

Novell DirXML[®] Driver for PeopleSoft^{*}

3.6.2

www.novell.com

IMPLEMENTATION GUIDE

January 15, 2004



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.

You may not export or re-export this product in violation of any applicable laws or regulations including, without limitation, U.S. export regulations or the laws of the country in which you reside.

Copyright © 2004 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.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

DirXML Driver for PeopleSoft Implementation Guide

[January 15, 2004](#)

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

ConsoleOne is a registered trademark of Novell, Inc. in the United States and other countries.

DirXML is a registered trademark of Novell, Inc. in the United States and other countries.

eDirectory is a trademark of Novell, Inc.

NDS is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

Nsure is a trademark of Novell, Inc. in the United States and other countries.

Contents

- About This Guide** **7**
- 1 Introducing the DirXML Driver for PeopleSoft** **9**
 - Benefits 9
 - Driver Features 10
 - DirXML 2.0 New Features 11
 - Driver Components 12
 - Driver Configuration. 12
 - Driver Shim 13
 - Event Server 13
 - PeopleSoft Service Agent. 13
 - Publishing to eDirectory 13
 - Event Descriptions 14
 - Subscribing from eDirectory 16
- 2 Installing the Driver** **17**
 - Driver Requirements 17
 - Configuration Information. 17
 - Importing the Driver Configuration 19
 - Activating the Driver 20
 - Upgrading the Driver 20
- 3 Configuring Your PeopleSoft System** **21**
 - Installing the PeopleSoft Service Agent 21
 - Installing the PSA on PeopleSoft 7.5 21
 - Installing the PSA on PeopleSoft 8.1 25
 - Running the Message Agent Test Program 29
 - Testing the DIRXML_TRANS01 PeopleSoft Message Agent. 29
 - Testing the DIRXML_SCHEMA01 PeopleSoft Message Agent. 30
 - Testing the DIRXML_SCHEMA01_UPDATE PeopleSoft Message Agent 31
 - Testing the DIRXML_SCHEMA01_QUERY PeopleSoft Message Agent. 31
 - Installing the Event Server 32
 - Manually Installing and Configuring the Event Server 32
- 4 Troubleshooting the Driver** **37**
 - Resolving Errors 37
 - The Event Server Will Not Load 37
 - The Driver Will Not Start 37
 - The Driver Is Not Communicating with the Event Server 37
 - The Event Server Receives Message Agent Errors 37
 - Attributes Do Not Get Refreshed on the Data Map Object 38
 - Driver Only Appears to Process Transactions 38
 - Data Does Not Show up in eDirectory on the Publisher Channel. 38
 - Data Does Not Update in PeopleSoft on the Subscriber Channel 38
 - No Transactions Are Coming Across the Publisher Channel. 39
 - Transactions Do Not Get Placed in the PeopleSoft Queue 39

No Data Is Returned When Running the Message Test Program	39
Transactions Are Left in Selected State and Not Processed	39
Receiving Errors on the Publisher Channel When Processing a Transaction	39
Message Agent Relationships Not Functioning	40

About This Guide

This document is for network administrators, consultants, and PeopleSoft* administrators who are using PeopleSoft 7.5 or 8.1.

The DirXML[®] Driver for PeopleSoft, subsequently referred to as the driver, is designed to share data between Novell[®] eDirectory[™] and PeopleSoft. This configurable solution gives organizations the ability to increase productivity and streamline business processes by integrating PeopleSoft and eDirectory.

This driver connects to PeopleSoft via the Message Agent, a PeopleTools interface. The driver can be configured to work with any PeopleSoft module.

Additional Documentation

For documentation on using the DirXML drivers, see the [DirXML Documentation Web site \(http://www.novell.com/documentation/lg/dirxml20/index.html\)](http://www.novell.com/documentation/lg/dirxml20/index.html).

Documentation Updates

For the most recent version of this document, see the [DirXML Documentation Web site \(http://www.novell.com/documentation/lg/dirxml20/index.html\)](http://www.novell.com/documentation/lg/dirxml20/index.html)

Documentation Conventions

The term *driver* refers to all components of the driver for PeopleSoft and not to any one particular component. All references to NPS in this document represent Novell/PeopleSoft and not Novell Portal Services.

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

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

1

Introducing the DirXML Driver for PeopleSoft

A PeopleSoft application is composed of two primary pieces:

- ◆ PeopleTools, which is an infrastructure
- ◆ One or more modules

Example modules are Human Resources Management System (HRMS), Financials, Student Administration, and Customer Resource Management.

DirXML[®] Driver for PeopleSoft interacts with PeopleSoft at the PeopleTools level. By using object definitions within the PeopleSoft modules, along with a collection of preconfigured objects, you enable PeopleSoft so that you can do the following:

- ◆ Trap events in any PeopleSoft module.
- ◆ Expose trapped events to Novell[®] eDirectory[™].
- ◆ Collect and process the exposed data.
- ◆ Expose the data through a PeopleTools interface.

The driver creates an automated link between PeopleSoft and eDirectory. As new records are added, modified, or deactivated (disabled) in PeopleSoft, network tasks associated with these events can be automatically processed.

For example, when hires, re-hires, terminations, and employee updates occur within the Human Resources module, resultant tasks are often created for the Information Services department. These tasks might include setting up, modifying or disabling an eDirectory user account, creating an e-mail account, or creating a domain account. By using the driver, you can automate and maintain these and other business processes.

Benefits

As the following examples illustrate, the driver enables you to automate and maintain business processes.

- ◆ Automatically create an eDirectory account when an individual is hired or a student is admitted.
- ◆ Automatically delete or deactivate eDirectory accounts when an employee terminates.
- ◆ Synchronize bidirectional data between PeopleSoft and eDirectory.
- ◆ Maintain accurate and consistent eDirectory IDs.
- ◆ Define password policies (for example, a birthdate, social security number, and first and last name combinations).
- ◆ Via eDirectory, seamlessly allow integration between PeopleSoft and multiple applications (for example, eDirectory, Lotus Notes*, Netscape*, Exchange, Active Directory*).

- ◆ Create other eDirectory objects associated with a PeopleSoft object (for example, account codes or department records).
- ◆ Synchronize attributes between PeopleSoft and eDirectory.
- ◆ Synchronize data from eDirectory to PeopleSoft.

The driver includes the following:

- ◆ The Event Server program
- ◆ The driver shim
- ◆ The PeopleSoft Service Agent (PSA)
- ◆ A driver configuration file

You can configure the PSA and the driver objects to enhance your organization's business processes. Before installing and configuring the driver, you evaluate and define those processes. During installation, you configure the driver's policies to automate these processes wherever possible.

For more information about DirXML, refer to the *DirXML Administration Guide* (<http://www.novell.com/documentation>).

Driver Features

The driver provides the following features:

- ◆ Support for multiple PeopleTools.

A separate Event Server program is available for each PeopleTools release. This program connects to the driver shim in a Windows* socket.

Remote processing between the shim and the Event Server is also possible. This means that the Event Server can run on a different machine than the driver shim.

- ◆ Enhanced error processing.

The Event Server can run as a DOS window on a Windows NT* machine. The screen shows debug messages as an event is processed. The information includes the status of the transaction displayed on the Event Server. This enables you to determine exactly which transaction is being processed.

- ◆ Enhanced DSTRACE capability.

DSTRACE messaging includes a detailed description of the event and event processing.

- ◆ DirXML query support.

You can apply a Matching policy for both the Publisher and Subscriber channels. If an association between the eDirectory User object and the PeopleSoft employee does not exist, the DirXML engine executes a Matching policy and requests a query to merge attributes between the two objects.

You can also query PeopleSoft for attribute values needed for data manipulation or processing within a policy without having to copy the data to eDirectory.

- ◆ Command line parameters for the Event Server.

