

Novell IDM Driver for Schools Interoperability Framework

3.5.1

www.novell.com

IMPLEMENTATION GUIDE

September 28, 2007

N

Novell®

Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. See the [Novell International Trade Services Web page \(http://www.novell.com/info/exports/\)](http://www.novell.com/info/exports/) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright ©2003-2007 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed on the [Novell Legal Patents Web page \(http://www.novell.com/company/legal/patents/\)](http://www.novell.com/company/legal/patents/) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online Documentation: To access the latest online documentation for this and other Novell products, see [the Novell Documentation Web page \(http://www.novell.com/documentation\)](http://www.novell.com/documentation).

Novell Trademarks

For Novell trademarks, see [the Novell Trademark and Service Mark list \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

All third-party trademarks are the property of their respective owners.

Contents

- About This Guide** **9**

- 1 Introducing the Identity Manager Driver for SIF** **11**
 - 1.1 What's New 11
 - 1.1.1 Driver Features 11
 - 1.1.2 Identity Manager Features 11
 - 1.2 Key Driver Features 11
 - 1.2.1 Local Platforms 11
 - 1.2.2 Remote Platforms 12
 - 1.2.3 Role-Based Entitlements 12
 - 1.2.4 Password Synchronization Support 12
 - 1.2.5 Synchronized Objects 12
 - 1.3 About the Identity Manager Driver for SIF 12
 - 1.4 Understanding the Driver Configuration 14
 - 1.4.1 How the Identity Vault Is Updated When Data Changes in the Student Information System 14
 - 1.4.2 Data Mapping 16
 - 1.4.3 Sending Data from the Identity Vault to SIF 17

- 2 Planning** **19**
 - 2.1 Outlining Your Groups Of Students 19
 - 2.2 Creating Your Tree Structure 21
 - 2.2.1 Creating the Hierarchy of Containers for Students and Staff 21
 - 2.2.2 Identifying "Incomplete" Containers 25
 - 2.2.3 Identifying the "Search" Container 26
 - 2.2.4 Identifying "Disabled" Containers (Optional) 26
 - 2.2.5 Identifying eDirectory Templates 27
 - 2.3 Planning Driver and Replica Placement on Your Servers 27
 - 2.3.1 Determining How Many Zones You Have 27
 - 2.3.2 Planning Replica Placement 27
 - 2.3.3 Examples of Driver and Replica Placement 28
 - 2.4 Specifying the Pattern for User IDs 31
 - 2.5 Deciding Whether You Want the Driver to Manage Existing User Accounts 33
 - 2.6 Password Synchronization 33
 - 2.7 Gathering Information for the Driver Configuration 34

- 3 Installing the Driver** **35**
 - 3.1 Prerequisites 35
 - 3.1.1 Software Requirements 35
 - 3.1.2 Hardware Considerations 36
 - 3.2 Installing the Identity Manager Driver for SIF 36
 - 3.3 What's Next 36
 - 3.4 Activating the Driver 36

- 4 Upgrading the Driver** **37**
 - 4.1 Upgrading the Driver in Designer 37

4.2	Upgrading the Driver in iManager.	40
5	Deploying the Driver	41
5.1	Creating and Configuring the Driver	41
5.1.1	Prerequisites	41
5.1.2	Importing the Driver Configuration File in Designer	42
5.1.3	Importing the Driver Configuration File in iManager.	42
5.1.4	Configuration Parameters	43
5.1.5	Post-Configuration Tasks.	44
5.2	Preparing the ZIS and the Student Information System.	45
5.2.1	Configuring the ZIS to Recognize the Driver	45
5.2.2	Optimizing Data in the Student Information System	46
5.3	Starting and Testing the Driver.	46
5.4	Synchronizing the Identity Vault the First Time	48
5.4.1	Option 1: Populate the Identity Vault Using Migrate into Identity Vault	48
5.4.2	Option 2: Manage Existing Identity Vault User Accounts.	49
5.4.3	Option 3: Don't Manage Existing Identity Vault User Accounts	50
5.4.4	Using Migrate into Identity Vault to Populate or Update the Identity Vault.	51
5.5	Synchronizing the Identity Vault Each School Year	52
5.5.1	New Year Options for Students in School or Grade Containers	52
5.5.2	New Year Tasks for Students in Graduation Year Containers.	55
6	Customizing the Driver	57
6.1	Driver Parameters	57
6.2	Setting Up Security.	57
6.2.1	Server Authentication.	57
6.2.2	Client Authentication	58
6.3	Identity Manager Association Keys	59
6.4	Mapping SIF XML to the eDirectory Schema	59
7	Activating the Driver	61
8	Synchronizing Objects	63
8.1	What Is Synchronization?.	63
8.2	When Is Synchronization Done?	63
8.3	How Does the Metadirectory Engine Decide Which Object to Synchronize?.	64
8.4	How Does Synchronization Work?	65
8.4.1	Scenario One	65
8.4.2	Scenario Two	67
8.4.3	Scenario Three.	68
9	Managing the Driver	71
9.1	Starting, Stopping, or Restarting the Driver	71
9.1.1	Starting the Driver in Designer	71
9.1.2	Starting the Driver in iManager	71
9.1.3	Stopping the Driver in Designer.	71
9.1.4	Stopping the Driver in iManager.	71
9.1.5	Restarting the Driver in Designer	72
9.1.6	Restarting the Driver in iManager	72
9.2	Using the DirXML Command Line Utility	72
9.3	Viewing Driver Versioning Information	72

9.3.1	Viewing a Hierarchical Display of Versioning Information	72
9.3.2	Viewing the Versioning Information as a Text File	74
9.3.3	Saving Versioning Information	76
9.4	Reassociating a Driver Set Object with a Server Object	77
9.5	Changing the Driver Configuration	78
9.6	Storing Driver Passwords Securely with Named Passwords	78
9.6.1	Using Designer to Configure Named Passwords	79
9.6.2	Using iManager to Configure Named Passwords	79
9.6.3	Using Named Passwords in Driver Policies	81
9.6.4	Using the DirXML Command Line Utility to Configure Named Passwords	81
9.7	Adding a Driver Heartbeat	85
10	Troubleshooting	87
10.1	Viewing Status Messages for the Identity Manager Driver for SIF	87
10.1.1	Using the Status Logs	87
10.1.2	Using the DSTrace Screen	88
10.1.3	Identity Manager Status Levels	88
10.2	Error Messages	88
10.3	Common HTTP Status Codes	102
10.4	ZIS Return Status	103
10.5	Troubleshooting Driver Processes	103
10.5.1	Viewing Driver Processes	104
11	Backing Up the Driver	111
11.1	Exporting the Driver in Designer	111
11.2	Exporting the Driver in iManager	111
12	Security: Best Practices	113
A	DirXML Command Line Utility	115
A.1	Interactive Mode	115
A.2	Command Line Mode	124
B	Properties of the Driver	129
B.1	Driver Configuration	129
B.1.1	Driver Module	130
B.1.2	Driver Object Password	130
B.1.3	Authentication	131
B.1.4	Startup Option	132
B.1.5	Driver Parameters	133
B.2	Global Configuration Values	134
B.2.1	Global Configuration Values > Driver Configuration	135
B.2.2	Global Configuration Values > Student Configuration	138
B.2.3	Global Configuration Values > Staff and Employee Configuration	138
B.2.4	Global Configuration Values > Zone Configuration	139
B.2.5	Global Configuration Values > Student Placement	141
B.2.6	Global Configuration Values > SIF Provider Configuration	142
B.2.7	Global Configuration Values > Password Configuration	143
B.3	Named Passwords	144
B.4	Engine Control Values	144
B.5	Log Level	146

B.6	Driver Image	147
B.7	Security Equals	148
B.8	Filter	148
B.9	Edit Filter XML	148
B.10	Misc	149
B.11	Excluded Users	150
B.12	Driver Manifest	150
B.13	Driver Inspector	150
B.14	Driver Cache Inspector	151
B.15	Inspector	151
B.16	Server Variables	152

Glossary **155**

About This Guide

The Identity Manager Driver for Schools Interoperability Framework (SIF*) lets you automatically provision users in the Identity Vault and synchronize user accounts the Identity Vault with user data from a SIF-enabled student information system.

This configurable solution gives you the ability to increase productivity, streamline school processes, and reduce errors by automating the transfer of user data to the Identity Vault.

The guide contains the following sections:

- ◆ Chapter 1, “Introducing the Identity Manager Driver for SIF,” on page 11
- ◆ Chapter 2, “Planning,” on page 19
- ◆ Chapter 3, “Installing the Driver,” on page 35
- ◆ Chapter 4, “Upgrading the Driver,” on page 37
- ◆ Chapter 5, “Deploying the Driver,” on page 41
- ◆ Chapter 6, “Customizing the Driver,” on page 57
- ◆ Chapter 7, “Activating the Driver,” on page 61
- ◆ Chapter 8, “Synchronizing Objects,” on page 63
- ◆ Chapter 9, “Managing the Driver,” on page 71
- ◆ Chapter 10, “Troubleshooting,” on page 87
- ◆ Chapter 11, “Backing Up the Driver,” on page 111
- ◆ Chapter 12, “Security: Best Practices,” on page 113
- ◆ “Glossary” on page 155
- ◆ Appendix A, “DirXML Command Line Utility,” on page 115
- ◆ Appendix B, “Properties of the Driver,” on page 129

SIF is an open standard created to allow K-12 education applications to exchange data effectively. The driver for SIF works as a SIF Agent.

The 3.5. release of the driver conforms to SIF Implementation Specifications 1.1 and 1.5r1. For information about the SIF specifications, see the [Schools Interoperability Framework Web site \(http://www.sifinfo.org\)](http://www.sifinfo.org).

This release supports only English versions of NetWare[®] and Windows*.

Audience

This guide is for school administrators and others who implement the Identity Manager Driver for Schools Interoperability Framework (SIF) in a K-12 school environment.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the

online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

For the most recent version of this document, see the [Drivers Documentation Web site \(http://www.novell.com/documentation/idm35drivers/index.html\)](http://www.novell.com/documentation/idm35drivers/index.html).

Additional Documentation

For documentation on using Identity Manager and the other drivers, see the [Identity Manager Documentation Web site \(http://www.novell.com/documentation/idm35/\)](http://www.novell.com/documentation/idm35/).

Documentation Conventions

In Novell® documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

When a single pathname can be written with a backslash for some platforms or a forward slash for other platforms, the pathname is presented with a backslash. Users of platforms that require a forward slash, such as Linux* or UNIX*, should use forward slashes as required by your software.

Introducing the Identity Manager Driver for SIF

1

Managing student and staff accounts manually in a K-12 school system can be a time-consuming job for an administrator. Support time and opportunities for human error are multiplied during the influx of students at the beginning of the year, the changes in student enrollment or staff employment throughout the school year, and the end-of-year issues such as disabling accounts or moving students to reflect a school or grade change.

The Identity Manager Driver for Schools Interoperability Framework (SIF) lets you automatically synchronize student, faculty, and staff data in K-12, SIF-enabled applications with user objects in the Identity Vault. Using the driver for managing user accounts provides a great return on investment.

A configuration file is provided for provisioning students and staff, using the Student Information System as the authoritative data source. You can also customize the configuration.

- ◆ [Section 1.1, “What’s New,” on page 11](#)
- ◆ [Section 1.2, “Key Driver Features,” on page 11](#)
- ◆ [Section 1.3, “About the Identity Manager Driver for SIF,” on page 12](#)
- ◆ [Section 1.4, “Understanding the Driver Configuration,” on page 14](#)

1.1 What’s New

In this section:

- ◆ [Section 1.1.1, “Driver Features,” on page 11](#)
- ◆ [Section 1.1.2, “Identity Manager Features,” on page 11](#)

1.1.1 Driver Features

There are no new features for the SIF driver with the release of Identity Manager 3.5.1.

1.1.2 Identity Manager Features

For information about the new features in Identity Manager, see “[What's New in Identity Manager 3.5.1?](#)” in the *Identity Manager 3.5.1 Installation Guide*.

1.2 Key Driver Features

The sections below contains a list of the key driver features.

1.2.1 Local Platforms

The SIF driver can be installed locally on a NetWare® or Windows server.

1.2.2 Remote Platforms

The SIF driver can use the Remote Loader service. The Remote Loader service for the SIF driver can be installed on NetWare and Windows Platforms. For more information about installing the Remote Loader services, see “[Installing the Remote Loader](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

1.2.3 Role-Based Entitlements

The SIF driver does not support Role-Based Entitlements.

1.2.4 Password Synchronization Support

The SIF driver can synchronize passwords between the Identity Vault and the Zone if the SIF driver and the Zone are using SIF Specification 1.5r1 or later. For more information, see [Section 2.6, “Password Synchronization,”](#) on page 33.

1.2.5 Synchronized Objects

The SIF driver synchronizes StudentPersonal, StudentSchoolEnrollment, SchoolInfo, and StaffPersonal objects from the SIF system to users object in the Identity Vault. It also synchronizes container objects. For more information about the SIF driver, see [Section 1.3, “About the Identity Manager Driver for SIF,”](#) on page 12.

1.3 About the Identity Manager Driver for SIF

Schools use many applications to organize data for a K-12 education environment, such as systems for student administration, network access, food services, and library automation. These diverse systems often contain duplicate information. If the applications do not communicate with each other to share information, school administrators and information technology personnel must deal with the challenges of manually provisioning students and using redundant data entry to keep the systems synchronized.

For example, when new students enroll at a school, they need network access and a home directory for their files. If the [Student Information System \(SIS\)](#) does not communicate with the Identity Vault, the network administrator must manually create a user account and assign network resources for each new student, one at a time. Without interoperability between the systems, each subsequent change to student data also requires manual intervention to keep the Identity Vault users updated.

To create interoperability between the Student Information System and the Identity Vault, Novell® provides the Identity Manager Driver for Schools Interoperability Framework (SIF).

SIF is an open standard created to allow K-12 education applications to exchange data effectively. The Identity Manager Driver for SIF works as a SIF Agent. The 3.5 release of the driver conforms to SIF Implementation Specifications 1.1 and 1.5r1. For information about the specifications, see the [Schools Interoperability Framework Web site \(http://www.sifinfo.org\)](http://www.sifinfo.org).

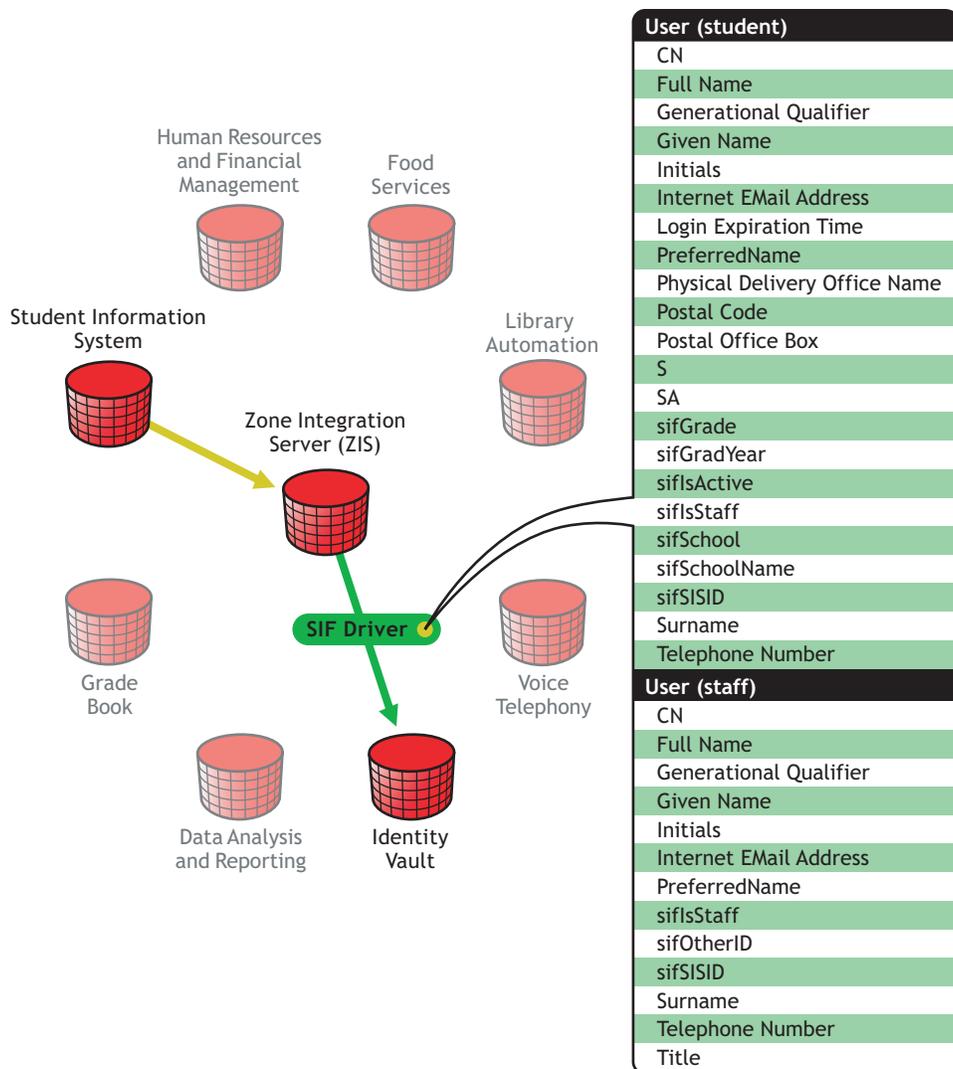
The driver eliminates the need to manually provision, change, or delete User objects for a school system in the Identity Vault. Instead, the changes in the Identity Vault are made automatically, mirroring the data from the Student Information System. When a student is entered in the Student Information System, he or she is automatically given a User object in the Identity Vault, in the

correct container, with network resources. If the student's status changes, such as a grade-level change or a move to a different school, the change is reflected in the Identity Vault and the User object is moved to a different container, if appropriate. If a student leaves the school system, the user object's login is disabled. The same kind of synchronization is done for staff and faculty users.

In a school network that uses the SIF standards, the Student Information System publishes information to the **Zone Integration Server (ZIS)**.

The driver, like other **SIF Agents**, registers with the ZIS so it can receive information. The driver receives the StudentPersonal, StudentSchoolEnrollment, and SchoolInfo objects for students, and the StaffPersonal object for faculty and staff. The driver uses that information to create User objects for students and staff, give them appropriate attributes, and automatically place them in the correct container in the Identity Vault. This flow of information and the list of the attributes that are populated in the Identity Vault are shown in the following diagram.

Figure 1-1 The Role of the Identity Manager Driver for SIF



In the driver configuration provided, the Identity Vault receives information from the Student Information System. You can customize the configuration to change how students and staff are provisioned, and cause the Identity Vault to send information to the ZIS.

1.4 Understanding the Driver Configuration

After you install Identity Manager and the driver, you create a Driver object. A Driver object represents an instance of the Identity Manager Driver for SIF.

A driver configuration file, `SIFAgent-IDM3_5_0-V1.xml`, is provided to get you up and running with a minimum of customization. This section explains what the driver configuration does.

- ♦ “How the Identity Vault Is Updated When Data Changes in the Student Information System” on page 14
- ♦ “Data Mapping” on page 16
- ♦ “Sending Data from the Identity Vault to SIF” on page 17

For information about Identity Manager in general, see the *Novell Identity Manager 3.5.1 Administration Guide*.

1.4.1 How the Identity Vault Is Updated When Data Changes in the Student Information System

The following tables describe what the configuration does to provision user accounts and keep the Identity Vault updated when changes occur in the Student Information System. There are two types of user accounts; students and staff. [Table 1-1](#) contains information about student provisioning and [Table 1-2](#) contains information about staff provisioning.

Table 1-1 Student Provisioning

Change in Student Data	Synchronization in the Identity Vault
A student is added	<ul style="list-style-type: none">♦ Creates an the Identity Vault User object with a unique user ID.♦ Populates the User object attributes with data from the Student Information System. The attributes are listed in “Data Mapping” on page 16.♦ Places the user in the correct container as determined by the student’s school and grade level or graduation year.♦ Uses a template (if you specify one) to set default properties for the user, group membership, login restrictions, and password restrictions.♦ (NetWare only) Creates a home directory in the file system. (You must use a template to specify this.)

Change in Student Data	Synchronization in the Identity Vault
A student's information is modified	<ul style="list-style-type: none"> ◆ Modifies the Identity Vault User object attributes accordingly. The attributes are listed in "Data Mapping" on page 16. ◆ If appropriate, moves the User object to a different container in the tree. <p>For example, a school or grade level/graduation year change could trigger moving the user to a different container.</p> ◆ (Optional) If any of the attributes creates a User ID change, the user account is renamed. ◆ The home directory is not moved.
A student withdraws from school or graduates	<ul style="list-style-type: none"> ◆ On the Exit Date, disables the login of the User object in the Identity Vault. ◆ (Optional) On the Exit Date, moves the user account to the Disabled directory. ◆ The home directory is not deleted.
A student returns to the school system (an Entry Date that is newer than the Exit Date is entered in the Student Information System)	<ul style="list-style-type: none"> ◆ Enables the login of the User object in the Identity Vault. ◆ Moves the user account from the Disabled directory to the correct student container. ◆ The User object still has rights to the home directory.
A student is removed from the Student Information System	<ul style="list-style-type: none"> ◆ On the Exit Date, disables the login of the User object in the Identity Vault. ◆ (Optional) Moves the user account to the Disabled directory. ◆ The home directory is not deleted.

Table 1-2 *Staff Provisioning*

Change in Staff Data	Synchronization in the Identity Vault
Staff is added	<ul style="list-style-type: none"> ◆ Creates an the Identity Vault User object with a unique User ID. ◆ Populates the User object attributes with data from the Student Information System. The attributes affected are listed in "Data Mapping" on page 16. ◆ Places the user in the correct container, as determined by the Zone. ◆ Uses a template (if you specify one) to set default properties for the user, including group membership, login restrictions, and password restrictions. ◆ (NetWare only) Creates a home directory in the file system. (You must use a template to specify this.)
Staff information is modified	<ul style="list-style-type: none"> ◆ Modifies the Identity Vault user accordingly. The attributes maintained are listed in "Data Mapping" on page 16. ◆ (Optional) If any of the attributes creates a User ID change, the user account is renamed.

Change in Staff Data	Synchronization in the Identity Vault
Staff removed from the Student Information System	<ul style="list-style-type: none"> ◆ Disables the User object in the Identity Vault. ◆ (Optional) Moves the user account to the Disabled directory. ◆ The home directory is not removed from the file system.

1.4.2 Data Mapping

The Identity Manager Driver for SIF uses data from the Student Information System to synchronize the following User class attributes in the Identity Vault. [Table 1-3](#) contains a list of the eDirectory attribute, the SIF objects, and the SIF attributes.

Table 1-3 *User Class Attributes*

eDirectory Attribute	SIF Object	SIF Attribute
CN	StudentPersonal or StaffPersonal	CN is formed from the combination of several SIF attributes.
Full Name	StudentPersonal or StaffPersonal	Name/FullName
Generational Qualifier	StudentPersonal or StaffPersonal	Name/Suffix
Given Name	StudentPersonal or StaffPersonal	Name/FirstName
Initials	StudentPersonal or StaffPersonal	Name/MiddleName
Internet EMail Address	StudentPersonal or StaffPersonal	Email
Login Expiration Time	StudentSchoolEntrollment	EntryDate and ExitDate When ExitDate is newer than EntryDate, the login is set to expire on the ExitDate. When the EntryDate is newer than the ExitDate, the expiration date is removed.
personalTitle	StudentPersonal or StaffPersonal	Name/Prefix
preferredName	StudentPersonal or StaffPersonal	Name/PreferredName
Physical Delivery Office Name	StudentPersonal or StaffPersonal	Address/City
Postal Code	StudentPersonal or StaffPersonal	Address/PostalCode
Postal Office Box	StudentPersonal or StaffPersonal	Address/Street/Line2

eDirectory Attribute	SIF Object	SIF Attribute
S	StudentPersonal or StaffPersonal	Address/StatePr
SA	StudentPersonal or StaffPersonal	Address/Street/Line1
Surname	StudentPersonal or StaffPersonal	Name/LastName
Telephone Number	StudentPersonal or StaffPersonal	PhoneNumber
Title	StaffPersonal	Name/Title
DirXML-sifGrade	StudentSchoolEnrollment	GradeLevel
DirXML-sifGradYear	StudentPersonal	GradYear
DirXML-sifIsStaff	StudentPersonal or StaffPersonal	Not set from a particular attribute. Set to True if the SIF object is StaffPersonal. Otherwise, it is set to False.
DirXML-sifSchool	SchoolInfo	IdentificationInfo
DirXML-sifSchoolName	SchoolInfo	SchoolName
DirXML-sifSISID	SchoolInfo	RefId
DirXML-sifSSEGUID	StudentSchoolEnrollment	RefId

1.4.3 Sending Data from the Identity Vault to SIF

The SIF Driver is generally used to provision users from a SIF-enabled Student Information System to the Identity Vault. The driver is configured, by default, to send no data from the Identity Vault to the Zone Integration Server (ZIS) and the Student Information System. The Student Information System is considered to be the authoritative data source.

However, the driver is capable of bidirectional synchronization and can send data to the ZIS and SIF. There are two ways you might choose to use this bidirectional capability:

- ◆ Configure the driver as the authoritative source for some user attributes or for new users.

If you want the Identity Vault to be the authoritative source for some user attributes, you could configure the driver to send certain attributes from the Identity Vault to SIF.

If your business practices allow users to be entered manually in the Identity Vault who are not entered in the Student Information System first, you could also configure the driver to send new users from the Identity Vault to SIF.

- ◆ Configure the driver to be the SIF provider for all student and staff data.

If your Student Information System is not SIF-enabled, but you have other SIF-enabled applications, you might choose to configure the SIF Driver to function as the authoritative source for students and staff.

In this role, the SIF Driver is the SIF provider for StudentPersonal, StudentSchoolEnrollment, SchoolInfo, StaffPersonal, and SIF Authorization objects. Being the provider means this driver

responds when other SIF-enabled applications send SIF queries for information about students and staff.

For example, you could export student and staff information from your Student Information System and import it into the Identity Vault, using a database import. At the start of the school year, the other SIF Agents in the Zone would populate their databases by querying for all students. If you register the SIF Driver as the provider for the Zone, the queries would be routed to the SIF Driver. During the school year, as student and staff information in the Identity Vault is updated, either by database import or by updating manually, the SIF Driver would send those updates to SIF, thereby keeping the other SIF-enabled applications current.

You would not enable this option if you have a SIF-enabled Student Information System. Only one Agent in a Zone can be the provider. If you have a SIF-enabled Student Information System, we recommend that the Student Information System be the provider.

If you configure the SIF Driver to send new users or to be the provider of all student and staff information, at a minimum you must provide the user attributes listed in [Table 1-4](#) when creating a user object in the Identity Vault. A new user object is not sent from the Identity Vault to SIF unless these attributes have values.

Table 1-4 *Required User Attributes*

Type of User Account	Attribute
Student	Given Name
	Surname
	DirXML-sifGrade
	DirXML-sifGradYear
	DirXML-sifSchool
	DirXML-sifSISID
Staff	Given Name
	Surname
	DirXML-sifSISID

Installing Identity Manager and the Identity Manager Driver for SIF is a simple process. However, you need to do some planning to make sure that you prepare your tree structure, know where to place your partition replicas and the driver, consider some of the choices you make when configuring the driver, and gather additional configuration information.

In this section:

- ◆ [Section 2.1, “Outlining Your Groups Of Students,” on page 19](#)
- ◆ [Section 2.2, “Creating Your Tree Structure,” on page 21](#)
- ◆ [Section 2.3, “Planning Driver and Replica Placement on Your Servers,” on page 27](#)
- ◆ [Section 2.4, “Specifying the Pattern for User IDs,” on page 31](#)
- ◆ [Section 2.5, “Deciding Whether You Want the Driver to Manage Existing User Accounts,” on page 33](#)
- ◆ [Section 2.6, “Password Synchronization,” on page 33](#)
- ◆ [Section 2.7, “Gathering Information for the Driver Configuration,” on page 34](#)

2.1 Outlining Your Groups Of Students

Outlining the groups of students that you want the Identity Manager Driver for SIF to manage provides the following benefits:

- ◆ It helps you prepare for creating the tree structure you want to use for the containers that hold student users, described in the next section, [Section 2.2, “Creating Your Tree Structure,” on page 21](#).
- ◆ It helps you configure the driver more quickly. In the driver configuration, you must specify each group of students, their location, and the Template object to use.

As a planning tool, we recommend that you create a table to represent the groups of students. This table is helpful you when you are configuring the driver, to make sure you have all the containers and templates you need.

When identifying the groups, use the identifiers used by your student information system for school code and for grade or graduation year. To configure the driver correctly, you need to know the codes your student information system uses.

You can choose to group students by grade level, graduation year, school, or in a single container. (Example tree structures are shown in [“Creating Containers for Students” on page 21](#).)

For example, consider a school district named Alpine District, with one Zone and three schools: Canyon Elementary, Sunset Middle School, and Highland High School. To group the students by grade level, you would create a table like [Table 2-1](#).

Table 2-1 *An Example Planning Table*

School Code	Grade or Graduation Year	Container DN	Template DN
CElem	KG		
	01		
	02		
	03		
	04		
	05		
SMiddle	06		
	07		
HHS	08		
	09		
	10		
	11		
	12		

After completing the planning section [Section 2.2, “Creating Your Tree Structure,”](#) on page 21, you would fill in the rest of the table with the container DN and template for each student group.

For example, if you were using one container per grade level, and decided to use one template per school with the templates placed in the Alpine container, your table would now look like [Table 2-2](#).

Table 2-2 *Container DN Mapped to a Template for Each User Group*

School Code	Grade or Graduation Year	Container DN	Template DN
CElem	KG	Alpine\District\Canyon Elem\K	Alpine\Elementary
	01	Alpine\District\Canyon Elem\01	Alpine\Elementary
	02	Alpine\District\Canyon Elem\02	Alpine\Elementary
	03	Alpine\District\Canyon Elem\03	Alpine\Elementary
	04	Alpine\District\Canyon Elem\04	Alpine\Elementary
	05	Alpine\District\Canyon Elem\05	Alpine\Elementary
SMiddle	06	Alpine\District\Canyon Elem\06	Alpine\Elementary
	07	Alpine\District\Sunset Middle\07	Alpine\Middle
HHS	08	Alpine\District\Sunset Middle\08	Alpine\Middle
	09	Alpine\District\Highland High\09	Alpine\HighSchool

School Code	Grade or Graduation Year	Container DN	Template DN
	10	Alpine\District\Highland High\10	Alpine\HighSchool
	11	Alpine\District\Highland High\11	Alpine\HighSchool
	12	Alpine\District\Highland High\12	Alpine\HighSchool

Use the table as a reference when you configure the driver, as described in [Section 5.1, “Creating and Configuring the Driver,”](#) on page 41.

2.2 Creating Your Tree Structure

In this planning step, you review your tree and add or update the containers you want to use to hold student and users, add containers for incomplete or disabled user objects, and make sure you have the eDirectory™ template objects you need.

- ♦ [“Creating the Hierarchy of Containers for Students and Staff”](#) on page 21
- ♦ [“Identifying “Incomplete” Containers”](#) on page 25
- ♦ [Section 2.2.4, “Identifying “Disabled” Containers \(Optional\),”](#) on page 26
- ♦ [“Identifying eDirectory Templates”](#) on page 27

2.2.1 Creating the Hierarchy of Containers for Students and Staff

We recommend that your eDirectory tree have a hierarchal structure for holding User objects.

This part of the tree should begin at least one level down from the root container, so that the root container can contain the Admin user and other objects you don’t want the driver to manage. We recommend that students and staff be kept in separate eDirectory containers.

As part of your planning, you need to decide how you want to group your student users.

The container names don’t need to be identical to the school code or grade code used in the student information system.

In this section:

- ♦ [“Creating Containers for Students”](#) on page 21
- ♦ [“Creating One or More Containers for Staff Users”](#) on page 23

Creating Containers for Students

The tree structure can be created according to your needs; the only thing required by the driver is that you specify which containers students and staff are placed in. In the examples in this manual, separate school containers are shown, and sometimes grade or graduation year containers as well, but this is not required.

One example tree structure would be to create a single district container below the root container. Below the district container you could create containers representing each school. Below each school container could be containers representing the grade levels or graduation years in the school.

Figure 2-1 illustrates this example hierarchy, with the District container, the Highland High school container, and the 12th grade container.

