` packages that you are preparing for distribution to Configuration Manager Server are stored in the AdminStudio Enterprise Server Application Catalog. All of the Windows Installer packages in the open Application Catalog are listed on the Distribute Package page, making them available for distribution. On this page, select the package that you want to distribute to a Configuration Manager Server and click **Distribute**.

This page includes the following information and functionality.

Table 21-6 • Distribute Package Page Options

Option	Description
Application Name	Lists the names of all applications available for distribution. Click the radio button next to the Application Name to select it for distribution. Only one application can be selected at a time.
Company Name	Lists the company affiliated with the listed application.
Distribute	Click to initiate distribution of the selected application.

PDF Selection Page

A Package Definition File (`.pdf`) is a text file that contains predefined programs and property settings for a package and is used to automate package creation. When you use a package definition file, appropriate fields on the Configuration Manager Package Configuration Page are populated with the information in the package definition file you selected. A package definition file typically has an `.sms` file name extension (`.pdf` in previous versions of Configuration Manager).

This page contains the following fields:

Table 21-7 • PDF Selection Page Options

Option	Description
Distribute Using Package Definition	Select this option if you want to distribute this application using an existing Configuration Manager Package Definition File (`.pdf` or `.sms`). When you select this check box, the **Use Existing Package Definition** option is automatically selected, and the list of existing Package Definitions available on the Configuration Manager Server are shown on the following grid.
Use Existing Package Definition	Select this option to distribute this application using any Package Definitions available on the Configuration Manager Server. When the **Distribute using Package Definition** check box is selected, this option is automatically selected, and the list of existing Package Definitions available on the Configuration Manager Server are shown on the following grid.

Table 21-7 • PDF Selection Page Options (cont.)

Option	Description
Use Package Definition from the following file	Select this option if you want to distribute this application using a `.pdf` or `.sms` file located on the machine where Configuration Manager Console is installed. When you select this option, the list below it is populated with a list of all of the `.pdf` or `.sms` files in the following directory: `<Configuration Manager Web Console Installation Directory>\ SMS Web Console Application\Distribution Providers\SMS\PDFs`
Next	Click to proceed to the Configuration Manager Package Configuration Page where appropriate fields are populated with the information in the Package Definition you selected.

Configuration Manager Package Configuration Page

The **Configuration Manager Package Configuration** page, which you use to specify distribution configuration fields that the Configuration Manager Server utilizes in the distribution of the package, consists of the following views:

Table 21-8 • Configuration Manager Package Configuration Page Views

View	Description
Package Settings View	General information about the package, along with data source, data access, distribution settings, and reporting information.
Access Accounts View	Specify which administrators are allowed to distribute this particular package.
Distribution Points View	Specify which servers may act as distribution points for this particular package. A Distribution Point is a site system role that stores software package files so clients can access them during the software distribution process.
Programs View	Specify programs (that will run on clients) to execute the distribution of the application, and the types of clients on which the program can run (defined using directories, operating system versions, etc.).
Advertisements View	Identify the collection of users to receive this package, and specify when the package will be available for distribution and whether it is optional or mandatory.

Package Settings View

Information on the Package Settings View is presented on the following tabs:

- General Tab
- Data Source Tab
- Data Access Tab
- Distribution Settings Tab
- Reporting Tab

Enter and edit information on these tabs and then click **Update** to save your changes.

General Tab

The **General** tab of the **Package Settings View** contains the following options:

Table 21-9 • Package Settings View — General Tab Options

Option	Description
Icon	To change the displayed icon, select a different icon from the list. This list is populated with the files available from: `<Configuration Manager Web Console Installation Directory>\SMS Web Console Application\Distribution Providers\SMS\Icons`
Name	The name of the package, up to 50 characters. This field is required.
Version	The version number of the software package, up to 32 characters.
Publisher	The name of the software publisher, up to 32 characters.
Language	The language version of the software, up to 32 characters.
Comment	Optional text about the package, such as a description. You can use up to 127 characters.

Data Source Tab

The **Data Source** tab of the **Package Settings View** contains the following options:

Table 21-10 • Package Settings View — Data Source Tab Options

Option	Description
This Package contains source files	This package has source files that are required to run its programs when they are advertised to clients. By default, this check box is cleared. When this check box is cleared, Configuration Manager does not use distribution points for the package.
Source Directory	The network path to the package source files.

Table 21-10 • Package Settings View — Data Source Tab Options (cont.)

Option	Description
Use a compressed copy of the source directory	Configuration Manager will create a compressed version of the source files on the site server. When you specify additional or refresh distribution points for the package, the compressed version will be decompressed and copied to the distribution points, rather than being copied directly from the original source directory. Use this option if the source files might be removed from the specified path (for example, if the source files are on a compact disc). A compressed version of the source directory is always created and used when the package is sent to distribution points in child sites. This field is available only when the This Package contains source files check box is selected.
Always obtain files from source directory	Configuration Manager will obtain package source files (the files needed to run the package programs, such as .exe and .dll files) from the path specified in the Source directory box. Use this option if the source files are likely to remain at the specified path, such as a share on a server. This field is available only when the This package contains source files check box is selected. By default, this option is selected. ***Note*** • *This option is unavailable if the package is a compressed version replicated from a parent site.*

Data Access Tab

The **Data Access** tab of the **Package Settings View** contains the following options:

Table 21-11 • Package Settings View — Data Access Tab Options

Property	Description
Access distribution folder through common Configuration Manager package share	Users will access the package through the common Configuration Manager package share (SMSPKGx$, where *x* is the drive letter) on distribution points. By default, this option is selected.
Share Distribution folder	The package will be created on distribution points with an administrator-specified share (and path name) that is entered in the Share name text box.

Table 21-11 • Package Settings View — Data Access Tab Options (cont.)

Property	Description
Disconnect users from distribution points	Notifies users before disconnecting them from distribution points when Configuration Manager updates package data. By default, this check box is cleared. • **Number of retries before disconnecting users**—The number of times that Configuration Manager tries to update the package source files before starting to disconnect users who are connected to the distribution point. The default setting is 2; you can specify 0 to 99, inclusive. • **User grace period**—The number of minutes Configuration Manager should wait after notifying users before disconnecting them from distribution points. The default setting is 5 minutes; you can specify between 0 and 59 minutes, inclusive.

Distribution Settings Tab

The **Distribution Settings** tab of the **Package Settings View** contains the following options:

Table 21-12 • Package Settings View — Distribution Settings Tab Options

Option	Description
Sending priority	The priority of this package when sent to distribution points in child sites. Packages can be sent with High, Medium, or Low priority. The default setting is Medium priority. • If a package has **High** priority, it will be sent before packages with Medium or Low priority. • If a package has **Low** priority, it will be sent after packages with higher priority settings. • If packages have **identical** priorities, they will be sent in the order in which they were created in the Configuration Manager Administrator console. Sending priority for a package is carried over in packages sent from a parent site to a child site.
Preferred sender	The sender to be used to send this package to distribution points in other sites. You can specify any sender configured for the site, or No Preference (which uses any available sender). The default setting is No Preference. ***Note*** • *To use Courier Sender to distribute packages, you must select it as the preferred sender.*

Reporting Tab

The **Reporting** tab of the **Package Settings View** contains the following options:

Table 21-13 • Package Settings View — Reporting Tab Options

Property	Description
Use package properties for status MIF matching	Configuration Manager will use the properties in the **General** tab for status MIF file matching. By default, this option is selected.
Use these fields for status MIF matching	Configuration Manager will use the following values specified for status MIF file matching: • **MIF file name**—The name of the MIF file that contains the package status, up to 50 characters. • **Name**—The name of the package, up to 50 characters. • **Version**—The version number of the package, up to 32 characters. • **Publisher**—The software publisher of the package, up to 32 characters.

Access Accounts View

Use this view to create new users and groups. If you select a user's icon under Access Accounts in the tree, you can review (but not change) that user's permissions and can delete the user.

Table 21-14 • Access Accounts View Options

Option	Description
Domain User Listing	List of all defined users, displaying the following: • **Name**—User name using the syntax of Domain\User. • **Type**—Identifies whether this is a user or a group, and identifies whether they have Windows or Generic access. • **Permissions**—Lists the user or group's permission.

Table 21-14 • Access Accounts View Options (cont.)

Option	Description
Create New Access Account	To create a new access account, select whether you want to create a Windows User Access Account or a Generic Access Account, and perform the following steps: **To create a Windows User Access Account:** 1. Enter a **User Name** using the syntax of Domain\User. 2. Under **Account Type**, select **User** or **Group**. 3. Select **this** user's permission level from the **Permissions** list. 4. Click Create. **To create a Generic Access Account:** 1. Under **Account Type**, select **Users**, **Guests** or **Administrators**. 2. Select this generic account's permission level from the **Permissions** list. 3. Click **Create**.

Deleting a User or Group

To delete an individual user or group, select the user icon under **Access Accounts** in the tree and click **Delete**.

Distribution Points View

On the Distribution Points View, you can view a list of distribution points available on the Configuration Manager Server and can target them for this package.

A Distribution Point is a site system role that stores software package files so clients can access them during the software distribution process. See Distributing Software Through Configuration Manager Server for more information.

Table 21-15 • Distribution Points View Options

Option	Description
Distribution Point Listing	List of all defined distribution points, displaying the following: • **Name**—The name of the distribution point. • **Site**—The name of the site where the distribution point is located (for example, NYC - New York City). • **Type**—The type of distribution point (for example, Windows NT Server or NetWare Bindery).
Select New Distribution Point	To target a new distribution point(s), select a distribution point(s) from the list and click **OK**. The new distribution point will now be listed under **Distribution Points** in the tree.

Programs View

To view a listing of existing programs and to create a new one, use the main **Programs View**.

Table 21-16 • Programs View Options

Option	Description
Program Listing	List of all defined programs, displaying the program name, run time, disk space, and any comments associated with the program.
Create New Program	To create a new program, enter a name in the **Name** text box and click **Create**. The new program will now be listed under **Programs** in the tree. To edit the program settings on the detailed tab view, select the program in the tree.

To edit an existing program, select the program icon under **Programs** in the tree to access the detailed tab views (**General**, **Requirements**, **Environment**, and **Advanced**) where you can edit the program settings.

General Tab

Make edits in these fields and then click **Update** to save your edits. To delete the program, click **Delete**.

Table 21-17 • Programs View — General Tab Options

Option	Description
Icon	To change the displayed icon, select a different icon from the list. This list is populated with the files available from: `<Configuration Manager Web Console Installation Directory>\SMS Web Console Application\Distribution Providers\SMS\Icons`
Name	The name of the program, up to 50 characters. This field is required and must be unique within a package. This name is used when defining advertisements.
Comment	Optional text about the program, such as a description. You can use up to 127 characters.

Table 21-17 • Programs View — General Tab Options (cont.)

Option	Description

Command Line	The command line for the program, up to 511 characters. This field is required. File name extensions are strongly recommended.
	
	Click **Browse** to navigate to the file. By default, the package source directory appears, if it exists and is accessible. When the program is run on a client, the command line file name will first be searched for within the package.
	
	• If the file does not exist there, Configuration Manager searches the local Windows directory.
	• If Configuration Manager cannot find the file, the program fails.
	• If the file name does not have a file name extension specified, Configuration Manager tries .pif, .com, .exe, and .bat.
	• If the file name has an extension but is not an executable, Configuration Manager tries to apply a local association. For example, if the command line is readme.gif, Configuration Manager will start the application specified on the client for opening .gif files. For example:
	
	`setup.exe /a`
	
	`command.com /c copy Jan98.dat c:\sales\Jan98.dat`
	
	Note • *On 16-bit clients, batch files MUST specify the .bat file name extension, or the command line will fail.*

Start in	The executable directory for the program, up to 127 characters. This directory can be an absolute path on the client, or a path relative to the distribution point folder that contains the package. This field is optional. For example:
	
	`c:\office97`
	`i386`

Run	The program mode. By default, programs run in **Normal** mode. You can also run a program in **Minimized**, **Maximized**, or **Hidden** mode.

Table 21-17 • Programs View — General Tab Options (cont.)

Option	Description
After Running	The action that occurs after this program is completed successfully. • **No action required**—No restart or logoff is required. This is the default value. • **Program restarts computer**—The program requires a restart and performs the restart automatically. • **Configuration Manager restarts computer**—Configuration Manager restarts the computer. • **Configuration Manager logs user off**—Configuration Manager logs the current user off the computer after the program is completed. Use this option if the program requires that the user log off and then log on again before it can run. ***Note*** • *On 16-bit clients, only No action required is supported.* ***Caution*** • *Both the Configuration Manager restarts computer and the Configuration Manager logs user off actions are done forcefully after providing the user with a grace period. Applications that are running on clients when this occurs will not have an opportunity to save their states or data.*

Requirements Tab

Make edits in these fields and then click **Update** to save your edits. To delete the program, click **Delete**.

Table 21-18 • Programs View — Requirements Tab Options

Option	Description
Estimated disk space	A whole number greater than or equal to zero that represents the amount of disk space the software will require when installed on the client. Also, specify the units for the value.
Maximum allowed run time	A whole number greater than zero that represents the time (in minutes) required to run the program on the client.
Notify user if the program runs 15 minutes longer than estimated	Configuration Manager will monitor the run time of the program and warn the user if the program is taking longer to run than the estimated time. By default, this check box is cleared. ***Note*** • *This feature is not supported on 16-bit clients.*
This program can run on any platform	The program can run on any platform. If this option is selected, Configuration Manager does not check the platform type when advertising the program to clients. By default, this option is selected.

Table 21-18 • Programs View — Requirements Tab Options (cont.)

Option	Description
This program can run only on specified client platforms	The processors, operating systems, and service packs on which this program will run. When this option is selected, at least one platform must also be selected. By default, no platforms are selected. Configuration Manager uses this information when evaluating which clients in a collection are to receive the advertised program. ***Note*** • *If you advertise this program to clients in Configuration Manager 1.2 sites, Configuration Manager performs a less in-depth level of platform checking than it does for Configuration Manager 2.0 clients. Platform checking for Configuration Manager 1.2 clients is limited to Windows 3.1, Windows 95, x86 Windows NT, Alpha Windows NT, and Macintosh.*
Additional Requirements	Other information for clients, up to 127 characters. This field is optional. For example, you can notify users that they should shut down all other applications before running this program.

Environment Tab

The **Environment** tab of the **Programs View** contains the following options:

Table 21-19 • Programs View — Environment Tab Options

Option	Description
Program can run	Specify whether the program requires that a user be logged on to the client computer to run: • **Only when a user is logged on**—Prevents the program from running if no user is logged on to the computer. This is the default setting. Use this option for clients that do not run a Windows NT operating system. • **Only when no user is logged on**— Prevents the program from running until the user logs off the computer. This option is valid only for Windows NT clients. This option forces the program to run under the local Administrator account on the client. If a user logs on while the installation is running, the program continues to run. • **Whether or not a user is logged on**—Enables the program to run with or without user interaction. This option is valid only for Windows NT clients. This option forces the program to run under the local Administrator account on the client.
User input required	Requires that the user interact with the program when it runs. This check box is selected when Program can run is set to Only when a user is logged on. If the program does not require user input, clear this check box.

Table 21-19 • Programs View — Environment Tab Options (cont.)

Option	Description
Run mode	Select one of the following run modes: • **Run with user's rights** • **Run with administrative rights** To specify that the program requires administrative privileges on the computer in order to run, select Run with administrative rights. This option is available when Program can run is set to Only when a user is logged on. By default, this check box is cleared. This option is valid only for Windows NT clients. The **Run with administrative rights** option forces the program to run under the local Administrator account on the client.
Use software installation account	Run the program under an administrator-specified account (the Configuration Manager Windows NT Client Software Installation Account) on Windows NT clients. This check box is available only when Program can run is set to Only when no user is logged on or Whether or not a user is logged on. By default, this check box is cleared. This option should be used only when the program requires access to network resources other than the package files on the distribution point.
Drive mode	Select the drive mode used for network connections to the distribution points. • **Runs with UNC name**—Enables the program to run with a universal naming convention (UNC) name. This is the default setting. • **Requires drive letter**—Indicates that the program requires a drive letter to fully qualify its location, but Configuration Manager can use any available drive letter on the client. • **Requires specific drive letter**—Indicates that the program requires the drive letter you specify (A to Z) to fully qualify its location (for example, H). If the specified drive letter is already used on a client, the program will not run.
Reconnect to Distribution point at logon	The computer will reconnect to the drive when the user logs on. By default, this check box is cleared. ***Note*** • *This feature is not supported on 16-bit clients.*

Advanced Tab

The **Advanced** tab of the **Programs View** contains the following options:

Table 21-20 • Programs View — Advanced Tab Options

Option	Description
Run another program first	This program requires another program (in another package that is targeted for distribution using Configuration Manager Web Console or in the current package) to run before it. By default, this check box is cleared. ***Note*** • *This feature is not supported on 16-bit clients.* ***Note*** • *The program you specify to run first does not need to be advertised separately.*
Package	The package that contains the program that must run before this program.
Program	The program that must run first. ***Note*** • *If the depended-upon program fails on a client, the depending program will not run. Also, if the program that runs first has been run successfully, Configuration Manager will not automatically re-run it, even if the depending program is on a recurring schedule.*
When this program is assigned to a computer	When Program can run (in the **Environment** tab) is set to **Only when a user is logged on**, you can specify how the program is assigned to users. • **Run once for first user who logs on**—The program should run only once on the computer. This is the default setting. For example, a virus checker only needs to run once on the client. • **Run once for every user who logs on**—The program should run once for each new user who logs on. ***Note*** • *On 16-bit clients, advertised programs will run only for the first user who logs on.*

Table 21-20 • Programs View — Advanced Tab Options (cont.)

Option	Description
Remove software when it is no longer advertised	Indicates that Configuration Manager will remove this program from client computers when it is no longer advertised to those clients. This option is mainly useful for shared network applications and is available only to programs that register with Add/Remove Programs in Control Panel. To use this option, enter the name of the Uninstall Registry Key the program creates in the following registry key: `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\ CurrentVersion\Uninstall` After all advertisements for the program are revoked from the client, Configuration Manager looks up the program's uninstall command line in the `UninstallString` or `QuietUninstallString` property in this registry key. By default, this check box is cleared. An advertisement is considered revoked from a client when any of the following are true: • The client is no longer a member of the collection specified in the advertisement. • The advertisement is deleted. • The advertisement expires. • Uninstall registry key • The registry key that contains the uninstall command for this program. You can use up to 127 characters. ***Note*** • *This feature is not supported on 16-bit clients.*
Disable this program on computers where it is advertised	Temporarily disables all advertisements that contain this program. The program will be removed from the list available for users to run and will not be run via assignment until it is re-enabled. By default, this check box is cleared.

Advertisements View

On the Advertisements View, you identify the collection of users to receive this package, and specify when the package will be available for distribution and whether it is optional or mandatory. You can view a list of existing advertisements and create new advertisements.

Table 21-21 • Advertisements View Options

Option	Description
Advertisement Listing	List of all existing advertisements, displaying the following: • **Name**—Name of the advertisement • **Program**—The program to run on the client. • **Collection**—The target collection of computers, users, or user groups that will receive the program. • **Available After**—Date after which the advertisement will be available • **Expires After**—Expiration date of advertisement • **Advertisement ID**—Number uniquely identifying the advertisement • **Status**—Advertisement's status
Create New Advertisement	To create a new advertisement, enter a name in the Name text box and click Create. The new advertisement will now be listed under Advertisements in the tree.

If you select an existing advertisement under Advertisements in the tree, you can edit it on the detailed tab view (**General** and **Schedule**).

General Tab

To edit or delete an advertisement, select the advertisement icon under Advertisements in the tree. The General tab appears. Make edits in these fields and then click Update to save your edits. To delete the advertisement, click Delete.

Table 21-22 • Advertisements View — General Tab Options

Option	Description
Name	The name of the advertisement, up to 50 characters. This field is required.
Comment	Optional text about the advertisement, such as a description. You can use up to 127 characters.
Program	The program (within the current package) to advertise to clients. This field is required.

Table 21-22 • Advertisements View — General Tab Options (cont.)

Option	Description
Collection	Select the existing collection that will receive the advertisement. Click Browse to open the Select a Configuration Manager Collection Dialog Box, where you can select the Configuration Manager collection that you want to use in this Advertisement definition. This field is required. ***Note*** • *Configuration Manager will not advertise a program to 16-bit clients based on user groups or user accounts, but this is supported for 32-bit clients.*
Include members of subcollections	Specifies that the program also be advertised to clients and users in the specified collection's subcollections. By default, this check box is selected.

Schedule Tab

Make edits in these fields and then click **Update** to save your edits. To delete the advertisement, click **Delete**.

Table 21-23 • Advertisements View — Schedule Tab Options

Option	Description
Advertisement start time	The date and time when the program is advertised and available to run on clients. By default, the current date and time is specified. **Greenwich Mean Time**—Sets the time independent of time zones. That is, if you set the time to 1:00 P.M. and select this option, the program will be advertised on all clients simultaneously—1:00 P.M. for clients in the Greenwich mean time zone, 10:00 A.M. for clients in the Eastern time zone, and 7:00 A.M. for clients in the Pacific time zone, and so on. This check box is cleared by default, which enables you to specify a relative time for advertising the program. Configuration Manager will treat the specified time as a local time. In this case, if the time is set to 1:00 P.M., the program will be advertised on clients at 1:00 P.M. in each time zone.
Mandatory assignments (Assign immediately after this event)	If you want to ensure that the program runs on all clients (users cannot choose not to run the program), select: • As soon as possible • Logon • Logoff ***Note*** • *Recurring assignments are not supported on 16-bit clients. Programs advertised to 16-bit clients run only once on those clients.*

Table 21-23 • Advertisements View — Schedule Tab Options (cont.)

Option	Description
Assignments are not mandatory over slow links	Suspends assignments for clients on a slow link. By default, this check box is selected. (This option is available only if at least one assignment is defined.)
Allow users to run the program independently of assignments	Enables users to run the program independent of when it is assigned. By default, this check box is cleared. (This option is available only if at least one assignment is defined.) Typically, only non-assigned programs are visible to users in the Configuration Manager client interface. Use this option to display an assigned program in the client interface. For example, to require users to upgrade their Microsoft Office software, you specify an assignment in the advertisement properties. But if you want to enable users to run the installation program before it is scheduled to run automatically, select this option so that the assigned program appears in Advertised Programs Wizard in the Control Panel on client computers.
Advertisement will expire	Advertises the program to clients only for a limited time. By default, this option is cleared. This feature is not supported on 16-bit clients; the program appears until it runs on the client. ***Note*** • *When a program expires, it no longer appears in Advertised Programs Monitor or Advertised Programs Wizard in Control Panel on client computers.To specify the cutoff date and time, click the arrows next to the date and time. The default values are six months from the current date and time.* **Greenwich Mean Time**—Set the time independent of time zones. That is, if you set the time to 1:00 P.M. and select this option, the program will expire at all computers simultaneously—1:00 P.M. for clients in the Greenwich mean time zone, 10:00 A.M. for clients in the Eastern time zone, and 7:00 A.M. for clients in the Pacific time zone, and so on. This check box is cleared by default, which enables you to specify a relative time. Configuration Manager will treat the specified time as a local time. In this case, if the time is set to 1:00 P.M., the program will expire at 1:00 P.M. on clients in each time zone.
Priority	The advertisement's priority when sent to child sites: High, Medium, or Low. The default priority is Medium.

Package Summary Page

This page, which lists a summary of the information in the `.pdf` file that you are using for Configuration Manager distribution, appears after you have reviewed the `.pdf` file on the Configuration Manager Package Configuration Page.

Click **Commit Changes to Configuration Manager** to save the package on the Configuration Manager Server.

A message will appear informing you that the package changes were successfully saved to Configuration Manager.

Select a Configuration Manager Collection Dialog Box

This dialog box, which appears when you click **Browse** next to the **Collection** field on the **General** tab of the Advertisements View of the Configuration Manager Package Configuration Page, lists all defined collections on the Configuration Manager Server.

Select the collection that you want to use for the Advertisement definition, and click **OK**.

Distribution Settings Page

On the **Distribution Settings** page, which opens when you click **Distribution Settings** on the **Configuration Manager Web Console** Home Page, you set the default Configuration Manager Server connection settings.

Table 21-24 • Distribution Settings Page Options

Option	Description
Site Server Name	Enter the machine name on which Configuration Manager Server is installed. (This could be a local or remote machine.) When you make a selection in this field, the Site Code field is populated with a list of sites available on the Configuration Manager Server.
Site Code	Select the Configuration Manager site you want to use for software distribution.
User Name	Enter the user name appropriate for the Configuration Manager Site you want to connect to.
Update	Click to save your entries.

Note • *The values on entered on this page will be used as the default values for any new package you target for distribution. However, you can change these values while configuring the package.*

Package Administration Page

On this page you can view the status of distributed applications, modify settings of distributed packages, and delete an existing distribution.

Table 21-25 • Package Administration Page Options

Option	Description
Configuration Manager Package ID	Package ID assigned to this package by Configuration Manager.
Package Name	Name of the package.
Version	Version of the package.
Language	Language that the package was written for.
Description	Description of package.
Status	Lists the distribution status of the package: **Distributed** or **Not Distributed**.
Application Name	Name of application.
Company Name	Manufacturer of application.
Modify Settings	Click to modify the selected package's distribution settings. After you connect to the appropriate Configuration Manager Server, the Configuration Manager Package Configuration Page appears, where you can modify the distribution settings.
View Status	Click to view the status of a distributed application. Configuration Manager Web Console connects to the Configuration Manager Server, retrieves the status of that application, and displays that status as a text message.
Delete Distribution	Click to delete the selected application from Configuration Manager Web Console and/or the Configuration Manager Server.

Package Status Page

The Package Status page opens when you select a Distribution Package on the Package Administration Page, and click **View Status**.

A summary of this distributed package status is displayed in two main sections: **Package Status** and **Advertisement Status**.

Package Status

The Package Status section includes the following information:

Table 21-26 • Package Status Page — Package Status Information

Option	Description
Site	Server location where package was distributed.
Distribution Point	Distribution point where this package was distributed. A distribution point is a site system role that stores software package files so clients can access them during the software distribution process.
State	Identifies the state of the source files on the distribution point. Values are: • INSTALLED (0) • INSTALL_PENDING (1) • INSTALL_RETRYING (2) • INSTALL_FAILED (3) • REMOVAL_PENDING (4) • REMOVAL_RETRYING (5) • REMOVAL_FAILED (6)
Last Copied	Date and time (GMT) when the package source files were last successfully copied to the distribution point.
Source Version	The number of times an administrator has updated the package source files and requested an update to distribution points (specified on the Data Source tab of the Package Properties dialog box).
Targeted	The number of distribution points this package was sent to.
Installed	The number of distribution points that received this package.
Retrying	The number of distribution points that are having problems receiving the package, but that Configuration Manager is still attempting to deliver to.
Failed	The number of distribution points that failed to receive the package. Failure means Configuration Manager tried to deliver the package several times, but could not do so successfully.
Summary Date	Date and time (GMT) when a change in package status for the sites was most recently reported.
Path	NAL path to the package's source files.

Advertisement Status

The **Advertisement Status** section includes the following information

Table 21-27 • Package Status Page — Advertisement Status Information

Option	Description
Advertisement Name	Name of the advertisement.
Site	Server location where package was distributed.
Target Collection	Name of the collection to which this advertisement is advertised
Program	Name of the program (related to **PackageID)** that this advertisement advertises.
Available After	A program available to members of a collection after this time.
Expires After	A program will no longer be available to members of a collection after this time.
Received	Total number of users, user groups, or client computers that are reporting successful receipt of the advertisement.
Failures	Total number of users, user groups, or client computers that experienced an error in processing the advertisement or its associated program, or that attempted to run the advertised program but failed before the program could be started.
Programs Started	Total number of users, user groups, or client computers in the site that were able to successfully start running the advertised program.
Program Errors	Total number of users, user groups, or client computers that reported errors while running the advertised program. A program is considered in error when it produces either: 1. A non-zero exit code, or 2. An install-status MIF file with a failure-status attribute. This file, if present, will override an exit code.
Program Success	Total number of users, user groups, or client computers, or both, that are reporting that the advertisement ran successfully. A program is considered successful when it produces either: 1. A exit code of zero, or 2. An install-status MIF file with a success-status attribute. This file, if present, will override an exit code.
Advertisement ID	Unique identifier that Configuration Manager assigns to each advertisement.

Confirm Delete Page

The **Confirm Delete** page opens when you select a package on the Package Administration Page, and click **Delete Distribution**.

The messages shown on this page depend upon the status of the selected package: **Distributed** or **Not Distributed**:

- For both package status types, you are prompted to confirm that you want to delete the package from the Application Catalog.
- If you are deleting a **Distributed** package, you are also prompted to confirm that you want to delete this package from Configuration Manager.

Select **Yes** or **No** to indicate your deletion preferences:

Table 21-28 • Package Deletion Options

Package Status	Delete from Catalog?	Delete from Configuration Manager?
Not Distributed	Choose **Yes** to confirm the deletion from the Application Catalog or choose **No** to decline the deletion. If you choose **Yes**, the package will be deleted from the Application Catalog, along with all of the package configuration information that you entered on the **Configuration Manager Package Configuration** page.	N/A
Distributed	Choose **Yes** to confirm the deletion from the Application Catalog, or choose **No** to decline the deletion. If you delete a package from Configuration Manager but do not delete it from the Application Catalog, all of the package configuration information that you entered on the **Configuration Manager Package Configuration** page is saved, making it easy to distribute this package again in the future.	Choose **Yes** to confirm the deletion from Configuration Manager, or choose **No** to decline the deletion. Unless you are certain that you will never want to distribute this package again, it is recommended that if you delete a package from Configuration Manager, you do not delete it from the Application Catalog.

Click **OK** to confirm your selections.

Caution • *Deleting a distributed package from Configuration Manager will also delete its programs and any advertisements of the programs. If the package has source files, they will be removed from distribution points. If there are access accounts for this package, they will be deleted. Also, Configuration Manager Administrators' security rights to the package will be deleted.*

Caution • *When you delete a package from the Application Catalog, all package configuration information is lost, even if you do not delete it from Configuration Manager. If you delete a package from the Application Catalog but not from Configuration Manager, the package would still be available on the Configuration Manager Server, but all connection between the Application Catalog and Configuration Manager is lost. Should you want to distribute that package again in the future, you would have to re-import the application into the Application Catalog, and then re-enter package configuration information on the Configuration Manager Package Configuration Page.*

Distribute New Package Page

Edition • *This feature is included with AdminStudio Standard Edition only.*

In the **Application Name** field, enter the name of the application that you want to distribute to a Configuration Manager Server, and click **Distribute** to continue.

22

Analyzing Installations Prior to Deployment

Edition • *Predeployment Test is included with AdminStudio Professional and Enterprise Editions.*

The Predeployment Test tool allows system administrators to determine if a Windows Installer `.msi` package will succeed or fail when it is installed in production. Predeployment Test finds issues with disk space, install conditions, and will find conflicts with Windows installer packages already installed on the system.

The Predeployment Test user documentation is organized into the following sections:

Table 22-1 • Predeployment Test User Documentation

Section	Description
About the Predeployment Test Tool	This section provides an overview of the Predeployment Test tool and explains how it works.
Configuring the Predeployment Test Environment	This section explains how to configure the Predeployment Test Results Web Site and how to set the Predeployment Test Results Web site addresses.
Performing Predeployment Analysis	This section provides step-by-step procedures explaining how to generate and distribute a test-ready Windows Installer package (a package that does not make any system changes when it is deployed), and how to generate test reports.
Viewing Predeployment Test Results	This section describes each of the reports available on the Predeployment Test Results Web site.
Predeployment Test Reference	Every view, dialog box, and Wizard available in the Predeployment Test tool is discussed in this section.

About the Predeployment Test Tool

The Predeployment Test tool allows system administrators to determine if a Windows Installer `.msi` package will succeed or fail when it is installed in production. Predeployment Test finds issues with disk space, install conditions, and will find conflicts with Windows Installer packages already installed on the system.

This section includes the following topics:

- How the Predeployment Test Tool Works
- Predeployment Test Tool Components
- Overview of Predeployment Test Tool Results

Caution • *Predeployment Test features cannot be installed on 64 bit machines because Microsoft® .NET Framework 2.0 does not support 64 bit machines.*

How the Predeployment Test Tool Works

When using the Predeployment Test Tool to determine if a Windows Installer package will succeed or fail when it is installed in production, you perform the following steps:

- **Create test-ready packages**—Use the Predeployment Test Preparation Wizard to convert an original Windows Installer `.msi` package into a "test-ready" package which incorporates custom actions. A test-ready `.msi` package does not make any system changes when it is installed; it is used to collect data on whether that package would succeed or fail if it was installed on a particular machine. Using the wizard, you select the package to test and you specify deployment and conflict tests that you want to execute on the deployment machines.
- **Install test-ready packages**—You install the test-ready packages in your enterprise. When a test-ready `.msi` package is deployed, Type 1 `.msi` custom actions are executed. Each custom action performs a test and sends the results to a Web service, and the Web service saves those results into a database.
- **Analyze results**—You can then view the compiled test results in a Web browser.

Predeployment Test Tool Components

The Predeployment Test Tool consists of five components:

- **Predeployment Test Preparation Wizard**—This wizard-based tool converts the original `.msi` package into a "test-ready" `.msi` package, a package that does not make any system changes when it is deployed.
- **Custom Actions**—Type 1 `.msi` custom actions are executed when the "test-ready" `.msi` package is deployed. Each custom action performs a test and sends the results to the .NET Web service.
- **.NET Web Service**—This .NET Web service accepts the results from the custom action execution and saves them into database tables in the Application Catalog.

- **Database Tables**—Test results from the custom action execution are stored in database tables in the Application Catalog. The .NET Web service writes into this database.
- **Predeployment Test Results**—The Predeployment Test tool accesses the saved test results in the database tables. You can click a link on the **Predeployment Test Start Page** to view the test results in a Web browser.

Overview of Predeployment Test Tool Results

When each test-ready package is deployed, test results are added to its corresponding job on the Predeployment Test Results Web site and those results are displayed in the following reports:

Table 22-2 • Predeployment Test Results

Report Name	Description
Predeployment Test Results By Job Report	Lists all of the “test-ready” package jobs that have been run. Each time a test-ready package is installed on a machine in your network, test results are added to its corresponding job on the Predeployment Test Results Web site.
Job Summary Report	Lists the total number of times the test-ready package was run, and the number of times it Passed, Failed, and was Incomplete.
Test Result Summary Report	Lists the all the computers that this job was run on (All Results), or all the computers that had either: Passed, Failed, or Incomplete results.
Job Details Report	Lists all of the deployment and conflict tests that were run on this machine for this job.
Test Details Report	Lists detailed information on errors that were generated during a Deployment Test.
File/Registry Conflicts Report	Lists the results of File Conflict checks (ACE07, ACE23) and Registry Conflict checks (ACE10, ACE24).
Predeployment Test Results By Machine Report	Lists all of the machines in the network where “test-ready” packages have been installed.
Machine Result Summary Report	Lists all of the test-ready package jobs run on a specific machine.

Configuring the Predeployment Test Environment

To configure the Predeployment Test environment, you specify the following:

- **Predeployment Test Results Web Site**—Web site address where the Predeployment Test Results can be viewed, and the Application Catalog that stores the reports.
- **Predeployment Test Web Service**—Web site address of the Predeployment Test Web Service application, and the Application Catalog that stores the data generated when a test-ready application is installed.

This section includes the following topics:

- Configuring the Predeployment Test Results Web Site
- Configuring the Predeployment Test Web Service

Configuring the Predeployment Test Results Web Site

To configure the Predeployment Test Results Web site, open the Options Dialog Box and specify the following information:

- **Web Address**—Enter the URL of the Predeployment Test Results Web site. See Setting the Predeployment Test Results Web Site Address.
- **Application Catalog**—Configure a connection to an Application Catalog to store the Predeployment Test results. You will connect to this database to view the reports. Configuring the Predeployment Test Results Web Site Application Catalog

Note *• To use a Microsoft Access Application Catalog database on the Predeployment Test Results Web site, the database file must have ASP.NET permissions assigned to it. Files in the* `AdminStudio Shared` *directory inherit ASP.NET permissions from that directory. If you want to open a Microsoft Access database that is in another directory, you must assign ASP.NET permissions to that database file or the directory that it is in. See Assigning ASP.NET Permissions to Files or Folders.*

Setting the Predeployment Test Results Web Site Address

Task: ***To configure the Predeployment Test Results Web site:***

1. Launch the Predeployment Test tool. The **Predeployment Test Start Page** opens.
2. On the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. In the **URL** field of the **Predeployment Test Results Web Site** area, enter the Web address of the Predeployment Test Results Web site.

When Predeployment Test is installed, the default Predeployment Test Results URL is:

`http://IPADDRESS/Predeploytestreports/default.aspx`

where `IPADDRESS` is the Predeployment Test Service virtual directory address.

4. Click **OK**.

Configuring the Predeployment Test Results Web Site Application Catalog

Task: ***To configure the Predeployment Test Results Web Site Application Catalog:***

1. Launch the Predeployment Test tool. The **Predeployment Test Start Page** opens.
2. On the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. To select the Application Catalog that will store the reports, click **Configure Application Catalog** in the **Predeployment Test Results Web Site** area. The **Connection to New Catalog** page opens.
4. To configure a **Standalone** Application Catalog, perform the following steps:
   a. Open the **Standalone** tab.
   b. Enter the name of the **Server** to which you want to connect.
   c. For **How should SQL Server verify the authenticity of login ID?**, select one of the following options:
      - **With Windows NT authentication using the Network Login ID**—Use Windows network authentication when connecting to this database.
      - **With Database Server authentication using Login ID and password entered by user**—Use database server login identification for authentication when connecting to this database.
   d. If you choose **Database Server authentication**, enter the **Login ID** and **Password**.
   e. In the **Catalog Name** field, enter the name of an Application Catalog database available on the Microsoft SQL Server database.
   f. Click **Save** to set the selected database as the Application Catalog for the Predeployment Test Results Web site.
5. To configure the **AdminStudio Enterprise Server** Application Catalog, perform the following steps:
   a. Open the **Enterprise Server** tab. The address of the AdminStudio Enterprise Server is listed.
      - **If the AdminStudio Enterprise Server has been configured**, this address will be a URL to a running instance of an AdminStudio Enterprise Server.
      - **If the AdminStudio Enterprise Server has not yet been configured**, this address will be the default value of `http:\\localhost`.
   b. **If the AdminStudio Enterprise Server site has not yet been configured**, or if the URL is incorrect, click the address link to open the **Select AdminStudio Enterprise Server URL** page, enter the correct URL, and click **Save**.

c. On the **Enterprise Server** tab, enter a valid **User Name** and **Password** for the specified AdminStudio Enterprise Server.

d. Click **Login**.

Configuring the Predeployment Test Web Service

To configure the Predeployment Test Web Service, open the Options Dialog Box and specify the following information:

- **Web Address**—Enter the URL of Predeployment Test Web Service. See Setting the Predeployment Test Web Service Address.
- **Application Catalog**—Configure a connection to an Application Catalog to receive and store the test result data generated when test-ready packages are installed. See Configuring the Predeployment Test Web Service Application Catalog.

Note • *To use a Microsoft Access Application Catalog database on the Predeployment Test Results Web site, the database file must have ASP.NET permissions assigned to it. Files in the* `AdminStudio Shared` *directory inherit ASP.NET permissions from that directory. If you want to open a Microsoft Access database that is in another directory, you must assign ASP.NET permissions to that database file or the directory that it is in. See Assigning ASP.NET Permissions to Files or Folders.*

Setting the Predeployment Test Web Service Address

Task: ***To set the Predeployment Test Web Service address:***

1. Launch the Predeployment Test tool. The **Predeployment Test Start Page** opens.
2. On the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. In the **URL** field of the **Predeployment Test Web Service** area, enter the URL of the Predeployment Test Web Service.

   When Predeployment Test is installed, the default Predeployment Test Web Service URL is:

   `http://IPADDRESS/PreDeploytestservice/predeploytestservice.asmx`

   where `IPADDRESS` is the Predeployment Test Service virtual directory address.
4. Click **OK**.

Configuring the Predeployment Test Web Service Application Catalog

Task: ***To configure the Predeployment Test Web Service Application Catalog:***

1. Launch the Predeployment Test tool. The **Predeployment Test Start Page** opens.
2. On the **Tools** menu, select **Options**. The **Options** dialog box opens.
3. To select the Application Catalog that will receive and store the test result data generated when test-ready packages are installed, click **Configure Application Catalog** in the **Predeployment Test Web Service** area. The **Configure Application Catalog for Predeployment Web Service** page opens.
4. To configure a **Standalone** Application Catalog, perform the following steps:
   a. Open the **Standalone** tab.
   b. Enter the name of the **Server** to which you want to connect.
   c. For **How should SQL Server verify the authenticity of login ID?**, select one of the following options:
      - **With Windows NT authentication using the Network Login ID**—Use Windows network authentication when connecting to this database.
      - **With Database Server authentication using Login ID and password entered by user**—Use database server login identification for authentication when connecting to this database.
   d. If you choose **Database Server authentication**, enter the **Login ID** and **Password**.
   e. In the **Catalog Name** field, enter the name of an Application Catalog database available on the Microsoft SQL Server database.
   f. Click **Save**.
5. To configure the AdminStudio Enterprise Server Application Catalog, perform the following steps:
   a. Open the **Enterprise Server** tab. The address of the AdminStudio Enterprise Server is listed.
      - **If the AdminStudio Enterprise Server has been configured**, this address will be a URL to a running instance of an AdminStudio Enterprise Server.
      - **If the AdminStudio Enterprise Server has not yet been configured**, this address will be the default value of `http:\\localhost`.
   b. If the AdminStudio Enterprise Server site has not yet been configured, or if the URL is incorrect, click the address link to open the **Select AdminStudio Enterprise Server URL** page, enter the correct URL, and click **Save**.
   c. On the **Enterprise Server** tab, enter a valid **User Name** and **Password** for the specified AdminStudio Enterprise Server.
   d. Click **Login**.

Caution • *Predeployment Test features cannot be installed on 64 bit machines because Microsoft® .NET Framework 2.0 does not support 64 bit machines.*

Performing Predeployment Analysis

To analyze a package prior to deployment using the Predeployment Test tool, you create and distribute a test-ready Windows Installer package. You can then view the test results on the Predeployment Test Results Web site.

- Creating a Test-Ready Windows Installer Package
- Setting Predeployment Test Command Line Parameters
- Distributing a Test-Ready Windows Installer Package
- Viewing Predeployment Test Results
- Managing Job Data
- Setting Predeployment Test Command Line Parameters

Creating a Test-Ready Windows Installer Package

You use the **Predeployment Test Preparation Wizard** to convert an original Windows Installer `.msi` package into a "test-ready" `.msi` package, a package that does not make any system changes when it is deployed.

This test-ready `.msi` package incorporates Type 1 `.msi` custom actions which are executed when the package is deployed. Each custom action performs a test and sends the results to a Web service, and the Web service saves those results into a database. You can then view the test results in a Web browser.

Task: ***To create a test-ready Windows Installer package:***

1. Launch the Predeployment Test tool. The **Predeployment Test Start Page** opens.
2. Click the **Predeployment Test Preparation Wizard** link. The **Welcome Panel** of the **Predeployment Test Preparation Wizard** opens.
3. Click **Next**. The **Windows Installer File Selection Panel** opens.
4. In the **Source .MSI Package** field, select the Windows Installer `.msi` package that you want to test with the Predeployment Test tool—the `.msi` package that is being prepared for deployment.

   If you launched the Predeployment Test Preparation Wizard from the Application Manager by selecting a package and selecting **Distribute Package** from the context menu, the name in the **Source .MSI Package** field is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository:

- **Not in the Software Repository**—The full name and path of the Windows Installer file is displayed, and you can edit this entry or click **Browse** and select a different package.
- **In the Software Repository**—Only the name of the Windows Installer file is displayed (not the full path) and this entry cannot be edited or changed.

5. In the **Target .MSI Package** field, specify the name and location of the test-ready `.msi` package that will be generated at the end of this wizard.

   This field is automatically populated with the name of the package specified in the **Source .MSI Package** field with `_analysis` appended to the end of the name. For example, if `WinZip.msi` is selected as the **Source .MSI Package**, the **Target .MSI Package** will automatically be named `WinZip_Analysis.msi`. You can edit the name and the location of this file, if desired.

   If you launched the Predeployment Test Preparation Wizard from the Application Manager by selecting a package and selecting **Distribute Package** from the context menu, and the package is not in the Software Repository, the full name and path of the Target .MSI Package is displayed, but if the package is in the Software Repository, only the name of the Target .MSI Package is displayed (not the full path).

6. Click **Next**. The **Web Service URL Panel** opens.
7. In the **Web Service URL** field, enter the HTTP Web Service URL where you want to post the Predeployment Test results. This field automatically displays the Web Service URL that was entered the last time the wizard was run. By default, this field is set to:

   `http://`*`serveraddress`*`/PreDeployTestService/PreDeployTestService.asmx`

8. In the **Job Identifier** field, enter a title to identify the Predeployment Test results in the analysis reports. By default, this field is populated with the package name.
9. Click **Next**. The **Select Application Catalog Panel** opens.
10. Select the Application Catalog that will store the test results. Choose one of the following options:
    - **Send test results to the Application Catalog registered with the Web Service**—Select this option to send the Predeployment Test results to the Application Catalog registered with the Web Service. That Application Catalog's Server and Catalog names are listed
    - **Send test results to an alternate Application Catalog**—Click Browse to open the **Select Application Catalog** dialog box and select an Application Catalog to receive the test results.
11. Click **Next**. The **Deployment & Conflict Test Selection Panel** opens.
12. Select one or more deployment and conflict tests that you want to execute on the deployment machines when the test-ready package is deployed. Selecting a folder automatically selects all tests in that folder. Selecting **Predeployment Tests** automatically selects all deployment and conflict tests in all folders.

    The selections made on this panel are saved in an XML file (`PDTests.xml` in the **Support** subfolder of the AdminStudio installation folder) so that the same test selections are displayed the next time the wizard is run.

13. Click **Next**. The **Review Panel** opens.
14. Review the settings you specified in this wizard, and then click **Start** to initiate the generation of the test-ready `.msi` package. The **Progress Panel** opens, and indicates the progress of the testing.

    To edit a setting before initiating testing, click **Back** to return to previous panels.

15. When testing is complete, the **Next** button is activated. Click **Next** to proceed to the **Summary Panel**.
16. The **Summary Panel** reports that the test-ready `.msi` package was successfully generated, and lists a summary of the package settings. Perform one of the following options:
    a. **To exit the wizard without launching distribution** of the test-ready `.msi` package, click **Finish**. Then to distribute the test-ready package at a later time, launch the **Distribution Wizard**, select the test-ready `.msi` package on the **Installation Package Information Panel**, and proceed with distribution.
    b. **To automatically launch the Distribution Wizard** so that you can distribute this test-ready `.msi` package immediately, select the **Distribute the test-ready Windows Installer .msi package now** option and click **Finish**. The **Distribution Wizard** is launched with the generated test-ready `.msi` package automatically selected as the distribution source. You can then use the distribution method of your choice to distribute the test-ready package.

Setting Predeployment Test Command Line Parameters

This section includes the following topics:

- Command Line Parameters for the Predeployment Test Preparation Wizard
- Command Line Arguments for Distributing Test-Ready Packages

Command Line Parameters for the Predeployment Test Preparation Wizard

The Predeployment Test tool supports command-line parameters so that all the information entered in the **Predeployment Test Preparation Wizard** can also be provided via command line. The Predeployment Test Preparation Wizard works in silent mode when the following command line parameters are provided:

Table 22-3 • Predeployment Test Command Line Parameters

Parameter	Description
A	Application Catalog to use
S	Source Windows Installer `.msi` package file
T	Target test-ready .Windows Installer msi package file that will be generated
W	Web service URL
I	Test identifier
C	Comma-separated list of conflict tests to run
D	Comma-separated list of deployment tests to run

The following is an example of how to use these command line parameters:

```
PreDeploymentWizard.exe -S"c:\temp\my.msi" -T"c:\temp\My_Analysis.msi" -W"http://localhost/
    MyWebService.asmx"

PreDeploymentWizard.exe -C"File,Registry" -
    D"AppSearch,LaunchCondition,FileWrite,RegistryPermissions" -I"Test1"
```

Command Line Argument to Specify an Application Catalog

You can use the `A` command line argument to specify the Application Catalog that the Predeployment Test Preparation Wizard should use:

```
-A ConnectionString
```

When `-A` is not specified, the Wizard will always use the Web Service catalog. Using `-A` implicitly enables you to specify an alternate Application Catalog for the Wizard to use.

The connection string must be the complete OLE DB connection string including provider information.

Command Line Arguments for Distributing Test-Ready Packages

All test service settings are configurable from the command line when running a test-ready package. Here is a list of installer properties and their affect on the running package.

Table 22-4 • Command Line Arguments for Test-Ready Package

Property	Description
PD_ALTERNATE_CATALOG_PROVIDER	Property defining the database vendor for target Application Catalog. This property is not used unless the following property is defined: PD_ALTERNATE_CATALOG_CXN_STRING
PD_ALTERNATE_CATALOG_CXN_STRING	Connection string for an alternate Application Catalog. When this property is defined, the Web Server is forced to use an alternate Application Catalog. This property must contain the full connection string required to connect to the application catalog.

Distributing a Test-Ready Windows Installer Package

On the last panel of the **Predeployment Test Preparation Wizard**, the **Summary Panel**, you can select an option to automatically launch the **Distribution Wizard** so that you can distribute this test-ready `.msi` package immediately.

Task: ***To distribute a test-ready Windows Installer .msi package:***

1. Generate a test-ready Windows Installer `.msi` package, as described in **Creating a Test-Ready Windows Installer Package**.
2. On the **Summary Panel** of the **Predeployment Test Preparation Wizard**, select the **Distribute the test-ready Windows Installer .msi package now** option and click **Finish**.

   The **Distribution Wizard** will launch with the generated test-ready `.msi` package automatically selected as the distribution source. You can then use the distribution method of your choice to distribute the test-ready package.

Note • *If you have already completed the Predeployment Test Preparation Wizard and did not select the* ***Distribute the test-ready Windows Installer .msi package now*** *option on the* ***Summary Panel****, launch the Distribution Wizard and select the test-ready .msi package on the* ***Installation Package Information Panel****, and proceed with distribution.*

Running Diagnostic Tests When Distributing Test-Ready Packages

You can use the Predeployment Test Diagnostic Tool to run diagnostic tests on the critical parts of the predeployment reporting framework.

Diagnostic Tests Performed

The Diagnostic Tool performs the following tests:

- Does the specified Web Service exist?
- Can the Predeployment Test Tool connect to the Web Service?
- Does the Web Service have an Application Catalog specified?
- Does the package have an Application Catalog specified?
- Displays the details of the target Application Catalog.
- Can the Web Service connect to the Application Catalog?
- Can the Web Service create and retrieve the test data?
- Can the results XML file be posted to the Web Service?

- Can the Web Service open the results XML file and validate the contents?

The Diagnostic Tool is embedded into test-ready packages, enabling you to test machines where the Predeployment Test tool is not installed, and is launched via the command line.

Running the Diagnostic Tool

When running in diagnostic mode, the test-ready package runs just as it does when not running in diagnostic mode. The difference is that when the results are posted, additional diagnostic information is displayed to assist in troubleshooting.

To have a test-ready package launch the Diagnostic Tool, set the `PD_RUN_DIAGNOSTICS` property. This property can be set:

- in the property table, or
- on the command line when launching the instrumented package.

```
msiexec.exe /i pkgname_analysis2.msi PD_RUN_DIAGNOSTICS=1
```

Task: ***To run the Diagnostic Tool during test-ready package installation:***

1. Open the Command Prompt and enter the following command to launch installation of the test-ready package:

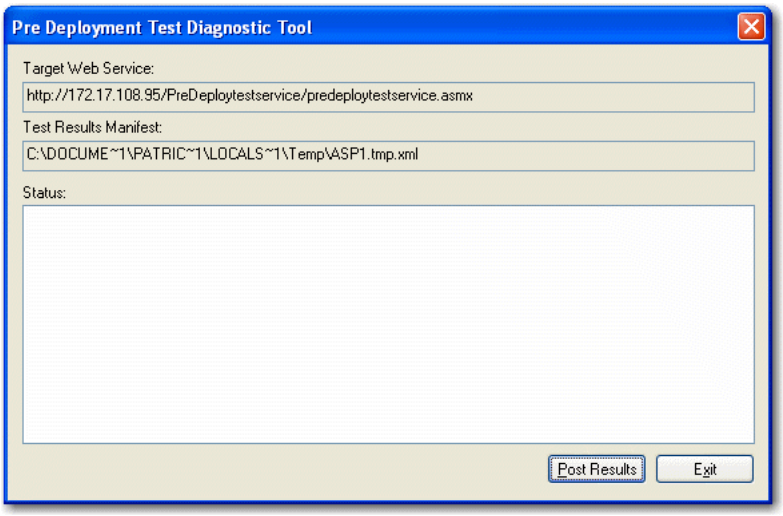
   ```
   msiexec.exe /i pkgname_analysis.msi PD_RUN_DIAGNOSTICS=1
   ```

   If you want to send the results to a log file in the same directory as the test-ready package, add the following parameter:

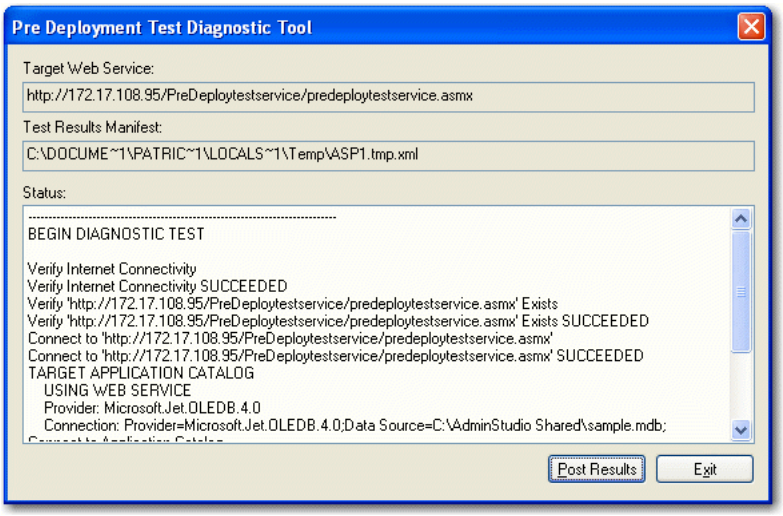
   ```
   /l*v log.log
   ```

   where *log.log* is the name of the log file.

   The package installation will begin. When the installation is almost complete, the Diagnostic Tool interface opens:



2. Click **Post Results**. The results of the diagnostic tests are then listed in the **Status** window.



3. Review the results of the diagnostic tests. If you find an error message that could be resolved by making a minor adjustment to your system, such as adjusting your server settings, you could make that adjustment and click **Post Results** again. The Diagnostic Tool will run the diagnostic tests again (without having to launch the test-ready package installation again) and generate new results. You can then check to see if your adjustment fixed the problem.

4. When you have finished reviewing the results of the diagnostic tests, click **Exit**.

   The installation of the test-ready package now finishes, and the results of the diagnostic tests are entered in the Predeployment Test Web Service Application Catalog, and, if you specified the log option, into the log file.

Note • *If an instrumented package is run in silent mode and* `PD_RUN_DIAGNOSTICS` *is set, the Diagnostic Tool interface will not be displayed, but it will still run the diagnostic tests and write the results to the log file.*

Viewing Predeployment Test Results

Use the **View Predeployment Test Results** link on the **Predeployment Test Start Page** to view the Predeployment Test Results.

Task: ***To generate Predeployment Test results:***

1. Launch the Predeployment Test tool. The **Predeployment Test Start Page** opens.

2. Select **View Predeployment Test Results**. The Predeployment Test Results By Job Report opens in a Web browser, listing all of the test-ready packages that you have installed. All of the other reports can be accessed from this report.

The following table lists all of the available reports and how to access them:

Report Name	Description	How To View
Predeployment Test Results By Job Report	Lists all of the "test-ready" package jobs that have been run. Each time a test-ready package is installed on a machine in your network, test results are added to its corresponding job on the Predeployment Test Report Web site.	On the **Predeployment Test Start Page**, click **View Predeployment Test Results**.
Job Summary Report	Lists the total number of times the test-ready package was run, and the number of times it **Passed**, **Failed**, and was **Incomplete**.	On the Predeployment Test Results By Job Report, click the **Job Name** link.
Test Result Summary Report	Lists the all the computers that this job was run on **(All Results)**, or all the computers that had either: **Passed**, **Failed**, or **Incomplete** results.	On the Job Summary Report, click the **All Results**, **Passed**, **Failed**, or **Incomplete** link. If the total number of tests in one of the result type categories is 0, the result type category name is not a link.
Job Details Report	Lists all of the deployment and conflict tests that were run on this machine for this job.	On a **Test Result Summary Report**, click on a **Computer Name** link.
Test Details Report	Lists detailed information on errors that were generated during a Deployment Test.	Click on a **Test Name** link in the **Deployment Test Results** list on the Job Details Report. (If no errors were found for that Test, the Test Name will not be a link.)
File/Registry Conflicts Report	Lists the results of the File Conflict checks (ACE07, ACE23) and the Registry Conflict checks (ACE10, ACE24).	On the Job Details Report, click the **File Conflicts** or **Registry Conflicts** link in the **Conflict Test Results** list.
Predeployment Test Results By Machine Report	Lists all of the machines in the network where "test-ready" packages have been installed.	On the Predeployment Test Results By Job Report, click the **Click here to see results by Machine** link.

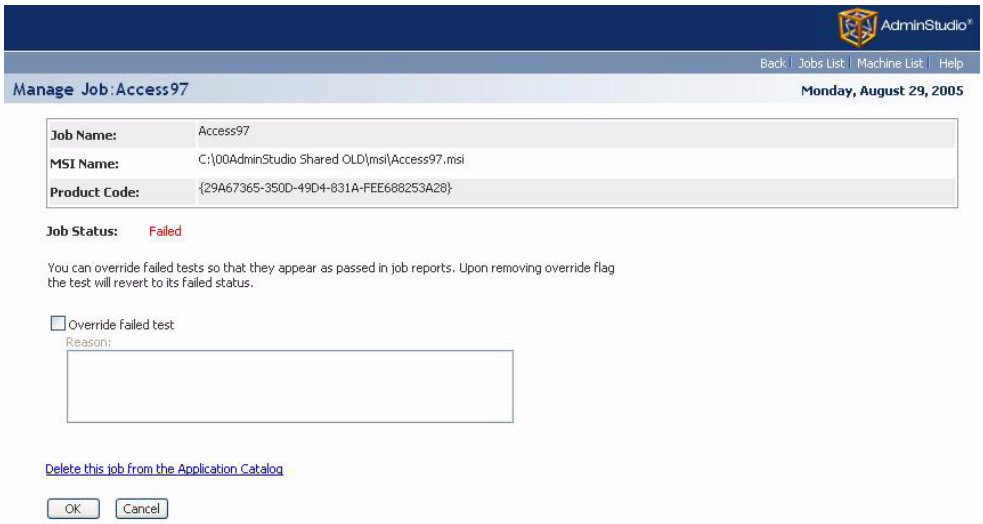
Report Name	Description	How To View
Machine Result Summary Report	Lists all of the test-ready package jobs run on a specific machine.	On the Predeployment Test Results By Machine Report, click on a **Machine Name** link.

Managing Job Data

After a Predeployment report is created, you can use the **Manage Job** feature to:

- Set a job with a **Failed** status to a **Passed** status, and enter a comment to explain why the status was changed.
- Delete the job.

If you click the **Manage Job** link on a **Job Summary Report** or **Job Details Report**, the **Manage Job Page** opens.

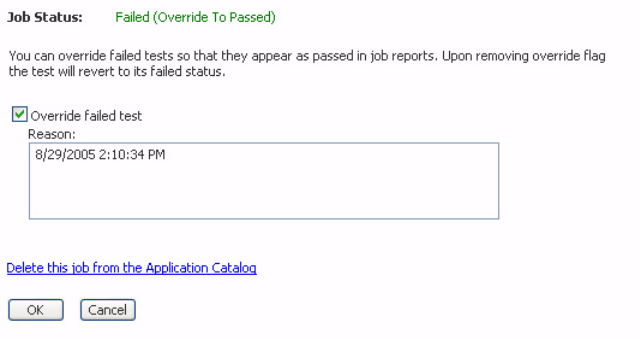


On the Manage Job Page, you can change the **Job Status** of a job from **Failed** to **Passed** or delete the job from the Application Catalog.

Changing the Status of a Job from Failed to Passed

Task: ***To change the Job Status of a job:***

1. Open the **Job Summary Report** or **Job Details Report** for a **Failed** job and click **Manage Job**. The **Manage Job Page** opens.
2. Select the **Override failed test** check box in the **Job Status** area. The job's status is then changed to **Failed (Override to Passed)**.



3. Enter a comment in the **Reason** text box to document why you changed this job's status and click **Save Changes**. The **Job Status** for this job is now listed as **Passed** in the Predeployment Test Results.

Marking a **Failed** job as being **Passed** does not modify any of the underlying test detail failure information that would normally be included in the **Job Details Report** or the **Test Details Report**. The original test results are always preserved for future reference, even though the predeployment test summary can be modified to appear as if the deployment tests succeeded.

Deleting a Job

Task: ***To delete a job:***

1. Open the Job Summary Report or Job Details Report for a job and click **Manage Job**. The Manage Job Page opens.
2. Click **Delete this job from the Application Catalog**.

Managing a Job Globally or in the Context of a Machine

This **Manage Job** page can be used to manage a job globally or to manage the job only in the context of a given machine. When the **Manage Job** page is launched from a machine-specific report page, the name of the hyperlink will change to **Delete the job data for this machine** and the check box label will change to **Mark test as Passed for this machine**.

Predeployment Test Reference

Predeployment Test Reference includes the same topics displayed when you click a help button from a Wizard or dialog box in the Predeployment Test interface. The following reference topics are available:

- Wizards and Dialog Boxes
- Technical Information
- Predeployment Test Results
- ConflictSolver ACE Reference

Wizards and Dialog Boxes

- Predeployment Test Start Page
- Predeployment Test Preparation Wizard
- Options Dialog Box
- Manage Job Page
- Configure Application Catalog for Predeployment Web Service Page

Predeployment Test Start Page

The Predeployment Test tool allows system administrators to determine if a Windows Installer `.msi` package will succeed or fail when it is installed in production. AdminStudio will find issues with disk space, install conditions, and will find conflicts with `.msi` packages already installed on the system.

The Predeployment Test Start Page is launched by selecting the Predeployment Test icon on the AdminStudio Start Page or in the Tools Gallery. You can also launch it from the **Test Page** of the Process Assistant.

Predeployment testing is a three-step process:

Table 22-5 • Predeployment Test Process

#	Step	Description
1	**Create "Test-Ready" Package**	Select this option to launch the **Predeployment Test Preparation Wizard** to prepare a package for predeployment testing. See Creating a Test-Ready Windows Installer Package for more information.
2	**Distribute "Test-Ready" Package**	Predeployment testing does not occur until the test-ready package is distributed. Select the **Launch Distribution Wizard** option to distribute a test-ready `.msi` package. Or, you can distribute the test-ready package using another distribution method. See Distributing a Test-Ready Windows Installer Package.
3	**View Test Results**	Select this option to view the Predeployment Test Results. See Viewing Predeployment Test Results for more information.

Predeployment Test Preparation Wizard

The AdminStudio Predeployment Test feature helps to determine if a Windows Installer `.msi` package will succeed or fail when it is deployed in production. This wizard is the first step in the Predeployment analysis process.

The Predeployment Test Preparation Wizard consists of the following panels:

- Welcome Panel
- Windows Installer File Selection Panel
- Web Service URL Panel
- Select Application Catalog Panel
- Deployment & Conflict Test Selection Panel
- Review Panel
- Progress Panel
- Summary Panel

Welcome Panel

The first panel of the Predeployment Test Preparation Wizard welcomes you to the Wizard. Click **Next** to proceed to the Windows Installer File Selection Panel.

This panel, and others in the Wizard, have four buttons located at the bottom of the Wizard. Depending on where you are in the Wizard, certain buttons may be disabled. The buttons are:

Table 22-6 • Wizard Buttons

Button	Description
Next	Advances you to the next panel in the Wizard.
Back	Moves you to the previous panel in the Wizard.
Cancel	Terminates the Wizard.
Help	Brings up help about the specific panel.

Windows Installer File Selection Panel

On this panel you specify the location of the original `.msi` package that is being prepared for deployment, and the name and location of the test-ready `.msi` package that you want to be generated at the end of this wizard.

Table 22-7 • Windows Installer File Selection Panel Options

Option	Description
Source .MSI Package	Select the Windows Installer `.msi` package that you want to test with the Predeployment Test tool. In the **Source .MSI Package** field, select the Windows Installer `.msi` package that you want to test with the Predeployment Test tool—the `.msi` package that is being prepared for deployment. If you launched the Predeployment Test Preparation Wizard from the Application Manager by selecting a package and selecting **Perform Pre-Deployment Testing** from the context menu, the name in the **Source .MSI Package** field is already entered. The ability to edit this entry depends upon whether the package you are distributing is managed by the Software Repository: • **Not in the Software Repository**—The full name and path of the Windows Installer file is displayed, and you can edit this entry or click **Browse** and select a different package. • **In the Software Repository**—Only the name of the Windows Installer file is displayed (not the full path) and this entry cannot be edited or changed.
Target .MSI Package	Specify the name and path of the "test-ready" `.msi` package that this wizard will create. This field is automatically populated with the name of the package specified in the Source .MSI Package field with _analysis appended to the end of the name. For example, if `WinZip.msi` is selected as the Source .MSI Package, the Target .MSI Package will automatically be named `WinZip_Analysis.msi`. You can edit the name and the location of this file, if desired. If you launched the Predeployment Test Preparation Wizard from the Application Manager by selecting a package and selecting **Perform Pre-Deployment Testing** from the context menu, and the package is not in the Software Repository, the full name and path of the Target .MSI Package is displayed, but if the package is in the Software Repository, only the name of the Target .MSI Package is displayed (not the full path).

Web Service URL Panel

On the Web Service URL Panel, enter the Web Service address where the Predeployment Test Web Service application is installed. Predeployment Test analysis test results will be posted to this Web Service.

When Predeployment Test is installed, the default location of the Predeployment Test Web Service is:

`http://IPADDRESS/PreDeploytestservice/predeploytestservice.asmx`

where *IPADDRESS* is the virtual directory address of the Predeployment Test Web Service application.

Table 22-8 • Web Service URL Panel Options

Option	Description
Predeployment Test Web Service URL	Enter the HTTP Web Service URL where you want to post the Predeployment Test results. This field automatically displays the Web Service URL that was entered the last time the wizard was run. By default, this field is set to: `http://adminstudioserver/PreDeployTestService/PreDeployTestService.asmx`
Job Identifier	Enter a title to identify the Predeployment Test results in the analysis reports. By default, this field is populated with the package name.

Select Application Catalog Panel

On this Panel, select the Application Catalog that will store the test results that are posted to the Predeployment Test Web Service.

Table 22-9 • Select Application Catalog Panel Options

Option	Description
Send test results to the Application Catalog registered with the web service	Select this option to send the Predeployment Test results to the Application Catalog registered with the Web Service. That Application Catalog's **Server** and **Catalog** names are listed. If the Web Service selected on the Web Service URL Panel has been configured with an Application Catalog, this radio button will be selected by default the first time the Predeployment Test Preparation Wizard is run. On subsequent runs, this panel will default to the data entered in the previous run of the Wizard.

Table 22-9 • Select Application Catalog Panel Options (cont.)

Option	Description
Send test results to an alternate Application Catalog	Click **Browse** to open the **Select Application Catalog Dialog Box** and select an Application Catalog to receive the test results. If the Web Service selected on the Web Service URL Panel has not been configured with an Application Catalog, this radio button will be selected by default the first time the Predeployment Test Preparation Wizard is run, and the Application Catalog that AdminStudio is currently connected to will be selected. On subsequent runs, this panel will default to the data entered in the previous run of the Wizard. Registering is not required, but can significantly streamline the process of creating test-ready packages. In addition, when you select an "alternate" package, if you ever want to change the Application Catalog, you would have to rebuild all of your test-ready packages. With registered catalogs you just change the Application Catalog registered with the test service and then all of the new test runs will be committed to the newly registered catalog. It would be expected that the most common uses of the alternate package setting would be for people evaluating the tool and for people who need to occasionally target alternate Application Catalogs. ***Note*** • *If you select a Microsoft Access database, you must enter a path relative to the machine running the test service virtual directory.* ***Note*** • *You are permitted to specify one Application Catalog for the Web Service and specify a different Application Catalog in the instrumented package. In this instance, the catalog information specified in the instrumented package takes precedence.*

Deployment & Conflict Test Selection Panel

On the **Deployment & Conflict Test Selection Panel**, select one or more **Deployment** and **Conflict** tests that you want to execute on the deployment machines when the test-ready package is deployed.

The following **Deployment** tests are available:

Table 22-10 • Deployment Tests

Test	Description
System Search	Locates a particular file, folder, registry key, `.ini` file value, or `.xml` file value on a target system prior to installation. Essentially, this feature lets you perform application, version, and configuration data searches.
Launch Conditions	Evaluates a set of conditions that must return **True** if the installation is to continue. If any condition fails, the user is presented with an error message and the installation ends.

Table 22-10 • Deployment Tests (cont.)

Test	Description
Directory Creation Permissions	Checks to determine if the necessary directory creation permissions exist for product installation.
File Read Permissions	Checks to determine if the necessary file read permissions exist for product installation.
Disk Space	Checks to determine if there is enough disk space for successful product installation.
Registry Permissions	Checks to determine if the necessary Registry permissions exist for product installation.
.NET Framework	Checks to see if the .NET Framework was installed correctly.
File Association	Checks to see if all file extensions have been installed and associated correctly. This involves launching a file with this extension, and determining if the correct application was used.

The following **Conflict** tests are available:

Table 22-11 • Conflict Tests

Test	Description
File	File Conflict Checks (ACE07, ACE23).
Registry	Registry Checks (ACE10, ACE24).

Selecting a folder automatically selects all tests in that folder, and selecting the **Predeployment Tests** folder automatically selects all **Deployment** and **Conflict** tests in all folders.

Note • *The **Deployment & Conflict Test Selection Panel** automatically displays the selections that were made during the previous run of this wizard. These selections are saved in an XML file, PDTests.xml, in the **Support** subfolder of the AdminStudio installation directory.*

Note • *If you select to include File or Registry conflict tests, it will take a longer time to complete the analysis during deployment of a test-ready package.*

Review Panel

On this panel you can review the settings you specified in this wizard, and then initiate the testing.

Click Start to initiate the generation of the test-ready `.msi` package and proceed to the **Progress Panel**. To edit a setting before initiating testing, click Back to return to previous panels.

***Note** • If a test-ready `.msi` file with same name as the package you are generating already exists, it will be overwritten.*

Progress Panel

The Progress Panel indicates the progress of the testing. When testing is complete, the **Next** button is activated. Click **Next** to proceed to the **Summary Panel**.

Summary Panel

The Summary Panel lets you know that the test-ready `.msi` package was successfully generated, and lists a summary of the package settings.

Click **Finish** to exit the Wizard.

Or, to automatically launch the **Distribution Wizard** so that you can distribute this test-ready `.msi` package, select the **Distribute the test-ready Windows Installer .msi package** now option and click **Finish**. The Distribution Wizard will be launched with the generated test-ready `.msi` package automatically selected as the distribution source.

Options Dialog Box

Enter information on the Options dialog box to configure the Predeployment Test Results Web Site and the Predeployment Test Web Service. To open the **Options** dialog box, open the **Predeployment Test Start Page** and click **Options** on the **Tools** menu.

Table 22-12 • Predeployment Options Dialog Box Options

Option	Description
Predeployment Test Results Web Site	Configure the Predeployment Test Results Web Site, the Web location where predeployment test results are displayed in Report Format. See Predeployment Test Results for report examples. Specify the following: • **URL**—Enter the URL of the Predeployment Test Results Web site. • **Configure Application Catalog**—Click to open the Assigning ASP.NET Permissions to Files or Folders to configure a connection to an Application Catalog to store the Predeployment Test results.

Table 22-12 • Predeployment Options Dialog Box Options (cont.)

Option	Description
Predeployment Test Web Service	Configure the Predeployment Test Web Service, the application that receives predeployment test result data. Specify the following: • **URL**—Enter the URL of the Predeployment Test Web Service. • **Configure Application Catalog**—Click to open the Assigning ASP.NET Permissions to Files or Folders to configure a connection to an Application Catalog that will store the test result data generated when test-ready packages are installed.

Configure Application Catalog for Predeployment Web Service Page

In Predeployment Test, you can connect to an SQL Server Application Catalog. The procedure for opening an Application Catalog is the same as the procedure used in the AdminStudio interface, Application Manager, and ConflictSolver. See Connecting to an Existing Application Catalog.

Technical Information

This section includes the following topics:

- Predeployment Custom Properties
- Predeployment Test Identifiers
- Predeployment Test Results XML File
- Changes Made During Test-Ready Package Creation

Predeployment Custom Properties

The **Predeployment Test Preparation Wizard** adds some custom properties to the test-ready MSI package. These properties, which are used by the custom actions, can be passed as parameters to msiexec.exe to override their values in the `.msi` file.

Table 22-13 • Predeployment Custom Properties

Property	Description
PDTESTSTORUN	Comma separated list of deployment and conflict test identifiers to run. See Predeployment Test Identifiers for a list of these tests. If this property value is an asterisk (*), it indicates that all the tests will be run. This list populates the list shown on the Deployment & Conflict Test Selection Panel. The selections made on this panel are saved in an XML file (PDTests.xml in the Support subfolder of the AdminStudio program folder) so that the same test selections are displayed the next time the wizard is run.
PDJOBID	The descriptive test identifier.
PDPKGSRC	The original Windows Installer `.msi` package file.
PDWEBSERVICEURL	Web Service URL where the Predeployment Test results will be posted.