Command line parameters define how the Event Server is to be executed, how to connect to the PeopleSoft DirXML driver shim, and whether the Event Server should be installed as a service. The parameters also designate the path of the PeopleSoft client environment. For more information, refer to **“Installing the PeopleSoft Service Agent” on page 21.**

- ◆ Auto-start ability.

You can set the driver to auto-start by installing the Event Server as a Windows service and setting the auto-start flag on the driver. For more information, refer to “[Manually Installing and Configuring the Event Server](#)” on page 32.

- ◆ Support for schema query.

The driver can query the eDirectory and PeopleSoft schemas for attributes that may be used in the Mapping policy or query PeopleSoft for objects defined as policy objects on the Driver object.

DirXML 2.0 New Features

DirXML 2.0 includes the following new features. For more information, refer to the *Nsure Identity Manager 2 Administration Guide* (<http://www.novell.com/documentation/lg/dirxml20/admin/data/alxnk27.html>).

Password Management

The new password management framework includes the following benefits:

- ◆ New Password Policies let you create rules for passwords and assign them to users, containers, or the whole eDirectory tree. You can enable Universal Password, which lets you enforce detailed criteria for passwords and allows for special characters.
- ◆ Password Synchronization 2.0 is now cross-platform, and it lets you enforce your Password Policies across connected systems. New notification templates let you automatically send messages to users about their password synchronization status.
- ◆ Using Password Policies, you can also provide Forgotten Password Self-Service and Reset Password Self-Service to your users. These new features can help you reduce help desk calls. Notification templates are also included for automatically sending forgotten password and password hint messages to users.

Policy Builder Interface and DirXML Script for Creating Policies

For the most common tasks, you can now use the new Policy Builder interface to create policies for your driver without writing XSLT code. The Policy Builder helps you set up policies using the new DirXML Script.

Role-Based Entitlements

For many drivers, Role-Based Entitlements is an option in the sample driver configuration that you can choose when importing the driver.

Role-Based Entitlements let you grant entitlements on connected systems to a group of Novell® eDirectory™ users. Using Entitlement Policies, you can streamline management of business policies and reduce the need to configure your DirXML drivers.

Novell Nsure Audit

Novell Nsure™ Audit is a centralized, cross-platform auditing service. It collects event data from multiple applications across multiple platforms and writes the data to a single, non-repudiable data store. Nsure Audit is also capable of creating filtered data stores. Based on criteria you define, Nsure Audit captures specific types of events and writes those events to secondary data stores.

Global Configuration Values

Global configuration values (GCVs) are new settings that are similar to driver parameters. Global configuration values can be specified for a driver set as well as an individual driver. If a driver does not have a value for a particular GCV, the driver inherits the value for that GCV from the driver set.

GCVs allow you to specify settings for new DirXML features such as Password Synchronization, 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. You can refer to these values in a policy to help you customize your driver configuration.

Driver Heartbeat

The DirXML engine now accepts driver heartbeat documents from drivers, and drivers can be configured to send them.

Driver Components

The driver uses the following components:

- ◆ PeopleSoft Service Agent (PSA)
- ◆ Driver Shim
- ◆ XML Import File
- ◆ Event Server
- ◆ PeopleSoft Message Agent (delivered through PeopleSoft)

Component	For PeopleSoft 7.5	For PeopleSoft 8.1
PSA	DIRXML_DRIVER 3_6 PSA.EXE	DIRXML_DRIVER 3_6 PSA.EXE
Driver shim	NPSSHIM.DLL	NPSSHIM.DLL
XML Import File	PEOPLESOFT36.XML	PEOPLESOFT36.XML
Event Server	NPS75EVENTSERVER.EXE	NPS81EVENTSERVER.EXE

Driver Configuration

The driver is responsible for reporting object change events that it receives from PeopleSoft to the DirXML engine. The driver consists of both an Event Server program and the NPSSHIM.DLL file.

The driver contains both a Publisher channel and a Subscriber channel. The Subscriber channel receives XML-formatted eDirectory events from the DirXML engine. The driver then converts these documents to an appropriate data stream, forwards them to the Event Server, and updates PeopleSoft via the Message Agent interface.

The Publisher channel receives XML-formatted PeopleSoft events from the Event Server and submits them to the DirXML engine for publication into eDirectory. The engine processes the document by sequentially applying all configured policies based on standard DirXML driver process flow.

Each policy performs a transformation on the XML document. The engine processes the event document according to the policies and the filter found in the Driver Publisher object.

Driver Shim

The driver shim handles communication between the event server and the DirXML engine. For this version of the driver, the driver shim is named NPSSHIM.DLL.

Event Server

The Event Server is a standalone executable process. It establishes and maintains connectivity between the driver shim and the PeopleSoft Message Agent.

The Event Server communicates with the driver via a sockets interface and communicates with the Message Agent via proprietary PeopleSoft Message Agent APIs.

The Event Server is a bidirectional component with Publisher and Subscriber channels. The Publisher channel polls the Message Agent for transactions. When transactions are available, the Event Server reads the record data via the Message Agent and transforms the proprietary format of the PeopleSoft records into an XML-formatted representation. The XML records are sent to the driver for submission to eDirectory.

Likewise, the Subscriber channel receives XML-formatted eDirectory events from the driver and transforms them into the PeopleSoft proprietary format. The event data is then sent to the Message Agent for publication into the PeopleSoft database.

PeopleSoft Service Agent

The PeopleSoft Service Agent (PSA) is software that you receive as a PeopleSoft project. A PSA is a collection of objects. This collection customizes the PeopleSoft module that the objects are applied to.

You can install the PSA components and then configure them for any PeopleSoft application.

The PSA serves three purposes:

- ◆ Traps events as they occur in PeopleSoft
- ◆ Places a transaction for trapped events into the worklist
- ◆ Exposes the transaction (which is joined with relevant, current data) to the driver via the Message Agent

The PSA includes a sample of all objects necessary in PeopleSoft so that any PeopleSoft module can have connectivity to and from the Event Server.

Publishing to eDirectory

The publishing of events to eDirectory begins with updates in PeopleSoft. As these updates occur, transactions are placed in a worklist queue. These events are then made available to eDirectory through the Event Server.

As the Event Server receives transactions from PeopleSoft, it transforms the data into an XML document and passes the document to the driver shim. The driver shim then passes the document to the DirXML engine, which processes the transaction in eDirectory based on the policies as defined in the PeopleSoft DirXML Driver.

The transaction in the worklist is also updated with either a *worked* or *error* status. As the status is being updated in PeopleSoft, additional PeopleCode can be processed to trigger e-mail notifications to users in a defined PeopleSoft workflow role. The e-mail message communicates information regarding both successful transactions and transactions that receive errors.

Event Descriptions

As updates or actions occur within PeopleSoft, workflow triggers are executed. These triggers place transactions in a PeopleSoft worklist queue. Each transaction is given key fields that uniquely identify the transaction to the driver.

One of these key fields is the Event Name field. Event Names are assigned based on a PeopleSoft action. The driver monitors the queue, checking for transactions that meet the criteria for processing.

The criteria for processing include the following:

- ◆ The transactions have the status of 0.
- ◆ Action, date, and time are less than or equal to current date and time.

After receiving the event, the Event Server converts the event into XML and sends the XML to the driver shim. This shim passes the XML document to the DirXML engine. The engine then applies the appropriate policies for that event type (based on the EventName field). The driver changes the XML to eDirectory commands and sends the events to eDirectory.

The delivered solution supports the following workflow events. By default, an event in PeopleSoft equates to an XML functional event or document.

PeopleSoft Event	Typical PeopleSoft Action	XML Functional Event
ADD	New hire, new student, new account code, new record.	Add
UPD	Any change or modification to data. For example, a field on a panel is modified.	Modify
DIS	Termination. A record becomes inactive or disabled.	Delete

To trigger the appropriate event for the type of XML document that is to be generated, you must configure PeopleSoft appropriately. Without proper configuration and review of your business processes, a termination (action) could trigger a UPD event instead of a DIS event. The resulting XML document would be a Modify document instead of a Delete document.