Figure 2-1 Example Tree Structure, with Grade Level Containers

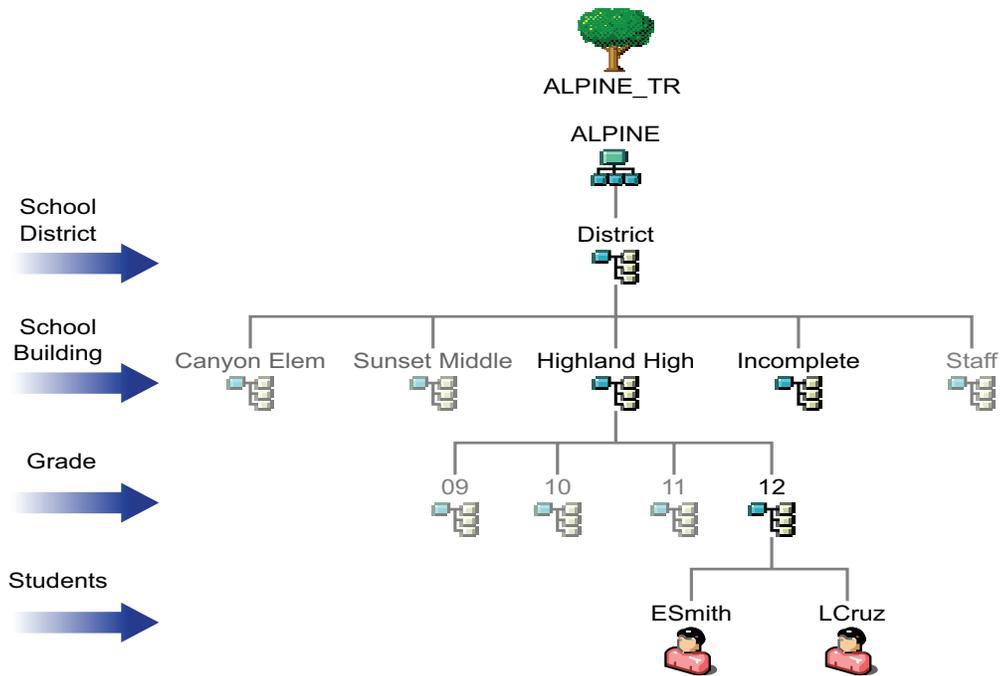
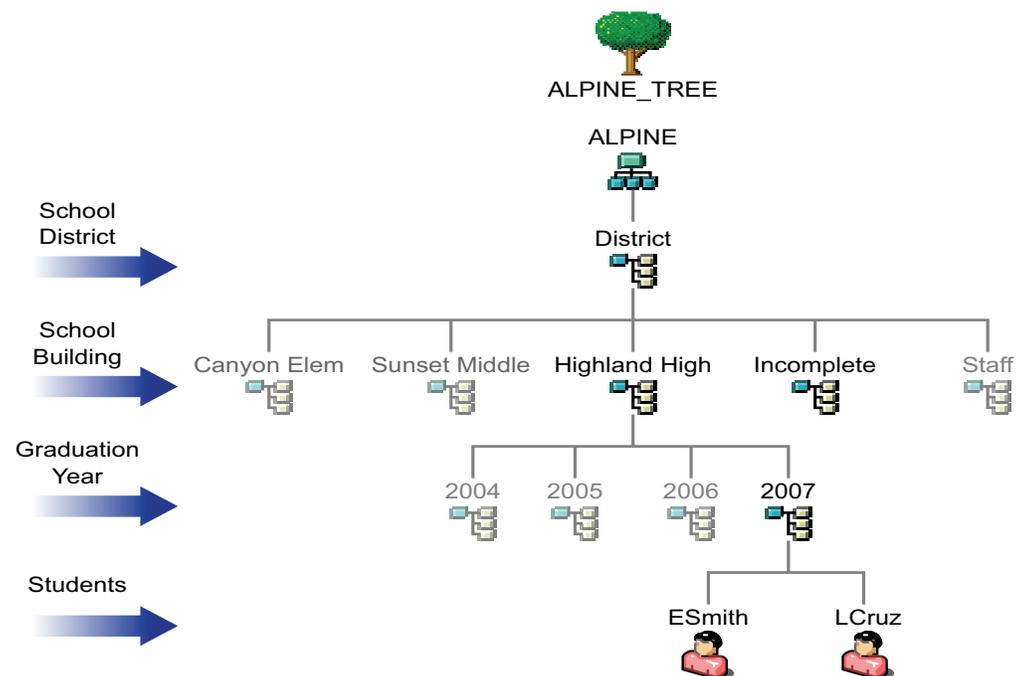


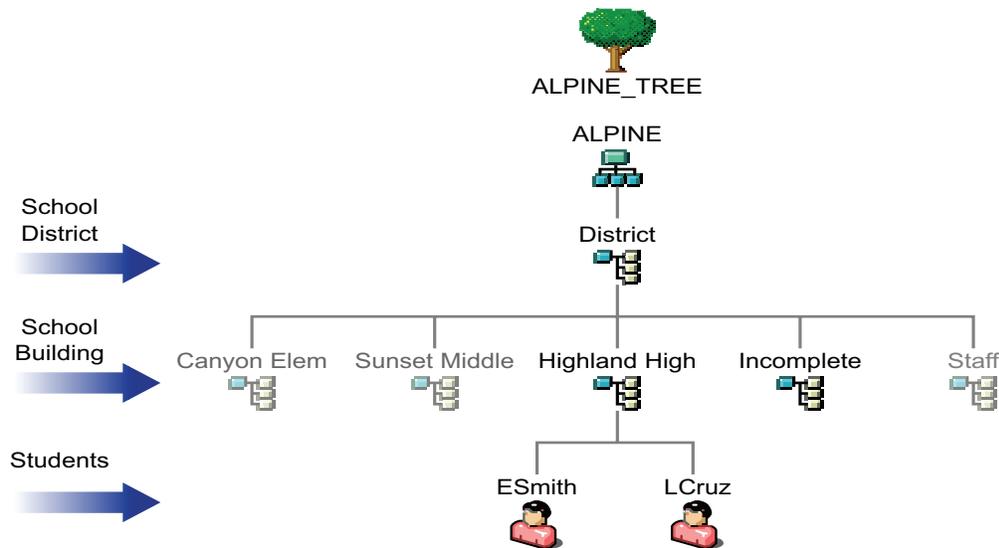
Figure 2-2 illustrates using the same kind of structure with graduation year containers instead of grade level containers.

Figure 2-2 Example Tree Structure, with Graduation Year Containers



Another way you could organize the tree is to eliminate the optional grade or graduation year level, and use only school containers, as shown in [Figure 2-3](#).

Figure 2-3 Example Tree Structure, without Grade Level Containers



Creating One or More Containers for Staff Users

In this planning step, you review your tree and identify or create the container you want to use to hold staff users.

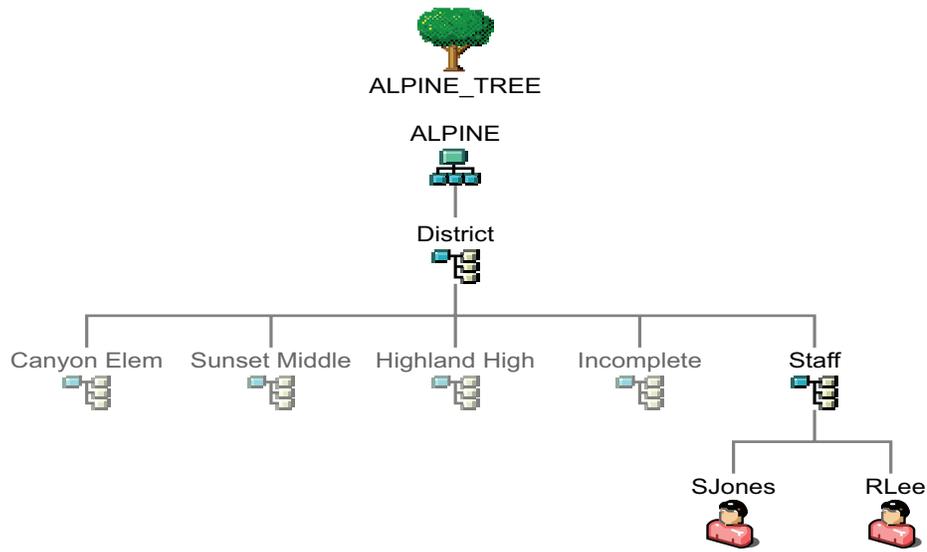
This container should be at least one level down from the root container, so that the root container can contain the Admin user and other objects you don't want the driver to change.

Each Zone that you configure specifies a Staff container.

For this part of your planning, it's helpful to know how many Zones you have, as discussed in [Section 2.3, "Planning Driver and Replica Placement on Your Servers,"](#) on page 27.

If you have a single Zone, you could place your staff users in a container below the district-level container, as illustrated in the following figure.

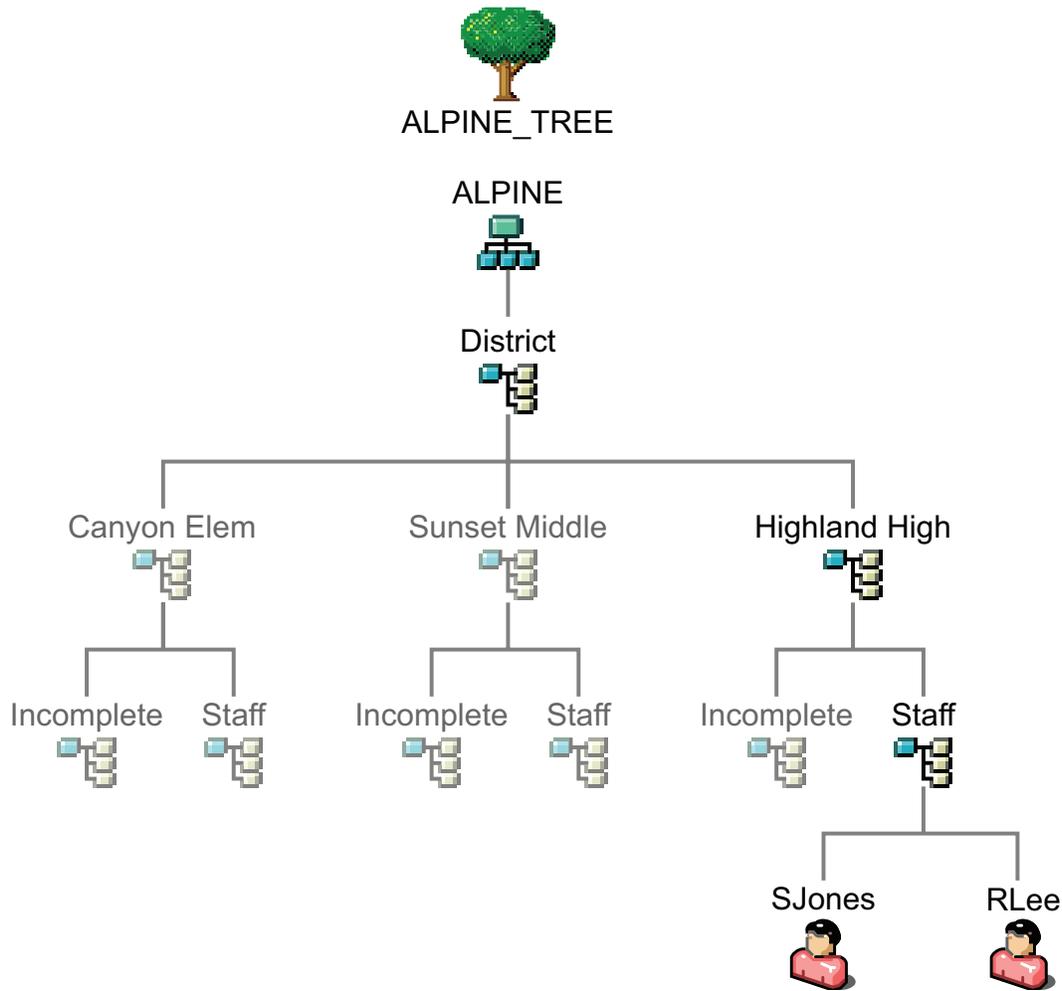
Figure 2-4 Example Tree Structure for Staff for Single Zone



If you have multiple zones, you have two choices for placing staff users:

- ◆ You could specify the same container for all your staff Driver objects, so all staff users are created in the same container regardless of which Zone they are represented in. This would be like the scenario illustrated in the previous figure.
- ◆ You could create one container for staff for each Zone, as illustrated in the following figure.

Figure 2-5 Example Tree Structure for Staff for Multiple Zones



2.2.2 Identifying “Incomplete” Containers

You need to identify a container to be used as the Incomplete container, so the driver has a place to hold information for students that it can't place correctly in the tree. This container is needed when a student's information is incomplete. If desired, you can specify an existing container to be used for this purpose instead of creating a new one.

If you have only one Zone, we recommend that you create one Incomplete container below the district container, as illustrated in [Figure 2-6 on page 29](#).

If you have multiple Zones, we recommend that you create an Incomplete container below each school container, as illustrated in [Figure 2-8 on page 30](#). This way, the users who are not yet placed correctly or who require administrator intervention are grouped by school.

Here are two examples of situations in which the driver would place students in the Incomplete container:

- ♦ A student has been entered into the student information system but the grade level or graduation year has not yet been entered.

When the grade level or graduation year is entered into the student information system, Identity Manager automatically creates the user in the correct container, using the correct template.

- ♦ A student has a school and grade level for which no container has been specified in the rules, the container specified does not exist, or there is a syntax problem with a rule.

For example, if the driver were set up for students in grades K-6 at Canyon Elementary, but the eDirectory administrator setting up the driver mistakenly left out a rule for the 5th grade, then Identity Manager would not know where to place students with the school of Canyon Elementary and the grade level of 05. Identity Manager would place them in the Incomplete container awaiting intervention by the eDirectory administrator.

The administrator needs to fix the rules and then place the students in the right container using the right template. If no template is desired, the User objects could simply be manually moved to the correct container. If the User objects need to be created by using a template, the administrator needs to first delete them from the Incomplete container. Then, they need to be re-created with the correct template in the correct container either manually or by using the Migrate into the Identity Vault command to cause the driver to re-create them. (See [“Using Migrate into Identity Vault to Populate or Update the Identity Vault” on page 51.](#))

You need to specify the DN of the Incomplete container when you configure the Driver object.

2.2.3 Identifying the “Search” Container

The search container is the point in the Identity Vault below which User IDs must be unique. When creating a new User object, the driver searches the Identity Vault to verify that the new User ID is not already in use. The search container and all subcontainers are searched. Choose the district container or a container that is high enough in the tree that user IDs are unique for all students and staff. For example, for the environment shown in [Figure 2-6, “Example Tree for One Zone, Showing Partitioning,” on page 29](#), you would specify the District container. A single search container is used for all zones.

If you specify Yes in the *Send New Users to SIF field*, only users created in this container and its subcontainers are sent to SIF.

2.2.4 Identifying “Disabled” Containers (Optional)

The driver configuration gives you the option to automatically move a student user to a different container if the user’s login is disabled. This option makes it easy for the administrator to identify all disabled accounts.

If you want to use this option, you must specify which container or containers you want the user objects to be placed in. If desired, you can use an existing container for this purpose instead of creating a new one.

If you have only one Zone, we recommend that you create one Disabled container below the district container, as illustrated in [Figure 2-6 on page 29](#).

If you have multiple Zones, we recommend that you create a Disabled container below each school container, as illustrated in [Figure 2-7 on page 29](#). This way, the users who have login disabled are grouped by school.

A student account is disabled when an exit date is set in the student information system. For example, this could happen when a student withdraws from school. If the student returns to school, a

new entry date is set in the student information system. The student's account is then enabled and moved to the appropriate student container.

2.2.5 Identifying eDirectory Templates

Decide which eDirectory User Template objects you want the Identity Manager Driver for SIF to use when creating new users. An eDirectory User Template object is not required for the driver, but it helps automate User object creation by allowing you to specify standard properties that can then be applied to new User objects.

For example, you might decide to have one Template object corresponding to each container where student users are grouped, such as one per school or grade, and a different Template object for staff users.

To prepare for configuring the driver, review the Template objects you have and update or add new ones if necessary. The Template objects that the driver needs access to must be on the server where the driver is running.

2.3 Planning Driver and Replica Placement on Your Servers

In this section:

- ♦ [“Determining How Many Zones You Have” on page 27](#)
- ♦ [“Planning Replica Placement” on page 27](#)
- ♦ [“Examples of Driver and Replica Placement” on page 28](#)
 - ♦ [“Example: Placing Drivers and Replicas for One Zone” on page 28](#)
 - ♦ [“Example: Placing Drivers and Replicas for Multiple Zones” on page 29](#)

2.3.1 Determining How Many Zones You Have

Consult with your student information systems administrator to find out how many Zones your environment is using and what they are managing.

Some SIF-enabled student information systems use one Zone for a whole district; some use multiple Zones, such as one per school.

A single instance of the Identity Manager Driver for SIF supports up to 10 Zones. If you have more than 10 Zones we recommend that you install Identity Manager and the SIF driver on more than one server. Each server with Identity Manager and the SIF Driver can service up to 10 Zones.

If you have multiple Zones, compare what the Zone manages to the containers in your eDirectory tree, to see which containers hold objects that are managed by each Zone.

For additional information about planning your containers for managing students, see [“Creating the Hierarchy of Containers for Students and Staff” on page 21](#).

2.3.2 Planning Replica Placement

This information is based on [“Replicating the Objects that Identity Manager Needs on the Server”](#) in the *Identity Manager 3.5.1 Installation Guide*.

For each Driver object, the server where it runs must hold full master or read/write replicas of the following objects:

- ◆ The User objects that you want this instance of the driver to synchronize.

The driver can't synchronize objects unless a replica of those objects is on the same server as the driver. This might necessitate some changes, for example, aggregating replicas onto a single server if the driver needs a tree-wide view of eDirectory data.

- ◆ The Driver Set object for that server.

You should have one Driver Set object for each server that is running Identity Manager. Inside this Driver Set object is the Driver object that represents the driver that is running on that server. Unless you have specific needs, don't associate more than one server with the same Driver Set object.

- ◆ The Template objects you want the driver to use when creating users, if you choose to use templates.

The driver does not require you to specify templates for use when creating users. But if you want the driver to use templates, the Template objects must be on the server where the driver is running.

- ◆ The Server object for that server.

The Server object is necessary because it allows the driver to generate key pairs for objects. It also is important for Remote Loader authentication.

- ◆ Containers

All containers specified in the configuration must be visible on the server, such as the Incomplete container and the Disabled container.

2.3.3 Examples of Driver and Replica Placement

In this section:

- ◆ [“Example: Placing Drivers and Replicas for One Zone” on page 28](#)
- ◆ [“Example: Placing Drivers and Replicas for Multiple Zones” on page 29](#)

Example: Placing Drivers and Replicas for One Zone

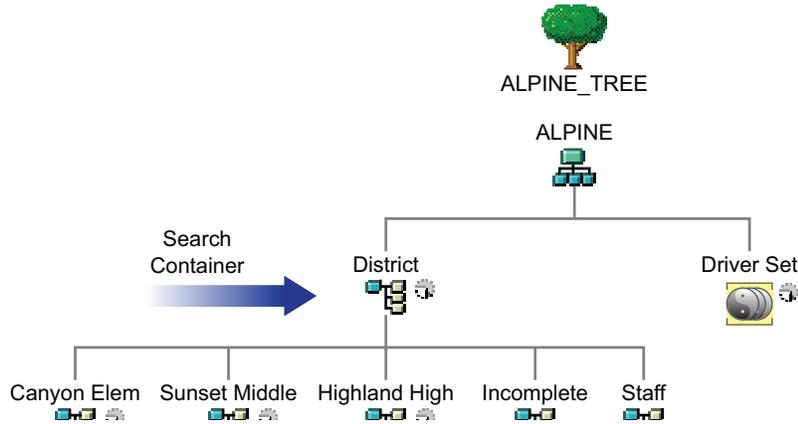
The following figures show an example of how to place the driver and partition replicas based on an example tree, for an environment with only one Zone that manages the whole district.

[Figure 2-6](#) shows how the example tree is partitioned, and [Figure 2-7](#) shows which replicas are needed on the server.

In this example tree, each school container is in a separate partition. The Driver Set object is also in a separate partition.

In this case, you should specify the District container as the search container. (In the driver configuration, you specify which container is the search container, meaning the container and subcontainers that should be searched to find out if there are duplicate User IDs.)

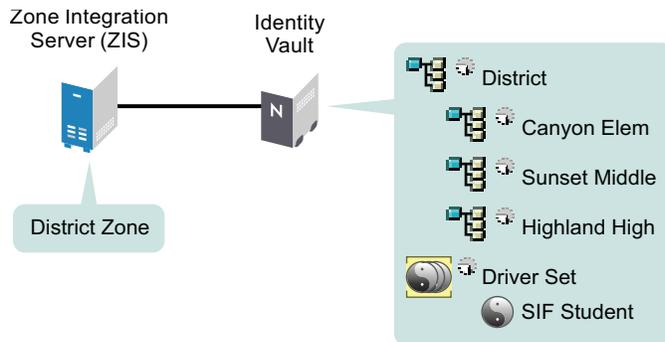
Figure 2-6 Example Tree for One Zone, Showing Partitioning



In this example, a single Identity Vault server is used for the district. Identity Manager and the driver software must be installed on the server so the server can run the driver.

The partitions that are needed on the Identity Vault server with the driver are shown in [Figure 2-7](#).

Figure 2-7 Partitions Containing Users Must Be Replicated on the Same Server as the Driver



Example: Placing Drivers and Replicas for Multiple Zones

This section gives an example of how to place drivers and replicas on servers, based on an example tree, for an environment with multiple Zones. There are three Zones, one for each school.

[Figure 2-8](#) shows how the example tree is partitioned, and [Figure 2-9](#) shows that all replicas must be on the Identity Vault server.

In this example tree, each school container is in a separate partition, as shown in [Figure 2-8](#). The Driver Set object is also in a separate partition.

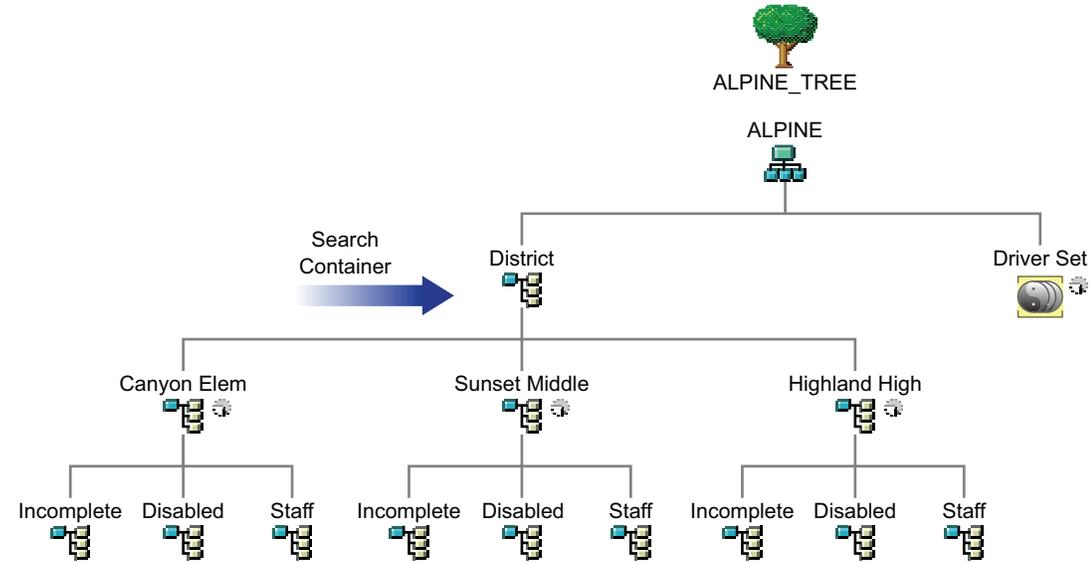
For this example, the District container is the search container. The search container should be high enough in the tree to include all students and staff.

In this example, each school contains its own Incomplete container and Disabled container.

NOTE: This is not required; you could use a single Incomplete container. However, we recommend this implementation for school systems with multiple Zones because it makes it easier to see which Zone needs attention if a student account is “stuck” in the Incomplete container, and because it

reduces the number of partitions you might need on each server. If you use a single Incomplete container for all Zones, you need to keep a master or read/write replica of it on every server.

Figure 2-8 Example Tree for Multiple Zones, Showing Partitioning

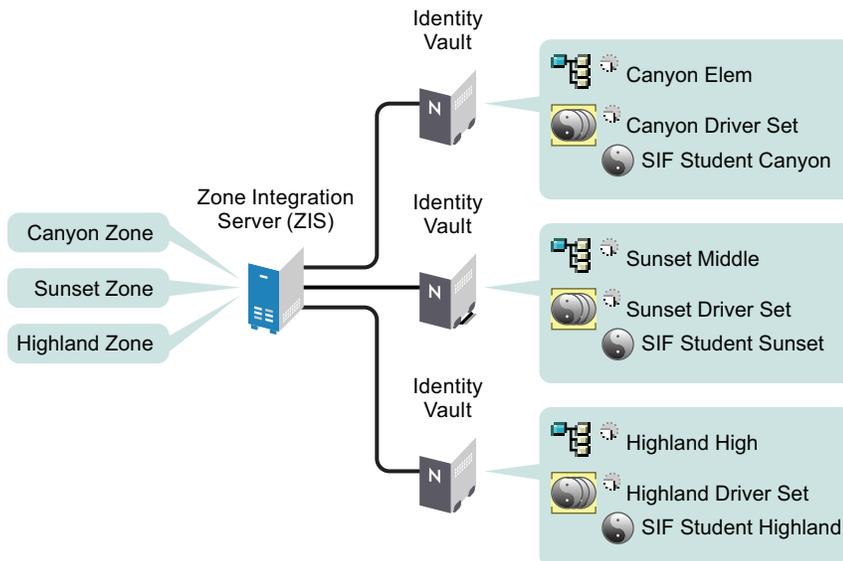


In this example there are three Zones, one per school. Identity Manager and the Identity Manager Driver for SIF are installed on a server that holds replicas of the partitions from each school. One driver is configured to connect to all three Zones.

NOTE: Unlike the example for a single Zone, in this example it's not necessary to replicate the District container on each server in order to get a replica of the Incomplete container, because separate Incomplete containers for each Zone are inside each individual school container.

Figure 2-9 illustrates the driver and the partitions that are replicated on the server.

Figure 2-9 Multiple Zones



2.4 Specifying the Pattern for User IDs

Using Identity Manager for provisioning ensures that eDirectory User IDs follow a consistent pattern, and it eliminates human error in creating User IDs. Consistently following a good pattern reduces support time because you don't need to go in to the Identity Vault to look up User IDs; instead, the student can predict the ID by knowing the pattern (such as last name, first initial, and student ID) and applying it to his or her own information.

You need to plan the pattern you want the driver to follow when creating an eDirectory User ID.

The driver configuration gives you a lot of flexibility in specifying the pattern for creating User IDs. You specify one pattern for student User IDs and a separate pattern for creating staff User IDs. You can create User IDs that are a combination of up to 5 parts.

The following figure shows an example of the options that are provided for User IDs in the driver configuration.

Figure 2-10 Five Parts You Can Use to Create a Pattern for User IDs

The screenshot shows a configuration window titled "Student Configuration". It contains several settings for user ID creation:

Setting	Value
<input type="checkbox"/> Manage student accounts ⓘ	Yes ▾
<input type="checkbox"/> Student user ID format ⓘ	Show ▾
<input type="checkbox"/> Part 1 attribute ⓘ	Last Name ▾
<input type="checkbox"/> Part 1 attribute length ⓘ	All ▾
<input type="checkbox"/> Part 2 attribute ⓘ	First Name ▾
<input type="checkbox"/> Part 2 attribute length ⓘ	1 ▾
<input type="checkbox"/> Part 3 attribute ⓘ	Middle Name ▾
<input type="checkbox"/> Part 3 attribute length ⓘ	1 ▾
<input type="checkbox"/> Part 4 attribute ⓘ	None ▾
<input type="checkbox"/> Part 4 attribute length ⓘ	All ▾
<input type="checkbox"/> Part 5 attribute ⓘ	None ▾
<input type="checkbox"/> Part 5 attribute length ⓘ	All ▾
<input type="checkbox"/> Text for user ID ⓘ	

You can use the following attributes from the student information system:

- ◆ Last Name
- ◆ First Name
- ◆ Middle Name
- ◆ Student ID number

TIP: Formats that include part of the student ID number are more likely to produce unique User IDs.

- ◆ Graduation Year

In addition to using attributes, you also have the option to specify one of the following values:

- ◆ Text

You could incorporate a text string you specify. The text field where you enter the string is the last field shown in [Figure 2-10 on page 31](#).

- ◆ None

Using this option for one of the parts of the User ID indicates that the part has no value and is not being used. For example, if you wanted the User ID to be made up of only three parts, you could specify None as the value for parts 4 and 5.

For each part, you specify a length. The length indicates the number of characters or digits to use from the attribute. For Last Name, First Name, Middle Name, and Text, the left-most characters are used. For Student ID and Graduation Year, the right-most digits are used.

Example

In this figure showing the User ID section of the driver configuration, the administrator has chosen to use four parts for the User ID. Because the fifth part is not needed, it is set to None.

Figure 2-11 Sample User ID Pattern

Student Configuration	
<input type="checkbox"/> Manage student accounts ⓘ	Yes ▾
<input type="checkbox"/> Student user ID format ⓘ	Show ▾
<input type="checkbox"/> Part 1 attribute ⓘ	Text (enter below) ▾
<input type="checkbox"/> Part 1 attribute length ⓘ	All ▾
<input type="checkbox"/> Part 2 attribute ⓘ	Last Name ▾
<input type="checkbox"/> Part 2 attribute length ⓘ	1 ▾
<input type="checkbox"/> Part 3 attribute ⓘ	First Name ▾
<input type="checkbox"/> Part 3 attribute length ⓘ	1 ▾
<input type="checkbox"/> Part 4 attribute ⓘ	Student ID ▾
<input type="checkbox"/> Part 4 attribute length ⓘ	All ▾
<input type="checkbox"/> Part 5 attribute ⓘ	None ▾
<input type="checkbox"/> Part 5 attribute length ⓘ	All ▾
<input type="checkbox"/> Text for user ID ⓘ	<input type="text"/>

[Table 2-3 on page 33](#) represents the same choices, and what the resulting parts of the User ID would be for an example user, Michelle Jones. For this example, the resulting User ID would be “S-JonesM3842.”

Table 2-3 Resulting Parts of the User ID

Part	Value Specified	Length Specified	Attribute Value	Resulting Value
1	Text	All	S-	S-
2	Last Name	All	Jones	Jones
3	First Name	1	Michelle	M
4	Student ID	4	7683842	3842
5	None	All		

2.5 Deciding Whether You Want the Driver to Manage Existing User Accounts

During your planning, decide whether you want the driver to manage existing Identity Vault user accounts. This decision lets you know whether to specify Yes or No for the Manage Existing eDirectory Users field when configuring the driver. This field is on the Global Config Variables page for the driver.

The driver gives you the following options for how to handle existing accounts. You can choose one to start with, and later switch to another option as needed.

- ◆ **Yes.** Use this option if you have one of the following scenarios:
 - ◆ The Identity Vault has no users, and you want to populate the Identity Vault by migrating all students from the student information system into the Identity Vault.
 - ◆ You want to remove all existing users in the Identity Vault and home directories, then populate the Identity Vault by migrating all students from the student information system into the Identity Vault.
 - ◆ You want to manage all existing Identity Vault User objects and new students, without deleting any existing user accounts.
Add the student ID from the student information system to the DirXML-sifSISID attribute for existing accounts in the Identity Vault, so the driver can manage them.
- ◆ **No.** Use this option if you don't want to manage existing Identity Vault users; you want to use the driver only to provision new students.

For more information on these options, the reasons why you might choose them, and how to set them up, see [Section 5.4, “Synchronizing the Identity Vault the First Time,”](#) on page 48.

2.6 Password Synchronization

The SIF driver can synchronize passwords between the Identity Vault and the Zone if the SIF driver and the Zone are using SIF Specification 1.5r1 or later. In order to properly synchronize passwords with the Identity Vault, you must be familiar with “[Password Synchronization across Connected Systems](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*. There are two prompts in the SIF driver's global configuration values (GCVs) that control password sharing with SIF. Set these two prompts to True if you want to synchronize or share passwords.

- ◆ SIF Driver sends user passwords to the Zone

If set to True, the SIF driver sends user passwords in the Identity Vault to the Zone. Passwords are sent as SIF Authorization objects. Other SIF-enabled applications can subscribe to the Zone to receive the passwords.

You would set this parameter to True when other SIF-enabled applications want to use the user's network password. When a Distribution Password is set for a new user or when a Distribution Password is changed in the Identity Vault, the SIF driver sends a SIF Authorization object containing the password to the Zone.

- ◆ SIF Driver accepts user passwords from the Zone

If set to True, the SIF Driver sets user passwords in the Identity Vault to the passwords received from the Zone. The passwords are received as SIF Authorization objects. The passwords are published to the Zone by other SIF-enabled applications.

You would set this parameter to True if you want the network password to be generated by another SIF-enabled application. For example, you have a SIF-enabled application in the Zone that generates a password for each user. When the SIF driver receives the password in a SIF Authorization object, the corresponding user's the Identity Vault password is set to this value.

If this parameter is set to True, we recommend that the SIF driver also be configured to set an initial password for each new user. There might be a delay between the creation of the user account and when the password is received, and it is best to make sure the account is protected by a password at all times.

2.7 Gathering Information for the Driver Configuration

After you create a Driver object with the `SIFAgent-IDM3_5_0-V1.xml` configuration, you need to configure driver settings on the Global Configuration Values page.

As part of your planning, review the table in [Section 5.1, "Creating and Configuring the Driver," on page 41](#), which lists the settings in the driver configuration that you need to complete.