Predeployment Test Identifiers

The following terms identify the tests that are run during Predeployment testing:

```
AppSearch
LaunchCondition
CreateFolder
FileRead
FileWrite
DiskSpace
Registry
DotNetFramework
FileAssociation
FileConflicts
RegConflicts
```

A comma separated list of the above identifiers is saved as a property, PDTestsToRun, in the test-ready `.msi` package. This property value is then used by the custom actions to determine the tests to run during the analysis. The list of tests in the PDTestsToRun property populates the list shown on the Deployment & Conflict Test Selection Panel.

Predeployment Test Results XML File

Test results are sent to a .NET Web service and are saved into database tables in the Application Catalog. The Predeployment Test tool accesses the saved test results in the database tables and displays them in a Web browser. This is an example of an XML file containing those test results.

```
<div class="dtd-fragment">
<pre>
<Predeployment version="1.0" ComputerName="SCH01SINGHDXP"
   IP="172.17.100.22" OS="Windows Server 2003 version 5.2
   (Build 3790)" Language="English" UserName="JohnSmith"
   JobID="CorelDraw" RunStartDate="6/15/2004 18:19"
   xmlns="http://www.installshield.com/webservices/">
  <Original>
    <PackageCode>{FAD3134F-5474-4778-9E34-204662673F6D}</PackageCode>
    <ProductCode>{5A92C452-89F2-435C-8C13-26EE70F51059}</ProductCode>
    <Name>Corel Draw</Name>
    <Version>9.5</Version>
    <Language>1033</Language>
    <Source>\\server\SharedUsers\JohnSmith\_Performance\MSIs\
      Corel Draw.msi</Source>
  </Original>
  <Instrumentation NewPackageCode="{FAD3134F-5474-4778-9E34-
    204662673F6D}" NewProductCode="
    {59293687-ABCF-4e96-9CF5-5F3A08E2166A}" />
  <TestResults>
    <Deployment>
      <Test ID="AppSearch">
        <Result Total="0" Success="0" Failed="0" />
      </Test>
      <Test ID="LaunchCondition">
        <Result Total="0" Success="0" Failed="0" />
      </Test>
      <Test ID="CreateFolder">
        <Message Key="INSTALLDIR" Folder="C:\Program Files\Corel\
           Graphics9\" Result="0" />
        <Message Key="Directory26" Folder="C:\Program Files\
           Corel\Graphics9\Workspace\Corel Photo-Paint9
           _BootDefault\" Result="0" />
        <Message Key="WindowsFolder" Folder="C:\WINDOWS\" Result="0" />
        <Message Key="MESHWARP" Folder="C:\Program Files\Corel\
           Graphics9\Custom\MeshWarp\" Result="0" />
        <Message Key="TEMPLATE" Folder="C:\Program Files\Corel\
           Graphics9\Draw\Template\" Result="0" />
        <Message Key="PAPERTYPES" Folder="C:\Program Files\Corel\
           Graphics9\Papertypes\" Result="0" />
        <Message Key="SYMBOLS" Folder="C:\Program Files\Corel\
           Graphics9\Symbols\" Result="0" />
        <Message Key="BOOTDEFAULT" Folder="C:\Program Files\Corel\
           Graphics9\Workspace\CorelDRAW9\_BootDefault\" Result="0" />
        <Message Key="PROFILES" Folder="C:\WINDOWS\Profiles\"
           Result="0" />
        <Message Key="INSTALLDIR.A67717E9_4C97_4D36_947C_FC3157B0BD67"
           Folder="C:\WINDOWS\system32\" Result="0" />
        <Message Key="DIR51" Folder="C:\Documents and Settings\
```

```
   JohnSmith\Favorites\corel on the web\" Result="0" />
<Message Key="DIR52" Folder="C:\Program Files\Corel\
   Graphics9\color\" Result="0" />
<Message Key="DIR53" Folder="C:\Program Files\Corel\
   Graphics9\config\" Result="0" />
<Message Key="DIR54" Folder="C:\Program Files\Corel\
   Graphics9\custom\" Result="0" />
<Message Key="DIR55" Folder="C:\Program Files\Corel\
   Graphics9\custom\brushes\" Result="0" />
<Message Key="DIR56" Folder="C:\Program Files\Corel\
   Graphics9\custom\bumpmap\" Result="0" />
<Message Key="DIR57" Folder="C:\Program Files\Corel\
   Graphics9\custom\canvas\" Result="0" />
<Message Key="DIR58" Folder="C:\Program Files\Corel\
   Graphics9\custom\displace\" Result="0" />
<Message Key="DIR59" Folder="C:\Program Files\Corel\
   Graphics9\custom\duotone\" Result="0" />
<Message Key="DIR60" Folder="C:\Program Files\Corel\
   Graphics9\custom\frames\" Result="0" />
<Message Key="DIR61" Folder="C:\Program Files\Corel\
   Graphics9\custom\layouts\" Result="0" />
<Message Key="DIR62" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\" Result="0" />
<Message Key="DIR64" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\cmyk\misc\" Result="0" />
<Message Key="DIR65" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\cmyk\nature\" Result="0" />
<Message Key="DIR66" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\cmyk\people\" Result="0" />
<Message Key="DIR67" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\cmyk\things\" Result="0" />
<Message Key="DIR69" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\rgb\misc\" Result="0" />
<Message Key="DIR70" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\rgb\nature\" Result="0" />
<Message Key="DIR71" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\rgb\people\" Result="0" />
<Message Key="DIR72" Folder="C:\Program Files\Corel\
   Graphics9\custom\palettes\rgb\things\" Result="0" />
<Message Key="DIR73" Folder="C:\Program Files\Corel\
   Graphics9\custom\patterns\" Result="0" />
<Message Key="DIR74" Folder="C:\Program Files\Corel\
   Graphics9\custom\shearmap\" Result="0" />
<Message Key="DIR75" Folder="C:\Program Files\Corel\
   Graphics9\custom\tiles\" Result="0" />
<Message Key="DIR76" Folder="C:\Program Files\Corel\
   Graphics9\custom\tonecrve\" Result="0" />
<Message Key="DIR77" Folder="C:\Program Files\Corel\
   Graphics9\custom\userdef\" Result="0" />
<Message Key="DIR78" Folder="C:\Program Files\Corel\
   Graphics9\draw\" Result="0" />
<Message Key="DIR79" Folder="C:\Program Files\Corel\
   Graphics9\draw\custommediastrokes\" Result="0" />
<Message Key="DIR82" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\fountain\" Result="0" />
<Message Key="DIR83" Folder="C:\Program Files\Corel\
```

```
   Graphics9\draw\fill_out\fills\fountain\cylinder\"
   Result="0" />
<Message Key="DIR84" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\fountain\radial\"
   Result="0" />
<Message Key="DIR86" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\pattern\2color\" Result="0" />
<Message Key="DIR87" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\pattern\bitmap\" Result="0" />
<Message Key="DIR88" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\pattern\color\" Result="0" />
<Message Key="DIR89" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\texture\" Result="0" />
<Message Key="DIR90" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\texture\biology\"
   Result="0" />
<Message Key="DIR91" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\texture\elements\"
   Result="0" />
<Message Key="DIR92" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\texture\material\"
   Result="0" />
<Message Key="DIR93" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\fills\texture\rocks\"
   Result="0" />
<Message Key="DIR94" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\outlines\" Result="0" />
<Message Key="DIR95" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\outlines\arrows\" Result="0" />
<Message Key="DIR96" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\outlines\dot_dash\" Result="0" />
<Message Key="DIR97" Folder="C:\Program Files\Corel\
   Graphics9\draw\fill_out\outlines\width\" Result="0" />
<Message Key="DIR98" Folder="C:\Program Files\Corel\
   Graphics9\draw\find\" Result="0" />
<Message Key="DIR99" Folder="C:\Program Files\Corel\
   Graphics9\draw\net_fav\" Result="0" />
<Message Key="DIR100" Folder="C:\Program Files\Corel\
   Graphics9\draw\net_fav\links\" Result="0" />
<Message Key="DIR101" Folder="C:\Program Files\Corel\
   Graphics9\draw\samples\" Result="0" />
<Message Key="DIR103" Folder="C:\Program Files\Corel\
   Graphics9\draw\scripts\presets\" Result="0" />
<Message Key="DIR104" Folder="C:\Program Files\Corel\
   Graphics9\draw\scripts\scripts\" Result="0" />
<Message Key="DIR105" Folder="C:\Program Files\Corel\
   Graphics9\filters\" Result="0" />
<Message Key="DIR106" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\" Result="0" />
<Message Key="DIR107" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\brushtxr\" Result="0" />
<Message Key="DIR108" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\imglists\" Result="0" />
<Message Key="DIR109" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\net_fav\" Result="0" />
<Message Key="DIR110" Folder="C:\Program Files\Corel\
```

```
   Graphics9\photopnt\net_fav\links\" Result="0" />
<Message Key="DIR111" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\paths\" Result="0" />
<Message Key="DIR112" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\samples\" Result="0" />
<Message Key="DIR114" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\scripts\effects\" Result="0" />
<Message Key="DIR115" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\scripts\scripts\" Result="0" />
<Message Key="DIR116" Folder="C:\Program Files\Corel\
   Graphics9\photopnt\tables\" Result="0" />
<Message Key="DIR117" Folder="C:\Program Files\Corel\
   Graphics9\plugins\" Result="0" />
<Message Key="DIR118" Folder="C:\Program Files\Corel\
   Graphics9\plugins\digimarc\" Result="0" />
<Message Key="DIR119" Folder="C:\Program Files\Corel
   Graphics9\plugins\squizz\" Result="0" />
<Message Key="DIR120" Folder="C:\Program Files\Corel\
   Graphics9\programs\" Result="0" />
<Message Key="DIR121" Folder="C:\Program Files\Corel\
   Graphics9\programs\data\" Result="0" />
<Message Key="DIR122" Folder="C:\Program Files\Corel\
   Graphics9\register\" Result="0" />
<Message Key="DIR123" Folder="C:\Program Files\Corel\
   Graphics9\scripts\" Result="0" />
<Message Key="DIR125" Folder="C:\Program Files\Corel\
   Graphics9\shared\help\" Result="0" />
<Message Key="DIR126" Folder="C:\Program Files\Corel\
   Graphics9\trace\" Result="0" />
<Message Key="DIR127" Folder="C:\Program Files\Corel\
   Graphics9\trace\samples\" Result="0" />
<Message Key="DIR130" Folder="C:\Program Files\Corel\
   Graphics9\tutors\draw\htmldocs\" Result="0" />
<Message Key="DIR131" Folder="C:\Program Files\Corel\
   Graphics9\tutors\draw\htmldocs\htmlpics\" Result="0" />
<Message Key="DIR133" Folder="C:\Program Files\Corel\
   Graphics9\tutors\paint\htmldocs\" Result="0" />
<Message Key="DIR134" Folder="C:\Program Files\Corel\
   Graphics9\tutors\paint\htmldocs\htmlpics\" Result="0" />
<Message Key="DIR136" Folder="C:\Program Files\Corel\
   Graphics9\workspace\corel photo-paint9\" Result="0" />
<Message Key="DIR137" Folder="C:\Program Files\Corel\
   Graphics9\workspace\corel photo-paint9\adobe(r)
   photoshop(r)\" Result="0" />
<Message Key="DIR138" Folder="C:\Program Files\Corel\
   Graphics9\workspace\corel photo-paint9\corel
   photo-paint 8\" Result="0" />
<Message Key="DIR139" Folder="C:\Program Files\Corel\
   Graphics9\workspace\coreldraw9\" Result="0" />
<Message Key="DIR140" Folder="C:\Program Files\Corel\
   Graphics9\workspace\coreldraw9\adobe(r) illustrator(r)\"
   Result="0" />
<Message Key="DIR141" Folder="C:\Program Files\Corel\
   Graphics9\workspace\coreldraw9\coreldraw8\" Result="0" />
<Message Key="DIR145" Folder="C:\Program Files\corel\
   shared\writing tools\9.0\" Result="0" />
```

```
  <Message Key="DIR146" Folder="C:\WINDOWS\corel\" Result="0" />
  <Message Key="DIR148" Folder="C:\WINDOWS\corel\
     coreldraw 9\9\" Result="0" />
  <Message Key="DIR149" Folder="C:\Program Files\Corel\
     Graphics9\filters\convert\" Result="0" />
  <Message Key="SystemFolder" Folder="C:\WINDOWS\system32\"
     Result="0" />
  <Message Key="DIR29.5B60FF9E_851D_11D4_A752_00B0D0428C0C"
     Folder="Y:\Program Files\Common Files\Microsoft
     Shared\DAO\" Result="0" />
  <Message Key="INSTALLDIR.46C3930E_0E70_11D5_BF08_0010A4EF69FF"
     Folder="C:\Program Files\Borland\Common Files\Default\"
     Result="0" />
  <Message Key="Dir388" Folder="C:\Documents and Settings\
     JohnSmith\Start Menu\Programs\CorelDRAW 9\
     Graphics Utilities\" Result="0" />
  <Message Key="Dir387" Folder="C:\Documents and Settings\
     JohnSmith\Start Menu\Programs\CorelDRAW 9\" Result="0" />
  <Message Key="Dir389" Folder="C:\Documents and Settings\
     JohnSmith\Start Menu\Programs\CorelDRAW 9\
     Productivity Tools\" Result="0" />
  <Message Key="Dir390" Folder="C:\Documents and Settings\
     JohnSmith\Start Menu\Programs\CorelDRAW 9\
     Setup and Notes\" Result="0" />
  <Message Key="Directory16" Folder="C:\Program Files\Corel\
     Graphics9\Config\" Result="0" />
  <Message Key="Directory17" Folder="C:\Program Files\Corel\
     Graphics9\Custom\Tiles\" Result="0" />
  <Message Key="Directory5" Folder="C:\Program Files\Corel\
     Graphics9\Draw\" Result="0" />
  <Message Key="Directory19" Folder="C:\Program Files\Corel\
     Graphics9\Filters\" Result="0" />
  <Message Key="Directory20" Folder="C:\Program Files\Corel\
     Graphics9\Photopnt\" Result="0" />
  <Message Key="Directory22" Folder="C:\Program Files\Corel\
     Graphics9\Programs\Data\" Result="0" />
  <Message Key="Directory23" Folder="C:\Program Files\Corel\
     Graphics9\Register\" Result="0" />
  <Message Key="Directory24" Folder="C:\Program Files\Corel\
     Graphics9\Trace\" Result="0" />
  <Message Key="Directory27" Folder="C:\Program Files\Corel\
     Graphics9\Workspace\Corel Photo-Paint9\
     Adobe(R) Photoshop(R)\" Result="0" />
  <Message Key="TempFolder" Folder="C:\DOCUME~1\DARSHA~1\
     LOCALS~1\Temp\" Result="0" />
  <Message Key="CommonFilesFolder" Folder=
     "Y:\Program Files\Common Files\" Result="0" />
  <Message Key="StartupFolder" Folder="C:\Documents and Settings\
     JohnSmith\Start Menu\Programs\Startup\" Result="0" />
  <Result Total="112" Success="112" Failed="0" />
</Test>
<Test ID="FileRead">
  <Result Total="3399" Success="3399" Failed="0" />
</Test>
<Test ID="FileWrite">
  <Result Total="3399" Success="3399" Failed="0" />
```

```
      </Test>
      <Test ID="Registry">
        <Result Total="802" Success="802" Failed="0" />
      </Test>
      <Test ID="DotNetFramework">
        <Message AssembliesCount="0" V10="0" V11="1" V12="0" V20="0"
           InstallRoot="C:\WINDOWS\Microsoft.NET\Framework\" />
        <Result Total="1" Success="1" Failed="0" />
      </Test>
      <Test ID="DiskSpace">
        <Result Total="1" Success="1" Failed="0" />
      </Test>
      <Test ID="FileAssociation">
        <Result Total="11" Success="11" Failed="0" />
      </Test>
    </Deployment>
    <Conflict>
    </Conflict>
  </TestResults>
  <Conclusion>
    <RunEndDate>6/15/2004 18:20</RunEndDate>
    <DeploymentTests Run="9" Success="9" Failed="0" />
    <ConflictTests Run="0" Success="0" Failed="0" />
  </Conclusion>
</Predeployment>
</pre>
</div>
```

Changes Made During Test-Ready Package Creation

The Predeployment Test Preparation Wizard makes a copy of the selected source Windows Installer `.msi` package and makes the following changes to create a “test-ready” package. These changes are made to ensure that the test-ready package does not make any system changes when it is deployed.

- Add a Type 1 custom action for each test into the **CustomAction** table.
- Add or update the **EXECUTEMODE** property and set its value to None.
- Add or update the **REBOOT** property and set its value to ReallySuppress.
- Add or update the **ARPNOMODIFY** property and set its value to 1.
- Add or update the **ARPNOREMOVE** property and set its value to 1.
- Add or update the **ARPNOREPAIR** property and set its value to 1.
- Update the product code, and save the original product code in a different, new property.
- Update the product code in the Product table.
- Add other properties, such as Web service URL, test descriptive identifier, tests to run, etc. based on the data connected in the wizard.
- Remove the existing custom actions and their references from all the user interface and execute sequence tables.

- Update the product name.
- Delete all rows from **_Validation** table.

Predeployment Test Results

The following table lists all of the available Predeployment Test results and how to access them on the Predeployment Test Report Web site:

Table 22-14 • Predeployment Test Results

Report Name	Description	How To View
Predeployment Test Results By Job Report	Lists all of the "test-ready" package jobs that have been run. Each time a test-ready package is installed on a machine in your network, test results are added to its corresponding job on the Predeployment Test Report Web site.	On the **Predeployment Test Start Page**, click **View Predeployment Test Results**.
Job Summary Report	Lists the total number of times the test-ready package was run, and the number of times it **Passed**, **Failed**, and was **Incomplete**.	On the Predeployment Test Results By Job Report, click the **Job Name** link.
Test Result Summary Report	Lists the all the computers that this job was run on **(All Results)**, or all the computers that had either: **Passed**, **Failed**, or **Incomplete** results.	On the Job Summary Report, click the **All Results**, **Passed**, **Failed**, or **Incomplete** link. If the total number of tests in one of the result type categories is 0, the result type category name is not a link.
Job Details Report	Lists all of the deployment and conflict tests that were run on this machine for this job.	On a **Test Result Summary Report**, click on a **Computer Name** link.
Test Details Report	Lists detailed information on errors that were generated during a Deployment Test.	Click on a **Test Name** link in the **Deployment Test Results** list on the Job Details Report. (If no errors were found for that Test, the Test Name will not be a link.)
File/Registry Conflicts Report	Lists the results of the File Conflict checks (ACE07, ACE23) and the Registry Conflict checks (ACE10, ACE24).	On the Job Details Report, click the **File Conflicts** or **Registry Conflicts** link in the **Conflict Test Results** list.

Table 22-14 • Predeployment Test Results (cont.)

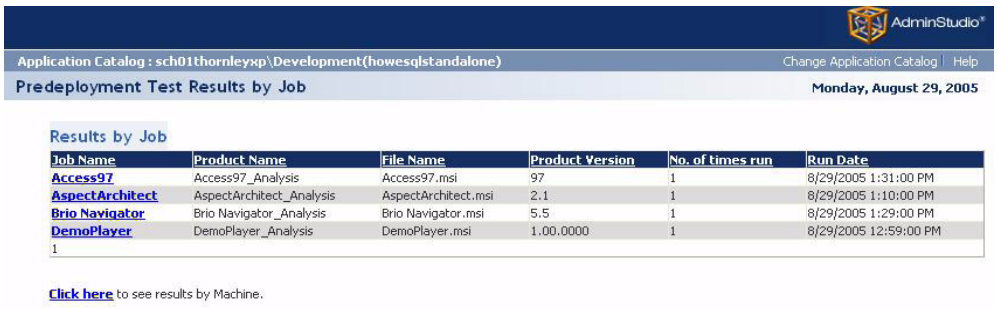
Report Name	Description	How To View
Predeployment Test Results By Machine Report	Lists all of the machines in the network where “test-ready” packages have been installed.	On the Predeployment Test Results By Job Report, click the **Click here to see results by Machine** link.
Machine Result Summary Report	Lists all of the test-ready package jobs run on a specific machine.	On the Predeployment Test Results By Machine Report, click on a **Machine Name** link.

The following Reference topics are also included:

- Connection to New Catalog Page
- Manage Job Page
- Assigning ASP.NET Permissions to Files or Folders

Predeployment Test Results By Job Report

The Predeployment Test Results By Job report lists all of the “test-ready” package jobs that have been run. Each time a test-ready package is installed on a machine in your network, test results are added to its corresponding job on the Predeployment Test Report Web site. This report is the first page you see when opening the Predeployment Test Report Web site.



The following information is included:

Table 22-15 • Predeployment Test Results By Job Report Data

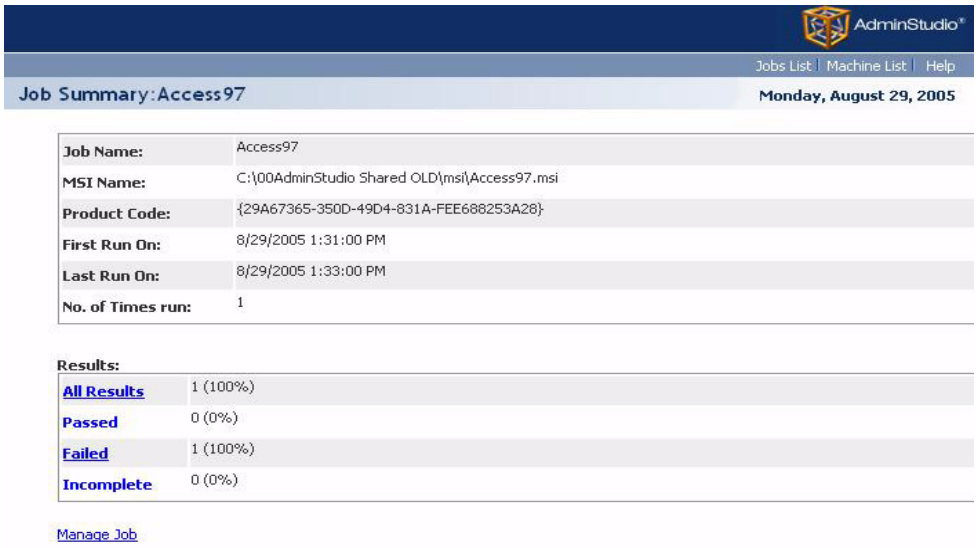
Item	Description
Job Name	Name of the test-ready package job. Corresponds to the Job Identifier that was entered on the Web Service URL Panel of the Predeployment Test Preparation Wizard when this test-ready package was created. Click on this link to view the Job Summary Report for this job.

Table 22-15 • Predeployment Test Results By Job Report Data (cont.)

Item	Description
Product Name	Name of the test-ready `.msi` package associated with this job. (The `.msi` extension is not shown.)
File Name	Name of the original `.msi` package that was used to create the test-ready package.
Product Version	Version of the original `.msi` package.
No. of times run	Number of times this job has been run (this test-ready package has been installed) on computers in your network.
Run Date	Date and time that this job was last run.
Change Application Catalog	Click to access the Connection to New Catalog Page, where you can open a new Application Catalog while you are using the Predeployment Test Results Web site.
Click here to see results by machine	Click to view the Predeployment Test Results By Machine Report.

Job Summary Report

The Job Summary report lists the total number of times the test-ready package was run, and the number of times it Passed, Failed, and was Incomplete. To access this report, go to the Predeployment Test Results By Job Report, and click the **Job Name** link.



The following information is included:

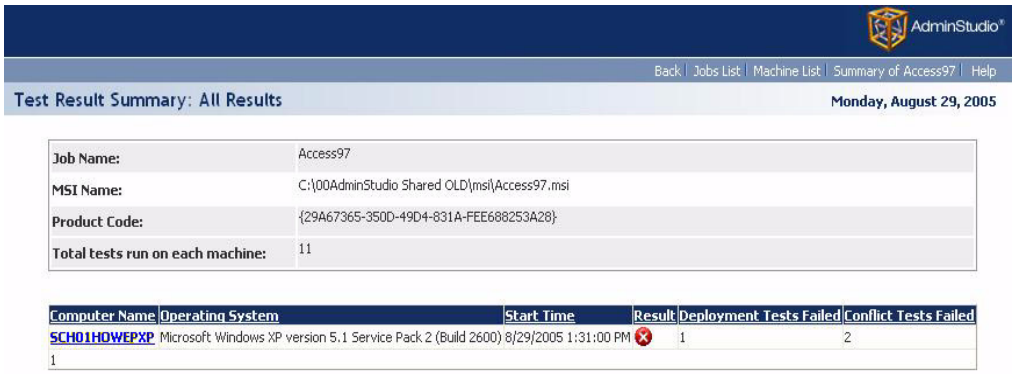
Table 22-16 • Job Summary Report Data

Item	Description
Job Name	Name of the test-ready package job. Corresponds to the Job Identifier that was entered on the Web Service URL Panel of the Predeployment Test Preparation Wizard when this test-ready package was created. Click on this link to view the Job Summary Report for this job.
MSI Name	Name and location of the test-ready `.msi` package associated with this job.
Product Code	Product GUID of the `.msi` package that was used to create the test-ready package.
First Run On & Last Run On	Date the test-ready package was run the first time and the most recent time.
No. of Times run	Number of times this job has been run (this test-ready package has been installed) on computers in your network.
Results	Four category totals are provided to summarize the test results: All Results, Passed, Failed, and Complete. Click on one of these links to view the Test Result Summary Report for the selected category. If there were no results in a category, the name of that category is not a link.
Manage Job	Click to open the Manage Job Page, where you can choose to override a failed test or delete this Job from the Application Catalog.
Jobs List	Click to access the Predeployment Test Results By Job Report.
Machine List	Click to access the Predeployment Test Results By Machine Report.

Test Result Summary Report

The Test Results Summary reports list the all the computers that this test-ready package job was run on (for the All Results report), or all the computers where the test-ready package tests Passed, Failed, or had Incomplete results.

To access the Test Results Summary Reports, open the Job Summary Report, and click the All Results, Passed, Failed, or Incomplete link. If the total number of tests in one of the result type categories is 0, the result type category name is not a link.



The following information is included:

Table 22-17 • Test Results Summary Report Data

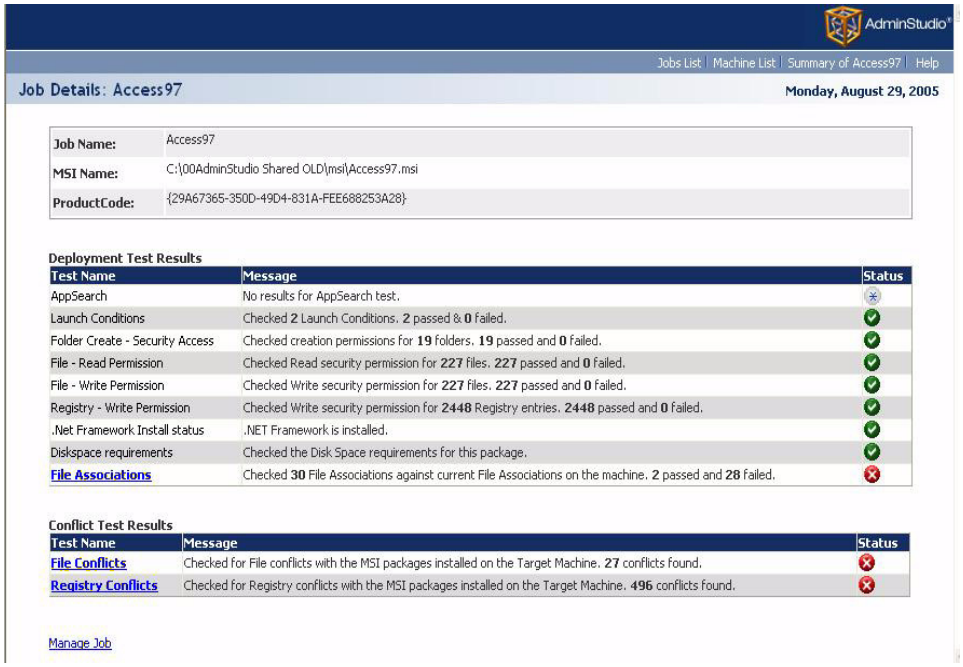
Item	Description
Job Name	Name of the test-ready package job. Corresponds to the Job Identifier that was entered on the Web Service URL Panel of the Predeployment Test Preparation Wizard when this test-ready package was created. Click on this link to view the Job Summary Report for this job.
MSI Name	Name and location of the test-ready `.msi` package associated with this job.
Product Code	Product GUID of the `.msi` package that was used to create the test-ready package.
Total tests run on each machine	The total number of Deployment and Conflict tests associated with this test-ready package.
Computer Name	Unique identifier of an individual computer in the network.
Operating System	Operating system of the machine.
Start Time	Time the installation of the test-ready package started on this machine.
Result	Icon indicating whether the test-ready package installation (and associated tests) **Passed** (), **Failed** (), or were **Incomplete.**
Deployment Tests Failed	Total number of deployment tests that failed during the installation of the test-ready package on this computer.
Conflict Tests Failed	Total number of conflict tests that failed during the installation of the test-ready package on this computer.
Jobs List	Click to access the Predeployment Test Results By Job Report.

Table 22-17 • Test Results Summary Report Data (cont.)

Item	Description
Machine List	Click to access the Predeployment Test Results By Machine Report.
Summary of [Test Name]	Click to view the Job Summary Report for this test-ready package job.

Job Details Report

The Job Details report lists all of the deployment and conflict tests that were run on this machine for this job. To access it, click the Computer Name link on a Test Result Summary Report.



The following information is included:

Table 22-18 • Job Details Report Data

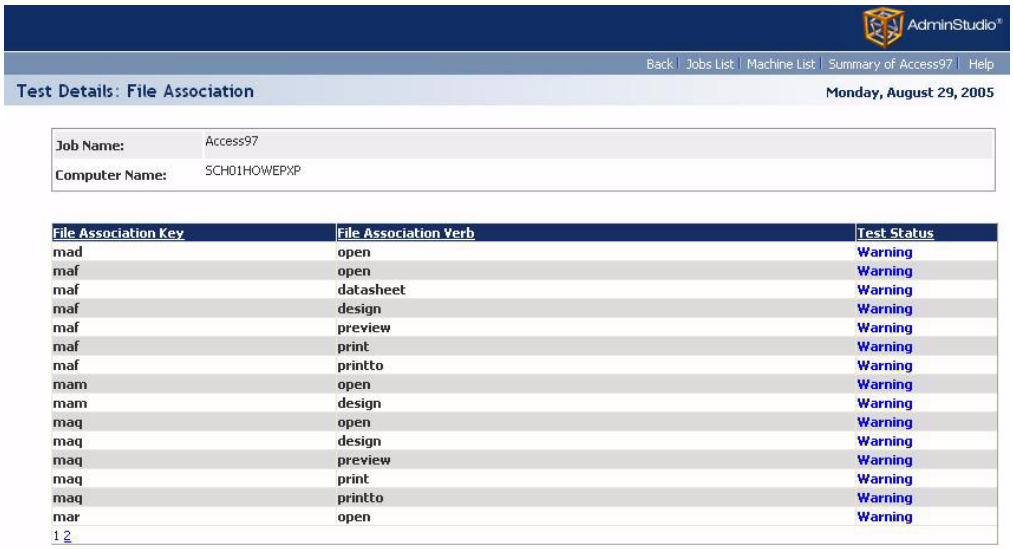
Item	Description
Job Name	Name of the test-ready package job. Corresponds to the Job Identifier that was entered on the Web Service URL Panel of the Predeployment Test Preparation Wizard when this test-ready package was created. Click on this link to view the Job Summary Report for this job.
MSI Name	Name and location of the test-ready `.msi` package associated with this job.
Product Code	Product GUID of the `.msi` package that was used to create the test-ready package.

Table 22-18 • Job Details Report Data (cont.)

Item	Description
Deployment and Conflict Test Results	For each deployment and conflict test that was included with this test-ready package, the following information listed: • **Test Name**—Name of the deployment or conflict test that was included with this test-ready package. • **Message**—Summary message generated when this deployment or conflict test was run on this computer indicating quantity of items (folders, files, Registry entries, etc.) checked, and how many of those items passed the test, and how many failed. • **Status**—Icon indicating the deployment or conflict test status: **Passed** (), **Failed** (), or **Incomplete**.
Manage Job	Click to open the Manage Job Page, where you can choose to override a failed test or delete this Job from the Application Catalog.
Jobs List	Click to access the Predeployment Test Results By Job Report.
Machine List	Click to access the Predeployment Test Results By Machine Report.

Test Details Report

The Test Details report lists detailed information on errors that were generated during a Deployment Test. To access it, click on a Test Name link in the **Deployment Test Results** list on the Job Details Report. (If no errors were found for that Test, the Test Name will not be a link.)



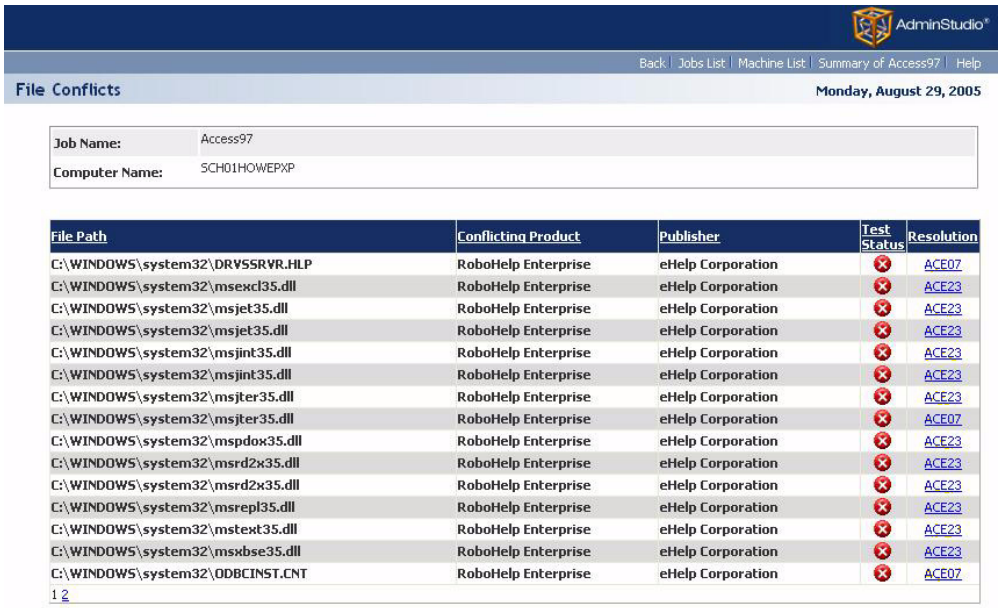
The following information is included:

Table 22-19 • Test Details Report Data

Item	Description
Job Name	Name of the test-ready package job. Corresponds to the Job Identifier that was entered on the Web Service URL Panel of the Predeployment Test Preparation Wizard when this test-ready package was created. Click on this link to view the Job Summary Report for this job.
Computer Name	Unique identifier of an individual computer in the network.
[Conflicting Item in Product]	The item in a product installed on the computer that the test-ready package was installed on that conflicted with something in the test-ready package. The information displayed in this report varies depending upon the type of test that you are viewing.
Test Status	Icon indicating whether the item generated a warning, failed to install: (), successfully installed: (), or was Incomplete.
Jobs List	Click to access the Predeployment Test Results By Job Report.
Machine List	Click to access the Predeployment Test Results By Machine Report.
Summary of [Job Name]	Click to access the Job Summary Report for this test-ready package job.

File/Registry Conflicts Report

The **File** / **Registry Conflicts Report** lists File or Registry conflicts that were detected against the selected package. To open this report, click the **File Conflicts** or **Registry Conflicts** link in the **Conflict Test Results** list on the Job Details Report.



For each test-ready package job, the following information is included:

Table 22-20 • Machine Result Summary Report Data

Item	Description
Job Name	Name identifying a test-ready package that was installed on this machine. The Job Identifier is entered on the Web Service URL Panel of the Predeployment Test Preparation Wizard. By default, the Job Identifier is the package name. Click on this link to view the Job Details Report.
Computer Name	Unique identifier of an individual computer in the network.
File Path	Path and file name of installation file or component that caused the conflict.
Conflicting Product	Products that ACE errors were detected against.
Publisher	Manufacturer of conflicting product.
Test Status	Icon indicating whether the test-ready package successfully installed on the machine: Passed (✓) or Failed (✗),

Table 22-20 • Machine Result Summary Report Data (cont.)

Item	Description
Resolution	Links to the ACE conflicts topics, which describe the ACE conflicts and also explain how to resolve them.

Information on detected ACE conflicts is presented in the following topics:

- Predeployment File Conflict Check: ACE07
- Predeployment File Conflict Check: ACE 23
- Predeployment Registry Check: ACE 10
- Predeployment Registry Check: ACE 24
- Resolving File Conflict Issues

Predeployment File Conflict Check: ACE07

ACE07 checks for the existence of the same file in components with different ComponentIds, and reports four types of errors, depending on the various combinations of source and target files and whether or not they originated in a merge module.

In Predeployment Test, component names are used to uniquely identify components across products, while in ConflictSolver, ACE07 uses the Component ID.

Predeployment File Conflict Check: ACE 23

ACE23 identifies file duplication between source and target packages. ACE23 checks to see if files with the same name and destination directory have the same size, version, and language when comparing a source package against a target package. If a file with the same name and destination directory is found in both the source and target packages, but the file has a different size, version, or language, ACE23 fails.

Predeployment Registry Check: ACE 10

ACE10 checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds.

In Predeployment Test, component names are used to uniquely identify components across products, while in ConflictSolver, ACE10 uses the Component ID.

Predeployment Registry Check: ACE 24

ACE24 checks to see if registry entries with the same registry hive, key, and value name have the same data type and value. If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target package, but the registry entry has a different data type or value, ACE24 fails.

Resolving File Conflict Issues

In the File Conflict Results Page, the **Resolution** column lists the name of the ConflictSolver ACE for the detected failure, and a link to view ACE resolution information.

Predeployment Test Results By Machine Report

The Predeployment Test Results by Machine report lists all of the machines in your organization where test-ready packages have been installed. One way to access this report, is to by clicking the **Click here to see results by Machine** link on the Predeployment Test Results By Job Report.

For each machine, the following information is included:

Table 22-21 • Predeployment Test Results by Machine Report Data

Option	Description
Machine Name	Name that identifies the machine on the network. Click this link to view the Machine Result Summary Report for this machine.
IP Address	Number that identifies the machine on the network.
Operating System	Operating system and version installed on the machine.
Language	Default language of operating system.
Total Jobs Run	Total number of test-ready packages that have been installed on the machine.
Jobs Passed and Jobs Failed	Number of test-ready packages that installed successfully on this machine and the number of test-ready packages that failed to install on this machine.
Last Run	Date and time that the last test-ready application was installed on the machine.
Click here to see results by Job.	Click to view the Predeployment Test Results By Job Report.
Jobs List	Click to view the Predeployment Test Results By Job Report.

Machine Result Summary Report

The Machine Result Summary report lists all of the test-ready package jobs run on a specific machine. To access this report, open the Predeployment Test Results By Machine Report, and click on a Machine Name link.

For each test-ready package job, the following information is included:

Table 22-22 • Machine Result Summary Report Data

Item	Description
Job Name	Name identifying a test-ready package that was installed on this machine. The Job Identifier is entered on the Web Service URL Panel of the Predeployment Test Preparation Wizard. By default, the Job Identifier is the package name. Click on this link to view the Job Details Report.
File Name	Name of the original source `.msi` package.
User Name	Name of the user who installed the test-ready package.
Job Started and Job Finished	Date and time when the installation of the test-ready package began and ended.
Result	Icon indicating whether the test-ready package successfully installed on the machine: Passed ( ) or Failed ( ),
Deployment Tests Failed	Total number of deployment tests that were included in the test-ready package that failed during installation. The tests that are included in a test-ready package are selected on the Deployment & Test Selection Panel of the Predeployment Test Preparation Wizard.
Conflict Tests Failed	Total number of conflict tests that were included in the test-ready package that failed during installation. The tests that are included in a test-ready package are selected on the Deployment & Test Selection Panel of the Predeployment Test Preparation Wizard.
Jobs List	Click to access the Predeployment Test Results By Job Report.
Machine List	Click to access the Predeployment Test Results By Machine Report.

Connection to New Catalog Page

You can choose to connect to a standalone Application Catalog database or the AdminStudio Enterprise Server Application Catalog database.

- Connecting to a Standalone Application Catalog
- Connecting to the AdminStudio Enterprise Server Application Catalog

Connecting to a Standalone Application Catalog

Task: ***To connect to a standalone Application Catalog:***

1. Open the Predeployment Test Reports Web site.
2. On the **Predeployment Test Results by Job** page, click **Change Application Catalog**. The **Connection to New Catalog** page opens.
3. Open the **Standalone** tab.
4. Enter the name of the **Server** to which you want to connect.
5. For **How should SQL Server verify the authenticity of login ID?**, select one of the following options:
   - **With Windows NT authentication using the Network Login ID**—Use Windows network authentication when connecting to this database.
   - **With Database Server authentication using Login ID and password entered by user**—Use database server login identification for authentication when connecting to this database.
6. If you choose **Database Server authentication**, enter the **Login ID** and **Password**.
7. In the **Catalog Name** field, enter the name of an Application Catalog database available on the server.
8. Click **Save**.

Connecting to the AdminStudio Enterprise Server Application Catalog

Task: ***To connect to a standalone Application Catalog:***

1. Open the Predeployment Test Reports Web site.
2. On the **Predeployment Test Results by Job** page, click **Change Application Catalog**. The **Connection to New Catalog** page opens.
3. Open the **Enterprise Server** tab. The address of the AdminStudio Enterprise Server is listed.
   - **If the AdminStudio Enterprise Server has been configured**, this address will be a URL to a running instance of an AdminStudio Enterprise Server.
   - **If the AdminStudio Enterprise Server has not yet been configured**, this address will be the default value of `http:\\localhost`.

4. **If the AdminStudio Enterprise Server site has not yet been configured**, or if the URL is incorrect, click the address link to open the **Select AdminStudio Enterprise Server URL** page, enter the correct URL, and click **Save**.
5. On the **Enterprise Server** tab, enter a valid **User Name** and **Password** for the specified AdminStudio Enterprise Server.
6. Click **Login**.

Manage Job Page

On the **Manage Job** page, you can change the **Job Status** of a job from **Failed** to **Passed** or delete the job from the Application Catalog.

If you click the **Manage Job** link on a **Job Summary Report** or **Job Details Report**, the **Manage Job** page opens. The following options are included:

Table 22-23 • Manage Job Page Options

Option	Description
Job Name	Name of the job that you are viewing.
MSI Name	Name and directory path of the test-ready package that was used to run this job.
Product Code	Product code of the test-ready package.
Job Status	Status of this job: • Failed • Passed Marking a **Failed** job as being **Passed** does not modify any of the underlying test detail failure information that would normally be included in the Job Details Report or the Test Details Report. The original test results are always preserved for future reference, even though the predeployment test summary can be modified to appear as if the deployment tests succeeded.
Override failed test	Select this check box to change the status of a **Failed** job to **Failed (Override to Passed)**. The job will then be listed in all other reports as **Passed**. If the **Manage Job** page was launched from a machine-specific report page, the name of this hyperlink is **Mark test as Passed for this machine**.
Reason	Enter comments to document the reason the job's status was changed.
Save Changes	Click to save your selections and return to the report.

Table 22-23 • Manage Job Page Options (cont.)

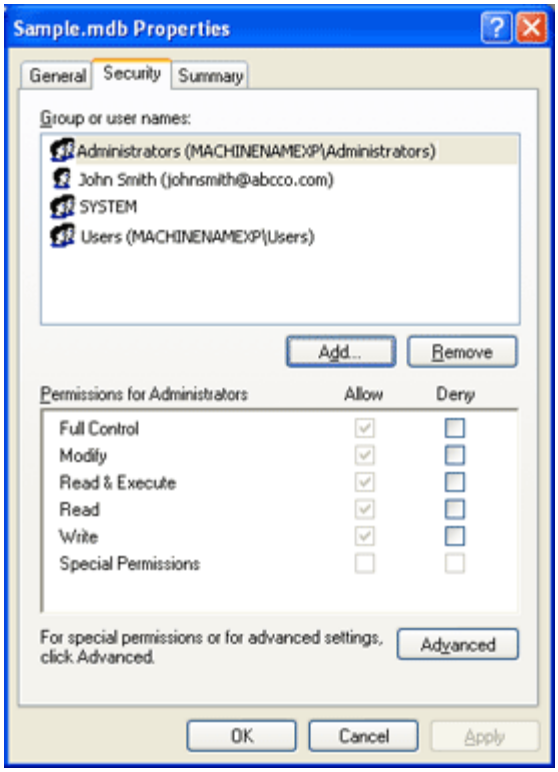
Option	Description
Delete this job from the Application Catalog	Click to delete this job from the Application Catalog. If the Manage Job page was launched from a machine-specific report page, the name of this hyperlink is **Delete the job data for this machine from the Application Catalog**.

Assigning ASP.NET Permissions to Files or Folders

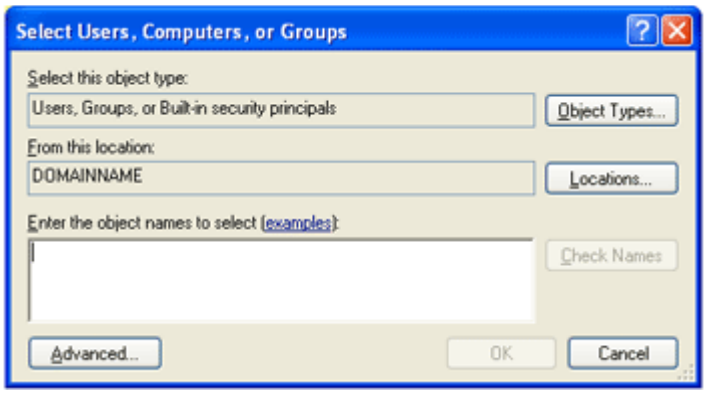
To use a Microsoft Access Application Catalog database on the Predeployment Test Results Web site, the database file must have ASP.NET permissions assigned to it. Files in the AdminStudio Shared directory inherit ASP.NET permissions from that directory. If you want to open a Microsoft Access database that is in another directory, you must assign ASP.NET permissions to that database file or the directory that it is in.

Task: ***To assign ASP.NET permissions to a file or folder:***

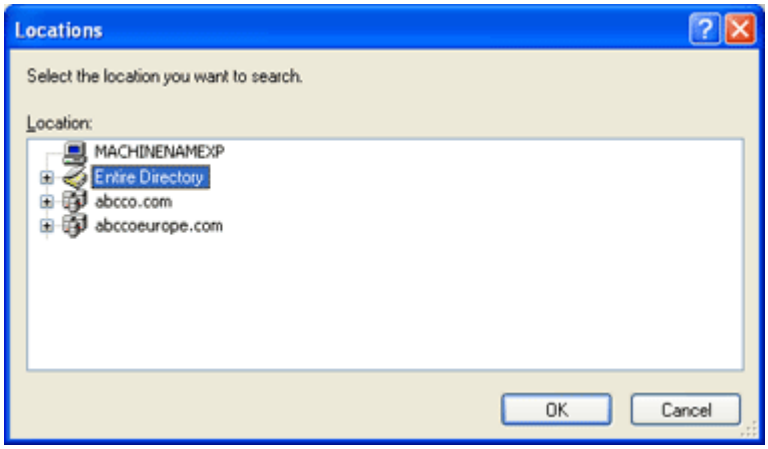
1. In Windows Explorer, select the file or folder that you want to assign ASP.NET permissions to select **Properties** from the context menu. The Properties dialog opens.
2. Click **Security** to open the **Security** tab.



3. Click **Add**. The **Select Users, Computers, or Groups** dialog box opens.



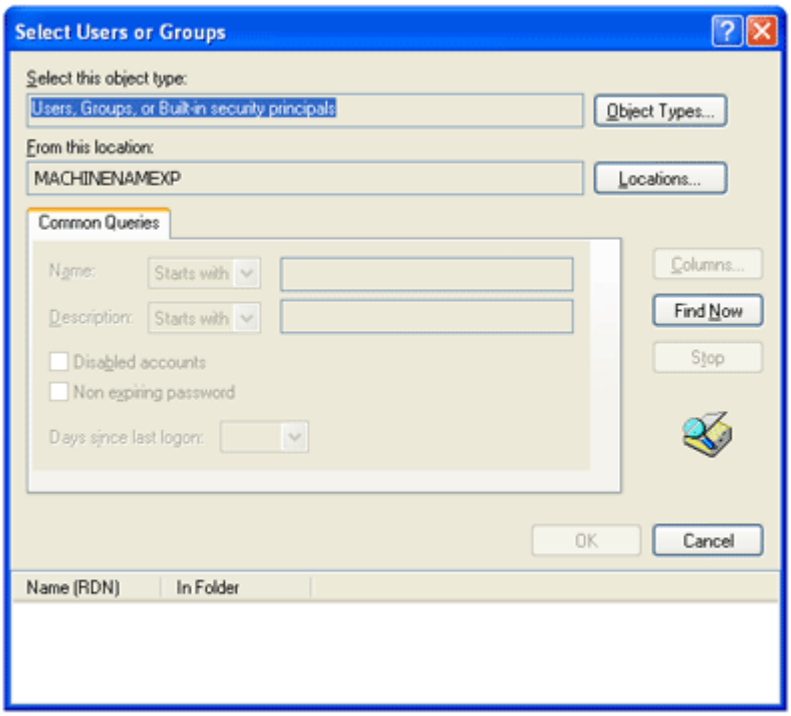
4. Click **Locations**. The **Locations** dialog box opens.



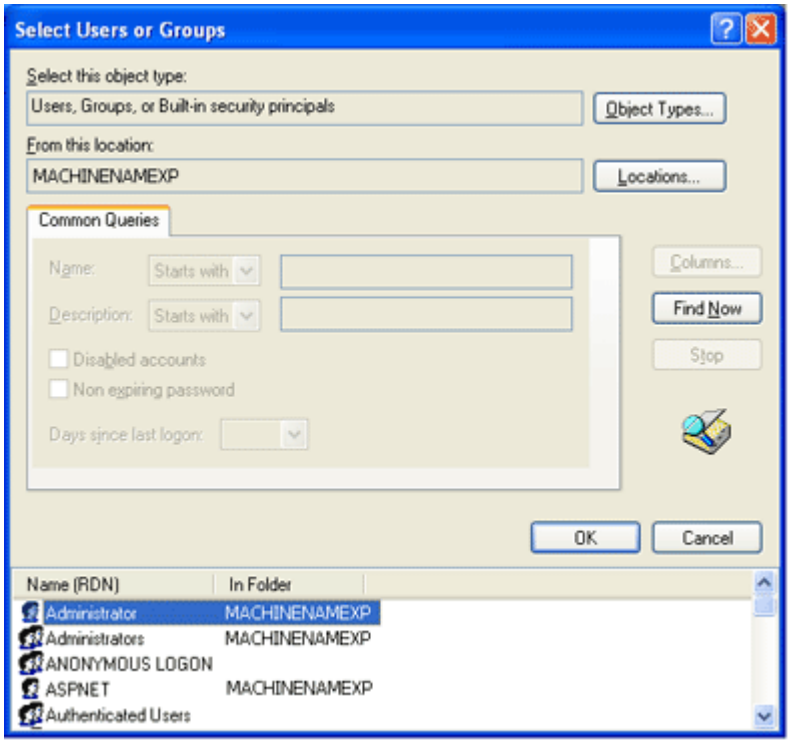
5. In the **Locations** list, select the name of your computer and click **OK** to return to the **Select Users or Groups** dialog box. The name of your computer is now listed in the **From this location** field.
6. In the **Enter the object names to select** list, enter `ASPNET` and click C**heck Names**. The following is now listed in the **Enter the object names to select** list:

   `<YourComputerName>\ASPNET`
7. If `<YourComputerName>\ASPNET` was found, continue to Step 7. If it was not found, perform the following additional steps:

a. On the **Select Users or Groups** dialog box, click **Advanced**. The **Advanced** View of the **Select Users or Groups** dialog box opens.



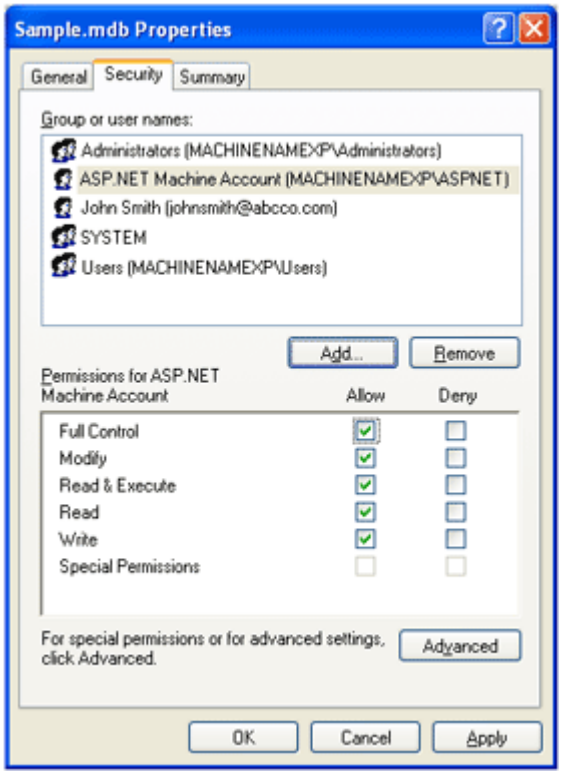
b. Click **Find Now**. All of the users and groups on your machine are listed.



c. In the **Name (RDN)** list, select **ASPNET** and click **OK** to return to the **Select Users or Groups** dialog box.

Caution • *If ASPNET is still not listed, then ASP.NET is not installed on your machine. ASP.NET is automatically installed when you install AdminStudio. Re-install AdminStudio using the original installation CD.*

8. Click **OK** to return to the **Security** tab of the **Properties** dialog box. ASP.NET Machine account is now listed in the **Group or user names** list.
9. In the **Group or user names** list, select **ASP.NET Machine account**, and then select **Full Control** in the **Allow** column in the **Permissions for ASP.NET Machine Account** list:



10. Click **OK** to exit the **Properties** dialog box.

ConflictSolver ACE Reference

The following ACEs are used by Predeployment Test to identify potential conflicts between installation packages.

Table 22-24 • ACEs Used by Predeployment Test

ACE Rule	Conflict Type	Brief Description of ACE
ACE07	**Files**	Checks for the existence of the same file in components with different ComponentIds.
ACE10	**Registry**	Checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds.
ACE23	**Files**	Identifies file duplication between source and target packages. ACE23 checks to see if files with the same name and destination directory have the same size, version, and language when comparing a source package against a target package. If a file with the same name and destination directory is found in both the source and target packages, but the file has a different size, version, or language, ACE23 fails.
ACE24	**Registry**	Checks to see if registry entries with the same registry hive, key, and value name have the same data type and value. If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target package, but the registry entry has a different data type or value, ACE24 fails.

ACE07

ACE07 checks for the existence of the same file in components with different ComponentIds. ACE07 reports four types of errors, depending on the various combinations of source and target files and whether or not they originated in a merge module:

Table 22-25 • Four Types of ACE07 Errors

Source File	Target File	Severity	CARD-Enabled	Conflict View Message	Output Message
Not Merge Module	**Merge Module**	Warning	No	The file [FILENAME] in the component [COMPONENTNAME] is identical to the merge module installed file in the component [COMPONENTNAME]. Confirm this error by running this package against ACE12.	The file [FILENAME] in the component [COMPONENTNAME] in the package [PACKAGENAME] is identical to the merge module installed file in the component [COMPONENTNAME] in the package [PACKAGENAME]. Confirm this error by running this package against ACE12.
Not Merge Module	**Not Merge Module**	Error	Yes	The file [FILENAME] in the component [COMPONENTNAME] is identical to a file in the component [COMPONENTNAME], but the components have different GUIDs.	The file [FILENAME] is identical in both the component [COMPONENTNAME] in the package [PACKAGENAME] and the component [COMPONENTNAME] in the package [PACKAGENAME], but the components have different GUIDs.
Merge Module	**Merge Module**	Error	No	The file [FILENAME] in the merge module installed component [COMPONENTNAME] is identical to a file in another merge module installed component [COMPONENTNAME], but the components have different GUIDs. Run ACE12 to determine which merge module is most appropriate to use.	The file [FILENAME] is identical and installed by merge modules in both the component [COMPONENTNAME] in the package [PACKAGENAME] and the component [COMPONENTNAME] in the package [PACKAGENAME], but the components have different GUIDs. Run ACE12 to determine which merge module is most appropriate to use.

Table 22-25 • Four Types of ACE07 Errors

Source File	Target File	Severity	CARD-Enabled	Conflict View Message	Output Message
Merge Module	**Not Merge Module**	Error	No	The file [FILENAME] in the merge module installed component [COMPONENTNAME] is identical to the file in the component [COMPONENTNAME]. Confirm this error by running the [PACKAGENAME] package against ACE12.	The file [FILENAME] in the merge module installed component [COMPONENTNAME] in the package %s is identical to the file in the component [COMPONENTNAME] in the package [PACKAGENAME]. Confirm this error by running the [PACKAGENAME] package against the ACE12.

Summary

Table 22-26 • ACE07 Summary

Conflict Type:	File
Description:	Checks for the existence of the same file in components with different ComponentIds.
Result:	ACE07 reports four types of errors, depending on the various combinations of source and target files and whether or not they originated in a merge module. (See table above.)
Resolution Type:	Automatic (CARD07) only if both the source and target files originated in a Merge Module. For all other errors, confirm the error by running the package against ACE12, and then, based upon the results, decide how to proceed.
Resolution:	The Source package ComponentId is changed to match the Target package Component Id.

ACE10

ACE10 checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds. The ACE10 Error String is displayed in the following format:

```
The registry entry [REGISTRY_ENTRY] [REGISTRY_KEY] in the component [COMPONENT1] in the package
    [PACKAGE1] conflicts with the same registry entry in the component [COMPONENT1] in the package
    [PACKAGE2].
```

Note • *ACE10 is a warning, not an error.*

Summary

Table 22-27 • ACE10 Summary

Conflict Type:	Registry
Description:	Checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds.
Result:	If the same Root/Key/Name registry combination is in more than one component, a warning is generated.
Resolution Type:	Manual
Resolution:	Change the ComponentId of the Source component to match that of the Target component.

Cause

The same Root/Key/Name registry combination is found in more than one component.

Conditions When an ACE10 Error Can Be Ignored

ACE10 uses data from the MSI Registry table to check for identical registry Root/Key/Name combinations in different components. However, there may be situations in which ACE10 will report an error unnecessarily. MSI supports a grammar for the Registry table in which if the Value field is preceded or terminated by the sequence tilde '[~]', then the registry value will be appended or prepended, respectively, to the existing registry Value. This sort of operation may be perfectly acceptable if the applications in question are modifying a common registry key in a manner consistent with its purpose.

You will need to decide individually if an ACE10 error is valid, but please consider checking the Registry Value field, as its contents may prove useful in helping you decide.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open the MSI package in InstallShield Editor.
2. Once the project is open, navigate to the Components View and find the component that needs to be modified by referring to the component name displayed in the error message.
3. In the Component's Property grid, change the value in the Component Code field to match the Component Code (ComponentId) of the component in the other project.

   If it is unclear what the value should be changed to, follow these steps:

a. Open up the package that will not be modified in Direct Edit mode by selecting Direct Editor under Additional Tools. A list of the package tables is displayed.

b. Select the Component table and search for the Component Name displayed in the conflict.

c. Once the entry is found, take note of the value displayed in the ComponentId column. This is the Component Code that should be used for the component in the other package.

4. Select Save As from the File menu and save the changes as a Windows Installer Transform (.mst) file.

5. Open ConflictSolver and reimport this package with its newly created transform file into your Application Catalog, and then use the Conflict Wizard to check it against ACE10 again.

ACE23

ACE23 identifies file duplication between source and target packages. ACE23 checks to see if files with the same name and destination directory have the same size, version, and language when comparing a source package against a target package. If a file with the same name and destination directory is found in both the source and target packages, but the file has a different size, version, or language, ACE23 fails.

The ACE23 Error String is displayed in the following format:

```
The file [EXECUTABLE_FILENAME, SIZE, VERSION, LANGUAGE] installed to location [PATHNAME] by
    component [COMPONENT1] in Package [PACKAGE1] conflicts with the same file in component
    [COMPONENT2] in Package [PACKAGE2].
```

Note • *ACE23 can be used in conflict analysis when MSI Package, OS Snapshot File, or Marimba NCP File data types are selected as either source or target packages. However, if you specify only Marimba NCP or OS Snapshot files as your source packages, ACE23 and ACE24 will be the only ACEs that you can select on the* ***Conflict Options Panel*** *of the Conflict Wizard. All other ACEs will be unavailable.*

Edition • *Import support for Marimba Native Channel Packager (.ncp) files is included with the AdminStudio Enterprise Edition.*

Summary

Table 22-28 • ACE23 Summary

Conflict Type:	Files
Description:	Checks to see if files with the same name and destination directory have the same size, version, and language.
Result:	If a file with the same name and destination directory is found in both the source and target package (and, in the case of an MSI package comparison, the packages have different ComponentId values), but the file has a different size, version, or language, ACE23 fails. If the Source and Target packages are MSIs and have the same Component Id value, no error is reported.
Resolution Type:	Manual
Resolution:	Investigate and decide which file has precedence. If the source file has precedence over the target file, remove the target file.

Cause

A file with the same name and destination directory is found in both the source and target package, but the file has a different size, version, or language.

Resolution

This conflict requires a manual resolution. Use ONE of the following solutions:

Task: ***Replace the file in the source package with the same file found in the target package***

1. Retrieve a copy of the file from the operating system in which the target package was taken.
2. Open a transform file or MSI package in InstallShield Editor.
3. Once the project is open, navigate to the Files and Folders view, and find the file causing the conflict. Replace that file with the one retrieved from the operating system.
4. Select Save As from the File menu and save the changes as a Windows Installer Transform (.mst) file.
5. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the Conflict Wizard to check it against ACE23 again.

Task: ***Change the destination of the component that installs the file in the MSI package***

1. Open a transform file or MSI package in InstallShield Editor.
2. Once the project is open, navigate to the Components view, and select the component that contains the file causing the conflict.

   In order to quickly find the component name, go to the File Table via the Direct Editor view and search for the file name. Then, check the component column for the component name.
3. Select the Destination field of the component from the Property grid and change it to a new destination.

   Note • *Changing the destination may cause the application to break. Before changing the destination, verify that the application will still work using the new destination.*
4. Select Save As from the File menu and save the changes as a Windows Installer Transform (.mst) file.
5. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the Conflict Wizard to check it against ACE23 again.

ACE24

ACE 24 checks to see if registry entries with the same registry hive, key, and value name have the same data type and value. If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target package, but the registry entry has a different data type or value, ACE24 fails.

The ACE24 Error String is displayed in the following format:

```
The registry entry [REGISTRY_ENTRY] in [PACKAGE1] conflicts with the same registry entry in the
    [COMPONENT] in [PACKAGE2].
```

Note • *ACE24 can be used in conflict analysis when MSI Package, OS Snapshot File, or Marimba NCP File data types are selected as either source or target packages. However, if you specify only Marimba NCP Files as your source packages, ACE23 and ACE24 will be the only ACEs that you can select on the* ***Conflict Options Panel*** *of the Conflict Wizard. All other ACEs will be unavailable.*

Edition • *Import support for Marimba Native Channel Packager (.ncp) files is included with the AdminStudio Enterprise Edition.*

Summary

Table 22-29 • ACE24 Summary

Conflict Type:	Registry
Description:	Checks to see if registry entries with the same registry hive, key, and value name have the same data type and value.
Result:	If a registry entry with the same registry hive, key, and value name in a package is found in both the source and target packages, but the registry entry has a different data type or value, ACE24 fails.
Resolution Type:	Manual
Resolution:	Investigate and decide which registry entry has precedence. If the source registry entry has precedence over the target registry entry, remove the target registry entry.

Cause

A registry entry with the same registry hive, key, and value name in a package is found in both the source and target package, but the registry entry has a different data type or value.

Resolution

This conflict requires a manual resolution. Follow these steps:

Task: ***To resolve this conflict:***

1. Open a transform file or MSI package in InstallShield Editor.
2. Once the project is open, navigate to the Registry view, and find the registry value causing the conflict. Replace that value with the same registry value and data type found in the target package.
3. Select Save As from the File menu and save the changes as a Windows Installer Transform (.mst) file.
4. Open ConflictSolver and reimport this package with its transform file into your Application Catalog, and then use the Conflict Wizard to check it against ACE24 again.

23

License-Enabling Packages Using FLEXwrap

AdminStudio comes with two tools you can use to FLEX-enable packages:

- **FLEXwrap Assistant**—Allows you to place Windows Installer packages (`.msi`) that do not have embedded license management under the control of the FLEXnet software licensing technology for easy control and administration. See Enabling License Tracking of Windows Installer Packages
- **FLEXwrap Server Configuration Tool**—Allows you to monitor the usage of FLEX-enabled applications and manage FLEXwrap license files on the server. See Using the FLEXwrap Server Configuration Tool to Manage the FLEXwrap License Server.

  You can also use the FLEXwrap Server Configuration Tool to FLEX-enable individual existing executable files (.exe). See Using the FLEXwrap Server Configuration Tool to FLEX-Enable Executable (.exe) Files

FLEXwrap user documentation is presented in the following sections:

Table 23-1 • FLEXwrap User Documentation

Section	Description
Enabling License Tracking of Windows Installer Packages	Provides an overview of FLEXnet Software Licensing technology and how to use the FLEXwrap Assistant to inject licensing capabilities into existing Windows Installer packages. The following topics are included: • About FLEXnet Licensing Technology and FLEXwrap • Using the FLEXwrap Assistant • FLEXwrap Assistant Reference

Table 23-1 • FLEXwrap User Documentation (cont.)

Section	Description
Managing FLEXwrap License Files	Explains how to manage FLEXwrap license files using the FLEXwrap Server Configuration Tool. The following topics are included: • Overview of the FLEXwrap License Server • Using the FLEXwrap Server Configuration Tool to Manage the FLEXwrap License Server • Using the FLEXwrap Server Configuration Tool to FLEX-Enable Executable (.exe) Files • FLEXwrap Server Configuration Tool Reference

Enabling License Tracking of Windows Installer Packages

This section provides an overview of FLEXnet Software Licensing technology and how to use the FLEXwrap Assistant to inject licensing capabilities into existing Windows Installer packages. The information is presented in the following sections:

Table 23-2 • FLEXwrap Assistant User Documentation

Section	Description
About FLEXnet Licensing Technology and FLEXwrap	Describes the FLEXnet Software Licensing technology and provides and overview of the two components of FLEXwrap: the FLEXwrap Assistant, and the FLEXwrap Server Configuration Tool.
Using the FLEXwrap Assistant	Provides complete instructions on how to FLEX-enable Windows Installer packages using the FLEXwrap Assistant.
FLEXwrap Assistant Reference	Contains an description of each Page and Dialog Box in the FLEXwrap Assistant.

About FLEXnet Licensing Technology and FLEXwrap

The FLEXnet software licensing management platform is the de facto industry standard for the electronic licensing of software. The software operates on over forty operating systems.

Network license administrators can use FLEXwrap to inject FLEXnet software licensing technology into existing Windows Installer (`.msi`) packages so that they can track usage of the executables in those packages.

Once you have distributed a FLEX-enabled package to the users in your enterprise, you can manage the licensing of that application using FLEXnet Manager.

Benefits of FLEXnet Software Licensing Technology

By using FLEXnet software licensing technology, you can significantly reduce the cost of your software investment by being able to control and monitor the usage of the software licenses on your network.

- **Track usage**—You can track concurrent usage of applications within the enterprise. You can track a list of specific users within the enterprise, and can trace users to a specific individual and machine. You can then use FLEXnet Manager to generate the following reports:
  - Concurrent usage
  - Installations
  - Report based on users and machine names
- **Enforce application license limits**—You can set the number of concurrent users permitted to run an application equal to the number of licenses that you have purchased, enabling you to comply with the terms and conditions of your software license agreements.
- **Enable overdraft licenses during peak usage times to determine accurate license requirements**—You can choose to allow users to run a software package even after its license limit has been reached, and then generate a report on actual usage. You could then purchase any additional licenses that are required, enabling you to only purchase the number of licenses that your organization actually needs.
- **Enable application usage during server down time**—You can choose to allow the wrapped application to run without a license when the FLEXwrap License Server is down or cannot be contacted, providing uninterrupted access to your wrapped applications.
- **Specify handling options for concurrent use**—Instead of requiring a license for each concurrent user (the default setting), you can choose to change this restriction by selecting a different grouping option to determine license requirements. You can specify that any of the following groupings would require only one license:
  - **Host**—All concurrent requests from the same computer
  - **User**—All concurrent requests from the same user
  - **User + Host**—All concurrent requests from the same user on the same computer
  - **Site**—All concurrent requests from the same site
- **Set license expiration date**—You can specify a date when all licenses for an application would expire.
- **Queuing behavior**—When the license limit is reached, you can specify whether the application would launch automatically when a license becomes available, ask the user if he wants to wait for an available license and display a list of current users, or inform the user that he should attempt to run the application at a later time.
- **Flexible instrumentation**—Packages can be FLEX-enabled either before or after repackaging or customization.

Types of Applications that Can Be FLEX-enabled

FLEXwrap can be used with:

- Internally developed applications
- Applications without built-in license enforcement
- Applications with a built-in, non-FLEXnet license manager

***Note** • FLEXwrap does not replace a built-in license manager. If you wrap an application that has a built-in license manager, the application must obtain a license from FLEXwrap in order to try to ask for a license from its built-in license manager.*

However, if an application already contains a licensing mechanism, we do not recommend license-enabling it using FLEXwrap.

About FLEXwrap License Files

A FLEXwrap license file is a specialized license text file containing the applications under FLEXwrap management. This file also contains information about where the License Server runs, all the licensing settings, and the location of the log and options files. The license file (`.lic`) is read by the License Server and, when the **Allow application to run if the license server is unavailable** option is selected on the Configure Connection Page, by the Wrapper.

Evaluation License File

When first using FLEXwrap Assistant, you may want to use a sample evaluation license file provided by AdminStudio. The sample evaluation license file, `eval.lic`, is installed with the product in the following location:

`AdminStudio Installation Directory``\FLEXwrap\Server`

When you choose this file on the Configure Connection Page, the FLEXwrap Server Name Dialog Box opens and you are prompted to enter the machine name of the computer where the FLEXwrap server is installed.

***Note** • The Evaluation License File has a license limit count of **5**, which means that only five licenses are available for an application that is FLEX-enabled using this license.*

***Important** • The evaluation license file (`eval.lic`) is for evaluation purposes only; you are not permitted to use it in a production environment, and it can only be used for a limited amount of time before expiring. When purchasing FLEXwrap with AdminStudio, you do not receive a production license file. To obtain a production license file, you need to purchase FlexNet Manager. For more information, see Production License Files.*

Production License Files

You purchase production license files from AdminStudio Support. To install a production license file, you first purchase license file credentials from AdminStudio Support, and then create the license file by entering the license file credentials on the **Certificate** tab of the FLEXwrap Server Configuration Tool. See Installing Your Production FLEXwrap License File.

Selecting the Executable File to Wrap

On the **Select Files** page of the FLEXwrap Assistant, you are prompted to select a Windows Installer (`.msi`) package, and then to select the executable (`.exe`) files from that Windows Installer package that you want to license inject.

A Windows Installer package often has multiple executable files associated with it, so it may be unclear as to which of those files you should license inject. By default, the **Only show executable files with associated components with shortcuts option** is selected, which filters the list to show only those executables which can be launched separately using a Shortcut.

For example, if you were license-injecting a Windows Installer package that installs a suite of applications such as:

```
wordprocessing.exe, spreadsheet.exe, presentation.exe
```

you would need to license-inject all three of those executable files in order to enforce license management.

But if only one of a Windows Installer package's executable files launches separately (and all the others are called by the main executable to run behind the scenes to perform specific functions), you would only have to license-inject the main, launchable executable file.

You also have the option of clearing the selection of any executable. For example, if an executable is a command line utility that you do not want to enforce licensing on, you can just clear the check box for that executable.

About Output Folders

The **Output folder** on the Finalize Package Page defaults to the folder specified for **Default Output Folder** on the Options Dialog Box.

When the selected package is wrapped, FLEXwrap Assistant creates a subfolder in the default output folder and gives it the same name as the Windows Installer file. Then FLEXwrap Assistant copies the Windows Installer file and all of its associated files and subfolders to this new subfolder before starting the wrapping process.

For example, if you were license enabling a Windows Installer package named `Graph2005.msi`, and if the **Output folder** on the Finalize Package Page was set to:

```
C:\WrappedPackages
```

the selected Windows Installer file and all of its associated files and subfolders would be copied to:

```
C:\WrappedPackages\Graph2005
```

How FLEXwrap Assistant Works

The FLEXwrap Assistant, a tool in the AdminStudio Tools Gallery, is used to FLEX-enable Windows Installer packages. After a Windows Installer package is wrapped to include the licensing technology, the output is still a Windows Installer package which has full functionality and manageability, with all Windows Installer capabilities intact.

Actions You Take

When you use the FLEXwrap Assistant to inject licensing capabilities into a Windows Installer package, you perform the following steps:

Table 23-3 • Wrapping a Windows Installer Package

Icon	Step	Description
	Step 1: Select Files	Identify the executables (`.exe` files) in a Windows Installer package (`.msi` file) that you want to wrap, and specify whether to encrypt the original executables to prevent end users from bypassing licensing limits.
	Step 2: Set Limits	Specify the number of concurrent licenses available for this application, the number of overdraft licenses available for peak usage, and the action the application will take if no licenses are currently available.
	Step 3: Define Access	Defining which requests or group of requests require a separate license. You can also restrict application usage by specific computers.
	Step 4: Configure Connection	Configure a connection to the FLEXwrap License Server.
	Step 5: Finalize Package	Specify an output folder and initiate the creation a new FLEX-enabled Windows Installer package.