The delivered driver configuration identifies two fields (Email ID and Description) on the subscriber channel. These fields synchronize data from eDirectory to PeopleSoft. Additional fields can be subscribed to PeopleSoft from eDirectory, based on the needs of your organization. For Subscriber channel information, refer to [“Subscribing from eDirectory” on page 16](#).

ADD

The ADD event is generally triggered when a new record is added. For the HR module, this is when an employee is hired. When a new Hire record is created within PeopleSoft, the following steps occur:

1. A workflow definition is triggered.
2. An ADD transaction is written to the worklist.
3. The driver reads the worklist to obtain a transaction, which is joined with user-specified data.
4. The driver transforms the data stream into an XML document.
5. The driver passes the document to the DirXML engine.

6. The engine applies the policies that have been configured.
7. The engine creates an object.

The SSCreate policy is configured as a sample Create policy. You use this policy to create the User object. You can define the eDirectory Common Name, Initial Login Password, and other Create policies within this policy.

Depending on your business requirements, you can apply additional configurations to the PSA to expose additional data elements. You can also configure the policies to meet business objectives.

During the ADD event processing, an association is made between the employee's PeopleSoft record and the eDirectory object. During the creation of the object, an association is made between the eDirectory object and the unique key from PeopleSoft. For the HR module, the EmplID field is used as the key field and saved in the association attribute on the User object.

When an object is created, various attributes or data elements defined in PeopleSoft could be passed through the interface and used to place the object. The Placement policy on the Driver object would use these values.

UPD

UPD events are generally triggered when data on a specified PeopleSoft field is modified. A workflow is triggered through the use of PeopleCode, and a UPD event is written to the worklist. Through the same process as the ADD transaction, the driver retrieves the transaction and applies the appropriate policies. In the delivered solution, the eventXform policy is used to apply these events. The policies might vary, based on the individual needs of an organization.

To determine the PeopleSoft fields that should be synchronized and used as part of the policies within eDirectory, analyze your business processes.

For example, a user's location, department, or company can determine the location of that individual's User object within eDirectory. If one of these fields is updated from PeopleSoft, the eDirectory User object must be moved to the appropriate container, based on the new information.

PeopleSoft fields can also be stored in eDirectory as attributes. Telephone numbers, department, preferred names, business titles, and locations are some common fields that can be shared between the two systems. As you analyze your business processes and needs, you determine the necessary data mappings between the systems.

DIS

The DIS event is generally triggered when records are disabled. Many different actions within PeopleSoft represent a disabled process. In the HR module, this event occurs when an employee terminates. When a termination record is created within PeopleSoft, a workflow is triggered and a DIS entry is written to the worklist.

IMPORTANT: If a DIS transaction is triggered, a Delete XML document is the result.

An organization can choose to delete, disable, or disable and move the eDirectory account. As with the modify/update process, the eventXform policy is delivered to show how a Delete XML document is transformed into a Modify document.

NOTE: By default, when the Publisher channel processes any event, the eDirectory ID and eDirectory DN (distinguished name) are updated in PeopleSoft. You can disable this process by updating the driver's XML properties.

Subscribing from eDirectory

PeopleSoft is the authoritative process for ADD and DIS (delete or disable) events. Therefore, the default PeopleSoft DirXML Driver only allows updates to PeopleSoft on the Subscriber channel.

Because PeopleSoft is a relational database (instead of a hierarchical database), the Move and Rename processes generally do not apply. If these processes occur in eDirectory against an object that the PeopleSoft DirXML Driver is subscribed to, the driver converts them to a Modify event. The driver does this to update the eDirectory ID and eDirectory Distinguished Name in PeopleSoft, provided the update process is not disabled.

The driver subscribes to Modify events that occur within eDirectory. PeopleSoft is designed to be the authoritative owner of data in the delivered solution. Therefore, if a User object is created in eDirectory, an employee record in PeopleSoft will not be created with the default configuration.

The data elements that are being subscribed from eDirectory to PeopleSoft must

- ◆ Be found on the Subscriber channel filter
- ◆ Exist on the DIRXML_SCHEMA01_UPDATE Message Definition and on the DIRXML_SCHEMA01 Message Definition in PeopleSoft (or equivalent definitions)

When the driver is notified of a modification to an attribute selected in its Subscriber filter, the driver sends the event to the Event Server. The Event Server formats the event and then updates PeopleSoft by communicating through the PeopleSoft Message Agent. Data received from eDirectory can be placed on a panel or page inside PeopleSoft.

Because this data is being updated and maintained outside of PeopleSoft, it should typically be set to Display Only from within PeopleSoft. It is not advisable to allow the same data element to be modified inside and outside of PeopleSoft. For this to occur, the data element would have to be defined in both the Publisher and the Subscriber channel on the driver.

You can update data in PeopleSoft to an existing PeopleSoft record. To do this, you must write appropriate PeopleCode so that data elements updated on the DIRXML_STAGE01 record definition are also copied to the desired record within the PeopleSoft environment.

2

Installing the Driver

This section contains information to help you:

- ◆ Understand “[Driver Requirements](#)” on page 17.
- ◆ Review “[Configuration Information](#)” on page 17
- ◆ Complete procedures for “[Importing the Driver Configuration](#)” on page 19.
- ◆ Understand the process of “[Activating the Driver](#)” on page 20.

Driver Requirements

- ◆ Novell Nsure™ Identity Manager 2
- ◆ Windows NT* 4.0 with Service Pack 5 or higher, or Windows 2000
- ◆ The appropriate version of PeopleTools and Tuxedo* Application Server

PeopleSoft Platform	PeopleTools Version
PeopleSoft 7.5	PeopleTools 7.57 or higher
PeopleSoft 8.1	PeopleTools 8.17 or higher

Configuration Information

As you import the driver configuration file, you will be prompted for the following information.

Parameter Name	Parameter Description
Driver name	The actual name you want to use for the driver.
Active Users Container	The name of the Organizational Unit object where Active users are placed.
Inactive Users Container	The name of the Organizational Unit where Inactive users are placed.
Active Employees Group	The name of the Group Object to which Active “Employee” users are added.
Active Managers Group	The name of the Group Object to which Active “Manager” users are added.
Event Server Host Name and Port	The host name or IP address and port number of the computer where the PeopleSoft Event Server.

Parameter Name	Parameter Description
PeopleSoft Connection String	The host name or IP address and port number for connecting to the appropriate PeopleSoft Application server. This is typically referred to as the PeopleSoft application server connection string. The default port is 9000.
PeopleSoft User ID	The PeopleSoft User ID the driver uses for authentication to PeopleSoft.
PeopleSoft User Password	The PeopleSoft User password this driver uses for authentication to PeopleSoft.
Configure Data Flow	Dataflow can be configured to one of the following options: <ul style="list-style-type: none"> ◆ Bidirectional: PeopleSoft and eDirectory are both authoritative sources of the data synchronized between them. ◆ PS-to-eDirectory: PeopleSoft is the authoritative source. ◆ eDirectory-to-PS: eDirectory is the authoritative source.
Install Driver as Remote/Local	Configure the driver for use with the Remote Loader service by selecting the Remote option, or select Local to configure the driver for local use. If Local is selected, you can skip the remaining parameters.
Remote Host Name and Port	Specify the host name or IP address and port number for where the Remote Loader service has been installed and is running for this driver. The default port is 8090.
Driver Password	The driver object password is used by the Remote Loader to authenticate itself to the DirXML server. It must be the same password that is specified as the driver object password on the DirXML Remote Loader.
Remote Password	The Remote Loader password is used to control access to the Remote Loader instance. It must be the same password that is specified as the Remote Loader password on the DirXML Remote Loader.