In the previous planning sections, you have already gathered some of the information you need.

Installing the Driver

3

If you are upgrading from a previous version of the Identity Manager Driver for SIF, follow the instructions in [Chapter 4, “Upgrading the Driver,” on page 37](#).

This section gives the prerequisites and the steps for a new installation of the driver.

- ◆ [Section 3.1, “Prerequisites,” on page 35](#)
- ◆ [Section 3.2, “Installing the Identity Manager Driver for SIF,” on page 36](#)
- ◆ [Section 3.4, “Activating the Driver,” on page 36](#)

After completing these tasks, follow the instructions in [Chapter 5, “Deploying the Driver,” on page 41](#) to create and test a Driver object.

3.1 Prerequisites

This section lists the software and hardware requirements you must meet to run the driver.

- ◆ [“Software Requirements” on page 35](#)
- ◆ [“Hardware Considerations” on page 36](#)

3.1.1 Software Requirements

- ❑ Identity Manager with the latest patches and product updates

For patches and product updates for Novell® products, see [Product Updates \(http://support.novell.com/filefinder/5069/index.html\)](http://support.novell.com/filefinder/5069/index.html).

- ❑ The software requirements for Identity Manager listed in the [“Installing Identity Manager”](#) section found in the *Identity Manager 3.5.1 Installation Guide*.
- ❑ A Zone Integration Server and student information environment that complies with SIF standard 1.1 or 1.5r1.

NOTE: If necessary, the driver can be used with a Student Information System that is not SIF-enabled, as described in [“Sending Data from the Identity Vault to SIF” on page 17](#).

- ❑ One of the following server operating systems:
 - ◆ Novell NetWare® 6 or 6.5 with the latest Support Pack (you must obtain and install JVM* 1.4.2 on NetWare)
 - ◆ Windows NT*, 2000, or 2003 with the latest Service Pack

The Identity Manager Driver for SIF supports only English versions of NetWare and Windows.

- ❑ One of the following eDirectory™ versions:
 - ◆ eDirectory 8.7.3 with the latest Support Pack
 - ◆ eDirectory 8.8

3.1.2 Hardware Considerations

- ♦ Identity Manager and the Identity Manager Driver for SIF use approximately 5% of the system's memory and CPU for each Zone they connect to.
- ♦ An Identity Manager-dedicated NetWare system with a 1 GHz processor and 1 GB memory can support connecting to 10 Zones.
- ♦ In production, the driver's poll rate should be set at 900 seconds or higher.

3.2 Installing the Identity Manager Driver for SIF

Installing Identity Manager and the driver software is a simple process.

For a new installation, install Identity Manager and the SIF driver shim on either NetWare or Windows, as described in the “[Installing Identity Manager](#)” section found in the *Identity Manager 3.5.1 Installation Guide*.

If you are upgrading from a previous version of Identity Manager and the SIF driver, see “[Upgrading the Driver](#)” on page 37.

NOTE: Install Identity Manager and the SIF driver shim only once per server, even if a server is running multiple instances of the driver. Multiple instances of the driver are not necessary unless you have more than 10 Zones.

Keep in mind that installing the driver software lets you get the driver up and running, but it does not install the product license. Without the license and activation, the driver will not run after 90 days.

3.3 What's Next

To begin using the driver, create a new Driver object, as explained in [Chapter 5, “Deploying the Driver,”](#) on page 41.

3.4 Activating the Driver

For activation information, refer to “[Activating Novell Identity Manager Products](#)” in the *Identity Manager 3.5.1 Installation Guide*.

Upgrading the Driver

4

If you have been using a previous version of the Identity Manager Driver for SIF, follow these instructions instead of the ones in [Chapter 3, “Installing the Driver,”](#) on page 35.

The Identity Manager 3.5.1 engine is backward compatible with the Identity Manager 3.x, Identity Manager 2.x, and DirXML 1.1a SIF driver shim and driver configuration. We recommend that you upgrade Identity Manager and the SIF driver at the same time. The SIF driver and configuration are tightly coupled. Both must be used together.

If you are upgrading from Identity Manager 3.5.0 to Identity Manager 3.5.1, the following information does not apply.

Identity Manager 3.5.0 contains a new architecture for how policies reference one another. To take advantage of this new architecture, the driver configuration file provided for SIF must be upgraded. For more information on the new architecture, see [“Upgrading Identity Manager Policies”](#) in the [Understanding Policies for Identity Manager 3.5.1](#). You can upgrade the driver in Designer or iManager.

- ♦ [Section 4.1, “Upgrading the Driver in Designer,”](#) on page 37
- ♦ [Section 4.2, “Upgrading the Driver in iManager,”](#) on page 40

4.1 Upgrading the Driver in Designer

- 1 Make sure you have updated your driver with all the patches for the version you are currently running.

We recommend this step for all drivers, to help minimize upgrade issues.

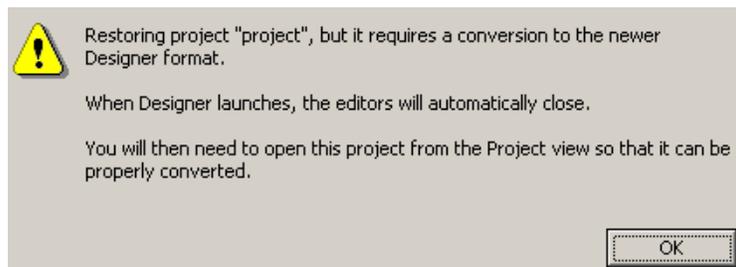
- 2 Back up the driver.

See [Chapter 11, “Backing Up the Driver,”](#) on page 111 for instructions on how to back up the driver.

- 3 Install Designer version 2.1 or above, then launch Designer.

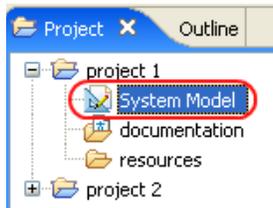
If you had a project open in Designer when you upgraded Designer, proceed to [Step 4](#). If you didn't have a project open in Designer when you upgraded Designer, skip to [Step 5](#).

- 4 If you had a project open when upgrading Designer, the following warning message is displayed. Read the warning message, then click *OK*.

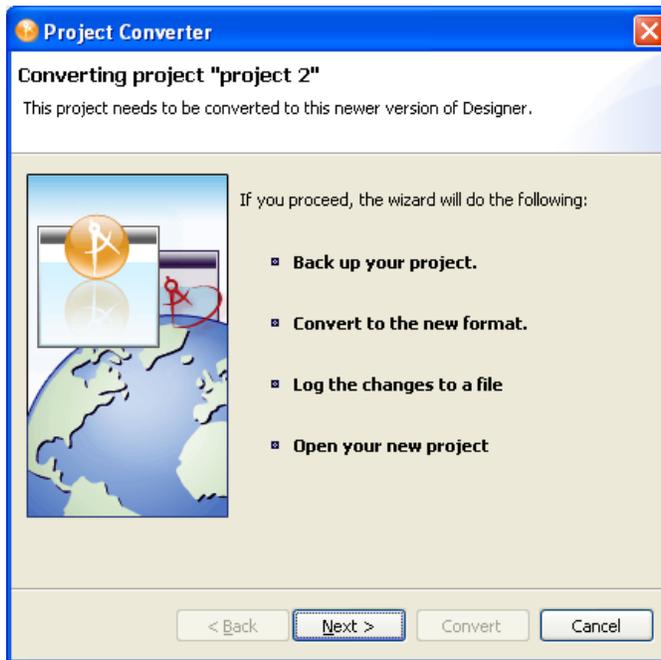


Designer closes the project to preform the upgrade.

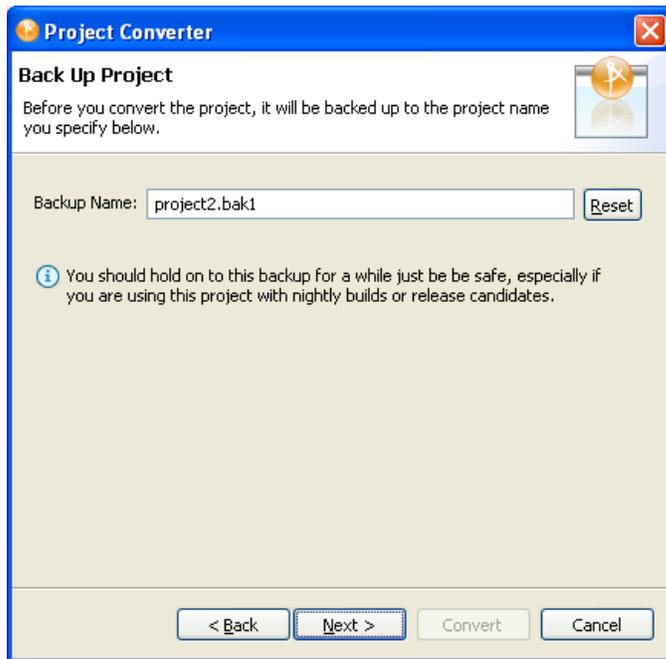
5 In the Project view, double-click *System Model* to open and convert the project.



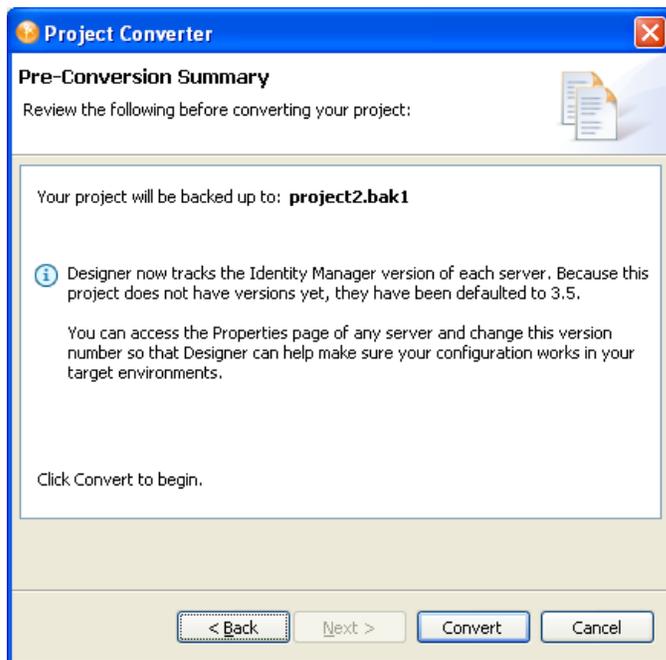
6 Read the Project Converter message explaining that the project is backed up, converted to the new format, changes logged to a file, and the new project is opened, then click *Next*.



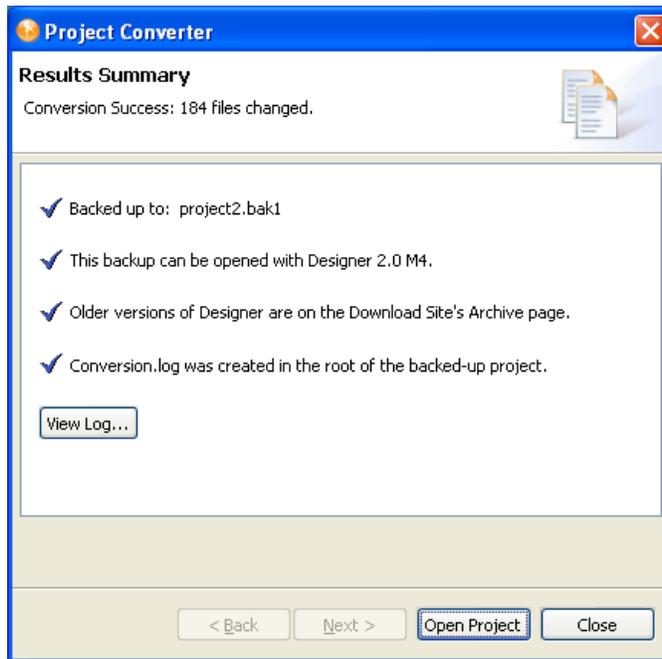
7 Specify the name of the backup project name, then click *Next*.



8 Read the project conversion summary, then click *Convert*.



- 9 Read the project conversion result summary, then click *Open Project*.



If you want to view the log file that is generated, click *View Log*.

4.2 Upgrading the Driver in iManager

- 1 Make sure you have updated your driver with all the patches for the version you are currently running.

We recommend this step for all drivers, to help minimize upgrade issues.

- 2 Back up the driver. See [Chapter 11, “Backing Up the Driver,” on page 111](#) for instruction on how to back up the driver.
- 3 Verify that Identity Manager 3.5.1 has been installed and you have the current plug-ins installed, then launch iManager.
- 4 Click *Identity Manager > Identity Manager Overview*.
- 5 Click *Search* to find the Driver Set object, then click the driver you want to upgrade.
- 6 Read the message that is displayed, then click *OK*.



- 7 If there is more than one driver to upgrade, repeat [Step 2](#) through [Step 6](#).

Deploying the Driver

5

You can use the Identity Manager Driver for SIF to manage Identity Vault accounts for students, staff, and faculty. This automation allows new users to have network access and a home directory right away, without manual intervention by the Identity Vault administrator. When changes occur to student, staff, and faculty information, the Identity Vault is automatically updated.

This section contains the information you need to set up the driver object and configure it, after it is installed (as explained in [Chapter 3, “Installing the Driver,”](#) on page 35).

- ◆ [Section 5.1, “Creating and Configuring the Driver,”](#) on page 41
- ◆ [Section 5.2, “Preparing the ZIS and the Student Information System,”](#) on page 45
- ◆ [Section 5.3, “Starting and Testing the Driver,”](#) on page 46

After you complete these tasks, decide how you want to synchronize student data from the Student Information System into the Identity Vault.

- ◆ [Section 5.4, “Synchronizing the Identity Vault the First Time,”](#) on page 48
- ◆ [Section 5.5, “Synchronizing the Identity Vault Each School Year,”](#) on page 52

5.1 Creating and Configuring the Driver

The Identity Manager Driver for SIF comes with a driver configuration file named `SIFAgent-
IDM3_5_0-v1.xml`.

You use a wizard to create a new Driver object based on this configuration file. When you import the configuration file to create or upgrade a driver object, only a few prompts are presented. Most of the driver configuration is done after you import, on the Global Configuration Values page for the driver.

- ◆ [Section 5.1.1, “Prerequisites,”](#) on page 41
- ◆ [Section 5.1.2, “Importing the Driver Configuration File in Designer,”](#) on page 42
- ◆ [Section 5.1.3, “Importing the Driver Configuration File in iManager,”](#) on page 42
- ◆ [Section 5.1.4, “Configuration Parameters,”](#) on page 43
- ◆ [Section 5.1.5, “Post-Configuration Tasks,”](#) on page 44

5.1.1 Prerequisites

- ❑ You have installed Identity Manager and the Identity Manager Driver for SIF on the Identity Vault server, and installed the Identity Manager plug-ins and the driver configuration files on the iManager Web server, as explained in [Section 3.2, “Installing the Identity Manager Driver for SIF,”](#) on page 36.
- ❑ You restarted NetWare® (for a NetWare server) or eDirectory™ (for a Windows server) after installing the driver.

- ❑ You have followed the instructions in [“Planning” on page 19](#) to complete the following tasks:
 - ◆ Identify or create the Identity Vault objects you need: the necessary containers for your students and staff, the Incomplete and Disabled containers, and the Template objects. In the driver configuration, you need to specify the DN for these objects.
 - ◆ Gather the other information you need for setting up the driver configuration, as explained in [Section 2.7, “Gathering Information for the Driver Configuration,” on page 34](#).

5.1.2 Importing the Driver Configuration File in Designer

Designer allows you to import the basic driver configuration file for SIF. This file creates and configures the objects and policies needed to make the driver work properly. The following instructions explain how to create the driver and import the driver’s configuration.

There are many different ways of importing the driver configuration file. This procedure only documents one way.

- 1 Open a project in Designer and in the modeler, right-click the Driver Set object and select *New > Driver*.
- 2 Browse to and select the SIF driver from the drop-down list, then click *Run*.
- 3 Configure the driver by filling in the fields. Specify information specific to your environment. For information on the settings, see [Table 5-1 on page 43](#) for more information.
- 4 After specifying parameters, click *Finish* to import the driver.
- 5 After the driver is imported, customize and test the driver.
- 6 After the driver is fully tested, deploy the driver into the Identity Vault. See [“Deploying a Driver to an Identity Vault”](#) in the *Designer 2.1 for Identity Manager 3.5.1*.

5.1.3 Importing the Driver Configuration File in iManager

The SIF preconfiguration file is an example configuration file. You installed this file when you installed the Identity Manager Web components on an iManager server. Think of the preconfiguration file as a template that you import and customize or configure for your environment.

- 1 In iManager, select *Identity Manager Utilities > Import Configurations*.
- 2 Select a driver set, then click *Next*.

Where do you want to place the new drivers?

In an existing driver set
  

In a new driver set

If you place this driver in a new driver set, you must specify a driver set name, context, and associated server.

- 3 Select how you want the driver configurations sorted:
 - ◆ All configurations

- ◆ Identity Manager 3.5 configurations
- ◆ Identity Manager 3.0 configurations
- ◆ Configurations not associated with an IDM version

4 Select the *SIF* driver, then click *Next*.



5 Configure the driver by filling in the configuration parameters, then click *Next*. For information on the settings, see [Table 5-1 on page 43](#).

6 Define security equivalences, using a user object that has the rights that the driver needs to have on the server

The tendency is to use the Admin user object for this task. However, you might want to create a DriversUser (for example) and assign security equivalence to that user. Whatever rights that the driver needs to have on the server, the DriversUser object must have the same security rights.

7 Identify all objects that represent administrative roles and exclude them from replication.

Exclude the security-equivalence object (for example, DriversUser) that you specified in [Step 6](#). If you delete the security-equivalence object, you have removed the rights from the driver. Therefore, the driver can't make changes to Identity Manager.

8 Click *Finish*.

5.1.4 Configuration Parameters

The following table explains the parameters you must provide during initial driver configuration.

Table 5-1 Configuration Parameters for the SIF Driver

Field Name	Description
<i>Driver name</i>	Specify the name you want to use for the driver object in the Identity Vault.
<i>Sif Agent Name</i>	Specify the name this driver uses to register as a SIF Agent with the Zone Integration Server (ZIS). The driver must have a Zone-unique, case-sensitive name. We recommend that you use the default name, Novell Identity Manager. You need to coordinate with the ZIS administrator to make sure that the same name is used when configuring the ZIS, as described in "Configuring the ZIS to Recognize the Driver" on page 45 .
<i>Sif Specification version</i>	Specify the SIF Specification version you want this driver to use, either SIF Specification 1.1, or SIF Specification 1.5r1.

Field Name	Description
<i>Manage preexisting eDirectory users</i>	<p>The SIF Driver can match students and staff in the Student Information System (SIS) with preexisting Identity Vault users only if the eDirectory user attribute DirXML-sifSISID contains the student's or staff's ID number.</p> <p>Specify <i>Yes</i> if one of the following is true:</p> <ul style="list-style-type: none"> ◆ You want to manage preexisting Identity Vault users, and the DirXML-sifSISID is set on all users. ◆ No users currently exist in the Identity Vault. <p>Otherwise, specify <i>No</i>.</p> <p>If <i>Yes</i> is specified, the <i>Migrate into Identity Vault</i> command can be used to add or update all SIF users into the Identity Vault.</p> <p>If <i>No</i> is specified, the <i>Migrate into Identity Vault</i> command is ignored to prevent duplicate users from being created in the Identity Vault.</p> <p>This field does not apply to users added to the Identity Vault by this driver. Identity Manager can always match these Identity Vault users with Student Information System users, and these Identity Vault users are always kept current with changes from the Student Information System.</p> <p>For more information on how to make this decision, see Section 5.4, "Synchronizing the Identity Vault the First Time," on page 48.</p>
<i>Driver is Local/Remote</i>	<p>Specify whether to run the driver locally or using Remote Loader.</p> <p>If you specify <i>Remote</i>, then click <i>Next</i>, another page presents a few more items for you to specify regarding Remote Loader configuration.</p> <p>For information about running the driver remotely, see "Deciding Whether to Use the Remote Loader" in the <i>Novell Identity Manager 3.5.1 Administration Guide</i>.</p>

5.1.5 Post-Configuration Tasks

- 1** After you create the Driver object, configure settings such as the containers to use for students and staff.
 - 1a** In iManager, click *Identity Manager* > *Identity Manager Overview*. Search for and select the driver set.
 - 1b** Browse to and click the driver icon, then in the next page, click the driver icon again.
- 2** Click the *Global Config Values* tab, then specify the desired settings. Some of them were specified when creating the driver object; for those items you can simply review the settings to make sure they are correct. See [Section B.2, "Global Configuration Values,"](#) on page 134 for a detailed list of all of the fields.
- 3** Follow the instructions in [Section 5.2, "Preparing the ZIS and the Student Information System,"](#) on page 45 to configure the ZIS to recognize the driver as a SIF Agent.

5.2 Preparing the ZIS and the Student Information System

The Zone Integration Server (ZIS) must be configured to recognize the driver as a SIF Agent, just as you would do for other SIF Agents. The driver works with but does not change data from the Student Information System, but you can optimize the data if desired.

You can complete these steps either before or after you create a Driver object in the Identity Vault for the Identity Manager Driver for SIF, but you must complete them for the driver to receive information.

- ♦ [Section 5.2.1, “Configuring the ZIS to Recognize the Driver,” on page 45](#)
- ♦ [Section 5.2.2, “Optimizing Data in the Student Information System,” on page 46](#)

5.2.1 Configuring the ZIS to Recognize the Driver

The Zone Integration Server (ZIS) must be configured to recognize the Identity Manager Driver for SIF as a SIF Agent. The driver can't receive any information about students or staff until this has been done.

The ZIS administrator should configure the ZIS to recognize the driver by doing the following tasks. Refer to your ZIS documentation for instructions.

1. Specify the SIF Agent name for the driver, such as Novell Identity Manager.

This is the name the driver uses to register with the ZIS. It must be the same name you specify in the *SIF Agent Name* field when you create the Driver object, as described in [Section 5.1, “Creating and Configuring the Driver,” on page 41](#).

The default name is Novell Identity Manager. If you want to use a different name, keep in mind that it must be unique within each Zone, and it is case sensitive.

2. Specify the SIF objects the driver has access to:

- ♦ StudentPersonal
- ♦ StudentSchoolEnrollment
- ♦ SchoolInfo
- ♦ StaffPersonal
- ♦ EmployeePersonal
- ♦ Authorization

For these SIF objects, the driver should have Add, Change, Delete, Subscribe, and Request rights.

If the driver is also the SIF Provider, it should have Publish and Response rights.

3. Specify that the driver is a pull agent.
4. Give the driver permission to request the ZoneStatus object.
5. Set up security, if desired. This is explained in [Section 6.2, “Setting Up Security,” on page 57](#).

5.2.2 Optimizing Data in the Student Information System

The Identity Manager Driver for SIF is designed to work as a SIF Agent without requiring any change in the Student Information System, but there is one aspect of the Student Information System that can be optimized.

According to the SIF implementation specification, the StudentPersonal object provides the student's name, the StudentSchoolEnrollment object provides the grade, and the SchoolInfo object provides the school code. However, some Student Information Systems can be configured to also provide school and grade information with the StudentPersonal object, in the OtherID attribute.

Student placement is done most efficiently when the Student Information System provides the school and grade to the driver using the OtherId attribute of the StudentPersonal object. If possible, have the Student Information System administrator configure it this way.

No corresponding change to the driver configuration is necessary. These values are handled in the Input Transformation, which is configured to accept the school and grade information from either the StudentSchoolEnrollment object or the OtherID attribute.

5.3 Starting and Testing the Driver

After creating the Driver object and completing the rules for placing groups of students, you can start the driver and test it.

The default polling rate on a new Driver object is 900 seconds. This is appropriate for a production environment, but you should make it shorter for testing purposes.

Prerequisites

- ❑ You have configured the ZIS to recognize the driver as a SIF Agent, as described in [“Configuring the ZIS to Recognize the Driver” on page 45](#).

If you don't complete this step, you will get errors in the status log when you start the driver.

- ❑ If you have existing users in the Identity Vault, and the *Manage preexisting eDirectory users* parameter for the driver is set to *Yes*, review your options before starting the driver to avoid duplicate users being created. See [Section 5.4, “Synchronizing the Identity Vault the First Time,” on page 48](#). If you select *Yes* for *Manage preexisting eDirectory users*, you should follow the steps given and fill in the DirXML-sifSISID attribute for existing user objects before starting the driver.

Procedure

- 1** For testing purposes, set the polling rate for the Driver object to 15 seconds.
 - 1a** In iManager, click *Identity Manager > Identity Manager Overview*. Search for the driver set.
 - 1b** In the driver set, click the icon for the driver. On the Identity Manager Driver Overview page that appears, click the driver icon again.
 - 1c** Click the *Identity Manager* tab, then click *Driver Configuration*. On the Driver Configuration page, find the *Publisher Settings* and *Poll rate in seconds*.

1d Change the poll rate to 15 seconds, then click *OK*.

Modify Object: SIF.DriverSet.novell

Identity Manager | **Server Variables** | **General**

Driver Configuration | [Global Config Values](#) | [Named Passwords](#) | [Engine Control Values](#) | [Linkage](#) | [Log Level](#) | [Driver Image](#) | [Security Equals](#) | [Filter](#) | [Edit Filter XML](#) | [Misc](#) | [Excluded Users](#) | [Driver Manifest](#) | [Associations](#)

Driver keystore name

Driver certificate password

Authentication level

Encryption level

Subscriber Settings

Publisher Settings

Poll rate in seconds

StudentSchoolEnrollment TimeFrame

- 2 Set the startup option for the driver. On the same Driver Configuration page, scroll to *Startup Option*. Select how you want the driver to be started, then click *OK*.
- 3 Start the driver. On the Identity Manager Driver Overview page, click the icon in the upper right corner of the driver icon, then click *Start Driver*.
- 4 Check for errors.

If you see errors you need to fix, you might want to clear the log before you make changes so you can see which errors are new.

See [Section 10.1, “Viewing Status Messages for the Identity Manager Driver for SIF,” on page 87](#) and [Section 10.2, “Error Messages,” on page 88](#).

Disregard the error message `No object name provided`. It does not indicate a problem.

- 5 After you test the driver, set the polling rate to a longer period appropriate for your environment, such as 900 seconds.

Modify Object: SIF.DriverSet.novell

Identity Manager | **Server Variables** | **General**

Driver Configuration | [Global Config Values](#) | [Named Passwords](#) | [Engine Control Values](#) | [Linkage](#) | [Log Level](#) | [Driver Image](#) | [Security Equals](#) | [Filter](#) | [Edit Filter XML](#) | [Misc](#) | [Excluded Users](#) | [Driver Manifest](#) | [Associations](#)

Driver keystore name

Driver certificate password

Authentication level

Encryption level

Subscriber Settings

Publisher Settings

Poll rate in seconds

StudentSchoolEnrollment TimeFrame

5.4 Synchronizing the Identity Vault the First Time

After you have imported the driver and tested it, you need to decide how to handle synchronizing the Identity Vault user accounts with user data in the Student Information System the first time.

When you configure the driver, you specify either *Yes* or *No* for the *Manage preexisting eDirectory users* field as described in [Section 5.1, “Creating and Configuring the Driver,” on page 41](#). This setting determines whether the driver tries to synchronize existing users in the Identity Vault, or ignores them and only manages new students and staff. You specify this setting on the Global ConfigValues page for the driver.

The Identity Manager Driver for SIF gives you three options for synchronizing existing accounts. Regardless of the option you choose, the driver provisions and manages any new accounts entered into the Student Information System in the future.

This section describes the three options, the reasons why you might choose one, and how to set them up.

- ◆ [“Option 1: Populate the Identity Vault Using Migrate into Identity Vault” on page 48](#)
- ◆ [“Option 2: Manage Existing Identity Vault User Accounts” on page 49](#)
- ◆ [“Option 3: Don’t Manage Existing Identity Vault User Accounts” on page 50](#)

To help you set up these options, this section also provides instructions for the following task:

- ◆ [“Using Migrate into Identity Vault to Populate or Update the Identity Vault” on page 51](#)

5.4.1 Option 1: Populate the Identity Vault Using Migrate into Identity Vault

For this option, you remove all existing accounts and home directories, and re-create them “from scratch,” using the *Migrate into Identity Vault* option to populate the Identity Vault.

Why Would You Use This Option?

- ◆ You want the driver to manage all accounts.
- ◆ You have decided you want to “start from scratch” by removing existing users from the Identity Vault, or you have not yet put any users into the Identity Vault.
- ◆ You don’t need to preserve the files that are currently in the home directories.

For example, if you were implementing the driver before the beginning of the school year, and you didn’t need to keep home directories from the previous year, you could get a fresh start in the Identity Vault using this option.

How To Set It Up

- 1 Remove existing user accounts (User objects) from the Identity Vault.
- 2 Remove the home directories from the server.

IMPORTANT: If existing home directories are not deleted along with existing user accounts, the users who are migrated won’t have a home directory. Identity Manager must create the

home directory at the same time it creates a user. It can't grant the newly created user rights to an existing home directory; instead, it gives an error.

If you had existing user accounts with home directories and you didn't delete them before using Migrate into Identity Vault, you need to delete them and repeat the migration.

3 Set *Manage preexisting eDirectory users* to *Yes*.

You set this on the Global Config Values page for the driver.

4 Populate the Identity Vault by using *Migrate into Identity Vault* to request all user data from the Student Information System.

See [“Using Migrate into Identity Vault to Populate or Update the Identity Vault” on page 51](#).

You should use *Migrate into Identity Vault* when demand for the server is low, such as on a weekend. If you have more than one Zone configured, we recommend you perform the migration one Zone at a time. The migration can take approximately 20 seconds per user and places a load on the server.

Identity Manager creates all students and staff in the Student Information System as User objects in the Identity Vault. As they are created, the objects are automatically associated with the ID in the Student Information System, so Identity Manager can manage them.

5.4.2 Option 2: Manage Existing Identity Vault User Accounts

For this option, you leave existing accounts in the Identity Vault. You manually put the student or staff ID from the Student Information System into the DirXML-sifSISID attribute of each existing Identity Vault user object, so the driver can match it with the corresponding individual in the Student Information System. After you put in the Student Information System ID, the driver can manage existing user accounts, so any new changes to those individuals in the Student Information System are reflected in the Identity Vault.

If you want current data from the Student Information System to be synchronized to the Identity Vault (for example, because you are concerned that existing user account data doesn't currently match the Student Information System), use *Migrate into Identity Vault* after you add the Student Information System ID to the DirXML-sifSISID attribute.

If you choose this option, you need to fill in the DirXML-sifSISID immediately. If you don't, and a change comes through for an account, the driver cannot find the matching User object and a duplicate is created.

Why Would You Use This Option?

- ◆ You already have User objects in the Identity Vault, and you don't want to delete them, but you do want the driver to manage them.
- ◆ You want to preserve the files that are currently in the home directories.

For example, if you were implementing the driver during the school year, and you wanted to keep home directories intact and minimize the risk of any problems with accounts, you might decide to keep existing accounts in place. With this option, you could keep accounts that are currently working and take the time to manually add the Student Information System ID to each of them, so the driver can recognize and manage them.

How To Set It Up

- 1 For all existing Identity Vault User objects, manually enter the Student Information System ID into the DirXML-sifSISID attribute. Make sure it is correct.

This is a one-time effort.

IMPORTANT: If the ID is not entered or is not correct, *Migrate into Identity Vault* creates duplicate User objects instead of updating existing User objects. There is no command to “undo” *Migrate into Identity Vault*, so you would need to remove the duplicates manually.

- 2 Set *Manage preexisting eDirectory users* to *Yes*.
You set this on the Global Config Values page for the driver.
- 3 (Optional) If you want to synchronize existing accounts in the Identity Vault with all data from the Student Information System, you can use *Migrate into Identity Vault*.

See “[Using Migrate into Identity Vault to Populate or Update the Identity Vault](#)” on page 51.

If you are only concerned about synchronizing new changes that occur, you don’t need to do this step.

You should use *Migrate into Identity Vault* when demand for the server is low, such as on a weekend. If you have more than one Zone configured, we recommend you perform the migration one Zone at a time. The migration can take approximately 20 seconds per user and places a load on the server.

After following these steps, Identity Manager can manage existing Identity Vault user accounts because you have manually made the association with the Student Information System ID. New users are also managed because Identity Manager automatically creates the association when it creates a new user.

5.4.3 Option 3: Don’t Manage Existing Identity Vault User Accounts

For this option, you set the driver to ignore existing accounts and manage only new students who are entered in the Student Information System. You don’t use *Migrate into Identity Vault* as part of setting up this option.

Existing student accounts in the Identity Vault are not affected by the driver; changes that occur for these accounts in the Student Information System are ignored by the driver.

New students added to the Student Information System after the driver is started are provisioned in the Identity Vault and are thereafter managed by the driver. The Identity Vault users created by the driver are always kept current with changes from the Student Information System.

Don’t run *Migrate into Identity Vault* if you are using this option.

Why Would You Use This Option?

- ◆ You don’t want the driver to affect existing student accounts.
- ◆ You only want the driver to provision and manage new students who are added to the Student Information System.
- ◆ You need to preserve the files that are currently in home directories.