Actions Taken by the FLEXwrap Assistant

The following diagram details the main actions performed by the FLEXwrap Assistant when you wrap a Windows Installer package:

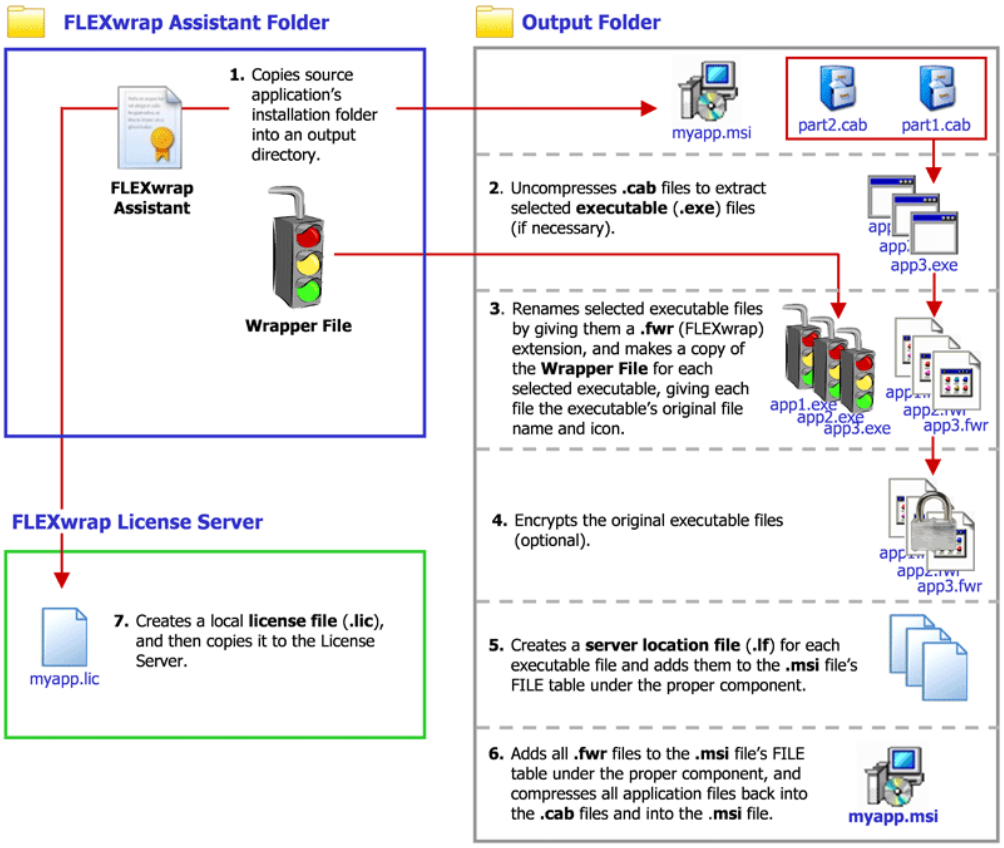


Figure 23-1: FLEXwrap Assistant Overview

When you click **Build Package** on the **Finalize Package Panel**, the FLEXwrap Assistant takes the following actions:

1. **Copies package to output folder**—Copies the source Windows Installer folder to a subfolder of the output folder you specified.
2. **Extracts executables**—Uncompresses this package's .cab files (if any) to extract the executables (`.exe` files) that you selected.
3. **Renames executables and replaces them with Wrapper**—Renames selected `.exe` files to `.fwr`, and makes a copy of the copy of the Wrapper file for each selected `.exe` file, giving each file the executable's original file name and icon.
4. **Encrypts executables**—Encrypts the original executables, if requested.
5. **Specifies location of License Server**—For each executable, the FLEXwrap Assistant creates a location file (`.lf`) containing the location of the License Server, and adds the `.lf` files to the `.msi` File table under the proper component.

6. **Recompresses Windows Installer package**—Adds all `.fwr` files to the package's File table under the proper component, and compresses all application files back into the `.cab` files and into the `.msi`.
7. **Creates license file and copies it to the License Server**—Creates a local license file (`myapp.lic`)for this application, and then copies it to the License Server.

Launch Process for a Wrapped Application

When a user attempts to launch a wrapped application, the following occurs:

- **Wrapper launches**—When a user chooses to open an application, the Wrapper launches (which is transparent to the user).
- **Wrapper obtains location of License Server**—The Wrapper obtains the location of the License Server from the server location file (`.lf`).
- **Wrapper contacts License Server to obtain a license**—The Wrapper attempts to obtain a license for the wrapped application from the License Server.
- **Wrapper launches the wrapped application**—One of the following occurs:

If this is true ...	BUT this is also true,	then this happens
License is obtained		Wrapper runs the wrapped application's binary.
No license is obtained	**Enable overdraft licenses** option on the Set Limits Page is selected	Wrapper runs the wrapped application's binary.
License Server cannot be contacted	**Allow application to run if the License Server is unavailable** option on the Configure Connection Page page is selected	Wrapper runs the wrapped application's binary.

If this is true ...	BUT this is also true,	then this happens
No license is obtained	Neither of the above options was selected.	Wrapper follows the option that was selected under **Action to take when license limit is reached** on the Set Limits Page page: • **Ask User**—A list of current license users opens, and the user is prompted to select whether he wants to wait for a license to become available. • **Always Queue**—The user is added to a waiting list and then the application is automatically launched when a license becomes available. • **Never Queue**—Application is not launched and the user is instructed to attempt to launch the application later.

License checkouts are nearly instantaneous, so users should not notice any delay in starting wrapped applications.

- **Usage information written to report log**—If a license is obtained and if report logging is enabled, license usage information is written to the FLEXwrap report log. You could then use FLEXnet Manager reporting to generate usage reports.

The following diagram illustrates the activities that occur when a wrapped application is launched:

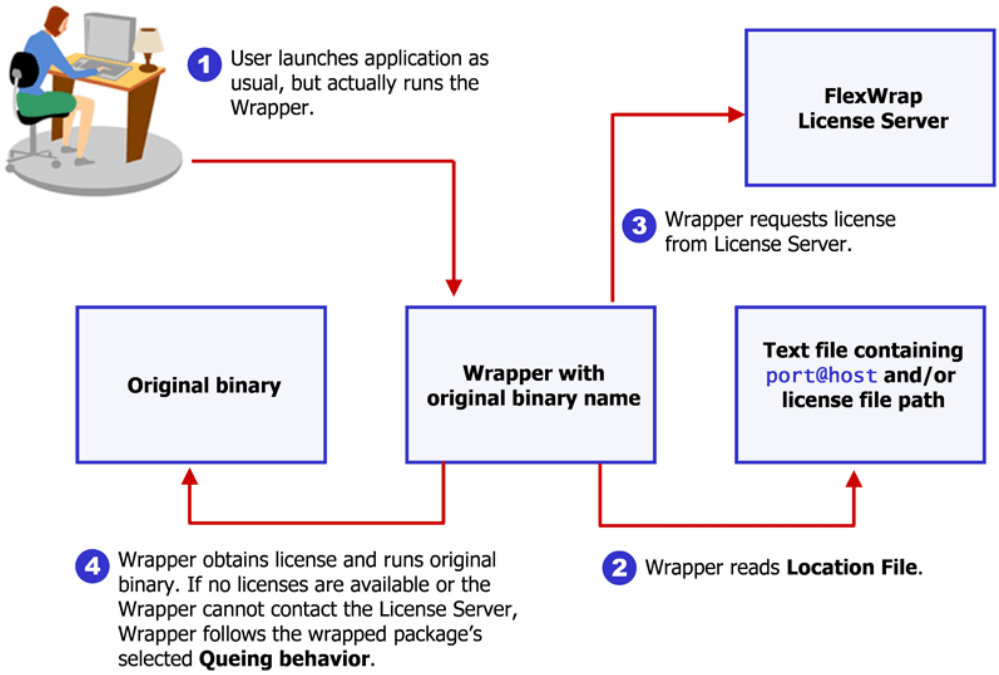


Figure 23-2: Diagram of activities that occur when a wrapped application is launched

Note • *This diagram assumes that no overdraft licenses were provided and the application is not permitted to run if the License Server cannot be connected*

Using the FLEXwrap Assistant

The procedures listed in this section explain how to use the FLEXwrap Assistant to license-enable Windows Installer packages, and describes the various licensing options.

- Launching the FLEXwrap Assistant
- License-Enabling a Windows Installer Package
- Setting Licensing Options

Launching the FLEXwrap Assistant

The FLEXwrap Assistant can be launched from AdminStudio from the following locations:

- Click the **FLEXwrap Assistant** icon in the Tools Gallery on the AdminStudio 10.0 SP2 Start Page.
- Click the **FLEXwrap Assistant** icon in the Tools Gallery on the AdminStudio Tools Tab.

- Click the **Prepare Application for Usage Management** link in the Application Tasks list on the AdminStudio Start Page.
- Click the **Prepare Application for Usage Management** link in the More Options list on the Distribute Page of the Process Assistant.
- On the Windows **Start Menu**, point to **All Programs**, **AdminStudio**, **AdminStudio 10.0 SP2 Tools**, **FLEXwrap**, and click **FLEXwrap Assistant**.

License-Enabling a Windows Installer Package

To license-enable a Windows Installer package, perform the following tasks:

- Selecting a Windows Installer Package to License-Enable
- Setting License Limits and Queuing Behavior
- Defining License Requirements and Access Restrictions
- Configuring a Connection to the FLEXwrap License Server
- Specify the Output Folder and Build Package

Selecting a Windows Installer Package to License-Enable

The first task is to select the Windows Installer package that you want to FLEX-enable, and specify whether to encrypt the original executables to prevent end users from bypassing licensing limits.

Task: ***To license-enable a Windows Installer package:***

1. Launch the FLEXwrap Assistant. The **FLEXwrap Home Page** opens.
2. Click the **Select Files** icon in the navigation bar. The **Select Files Page** opens.
3. Click the **Browse** button and select the Windows Installer (`.msi`) package file that you want to license-enable.

   The names of all of the executables contained the selected Windows Installer package are now listed in the **Select Executable File(s) to License-Inject** list, and by default all are selected.
4. Often a Windows Installer package contains multiple executable (`.exe`) files, but usually only a subset of these executables have a Windows shortcut associated with it. If the **Only show executable files with associated components with shortcuts** option is selected, only those executables that the Windows Installer package creates a shortcut for are listed. To view a list of all executables regardless of Shortcut status, clear this option.
5. Clear the selection of any executables that you do not wish to license-enable. You can use the **Select All** or **Clear All** buttons to quickly select or clear selections.
6. If more than one executable is selected, the **Treat selected executable files as a suite** option is enabled. If you want to license-enable these executables as a suite, meaning that only one license is required to launch multiple executables from this package, select this option and enter a **Suite name**.

For more information, see Benefits of Using the Suite Option.

7. If you want to prevent users from running non-license-injected versions of the selected file(s) by renaming and using the original executable, select the **Encrypt original executable file(s)** option.

   See Benefits of Encrypting the Source Executables for more information.

8. Click the **Set Limits** icon in the navigation bar. The **Set Limits Page** opens.

9. Proceed with Setting License Limits and Queuing Behavior.

Setting License Limits and Queuing Behavior

In this task you specify the number of concurrent licenses available for this application, the number of overdraft licenses available for peak usage, and the action the application will take if no licenses are currently available.

Task: ***To set license limits and overdraft behavior:***

1. After you have performed the steps described in Selecting a Windows Installer Package to License-Enable and the **Set Limits Page** is open, enter the number of concurrent licenses available for this application in the **License limit** box. For help in determining your license limit, see Setting Your License Limit.

2. If you want to enable users to continue to use this application even when the license limit has been reached, select the **Enable overdraft licenses** option and enter a value in the **Overdraft limit** box.

   See Using Overdraft Support to Determine Application Usage for a description of your overdraft support options.

3. If you want this application's licenses to expire on a specific date, select the **Set an expiration date for the licenses** option and select a date from the **Licenses expires on** list. If you do not want these licenses to expire, leave this option unselected.

   See Setting a License Expiration Date for more information.

4. Under **Queueing behavior**, select one of the following options:

   - **Ask user**—If all licenses are in use and another user attempts to launch the application, a list of the current license users opens, and the user is prompted to select whether he wants to wait for a license to become available (when one of the current license users closes the application). If he selects this option, the application will launch when a license becomes available.

     By displaying a list of current license users, the user who is waiting has the option of contacting one of the license users to see if they will close the application to make a license available.

   - **Always queue**—If all licenses are in use and another user attempts to launch the application, always add the user to a waiting list and then launch the application when a license becomes available.

   - **Never queue**—If all licenses are in use and another user attempts to launch the application, inform user to attempt to launch the application later.

   For more information, see Choosing the Right Queuing Option for Your Application.

5. Click the **Define Access** icon in the navigation bar. The **Define Access Page** opens. Proceed with Defining License Requirements and Access Restrictions.

Defining License Requirements and Access Restrictions

In this task you define which requests or group of requests require a separate license. You can also restrict application usage by specific hostids (computer names).

Task: ***To define license requirements and set access restrictions:***

1. After you have performed the steps described in Setting License Limits and Queuing Behavior and the **Define Access Page** is open, select one of the following license requirement grouping options:
2. You can change this restriction by selecting one of the other four grouping options:
   - **Process**—Each concurrent instance of a license-enabled application requires a separate license. (Default)
   - **Host**—All concurrent requests from the same computer use only one license, regardless of whether multiple users are accessing the application from the same computer.
   - **User**—All concurrent requests from the same user use only one license, regardless of whether the same user is accessing the application from multiple computers.
   - **User + Host**—All concurrent requests from the same user on the same computer use only one license. This is the default selection if grouping concurrent license requests.
   - **Site**—All concurrent requests use the same license.

   For more information, see Grouping Options for Concurrent Use.
3. A **hostid** is the address of a computer in a TCP/IP network that uniquely identifies it. To restrict application usage by hostid, select the **Restrict access by hostids listed below** option, and perform the following steps to add hostids to the **Permitted hostids** list:
   a. Click **Add**. The **Permitted Hostid** dialog box opens.
   b. In the **Hostid Type** list, select the type of hostid that you want to restrict usage by: **Ethernet address**, **Internet IP address**, **Machine name**, **User name**, **Windows disk serial number**, or **Other**.
   c. In the **Hostid Value** text box, enter a value for the selected **Hostid Type**.

      To determine a machine's hostid, launch **FLEXwrap LMTOOLS** on that machine, and open the **System Settings** tab:

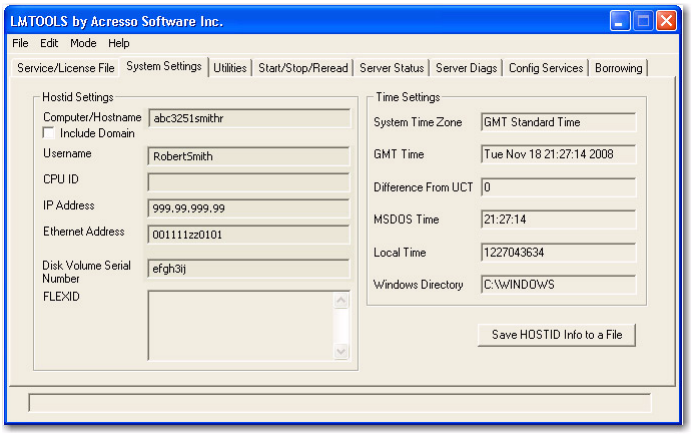


Figure 23-3: Hostid information on the Systems Settings tab of FLEXwrap LMTOOLS

d. Click **OK**. The **Permitted Hostid** dialog box closes and the hostid that you just defined in now listed in the **Permitted hostids** list.

e. If you want to edit an existing hostid, select an item in the **Permitted hostids** list and click **Edit**

For more information, see About Hostids.

4. Click the Configure Connection icon in the navigation bar. The Configure Connection Page opens. Proceed with Configuring a Connection to the FLEXwrap License Server.

Configuring a Connection to the FLEXwrap License Server

In this task you configure a connection to the FLEXwrap License Server.

Task: ***To define license requirements and set access restrictions:***

1. After you have performed the steps described in Defining License Requirements and Access Restrictions and the **Configure Connection Page** is open, do one of the following:

   If you already have a license file, perform the following steps:

   a. Click **Browse** and select a **FLEXwrap server license (.lic) file**. This file is not modified and is only used to get your FLEXwrap License Server settings.

   b. After you select a license file, the **Host name** and **Port** fields are automatically populated with the information in the license file.

   If you do not have a license file, perform the following steps to obtain and open an evaluation license file:

   a. Click the **Request a FLEXwrap server license** link to open a request form on the AdminStudio Web site. Fill out the requested information and submit the form. You will then receive an evaluation license file named `eval.lic` via email.

   b. Copy the `eval.lic` file to the following location:

AdminStudio Installation Directory\FLEXwrap\Server

c. On the **Configure Connection Page**, click **Browse** and select the file that you just obtained: `eval.lic`. The FLEXwrap Server Name dialog box opens, prompting you to enter the machine name of the computer where the FLEXwrap server is installed.

d. Enter a machine name and click OK.

Note • *The Evaluation License File has a license limit count of **5**, which means that only five licenses are available for an application that is FLEX-enabled using this license.*

2. If you want to be able to specify the server location when this application is installed, select the **Allow host name and port redefinition at install time** option.

   If this option is selected, FLEXwrap injects a custom action into the `.msi` file that enables you to set the server name and port from the command line. For example:

   ```
   msiexec /i "<path_to_msi_file>\wrapped.msi" SWRAP_LICENSE_FILE=9987@testserver
   ```

   You would choose this option if you wanted the flexibility to reassign the FLEXwrap License Server host name/port at install time for any reason, such as if you wanted to install the same wrapped application to multiple servers across different locations at the same organization.

3. To allow this wrapped application to be able to launch even when the FLEXwrap License Server is down or is having communication problems, select the **Allow application to run if the License Server is unavailable** option. This option stores the license file on the local machine. By default, this is unselected and is not recommended because it may enable users to run unlicensed versions of an application, circumventing the protection that FLEXwrap provides.

4. If you want to allow an application to run even when the License Server becomes unavailable, select the Specify timeout value option. Then, in the **Timeout value** box, enter the number of minutes that the connection has to be lost before the application is closed.

   - By default, this option is not selected and therefore there is no timeout period. The application will continue to run without a license and will continue to attempt to regain its license indefinitely.
   - If this option is selected and a **Timeout value** is entered, the application warns the user and then exits if the license cannot be regained within the timeout period.

5. Click the **Finalize Package** icon in the navigation bar. The **Finalize Package Page** opens. Proceed with Specify the Output Folder and Build Package.

Specify the Output Folder and Build Package

In this task, you specify an output folder and initiate the creation a new FLEX-enabled Windows Installer package.

Task: ***To specify the Output folder and enable licensing:***

1. After you have performed the steps described in Configuring a Connection to the FLEXwrap License Server and the **Finalize Package Page** is open, you can click **Browse** and select the **Output folder** where you want the wrapped application to be stored or you can just accept the default location.
2. If you want to automatically update the package code of this wrapped package to differentiate it from the original package, select the **Update package code** option. By default, this option is selected. See When to Update the Package Code for more information.
3. If you want to add a comment to this package to indicate that it is a wrapped package or to provide information on next steps, select the **Modify package comments** option and enter comments in the box. These comments will be visible from Windows Explorer if you right click on this `.msi` file and select **Properties** from the context menu. See Purpose of Packaging Comments for more information.
4. Click **Build Package** to create a new Windows Installer package with built-in licensing and store it in the directory that you specified.

Next Steps

After you have license-enabled a package, the next step is to restart your FLEXwrap License Server or issue a reread command. For instructions on restarting your License Server, see Starting the License Server.

You also may want to use FLEXnet Manager and set up usage reporting so that you can track concurrent usage of this application within the enterprise. You can track a list of specific users within the enterprise, and can trace users to a specific individual and machine. For more information on FLEXnet Manager's reporting features, see FLEXnet Manager. After you have FLEX-enabled a package, you cannot open up the wrapped package in the FLEXwrap Assistant to edit any of the settings. Instead, select the original Windows Installer file and follow the License-Enabling a Windows Installer Package procedure to create another wrapped version of the package.

Setting Licensing Options

When using the FLEXwrap Assistant to license-enable a Windows Installer package, you can use the following licensing options:

- Benefits of Using the Suite Option
- Benefits of Encrypting the Source Executables
- Setting Your License Limit
- Using Overdraft Support to Determine Application Usage
- Setting a License Expiration Date
- Choosing the Right Queuing Option for Your Application

- Grouping Options for Concurrent Use
- Setting Licensing Behavior During Server Down Time
- Setting Timeout Periods

Benefits of Using the Suite Option

If a Windows Installer package contains multiple executables, you can specify that you want all of the executables to belong to the same suite and therefore require only one license.

To do this, select multiple executable files on the Select Files Page, select the **Treat selected executable files as a suite** option, and specify a **Suite name**.

For example, if a Windows Installer package named `GraphicsUtilties.msi` includes a drawing application (`draw.exe`), a screen capture application (`capture.exe`), and a page layout application (`layout.exe`), you can choose to treat these applications as a suite so that the user would only have to purchase one license to activate all applications in the suite.

Benefits of Encrypting the Source Executables

When the FLEXwrap Assistant wraps a Windows Installer package, it renames the selected `.exe` files to `.fwr`, and makes a copy of the Wrapper file for each selected `.exe` file, giving each file the executable's original file name and icon. All of these files are compressed back into the Windows Installer `.msi` file.

Because the original, unwrapped executables are installed along with the wrapped executables, it is possible for a user to rename the executable files with the `.fwr` extension back to having an `.exe` extension, and to launch the original, unwrapped executables and bypass the licensing limits you have set.

Therefore, if you want to prevent users from running non-license-injected versions of the wrapped executables, select the **Encrypt original executable file(s)** option on the Select Files Page. This would make it impossible for the original executable files to run.

Caution • *This encryption feature of FLEXwrap is not guaranteed because some applications can be configured to discourage any alteration, and encryption of the application binary may not be allowed.*

Setting Your License Limit

On the Set Limits Page, you should set the **License limit** (the number of concurrent users permitted to run an application) equal to the number of licenses that you have purchased, enabling you to comply with the terms and conditions of your software license agreements.

However, the FLEXwrap Assistant provides several other options you can use to manage access to an application while still complying with your license agreements:

- **Overdraft support**—By selecting the **Enable overdraft licenses** option on the Set Limits Page, you can choose to allow a specified number of users to run a software package during peak usage times even after its license limit has been reached, and then generate a report on actual usage. You could then purchase any

additional licenses that are required, enabling you to only purchase the number of licenses that your organization actually needs.

- **Grouping options**—By default, each application instance requires its own license. You can change this restriction by selecting a different grouping option so that license requests are grouped by computer, by user, by user and computer, or by site. For more information, see Grouping Options for Concurrent Use.

Using Overdraft Support to Determine Application Usage

By selecting the **Enable overdraft licenses** option on the Set Limits Page, you can choose to allow a specified number of users to run the software package during peak usage times even after the application's license limit has been reached.

You can then use FLEXnet Manager to generate a report on actual usage, enabling you to accurately determine your organization's license requirements, and purchase any additional licenses that are required. Implementing Overdraft support gives you the information you need to purchase only the number of licenses that your organization actually needs.

Setting a License Expiration Date

On the Set Limits Page of the FLEXwrap Assistant, you can choose to enter an expiration date for this application's licenses. You would want to set an expiration date if you were aware that your license agreement for this application expires on a specific date, and you want to make sure that no one in your organization can launch the application after that date. This enables you to comply with the terms and conditions of your software license agreement.

If you do not have plans to let this application's license agreement expire, do not select this option.

Choosing the Right Queuing Option for Your Application

On the Set Limits Page under **Action to take when license limit is reached**, you can select one of the following options for the selected package:

- **Ask user**—If all licenses are in use and another user attempts to launch the application, a list of the current license users opens, and the user is prompted to select whether he wants to wait for a license to become available (when one of the current license users closes the application). If he selects this option, the application will launch when a license becomes available.

  By displaying a list of current license users, the user who is waiting has the option of contacting one of the license users to see if they will close the application to make a license available. You might want to choose this option if most of the users who would be using this package know each other and contact each other on a frequent basis.

- **Always queue**—If all licenses are in use and another user attempts to launch the application, always add the user to a waiting list and then launch the application when a license becomes available. You might want to choose this option if most of the users who would be using this package do not know each other and, therefore, probably would not contact any of the current users on the list.

- **Never queue**—If all licenses are in use and another user attempts to launch the application, inform the user to attempt to launch the application later. You might want to use this option when most of the people using this application usually use it for long periods of time (and therefore would not be likely to close it any time soon).

Grouping Options for Concurrent Use

By default, license requests are grouped by **Process**: each concurrent request to launch the application uses one license. Instead of requiring a license for each concurrent user, you can choose to change this restriction by selecting a different grouping option to determine license requirements. You can specify that any of the following groupings would require only one license:

- **Host**—All concurrent requests from the same computer.
- **User**—All concurrent requests from the same user.
- **User & Host**—All concurrent requests from the same user on the same computer.
- **Site**—All concurrent requests from the same site.

You should select the grouping option that concurs with your licensing agreement for this software package.

Setting Licensing Behavior During Server Down Time

By selecting the **Allow application to run if the license server is unavailable** option on the Configure Connection Page, you can choose to allow the wrapped application to run without a license when the FLEXwrap License Server is down or cannot be contacted, providing uninterrupted access to this application. When this option is chosen, the license file is stored on the local machine.

You would not want to select this option if you were concerned that having it selected might enable some users to use this application without a valid license.

Setting Timeout Periods

You would select the **Specify timeout value** option on the Configure Connection Page to specify the number of minutes that the connection to the License Server has to be lost before the application is closed.

By default, the **Specify timeout value** option is not selected and therefore there is no timeout period. The application will continue to run without a license and will continue to attempt to regain its license indefinitely.

If the **Specify timeout value** option is selected and a **Timeout value** is entered, the application warns the user and then exits if the connection to the License Server cannot be regained within the timeout period. You would want to specify a timeout value if you were concerned that if no timeout value was specified, some users would be able to use this application without a valid license and violate your licensing agreement.

Setting Configuration Options

The following topics describe FLEXwrap Assistant configuration and communication concepts:

- About Hostids
- Changing the Host Name or Port at Install Time
- When to Update the Package Code
- FLEX-enabling a Signed Package
- Purpose of Packaging Comments

About Hostids

You can choose to restrict application access by hostid so that only permitted hostids would be able to use the application. A **hostid** is the address of a computer in a TCP/IP network that uniquely identifies it. The hostid could be the Ethernet address, IP address, machine name, user name, Windows disk serial number, or any other unique identifier for that machine.

To lock a license file to a specific group of machines, choose the **Restrict access by hostids listed below** option on the **Define Access Page** of the FLEXwrap Assistant, and add **Permitted hostids** to the list.

Changing the Host Name or Port at Install Time

If you want to be able to specify the server location when this application is installed, select the **Allow host name and port redefinition at install time** option on the Configure Connection Page.

If this option is selected, FLEXwrap injects a custom action into the `.msi` file that enables you to set the server name and port from the command line. For example:

```
msiexec /i "<path_to_msi_file>\wrapped.msi" SWRAP_LICENSE_FILE=9987@testserver
```

You would choose this option if you wanted the flexibility to reassign the License Server host name/port at install time for any reason, such as if you wanted to install the same wrapped application to multiple servers across different locations at the same organization.

When to Update the Package Code

If you want to automatically update the package code of this wrapped package to differentiate it from the original package, select the **Update package code** option. By default, this option is selected.

You would want to select this option perhaps if you wanted to import both the wrapped and the original package into the AdminStudio Application Catalog, which requires the each package to have a different package code. If you only wanted to import the wrapped package into the Application Catalog, you would not need to select this option.

If the wrapped package used the same package code as the original package, it would cause issues with patches and anything else that would treat both packages the same. If patches intended to update the original application's binary were applied to the wrapped package, the patch would fail. By creating a new package code, this problem is avoided.

FLEX-enabling a Signed Package

Packages can include a digital signature, and those packages are referred to as signed packages. A signed package is identical to an unsigned package except for the digital signature.

If a package has a valid digital signature, this ensures that the package has not been modified since the signature was applied to the package. Using signed packages provides a secure method of downloading or installing packages because the digital signature can be verified before the package is added to a system.

If you choose to FLEX-enable a signed package, the following warning message appears and the digital signature is invalidated:

```
Warning: The Windows Installer package you selected is digitally signed.
```

If you choose to continue and use the FLEXwrap Assistant to modify this package, you lose this package's digital signature and the application can no longer be verified before download or installation. However, all of the package's functionality remains intact and it installs properly.

Purpose of Packaging Comments

When finalizing your wrapped application, you may want to attach a note stating that this is a license-injected package, or provide "next steps" instructions such as how to restart the process with another package, how to restart the License Server, or how to set up usage reporting. The exact text should be decided upon by the development and documentation team.

To associate comments with a package, select the **Modify package comments** option on the Finalize Package Page, and enter comments in the box. These comments are stored in the System Information Stream of the package, and will be visible from Windows Explorer if you right click on this `.msi` file and select **Properties** from the context menu.

FLEXwrap Assistant Reference

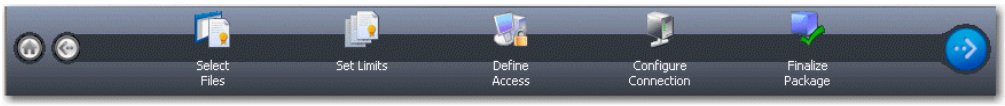
This section describes each of the Pages and dialog boxes that you encounter when using the FLEXwrap Assistant.

- Home Page
- Select Files Page
- Set Limits Page
- Define Access Page
- Configure Connection Page
- Finalize Package Page

- Permitted Hostid Dialog Box
- Options Dialog Box
- Output Window

Home Page

The FLEXwrap Assistant Home Page includes a graphic that helps you visualize the injection process, and includes a Navigation Bar with links to the tasks performed using this Assistant:



The license-injection process is presented on the following five pages:

Table 23-4 • FLEXwrap Assistant Pages

FLEXwrap Assistant Page	Description
Select Files Page	Select a Windows Installer package and the applications within it that you want to license-enable.
Set Limits Page	Specify the number of concurrent licenses available for the license-injected application(s), set the license limits, set the overdraft license limits, and set the queuing behavior.
Define Access Page	Define the handling options for concurrent use of the license-enabled application(s). By default, each application instance (process) requires its own license. You can change this restriction by selecting a different grouping option. You can also restrict application usage by hostid.
Configure Connection Page	Specify a path to an existing server license file, specify how the license-enabled application will find the License Server, how strictly licenses should be enforced , and how to handle loss of connection to the License Server.
Finalize Package Page	Complete the license-injection process by optionally modifying settings and then clicking the **Build Package** button. All the files in the source Windows Installer package folder will be copied under a subfolder of the output folder you select below, then the license-injection process will start.

To return to this Home Page at any time, click the Home button on the left side of the Navigation Bar:

Select Files Page

On the Select Files page, select a Windows Installer package and the applications within it that you want to license-enable.

Table 23-5 • Select Files Page Options

Option	Description
Select Windows Installer Package	Select a Windows Installer package containing the applications that you want to license-enable. After you select an .msi file, all executables in that package are listed in the Select Executable File(s) to License Inject list.
Select Executable File(s) to License Inject	By default, this list only contains executables that also have associated shortcuts, and all of these executables are selected. This option can be unselected and all executables in the package will display in this list. If multiple executables are selected, the option to treat selected executables as a suite becomes available (by default, all executables with shortcuts are displayed and selected, so this option is by default enabled). If the administrator selects this option, they need to specify a name for the suite they are creating. If you select multiple executables and do not treat them as a suite, then each executable creates its own license file with identical options to other selected executables.
Only show executable files with associated components with shortcuts	By default, only the applications with a shortcut are listed in the Select Executable File(s) to License Inject list. To list all of the executables in the selected Windows Installer package, select this option.
Treat selected executable files as a suite	When a suite is created, all applications in the suite operate under a single license. In this case, a license represents the ability to run each of those executables concurrently. For example, if you had a suite consisting of Word.exe, Excel.exe, and PowerPoint.exe, you would be able to run one instance each of them simultaneously and still only use a single license for the suite. If you select multiple executables and do not treat them as a suite, then each executable creates its own license file with identical options to other selected executables. If you select this option, enter a Suite name to identify the suite.
Encrypt original executable file(s)	Select this option to encrypt the original executable file(s) to prevent users from circumventing license restrictions by running the non-license-injected versions of the executables. ***Caution*** • *This encryption feature of FLEXwrap is not guaranteed because some applications can be configured to discourage any alteration, and encryption of the application binary may not be allowed.*

Set Limits Page

On the Set Limits page, you specify the number of concurrent licenses available for the license-enabled application, set the license limits, set the overdraft license limits, and set the queuing behavior.

Table 23-6 • Set Limits Page

Option	Description
License limit	Specify the number of concurrent licenses available for the license-enabled application. By default, the limit is set to 1.
Enable overdraft licenses	Select this option if you want to be able to set a number of extra licenses above and beyond the license limit (**Overdraft limit**). This allows you to cover peak usage times until new licenses can be purchased. By default, this option is not selected.
Overdraft limit	You can optionally specify the number of overdraft licenses available for peak usage.
Set an expiration date for the licenses	Select this option to make the licenses for this package expire on a certain date. By default, this option is not selected.
Licenses expire on	Specify the date on which the licenses will expire. The expiration date defaults to one year from the current date.
Action to take when license limit is reached	Select one of the following options to specify action to take when the license limit has been reached (all licenses are currently in use): • **Ask user**—Display a list of current users, and prompt the user attempting to launch the application to choose to wait for a license to become available. The user has the option of contacting one of the other users determine if they plan to close the application soon. If the user queues for a license, then the application will run when a license becomes available. This is the default behavior. • **Always queue**—Always queue license requests if a license is currently unavailable, and launch the application when a license is available. • **Never queue**—If a license is not available, inform the user to run the application later.

Define Access Page

On the Define Access page, you can change how license requests are grouped, and you can choose to restrict application usage by computer.

Table 23-7 • Define Access Page Options

Option	Description
Group license requests by	By default, each concurrent request of an application requires one license. However, if you select this option, you can change how license requests are grouped • **Process**—Each request uses one license. (Default) • **Host**—All concurrent requests from the same computer use only one license, regardless of whether multiple users are accessing the application from the same computer. • **User**—All concurrent requests from the same user use only one license, regardless of whether the same user is accessing the application from multiple computers. • **User + Host**—All concurrent requests from the same user on the same computer use only one license. This is the default selection if grouping concurrent license requests. • **Site**—All concurrent requests from the same site use only one license.
Restrict access by hostids listed below	Select this option to allow access to this license-enabled application only if the host name of the computer is in the **Permitted hostids** list.
Permitted hostids	List of computers that are permitted access to this license-enabled application. Computer names are listed in the standard Windows format that is displayed on the **Computer Name** tab of the Microsoft Windows **System Properties** dialog box. If there are computers in this list and the **Restrict access by hostids listed below** option is cleared, the list becomes non-editable but is persisted. ***Note*** • *To determine a machine's hostid, launch **FLEXwrap LMTOOLS** on that machine, and view the information on the **System Settings** tab.*
Add	Click to open the Permitted Hostid Dialog Box, where you can enter a computer Hostid Value to add to the **Permitted hostids** list.
Edit	Select a **Permitted hostid** and click **Edit** to open the **Permitted Hostid** dialog box where you can edit that computer's **Hostid Type** and **Hostid Value.**
Delete	Select a Permitted hostid and click **Delete** to delete it from the **Permitted hostids** list.

Configure Connection Page

On the Configure Connection page, you specify a path to an existing server license file, enable the application to run if the License Server is unavailable, enable a user to enter the host name and port redefinition at install time, and specify how long the License Server needs to be disconnected before the user is notified.

Table 23-8 • Configure Connection Page Options

Option	Description
FLEXwrap server license (.lic) file	Browse to the license file location. After a license file has been selected, the **Host name** and **Port** for the License Server will be displayed. The selected license file will not be modified and a newly generated one will be copied to the same location.
Host name (read only)	The host name of the FLEXwrap License Server listed in the selected license file.
Port (read only)	The port of the FLEXwrap License Server listed in the selected license file.
Allow application to run if the License Server is unavailable	This option stores the license file on the local machine. By default, this is unselected.
Allow host name and port redefinition at install time	In order to accommodate the use case of having multiple servers across different location and needing to specify a different License Server name based on location or any other factor using the same package Select to allow customization of the server location when this application is installed.
Specify timeout value; If license is lost, user is warned and application is closed after this number of minutes	Select this option if you want to specify the number of minutes that an application will continue to run if the License Server becomes unavailable. By default, there is no timeout period, so the application will continue to run without a license and will continue to attempt to regain its license indefinitely. If this option is selected and a **Timeout value** is entered, the application warns the user and then exits if the license cannot be regained within the timeout period.
Timeout value (minutes)	This option is enabled if the **Specify timeout value** option is selected. For situations where the License Server becomes unavailable, enter the time in minutes that you want the application to attempt to re-establish a connection to the License Server before the application exits. The minimum amount if 5 minutes.

Table 23-8 • Configure Connection Page Options (cont.)

Option	Description
Request a FLEXwrap server license	Click to open an entry form on the AdminStudio Web site where you request a FLEXwrap evaluation license (eval.lic). The license will be sent to you via email. ***Note*** • *The Evaluation License File has a license limit count of **5**, which means that only five licenses are available for an application that is FLEX-enabled using this license.*

Finalize Package Page

On the Finalize Package page, you specify the location where the wrapped Windows Installer and its source files will be saved and click **Build Package** to create the wrapped application. You can also choose to update the package code or modify package comments.

Table 23-9 • Finalize Package Page

Option	Description
Output folder	Specify the folder where the wrapped Windows Installer application will be saved. All the files in the source Windows Installer package folder will be copied into a subfolder of this output folder. By default, the files will be copied to the path specified on the FLEXwrap Assistant Options dialog box, which is opened by selecting Options on the Tools menu.
Update package code	Select this option to automatically update the package code of this wrapped package to differentiate it from the original package. By default, this option is selected. If the wrapped package used the same package code as the original package, it would cause issues with patches and anything else that would treat both packages the same. If patches intended to update the original application's binary were applied to the wrapped package, the patch would fail. By creating a new package code, this problem is avoided.
Modify package comments	Select this option to note that this is a license-injected package or to enter any other comments that you feel would be helpful. The comments are stored in the MSI Package's System Information Stream and are visible from Windows Explorer if you right click on this .msi file and select **Properties** from the context menu.
Build Package	Click to create a new Windows Installer package with built-in licensing.

Note • *Make sure you restart your licensing server or issue a reread command when any license file is modified or added.*

Permitted Hostid Dialog Box

On the Permitted Hostid dialog box, you can restrict application usage by hostid, a value that uniquely identifies a machine on a network.

If you enter hostids on this dialog, only users on the specified hostids will be able to use this application.

Table 23-10 • Permitted Hostid Dialog Box Options

Option	Description
Hostid Type	Select the type of hostid that you want to use to restrict application usage. The following types of hostids are supported: • Ethernet address • Internet IP address • Machine name • User name • Windows disk serial number • Other
Hostid Value	Value identifying the hostid of the selected type. ***Note*** • *You can run lmutil.exe on any machine to get the needed hostid value if you do not know it.*

Options Dialog Box

The **Options** dialog box is opened by selecting **Options** on the **Tools** menu and includes the following options

Table 23-11 • Options Dialog Box Options

Option	Description
Default Output Folder	Click **Browse** and select the default output folder to use when license injecting a package. When using the FLEXwrap Assistant to license-inject a package, the directory you specify in this field will populate the **Output folder** field on the Finalize Package Page, where you can either accept this default location or specify another location.

Table 23-11 • Options Dialog Box Options

Option	Description
Show confirmation dialogs	To suppress the display of confirmation dialog boxes during the license-injection process, clear this option.

FLEXwrap Server Name Dialog Box

On the FLEXwrap Server Name dialog box, enter the machine name of the computer where the FLEXwrap server is installed. This dialog box opens the first time you open the FLEXwrap evaluation license file (`eval.lic`) on the Configure Connection Page.

To obtain a FLEXwrap evaluation license file, click the **Request a FLEXwrap server license** link on the Configure Connection Page.

Note • *The Evaluation License File has a license limit count of* ***5****, which means that only five licenses are available for an application that is FLEX-enabled using this license.*

Output Window

When you click **Build Package** on the Finalize Package Page to imitate the creation of a wrapped package, the Output Window opens and displays the messages that are generated during the process, including error messages.

To copy these comments so that you can paste them into another application (such as a word processing program), right-click in the Output window, select **Select All** from the context menu, and then select **Copy** from the context menu.

Managing FLEXwrap License Files

This user documentation explains how to use the FLEXwrap Server Configuration Tool, and is presented in the following sections:

Table 23-12 • FLEXwrap Server Configuration Tool Topics

Topic	Description
Overview of the FLEXwrap License Server	Explains how the FLEXwrap License Server is accessed, and discusses platform and communication issues.

Table 23-12 • FLEXwrap Server Configuration Tool Topics

Topic	Description
Using the FLEXwrap Server Configuration Tool to Manage the FLEXwrap License Server	Explains how to install a production FLEXwrap license file, test a wrapped application, start and stop the License Server, check the status of the License Server, and track projects.
Using the FLEXwrap Server Configuration Tool to FLEX-Enable Executable (.exe) Files	Explains how to FLEX-enable an executable file using the FLEXwrap Server Configuration Tool, and explains how to set up failsafe modes for an executable.
FLEXwrap Server Configuration Tool Reference	Documents FLEXwrap terminology and license parameters.

Note • *For an overview of AdminStudio Software Licensing technology and the FLEXwrap application, see About FLEXnet Licensing Technology and FLEXwrap.*

- *To license-enable Windows Installer packages, use the FLEXwrap Assistant. See Enabling License Tracking of Windows Installer Packages for more information.*

Overview of the FLEXwrap License Server

You can manage and monitor your FLEXwrap License Server using the FLEXwrap Server Configuration Tool. The FLEXwrap License Server serves licenses to wrapped applications based on the options that are set and the policies defined in the FLEXwrap license file.

Note • *For more information on license files, see About FLEXwrap License Files.*

When a user launches a wrapped application, the following occurs:

- **Wrapper launches**—When a user chooses to open an application, the Wrapper launches (which is transparent to the user).
- **Wrapper obtains location of License Server**—The Wrapper obtains the location of the FLEXwrap License Server from the server location file (`.lf`) or the `SWRAP_LICENSE_FILE` environment variable.
- **Wrapper contacts License Server to obtain a license**—The Wrapper attempts to obtain a license for the wrapped application from the License Server.
- **Wrapper launches the wrapped application**—If the Wrapper obtains a license for the application, the application is launched.

The following diagram illustrates how the FLEXwrap License Server is accessed:

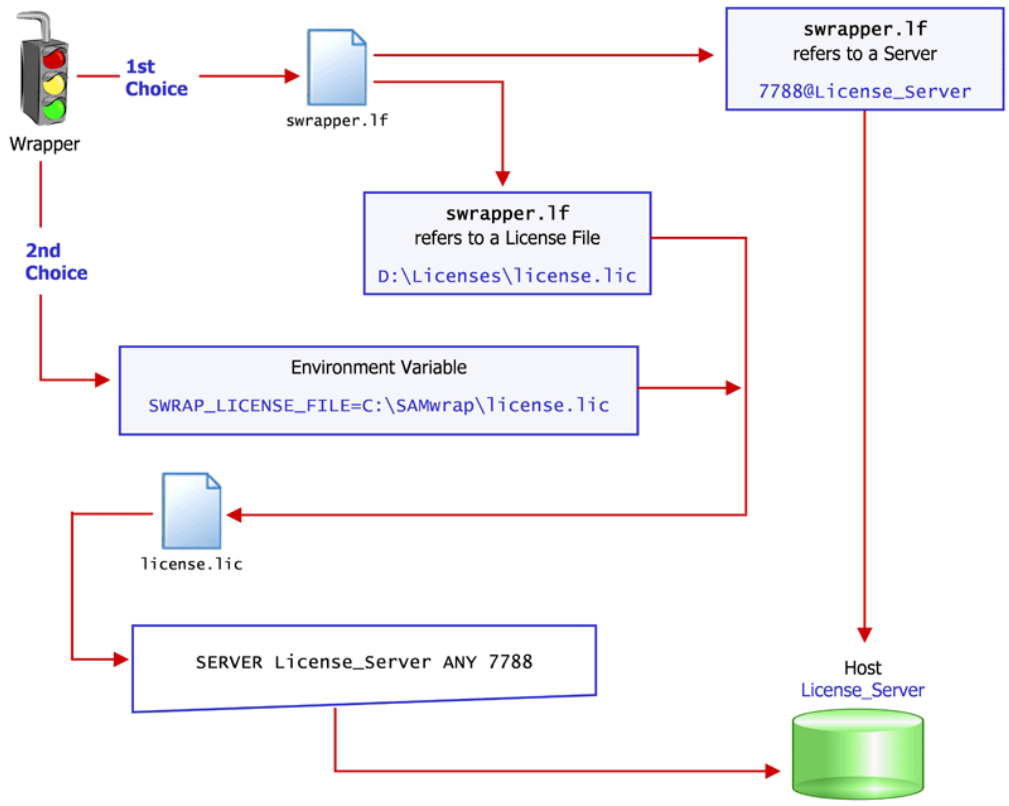


Figure 23-4: Accessing the FLEXwrap License Server

For detailed information on using the FLEXwrap Server Configuration Tool, see Managing FLEXwrap License Files.

Supported Platforms

Applications that run on Microsoft Windows 95 or later can be wrapped, regardless of the platform from which the application is served.

The License Server can run on:

- Windows NT 4 or later
- AIX, HP, Red Hat Linux, SGI, Solaris

Note • *For information about FLEX-enabling UNIX applications, contact AdminStudio Support.*

Communications Issues

FLEXwrap supports either TCP or UDP communications transport. The default for FLEXwrap is TCP. This should prove satisfactory in all circumstances; however, you have the option to select UDP communications. You can select UDP in either of the following ways:

Table 23-13 • Methods to Select UDP Communications

Method	Description
FLEXLM_COMM_TRANSPORT	Set the `FLEXLM_COMM_TRANSPORT` environment variable: `C:\>set FLEXLM_COMM_TRANSPORT= UDP`
COMM TRANSPORT	Select the comm transport in the options file with the line: `TRANSPORT UDP`

The order of precedence is:

1. Environment variable, if specified
2. Options file entry, if specified
3. Default TCP

Note • *When using UDP, FLEXwrap uses a default UDP time out of five minutes.*

- *Windows systems must be correctly installed and configured with the standard Windows Socket (WinSock) interface.*

Installation Notes

When you install the FLEXwrap software, there are some things you should keep in mind:

- **Do not run the License Server as root**—It is not necessary to install or run FLEXwrap as root and it is strongly recommended that the License Server not be run as root. The license file generator (`swsetup32.exe` and `swcrypter.exe`), however, should be protected from execution by your general user community.
- **License Server daemons need to be installed locally**—The License Server daemons (`lmgrd` and `swrap`) should be local to the system(s) on which they will run. So either install FLEXwrap on the machine where the License Server will run or copy `lmgrd` and `swrap` to the machine where the License Server will run. You should not run License Servers on diskless nodes.
- **Binaries to be wrapped need to be able to access the Wrapper**—Each binary to be wrapped must have access to the Wrapper binary, either on a local drive or through a permanent NFS mount point. So either copy the platform-specific Wrapper to each machine that contains a binary to be wrapped or NFS-mount the FLEXwrap drive to all machines that contain binaries to be wrapped.

About the FLEXwrap Server Configuration Tool

You can use the FLEXwrap Server Configuration Tool to monitor the usage of FLEX-enabled applications and manage FLEXwrap license files on the server. The FLEXwrap Server Configuration Tool serves the licenses to wrapped applications and can log usage of the licenses in a FLEXwrap report log. You can then generate usage reports using FLEXnet Manager.

For more information, see Managing FLEXwrap License Files and Using the FLEXwrap Server Configuration Tool to Manage the FLEXwrap License Server.

About the FLEXwrap LMTOOLS

FLEXwrap LMTOOLS is a utility used to manage the FLEXwrap License Server. Some of the functions FLEXwrap LMTOOLS performs include:

- Starting, stopping, and configuring FLEXwrap license server systems.
- Obtaining system information, including hostids.
- Obtaining server status.

The FLEXwrap LMTOOLS is launched from the Windows **Start** menu.

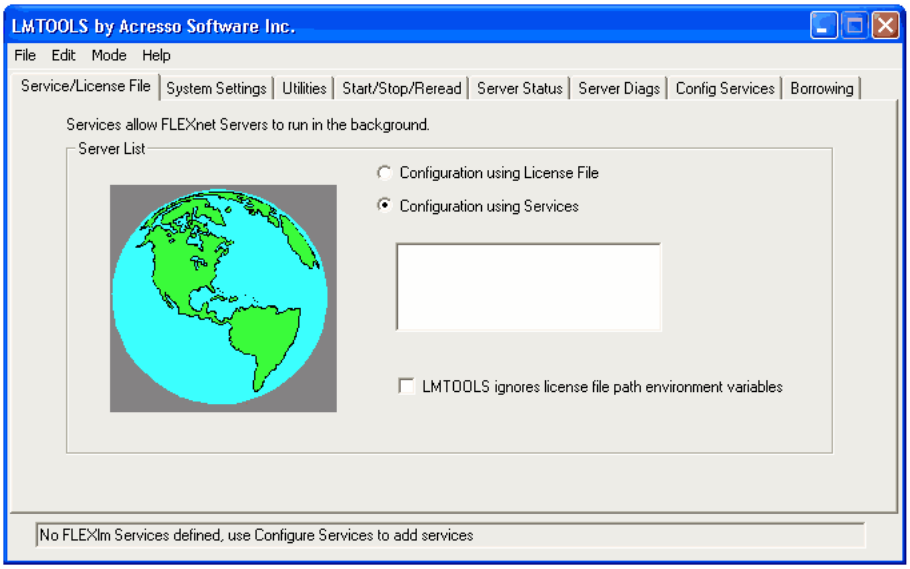


Figure 23-5: FLEXwrap LMTOOLS interface

LMTOOLS has two modes in which to configure a license server system:

- Configuring License Server Using a License File
- Configuring License Server Using Services

Configuring License Server Using a License File

If you configure your license server system using a license file, operations are performed on a particular license file. The file can be either local or remote. In this mode, you cannot start the `lmgrd` process, but you can do everything else.

Task: ***To configure your license server in license file mode:***

1. Launch LMTOOLS.
2. Click the **Configuration using License File** radio button.
3. Enter one or more the license file names or *port@host* specifications.

Configuring License Server Using Services

If you configure your license server system using services, operations are performed on a service, which allows starting `lmgrd` processes local to the system on which LMTOOLS is running. For details on configuring services, see Configuring FLEXwrap License Server to Run as a Service.

Using the FLEXwrap Server Configuration Tool to Manage the FLEXwrap License Server

This section explains how to use the FLEXwrap Server Configuration Tool to manage the FLEXwrap License Server, and includes the following procedures:

- About the FLEXwrap Server Configuration Tool
- Installing Your Production FLEXwrap License File
- Testing a Wrapped Application
- Starting the License Server
- Stopping the License Server
- Checking the Status of the FLEXwrap License Server
- Tracking Projects with FLEXwrap

Installing Your Production FLEXwrap License File

These instructions assume that you have already installed FLEXwrap and been working with a FLEXwrap evaluation license file (`eval.lic`).

Note • *The FLEXwrap sample evaluation license file is installed with the product in the following location:*

AdminStudio Installation Directory\FLEXwrap\Server

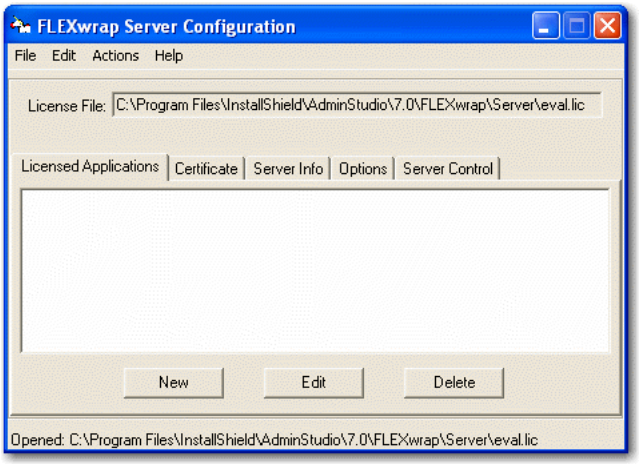
***Important** • The evaluation license file (eval.lic) is for evaluation purposes only; you are not permitted to use it in a production environment, and it can only be used for a limited amount of time before expiring. When purchasing FLEXwrap with AdminStudio, you do not receive a production license file. To obtain a production license file, you need to purchase FlexNet Manager. For more information, see About FLEXwrap License Files.*

The hostid(s) from the SERVER line(s) of your production FLEXwrap license file is encrypted into the license keys generated for FEATURE lines for wrapped applications that you add to the FLEXwrap license. You can update your FLEXwrap evaluation license file with the production FLEXwrap license information either using the FLEXwrap Server Configuration Tool or manually:

- **Using the FLEXwrap Server Configuration Tool**—If you update your FLEXwrap evaluation license with your production FLEXwrap license information using the FLEXwrap Server Configuration Tool, any license keys that you generated using your evaluation license will be regenerated automatically.
- **Manually**—If you manually update your FLEXwrap evaluation license file with your production FLEXwrap license information by editing the file, you will have to use the FLEXwrap License Encryption Utility (swcrypter.exe) to regenerate its license keys for wrapped applications. The FLEXwrap License Encryption Utility is used to manually sign FLEXwrap licenses.

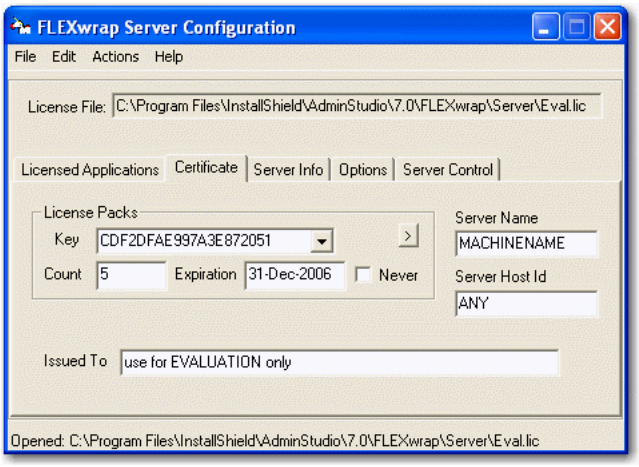
Task: ***To update your FLEXwrap license file from the evaluation file:***

1. Backup your FLEXwrap evaluation license file (eval.lic) to another file name (such as evallic.org) so that you do not accidentally modify it. Using a different extension than .lic is recommended to prevent the Server from automatically loading the license file when it starts.
2. Launch the FLEXwrap Server Configuration Tool from the Windows **Start** menu. The **FLEXwrap Server Configuration** dialog box opens.

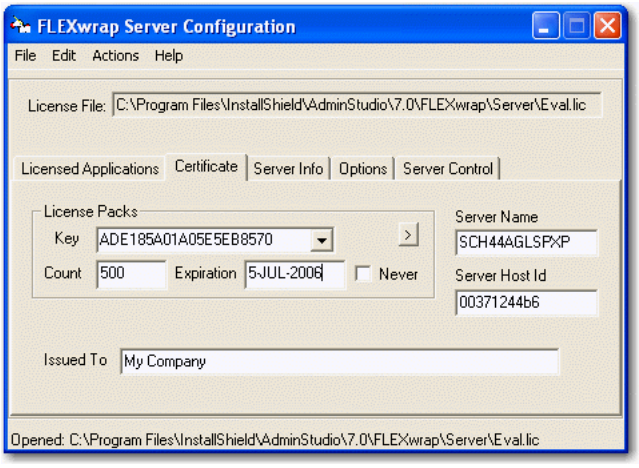


You use this dialog box to install your FLEXwrap license file, to generate licenses for the applications to be wrapped, and to wrap individual executable files.

3. If you have not already opened your evaluation license file, click **Open** on the **File** menu and select your FLEXwrap evaluation license file (`eval.lic`).
4. If you are running the Server, open the **Server Control Tab** and stop the Server.
5. Click the **Certificate** tab. The **Certificate Tab** opens.



6. Replace the entries in the **Certificate Tab** fields with the exact production FLEXwrap license information you received from AdminStudio Support.
   - **Key**—License key provided by AdminStudio Support.
   - **Count**—The number of concurrent licenses available for this FLEX-enabled application.
   - **Expiration**—The date these licenses will expire.
   - **Server Host Id**—The host name of the FLEXwrap License Server listed in the production license file.
   - **Issued To**—The name of the company that purchased the license.
7. Enter the **Server Name** associated with the **Server Host Id** that you entered.



8. On the **File** menu, click **Save** to save the license file and enter a name using the `.lic` extension, such as `flexwrap.lic`.

   Because you have updated your FLEXwrap license using the FLEXwrap Server Configuration Tool, any `FEATURE` lines in the license file that you created with your FLEXwrap evaluation license information will be rekeyed using your production FLEXwrap license information.

9. Start the FLEXwrap License Server from the **Server Control** tab.

Note • *If you generate a new Windows Installer package using the FLEXwrap Assistant using the evaluation license file, and if you had selected the option to run even if the server is down, you should rewrap the package once you receive the production license because the local license copy installed will still be the old one.*

Starting the License Server

There are three options for starting the FLEXwrap License Server:

Table 23-14 • Methods of Launching the FLEXwrap License Server

Option	Description
Configuring FLEXwrap License Server to Run as a Service	Launch an installation from the FLEXwrap interface to configure the License Server to run as a service (to start automatically when the system is booted).
Manually Starting the FLEXwrap License Server from the FLEXwrap Interface	Click a button in the FLEXwrap interface.
Manually Starting the FLEXwrap License Server from the Command Line	Run a command at a Windows console prompt.

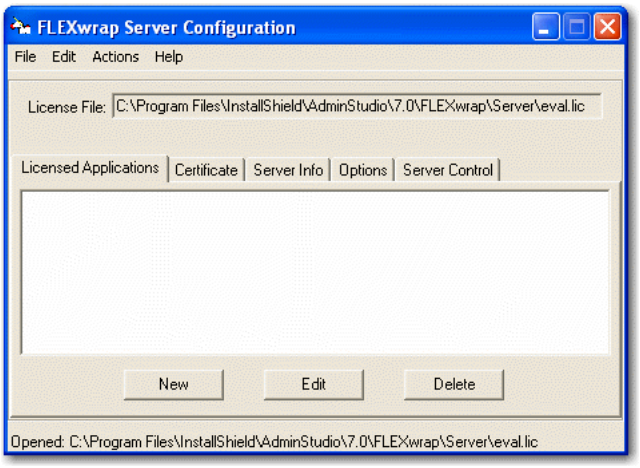
Note • *When you purchase FLEXwrap, you specify the hostid to which the License Server will be locked.*

Configuring FLEXwrap License Server to Run as a Service

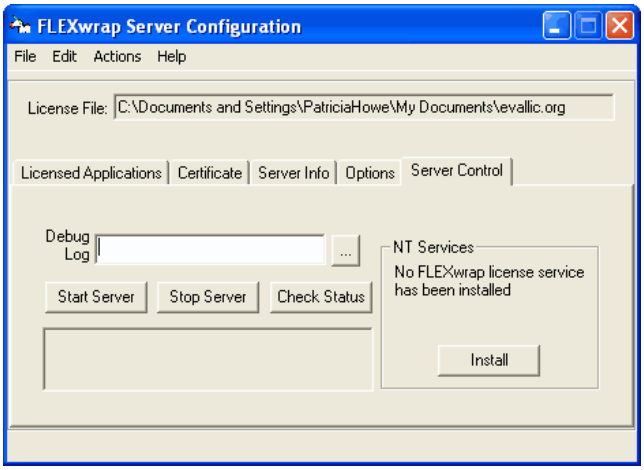
You can configure the FLEXwrap License Server to run as a service by selecting the **NT Services** option on the **Server Control** tab of the **FLEXwrap Server Configuration** dialog box.

Task: ***To configure the FLEXwrap License Server to run as a service:***

1. Launch the FLEXwrap Server Configuration Tool from the **Start** menu. The **FLEXwrap Server Configuration** dialog box opens.



2. On the **File** menu, click **Open** and select the FLEXwrap license file.
3. Click the **Server Control** tab. The **Server Control** tab opens.



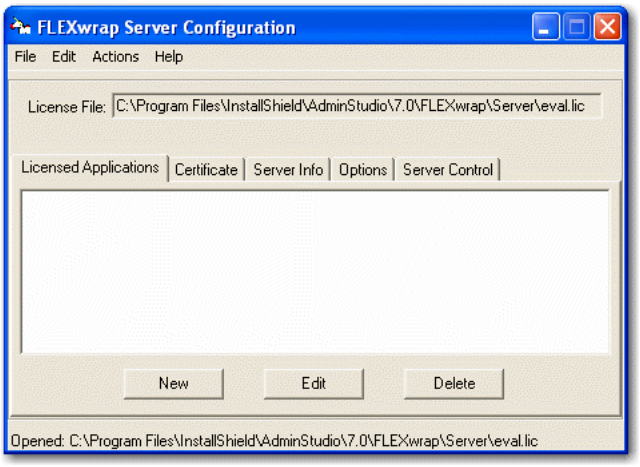
4. Click the **Install** button in the **NT Services** area. The selected license file is now configured to run as a service.

Manually Starting the FLEXwrap License Server from the FLEXwrap Interface

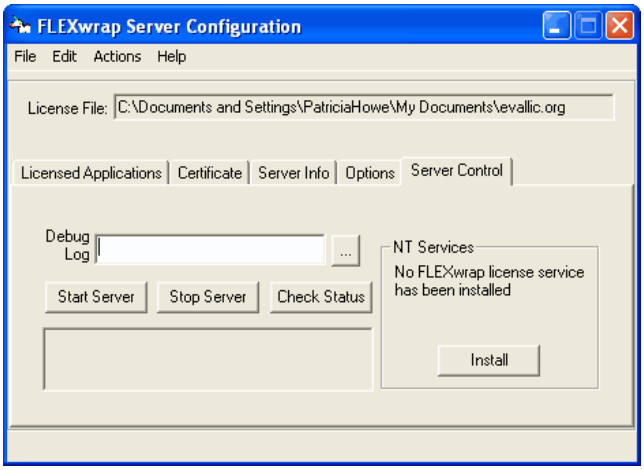
You can manually start the FLEXwrap License Server on the **Server Control** tab of the **FLEXwrap Server Configuration** dialog box.

Task: ***To start the FLEXwrap License Server manually from the FLEXwrap interface:***

1. Launch the FLEXwrap Server Configuration Tool from the **Start** menu. The **FLEXwrap Server Configuration** dialog box opens:



2. On the **File** menu, click **Open** and select the FLEXwrap license file.
3. Click the **Server Control** tab. The **Server Control Tab** opens.



4. In the **Debug Log** field, enter the path to a debug log or select an existing debug log.

Note • *If you do not specify a debug log when you start the License Server, at least two console windows will appear—one for the* `lmgrd` *output, one for the* `swrap` *output, and one for each of any other vendor daemons started by this* `lmgrd`*.*

5. Click the **Start Server** button. All licenses (`*.lic`) in the directory of the opened license are loaded.

Manually Starting the FLEXwrap License Server from the Command Line

You can manually start the FLEXwrap License Server by running a command at the Windows console prompt.

Task: ***To start the FLEXwrap License Server from the command line:***

Run the following command at a Windows console prompt:

```
C:\path_to_lmgrd.exe -c path_to_license_file
```

For example:

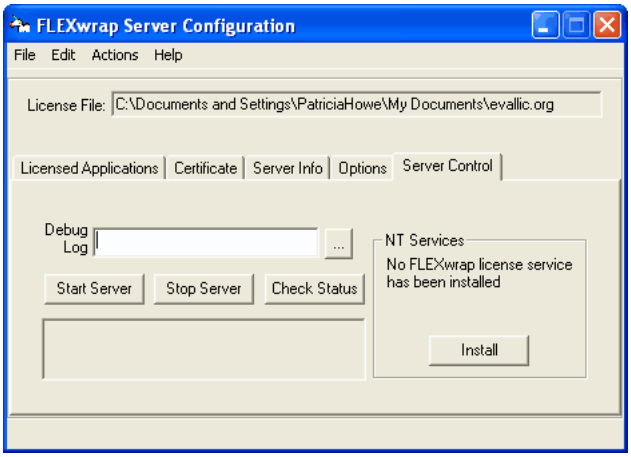
```
"C:\Program Files\SAMwrap\v8.2\lmgrd.exe" -c "C:\Program Files\SAMwrap\v8.2\FLEXwrap.lic"
```

Stopping the License Server

You can stop the FLEXwrap License Server on the **Server Control** tab of the **FLEXwrap Server Configuration** dialog box.

Task: ***To stop the License Server:***

1. Launch the FLEXwrap Server Configuration Tool from the **Start** menu. The **Licensed Applications** tab of the **FLEXwrap Server Configuration** dialog box opens.
2. On the **File** menu, click **Open** and select the FLEXwrap license file.
3. Click the **Server Control** tab. The **Server Control** tab opens.



4. Click **Stop Server.**

Checking the Status of the FLEXwrap License Server

You can check the status of the FLEXwrap License Server on either the **Server Control** tab of the **FLEXwrap Server Configuration** dialog box or on the **Server Status** tab of the **LMTOOLS** dialog box. The LMTOOLS status report is slightly more detailed than the report generated by the FLEXnet interface.

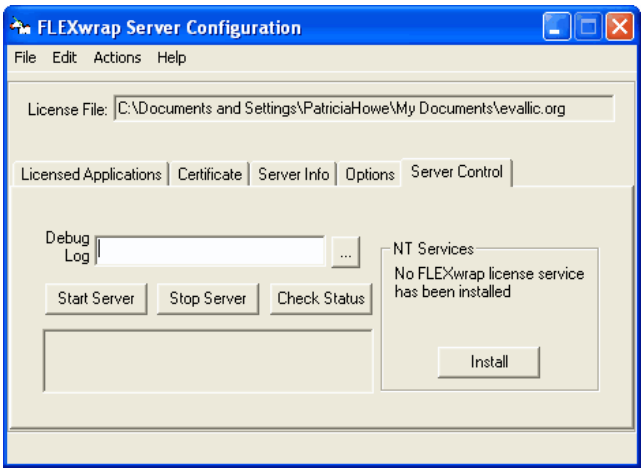
Checking the License Server Status from the FLEXwrap Server Configuration Tool

You can check the status of the FLEXwrap License Server using the FLEXwrap Server Configuration Tool.

Task: ***To check the status of the License Server from the FLEXwrap interface:***

1. Launch the FLEXwrap Server Configuration Tool from the **Start** menu. The **Licensed Applications** tab of the **FLEXwrap Configuration** dialog box opens.
2. On the **File** menu, click **Open** and select the FLEXwrap license file.

3. Click the **Server Control** tab. The **Server Control** tab opens.



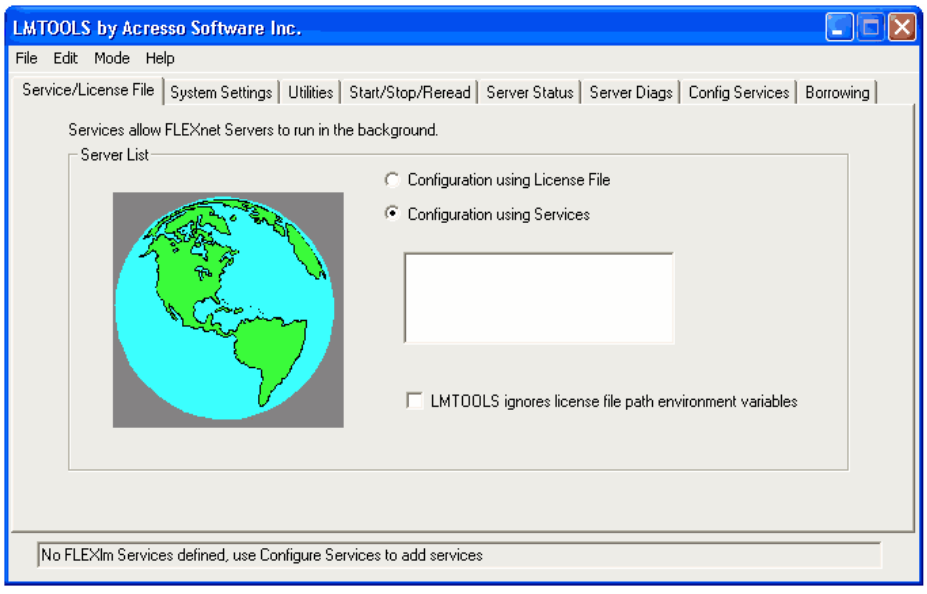
4. Click **Check Status**. A status message is displayed.

Checking the License Server Status from the LMTOOLS Interface

You can use the FLEXwrap LMTOOLS utility to check the status of the FLEXwrap License Server.

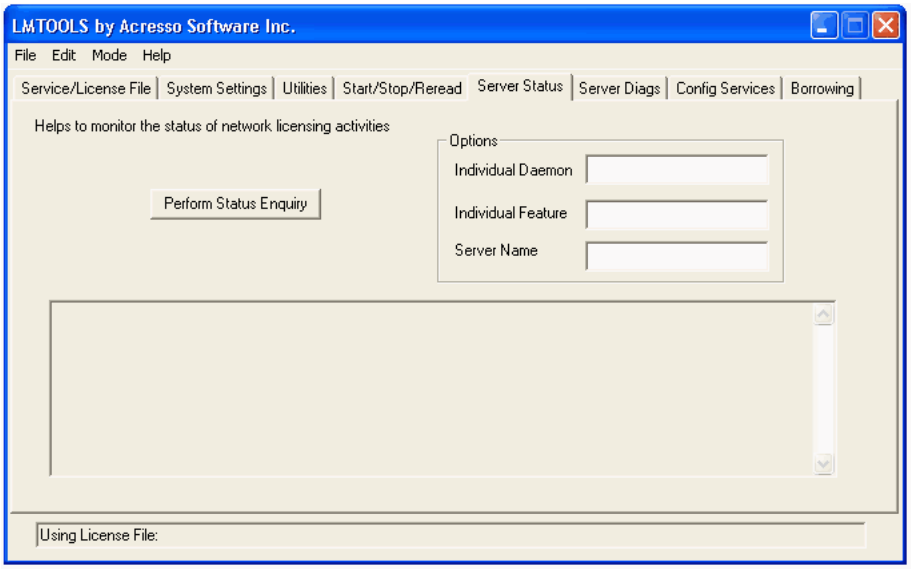
Task: ***To check the status of the License Server from the LMTOOLS interface:***

1. Open the **FLEXwrap LMTOOLS** utility from the **Start** menu. The **Service/License File** tab of the **LMTOOLS** dialog box opens.



2. On the **File** menu, click **Open** and select the FLEXwrap license file.

3. Click the **Server Status** tab. The Server Status tab opens:



4. Click **Perform Status Inquiry**. A Status Report opens.

Tracking Projects with FLEXwrap

FLEXwrap allows you to track usage of FLEXwrap-managed applications by project (or department or account). Prior to running the wrapped application, the user must set the `LM_PROJECT` environment variable to the name of the project to be tracked. Enter the following at the command prompt:

```
C:\> set LM_PROJECT=my_project
```

The project name is written to the report log when a license checkout occurs. You can then filter, summarize, and sort on these projects when you generate reports using FLEXnet Manager.

Using the FLEXwrap Server Configuration Tool to FLEX-Enable Executable (.exe) Files

In addition to managing the FLEXwrap Server, you can also use the FLEXwrap Server Configuration Tool to FLEX-enable an executable (`.exe`) file. The procedure for doing so are in this section.

However, if you want to FLEX-enable all of the executables in a Windows Installer package, use the FLEXwrap Assistant. See Enabling License Tracking of Windows Installer Packages.

Overview

Putting an application under FLEXwrap management involves the following five basic steps:

Table 23-15 • Basic Steps of Wrapping an Application

#	Step Name	Description
1	**Add FEATURE Line to License File**	Generate a FEATURE line for the application to be wrapped and add this FEATURE line to the FLEXwrap license file.
2	**Start the FLEXwrap License Server**	Make the FLEXwrap License Server serve licenses for the new feature by issuing a "reread" command (if the License Server is already running) or by starting the FLEXwrap License Server (if it is not already running).
3	**Rename Executable File**	Rename the application executable to use a `.fwr` extension.
4	**Replace Original Executable File with Wrapper File**	Copy the Wrapper file (`swrap_nt.exe`) to the application's location, and give it the original name of the application.
5	**Identify License File Path**	In the wrapped application's directory, create a license file named `swrapper.lf`, a plain text file containing `port@host` for the `lmgrd` process (a License Server daemon) and/or the path to a copy of the license file. `swrapper.lf` is used by the Wrapper binary to request a license from the License Server.