The additional driver parameters are set to default values during the import process, but they can be modified in iManager (by clicking the Driver Configuration tab on the driver object.)

Parameter Name	Description
Event Server Port Number	The IP port number on which the driver Event Server component listens for connection from the driver shim.
Data Record ID Field	The name of the field in the Data Schema CI that uniquely identifies a PeopleSoft object. The value in this field in the Data Schema that uniquely identifies a PeopleSoft object. The value in this field is used as the DirXML object association identifier. The default field name is DIRXML_ASSOC_ID.
Schema Data Subscriber Activity	The name of the PeopleSoft Activity object that defines the data that can be synchronized from eDirectory to the PeopleSoft application. The default Subscriber Activity is DIRXML_SCHEMA01_UPDATE. You should ensure that the Message Definition within this Activity matches the name of the Activity.
Schema Data Query Activity	The name of the PeopleSoft Activity object that defines the data that the driver reads from the PeopleSoft application. This data is usually the subset of application data elements contained within the Schema Data Publisher Activity. The default Query Activity is DIRXML_SCHEMA01_QUERY. You should ensure that the Message Definition within this Activity matches the name of the Activity.

Parameter Name	Description
Queue Poll Interval (seconds)	<p>The number of seconds the driver waits between attempts to process transaction records. This poll interval is only applied when no transactions are available for processing.</p> <p>The default Poll Interval is 5 seconds.</p>
Queue Retrieval Limit	<p>This parameter specifies how many transactions will be retrieved by the driver from the PeopleSoft worklist queue each time the driver accesses the Transaction Access Activity.</p> <p>The default value is 5.</p>
Transaction Access Activity	<p>The name of the PeopleSoft Activity object that defines the set of fields required for the DirXML Transaction interface. The set of fields in the specified transaction activity must contain the same fields and keys identified in the default transaction activity in order for the driver to work.</p> <p>The default Transaction Activity is DIRXML_TRANS01.</p> <p>You should ensure that the Message Definition within this Activity matches the name of the Activity.</p>
Schema Data Publisher Activity	<p>The name of the PeopleSoft Activity object that defines the set of data to be synchronized from the PeopleSoft application to eDirectory.</p> <p>The default Publisher Activity is DIRXML_SCHEMA01.</p> <p>You should ensure that the Message Definition within this Activity matches the name of the Activity.</p> <p>NOTE: Additional transaction control key fields (field names are all capitalized) contained within the Message Definition are used for processing purposed only and are not synchronized.</p>

Importing the Driver Configuration

The Create Driver Wizard helps you import the basic driver configuration file. 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.

1 In Novell iManager, click DirXML Utilities > Create Driver.

2 Select a 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 Import a Driver Configuration from the Server, then select PeopleSoft36.xml.

The driver configuration files are installed on the Web server when you install DirXML. During the import, you will be prompted for the driver's parameters and other information. Refer to "[Configuration Information](#)" on page 17 for more information.

4 Specify the driver's parameters, then click OK to import the driver.

When the import is finished, you can define security equivalences and exclude administrative roles from replication.

The driver object must be granted sufficient eDirectory rights to any object it reads or writes. You can do this by granting Security Equivalence to the driver object. The driver must have Read/Write access to users, post offices, resources, and distribution lists, and Create, Read, and Write rights to the post office container. Normally, the driver should be given security equal to Admin.

- 5 Review the driver objects in the Summary screen, and then click Finish.

Activating the Driver

DirXML and DirXML drivers must be activated within 90 days of installation, or they will not run. At any time during the 90 days, or afterward, you can choose to activate DirXML products to a fully licensed state.

To activate your driver, you should:

- ◆ Purchase DirXML licenses
- ◆ Generate a Product Activation Request
- ◆ Submit the Product Activation Request
- ◆ Install the Product Activation Credential received from Novell

For more information about completing these tasks, refer to [Activating Your DirXML Product \(http://www.novell.com/documentation/lg/dirxml20/index.html\)](http://www.novell.com/documentation/lg/dirxml20/index.html).

Upgrading the Driver

For detailed instructions regarding how to upgrade the 3.51 or 3.6 driver to 3.6a, you should refer to TID #2963871 at <http://support.novell.com> (<http://support.novell.com>).

3

Configuring Your PeopleSoft System

Use the information in the following section to configure your PeopleSoft system to share and synchronize data with Novell® eDirectory™.

- ♦ [“Installing the PeopleSoft Service Agent” on page 21](#)
- ♦ [“Running the Message Agent Test Program” on page 29](#)
- ♦ [“Installing the Event Server” on page 32](#)

Installing the PeopleSoft Service Agent

The process for installing the PSA is different between versions of PeopleSoft. You can use the information contained in this section to install the PSA for your PeopleSoft application version.

- ♦ [“Installing the PSA for PeopleSoft 7.5” on page 22](#)
- ♦ [“Installing the PSA on PeopleSoft 8.1” on page 25](#)

NOTE: Any references to specific paths in these sections represent the defaults indicated during the installation procedures. Apply the necessary changes if applicable.

Installing the PSA on PeopleSoft 7.5

If you are using PeopleSoft 7.5, you should complete the following tasks to install the PSA:

- ♦ [“Extracting the PSA Files” on page 21](#)
- ♦ [“Installing the PeopleSoft Service Agent” on page 22](#)
- ♦ [“Copying the Novell Integration Project into a Target Database” on page 23](#)
- ♦ [“Establishing Security to PSA Objects in PeopleSoft 7.5” on page 24](#)

Extracting the PSA Files

The PSA project comes as a self-extracting project. To extract the project:

- 1** From the location you specified when you installed the components for PeopleSoft, run DIRXML_DRIVER_36.EXE. The default location is Novell\NDS\PS.
- 2** Click Next, select the destination directory, then click Next to begin extracting the PSA files.
- 3** Click Finish to close the extraction process.

The PSA is now ready to be installed into your PeopleSoft environment.

Installing the PeopleSoft Service Agent

Before you can install and configure the PeopleSoft objects, you need access to a PeopleSoft user ID and password with Administrator or appropriate developer rights. You can create a unique user ID and password for implementing these objects.

To apply the delivered PSA set of objects, follow PeopleSoft's standard upgrade procedures. For information on PeopleSoft upgrade procedures, refer to PeopleSoft documentation.

Installing the PSA for PeopleSoft 7.5

Data Mover is an application tool used to import external PeopleSoft objects into an existing PeopleSoft 7.5 database. Using the standard PeopleSoft upgrade process, you use Data Mover to move the DirXML for PeopleSoft projects into an organization's PeopleSoft environment.

A Data Mover script (DMS) is provided with the driver. This script imports new and modified objects into the data dictionary tables. You should only run this script on a refreshed Application Update Database (AUD). Refreshing the database resolves any conflicts between the delivered objects and any objects that already exist in the AUD database.

WARNING: If the DMS is run against a database other than the AUD, it should fail or overwrite existing data. If you run the script directly against the PeopleSoft development database, it will corrupt the database.

To ensure that the DMS files were copied during the component installation, check the C:\PSA directory. This is the default directory that the installation program uses. If you altered this setting during the installation, check the directory that you entered. The following are default steps to import the external PeopleSoft objects:

- 1 From the Start menu, click Programs > PeopleSoft > Data Mover.
- 2 Log in to the AUD database.
- 3 Select File > Open.
- 4 Browse to the appropriate file for your installation > select the DMS script with the appropriate release number. (For example, DIRXML_DRIVER_PSA_36IMP.DMS.)

The release number should be the same as the release number of the PeopleSoft database.

- 5 In the Data Mover syntax dialog box, verify that the input and log files are pointing to the directory containing the data (.DAT) files on the local drive.