For example, you could use this option if you were deploying the driver during the middle of the school year, and you wanted to eliminate risk to any existing accounts. Perhaps you don't have time to manually create the association with the Student Information System for each existing object. With this option, you can keep existing accounts as they are but take advantage of the driver's functionality to provision any new students.

How To Set It Up

- 1 Set *Manage preexisting eDirectory users* to *No*.

You set this on the Global Config Values page.

- 2 Don't use *Migrate into Identity Vault*.

If *Manage preexisting eDirectory users* is set to *No*, *Migrate into Identity Vault* is ignored.

Should I use Migrate into Identity Vault or Synchronize Options?

The *Migrate into Identity Vault* option requests all student and staff records from the Student Information System and tries to match each record with an user account in the Identity Vault. If a match is found, the Identity Vault user account is updated with the information from the Student Information System. If a match is not found, a new user account is created in the Identity Vault.

For each user account in the Identity Vault, the *Synchronize* option queries the Student Information System for its attribute values and updates the Identity Vault user account with the received information.

Migrate into Identity Vault is more efficient. Only one query is sent to the SIS. *Synchronize* sends a separate query for each user account in the Identity Vault. *Migrate into Identity Vault* updates existing the Identity Vault user accounts and creates new Identity Vault user accounts. *Synchronize* only updates existing Identity Vault user accounts.

5.4.4 Using Migrate into Identity Vault to Populate or Update the Identity Vault

Migrate into Identity Vault lets you request records for all individuals from the Student Information System. If a matching user is not found in the Identity Vault, a new account is created. If an account already exists in the Identity Vault for the student, and the DirXML-sifSISID attribute contains the correct Student Information System ID, the driver updates the account to match the information in the Student Information System.

You can run *Migrate into Identity Vault* at the start of a school year to initially populate the Identity Vault. You can also run it any time you want to ensure the Identity Vault is synchronized with the Student Information System.

You should use this option only if the following two conditions are met:

- ♦ If you have any users in the Identity Vault, they must either have been created by the driver (which means they have an Identity Manager association created by the driver), or they must have the correct ID manually entered in the DirXML-sifSISID attribute.

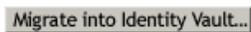
This allows the driver to match an individual in the Student Information System with an existing User object.

IMPORTANT: If this condition is not met, *Migrate into Identity Vault* creates duplicate User objects instead of updating existing User objects. There is no command to “undo” *Migrate into Identity Vault*, so you would need to remove the duplicates manually.

- ♦ The Driver object’s *Manage preexisting eDirectory users* parameter is set to *Yes*.
If it is set to *No*, *Migrate into Identity Vault* is ignored.

You should use *Migrate into Identity Vault* when demand for the server is low, such as on a weekend. If you have more than one *Zone* configured, we recommend you perform the migration one *Zone* at a time. The migration can take approximately 20 seconds per user and places a load on the server.

- 1 In iManager, click *Identity Manager > Identity Manager Overview*, and search for the driver set.
- 2 Click the driver icon for the driver.
- 3 If the driver is not running, click the icon in the upper-right corner of the driver icon, then select *Start Driver*.
- 4 Click the *Migrate into Identity Vault* button.



- 5 In the *Migrate Data into the Identity Vault* dialog box, click *Edit List*.
The *Edit Migration Criteria* dialog box appears.
- 6 In the left column, select *User*, then click *OK*.
- 7 On the *Migrate Data into the Identity Vault* dialog box, click *OK*.
The driver continues to run the migration, even if you close iManager.

5.5 Synchronizing the Identity Vault Each School Year

You can synchronize student data in the Identity Vault so that it matches the Student Information System at the beginning of the school year. To accomplish this, you have options similar to the ones outlined in [Section 5.4, “Synchronizing the Identity Vault the First Time,” on page 48](#). Consult with your Student Information System administrator; the way your application works might influence your choice, and your application vendor might have a recommended approach.

This section describes the options and issues you should consider.

- ♦ [“New Year Options for Students in School or Grade Containers” on page 52](#)
- ♦ [“New Year Tasks for Students in Graduation Year Containers” on page 55](#)

5.5.1 New Year Options for Students in School or Grade Containers

In this section:

- ♦ [“Option 1 for a New Year: Repopulate the Identity Vault Using Migrate into Identity Vault” on page 53](#)

- ◆ [“Option 2 for a New Year: Update Existing Accounts Using Migrate into Identity Vault” on page 53](#)
- ◆ [“Option 3 for a New Year: Maintain Existing Accounts All Summer” on page 54](#)

Option 1 for a New Year: Repopulate the Identity Vault Using Migrate into Identity Vault

For this option, you delete existing student accounts and home directories in the Identity Vault and use *Migrate into Identity Vault* to repopulate the Identity Vault “from scratch” at the beginning of the year.

Why Would You Use This Option?

- ◆ Your Student Information System application recommends this kind of approach.
We recommend this approach; however, you should consult with the administrator of the your Student Information System.
- ◆ You don’t need to preserve the files that are currently in the home directories.
- ◆ You have students who are moving to new schools, their home directories need to be moved to a new server, and you don’t want to move them manually.
- ◆ You have specified different eDirectory templates for different containers or schools, and you need accounts to be updated to match a new eDirectory template when users move to a new container or school.

How to Set It Up

- 1 Stop the driver at the beginning of the summer.
- 2 Remove the eDirectory accounts and the home directories.

IMPORTANT: If existing home directories are not deleted along with existing user accounts, the users who are migrated won’t have a home directory. Identity Manager must create the home directory at the same time it creates a user. It can’t grant the newly created user rights to an existing home directory; instead, it gives an error.

If you had existing user accounts with home directories and you didn’t delete the home directories before using *Migrate into Identity Vault*, you need to delete them and repeat the migration.

- 3 At the end of the summer when the Student Information System is up-to-date for the next school year, start the driver again and use *Migrate into Identity Vault* to repopulate the Identity Vault.

See [“Using Migrate into Identity Vault to Populate or Update the Identity Vault” on page 51](#).

You should use *Migrate into Identity Vault* when demand for the server is low, such as on a weekend. If you have more than one Zone configured, we recommend you perform the migration one Zone at a time. The migration can take approximately 20 seconds per user and places a load on the server.

Option 2 for a New Year: Update Existing Accounts Using Migrate into Identity Vault

For this option, you keep your existing Identity Vault student accounts and update them all at once using *Migrate into Identity Vault* at the beginning of the year.

This option involves stopping the driver at the beginning of summer. At the end of the summer when the Student Information System data is ready for the new year, you start the driver again and use *Migrate into Identity Vault* to update existing accounts all at once.

To use this option, the driver must be able to associate existing user accounts with a record in the Student Information System. Therefore, all existing user accounts must have either the Student Information System ID entered in the DirXML-sifSISID attribute (you need to do this manually for users who were originally created by hand), or an Identity Manager association created (the driver does this for user accounts it creates).

IMPORTANT: If the ID is not entered or is not correct, *Migrate into Identity Vault* creates duplicate User objects instead of updating existing User objects. There is no command to “undo” *Migrate into Identity Vault*, so you would need to remove the duplicates manually.

Using *Migrate into Identity Vault* moves student accounts to new containers if necessary. However, the driver does not move home directories, so if the student account moves to a container on a new server and you want the home directory to be on the same server, you must move the home directories manually or with third-party software.

Why Would You Use This Option?

- ◆ Your Student Information System application recommends this kind of approach.
- ◆ You don’t need student accounts to be re-created based on a new eDirectory template when they move to a new grade or school.
- ◆ You want to preserve the files in the home directories.

How To Set It Up

- 1 Stop the driver at the beginning of the summer.
- 2 When the Student Information System is up-to-date for the next school year, start the driver again and use *Migrate into Identity Vault* to synchronize the Identity Vault.

See “[Using Migrate into Identity Vault to Populate or Update the Identity Vault](#)” on page 51.

You should use *Migrate into Identity Vault* when demand for the server is low, such as on a weekend. If you have more than one Zone configured, we recommend you perform the migration one Zone at a time. The migration can take approximately 20 seconds per user and places a load on the server.

- 3 Move home directories as necessary, such as for students who are moving to a new school and whose accounts need to be on a different server.

You can do this manually. Third-party software is also available to move home directories.

Option 3 for a New Year: Maintain Existing Accounts All Summer

For this option, you keep your existing Identity Vault student accounts, and keep them up-to-date by receiving changes as they are entered in the Student Information System over the summer.

You leave the driver running all summer to receive incremental changes from the Student Information System.

The driver moves students from one container to another as their schools and grades are updated in the Student Information System. However, the driver does not move home directories, so if the

student account moves to a container on a new server and you want the home directory to be on the same server, you must move the home directories manually or with third-party software.

Migrate into Identity Vault is not required for this option.

Why Would You Use This Option?

- ◆ Your Student Information System application recommends this kind of approach.
- ◆ You want to preserve the files in the home directories.
- ◆ You don't need student accounts to be re-created based on a new eDirectory template when they move to a new grade or school.
- ◆ You need student accounts to be up-to-date all summer, such as for year-round schedules or summer school.

How to Set It Up

- 1 Keep the driver running all summer.
- 2 Move home directories as necessary, such as for students who are moving to a new school and whose accounts need to be on a different server.

You can do this manually. Third-party software is also available to move home directories.

5.5.2 New Year Tasks for Students in Graduation Year Containers

If you put students in graduation year containers (see the example in [Figure 2-2 on page 22](#)), you need to update your tree structure each year to accommodate groups of students moving to new schools.

- 1 Manually create new graduation year containers under the school containers they are moving to.
- 2 In the Global Configuration Values for the driver, update the container DN and template assignments for all groups of students that are moving to a new school.
See [Section 5.1, "Creating and Configuring the Driver," on page 41](#).
- 3 Make sure the students are placed in the new container. You have three options for doing this, based on how you want to handle student accounts for each new school year:
 - ◆ ["Option 1 for a New Year: Repopulate the Identity Vault Using Migrate into Identity Vault" on page 53](#)
 - ◆ ["Option 2 for a New Year: Update Existing Accounts Using Migrate into Identity Vault" on page 53](#)
 - ◆ ["Option 3 for a New Year: Maintain Existing Accounts All Summer" on page 54](#)For this option, if you create the new graduation year containers and update the Global Config Values for the driver *after* the school changes for students have been made in the Student Information System, then you still need to move students manually to the correct container.
- 4 After you have tested the change, and all the students have been moved to the new graduation year containers, delete the old containers.
- 5 Move home directories as necessary.

You can do this manually. Third-party software is also available to move home directories.

Customizing the Driver

6

You can custom the driver to met your needs by changing:

- ♦ [Section 6.1, “Driver Parameters,” on page 57](#)
- ♦ [Section 6.2, “Setting Up Security,” on page 57](#)
- ♦ [Section 6.3, “Identity Manager Association Keys,” on page 59](#)
- ♦ [Section 6.4, “Mapping SIF XML to the eDirectory Schema,” on page 59](#)

6.1 Driver Parameters

If you need to change the driver parameters, after the driver is configured, you can do that through the properties of the driver object. The driver parameters can be customized to fit your environment. For a list of the driver parameters and how to access them, see [Section B.1.5, “Driver Parameters,” on page 133](#) in the [Appendix B, “Properties of the Driver,” on page 129](#).

6.2 Setting Up Security

You should initially use HTTP to connect the driver to the ZIS. After the connection is shown to be working, switch to using HTTPS. When passing real student information, we recommend that you use secure HTTP (HTTPS) between the driver and the Zone Integration Server (ZIS). Secure HTTP connections use server authentication. The server is the ZIS. In server authentication, the client (the driver) authenticates that it is communicating with the expected ZIS server. The ZIS server might also require client authentication. Client authentication occurs after the server authentication is complete. The ZIS server authenticates that it is communicating with a known client (the driver).

6.2.1 Server Authentication

For secure HTTP to work, you must import the Certification Authority (CA) certificate used by the ZIS into the `jssecacerts` keystore file to show you trust the CA. To prove that a server belongs to the organization that it claims to represent, the server presents its public key certificate to the driver. This certificate is validated against the CA certificate so the client can be sure of the identity of the server.

The CA certificate must be added to the `java-home/lib/security/jssecacerts` keystore file. For NetWare[®] systems, `java-home` is typically `sys:/java`. For Windows systems, `java-home` is typically `Novell\Nds\jre`. The CA certificate is added to the keystore by using the `keytool` utility (<http://java.sun.com/j2se/1.3/docs/tooldocs/solaris/keytool.html>). For example,

```
java-home/jre/bin/keytool -import -alias zisca -file zisca.cer -
keystore
java-home/jre/lib/security/jssecacerts -storepass changeit
```

This sets the initial password of the `jssecacerts` keystore file to “changeit.” The system administrator should change that password and the default access permission of that file.

6.2.2 Client Authentication

When client authentication (in other words, mutual authentication) is also desired, the client public key and certificate must be stored in a separate keystore file, for example `java-home/lib/security/sifagentcerts`. This keystore file should only hold the one client key. The name of this file is also entered in the driver configuration. You must import the client's CA certificate into the client's trusted-certificate store and the ZIS trusted-certificate store. You first need a client key pair, then a CA must sign the key pair.

One way to get the key pair signed is to use the Novell CA:

- 1** In ConsoleOne[®], open the Security container > select the Organizational CA > *Properties* > *Certificates tab* > *Self Signed Certificate* > click *Export* to export the Novell[®] CA trusted root certificate.
- 2** Select *No*, then click *Next*.
- 3** Save the certificate in Base64 format as `NOVELLCASELFSIGNEDCERT.B64`.
- 4** Import this certificate into the client's trusted-certificate keystore.

```
java-home/jre/bin/keytool -import -alias novellca -file  
NOVELLCASELFSIGNEDCERT.B64 -keypass novell1 -keystore  
java-home/jre/lib/security/cacerts -storepass novell2
```
- 5** This certificate must also be imported into the ZIS trusted-certificate keystore. Consult the ZIS documentation on how this is done.
- 6** Generate a public and private key pair for the agent in a new keystore file. The `-dname` parameter must contain the IP address of the client system or SIF Level 3 Authentication will not work. The `-keyalg` parameter must be RSA.

```
java-home/jre/bin/keytool -genkey -alias sifagent -keyalg RSA -  
dname "CN=137.65.146.24, OU=DirXML, O=Novell, L=Provo, S=Utah,  
C=US" -keypass novell1 -keystore  
java-home/jre/lib/security/sifagentcert -storepass novell2
```
- 7** To guarantee the identity of the client, a certificate is needed to authenticate the key pair ownership. To do this, generate a Certificate Signing Request (CSR) in the `novellagent.csr` file.

```
java-home/jre/bin/keytool -certreq -alias sifagent -file  
novellagent.csr -keypass novell1 -keystore  
java-home/jre/lib/security/sifagentcert -storepass novell2
```
- 8** Now use the Novell CA to generate a certificate for the client's key pair. In ConsoleOne, select *Tools* > *Issue Certificate*.
- 9** In the *Filename* field, browse to and select the `novellagent.csr` file, then click *Next*.
- 10** Select Organizational Certificate Authority, then click *Next*.
- 11** Specify SSL or TSL as the Type, then click *Next*.
- 12** Review the certificate parameters, click *Next*, then click *Finish*.
- 13** Save the certificate in Base64 format as `ISSUEDCERTIFICATE.B64`.
- 14** The certificate now needs to be stored in the `sifagentcert` keystore with the key pair.

```
java-home/jre/bin/keytool -import -trustcacerts -alias sifagent -  
file ISSUEDCERTIFICATE.B64 -keypass novell1 -keystore  
java-home/jre/lib/security/sifagentcert -storepass novell2
```

At this point, your `sifagentcert` keystore consists of the client's CA self-signed certificate and your key and a Certificate Authority has signed it.

- 15 View the `sifagent` keystore. There should be two entries. Your key entry should show "Certificate chain length: 2." The first certificate is your key; the second certificate is the CA that signed it. When the server (ZIS) asks for a certificate, the signed certificate is returned to the server for authentication.

```
java-home/jre/bin/keytool -list -v -keystore
java-home/jre/lib/security/sifagentcerts -storepass novell2
```

6.3 Identity Manager Association Keys

SIF objects have a GUID assigned to them by the application that creates the object. For example, the Student Information System assigns a GUID to each `StudentPersonal` object when a new student is created. The GUID uniquely identifies the SIF object. The GUID is part of the SIF object and is called the `RefId`. The `RefId` is always sent as part of the object. The driver uses the `RefId` for the Identity Manager Association Key.

6.4 Mapping SIF XML to the eDirectory Schema

SIF XML uses names that are hierarchical. Element names include the path to the element, relative to the data object. For example, the name of the `City` element in the `StudentPersonal` object is `StudentAddress/Address/City`.

The Identity Manager driver for SIF uses the path name as the element name, for example, `Name/FirstName` and `Name/LastName`. SIF element names are case sensitive. SIF application names in the Schema Map policy must use the path name. An example segment from a Schema Map policy follows.

```
<attr-name class-name="User">
  <nds-name>Given Name</nds-name>
  <app-name>Name/FirstName</app-name>
</attr-name>
<attr-name class-name="User">
  <nds-name>Surname</nds-name>
  <app-name>Name/LastName</app-name>
</attr-name>
<attr-name class-name="User">
  <nds-name>Telephone Number</nds-name>
  <app-name>PhoneNumber</app-name>
</attr-name>
```

SIF elements can contain attributes. Usually the attribute qualifies the element. For example, the element `Name` has the attribute `Type="02"`. The value "02" qualifies the name as the legal name. SIF attribute values are enumerated in the SIF Implementation Specification or some other recognized standard. The SIF shim does not filter out these attributes or use schema mapping to change their names. The driver simply passes them through so the style sheets can process them. The attribute names are passed through using the namespace "sif" so they are not confused with Identity Manager reserved words. For example:

```
<add-attr attr-name="OtherId" sif:Type="06">
  <value type="string">360367</value>
</add-attr>
<add-attr attr-name="Name/LastName" sif:Type="02">
```

```

    <value type="string">Appleseed</value>
  </add-attr>
<add-attr attr-name="Name/FirstName" sif:Type="02">
  <value type="string">Johnny</value>
</add-attr>
  <add-attr attr-name="TelephoneNumber" sif:Format="NA"
sif:Type="HP">
  <value type="string">123-456-7890</value>
</add-attr>

```

Some SIF elements use an attribute field to specify the element value. For these special attributes, the driver takes the attribute value and passes it to eDirectory™ as the element value. Attributes whose values are used as the element value are specified in the `sifobjects.conf` file. Two examples are:

```

<StatePr Code="PA"/>
<Country Code="US"/>

```

These attributes are changed to:

```

<add-attr attr-name="StudentAddress/Address/StatePr" sif:Code="PA">
  <value type="string">PA</value>
  </add-attr><add-attr attr-name="StudentAddress/Address/Country"
sif:Code="US">
  <value type="string">US</value>
  </add-attr>

```

Activating the Driver

7

Novell® Identity Manager, Integration Modules, and the Provisioning Module must be activated within 90 days of installation, or they shut down. At any time during the 90 days, or afterward, you can choose to activate Identity Manager products.

To activate the driver, see [Activating Novell Identity Manager Products \(http://www.novell.com/documentation/idm/install/data/afbx4oc.html\)](http://www.novell.com/documentation/idm/install/data/afbx4oc.html).

Synchronizing Objects

8

This section explains driver and object synchronization in DirXML[®] 1.1a, Identity Manager 2.0, and Identity Manager 3.x. Driver synchronization was not available for DirXML 1.0 and DirXML 1.1.

After the driver is created, instead of waiting for objects to be modified or created, the data between the two connected systems can be sent through the synchronization process.

- ♦ [Section 8.1, “What Is Synchronization?” on page 63](#)
- ♦ [Section 8.2, “When Is Synchronization Done?” on page 63](#)
- ♦ [Section 8.3, “How Does the Metadirectory Engine Decide Which Object to Synchronize?” on page 64](#)
- ♦ [Section 8.4, “How Does Synchronization Work?” on page 65](#)

8.1 What Is Synchronization?

The actions commonly referred to as “synchronization” in Identity Manager refer to several different but related actions:

- ♦ Synchronization (or merging) of attribute values of an object in the Identity Vault with the corresponding attribute values of an associated object in a connected system.
- ♦ Migration of all Identity Vault objects and classes that are included in the filter on the Subscriber channel.
- ♦ Generation of the list of objects to submit to the driver’s Subscriber channel for synchronization or migration in response to a user request (a manual synchronization).
- ♦ Generation of the list of objects to submit to the driver’s Subscriber channel for synchronization or migration in response to enabling a formerly disabled driver, or in response to a cache error.

8.2 When Is Synchronization Done?

The Metadirectory engine performs object synchronization or merging in the following circumstances:

- ♦ A `<sync>` event element is submitted on the Subscriber or Publisher channel.
- ♦ A `<sync>` event element is submitted on the Subscriber channel in the following circumstances:
 - ♦ The state of the object’s association value is set to “manual” or “migrate.” (This causes an eDirectory[™] event, which in turn causes the Identity Manager caching system to queue an object synchronization command in the affected driver’s cache.)
 - ♦ An object synchronization command is read from the driver’s cache.
- ♦ A `<sync>` event element is submitted on the Publisher channel in the following circumstances:
 - ♦ A driver submits a `<sync>` event element. No known driver currently does this.

- ♦ The Metadirectory engine submits a <sync> event element for each object found as the result of a migrate-into-NDS query. These <sync> events are submitted using the Subscriber thread, but are processed using the Publisher channel filter and policies.
- ♦ An <add> event (real or synthetic) is submitted on a channel and the channel Matching policy finds a matching object in the target system.
- ♦ An <add> event with an association is submitted on the Subscriber channel. This normally occurs only in exceptional cases, such as the bulk load of objects into eDirectory with DirXML-Associations attribute values.
- ♦ An <add> event is submitted on the Publisher channel and an object is found in eDirectory that already has the association value reported with the <add> event.

The Metadirectory engine generates synchronization requests for zero or more objects in the following cases:

- ♦ The user issues a manual driver synchronization request. This corresponds to the *Resync* button in the Driver Set property page in ConsoleOne[®], or to the *Synchronize* button on the iManager Identity Manager Driver Overview page.
- ♦ The Metadirectory engine encounters an error with the driver's cache and cannot recover from the cache error. The driver's cache is deleted and the engine generates object synchronization commands as detailed in [Section 8.3, "How Does the Metadirectory Engine Decide Which Object to Synchronize?," on page 64.](#)

8.3 How Does the Metadirectory Engine Decide Which Object to Synchronize?

The Metadirectory engine processes both manually initiated and automatically initiated synchronization requests in the same manner. The only difference in the processing of manually initiated versus automatically initiated driver synchronization requests is the starting filter time used to filter objects being considered for synchronization.

The starting filter time is used to filter objects that have modification or creation times that are older than the starting time specified in the synchronization request.

For automatically initiated driver synchronization, the starting filter time is obtained from the time stamps of cached eDirectory events. In particular, the starting filter time is the earliest time for the cached events that haven't yet been successfully processed by the driver's Subscriber channel.

For manually initiated driver synchronization, the default starting filter time is the earliest time in the eDirectory database. In Identity Manager 2 and Identity Manager 3, an explicit starting filter time can also be set. In DirXML 1.1a there is no facility to set the starting filter time value for synchronization when manually initiating driver synchronization.

The Metadirectory engine creates a list of objects to be synchronized on the Subscriber channel in the following manner:

1. It finds all objects that:
 - ♦ Have an entry modification time stamp greater than or equal to the starting filter time and
 - ♦ Exist in the filter on the Subscriber channel.

2. It finds all objects that have an entry creation time stamp greater than or equal to the starting filter time.
3. It adds a `synchronize object` command to the driver cache for each unique object found that has an entry modification time stamp greater than or equal to the starting filter time and all objects and classes that are in the Subscriber filter channel in the driver being synchronized.

8.4 How Does Synchronization Work?

After the Metadirectory engine determines that an object is to be synchronized, the following processes occur:

1. Each system (the Identity Vault and the connected system) is queried for all attribute values in the appropriate filters.
 - ♦ eDirectory is queried for all values in the Subscriber filter, and for values that are marked for synchronization in Identity Manager 2.x and Identity Manager 3.x.
 - ♦ The connected system is queried for all values in the Publisher filter, and for values that are marked for synchronization in Identity Manager 2.x and Identity Manager 3.x.
2. The returned attribute values are compared and modification lists are prepared for the Identity Vault and the connected system according to [Table 8-1 on page 66](#), [Table 8-2 on page 67](#), and [Table 8-3 on page 69](#).

In the tables the following pseudo-equations are used:

- ♦ “Left = Right” indicates that the left side receives all values from the right side.
- ♦ “Left = Right[1]” indicates that the left side receives one value from the right side. If there is more than one value, it is indeterminate.
- ♦ “Left += Right” indicates that the left side adds the right side values to the left side’s existing values.
- ♦ “Left = Left + Right” indicates that the left side receives the union of the values of the left and right sides.

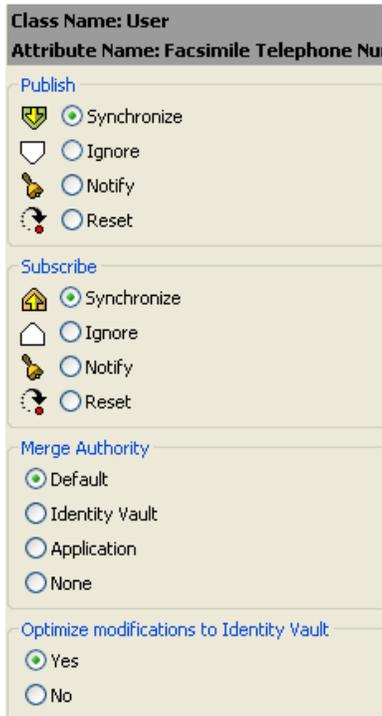
There are three different combinations of selected items in the filter, and each one creates a different output.

- ♦ [Section 8.4.1, “Scenario One,” on page 65](#)
- ♦ [Section 8.4.2, “Scenario Two,” on page 67](#)
- ♦ [Section 8.4.3, “Scenario Three,” on page 68](#)

8.4.1 Scenario One

The attribute is set to *Synchronize* on the Publisher and Subscriber channels, and the merge authority is set to *Default*.

Figure 8-1 Scenario One



The following table contains the values that the Metadirectory engine synchronizes when the attribute is sent through a filter that is set to the configuration for Scenario One. The table shows different outputs depending upon whether the attribute comes from the Identity Vault or the Application, if the attribute is single-valued or multi-valued, and if the attribute is empty or non-empty.

Table 8-1 Output of Scenario One

	Identity Vault single-valued empty	Identity Vault single-valued non-empty	Identity Vault multi-valued empty	Identity Vault multi-valued non-empty
Application single-valued empty	No change	App = Identity Vault	No change	App = Identity Vault[1]
Application single-valued non-empty	Identity Vault = App	App = Identity Vault	Identity Vault = App	Identity Vault + = App
Application multi-valued empty	No change	App = Identity Vault	No change	App = Identity Vault
Application multi-valued non-empty	Identity Vault = App[1]	App + = Identity Vault	Identity Vault = App	App = App + Identity Vault Identity Vault = App + Identity Vault

8.4.2 Scenario Two

The attribute is set to *Synchronize* only on the Subscriber channel, or it is set to *Synchronize* on both the Subscriber and Publisher channels. The merge authority is set to *Identity Vault*.

Figure 8-2 Scenario Two

Class Name: User

Attribute Name: Description

Publish

Synchronize

Ignore

Notify

Reset

Subscribe

Synchronize

Ignore

Notify

Reset

Merge Authority

Default

Identity Vault

Application

None

Optimize modifications to Identity Vault

Yes

No

The following table contains the values that the Metadirectory engine synchronizes when the attribute is sent through a filter that is set to the configuration for Scenario Two. The table shows different outputs depending upon whether the attribute comes from the Identity Vault or the Application, if the attribute is single-valued or multi-valued, and if the attribute is empty or non-empty.

Table 8-2 Output of Scenario Two

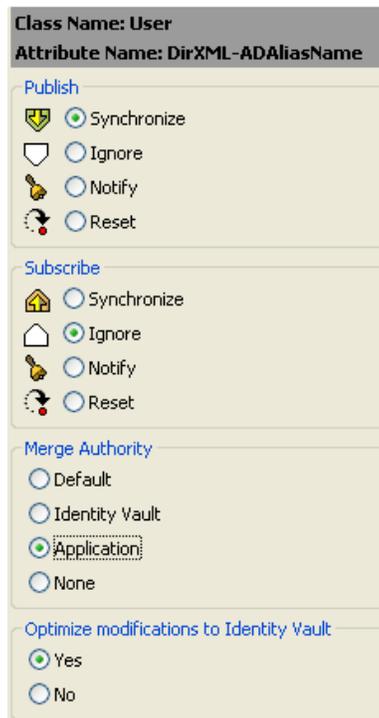
	Identity Vault single-valued empty	Identity Vault single-valued non-empty	Identity Vault multi-valued empty	Identity Vault multi-valued non-empty
Application single-valued empty	No change	App = Identity Vault	No change	App = Identity Vault[1]
Application single-valued empty	App = empty	App = Identity Vault	Identity Vault = App	App = Identity Vault[1]
Application multi-valued empty	No change	App = Identity Vault	No change	App = Identity Vault

	Identity Vault single-valued empty	Identity Vault single-valued non-empty	Identity Vault multi-valued empty	Identity Vault multi-valued non-empty
Application multi-valued non-empty	App = empty	App = Identity Vault	App = empty	App = Identity Vault

8.4.3 Scenario Three

The attribute is set to *Synchronize* on the Publisher channel or the merge authority is set to *Application*.

Figure 8-3 Scenario Three



The following table contains the values that the Metadirectory engine synchronizes when the attribute is sent through a filter that is set to the configuration for Scenario Three. The table shows different outputs depending upon whether the attribute comes from the Identity Vault or the Application, if the attribute is single-valued or multi-valued, and if the attribute is empty or non-empty.

Table 8-3 *Output of Scenario Three*

	Identity Vault single-valued empty	Identity Vault single-valued non-empty	Identity Vault multi-valued empty	Identity Vault multi-valued non-empty
Application single-valued empty	No change	Identity Vault = empty	No change	Identity Vault = empty
Application single-valued non-empty	Identity Vault = App	Identity Vault = App	Identity Vault = App	Identity Vault = App
Application multi-valued empty	No change	Identity Vault = empty	No change	Identity Vault = empty
Application multi-valued non- empty	Identity Vault = App[1]	Identity Vault = App[1]	Identity Vault = App	Identity Vault = App

The driver can be managed through Designer, iManager, or the DirXML[®] Command Line utility.

- ◆ Section 9.1, “Starting, Stopping, or Restarting the Driver,” on page 71
- ◆ Section 9.2, “Using the DirXML Command Line Utility,” on page 72
- ◆ Section 9.3, “Viewing Driver Versioning Information,” on page 72
- ◆ Section 9.4, “Reassociating a Driver Set Object with a Server Object,” on page 77
- ◆ Section 9.5, “Changing the Driver Configuration,” on page 78
- ◆ Section 9.6, “Storing Driver Passwords Securely with Named Passwords,” on page 78
- ◆ Section 9.7, “Adding a Driver Heartbeat,” on page 85

9.1 Starting, Stopping, or Restarting the Driver

- ◆ Section 9.1.1, “Starting the Driver in Designer,” on page 71
- ◆ Section 9.1.2, “Starting the Driver in iManager,” on page 71
- ◆ Section 9.1.3, “Stopping the Driver in Designer,” on page 71
- ◆ Section 9.1.4, “Stopping the Driver in iManager,” on page 71
- ◆ Section 9.1.5, “Restarting the Driver in Designer,” on page 72
- ◆ Section 9.1.6, “Restarting the Driver in iManager,” on page 72

9.1.1 Starting the Driver in Designer

- 1 Open a project in the Modeler, then right-click the driver line.
- 2 Select *Live > Start Driver*.

9.1.2 Starting the Driver in iManager

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Browse to the driver set where the driver exists, then click *Search*.
- 3 Click the upper right corner of the driver icon, then click *Start driver*.

9.1.3 Stopping the Driver in Designer

- 1 Open a project in the Modeler, then right-click the driver line.
- 2 Select *Live > Stop Driver*.

9.1.4 Stopping the Driver in iManager

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Browse to the driver set where the driver exists, then click *Search*.

- 3 Click the upper right corner of the driver icon, then click *Stop driver*.

9.1.5 Restarting the Driver in Designer

- 1 Open a project in the Modeler, then right-click the driver line.
- 2 Select *Live > Restart Driver*.

9.1.6 Restarting the Driver in iManager

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Browse to the driver set where the driver exists, then click *Search*.
- 3 Click the upper right corner of the driver icon, then click *Restart driver*.

9.2 Using the DirXML Command Line Utility

The DirXML Command Line utility provides command line access to manage the driver. This utility is not a replacement for iManager or Designer. The primary use of this utility is to allow you to create platform-specific scripts to manage the driver.

For example, you could create a shell script on Linux to check the status of the driver. See [Appendix A, “DirXML Command Line Utility,” on page 115](#) for detailed information about the DirXML Command Line utility. For daily tasks, use iManager or Designer.