These basic steps for wrapping an application are illustrated in the following diagram:

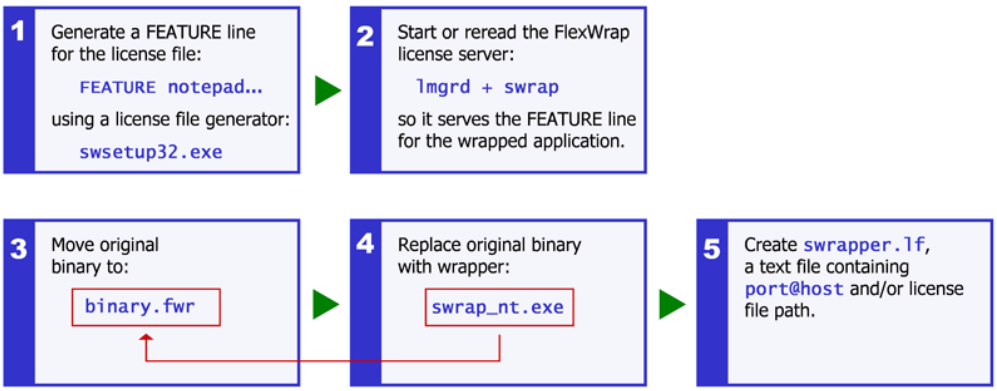


Figure 23-6: Basic steps for wrapping an application

When following these steps to wrap an application, note the following:

- If an application is installed locally on multiple machines, **Step 3** through **Step 5** must be repeated for each installed copy.

- For License Server failsafe mode, additional steps are needed. See Setting Up License Server Failsafe Mode for instructions.

After you perform these steps, no additional setup needs be done by your end users. The following diagram provides an overview of the basic steps for wrapping an executable file:

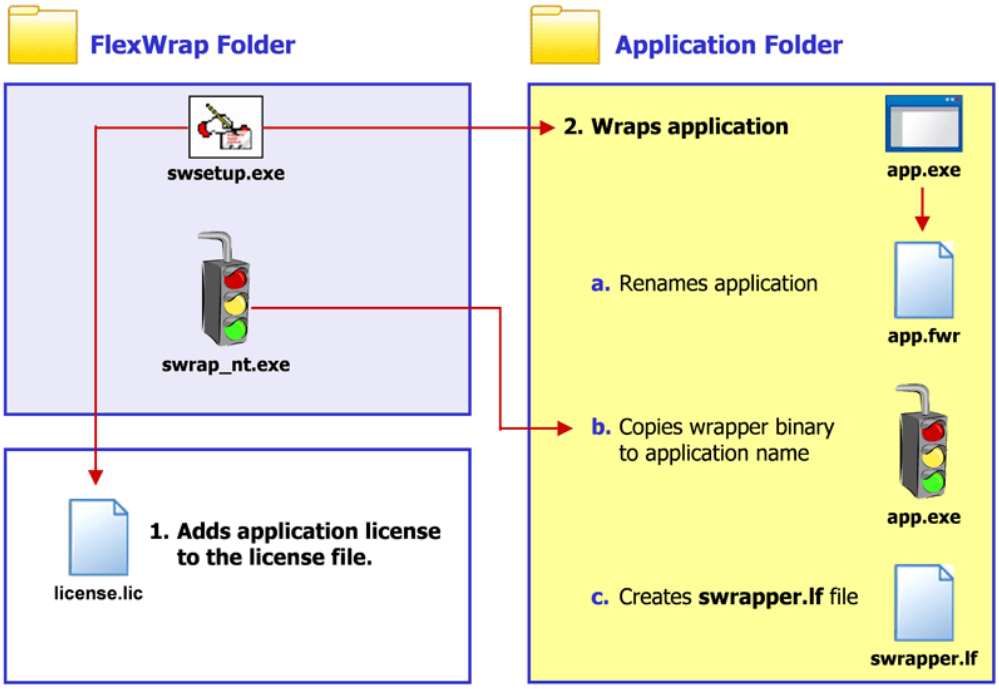


Figure 23-7: Saving the license file and application

Note • *To license-enable Windows Installer packages, use the FLEXwrap Assistant. See Enabling License Tracking of Windows Installer Packages for more information.*

Available Options

When using the FLEXwrap Server Configuration Tool to wrap an executable, you can set the same limits, behavior, restriction, and configuration options that you can set when using the FLEXwrap Assistant to wrap a Windows Installer package.

Table 23-16 • FLEXwrap Server Configuration Tool Options when Wrapping an Executable

Option	Description
Set License Limit	Restrict usage of a software application to a specified number of users to run on any machine (counted).
Restrict Usage by Machine	Restrict usage of a software application to either an unlimited number of users or a specified number of users on a specified computer
Expiration Date	Set an expiration date.
Overdraft Support	Allow overdraft usage (reports show usage in excess of license limit).
Queuing	Permit queueing when licenses are not available.
Suite Option	Enforce licensing terms and conditions of product suites.
Restrict Usage by Hostids	Reserve licenses for particular users, machines, or sites.
Group License Requests	Include or exclude specific users, hosts, or displays or groups of users or hosts.
Reporting	Logging license usage

FLEX-Enabling an Executable (.exe) File

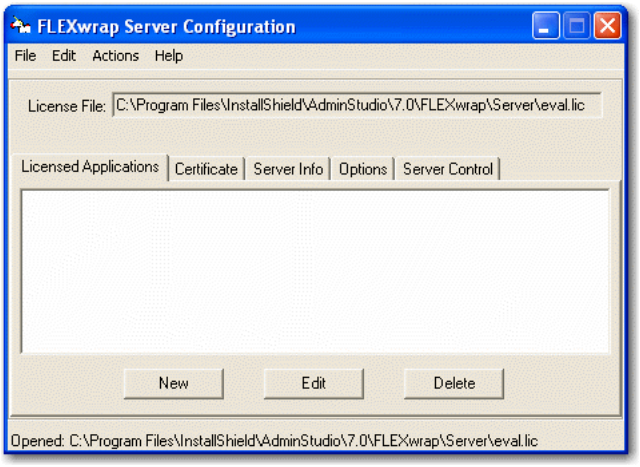
License-enabling an executable (.exe) file is a four step process:

- Step 1: Specifying Basic Settings for the License
- Step 2: Specifying Advanced Settings for the License and Binary
- Step 3: Creating an Options File
- Step 4: Saving the New License File and Wrapped Application

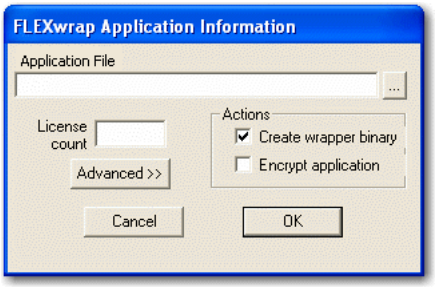
Specifying Basic Settings for the License

Task: ***To specify basic settings for the license:***

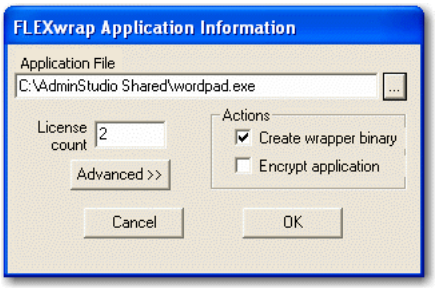
1. Launch the FLEXwrap Server Configuration Tool from the Windows **Start** menu. You will be prompted whether to open the last license file that you worked with in the FLEXwrap Setup dialog box.
2. Click the **No** button.The **FLEXwrap Server Configuration** dialog box opens.



3. On the **File** menu, click **Open** and select your production FLEXwrap license file. After you select the license file, the **FLEXwrap Server Configuration** dialog box displays the path to this license file and a list of programs (if any) that have been licensed with FLEXwrap.
4. Click **New**. The **FLEXwrap Application Information** dialog box opens.



5. In the **Application File** field, select the application binary that you are going to wrap.
6. In the **License Count** field, type the license limit for the application. This is the number of licenses that the License Server will allow to be checked out simultaneously for this application. The license limit will appear in reports as the maximum number of licenses available. The total of the counts of licenses on all the `FEATURE` lines you generate may not exceed the count on the FLEXwrap `INCREMENT` line in your license file. If you are just testing FLEXwrap's functionality, enter 2.
7. Make sure the **Create Wrapper binary** option is selected.

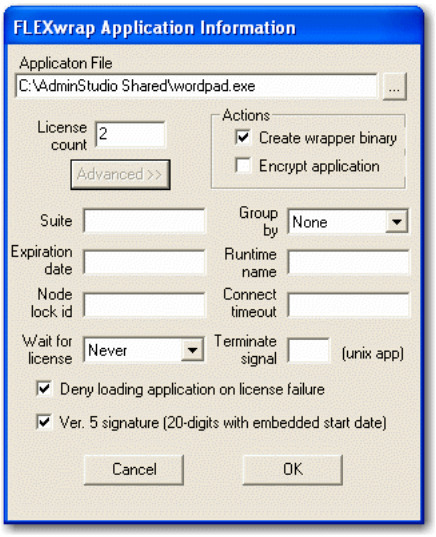


8. Continue with the steps in Specifying Advanced Settings for the License and Binary.

Specifying Advanced Settings for the License and Binary

Task: ***To specify advanced settings for the license and binary:***

1. On the **FLEXwrap Application Information** dialog box, click the **Advanced** button. The dialog box expands and the advanced licensing options appear.



2. Specify the desired options. See Specifying FLEXwrap License Parameters for detailed instructions.

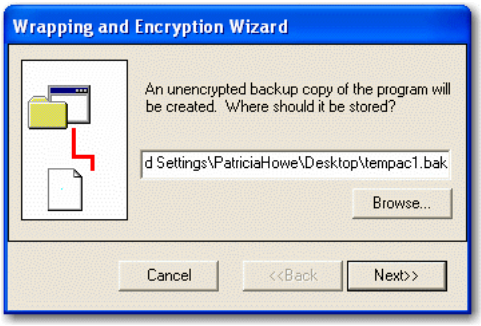
   The most important advanced option to consider is whether you want the wrapped application to be able to run without being able to check out a license from a running License Server (License Failsafe Mode). By default, the **Deny loading application on license failure** option is selected.

   - **Run when no license is available**—If you want the wrapped application to be able to run whether or not it can get a license from a running FLEXwrap License Server (License Failsafe Mode), clear the **Deny loading application on license failure** option. See Setting Up Regular (License) Failsafe Mode.
   - **Run when cannot connect to License Server**—If you want the wrapped application to be able to run whether or not the FLEXwrap License Server is running (License Server Failsafe Mode), clear the **Deny**

loading application on license failure option and follow the instructions in Setting Up License Server Failsafe Mode.

- **Cannot run without a license**—If you wish to require the Wrapper to obtain a license in order to run the wrapped application, select the **Deny loading application on license failure** option.

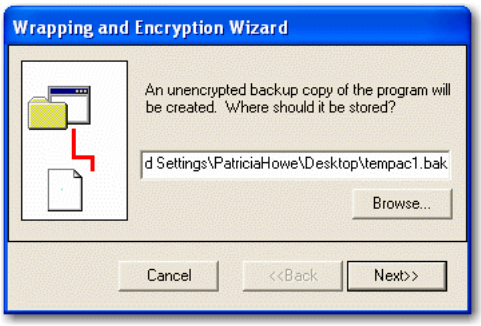
3. In some cases, the application binary can be encrypted so that it can be run only by the Wrapper. This would prevent users from running the wrapped application binary (`binary.fwr`) directly. If you want to encrypt the application binary, select the **Encrypt application** option.
4. Click **OK**. If you selected the **Encrypt application** check box, the **Wrapping and Encryption Wizard** dialog box opens in which you can specify where you would like a backup copy of the application binary to be saved. The default is `binary.bak` in the application directory.



If you did not select the **Encrypt application** option skip to **Step 11**.

5. Click **Next**. The second panel of the **Wrapping and Encryption Wizard** opens, where you are prompted to identify the path to the License Server.

6. Select the **Host and port number reference** option and enter the host and port number, or select the **Path to the license file** option and enter the appropriate path. This information is written into `swrapper.lf`. You would normally want to select `port@host`.
7. Click **Next**. You are prompted to identify the location where you want FLEXwrap will save a copy of the original binary prior to encryption.

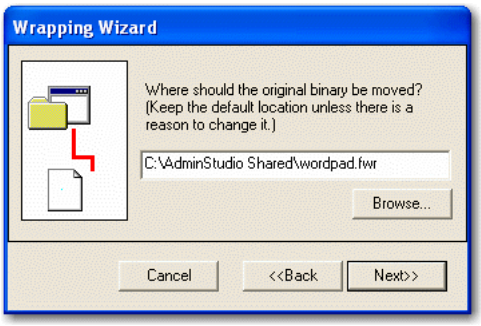


8. Identify the location and click **Next**. A message appears stating that FLEXwrap is ready to encrypt and wrap the application.
9. Click **Finish** A message appears confirming that the application was encrypted successfully.
10. Click **OK**. You have successfully wrapped and encrypted the executable. Now continue with Creating an Options File.

Note • *This encryption feature of FLEXwrap is not guaranteed because some applications can be configured to discourage any alteration, and encryption of the application binary may not be allowed.*

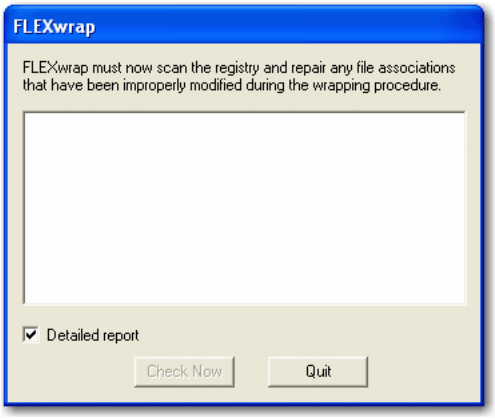
11. If you did not choose to encrypt the application, click **OK** in the **FLEXwrap Application Information** dialog box. The **Wrapping Wizard** dialog box opens and you are prompted to select how you want the Wrapper to find the FLEXwrap License Server.

12. Select the **Host and port number reference** option and enter the host and port number, or select the **Path to the license file** option and enter the appropriate path. This information is written into `swrapper.lf`. You would normally want to select **port@host**.
13. Click **Next**. If you are setting up an uncounted node-locked license, you will be prompted to select whether you want uncounted node-locked licenses to be logged. If you select both logging uncounted node-locked licenses and failsafe mode, failsafe mode may not work correctly.
14. Decide whether to select this option and click **Next**. You are prompted to specify where you would like a backup copy of the application binary to be saved. The default is `binary.fwr` in the application directory.

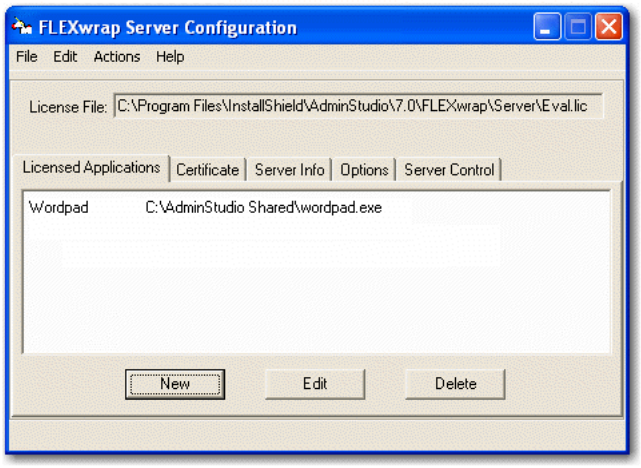


15. Specify the location and click **Next**. A message appears stating that FLEXwrap is ready to wrap the application.

16. Click **Finish**. A command window and a FLEXwrap dialog box appear.



17. Click **Check Now** to scan/repair the registry.
18. When the dialog box displays `Done`, click **Quit**. The **Wrapping Wizard** dialog box should display `Finished`. The FLEXwrap license has been modified, but not yet saved.
19. Click **OK** in the **Wrapping Wizard** dialog box. In the **FLEXwrap Server Configuration** dialog box, you should now see the application in the list of **Licensed Applications**.



20. Continue with the steps in Creating an Options File.

Creating an Options File

If you want to record license usage in a report log, create a FLEXwrap options file. You could then generate usage reports using FLEXnet Manager.

Task: ***To create an options file:***

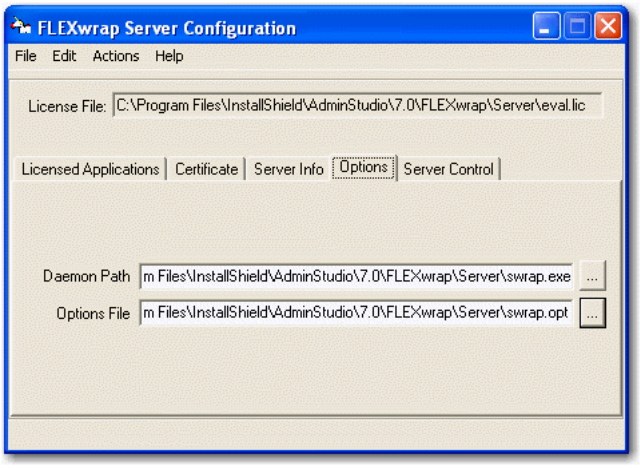
1. Create a plain text file called `swrap.opt` and save it in the same directory with your license file.
2. Type a `REPORTLOG` line in the options file:

   ```
   REPORTLOG +path_to_report_log
   ```

   For example:

   ```
   REPORTLOG +C:\flexlm\swrap\swrap.rl
   ```

   If you already have a FLEXwrap options file, edit it to contain a `REPORTLOG` line.
3. In the **FLEXwrap Server Configuration** dialog box, click the **Options** tab.
4. In the **Options File** box, enter the path (including file name) to the FLEXwrap options file.



5. Continue with the steps in Saving the New License File and Wrapped Application.

Setting the Path to the FLEXwrap Options File

When you save the FLEXwrap license file in the next section, the path to the FLEXwrap options file will follow the path to the `swrap` daemon:

```
DAEMON swrap swrap.exe path_to_options_file
```

For example:

```
DAEMON swrap swrap.exe C:\flexlm\swrap\swrap.opt
```

When `lmgrd` starts the `swrap` daemon, the daemon will read the options file and append license usage information to the report log.

Saving the New License File and Wrapped Application

Task: ***To save the new license file and wrapped application:***

1. On the **File** menu of the **FLEXwrap Server Configuration** dialog box, click **Save** to save the updated FLEXwrap license file in its original location.
2. In the folder that contains the application binary file, you will see three files:
   - `program.exe` is now a copy of the Wrapper binary (notice the difference in file size)
   - `program.fwr` is the original program file that has been renamed
   - `swrapper.lf` is the location file that the Wrapper binary uses to locate the License Server

Testing a Wrapped Application

When you launch the wrapped application, the Wrapper reads the location file (`swrapper.lf`) to obtain the location of the License Server (`port@host`), contacts the License Server at that location, obtains a license, and launches the wrapped application (*`binary`*`.fwr`).

Task: ***To test the wrapped application:***

1. Launch the wrapped application in the same way as you launched it before it was wrapped, such as double-clicking the icon.

   Note • *If the application doesn't run, and you chose to encrypt the application, it may be that the application cannot be encrypted. Restore the backup copy of the application binary and rewrap the binary without encrypting.*

2. Check the status of the License Server either with LMTOOLS or at the command line with the following command:

   ```
   % lmutil lmstat -a -c path_to_license_file
   ```

3. To test what happens when all available licenses are checked out, invoke the wrapped application up to its license limit and then invoke one more.

   - **If you have set up license failsafe mode**, the additional instance of the wrapped application should run.
   - **If you did not set up license failsafe mode**, the additional instance of the wrapped application should not run when it fails to get a license. Instead, you will see a message like the following if you set the Wait for license choice list to **Ask** or **Always**:

     ```
     The maximum number of users for "application" has been reached.

     Do you wish to be notified when a license becomes available?

     Current user(s) are:

     dodi at zippy on /dev/ttypa (v1.0), started on Tuesday 6/17 at 17:51

     dodi at speedy on /dev/ttyq8 (v1.0), started on Tuesday 6/17 at 17:51

     If you choose Yes, the license server will queue for a license. You will see a
        dialog that says, "Waiting for [application] license" and when a license
        becomes available, the application will run.
     ```

4. To generate a report of your usage, invoke the wrapped application a few more times, allowing some time to elapse before checking licenses back in.

5. Then send a reread command to the License Server. The reason for the reread is that v5+ report logs are compressed and authenticated and information is written to the report log only periodically. A reread flushes

all usage information into the report log. If you require up-to-the-minute reporting, you must send a reread command to the License Server before generating a report.

A reread can be done using the LMTOOLS program or at the command line with the following command:

```
C:\Program Files\SAMwrap\lmutil lmreread -c C:\Program Files\SAMwrap\v8.2\FLEXwrap.lic
```

Now you can generate a license usage report using FLEXnet Manager.

Setting Up Fail Safe Modes

There are two instances when a wrapped application will be unable to obtain a license. You may want to allow a wrapped application to run without a license in either or both of these instances:

- **Regular (License) Fail Safe Mode**—The Wrapper contacts the License Server, but all licenses for the feature are checked out. Having failsafe mode specified in the license file allows the wrapped application to run without a license. This mode relies on the Wrapper contacting the License Server.
- **License Server Fail Safe Mode**—The Wrapper is unable to contact the License Server, either because the License Server is not running or because of a network problem. Having failsafe mode specified in the license file and ensuring that the Wrapper can read a current local copy of the license file allows the wrapped application to run without a license.

The topics in this section explain how to set up both license and License Server failsafe modes:

- Setting Up Regular (License) Failsafe Mode
- Setting Up License Server Failsafe Mode

Setting Up Regular (License) Failsafe Mode

In the case where the Wrapper can contact the License Server but all licenses for the feature are checked out, a specification of failsafe mode in the `FEATURE` line for the wrapped application allows the wrapped application to run without a license.

The Wrapper uses `port@host` in `swrapper.lf` to contact the License Server and learns from the License Server that, although all licenses are checked out, failsafe mode was enabled in the license file. A denial is logged in the report log, the Wrapper runs the application binary, and the usage of the wrapped application is not recorded.

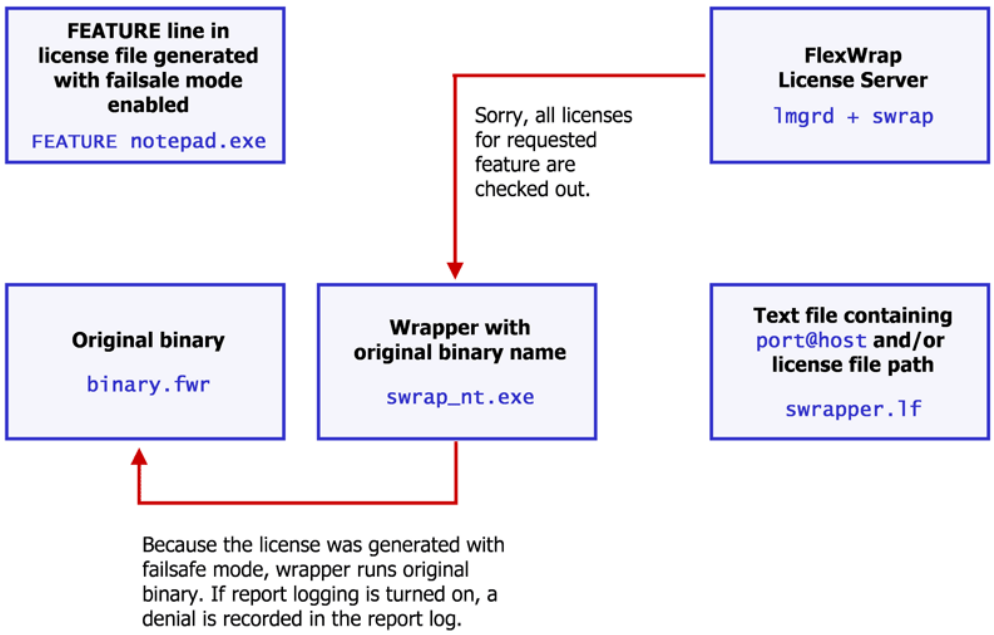


Figure 23-8: Regular (License) Failsafe Mode

This level of failsafe is often adequate. For example, if the wrapped application is being served from the same machine on which the FLEXwrap License Server runs, network communication is not a concern—keeping the License Server up and running would be the only thing necessary to ensure that the wrapped application can always run.

Setting Up License Server Failsafe Mode

In the case where the Wrapper cannot contact the License Server, if `port@host` is the only information in `swrapper.lf`, there is no way for the Wrapper to learn that failsafe mode was enabled in the license file.

Unless steps have been taken to ensure that the Wrapper can read a copy of the license file, the wrapped application will not run, even though failsafe mode was specified in the license file. The drawback to ensuring access to the license file is that you may need to make and maintain extra copies of the license file.

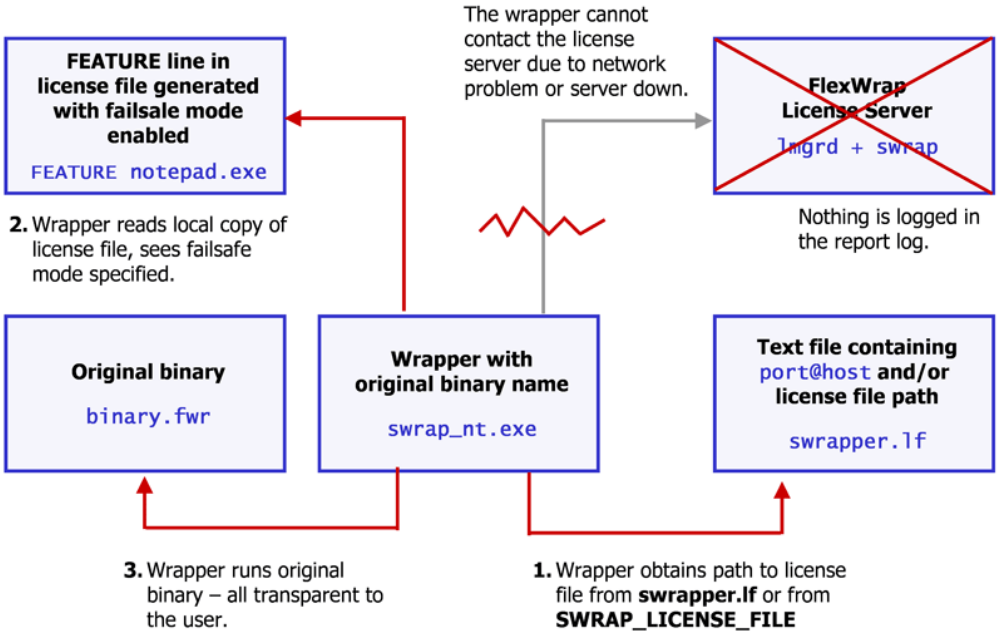


Figure 23-9: License Server Failsafe Mode

Perform the following steps to ensure License Server failsafe.

Task: ***To ensure License Server failsafe:***

1. Make sure you specified failsafe mode when you generated the `FEATURE` line for the wrapped application. The `FEATURE` line will contain the words: `VENDOR_STRING=failsafe`. This is part of the license key, so you cannot just add these words to an existing `FEATURE` line; you must specify failsafe mode when generating the license. See FLEX-Enabling an Executable (.exe) File for instructions on generating license files.
2. If the FLEXwrap License Server runs on a different machine from the machine where the wrapped binary resides, remove the network dependency by copying the FLEXwrap license file to a drive local to the wrapped binary. We recommend putting the license file in the directory where the wrapped binary resides and calling it `FLEXwrap.lic`.
3. Edit `swrapper.lf` in the directory where the wrapped binary resides and add a semi-colon (;) and the path to the license file on the local disk. For example:

   ```
   7788@machinea;"C:\Program Files\SAMwrap\v8.2\FLEXwrap.lic
   ```

   If, when installing FLEXwrap on Windows, you chose to use the path to the license file instead of `port@host` to find the FLEXwrap license file, you will not have to edit `swrapper.lf` because it already contains the information it needs to find the license file.

   The Wrapper will also check the `SWRAP_LICENSE_FILE` environment variable after checking `swrapper.lf`.

Note • *FLEXwrap does not replace a built-in license manager. If you wrap an application that has a built-in license manager, the application must obtain a license from FLEXwrap in order to try to ask for a license from its built-in license manager. By enabling failsafe mode (such as allowing a wrapped application to run even if it cannot get a license from a running license server), you can guarantee that a FLEXwrap License Server will never prevent the application from getting to its own license manager.*

FLEXwrap Server Configuration Tool Reference

FLEXwrap Reference information is organized into the following areas:

- FLEXwrap Terminology
- About FLEXwrap License Parameters

FLEXwrap Terminology

The following terms are defined in the context of the FLEXwrap software and are used throughout this documentation:

Table 23-17 • FLEXwrap Terminology

Term	Description
Application	In this documentation, an application consists of one binary file to be put under FLEXwrap license management. The application might be a word processor, compiler, or CAD tool, for example.
Failsafe mode or License failsafe mode	A mode of operation that allows the Wrapper to run the wrapped application without a license when the FLEXwrap License Server is running. This mode is enabled by specifying failsafe mode when the license is generated. See Setting Up Fail Safe Modes for more information.
FEATURE line	A line of text generated by the FLEXwrap administrator at your site, using a license file generator. This line describes the license policies for a wrapped application and is appended to your FLEXwrap license file.
License	A software “token” that represents the right to use a wrapped application.
License administration tools	Programs that allow the FLEXwrap administrator to manage the License Server: `FLEXlm utilities—lmutil.exe, lmtools.exe (LMTOOLS)`

Table 23-17 • FLEXwrap Terminology (cont.)

Term	Description
License file	A text file containing the FEATURE lines for all the applications under FLEXwrap management. This file also contains information about where the License Server runs, the location of the swrap daemon, and the location of the options file. The license file is usually called FLEXwrap.lic or license.dat and is read by the License Server and, under License Server failsafe mode, by the Wrapper.
License file generator	The FLEXwrap program that allows the FLEXwrap license administrator at your site to generate FEATURE lines for wrapped applications. swsetup32.exe or swcrypter.exe This program should be protected from execution by your general user community. Otherwise, your users would be able to regenerate your license files, possibly compromising the integrity of your license management system.
License key	The 12- or 20-digit authentication code on a FEATURE line in a license file. Also called encryption code.
License server	The combination of one running lmgrd process and one running swrap vendor daemon process that allows wrapped applications to request and obtain licenses.
License server failsafe mode	A mode of operation that allows the Wrapper to run the wrapped application when the FLEXwrap License Server is down or cannot be contacted. This mode is enabled by specifying failsafe mode when the license is generated and making sure the license file is accessible to the Wrapper even when the License Server is down. See Setting Up Fail Safe Modes for more information.
License server machine	A computer system that is running both lmgrd and the swrap daemon. A copy of the license file will be on this machine, as will the FLEXwrap report log if turned on.
lmgrd	The license manager daemon process that starts the swrap vendor daemon and directs the Wrapper to the swrap daemon: lmgrd.exe
swrap	The FLEXwrap vendor daemon process that serves licenses to wrapped applications and writes a report log that FLEXnet Manager uses to generate reports: swrap.exe
Suite	A group of products sold by a software vendor where, according to the license agreement, each individual product cannot be used independently of the product suite.

Table 23-17 • FLEXwrap Terminology (cont.)

Term	Description

Wrapper	The FLEXwrap binary which is run when an end user attempts to run a wrapped application:
	`swrap_nt.exe (GUI apps)`
	or
	`swrap_ntc.exe (console apps)`
	The Wrapper requests a license from the License Server and runs the wrapped application if a license is obtained or if license failsafe mode or License Server failsafe mode is enabled.

About FLEXwrap License Parameters

This section lists the field name and explanation of each license parameter, and includes an example of a license file using each of these parameters. It also includes a list of all of the possible elements of a FLEXwrap FEATURE line.

- Specifying FLEXwrap License Parameters
- Understanding FLEXwrap License File FEATURE Lines

Specifying FLEXwrap License Parameters

To generate FLEXwrap licenses, you use the swsetup32 program and specify parameters using a FEATURE line. The following table lists the field name and explanation of each license parameter.

Table 23-18 • FLEXwrap License Parameters

Parameter	Description
Application Program	This parameter must be identical (including case and excluding .exe) to the name of the binary to be wrapped. It must conform to FLEXwrap feature-naming rules, including: • The name must be <= 30 characters. • The name must start with either a letter, a number, or an underscore "_". This will appear after the word FEATURE on the FEATURE line.

Table 23-18 • FLEXwrap License Parameters (cont.)

Parameter	Description
License Limit	This parameter is the license limit—the number of licenses that the License Server will allow to be checked out simultaneously, not including overdraft. To strictly enforce the license limit for this feature, enter the number of licenses that you are licensed to run. If you do not wish to restrict licensing, but you want complete reports on all usage of the wrapped application, enter the largest number of instances of this binary that you think will be run simultaneously. If you are just testing FLEXwrap's functionality, you might want to enter 2 here. The license count will appear after the expiration date on the FEATURE line. For uncounted licenses, enter `0` or `uncounted` for the license limit.
Overdraft	This parameter is the number of licenses above the license limit that the License Server will allow to be checked out. Use this parameter if you want usage reports on usage and do not want to enforce license limits. Reports will show the license limit as the maximum number of licenses available, and any usage exceeding this limit will be overdraft usage.

Program Suite	Some vendors sell their products as a suite. FLEXwrap defines a suite as a group of independent programs which must, for licensing purposes, be treated as a unit.
	
	For example, if a product suite, Office, consists of programs Word, Paint, and Mail, and if you have only one license for the suite, then while one user was using Word, that same user could use products Paint or Mail, but a different user could not use any products—even Paint or Mail, which are not currently in use.
	
	Typically, to create the licenses for a product suite, you would create licenses with the Group by (duplicate license request) parameter set so that multiple requests from the same user (or the same user on the same host) count as one license request.
	
	When you create a suite license with FLEXwrap, the Wrapper checks out both the individual program license (Word in the example above) and a suite license (Office in the example). Since a second license request for Office does not require an additional license, when the same user runs program Paint, he is able to check out the Office suite license, and is also allowed to check out the Paint license. If another user attempts to check out a license for Paint, she would be unable to get the Office suite license and her request would be denied (or queued).
	
	An application is made part of a suite by using the optional PREREQ attribute of a FLEXwrap license FEATURE line. PREREQ is set to a FEATURE that must also be checked out before the wrapped application will be executed.
	
	To create a suite you must add a PREREQ line to all FEATUREs that will be part of the suite. You must then create the suite's FEATURE in the license file. A full suite FEATURE looks like:
	
	`FEATURE suite name swrap 1.0 expiration count 0 \`
	`  DUP_GROUP=[UHDS]`
	
	This is identical to an application's FEATURE line except that only the DUP_GROUP optional attribute is pertinent.

Table 23-18 • FLEXwrap License Parameters (cont.)

Parameter	Description
Expiration Date	The date the license for this FEATURE will expire. Licenses can no longer be obtained after this date.
Node-lock Hostid	This is the hostid of the only machine(s) on which the wrapped application can run. This field can contain multiple hostids separated by spaces and surrounded by double quotes.
Application File to Run	Unless you specify otherwise, when the Wrapper obtains a license, it will run `binary.orig` (or `binary.fwr`) in the directory from which the Wrapper ran. If you want it to run a different binary, specify the full path to the binary name here. This will become part of the FEATURE line as `w_binary=full_path_to_binary`.
Queueing Behavior	• **Queue/Always**—Queue for a license if one cannot be obtained. When a license becomes available, the binary will run. This is for use when failsafe is not on. This is recommended for batch jobs. (`w_queue=1`) • **Query/Ask**—If a license cannot be obtained, shows who is using the licenses and asks the user if they want to queue for a license. If the user answers **Y**, the Wrapper will wait until a license is available. When one becomes available, the binary will run. (`w_queue=2`) • **Exit/Never**—No queueing.
Duplicate Grouping	FLEXwrap allows you to specify that simultaneous uses of the wrapped application by the same user (or host, or user/host combination, etc.) will not require a license. The default is that each process requires its own license. To accept this default, in `swsetup32`, choose **None**. You can group on: • **User**—All requests from the same user use only one license. • **Host**—All requests from the same host use only one license. • **User + Host**—All requests from the same user on the same host use one license. • **Site**—All requests use one license.
Regular (License) Failsafe Mode	If you wish to strictly enforce your license limits, check in `swsetup32`. If a license cannot be obtained, the binary will not run. This will not appear on the FEATURE line. If you want the binary to run even if it cannot obtain a license, uncheck in `swsetup32`. This will appear as `VENDOR_STRING=failsafe` on the FEATURE line. ***Note*** • *For License Server failsafe mode, where the application will never fail to run because it cannot connect to the FLEXwrap License Server, the Wrapper must be able to see the license file when the License Server is down. See Setting Up License Server Failsafe Mode for instructions.*

Table 23-18 • FLEXwrap License Parameters (cont.)

Parameter	Description
Generate 20-Digit License Keys	The default length of the license key on a FEATURE line generated by FLEXwrap v8.2 is 12 digits. • If you run swsetup32 on Windows or run swcrypter from the command line, you can generate a 20-digit license key with embedded start date for backward compatibility. • A FLEXwrap v8.2 license file can contain a mixture of 12- and 20-digit license keys. • To generate 20-digit keys, set the license format to FLEXlm v5 in swsetup32 or swcrypter.
Shutdown Signal	Some applications require a special shutdown signal in order to perform application-specific cleanup. FLEXwrap will normally send a SIGHUP to the application, but if the application requires a different signal, you can specify it in this parameter. For example, vi will leave the terminal in "raw" mode unless it receives a SIGHUP.

Connection Timeout	If the License Server becomes unavailable (due to a License Server node failure, a License Server process failure, or a network communication problem), the application will lose its license.
	The Wrapper will attempt to regain its license until the timeout (or exit) period elapses. By default, there is no timeout (or exit) period, so the application will continue to run without a license and will continue to attempt to regain its license indefinitely.
	If you specify a timeout period, the wrapped application will continue to run for the specified amount of time while the Wrapper attempts to regain its license. The Wrapper will give periodic warnings to the user. For example, if you set the timeout to ten minutes, the user will get ten messages like this:
	`Lost license for feature.`
	`  Reconnection attempt n of 10.`
	If a license can be obtained before the timeout period ends, the user will see `Reacquired feature license`. If the timeout period is reached and a license has not been obtained, the application will exit with the message `Lost license. Exiting`.

Table 23-18 • FLEXwrap License Parameters (cont.)

Parameter	Description
Runtime Name	Some application programs require that their `argv[0]` is different from the last part of the application file name. To accommodate these rare cases, FLEXwrap allows you to specify what value to pass as `argv[0]` to the application. For example, the `vi` program will only operate in what would be considered a "normal" manner if its `argv[0]` is `vi`. If you were to rename the binary for `vi` to `/usr/ucb/vi_real`, for example, then you would want to specify `vi` as the `argv[0]` parameter. Another reason to use the `argv[0]` parameter would be for programs that do an internal restart by calling the `exec()` function. When the function which causes the product to call `exec()` is executed, the `exec()` call causes a second copy of the FLEXwrap Wrapper to run, thereby checking out a second license. By setting `argv[0]` to `program.orig` (and using the standard renaming convention of renaming program to `program.orig`), then the `exec()` call executes the real program rather than the Wrapper, and the second license is not checked out. ***Note*** • *If the user does not have the program in his path, then the* `argv[0]` *parameter will need to be the full path name of the wrapped program. This will not work in the case where the Wrapper is used to wrap multiple architecture binaries (since they will be in different directories and only one directory can be specified to* `argv[0]` *in this manner). In this case, the user will need to place the binary directory into his shell path and set* `argv[0]` *to the program name, without the path. There are very, very few cases in Windows when this setting is needed.*

Note • *See Understanding FLEXwrap License File FEATURE Lines for additional explanation of these parameters.*

A `FEATURE` line with all possible attributes enabled would look like this:

```
FEATURE feature_name swrap 1.0 expiration_date count 0 \
    w_binary=full_path_to_binary \
    w_argv=argv \
    PREREQ=suite_name \
    W_LIC_LOSS=#minutes \
    VENDOR_STRING=failsafe \
    HOSTID=node_lock_hostid \
    DUP_GROUP=[U, H, UH, S] \
    w_queue=[not set, 1, 2] \
    w_term_signal=signal#
```

Understanding FLEXwrap License File FEATURE Lines

The simplest FEATURE line in a FLEXwrap license file will look something like this:

```
FEATURE testvi2 swrap 1.000 1-jan-0000 2 BB8C401190F6
```

The most complex FEATURE line, in which all the options have been exercised, will look something like this:

```
FEATURE featurea swrap 1.000 15-dec-2001 2 6B7C60F1803D \
    VENDOR_STRING=failsafe HOSTID=1234 \
    DUP_GROUP=U w_binary=/t/dodi/real w_argv=real w_queue=1 \
    w_term_signal=2 W_LIC_LOSS=5 PREREQ=suite
```

Here are all the possible elements of a FLEXwrap FEATURE line. Keywords in upper case letters are part of the license encryption, so they cannot be manually edited without corrupting the FEATURE line. Those in lower case are not part of the encryption, so they can be changed manually without regenerating the license.

Table 23-19 • FLEXwrap FEATURE Line Elements

Element	Possible Values	Explanation
FEATURE	FEATURE	
feature_name	Same as wrapped application's original binary name (including case).	
daemon_name	swrap	
version	1.000	
expiration_date	Any valid date in the format dd-mmm-yy[yy]. • Any year shown as two digits is assumed to be 19xx. For example: 31-dec-99 is December 31, 1999. • Any year after 1999 must be shown with four digits. For example: 1-jan-2000. • Unexpiring can be shown as: 1-jan-0 or 1-jan-00 or 1-jan-0000, permanent.	
count	Any positive integer. For uncounted licenses, the count is set to 0 or uncounted.	The number of licenses that can be checked out simultaneously without going into overdraft. This number will appear as the license limit on reports.

Table 23-19 • FLEXwrap FEATURE Line Elements (cont.)

Element	Possible Values	Explanation
license_key	12- or 20-digit hexadecimal number.	FLEXwrap license authorization key generated by the FLEXwrap license file generator. 12-digit license keys are generated by default. A FLEXwrap v8.2 license file can contain a mixture of 12- and 20-digit license keys. For backward compatibility, 20-digit license keys containing an embedded start date can be generated using `swsetup32` on Windows or by using an option to `swcrypter`: `swcrypter license_file -verfmt 5`
VENDOR_STRING=	failsafe	In regular (license) failsafe mode, the wrapped application specified on the `FEATURE` line can run without obtaining a license if the Wrapper can connect to the License Server. To set up License Server failsafe mode in which the wrapped application can run even if the License Server is not running, see Setting Up License Server Failsafe Mode
HOSTID=	node_lock_hostid	The only machine(s) where the wrapped application can run. Multiple hostids are separated by spaces and have double quotes around the string.
DUP_GROUP=	• **U**—User • **H**—Host • **D**—Display • **UH**—User and Host • **UD**—User and D • **UHD**—User, Host and D • **HD**—Host and D • **S**—Site	Additional simultaneous instances by the same (U)ser, (H)ost, (U)ser & (H)ost, (S)ite, etc. will not be counted against the license limit.

Table 23-19 • FLEXwrap FEATURE Line Elements (cont.)

Element	Possible Values	Explanation
w_binary=	full_path_to_binary	Binary to run when license has been obtained.
w_argv=	argv	When wrapped, the application binary's name is changed to `binary.orig`. Some applications do not work properly unless their name is binary. In those cases, specify binary as the `argv`.
w_queue=	1 2	If license cannot be obtained, and failsafe is not set: **1** = Queue automatically **2** = Ask if user wishes to queue
w_term_signal	signal#	Signal sent to wrapped application when the Wrapper program is killed. (default=1)
W_LIC_LOSS=	#minutes	This is the timeout value. If a timeout is specified, and if the Wrapper loses its license, the Wrapper will warn the user and exit after this number of minutes. The minimum connection timeout is five minutes.
PREREQ=	suite_name	Feature name of suite license that must also be obtained in order for this license to be obtained.

Dialog Boxes and Wizards

The FLEXwrap Server Configuration Tool uses the following dialog boxes and Wizards:

- FLEXwrap Server Configuration Dialog Box
- FLEXwrap Application Information Dialog Box
- Wrapping Wizard
- Encryption Wizard
- Wrapping and Encryption Wizard

- FLEXwrap Command Dialog Box
- Invalid Certificate Dialog Box

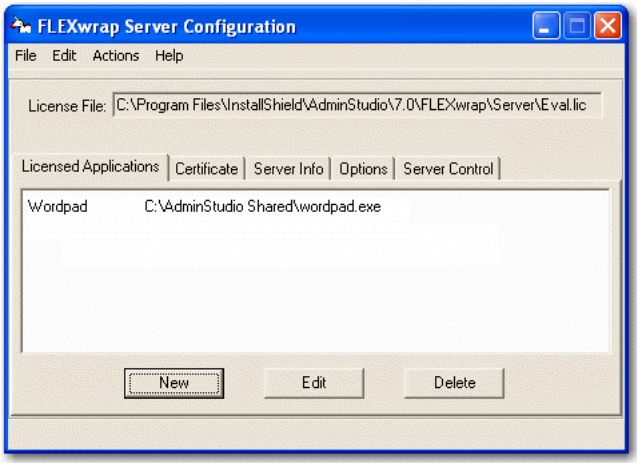
FLEXwrap Server Configuration Dialog Box

The **FLEXwrap Server Configuration** dialog box includes the following tabs:

- Licensed Applications Tab
- Certificate Tab
- Server Info Tab
- Options Tab
- Server Control Tab

Licensed Applications Tab

The **Licensed Applications** tab lists all of the licensed applications in the currently selected License File.



The following options are included:

Table 23-20 • Server Configuration Tool Dialog Box—Licensed Applications Tab

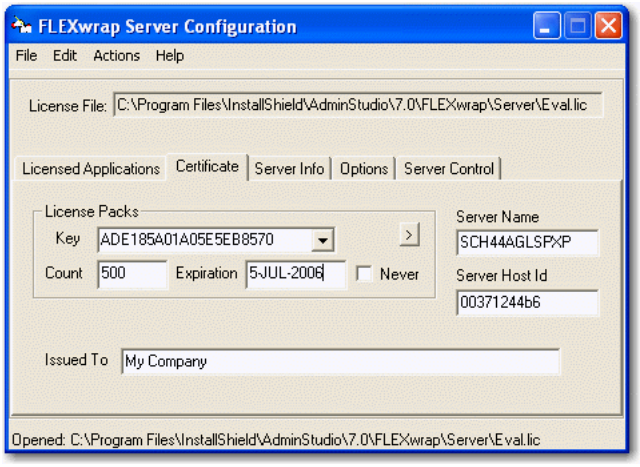
Option	Description
License File	The location of the selected License File. To open a License File, click **Open** on the **File** menu.
Licensed Applications	A list of all of the licensed applications in the currently selected License File.
New	Click to open the FLEXwrap Application Information Dialog Box where you select an application executable file to FLEX-enable.

Table 23-20 • Server Configuration Tool Dialog Box—Licensed Applications Tab

Option	Description
Edit	Select a listed application and click Edit to edit that application's settings on the FLEXwrap Application Information Dialog Box.
Delete	Click to delete a licensed application from the list.

Certificate Tab

On the **Certificate** tab, you enter the production FLEXwrap license information you received from AdminStudio Support for an application.



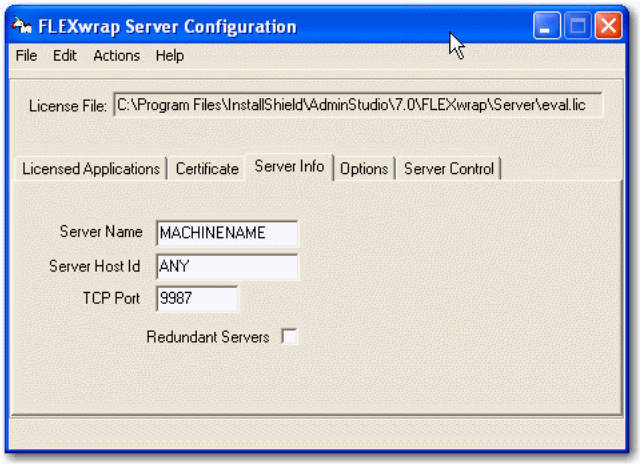
The following options are included:

Table 23-21 • Server Configuration Tool Dialog Box—Certificate Tab

Option	Description
Key	License key provided by AdminStudio Support.
Count	The number of concurrent licenses available for this FLEX-enabled application.
Expiration	The date these licenses will expire.
Never	Select if you do not want these license to expire.
Server Name	Name of the FLEXwrap License Server machine.
Server Hostid	The host name of the FLEXwrap License Server listed in the production license file.
Issued To	The name of the company that purchased the license.

Server Info Tab

This tab lists the identification information for the FLEXwrap License Server.



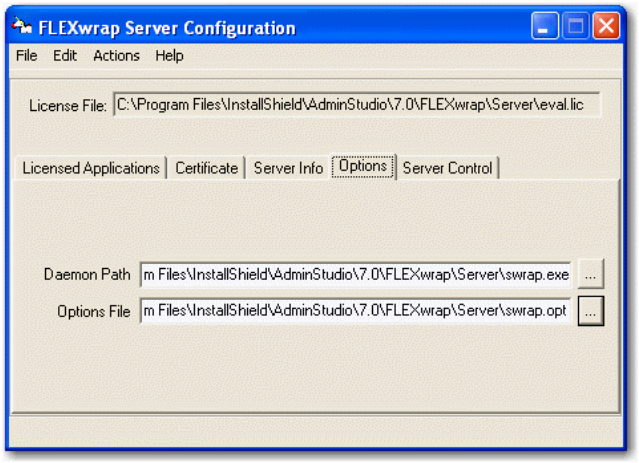
The following options are included:

Table 23-22 • Server Configuration Tool Dialog Box—Server Info Tab

Option	Description
Server Name	Name of the FLEXwrap License Server
Server Host Id	Hostid of the FLEXwrap License Server.
TCP Port	TCP Port of the FLEXwrap License Server.
Redundant Servers	Select this option if there are any redundant (back-up) servers associated with the specified server.

Options Tab

On the Options tab, you can set the path for the Daemon and the Options File.



The following options are included:

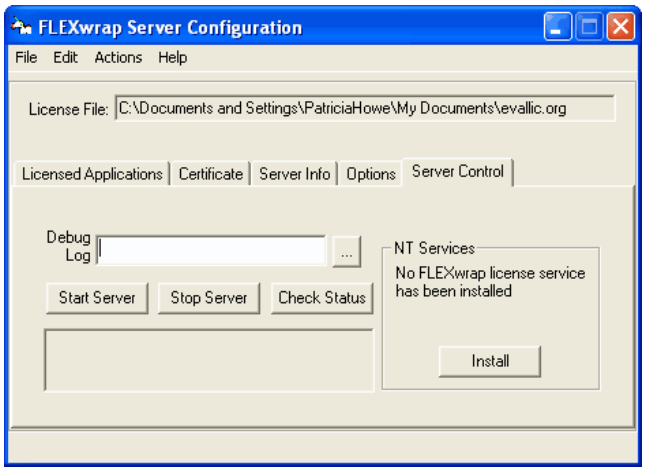
Table 23-23 • Server Configuration Tool Dialog Box—Options Tab

Option	Description
Daemon Path	If your `swrap` daemon is located in the same folder as the FLEXwrap Server Configuration tool (`lmgrd.exe`), you can leave the **Daemon Path** field blank. Otherwise, enter the full path to the `swrap` daemon.
Options File	If you want to record license usage in a report log, create a FLEXwrap options file. You could then generate usage reports using FLEXnet Manager. Create a plain text file called `swrap.opt` and save it in the same directory with your license file. In the **Options File** field, enter the full path to this options file. See Creating an Options File for more information.

Server Control Tab

On the Server Control tab, you can:

- Configure the FLEXwrap License Server to run as a service.
- Manually start the FLEXwrap License Server.
- Stop the FLEXwrap License Server.
- Check the status of the FLEXwrap License Server.



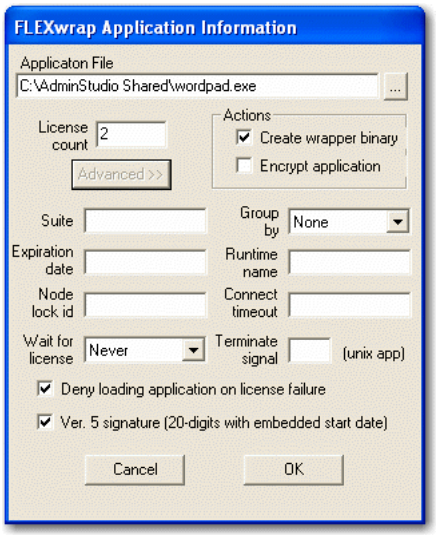
The following options are included:

Table 23-24 • Server Configuration Tool Dialog Box—Server Control Tab

Option	Description
Debug Log	To manually start the FLEXwrap License Server, you first need to enter the path to a debug log or select an existing debug log. If you do not specify a debug log when you start the License Server, at least two console windows will appear—one for the `lmgrd` output, one for the `swrap` output, and one for each of any other vendor daemons started by this **lmgrd**.
Start Server	To start the FLEXwrap License Server, click the **Start Server** button. All licenses (`*.lic`) in the directory of the opened license are loaded.
Stop Server	To stop the FLEXwrap License Server, click the **Stop Server** button.
Check Status	To check the status of the FLEXwrap License Server, click the **Check Status** button A Status Report opens.
Install	To configure the FLEXwrap License Server to run as a service, click the **Install** button in the **NT Services** area. The selected license file is now configured to run as a service.

FLEXwrap Application Information Dialog Box

On the FLEXwrap Application Information dialog box, you select the application binary that you are going to wrap, enter the license limit for the application, and set other advanced options.



The following options are included:

Table 23-25 • FLEXwrap Application Information Dialog Box Options—Advanced View

Option	Description
Application File	Select the application binary that you are going to wrap. Unless you specify otherwise, when the Wrapper obtains a license, it will run `binary.orig` (or `binary.fwr`) in the directory from which the Wrapper ran. If you want it to run a different binary, specify the full path to the binary name here. This will become part of the `FEATURE` line as `w_binary=full_path_to_binary`.
Create wrapper binary	Make sure this option is selected.
Encrypt application	In some cases, the application binary can be encrypted so that it can be run only by the Wrapper. This would prevent users from running the wrapped application binary (`binary.fwr`) directly. If you want to encrypt the application binary, select the this option.

Table 23-25 • FLEXwrap Application Information Dialog Box Options—Advanced View (cont.)

Option	Description
License Count	Enter the license limit for the application—the number of licenses that the License Server will allow to be checked out simultaneously, not including overdraft. • To **strictly enforce the license limit** for this feature, enter the number of licenses that you are licensed to run. • If you **do not wish to restrict licensing**, but you want complete reports on all usage of the wrapped application, enter the largest number of instances of this binary that you think will be run simultaneously. • For uncounted licenses, enter `0` or `uncounted` for the license limit. • If you are just testing FLEXwrap's functionality, you might want to enter 2 here. • The license count will appear after the expiration date on the `FEATURE` line. • The license limit will appear in reports as the maximum number of licenses available. • The total of the counts of licenses on all the `FEATURE` lines you generate may not exceed the count on the FLEXwrap `INCREMENT` line in your license file.
Advanced	Click to display advanced options.

Suite	If you want to treat the selected executable files as a suite, enter a suite name in this field.
	A suite is a group of independent programs which must, for licensing purposes, be treated as a unit. Only one license is required for one person to use all of the applications in the suite.
	An application is made part of a suite by using the optional `PREREQ` attribute of a FLEXwrap license `FEATURE` line. `PREREQ` is set to a `FEATURE` that must also be checked out before the wrapped application will be executed.
	To create a suite you must add a `PREREQ` line to all `FEATURE`s that will be part of the suite. You must then create the suite's `FEATURE` in the license file. A full suite `FEATURE` looks like:
	`FEATURE suite name swrap 1.0 expiration count 0 \`
	`  DUP_GROUP=[UHDS]`
	This is identical to an application's `FEATURE` line except that only the `DUP_GROUP` optional attribute is pertinent.

Expiration date	The date the license for this `FEATURE` will expire. Licenses can no longer be obtained after this date.
Node lock id	This is the hostid of the only machine(s) on which the wrapped application can run. This field can contain multiple hostids separated by spaces and surrounded by double quotes.

Table 23-25 • FLEXwrap Application Information Dialog Box Options—Advanced View (cont.)

Option	Description
Wait for license	Select one of the following options to define the queuing behavior of this wrapped application: • **Always**—Queue for a license if one cannot be obtained. When a license becomes available, the binary will run. This is for use when failsafe is not on. This is recommended for batch jobs. (`w_queue=1`) • **Ask**—If a license cannot be obtained, shows who is using the licenses and asks the user if they want to queue for a license. If the user answers **Y**, the Wrapper will wait until a license is available. When one becomes available, the binary will run. (`w_queue=2`) • **Never**—No queueing.
Group by	FLEXwrap allows you to specify that simultaneous uses of the wrapped application by the same user (or host, or user/host combination, etc.) will not require a license. The default is that each process requires its own license. To accept this default, in `swsetup32`, choose **None**. You can group on: • **User**—All requests from the same user use only one license. • **Host**—All requests from the same host use only one license. • **User + Host**—All requests from the same user on the same host use one license. • **Site**—All requests use one license.
Runtime name	Some application programs require that their `argv[0]` is different from the last part of the application file name. To accommodate these rare cases, FLEXwrap allows you to specify what value to pass as `argv[0]` to the application by entering a name in this box. For more information on specifying a **Runtime name**, see Runtime Name.

Table 23-25 • FLEXwrap Application Information Dialog Box Options—Advanced View (cont.)

Option	Description

Connect timeout	If the License Server becomes unavailable (due to a License Server node failure, a License Server process failure, or a network communication problem), the application will lose its license.
	
	The Wrapper will attempt to regain its license until the timeout (or exit) period elapses. By default, there is no timeout (or exit) period, so the application will continue to run without a license and will continue to attempt to regain its license indefinitely.
	
	If you specify a **Connect timeout** period, the wrapped application will continue to run for the specified amount of time while the Wrapper attempts to regain its license. The Wrapper will give periodic warnings to the user. For example, if you set the timeout to ten minutes, the user will get ten messages like this:
	
	`Lost license for feature.`
	`  Reconnection attempt n of 10.`
	
	If a license can be obtained before the timeout period ends, the user will see `Reacquired feature license`. If the timeout period is reached and a license has not been obtained, the application will exit with the message `Lost license. Exiting`.

Terminate signal	Some applications require a special shutdown signal in order to perform application-specific cleanup. FLEXwrap will normally send a `SIGHUP` to the application, but if the application requires a different signal, you can specify it in this parameter. For example, `vi` will leave the terminal in “raw” mode unless it receives a `SIGHUP`.
Deny loading application on license failure	By default, the **Deny loading application on license failure** option is selected. • **Run when no license is available**—If you want the wrapped application to be able to run whether or not it can get a license from a running FLEXwrap License Server (License Failsafe Mode), clear the **Deny loading application on license failure** option. See Setting Up Regular (License) Failsafe Mode. • **Run when cannot connect to License Server**—If you want the wrapped application to be able to run whether or not the FLEXwrap License Server is running (License Server Failsafe Mode), clear the **Deny loading application on license failure option** and follow the instructions in Setting Up License Server Failsafe Mode. • **Cannot run without a license**—If you wish to require the Wrapper to obtain a license in order to run the wrapped application, select the **Deny loading application on license failure** option

Table 23-25 • FLEXwrap Application Information Dialog Box Options—Advanced View (cont.)

Option	Description
Ver. 5 signature (20-digits with embedded start date)	Select this option if you want to generate 20-digit license keys with an embedded start date. • The default length of the license key on a `FEATURE` line generated by FLEXwrap v8.2 is 12 digits. • A FLEXwrap v8.2 license file can contain a mixture of 12- and 20-digit license keys. • If you run `swsetup32` on Windows or run `swcrypter` from the command line, you can generate a 20-digit license key with embedded start date for backward compatibility.

Wrapping Wizard

The Wrapping Wizard includes the following panels:

- Select File to Wrap Panel
- Identify the License Server and License File Panel
- Identify Location of Original Binary Panel
- Ready to Wrap Panel

Select File to Wrap Panel

Select the application executable (`.exe`) that you wish to wrap and click **Next**.

Identify the License Server and License File Panel

On this panel, you are prompted to identify the path to the License Server. You can choose to enter a host and port number or a path to the location of the license file.

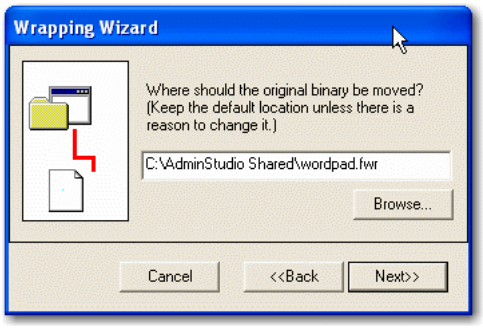
The following options are included:

Table 23-26 • Wrapping Wizard—License File Location Panel

Option	Description
Host and port number reference	If you select this option, enter the host and port number of the license server.
Path to the license file	If you select this option, enter the appropriate path to the license server. This information is written into `swrapper.lf`. You would normally want to select **port@host**.

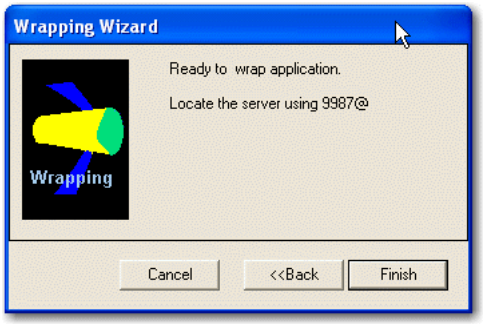
Identify Location of Original Binary Panel

On this panel, you are prompted to identify the location where you want FLEXwrap to save a copy of the original binary prior to wrapping.



Ready to Wrap Panel

To begin wrapping the selected package, click **Finish**.



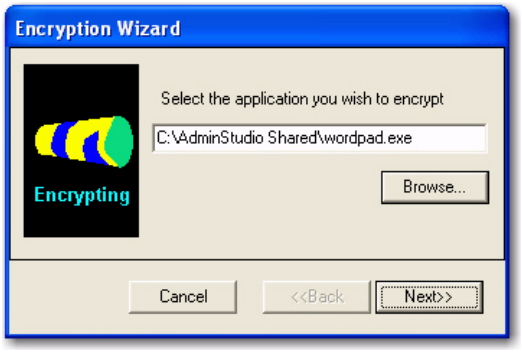
Encryption Wizard

The Encryption Wizard includes the following panels:

- Select Application Panel
- Location for Original Binary Panel
- Ready to Encrypt Panel

Select Application Panel

Select the application executable (`.exe`) that you wish to encrypt and click **Next**.



Location for Original Binary Panel

On this panel, you are prompted to identify the location where you want FLEXwrap to save a copy of the original binary prior to encryption.

Ready to Encrypt Panel

To begin encryption of the selected package, click **Finish**.

Wrapping and Encryption Wizard

The Wrapping and Encryption Wizard includes the following panels:

- Backup Copy Panel
- License File Location Panel
- Original Binary Location Panel
- Ready to Encrypt and Wrap Panel

Backup Copy Panel

On this panel you specify where you would like a backup copy of the application binary to be saved. The default is `binary.bak` in the application directory

License File Location Panel

On the License File Location Panel, you are prompted to identify the path to the License Server. You can choose to enter a host and port number or a path to the location of the license file.

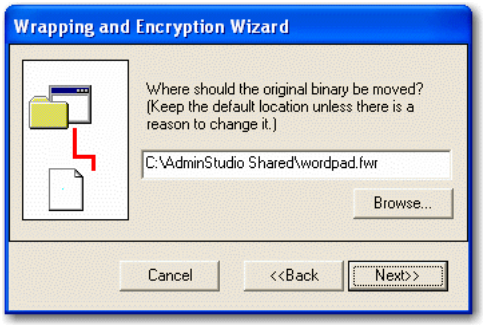
The following options are included:

Table 23-27 • Wrapping and Encryption Wizard—License File Location Panel

Option	Description
Host and port number reference	If you select this option, enter the host and port number of the license server.
Path to the license file	If you select this option, enter the appropriate path to the license server. This information is written into `swrapper.lf`. You would normally want to select **port@host**.

Original Binary Location Panel

On this panel you are prompted to identify the location where you want FLEXwrap will save a copy of the original binary prior to encryption.



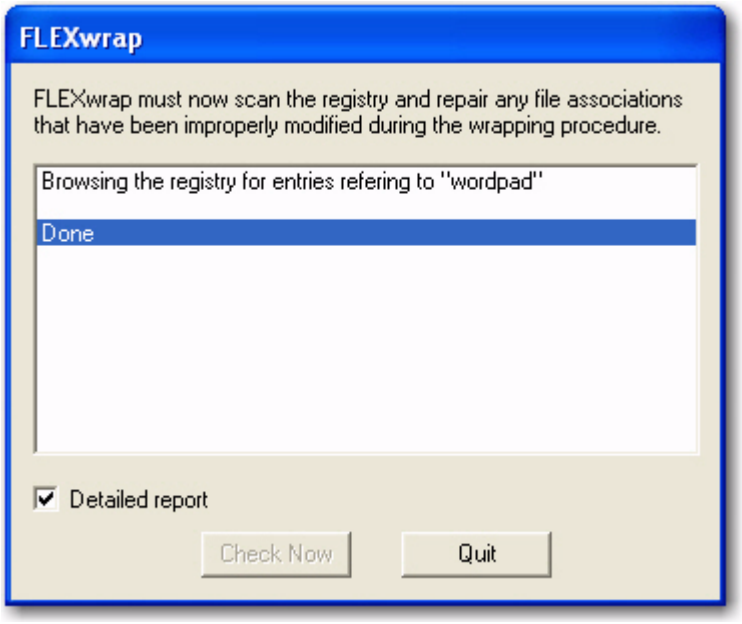
Ready to Encrypt and Wrap Panel

To begin encryption and wrapping of the selected package, click **Finish**.

FLEXwrap Command Dialog Box

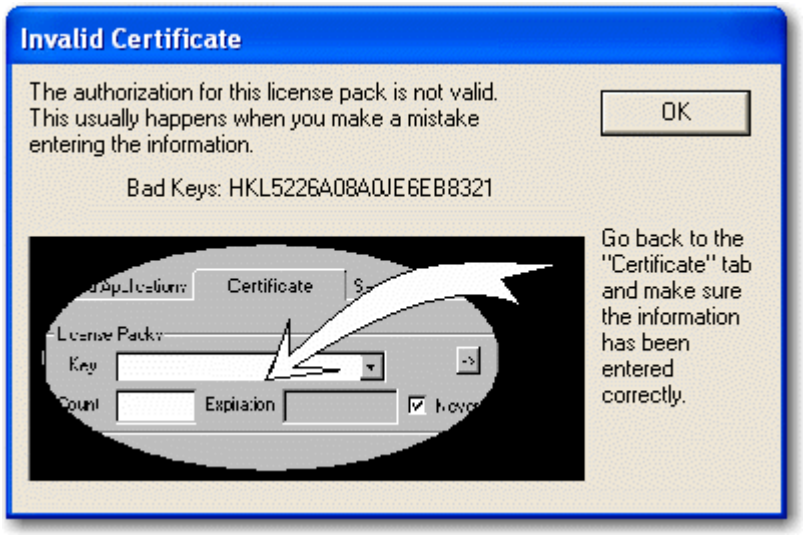
When the Wrapping Wizard is finished, this dialog box opens, informing you that FLEXwrap must now scan the Registry and repair any file associations that were improperly modified during the wrapping procedure.

Click **Check No** to scan/repair the registry. To get a detailed report of the Registry scan, select the **Detailed report** option.



Invalid Certificate Dialog Box

This dialog box opens when an invalid license key was entered in the License Packs **Key** field on the Certificate Tab of the FLEXwrap Server Configuration Dialog Box. You should check to see whether the **Key** was entered correctly.



You purchase production license file credentials from AdminStudio Support, and then create the license file by entering the license file credentials on the Certificate Tab.

Part 6

Using AdminStudio Enterprise Server Tools

This part of the User Guide includes the following chapters:

- Generating and Viewing Reports in Report Center
- Automating Tasks Using Job Manager

24

Generating and Viewing Reports in Report Center

Edition • *Report Center is included with AdminStudio Enterprise Edition*

Report Center provides a centralized view of all of the information regarding packages in your Application Catalog. You can use Report Center to generate reports on packages stored in the Application Catalog. You can also use SQL queries to generate custom reports on data stored in the Application Catalog.

- **Package Report**—Includes consolidated package information including conflicts, validation results, and extended attributes. See Viewing Package Reports.
- **Custom SQL Query Report**—A custom report defined by entering an SQL query in the Report Wizard. See Generating a Custom SQL Query Report for AdminStudio.

Generating and Viewing AdminStudio Reports

You can use Report Center to obtain a centralized view of all of the information regarding packages in your AdminStudio Application Catalog. Because Report Center is a Web application, it can be easily accessed by a geographically dispersed workforce without requiring any software installation or data transfer. Report Center makes it easy to get the application data you need to diagnose and repair software problems and to manage applications across your organization.

A catalog-level search tool enables you to generate detailed, customizable reports on packages with particular characteristics. These reports are accessible anywhere via a Web interface and can be exported to PDF or Excel format for sharing and archiving.

Information on generating and viewing AdminStudio reports in Report Center is presented in the following sections:

Table 24-1 • Information About Generating AdminStudio Reports

Section	Description
Viewing Package Reports	Explains how to generate a Package Report on a selected package. Also explains how to filter the package tree by specified criteria in order to find a specific package in the Application Catalog. This section also lists the contents of all of the sections of a Package Report.
Archiving a Package Report	Explains how to create a PDF-snapshot of a Package Report at a particular date and time and how to open an archived report.
Exporting a Package Report	Explains how to export a Package Report to Excel or PDF format.
Generating a Custom SQL Query Report for AdminStudio	Explains how to enter an SQL query to specify the data to be displayed in a custom report.

Viewing Package Reports

You can generate AdminStudio Package Reports on the **Search Packages** subtab of the Report Center. On this tab, you can perform a search of all of the applications in the Application Catalog to locate the package you would like to generate a report for.

- Searching for a Package on the Search Packages Page
- Information Included in Package Reports
- Navigating Through a Package Report
- Archiving a Package Report
- Exporting a Package Report

Searching for a Package on the Search Packages Page

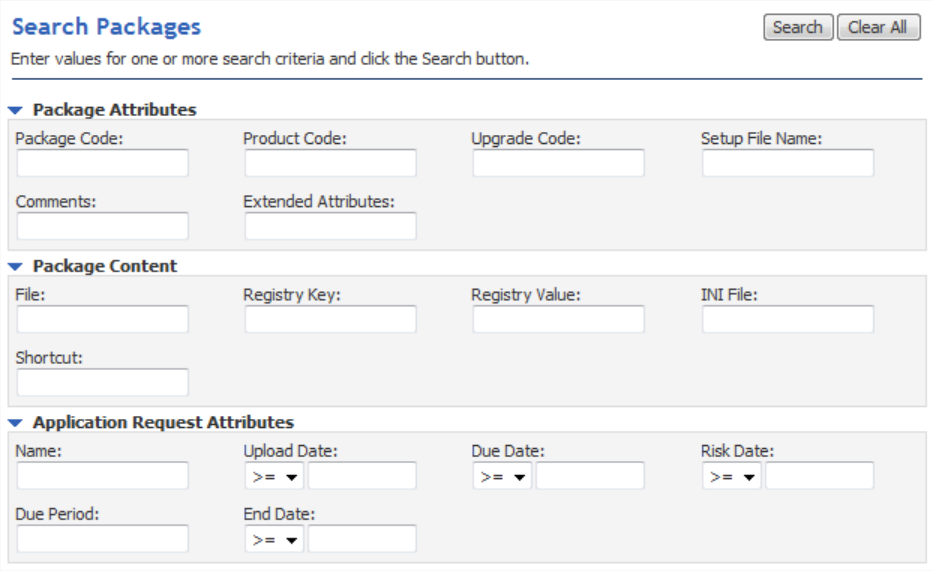
On this **Search Packages** page, you can filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria, which are grouped into three categories:

- **Package Attributes**—Search by properties assigned to the Windows Installer package. See Package Attributes.
- **Package Content**—Search by files, registry entries, `.ini` files, or shortcuts contained in the Windows Installer package. See Package Content.
- **Application Request Attributes**—Search by information related to a package's associated Application Request. See Application Request Attributes.

To filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria, perform the following steps.

Task: ***To search for a package on the Search Packages page:***

1. In the **Search Packages** area of the **Search Packages** page, expand the criteria category that you want to use by clicking the arrow. When all three categories are expanded, the following fields are available:



2. Enter values in the criteria fields that you want to search on. You can search for packages in the Application Catalog based on metadata in three categories:

   - **Package Attributes**—Search by properties assigned to the Windows Installer package. See Package Attributes.
   - **Package Content**—Search by files, registry entries, `.ini` files, or shortcuts contained in the Windows Installer package. See Package Content.
   - **Application Request Attributes**—Search by information related to a package's associated Application Request. See Application Request Attributes.

3. After you have entered the search criteria, click **Search**. The packages that meet the criteria are now listed in the package tree grouped by application.

Package Attributes

You can search for packages in a catalog based on one or more of any of the following Package attribute metadata:

Table 24-2 • Package Attribute Search Fields

Metadata	Description
Package Code	Enter the GUID that identifies a particular Windows Installer `.msi` package. The Package Code associates an `.msi` file with an application or product and is represented as a string GUID—a text string that has a special format: `{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}` where each X character is a hex digit (0 through 9 or uppercase A through F).
Product Code	Enter the GUID that uniquely identifies the particular product release of the package. The **ProductCode** is a Windows Installer property and is represented as a string GUID—a text string that has a special format: `{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}` where each X character is a hex digit (0 through 9 or uppercase A through F).
Upgrade Code	Enter the GUID that identifies the family of products that are in the same upgrade path. The **UpgradeCode** is a Windows Installer property and is represented as a string GUID—a text string that has a special format: `{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXsXXXXX}` where each X character is a hex digit (0 through 9 or uppercase A through F). ***Note*** • *Each stand-alone product usually has its own **UpgradeCode** GUID. Every version of XYZ Product typically uses the same GUID for the **UpgradeCode**. In other words, Product A Version 1.0 has the same UpgradeCode as Product A Version 2.0, but has a different UpgradeCode than Product B.*
Setup File Name	Name of the Windows Installer (.msi) file that was imported into the Application Catalog.
Comments	Enter the text of any comments associated with the package.
Extended Attributes	Enter the value of any of the Extended Attributes associated with the package.