The following is an example of the delivered syntax:

```
SET OUTPUT c:\PSA\AU_CHECK.DAT;  
  
SET INPUT c:\PSA\PSA-PS75\DIRXML_DRIVER_PSA_36IMP.DAT;  
  
SET LOG c:\PSA\DIRXML_DRIVER_PSA_36IMP.LOG;  
  
SET NO TRACE;  
  
SET IGNORE_DUPS;  
  
EXPORT AU_CHECK;  
  
IMPORT *;
```

- 6 Select File > Run Script.

From the report in the bottom status panel, verify the import was successful. If the import fails, make sure the input and output paths point the proper directories.

- 7 Close Data Mover.

Copying the Novell Integration Project into a Target Database

Your target database contains the system catalog, data dictionary structures, and sample application data. When moving from an AUD database to the target database, objects from the data dictionary in the AUD are copied to the data dictionary of the target database.

After the copy is complete, you must build the database. The build creates the physical tables or views in the underlying database so that the objects are accessible. You should be logged in as an administration user when completing these procedures.

We recommend the following:

- ◆ Use a clean copy of a target database for this procedure.
If you apply a copy to a database with production data, do a copy and compare before you apply the copy.
- ◆ Complete an installation and test that installation against a target database.
Do this before you change the configuration and before you apply the database to a development or production environment.

To copy the project into a target database:

- 1** From the Start menu, click Programs > PeopleSoft > PeopleTools.
- 2** Log in to the AUD database.
- 3** Click Go > PeopleTools > Application Designer > File > Open.
- 4** Click Object Type, then click Project.
- 5** Click Selection Criteria, Name, then type DIRXML_DRIVER_PSA_36, then click Select.
- 6** Click the project, then click Select.
- 7** Click Tools > Upgrade > Copy.
- 8** Type the target database name (the project will be copied to this database), operator ID, and password, then click OK.
Be sure to select all objects with the project, including the project itself.
- 9** Check the Export Project box to export the objects to the target database, then click Copy.
All objects are copied from the AUD to the target DMO database.
- 10** Close the Application Designer window.

Building Project Record Definitions

After you have imported the project into the PeopleSoft database, you should build project record definitions and then build project views.

- 1** From the Start menu, click Programs > PeopleSoft > PeopleTools.
- 2** Log in to the target database using an administrator username that has administrative and development rights.
- 3** In the Application Designer, click File, then click Open.
- 4** Select the DIRXML_DRIVER_PSA_36 project, and click Open.
- 5** Click Build > Project.
- 6** From Build Options, click Create Tables and Execute SQL Now.

- 7** Click Build to create sample project tables. After project tables are created, click Close to close the Build Progress window.

NOTE: You must create project tables before creating the views. Views are created using information from table fields.
- 8** From the Application Designer, select Build > Project.
- 9** From Build Options, click Create Views and Execute SQL Now.
- 10** Click Build to create the sample project views. After views are created, click Close to close the Build Progress window.

You should now establish roles and initiate workflow. Refer to [“Establishing Roles and Initiating Workflow” on page 24](#).

Establishing Roles and Initiating Workflow

A role is a class of users who perform the same type of work. Business policies typically determine what role performs what activity. The eDirectory administrator has been designated as the role performing the ADD, UPD, and DIS events. This means that workflow must be activated and a role user must be assigned to the eDirectory administrator role. To activate a workflow, complete the following:

- 1** Click Go > PeopleTools > Workflow Administrator.
- 2** Click Use > Workflow System Defaults > Defaults.
- 3** Check the Worklist Active box.
- 4** If using e-mail and other electronic forms, check the applicable boxes.
- 5** Ensure that a default role user, typically WF Admin, is also assigned, and click Save. If workflow is not currently being used by other processes, change this to the role of eDirectory Administrator.

You should now assign security rights to the new objects (see [“Establishing Security to PSA Objects in PeopleSoft 7.5” on page 24](#)).

Establishing Security to PSA Objects in PeopleSoft 7.5

You can assign security rights to all new panels and pages that have been added to a project. Use an administrator-type account to establish security.

Applying Security for the New PSA Objects

- 1** Click Go > PeopleTools > Security Administrator.
- 2** Click File > Open. Select the ALLPANLS Operator Class, then click OK.
- 3** Click the Menu Items icon from the left pane.
- 4** Click Insert > Menu Name.
- 5** Double-click DirXML_Administrator_36, click Select All, then click OK.
- 6** Click Select All, then click OK.
- 7** Save. After you save, you can exit the Security Administrator interface.

Verifying Access to PSA Object

After you have applied security to the new PSA objects, you should use the sample application to verify that you can access, change, and save data.

You can test to ensure that transactions are created by entering a new person using the PeopleSoft DirXML® sample application. This example uses Departments, so you will need to create a sample department, and then add a person (assigning him or her to that department) to validate that the application works.

To test sample PeopleSoft applications:

- 1** Connect to the PeopleSoft database as administrator in two tier mode.
- 2** From the Application Designer, select Go > DirXML > DirXML Administrator 36 EP.
- 3** Select Use > DirXML Sample Department.
- 4** Click an empty Department field row to add sample department and description.
- 5** Click Save to add the Department.
- 6** Select Use > DirXML Sample People > Add.
- 7** Enter an ID number for the new person and click OK.
- 8** Enter data into the various fields for this ID and click Save.
(The required fields include: first name, last name, birth date, status, title, and department ID.)
- 9** To validate that an ADD transaction was created, access the eDirectory Worklist Maintenance menu. Click Use > eDirectory Worklist Maintenance > eDirectory Worklist. You can then search for the ID you just created.
- 10** Click the Search button.
- 11** Double-click event you created. Look at the panels and verify that the data you entered on exists.
- 12** Close the PeopleSoft Client.

Now that you have completed the required tasks for installing and configuring the PSA, you should continue with [“Running the Message Agent Test Program” on page 29](#).

Installing the PSA on PeopleSoft 8.1

Complete the following tasks to install the sample project for testing and configuration purposes:

1. [“Extracting the PSA Files” on page 25](#)
2. [“Importing the PSA Project into the PeopleSoft Database” on page 26](#)
3. [“Building Project Record Definitions” on page 26](#)
4. [“Testing Sample PeopleSoft Applications” on page 26](#)

NOTE: Any references to specific paths in these sections represent the defaults indicated during the installation procedures. Apply the necessary changes if applicable.

Extracting the PSA Files

The PSA project comes as a self-extracting project. To extract the project:

- 1** From the location you specified when you installed the components for PeopleSoft, run DIRXML_DRIVER_36.EXE. The default location is Novell\NDS\PS.

- 2 Click Next, select the destination directory, then click Next to begin extracting the PSA files.
- 3 Click Finish to close the extraction process.

The PSA is now ready to be installed into your PeopleSoft environment.

Importing the PSA Project into the PeopleSoft Database

With PeopleSoft 8, projects are now delivered in a cache directory structure, which is similar to the cache structure found within PeopleSoft. With previous versions of PeopleSoft you needed to use the Data Mover script and compare, but this process is no longer necessary.

To import the PSA into the PeopleSoft database:

- 1 Connect to the PeopleSoft database as administrator in two tier mode.
- 2 From the Application Designer, select File > Copy Project From File.
- 3 Click Browse and select the PSA project directory: C:\PSA\PSA-PSA8\.
- 4 Click Copy.
- 5 With all object types selected, click Copy to copy all project components into the PeopleSoft database.

Building Project Record Definitions

After you have imported the project into the PeopleSoft database, you should build project record definitions and then build project views.

- 1 Log into PeopleSoft using an administrator username that has administrative and development rights.
- 2 From the Application Designer, select Build > Project.
- 3 From Build Options, click Create Tables and Execute SQL Now.
- 4 Click Build to create sample project tables. After project tables are created, click Close to close the Build Progress window.
NOTE: You must create project tables before creating the views. Views are created using information from table fields.
- 5 From the Application Designer, select Build > Project.
- 6 From Build Options, click Create Views and Execute SQL Now.
- 7 Click Build to create the sample project views. After views are created, click Close to close the Build Progress window.