9.3 Viewing Driver Versioning Information

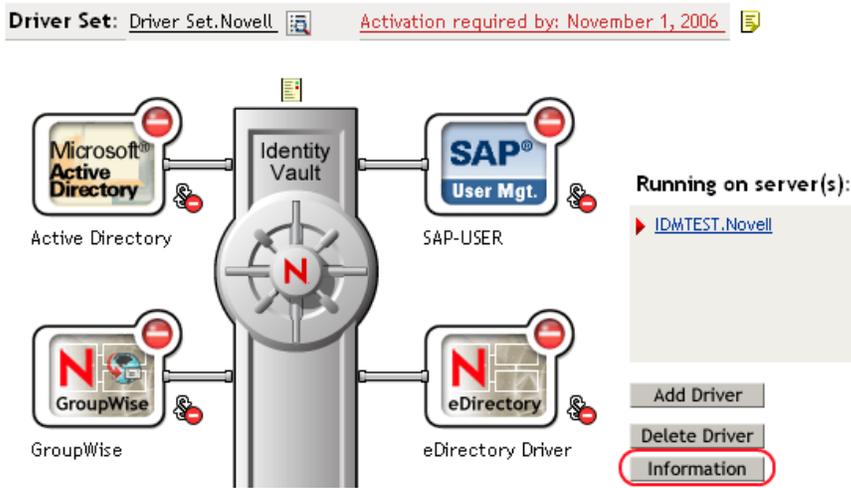
The Versioning Discovery tool only exists in iManager.

- ♦ [Section 9.3.1, “Viewing a Hierarchical Display of Versioning Information,” on page 72](#)
- ♦ [Section 9.3.2, “Viewing the Versioning Information as a Text File,” on page 74](#)
- ♦ [Section 9.3.3, “Saving Versioning Information,” on page 76](#)

9.3.1 Viewing a Hierarchical Display of Versioning Information

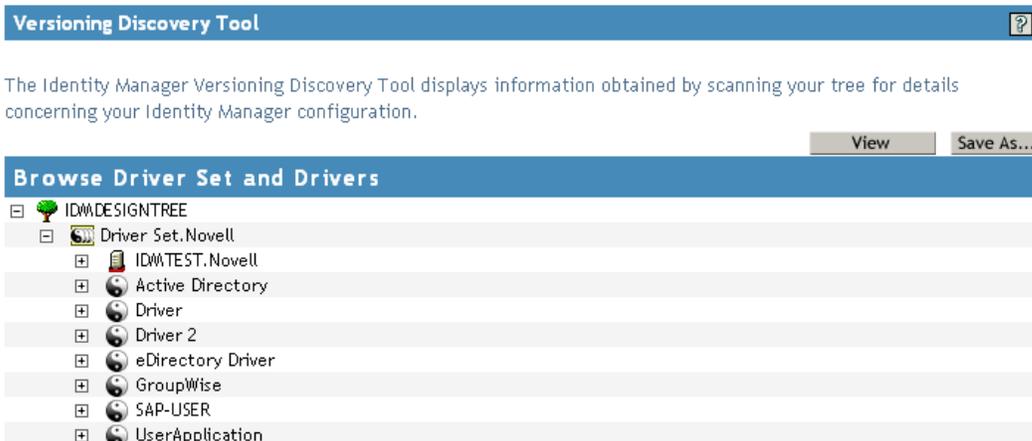
- 1 To find your Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.

2 In the Identity Manager Overview, click *Information*.



You can also select *Identity Manager Utilities > Versions Discovery*, browse to and select the Driver Set object, then click *OK*.

3 View a top-level or unexpanded display of versioning information.



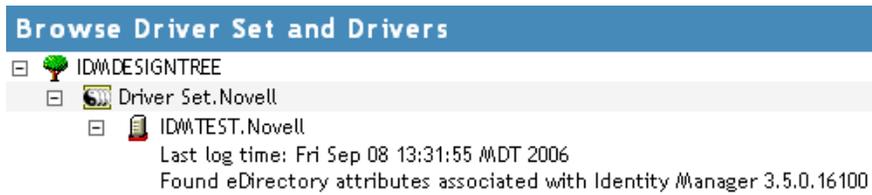
The unexpanded hierarchical view displays the following:

- ◆ The eDirectory™ tree that you are authenticated to
- ◆ The Driver Set object that you selected
- ◆ Servers that are associated with the Driver Set object

If the Driver Set object is associated with two or more servers, you can view Identity Manager information on each server.

- ◆ Drivers

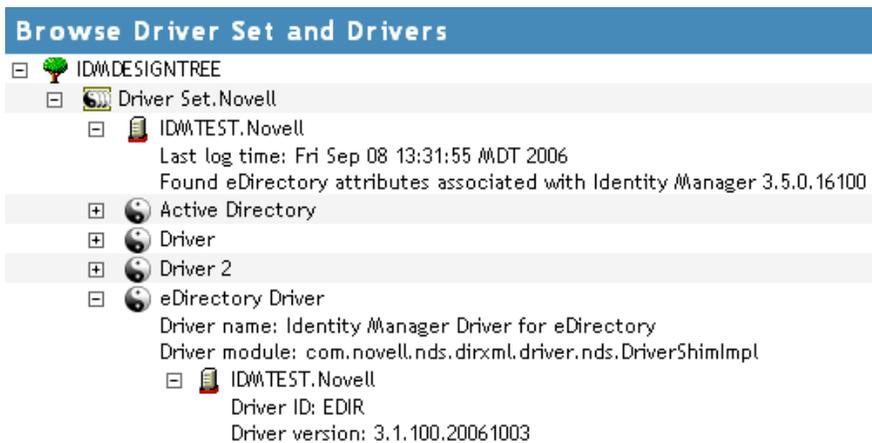
- 4 View versioning information related to servers by expanding the server icon.



The expanded view of a top-level server icon displays the following:

- ◆ Last log time
- ◆ Version of Identity Manager that is running on the server

- 5 View versioning information related to drivers by expanding the driver icon.



The expanded view of a top-level driver icon displays the following:

- ◆ The driver name
- ◆ The driver module (for example, `com.novell.nds.dirxml.driver.delimitedtext.DelimitedTextDriver`)

The expanded view of a server under a driver icon displays the following:

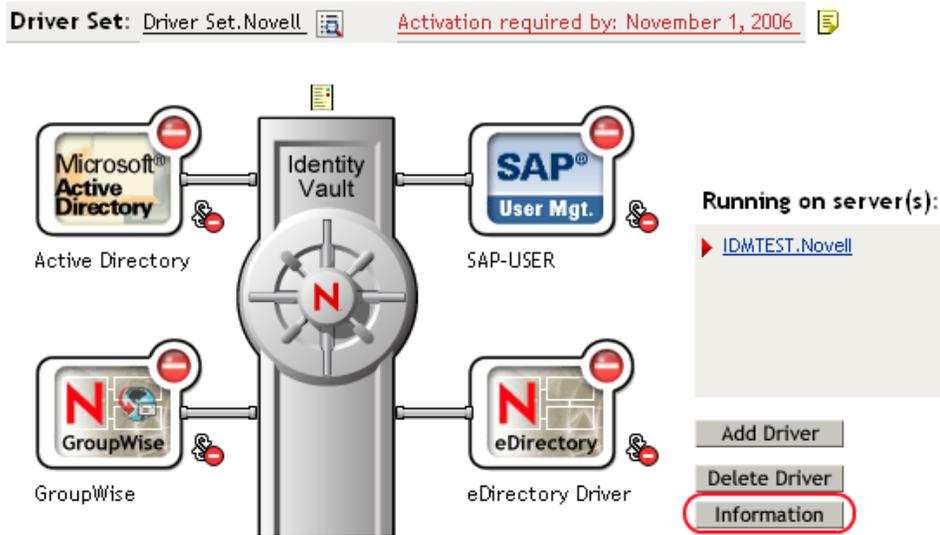
- ◆ The driver ID
- ◆ The version of the instance of the driver running on that server

9.3.2 Viewing the Versioning Information as a Text File

Identity Manager publishes versioning information to a file. You can view this information in text format. The textual representation is the same information contained in the hierarchical view.

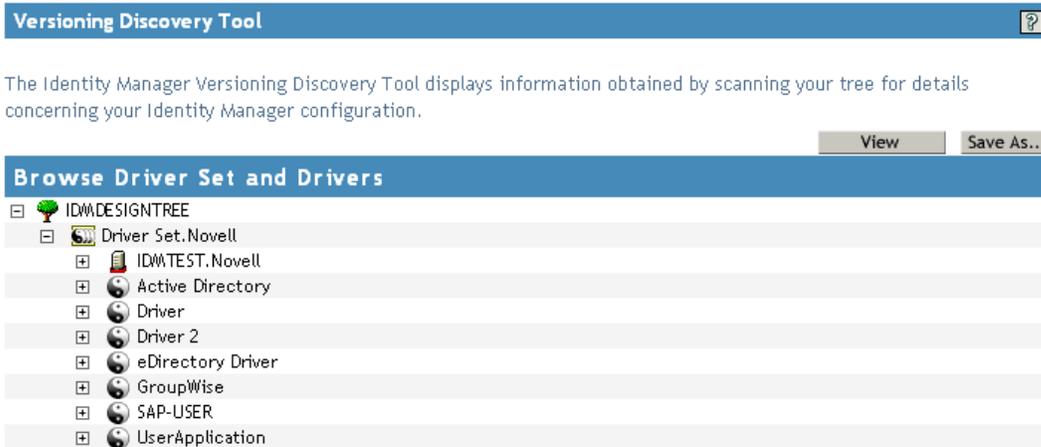
- 1 To find your Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.

2 In the Identity Manager Overview, click *Information*.



You can also select *Identity Manager Utilities > Versioning Discovery*, browse to and select the Driver Set object, then click *Information*.

3 In the Versioning Discovery Tool dialog box, click *View*.



The information is displayed as a text file in the Report Viewer window.

Versioning Discovery Tool - Report Viewer

```
Identity Manager Version Discovery Tool v2.0
Novell, Inc. Copyright 2003, 2004

Version Query started Saturday, January 20, 2007 11:02:52 AM MST

Parameter Summary:
  Default server's DN:  IDMTEST.Novell
  Default server's IP address:  137.65.151.208
  Logged in as admin, context Novell
  Tree name:  IDMDESIGNTREE
  Found 7 Identity Manager Drivers

Driver Set:  Driver Set.Novell
  Driver Set running on Identity Vault:  IDMTEST.Novell
  Last log time:  Fri Sep 08 13:31:55 MDT 2006
  Found eDirectory attributes associated with Identity Manager 3.5.0.1
  Driver:  Active Directory.Driver Set.Novell
  Driver name:  Identity Manager Driver for Active Directory and Exchange
  Driver module:  addriver.dll
  Driver Set running on Identity Vault:  IDMTEST.Novell
  Didn't find any DirXML-DriverVersion attributes associated with
  This may mean the Metadirectory engine is older than
  It does not indicate anything about the version of the
  Driver:  Driver.Driver Set.Novell
  Driver name:  Identity Manager Driver for Peoplesoft
  Driver module:  NPSShim.dll
  Driver Set running on Identity Vault:  IDMTEST.Novell
```

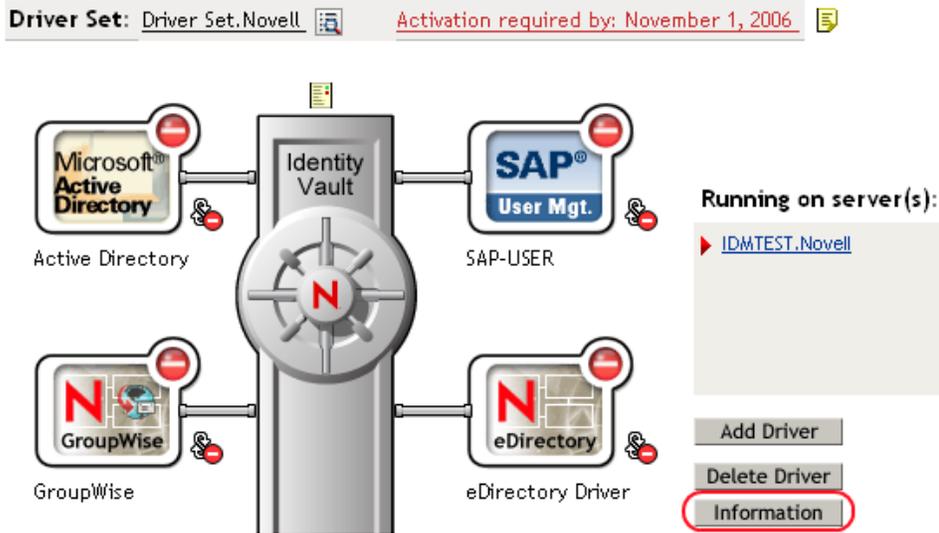
OK

9.3.3 Saving Versioning Information

You can save versioning information to a text file on your local or network drive.

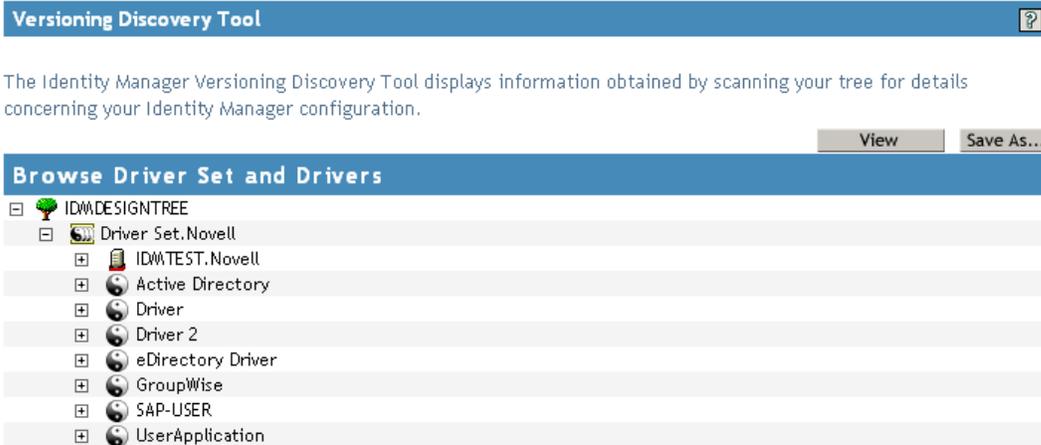
- 1 To find the Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.

2 In the Identity Manager Overview, click *Information*.



You can also select *Identity Manager Utilities > Versioning Discovery*, browse to and select the Driver Set object, then click *Information*.

3 In the Versioning Discovery Tool dialog box, click *Save As*.



4 In the File Download dialog box, click *Save*.

5 Navigate to the desired directory, type a filename, then click *Save*.

Identity Manager saves the data to a text file.

9.4 Reassociating a Driver Set Object with a Server Object

The driver set object should always be associated with a server object. If the driver set is not associated with a server object, none of the drivers in the driver set can start.

If the link between the driver set object and the server object becomes invalid, you see one of the following conditions:

- ◆ When upgrading eDirectory your Identity Manager server, you get the error UniqueSPIException error -783.
- ◆ No server is listed next to the driver set in the Identity Manager Overview window.
- ◆ A server is listed next to the driver set in the Identity Manager Overview window, but the name is garbled text.

To resolve this issue, disassociate the driver set object and the server object, then reassociate them.

- 1 In iManager click *Identity Manager > Identity Manager Overview*, then click *Search* to find the driver set object that the driver should be associated with.
- 2 Click the *Remove server* icon, then click *OK*.
- 3 Click the *Add server* icon, then browse to and select the server object.
- 4 Click *OK*.

9.5 Changing the Driver Configuration

If you need to change the driver configuration, Identity Manager allows you to make the change through iManager or Designer.

To change the driver configuration in iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties*.

To change the driver configuration in Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties*.

For a listing of all of the configuration fields, see [Appendix B, “Properties of the Driver,” on page 129](#).

9.6 Storing Driver Passwords Securely with Named Passwords

Identity Manager allows you to store multiple passwords securely for a particular driver. This functionality is referred to as Named Passwords. Each different password is accessed by a key, or name.

You can also use the Named Passwords feature to store other pieces of information securely, such as a user name.

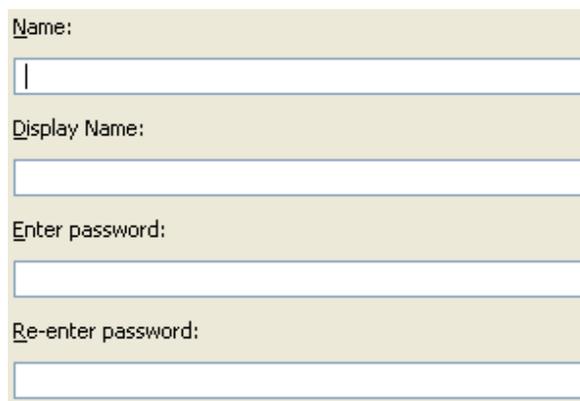
To use a Named Password in a driver policy, you refer to it by the name of the password, instead of using the actual password, and the Metadirectory engine sends the password to the driver. The

method described in this section for storing and retrieving Named Passwords can be used with any driver without making changes to the driver shim.

- ♦ [Section 9.6.1, “Using Designer to Configure Named Passwords,” on page 79](#)
- ♦ [Section 9.6.2, “Using iManager to Configure Named Passwords,” on page 79](#)
- ♦ [Section 9.6.3, “Using Named Passwords in Driver Policies,” on page 81](#)
- ♦ [Section 9.6.4, “Using the DirXML Command Line Utility to Configure Named Passwords,” on page 81](#)

9.6.1 Using Designer to Configure Named Passwords

- 1 Right-click the driver object, then select *Properties*.
- 2 Select *Named Password*, then click *New*.



Name:

Display Name:

Enter password:

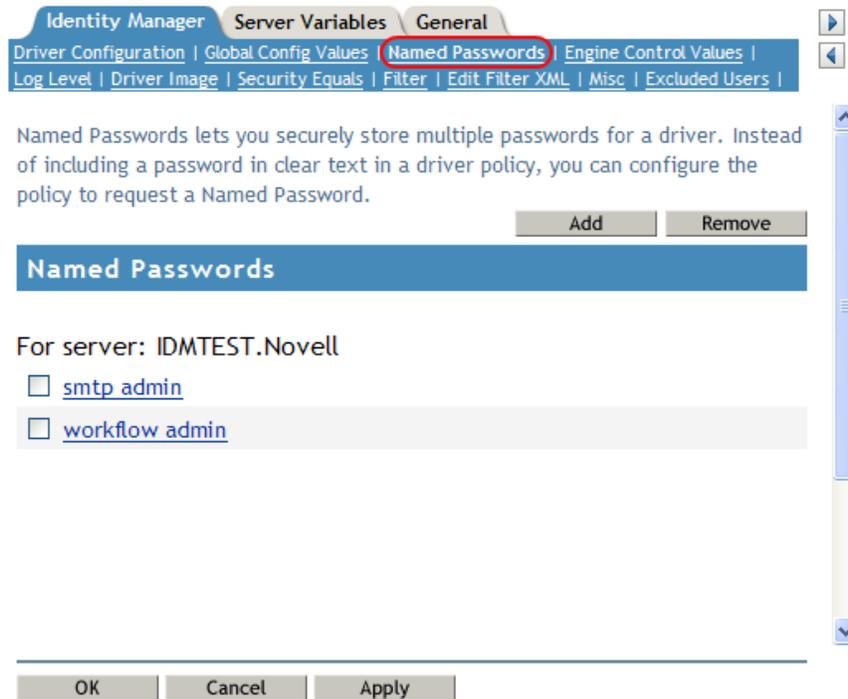
Re-enter password:

- 3 Specify the *Name* of the Named Password.
- 4 Specify the *Display name* of the Named Password.
- 5 Specify the Named Password, then re-enter the password.
- 6 Click *OK* twice.

9.6.2 Using iManager to Configure Named Passwords

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 In the Identity Manager Overview, click the upper right corner of the driver icon, then click *Edit properties*.
- 3 On the Modify Object page on the Identity Manager tab, click *Named Passwords*.

The Named Passwords page appears, listing the current Named Passwords for this driver. If you have not set up any Named Passwords, the list is empty.



- 4 To add a Named Password, click *Add*, complete the fields, then click *OK*.

Named Password

Named Passwords lets you securely store multiple passwords for a driver. Instead of including a password in clear text in a driver policy, you can configure the policy to request a Named Password.

Name:

Display name:

Enter password:

Reenter password:

- 5 Specify a name, display name and a password, then click *OK* twice.
You can use this feature to store other kinds of information securely, such as a username.
- 6 Click *OK* to restart the driver and have the changes take effect.
- 7 To remove a Named Password, select the password name, then click *Remove*.
The password is removed without prompting you to confirm the action.

9.6.3 Using Named Passwords in Driver Policies

- ♦ “Using the Policy Builder” on page 81
- ♦ “Using XSLT” on page 81

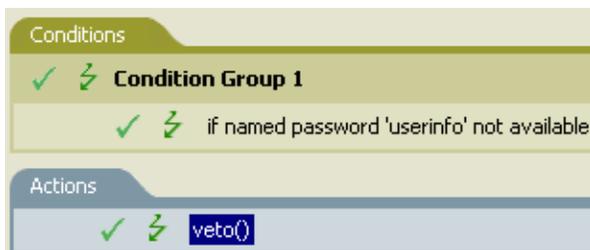
Using the Policy Builder

Policy Builder allows you to make a call to a Named Password. Create a new rule and select Named Password as the condition, then set an action depending upon if the Named Password is available or not available.

- 1 In Designer, launch Policy Builder, right-click, then click *New > Rule*.
- 2 Specify the name of the rule, then click *Next*.
- 3 Select the condition structure, then click *Next*.
- 4 Select *named password* for the *Condition*.
- 5 Browse to and select the Named Password that is stored on the driver.
In this example, it is *userinfo*.
- 6 Select whether the Operator is available or not available.
- 7 Select an action for the *Do* field.
In this example, the action is *veto*.

The example indicates that if the *userinfo* Named Password is not available, then the event is vetoed.

Figure 9-1 A Policy Using Named Passwords



Using XSLT

The following example shows how a Named Password can be referenced in a driver policy on the Subscriber channel in XSLT:

```
<xsl:value-of  
select="query:getNamedPassword($srcQueryProcessor, 'mynamedpassword')"  
xmlns:query="http://www.novell.com/java/  
com.novell.nds.dirxml.driver.XdsQueryProcessor/>
```

9.6.4 Using the DirXML Command Line Utility to Configure Named Passwords

- ♦ “Creating a Named Password in the DirXML Command Line Utility” on page 82
- ♦ “Using the DirXML Command Line Utility to Remove a Named Password” on page 83

Creating a Named Password in the DirXML Command Line Utility

- 1 Run the DirXML Command Line utility.

For information, see [Appendix A, “DirXML Command Line Utility,”](#) on page 115.

- 2 Enter your username and password.

The following list of options appears.

```
DirXML commands
1: Start driver
2: Stop driver
3: Driver operations...
4: Driver set operations...
5: Log events operations...
6: Get DirXML version
7: Job operations...
99: Quit
Enter choice:
```

- 3 Enter 3 for driver operations.

A numbered list of drivers appears.

- 4 Enter the number for the driver you want to add a Named Password to.

The following list of options appears.

```
Select a driver operation for:
driver_name
1: Start driver
2: Stop driver
3: Get driver state
4: Get driver start option
5: Set driver start option
6: Resync driver
7: Migrate from application into DirXML
8: Submit XDS command document to driver
9: Submit XDS event document to driver
10: Queue event for driver
11: Check object password
12: Initialize new driver object
13: Passwords operations
14: Cache operations
99: Exit
Enter choice:
```

- 5 Enter 13 for password operations.

The following list of options appears.

```
Select a password operation
1: Set shim password
2: Reset shim password
3: Set Remote Loader password
4: Clear Remote Loader password
5: Set named password
6: Clear named password(s)
7: List named passwords
```

```
8: Get passwords state
99: Exit
Enter choice:
```

- 6 Enter 5 to set a new Named Password.

The following prompt appears:

```
Enter password name:
```

- 7 Enter the name by which you want to refer to the Named Password.

- 8 Enter the actual password that you want to secure at the following prompt:

```
Enter password:
```

The characters you type for the password are not displayed.

- 9 Confirm the password by entering it again at the following prompt:

```
Confirm password:
```

- 10 After you enter and confirm the password, you are returned to the password operations menu.

- 11 After completing this procedure, you can use the 99 option twice to exit the menu and quit the DirXML Command Line Utility.

Using the DirXML Command Line Utility to Remove a Named Password

This option is useful if you no longer need Named Passwords that you previously created.

- 1 Run the DirXML Command Line utility.

For information, see [Appendix A, “DirXML Command Line Utility,” on page 115](#).

- 2 Enter your username and password.

The following list of options appears.

```
DirXML commands
1: Start driver
2: Stop driver
3: Driver operations...
4: Driver set operations...
5: Log events operations...
6: Get DirXML version
7: Job operations
99: Quit
Enter choice:
```

- 3 Enter 3 for driver operations.

A numbered list of drivers appears.

- 4 Enter the number for the driver you want to remove Named Passwords from.

The following list of options appears.

```
Select a driver operation for:
driver_name
1: Start driver
2: Stop driver
3: Get driver state
4: Get driver start option
5: Set driver start option
6: Resync driver
```

```
7: Migrate from application into DirXML
8: Submit XDS command document to driver
9: Submit XDS event document to driver
10: Queue event for driver
11: Check object password
12: Initialize new driver object
13: Passwords operations
14: Cache operations
99: Exit
Enter choice:
```

5 Enter 13 for password operations.

The following list of options appears.

Select a password operation

```
1: Set shim password
2: Reset shim password
3: Set Remote Loader password
4: Clear Remote Loader password
5: Set named password
6: Clear named password(s)
7: List named passwords
8: Get passwords state
99: Exit
Enter choice:
```

6 (Optional) Enter 7 to see the list of existing Named Passwords.

The list of existing Named Passwords is displayed.

This step can help you make sure you are removing the correct password.

7 Enter 6 to remove one or more Named Passwords.

8 Enter No to remove a single Named Password at the following prompt:

```
Do you want to clear all named passwords? (yes/no):
```

9 Enter the name of the Named Password you want to remove at the following prompt:

```
Enter password name:
```

After you enter the name of the Named Password you want to remove, you are returned to the password operations menu:

```
Select a password operation
1: Set shim password
2: Reset shim password
3: Set Remote Loader password
4: Clear Remote Loader password
5: Set named password
6: Clear named password(s)
7: List named passwords
8: Get passwords state
99: Exit
Enter choice:
```

10 (Optional) Enter 7 to see the list of existing Named Passwords.

This step lets you verify that you have removed the correct password.

- 11 After completing this procedure, you can use the 99 option twice to exit the menu and quit the DirXML Command Line utility.

9.7 Adding a Driver Heartbeat

The driver heartbeat is a feature of the Identity Manager drivers that ship with Identity Manager 2 and later. Its use is optional. The driver heartbeat is configured by using a driver parameter with a time interval specified. If a heartbeat parameter exists and has an interval value other than 0, the driver sends a heartbeat document to the Metadirectory engine if there is no communication on the Publisher channel for the specified interval of time.

The intent of the driver heartbeat is to give you a trigger to allow you to initiate an action at regular intervals, if the driver does not communicate on the Publisher channel as often as you want the action to occur. To take advantage of the heartbeat, you must customize your driver configuration or other tools. The Metadirectory engine accepts the heartbeat document but does not take any action because of it.

For most drivers, a driver parameter for heartbeat is not used in the sample configurations, but you can add it.

A custom driver that is not provided with Identity Manager can also provide a heartbeat document, if the driver developer has written the driver to support it.

To configure the heartbeat:

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Browse to and select your driver set object, then click *Search*.
- 3 In the Identity Manager Overview, click the upper right corner of the driver icon, then click *Edit properties*.
- 4 On the Identity Manager tab, click *Driver Configuration*, scroll to *Driver Parameters*, then look for Heart Beat or a similar display name.

If a driver parameter already exists for heartbeat, you can change the interval and save the changes, and configuration is then complete.

The value of the interval cannot be less than 1. A value of 0 means the feature is turned off.

The unit of time is usually minutes; however, some drivers might choose to implement it differently, such as using seconds.

- 5 If a driver parameter does not exist for heartbeat, click *Edit XML*.
- 6 Add a driver parameter entry like the following example, as a child of <publisher-options>.

```
<pub-heartbeat-interval display-name="Heart Beat">10</pub-heartbeat-interval>
```

TIP: If the driver does not produce a heartbeat document after being restarted, check the placement of the driver parameter in the XML.

- 7 Save the changes, and make sure the driver is stopped and restarted.

After you have added the driver parameter, you can edit the time interval by using the graphical view. Another option is to create a reference to a global configuration value (GCV) for the time interval. Like other global configuration values, the driver heartbeat can be set at the driver set level

instead of on each individual driver object. If a driver does not have a particular global configuration value, and the driver set object does have it, the driver inherits the value from the driver set object.

You can log Identity Manager events using Novell® Audit. Using this service in combination with the driver log level setting provides you with tracking control at a very granular level. For more information, see “[Integrating Identity Manager with Novell Audit](#)” in the *Identity Manager 3.5.1 Logging and Reporting*.

- ♦ [Section 10.1, “Viewing Status Messages for the Identity Manager Driver for SIF,” on page 87](#)
- ♦ [Section 10.2, “Error Messages,” on page 88](#)
- ♦ [Section 10.3, “Common HTTP Status Codes,” on page 102](#)
- ♦ [Section 10.4, “ZIS Return Status,” on page 103](#)
- ♦ [Section 10.5, “Troubleshooting Driver Processes,” on page 103](#)

10.1 Viewing Status Messages for the Identity Manager Driver for SIF

When configuring the driver, status messages can be viewed in the driver set status log, Publisher channel status log, Subscriber channel status log, or in the DSTrace screen. The status log contains error messages. The DSTrace screen contains a trace of the SIF driver activity.

You can also set up logging using Novell Audit. See “[Integrating Identity Manager with Novell Audit](#)” in the *Identity Manager 3.5.1 Logging and Reporting*.

In this section:

- ♦ [“Using the Status Logs” on page 87](#)
- ♦ [“Using the DSTrace Screen” on page 88](#)
- ♦ [“Identity Manager Status Levels” on page 88](#)

10.1.1 Using the Status Logs

To view messages in the Publisher or Subscriber status log:

- 1 In Novell iManager, click *Identity Manager > Identity Manager Overview*. Search for the driver set.
- 2 Click the driver icon.
- 3 In the page that appears showing the configuration for the driver, click the status log icon  for either the Publisher or Subscriber channel.

To view messages in the Driver Set status log:

- 1 In Novell iManager, click *Identity Manager > Identity Manager Overview*. Search for the driver set.
- 2 Click the status log icon .

If you see errors you need to fix, you might want to clear the log so you can see which errors are new.

For a description of messages, see [Section 10.2, “Error Messages,” on page 88](#).

10.1.2 Using the DSTrace Screen

To obtain SIF Driver traces:

- 1 In the DSTrace window, click *Edit > Options > Events > Clear All > select Identity Manager Drivers > Save Default > OK*.
- 2 In iManager, click *Identity Manager > Identity Manager Overview*. Search for the driver set, then click the driver object icon. On the driver parameters page that appears, click the *Misc* tab. Set the trace level to 3, then click *Apply*.

For information about view the driver processes, see [Section 10.5, “Troubleshooting Driver Processes,” on page 103](#).

10.1.3 Identity Manager Status Levels

For each event or operation received from the Identity Vault, the driver returns an XML document containing a status report. If the status report does not indicate success, the document also contains a reason. The table in [Section 10.2, “Error Messages,” on page 88](#) contains error text returned by the driver to Identity Manager.

Possible values for levels are:

- ◆ **Success:** The operation or event was successful.
- ◆ **Warning:** The operation was not successful, but can be ignored without consequences.
- ◆ **Error:** The operation failed.
- ◆ **Fatal:** A fatal error occurred, and the driver will be shut down.
- ◆ **Retry:** The ZIS is unavailable. Identity Manager retries the operation every 30 seconds.

Here are examples of return status in the trace screen:

```
<status event-id="0" level="success"/><status event-id="0" level="warning">SIFdoes not support the Move operation.</status>
```

10.2 Error Messages

This topic contains errors that can be seen in the Status Log or DSTrace screen. The Error Condition column contains the error text returned to Identity Manager. The Level column specifies the status level. The Description column describes situations that might cause the condition and possible actions you can take to fix the problem. The message text and status level are recorded in the Driver Identity Manager log.

A SIF Agent Name must be provided

Source: The status log or DSTrace screen

Explanation: The SIF Agent name is missing from the driver configuration.

Possible Cause: The SIF Agent name was not provided during the configuration of the driver.

Level: Fatal

Action: Edit the driver parameters and add the SIF Agent name. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

A Zone URL must be provided

Source: The status log or DSTrace screen.

Explanation: A Zone URL is missing from the driver parameters.

Possible Cause: A Zone URL was not provided during the configuration of the driver.

Action: Edit the driver parameters and add the Zone URL. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

Authentication level must 0-3

Source: The status log or DSTrace screen.

Explanation: There is no authentication level specified.

Possible Cause: There is no authentication level specified.