Package Content

You can search for packages in a catalog based on one or more of any of the following Package Content metadata

Table 24-3 • Package Content Search Fields

Metadata	Description
File	Enter the file name of one of the files in the Windows Installer package.
Registry Key	Enter a registry key to search on.
Registry Value	Enter a registry value to search on.
INI File	Enter any changes to an `.ini` file that are made when the product is installed.
Shortcut	Enter the name of a shortcut that is created when the product is installed.

Application Request Attributes

You can search for packages in a catalog based on one or more of any of the following attributes of the package's associated Application Request:

Table 24-4 • Application Request Attributes Search Fields

Metadata	Description
Name	Enter the name of the package's associated Application Request.
Upload Date	Enter the date the Application Request was created.
Due Date	Enter the date the Application Request is scheduled to be completed, based upon its value for **Application Due Period**.
Risk Date	Enter the date at which the Application Request's status will change to **At Risk**, which is based upon its value for **Application At Risk Period**.
Due Period	Enter, in days, the length of time this Application Request needs to be completed in order to meet its Project's Service Level Agreement (SLA) requirements.
End Date	Enter the date the Application Request was completed.

Information Included in Package Reports

A Package Report presents information in a tabbed interface. See Navigating Through a Package Report.

A Package Report includes the following major sections:

- Package Summary Information View

- Files View
- Registry View
- Shortcuts View
- ODBC Drivers View
- ODBC DS View
- Extended Attributes View
- Validation View
- Conflicts View
- History View
- Dependencies View

Note • *Additional information may be available for App-V reports.*

Package Summary Information View

The initial view (Page 1) of a Package Report is the **Package Summary Information** view.

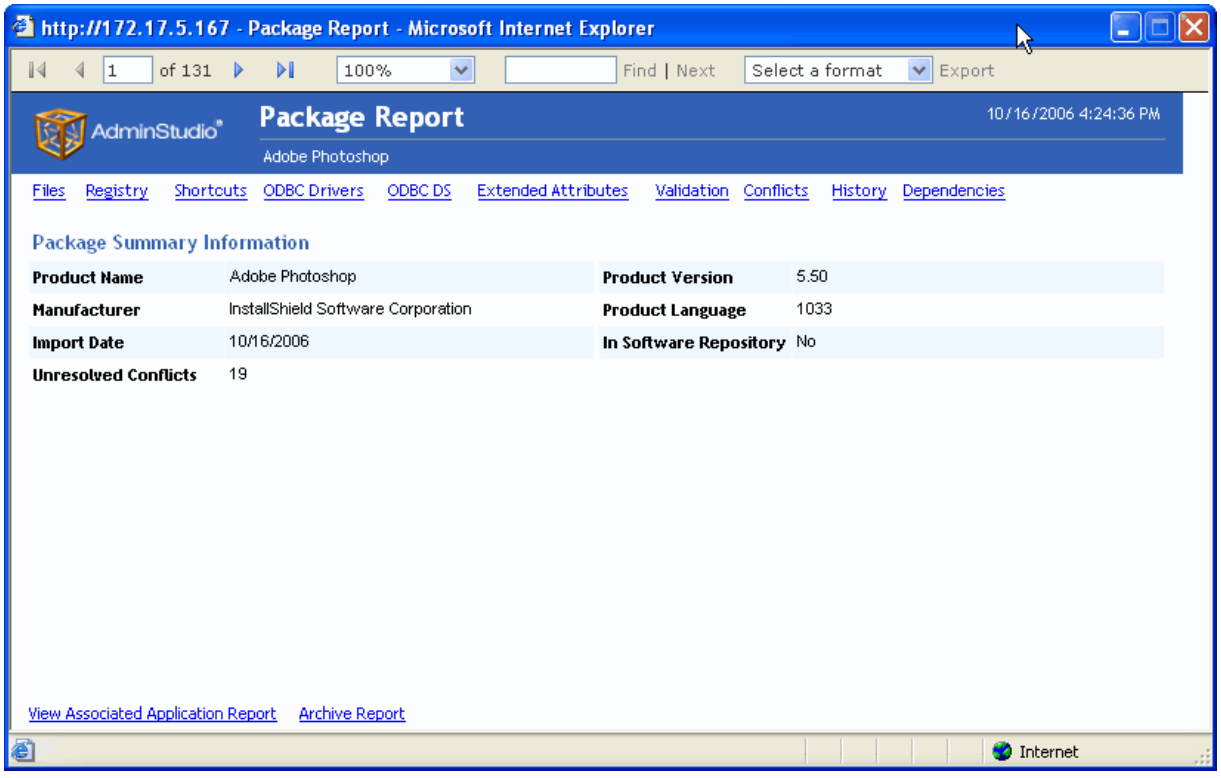


Figure 24-1: Package Report / Package Summary Information View

The Package Summary Information View lists the following information:

Table 24-5 • Package Report / Package Summary Information

Item	Description
Product Name	Name assigned to the package.
Manufacturer	Company that authored the package.
Import Date	The date and time the package was imported into the Application Catalog.
Unresolved Conflicts	The number of detected conflicts, generated during conflict analysis of this package, which have not yet been resolved—either automatically or manually.
Product Version	Version of package that is recorded in the package's Windows Installer file.
Product Language	Decimal-based code identifying the language that this software package was authored for. For example, English is 1033, German is 1031, and Japanese is 1041.
In Software Repository	Indicates whether or not this package and its associated files are managed by the Software Repository.

Files View

The **Files** view lists all of the files included in the selected package, and the location where these files will be installed.

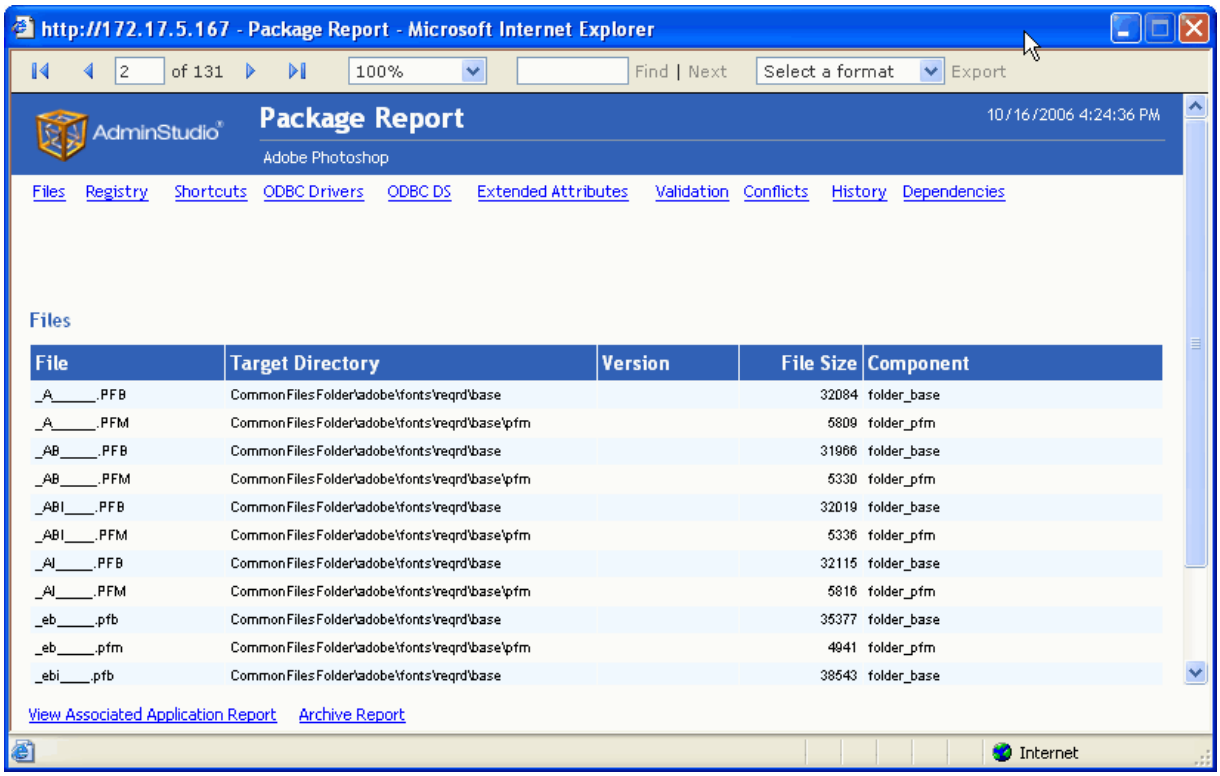


Figure 24-2: Package Report / Files View

For each file, the following information is listed:

Table 24-6 • Package Report / Files Information

Item	Description
File	Name of file included with this package.
Target Directory	Name of directory where the file is installed.
Version	Version number of the file.
File Size	Size of the installed file.
Component	Component that the file is associated with.

Registry View

The **Registry** view lists the registry entries that will be created when this package is installed.

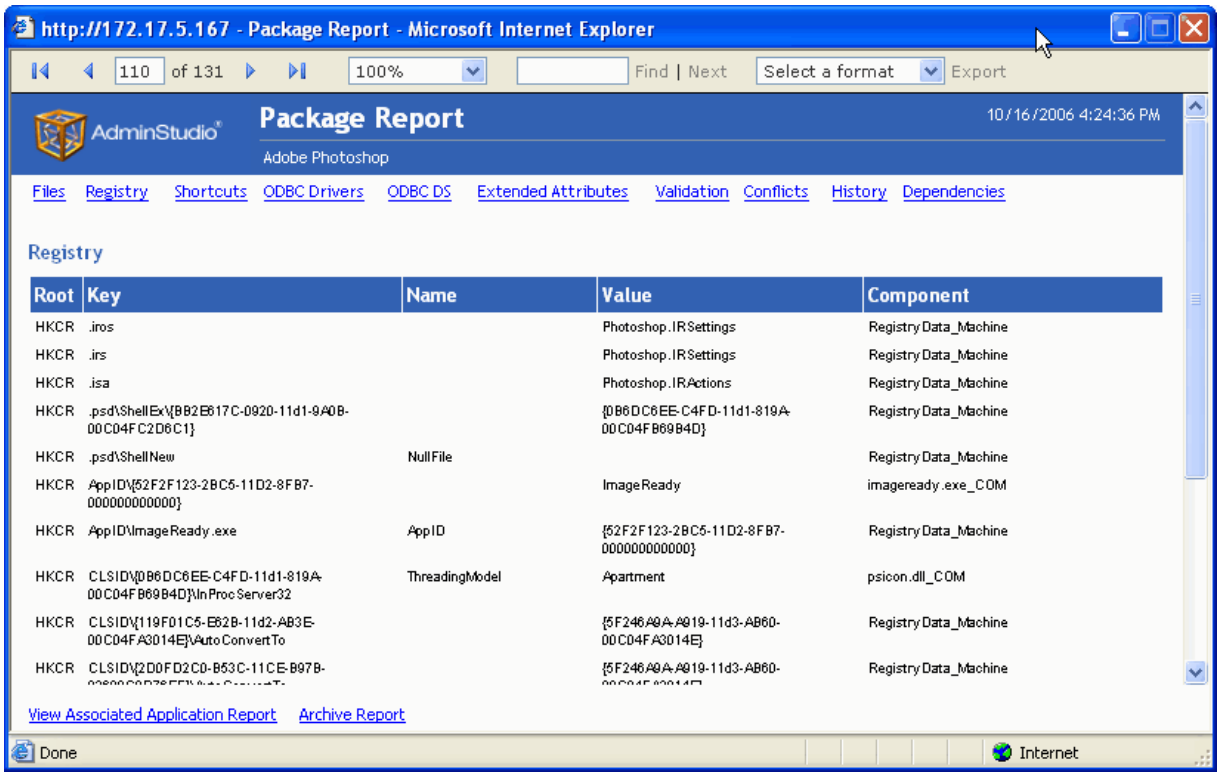


Figure 24-3: Package Report / Registry View

For each registry entry, the following information is listed:

Table 24-7 • Package Report / Registry Information

Item	Description
Root	Identifies the predefined “root” key that contains the registry entry.
Key	A registry key.
Name	Name identifying the registry entry.
Value	The string of data that defines the value of the key.
Component	Package component that the registry entry is associated with.

Shortcuts View

The **Shortcuts** view lists all of the shortcuts that will be created when this package is installed.

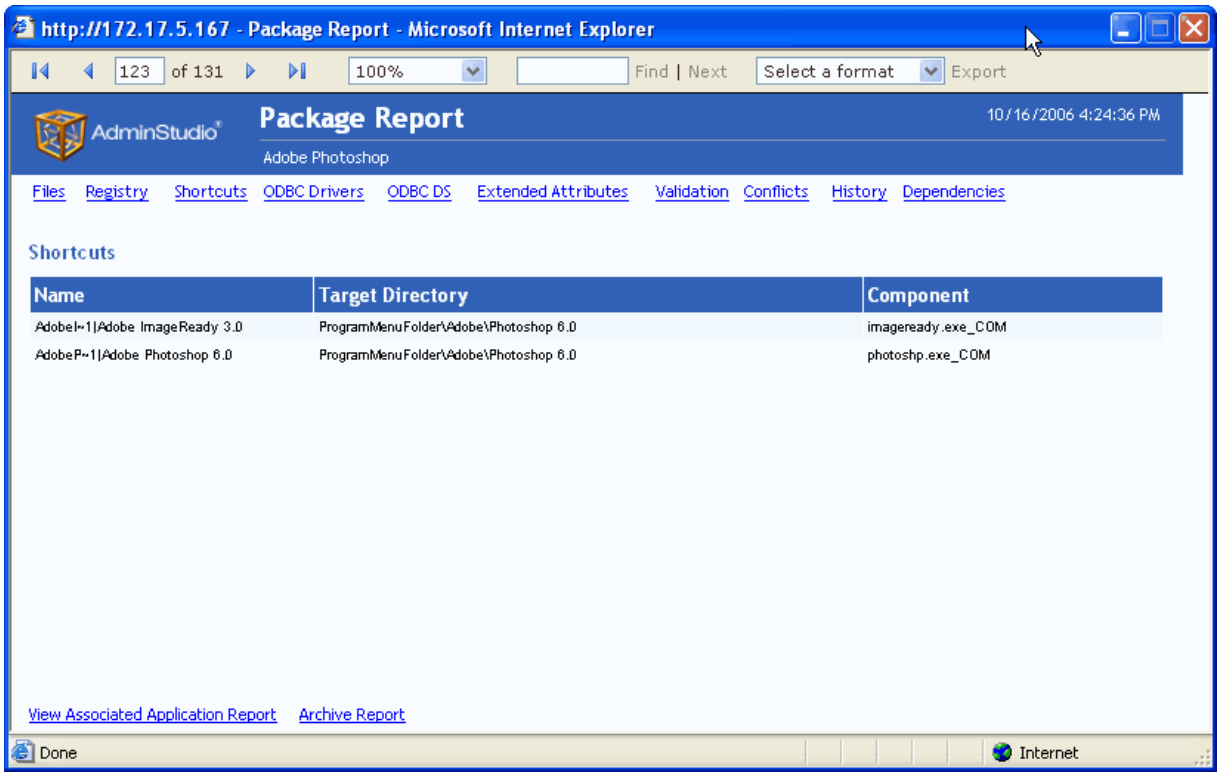


Figure 24-4: Package Report / Shortcuts View

For each shortcut, the following information is listed:

Table 24-8 • Package Report / Shortcuts Information

Item	Description
Name	Name identifying the shortcut.
Target Directory	Directory and executable that the shortcut invokes.
Component	Component associated with the shortcut.

ODBC Drivers View

The **ODBC Drivers** view lists all of the Open Database Connectivity (ODBC) drivers in the package.

ODBC Resources are ones that involve interaction with databases. ODBC drivers are libraries that implement functions involving ODBC. Each database type has its own ODBC driver.

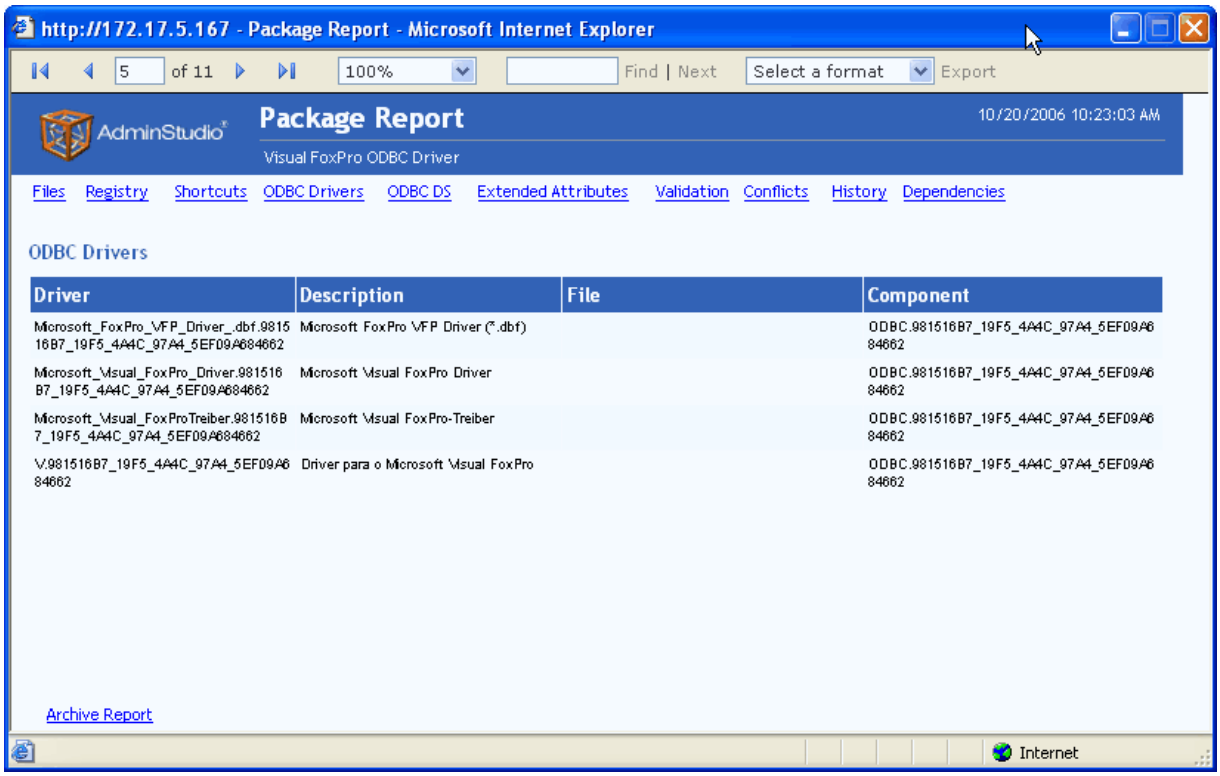


Figure 24-5: Package Report / ODBC Drivers View

For each ODBC driver, the following information is listed:

Table 24-9 • Package Report / ODBC Drivers Information

Item	Description
Driver	Name of an Open Database Connectivity (ODBC) driver in the package. Each database type has its own ODBC driver.
Description	Description of the ODBC driver identifying its associated database type.
File	File associated with the ODBC driver.
Component	Component associated with the ODBC driver.

ODBC DS View

The **ODBC DS** view lists all of the Open Database Connectivity (ODBC) data sources in the package. An ODBC data source identifies the source database type and provides information on how to connect to that database.

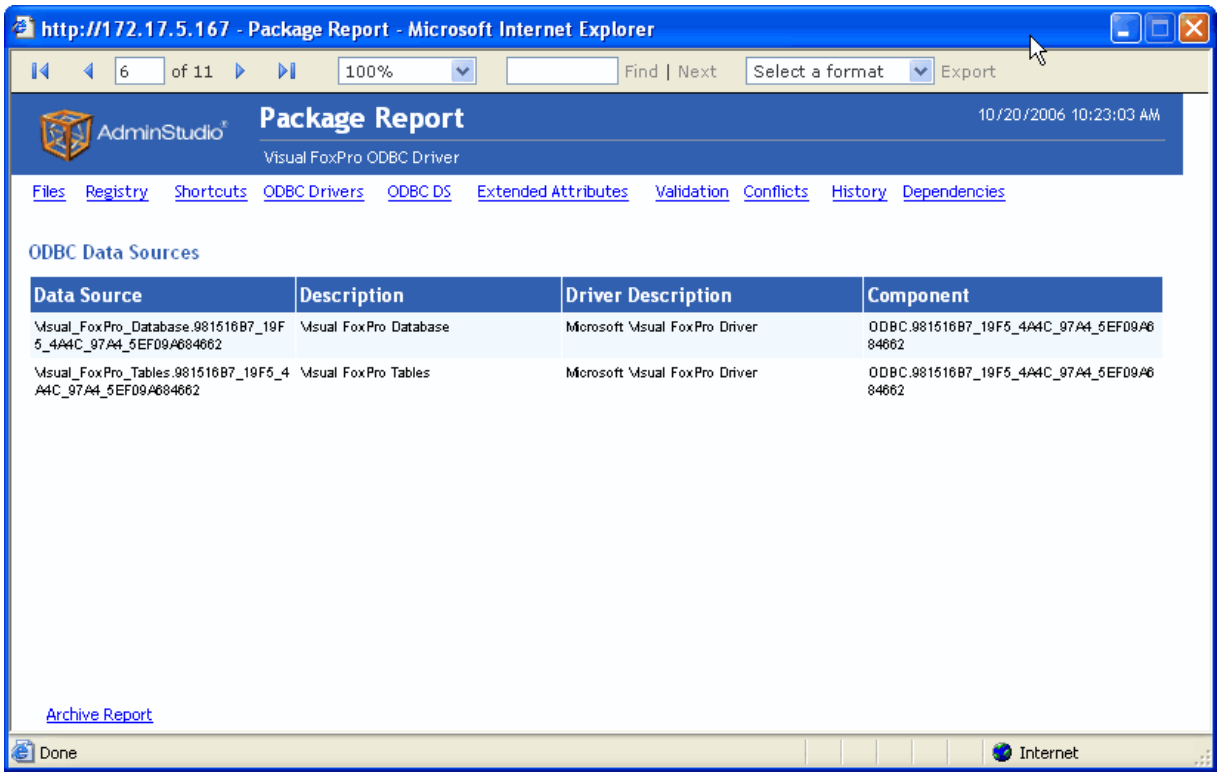


Figure 24-6: Package Report / ODBC Data Sources View

For each ODBC DS, the following information is listed:

Table 24-10 • Package Report / ODBC DS Information

Item	Description
Data Source	Name of the ODBC data source, which identifies the source database type and provides information on how to connect to that database.
Description	Identifies the database type.
Driver Description	Name of this ODBC data source's associated ODBC driver.
Component	Component that this ODBC data source is affiliated with.

Extended Attributes View

The **Extended Attributes** view lists all of the extended attribute metadata that has been entered for this package.

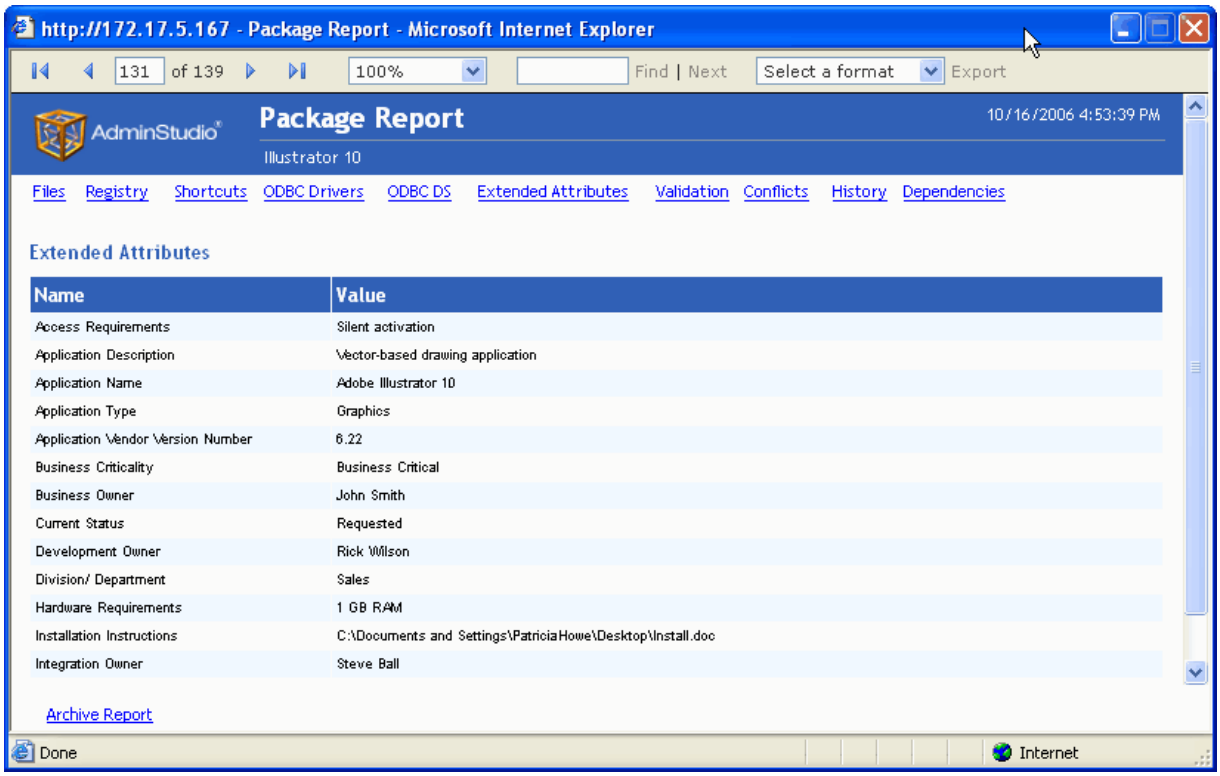


Figure 24-7: Package Report / Extended Attributes View

For each Extended Attribute, the following information is listed:

Table 24-11 • Package Report / Extended Attributes Information

Item	Description
Name	Name identifying the attribute.
Value	Content entered for the attribute.

Validation View

The **Validation** view lists all of the ICE rule errors and warnings that were generated when the package was validated against Microsoft ICEs (Internal Consistency Evaluators)—custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards.

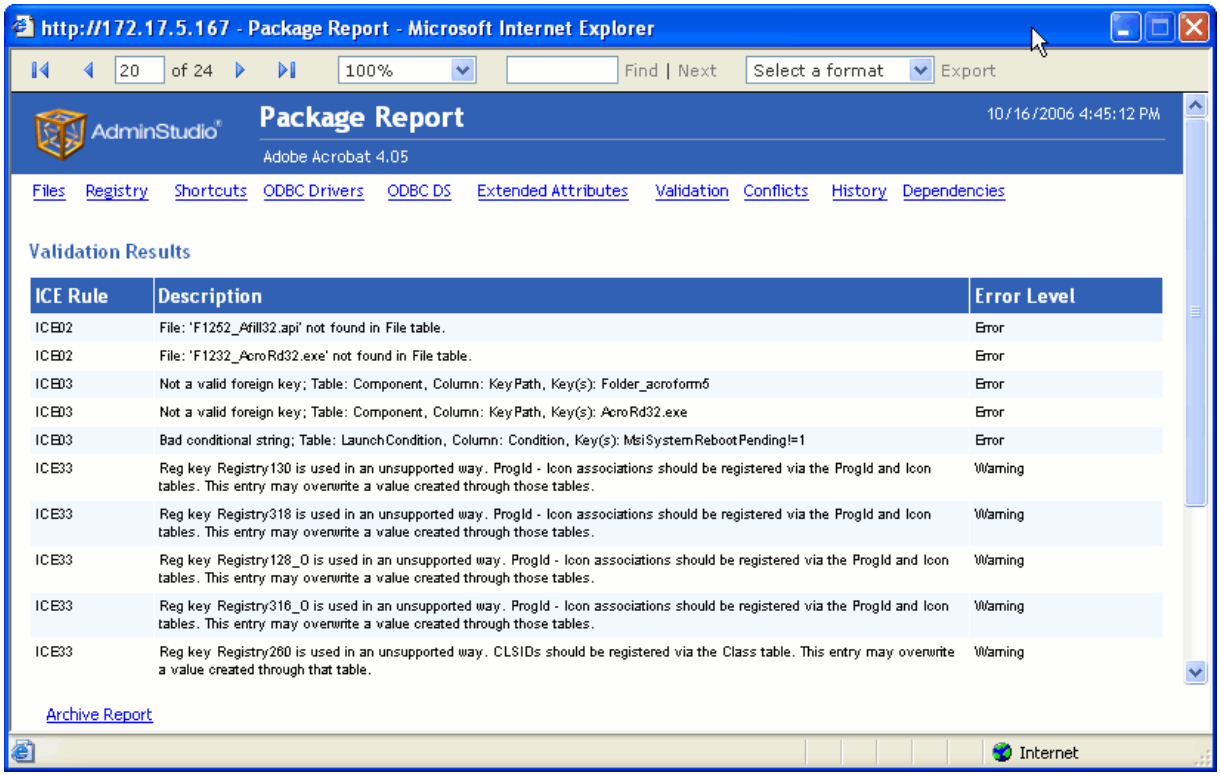


Figure 24-8: Package Report / Validation View

For each error or warning, the following information is listed:

Table 24-12 • Package Report / Validation Information

Item	Description
ICE Rule	Name of ICE Rule that generated an error or warning message.
Description	Error or warning message.
Error Level	Indicates the severity of the message as either being a Warning or an Error. • **Errors**—Package authoring that will cause incorrect behavior. • **Warnings**—Package authoring that could possibly cause incorrect behavior. Warnings can also report unexpected side-effects of package authoring.

Conflicts View

The **Conflicts** view lists all of the unresolved errors that were found when conflict analysis was performed on this package.

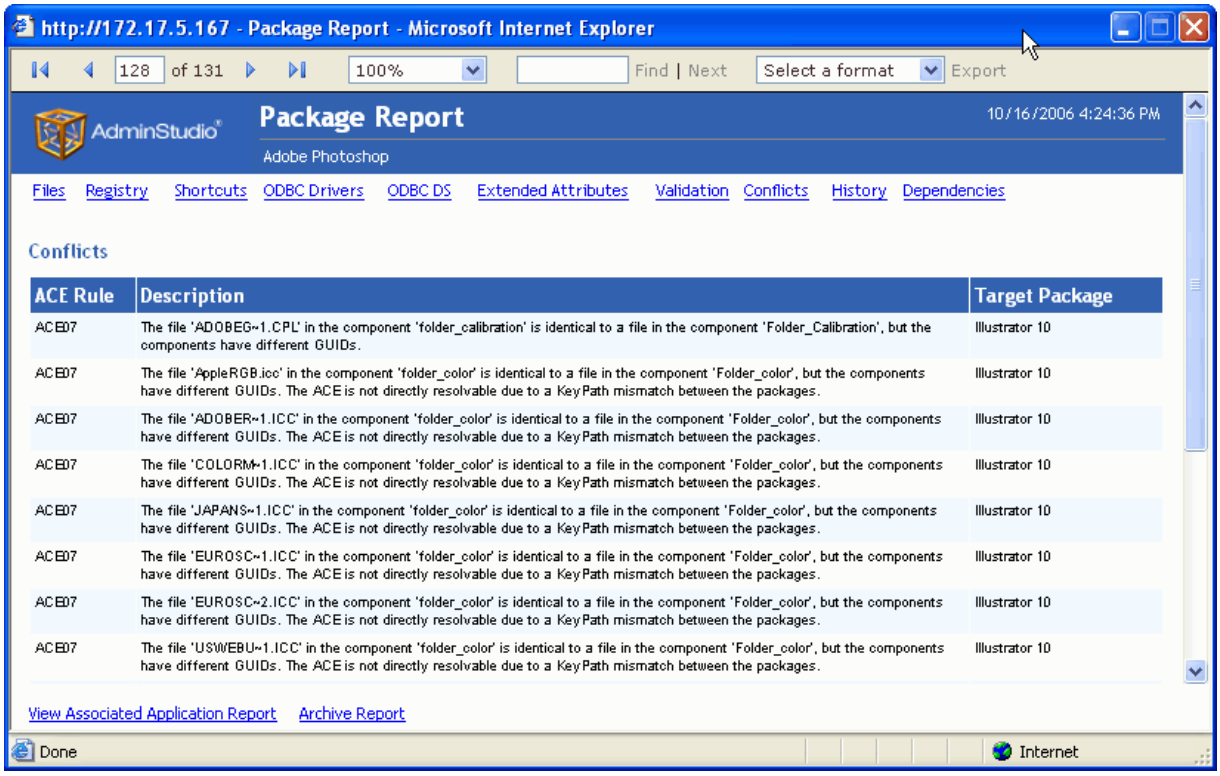


Figure 24-9: Package Report / Conflicts View

For each error, the following information is listed:

Table 24-13 • Package Report / Conflicts Information

Item	Description
ACE Rule	Name of ACE Rule that generated the message.
Description	Message generated during conflict analysis.
Target Package	Package that conflicted with this package.

History View

The **History** view lists all of the actions that have been performed on this package since it was imported into the Application Catalog.

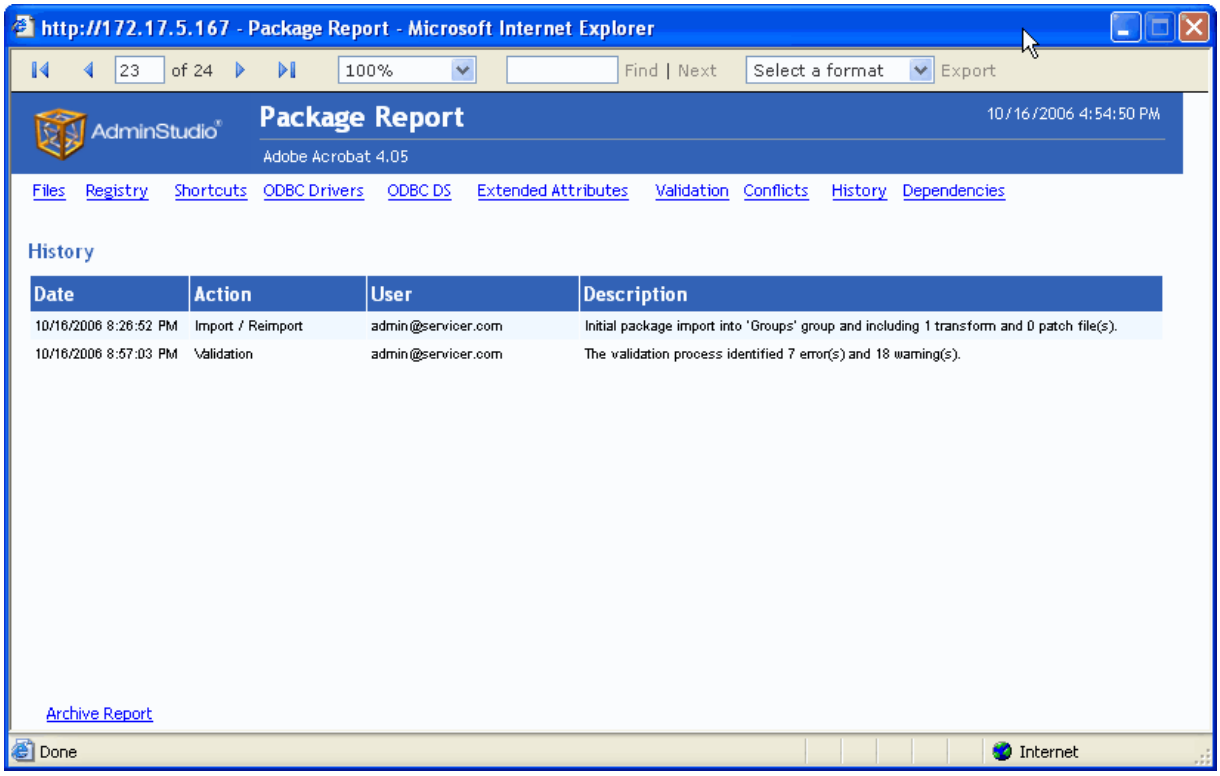


Figure 24-10: Package Report / History View

For each action, the following information is listed:

Table 24-14 • Package Report / History Information

Item	Description
Date	Day and time the event occurred.
Action	Identifies the event that occurred.
User	Identifies the user who executed the event.
Description	Description of the event that occurred.

Dependencies View

The **Dependencies** view lists all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog.

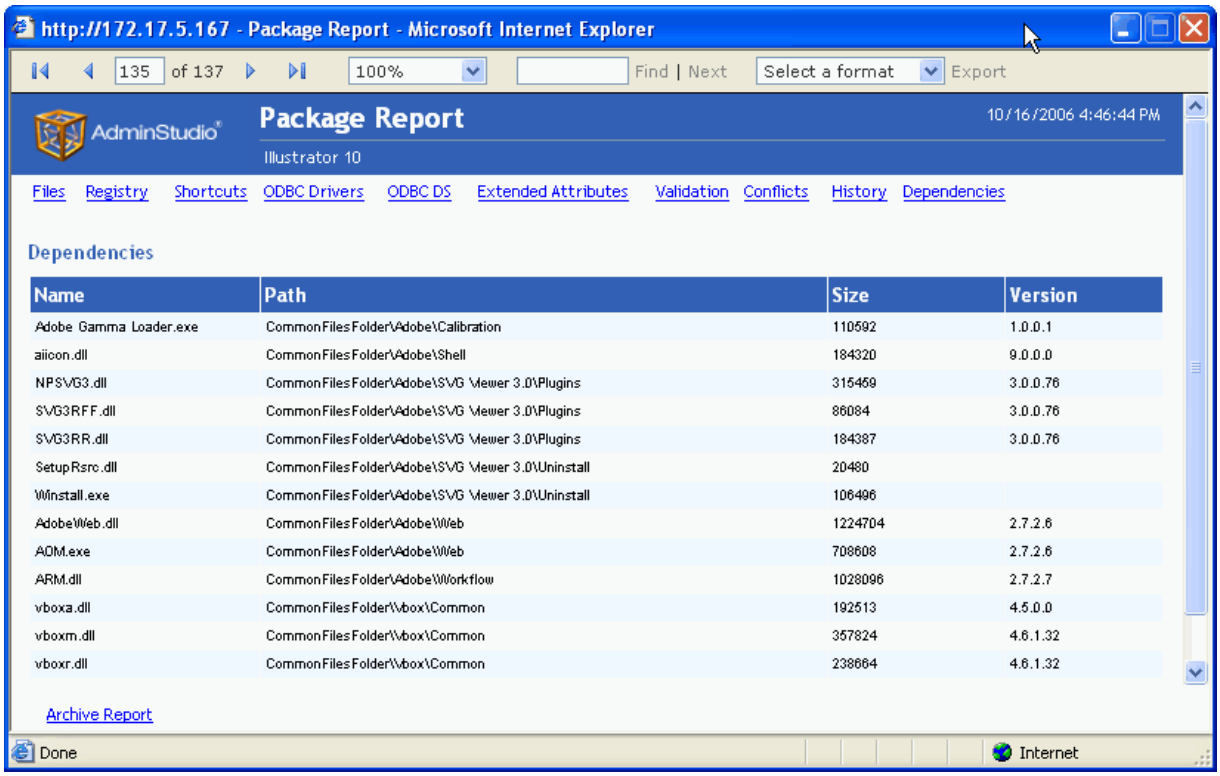


Figure 24-11: Package Report / Dependencies View

For each dependency, the following information is listed:

Table 24-15 • Package Report / Dependencies Information

Item	Description
Name	Name of a file associated with this package that has dependencies with files used by other products or operating systems in the Application Catalog.
Path	Location where this dependent file is installed.
Size	Size of the dependent file.
Version	Version of the dependent file.

Navigating Through a Package Report

The Package Report consists of the initial Package Summary View and 10 other multi-page views which are accessed by clicking the links at the top of the report:

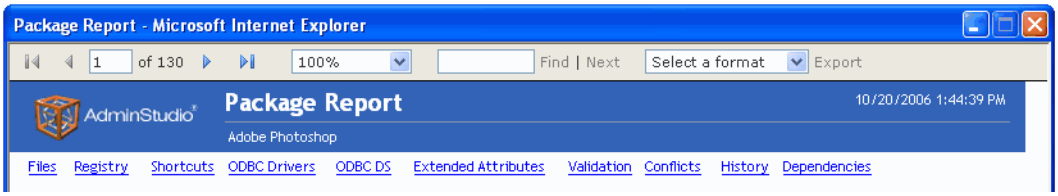


Figure 24-12: Navigation Links on the Package Report

Scrolling Through Pages of a View

Each of the Package Report views can be either a single page or multi-page, depending upon the content. The Package Report window is not resizable, so you cannot enlarge the window to display more items. Instead, you can use the Page Scrolling controls in the toolbar.

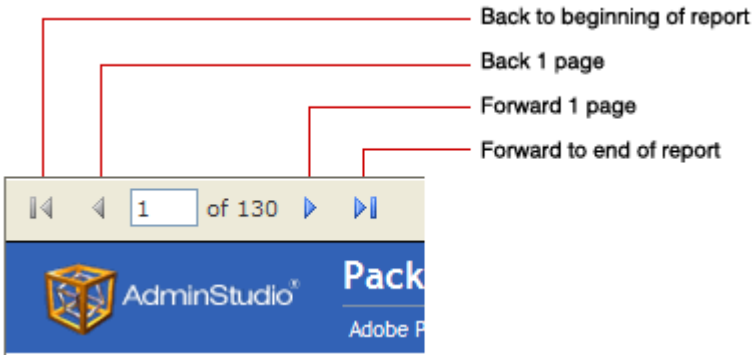


Figure 24-13: Page Scrolling Controls on Package Report

The total number of pages of the Package Report is listed in the toolbar, along with the number of the page that you are currently viewing. To jump to a specific page, enter a number in the box and click **Enter**.

Page 1 of the Package Report is the **Package Summary Information** view. Following this view, the rest of the views follow in the order in which they appear in the navigation links. The total number of pages in a Package Report is determined by adding the number of pages of all of the different views together.

Using Zoom Capability to Modify the Report Size

You can make selections from the Zoom list in the Package Report tool bar to enlarge or decrease the size of the report.

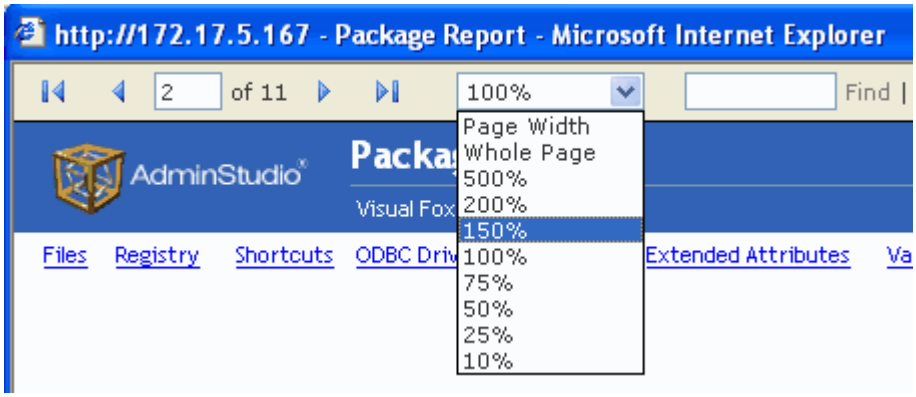


Figure 24-14: Zoom List on the Package Report

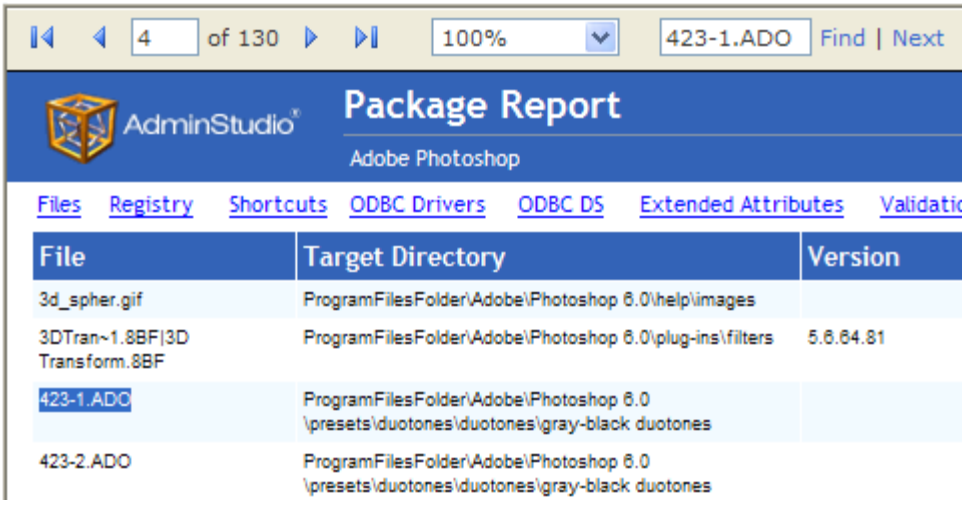
Note • *When you use the Zoom list to change the size of a Package Report, the size of the font used in the text is increased or decreased; however, the amount of information displayed on one page does not change.*

Searching for Information in a Package Report

You can use the **Find** box in the Package Report tool bar to search for specific information in the Package Report.

Task: ***To search a Package Report:***

1. In the Package Report toolbar, enter the text you want to search for in the **Find** box and click **Find**. The page containing the first instance of that text is opened, and the text you searched for is highlighted.



2. Click **Next** in the tool bar to find the next instance of the text.

Archiving a Package Report

You can archive a Package Report to document a snapshot of a package's information as of a specific date and time.

Package Reports are saved in PDF format, and therefore can be easily distributed. An archived report looks very similar to the original report, except that it is a multiple-page PDF:

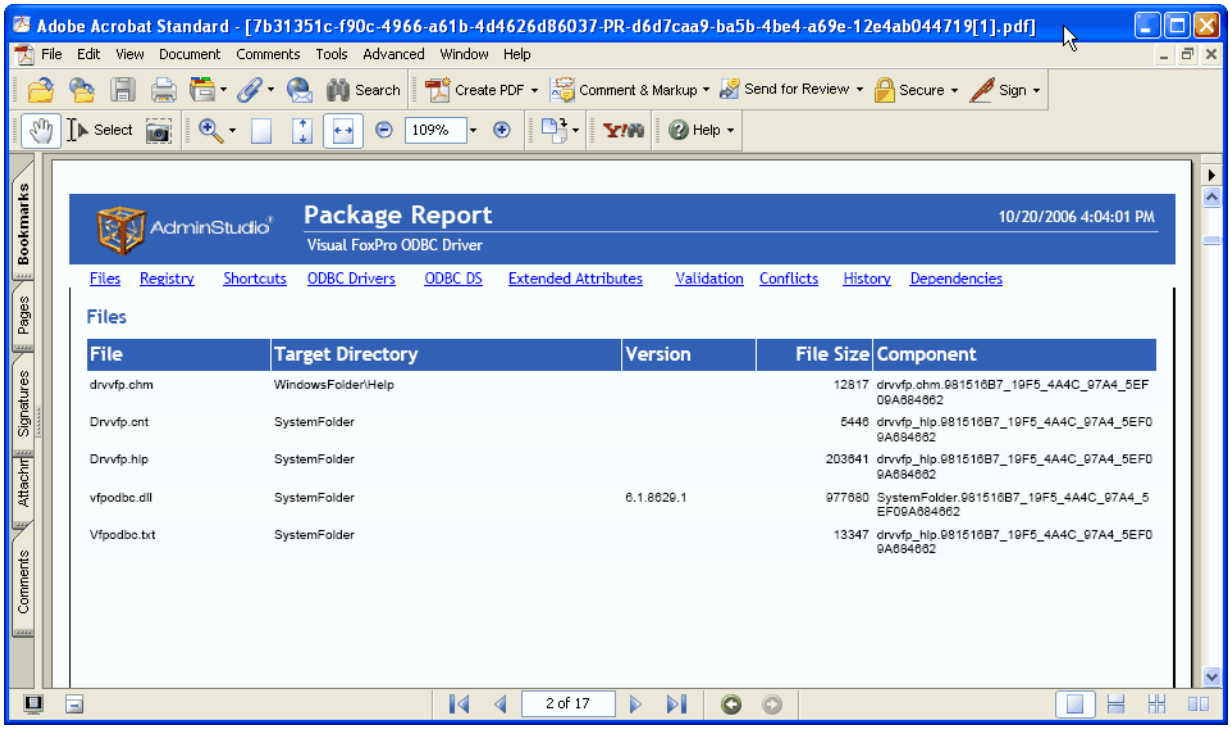


Figure 24-15: Archived Package Report

Note • *In an archived Package Report PDF, the navigation links at the top of the report (**Files**, **Registry**, **Shortcuts**, etc.) are not active. To scroll through the PDF, use the standard Adobe Reader controls.*

To archive a Package Report, perform the following steps.

Task: ***To archive a Package Report:***

1. Open a Package Report.
2. Click the **Archive Report** link in the lower left corner any of the Package Report pages. The report is archived in PDF format and the following message is displayed:

   ```
   The report has been archived.
   ```

3. Click the Report Center **All Reports** tab. The **All Reports** page opens, and the report that you just archived is listed.

   Note • *Each user's* ***Archived Reports*** *list only includes those reports that they archived. If you want others in your organization to view an archived report, you need to distribute the PDF via email or other delivery method.*

4. Click **View** next to the Package Report that you want to view. The report is opened in a PDF browser.

Deleting an Archived Package Report from the Archived Reports List

To delete an archived Package Report, perform the following steps.

Task: ***To delete an archived Package Report from the Archived Reports list:***

1. In the Archived Reports list on the **All Reports** page, right-click on the archived report you want to delete, and then click **Delete**. You are prompted to confirm the deletion.
2. Click **OK**. The archived report is deleted.

Exporting a Package Report

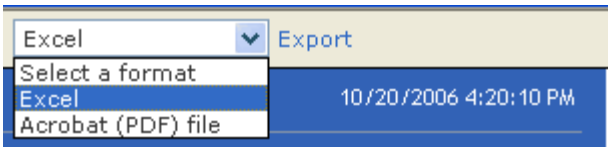
You can export the contents of a Package Report to an Excel (`.xls`) or Acrobat (`.pdf`) file.

- **Excel .xls file**—When a Package Report is exported to Microsoft Excel format, each of the Package Report views are displayed on a different worksheet.
- **Acrobat .pdf file**—An exported Package Report in PDF format is the same as the PDF created when a Package Report is archived. See Archiving a Package Report.

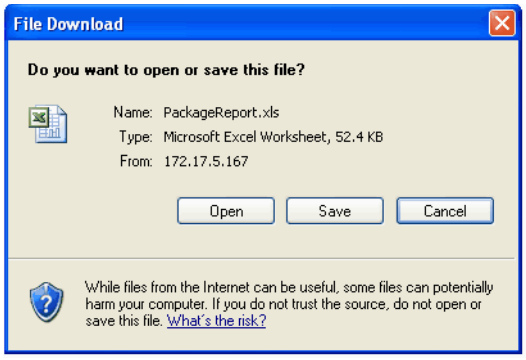
To export a Package Report, perform the following steps.

Task: ***To export a Package Report:***

1. In the Package Report tool bar, select **Excel** or **Acrobat (PDF) file** from the list.



2. Click **Export**. The File Download dialog box opens.



3. Click **Save**. The **Save As** dialog box opens.
4. Select a location for the exported file and click **Save**.

Viewing the Application Readiness Dashboard

The **Application Readiness Dashboard**, which is opened by selecting the **Application Catalog Reports** subtab of the **Report Center** tab, provides graphical representations of summary data concerning the readiness of Windows Installer and App-V packages for distribution.

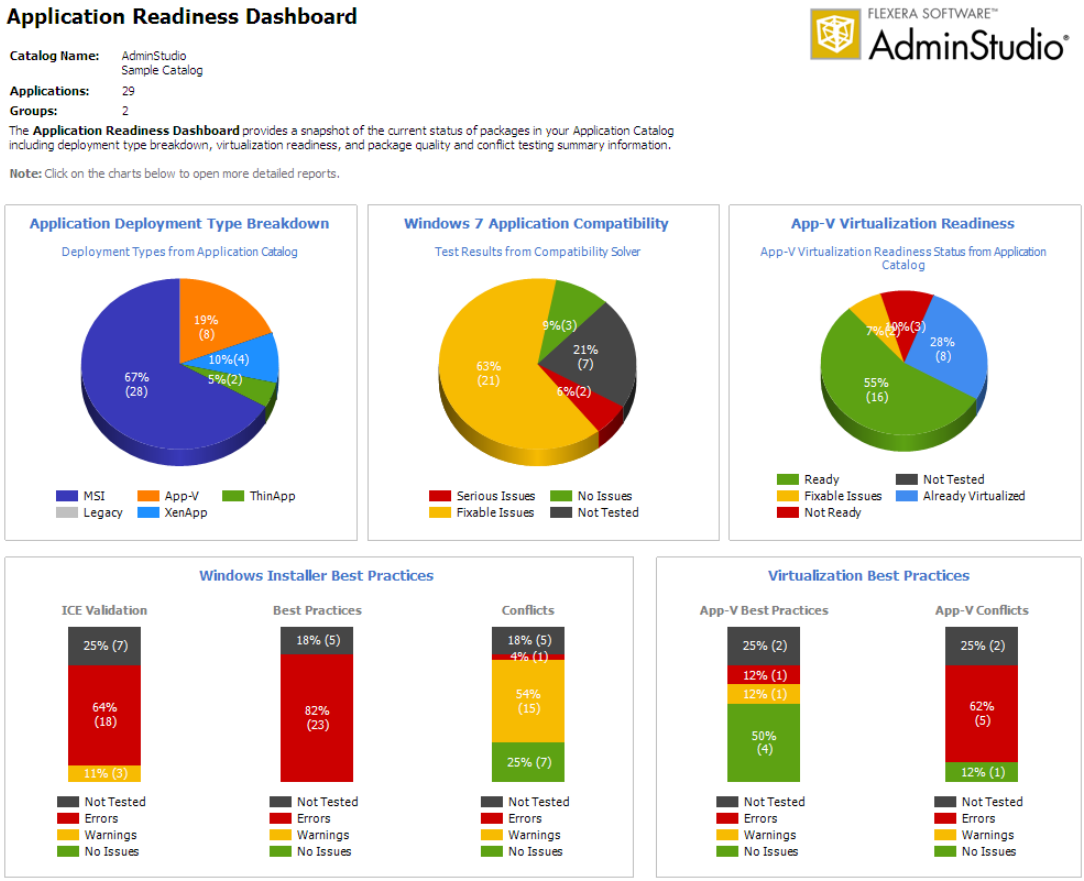


Figure 24-16: Application Readiness Dashboard

Tip • *Click on specific categories of these charts to open more detailed reports.*

Important • *In addition to the Application Readiness Dashboard, additional reports can be viewed on the* ***Application Catalog Reports*** *subtab of the* ***Report Center*** *tab. See Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards for a detailed description of those reports.*

The Application Readiness Dashboard includes the following information:

Table 24-16 • Application Readiness Dashboard

Description	Chart
Application Deployment Type Breakdown	Provides a graph listing the percentage of packages in the Application Catalog by deployment type (Windows Installer, App-V, XenApp, ThinApp, or Legacy). See Application Deployment Type Breakdown.
Windows 7 Application Compatibility	Lists the results of Windows 7 application compatibility tests that were performed in Compatibility Solver on packages that were loaded into the Compatibility Solver database. Packages are grouped according to the type of issues found: **Serious Issues**, **Fixable Issues**, **No Issues**, or **Not Tested**. See Windows 7 Application Compatibility.
App-V Virtualization Readiness	Lists the percentage of and number of Windows Installer packages in the Application Catalog in each of the following App-V virtualization readiness categories: **Ready**, **Fixable Issues**, **Not Ready**, **Not Tested**. This chart also lists the number of App-V packages in the Application Catalog in the **Already Virtualized** category. See App-V Virtualization Readiness.
Windows Installer Best Practices	Lists the Microsoft ICE Validation status of Windows Installer packages, and the AdminStudio Best Practices and Conflict test status of Windows Installer packages. See Windows Installer Best Practices.
Virtualization Best Practices	Lists the AdminStudio Best Practices and Conflict test status of App-V packages. See Virtualization Best Practices.

Note • *For more information, see Application Readiness Dashboard.*

Generating a Custom SQL Query Report for AdminStudio

You can generate a Custom SQL Query Report to include data generated by both AdminStudio and Workflow Manager. To generate a Custom SQL Query Report, perform the following steps.

Task: ***To generate a new Custom SQL Query report:***

1. Open the **Report Center** tab. The **All Reports** page opens.
2. Click **Add**. The **Select Report Type** page opens.
3. Click **Custom SQL Query Report**. The **Enter SQL Query Panel** opens.
4. Enter an SQL Query to retrieve the data for this report. Click the **Test Query** button to verify the query syntax.

   Tip • *To assist you in writing queries to retrieve data, see Wildcard Support in Report Center SQL Queries.*
5. Click **Next**. The **General Information Panel** opens.
6. Enter a **Report Name** and **Description** to clearly identify the contents and purpose of this report. This name and description will be listed on the **All Reports** page.
7. Select the Roles that you want to have permission to view this report.
8. Click **Next**. The **Summary Panel** opens, which displays all the information needed to create the report.
9. Click **Finish**. The report is generated in the **Report View**. This report is also saved and now appears in the list on the **All Reports** page.

Wildcard Support in Report Center SQL Queries

In Report Center searches, the `LIKE` operator is always used. You can combine the `LIKE` operator with a wildcard character, and the following rules apply:

Table 24-17 • Wildcard Support in Report Center Queries

Situation	Rule

When no wildcards are used	If you do not enter a wildcard character in the **Search** box, then Report Center performs a “LIKE” search, which searches for any occurrence of that text anywhere in the item that is being searched for.
	
	For example, if you are searching for a file name that has the word `test` anywhere in the file name, and you entered `test` in the **Search** box, it would be interpreted by Report Center as:
	
	`*test*`
	
	And the following files would be found:
	
	`MyTestFile and TestFile`

Table 24-17 • Wildcard Support in Report Center Queries

Situation	Rule
When wildcards are used	You can specify a * wildcard in the **Search** box to narrow the search results. For example, if you are searching for a file name that includes the word `test`, but does not begin with it, and you entered `*test` in the **Search** box, `MyTest` would be returned, but not `TestFile`.

Report Center Reference

This section includes reference information on the following pages, views, and reports:

- All Reports Page
- Report View
- Search Packages Page
- Package Report
- Reports Wizard

All Reports Page

The **All Reports** page provides access to the archived Package Reports and Custom SQL Query reports.

The All Reports page includes the following options:

Option	Description
Report Name	Name of a saved report.
Report Description	Text describing the contents of the report.
Add	Click to design and generate a new report.

Available Reports

From the **All Reports** page, you can choose to create a new Custom SQL Query Report, or view a report that was already created. From this page you can also open an archived Package Report.

Viewing an Existing Custom Report from the All Reports Page

To view a report, click **View** to launch the **Report View**.

Report View

The Report View, which is opened by clicking **View** next to an existing Custom Report on the **All Reports** page, displays the contents of a saved report.

Table 24-18 • Report View

Option	Description
Report Name	Name of this report.
Description	Text describing the contents of the report.
Default Filter	Name of filter, if any, applied to this report.
Export	Click to export this report to Microsoft Excel format.

Search Packages Page

From the **Search Packages** page, you can select or search for a specific package, and then generate a detailed Package Report.

On the **Search Packages** page, you can filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria, which are grouped into three categories:

- **Package Attributes**—Search by properties assigned to the Windows Installer package. See Package Attributes.
- **Package Content**—Search by files, registry entries, `.ini` files, or shortcuts contained in the Windows Installer package. See Package Content.
- **Application Request Attributes**—Search by information related to a package's associated Application Request. See Application Request Attributes.

To filter the list of packages displayed in the package tree to display only those packages that meet specific search criteria, enter values in the criteria fields that you want to search on, and click **Search**. The packages that meet *any of the criteria* are then listed in the package tree in alphabetical order and are no longer grouped.

Package Attributes

You can search for packages in a catalog based on one or more of any of the following Package attribute metadata:

Table 24-19 • Package Attribute Search Fields

Metadata	Description
Package Code	Enter the GUID that identifies a particular Windows Installer `.msi` package. The Package Code associates an `.msi` file with an application or product and is represented as a string GUID—a text string that has a special format: `{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}` where each X character is a hex digit (0 through 9 or uppercase A through F).
Product Code	Enter the GUID that uniquely identifies the particular product release of the package. The **ProductCode** is a Windows Installer property and is represented as a string GUID—a text string that has a special format: `{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX}` where each X character is a hex digit (0 through 9 or uppercase A through F).

Table 24-19 • Package Attribute Search Fields (cont.)

Metadata	Description
Upgrade Code	Enter the GUID that identifies the family of products that are in the same upgrade path. The **UpgradeCode** is a Windows Installer property and is represented as a string GUID—a text string that has a special format: `{XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXsXXXXX}` where each X character is a hex digit (0 through 9 or uppercase A through F). ***Note*** • *Each stand-alone product usually has its own **UpgradeCode** GUID. Every version of XYZ Product typically uses the same GUID for the **UpgradeCode**. In other words, Product A Version 1.0 has the same UpgradeCode as Product A Version 2.0, but has a different UpgradeCode than Product B.*
Setup File Name	Name of the Windows Installer (.msi) file that was imported into the Application Catalog.
Comments	Enter the text of any comments associated with the package.
Extended Attributes	Enter the value of any of the Extended Attributes associated with the package.

Package Content

You can search for packages in a catalog based on one or more of any of the following Package Content metadata

Table 24-20 • Package Content Search Fields

Metadata	Description
File	Enter the file name of one of the files in the Windows Installer package.
Registry Key	Enter a registry key to search on.
Registry Value	Enter a registry value to search on.
INI File	Enter any changes to an `.ini` file that are made when the product is installed.
Shortcut	Enter any shortcuts that are created when the product is installed.

Application Request Attributes

You can search for packages in a catalog based on one or more of any of the following attributes of the package's associated Application Request:

Table 24-21 • Application Request Attributes Search Fields

Metadata	Description
Name	Enter the name of the package's associated Application Request.
Upload Date	Date the Application Request was submitted.
Due Date	Enter the date the Application Request is scheduled to be completed, based upon its value for **Application Due Period**.
Risk Date	Enter the date at which the Application Request's status will change to **At Risk**, which is based upon its value for **Application At Risk Period**.
Due Period	Enter, in days, the length of time this Application Request needs to be completed in order to meet its Project's Service Level Agreement (SLA) requirements.
End Date	Enter the date the Application Request was completed.

Application Catalog Reports Page

AdminStudio 10.0 provides an enhanced reporting infrastructure. AdminStudio has incorporated Microsoft SQL Reporting Services to provide a summary dashboard of Application Catalog readiness data and to enable you to create customizable reports.

- Viewing the Application Readiness Dashboard
- Exporting a Report in PDF, Excel, or Word Format

Viewing the Application Readiness Dashboard

The **Application Readiness Dashboard**, which is opened by selecting the **Application Catalog Reports** subtab of the **Report Center** tab, provides graphical representations of summary data concerning the readiness of Windows Installer and App-V packages for distribution.

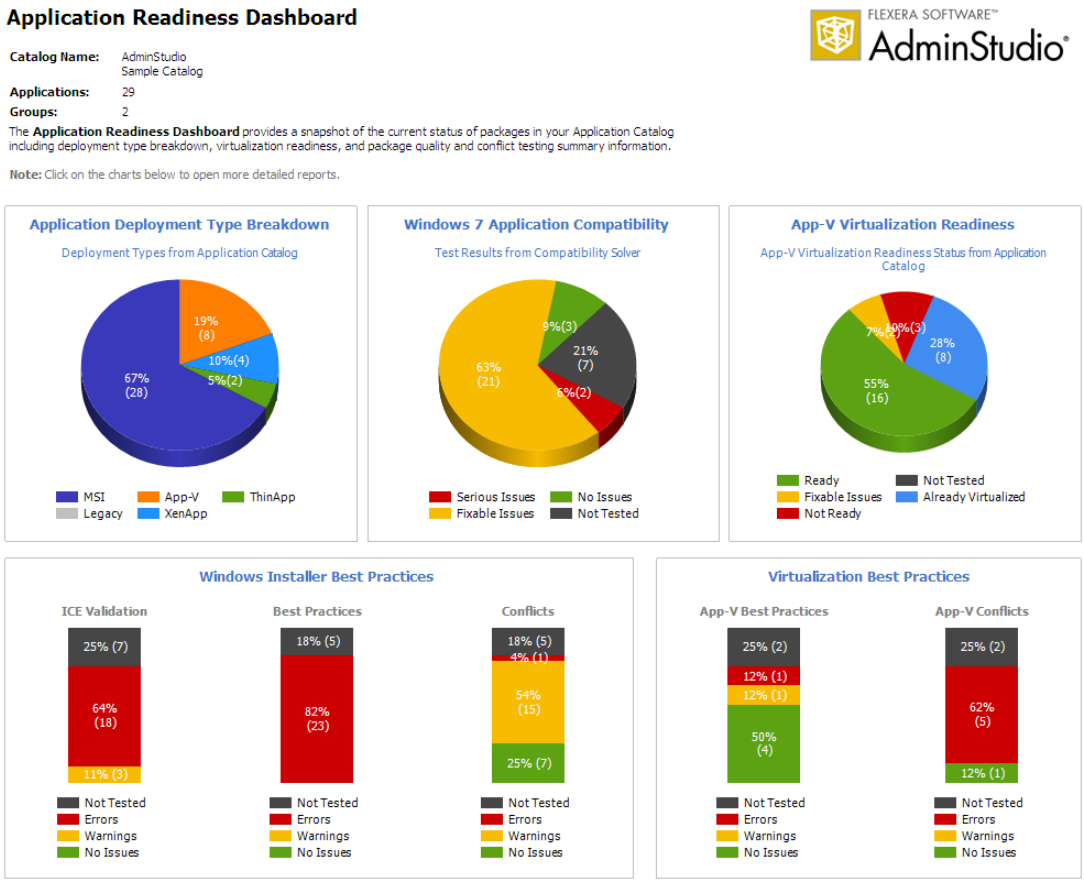


Figure 24-17: Application Readiness Dashboard

Important • *In addition to the Application Readiness Dashboard, additional reports can be viewed on the* ***Application Catalog Reports*** *subtab of the* ***Report Center*** *tab. See Application Catalog View (Enterprise Edition): Advanced Application Catalog Reporting and Dashboards for a detailed description of those reports.*

The Application Readiness Dashboard includes the following information:

Table 24-22 • Application Readiness Dashboard

Description	Chart
Application Deployment Type Breakdown	Provides a graph listing the percentage of packages in the Application Catalog by deployment type (Windows Installer, App-V, XenApp, ThinApp, or Legacy). See Application Deployment Type Breakdown.
Windows 7 Application Compatibility	Lists the results of Windows 7 application compatibility tests that were performed in Compatibility Solver on packages that were loaded into the Compatibility Solver database. Packages are grouped according to the type of issues found: **Serious Issues**, **Fixable Issues**, **No Issues**, or **Not Tested**. See Windows 7 Application Compatibility.
App-V Virtualization Readiness	Lists the percentage of and number of Windows Installer packages in the Application Catalog in each of the following App-V virtualization readiness categories: **Ready**, **Fixable Issues**, **Not Ready**, **Not Tested**. This chart also lists the number of App-V packages in the Application Catalog in the **Already Virtualized** category. See App-V Virtualization Readiness.
Windows Installer Best Practices	Lists the Microsoft ICE Validation status of Windows Installer packages, and the AdminStudio Best Practices and Conflict test status of Windows Installer packages. See Windows Installer Best Practices.
Virtualization Best Practices	Lists the AdminStudio Best Practices and Conflict test status of App-V packages. See Virtualization Best Practices.

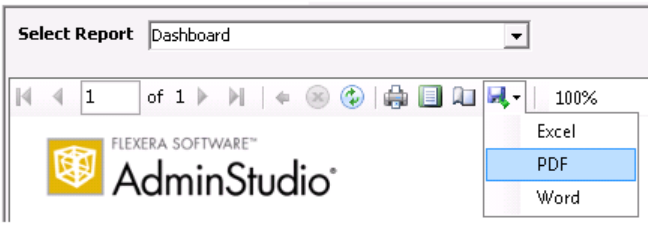
Note • *For more information, see Application Readiness Dashboard.*

Exporting a Report in PDF, Excel, or Word Format

You can save the Application Readiness Dashboard report, or any of the drill-through reports, in PDF, Microsoft Excel, or Microsoft Word format.

Task: ***Saving a report:***

1. View the report that you want to save.
2. In the toolbar, click the **Save** icon.



3. From the menu, select either **Excel**, **PDF**, or **Word**. The report is exported and you are prompted for a location to store the report.
4. Specify a location and click **Save**.

Note • *You can also print the currently viewed report by clicking the **Print** icon in the toolbar.*

Package Report

This topic lists the data that is displayed in each section of a Package Report.

- Package Summary Information View
- Files View
- Registry View
- Shortcuts View
- ODBC Drivers View
- ODBC DS View
- Extended Attributes View
- Validation View
- Conflicts View
- History View
- Dependencies View

Note • *See also see Information Included in Package Reports.*

Note • *Additional information may be available for App-V packages.*

Package Summary Information View

The initial view (Page 1) of a Package Report is the **Package Summary Information** view, and it lists the following information:

Table 24-23 • Package Report / Package Summary Information

Item	Description
Product Name	Name assigned to the package.
Manufacturer	Company that authored the package.
Import Date	The date and time the package was imported into the Application Catalog.
Unresolved Conflicts	The number of detected conflicts, generated during conflict analysis of this package, which have not yet been resolved—either automatically or manually.
Product Version	Version of package that is recorded in the package's Windows Installer file.

Table 24-23 • Package Report / Package Summary Information (cont.)

Item	Description
Product Language	Decimal-based code identifying the language that this software package was authored for. For example, English is 1033, German is 1031, and Japanese is 1041.
In Software Repository	Indicates whether or not this package and its associated files are managed by the Software Repository.

Files View

The **Files** view lists the all of the files included in the selected package, and the location where these files will be installed. For each file, the following information is listed:

Table 24-24 • Package Report / Files Information

Item	Description
File	Name of file included with this package.
Target Directory	Name of directory where the file is installed.
Version	Version number of the file.
File Size	Size of the installed file.
Component	Component that the file is associated with.

Registry View

The **Registry** view lists the registry entries that will be created when this package is installed. For each registry entry, the following information is listed:

Table 24-25 • Package Report / Registry Information

Item	Description
Root	Identifies the predefined “root” key that contains the registry entry.
Key	A registry key.
Name	Name identifying the registry entry.
Value	The string of data that defines the value of the key.
Component	Package component that the registry entry is associated with.

Shortcuts View

The **Shortcuts** view lists all of the shortcuts that will be created when this package is installed. For each shortcut, the following information is listed:

Table 24-26 • Package Report / Shortcuts Information

Item	Description
Name	Name identifying the shortcut.
Target Directory	Directory and executable that the shortcut invokes.
Component	Component associated with the shortcut.

ODBC Drivers View

The **ODBC Drivers** view lists all of the Open Database Connectivity (ODBC) drivers in the package.

ODBC Resources are ones that involve interaction with databases. ODBC drivers are libraries that implement functions involving ODBC. Each database type has its own ODBC driver. For each ODBC driver, the following information is listed:

Table 24-27 • Package Report / ODBC Drivers Information

Item	Description
Driver	Name of an Open Database Connectivity (ODBC) driver in the package. Each database type has its own ODBC driver.
Description	Description of the ODBC driver identifying its associated database type.
File	File associated with the ODBC driver.
Component	Component associated with the ODBC driver.

ODBC DS View

The **ODBC DS** view lists all of the Open Database Connectivity (ODBC) data sources in the package. An ODBC data source identifies the source database type and provides information on how to connect to that database. For each ODBC DS, the following information is listed:

Table 24-28 • Package Report / ODBC DS Information

Item	Description
Data Source	Name of the ODBC data source, which identifies the source database type and provides information on how to connect to that database.
Description	Identifies the database type.

Table 24-28 • Package Report / ODBC DS Information (cont.)

Item	Description
Driver Description	Name of this ODBC data source's associated ODBC driver.
Component	Component that this ODBC data source is affiliated with.

Extended Attributes View

The **Extended Attributes** view lists all of the extended attribute metadata that has been entered for this package. For each Extended Attribute, the following information is listed:

Table 24-29 • Package Report / Extended Attributes Information

Item	Description
Name	Name identifying the attribute.
Value	Content entered for the attribute.

Validation View

The **Validation** view lists all of the ICE rule errors and warnings that were generated when the package was validated against Microsoft ICEs (Internal Consistency Evaluators)—custom actions written by Microsoft which can be executed to determine if an installation package is built according to Windows Installer standards.

For each error or warning, the following information is listed:

Table 24-30 • Package Report / Validation Information

Item	Description
ICE Rule	Name of ICE Rule that generated an error or warning message.
Description	Error or warning message.
Error Level	Indicates the severity of the message as either being a Warning or an Error. • **Errors**—Package authoring that will cause incorrect behavior. • **Warnings**—Package authoring that could possibly cause incorrect behavior. Warnings can also report unexpected side-effects of package authoring.

Conflicts View

The **Conflicts** view lists all of the unresolved errors that were found when conflict analysis was performed on this package. For each error, the following information is listed:

Table 24-31 • Package Report / Conflicts Information

Item	Description
ACE Rule	Name of ACE Rule that generated the message.
Description	Message generated during conflict analysis.
Target Package	Package that conflicted with this package.