Testing Sample PeopleSoft Applications

You can test to ensure that transactions are created by entering a new person using the PeopleSoft DirXML sample application. This example uses Departments, so you will need to create a sample department, and then add a person (assigning him or her to that department) to validate that the application works.

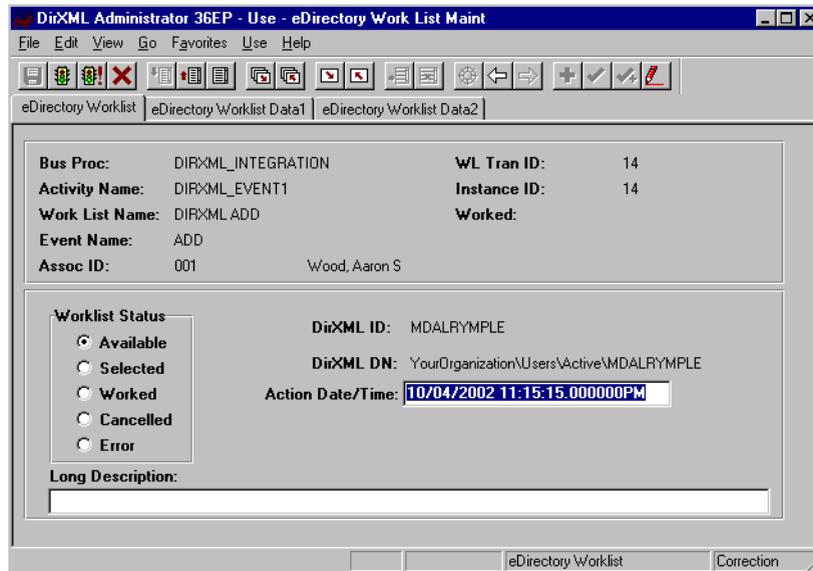
To test sample PeopleSoft applications:

- 1 Connect to the PeopleSoft database as administrator in two tier mode.
- 2 From the Application Designer, select Go > DirXML Administrator36.
- 3 From the DirXML Administrator menu, select Use > DirXML Sample Department.

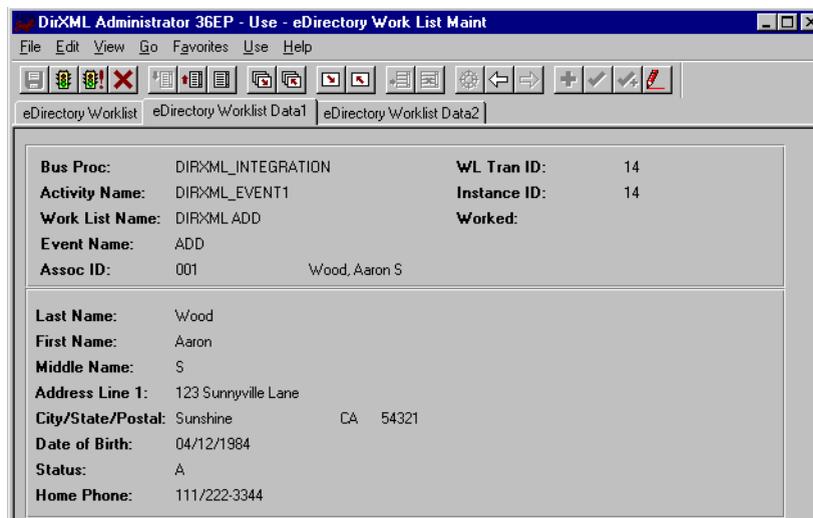
- 4** Click an empty Department field row to add sample department and description.
- 5** Click Save to add the Department.
- 6** From the DirXML Administrator 36 menu, select Use > DirXML Sample People > Add.
- 7** Enter an ID number and click OK.
- 8** Enter data into the various fields for this ID and click Save.
(The required fields include: first name, last name, birth date, status, title, and department ID.)
- 9** To validate that an ADD transaction was created, click Use > eDirectory Worklist Maintenance > eDirectory Worklist.
- 10** Click the Search button to search for the user you added.

Date/Time	Event Name	Assoc ID
2002-09-24-16.40.27.000000	ADD	P0001
2002-09-25-13.40.42.000000	ADD	P0002
2002-09-25-13.44.01.000000	ADD	P0003
2002-09-25-13.48.30.000000	ADD	P0004
2002-09-25-14.02.56.000000	ADD	P0005
2002-09-25-14.07.45.000000	ADD	P0006
2002-09-25-14.19.29.000000	ADD	P0007
2002-09-25-14.32.26.000000	ADD	P0008
2002-09-25-14.32.50.000000	UPD	P0004
2002-09-27-14.41.50.000000	ADD	P1919
2002-09-27-15.50.51.000000	UPD	P1919
2002-10-04-23.15.15.000000	ADD	001

- 11** Double-click the event to view its details.



- 12** After you know that events are configured to process properly, you want to ensure that you can make changes to the data and save it. Click Use > eDirectory Subscriber.
- 13** Click Search and locate the sample person you created.
- 14** Enter a sample value (or modify an existing value) in the phone number field, and click Save.
- 15** Click Use > eDirectory Query > DirXMLQuery01.
- 16** Click Search and double-click your sample person.
- 17** Verify that the phone number you added or modified has changed.



- 18** You can now close the PeopleSoft client.

Now that you have completed the required steps for configuring the PSA, you should refer to the [“Running the Message Agent Test Program” on page 29](#) section.

Running the Message Agent Test Program

You can test the following PeopleSoft Message Agents:

- ◆ DIRXML_TRANS01
- ◆ DIRXML_SCHEMA01
- ◆ DIRXML_SCHEMA01_UPDATE
- ◆ DIRXML_SCHEMA01_QUERY

For detailed troubleshooting information regarding the Message Agent Test System, see [Chapter 4, “Troubleshooting the Driver,”](#) on page 37.

Testing the DIRXML_TRANS01 PeopleSoft Message Agent

DIRXML_TRANS01.IST verifies that the Message Agent is receiving events from the worklist.

NOTE: Script files that have 8 after the script name are for PeopleSoft 8.1. Other IST scripts are for PeopleSoft 7.5.

- 1** Click the C:\PSA\PSA-P75\ or C:\PSA\PSA-P8\ directory.
- 2** Copy the appropriate test program for a message agent to the {PS_HOME}\BIN\CLIENT\WINX86 directory.

PeopleSoft Version	Test Program
PeopleSoft 7.5	PSMTST32v75.EXE
PeopleSoft 8.1	PSMTST32v80.EXE

- 3** Create a shortcut on the desktop to the Message Agent Test program.
- 4** View the program’s properties, and validate that the .IST file is not set to read-only.
- 5** Double-click the test program.
- 6** Click File > Open.
- 7** Click the DIRXML_TRANS01.IST file > Open.
- 8** Ensure that the appropriate connect string, operator ID, and password are listed for connecting to the application server.
- 9** Click Execute!.

If the Message Agent is working properly, a list of processing key fields will be populated.

The following is a sample script:

```
Connect 137.65.147.162:7000,PS,PS;  
StartMessage DIRXML_TRANS01,DIRXML_TRANS01;  
ProcessMessage;  
GetOutputAll;  
Disconnect;
```

In this script the connect string is 137.65.147.162:7000. The User ID is PS, and the password is PS.

You can now close the DIRXML_TRANS01 window. Do not close the Message Agent Test program, as you will use it in the following steps.

Testing the DIRXML_SCHEMA01 PeopleSoft Message Agent

DIRXML_SCHEMA01.IST verifies that all of the fields on the DIRXML_SCHEMA01 business process definition are being passed to the Message Agent.