Action: Edit the driver parameters and set the authentication level to 0, 1, 2, or 3. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

Connection to ZIS not yet established

Source: The status log or DSTrace screen.

Explanation: The Metadirectory engine sent a command to the driver Subscriber channel. The driver cannot handle the command at this time because it does not have a connection to the ZIS. The engine retries the operation every 30 seconds.

Possible Cause: Driver parameters are not configured properly. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Possible Cause: There is a communication problem between the driver and the ZIS.

Action: Review the driver parameters to make sure they are configured properly. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Retry

Encryption level must be 0-4

Source: The status log or DSTrace screen.

Explanation: No encryption level is specified.

Possible Cause: No encryption level is specified.

Action: Edit the driver parameters and add an encryption level to 0, 1, 2, 3, or 4. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

Error connecting to Zone = java.io.FileNotFoundException: http://ZISserver/zone1

Source: The status log or DSTrace screen.

Explanation: The specified ZIS server cannot be located.

Possible Cause: The DNS server is not configured properly.

Action: Use the IP address of the ZIS server instead of a DNS name.

Level: Error

Error connecting to Zone = java.net.MalformedURLException: no protocol: ...

Source: The status log or DSTrace screen.

Explanation: The URL is malformed.

Action: The URL must begin with http:// or https://. Correct the URL in the driver parameters. See [Section B.1.5, "Driver Parameters," on page 133](#) for more information.

Error connecting to Zone = java.net.SocketException: Connection refused: Connection refused

Source: The status log or the DSTrace screen.

Explanation: The driver cannot establish a connection to the ZIS.

Possible Cause: The ZIS server is up but the ZIS is not running.

Action: Start the ZIS.

Level: Error

Error connecting to Zone = javax.net.ssl.SSLException: Received fatal alert: handshake_failure (no cipher suites in common)

Source: The status log or DSTrace screen.

Explanation: The SSL handshake failed.

Possible Cause: The SSL connection is not configured properly.

Action: Verify that the SSL certificates are valid.

Level: Error

Error processing sifobjects.conf, code = xxx

Source: The status log or DSTrace screen.

Explanation: An error occurred processing the `sifobject.conf` file. The status log or DSTrace screen contains additional information.

Possible Cause: The `sifobject.conf` file could not be accessed.

Possible Cause: The `sifobject.conf` file contains errors.

Action: Verify that the `sifobject.conf` file exists and is accessible.

Action: Verify that the `sifobject.conf` file contains no errors.

Level: Fatal

Error processing sifschema.xml

Source: The status log or DSTrace screen.

Explanation: An error occurred processing the `sifschema.xml` file. The `sifschema.xml` file is required. The status log or DSTrace screen contains additional information.

Possible Cause: The `sifschema.xml` file cannot be accessed.

Possible Cause: The `sifschema.xml` file contains errors.

Action: Verify that the `sifschema.xml` file exists and is accessible.

Action: Verify that there are no errors in the `sifschema.xml` file.

Level: Fatal

java.io.IOException: HTTPS hostname wrong: should be <x.x.x.x>, but cert says <y.y.y.y>

Source: The status log or DSTrace screen.

Explanation: The specified communications protocol is secure HTTP (HTTPS). The ZIS has sent its public key certificate to the driver for authentication. The certificate cannot be authenticated. The `jssecacerts` keystore file contains the server's CA trusted root certificate but the certificate contains the wrong hostname (cn=).

Possible Cause: The `jssecacerts` keystore file contain the server's CA trusted root certificate but the certificate contains the wrong hostname (cn=).

Action: Change the certificate to contain the correct hostname.

Level: Error

java.net.SocketException: Connection reset by peer: JVM_recv in socket input stream read

Source: The status log or DSTrace screen.

Explanation: The driver is expecting a secure connection over HTTPS and the connection is HTTP.

Possible Cause: The URL for the ZIS is incorrect.

Action: Edit the driver parameters to change the ZIS URL. See [Section B.1.5, "Driver Parameters," on page 133](#) for more information.

Level: Error

javax.net.SocketException: Connection timed out: Connection timed out

Source: The status log or DSTrace screen.

Explanation: The specified ZIS server cannot be reached.

Possible Cause: An incorrect address for the ZIS server.

Possible Cause: A firewall is blocking the connection.

Action: Specify the correct address for the ZIS server in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Action: Verify that the firewall allows the driver to connect to the ZIS server.

Level: Error

java.net.SocketException: Network is unreachable

Source: The status log or DSTrace screen.

Explanation: The driver cannot communicate.

Possible Cause: TCP/IP is not configured properly.

Action: Verify that the gateway and subnet mask are correct.

Level: Error

java.net.SocketException: Software caused connection abort:

Source: The status log or DSTrace screen.

Explanation: The driver cannot communicate.

Possible Cause: TCP/IP is not configured properly.

Action: Verify that the gateway and subnet mask are correct.

Level: Error

javax.net.ssl.SSLException: Received fatal alert: bad_certificate

Source: The status log or DSTrace screen.

Explanation: The specified communications protocol is secure HTTP (HTTPS) with client authentication.

Possible Cause: An incorrect or no driver keystore file is specified.

Possible Cause: An incorrect or no driver certificate password is specified.

Action: Specify the correct agent keystore file in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Action: Specify the correct driver certificate password in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Error

javax.net.ssl.SSLException: Received fatal alert: certificate unknown

Source: The status log or DSTrace screen.

Explanation: The specified communications protocol is secure HTTP (HTTPS) with client authentication. The server could not authenticate the client’s key.

Possible Cause: The client key in the agent keystore file is incorrect.

Possible Cause: The client’s CA trusted root certificate is not contained in the server’s (ZIS) trusted keystore.

Action: Generate a new client keystore file.

Action: Generate a new client CA trusted root certificate with the ZIS server trusted keystore.

Level: Error

javax.net.ssl.SSLException: Unrecognized SSL handshake.

Source: The status log or DTrace screen.

Explanation: The contacted server is not expecting a secure connection.

Possible Cause: The specified ZIS URL is not correct.

Action: Verify that the ZIS URL is correct in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Action: Use the ZIS server IP address instead of the DNS name.

Level: Error

javax.net.ssl.SSLException: untrusted server cert chain

Source: The status log or DTrace screen.

Explanation: The ZIS has sent its public key certificate to the driver for authentication. The certificate cannot be authenticated.

Possible Cause: The `jssecacerts` keystore file does not contain the server’s CA trusted root certificate.

Action: Generate a new `jssecacerts` keystore file to contain the server’s CA trusted root certificate.

Manage existing users must be ‘yes’ or ‘no’

Source: The status log or DTrace screen.

Explanation: The *Manage existing users* field does not have a value.

Action: Set *Manage existing users* to *Yes* or *No* in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

Migrate not supported when Match Existing Users is set to no

Source: The status log or DTrace.

Explanation: The SIF driver does not process a *Migrate into the Identity Vault* command when *Manage Existing eDirectory Users* is set to *No*.

Action: If desired, change *Manage Existing eDirectory Users* to *Yes*.

Level: Warning

No initialization parameters

Source: The status log or DTrace.

Explanation: On driver initialization, the Metadirectory engine provided no parameters.

Action: Verify that the Metadirectory engine is properly installed and configured.

Level: Fatal

No publisher initialization parameters

Source: The status log or DSTrace.

Explanation: On driver initialization, the Metadirectory engine provided no parameters.

Action: Verify that the Metadirectory engine is properly installed and configured.

Level: Fatal

No subscriber initialization parameters

Source: The status log or DSTrace.

Explanation: On driver initialization, the Metadirectory engine provided no parameters.

Action: Verify that the Metadirectory engine is properly installed and configured.

Level: Fatal

Publisher filter missing

Source: The status log or DSTrace.

Explanation: On driver initialization, the Metadirectory engine provided no parameters.

Action: Verify that the Metadirectory engine is properly installed and configured.

Level: Fatal

No Publisher Option Parameter

Source: The status log or DSTrace screen.

Explanation: On Publisher channel initialization, the Metadirectory engine provided no Publisher options.

Action: Verify that the driver parameters are configured properly. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

Poll rate is not defined

Source: The status log or DSTrace screen.

Explanation: No valid poll rate has been specified for the driver.

Action: Add a valid poll rate to the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information. The poll rate is an integer greater than 5.

Level: Fatal

Poll rate must be an integer value

Source: The status log or DSTrace screen.

Explanation: No valid poll rate has been specified for the driver.

Action: Add a valid poll rate to the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information. The poll rate is an integer greater than 5.

Level: Fatal

SAXException = Missing whitespace before SYSTEM literal URI. null response received

Source: The status log or DSTrace screen.

Explanation: The specified URL is not correct. The contacted server is not a ZIS.

Possible Cause: An incorrect URL was specified during the configuration of the driver.

Action: Specify a correct URL for the ZIS server. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Error

SIF Category = xxx, SIF Code = xxx, SIF Description = xxx, yyy

Source: The status log or DSTrace screen.

Explanation: An error has been returned from the ZIS. This error is reported to Identity Manager.

Action: Additional information about the category, code, and description can be found in the [SIF Implementation Specification \(http://www.sifinfo.org\)](http://www.sifinfo.org).

Level: Error

SIF does not support the Move operation

Source: The status log or DSTrace screen.

Explanation: The SIF Implementation Specification does not contain the notion of containers.

Level: Warning

SIF does not support the Rename operation

Source: The status log or DSTrace screen.

Explanation: The SIF Implementation Specification does not provide for renaming objects.

Level: Warning

SIF objects to process not provided

Source: The status log or DSTrace screen.

Explanation: The SIF objects to process are not specified in the driver parameters. The values are student, staff, or the names of defined SIF objects.

Action: Add valid objects to process in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

SIF Schema not available, error processing sifobjects.conf, code=xxx

Source: The status log or DSTrace screen.

Explanation: The `sifobjects.conf` file is missing or invalid.

Possible Cause: The file could not be accessed or it contains errors and cannot be processed.

Action: Verify the `sifobjects.conf` file is available and valid.

Level: Fatal

The Incomplete container must be specified

Source: The status log or DSTrace screen.

Explanation: The *Incomplete container* field does not contain a valid DN of an organizational unit (container).

Action: Specify a valid DN for the incomplete container in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

The Incomplete container does not reference an Organization Unit

Source: The status log or DSTrace screen.

Explanation: The *Incomplete container* field does not contain a valid DN of an organizational unit (container).

Action: Specify a valid DN for the incomplete container in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Fatal

Unable to add home directory: novell.jclient.JCException: licenseConnection -1 DSERR_INSUFFICIENT_SPACE

Source: The status log or DSTrace screen.

Explanation: The NetWare[®] system must have sufficient user licenses installed.

Possible Cause: An available license does not exist that can be used for creating the user home directory.

Action: Add more user licenses.

Action: Verify that the licenses for the NetWare server are functional.

Level: Error

Unsupported Subscriber Channel operation =

Source: The status log or DSTrace screen.

Explanation: The Metadirectory engine has passed the driver a command that it cannot convert to a SIF operation.

Level: Error

xmlDoc is null

Source: The status log or DSTrace screen.

Explanation: The Metadirectory engine passed a null document to the Subscriber channel.

Possible Cause: The Metadirectory engine is not configured properly.

Action: Verify that the Metadirectory engine is properly installed and configured.

Level: Error

ZIS connection operational. (This is not an error.)

Source: The status log or DSTrace screen.

Explanation: When the driver has established an operational connection with the ZIS, this message is logged. If the connection is lost, an error is logged. When the connection is reestablished, this message is logged. Because only error messages are logged, this message shows in the log file as an error.

Level: Informational

Zone is not responding = java.net.ConnectException: Connection refused: connect

Source: The status log or DSTrace screen.

Explanation: The ZIS is not up or is not accepting connections.

Possible Cause: The ZIS IP address or port is not correct.

Action: Verify that the ZIS IP address and port number are correct in the driver parameters. See [Section B.1.5, "Driver Parameters," on page 133](#) for more information.

Level: Error

Zone is not responding = java.net.ConnectException: Operation timed out: connect

Source: The status log or DSTrace screen.

Explanation: The specified ZIS server cannot be reached.

Possible Cause: The ZIS IP address is incorrect.

Action: Verify that the ZIS IP address is correct in the driver parameters. See [Section B.1.5, "Driver Parameters," on page 133](#) for more information.

Level: Error

novell.jclient.JCException: modifyEntry -672 ERR_NO_ACCESS

Source: The status log or DSTrace screen.

Explanation: Driver Security Equals is not defined or does not have sufficient rights to perform operation.

Action: Verify that the driver has sufficient rights to perform the operations.

Level: Error

provideUsers is not defined

Source: The status log or DSTrace screen.

Explanation: The driver Subscriber channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

modifyUsers is not defined

Source: The status log or DSTrace screen.

Explanation: The driver Subscriber channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

addUsers is not defined

Source: The status log or DSTrace screen.

Explanation: The driver Subscriber channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

Driver parameter <schools> malformed

Source: The status log or DSTrace screen.

Explanation: The driver Subscriber channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

Driver parameter <schools> parameter is not available

Source: The status log or DSTrace screen.

Explanation: The driver Subscriber channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

School Information zone n does not reference an enabled zone

Source: The status log or DSTrace screen.

Explanation: The Zone number specified in the Subscriber channel section of the Global Configuration Values does not reference an enabled Zone.

Action: Change the Zone number to a reference an enabled Zone. See [Section B.2.4, "Global Configuration Values > Zone Configuration,"](#) on page 139 for more information.

Level: Fatal

At least one School Information must be configured when providing users

Source: The status log or DSTrace screen.

Explanation: When *Be the SIF Default Provider for Students and Staff* is set to *Yes*, one or more School information sets must be configured in the Subscriber channel section of Global Config Values.

Action: Add a School information set to the Subscriber channel. See [Section B.2.2, “Global Configuration Values > Student Configuration,” on page 138](#) or [Section B.2.3, “Global Configuration Values > Staff and Employee Configuration,” on page 138](#) for more information.

Level: Fatal

Connection to ZIS not yet established

Source: The status log or DSTrace screen.

Explanation: The driver has not yet established a connection the Zone when an event is received from the Metadirectory engine

Action: Verify that communication between the driver and the Zone is working.

Action: Verify that the correct Zone information is specified in the driver parameters. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

Level: Retry

A Search Container must be provided

Source: The status log or DSTrace screen.

Explanation: A search container must be specified in the global configuration values.

Action: Add a valid search container in the global configuration values. See [Section B.2.5, “Global Configuration Values > Student Placement,” on page 141](#) for more information.

Level: Fatal

Zone n - Enabled zone must have a URL

Source: The status log or DSTrace screen.

Explanation: In the global configuration values, a Zone is enabled but a URL is not specified or is not correctly formed.

Action: Add a valid URL to the Zone global configuration values. See [Section B.2.4, “Global Configuration Values > Zone Configuration,” on page 139](#) for more information.

Level: Fatal

Zone n - Malformed Zone URL =

Source: The status log or DSTrace screen.

Explanation: In the global configuration values, a Zone is enabled but a URL is not specified or is not correctly formed.

Action: Add a valid URL to the Zone global configuration values. See [Section B.2.4, “Global Configuration Values > Zone Configuration,” on page 139](#) for more information.

Level: Fatal

Zone n - URL not http or https =

Source: The status log or DSTrace screen.

Explanation: In the global configuration values, a Zone is enabled but a URL is not specified or is not correctly formed.

Action: Add a valid URL to the Zone global configuration values. See [Section B.2.4, “Global Configuration Values > Zone Configuration,” on page 139](#) for more information.

Level: Fatal

Zone n - Incomplete container must be configured for enabled zone

Source: The status log or DSTrace message.

Explanation: In the global configuration values, a Zone is enabled but an Incomplete container is not specified.

Action: Add an Incomplete container to the Zone global configuration values. See [Section B.2.4, “Global Configuration Values > Zone Configuration,” on page 139](#) for more information.

Level: Fatal

Zone n - Staff container must be configured for enabled zone

Source: The status log or DSTrace screen.

Explanation: In the global configuration values, a Zone is enabled but a Staff container is not specified.

Action: Specify a Staff container in the Zone global configuration values. See [Section B.2.4, “Global Configuration Values > Zone Configuration,” on page 139](#) for more information.

Level: Fatal

At least one zone must be enabled

Source: The status log or DSTrace screen.

Explanation: In the global configuration values, at least one Zone must be enabled.

Action: Add one enabled Zone in the Zone global configuration values. See [Section B.2.4, “Global Configuration Values > Zone Configuration,” on page 139](#) for more information.

Level: Fatal

Stopping driver because configured objects were not found in the Identity Vault

Source: The status log or DSTrace screen.

Explanation: One of the DNs specified in the global configuration values does not reference a valid object in the Identity Vault.

Possible Cause: The DN is incorrect.

Action: Verify that the DNs specified in the global configuration values are correct. See [Section B.2, “Global Configuration Values,” on page 134](#) for more information.

Level: Fatal

Stopping the drive because nothing is on sendDocToEngine

Source: The status log or DSTrace screen.

Explanation: Internal driver error.

Level: Fatal

Stopping the drive because doc = null

Source: The status log or DSTrace screen.

Explanation: Internal driver error.

Level: Fatal

Stopping the drive because something is on sendDocToEngine

Source: The status log or DSTrace screen.

Explanation: Internal driver error.

Level: Fatal

SIF objects to process not provided

Source: The status log or DSTrace screen.

Explanation: The driver Publisher channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

SIF objects must include StudentPersonal and/or StaffPersonal

Source: The status log or DSTrace screen.

Explanation: The driver Publisher channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

Driver parameter zone malformed

Source: The status log or DSTrace screen.

Explanation: The driver Publisher channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

zone parameter not available

Source: The status log or DSTrace screen.

Explanation: The driver Publisher channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

Driver parameter student malformed

Source: The status log or DSTrace screen.

Explanation: The driver Publisher channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

student parameter not available

Source: The status log or DSTrace screen.

Explanation: The driver Publisher channel configuration is not set up correctly.

Action: Remove the current driver object and add the driver again.

Level: Fatal

No object name provided

Source: The status log or DSTrace screen.

Explanation: Disregard this error message. It does not indicate a problem.

10.3 Common HTTP Status Codes

Below is a list of common HTTP status codes. For a complete list of HTTP codes, see [RFC 2616 \(http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html\)](http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html).

404 Not Found

Source: The status log or DSTrace screen.

Explanation: The requested resource is not available.

Possible Cause: The driver is not configured properly.

Action: Verify that the driver is configured properly. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

401 Unauthorized

Source: The status log or DSTrace screen.

Explanation: The request requires HTTP authentication.

Possible Cause: The authentication information provided is incorrect.

Action: Verify that the driver is configured with the proper authentication information. See [Section B.1.5, “Driver Parameters,” on page 133](#) for more information.

500 Internal Server Error

Source: The status log or DTrace screen.

Explanation: An error occurred in the HTTP server that prevents it from fulfilling the request.

Possible Cause: There is a problem with the HTTP server.

Action: Troubleshoot and resolve any issues with the HTTP server.

503 Service Unavailable

Source: The status log or DTrace screen.

Explanation: The HTTP server is temporarily overloaded and unable to handle the request.

Possible Cause: The HTTP server is providing too many services.

Action: Find out what is causing the server to be overloaded.

10.4 ZIS Return Status

For each event or operation sent to or received from the ZIS, an XML document containing a SIF_Ack message is returned. A SIF_Ack contains either a SIF_Status element acknowledging a successful result or a SIF_Error element indicating the error. The SIF_Error element contains an error number as well as a description of the error. The error number and descriptions are defined in the [SIF Implementation Specification \(http://www.sifinfo.org\)](http://www.sifinfo.org).

Examples

```
<SIF_Status>
  <SIF_Code>0</SIF_Code>
  <SIF_Data>Success</SIF_Data>
</SIF_Status>
<SIF_Error>
  <SIF_Category>1</SIF_Category>
  <SIF_Code>1</SIF_Code>
  <SIF_Desc>Message is not well-formed</SIF_Desc>
  <SIF_ExtendedDesc>Next character must be ">" terminating element
  "Name".</SIF_ExtendedDesc>
</SIF_Error>
```

10.5 Troubleshooting Driver Processes

Viewing driver processes is necessary to analyze unexpected behavior. To view the driver processing events, use DTrace. You should only use it during testing and troubleshooting the driver. Running DTrace while the drivers are in production increases the utilization on the Identity Manager server and can cause events to process very slowly.

- ◆ [Section 10.5.1, “Viewing Driver Processes,” on page 104](#)

10.5.1 Viewing Driver Processes

In order to see the driver processes in DSTrace, values are added to the driver set and the driver objects. You can do this in Designer and iManager.

- ♦ “Adding Trace Levels in Designer” on page 104
- ♦ “Adding Trace Levels in iManager” on page 106
- ♦ “Capturing Driver Processes to a File” on page 106

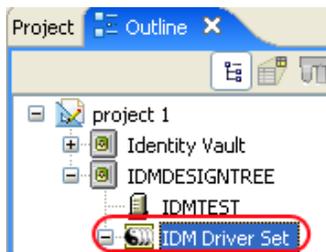
Adding Trace Levels in Designer

You can add trace levels to the driver set object or to each driver object.

- ♦ “Driver Set” on page 104
- ♦ “Driver” on page 105

Driver Set

- 1 In an open project in Designer, select the driver set object in the *Outline* view.



- 2 Right-click and select *Properties*, then click *5. Trace*.
- 3 Set the parameters for tracing, then click *OK*.

Parameter	Description
Driver trace level	As the driver object trace level increases, the amount of information displayed in DSTrace increases. Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5.
XSL trace level	DSTrace displays XSL events. Set this trace level only when troubleshooting XSL style sheets. If you do not want to see XSL information, set the level to zero.
Java debug port	Allows developers to attach a Java* debugger.
Java trace file	When a value is set in this field, all Java information for the driver set object is written to a file. The value for this field is the path for that file. As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.

Parameter	Description
Trace file size limit	<p>Allows you to set a limit for the Java trace file. If you set the file size to <i>Unlimited</i>, the file grows in size until there is no disk space left.</p> <hr/> <p>NOTE: The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.</p>

If you set the trace level on the driver set object, all drivers appear in the DSTrace log. For more information, see [“Capturing Driver Processes to a File” on page 106](#).

Driver

- 1 In an open project in Designer, select the driver object in the *Outline* view.
- 2 Right-click and select *Properties*, then click *Trace*.
- 3 Set the parameters for tracing, then click *OK*.

Parameter	Description
Trace level	<p>As the driver object trace level increases, the amount of information displayed in DSTrace increases.</p> <p>Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5.</p> <p>If you select <i>Use setting from Driver Set</i>, the value is taken from the driver set object.</p>
Trace file	<p>Specify a filename and location for where the Identity Manager information is written for the selected driver.</p> <p>If you select <i>Use setting from Driver Set</i>, the value is taken from the driver set object.</p>
Trace file size limit	<p>Allows you to set a limit for the Java trace file. If you set the file size to <i>Unlimited</i>, the file grows in size until there is no disk space left.</p> <p>If you select <i>Use setting from Driver Set</i>, the value is taken from the driver set object.</p> <hr/> <p>NOTE: The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.</p>
Trace name	<p>The driver trace messages are prepended with the value entered instead of the driver name. Use this option if the driver name is very long.</p>

If you set the parameters only on the driver object, only information for that driver appears in the DSTrace log. For more information, see [“Capturing Driver Processes to a File” on page 106](#).

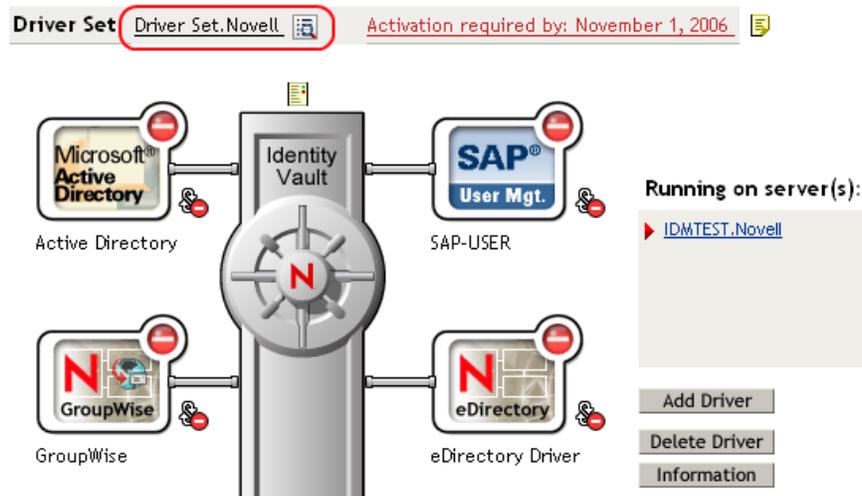
Adding Trace Levels in iManager

You can add trace levels to the driver set object or to each driver object.

- ♦ “Driver Set” on page 106
- ♦ “Driver” on page 106

Driver Set

- 1 In iManager, select *Identity Manager > Identity Manager Overview*.
- 2 Browse to the driver set object, then click *Search*.
- 3 Click the driver set name.



- 4 Select the *Misc* tab for the driver set object.
- 5 Set the parameters for tracing, then click *OK*.
See “Driver trace level” on page 104 for the parameters.

Driver

- 1 In iManager, select *Identity Manager > Identity Manager Overview*.
- 2 Browse to the driver set object where the driver object resides, then click *Search*.
- 3 Click the upper right corner of the driver object, then click *Edit properties*.
- 4 Select the *Misc* tab for the driver object.
- 5 Set the parameters for tracing, then click *OK*.
See “Trace level” on page 105 for the parameters.
The option *Use setting from Driver Set* does not exist in iManager.

Capturing Driver Processes to a File

You can save driver processes to a file by using the parameter on the driver object or by using DSTrace. The parameter on the driver object is the *Trace file* parameter, under the *MISC* tab.

The driver processes that are captured through DSTrace are the processes that occur on the Identity Manager engine. If you use the Remote Loader, you need to capture a trace on the Remote Loader at the same time as you are capturing the trace on the Identity Manager engine.

The following methods help you capture and save Identity Manager processes through DSTrace on different platforms.

- ◆ “NetWare” on page 107
- ◆ “Windows” on page 107
- ◆ “UNIX” on page 108
- ◆ “iMonitor” on page 108
- ◆ “Remote Loader” on page 109

NetWare

Use `dstrace.nlm` to display trace messages on the system console or trace messages to a file (`sys:\system\dstrace.log`). Use `dstrace.nlm` to display the trace messages to a screen labeled DSTrace Console.

- 1 Enter `dstrace.nlm` at the server console to load `dstrace.nlm` into memory.
- 2 Enter `dstrace screen on` at the server console to allow trace messages to appear on the DSTrace Console screen.
- 3 Enter `dstrace file on` at the server console to capture trace messages sent to the DSTrace Console to the `dstrace.log` file.
- 4 (Optional) Enter `dstrace -all` at the server console to make it easier to read the trace log.
- 5 Enter `dstrace +dxml dstrace +dvr` at the server console to display Identity Manager events.
- 6 Enter `dstrace +tags dstrace +time` at the server console to display message tags and time stamps.
- 7 Toggle to the DSTrace Console screen and watch for the event to pass.
- 8 Toggle back to the server console.
- 9 Enter `dstrace file off` at the server console.
This stops capturing trace messages to the log file. It also stops logging information into the file.
- 10 Open the `dstrace.log` in a text editor and search for the event or the object you modified.

Windows

- 1 Open the *Control Panel* > *NDS Services* > `dstrace.dlm`, then click *Start* to display the NDS Server Trace utility window.
- 2 Click *Edit* > *Options*, then click *Clear All* to clear all of the default flags.
- 3 Select *DirXML* and *DirXML Drivers*.
- 4 Click *OK*.
- 5 Click *File* > *New*.
- 6 Specify the filename and location where you want the DSTrace information saved, then click *Open*.

- 7 Wait for the event to occur.
- 8 Click *File > Close*.
This stops the information from being written to the log file.
- 9 Open the file in a text editor and search for the event or the object you modified.

UNIX

- 1 Enter `ndstrace` to start the `ndstrace` utility.
- 2 Enter `set ndstrace=nodebug` to turn off all trace flags currently set.
- 3 Enter `set ndstrace on` to display trace messages to the console.
- 4 Enter `set ndstrace file on` to capture trace messages to the `ndstrace.log` file in the directory where eDirectory is installed. By default it is `/var/nds`.
- 5 Enter `set ndstrace=+dxml` to display the Identity Manager events.
- 6 Enter `set ndstrace=+dvrs` to display the Identity Manager driver events.
- 7 Wait for the event to occur.
- 8 Enter `set ndstrace file off` to stop logging information to the file.
- 9 Enter `exit` to quite the `ndstrace` utility.
- 10 Open the file in a text editor. Search for the event or the object that was modified.

iMonitor

iMonitor allows you to get DSTrace information from a Web browser. It does not matter where Identity Manager is running. The following files run iMonitor:

- ◆ `ndsimon.nlm` runs on NetWare®.
- ◆ `ndsimon.dlm` runs on Windows.
- ◆ `ndsimonitor` runs on UNIX.

- 1 Access iMonitor from `http://server_ip:8008/nds`.

Port 8008 is the default.

- 2 Specify a username and password with administrative rights, then click *Login*.
- 3 Select *Trace Configuration* on the left side.
- 4 Click *Clear All*.
- 5 Select *DirXML* and *DirXML Drivers*.
- 6 Click *Trace On*.
- 7 Select *Trace History* on the left side.
- 8 Click the document with the *Modification Time of Current* to see a live trace.
- 9 Change the *Refresh Interval* if you want to see information more often.
- 10 Select *Trace Configuration* on the left side, then click *Trace Off* to turn the tracing off.
- 11 Select *Trace History* to view the trace history.

The files are distinguished by their time stamp.

Option	Short Name	Parameter	Description
-tracefilemax	-tfm	size	<p>Specifies the approximate maximum size that trace file data can occupy on disk. If you specify this option, there is a trace file with the name specified using the tracefile option and up to 9 additional "roll-over" files. The roll-over files are named using the base of the main trace filename plus "_n", where n is 1 through 9.</p> <p>The size parameter is the number of bytes. Specify the size by using the suffixes K, M, or G for kilobytes, megabytes, or gigabytes.</p> <p>If the trace file data is larger than the specified maximum when the Remote Loader is started, the trace file data remains larger than the specified maximum until roll-over is completed through all 10 files.</p> <p>Example: <code>-tracefilemax 1000M</code> or <code>-tfm 1000M</code></p>

Backing Up the Driver

11

You can use Designer or iManager to create an XML file of the driver. The file contains all of the information entered into the driver during configuration. If the driver becomes corrupted, the exported file can be imported to restore the configuration information.

IMPORTANT: If the driver has been deleted, all of the associations on the objects are purged. When the XML file is imported again, new associations are created through the migration process.

Not all server-specific information stored on the driver is contained in the XML file. Make sure this information is documented through the Doc Gen process in Designer. See “[Documenting Projects](#)” in the *Designer 2.1 for Identity Manager 3.5.1*.

- ♦ [Section 11.1, “Exporting the Driver in Designer,” on page 111](#)
- ♦ [Section 11.2, “Exporting the Driver in iManager,” on page 111](#)

11.1 Exporting the Driver in Designer

- 1 Open a project in Designer, then right-click the driver object.
- 2 Select *Export to Configuration File*.
- 3 Specify a unique name for the configuration file, browse to location where it should be saved, then click *Save*.
- 4 Click *OK* in the Export Configuration Results window.

11.2 Exporting the Driver in iManager

- 1 In iManager, select *Identity Manager > Identity Manager Overview*.
- 2 Browse to and select the driver set object, then click *Search*.
- 3 Click the driver icon.
- 4 Select *Export* in the Identity Manager Driver Overview window.
- 5 Browse to and select the driver object you want to export, then click *Next*.
- 6 Select *Export all policies, linked to the configuration or not* or select *Only export policies that are linked to the configuration*, depending upon the information you want to have stored in the XML file.
- 7 Click *Next*.
- 8 Click *Save As*, then click *Save*.
- 9 Browse and select a location to save the XML file, then click *Save*.
- 10 Click *Finish*.

Security: Best Practices

12

For more information on how to secure the driver and the information it is synchronizing, see “[Security: Best Practices](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

DirXML Command Line Utility

A

The DirXML[®] Command Line utility allows you to use a command line interface to manage the driver. You can create scripts to manage the driver with the commands.

The utility and scripts are installed on all platforms during the Identity Manager installation. The utility is installed to the following locations:

- ♦ Windows: \Novell\Nds\dxcmd.bat
- ♦ NetWare[®]: sys:\system\dxcmd.ncf
- ♦ UNIX: /usr/bin/dxcmd

There are two different methods for using the DirXML Command Line utility:

- ♦ [Section A.1, “Interactive Mode,” on page 115](#)
- ♦ [Section A.2, “Command Line Mode,” on page 124](#)

A.1 Interactive Mode

The interactive mode provides a text interface to control and use the DirXML Command Line utility.

- 1 At the console, enter dxcmd.
- 2 Enter the name of a user with sufficient rights to the Identity Manager objects, such as admin.novell.
- 3 Enter the user’s password.

```
DirXML commands
1: Start driver
2: Stop driver
3: Driver operations...
4: Driver set operations...
5: Log events operations...
6: Get DirXML version
7: Job operations...
99: Quit
Enter choice:
```

- 4 Enter the number of the command you want to perform.
[Table A-1 on page 116](#) contains the list of options and what functionality is available.
- 5 Enter 99 to quit the utility.