History View

The **History** view lists all of the actions that have been performed on this package since it was imported into the Application Catalog. For each action, the following information is listed:

Table 24-32 • Package Report / History Information

Item	Description
Date	Day and time the event occurred.
Action	Identifies the event that occurred.
User	Identifies the user who executed the event.
Description	Description of the event that occurred.

Dependencies View

The **Dependencies** view lists all of a package's files that have dependencies with files used by other products or operating systems in the Application Catalog. For each dependency, the following information is listed:

Table 24-33 • Package Report / Dependencies Information

Item	Description
Name	Name of a file associated with this package that has dependencies with files used by other products or operating systems in the Application Catalog.
Path	Location where this dependent file is installed.
Size	Size of the dependent file.
Version	Version of the dependent file.

Reports Wizard

Using the Reports Wizard, you can generate a Custom SQL Query report. See Generating a Custom SQL Query Report for AdminStudio and the Enter SQL Query Panel.

Select Report Type Panel

On the **Select Report Type Panel** of the Reports Wizard, select one of the following options:

Table 24-34 • Select Report Type Panel Options

Option	Description
Custom Report	Select to create a custom report.
Activity Report	Select to create report that exclusively lists Application Request activities.
Custom SQL Query Report	Select to create a custom report defined by entering an SQL query in the Report Wizard.

Enter SQL Query Panel

On the **Enter SQL Query Panel** of the **Custom SQL Query Reports Wizard**, enter an SQL query to retrieve the data for this report. Click the **Test Query** button to verify the query syntax, and click **Next** to proceed.

Wildcard Support in Report Center SQL Queries

In Report Center searches, the `LIKE` operator is always used. You can combine the `LIKE` operator with a wildcard character, and the following rules apply:

Table 24-35 • Wildcard Support in Report Center Queries

Situation	Rule

When no wildcards are used	If you do not enter a wildcard character in the **Search** box, then Report Center performs a “LIKE” search, which searches for any occurrence of that text anywhere in the item that is being searched for.
	
	For example, if you are searching for a file name that has the word `test` anywhere in the file name, and you entered `test` in the **Search** box, it would be interpreted by Report Center as:
	
	`*test*`
	
	And the following files would be found:
	
	`MyTestFile and TestFile`

Table 24-35 • Wildcard Support in Report Center Queries

Situation	Rule
When wildcards are used	You can specify a * wildcard in the **Search** box to narrow the search results. For example, if you are searching for a file name that includes the word `test`, but does not begin with it, and you entered `*test` in the **Search** box, `MyTest` would be returned, but not `TestFile`.

General Information Panel

On the **General Information Panel** of the Reports Wizard, enter a **Report Name** and **Description** to clearly identify the contents and purpose of this report. This name and description will be listed on the **All Reports** page.

Next, select the **Roles** that you want to have permission to view this report.

Click **Next** to continue with the Reports Wizard.

Note • *You can change the selected Roles at any time after this report is created by clicking Edit next to the Report Name on the **All Reports** page.*

Summary Panel

On the **Summary Panel** of the Reports Wizard, the following information is listed:

Table 24-36 • Summary Panel of the Reports Wizard

Option	Description
Report Name	Name of report.
Report Fields	List of fields that you selected to be included in this report.
Template Data	A list of the Template data you selected to be in this report.
Filters	A list of filters applied to this report.

Click **Finish** to generate the Report. The report is generated in the **Report View**. This report is also saved and now appears in the list on the **All Reports** page.

25

Automating Tasks Using Job Manager

Edition • *Job Manager is included with AdminStudio Enterprise Edition.*

You can use Job Manager to automate time consuming tasks such as package import, validation, conflict analysis, and conflict resolution. With Job Manager, you can schedule these tasks to run automatically at scheduled times, perhaps when system availability is highest. This enables your organization to increase the efficiency of your packaging process and to enforce standardized business practices.

Table 25-1 • Job Manager User Documentation

Section	Description
About Job Manager	Describes the purpose of Job Manager, and the types of tasks that you can automate.
Directory Monitoring and Job Manager	Describes how the **Directory Monitoring** Job Step can be included in both Template Jobs and Custom Jobs to examine a specified directory for new or modified packages and then to automatically import or reimport those packages into the Application Catalog.
Managing Jobs	Explains how to create a new job, reschedule a job, set email alerts for a job, and view job status.
Managing Templates	Explains how to create and edit Job Templates.
Setting Job Manager Conflict Detection Options	Explains how to set import, validation, and detection options.
Managing Jobs Using Job Manager Engine	Explains how to use Job Manager Engine to manage jobs from multiple Application Catalogs.

Table 25-1 • Job Manager User Documentation (cont.)

Section	Description
Job Manager Reference	Provides detailed information on all of the Job Manager pages.

About Job Manager

You can use Job Manager to automate time consuming tasks such as package import, validation, conflict analysis, and conflict resolution. Job Manager provides:

- **Increased efficiency**—With Job Manager, you can schedule common tasks—such as package import, conflict analysis, and conflict resolution—to run automatically at scheduled times (such as when system availability is highest), and no human intervention is required.
- **Enforcement of consistent business practices**—Job Manager enables organizations to enforce consistent business practices in the deployment preparation process. It allows you to perform the exact same series of steps on any package before it is deployed in your enterprise, such as checking against specific ICE or ACE rules, and against a specific operating system.
- **Dynamic group content identification**—You can create a Job that will perform actions on whatever packages are in a selected group at the time the Job is executed. You can schedule a reoccurring Job to automatically perform actions on the exact contents of a group without ever having to select any packages or create any new Jobs. This allows you to enforce consistent business practices on a group's contents even as the content changes.

This section includes the following topics:

- Job Steps: Tasks You Can Automate
- Template Jobs vs. Custom Jobs
- Selecting Contextual Packages in a Custom Job
- Dynamic Group Content Identification
- Conflict Resolution and the Software Repository
- User Permissions and Job Manager

Job Steps: Tasks You Can Automate

Jobs are comprised of one or more Job Steps. The following Job Step Types are available:

Table 25-2 • Job Manager Step Types

Task	Description
Scan for Dependencies	The **Scan for Dependencies** Job Step analyzes a Windows Installer package for dependent files. This information is used primarily to enhance the quality of any future Patch Impact Analysis operation that may be performed on this package.
Directory Monitoring	You can use the **Directory Monitoring** Job Step to establish a consistent, repeatable process in which a directory and, optionally, its subdirectories are monitored for any new or modified Windows Installer package files, and if changes are detected, those new or modified files are then either imported or reimported into the Application Catalog. See About Automatically Importing Packages from a Network Directory for more information.
Validation	The **Validation** Job Step verifies the internal data integrity of a Windows installer package.
Conflict Detection	The **Conflict Detection** Job Step performs conflict detection against selected packages.
Import	The **Import** Job Step imports a specified package (of any package type except Merge Modules) into the Application Catalog. An **Import** Job Step can only be included in a Custom Job, not a Job based on a Template.
Best Practice Detection	The **Best Practice ACEs** Job Step operates against a single package to enforce Microsoft Windows Installer standards.
Resolution	The **Resolution** Job Step performs automatic conflict resolution on one Windows Installer package or a set of packages. To configure the Resolution Job Step, you first select the packages to perform conflict resolution on, and then you select at least one resolution.

About Automatically Importing Packages from a Network Directory

In AdminStudio Application Manager, you can use the **Network Directory** option of **Package Auto Import** to monitor a directory location on a network (or a local directory) and automatically import or re-import packages in that directory at scheduled intervals.

You can use this **Network Directory** option in combination with the **Directory Monitoring** Job Step to establish a consistent, repeatable process in which a directory and, optionally, its subdirectories are monitored for any new or modified Windows Installer package files. If changes are detected, those new or modified files can then be either imported or reimported into the Application Catalog.

Note • *For more information, see Automatically Importing Packages from a Network Directory.*

Template Jobs vs. Custom Jobs

When you create a new Job, you can either base it on an existing Template that has pre-defined Job Steps, or build a Custom Job by choosing which Job Steps you want to perform and in what order. Each type of Job serves a particular purpose and provides certain benefits.

Template Jobs

Creating Jobs based on pre-defined Templates enables an organization to enforce consistent business practices. System Administrators can create a Template to define a series of Job Steps that they always want performed on a certain type of package before it is deployed. Then, when a package that falls into that category needs to be prepared for deployment, a Job based on that Template can be created and run.

For example, if all users of a specific department of an organization use the same disk image (Operating System and set of applications), all packages that are distributed in that department would need to have conflict analysis performed against that OS Snapshot and those target applications. The System Administrator could create a Template that uses those specific target applications, and could include very specific ICE and ACE rules in the conflict analysis, based upon requirements specific to that target environment.

In Template Jobs:

- **Target packages** are selected when a Template is created.
- **Source packages** are selected when a Job based on a Template is created.
- **An Import Job Step cannot be included**.
- **A Directory Monitoring Job Step can be included** to examine the specified directory for new or modified packages and then automatically import or reimport those packages into the Application Catalog. This is equivalent to using the Application Manager **Package Auto Import Wizard** to automatically import packages from a network directory. However, subsequent Job Steps cannot perform any actions on packages imported or reimported as part of a Directory Monitoring Job Step.

Note • *See About Automatically Importing Packages from a Network Directory for more information.*

Custom Jobs

When creating a new Custom Job, you can include only the Job Steps that you want to perform, in the order you want to perform them, using the source and target packages that you select when you create the Job. In Custom Jobs:

- **Both Source and Target packages are selected** when the Custom Job is created.
- **An Import or Directory Monitoring Job Step can be included**, and you can use packages imported as part of those steps as Source packages in subsequent Job Steps.

Because you can insert an **Import** or **Directory Monitoring** Job Step, a Custom Job is well-suited to automate a bulk import process.

Selecting Contextual Packages in a Custom Job

When defining a Custom Job, the packages that you import in prior **Import** or **Directory Monitoring** Job Steps are available for selection as source or target packages when defining operations in subsequent Job Steps.

The packages that will be imported in an **Import** or **Directory Monitoring** Job Step are selected when the Job is executed, not when the Job Step is defined, so the names of those packages are unknown.

Therefore to make these imported packages available for selection when defining subsequent Job Steps, they are listed in the **Contextual Packages** group in the package tree and are referred to in context of their associated import step, such as `Job Step` *n*, where *n* is the number of the Job Step that performed the import.

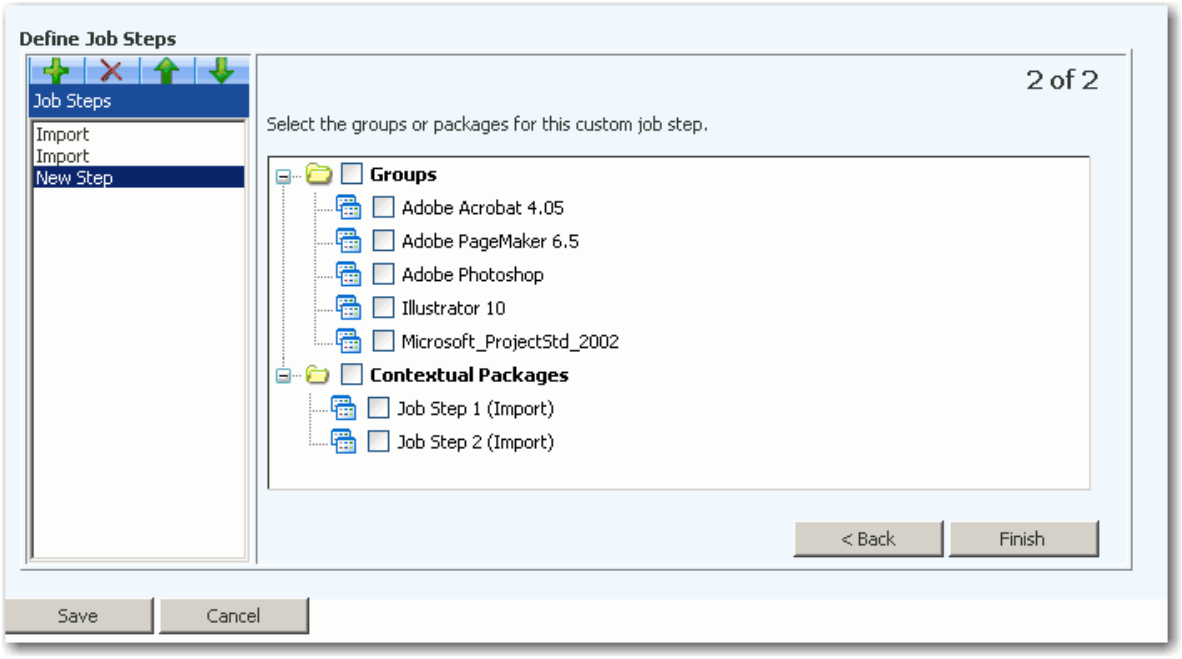


Figure 25-1: Contextual Packages Group

Dynamic Group Content Identification

You can create a Custom or Template Job that will perform actions on whatever packages are in a selected group *at the time the Job is executed.*

Therefore, you can schedule a reoccurring Job to automatically perform actions on the exact contents of a group without ever having to select any packages or create any new Jobs. This allows you to enforce consistent business practices on a group's contents even as the content changes.

For example, your organization could have a policy of importing all new packages into a specific group, such as a New Products Group. You could then choose to schedule a Job to run each night that would automatically perform conflict analysis and resolution using the packages in the New Products Group as the source packages and using selected packages in the Application Catalog as the target packages. In the morning, you could quickly see the current state of each package and continue the evaluation process.

Conflict Resolution and the Software Repository

If you include a **Conflict Detection** Job Step in a Job to identify conflicts, you can also include a **Resolution** Job Step to automatically resolve the conflicts that have associated Conflict Application Resolution Definitions (CARDs), and then reimport the modified package into the Application Catalog.

Resolution Behaviors

Because conflict resolution requires that the package be modified, if a package is stored in the Software Repository and is currently checked-out by another user, it cannot be modified, and therefore cannot be resolved. This resolution behavior is detailed in the following table.

Table 25-3 • Job Manager Resolution Behavior Regarding the Software Repository

Package State	Resolution Behavior
Package in Software Repository / Checked Out	Job Manager ignores the resolution process for these packages. The packages are already being edited by someone else.
Package in Software Repository / Not Checked Out	The package will be checked out by the virtual user **Job Manager Engine** prior to the normal operation of the resolution process. If the resolution process results in a no-operation or the user does not elect to re-import the package, then the check-out operation will be cancelled.
Package Not in Software Repository	The resolution process will operate normally.

Re-Import Behaviors

When defining a Resolution Job Step, you have to indicate the reimport option for resolved packages. When defining a **Resolution** Job Step, you have the choice of three re-import options for resolved packages: re-import as a new version, re-import by overwriting the existing version, or do not re-import.

The choice you make depends upon whether the package is stored in the Software Repository:

Table 25-4 • Resolved Package Re-Import Options

Resolved Packages Re-import options	Package in Software Repository	Package Not in Software Repository
As a new version	Package will be re-imported and checked-in as a new version.	N/A Package cannot be imported as a new version unless it is in the Software Repository.
By overwriting existing version	Package will be re-imported and checked-in, replacing the existing version.	Package will be re-imported, replacing the original package.
Do not import the resolved packages	Package will not be re-imported.	Package will not be re-imported.

User Permissions and Job Manager

The list of Job Step types that are available for you to select when creating a Job is controlled by the permissions of the Roles you are assigned to. For example, if you are not assigned to a Role that includes conflict resolution permission, the **Resolution** Job Step will not be available for your selection.

You can obtain access to the complete list of Job Step types by being assigned to additional Roles or having the Roles that you are assigned to modified to provide additional permissions.

Directory Monitoring and Job Manager

A **Directory Monitoring** Job Step can be included in both Template Jobs and Custom Jobs to examine a specified directory for new or modified packages and then to automatically import or reimport those packages into the Application Catalog. This is equivalent to using the Application Manager **Package Auto Import Wizard** to automatically import packages from a network directory.

Note • *See About Automatically Importing Packages from a Network Directory for more information.*

However, the use of these imported packages varies between Template Jobs and Custom Jobs:

- **Template Jobs** can include a **Directory Monitoring** Job Step to import packages, but subsequent Job Steps in that Job *cannot* perform any actions on any packages that are imported/reimported as part of that Directory Monitoring Job Step.

- **Custom Jobs** can include a **Directory Monitoring** Job Step, and you *can* use packages imported as part of that Directory Monitoring Job Step as source packages in subsequent Job Steps. See Selecting Contextual Packages in a Custom Job.

Managing Jobs

This section explains how to create and submit Jobs for execution. Topics include how to create and edit Template and Custom Jobs, reschedule a Job, set email alerts for a Job, and view Job status. The following topics are included:

- Creating a New Job
- Editing an Existing Job
- Rescheduling a Job
- Setting Email Alerts
- Viewing Job Status
- Deleting a Job

Creating a New Job

When creating a new Job to execute, you have the choice of either creating a Job based on a Template or a new Custom Job:

- **Template Job**—A Job based on a Template consists of a series of pre-defined Job Steps. See Creating a New Job Based on a Template.
- **Custom Job**—Allows you to include only the Job Steps that you want to perform, in the order you want to perform them, and using the source and target packages that you select. Custom Jobs can include the **Import** Job Step, while Template Jobs cannot. See Creating a New Custom Job.

Creating a New Job Based on a Template

A Template Job consists of a series of pre-defined Job Steps that are performed on the package(s) or group(s) you select from the Application Catalog.

To create a job based on a Template, perform the following steps.

Task: ***To create a Job based on a Template:***

1. Login to AdminStudio Enterprise Server.
2. Click the **Job Manager** tab. The **Jobs Queue** page opens.
3. Click the **Add** button. The **Add Job** page opens with the **Template** option selected.

4. From the **Template** list, select the appropriate Template.
5. In the **Name** field, enter a name for this job.
6. In the **Package List** tree, select the package(s) or group(s) that this Job will perform actions on. For example, if you chose a Job Template named *Scan_Dependencies_Template*, select the packages that you want to scan.

   If you choose a group, this Job's actions will be performed on all the packages in that group *at the time the Job is run*. With this **dynamic group content identification** feature, you can schedule a reoccurring Job to automatically perform actions on the exact contents of a group without ever having to select any packages or create any new Jobs.
7. **Schedule** this Job by selecting one of the following options from the Execution list:
   - **Immediate**—Execute this Job immediately.
   - **Scheduled**—Execute this Job at the **Date** and **Time** selected.

   Note • *You can also choose to set this job to reoccur at a specified interval. See Rescheduling a Job*
8. Click **Save**. The new Job is now listed on the **Jobs Queue** page and will be run at its scheduled time.

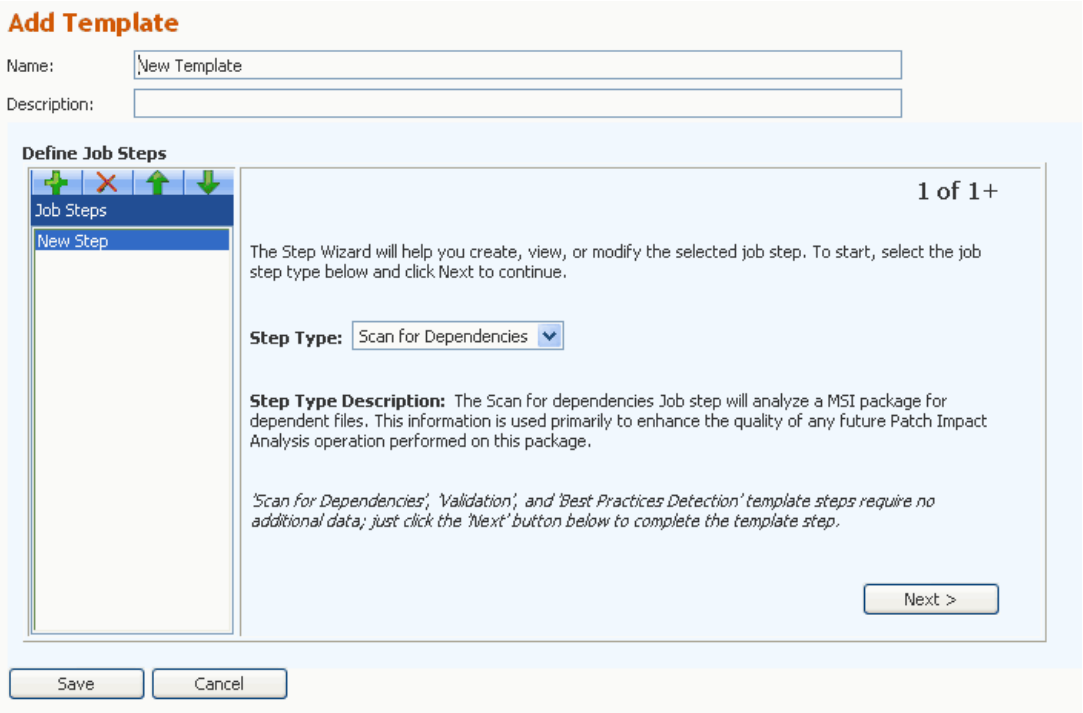
Creating a New Custom Job

When you create a Custom Job, you can include only the Job Steps that you want to perform, in the order you want to perform them, and using the source and target packages that you select. Custom Jobs can include the **Import** Job Step, while Template Jobs cannot.

To create a Custom Job, perform the following steps.

Task: ***To create a new Custom Job:***

1. Login to AdminStudio Enterprise Server.
2. Click the **Job Manager** tab. The **Jobs Queue** page opens.
3. Click the **Add** button. The **Add Job** page opens with the **Template** option selected.
4. Enter a **Name** for this new job.
5. Clear the **Template** option. An empty **Define Job Steps** area opens.



6. Select one of the listed tasks from the **Step Type** list:

   - **Scan for Dependencies**—Analyze a Windows Installer package for dependent files.
   - **Directory Monitoring**—Use to establish a consistent, repeatable process in which a directory and, optionally, its subdirectories are monitored for any new or modified Windows Installer package files, and if any changes are detected, those new or modified files are then either imported or re-imported into the Application Catalog.
   - **Validation**—Verifies the internal data integrity of a Windows Installer package.
   - **Conflict Detection**—Performs conflict detection on selected packages using Conflict ACE Rules
   - **Import**—Import a selected package into the Application Catalog.
   - **Best Practice Detection**—Performs conflict detection on a single package using Best Practice ACE Rules to enforce Windows Installer standards.
   - **Resolution**—Performs automatic conflict resolution on selected packages.

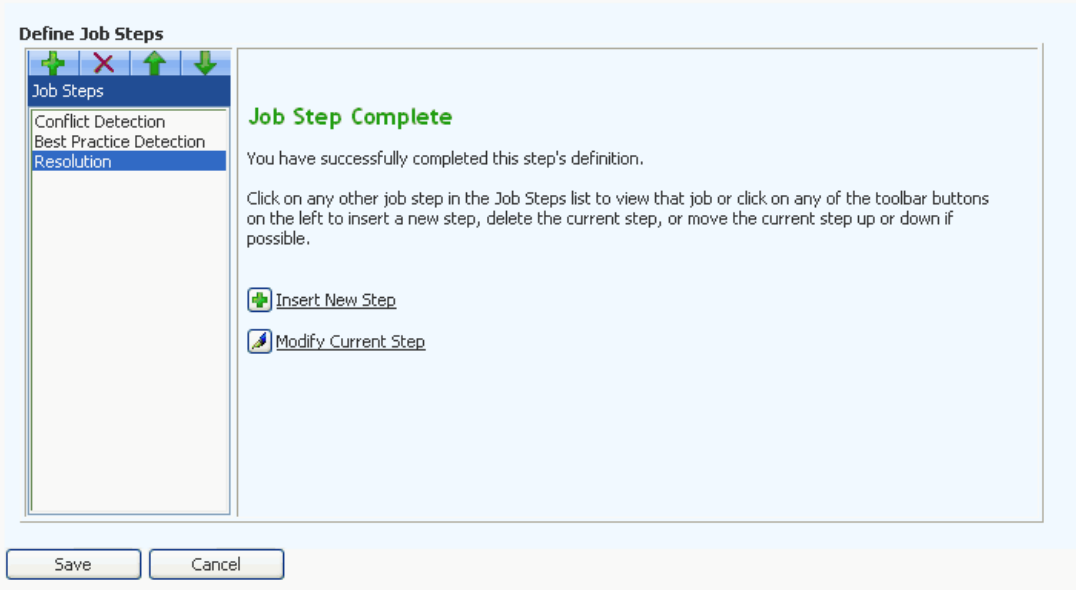
   Each time you select a **Step Type**, you are prompted to specify additional information:

Step Type	Required Selections
Scan for Dependencies	1. Click **Next**. The **Step 2 of 2** View opens. 2. Select the packages or groups of packages that you want to scan, and then click **Finish** to save this Job Step.

Step Type	Required Selections
Directory Monitoring	1. Click **Next**. The **Step 2 of 2** View opens. 2. In the **Directory Name** box, enter the UNC path of the directory that you want to monitor. 3. Choose whether to **Include subdirectories in the 'Directory Monitoring' process**. 4. Click **Finish** to save this Job Step.
Validation	1. Click **Next**. The **Step 2 of 2** View opens. 2. Select the packages or groups of packages that you want to validate. 3. Click **Finish** to save this Job Step.
Conflict Detection	1. Click **Next**. The **Step 2 of 4** vew opens. 2. Select the source packages or groups of packages that you want to include in the conflict detection process, and then click **Next**. The **Step 3 of 4** View opens. 3. Select the target packages or groups of packages that you want to use in the conflict detection process and click **Next**. The **Step 4 of 4** View opens. 4. Select the set of Conflict ACE Rules that you want to include in the conflict detection process and click **Finish** to save this Job Step.
Import	1. Click **Next**. The **Step 2 of 4** View opens. 2. Select a package to import. 3. Specify whether to include the package in the Software Repository 4. Click **Next**. The **Step 3 of 4** View opens. 5. Select any transform (`.mst`) or patch (`.msp`) files to import. 6. Click **Next**. The Step 4 of 4 View opens. 7. Select one or more destination groups into which the package will be imported. 8. Click **Finish** to save this Job Step.
Best Practice Detection	1. Click **Next**. The **Step 2 of 2** View opens. 2. Select the packages or groups of packages that you want to perform Best Practice conflict detection on. 3. Click **Finish** to save this Job Step.

Step Type	Required Selections
Resolution	1. Click **Next**. The **Step 2 of 3** View opens. 2. Select the packages or groups of packages that you want to perform conflict resolution on, and then click **Next**. The **Step 3 of 3** View opens. 3. Select the Conflict ACE Rules that you want to resolve automatically. 4. Select a **Resolved packages re-import option**. See Re-Import Behaviors. 5. Click **Finish** to save this Job Step.

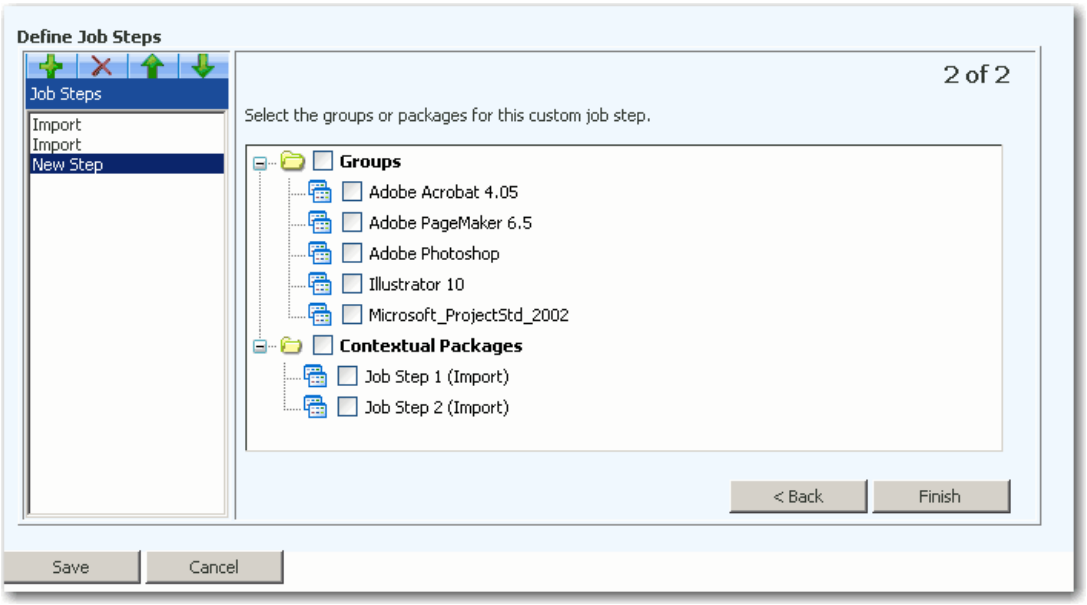
7. Each time you click **Finish**, the new Job Step is saved and you are prompted to add another Job Step.



8. Continue adding desired Job Steps by clicking **Insert New Step**.

9. If your Job includes **Import** or **Directory Monitoring** Job Steps, you can perform actions on those imported or reimported packages in subsequent Job Steps. For example, if you add a **Validation** Job Step after an **Import** or **Directory Monitoring** Job Step, you can choose to validate the newly imported/reimported package or packages from the previous Steps.

   To make these packages available for selection when defining subsequent Job Steps, they are listed in the **Contextual Packages** group in the package tree and are referred to in context of their associated step, such as `Job Step` *n*, where *n* is the number of the Job Step that performed the import.



Note • *During a **Directory Monitoring** Job Step, one or more packages in the monitored directory may be imported or reimported into the Application Catalog. Therefore, if you select a **Directory Monitoring** Job Step as a Source for a subsequent Job Step, that Step's action will be performed on all of the newly imported or reimported packages.*

10. **Schedule** this Job by selecting one of the following options from the Execution list:
    - **Immediate**—Execute this Job immediately.
    - **Scheduled**—Execute this Job at the **Date** and **Time** selected.

Note • *You can also choose to set this job to reoccur at a specified interval. See Rescheduling a Job*

11. Click **Save**. The new Job is now listed on the **Jobs Queue** page and will be run at its scheduled time.

Editing an Existing Job

You can edit an existing Job by selecting the Job on the **Jobs Queue** page to open the **Edit Job** page. The following information can be edited:

Table 25-5 • Editable Items in an Existing Job

Job Step Type	Editable Items
General	• Job Name • Execution options • Step Type
Scan for Dependencies Job Step	• Selected packages and groups
Directory Monitoring Job Step	• Directory to monitor • Option to monitor subdirectories
Validation Job Step	• Selected packages and groups
Conflict Detection Job Step	• Selected source packages and groups • Selected target packages and groups • Set of conflict detection rules to use during the conflict detection process
Import Job Step	• Package to import • Software Repository option • Transform (.mst) and Patch (.msp) files to include during import • Destination group for imported package
Best Practice Detection Job Step	• Selected packages and groups
Resolution Job Step	• Selected packages and groups • Conflicts to resolve • Resolved package re-import options

To edit an existing Job, perform the following steps:

Task: ***To edit an existing Job:***

1. Login to AdminStudio Enterprise Server.
2. Click the **Job Manager** tab. The **Jobs Queue** page opens.

3. Select the Job that you want to edit. The **Edit Job** page opens.
   - Only one Job can be edited at a time.
   - You cannot edit a Job with a **Status** of **Running** or **Pending**.
   - You cannot change a **Custom Job** to a **Template Job** or vice versa.
4. Edit the Job following the instructions in Creating a New Job Based on a Template or Creating a New Custom Job.
5. When you have finished editing the Job, click **Save**.

Rescheduling a Job

You can reschedule a completed Job so that it is run again. You can also reschedule Jobs that failed, were cancelled, or are scheduled to be run in the future.

If a Job is in **Scheduled**, **Complete**, **Failed**, or **Cancelled** Status, open the **Jobs Queue** page and click the link in the **Schedule** column for that Job. The **Schedule Job** page opens.

Task: ***To reschedule a job:***

1. On the **Jobs Queue** page, click the date and time listed in the **Schedule** column for the Job you want to reschedule. The **Schedule Job page** opens.
2. Select either **For Immediate Execution** or **For Scheduled Execution**.
   - If you selected **For Immediate Execution**, click **Save** and your Job will be executed immediately.
   - If you selected **For Scheduled Execution**, continue with step **4**.
3. Click the calendar icon next to the **Select Date** field and select the date when you want the Job to be run.
4. In the **Enter Time** fields, select an hour of the day from the first list, select the minutes after the hour from the second list, and select **AM** or **PM** from the third list.
5. If you want this Job to reoccur at a specified interval, set the **Recurrence Interval** option to **Daily**, **Weekly**, or **Monthly**, and perform the following steps:
   a. If you want this Job to reoccur a specific number of times, select the N**umber of Times** option, and enter a number in the box.
   b. If you want this Job to reoccur until a specified expiration date, select the **Expiration Date** option, click the calendar icon, and select a date from the calendar.
6. Click **Save**.

Setting Email Alerts

You can choose to send an email Alert when a Job is completed and when it fails. To do this, open the **Jobs Queue** page, and click **Add** in the **Alerts** column for the Job that you want to set an alert for. You then can create an email message on the **Job Alerts** page.

Tip • *You can choose to send a Job alert upon both Job failure and Job success.*

When composing the **Subject** line and the **Message**, you can also insert System Variables to instruct Job Manager to automatically insert the variable data regarding that Job at runtime.

Task: ***To set email alerts:***

1. On the **Jobs Queue** page, click **Add** in the **Alerts** column for the Job that you want to set an alert for. The **Job Alerts** page opens.
2. From the **Send the following e-mail alert when the job** list, select **fails** or **succeeds**.
3. In the **To** field, enter the email addresses that you want to be notified when a Job is completed or when it fails. Separate multiple email addresses with a semicolon (**;**).
4. In the **Subject** box, enter text to describe the purpose of the email. You can select System Variables from the list so that the value of the variable for this Job is inserted when the email is sent.
5. In the **Message** box, enter the text of the message. You can select System Variables from the list so that the value of the variable for this Job is inserted when the email is sent.
6. Click **Save**.

System Variables

The following system variables are available to use in email alerts:

Table 25-6 • System Variables Available to Use in Email Alerts

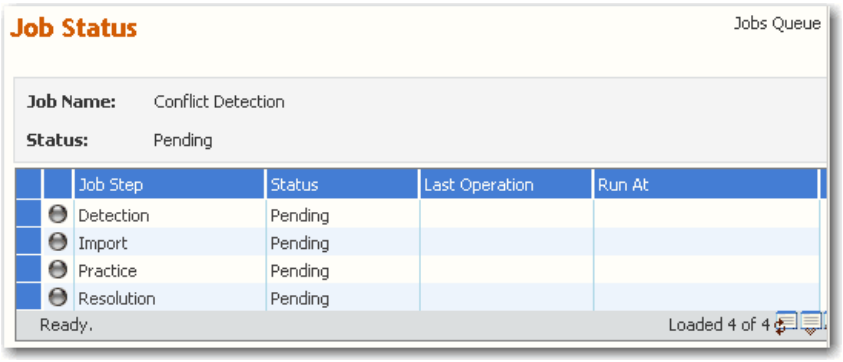
Variable	Description
QueueName	Name of Job.
LastAction	The last Job Step that was completed for this Job. If a Job's status is Running, this is the Job Step that is currently being executed. If a Job's status is Complete, this is the last Job Step of the Job. If a Job Failed, this is the Job Step that was running when the failure occurred.
CompletionTime	Time job was completed.
UserName	Person who submitted the Job.

Table 25-6 • System Variables Available to Use in Email Alerts

Variable	Description
JobName	Name of Template.
JobType	Number of Job Steps defined in the Template.
Description	Description that was entered to identify the Template.
ProductVersion	MSI specific property from package.
ProductName	MSI specific property from package.
Manufacturer	MSI specific property from package.
ProductLanguage	MSI specific property from package.
FileName	Base file path (either “Managed from the Software Repository” or appropriate path).
Comments	Product comments.
ImportDate	Date package was imported into the Application Catalog.
DisplayedProductName	Product name as displayed in Application Manager/ConflictSolver.
OriginalMsiFileName	Name of `.msi` file that was imported into the Application Catalog.
OriginalPackageLocation	Location of the original `.msi` file at the time it was imported into the Application Catalog.

Viewing Job Status

On the **Jobs Queue** page, a Job’s current status is listed as a hyperlink in the **Status** column. If you click on this hyperlink, the **Job Status** page opens, displaying the Job Name, the Job Status, and the status of each of the Job Steps in that job.



The possible statuses are:

- **Scheduled**—Job is scheduled to be executed at a future date and time.
- **Running**—Job is currently being run. Only one job per Application Catalog connection will ever be running at one time.
- **Pending**—Job is available to run but is waiting for the running job to complete. A job set for immediate execution will be automatically assigned a Pending job status.
- **Complete**—Job has been executed.
- **Canceled**—Job was canceled before it was executed or before its execution was complete.
- **Failed**—Job was unable to run either because of a software error, a network communication problem, a hardware failure, or any other reason.

Deleting a Job

You can delete a Job by opening the **Jobs Queue** page and clicking **Delete** in the **Action** column of the Job you want to delete.

Note • *Deleting a Template Job in no way effects the Job Template that it was based on.*

Task: ***To delete a Job:***

1. Login to AdminStudio Enterprise Server.
2. Click the **Job Manager** tab. The **Jobs Queue** page opens.
3. Click **Delete** in the **Action** column of the Job that you want to delete. The **Delete Job** page opens and you are prompted to confirm the deletion.

   Note • *You cannot delete a Job with a **Status** of **Running**.*

4. Click **Delete**. The Job is no longer listed on the **Jobs Queue** page.

Managing Templates

Creating Jobs based on pre-defined Templates enables an organization to enforce consistent business practices. System Administrators can create a Template to define a series of Job Steps that they always want performed on a certain type of package before it is deployed. Then, when a package that falls into that category needs to be prepared for deployment, a Job based on that Template can be created and run.

For example, if all users of a specific department of an organization use the same disk image (Operating System and set of applications), all packages that are distributed in that department would need to have conflict analysis performed against that OS Snapshot and those target applications. The System Administrator could create a Template that uses those specific target applications, and could include very specific ICE and ACE rules in the conflict analysis, based upon requirements specific to that target environment.

This section includes the following topics:

- Source and Target Packages in Template Jobs vs. Custom Jobs
- Creating a New Template
- Editing an Existing Template
- Deleting a Template

Source and Target Packages in Template Jobs vs. Custom Jobs

The selection of source and target packages in a Template Job differs from how source and target packages are selected in Custom Jobs.

For Template Jobs:

- **Target packages** are selected when a Template is created.
- **Source packages** are selected when a Job based on a Template is created.
- **An Import Job Step cannot be included**. All actions must be performed on packages that have already been imported into the Application Catalog.

For Custom Jobs:

- **Both Source and Target packages** are selected when the Custom Job is created.
- **An Import Job Step can be included**, and you can use newly imported packages as Source packages in subsequent Job Steps.

Creating a New Template

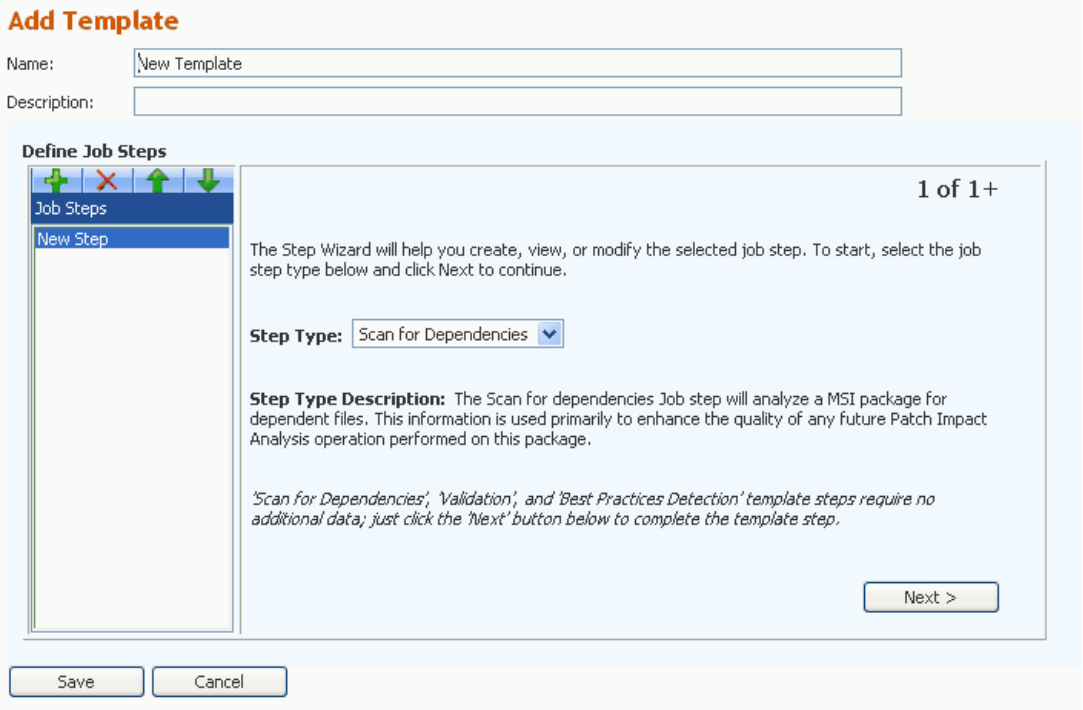
You create a Template much like the steps described in Creating a New Custom Job, except:

- You do not **Schedule** a Template
- You do not set **Alerts** for a Template
- You cannot include an **Import** Job Step in a Template.

You create Templates on the **Manage Templates** tab. The **Job Template List** page lists all defined Templates. From this page, you can **Add** a new Template, **Edit** an existing Template, or **Delete** a Template.

Task: ***To create a new Template:***

1. Login to AdminStudio Enterprise Server.
2. Click the **Job Manager** tab. The **Jobs Queue** page opens.
3. Click the **Manage Templates** subtab. The **Job Template List** page opens.
4. Click the **Add** button. The **Add Template** page opens, displaying an empty **Define Job Steps** area.



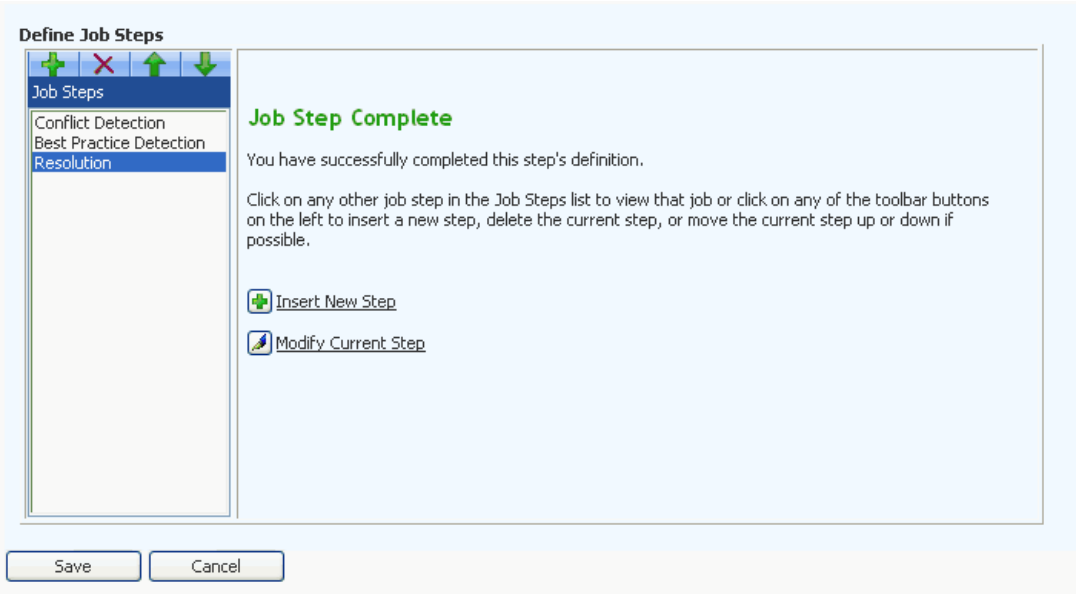
5. Enter a **Name** for this new Template.
6. Enter a **Description** to define the purpose of this new Template.
7. Select one of the listed tasks from the **Step Type** list:

- **Scan for Dependencies**—Analyze a Windows Installer package for dependent files.
- **Directory Monitoring**—Use to establish a consistent, repeatable process in which a directory and, optionally, its subdirectories are monitored for any new or modified Windows Installer package files, and if any changes are detected, those new or modified files are then either imported or re-imported into the Application Catalog.
- **Validation**—Verifies the internal data integrity of a Windows Installer package.
- **Conflict Detection**—Performs conflict detection on selected packages using Conflict ACE Rules
- **Best Practice Detection**—Performs conflict detection on a single package using Best Practice ACE Rules to enforce Windows Installer standards.
- **Resolution**—Performs automatic conflict resolution on selected packages.

After you select a **Step Type**, you may be prompted to specify additional information. Make the required selections, as listed in the following table, and click **Finish** to save the Step.

Step Type	Required Selections
Scan for Dependencies	Click **Next** to save this Job Step. (Source packages will be selected when a Job based on this Template is created.)
Directory Monitoring	1. Click **Next**. The **Step 2 of 2** view opens. 2. In the **Directory Name** box, enter the UNC path of the directory that you want to monitor. 3. Choose whether to **Include subdirectories in the 'Directory Monitoring' process**. 4. Click **Finish** to save this Job Step.
Validation	Click **Next** to save this Job Step. (Source packages will be selected when a Job based on this Template is created.)
Conflict Detection	1. Click **Next**. The **Step 2 of 3** view opens. 2. Select the target packages or groups of packages that you want to use in the conflict detection process and click **Next**. The **Step 3 of 3** view opens. 3. Select the set of Conflict ACE Rules that you want to include in the conflict detection process and click **Finish** to save this Job Step.
Best Practice Detection	Click **Next** to save this Job Step. (Source packages will be selected when a Job based on this Template is created.)
Resolution	1. Click **Next**. The **Step 2 of 2** view opens. 2. Select the Conflict ACE Rules that you want to resolve automatically. 3. Select a **Resolved packages re-import option**. See Re-Import Behaviors. 4. Click **Finish** to save this Job Step.

8. Each time you click **Finish**, the new Job Step is saved and you are prompted to add another Job Step.



9. Continue adding desired Job Steps.
10. When you have finished adding Job Steps, click **Save**. The Template is now listed on the **Job Template List** page and in the **Template** list on the **Add Job** page.

Editing an Existing Template

You can edit an existing Template by selecting the Template on the **Job Template List** page to open the **Edit Template** page.

Task: ***To edit an existing Template:***

1. Login to AdminStudio Enterprise Server.
2. Click the **Job Manager** tab. The **Jobs Queue** page opens.
3. Click the **Manage Templates** tab. The
4. Select the Template you want to edit. The **Edit Template** page opens.

Note • *Only one Template can be edited at a time.*

5. Edit the Template following the instructions in Creating a New Template.
6. When you have finished editing the Template, click **Save**.

Deleting a Template

You can delete a Template by opening the **Job Template List** page and clicking **Delete** in the **Action** column of the Template you want to delete.

Only Templates which are not in use are allowed to be deleted.

- A Template is considered in use as long as a Template Job associated with the Template exists.
- If a Template Job associated with this Template is scheduled to be run in the future or if it is already completed, you will not be permitted to delete the Template.

Task: ***To delete a Template:***

1. Login to AdminStudio Enterprise Server.
2. Click the **Job Manager** tab. The **Jobs Queue** page opens.
3. Click the **Manage Templates** tab. The **Job Template List** page opens.
4. Click **Delete** in the **Action** column of the Template that you want to delete. The **Delete Template** page opens, prompting you to confirm the deletion.
5. Click **Delete**. The Template is no longer listed on the **Job Template List** page.

Setting Job Manager Conflict Detection Options

You select the rules that you want Job Manager to use by default during conflict detection on the **Conflict Detection Options** page of the **Configuration tab**.

Note • *By setting these Conflict Detection options, you are only setting the defaults for the Conflict Detection Job Step in a Job. When you are defining the Conflict Detection Job Step, you can change these selections.*

Task: ***To set Detection options:***

1. Login to AdminStudio Enterprise Server.
2. Click the Job Manager tab. The **Jobs Queue** page opens.
3. Click the **Configuration** subtab. The **Conflict Detection Options page** opens.

4. In the **Conflict Types** tree, select the check box for ACE rules that you want to be performed during conflict identification. Conflict rules (ACEs) associated with unselected boxes will not be performed during conflict identification. The following rules are available:

Category	ACEs
Components	• **ACE30**—A check for KeyPath conflicts across components. • **ACE02**—Check destination directories for identical components. • **ACE09**—Check for different merge module versions.
File Extensions	• **ACE17**—Confirm that file extensions exist in only one component.
Files	• **ACE23**—Identifies file duplication between source and target packages. • **ACE12**—Checks for components that contain files that can be replaced by one of the merge modules that you have imported into the Application Catalog database. • **ACE08**—identifies components with identical ComponentIds, and checks those components to see if the versions of the files in each component match. • **ACE07**—Checks for the existence of the same file in components with different ComponentIds. • **ACE03**—Checks to see if components in different packages that have matching ComponentIds also contain the same files.
INI Files	• **ACE22**—Checks file name/target directory pairs in the File table to see if they conflict with similar entries in the IniFile table. • **ACE21**—Checks entries in the IniFile table to see if they conflict with similar entries in the File table. • **ACE14**—Checks for the existence of components with different ComponentIds that modify the same INI file entry, such as the [File Name/Section/Key/Value] entry.
NT Services	• **ACE16**—Checks for the existence of identical NT Services in components with different ComponentIds.
ODBC Resources	• **ACE15**—Checks for the existence of identical ODBC entries in components with different ComponentIds.
Product Properties	• **ACE20**—Checks the Upgrade Code to see if it is unique. • **ACE19**—Checks the Product Code to see if it is unique. • **ACE18**—Checks the Package Code to see if it is unique.
Registry	• **ACE24**—Checks to see if registry entries with the same registry hive, key, and value name have the same data type and value. • **ACE10**—Checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds.

Category	ACEs
Shortcuts	• **ACE13**—Checks for the existence of the same shortcut within different packages in components with different ComponentIds.
Windows Terminal Server Validation	• **WST01**—Checks to see if the ALLUSERS property is defined. • **WST02**—Checks for any registry entries installed to per-user specific locations. • **WST03**—Checks for any file entries installed to per-user specific locations. • **WST04**—Checks for any ODBC data source entries installed to per-user specific locations. • **WST05**—Checks for any environment strings installed to per-user specific locations.

5. Click **Save**.

Managing Jobs Using Job Manager Engine

Job Manager Engine executes the Jobs created with Job Manager. Job Manager Engine can connect to any of the Application Catalogs in your network and allow you to configure how the engine operates and manage the state of any particular Application Catalog.

This section includes the following topics:

- About Job Manager Engine
- Opening the Job Manager Engine User Interface
- Adding an Application Catalog
- Deleting a Connection to an Application Catalog
- Executing Jobs
- Pausing, Shutting Down, and Restarting Application Catalog Jobs
- Setting Configuration Options

About Job Manager Engine

The Job Manager Engine's **Manage Application Catalogs View** lists the **Catalog Name** of each of the Application Catalogs that it is connected to, the **Status** (Running, Pending, Paused) of each Application Catalog, and the **# of Jobs** from that Application Catalog that are currently running or waiting to run.

Opening the Job Manager Engine User Interface

In order to execute jobs, Job Manager Engine needs to be running somewhere in the network. Because Jobs are often scheduled to be run overnight, Job Manager Engine usually needs to be running continuously. There are two ways to launch Job Manager Engine:

- **Launch at startup**—The first time that Job Manager Engine launches, you are prompted to select whether you want to automatically launch Job Manager Engine at startup. If you choose this option, Job Manager Engine launches, but then is automatically minimized, and its icon is shown in the Windows system tray.
- **Select shortcut on Start menu**—You can open Job Manager Engine by selecting it from the **AdminStudio 10.0 SP2 Tools** subfolder on the **Start** menu. However, when it is launched, it is still automatically minimized, and its icon is shown in the Windows system tray.

If you want to view the Job Manager Engine interface, click on the Job Manager Engine system tray icon. The Manage Application Catalogs View of the Job Manager Engine opens.

For more information on the Job Manager Engine interface, see Job Manager Engine.

Note • *While everyone who installs AdminStudio Enterprise Server gets the Job Manager Engine, only one Job Manager Engine can be connected at any one time to any one Application Catalog.*

Adding an Application Catalog

To add an Application Catalog to Job Manager Engine, perform the following steps.

Task: ***To add an Application Catalog to Job Manager Engine:***

1. Open Job Manager Engine.
2. On the Manage Application Catalogs View, click the **Add** button. The Select Application Catalog dialog box opens.

Deleting a Connection to an Application Catalog

When you delete an Application Catalog from Job Manager Engine, it will first shut down the Application Catalog, allowing any currently executing Job Steps to complete. Therefore, this operation may take some time to complete.

Task: ***To delete the connection of Job Manager Engine to an Application Catalog***

1. Open Job Manager Engine.
2. On the Manage Application Catalogs View, select the Application Catalog that you want to delete and click the **Delete** button. You are prompted to confirm the deletion.
3. Click **Yes**. The Application Catalog is no longer listed.

Executing Jobs

When Job Manager Engine opens, it automatically identifies all of the Jobs that are in **Pending** status. Then, as the scheduled time interval of each Pending Job is reached, Job Manager Engine executes it.

Job Manager Engine can be connected to multiple Application Catalogs at once, and multiple Job Manager Engines may run on the network. However an Application Catalog can only be connected to one Job Manager Engine at a time.

Pausing, Shutting Down, and Restarting Application Catalog Jobs

From the Manage Application Catalogs View of Job Manager Engine, you can Pause, Shutdown, and Restart Jobs in an Application Catalog:

Table 25-7 • Pausing, Shutting Down, and Restarting Application Catalog Jobs

Button	Action That is Taken by Job Manager Engine
Pause	Click to pause the execution of the Jobs in the selected Application Catalog. Clicking the **Pause** button tells Job Manager Engine not to start the execution of any new Jobs in the selected Application Catalog, but allow all currently executing jobs to complete normally. Job Manager Engine remains fully aware of the Application Catalog, but does not use any CPU resources to run any Jobs. Therefore, the **Pause** operation could be used by someone to temporarily give another Application Catalog better access to system resources so that it could complete its jobs faster. When you select a **Paused** Application Catalog and click **Run/Restart,** its Pending Jobs begin executing immediately.

Table 25-7 • Pausing, Shutting Down, and Restarting Application Catalog Jobs

Button	Action That is Taken by Job Manager Engine
Shutdown	Click to stop all Jobs on the selected Application Catalog. Any currently running Job Step is allowed to finish, but no additional Jobs will execute. Clicking **Shutdown** tells Job Manager Engine to fail any currently running Jobs and to unload itself from memory. All resources used by Job Manager Engine for this Application Catalog are freed. When you select a **Shutdown** Application Catalog and click **Run/Restart**, the restart operation will need to relaunch and initiate the background thread for the Application Catalog, and therefore a **Shutdown** Application Catalog takes longer to restart than a **Paused** Application Catalog.
Run/Restart	Click to initiate execution of Jobs from a selected Application Catalog that was Paused or Shutdown.

When you select an Application Catalog, an icon is displayed to indicate its status:

Table 25-8 • Job Manager Engine Status Icons

Icon	Status
	All Pending Jobs and any future Jobs sent from this Application Catalog will be executed.
	The **Pending** Jobs from this Application Catalog will not be executed until the connection is restarted by clicking the **Run/Start** button.
	No Jobs from this Application Catalog will be executed until the Application Catalog is restarted by clicking the **Run/Start** button. All Jobs from this Application Catalog that were in **Pending** status when it was shutdown were **Failed** and must be resubmitted to execute.

Viewing Job Manager Engine Status in Job Manager

The status (**Active** or **Not Active**) of the Job Manager Engine associated with an installation of Job Manager is displayed in the **Job Manager Engine Status** area of the left pane of all of the Job Manager pages:

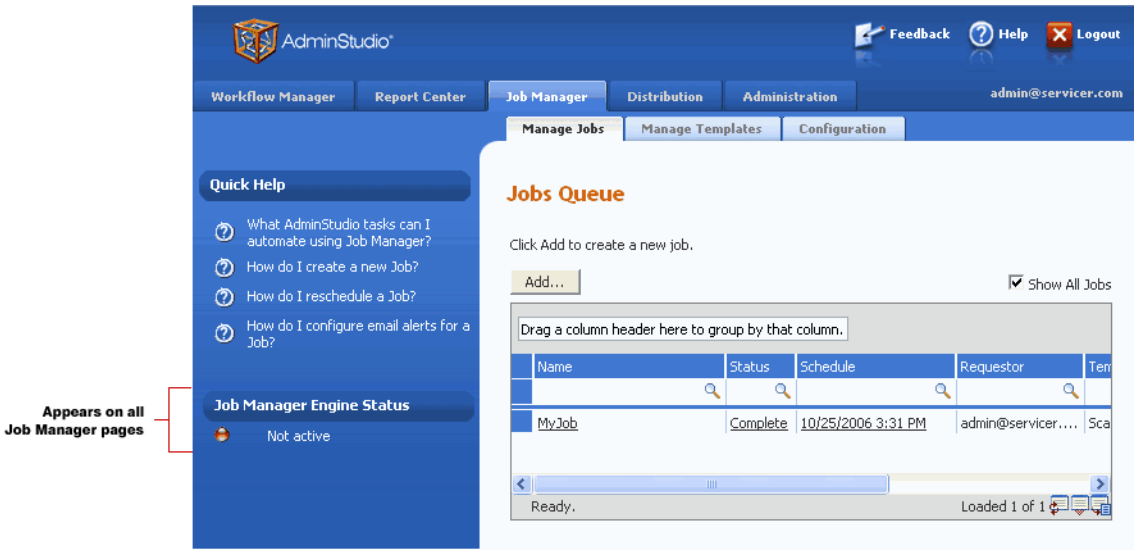


Figure 25-2: Job Manager Engine Status Displayed in the Job Manager User Interface

Setting Configuration Options

Usually the Job Manager Engine is installed on the same Server as the AdminStudio client tools, and the Job Manager Configuration options are set on the **Rules** tab and **Validate** tab of the ConflictSolver **Options** dialog box.

However, if you choose to install Job Manager Engine on a different server, you can set the following configuration options from the Job Manager Engine interface:

- Conflict Detection Configuration
- Validation Configuration
- Email Alerts Configuration

Conflict Detection Configuration

On the **Conflict Detection Configuration View**, you can set the following configuration options:

Table 25-9 • Conflict Detection View Options

Option	Description
ACE File	File containing the pre-defined ACE rule information for all ConflictSolver Conflict and Best Practice ACEs. It is installed in the following location on the machine where AdminStudio is installed: *AdminStudio Installation Directory* \ConflictSolver\Support

Table 25-9 • Conflict Detection View Options

Option	Description
Custom ACE File	You use user-defined custom ACEs to extend the functionality of pre-defined ACEs with company-specific functionality. By selecting different user-defined ACE files, you can organize rules appropriate for individual users in your organization. The user-defined ACE file specified here is run after the pre-defined ACE rules are run. The selection of this user-defined ACE file will affect the default Conflict Types displayed on the Job Manager interface when you are creating Job Steps. By default, an empty user-defined ACE file is installed in the following location on the machine where AdminStudio is installed. `C:\AdminStudio Shared\ConflictSolver\ CustomConflictFile.ace` Only one user-defined ACE file can be active at one time.

Validation Configuration

On the **Validation Configuration View**, you can set the following configuration options:

Table 25-10 • Validation View Options

Option	Description
CUB File	Validation involves comparing a Windows Installer package against a known criteria to identify deviations from those rules. By default, ConflictSolver compares packages against the Full MSI Validation Suite. This suite contains a comprehensive set of Internal Consistency Evaluators (ICEs)—guidelines created by Microsoft to ensure an installation package works correctly with the Windows Installer engine. The file specified in this field is file containing Internal Consistency Evaluators (ICEs) used for validation. The CUB file is configured as a part of the Job Server Engine installation, so normally you do not need to modify this entry.

Email Alerts Configuration

On the **E-mail Alerts View**, you can enter configuration settings so that email alerts can be sent out to report the status of a completed or failed job.

Table 25-11 • E-mail Alerts View Options

Option	Description
SMTP Mail Server Name	Enter the SMTP mail server the Job Manager Server should use when sending messages. Enter the MTP Mail Server in the format of: `mail.mycompany.com`

Table 25-11 • E-mail Alerts View Options

Option	Description
Senders E-mail Address	Enter the E-mail Address of the person who you want to be displayed as the email sender.
SMTP Server Port	When someone sends a mail message to your SMTP Server, the mail will attempt to come in to your server on Port **25**. This is the standard SMTP (Simple Mail Transport protocol) port. Port 2525 is an alternate port used by forward services. This is a non-standard port, but useful if the standard SMTP port is blocked

Job Manager Reference

Job Manager Reference includes the same topics displayed when you select help from the Job Manager or Job Manager Engine interface. Reference information for Job Manager is organized into the following areas:

Table 25-12 • Job Manager Reference Organization

Page	Views
Jobs Queue Page	• Add Job Page • Edit Job Page • Delete Job/Template Page • Schedule Job Page • Job Alerts Page • Job Status Page
Job Template List Page	• Add Template Page • Edit Template Page
Conflict Detection Options Page	
Job Manager Engine	• Manage Application Catalogs View • Conflict Detection Configuration View • Validation Configuration View • E-mail Alerts View

Jobs Queue Page

The **Jobs Queue** page lists all defined Jobs. From this page you can choose to add a new Job, edit an existing Job, delete a Job, schedule a job, or define the email alerts that will be sent out when this job is executed.

The following information and options are available:

Table 25-13 • Jobs Queue Page Options

Option	Description
Add Button	Click to open the Add Job Page where you can create a new Job.
Name	Name of Job
Status	Current status of the Job in the Job Execution process: • **Scheduled**—Job is scheduled to be executed at a future date and time. • **Running**—Job is currently being run. Only one job per Application Catalog connection will ever be running at one time. • **Pending**—Job is available to run but is waiting for the running job to complete. A job set for immediate execution will be automatically assigned a **Pending** job status. • **Complete**—Job has been executed. • **Canceled**—Job was canceled before it was executed or before its execution was complete. • **Failed**—Job was unable to run either because of a software error, a network communication problem, a hardware failure, or any other reason. The values in the **Status** column are also hyperlinks. Click on a status value to open the Job Status Page that lists the current status of each Job Step in this Job.
Schedule	Lists the scheduled date and time for this job. Click this link to open the Schedule Job Page where you can specify when to run this job and also configure the selected Job to reoccur over a specified interval.
Requestor	User who created this Job.
Template	Template that was used to create the Job. If no Template is listed, it is a Custom Job.
Alerts	Click the link in this column to open the Job Alerts Page where you can add or edit an email to send when the selected Job is completed and when it fails.
Action	Click **Delete** in this column for a Job you want to delete.

Add Job Page

The **Add Job** page includes the Template View and the Custom Job View.

Template View

On the **Job Template** View, you create a new Job based on a pre-existing Job Template. The **Job Template** View contains the following options:

Table 25-14 • Add Job Page / Job Template View Options

Option	Description
Job Name	Enter a name to identify the Job.
Template	Select this option and select a Template from the list to create a Template Job. .
Package List	Select the package(s) or group(s) to be used by the Job Template. If you select a group, the Job will perform actions on whatever packages are in the selected group at *the time the Job is executed*.
Execution	Use the following options to schedule this Job: • **Immediate**—Select to execute this Job immediately. • **Scheduled**—Select to execute this Job at the **Date** and **Time** you select.

Custom Job View

The Custom Job View contains the following options:

Table 25-15 • Add Job Page / Custom Job View Options

Option	Description
Name	Enter a name to identify the Job.
Template	To create a Custom Job, clear this selection. When this option is not selected, the **Define Job Steps** area opens.

Table 25-15 • Add Job Page / Custom Job View Options (cont.)

Option	Description
Define Job Steps	You create Job Steps using this area. To create the first Job Step, make a selection from the **Step Type** list and click **Next**. To create all subsequent Job Steps, click **Insert New Step**. The following Job Step Types are available: • **Import**—Import a selected package into the Application Catalog. • **Scan for Dependencies**—Analyze a Windows Installer package for dependent files. • **Directory Monitoring**—Use to establish a consistent, repeatable process in which a directory and, optionally, its subdirectories are monitored for any new or modified Windows Installer package files, and if any changes are detected, those new or modified files are then either imported or re-imported into the Application Catalog. See About Automatically Importing Packages from a Network Directory for more information. • **Validation**—Verifies the internal data integrity of a Windows Installer package. • **Conflict Detection**—Performs conflict detection on selected packages using Conflict ACE Rules • **Best Practice Detection**—Performs conflict detection on a single package using Best Practice ACE Rules to enforce Windows Installer standards. • **Resolution**—Performs automatic conflict resolution on selected packages. For each type of Job Step, you are prompted to specify additional information. See Information Requested When Defining Job Steps in a Custom Job. Use the **Up** and **Down** arrows to rearrange the order of the Job Steps, and use the **Delete** button to delete a Job Step. Click **Finish** to save each Step and click **Save** to save the Job.
Schedule Job	Use the following options to schedule this Job: • **Execute**—Select to execute this job **Immediately** or at a **Scheduled** time. • **Select Date**—If you chose to execute this job at a Scheduled time, click the calendar icon and select a date. • **Enter Time**—If you chose to execute this job at a Scheduled time, enter the time in hours and minutes.

Information Requested When Defining Job Steps in a Custom Job

Each time you select a **Step Type**, you are prompted to specify additional information:

Table 25-16 • Information Requested When Defining a Job Step in a Custom Job

Step Type	Required Selections
Scan for Dependencies	1. Click **Next**. The **Step 2 of 2** View opens. 2. Select the packages or groups of packages that you want to scan, and then click **Finish** to save this Job Step.
Directory Monitoring	1. Click **Next**. The **Step 2 of 2** View opens. 2. In the **Directory Name** box, enter the UNC path of the directory that you want to monitor. 3. Choose whether to **Include subdirectories in the 'Directory Monitoring' process**. 4. Click **Finish** to save this Job Step.
Validation	1. Click **Next**. The **Step 2 of 2** View opens. 2. Select the packages or groups of packages that you want to validate. 3. Click **Finish** to save this Job Step.
Conflict Detection	1. Click **Next**. The **Step 2 of 4** View opens. 2. Select the source packages or groups of packages that you want to include in the conflict detection process, and then click **Next**. The **Step 3 of 4** View opens. 3. Select the target packages or groups of packages that you want to use in the conflict detection process and click **Next**. The **Step 4 of 4** View opens. 4. Select the set of Conflict ACE Rules that you want to include in the conflict detection process and click **Finish** to save this Job Step.
Import	1. Click **Next**. the **Step 2 of 4** View opens. 2. Select a package to import. 3. Specify whether to include the package in the Software Repository 4. Click **Next**. The **Step 3 of 4** View opens. 5. Select any transform (`.mst`) or patch (`.msp`) files to import. 6. Click **Next**. The Step 4 of 4 View opens. 7. Select one or more destination groups into which the package will be imported. 8. Click **Finish** to save this Job Step.

Table 25-16 • Information Requested When Defining a Job Step in a Custom Job (cont.)

Step Type	Required Selections
Best Practice Detection	1. Click **Next**. The **Step 2 of 2** View opens. 2. Select the packages or groups of packages that you want to perform Best Practice conflict detection on. 3. Click **Finish** to save this Job Step.
Resolution	1. Click **Next**. The **Step 2 of 3** View opens. 2. Select the packages or groups of packages that you want to perform conflict resolution on, and then click **Next**. The **Step 3 of 3** View opens. 3. Select the Conflict ACE Rules that you want to resolve automatically. 4. Select a **Resolved packages re-import option**. See Re-Import Behaviors. 5. Click **Finish** to save this Job Step.

Edit Job Page

You can edit an existing Job by selecting the Job on the **Jobs Queue** page to open the **Edit Job** page.

Note • *When editing Jobs note that:*

- *Only one Job can be edited at a time.*
- *You cannot edit a Job with a **Status** of **Running**.*
- *You cannot change a **Custom Job** to a **Template Job** or vice versa.*

You edit a Job following the instructions in Creating a New Job Based on a Template or Creating a New Custom Job. When you have finished editing the Job, click **Save**.

Editable Items in an Existing Job

The following information can be edited:

Table 25-17 • Editable Items in an Existing Job

Custom Jobs	Template Jobs
General • Job Name • Execution options • Step Type **Scan for Dependencies Job Step** • Selected packages and groups **Directory Monitoring Job Step** • Directory to monitor • Option to monitor subdirectories **Validation Job Step** • Selected packages and groups **Conflict Detection Job Step** • Selected source packages and groups • Selected target packages and groups • Set of conflict detection rules to use during the conflict detection process **Import Job Step** • Package to import • Software Repository option • Transform (.mst) and Patch (.msp) files to include during import • Destination group for imported package **Best Practice Detection Job Step** • Selected packages and groups **Resolution Job Step** • Selected packages and groups • Conflicts to resolve • Resolved package re-import options	**General** • Job Name • Execution options **General** • Selected packages and groups

Delete Job/Template Page

The **Delete Job/Template** page opens when you attempt to delete a Job or Template and prompts you to confirm the deletion. To delete a Job or Template:

- **Deleting a Job**—Open the **Jobs Queue** page and click **Delete** in the **Action** column of a Job that you want to delete.
- **Deleting a Template**—Open the **Job Template List** page and click **Delete** in the **Action** column of a Template that you want to delete.

To confirm the deletion, click **Delete**. To return to the **Jobs Queue** or **Job Template List** page without making the deletion, click **Cancel**.

Schedule Job Page

You can reschedule a completed Job so that it is run again. You can also reschedule Jobs that failed, were cancelled, or are scheduled to be run in the future.

To reschedule a Job, open the **Jobs Queue** page and click the date and time listed in the **Schedule** column for the Job you want to reschedule. The **Schedule Job page** opens.

The **Schedule Job** page includes the following options:

Table 25-18 • Schedule Job Page Options

Option	Description
Execute	Select **For Immediate Execution** or **For Scheduled Execution**.
Select Date	If you selected **For Scheduled Execution**, click the calendar icon and select the date when you want this Job to run.
Enter Time	If you selected **For Scheduled Execution**, identify the time of day that you want this Job to run by selecting an hour of the day from the first list, selecting the minutes after the hour from the second list, and selecting **AM** or **PM** from the third list.
Recurrence Interval	If you want this Job to reoccur at a specified interval, select **Weekly, Daily, or Monthly** from this list, and perform the following steps: • If you want this Job to reoccur a specific number of times, select the **Number of Times** option, and enter a number in the box. • If you want this Job to reoccur until a specified expiration date, select the **Expiration Date** option, click the calendar icon, and select a date from the calendar.
Save	Click to save this Job schedule.

Job Alerts Page

You can choose to send an email when a Job is completed and when it fails. To do this, open the **Jobs Queue** page and click **Add** in the **Alerts** column of the Job that you want to set an alert for. You then can create an email message on the **Job Alerts** page.

When composing the **Subject** line and the **Message**, you can insert system variables to instruct Job Manager to automatically insert the variable data regarding that Job at runtime.

The **Job Alerts** page includes the following options:

Table 25-19 • Job Alerts Page Options

Options	Description
To	Enter the email addresses that you want to be notified when a Job is completed or when it fails. Separate multiple email addresses with a semicolon (;).
Subject	Enter text to describe the purpose of the email.
Message	Enter the text of the email message. You can select System Variables from the list so that the value of the variable for this Job is inserted when the email is sent.
Conditional Forward	Select one of the following options: • **Failed**—Send an email when this Job fails. • **Complete**—Send an email when this Job is completed.
Save	Click to save your entries.

Job Status Page

On the **Jobs Queue** page, a Job's current status is listed as a hyperlink in the **Status** column. If you click on this hyperlink, the **Job Status** page opens, displaying the **Job Name** and **Job Status**, and the status of each of the Job Steps in that job.

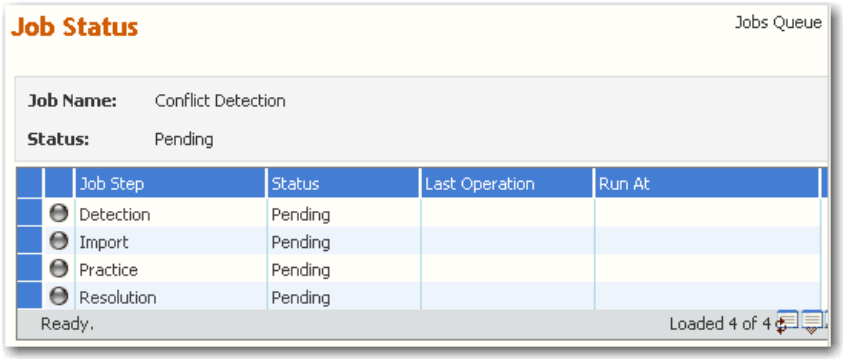


Figure 25-3: Job Status Page

The possible statuses are:

- **Scheduled**—Job is scheduled to be executed at a future date and time.
- **Running**—Job is currently being run. Only one job per Application Catalog connection will ever be running at one time.
- **Pending**—Job is available to run but is waiting for the running job to complete. A job set for immediate execution will be automatically assigned a **Pending** job status.
- **Complete**—Job has been executed.
- **Canceled**—Job was canceled before it was executed or before its execution was complete.
- **Failed**—Job was unable to run either because of a software error, a network communication problem, a hardware failure, or any other reason.

Click **Back** to return to the **Jobs Queue** page.

Job Template List Page

The **Job Template List** page lists all defined Templates. From this page you can choose to add a new Template, edit an existing Template, or delete a Template.

The following information and options are available:

Table 25-20 • Job Template List Page Options

Option	Description
Add	Click to open the **Add Template** page where you can create a new Template.
Template Name	Name of Template
# of Steps	Number of Job Steps defined in the Template.
Description	Description that was entered to identify the Template.
Action	Click **Delete** in this column to delete a listed Template.