- 1 Double-click the message test program.
- 2 Click File > Open.
- 3 Click the DIRXML_SCHEMA01.IST file > Open.
- 4 Verify that the appropriate connect string, operator ID, and password are listed.
- 5 Modify the INSTANCEID, INSTSTATUS, ACTIVITY NAME, and EVENTNAME fields.

You can find this information by searching for the person and viewing the DIRXML_SCHEMA01 panel on the eDirectory Work List Maintenance Search panel shown below:

Date/Time	Event Name	Assoc IC
2002-09-24-16:40:27.000000	ADD	P0001

These fields must correspond to the event that you are trying to receive the output data for. Copy and paste appropriate values from the output from the previous test. This output was generated from testing DIRXML_TRANS01.IST.

- 6 Click Save.
- 7 Click Execute!

If the Message Agent is working properly, all the fields that are populated in PeopleSoft will be populated in the output.

The following is a sample script:

```
Connect 137.65.147.182:7000,PS,PS;
StartMessage DIRXML_SCHEMA01,DIRXML_SCHEMA01;
Setfield BUSPROCNAME=DIRXML_INTEGRATION;
Setfield ACTIVITYNAME=DIRXML_EVENT1;
Setfield EVENTNAME=ADD;
Setfield WORKLISTNAME=DIRXML_ADD;
Setfield INSTANCEID=5;
Setfield INSTSTATUS=0;
Setfield COMMENTSHORT=Selected;
ProcessMessage;
```

```
GetOutputall;  
Disconnect;
```

- 8 Close the log files.

Testing the DIRXML_SCHEMA01_UPDATE PeopleSoft Message Agent

DIRXML_SCHEMA01_UPDATE.IST tests updates to the PeopleSoft record DIRXML_SCHEMA01_UPDATE. This process tests events that come from eDirectory rather than events that are sent from PeopleSoft.

- 1 Double-click the message test program.
- 2 Click File > Open.
- 3 Click the DIRXML_SCHEMA01_UPDATE.IST file > Open.
- 4 Verify that the appropriate connect string, operator ID, and password are listed.
- 5 Modify the ASSOC ID, NDS ID, NDS CONTEXT, and Description fields.

These fields must correspond to the employee that you are trying to update.

- 6 Click Save.
- 7 Click Execute!

If the Message Agent is working properly, the eDirectory Subscriber page (the NDS Data Update panel for PeopleSoft 7.5) will be populated with the appropriate eDirectory ID, Distinguished Name, e-mail, and Description fields for the selected employee.

The following is a sample script:

```
Connect 137.65.139.178:7000,PS,PS;  
StartMessage DIRXML_SCHEMA01_UPDATE,DIRXML_SCHEMA01_UPDATE;  
SetField Assoc ID=8964;  
SetField NDS ID=GHORNBE;  
SetField NDS CONTEXT=NCS.PRIV.NOVELL;  
SetField Description=Updated;  
SetField email=unknown;  
ProcessMessage;  
GetOutputall;  
Disconnect;
```

- 8 Close the log file.

Testing the DIRXML_SCHEMA01_QUERY PeopleSoft Message Agent

DIRXML_SCHEMA01_QUERY.IST tests the query function for the driver from PeopleSoft. This process tests the ability to retrieve data from PeopleSoft based on ASSOC ID.

The Output fields defined on the Query object need to be equivalent to those that are defined on the DIRXML_SCHEMA01 object.

The primary difference between these two objects is the key input fields. This object only has the object ID as the key field (for example, Assoc ID).

- 1 Double-click the message test program.
- 2 Click File > Open.
- 3 Click the DIRXML_SCHEMA01_QUERY.IST file > Open.

4 Verify that the appropriate connect string, operator ID, and password are listed.

5 Modify the Assoc ID.

This field must correspond to the employee that you are trying to access.

6 Click Execute!.

If the Message Agent is working properly, all the fields that are populated in PeopleSoft will be populated in the output.

The following is a sample script:

```
Connect 137.65.139.178:7000,PS,PS;  
StartMessage DIRXML_SCHEMA01_QUERY,DIRXML_SCHEMA01_QUERY;  
SetField Assoc ID=8964;  
ProcessMessage;  
GetOutPutall;  
Disconnect
```

7 Close the log file.

Now that you have tested your PeopleSoft message agents, you should install the event server. Refer to [“Installing the Event Server” on page 32](#).

Installing the Event Server

Manually Installing and Configuring the Event Server

Typically, the Event Server is installed when you run the installation program. If either of the following has occurred, manually install or configure the desktop shortcut for the Event Server:

- ◆ You did not install the Event Server when you installed other components.
- ◆ Various directory paths were unavailable during the installation.

Installing the Event Server

The following steps assume that eDirectory has been installed on drive C: of the server in the \NOVELL\RemoteLoader\ directory.

1 Copy the following to the C:\NOVELL\NDS directory.

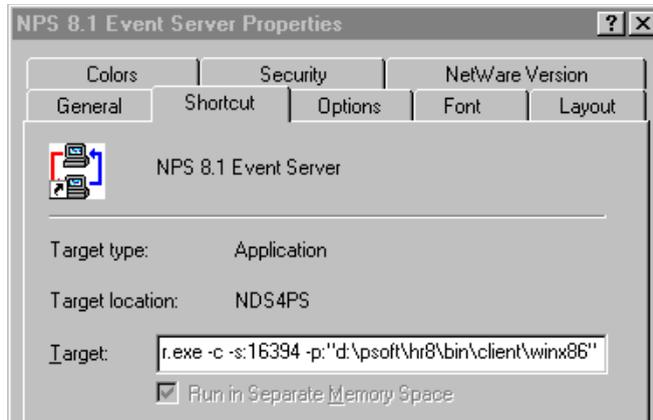
- ◆ NPS75EVENTSERVER.EXE for PeopleSoft 7.5
- ◆ NPS81EVENTSERVER.EXE for PeopleSoft 8.1

2 Create a shortcut for the Event Server application on the desktop.

IMPORTANT: If you are running multiple PeopleSoft drivers, create one shortcut for each Event Server.

3 Edit the command line in the Target field.

The following graphic illustrates this field:



Right-click the shortcut > click Properties > Shortcut.

Enter parameters for the shortcut. In the following table, {PS_Home} is the root of the PeopleSoft directory structure.

PeopleSoft Platform	Parameters
PeopleSoft 7.5	C:\Novell\NDS\NPS75EventServer.exe -c -s:16386 -p:{PS_Home}\bin\client\winx86
PeopleSoft 8.1	C:\Novell\NDS\NPS81EventServer.exe -c -s:16394 -p:{PS_Home}\bin\client\winx86

The -c parameter configures the Event Server so that it runs as a console application. When you use this parameter, configure the DOS window properties with the following:

- ◆ A scroll bar
- ◆ A different color layout

The Event Server communicates with the driver using a Windows sockets interface or listening port. The following table lists the delivered socket number for each Event Server. If these numbers conflict with sockets already being used in your organization, use a different port set.

Event Server	Port
PeopleSoft 7.5 Event Server	16386
PeopleSoft 8.1 Event Server	16394

Because each Event Server requires four consecutive sockets or ports, socket numbers for multiple Event Servers must be at least four digits apart.

The Event Server must establish communications with the PeopleSoft Message Agent. It does this by connecting to the PeopleSoft Message Agent DLL files.

These DLL files are typically located in the {PS_HOME}\BIN\CLIENT\WINX86 directory.

For the NPS75EventServer shortcut, an example path to the PeopleSoft Message DLLs is -p:d:\psoft\hr757\bin\client\winx86. In this example, {PS_HOME} is equal to d:\psoft\hr757.

4 Enter the following in the Start In field:

c:\Novell\NDS

This is the directory where eDirectory is installed. The following figure illustrates this field:



5 Select Apply > Click OK.

6 Open the application via the shortcut.

You can customize the scroll bar and color changes by customizing the Property tab on the window.