NOTE: If you are running eDirectory™ 8.8 on UNIX or Linux, you must specify the -host and -port parameters. For example, dxcmd -host 10.0.0.1 -port 524. If the parameters are not specified, a jclient error occurs:

```
novell.jclient.JCException: connect (to address) 111 UNKNOWN ERROR
```

By default, eDirectory 8.8 is not listening to localhost. The DirXML Command Line utility needs to resolve the server IP address or hostname and the port to be able to authenticate.

Table A-1 *Interactive Mode Options*

Option	Description
1: <i>Start Driver</i>	Starts the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to start the driver.
2: <i>Stop Driver</i>	Stops the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to stop the driver.
3: <i>Driver operations</i>	Lists the operations available for the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to see the operations available. See Table A-2 on page 117 for a list of operations.
4: <i>Driver set operations</i>	Lists the operations available for the driver set. <ul style="list-style-type: none">◆ 1: Associate driver set with server◆ 2: Disassociate driver set from server◆ 99: Exit
5: <i>Log events operations</i>	Lists the operations available for logging events through Novell® Audit. See Table A-5 on page 121 for a description of these options.
6: <i>Get DirXML version</i>	Lists the version of the Identity Manager installed.
7: <i>Job operations</i>	Manages jobs created for Identity Manager.
99: <i>Quit</i>	Exits the DirXML Command Line utility

Figure A-1 *Driver Options*

```
Select a driver operation for:
Active Directory.Driver Set.Novell.IDMDESIGNTREE.

1: Start driver
2: Stop driver
3: Get driver state
4: Get driver start option
5: Set driver start option
6: Resync driver
7: Migrate from application into DirXML
8: Submit XDS command document to driver
9: Submit XDS event document to driver
10: Queue event for driver
11: Check object password
12: Initialize new driver object
13: Passwords operations
14: Cache operations
99: Exit

Enter choice:
```

Table A-2 *Driver Options*

Options	Description
1: <i>Start driver</i>	Starts the driver.
2: <i>Stop driver</i>	Stops the driver.
3: <i>Get driver state</i>	Lists the state of the driver. <ul style="list-style-type: none">◆ 0 - Driver is stopped◆ 1 - Driver is starting◆ 2 - Driver is running◆ 3 - Driver is stopping
4: <i>Get driver start option</i>	Lists the current driver start option. <ul style="list-style-type: none">◆ 1 - Disabled◆ 2 - Manual◆ 3 - Auto
5: <i>Set driver start option</i>	Changes the start option of the driver. <ul style="list-style-type: none">◆ 1 - Disabled◆ 2 - Manual◆ 3 - Auto◆ 99 - Exit
6: <i>Resync driver</i>	<p>Forces a resynchronization of the driver. It prompts for a time delay: <i>Do you want to specify a minimum time for resync? (yes/no)</i>.</p> <p>If you enter Yes, specify the date and time you want the resynchronization to occur: <i>Enter a date/time (format 9/27/05 3:27 PM)</i>.</p> <p>If you enter No, the resynchronization occurs immediately.</p>
7: <i>Migrate from application into DirXML</i>	<p>Processes an XML document that contains a query command: <i>Enter filename of XDS query document:</i></p> <p>Create the XML document that contains a query command by using the Novell nds.dtd (http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsstdtd/query.html).</p> <p>Examples:</p> <p>NetWare: <code>sys:\files\query.xml</code></p> <p>Windows: <code>c:\files\query.xml</code></p> <p>Linux: <code>/files/query.xml</code></p>

Options	Description
8: <i>Submit XDS command document to driver</i>	<p>Processes an XDS command document:</p> <p><i>Enter filename of XDS command document:</i></p> <p>Examples:</p> <p>NetWare: <code>sys:\files\user.xml</code></p> <p>Windows: <code>c:\files\user.xml</code></p> <p>Linux: <code>/files/user.xml</code></p> <p><i>Enter name of file for response:</i></p> <p>Examples:</p> <p>NetWare: <code>sys:\files\user.log</code></p> <p>Windows: <code>c:\files\user.log</code></p> <p>Linux: <code>/files/user.log</code></p>
9: <i>Submit XDS event document to driver</i>	<p>Processes an XDS event document:</p> <p><i>Enter filename of XDS event document:</i></p> <p>Examples:</p> <p>NetWare: <code>sys:\files\add.xml</code></p> <p>Windows: <code>c:\files\add.xml</code></p> <p>Linux: <code>/files/add.xml</code></p>
10: <i>Queue event for driver</i>	<p>Adds an event to the driver queue:</p> <p><i>Enter filename of XDS event document:</i></p> <p>Examples:</p> <p>NetWare: <code>sys:\files\add.xml</code></p> <p>Windows: <code>c:\files\add.xml</code></p> <p>Linux: <code>/files/add.xml</code></p>
11: <i>Check object password</i>	<p>Validates that an object's password in the connected system is associated with a driver. It matches the object's eDirectory password (Distribution Password, used with Universal Password).</p> <p><i>Enter user name:</i></p>
12: <i>Initialize new driver object</i>	<p>Performs an internal initialization of data on a new Driver object. This is only for testing purposes.</p>
13: <i>Password operations</i>	<p>There are nine Password options. See Table A-3 on page 119 for a description of these options.</p>
14: <i>Cache operations</i>	<p>There are five Cache operations. See Table A-4 on page 120 for a description of these options.</p>

Options	Description
99: <i>Exit</i>	Exits the driver options.

Figure A-2 Password Operations

```

Select a password operation

1: Set shim password
2: Clear shim password
3: Set Remote Loader password
4: Clear Remote Loader password
5: Set named password
6: Clear named password(s)
7: List named passwords
8: Get passwords state
99: Exit

Enter choice:

```

Table A-3 Password Operations

Operation	Description
1: <i>Set shim password</i>	Sets the application password. This is the password of the user account you are using to authenticate into the connected system with.
2: <i>Clear shim password</i>	Clears the application password.
3: <i>Set Remote Loader password</i>	The Remote Loader password is used to control access to the Remote Loader instance. Enter the Remote Loader password, then confirm the password by typing it again.
4: <i>Clear Remote Loader password</i>	Clears the Remote Loader password so no Remote Loader password is set on the Driver object.
5: <i>Set named password</i>	Allows you to store a password or other pieces of security information on the driver. See Section 9.6, "Storing Driver Passwords Securely with Named Passwords," on page 78 for more information. There are four prompts to fill in: <ul style="list-style-type: none"> ◆ <i>Enter password name:</i> ◆ <i>Enter password description:</i> ◆ <i>Enter password:</i> ◆ <i>Confirm password:</i>

Operation	Description
6: <i>Clear named passwords</i>	<p>Clears a specified Named Password or all Named Passwords that are stored on the driver object.</p> <p><i>Do you want to clear all named passwords? (yes/no).</i></p> <p>If you enter Yes, all Named Passwords are cleared. If you enter No, you are prompted to specify the password name that you want to clear.</p>
7: <i>List named passwords</i>	<p>Lists all Named Passwords that are stored on the driver object. It lists the password name and the password description.</p>
8: <i>Get password state</i>	<p>Lists if a password is set for:</p> <ul style="list-style-type: none"> ◆ Driver Object password ◆ Application password ◆ Remote loader password <p>The dxcmd utility allows you to set the Application password and the Remote Loader password. You cannot set the Driver Object password with this utility. It shows if the password has been set or not.</p>
99: <i>Exit</i>	<p>Exits the current menu and takes you back to the Driver options.</p>

Figure A-3 *Cache Operations*

```

Enter choice: 14

Select a cache operation

1: Get driver cache limit
2: Set driver cache limit
3: View cached transactions
4: Delete cached transactions
99: Exit

Enter choice:

```

Table A-4 *Cache Operations*

Operation	Description
1: <i>Get driver cache limit</i>	<p>Displays the current cache limit that is set for the driver.</p>
2: <i>Set driver cache limit</i>	<p>Sets the driver cache limit in kilobytes. A value of 0 is unlimited.</p>

Operation	Description
3: <i>View cached transactions</i>	<p>A text file is created with the events that are stored in cache. You can select the number of transactions to view.</p> <ul style="list-style-type: none"> ◆ <i>Enter option token (default=0):</i> ◆ <i>Enter maximum transactions records to return (default=1):</i> ◆ <i>Enter name of file for response:</i>
4: <i>Delete cached transactions</i>	<p>Deletes the transactions stored in cache.</p> <ul style="list-style-type: none"> ◆ <i>Enter position token (default=0):</i> ◆ <i>Enter event-id value of first transaction record to delete (optional):</i> ◆ <i>Enter number of transaction records to delete (default=1):</i>
99: <i>Exit</i>	Exits the current menu and takes you back to the Driver options.

Figure A-4 Log Event Operations

```

Select a log events operation

1: Set driver set log events
2: Reset driver set log events
3: Set driver log events
4: Reset driver log events
99: Exit

Enter choice:

```

Table A-5 Log Events Operations

Operation	Description
1: <i>Set driver set log events</i>	<p>Allows you to log driver set events through Novell Audit. There are 49 items you can select to log. See Table A-6 on page 122 for a list of these options.</p> <p>Type the number of the item you want to log. After the items are selected, enter 99 to accept the selections.</p>
2: <i>Reset driver set log events</i>	Resets all of the log event options.
3: <i>Set driver log events</i>	<p>Allows you to log driver events through Novell Audit. There are 49 items to select to log. See Table A-6 on page 122 for a list of these options.</p> <p>Type the number of the item you want to log. After the items are selected, enter 99 to accept the selections.</p>

Operation	Description
4: <i>Reset driver log events</i>	Resets all of the log event options.
99: <i>Exit</i>	Exits the log events operations menu.

Table A-6 *Driver Set and Driver Log Events*

Options

- 1: Status success
 - 2: Status retry
 - 3: Status warning
 - 4: Status error
 - 5: Status fatal
 - 6: Status other
 - 7: Query elements
 - 8: Add elements
 - 9: Remove elements
 - 10: Modify elements
 - 11: Rename elements
 - 12: Move elements
 - 13: Add-association elements
 - 14: Remove-association elements
 - 15: Query-schema elements
 - 16: Check-password elements
 - 17: Check-object-password elements
 - 18: Modify-password elements
 - 19: Sync elements
 - 20: Pre-transformed XDS document from shim
 - 21: Post input transformation XDS document
 - 22: Post output transformation XDS document
 - 23: Post event transformation XDS document
 - 24: Post placement transformation XDS document
 - 25: Post create transformation XDS document
 - 26: Post mapping transformation <inbound> XDS document
 - 27: Post mapping transformation <outbound> XDS document
-

Options

- 28: Post matching transformation XDS document
 - 29: Post command transformation XDS document
 - 30: Post-filtered XDS document <Publisher>
 - 31: User agent XDS command document
 - 32: Driver resync request
 - 33: Driver migrate from application
 - 34: Driver start
 - 35: Driver stop
 - 36: Password sync
 - 37: Password request
 - 38: Engine error
 - 39: Engine warning
 - 40: Add attribute
 - 41: Clear attribute
 - 42: Add value
 - 43: Remove value
 - 44: Merge entire
 - 45: Get named password
 - 46: Reset Attributes
 - 47: Add Value - Add Entry
 - 48: Set SSO Credential
 - 49: Clear SSO Credential
 - 50: Set SSO Passphrase
 - 51: User defined IDs
 - 99: Accept checked items
-

Table A-7 Enter Table Title Here

Options	Description
1: <i>Get available job definitions</i>	Allows you to select an existing job. <i>Enter the job number:</i> <i>Do you want to filter the job definitions by containment? Enter Yes or No</i> <i>Enter name of the file for response:</i> Examples: NetWare: <code>sys:\files\user.log</code> Windows: <code>c:\files\user.log</code> Linux: <code>/files/user.log</code>
2: <i>Operations on specific job object</i>	Allows you to perform operations for a specific job.

A.2 Command Line Mode

The command line mode allows you to use script or batch files. [Table A-8 on page 124](#) lists the different options that are available.

To use the command line options, decide which items you want to use and string them together.

Example: `dxcmd -user admin.headquarters -host 10.0.0.1 -password n0vell -start test.driverset.headquarters`

This example command starts the driver.

Table A-8 Command Line Options

Option	Description
Configuration	
<code>-user <user name></code>	Specify the name of a user with administrative rights to the drivers you want to test.
<code>-host <name or IP address></code>	Specify the IP address of the server where the driver is installed.
<code>-password <user password></code>	Specify the password of the user specified above.
<code>-port <port number></code>	Specify a port number, if the default port is not used.
<code>-q <quiet mode></code>	Displays very little information when a command is executed.
<code>-v <verbose mode></code>	Displays detailed information when a command is executed.

Option	Description
-s <stdout>	Writes the results of the <code>dxcmd</code> command to <code>stdout</code> .
-? <show this message>	Displays the help menu.
-help <show this message>	Displays the help menu.
Actions	
-start <driver dn>	Starts the driver.
-stop <driver dn>	Stops the driver.
-getstate <driver dn>	Shows the state of the driver as running or stopped.
-getstartoption <driver dn>	Shows the startup option of the driver.
-setstartoption <driver dn> <disabled manual auto> <resync noresync>	Sets how the driver starts if the server is rebooted. Sets whether the objects are to be resynchronized when the driver restarts.
-getcachelimit <driver dn>	Lists the cache limit set for the driver.
-setcachelimit <driver dn> <0 or positive integer>	Sets the cache limit for the driver.
-migrateapp <driver dn> <filename>	Processes an XML document that contains a query command. Create the XML document that contains a query command by using the Novell <code>nds.dtd</code> (http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy_dtd/data/dtdndsoverview.html#dtdndsoverview).
-setshimpassword <driver dn> <password>	Sets the application password. This is the password of the user account you are using to authenticate into the connected system with.
-clearshimpassword <driver dn> <password>	Clears the application password.
-setremoteloaderpassword <driver dn> <password>	Sets the Remote Loader password. The Remote Loader password is used to control access to the Remote Loader instance.
<clearremoteloaderpassword <driver dn>	Clears the Remote Loader password.

Option	Description
-sendcommand <driver dn> <input filename> <output filename>	<p>Processes an XDS command document.</p> <p>Specify the XDS command document as the input file.</p> <p>Examples:</p> <p>NetWare: sys:\files\user.xml</p> <p>Windows: c:\files\user.xml</p> <p>Linux: /files/user.log</p> <p>Specify the output filename to see the results.</p> <p>Examples:</p> <p>NetWare: sys:\files\user.log</p> <p>Windows: c:\files\user.log</p> <p>Linux: /files/user.log</p>
-sendevent <driver dn> <input filename>	<p>Submits a document to the driver's Subscriber channel, bypassing the driver cache. The document is processed ahead of anything that might be in the cache at the time of the submission. It also means that the submission fails if the driver is not running.</p>
-queueevent <driver dn> <input filename>	<p>Submits a document to the driver's Subscriber channel by queuing the document in the driver cache. The document gets processed after anything that might be in the cache at the time of the submission. The submission won't fail if the driver isn't running.</p>
-setlogevents <dn> <integer ...>	<p>Sets Novell Audit log events on the driver. The integer is the option of the item to log. See Table A-6 on page 122 for the list of the integers to enter.</p>
-clearlogevents <dn>	<p>Clears all Novell Audit log events that are set on the driver.</p>
-setdriverset <driver set dn>	<p>Associates a driver set with the server.</p>
-cleardriverset	<p>Clears the driver set association from the server.</p>
-getversion	<p>Shows the version of Identity Manager that is installed.</p>
-initdriver object <dn>	<p>Performs an internal initialization of data on a new Driver object. This is only for testing purposes.</p>
-setnamedpassword <driver dn> <name> <password> [description]	<p>Sets Named Passwords on the driver object. You specify the name, the password, and the description of the Named Password.</p>
-clearnamedpassword <driver dn> <name>	<p>Clears a specified Named Password.</p>
-startjob <job dn>	<p>Starts the specified job.</p>

Option	Description
-abortjob <job dn>	Aborts the specified job.
-getjobrunningstate <job dn>	Returns the specified job's running state.
-getjobenabledstate <job dn>	Returns the specified job's enabled state.
-getjobnextruntime <job dn>	Returns the specified job's next run time.
-updatejob <job dn>	Updates the specified job.
-clearallnamedpasswords <driver dn>	Clears all Named Passwords set on a specific driver.

If a command line is executed successfully, it returns a zero. If the command line returns anything other than zero, it is an error. For example 0 means success, and -641 means invalid operation. -641 is an eDirectory error code. [Table A-9 on page 127](#) contains other values for specific command line options.

Table A-9 *Command Line Option Values*

Command Line Option	Values
-getstate	0- stopped 1- starting 2- running 3- shutting down 11- get schema Anything else that is returned is an error.
-getstartoption	0- disabled 1- manual 2- auto Anything else that is returned is an error.
-getcachelimit	0- unlimited Anything else that is returned is an error.
-getjobrunningstate	0- stopped 1- running Anything else that is returned is an error.
-getjobenabledstate	0- disabled 1- enabled 2- configuration error Anything else that is returned is an error.

Command Line Option	Values
-getjobnextruntime	Return is the next scheduled time for the job in eDirectory time format (number of seconds since 00:00:00 Jan 1, 1970 UTC).

Properties of the Driver

B

There are many different fields and values for the driver. Sometimes the information is displayed differently in iManager than in Designer. This section is a reference for all of the fields on the driver as displayed in iManager and Designer.

The information is presented from the viewpoint of iManager. If a field is different in Designer, it is marked with an  icon.

- ◆ [Section B.1, “Driver Configuration,” on page 129](#)
- ◆ [Section B.2, “Global Configuration Values,” on page 134](#)
- ◆ [Section B.3, “Named Passwords,” on page 144](#)
- ◆ [Section B.4, “Engine Control Values,” on page 144](#)
- ◆ [Section B.5, “Log Level,” on page 146](#)
- ◆ [Section B.6, “Driver Image,” on page 147](#)
- ◆ [Section B.7, “Security Equals,” on page 148](#)
- ◆ [Section B.8, “Filter,” on page 148](#)
- ◆ [Section B.9, “Edit Filter XML,” on page 148](#)
- ◆ [Section B.10, “Misc,” on page 149](#)
- ◆ [Section B.11, “Excluded Users,” on page 150](#)
- ◆ [Section B.12, “Driver Manifest,” on page 150](#)
- ◆ [Section B.13, “Driver Inspector,” on page 150](#)
- ◆ [Section B.14, “Driver Cache Inspector,” on page 151](#)
- ◆ [Section B.15, “Inspector,” on page 151](#)
- ◆ [Section B.16, “Server Variables,” on page 152](#)

B.1 Driver Configuration

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration*.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and click *Properties > Driver Configuration*.

There are different sections under *Driver Configuration*. In this document, each section is listed in a table. The table contains a description of the fields, and the default value or an example of the value that should be specified in the field.

B.1.1 Driver Module

The driver module changes the driver from running locally to running remotely or the reverse.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Driver Module*.
See [Table B-1](#) for a list of the driver module options.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Select the *Driver Module* tab.
See [Table B-1](#) for a list of the driver module options.

Table B-1 *Driver Module Options*

Option	Description
<i>Java</i>	Used to specify the name of the Java class that is instantiated for the shim component of the driver. This class can be located in the <code>classes</code> directory as a class file, or in the <code>lib</code> directory as a <code>.jar</code> file. If this option is selected, the driver is running locally.
<i>Native</i>	Used to specify the name of the <code>.dll</code> file that is instantiated for the application shim component of the driver. If this option is selected, the driver is running locally.
<i>Connect to Remote Loader</i>	Used when the driver is connecting remotely to the connected system.
 <i>Remote Loader Client Configuration for Documentation</i>	 Includes the Remote Loader client configuration information in the driver documentation that is generated by Designer.

B.1.2 Driver Object Password

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Driver Object Password > Set Password*.
See [Table B-2](#) for more information.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and click *Properties > Driver Configuration*.
- 2 Click *Driver Module > Connect to Remote Loader > Driver Object Password > Set Password*.
See [Table B-2](#) for more information.

Table B-2 *Driver Object Password*

Option	Description
<i>Driver Object Password</i>	Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page or the remote driver does not run. This password is used by the Remote Loader to authenticate itself to the remote driver shim.

B.1.3 Authentication

The authentication section stores the information required to authenticate to the connected system.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Authentication*.
See [Table B-3](#) for a list of the authentication options.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Click *Authentication*.
See [Table B-3](#) for a list of the authentication options.

Table B-3 *Authentication Options*

Option	Description
<i>Authentication ID</i> or  <i>User ID</i>	Specify a user application ID. This ID is used to pass Identity Vault subscription information to the application. Example: Administrator
<i>Authentication Context</i> or  <i>Connection Information</i>	Specify the IP address or name of the server the application shim should communicate with.

Option	Description
<i>Remote Loader Connection Parameters</i> or  <i>Host name</i>  <i>Port</i>  <i>KMO</i>  <i>Other parameters</i>	Used only if the driver is connecting to the application through the Remote Loader. The parameter to enter is <code>hostname=xxx.xxx.xxx.xxx port=xxxx</code> <i>kmo=certificatename</i> , when the host name is the IP address of the application server running the Remote Loader server and the port is the port the Remote Loader is listening on. The default port for the Remote Loader is 8090. The <i>kmo</i> entry is optional. It is only used when there is an SSL connection between the Remote Loader and the Metadirectory engine. Example: <code>hostname=10.0.0.1 port=8090</code> <code>kmo=IDMCertificate</code>
<i>Driver Cache Limit (kilobytes)</i> or  <i>Cache limit (KB)</i>	Specify the maximum event cache file size (in KB). If it is set to zero, the file size is unlimited.  Click <i>Unlimited</i> to set the file size to unlimited in Designer.
<i>Application Password</i> or  <i>Set Password</i>	Specify the password for the user object listed in the <i>Authentication ID</i> field.
<i>Remote Loader Password</i> or  <i>Set Password</i>	Used only if the driver is connecting to the application through the Remote Loader. The password is used to control access to the Remote Loader instance. It must be the same password specified during the configuration of the Remote Loader on the connected system.

B.1.4 Startup Option

The Startup Option allows you to set the driver state when the Identity Manager server is started.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Startup Option*.
See [Table B-4](#) for a list of the startup options.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Click *Startup Option*.
See [Table B-4](#) for a list of the startup options.

Table B-4 Startup Options

Option	Description
<i>Auto start</i>	The driver starts every time the Identity Manager server is started.
<i>Manual</i>	The driver does not start when the Identity Manager server is started. The driver must be started through Designer or iManager.
<i>Disabled</i>	The driver has a cache file that stores all of the events. When the driver is set to Disabled, this file is deleted and no new events are stored in the file until the driver state is changed to Manual or Auto Start.
 <i>Do not automatically synchronize the driver</i>	This option only applies if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it is started.

B.1.5 Driver Parameters

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Driver Parameters*.
See [Table B-5](#) for a list of the driver parameters.

In Designer:

- 1 Open a project in the modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Click *Driver Parameters*.
See [Table B-5](#) for a list of the driver parameters.

Table B-5 Driver Parameters

Option	Description
Driver Settings	
<i>SIF Agent Name</i>	Specifies the name the driver uses to register as a SIF Agent with the Zone Integration Server (ZIS). The driver must have a unique, case-sensitive Zone name. You need to coordinated with the Zone Integration Server (ZIS) administrator to make sure that the same name is used when configuring the ZIS, as described in Section 5.2.1, "Configuring the ZIS to Recognize the Driver," on page 45.
<i>SIF Specification version</i>	Specifies the SIF specification version you want this driver to use. Select <i>SIF Spec 1.1</i> or <i>SIF Spec 1.5</i> depending upon the SIF specification version the zone is running.

Option	Description
<i>Driver keystore file</i>	<p>If the ZIS is configured to request client authentication, this is the path and name of the client keystore file.</p> <p>For example: <java-home>\jre\lib\security\sifagentcert</p> <p>The keystore file should only hold the client key and certificate. Leave this field blank when client authentication is not used. For more information, see Section 6.2, "Setting Up Security," on page 57.</p>
<i>Driver certificate password</i>	<p>If the ZIS is configured to request client authentication, this is the key password (not the keystore password). Leave this field blank when client authentication is not used. For more information, see Section 6.2, "Setting Up Security," on page 57.</p>
<i>Authentication level</i>	<p>Specifies the security requirements of the communication channel between the ZIS and the recipient agents. <i>Authentication level</i> and <i>Encryption level</i> define the minimum level of security a data transport channel must provide.</p> <p>See the SIF Specification (http://www.sifinfo.org) for more information about authentication levels.</p>
<i>Encryption level</i>	<p>Specifies the security requirements of the communication channel between the ZIS and the recipient agents. <i>Authentication level</i> and <i>Encryption level</i> define the minimum level of security a data transport channel must provide.</p> <p>See the SIF Specification (http://www.sifinfo.org) for more information about authentication levels.</p>
Publisher Settings	
<i>Poll rate in seconds</i>	<p>The rate the driver polls the ZIS for incoming messages. Novell recommends 900 seconds.</p>
<i>StudentSchoolEnrollment TimeFrame</i>	<p>The time frame of the student enrollment as specified by the Student Information System. StudentSchoolEnrollment objects with a time frame not specified are ignored.</p> <p>Normally, the setting for this parameter should be <i>Current</i>. Specify other combinations only if you Student In</p>

B.2 Global Configuration Values

Global configuration values (GCVs) allow you to specify settings for the Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own.

IMPORTANT: Password Synchronization settings are GCVs, but it's best to edit them in the graphical interface provided on the Server Variables page for the driver, instead of the GCV page. The Server Variables page that shows Password Synchronization settings is accessible as a tab like other driver parameters, or by clicking *Password Management > Password Synchronization*,

searching for the driver, and clicking the driver name. The page contains online help for each Password Synchronization setting.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Global Config Values*.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Global Config Values*.

The SIF driver has different categories of global configuration values.

- ◆ [Section B.2.1, “Global Configuration Values > Driver Configuration,” on page 135](#)
- ◆ [Section B.2.2, “Global Configuration Values > Student Configuration,” on page 138](#)
- ◆ [Section B.2.3, “Global Configuration Values > Staff and Employee Configuration,” on page 138](#)
- ◆ [Section B.2.4, “Global Configuration Values > Zone Configuration,” on page 139](#)
- ◆ [Section B.2.5, “Global Configuration Values > Student Placement,” on page 141](#)
- ◆ [Section B.2.6, “Global Configuration Values > SIF Provider Configuration,” on page 142](#)
- ◆ [Section B.2.7, “Global Configuration Values > Password Configuration,” on page 143](#)

B.2.1 Global Configuration Values > Driver Configuration

The driver configuration GCVs control how the driver synchronizes information.

Table B-6 *Global Configuration Values > Driver Configuration*

GCV	Description
<i>Search container DN</i>	<p>The container below which User IDs must be unique.</p> <p>When creating a new User object, the driver searches the Identity Vault to verify that the new User ID is not already in use. This container and all subcontainers are searched. Choose the district container or a container that is high enough in the tree that user IDs are unique for all students and staff.</p> <p>For example, for the environment shown in Figure 2-6 on page 29, you would specify the District container. This search container is used for all zones.</p> <p>If you select Yes in the <i>Send New Users to SIF</i> field, only users in this container and its subcontainers are sent to SIF.</p>

GCV	Description
<i>Manage preexisting eDirectory users</i>	<p>This option lets you decide whether you want the driver to manage accounts that you already have created in the Identity Vault, before using this driver.</p> <p>The SIF Driver can match students and staff in the Student Information System (SIS) with preexisting Identity Vault users only if the Identity Vault user attribute DirXML-sifSISID contains the student's or staff's ID number.</p> <p>Select <i>Yes</i> if one of the following is true:</p> <ul style="list-style-type: none"> ◆ You want to manage preexisting Identity Vault users, and the DirXML-sifSISID is set on all users. ◆ No users currently exist in the Identity Vault, and you plan to let the driver create them all using the <i>Migrate into the Identity Vault</i> command. <p>Otherwise, select <i>No</i>.</p> <p>If <i>Yes</i> is specified, the <i>Migrate into the Identity Vault</i> command can be used to add or update all SIF users into the Identity Vault.</p> <p>If <i>No</i> is specified, the <i>Migrate into the Identity Vault</i> command is ignored to prevent duplicate users from being created in the Identity Vault.</p> <p>This field does not apply to users added to the Identity Vault by this driver. Identity Manager can always match these Identity Vault users with Student Information System users, and these Identity Vault users are always kept current with changes from the Student Information System.</p> <p>For more information on how to make this decision, see Section 5.4, "Synchronizing the Identity Vault the First Time," on page 48.</p>
<i>Send user updates to SIF</i>	<p>Select <i>Yes</i> if you want changes made to users in the Identity Vault to be sent to SIF. You might want to do this for the following reasons:</p> <ul style="list-style-type: none"> ◆ The Identity Vault is the authoritative source for some student information and you want SIF applications notified when it changes. ◆ Your Student Information System is not SIF-enabled and you want the Novell SIF Driver to inform SIF of changes to student and staff information. <p>Otherwise, select <i>No</i>.</p>
<i>Send new users to SIF</i>	<p>Select <i>Yes</i> if you want new users in the Identity Vault to be sent to SIF. You might want to do this if your Student Information System is not SIF-enabled and you want the Novell SIF Driver to inform SIF of new students and staff.</p> <p>If you select <i>Yes</i> you should also set "Send user updates to SIF" to <i>Yes</i>.</p> <p>Otherwise, select <i>No</i>.</p>

GCV	Description
<i>Send email notification</i>	<p data-bbox="483 260 1219 312">Send an e-mail notification when an Identity Vault account's User ID is renamed or when a new user is created with a non-standard User ID.</p> <p data-bbox="483 338 1284 596">User IDs must be unique. When the driver receives information for a new student from the Student Information System, it follows the format for creating the User ID that you chose in the User ID Format. Before creating the User object, the driver searches for a duplicate ID starting with the container you specified in the Search container DN. If the driver finds the user ID already exists, the driver creates a unique ID by appending a digit to it. For example, if Dawn Smith had the User ID of DSmith, and a new user named David Smith were added, the driver places him in the appropriate container and gives David the User ID: DSmith1.</p> <p data-bbox="483 621 1284 705">Also, when an Identity Vault user account is renamed by the driver, an e-mail notification can be sent. Select <i>Yes</i> if you want e-mail notifications sent. You must have a local SMTP server. Otherwise, select <i>No</i>.</p> <p data-bbox="483 730 1208 783">If you select <i>Yes</i>, you are presented with the following four additional prompts:</p> <ul data-bbox="508 808 1284 1310" style="list-style-type: none"> <li data-bbox="508 808 1284 919">◆ <i>Recipient's email address</i> Replace the sample e-mail address with the recipient's e-mail address, for example, admin@school.com <li data-bbox="508 930 1284 1031">◆ <i>SMTP server address</i> Replace the sample address with the address of an SMTP server, for example, mail.school.com. You must have a local SMTP server. <li data-bbox="508 1041 1284 1167">◆ <i>Optional user account on SMTP server</i> Optional credential for authentication to the SMTP server. If the SMTP server requires authentication, enter the user account name. Otherwise, leave the field blank. <li data-bbox="508 1178 1284 1310">◆ <i>Optional password for user account on SMTP server</i> Optional credential for authentication to the SMTP server. If the SMTP server requires authentication, enter the password for the user account. Otherwise, leave the field blank. <p data-bbox="537 1325 1162 1409">For more information, see the following fields below: <i>Rename student users when naming attributes change</i> and <i>Rename staff users when naming attributes change</i>.</p>
<i>Specify the Student Information System you are using</i>	<p data-bbox="483 1434 1187 1455">Select the Student Information Management System you are using.</p> <ul data-bbox="508 1482 915 1671" style="list-style-type: none"> <li data-bbox="508 1482 854 1503">◆ <i>CSIU Administrative Software</i> <li data-bbox="508 1524 745 1545">◆ <i>Apple Powerschool</i> <li data-bbox="508 1566 773 1587">◆ <i>NCS Pearson SASIxp</i> <li data-bbox="508 1608 915 1629">◆ <i>SunGard Pentamation eSchoolPlus</i> <li data-bbox="508 1650 704 1671">◆ <i>Visual Software</i> <p data-bbox="483 1698 1265 1751">This information is used to accommodate unique features about each SIS. Select <i>Other</i> if the SIS you are using is not listed.</p> <p data-bbox="483 1778 1235 1829">Select <i>Yes</i> if you want to manage student accounts in the Identity Vault. Otherwise select <i>No</i>.</p>

B.2.2 Global Configuration Values > Student Configuration

The student configuration GCVs control how the student objects are created and synchronized.