Add Template Page

The **Add Template** page contains the following options:

Table 25-21 • Add Template Page Options

Option	Description
Name	Enter a name to identify the new Template.

Table 25-21 • Add Template Page Options (cont.)

Option	Description
Description	Enter a description to identify the purpose of this Template.
Define Job Steps	You create Job Steps using this area. Each time you want to create a Job Step, select one of the listed tasks from the **Step Type** list: • **Scan for Dependencies**—Analyze a Windows Installer package for dependent files. • **Directory Monitoring**—Use to establish a consistent, repeatable process in which a directory and, optionally, its subdirectories are monitored for any new or modified Windows Installer package files, and if any changes are detected, those new or modified files are then either imported or re-imported into the Application Catalog. See About Automatically Importing Packages from a Network Directory for more information. • **Validation**—Verifies the internal data integrity of a Windows Installer package. • **Conflict Detection**—Performs conflict detection on selected packages using Conflict ACE Rules • **Best Practice Detection**—Performs conflict detection on a single package using Best Practice ACE Rules to enforce Windows Installer standards. • **Resolution**—Performs automatic conflict resolution on selected packages. For each type of Job Step, you are prompted to specify additional information. See Information Requested When Defining Job Steps in a Template. Use the **Up** and **Down** arrows to rearrange the order of the Job Steps, and use the **Delete** button to delete a Job Step. Click **Finish** to save each Step and click **Save** to save the Job.

Information Requested When Defining Job Steps in a Template

Each time you select a **Step Type**, you are prompted to specify additional information:

Table 25-22 • Information Requested When Defining a Job Step in a Template

Step Type	Required Selections
Scan for Dependencies	Click **Next** to save this Job Step. (Source packages will be selected when a Job based on this Template is created.)

Table 25-22 • Information Requested When Defining a Job Step in a Template (cont.)

Step Type	Required Selections
Directory Monitoring	1. Click **Next**. The **Step 2 of 2** view opens. 2. In the **Directory Name** box, enter the UNC path of the directory that you want to monitor. 3. Choose whether to **Include subdirectories in the 'Directory Monitoring' process**. 4. Click **Finish** to save this Job Step.
Validation	Click **Next** to save this Job Step. (Source packages will be selected when a Job based on this Template is created.)
Conflict Detection	1. Click **Next**. The **Step 2 of 3** view opens. 2. Select the target packages or groups of packages that you want to use in the conflict detection process and click **Next**. The **Step 3 of 3** view opens. 3. Select the set of Conflict ACE Rules that you want to include in the conflict detection process and click **Finish** to save this Job Step.
Best Practice Detection	Click **Next** to save this Job Step. (Source packages will be selected when a Job based on this Template is created.)
Resolution	1. Click **Next**. The **Step 2 of 2** view opens. 2. Select the Conflict ACE Rules that you want to resolve automatically. 3. Select a **Resolved packages re-import option**. See Re-Import Behaviors. 4. Click **Finish** to save this Job Step.

Edit Template Page

You can edit an existing Template by selecting that Template on the **Job Template List** page to open the **Edit Template** page.

Edit the Template following the instructions in Creating a New Template. When you have finished editing the Template, click **Save**.

Conflict Detection Options Page

On the **Conflict Detection Options** page, you select the rules that you want Job Manager to use by default during conflict detection. The following rules are available.

Table 25-23 • Conflict Detection Options

Category	ACEs
Components	• **ACE30**—A check for KeyPath conflicts across components. • **ACE02**—Check destination directories for identical components. • **ACE09**—Check for different merge module versions.
File Extensions	• **ACE17**—Confirm that file extensions exist in only one component.
Files	• **ACE23**—Identifies file duplication between source and target packages. • **ACE12**—Checks for components that contain files that can be replaced by one of the merge modules that you have imported into the Application Catalog database. • **ACE08**—identifies components with identical ComponentIds, and checks those components to see if the versions of the files in each component match. • **ACE07**—Checks for the existence of the same file in components with different ComponentIds. • **ACE03**—Checks to see if components in different packages that have matching ComponentIds also contain the same files.
INI Files	• **ACE22**—Checks file name/target directory pairs in the File table to see if they conflict with similar entries in the IniFile table. • **ACE21**—Checks entries in the IniFile table to see if they conflict with similar entries in the File table. • **ACE14**—Checks for the existence of components with different ComponentIds that modify the same INI file entry, such as the [File Name/Section/Key/Value] entry.
NT Services	• **ACE16**—Checks for the existence of identical NT Services in components with different ComponentIds.
ODBC Resources	• **ACE15**—Checks for the existence of identical ODBC entries in components with different ComponentIds.
Product Properties	• **ACE20**—Checks the Upgrade Code to see if it is unique. • **ACE19**—Checks the Product Code to see if it is unique. • **ACE18**—Checks the Package Code to see if it is unique.

Table 25-23 • Conflict Detection Options (cont.)

Category	ACEs
Registry	• **ACE24**—Checks to see if registry entries with the same registry hive, key, and value name have the same data type and value. • **ACE10**—Checks for the existence of identical Root/Key/Name registry combinations in components with different ComponentIds.
Shortcuts	• **ACE13**—Checks for the existence of the same shortcut within different packages in components with different ComponentIds.
Windows Terminal Server Validation	• **WST01**—Checks to see if the ALLUSERS property is defined. • **WST02**—Checks for any registry entries installed to per-user specific locations. • **WST03**—Checks for any file entries installed to per-user specific locations. • **WST04**—Checks for any ODBC data source entries installed to per-user specific locations. • **WST05**—Checks for any environment strings installed to per-user specific locations.

Job Manager Engine

Job Manager Engine executes the Jobs created with Job Manager. Job Manager Engine can connect to any of the Application Catalogs in your network and allow you to configure how the engine operates and manage the state of any particular Application Catalog.

Job Manager Engine has the following Views:

- Manage Application Catalogs View
- Conflict Detection Configuration View
- Validation Configuration View
- E-mail Alerts View

Manage Application Catalogs View

Application Catalogs that are connected to the Job Manager Engine are listed on the Manage Application Catalogs View. For each Application Catalog, the following information is listed:

Table 25-24 • Connected Application Catalog Information

Item	Description
Catalog Name	Name of the connected Application Catalog database.
Status	Identifies whether the Job is Running, Pending, or Shutdown.

Table 25-24 • Connected Application Catalog Information (cont.)

Item	Description
# of Jobs	Number of Jobs in that Application Catalog that have been submitted for execution.

When Job Manager Engine is open, it automatically identifies all of the Jobs in all of the Application Catalogs that it is connected to that are in **Pending** status. Then, as the scheduled time interval of each Pending Job is reached, Job Manager Engine will execute it.

On the Manage Application Catalogs View, the following functions can be performed:

Table 25-25 • Job Manager Engine / Manage Application Catalogs View Options

Option	Description
Delete	Click to delete the connection of Job Manager Engine to an Application Catalog. ***Note*** • *When you delete an Application Catalog from Job Manager Engine, it will first shut down the Application Catalog, allowing any currently executing Job Steps to complete. Therefore, this operation may take some time to complete.*
Run/Restart	Click to initiate execution of Jobs from a selected Application Catalog that was Paused or Shutdown.
Pause	Click to pause the execution of the Jobs in the selected Application Catalog.
Shutdown	Click to stop all Jobs on the selected Application Catalog. Any currently running Job Step is allowed to finish, but no additional Jobs would execute.

When you select an Application Catalog on Job Manager Engine Manage Application Catalogs View, an icon is displayed to indicate its status:

Table 25-26 • Job Manager Engine Status Icons

Icon	Status
	All Pending Jobs and any future Jobs sent from this Application Catalog will be executed.
	The **Pending** Jobs from this Application Catalog will not be executed until the connection is restarted by clicking the **Run/Start** button.

Table 25-26 • Job Manager Engine Status Icons (cont.)

Icon	Status
	No Jobs from this Application Catalog will be executed until the Application Catalog is restarted by clicking the **Run**/**Start** button. All Jobs from this Application Catalog that were in **Pending** status when it was shutdown were **Failed** and must be resubmitted to execute.

Conflict Detection Configuration View

On the Conflict Detection Configuration View, you can set the following configuration options:

Table 25-27 • Conflict Detection Configuration View Options

Option	Description
ACE File	File containing the pre-defined ACE rule information for all ConflictSolver Conflict and Best Practice ACEs. It is installed in the following location on the machine where AdminStudio is installed: *AdminStudio Installation Directory* \ConflictSolver\Support
Custom ACE File	You use user-defined custom ACEs to extend the functionality of pre-defined ACEs with company-specific functionality. By selecting different user-defined ACE files, you can organize rules appropriate for individual users in your organization. The user-defined ACE file specified here is run after the pre-defined ACE rules are run. The selection of this user-defined ACE file will affect the default Conflict Types displayed on the Job Manager interface when you are creating Job Steps. By default, an empty user-defined ACE file is installed in the following location on the machine where AdminStudio is installed. C:\AdminStudio Shared\ConflictSolver\ CustomConflictFile.ace Only one user-defined ACE file can be active at one time.

Validation Configuration View

On the Validation Configuration View, you can set the following configuration options:

Table 25-28 • Validation Configuration View Options

Option	Description
CUB File	Validation involves comparing a Windows Installer package against a known criteria to identify deviations from those rules. By default, ConflictSolver compares packages against the Full MSI Validation Suite. This suite contains a comprehensive set of Internal Consistency Evaluators (ICEs)—guidelines created by Microsoft to ensure an installation package works correctly with the Windows Installer engine. The file specified in this field is file containing Internal Consistency Evaluators (ICEs) used for validation. The CUB file is configured as a part of the Job Server Engine installation, so normally you do not need to modify this entry.

E-mail Alerts View

On the E-mail Alerts View, you can enter configuration settings so that email alerts can be sent out to report the status of a completed or failed job.

Table 25-29 • E-mail Alerts View Options

Option	Description
SMTP Mail Server Name	Enter the SMTP mail server the Job Manager Server should use when sending messages. Enter the MTP Mail Server in the format of: `mail.mycompany.com`
Senders E-mail Address	Enter the E-mail Address of the person who you want to be displayed as the email sender.
SMTP Server Port	When someone sends a mail message to your SMTP Server, the mail will attempt to come in to your server on Port **25**. This is the standard SMTP (Simple Mail Transport protocol) port. Port 2525 is an alternate port used by forward services. This is a non-standard port, but useful if the standard SMTP port is blocked

Index

Symbols

-? 466, 562
.aacx files 734
.aot 573
  converting to Repackager project 573
  difference from .axt file 574
.axt 573
  converting to Repackager project 573
  difference from .aot file 574
.cab 1279
.cer file 1652
.cub 1268, 1272
.inc 562, 573, 674, 681
  converting to Repackager project 573
.ini Files Tab 670
.ipf 573
  converting to Repackager project 573
.irp 571, 681
  creating 571
.isl 577
  converting to Repackager project 577
.ism 109, 456, 562, 578, 674
  building in Repackager 578
.msi 578
  building in Repackager 578
.osc 338
.pvk file 1653
.spc file 1653
.spy 681
.txt 577
  converting to Repackager project 577
.wse 577
  converting to Repackager project 577
<Machines> 735
<Options> 735
<PackageList> 735
<Packages> 735
<Results> 735

Numerics

-10000 - Process Cancelled By User 947
-10001 - Suite File Missing 947
-10002 - Suite File is Duplicate 947
-10003 - Application File Missing 948
-10004 - INI File Missing 948
11000 - Excluding TCPIP Registry Entries 948
11001 - Fail on VMware 949
11003 - Control Panel Applet - Citrix 949
11004 - Control Panel Applet - ThinApp 949
11005 - QuickTime 7.4.1 Causes Fatal Error 950
11006 - Adobe Distiller Exclude AdobePDFSettings 950
11007 - Exclude URL Shortcut 950
64-bit applications 492
-9000 - Unknown Exception 895
-9001 - Unknown COM 896
-9002 - Error Opening Package 896
-9003 - Error Saving Package 896
-9004 - Process Cancelled By User 897
-9005 - Error Creating Temporary Folder 897
-9006 - Error Decompressing Package 898
-9007 - File With Extension Not Found 898
-9008 - Error Extracting Icon 899
-9009 - Unknown Provider 899
-9010 - Invalid Target File Name 899

-9011 - Error Reading MSI Table 900
-9012 - Unexpected Error in Method 900
-9013 - Type Library Not Found 901
-9014 - ShellExecute Failed 901
-9015 - Unable to Determine Full Path for Driver 902
-9016 - Contents of Table Ignored 902
-9017 - .NET 1.x Assembly Not Supported 903
-9018 - Custom Actions Warning 903
-9019 - Conditionalized Components 904
-9020 - Directory With Null Parent Error 905
-9021 - Unable to Extract COM Data 905
-9022 - Complus Table Error 906
-9024 - FileSFPCatalog 906
-9026 - LaunchCondition Table Warning 907
-9027 - LockPermissions Table Warning 907
-9028 - MoveFile Table Error 908
-9029 - MsiDriverPackages Table Error 909
-9030 - ODBCTranslator Table Warning 909
-9031 - RemoveFile Table Warning 910
-9032 - RemoveIniFile Table Warning 910
-9033 - RemoveRegistry Table Warning 911
-9036 - ISCEInstall Table Error 911
-9037 - ISComPlusApplication Table Error 912
-9038 - ISPalmApp Table Error 912
-9039 - ISSQLScriptFile Table Error 913
-9040 - ISVRoot Table Error 913
-9041 - ISXmlFile Table Error 914
-9051 - Package Decompression Canceled 914
-9100 - CreateInstance of Package Object Failed 914
-9101 - Create Operation of Package Object Failed 915
-9102 - Failed to Write Header Information 915
-9103 - Citrix Finalization Failed 916
-9104 - Citrix Save Failed 916
-9105 - Error Initializing Citrix Writer 916
-9106 - Error Initializing Citrix Package 917
-9107 - Error Writing Citrix File Entries 917
-9108 - Error Determining Source File Path 918
-9109 - Error Writing Citrix Folder Entries 918
-9110 - Error Writing Citrix Registry Entries 918
-9113 - Error Writing Citrix INI File Entries 919
-9114 - Error Writing Citrix Shortcuts 919
-9115 - Error Saving Citrix Profile 920
-9116 - Error Creating Empty Citrix Profile 920
-9117 - Error Creating Intermediate Folder 920
-9118 - Error Initializing Citrix Profile 921
-9119 - Error Creating Default Target in Citrix Profile 921
-9120 - Error Deleting File From Profile 921
-9121 - Failed to Copy File into Citrix Profile 922
-9122 - Target Does Not Exist in Citrix Profile 922
-9124 - No Shortcuts Created for this Profile 923
-9125 - Error Writing Citrix File Type Associations 923
-9126 - Failed to Sign Profile Using Certificate 924
-9127 - Could Not Create Script Execution 924
-9128 - Duplicate Shortcut 924
-9129 - Duplicate Shortcut Names 925
-9130 - Duplicate Shortcut Targets 925
-9131 - Unable to Resolve Installer Variable 926
-9132 - 16 Color Shortcut Icon Not Found 926
-9133 - Shortcut Icon Not Found 926
-9134 - Failure to Extract Icon from Executable 927
-9135 - Shortcut Target is 16-Bit 927
-9136 - Some Files May Not Be Decompressed 928
-9137 - Destination Directory Cannot Be Found 928
-9138 - DuplicateFile table warning 929
-9139 - 64-bit executables 929
-9200 - ThinApp Must Be Installed 930
-9201 - Extension for Shortcut Files Must Be .exe 930
-9202 - No Applications Were Created 931
-9203 - ThinApp Tool is Missing 931
-9204 - Duplicate Shortcut 931
-9205 - Identically-Named Shortcut Already Exists, But With Different Command Line Parameters 932
-9206 - Identically-Named Shortcut Already Exists But With a Different Target 932
-9207 - Error During Build Process (vregtool.exe) 933
-9208 - Error Occurred During Build Process (vftool.exe) 933
-9209 - Error Occurred During Build Process (tlink.exe) 934
-9210 - 64-Bit Executables (ThinApp) 934
-9300 - Unhandled Exception During AdviseFile Operation 934
-9301 - Unhandled Exception During AdviseRegistry Operation 935
-9302 - Unhandled Exception During Command Action 935
-9303 - Unhandled Exception During Alter File Action 935
-9304 - Unhandled Exception During Alter Registry Action 936
-9305 - Unhandled Exception During Create Action 936
-9306 - Unhandled Exception During Execution of Rules Engine 937
-9401 - Error Initializing App-V Writer 937
-9402 - Error Initializing App-V Package 937
-9403 - Error Writing App-V File Entries 938
-9404 - Error Writing App-V Folder Entries 938
-9405 - Error Writing App-V Registry Entries 938
-9406 - Error Writing App-V INI File Entries 939
-9407 - Error Writing App-V Shortcuts 939
-9408 - Error Writing App-V File Type Data 939
-9409 - Error Saving App-V Data 940
-9410 - Error Determining Source File Path 940
-9411 - OSD File Template Could Not Be Extracted 940
-9412 - OSD File Could Not Be Saved 941
-9413 - App-V OSD Real Save 941

-9414 - Local App-V Application Should Not Be Specified as a Dependency of the Primary Application 941
-9415 - Dependency Application Was Not Found 942
-9416 - Invalid Primary Application Directory 942
-9417 - Dependency Application's OSD File Contains an Invalid HREF Value 943
-9418 - Error While Privatizing Side-By-Side Assemblies 943
-9419 - Error Inserting Watermark 943
-9420 - Errror During App-V Package Upgrade 944
-9421 - 16-Bit Application 944
-9422 - Package Cannot Be Opened 945
-9423 - No Shortcuts Detected 946
-9500 - Shortcut Missing 946

A

aacx.exe 865
Abort Result setting 1025
About AdminStudio dialog 122
About Exclusions Editor dialog box 674
About InstallShield ConflictSolver dialog 1519
About QualityMonitor dialog 1702
About Repackager dialog box 633
access accounts 1810
accounts
  deleting a group account 142
  deleting a user account 142
ACE Rule Properties dialog 1519, 1520, 1521, 1522, 1523
  Additional Information tab 1520, 1521
  DLL Information tab 1523
  General Information tab 1519
  Where Clause tab 1522
ACE02 1370, 1406
ACE03 1370, 1406
ACE04 1370, 1408
ACE05 1408
ACE06 1370, 1409
ACE07 1370, 1410, 1901
ACE08 1370, 1412
ACE09 1370, 1413
ACE10 1370, 1414, 1901
ACE12 1370, 1416
ACE13 1370, 1418
ACE14 1370, 1419
ACE15 1370, 1420
ACE16 1370, 1421
ACE17 1370, 1422
ACE18 1370, 1423
ACE19 1370, 1424
ACE20 1370, 1424
ACE200 994, 1440
ACE201 994, 1441
ACE202 995, 1442
ACE203 995, 1444
ACE204 996, 1446
ACE205 996, 1447
ACE206 996, 1448
ACE207 997, 1448
ACE208 997, 1450
ACE209 1451
ACE21 1370, 1425
ACE210 1452
ACE211 1453
ACE212 1454
ACE213 1455
ACE214 1456
ACE215 998, 1457
ACE216 1458
ACE22 1370, 1426
ACE23 1370, 1428, 1901
ACE24 1370, 1429, 1901
ACE25 1431
ACE26 1431
ACE27 1432
ACE28 1433
ACE29 1434
ACE30 1434
ACE31 1435
ACE32 1436
ACE33 1437
ACE34 1437
ACE35 1438
ACE36 1439
ACEs 1399, 1400, 1406, 1408, 1409, 1410, 1412, 1413, 1414, 1416, 1418, 1419, 1420, 1421, 1422, 1423, 1424, 1425, 1426, 1428, 1429, 1467, 1469, 1471, 1474, 1477, 1479
  ACE02 1406
  ACE03 1406
  ACE04 1408
  ACE05 1408
  ACE06 1409
  ACE07 1410
  ACE08 1412
  ACE09 1413
  ACE10 1414
  ACE12 1416
  ACE13 1418
  ACE14 1419
  ACE15 1420
  ACE16 1421
  ACE17 1422

ACE18 1423
ACE19 1424
ACE20 1424
ACE200 1440
ACE201 1441
ACE202 1442
ACE203 1444
ACE204 1446
ACE205 1447
ACE206 1448
ACE207 1448
ACE208 1450
ACE209 1451
ACE21 1425
ACE210 1452
ACE211 1453
ACE212 1454
ACE213 1455
ACE214 1456
ACE215 1457
ACE216 1458
ACE22 1426
ACE23 1428
ACE24 1429
ACE25 1431
ACE26 1431
ACE27 1432
ACE28 1433
ACE29 1434
ACE30 1434
ACE31 1435
ACE32 1436
ACE33 1437
ACE34 1437
ACE35 1438
ACE36 1439
best practices ACEs 1432, 1435, 1437, 1438
creating custom 1469, 1471
creating DLL-based 1474
custom 1467
deleting user-defined 1477
DuplicateFileData ACE 1432
editing user-defined 1477
KeyPath ACE 1434
metrics 1477
MoveFileData ACE 1435
overview 1399
Predeployment Test 1901
reference 1400
RemoveFileData ACE 1437
RemoveIniFileData ACE 1437
RemoveRegistryData ACE 1438
specifying the Visual Studio C++ type library file path 1474
user-defined 1467
viewing ACE metrics 1477
WTS01 1459
WTS02 1460
WTS03 1461
WTS04 1462
WTS05 1463
WTS06 1463
WTS07 1464
WTS08 1465
WTS09 1466
activation 68
ports used in 68
silent 68
Active Directory 163
Active Directory Configurations Page 163
automatic login to Workflow Manager 157
controlling access to ThinApp applications 1118
importing users 159
LDAP attributes 163
Active Directory Configurations Page 163
Active Directory User Import View 159
Add AppLink Reference dialog box 1142, 1173
entering relative paths 1174
Add Ignore Table dialog 1523
Add New Tool dialog 123, 416
Add Packages 786
Add Tool Wizard 124, 133, 134
Command-Line Configurations panel 134
Command-Line Properties dialog 124
Tool Properties panel 134
Welcome panel 133
Add/Remove Program settings 1301
changing 1301
Add/Remove Programs 1302
disabling Modify button in 1302
disabling Remove button in 1302
disabling Repair button in 1302
Add/Remove Programs view 1301, 1338
changing properties in 1301
adding
ad-hoc test results 1584
test result message 1586
Additional Errors setting 1030
Additional Information panel 1546, 1548
Rules Wizard 1546, 1548
Additional Information Tab 1520, 1521
additional server locations 1299

configuring 1299
Additional Setup Programs dialog box 509, 559
Repackaging Wizard 509, 559
ad-hoc import 209
ad-hoc test results
about 1584
adding 1584
administration
copying roles 178
creating new roles 177
deleting roles 178
Administrative Install panel 1761, 1777
Distribution Wizard 1761, 1777
administrative installation 1730
creating with Distribution Wizard 1730
AdminStudio 89, 90, 91, 92, 94, 95, 112, 113, 116, 119, 133
activation 68
Add Tool Wizard 133
adding tools 133
checking for updates 95
configuring the interface 90
creating new Application Catalog 196
Customer Experience Improvement Program (CEIP) 126
database server permissions 464
default Application Catalog 199
disconnecting from Application Catalog 203
Interface reference 113
launching applications 91
menus 119
minimum permissions 464
minimum privileges 464
Process Assistant 490
Projects tab 116
Quick Start evaluation experience 193
sample database 193
setting shared location 92
setting task page help location 94
specifying default Application Catalog 199
specifying shared Application Catalog location 92
specifying the location of the AdminStudio Shared Directory 92
SQL Express 193
Start Page 114
Tool Properties dialog 130
toolbar 119
Tools Gallery 115
Tools tab 115
upgrading legacy databases 204
using 1263
using the Interface 89
virtualization 691
AdminStudio Application Catalog 196, 199, 203, 295, 722, 752, 818
creating new 196
disconnecting from 203
specifying default 199
AdminStudio Automated Application Converter Log Report 804
viewing 768
AdminStudio Automated Application Converter Log report 733
viewing 806
viewing debug messages 806
AdminStudio databases 204
upgrading 204
AdminStudio Enterprise Server
evaluating 68
forgetting your password 152
Report Center 1995
standalone Application Catalogs 192
AdminStudio Interface 89, 90, 95, 113, 118, 122, 133
configuring the interface 90
configuring to stay on top 95
dialogs 122
reference 113
using 89
wizards 133
Workflows Templates tab 118
AdminStudio options 124
AdminStudio Shared Directory 92, 124
configuring for Replication 312
Global Exclusions list 92
AdminStudio tasks 100
associating tools with 100
AdminStudio tools 99, 101, 115
deleting command-line configurations from 99
deleting tools from Tools Gallery 101
AdminStudio Workflow Manager. See Workflow Manager
Advanced Client 1799
advanced conversion options 587
configuring Repackager 587
Advanced Marimba Options dialog box 1770
Advanced Options dialog 450, 1659, 1660, 1661
Digital Signature tab 1661
Manifest Options tab 1660
Advanced Options view 587
Advanced Settings view 658
advertisement 1276
feature 1276
advertisements 1799, 1819
Advertisements View 1841
All Merge Modules view 411

All Reports page 2020
Allow application execution to the following user groups 1118
Allow Local Interaction setting 1009
ALLUSERS 674
alternate-language repackaging on clean machines 496
Altiris 1732
  Altiris Integration Panel 1764
  distribution 1732
  XML template 1765
AMS. See Workflow Manager
analysis options 339
  configuring in OS Snapshot Wizard 339
Analysis Options dialog 457
Analysis Options dialog box 561
  Repackaging Wizard 561
anonymous
  Customer Experience Improvement Program (CEIP) 126
anonymous account 153, 155
  logging in 155
  setting up 153
anti-virus software 618
  excluding directories when using Snapshot method 498
-app 562
Append Package Version setting 960
application
  saving after being wrapped 1961
  testing a wrapped application 1962
  using after being wrapped by FlexWrap 1916
Application Catalog 92, 196, 199, 202, 203, 295, 482, 1355, 1897
  AdminStudio default 199
  connecting to 194, 195
  connecting to a specific using command-line options 482
  connecting to existing 193
  connecting to existing in ConflictSolver 1355
  connecting to on the Predeployment Test Reports Web site 1897
  creating new 196
  disconnecting from 203
  enabling Software Repository 285
  integration with InstallShield Editor 1256
  replication 311, 312, 316
  sample 193
  searching 202
  specifying a default for an AdminStudio enterprise 199
  specifying default 199
  specifying shared location 92
  standalone 192
  upgrading 204
  upgrading 5.0 or 5.5 or 6.0 Application Catalogs 204
  upgrading legacy 204
  upgrading pre-AdminStudio 5.0 Application Catalogs 204, 464
  using the Software Repository 284
  version management 290
Application Catalog Replication 311
  command line options 466
  creating 321
  creating a new publication 321
  creating a new subscription 328
  deleting a publication 326
  dialogs 413
  editing a publication 324
  editing a publication access list 325
  managing publications 320
  managing subscriptions 327
  Publication Manager 320
  publishing 312, 316, 319
  publishing a publication 323
  publishing to 311
  reference 340
  setting a publication schedule 324
  subscribing 312, 316, 326
  subscribing to 311
  subscribing to in Application Manager 302
  Subscription Manager 327
  wizards 434
Application Catalogs
  merge replication using SQL Server 332
Application Configuration view 1337
  Tuner 1337
Application Conflict Evaluators 1399, 1400, 1406, 1408, 1409, 1410, 1412, 1413, 1414, 1416, 1418, 1419, 1420, 1421, 1422, 1423, 1424, 1425, 1426, 1428, 1429, 1467, 1477
  ACE02 1406
  ACE03 1406
  ACE04 1408
  ACE05 1408
  ACE06 1409
  ACE07 1410
  ACE08 1412
  ACE09 1413
  ACE10 1414
  ACE12 1416
  ACE13 1418
  ACE14 1419
  ACE15 1420
  ACE16 1421
  ACE17 1422
  ACE19 1423, 1424

ACE20 1424
ACE21 1425
ACE22 1426
ACE23 1428
ACE24 1429
custom 1467
deleting user-defined 1477
DuplicateFileData ACE 1432
editing user-defined 1477
KeyPath ACE 1434
metrics 1477
MoveFileData ACE 1435
overview 1399
reference 1400
RemoveFileData ACE 1437
RemoveIniFileData ACE 1437
RemoveRegistryData ACE 1438
user-defined 1467, 1477
Application Conversion Project Wizard 718, 813
using 721
Application Conversion Project Wizard Complete panel 730
Application Conversion Wizard 718, 801, 843
using 765
application isolation 598, 602, 1647, 1650, 1651, 1652, 1653, 1654, 1655, 1656
assemblies 1650
certificate store 1653
certificates 1652
code signing 1652
concept 1647
digital signatures 1652
filtering file listings when manually configuring 1656
manifests 598, 1651
methods 1650
modifying default recommendations 1655
private key 1653
setting assembly naming conventions 1654
software publishing credentials 1653
using Repackager vs. Application Isolation Wizard 1649
Application Isolation Wizard 1647, 1649, 1650, 1652, 1653, 1657, 1658, 1659, 1662, 1663, 1664
Advanced Options dialog 1659
Application Isolation Progress panel 1659
Application Manifest Properties dialog 1663
Assembly Properties dialog 1662
AssemblyType property 1664
CertificateFile property 1664
CertificateName property 1664
code signing technologies 1652
command-line options 1664
Company property 1664
comparison to isolation using Repackager 1649
Completing the Application Isolation Wizard panel 1659
configuration files 1664
digital signatures 1652, 1661
Division property 1664
IsolatedComponents property 1664
Isolation Method panel 1658
isolation methods 1650
launching 1649
Manifest and Assembly Design dialog 1662
Manifests property 1664
NewComponents property 1664
private key 1653
PVKFile property 1664
reference 1657
setting manifest options 1660
software publishing credentials 1653
SPCFile property 1664
Summary panel 1658
TimeStamp property 1664
using Repackager to isolate Windows Installer packages 1649
Welcome panel 1657
Windows Installer File Selection panel 1658
Application Link 1141
Application Manager 183, 205
adding groups 205
All Merge Modules view 411
applying patches during command-line import 479
applying transforms during command-line import 479
checking out packages 291
command line 466
Components view 412
connecting to a specific Application Catalog using command-line options 482
context menus 348
copying packages to multiple groups 206
creating a shortcut to a specific Application Catalog 482
deleting groups 206
deleting packages 207
Dependency view 412
editing groups 206
Exclusion view 412
extended attribute description file 266
extended attributes 265
extended attributes and Workflow Manager 268
Extended Attributes view 381, 400, 403, 404
Files view 413
Files/Components view 381
getting a copy of package in Software Repository 292
Group view 378

identifying packages in Software Repository 289
Import Wizard 437, 438, 450
importing 207, 213, 215, 224, 226, 227, 229
importing all packages in a directory using command line 480
importing Marimba NCP files into 226
importing merge modules 224
importing merge modules and Windows Installer packages simultaneously 480
importing MSI packages into 211
importing OS snapshots into 225
importing other setup types into 228
importing packages in multiple configurations into 229
importing Windows Installer packages into 211
INI File Changes view 383, 401, 405
linking to a network directory 307
linking to a remote Application Catalog 300, 301, 304
Marimba NCP Files 403
menus 341
Merge Module view 400, 411
Merge Modules View 347
Merge Modules view 388
merging Application Catalog information 299
modifying groups 206
monitoring a Network Directory 300
moving groups in 207
moving OS snapshots in 207
moving products in 207
organizing products into groups 205
OS Snapshot view 400
OSD Files 403
Other Setup Types view 402
Output Window 352
Package Auto Import 301, 306, 307
Package Auto Import Wizard 451, 452, 453, 454
patches 407
Patches Group View 407
Patches tab 407, 409
Patches view 407, 409
Product view 344, 378, 413
refreshing a linked package 305
Registry view 383, 401, 405
removing groups 206
running import silently 481
searching the Application Catalog 202
Shortcuts view 384, 402, 406
subscribing to an Application Catalog 302
Tables view 389, 402, 406
toolbar 341
user permissions 353
using a configuration file during command-line import 479
using a configuration file to import multiple merge modules 480
using a configuration file to import multiple Windows Installer packages 480
using extended attributes 265
viewing linked packages in Product View 306
Application Manager dialogs
Advanced Options 450
Command-Line Parameters 416
Extended Attribute Property 423
Find 421
Group Properties 423
Package Auto Import Properties 425, 427
Subscriptions 424
Application Manager extended attributes 265
using 265
Application Manager groups 205, 206
adding 205
deleting 206
editing 206
modifying 206
removing 206
Application Manager Output Window 352
Application Manager views
All Merge Modules 411
Components 412
Dependency 412
Exclusion 412
Files 413
Files/Components 381
Group 378
INI File Changes 383, 401, 405
Merge Module 400, 411
Merge Modules 388
OS Snapshot 400
Other Setup Types 402
Product 378
Products 413
Registry 383, 401, 405
Shortcuts 384, 402, 406
Tables 389, 402, 406
Application Manifest Properties dialog 1663
application manifests 598, 1651
application object template files 574
Application Sync 1144
application virtualization 693
applications 1653
isolating 1653
version management 290
Applications page 1090, 1156

creating a new application shortcuts 1066, 1132
excluding or deleting a shortcut 1067, 1133
including an existing shortcut 1067, 1132
AppLink 1141
adding reference 1142
Required and optional links 1143
AppLink Reference dialog box
relative links 1143
AppLink references 1171
AppLink settings 1170
collisions and order of import 1172
entering a relative path 1174
example references 1175
required and optional linked applications 1172
required vs. optional 1175
Required vs. Optional applications 1172
AppLink Settings dialog box 1142, 1170
deleting a reference 1143
AppSync Settings
Expiration tab 1178
AppSync settings 1144
benefits of 1144
Clear Sandbox option 1145
configuring 1145
expiration 1176
Expire Period 1178
Expire Period option 1146
Frequency of update 1145
frequency of update 1145
Message displayed 1145
messages 1147
setting expiration settings 1145
URL of update 1145
Use Application Expiration 1178
Warning Frequency 1179
Warning Frequency option 1146
Warning Message 1179
Warning Period 1179
Warning Period option 1146
AppSync Settings dialog box 1176
App-V
specifying client runtime drive 860
App-V 4.6 699
App-V Application Launcher 699, 770
controlling whether to include in build output 771
for testing with packages edited in Virtual Package Editor 991
location of in build output 770
opening 770
requirements for using 770
App-V applications 563, 655
about 695, 696, 1038, 1042
adding diagnostic tools to 1054, 1101
adding existing folder 1056
adding files to 1055
adding or deleting registry keys and values 1071
App-V Assistant 1036
App-VPackage directory 702
automatically creating from Repackager 692
benefits of using App-V Assistant to create 1038
benefits of using Automated Application Converter to create 696
building 1078
building a Windows Installer package with build output 1076
building using command line 1108
command line 563
components of 700
compressing packages for 1075
conversion error and warning messages 1108
creating 64-bit App-V applications 699
creating using InstallShield Microsoft App-V Assistant 692
creating with InstallShield 1036
defining shortcut executables 1066
deployment server 1086
deployment server protocols 1086
editing OSD file 703
editing OSD file to identify location of App-V server 702
evaluating AdminStudio's App-V support 70
excluding vs. deleting a shortcut 1069
feature block 1 1104
feature block 2 1104
feature blocks 1076
how transforms are included 1046
ICO file 702
including additional Windows Installer packages 1075
including an existing shortcut 1067
including diagnostic tools 1085
infrastructure of Microsoft Application Virtualization 1038
inheritance of isolation options from folders to files 1065
inheritance of isolation options in the registry 1072
integration of Windows Services 703
location of generated files 591, 702, 1042
managing files and folders 1055
Manifest file 702
methods to convert Windows Installer packages 691
modifying registry 1070, 1071
moving files and folders 1058
OSD file 702
overview of 1037
overview of isolation options 1063

package optimizations 1076
renaming a shortcut 1069
selecting application shortcuts 1065
setting file isolation options 1102
setting folder isolation options 1102
setting isolation options 1062
setting isolation options for folders and files 1064
SFT file 701
shortcut requirements 1066, 1091
specifying deployment server 1085
specifying host name 1086
specifying operating system requirements 1085
specifying package name 1051, 1084
specifying port 1086
specifying root folder name 1084
specifying root folder name of SFT package 1051
specifying virtual application server 1086
start up and shut down sequences 703
steps to create 1050
steps to create with App-V Assistant 1043
storage path of application 1087
support for Windows services 703, 1047
testing using App-V Application Launcher 770
App-V Assistant 590, 692, 1036, 1087, 1108
about 1043
adding an existing folder to an App-V application 1056
adding diagnostic tools to an App-V application 1054, 1101
adding files to an App-V application 1055
adding or deleting registry keys and values from an App-V application 1071
application features requiring pre- or post-conversion actions 1108
Applications page 1090
benefits of using 1038
Build Options page 1096
building a Windows Installer package with build output 1076
building an App-V application 1078
building App-V application in Direct Edit mode 1075
compared with Microsoft App-V Sequencer 1040
controlling the display of predefined folders 1059
creating 64-bit App-V applications 699
creating an App-V application 1050
creating new application shortcut executables 1066
creating new folder 1058
deleting files and folders 1059
deployment server 1053
deployment server host 1053
deployment server port 1054
Diagnostic Tools dialog box 1054, 1101
Dynamic Suite Composition page 1093
enabling App-V application building when editing a Windows Installer package 1075
error messages 1108
excluding or deleting an App-V application shortcut 1067
FILE protocol 1053
Files page 1087
Home page 1082
how transforms are included 1046
HTTP protocol 1053, 1086
HTTPS protocol 1053
including an existing shortcut 1067
inheritance of isolation options 1065
inheritance of isolation options in registry 1072
integration with Project Assistant and Installation Designer 1034
interface comparison to Microsoft App-V Sequencer 1047
Launch App-V Application dialog box 1103
managing files and folders 1055
modifying build options 1074
modifying registry settings 1070
modifying shortcuts 1065
moving files and folders 1058
overview of 1037
overview of isolation options 1063
Package Information page 1083
Package Optimizations dialog box 1077, 1104
performing dynamic suite composition 1072
Primary Application Directory dialog box 1105
reference 1081
Registry Isolation Options dialog box 1107
Registry page 1091
renaming a shortcut 1069
RTSP protocol 1053
RTSPS protocol 1053
selecting releases to build 1074
setting isolation options 1062
setting isolation options for folders and files 1064
setting isolation options on files 1102
setting isolation options on folders 1102
specifying deployment server 1053
specifying general settings 1051
specifying OS requirements 1052
specifying primary application directory 1060
specifying upgrade information 1052
steps to create an App-V application 1043
support for Windows services 1047
supported InstallShield project types 1046
when to exclude and when to delete shortcuts 1069
App-V Client 696
App-V Comments property 785

App-V Compression property 785
App-V Data Type setting 1015
App-V Diagnostics property 786
App-V Dynamic Suites property 785
App-V Feature Block 1 setting 1014
App-V GUID setting 1014
App-V infrastructure 696
App-V Override setting for a file or folder 1015
App-V Override setting for a registry key 1017
App-V package 955
  adding shortcuts to 980
  editing 955
  extracting files from 974
  file extensions 983
  including debug tools with 993
  registry isolation 975
  scripts for 986
App-V Package Name property 783
App-V Package Optimization property 785
App-V Package Upgrade Settings dialog box 1101
App-V packages
  components of 1042
  files included in 1042
App-V Root Folder Name property 784
App-V Runtime Drive property 786
App-V Sequencer 696
  clean machine requirement 698
  installation directory requirement 698
  vs. Automated Application Converter 696
App-V Server 696
App-V Server Host property 783
App-V server location
  in OSD file 702
App-V Server Path property 784
App-V Server Port property 784
App-V Server Protocol property 784
App-V Supported OS property 785
App-V Upgrade Package property 785
App-V Version setting 1014
App-V Versioning property 785
App-V VFS Path setting 1015
ARPCOMMENTS 1338
ARPCONTACT 1338
ARPHELPLINK 1338
ARPHELPTELEPHONE 1338
ARPNOMODIFY 1338
ARPNOREMOVE 1338
ARPNOREPAIR 1338
ARPURLINFOABOUT 1338
ARPURLUPDATEINFO 1338
as-hoc import 214, 284
ASP.NET permissions 1897
  assigning in Predeployment Test 1897
assemblies 602, 1650, 1652, 1653, 1654, 1657
  digital signatures 1652
  in application isolation 1650
  private key 1653
  servicing published shared 1657
  setting naming convention 602, 1654
  signing 1652
  software publishing credentials 1653
assemblies and manifests 1650
  isolation method 1650
assembly files 596
assembly manifests 598, 1651
Assembly Properties dialog 1662
AssemblyType 1664
Associate with Workflow Manager Application dialog 415
Associated Patches view 1506
associated tools 100, 108
  running in projects 100, 108
attributes 163
  creating Active Directory 163
authentication
  for Application Catalog or Microsoft SCCM Server 753
authentication modes 303
  preferred for replication 303
authoring packages 1253
Automated Application Converter 689
  about 713
  App-V client runtime drive 860
  benefits of 714
  benefits of using to create App-V applications 696
  best practices 879
  bulk conversion of App-V applications 699
  command line parameters 865
  compress the MSI wrapper option 860
  creating 64-bit App-V applications 699
  getting started 718
  launching 717
  location of output 705
  Machines tab 790
  menus and toolbars 800
  opening a new or existing project 719
  overview diagram 715
  preparing virtual machines 740
  project files 734
  reference section 775
  selecting rules when adding packages from a directory 826
  setting default project properties 774
  setting project options 774

support for Windows services 703
supported operating systems 716
supported virtual machines 716
testing App-V application using App-V Application Launcher 699
user interface 777
using in evaluation mode 737
using list features 808
using to create App-V application 696
viewing debug messages 881
vs. App-V Sequencer 696
wizard comparison 718, 815
workflow diagram 715
wrapper MSI 860
Automated Application Converter Log
viewing debug messages 881
Automated Repackaging on Virtual Machines panel 730, 767
automatic job execution
Job Manager 2035
automatic login 157
using Active Directory 157
Automatic Repackaging on Virtual Machines panel 838
automation
Job Manager 2035

B

-b 562
batch convert option
evaluating 70
batch converter 689
best practices 338, 879
ConflictSolver 1356, 1358, 1371, 1374
OS Snapshot 338
best practices ACEs 1432, 1435, 1437, 1438
Browse for Folder Dialog Box 855
Browse for Folder dialog box 758, 855
Browse local machine 726, 745, 829, 832
Browse local machine and network 723
Build App-V Packages 858
Build Citrix Profiles 858
Build Options page 1096, 1159
Build Settings page 1240
digitally signing a Citrix profile 1224
Build ThinApp application 1161
Build ThinApp Packages 858
Build Virtual Package 1162
Build Windows Installer Packages 858
Build Wrapper MSI setting 961
BuildCompressed 674
building App-V applications using command line 1108
building profiles using command line 1251
building ThinApp applications using command line 1180
-buildonly 563
business rules 1467, 1468
-bv 563

C

-C 466, 479, 480
C1083 error 1474
CAB file 710
CAB files 1279
Tuner 1279
cancel 801
CAPS (Client Access Points) 1799
Captured Installation view 642
capturing OS snapshots 338, 340
CARD 1399, 1406, 1408, 1409, 1410, 1420, 1423, 1424, 1479, 1481, 1482, 1483, 1484
CARD02 1406, 1481
CARD04 1408, 1482
CARD05 1408, 1482
CARD06 1409, 1482
CARD07 1410, 1483
CARD15 1420, 1483
CARD18 1423, 1483
CARD19 1424, 1484
CARD20 1424, 1484
CARDs 1480
CARD02 1481
CARD04 1482
CARD05 1482
CARD06 1482
CARD07 1483
CARD15 1483
CARD18 1483
CARD19 1484
CARD20 1484
overview 1479
reference 1480
CEIP (Customer Experience Improvement Program 126
certificate name in the store 1652
CertificateFile 1664
CertificateName 1664
certificates 1652, 1653
certificate store 1653
private key 1653
-cf 562, 563
channel publishing 1735
using Marimba with Distribution Wizard 1735
check in and check out functionality of packages 291

checking for updates 124
checklist 1314
Choose Registry Key dialog box 673
Citrix 563, 589, 656, 707
  Citrix profile 578, 656
  Citrix XenApp 589, 707
  command line 563
Citrix Assistant 590, 692, 1190, 1251
  about 1193
  adding an existing folder to a profile 1207
  adding diagnostic tools to profile 1200, 1244
  adding files to a Citrix profile 1207
  adding or deleting registry keys and values from a Citrix profile 1221
  adding pre-launch and post-exit scripts to profile 1203
  application features requiring pre- or post-conversion actions 1252
  Build Settings page 1240
  building a Citrix profile 1225
  building Citrix profile in Direct Edit mode 1225
  controlling the display of predefined folders 1210
  creating a Citrix profile 1198
  creating new folder 1209
  creating new profile shortcut 1217
  deleting files and folders 1210
  Diagnostic Tools dialog box 1200, 1244
  digitally signing a profile 1224
  enabling Citrix profile building when editing a Windows Installer package 1225
  error messages 1251
  excluding or deleting a profile shortcut 1218
  Home page 1227
  how requirements are applied at runtime 1202
  how transforms are included 1197
  including an existing profile shortcut 1218
  inheritance of isolation options 1214
  inheritance of isolation options in registry 1222
  integration with Project Assistant and Installation Designer 1034
  managing files and folders 1206
  modifying build settings 1222
  modifying registry settings 1220
  modifying shortcut settings 1214
  moving files and folders 1209
  overview 1190
  overview of 1190
  overview of Citrix profiles 705, 1194
  overview of isolation options 1212
  Profile Files page 1234
  Profile Information page 1229
  Profile Registry page 1239
  Profile Requirements page 1231
  Profile Shortcuts page 1237
  reference 1227
  Registry Isolation Options dialog box 1248
  renaming a shortcut 1220
  Script Execution dialog box 1243
  selecting releases to build 1223
  Service Packs Requirement dialog box 1250
  setting isolation options 1211
  setting isolation options for folders and files 1214
  setting isolation options on files 1245
  setting isolation options on folders 1247
  setting language requirements 1202
  setting operating system and service pack requirements 1201
  setting registry isolation options 1221
  setting Service Pack requirements 1250
  specifying profile information 1198
  specifying profile name and version 1199
  specifying profile requirements 1201
  specifying whether users can update applications 1199
  steps to create a Citrix profile 1191
  supported InstallShield project types 1046, 1197
  when to exclude or delete shortcuts 1219
Citrix Presentation Server 4.5
  benefits of 1110
Citrix profile 578, 589, 656, 707, 708
  about custom actions 951
  about services 951
  and COM+ applications 953
  automatically creating from Repackager 692
  benefits of deploying 711
  CAB file 710
  creating using InstallShield Citrix Assistant 692
  features requiring pre-or post-conversion actions 951
  hash file key for digital signatures 710
  hashes file 710
  ignored tables 951
  location of generated files 709
  methods to convert Windows Installer packages 691
  overview of 705, 709
  overview of Citrix XenApp 708
  profile manifest file 710
Citrix profiles 1192
  about 1194
  adding diagnostic tools to 1200, 1244
  adding existing folder 1207
  adding files to 1207
  adding or deleting registry keys and values 1221
  adding pre-launch and post-exit scripts 1203
  adding scripts 1203

benefits of 1110
benefits of deploying 1195
building using command line 1251
CAB file 1194
Citrix Assistant 1190
conditions when Shortcut should be excluded or deleted 1219
conversion error and warning messages 1251
creating with InstallShield 1190
digitally signing 1224
hashes.txt file 1194
how requirements are applied at runtime 1202
how shortcuts are implemented 1215
how transforms are included 1197
Icons file 1195
icons.bin file 1195
including an existing shortcut 1218
inheritance of isolation options from folders to files 1214
inheritance of isolation options in the registry 1222
managing files and folders 1206
moving files and folders 1209
myapp.profile file 1194
overview of 1194
overview of Citrix Assistant 1190
overview of Citrix XenApp 1192
overview of isolation options 1212
Profile Manifest file 1194
renaming a shortcut 1220
Scripts folder 1195
setting file isolation options 1245
setting folder isolation options 1247
setting isolation options 1211
setting isolation options for folders and files 1214
setting language requirements 1202
setting operating system and service pack requirements 1201
setting registry isolation options 1221
setting Service Pack requirements 1250
shortcut requirements 1217
shortcuts and the isolation environment 1215
specifying whether users can update applications 1199
steps to create with Citrix Assistant 1191
Citrix XenApp 707, 708, 1190, 1192
about 1192
about COM+ applications 953
and services 952
benefits of 711
overview of 708, 1192
overview of Citrix profiles 705, 709, 1194
Citrix XenApp Assistant. See Citrix Assistant.
Citrix XenApp Profiles (*.profile) 729, 837
class IDs 1677
checking 1677
Class IDs view 1714
clean machines 496
alternate-language repackaging on 496
Repackager 496
Clear Sandbox 1145
Client Access Points (CAPS) 1799
Client Agent 1799
Close wizard to configure packages and machines 728
code signing 1652
certificate name in the store 1652
credentials 1652
Collect Product Information panel 550
Repackaging Wizard 550
collection 1799
Column selector 803
columns 810
changing order 810
selecting on the Packages and Machines tabs 803
COM+ applications
and Citrix profile 953
and Citrix XenApp 953
command line 466, 479, 481, 539, 1108, 1180, 1251, 1698
applying patches to packages during import 479
applying transforms to packages during import 479
build App-V application 563
build Citrix profile 563
build ThinApp application 563
creating a log file during import 481
in Application Manager 466
Predeployment Test 1860
running QualityMonitor 1698
running Repackaging Wizard from 539
command line parameters 865
Command Line property 763, 782
Command tab 1316, 1526
command-line configurations 98, 99, 132
adding 98
creating 132
deleting 132
deleting from existing tools in AdminStudio 99
editing 132
modifying for existing tools 99
Command-Line Configurations panel 134
Add Tool Wizard 134
command-line import 479
using a configuration file in Application Manager 479
command-line options 480, 482, 562, 1299, 1664
Application Isolation Wizard 1664
connecting to a specific Application Catalog 482

Repackager 562
using a configuration file to import multiple merge modules in Application Manager 480
using a configuration file to import multiple Windows Installer packages in Application Manager 480
when to use Dialogs view instead 1299
Command-Line Parameters dialog 416
command-line properties 124
Add Tool Wizard 124
Command-Line Properties dialog 124
DevLocation variable 124
InstallLocation variable 124
ProjectName variable 124
SharedPoint variable 124
SourcePackage variable 124
TargetDir variable 124
TargetFileName variable 124
Command-Line Properties dialog 124
Add Tool Wizard 124
COMMapping 674
Comments 1338
comments 1303
adding and editing setup property 1303
Company 1664
Company property 763, 781
Compatibility Solver Database 125
Completing the Application Isolation Wizard panel 1659
component conflicts 1370
component settings options 588, 659
Components view 412
Compress Wrapper MSI setting 961
Compressed property 763, 782
Compressed setting 1008
Compression 860
compression of ThinApp applications 1131
Compression Type 1162
configuration file 479, 480
Application Manager 479
importing multiple merge modules using 480
importing multiple Windows Installer packages using 480
using with Application Manager command-line import 479
configuration files 1664
Application Isolation Wizard 1664
Configuration Manager 1798
Configuration Manager Server 1799
saving a package on using Configuration Manager Web Console 1822
Configuration Manager Web Console 1797, 1799, 1802
access accounts 1810
advertisements 1799, 1819
Advertisements View 1841
and Microsoft Configuration Manager 1798
Client Access Points (CAPS) 1799
Client Agent 1799
collection 1799
configuring distribution settings 1803
distributing a package to Configuration Manager Server 1822
distributing packages 1802, 1804
Distribution Points 1799
distribution points 1811
Distribution Points View 1833
Distribution Settings page 1844
getting started 1802
Management Points 1799
modifying distribution settings 1822
Package Administration Page 1845
package definition file 1827
package settings 1806
programs 1812
reference 1825
saving a package on Configuration Manager Server 1822
Select a Configuration Manager Collection dialog box 1844
setting package configuration options 1805
setting package configuration options\>\ 1805
target collection 1799
viewing status of distributed package 1822
Configuration tab 132
Tool Properties dialog 132
configuring command lines 134
conflict analysis
using multiple source packages 1371
Conflict Application Resolution Definitions (CARDs) 1399, 1406, 1408, 1409, 1410, 1420, 1423, 1424, 1479, 1480
CARD02 1481
CARD04 1482
CARD05 1482
CARD06 1482
CARD07 1483
CARD15 1483
CARD18 1483
CARD19 1484
CARD20 1484
overview 1479
reference 1480
conflict identification 1370, 1378
running again 1378
Conflict Information dialog 1523
conflict options 1372, 1529
changing ConflictSolver default 1372

ConflictSolver 1529
Conflict Options panel 1545
Conflict Wizard 1545
conflict persistence 1379
deleting persisted data 1379
conflict resolution 1370
Conflict Resolution dialog 1524
conflict types 1372
changing default in ConflictSolver 1372
Conflict Wizard 1374, 1540, 1541, 1542, 1545
Conflict Options panel 1545
identifying conflicts 1374
MSI Source Information panel 1542
Source Package panel 1542
Source Type panel 1542
Summary panel 1545
Target Information panel 1545
Welcome panel 1541
conflicts 1370, 1374, 1377, 1379, 1380, 1382, 1383, 1384
automatically resolving in ConflictSolver 1382
checking for during import 1377
component 1370
conflict resolution process in ConflictSolver 1380
deleting persisted 1379
file extension 1370
files 1370
identifying with Conflict Wizard 1374
INI file 1370
manually resolving in ConflictSolver 1383
NT services 1370
ODBC resources 1370
product code 1370
registry 1370
resolving 1379
resolving directly in Windows Installer package 1384
resolving using transforms 1384
shortcut 1370
solution using virtualization 694
viewing identified 1377
Conflicts Tab 1494
Conflicts tab 1529
in ConflictSolver Options dialog 1529
Conflicts view 1499, 1509
ConflictSolver 1355, 1361, 1362, 1363, 1368, 1369, 1372, 1373, 1374, 1377, 1379, 1380, 1382, 1383, 1384, 1385, 1393, 1467, 1477, 1485, 1488, 1492, 1497, 1499, 1504, 1505, 1506, 1507, 1509, 1518, 1540, 1541, 1542, 1545, 1546, 1548, 1549, 1550, 1551
ACE reference in Predeployment Test 1901
All Patches View 1633
Associated Patches View 1633
Associated Patches view 1506
automatically resolving conflicts 1382
best practices 1356, 1358, 1371, 1374
changing default conflict options 1372
changing default conflict types 1372
changing default validation options 1368
changing resolution options 1373
changing the default validation file in 1368
checking for conflicts during import 1377
conflict persistence 1379
conflict resolution process 1380
Conflict Wizard 1374, 1540, 1541, 1542, 1545
Conflicts view 1499, 1509
ConflictSolver database 1361
ConflictSolver tab on InstallShield Editor Options dialog 1261
connecting to an existing Application Catalog 1355
Consolidated Patch Report 1637
context menus 1492
Crystal Reports 1393
custom 1467
custom reports 1393
deleting user-defined ACEs 1477
Dependencies View 1633
Dependencies view 1505
dialogs 1518
editing user-defined ACEs 1477
handling invalid Windows Installer packages 1369
ICEs and Validation Errors 1369
Impact Analysis View 1633
Impacts View on Patches Tab 1517
manually resolving conflicts 1383
Marimba NCP Files 1508
menus 1485
Merge Module view 1507
OS Snapshot view 1507
OSD Files 1508
Other Setup Types view 1507
Output Window 1494
Patch Impact Analysis Wizard 1619, 1630, 1638
Patch Impact Information dialog box 1538
Patch Impact view 1504
Patch Impacts View 1633
patches 1513
Patches Tab 1517
Patches tab 1513, 1516, 1633
Patches Tab Views 1513
Patches view 1633
performing patch impact analysis 1630
persistence 1379
Product View 1488, 1497

reimporting packages 1385
reports 1390, 1391, 1392
resolution and the Software Repository 2040
resolving conflicts 1379
resolving conflicts directly in package 1384
resolving conflicts using transforms 1384
Rules Wizard 1545, 1546, 1548, 1549, 1550
scanning packages for dependencies 1505
Tables view 1507
Test Results view 1504, 1563
Testing packages using PackageExpert 1504
testing packages using PackageExpert 1563
toolbar 1485
user-defined ACEs 1467
Validate Package function 1364
validating external Windows Installer packages 1362
validation 1361, 1362
validation after import 1364, 1365, 1366, 1367
validation during import 1363
Validation Explanation dialog 1526
Validation view 1503
Validation Wizard 1550, 1551
viewing PackageExpert test results 1574
viewing patch and patch impact information 1633
viewing patch impact analysis results 1630, 1637
ConflictSolver Options dialog 1528, 1529, 1530, 1533, 1534, 1535, 1536
Conflicts tab 1528, 1529
Duplicate Package tab 1530
Extended Attributes tab 1528, 1533
General tab 1528, 1533
Import tab 1528, 1534
Resolution tab 1528, 1535
Validate tab 1528, 1536
ConflictSolver Output Window 1494
Connect to a Microsoft SCCM Server panel 753, 772, 820, 849
Connect to an AdminStudio Application Catalog panel 753, 772, 819, 848
Connect to Application Catalog dialog 416, 430
Connect to Machine 751, 795
Connection to New Catalog Page (Predeployment Test) 1897
Consolidated Patch Report 1637
Contact Person 1338
Contents 802
context file 775
context menus 348, 1492
Application Manager 348
ConflictSolver 1492
contextual packages 2039
Control Access via Active Directory 1118, 1153
converting 572, 573, 577
.aot 573
.axt 573
.inc 573
.ipf 573
.isl 577
.txt 577
.wse 577
InstallScript MSI 538
InstallShield log files 577
legacy setups 572
Novell ZENworks projects 573
Repackager 3.x output 573
SMS projects 573
WinINSTALL projects 577
Wise Installation projects 577
converting packages 721
Copy In 798
Copy Out 798
copying 206
packages to multiple Application Manager groups 206
roles 178
Create a new transform file option 1320
Create Report dialog box 633
CreateSetupExe 674
creating
roles 177
creating new workflows 107
credentials 1652
Crystal Reports 1390
generating in ConflictSolver 1393
-cs 562, 563
custom actions
and Citrix profiles 951
custom installation 1275
preventing feature display 1275
custom setups 1297
disabling 1297
custom tables 1534
Custom Test Results Editor dialog box 1611
customer
Customer Experience Improvement Program (CEIP) 126
Customer Experience Improvement Program (CEIP) 126
canceling membership 126
joining 126
Customize dialog 1526
Customize dialog box 1316

D

-D 466
data sources 1290
  adding new 1290
data type appearance 570
  changing Repackager 570
database
  sample 193
Database Name 820, 849
DDE Application setting 1023
DDE Command setting 1023
DDE Ifexec setting 1023
DDE Topic setting 1023
debug log 95
  generating 95
debug messages
  viewing 806, 881
Default Destination 1274
Default Organization 1274
Default.ini 681
Deleted Files view 650
Deleted Registry Entries view 651
Deleted.isr 681
deleting
  filters in PackageExpert 1580
  groups 142
  roles 178
  test results in PackageExpert 1575
  users 142
deleting added setups 510
dependencies 593, 1360
  detecting in Repackager 593
  scanning for 1360
  viewing in ConflictSolver 1633
dependencies of a virtual package 969
Dependencies view 1010, 1360, 1505
Dependency view 412
deploying SMS 1307
Deployment & Conflict Test Selection panel 1872
Deployment Status 1693
  install or configure products or features 1693
  reinstall components 1695
  reinstall features 1694
  verify files 1692
  view properties 1692
Deployment Status view 1724
deployment testing 1676
  Predeployment Test 1851, 1852
deployment tests 1713
  executing all from command line 1713
  executing all from Interface 1713
  running silently 1713
description file 266
  Application Manager extended attributes 266
Description property 1274
  editing for features 1274
description property 1275
Destination Group panel 450
  Import Wizard 450
destination variable 1275
DevLocation 124
diagnostic tools
  adding to App-V application 1054, 1101
  adding to Citrix profile 1200, 1244
  adding to ThinApp application 1119, 1165
Diagnostic Tools dialog box 1054, 1101, 1119, 1165, 1200, 1244
Diagnostics 860
dialog 1298
  editing properties for 1298
dialog boxes 1518
  ConflictSolver 1518
Dialog Properties dialog 1318
dialog sequences 1297
  restoring 1297
dialogs 122, 123, 124, 130, 1296, 1297, 1299, 1316, 1318, 1659
  About AdminStudio 122
  Add New Tool 123
  AdminStudio Interface 122
  AdminStudio Options 124
  Advanced Options 1659
  Command-Line Properties in Add Tool Wizard 124
  Dialog Properties dialog 1318
  hiding during UI sequences 1296
  suppressing License Agreement 1297
  suppression issues 1299
  Tool Properties 130
  Transform Summary 1316
  working with in Tuner 1296
Dialogs view 1296, 1297, 1298, 1299, 1318, 1338
  Dialog Properties dialog 1318
  disabling custom setups from 1297
  editing dialog properties in 1298
  hiding dialogs during UI sequences from 1296
  restoring dialog sequences from 1297
  suppressing License Agreement dialog from 1297
  when to use over MSI command-line options 1299
digital certificates 1652, 1653
  certificate store 1653
Digital Signature tab 1661

Advanced Options dialog 1661
digital signatures 1652, 1653, 1661
Application Isolation Wizard 1661
private key 1653
software publishing credentials 1653
digitally signing
a Citrix profile 1224
Direct Editor 1308, 1309, 1342
adding a new record 1309
adding a new row 1309
editing packages 1308
finding and replacing in 1309
from Validation tab 1309
directories 613
exclusions 613
Directories and Files Excluded During Analysis dialog box 672
directories and subdirectories 605
excluding 605
directory
package selection rules 826
directory exclusions 613
Directory Service Attributes 143
Directory Services 177
automatic login 152
creating Directory Service Attributes 143
creating new role with Directory Services 177
Directory Service Attributes 143
managing Directory Services configurations 143
Directory Services Page 160
Disable Log Monitor Tracing 1162
Disable Modify Button 1338
Disable Remove Button 1338
Disable Repair Button 1338
disabling 1302
Modify button in Add/Remove Programs 1302
Remove button in Add/Remove Programs 1302
Repair button in Add/Remove Programs 1302
disabling a Subscription 318
disabling a user account 141
disconnecting from Application Catalog 203
distributing packages 1802
distribution 1304, 1305, 1729
copying to FTP server 1305
copying to network location 1304
preparing packages 1729
Distribution Output panel 1786
Distribution Wizard 1786
Distribution Points 1799
distribution points 1811
distribution settings 1822
modifying in Configuration Manager Web Console 1822
Distribution Summary panel 1785
Distribution Wizard 1785
Distribution Type panel 1760
Distribution Wizard 1760
Distribution Wizard 1729, 1730, 1731, 1733, 1734, 1735, 1737, 1740, 1741, 1742, 1748, 1752, 1757, 1759, 1760, 1761, 1766, 1767, 1768, 1769, 1770, 1771, 1772, 1773, 1774, 1777, 1779, 1781, 1783, 1785, 1786, 1797
Administrative Install panel 1761, 1777
Advanced Marimba Options dialog box 1770
Altiris 1732
Altiris Integration Panel 1764
Altiris XML Template 1765
creating administrative installations using 1730
distributing packages to FTP servers 1731
distributing packages to network locations using 1737
distributing packages using Configuration Manager Web Console 1797
distributing to Microsoft Configuration Manager 1738
Distribution Output panel 1786
Distribution Summary panel 1785
Distribution Type panel 1760
FTP Location panel 1766
Installation Package panel 1785
LANDesk Integration Panel 1766
ManageSoft Package and Environmental Settings Panel 1734, 1767, 1768
Marimba Integration panel 1768, 1769
Marimba Patch Information Panel 1769
Network Location panel 1771
preparing for LANDesk distribution 1733
preparing for ManageSoft distribution 1734, 1767, 1768
preparing packages for distribution using 1729
preparing packages for Microsoft Configuration Manager 1797
preparing packages for SMS 1740
preparing packages for Tivoli 1742
preparing packages for ZENworks desktop application distribution 1748
preparing packages for ZENworks server distribution 1752
script-based setup.exe 1757
SMS Distribution Panel 1771
SPD Parameters panel 1773, 1774
Tivoli Integration panel 1772
Tivoli Settings panel 1772
using Marimba Channel Publishing 1735
Welcome panel 1759
ZENworks Configuration Management 1743, 1786
ZENworks Desktop Application Panel 1779

ZENworks Server Distribution - Distributor Panel 1783
ZENworks Server Distribution - Object Panel 1781
Distribution Wizard for ZENworks Configuration Management
ZENworks Configuration Management 1743, 1786
Division 1664
DLL Hell 1647
avoiding 1647
DLL Information tab 1523
DLL-Based ACEs panel 1550
Rules Wizard 1550
dual-mode patches 1629
duplicate package names 310
during auto import 310
Duplicate Package options 1530
ConflictSolver 1530
Duplicate Package tab 1530
in ConflictSolver Options dialog 1530
DuplicateFileData ACE 1432
Dynamic Dependency Scanner 597
Dynamic Suite Composition 969
configuring for an App-V package 969
dynamic suite composition 1072
Dynamic Suite Composition page 1093

E

Edit menu 800
Edit Registry Key dialog 674
Edit Registry Key Dialog Box 674
editing packages 1308
Direct Editor 1308
EnablePathVariables 674
Enforce Security Descriptors setting 1009
engines 489
including InstallScript engines with installation 489
enhancement
Customer Experience Improvement Program (CEIP) 126
enhancing packages 1253
ensuring package quality 1667
Enterprise Server Web tools
evaluating 68
Error -4308 - VM failed to start up 882
Error -4309 - VM failed to shut down 882
Error -4310 - Failed to connect to VM 883
Error -4312 - Failed to prepare Repackager 884
Error -4313 - Failed to access the package 885
Error -4314 - Failed to copy repackaged output from virtual machine 886
Error -4315 - Failed to send command to VM 886
Error -4316 - Failed getting response from VM 887
Error -4317 - Failed running pre-snapshot 887
Error -4318 - Failed running post-snapshot 888
Error -4319 - Failed running package installation 888
Error -4320 - Failed creating folder on VM 889
Error -4333 - Preparing command-line... 890
Error -4380 - Failed to prepare AppV 891
Error -4388 - Failed preparing for pre-snapshot 892
Error -4389 - Failed connecting to server 893
Error -4390 - Failed connecting to image 893
Error -4391 - Failed to reboot 894
Error -4395 - Failed to create VM directory 894
Error -4409 - Failed to delete package cache folder 895
error C1083 1474
error control level 1294
setting 1294
Error Control setting 1027
Errors property 797
evaluating
AdminStudio Enterprise Edition client tools 69
AdminStudio Enterprise Server Web tools 68
AdminStudio's Microsoft App-V support 70
batch convert option 70
evaluation files 1268, 1272
evaluation license file
obtaining 1942
evaluation mode 737
Event setting 1024
Excluded Processes dialog box 560
excluding files 604
exclusion list 681
specifying directory 92
exclusion lists 1683
Exclusion view 412
exclusions 604, 605, 606, 607, 608, 609, 610, 612, 613, 614, 615, 616, 617, 618, 668, 669, 670, 671
all files in a directory 605
all shortcuts in a directory 607
anti-virus software 618
configuring in Repackager 604
directories and subdirectories 605
directory 613
editing existing file 613
editing existing INI file 616
editing existing registry 617
external configuration file in Repackager 608
file 612
files 669
INI file 614
INI file sections 606, 615
INI files 606, 670

modifying Repackager external configuration file 609
OS snapshot 609
OS Snapshot Wizard 610
OS Snapshot Wizard global 610
project 609
registry 671
registry data 617
registry keys 605
registry values 606
removing existing file 614
removing existing INI file 616
removing registry 618
Repackager 610
Repackager file 604
Repackager global 610
repackaging 609
resetting to default values 668
shortcuts 607
shortcuts from subdirectories 607
specific file extensions 612
with specific file extensions 612
Exclusions Editor 609, 610, 612, 613, 614, 615, 616, 617, 618, 668, 669, 670, 671, 672, 673, 674
About dialog box 674
Choose Registry Key dialog box 673
Directories and Files Excluded During Analysis dialog box 672
directory exclusions 613
Edit Registry Key dialog box 674
editing existing file exclusions 613
editing existing INI file exclusions 616
editing existing registry exclusions 617
excluding INI file sections 615
File Exclusion Information dialog box 672
files 612
Files tab 669
INI File Exclusion Information dialog 673
INI files 614
INI Files tab 670
Keys Excluded During Registry Analysis dialog box 673
menus 668
OS Snapshot Wizard exclusions 610
reference 668
registry data 617
Registry tab 671
removing existing file exclusions 614
removing existing INI file exclusions 616
removing registry exclusions 618
Repackager exclusions 610
resetting exclusions to default values 668
specific file extensions 612
Exclusions tab 640
experience
Customer Experience Improvement Program (CEIP) 126
expiration 1145, 1176
Expiration Message 1179
Expire Period 1178
Explore 801
Expression Builder dialog 1527
extended attribute options 1533
ConflictSolver 1533
Extended Attribute Property dialog 423
extended attributes 265, 266, 268
and Workflow Manager 268
description file in Application Manager 266
in Application Manager 265
using in Application Manager 265
Extended Attributes tab 1533
in ConflictSolver Options dialog 1533
Extended Attributes view 381, 400, 403, 404
Application Manager 381, 400, 403, 404
external configuration file 608, 609
modifying Repackager 609
specifying in Repackager 608
ExtraHKCRPermanent 674

F

-F 466
failsafe modes
setting up 1963, 1964
FAQ 112
fatal error C1083 1474
feature 1273
changing visibility 1273
setting initial state of 1273
feature advertisement 1276
feature block 1 1104
Feature Block 1 Size setting 1008
feature block 2 1104
feature blocks 1076
FEATURE lines
overview of 1973
feature properties 1275
features 1274
editing description for 1274
Features view 1274, 1325
Description property 1274
FILE 784, 1053, 1086
file associations 1678
checking 1678
File Associations view 1715

file conflicts 1370
File Exclusion Information dialog box 672
file exclusions 604, 613, 614
  editing existing 613
  removing existing 614
  Repackager 604
file extension conflicts 1370
File Isolation Options dialog box 1245
File Locations tab 1316
File menu 800
files 1277, 1278, 1279
  adding 1277
  displaying from base Windows Installer package 1278
  excluding 604
  exclusion of specific extensions 612
  exclusions 612, 669
  preventing installation of from the Windows Installer package 1278
  removing added 1279
  storing in CAB in Tuner 1279
Files & Folders page 1153
  creating a new folder 1058, 1123
  deleting files and folders from a ThinApp application 1124
  deleting files and folders from an App-V application 1059
files and folders 1277
Files and Folders view 645, 1277, 1278, 1279, 1327
  adding files in 1277
  displaying files from the base Windows Installer package 1278
  preventing installation of files from the Windows Installer package 1278
  removing added files from 1279
Files page 1087
Files Report 1390, 1392
Files tab 669
  Exclusions Editor 669
Files view 413
Files/Components view 381
Filesystem 786, 860
Filter view 1607
filters
  deleting in PackageExpert 1580, 1584, 1607
  editing in PackageExpert 1579, 1583, 1607
Find dialog 421
finding and replacing 1309
  using Direct Editor 1309
Finishing INI File Import panel 1345
  Import REG File Wizard 1345
Finishing Registry Import panel 1346
  Import REG File Wizard 1346
First Action Delay setting 1029
First Error setting 1029
FlexWrap
  advanced license settings 1956
  applications to which it can be applied 1910
  basic license settings 1955
  benefits 1910
  checking the status of the license server 1949
  communication issues 1940
  components of 1966
  creating a new license 1951
  creating an options file 1960
  installation notes 1940
  installing FlexWrap license file 1942
  license parameters 1968
  license server overview 1938
  overview 1910
  overviewX 1929
  platform issues 1939
  policies and options 1954
  purposes of 1910
  saving a license file 1961
  setting up failsafe modes 1963
  setting up license server failsafe modes 1964
  setting up regular (license) failsafe modes 1963
  specifing license parameters 1968
  starting the license server 1945
  stopping the license server 1949
  terminology 1966
  testing a wrapped application 1962
  tracking projects with 1951
  understanding license file FEATURE lines 1973
  using a wrapped application 1916
  wrapping an executable 1951
FlexWrap license server
  checking the status 1949
  starting 1945
  stopping 1949
FlexWrap License Server Component 1937
Flicense file
  installing 1942
Folder Isolation Options dialog box 1102, 1166, 1247
frequency 124
frequently asked questions 112
FTP Location panel 1766
  Distribution Wizard 1766
FTP server 1305
  copying installation to 1305
FTP servers 1731
  distributing packages to using Distribution Wizard 1731
Full isolation option 1128

G

General Information panel 1546
  Rules Wizard 1546
General Information tab 1519
General Information view in Virtual Package Editor 1007
general options 1533
  ConflictSolver 1533
General tab 639, 1533
  in ConflictSolver Options dialog 1533
Generated property 797
generic transforms 1267, 1316
  creating 1267, 1316
Get Latest Version 292
Global Exclusions Editor
  exclusion list 92
Global Exclusions list 92
Group Dependencies setting 1028
Group Properties dialog 423
Group view 378
grouping 811
  lists 811
groups 205, 206, 207
  adding in Application Manager 205
  copying packages to multiple Application Manager 206
  deleting 142
  deleting from Application Manager 206
  editing properties in Application Manager 206
  moving in Application Manager 207
  organizing products into in Application Manager 205
  removing from Application Manager 206
Groups panel 453, 455
  Merge Wizard 455
  Package Auto Import Wizard 453
guest account 153, 155
  setting up 153
Guest Agent 856
Guest Password property 750, 794
Guest Username property 750, 794
GuestAgent Path property 750, 794

H

Hard Timeout property 764, 783
hashes file 710
hashes.txt 710
help
  Help Library conventions 73
help files 105, 1678
  associating with tasks 105
  checking 1678
Help Files view 1715
Help Telephone 1338
Help URL 1338
Help view 1323
Host setting 1010
HTTP 784, 1053, 1086
HTTPS 784, 1053, 1086
Hyper-V server 830

I

-I 466
ICE errors 1365
  deleting 1367
  supressing 1366
  viewing after validation 1365
ICEs 1272
ICO file 702
icons 344, 347, 1488
  Merge Modules View 347
  Product View 344, 1488
  used on Results tab 732
  used on Selected Package List panel 724
-IF 466, 479
-IMM 466
Impacts View on Patches Tab 1517
  ConflictSolver 1517
import 1377
  checking for conflicts during 1377
Import Conflict Options panel 1344, 1345
  Import INI File Wizard 1344
  Import REG File Wizard 1345
Import INI File panel 1344
  Import INI File Wizard 1344
Import INI File Wizard 1344
  Import Conflict Options panel 1344
  Import INI File panel 1344
  Welcome panel 1344
import options 1534
  ConflictSolver 1534
Import REG File Wizard 1345, 1346
  Finishing INI File Import panel 1345
  Finishing Registry Import panel 1346
  Import Conflict Options panel 1345
  Import Registry File panel 1345
  Welcome panel 1345
Import Registry File panel 1345
  Import REG File Wizard 1345
Import Tab 352, 1494
Import tab 1534
  in ConflictSolver Options dialog 1534

Import Wizard 437, 438, 450
Destination Group panel 450
Summary panel 450
Welcome panel 438
importing 207, 211, 213, 215, 224, 225, 226, 227, 228, 229, 480
ad-hoc data 209
Marimba NCP files 207, 226, 227
merge modules 207, 224
merge modules and Windows Installler packages simultaneously into Application Manager 480
MSI packages 207, 211, 215
OS patches using OS Security Patch Wizard 1622
OS Snapshots 207, 226
OS snapshots 225
other setup types 228, 229
specifying duplicate package identifiers in Application Manager 229
transforms 215
virtual packages into the Application Catalog 218
Windows Installer packages 207, 211, 213
importing data
importing into the Software Repository 287
performing ad-hoc import 214
Include App-V Launcher setting 960
Include SFT in Wrapper MSI setting 961
including additional MSIs in virtual package 1139, 1163
Index 802
INF file 1629
INI file 1344
Import INI File Wizard 1344
INI file actions 1287
modifying 1287
INI File Changes view 383, 401, 405
INI file conflicts 1370
INI File Exclusion Information dialog box 673
INI file exclusions 616
editing existing 616
removing existing 616
INI file keys 1287, 1289
modifying 1287
removing 1289
INI file sections 606, 615
excluding 606, 615
INI file values 1287
modifying 1287
INI files 606, 614, 670, 1285, 1286, 1287, 1288, 1289
adding 1286
adding new keys 1287
adding sections to 1287
exclusions 606, 614, 670
importing existing 1286
removing 1288
removing sections from 1289
INI Files tab 670
Exclusions Editor 670
INI Files view 649, 1286, 1287, 1288, 1289, 1333
adding INI files in 1286
adding new INI file keys 1287
adding sections to INI files in 1287
importing existing INI files into 1286
modifying INI file keys 1287
removing INI file keys from 1289
removing INI files from 1288
removing sections from INI files from 1289
Initial Configuration Complete panel 728, 836
initial state 1273
setting for features 1273
initial state property 1275
Install Microsoft App-V Client option 769
install monitoring 497, 498
installation 1304, 1305
copying to FTP server 1305
copying to network location 1304
Installation Designer
opening 1035
installation monitoring 497, 498
excluding processes from 560
Installation Package panel 1785
Distribution Wizard 1785
-installdir 562
installing
Novell ZENworks Desktop Management Agent 574
InstallLocation 124
InstallScript engines 489
InstallScript MSI 538
converting 552
converting to Basic MSI with InstallScript support 538
converting to Repackager project 538
InstallScript MSI Conversion Output panel 555
Repackaging Wizard 555
InstallScript MSI Identified panel 552
Repackaging Wizard 552
InstallScript Scan 489, 538
InstallShield
about the virtualization Assistants 1033
App-V Assistant 1036
Citrix Assistant 1190
integration of App-V Assistant 1034
integration of Citrix Assistant 1034
integration of ThinApp Assistant 1034
integration of virtualization Assistants 1034

ThinApp Assistant 1108, 1109
InstallShield 2008 Professional
compared to InstallShield Editor 1262
InstallShield Editor 596, 1253
ConflictSolver tab on InstallShield Editor Options dialog 1261
customizing and authoring packages 1253
Dynamic Dependency Scanner 597
editing generated Repackager projects 596
features 1262
integration with Application Catalogs 1256
Options dialog 1261
vs. InstallShield 2008 Professional 1262
InstallShield Editor project 578
building in Repackager 578
InstallShield log files 577
converting to Repackager project 577
InstallShield Professional 554, 662
SmartScan Wizard 663
InstallShield Professional Setup panel 554
Repackaging Wizard 554
InstallShield Professional setups 619
scanning in Repackager 619
InstallShield SmartScan options 588, 660
Interface 89
using AdminStudio 89
Interm directory 706, 1115
internal consistency evaluators 1272
invalid Windows Installer package 1270
handling 1270
-is 562, 564
ISCSAboutNew 1352
isolated components 596
Isolated Components Design dialog 1662
IsolatedComponents 1664
isolating applications 1647
Isolating Windows Installer Packages Using Application Isolation Wizard 597
isolation 596, 1647, 1650, 1653, 1655, 1656
applications 1653
filtering file listings when manually configuring 1656
methods 1650
modifying default recommendations 1655
reasons not to do 1647
reasons to do 1647
Isolation Method panel 1658
Application Isolation Wizard 1658
isolation methods 1650
assemblies and manifests 1650
Windows Installer isolated components 1650
isolation options
in ThinApp Assistant 1128
inheritance from folders to files in a Citrix profile 1214
inheritance from folders to files in a ThinApp application 1129
inheritance from folders to files in an App-V application 1065
inheritance in the registry in a Citrix profile 1222
inheritance in the registry in a ThinApp application 1137
inheritance in the registry in an App-V application 1072
overview of 1063, 1126, 1212
setting for folders and files in App-V Assistant 1064
setting for folders and files in Citrix Assistant 1214
setting for folders and files in ThinApp Assistant 1129
setting in App-V Assistant 1062
setting in Citrix Assistant 1211
setting in registry for Citrix profile 1221
setting in registry for ThinApp application 1136
setting in ThinApp Assistant 1126
Isolation Options dialog box 634
isolation tests 1686, 1688
performing 1686, 1688
Isolation Tests view 1723
isrepackager.context.ini 775
ISRepackager.ini 681, 684
isrepackager.ini 92
ISRIsolation 601
issnapshot.ini 457

J

Job Details Report (Predeployment Test) 1888
Job Manager 2035
about Templates 2053
available Job Steps 2037
configuration options 2057
conflict resolution and the Software Repository 2040
contextual packages 2039
creating a new Custom Job 2043
creating a new Job 2042
creating a new Template 2054
Custom Jobs 2039
deleting a Job 2052
deleting a Template 2057
dynamic group content identification 2039
editing a Job 2048
editing a Template 2056
email alerts 2050
evaluating 68
executing jobs 2061
Job Manager Engine 2059
rescheduling a Job 2049

selecting packages imported in previous Job Steps 2039
setting detection options 2057
Template Job vs. Custom Job 2038
user permissions 2041
viewing Job Manager Engine status 2062
viewing Job status 2051
Job Manager Engine 2059
adding an Application Catalog to 2060
deleting a connection to 2060
executing jobs 2061
opening 2060
setting conflict detection options 2063
setting validation options 2064
shutting down and restarting Application Catalog 2061
viewing status of in Job Manager 2062
Job Summary Report (Predeployment Test) 1885

K

KeyPath ACE 1434
Keys Excluded During Registry Analysis dialog 673

L

-L 466
LANDesk distribution 1733
creating with Distribution Wizard 1733
LANDesk Integration Panel 1766
Distribution Wizard 1766
language requirements
setting in Citrix profile 1202
Launch App-V Application dialog box 1103
Launch Package for Testing 746, 769
Launcher 786, 860
LDAP attributes 163
Active Directory 163
Legacy Client 1799
legacy setups 489, 504, 572
converting 572
repackaging 489, 504
Legacy Upgrade Wizard
upgrading legacy Application Catalogs 204, 464
license
advamced settings 1956
basic settings 1955
saving a wrapped application 1961
License Agreement dialog 1297
suppressing 1297
license file
creating in FlexWrap 1951
obtaining an evaluation license 1942
saving 1961
license parameters 1968
specifying 1968
license server
checking the status of 1949
overview 1938
starting 1945
stopping 1949
LimitedUI 674
limiting tool accessibility 101
list features 808
lists
changing column order 810
changing columns displayed 809
customizing 810
grouping 811
resizing 810
sorting 808
load order group 1294
setting 1294
Location panel 1346
Packaging Wizard 1346
lockdown and runtime tests 1686, 1687
performing 1686
running in restricted environments 1687
Lockdown and Runtime Tests view 1718
lockdown testing 1685
log file 481
creating during command-line import 481
Log Monitor 1140
Log Monitor tracing options 1140
log report 804
logging in to AdminStudio Enterprise Server 152, 153
anonymous 153
anonymous account 155
automatically 152
forgetting your password 152
logging in to Workflow Manager
guest account 153, 155

M

Machine Import Wizard 795
Machine property 749, 793
Machine Result Summary Report (Predeployment Test) 1893
Machines tab 790
context menu commands 795
editing virtual machine properties 748
properties 793
selecting columns to display 803
viewing machine information 792

Management Points 1799
ManageSoft distribution 1734, 1767, 1768
  creating with Distribution Wizard 1734, 1767, 1768
ManageSoft Package and Environmental Settings Panel 1734, 1767, 1768
Mandatory setting 1011
manifest
  in Citrix profile 710
Manifest and Assembly Design dialog 1662
Manifest file 702
manifest files 596
manifest options 1660
  Application Isolation Wizard 1660
Manifest Options tab 1660
  Advanced Options dialog 1660
Manifests 1664
manifests 598, 1651, 1666
  application 598, 1651
  assembly 598, 1651
  checking in QualityMonitor 1681
  checking with QualityMonitor
    QualityMonitor
      checking manifests 1681
  examples of 1666
  testing 1681
Manifests view 1716
Manual Test Perform dialog box 1613
Manual Test Resolve dialog box 1614
manual tests
  about 1588
  adding 1588
  creating in PackageExpert 1590
  Manual Test Perform dialog box 1613
  Manual Test Resolve dialog box 1614
  resolving 1591, 1592
  running 1591
  sample 1593
Mapped Network Drive Changes go to Sandbox 1152
Marimba 1735
  patch information 1769
  using channel publishing in Distribution Wizard 1735
Marimba advanced options 1770
Marimba Advanced Options dialog box 1770
Marimba Integration panel 1768, 1769
  Distribution Wizard 1768, 1769
Marimba NCP File View 403, 1508
Marimba NCP files 207, 226
  importing 207, 226, 227
  manifest.ncp 227
Marimba Patch Information Panel 1768
  Distribution Wizard 1769
menus 119, 341, 630, 668, 800, 1311, 1485, 1699
  AdminStudio 119
  Application Manager 341
  Application Manager context 348
  ConflictSolver 1485
  ConflictSolver context 1492
  Exclusions Editor 668
  QualityMonitor 1699
  Repackager 630
Merge Child Keys option 977
Merge Module view 400, 411, 1507
merge modules 224, 480
  importing 207, 224
  importing multiple using a configuration file 480
  importing simultaneously with Windows Installer packages 480
Merge Modules View 347
  Application Manager 347
  icons 347
Merge Modules view 388
merge replication
  using SQL Server 332
Merge Wizard 454, 455, 456
  Groups panel 455
  Progress panel 456
  Source Application Catalog panel 455
  Welcome panel 455
Merged isolation option 1128
merging Application Catalog information 299
Method Selection panel 547
  advanced settings 560
  Repackaging Wizard 547
metrics 1477
  viewing for ACEs 1477
Microsoft Application Virtualization (App-V). See App-V applications.
Microsoft Application Virtualization. See App-V applications.
Microsoft App-V applications. See App-V applications.
Microsoft App-V Assistant. See App-V Assistant.
Microsoft App-V Packages (*.sft) 729, 837
Microsoft App-V Sequencer 1040, 1047
  compared with App-V Assistant 1040
  interface comparison to App-V Assistant 1047
  quick start guide 1047
Microsoft Configuration Manager 1798
  using Distribution Wizard to prepare packages for 1738
Microsoft Configuration Manager Server 1799
Microsoft Hyper-V Server 716, 726, 745, 833
Microsoft patches
  Consolidated Patch Report 1637
  performing patch impact analysis 1630

Properties dialog box 1644, 1645, 1646
viewing information in ConflictSolver 1633
Microsoft SCCM 722, 818
Microsoft SCCM Server 752
Microsoft System Center Configuration Manager 1738, 1798
integration with Configuration Manager Web Console 1798
MIF 1307
instructing SMS to create file 1307
Minimum Client Version setting 1008
-mm 562, 564
-mode 562, 564
Modify button 1302
disabling in Add/Remove Programs 1302
modifying INI file actions 1287
modifying INI file keys 1287
modifying INI file values 1287
MoveFileData ACE 1435
-mp 562, 564
-ms 562, 564
MSI command-line options 1299
when to use Dialogs view instead 1299
MSI Doctor 1689
install or configure products or features 1693
reinstall components 1695
reinstall features 1694
verify files 1692
view properties 1692
MSI packages 211, 229, 1369, 1384
building in Repackager 578
handling invalid 1270
handling invalid in ConflictSolver 1369
importing 207, 211
prevalidating 1269
resolving conflicts directly within 1384
MSI Source Information panel 1542, 1551
Conflict Wizard 1542
Validation Wizard 1551
MSIs
about repackaging MSIs 764
MultiUserShortcuts 674

N

NCP files 226, 227
importing 226, 227
Marimba NCP File View 403, 1508
NCP View 403, 1508
nested .msi custom action 685
network location 1304
copying installation to 1304
Network Location panel 1771
Distribution Wizard 1771
network locations 1737
distributing packages to using Distribution Wizard 1737
New Workflow Project Wizard 107, 135, 136
creating workflows with 107
reference 135
Source Package panel 135
Target Directory and File Name panel 136
Welcome panel 135
Workflow Selection panel 135
New Workflow Project wizard 109
example using 109
new workflows 107
creating 107
NewComponents 1664
notes 104
creating for tasks 104
Novell ZENworks Desktop Management Agent
installing 574
Novell ZENworks projects 573
.aot and .axt files 573
application object template files 574
converting to Repackager project 573
NT service arguments 1292
NT service dependencies 1293
setting 1293
NT service description 1293
setting 1293
NT service display name 1293
setting 1293
NT service type arguments 1292
NT services 1292
NT services conflicts 1370
NT Services view 1292, 1293, 1294, 1295, 1335
setting the NT service arguments in 1292
setting the NT service dependencies in 1293
setting the NT service description in 1293
setting the NT service display name in 1293
setting the NT service error control level in 1294
setting the NT service load order group in 1294
setting the NT service overall install result in 1294
setting the NT service start name and password in 1295
setting the NT service start type in 1295
setting the NT service type in 1295

O

-o 562, 564
ODBC data source attributes 1290, 1291, 1292
adding new 1290

editing 1291
removing 1292
ODBC data sources 1291, 1682
checking 1682
removing existing 1291
ODBC Data Sources view 1717
ODBC driver attributes 1290, 1291
adding new 1290
editing 1291
removing 1291
ODBC drivers 1682
checking 1682
ODBC Drivers view 1717
ODBC resource conflicts 1370
ODBC resources 1289
ODBC Resources view 1290, 1291, 1292, 1334
adding new data sources 1290
adding new ODBC data source attributes 1290
adding new ODBC driver attributes 1290
editing ODBC data source attributes 1291
editing ODBC driver attributes 1291
removing existing ODBC data sources from 1291
removing ODBC data source attributes from 1292
-of 562, 564
-onp 565
Open a recent transform file option 1322
Open an existing transform file option 1323
Open Project panel 717, 816
Open QualityMonitor Project dialog 1706
operating systems 716
Optimize for Offline Use 1078
Optimize for Streaming 1077
opt-in or opt-out of Customer Experience Improvement Program (CEIP) 126
options
FlexWrap 1954
Options dialog 124, 637, 1528, 1529, 1530, 1533, 1534, 1535, 1536, 1707
Conflicts tab in ConflictSolver 1529
Duplicate Package tab in ConflictSolver 1530
Extended Attributes tab in ConflictSolver 1533
General tab in ConflictSolver 1533
Import tab in ConflictSolver 1534
in AdminStudio 124
Resolution tab in ConflictSolver 1535
Validate tab in ConflictSolver 1536
Options dialog box 1316
options file
creating in FlexWrap 1960
Options.ini file 674, 681
options.ini file
OtherComponentFileExtensions 674
support for user-defined extensions 674
Original InstallShield Professional Setup panel 662
Repackaging Wizard 662
OS requirements 658
option to match legacy system's OS requirements for SmartScan 658
OS Security Patch Wizard 1619, 1622
about missing patch files 1629
dual-mode patches 1629
OS Snapshot
display limits 1533
Shortcuts view 402
OS snapshot 609
exclusions 609
OS Snapshot Summary panel 457
OS Snapshot Wizard 457
OS Snapshot view 400, 1507
OS Snapshot Wizard 338, 339, 340, 456, 457, 610
Analysis Options dialog 457
best practices 338
concept 338
configuring analysis options 339
exclusions 610
OS Snapshot Summary panel 457
performing OS Snapshots 340
Performing Snapshot panel 457
Project Information panel 456
reference 456
Welcome panel 456
OS Snapshots
importing 207, 226
OS snapshots 207, 225, 338
capturing 338
importing 225
moving in Application Manager 207
taking 338
OSD 403, 1508
Marimba NCP File View 403, 1508
NCP View 403, 1508
OSD file 702
editing 703
osguard.cp file 1011
other setup types 229
importing 228, 229
Other Setup Types View 402, 1507
OtherComponentFileExtensions 674
OtherFilesNewComponents 674
Output Cache Path property 750, 794
output directory
setting 837

output formats 729
Output Path 858
Output Tab 352, 1494
Output Window 352, 1315, 1494
  Conflicts Tab 352, 1494
  Import Tab 352, 1494
  Output Tab 352, 1494
  Package Auto Import Tab 352, 1494
  Patch Impact Tab 352, 1494
  Search Results Tab 352, 1494
  Validate Tab 352, 1494
Output window 731, 802
output window
  in PackageExpert 1602
overall install result 1294
  setting 1294
Override Child Keys option 977

P

-P 466
package 1269, 1270, 1804
  distributing to Configuration Manager Server using Configuration Manager Web Console 1822
  distributing using Configuration Manager Web Console 1804
  handling invalid Windows Installer 1270
  modifying distribution settings in Configuration Manager Web Console 1822
  prevalidating MSI 1269
  saving a package on Configuration Manager Server using Configuration Manager Web Console 1822
  selecting for distribution using Configuration Manager Web Console 1804
Package Administration Page (Configuration Manager Web Console) 1845
Package Auto Import 300, 301, 304, 305, 306, 307
  Application Manager 300, 301, 307
  deleting a link to a network directory 310
  deleting remote Application Catalog link 306
  duplicate package names during import 310
  editing a link to a network directory 309
  editing in Application Manager 304
  forcing an import of packages in a network directory 310
  in Application Manager 304, 305, 306
  linking to a network directory 307
  refreshing linked packages in Application Manager 305
Package Auto Import Properties Dialog 425, 427
Package Auto Import Tab 352, 1494
Package Auto Import Wizard 306, 451, 452, 453, 454
  Folder Panel 453
  Groups panel 453, 454
  Monitoring Process Selection panel 452
  Remote Link Application Catalog panel 452
  Summary panel 454
  Welcome panel 451
package configuration options 1806
  access accounts 1810
  advertisements 1819
  distribution points 1811
  package settings 1806
  programs 1812
  setting in Configuration Manager Web Console 1805
package content 1277
  configuring 1277
package conversion options 587, 658
package definition file 1305, 1827
  Configuration Manager Web Console 1827
  creating 1305
package definition file (PDF) 1827
Package GUID setting 1008
Package Import Wizard 718, 752, 839
Package Information page 1083
Package Optimization 860
Package Optimizations
  Optimize for Offline Use 1078
  Optimize for Streaming 1077
package optimizations 1076
Package Optimizations dialog box 1104
Package property 781
Package Publish Wizard 771, 845
package quality 1667
Package Report 1390, 1391
package settings 1806
Package Summary panel 1347
  Packaging Wizard 1347
Package Validation view 1269, 1323
  prevalidating a Windows Installer package 1269
Package Version setting 1008
Package view 1304, 1305, 1341
  deploying to FTP server 1305
  deploying to network location 1304
Package.DAT 705, 1113
package.ini 1180
PackageExpert
  about 1556
  ad-hoc test results 1584
  automatically resolving errors 1587
  automatically running PackageExpert tests during Repackager build process 584, 655
  changing the severity of a message 1577
  changing the severity of a specific error 1576

choosing .mst or editing .msi to resolve errors 1559
Configuration view 1606
configuring 1558
Connect Application Catalog dialog box 1608
creating a manual test 1590
Custom Test Results Editor dialog box 1611
customizing test results 1576
deleting a package's local test results 1575
deleting a package's published test results 1576
deleting a result filter 1580, 1584, 1607
deleting a test result message 1587
deleting test results 1575
editing a filter 1579, 1583, 1607
editing a test result message 1586
excluding test results 1581, 1582
Filter view 1607
filtering 1581, 1582
Home Page 1596
icons 1599, 1600
interface 1596
managing test results 1565
Manual Test Perform dialog box 1613
Manual Test Resolve dialog box 1614
message severity icons 1599
Options view 1607
output window 1602
package tree 1597
package tree icons 1600
prompting to publish test results 1573
publishing test results to the Application Catalog 1572
reference 1595
resolving a manual test 1591, 1592
resolving errors 1587
run status 1602
running a manual test 1591
sample manual test 1593
Select a Package dialog box 1615
setting default test configuration 1558
setting default tests 1558
setting defaults 1558
setting resolution options 1558
specifying the resolution method 1559
status bar 1601
suppressing the display of a message 1581, 1582
Test & Resolve view 1603
Test Result Filter Editor dialog box 1616
Test Results view in ConflictSolver 1504, 1563
testing external packages 1561
testing packages 1560
testing packages from a local file system 1561
testing packages in an Application Catalog 1562
testing packages in ConflictSolver 1563
toggling between local and published test results 1573
toolbar menus 1598
Unpublished Packages dialog box 1617
viewing Catalog or Local results 1573
viewing test results 1565, 1566, 1567
viewing test results in ConflictSolver 1574
packages 206, 207, 1263, 1304, 1308, 1385, 1729, 1802
adding from a Microsoft SCCM Server 752
adding from an AdminStudio Application Catalog 752
adding from local machine or network 756
adding to project 752, 756
copying to multiple Application Manager groups 206
customizing 1263
deleting from Application Catalog 207
distributing with Configuration Manager Web Console 1802
editing with Direct Editor 1308
launching for testing 769
managing 751
preparing for distribution 1729
preparing in Tuner for distribution 1304
reimporting into ConflictSolver 1385
rules for adding from a directory 758
selection rules when adding from a directory 758
selection rules when adding packages from a directory 826
testing prior to deployment 1858
version management 290
Packages property 762
Packages tab 751, 778
Connect to Machine 751
editing package properties 761
editing properties 761
how virtual or repackaged packages are listed 733
icons used on 787
properties 762, 780
selecting columns to display 803
Status column 780, 787
viewing package information 779
packaging 1346
Packaging Wizard 1346
Packaging Wizard 1304, 1305, 1346, 1347
deploying to FTP server 1305
deploying to network location 1304
Location panel 1346
Package Summary panel 1347
Setup.exe panel 1347
SMS panel 1347
parameters 1308
Setup.ini 1308

password 152, 1295
forgetting your Workflow Manager password 152
resetting your Workflow Manager password 152
setting NT service 1295
Password Required dialog 638
patch 479
applying to package during command-line import 479
patch file
associating with package 284
performing an ad-hoc import 284
patch files
performing ad-hoc import 214
patch impact analysis 1630
Patch Impact Analysis Wizard 1619, 1630, 1638
OS Snapshots panel 1640
Source Patches panel 1639
Summary Information panel 1640
Target Products panel 1640
Welcome panel 1638
Patch Impact Tab 352, 1494
Patch Impact view 1504
Patch View 1516
patches 215
dual-mode 1629
Patches Group View 407
Patches View 407, 409, 1513, 1516
performing patch impact analysis 1630
viewing information in ConflictSolver 1633
viewing patch impact analysis results 1637
Patches Group View 407
Patches Tab
ConflictSolver 1517
Patches tab 407, 1513, 1633
Patches Group View 407
Patches View 409, 1516
Patches View 409, 1516
Path property 750, 763, 782, 793
Path to Executable File setting 1027
-pc 562, 565
PDF (package definition file) 1827
using in Configuration Manager Web Console 1827
People Administration Page 157
performing OS Snapshots 340
Performing Snapshot panel 457
OS Snapshot Wizard 457
PermanentSystemFiles 674
PermanentSystemFilesSubfolders 674
permissions 101, 111, 174
assigning to a publication 325
copying roles 178
creating new roles 177
database server 464
deleting roles 178
Job Manager 2041
limiting tool accessiblity 101
minimum for AdminStudio 464
persisted conflicts 1379
deleting 1379
persistence 1379
person
disabling their user account 141
Person Details View 158
Place packages under the following folder 729, 766
Place virtualized packages under the following folder 837
Platform column 727, 746
Platform property 749, 793
policies
FlexWrap 1954
ports
port 443 (HTTPS) 68
port 80 (HTTP) 68
port 8443 68
used in activation 68
PostReplication.sql 318
postvalidation 1268, 1271
Postvalidation view 1271, 1340
postvalidating transforms 1271
-pp 562, 565
predefined folders
controlling display of in App-V Assistant 1059
controlling display of in Citrix Assistant 1210
controlling display of in ThinApp Assistant 1125
predeployment analysis 1858
Predeployment Test 1851, 1858, 1869, 1870, 1871, 1872, 1873, 1874, 1883, 1884, 1885, 1888, 1893, 1897
about 1852
assigning ASP.NET permissions 1897
assigning permissions to Application Catalogs 1897
assigning permissions to Microsoft Access databases 1897
changes made during test-ready package creation 1882
configuring Predeployment Test Reports Web site 1867, 1874
configuring Reports Web site 1858
ConflictSolver ACE reference 1901
Connection to New Catalog Page 1897
creating a test-ready package 1858
custom properties 1876
distributing test-ready package 1862
Job Details Report 1888
Job Summary Report 1885
Machine Result Summary Report 1893

performing analysis 1862
Predeployment Test Preparation Wizard 1869
Predeployment Test Reports Web site 1867
Predeployment Test Results By Job Report 1884
Predeployment Test Results By Machine Report 1893
reference 1867
reports 1883
sample XML files 1877
setting command line parameters 1860
Start Page 1868
Test Details Report 1889
test identifiers 1876
Test Result Summary Report 1886
testing 1862
viewing reports 1864
wizard 1869
Predeployment test
wizard 1869
Predeployment Test Preparation Wizard 1858, 1869
setting command line parameters 1860
Predeployment Test Reports Web site 1874
configuring 1858
Predeployment Test Results By Job Report (Predeployment Test) 1884
Predeployment Test Results By Machine Report (Predeployment Test) 1893
PreReplication.sql 318
prevalidation 1268, 1269, 1270
handling invalid packages 1270
Prevalidation view 1323
primary application directory 1060
explicitly set 1061
location of shortcut in ProgramFilesFolder 1061
location of shortcut not in ProgramFilesFolder 1061
ProgramFilesFolder 1061
value of INSTALLDIR variable 1061
Primary Application Directory dialog box 1105
privacy
Customer Experience Improvement Program (CEIP) 126
private key 1653
procedures and tasks 1273, 1277, 1278, 1279, 1280, 1281, 1282, 1283, 1284, 1285, 1286, 1287, 1288, 1289, 1290, 1291, 1292, 1293, 1294, 1295, 1296, 1297, 1298, 1300, 1301, 1302, 1303, 1320, 1322, 1323
Process Assistant 490
Process Template Editor 105, 106
adding tools from 106
reordering tasks in 105
product code conflicts 1370
Product Info and Update URL 1338
Product Information view 1713
Product Name Change dialog 310
product properties 1274, 1275
default destination 1274
default organization 1274
destination variable 1275
Product Properties view 1274, 1275, 1324
changing destination variable 1275
setting the default destination 1274
setting the default organization 1274
Product property 763, 781
Product View 306, 344, 1488
Application Manager 344
ConflictSolver 1488
icons 344, 1488
Product view 378, 1497
products
moving in Application Manager 207
organizing into groups in Application Manager 205
Products view 413
Profile Files page 1234
creating a new folder 1209
deleting files and folders from a profile 1210
Profile Information page 1229
profile manifest file 710
Profile Registry page 1239
Profile Requirements page 1231
Profile Shortcuts page 1237
creating a new profile shortcut 1217
excluding or deleting a profile shortcut 1218
including an existing shortcut 1218
profiles 705, 709, 1194
prog IDs 1679
checking 1679
Prog IDs view 1715
programs (in Configuration Manager Web Console) 1812
Progress panel 456
Merge Wizard 456
project files 734
components of 736
creating a new 719
creating new 719
major elements 736
major elements used in 735
opening 719
opening existing 720
setting default properties 774
Project Information panel 456
OS Snapshot Wizard 456
Project Options dialog box 774, 857
project path 511, 521, 529, 537, 543
specifying in Repackaging Wizard 511, 521, 529, 537,

543
Project Properties dialog box 639, 640
Exclusions tab 640
General tab 639
Project Wizard 801
ProjectName 124
projects 100, 101, 106, 107, 108, 111, 116, 135
creating 106
creating new 135
deleting 108
executing 107
filtering 107
integration with Workflow Manager workflows 116
New Workflow Project Wizard 135
running associated tools in 100, 108
saving 108
setting global default virtual conversion settings 868
Source Package panel 135
using 106
Projects tab 116
ProjectTemplate 674
properties 134, 592, 1275, 1298, 1301, 1302, 1303
adding custom setup 1302
changing Add/Remove Programs 1301
configuring tools in AdminStudio 134
editing dialog 1298
feature 1275
modifying setup 1303
removing custom setup 1303
viewing Repackager project 592
Properties Dialog 1644
Properties dialog box 1316
Microsoft patches 1644, 1645, 1646
Properties Tab
Tool Properties dialog 130
Properties tab 130
Properties window 748, 762
protocol
of App-V deployment server 1086
Protocol setting 1009
pseudo-tokens 1549
Publication
subscribing 326
subscribing to more than one 318
publication 319
adding users 325
assigning permissions to 325
creating 321
deleting 326
editing 324
editing an access list 325
managing 319
publishing 323
setting a schedule 324
Publication Manager Dialog 428
Application Catalog Replication 340
using to manage publications 320
using to manage subscriptions 327
Publication Processing dialog 429
Publication Properties dialog 429
Publication Wizard 458, 459, 460, 461
Publication Access List Panel 460
Publication Details Panel 459
Task Scheduling dialog box 432
Publisher URL 1338
publishing 316
Application Catalog 316, 319
database 319
Purpose column 727, 746
Purpose property 749, 794
-pv 562, 565
PVKFile 1664

Q

-Q 466, 481
QualityMonitor 1667, 1669, 1672, 1676, 1677, 1678, 1679, 1680, 1681, 1682, 1685, 1697, 1698, 1699, 1701, 1711
checking class IDs 1677
checking deployment status 1689
checking file associations 1678
checking help files 1678
checking manifest files 1681
checking ODBC data sources 1682
checking ODBC drivers 1682
checking prog IDs 1679
checking services 1679
checking shortcuts 1680
checking type libraries 1680
Component Properties dialog 1702
creating exclusion lists 1683
deployment testing 1676
dialogs 1701
excluding system files 1683
exclusion lists 1683
Feature Properties dialog 1703
Files dialog 1705
Install or Configure Feature Dialog 1704
Install or Configure Product Dialog 1704
install or configure products or features 1693
lockdown and runtime testing 1685
menus and toolbar 1699

MSI Doctor 1689, 1692, 1693, 1694, 1695
Product Properties dialog 1709
reinstall components 1695
Re-install Feature Dialog 1710
reinstall features 1694
Re-install Product Dialog 1710
running from the command line 1698
Test Cases 1669
test reports 1697
verifying files 1692
viewing deployment status properties 1692
viewing Test Item details 1672
views 1711
QualityMonitor projects 1668, 1669
creating new 1668
opening existing 1669
Quick Start evaluation experience 193

R

reboot handling 565
Reboot Message setting 1029
reference 113, 135, 456, 775
AdminStudio Interface 113
New Workflow Project Wizard 135
OS Snapshot Wizard 456
REG files 1281
importing 1281
Registry 786, 860
registry 617, 618, 671, 1345
exclusions 617, 671
Import REG File Wizard 1345
modifying in App-V Assistant 1070, 1071
modifying in ThinApp Assistant 1135
removing exclusions 618
registry conflicts 1370
registry entries 1279
Registry Entries view 646
registry exclusions 617, 618
editing existing 617
removing global 618
registry information 1282
removing 1282
Registry Isolation Options dialog box 1107, 1168, 1248
registry keys 605, 1280
and App-V application 1071
and Citrix profile 1221
and ThinApp application 1136
creating 1280
excluding 605
Registry page 1157
Registry Report 1390, 1392
Registry tab 671
Exclusions Editor 671
registry values 606, 1281
creating 1281
excluding 606
Registry view 383, 401, 405, 1280, 1281, 1282, 1327
creating a registry key in 1280
creating a registry value in 1281
importing REG files in 1281
removing registry information from 1282
reimporting packages 1385
Remote Link Application Catalog panel 452
Removable Disk Changes go to Sandbox 1153
Remove 795
Remove button 1302
disabling in Add/Remove Programs 1302
Remove Selected 786
RemoveFileData ACE 1437
RemoveIniFileData ACE 1437
RemoveRegistryData ACE 1438
reordering 105
tasks 105
Repack.ini file 684
Repack.log 681
Repackage 798
Repackaged Output view 578, 652, 657
building InstallShield Editor projects 578
building MSI packages 578
creating virtual applications 692
Repackager 489, 496, 501, 562, 569, 570, 571, 572, 573, 577, 587, 592, 593, 594, 595, 596, 604, 605, 606, 607, 608, 609, 610, 628, 630, 632, 633, 637, 638, 639, 641, 642, 645, 646, 648, 649, 650, 651, 652, 657, 658, 660, 665, 667, 674, 685, 686
About Repackager dialog 633
additional setup programs 509
Advanced Settings view 658
and anti-virus software 498
automatically building a Citrix profile 589
automatically building a ThinApp application 589
automatically creating a Windows Installer package after creating the Editor project 589
automatically running PackageExpert tests upon build 584, 655
building a Citrix profile 578, 656, 707
building a ThinApp applications 704
building a VMware ThinApp application 656
building an App-V package 655
building Citrix profile automatically at project build 589
building InstallShield Editor projects 578

building MSI packages 578
building ThinApp application automatically at project build 589
Captured Installation view 642
changing data type appearance 570
clean systems 496
command-line options 562
component settings options 588, 659
configuring advanced conversion options 587
configuring exclusions 604
conversion problems 685, 686
converting .axt 573
converting .inc 573
converting .ipf 573
converting .isl 577
converting .txt 577
converting .wse 577
converting InstallShield log files 577
converting legacy setups 572
converting Novell ZENworks projects 573
converting Repackager 3.x output 573
converting SMS projects 573
converting WinINSTALL projects 577
converting Wise Installation projects 577
Create Report dialog box 633
creating projects 571
creating reports 594
Deleted Files view 650
Deleted Registry Entries view 651
detecting dependencies 593
dialog boxes 632
editing generated projects in InstallShield Editor 596
Error Building Table File error 686
excluding all files in a directory 605
excluding all shortcuts in a directory 607
excluding directories and subdirectories 605
excluding files 604
excluding INI file sections 606
excluding INI files 606
excluding registry keys 605
excluding registry values 606
excluding shortcuts 607
excluding shortcuts from subdirectories 607
exclusion list 681
exclusions 610
file exclusions 604
Files and Folders view 645
files associated with 681
INI Files view 649
installing on a clean machine 503
installing on clean machine 503
InstallShield SmartScan options 588, 660
ISDEV fatal error -5023 686
isolating applications 1649
Isolation Options dialog box 634
launching 569
launching Automated Application Converter 717
launching remotely 501
location of isrepackager.ini 92
menus 630
modifying external configuration file 609
Options dialog box 637
Options.ini file 674
OtherComponentFileExtensions 674
package conversion options 587, 658
Package Information view 657
Password Required dialog box 638
Project Properties dialog box 639
Registry Entries view 646
remote repackaging on Windows NT4 501
Repackaged Output view 652, 692
repackaging legacy setups 504
running automated tests against the .msi package at build 584, 655
saving projects 595
scanning InstallShield Professional setups 619
setting digital signature options 634
setting manifest options 634
Setup Intent Wizard 665
Shortcuts view 648
SmartScan Wizard 660, 663
SMS conversion problems 686
specifying external configuration file 608
support of user-defined extensions in options.ini 674
ThinApp applications 704
toolbar 630
troubleshooting 685, 686
viewing properties 592
views 642
virtualization 655, 656
VMware Repackaging Wizard 667
WinINSTALL Conversion dialog box 641
WinINSTALL conversion problems 685
Repackager 3.x output 573
converting to Repackager project 573
Repackager Cache Path property 750, 794
Repackager error building table file 686
Repackager projects 571, 595
creating 571
saving 595
Repackager Start Page 628
repackaging 489, 496, 501, 504, 609, 765

a Windows Installer package 540
alternate-language 496
an installation from a self-extracting .exe 507
and anti-virus software 618
excluding processes from 560
exclusions 609
legacy setups 489, 504
remotely on Windows NT4 501
support for 64-bit applications 492
viewing conversion results 768
Windows Installer packages 764
Repackaging an InstallScript MSI Setup to a Basic MSI Setup 538
Repackaging Method property 763, 782
repackaging methods 497, 498
selecting 515
Repackaging panel 556
Repackaging Using the Installation Monitoring Method 505
Repackaging Using the InstallScript Professional Logging Method 534
Repackaging Using the Snapshot Method 515
Repackaging Wizard 501, 539, 546, 547, 549, 550, 552, 554, 555, 556, 558, 559, 560, 561, 565
additional panels 559
additional setup programs 509
Additional Setup Programs dialog box 559
Analysis Options dialog box 561
Collect Product Information panel 550
deleting added setups 510
Excluded Processes dialog box 560
InstallScript MSI Conversion Output panel 555
InstallScript MSI Identified panel 552
InstallShield Professional Setup panel 554
Method Selection panel 547
Original InstallShield Professional Setup panel 662
reboot handling 565
repackaging a Windows Installer package 540
Repackaging panel 556
running from command line 539
running remotely on Windows NT4 501
Set Target Project Information and Capture Settings panel 554
Setup Information dialog box 560
Snapshot Method panel 549
specifying project path 511, 521, 529, 537, 543
Summary panel 558
using 505
Welcome panel 546
Repackaging Wizard Command-Line Options 562
Repair button 1302
disabling in Add/Remove Programs 1302
Replication
configuring servers for 312
replication 311, 332
Application Catalog 311, 312, 316
configuring servers for Replication 312
deleting a subscription 331
enabling/disablying a subscription 329
lifecycle 316
managing publications 320
publishing a publication 323
publishing Publication manually 318
setting a publication schedule 324
subscribing to more than one Publication 318
updating a subscription manually 318, 330
Replication Command Line Options dialog 466
report 804
viewing 768
Report Center 1995, 2020
available reports 1999
evaluating 68
Report View 2021
reports 594, 1390, 1393, 1883, 1884, 1885, 1888, 1893, 2020
All Reports page 2020
assigning Role permission to view 2034
ConflictSolver 1390, 1393
creating 2020
creating custom 1393
creating in Repackager 594
Crystal Reports 1390, 1393
editing Role permissions to view 2034
File and Registry 1392
Files 1390
Job Details Report (Predeployment Test) 1888
Job Summary Report (Predeployment Test) 1885
Machine Result Summary Report (Predeployment Test) 1893
Package 1390, 1391
Predeployment Test Results By Job Report (Predeployment Test) 1884
Predeployment Test Results By Machine Report (Predeployment Test) 1893
Predeployment Test) 1883
Registry 1390
Report Center 1995, 2020
Report View 2021
Report Wizard 2033
Test Details Report (Predeployment Test) 1889
Test Result Summary Report (Predeployment Test) 1886
viewing Predeployment Test result reports 1864
Reports Wizard 2033

requirements
how they are applied for a Citrix profile 1202
setting language requirements in Citrix profile 1202
Reset Period setting 1029
Reset Sandbox on Exit 1153
Resolution Details dialog 1539
Resolution Operations dialog 1523, 1539
resolution options 1373, 1535
changing ConflictSolver 1373
ConflictSolver 1535
Resolution Options dialog 1540
Resolution tab 1535
in ConflictSolver Options dialog 1535
resolving
a manual test in PackageExpert 1592
errors in PackageExpert 1587
resolving conflicts 1379
response transforms 1268
restricted environments 1687
running lockdown and runtime tests 1687
result filters
deleting in PackageExpert 1584, 1607
results
viewing 768
Results panel 666
Setup Intent Wizard 666
Results tab 731, 768, 796
context menu commands 799
icons used on 732, 798
properties 797
report 804
Role 177, 179
assigning permission to view reports 2034
creating 177
Role Administration Page 179
Role Details View 181
roles
copying 178
creating 177
deleting 178
system roles 174
user roles 174
root folder name 1084
Root Folder Name setting 1008
RTSP 784, 1009, 1053, 1086
RTSPS 784, 1009, 1053, 1086
Rules Viewer dialog 1540
Rules Wizard 1545, 1546, 1548, 1549, 1550
Additional Information panel 1546, 1548
Custom ACEs panel 1467, 1468
DLL-Based ACEs panel 1550
General Information panel 1546
inserting Column Names in Error or Display Strings 1549
inserting Product Name in Error or Display strings 1549
pseudo-tokens 1549
Summary panel 1550
Token Grammar 1549
User-Defined ACEs 1467, 1468
Welcome panel 1545
Where Clause Panel 1549
running associated tools 100, 108
runtime drive 860
runtime testing 1685

S

-S 466
sample.mdb 199
sandbox cache 1116, 1117
Sandbox Name 1152
saving 108
projects 108
workflows 108
-sb 562, 565
Scan for Dependencies 1505
ConflictSolver 1505
scanning for dependencies 1360
option disabled 1360
packages in software repository 1360
Scanning Media panel 664
SmartScan Wizard 664
Scanning panel 663
Scanning project panel 666
Setup Intent Wizard 666
Schedule panel 454
Package Auto Import Wizard 454
Script Execution dialog box 1243
script-based setup.exe 1757
creating with Distribution Wizard 1757
Search 802
Search Results Tab 352, 1494
Second Action Delay setting 1029
Second Error setting 1029
Security Console
evaluating 68
Select a Package dialog box 1615
Select Destination Folder panel 851
Select Group panel 852
Select Output Formats panel 729, 766, 837
Select Package Installation File dialog box 758, 861
Select Package Source panel 722, 752, 757, 818
Select Package Types to Publish panel 772, 853

Select Packages Installation File dialog box 861
Select Packages panel 723, 754, 757, 822
Select Publish Target panel 771, 847
Select Transform dialog box 862
Select Virtual Machine dialog box 769, 863
Select Virtual Machine Image File dialog box 864
Select Virtual Machine Source panel 726, 744, 829
Select Virtual Machines from a Microsoft Hyper-V Server panel 830
Select Virtual Machines from a VMware ESX or ESXi Server panel 831
Select Virtual Machines panel 727, 745, 832
Selected Package List panel 724, 754, 758, 824
  icons used on 724
Sequencer 1047
Sequencer. See also Microsoft App-V Sequencer.
sequences 1297
  restoring dialog 1297
Server Address property 750, 794
server component
  FlexWrap 1937
Server Host 859
server locations 1300, 1301
  adding 1300
  modifying 1300
  removing 1300
  reordering 1301
Server Locations view 1300, 1301, 1337
  adding server locations in 1300
  modifying server locations in 1300
  removing server locations in 1300
  reordering server locations in 1301
Server Name 819, 821, 830, 848, 850
Server Password property 750, 795
Server Port 859
Server Protocol 859
Server Username property 750, 794
Service Dependencies setting 1028
Service Is Interactive setting 1027
Service Packs Requirement dialog box 1250
service type 1295
  setting 1295
Service Type setting 1027
ServiceControlEvents 674
services 1679
  and Citrix profiles 951
  and Citrix XenApp 952
  checking 1679
Services view 1717
Set Target Project Information and Capture Settings panel 554
  Repackaging Wizard 554
Setup Cache Path property 750, 794
Setup Feature Tree panel 663
Setup Information dialog box 560
  Repackaging Wizard 560
Setup Intent Wizard 593, 665, 666
  Results panel 666
  Scanning project panel 666
  Welcome panel 665
Setup Organization 1272
setup programs
  adding additional in Repackaging Wizard 509
setup properties 1302, 1303
  adding custom 1302
  customizing 1302
  modifying 1303
  removing custom 1303
Setup Properties view 1302, 1303, 1338
  adding and editing comments 1303
  adding custom setup properties 1302
  modifying properties 1303
  removing custom setup properties from 1303
setup property comments 1303
  adding and editing 1303
Setup.exe 1307
  creating for package and transform 1307
Setup.exe panel 1347
  Packaging Wizard 1347
Setup.ini 1308
setups 489, 1297
  deleting added in Repackaging Wizard 510
  disabling custom 1297
  repackaging legacy 489, 504
severity
  changing for a PackageExpert message 1576, 1577
SFT file 701
SFT package
  specifying root folder name 1051
SFT_SOFTGRIDSERVER 1053
shared assemblies 1657
  servicing published 1657
shared location 92
SharedCommonFiles 674
SharedPoint 124
Shell New Enabled setting 1021
shortcut 482
  creating to a specific Application Catalog 482
shortcut conflicts 1370
shortcuts 607, 1282, 1283, 1284, 1285, 1329, 1331, 1332, 1680
  changing icon for 1283

changing location for 1284
changing target for 1284
checking 1680
creating 1283
creating a hot key for 1284
determining path of changed in Tuner 1285
excluding 607
excluding all in a directory 607
excluding from subdirectories 607
excluding or deleting from App-V application 1067
excluding or deleting from Citrix profile 1218
excluding or deleting from ThinApp application 1133
including in App-V application 1067
including in Citrix profile 1218
including in ThinApp application 1132
location 1332
properties 1329
removing 1285
target 1331
when to exclude or delete from App-V application 1069
when to exclude or delete from Citrix profile 1219
when to exclude or delete from ThinApp application 1134
Shortcuts view 384, 406, 648, 1283, 1284, 1285, 1328, 1715
changing a shortcut's icon 1283
changing a shortcut's location 1284
changing a shortcut's target in 1284
creating a hot key for a shortcut 1284
creating shortcuts in 1283
OS Snapshot 402
removing shortcuts from 1285
silent activation 68
silent mode 481
SISAuthor=Repackager 674
SmartScan Wizard 619, 660, 661, 664
automatic launch 660
conditions for automatic launch 660
launching automatically 660
option to match legacy system's OS requirements 658
Repackager 663
Scanning Media panel 664
Welcome panel 661
SMS 1306, 1740
creating file 1306
deploying 1307
using Distribution Wizard to prepare packages for 1740
SMS conversion problems 686
Repackager 686
SMS distribution 1741
SMS Distribution Panel 1771
Distribution Wizard 1771
SMS panel 1347
Packaging Wizard 1347
SMS project 573
converting to Repackager project 573
SMS templates 1741
-sn 562, 565
snapshot 497, 498
Snapshot method
and anti-virus software 498
Snapshot Method panel 549
Repackaging Wizard 549
Snapshot Name property 750, 795
snapshots 741
Soft Timeout property 764, 783
SoftGridUserSettings folder 971
software publishing credentials 1653
Software Repository 284
and ConflictSolver 293
and Distribution Wizard 293
and InstallShield Editor 293
and Predeployment Test Preparation Wizard 293
conflict resolution and 2040
enabling in Application Catalogs 285
Get Latest Version 292
getting a copy of a package 292
importing packages into 287
introduction 284
package check in and check out 291
using 284
software repository
scanning for dependencies for files stored in Software Repository 1360
sorting lists 808
Source Application Catalog panel 455
Merge Wizard 455
Source Package panel 135, 1542
Conflict Wizard 1542
in New Workflow Project Wizard 135
Source Type panel 1542
Conflict Wizard 1542
SourcePackage 124
sp_addpublication 318
SPCFile 1664
SPD File Details 1774
Distribution Wizard 1774
SPD Parameters panel 1773
Distribution Wizard 1773
Specifying duplicate package identifiers in Application Manager 229
SQL Express 193
SQL Server
merge replication 332

preferred authentication mode 303
SQL Server Enterprise Manager 318
using to create Publications and Subscriptions 318
standalone Application Catalog 192
Standard.nir 681
start name 1295
setting NT service 1295
Start Page 114
start type 1295
setting NT service 1295
Startup Type setting 1027
status 1822
viewing Job Manager Engine status 2062
viewing status of distributed package in Configuration Manager Web Console 1822
subscribing 326
subscribing to 311
Application Catalog 311, 312, 316, 326
Publication 326
Subscription
disabling 318
enabling 318
subscription 302
creating 328
deleting 331
enabling or disabling 329
in Application Manager 302
manually updating 330
Subscription Manager 326
Subscription Manager dialog 430
Subscription Wizard 461, 462, 463
Subscriptions dialog 424
Subsequent Action Delay setting 1030
Success Result setting 1025
suiting
in App-V Assistant 1072
Summary panel 450, 454, 558, 1545, 1550, 1551, 1658
Application Isolation Wizard 1658
Conflict Wizard 1545
Import Wizard 450
Package Auto Import Wizard 454
Repackaging Wizard 558
Rules Wizard 1550
Validation Wizard 1551
suppression issues 1299
SuppressReboot 1308
SuppressWin2k 1308
SysGuard File setting 1011
System Configuration view 1326
Tuner 1326
system roles 174
system tray mode 481

T

table 1309
adding a new record using Direct Editor 1309
adding a new row using Direct Editor 1309
Tables view 389, 402, 406, 1507
taking OS snapshots 338
target collection 1799
target directory 136
setting in New Workflow Project Wizard 136
Target Directory and File Name panel 136
in New Workflow Project Wizard 136
Target Information panel 1545
Conflict Wizard 1545
Target setting 1019
TargetDir 124, 136
TargetFileName 124, 136
task page help 94
setting location in AdminStudio 94
task properties 104
modifying 104
tasks 100, 104, 105, 106, 118
associating help files 105
associating tools with in AdminStudio 100
creating new 104
creating notes for 104
deleting 106
in workflows 118
renaming 105
reordering in Process Template Editor 105
template.pdf 1741
template.sms 1741
Terminate Children setting 1019
Test & Resolve view 1603
Test Cases 1669, 1672, 1673, 1674, 1675, 1695
adding comments 1672
clearing results 1673
creating custom 1695
filtering data 1675
manually setting status 1674
Test Cycle Summary view 1713
Test Details Report (Predeployment Test) 1889
Test Item details 1672
viewing 1672
Test Item Information dialog 1711
Test Items 1670, 1671, 1675
adding comments 1671
manually setting status 1675
running individual 1670

running multiple 1670
Test Progress dialog 1711
test reports 1697
Test Result dialog 1711
Test Result Filter Editor dialog box 1616
test result message
adding 1586
deleting in PackageExpert 1587
editing in PackageExpert 1586
Test Result Summary Report (Predeployment Test) 1886
test results
adding ad-hoc test results 1584
customizing in PackageExpert 1576
deleting 1575
deleting a test result message 1587
deleting catalog 1576
deleting local 1575
editing in PackageExpert 1586
viewing and managing 1565
viewing Catalog results 1573
viewing in ConflictSolver Test Results view 1574
viewing in PackageExpert 1566, 1567
viewing Local results 1573
Test Results view 1504, 1563
Test Virtualization Readiness 801
testing 1676, 1685
deployment 1676
in PackageExpert 1560
lockdown and runtime 1685
Predeployment Test 1851, 1852
repackaged MSI packages 769
source packages 769
virtual packages 769
testing packages 769
test-ready package 1858
installing 1862
ThinApp 563, 589, 704, 707
AppLink 1170
command line 563
Interm directory 706
Virtual Operating System 705
ThinApp applications 589, 704
about 704, 1109, 1113
adding an AppLink reference 1142
adding diagnostic tools to 1119, 1165
adding existing folder 1122
adding files to 1121
adding or deleting registry keys and values 1136
and Active Directory 1118
Application Link 1141
AppLink 1141, 1170
AppSync 1144
automatically creating from Repackager 692
benefits of deploying 707, 1110
building 1147
building a Windows Installer package with build output 1140
building using command line 1180
components of 705, 1113
compressing 1131
configuration file (package.ini) 1180
controlling access to 1118
conversion error and warning messages 1180
creating using InstallShield ThinApp Assistant 692
creating with InstallShield 1108
defining shortcut executables 1131
excluding vs. deleting a shortcut 1134
files included in 705, 1113
how transforms are included 1115
including an existing shortcut 1132
inheritance of isolation options from folders to files 1129
inheritance of isolation options in the registry 1137
Interm directory 706, 1115
intermediate data files 706, 1115
launching 1113
linking ThinApp applications 1141, 1170
linking to an application with more than one shortcut 1174
linking to an application with one shortcut 1174
linking to another ThinApp application 1174
location of 1114
managing files and folders 1121
methods to convert Windows Installer packages 691
modifying registry 1135
moving files and folders 1124
operating system 1109
order of import of linked ThinApp applications 1172
overview of isolation options 1126
overview of ThinApp Assistant 1109
Package.DAT 705
Package.DAT file 1113
prerequisites for building 707, 1119
relative paths when defining AppLink references 1143
renaming a shortcut 1135
sandbox cache 1117
sandbox name 1117
sandboxes 1116
security and authorization for linked applications 1173
selecting application shortcuts 1130
Setting AppLink options 1141
setting AppSync Expiration settings 1145
setting folder isolation options 1166

setting isolation options 1126, 1128
setting isolation options for folders and files 1129
setting Log Monitor tracing options 1140
setting registry isolation options 1136
shortcut requirements 1132, 1157
shortcuts 705
steps to create 1116
steps to create with ThinApp Assistant 1110
ThinApp Assistant 1108
ThinAppPackage directory 1114
updating applications 1144
updating using AppSync 1176
virtual environment 1109
virtual operating system 705
Virtual Operating System (VOS) 1109
ThinApp Assistant 590, 692, 1108, 1180
about 1110
about sandboxes 1116
adding an existing folder to a ThinApp application 1122
adding diagnostic tools to a ThinApp application 1119, 1165
adding files to a ThinApp application 1121
adding or deleting registry keys and values from a ThinApp application 1136
application features requiring pre- or post-conversion actions 1180
Applications page 1156
Build Options page 1159
building a ThinApp application 1147
building a Windows Installer package with build output 1140
building ThinApp application in Direct Edit mode 1139
clearing the cache 1164
Compression Type option 1131
compression types 1131
controlling access to ThinApp applications 1118
controlling the display of predefined folders 1125
creating a ThinApp application 1116
creating new application shortcut executables 1132
creating new folder 1123
default isolation options 1128
deleting files and folders 1124
Diagnostic Tools dialog box 1119, 1165
disabling Log Monitor Tracing 1141
enabling ThinApp application building when editing a Windows Installer package 1139
error messages 1180
excluding or deleting a ThinApp application shortcut 1133
Files & Folders page 1153
Home page 1149
how transforms are included 1115
including an existing shortcut 1132
inheritance of isolation options 1129
inheritance of isolation options in registry 1137
integration with Project Assistant and Installation Designer 1034
Interm directory 1115
isolation options 1128
managing files and folders 1120, 1121
modifying build options 1137
modifying registry settings 1135
modifying shortcuts 1130
moving files and folders 1124
overview 1109
overview of 1109
overview of isolation options 1126
Package Information page 1151
package.ini 1180
prerequisites for creating ThinApp application 1119
reference 1149
Registry Isolation Options dialog box 1168
Registry page 1157
renaming a shortcut 1135
sandbox cache 1117
selecting releases to build 1138
setting isolation options 1126, 1128
setting isolation options for folders and files 1129
setting isolation options on folders 1166
setting Log Monitor tracing options 1140
setting registry isolation options 1136
specifying access via Active Directory 1118
specifying general settings 1117
steps to create a ThinApp application 1110
supported InstallShield project types 1115
when to exclude and when to delete shortcuts 1134
Write Copy, Merged, and Full isolation options 1128
ThinApp cache
clearing 1164
ThinApp Log Monitor 1140
ThinApp Virtual Operating System (VOS) 1109
ThinApp Virtualization Suite Not Found 1180
ThinApp VOS 705
ThinAppPackage directory 1114
Timeout setting 1025
TimeStampAssemblies 1664
Tivoli 1742
using Distribution Wizard to prepare packages for 1742
Tivoli Integration panel 1772
Distribution Wizard 1772
Tivoli Settings panel 1772
Distribution Wizard 1772

Token Grammar 1549
Rules Wizard 1549
tool accessibility 101
Tool Properites panel 134
tool properties 130
viewing 130
Tool Properties dialog 130, 132
Configuration tab 132
Properties tab 130
Tool Properties panel 134
Add Tool Wizard 134
toolbar 119, 341, 630, 1311, 1485, 1699
AdminStudio 119
Application Manager 341
ConflictSolver 1485
QualityMonitor 1699
Repackager 630
toolbars 800
Toolbars tab 1316, 1526
tools 97, 98, 99, 100, 106, 108, 115, 130, 133
adding 97
adding command-line configurations 98
adding from Process Template Editor 106
adding in AdminStudio 133
adding using the Add Tool Wizard 133
associating with tasks in AdminStudio 100
deleting command-line configurations from in AdminStudio 99
editing properties for existing 98
modifying command-line configurations 99
running associated in projects 100, 108
viewing properties 130
Tools Gallery 101
deleting tools from 101
on Tools tab 115
Tools menu 801
Tools tab 115
Total File Size setting 1008
transform 479, 1271, 1307
applying to package during command-line import 479
creating Setup.exe for 1307
postvalidating 1271
transform files
associating with application catalog 209
associating with package using ad-hoc import 284
performing ad-hoc import 209, 214, 284
transform properties 1268
viewing 1268
Transform property 763, 782
Transform Summary dialog 1316
transforms 1264, 1265, 1266, 1267, 1268, 1316, 1384
creating generic 1267, 1316
creating new 1265
creating universal 1267, 1316
generic 1267, 1316
importing 215
opening existing 1266
opening recently accessed 1267
resolving conflicts using 1384
response 1268
Tuner 1264
universal 1267, 1316
working with 1264
trial mode 737
troubleshooting 869
first things to check 869
how to test a virtual machine 880
problems and solutions 872
Tuner 1263, 1264, 1268, 1279, 1296, 1304, 1308, 1309, 1342
adding and editing setup property comments 1303
adding custom setup properties 1302
adding files 1277
adding INI files 1286
adding new data sources 1290
adding new INI file keys 1287
adding new ODBC data source attributes 1290
adding new ODBC driver attributes 1290
adding sections to INI files 1287
adding server locations 1300
CAB files 1279
changing a feature's visibility 1273
changing a shortcut's icon 1283
changing a shortcut's location 1284
changing a shortcut's target 1284
changing Add/Remove Programs Properties 1301
creating a hot key 1284
creating a new transform file 1320
creating a registry key 1280
creating a registry value 1281
creating shortcuts 1283
Direct Editor 1309, 1342
disabling custom setups 1297
displaying files from the base Windows Installer package 1278
editing dialog properties 1298
editing ODBC data source attributes 1291
editing ODBC driver attributes 1291
hiding dialogs during UI sequences 1296
importing existing INI files 1286
importing REG files 1281
modifying server locations 1300

modifying setup properties 1303
opening a recent transform file 1322
opening an existing transform file 1323
preparing packages for distribution 1304
preventing installation of files from the Windows Installer package 1278
removing added files 1279
removing custom setup properties 1303
removing existing ODBC data sources 1291
removing INI file keys 1289
removing INI files 1288
removing ODBC data source attributes 1292
removing registry information 1282
removing sections from INI files 1289
removing server locations 1300
removing shortcuts 1285
reordering server locations 1301
restoring dialog sequences 1297
setting initial state of a feature 1273
setting the NT service arguments 1292
setting the NT service dependencies 1293
setting the NT service description 1293
setting the NT service display name 1293
setting the NT service error control level 1294
setting the NT service load order group 1294
setting the NT service overall install result 1294
setting the NT service start name and password 1295
setting the NT service start type 1295
setting the NT service type 1295
Setup.ini 1308
suppressing License Agreement dialog 1297
transforms 1264
validation in 1268
working with dialogs 1296
Tuner reference 1310, 1311, 1314, 1315, 1316, 1319, 1320, 1322, 1323
checklist 1314
customization steps 1314
Customize dialog box 1316
Help view 1323
InstallShield Start page 1320
menus 1311
Options dialog box 1316
Output Window 1315
Package Validation view 1323
Properties dialog box 1316
toolbar 1311
user interface reference 1310
View Bar 1314
views 1319
Tuner views 1319, 1320, 1323, 1324, 1325, 1326, 1327, 1328, 1333, 1334, 1335, 1337, 1338, 1339, 1340, 1341, 1342
Add/Remove Programs 1338
Additional Tools 1342
Application Configuration 1337
Dialogs 1338
Direct Editor 1342
Features 1325
Files and Folders 1327
Help 1323
INI Files 1333
Location 1342
NT Services 1335
ODBC Resources 1334
Package 1341
Package Preparation 1339
Package Validation 1323
Postvalidation 1340
Prevalidation 1323
Product Properties 1324
Registry 1327
Server Locations 1337
Setup Properties 1338
Setup.exe 1342
Shortcuts 1328
SMS 1342
System Configuration 1326
type libraries 1680
checking 1680
checking with QualityMonitor 1680
Type Libraries view 1715
type library file path 1474
specifying in Visual Studio C++ 1474

U

-U 466
UI sequences 1296
hiding dialogs during 1296
universal transforms 1267, 1316
creating 1267, 1316
Unpublished Packages dialog box 1617
update 330
manually updating a subscription 330
Updated.isr 681
upgrading
5.0 or 5.5 or 6.0 Application Catalogs 204
pre AdminStudio 5.0 Application Catalogs 204, 464
upgrading AdminStudio databases 204
Use Application Expiration 1178
UseHKCUProxy 674

UseMergeModules 674
user accounts 159
  disabling a user account 141
  importing from Active Directory 159
User Credentials panel 728, 747, 834
user interface 777
user interface reference 1310
user roles 174
user-defined ACE rules 1467, 1468
user-defined ACEs 1467, 1477
  deleting 1477
  editing 1477
User-Defined Tests view 1724
  Test Case view 1724
users
  adding to a publication access list 325
  deleting 142
  People Administration Page 157
  Person Details View 158
UseSrcFolder 674

V

validate options 1536
  ConflictSolver 1536
Validate Package 1364
  deleting ICE errors 1367
  suppressing ICE errors 1366
  viewing ICE error information 1365
Validate Tab 352, 1494
validation 1268, 1361, 1362, 1363, 1364, 1368, 1369, 1503
  before import 1362
  changing default ConflictSolver options 1368
  changing the default file in ConflictSolver 1368
  ConflictSolver 1361, 1362
  deleting ICE errors 1367
  during ConflictSolver import 1363
  Explanation dialog 1526
  handling invalid packages in ConflictSolver 1369
  in ConflictSolver 1362
  suppressing ICE errors 1366
  viewing ICE error information 1365
Validation tab 1309
  in ConflictSolver Options dialog 1536
  launching Direct Editor 1309
Validation view 1503
Validation Wizard 1362, 1550, 1551
  MSI Source Information panel 1551
  Summary panel 1551
  using 1362
variables 124
  DevLocation 124
  InstallLocation 124
  ProjectName 124
  SharedPoint 124
  SourcePackage 124
  TargetDir 124
  TargetFileName 124
verifying component deployment 1689
verifying files 1692
  using MSI Doctor in QualityMonitor 1692
-version 565
version 562
version management 284
  in Software Repository 290
  using the Software Repository 284
Version property 763, 781
Versioning 860
VFS folder 971
View Bar 1314
View menu 800
View Report 801
View Settings tab 1316
viewing tool properties 130
Views 642
views 1319, 1320, 1323, 1324, 1325, 1327, 1328, 1333, 1334, 1335, 1337, 1338, 1339, 1340, 1341, 1342, 1497, 1499, 1503, 1504, 1505, 1506, 1507, 1509
  Additional Tools 1342
  Associated Patches 1506
  Conflicts 1499, 1509
  ConflictSolver 1497
  Dependencies 1505
  Dialogs 1338
  Direct Editor 1342
  Features 1325
  Files and Folders 1327
  Help 1323
  INI Files 1333
  Location 1342
  Merge Module 1507
  NT Services 1335
  ODBC Resources 1334
  OS Snapshot 1507
  Other Setup Types 1507
  Package 1341
  Package Preparation 1339
  Package Validation 1323
  Patch Impact 1504
  Postvalidation 1340
  Prevalidation 1323
  Product 1497

Product Properties 1324
Registry 1327
Server Locations 1337
Setup Properties 1338
Setup.exe 1342
Shortcuts 1328
SMS 1342
Tables 1507
Validation 1503
virtual file system (VFS) folder 971
virtual file system folder 971
virtual machine
how to test 880
importing 743
Virtual Machine Import Wizard 718, 841
using 743
Virtual Machine Platform 730, 767
Virtual Machine Preparation setup 741
Virtual Machine Preparation Tool 869
virtual machines
connecting to active machines 751
editing virtual machine properties 748
how Automated Application Converter selects the machines to use 767
list of supported 716
managing 738
preparing virtual machines for use with the Automated Application Converter 740
preparing with Virtual Machine Preparation setup 741
taking a snapshot 741
VMware VIX API requirement 743
virtual package
deleting Windows Installer package association 223
Virtual Package Editor 955
associating targets with a dependency 970
Cmd.exe 993
configuring file extension associations 983
configuring virtual services 990
creating shortcuts 980
debug tools 993
defining targets 979
editing the virtual registry 975
editing tips 963
entry points 979
extracting files from an App-V package 974
HREF scripts 986
including files and folders 971
opening a virtual package in 959
Regedit 993
save options 960
saving as a new package 960
saving as an upgrade 960
SCRIPTBODY scripts 986
setting environment variables 982
starting 958
using the App-V Application Launcher 991
viewing package history 968
virtual packages
deleting an association 223
importing during import of Windows Installer package 219
importing into Application Catalog 218, 219
importing into Application Catalog after import of Windows Installer package 221
manually associating with Windows Installer package 223
virtual services in an App-V package 990
virtualization 563, 578
about 1031
benefits of 694, 695, 1032
benefits of Citrix XenApp 711
building a Citrix profile using Repackager 707
building a ThinApp application using Repackager 704
cancelling 801
capturing context 775
context file 775
creating virtual applications 1031
customizing applications 1031
diagram of 1032
example diagram 694
example of 1032
getting started 691
in InstallShield 1033
including additional MSIs 1139, 1163
methods to convert Windows Installer packages 691
overview of 693
overview of Citrix profiles 705, 709, 1194
overview of Citrix XenApp 708, 1192
via command line 563
viewing conversion results 768
virtualization Assistants
integration with Project Assistant and Installation Designer 1034
navigating in 1035
opening the Installation Designer 1035
showing and hiding 1036
Virtualization not recommended 725, 755, 759, 788
Virtualization not supported 788
Virtualization Readiness 825
Virtualization Readiness column 724, 755, 759
Virtualization Readiness property 781
Virtualization Technology property 750, 764, 783, 795

Virtualize packages with detected settings 728
virtualizing 765
visibility 1275
  changing for features 1273
  feature 1275
visible property 1275
visiblity 1273
VIX API 743
VMware 667
  ThinApp application 656
  VIX API requirement 743
VMware ESX or ESXi Server 716, 726, 745, 829, 833
  taking a snapshot 742
VMware Repackaging Wizard 667
  VMware Virtual Machines panel 667
  Welcome panel 667
VMware ThinApp 656
VMware ThinApp application 656
VMware ThinApp Packages (*.exe) 729, 837
VMware Virtual Machines panel 667
  VMware Repackaging Wizard 667
VMware VIX API 743, 869
VMware Workstation 716
  setting Snapshots options 742
  taking a snapshot 742

W

Warning Frequency 1179
Warning Message 1179
Warning Period 1179
Warnings property 797
watermark 70
Web Service URL panel 1871
Welcome Panel 1869
Welcome panel 133, 135, 136, 546, 661, 665, 667, 1344, 1345, 1541, 1545, 1657, 1759
  Add Tool Wizard 133
  Application Isolation Wizard 1657
  Conflict Wizard 1541
  Distribution Wizard 1759
  Import INI File Wizard 1344
  Import REG File Wizard 1345
  Import Wizard 438
  in New Workflow Project Wizard 135
  Merge Wizard 455
  OS Snapshot Wizard 456
  Package Auto Import Wizard 451
  Patch Impact Analysis Wizard 1638
  Predeployment Analysis Preparation Wizard 1869
  Repackaging Wizard 546
  Rules Wizard 1545
  Setup Intent Wizard 665
  SmartScan Wizard 661
  VMware Repackaging Wizard 667
Welcome to InstallShield Tuner 1320
Welcome to Tuner view
  Create a new transform file option 1320
Where Clause Panel 1549
  Rules Wizard 1549
Where Clause tab 1522
Windows 2000 1338
  Add/Remove Programs 1338
Windows Installer File Selection panel 1658, 1870
  Application Isolation Wizard 1658
Windows Installer isolated components 1650
  isolation method 1650
Windows Installer package 1269, 1270
  and .context.msi 775
  building in Repackager 578
  handing invalid 1270
  prevalidating 1269
Windows installer package
  repackaging using the Repackaging Wizard 540
Windows Installer packages 211, 480, 1369, 1384
  about repackaging 764
  handing invalid in ConflictSolver 1369
  importing 207, 211, 213
  importing multiple using a configuration file 480
  importing simultaneously with merge modules 480
  reasons to avoid repackaging 764
  resolving conflicts directly within 1384
  validating in ConflictSolver 1362
Windows Installer Packages (*.msi)
  specifying as an output format 729
Windows registry
  and App-V Assistant 1070
  and Citrix Assistant 1220
  and ThinApp Assistant 1135
Windows Services
  integration into an App-V application 703
Windows services
  App-V support of 703, 1047
  running within the virtual environment 703, 1047
Windows Terminal Server
  WTS01 ACE 1459
  WTS02 ACE 1460
  WTS03 ACE 1461
  WTS04 ACE 1462
  WTS05 ACE 1463
  WTS06 ACE 1463
  WTS07 ACE 1464

WTS08 ACE 1465
WTS09 ACE 1466
Windows XP 1338
Add/Remove Programs 1338
WinINSTALL Conversion dialog box 641
WinINSTALL conversion problems 685
Repackager 685
WinINSTALL project 577
converting to Repackager project 577
Wise Installation project 577
converting to Repackager project 577
wizards 133, 135, 815, 1869
AdminStudio Interface 133
Application Isolation Wizard 1659
Finishing INI File Import panel in Import REG File 1345
Finishing Registry Import panel in Import REG File 1346
Import Conflict Options panel in Import INI File 1344
Import Conflict Options panel in Import REG File 1345
Import INI File 1344
Import INI File panel in Import INI File 1344
Import REG File 1345
Import Registry File panel in Import REG File 1345
Location panel in Packaging 1346
New Workflow Project Wizard 135
OS Security Patch wizard 1619
Package Summary panel in Packaging 1347
Packaging 1346
Patch Impact Analysis wizard 1638
predeployment analysis 1869
Predeployment Preparation Analysis 1869
Setup.exe panel in Packaging 1347
SMS panel in Packaging 1347
Welcome panel in Import INI File 1344
Welcome panel in Import REG File 1345
workflow 109
example using the New Workflow Project wizard 109
Workflow Manager 116, 268
and extended attributes 268
integration with AdminStudio projects 116
logging in as a guest 155
Workflow Selection panel 135
in New Workflow Project Wizard 135
workflow tasks 100, 118
associating tools with 100
workflows 101, 102, 103, 107, 108, 111, 118, 135, 136
creating 102
creating new 103, 107
deleting 103
editing 102
filtering 103
New Workflow Project Wizard 135, 136
renaming 103
saving 108
tasks 118
Workflows Templates tab 118
Write Copy isolation option 1128
WTS01 ACE 1459
WTS02 ACE 1460
WTS03 ACE 1461
WTS04 ACE 1462
WTS05 ACE 1463
WTS06 ACE 1463
WTS07 ACE 1464
WTS08 ACE 1465
WTS09 ACE 1466

X

XenApp. See Citrix XenApp
XML file 266
as extended attributes description 266

Z

ZENworks 1748, 1752
using Distribution Wizard to prepare packages for desktop application distribution 1748
using Distribution Wizard to prepare packages for server distribution 1752
ZENworks Configuration Management
preparing for distribution 1743
ZENworks Configuration Management Server Login 1787
ZENworks Desktop Application Panel
Distribution Wizard 1779
ZENworks Desktop Application panel 1779
ZENworks Desktop Management Agent
installing 574
ZENworks projects
.aot and .axt files 573
application object template files 574
converting using Repackager 573
ZENworks Server Distribution - Distributor Panel 1783
Distribution Wizard 1783
ZENworks Server Distribution - Object Panel 1781