Table B-7 *Global Configuration Values > Student Configuration*

GCVs	Description
<i>Manage student accounts</i>	Select Yes if you want to manage student accounts in the Identity Vault. Otherwise, select No .
<i>Student user ID format</i>	Configure the Student user ID format. The format is composed of five parts. The five parts are concatenated to produce the user ID. See the description and example in Section 2.4, "Specifying the Pattern for User IDs," on page 31.
<i>Rename student users when naming attributes change</i>	Select Yes if you want student user accounts in the Identity Vault renamed when any of the attributes change that are used to build the User CN (the attributes you select in Student user ID format). Otherwise, select No . See <i>Send e-mail notifications</i> in the Driver Configuration options above.
<i>Student placement is by</i>	Select the criteria used to place students in the Identity Vault tree. <ul style="list-style-type: none">◆ <i>School and Grade</i>: Students are placed based on their school and grade level.◆ <i>School and Graduation Year</i>: Students are placed based on their school and graduation year.◆ <i>Grade Only</i>: Students are placed by grade level only.◆ <i>Graduation Year Only</i>: Students are placed by their graduation year only.◆ <i>School Only</i>: Students are placed by their schools only.
<i>Student password format</i>	Select a password format for students. <ul style="list-style-type: none">◆ <i>Student ID</i>: Student ID number.◆ <i>Preset text</i>: The password is the text specified in the field below.◆ <i>No password</i>: No password is specified; the user logs in without a password.
<i>Student preset text for password</i>	If you selected <i>Preset text</i> in the <i>Student password format</i> field above, specify the password you want to be assigned to new student users. Otherwise, leave this field blank.

B.2.3 Global Configuration Values > Staff and Employee Configuration

The staff and employee GCVs control how these objects are created and synchronized.

Table B-8 Global Configuration Values > Staff and Employee Configuration

GCVs	Description
<i>Manage staff and employee accounts</i>	<p>Select Yes if you want to manage staff and employee accounts in the Identity Vault. Otherwise, select No.</p> <p>Typically <i>StaffPersonal</i> objects are maintained by the SIS, and <i>EmployeePersonal</i> objects are maintained by the HR system.</p> <p>When you select Yes, there are additional options. These options are documented below.</p>
<i>SIF Staff and Employee objects to manage</i>	<ul style="list-style-type: none"> ◆ <i>StaffPersonal</i>: provisions SIS data into the Identity Vault. ◆ <i>EmployeePersonal</i>: provisions HR data in the Identity Vault. ◆ <i>StaffPersonal and EmployeePersonal</i>: Provisions both.
<i>Staff user ID format</i>	<p>Configure the <i>Staff user ID format</i>. The format is composed of five parts. The five parts are concatenated to produce the user ID.</p> <p>See the description and example in Section 2.4, “Specifying the Pattern for User IDs,” on page 31.</p>
<i>Rename staff users when naming attributes change</i>	<p>Select Yes if you want staff user accounts in the Identity Vault renamed when any of the attributes change that are used to build the User CN (the attributes you specify in Staff user ID format). Otherwise, select No. See <i>Send email notification</i> in the Driver Configuration options above.</p>
<i>Staff password format</i>	<p>Select a password format for staff.</p> <ul style="list-style-type: none"> ◆ <i>Staff ID</i>: Staff ID number. ◆ <i>Preset text</i>: Password is the text specified in the prompt below. ◆ <i>No password</i>: No password is specified; the user logs in without a password. You can modify the formats in the Publisher Create style sheet.
<i>Staff preset text for password</i>	<p>If you select <i>Preset text</i> in the <i>Staff password format</i> field above, specify the password you want to be assigned to new staff users. Otherwise, leave this field blank.</p>

B.2.4 Global Configuration Values > Zone Configuration

The Zone GCVs control how information is synchronized to and from the Zone.

Table B-9 Global Configuration Values > Zone Configuration

GCVs	Description
<i>Zone 1</i>	<p>Configuration information for each SIF Zone the driver connects to.</p> <p>Select <i>Show</i> to use the zone. Select <i>Hide</i> if you do not need the zone.</p> <p>The driver can connect up to ten Zones. You can use as many or as few Zones as needed for your environment. The order of the Zones is not important.</p> <p><i>Zone 1</i> through <i>Zone 10</i> contain the same fields. You specify the information for each Zone.</p>
<i>Connection to Zone</i>	<p>Select <i>Enabled</i> if the driver is to connect to this Zone. Select <i>Disabled</i> if the driver is to ignore these parameters. The connection to a configured Zone is disabled, for example, when testing an individual Zone or when a Zone is offline.</p>
<i>Zone URL</i>	<p>The URL of the SIF Zone Integration Server (ZIS) this driver connects to. The URL can be obtained from the ZIS administrator. It is case sensitive.</p> <p>The protocol is HTTP (Hypertext Transfer Protocol) or HTTPS (Secure Hypertext Transfer Protocol).</p> <p>If you have DNS, you can use the hostname. Otherwise, use the IP address.</p> <p>Example URLs are</p> <pre>http://www.myzis.com/Zone1 https://1.2.3.4:123/Zone2</pre> <p>When HTTPS is specified, the CA certificate for the ZIS must be placed in the <i>java-home\jre\lib\security\jssecacerts</i> keystore file. For more information on how to set this up after importing the driver, see Section 6.2, "Setting Up Security," on page 57.</p>
<i>Incomplete Container DN</i>	<p>The DN of the Incomplete container.</p> <p>If the grade or school for a student is not provided by the Student Information System, the user is created in the Incomplete container with login disabled. No template is used when creating the user.</p> <p>If you have users objects appear in the incomplete container, review the objects to find out what information is missing. Delete the objects from the Incomplete container, then adding the missing information to the users in the SIF system.</p> <p>Browse and select the Incomplete container you created for this Zone.</p> <p>This is the Incomplete container that you created during planning, in "Identifying "Incomplete" Containers" on page 25.</p>
<i>Disabled container DN</i>	<p>A student's login is disabled when he or she withdraws from school. If you want the student moved when the login is disabled, browse and select the Disabled container you created for this Zone. If you do not want the user moved, leave this field blank.</p>
<i>Staff container DN</i>	<p>If you are managing SIF staff users, browse and select the container where you want staff users to be placed for this Zone. Leave this field blank if you are not managing staff users.</p>
<i>Staff template DN</i>	<p>If you are managing SIF staff users, browse and select the eDirectory Template object you want to be used when creating staff users. Leave this field blank if you are not managing staff users or if you are not using a template.</p>

B.2.5 Global Configuration Values > Student Placement

The student placement GCVs control where the students are placed in the Identity Vault.

Table B-10 *Global Configuration Values > Student Placement*

GCVs	Descriptions
<i>School 1</i>	<p>Use this field to separate school configurations. Use this section to configure the placement of students in the same school. It places students in an eDirectory container based on their school code, graduation year, or grade level.</p> <p>You need to know the values your Student Information System (SIS) uses for schools, graduation years, and grades. Complete as many Student group placement entries as you need to in order to place all students.</p> <p>Use <i>Show</i> to use the <i>School</i> fields. Use <i>Hide</i> if you do not need all ten options.</p> <p><i>School 1</i> through <i>School 10</i> contain the same fields. Use the additional <i>School</i> field to define information specific for each school you administer.</p>
<i>School code or 'all'</i>	<p>The value of this field is based on your <i>Student placement is by</i> criteria. If you specified <i>School and Grade</i>, <i>School and Graduation Year</i>, or <i>School Only</i> enter the school code for this group of students exactly as it is specified in the Student Information System. Contact the administrator to find out the school code. This code might be alpha, numeric, or a combination.</p> <p>If you specified <i>Group Only</i> or <i>Graduation Year Only</i> in <i>Student placement is by</i>, type <code>all</code>. It must be all lowercase.</p>
<i>Student Group 1 Placement</i>	<p>This section lets you configure the placement of a group of students in the Identity Vault. Students are placed in an eDirectory container based on their school code, graduation year, or grade level. You need to know the values your Student Information System (SIS) uses for schools, graduation years and grades. Complete as many <i>Student Group x Placements</i> entries as you need to place all students.</p> <p><i>Student Group 1 Placement</i> through <i>Student Group 6 Placement</i> contain the same fields. Use the additional <i>Student Group Placement</i> fields to place additional groups of users.</p> <p>To use a <i>Student Group Placement</i> fields set the option to <i>Show</i>. If you do not need all six fields, set any fields not in use to <i>Hide</i>.</p> <p>If you need more than six <i>Student Group Placements</i> for this school, use additional <i>Student Group Placements</i> with the same school code.</p>
<i>Grade code, graduation year, or 'all'</i>	<p>Fill in this field based on your choice in the <i>Student Placement is by field</i>, in the STUDENT CONFIGURATION section.</p> <p>If you specified <i>School and Grade or Grade Only</i> in <i>Student Placement is by field</i>, specify the grade level code exactly as it is specified in the SIS.</p> <p>If you specified <i>School and Graduation Year</i> or <i>Graduation Year Only</i> in Student Placement Is by, specify the graduation year exactly as it is specified in the SIS.</p> <p>If you specified <i>School Only</i> in Student Placement Is by, type <code>all</code>. It must be all lowercase.</p>

GCVs	Descriptions
<i>Student container DN</i>	Browse and select the eDirectory container where you want this group of students to be placed.
<i>Student template DN</i>	Browse and select the eDirectory template you want to be used when creating users for this group of students. Leave this field blank if you are not using a template.

B.2.6 Global Configuration Values > SIF Provider Configuration

Configure this section only when this driver is the SIF provider for student and staff information, as described in [Section 1.4.3, “Sending Data from the Identity Vault to SIF,” on page 17](#). You might want to do this if your Student Information System is not SIF-enabled, and you want the driver to be the SIF provider of student and staff information. Being the provider means this driver responds to SIF queries for information about students and staff.

Table B-11 *Global Configuration Values > SIF Provider Configuration*

GCVs	Description
<i>Be the SIF default provider for students and staff</i>	<p>Select <i>Yes</i> if you want this driver to be the SIF provider for student and staff information. If you select <i>Yes</i>, other settings are displayed.</p> <p>You might want to do this if your Student Information System is not SIF-enabled and you want the Novell SIF Driver to be the SIF provider of student and staff information. Being the provider means this driver responds to SIF queries for information about students and staff. See “Sending Data from the Identity Vault to SIF” on page 17.</p> <p>If you select <i>Yes</i>, you must also set <i>Send User Updates to SIF</i> to <i>Yes</i> and <i>Send New Users to SIF</i> to <i>Yes</i>, and configure one or more sets of School Information.</p> <p>Otherwise, select <i>No</i>.</p>
<i>School information</i>	<p>This field is used to separate school configurations.</p> <p>This prompt and its sub-prompts are only used if you set <i>Be the SIF Default Provider for Students and Staff</i> to <i>Yes</i>.</p> <p>This information is used so the SIF Driver can provide the SIF SchoolInfo objects. You need to know the value your Student Information System uses for each school. Complete as many School Information entries as you need to define all schools.</p>
<i>School code</i>	Specify the school code exactly as it is specified in the Student Information System.
<i>School name</i>	Specify the school name as it is specified in the Student Information System.
<i>Zone number</i>	Specify the Zone number (1-10) this school belongs to.

B.2.7 Global Configuration Values > Password Configuration

For *Password Configuration*, you should only edit the first two settings listed in [Table B-12](#). The others are GCVs regarding Password Synchronization that are common to all drivers. They should

be edited using iManager in *Passwords > Password Synchronization*, not here. Some of them have dependencies on each other that are represented only in the iManager interface. They are explained in “[Password Synchronization across Connected Systems](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

Table B-12 *Global Configuration Values > Password Configuration*

Option	Description
<i>SIF Driver sends user passwords to the Zone</i>	<p>If set to <i>True</i>, the SIF driver sends user passwords in the Identity Vault to the Zone. Passwords are sent as SIF Authorization objects. Other SIF-enabled applications can subscribe to the Zone to receive the passwords.</p> <p>You would set this parameter to <i>True</i> when other SIF-enabled applications want to use the user’s network password. When a Distribution Password is set for a new user or when a Distribution Password is changed in the Identity Vault, the Novell SIF driver sends a SIF Authorization object containing the password to the Zone.</p>
<i>SIF Driver accepts user passwords from the Zone</i>	<p>If set to <i>True</i>, the SIF Driver sets user passwords in the Identity Vault to the passwords received from the Zone. The passwords are received as SIF Authorization objects. The passwords are published to the Zone by other SIF-enabled applications.</p> <p>You would set this parameter to <i>True</i> if you want the network password to be generated by another SIF-enabled application. For example, you have a SIF-enabled application in the Zone that generates a password for each user. When the Novell SIF driver receives the password in a SIF Authorization object, the corresponding user’s eDirectory password is set to this value.</p> <p>If this parameter is set to <i>True</i>, we recommend that the Novell SIF driver also be configured to set a password for each new user. There might be a delay between the creation of the user account and when the password is received, and it is best to make sure the account is protected by a password at all times.</p>
<i>Publish passwords to NDS password</i>	Use the password from the connected system to set the non-reversible NDS® password in eDirectory.
<i>Publish passwords to Distribution Password</i>	Use the password from the connected system to set the NMAS™ Distribution Password used for Identity Manager password synchronization.
<i>Require password policy validation before publishing passwords</i>	If <i>True</i> , applies NMAS password policies during publish password operations. The password is not written to the data store if it does not comply.
<i>Reset user’s external system password to the Identity Manager password on failure</i>	If <i>True</i> , on a publish Distribution Password failure, attempt to reset the password in the connected system using the Distribution Password from the Identity Manager data store.
<i>Notify the user of password synchronization failure via e-mail</i>	If <i>True</i> , notify the user by e-mail of any password synchronization failures.
<i>Connected System or Driver Name</i>	The name of the connected system, application or Identity Manager driver. This value is used by the e-mail notification templates.

B.3 Named Passwords

Identity Manager allows you to store multiple passwords securely for a particular driver. This functionality is referred to as Named Passwords. Each different password is accessed by a key, or name.

You can also use the Named Passwords feature to store other pieces of information securely, such as a user name. To configure Named Passwords, see [Section 9.6, “Storing Driver Passwords Securely with Named Passwords,”](#) on page 78.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Named Passwords*.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Named Passwords*.

B.4 Engine Control Values

The engine control values are a means through which certain default behaviors of the Metadirectory engine can be changed. The values can only be accessed if a server is associated with the Driver Set object.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Engine Control Values*.
See [Table B-13](#) for a list of the engine control values.

In Designer:

- 1 In the Modeler, right-click the driver line.
- 2 Select *Properties > Engine Control Values*.
- 3 Click the tooltip icon to the right of the *Engine Controls For Server* field. If a server is associated with the Identity Vault, and if you are authenticated, the Engine Control Values display in the large pane.
See [Table B-13](#) for a list of the engine control values.

Table B-13 Engine Control Values

Option	Description
<i>Subscriber channel retry interval in seconds</i>	The Subscriber channel retry interval controls how frequently the Metadirectory engine retries the processing of a cached transaction after the application shim's Subscriber object returns a retry status.
<i>Qualified form for DN-syntax attribute values</i>	The qualified specification for DN-syntax attribute values controls whether values for DN-syntax attribute values are presented in unqualified slash form or qualified slash form. A True setting means the values are presented in qualified form.
<i>Qualified form from rename events</i>	The qualified form for rename events controls whether the new-name portion of rename events coming from the Identity Vault is presented to the Subscriber channel with type qualifiers. For example, CN=. A True setting means the names are presented in qualified form.
<i>Maximum eDirectory replication wait time in seconds</i>	The maximum eDirectory™ replication wait time controls the maximum time that the Metadirectory engine waits for a particular change to replicate between the local replica and a remote replica. This only affects operations where the Metadirectory engine is required to contact a remote eDirectory server in the same tree to perform an operation and might need to wait until some change has replicated to or from the remote server before the operation can be completed (for example, object moves when the Identity Manager server does not hold the master replica of the moved object; file system rights operations for Users created from a template.)
<i>Use non-compliant backwards-compatible mode for XSLT</i>	<p>This control sets the XSLT processor used by the Metadirectory engine to a backward-compatible mode. The backward-compatible mode causes the XSLT processor to use one or more behaviors that are not XPath 1.0 and XSLT 1.0 standards-compliant. This is done in the interest of backward-compatibility with existing DirXML® style sheets that depend on the non-standard behaviors.</p> <p>For example, the behavior of the XPath “!=” operator when one operand is a node set and the other operand is other than a node set is incorrect in DirXML releases up to and including Identity Manager 2.0. This behavior has been corrected; however, the corrected behavior is disabled by default through this control in favor of backward-compatibility with existing DirXML style sheets.</p>
<i>Maximum application objects to migrate at once</i>	<p>This control is used to limit the number of application objects that the Metadirectory engine requests from an application during a single query that is performed as part of a Migrate Objects from Application operation.</p> <p>If java.lang.OutOfMemoryError errors are encountered during a Migrate from Application operation, this number should be set lower than the default. The default is 50.</p>

NOTE: This control does not limit the number of application objects that can be migrated; it merely limits the batch size.

Option	Description
<i>Set creatorsName on objects created in Identity Vault</i>	<p>This control is used by the Identity Manager engine to determine if the creatorsName attribute should be set to the DN of this driver on all objects created in the Identity Vault by this driver.</p> <p>Setting the creatorsName attribute allows for easily identifying objects created by this driver, but also carries a performance penalty. If not set, the creatorsName attribute defaults to the DN of the NCP™ Server object that is hosting the driver.</p>
<i>Write pending associations</i>	<p>This control determines whether the Identity Manager engine writes a pending association on an object during Subscriber channel processing.</p> <p>Writing a pending association confers little or no benefit but does incur a performance penalty. Nevertheless, the option exists to turn it on for backward compatibility.</p>
<i>Use password event values</i>	<p>This control determines the source of the value reported for the nspmDistributionPassword attribute for Subscriber channel Add and Modify events.</p> <p>Setting the control to False means that the current value of the nspmDistributionPassword is obtained and reported as the value of the attribute event. This means that only the current password value is available. This is the default behavior.</p> <p>Setting the control to True means that the value recorded with the eDirectory event is decrypted and is reported as the value of the attribute event. This means that both the old password value (if it exists) and the replacement password value at the time of the event are available. This is useful for synchronizing passwords to certain applications that require the old password to enable setting a new password.</p>
<i>Enable password synchronization status reporting</i>	<p>This control determines whether the Identity Manager engine reports the status of Subscriber channel password change events.</p> <p>Reporting the status of Subscriber channel password change events allows applications such as the Identity Manager User Application to monitor the synchronization progress of a password change that should be synchronized to the connected application.</p>

B.5 Log Level

Every driver set and driver has a log level field where you can define the level of errors that should be tracked. The level you indicate here determines which messages are available to the logs. By default, the log level is set to track error messages (this also includes fatal messages). Change the log level if you want to track additional message types.

Novell® recommends that you use Novell Audit instead of setting the log levels. See “[Integrating Identity Manager with Novell Audit](#)” in the *Identity Manager 3.5.1 Logging and Reporting*.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Log Level*.

See [Table B-14](#) for a list of the driver log levels.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Log Level*.

See [Table B-14](#) for a list of the driver log levels.

Table B-14 *Driver Log Levels*

Option	Description
<i>Use log settings from the DriverSet</i>	If this is selected, the driver logs events as the options are set on the Driver Set object.
<i>Log errors</i>	Logs just errors
<i>Log errors and warnings</i>	Logs errors and warnings
<i>Log specific events</i>	Logs the events that are selected. Click the  icon to see a list of the events.
<i>Only update the last log time</i>	Updates the last log time.
<i>Logging off</i>	Turns logging off for the driver.
<i>Turn off logging to DriverSet, Subscriber and Publisher logs</i>	If selected, turns all logging off for this driver on the Driver Set object, Subscriber channel, and the Publisher channel.
<i>Maximum number of entries in the log (50-500)</i>	Number of entries in the log. The default value is 50.

B.6 Driver Image

Allows you to change the image associated with the driver. You can browse and select a different image from the default image.

The image associated with a driver is used by the Identity Manager Overview plug-in when showing the graphical representation of your Identity Manager configuration. Although storing an image is optional, it makes the overview display more intuitive.

NOTE: The driver image is maintained when a driver configuration is exported.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Image*.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > iManager Icon*.

B.7 Security Equals

Use the Security page to view or change the list of objects that the driver is explicitly security equivalent to. This object effectively has all rights of the listed objects.

If you add or delete an object in the list, the system automatically adds or deletes this object in that object's "Security Equal to Me" property. You don't need to add the [Public] trustee or the parent containers of this object to the list, because this object is already implicitly security equivalent to them.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Security Equals*.

Designer does not list the users the driver is security equivalent to.

B.8 Filter

Launches the Filter editor. You can edit the filter from this tab.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Filter*.

The Filter editor is accessed through the outline view in Designer.

- 1 In an open project, click the *Outline* tab.
- 2 Select the driver you want to manage the filter for, then click the plus sign to the left.
- 3 Double-click the *Filter* icon to launch the Filter Editor.

B.9 Edit Filter XML

Allows you to edit the filter directly in XML instead of using the Filter Editor.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Filter*.

You can edit the Filter in XML through the Filter Editor in Designer.

- 1 In an open project, click the *Outline* tab.
- 2 Select the driver you want to manage the filter for, then click the plus sign to the left.

- 3 Double-click the *Filter* icon to launch the Filter Editor, then click *XML Source* at the bottom of the Filter Editor.

B.10 Misc

Allows you to add a trace level to your driver. With the trace level set, DSTrace displays the Identity Manager events as the Metadirectory engine processes the events. The trace level only affects the driver it is set for. Use the trace level for troubleshooting issues with the driver when the driver is deployed. DSTrace displays the output of the specified trace level.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Misc*.

See [Table B-15](#) for a list of the driver trace levels.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Trace*.

See [Table B-15](#) for a list of the driver trace levels.

Table B-15 *Driver Trace Levels*

Option	Description
<i>Trace level</i>	Increases the amount of information displayed in DSTrace. Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5.
<i>Trace file</i>	When a value is set in this field, all Java information for the driver is written to the file. The value for this field is the path for that file. As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.
<i>Trace file size limit</i>	Allows you to set a limit for the Java trace file. If you set the file size to Unlimited, the file grows in size until there is no disk space left. NOTE: The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.
<i>Trace name</i>	Driver trace messages are prepended with the value entered in this field.
 <i>Use setting from Driver Set</i>	This option is only available in Designer. It allows the driver to use the same setting that is set on the Driver Set object.

B.11 Excluded Users

Use this page to create a list of users or resources that are not replicated to the application. Novell recommends that you add all objects that represent an administrative role to this list (for example, the Admin object).

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Excluded Users*.

Designer does not list the excluded users.

B.12 Driver Manifest

The driver manifest is like a resumé for the driver. It states what the driver supports, and includes a few configuration settings. The driver manifest is created by default when the Driver object is imported. A network administrator usually does not need to edit the driver manifest.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Manifest*.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Manifest*.

B.13 Driver Inspector

The Driver Inspector page displays information about all of the objects associated with the driver.

- ♦ **Driver:** A link to run the *Driver Overview* on the driver that is being inspected.
- ♦ **Driver Set:** A link to run the *Driver Set Overview* of the driver set that holds the driver.
- ♦ **Delete:** Deletes the associations of the selected objects.
- ♦ **Refresh:** Select this option to re-read all of the objects associated with the driver and refresh the displayed information.
- ♦ **Actions:** Allows you to perform actions on the objects associated with the driver. Click *Actions* to expand the menu, which includes:
 - ♦ **Show All Associations:** Displays all objects associated with the driver.
 - ♦ **Filter for Disabled Associations:** Displays all objects associated with the driver that have a Disabled state.
 - ♦ **Filter for Manual Associations:** Displays all objects associated with the driver that have a Manual state.

- ♦ **Filter for Migrate Associations:** Displays all objects associated with the driver that have a Migrate state.
- ♦ **Filter for Pending Associations:** Displays all objects associated with the driver that have a Pending state.
- ♦ **Filter for Processed Associations:** Displays all objects associated with the driver that have a Processed state.
- ♦ **Filter for Undefined Associations:** Displays all objects associated with the driver that have an Undefined state.
- ♦ **Association Summary:** Displays the state of all objects associated with the driver.
- ♦ **Object DN:** Displays the DN of the associated objects.
- ♦ **State:** Displays the association state of the object.
- ♦ **Object ID:** Displays the value of the association.

B.14 Driver Cache Inspector

The Driver Cache Inspector page uses a table format to display information about the cache file that stores events while the driver is stopped.

- ♦ **Driver:** A link to run the *Driver Overview* on the driver that is associated with this cache file.
- ♦ **Driver Set:** A link to run the *Driver Set Overview* on the driver set that holds the driver.
- ♦ **Driver's cache on:** Lists the server object that contains this instance of the cache file.
- ♦ **Start/Stop Driver icons:** Displays the current state of the driver and enables you to start or stop the driver.
- ♦ **Edit icon:** Enables you to edit the properties of the currently selected Server object.
- ♦ **Delete:** Deletes the selected items from the cache file.
- ♦ **Refresh:** Enables you to re-read the cache file and refresh the displayed information.
- ♦ **Show:** Limits the number of items to be displayed. The options are:
 - ♦ 25 per page
 - ♦ 50 per page
 - ♦ 100 per page
 - ♦ Other: Enables you to specify a desired number.
- ♦ **Actions:** Enables you to perform actions on the entries in the cache file. Click *Actions* to expand the menu, which includes:
 - ♦ **Expand All:** Expands all of the entries displayed in the cache file.
 - ♦ **Collapse All:** Collapses all of the entries displayed in the cache file.
 - ♦ **Go To:** Enables you to access a specified entry in the cache file. Specify the entry number, then click *OK*.
 - ♦ **Cache Summary:** Summarizes all of the events stored in the cache file.

B.15 Inspector

The Inspector displays information about the connected system without directly accessing the system. Designer does not have this option.

B.16 Server Variables

This page lets you enable and disable password synchronization and the associated options for the selected driver.

When setting up password synchronization, consider both the settings on this page for an individual driver and the Universal Password Configuration options in your password policies.

This page lets you control which password Identity Manager updates directly, either the Universal Password for an Identity Vault, or the Distribution Password used for password synchronization by Identity Manager.

However, Novell Modular Authentication Service (NMAS™) controls whether the various passwords inside the Identity Vault are synchronized with each other. Password Policies are enforced by NMAS, and they include settings for synchronizing Universal Password, NDS® Password, Distribution Password, and Simple Password.

To change these settings in iManager:

- 1 In iManager, select *Passwords > Password Policies*.
- 2 Select a password policy, then click *Edit*.
- 3 Select *Universal Password*.

This option is available from a drop-down list or a tab, depending on your version of iManager and your browser.

- 4 Select *Configuration Options*, make changes, then click *OK*.

NOTE: Enabling or disabling options on this page corresponds to values of True or False for certain global configuration values (GCVs) used for password synchronization in the driver parameters. Novell recommends that you edit them here in the graphical interface, instead of on the GCVs page. This interface helps ensure that you don't set conflicting values for the password synchronization GCVs.

Option	Description
<i>Identity Manager accepts password (Publisher Channel)</i>	<p>If this option is enabled, Identity Manager allows passwords to flow from the connected system driver into the Identity Vault data store.</p> <p>Disabling this option means that no <code><password></code> elements are allowed to flow to Identity Manager. They are stripped out of the XML by a password synchronization policy on the Publisher channel.</p> <p>If this option is enabled, and the option below it for Distribution Password is disabled, a <code><password></code> value coming from the connected system is written directly to the Universal Password in the Identity Vault if it is enabled for the user. If the user's password policy does not enable Universal Password, the password is written to the NDS Password.</p>

Option	Description
<i>Use Distribution Password for password synchronization</i>	<p>To use this setting, you must have a version of eDirectory that supports Universal Password, regardless of whether you have enabled Universal Password in your password policies.</p> <p>If this option is enabled, a password value coming from the connected system is written to the Distribution Password. The Distribution Password is reversible, which means that it can be retrieved from the Identity Vault data store for password synchronization. It is used by Identity Manager for bidirectional password synchronization with connected systems. For Identity Manager to distribute passwords to connected systems, this option must be enabled.</p> <p>NMAS and Password policies control whether the Distribution Password is synchronized with other passwords in the Identity Vault. By default, the Distribution Password is the same as the Universal Password in the Identity Vault.</p> <p>If the password in the Identity Vault is to be independent of password synchronization, so that Identity Manager is a conduit only for synchronizing passwords among connected systems, change this default setting. In the Universal Password Configuration Options in a Password policy, disable <i>Synchronize Universal Password with Distribution Password</i>. This use of Identity Manager password synchronization is also referred to as “tunneling.”</p>
<i>Accept password only if it complies with user's Password Policy</i>	<p>To use this setting, users must have a Password policy assigned that has Universal Password enabled, and Advanced Password Rules enabled and configured.</p> <p>If this option is chosen, Identity Manager does not write a password from this connected system to the Distribution Password in the Identity Manager data store or publish it to connected systems unless the password complies with the user's Password policy.</p> <p>By using the notification option that is also on this page, you can inform users when a password is not set because it is not compliant.</p>

Option	Description
<i>If password does not comply, ignore Password Policy on the connected system by resetting user's password to the Distribution Password</i>	<p>This option lets you enforce Password policies on the connected system by replacing a password that does not comply. If you select this option, and a user's password on the connected system does not comply with the user's Password policy, Identity Manager resets the password on the connected system by using the Distribution Password from the Identity Vault data store.</p> <p>Keep in mind that if you do not select this option, user passwords can become out-of-sync on connected systems.</p> <p>By using the notification option that is also on this page, you can inform users when a password is not set or reset. Notification is especially helpful for this option. If the user changes to a password that is allowed by the connected system but rejected by Identity Manager because of the Password policy, the user won't know that the password has been reset until the user receives a notification or tries to log in to the connected system with the old password.</p> <hr/> <p>NOTE: Consider the connected system's password policies when deciding whether to use this option. Some connected systems might not allow the reset because they don't allow you to repeat passwords.</p>
<i>Always accept password; ignore Password Policies</i>	<p>If you select this option, Identity Manager does not enforce the user's Password policy for this connected system. Identity Manager writes the password from this connected system to the Distribution Password in the Identity Vault data store, and distributes it to other connected systems, even if the password does not comply with the user's Password policy.</p>
<i>Application accepts passwords (Subscriber Channel)</i>	<p>If you select this option, the driver sends passwords from the Identity Vault data store to this connected system. This also means that if a user changes the password on a different connected system that is publishing passwords to the Distribution Password in the Identity Vault data store, the password is changed on this connected system.</p> <p>By default, the Distribution Password is the same as the Universal Password in the Identity Vault, so changes to the Universal Password made in the Identity Vault are also sent to the connected system.</p> <p>If you want the password in the Identity Vault to be independent of password synchronization, so that Identity Manager is a conduit only for synchronizing passwords among connected systems, you can change this default setting. In the Universal Password Configuration Options in a password policy, disable <i>Synchronize Universal Password with Distribution Password</i>. This use of password synchronization is also referred to as "tunneling."</p>
<i>Notify the user of password synchronization failure via-email</i>	<p>If you select this option, e-mail is sent to the user if a password is not synchronized, set, or reset. The e-mail that is sent to the user is based on an e-mail template. This template is provided by the Password Synchronization application. However, for the template to work, you must customize it and specify an e-mail server to send the notification messages.</p> <hr/> <p>NOTE: To set up e-mail notification, select <i>Passwords > Edit EMail Templates</i>.</p>

Glossary

This glossary contains some basic Identity Manager terms and SIF terms used in this driver documentation.

Agent

A SIF term. SIF-enabled software that interfaces with an application on one side and a Zone Integration Server on the other side. The Agent is used to make the application's data available to the Zone or to consume data from the Zone and to make it available to the application. The Identity Manager Driver for SIF is a SIF Agent.

shim

An Identity Manager term. Another word for driver.

Schools Interoperability Framework (SIF)

A SIF term. The Schools Interoperability Framework (SIF) is an industry initiative to develop an open specification for ensuring that K-12 instructional and administrative software applications interact and share data seamlessly. SIF is not a product, but rather an industry-supported technical blueprint for K-12 software. For additional information about SIF, see the [Schools Interoperability Framework Web site \(http://www.sifinfo.org\)](http://www.sifinfo.org).

Connecting the applications with a common framework allows you to enter information once in an authoritative information source, such as a **Student Information System (SIS)**, and then publish that information so other systems can be updated automatically.

Student Information System (SIS)

A SIF term. A K-12 application for maintaining student information. Some Student Information Systems also store faculty and staff information.

Publisher channel

An Identity Manager term. The work of provisioning SIF student or staff from the SIS to Novell® eDirectory™ users is done through the Publisher channel of Identity Manager. You can customize the configuration that comes with the driver.

For more information on the channels, see *Novell Identity Manager 3.5.1 Administration Guide*.

Subscriber channel

An Identity Manager term. In the base configuration, the student information system is the authoritative data source for student information, so no data is sent from the Identity Vault to the SIS through the **Zone Integration Server (ZIS)**. The Subscriber channel is fully functional, but in the base configuration it is not used.

You can customize the Subscriber channel to send data changes made in the Identity Vault to SIF, if desired.

For more information on the channels, see the *Novell Identity Manager 3.5.1 Administration Guide*.

Zone

A SIF term. A grouping of SIF-enabled Agents for sharing data. A Zone might be small or large, servicing a school, several schools, or a district. An Agent must register with a Zone. The Zone manages the registered Agents.

Zone Integration Server (ZIS)

A SIF term. A software product that implements the SIF ZIS functionality and can also contain value-added management and configuration tools. A ZIS should be capable of supporting more than one Zone. The term ZIS is often used to mean a Zone.