Stopping the Event Server

You might need to stop the Event Server to refresh PeopleSoft connectivity, recycle the server, or debug. When you stop the Event Server, the Event Server sends a stop command to the driver shim. This command shuts down the driver shim before the Event Server exits.

To stop the Event Server from a console window:

1 Select the Event Server window.

2 Enter **Q**.

IMPORTANT: You can stop the Event Server by clicking the Exit icon in the window. However, click this icon only if the driver shim is not running. Before clicking the icon, shut down the driver shim. Do this from the DirXML Driver Set Properties menu.

Installing the Event Server as a Service

You can install the Event Server as an NT or Windows 2000 service. Normally, the Event Server is installed during the driver installation.

Two additional shortcuts are placed on the desktop during the installation process. These shortcuts enable you to create or remove the Event Server as a service.

To install the Event Server as a service:

1 Ensure that the Event Server program is available.

- ◆ On a local machine, verify that NPSxEvent Server.EXE is saved locally to the machine.
- ◆ If installing the Event Server on a remote server, where eDirectory is not installed, copy DIRXMLLIB.DLL and EXPAT.DLL from the NOVELL\NDS directory to the directory that the Event Server will be run from.

NOTE: The PeopleSoft client must always be installed on the server where the Event Server is going to run.

The remote server's IP address will be used when configuring the Event Server in the driver parameters.

2 At the command prompt, go to the directory where the executable file is stored > enter the following:

```
NPS75EventServer.exe -i -s:16386  
-p:{PS_HOME}\bin\client\winx86
```

The socket number and the path to the PeopleSoft PSMSG.DLL might be different, based on your environment and driver configuration.

- 3** From the Start menu, click Settings > Control Panel > Services > DirXML Event Server for PS 7.5 (or the appropriate release).
- 4** Click Startup > select the Automatic radio button to automatically start up the Event Server at login > click Start.

To stop the service:

- 1** From the Start menu, click Settings > Control Panel > Services.
- 2** Select the service > click Stop.

Manually Uninstalling the Event Server as a Service

You can uninstall or remove the Event Server as a service. Follow the same steps as for installing the Event Server. However, in Step 2 substitute the -u parameter for the -i parameter.

For example, enter `NPS75EventServer.exe -u -s:16386 -p: {PS_HOME}\bin\client\winx86`

NOTE: When the Event Server runs as a service, there is no console window. All debug information would only be available in the DSTRACE log.

4 Troubleshooting the Driver

This section contains potential problems and error codes you may encounter while configuring or using the driver.

Resolving Errors

The Event Server Will Not Load

- ◆ Check the path environment variable to assure that the PeopleSoft client and the DirXML[®] directory are in its path.

The Driver Will Not Start

- ◆ Check that the Event Server is running.
- ◆ Check that the NPSSHIM.DLL is accessible.
- ◆ Check that the connection parameters are set correctly.

The Driver Is Not Communicating with the Event Server

- ◆ Verify that the Event Server was started first.
- ◆ Verify that the Event Server and driver do not have multiple copies running.
- ◆ Verify that the connection was not previously broken.
- ◆ You might have to restart Novell[®] eDirectory[™] and then restart the driver components.

The Event Server Receives Message Agent Errors

- ◆ Verify that the APIs are configured appropriately. (Use the Message Agent Test Program.)
- ◆ Verify that the Application Server is running.
- ◆ Verify connectivity to the Message Agent.
- ◆ Make sure you have the correct PeopleSoft Application Server Connection String:
You can use either: HOSTNAME:PORT or IP:PORT for single instance of Application Server
or
Use Fail Over Mode: //HOSTNAME:PORT//HOSTNAME:PORT where the first entry is the first Application Server and the second entry is the second Application Server to which the driver tries to connect.
- ◆ Make sure the names of the Message Agent APIs are spelled correctly. These names are case sensitive.

- ◆ Check for errors when updating the Worklist transaction.
- ◆ Check for errors on the Application Server.
- ◆ Check for errors on the SMTP gateway.
- ◆ Verify password usage of the driver. This might be case sensitive depending on the platform you use.

Attributes Do Not Get Refreshed on the Data Map Object

- ◆ Verify that the Message Agent APIs are working correctly.
- ◆ Verify that both the worklist and data update APIs are working correctly (DIRXML_SCHEMA01 and DIRXML_SCHEMA01_UPDATE).

Driver Only Appears to Process Transactions

Transactions appear to be read from the worklist, but nothing gets processed by the driver. Messages repeat themselves. The following text output might appear on the Event Server console:

```
Record Count: 5
Processing record 1 of 5.
Processing record 2 of 5.
Processing record 3 of 5.
Processing record 4 of 5.
Processing record 5 of 5.
```

However, no data is being processed. This is a result of reading transactions out of the worklist, but unable to retrieve the data from the transaction. Make sure that you can read the data from the default DIRXML_SCHEMA01 Message definition. You should also:

- ◆ Verify that you can read a particular transaction with the Message Test Program.
- ◆ Verify that you only get one record when accessing the same transaction online.
- ◆ Verify that the ACTIONDTTM field has been removed as an input key field on the API.

Data Does Not Show up in eDirectory on the Publisher Channel

- ◆ Verify that the Mapping policy and filters are configured correctly.
- ◆ Verify that the APIs are working correctly and data is being produced by them.

Data Does Not Update in PeopleSoft on the Subscriber Channel

- ◆ Verify that the Mapping policy and filters are configured correctly.
- ◆ Verify that the APIs are working correctly.

No Transactions Are Coming Across the Publisher Channel

- ◆ Verify that the eDirectory Default Synchronization Check box is checked.
- ◆ Verify that there are active transactions in the queue ready for processing.
- ◆ Ensure that driver parameters are pointing to the correct PeopleSoft database. For example, transactions will not process if they are in the PROD database, but the driver is still pointing to the test database (which is configured to run with the driver, but holds no transactions).

Transactions Do Not Get Placed in the PeopleSoft Queue

- ◆ Verify that PeopleCode is working properly.
- ◆ Verify that you are not in correction mode when performing updates.

No Data Is Returned When Running the Message Test Program

No data is returned, particularly when you are running the Message Test Program. You might see errors such as, "No Text Available."

There are two typical reasons for this error:

- ◆ The Key Input elements are not pointing to the Search Record on the Field Mapping definition of the Activity.
- ◆ The field elements point to an invalid reference.

Make sure the Key Input elements are associated with the Search Record entry on the field mapping definition and not directly with the table. Also, make sure that the field elements are mapped to an appropriate field record definition that exists in the application buffer. You should also ensure that the data elements exist on a page within the application and that the links are configured correctly.

Transactions Are Left in Selected State and Not Processed

- ◆ Verify that all of the Message Activities can be processed and that the states can be updated to a 2 (worked) or 4 (error).

If e-mail is configured in PeopleSoft and the SMTP gateway is down, an error can occur causing the update of the transaction list to fail. You should verify that all online processing of the application works correctly. PeopleCode attached to the update might sometimes fail, thus causing the transaction to fail. If system connectivity is lost, the database or application server goes down during processing and causes the driver to abandon the transaction. The transaction is left in the state of "selected" with a status of 1.

Receiving Errors on the Publisher Channel When Processing a Transaction

These errors are sometimes hard to figure out. Here is a sampling of errors and generally what they represent:

- ◆ Operation vetoed by Create policy.
Possible required data missing in Create policy or other criteria in the Create policy has an error.

- ◆ generateKeyPair: -216 DSERR_PASSWORD_TOO_SHORT
Attribute used for initial password does not comply to policy, however, the user object will still be created.
- ◆ Unable to read current state of 8101
No association exists for this identity.
- ◆ nameToID: -601 ERR_NO_SUCH_ENTRY
Possible Placement policy error with an invalid container object designated.
- ◆ No DN generated by Placement policy
Possible missing or invalid data causing no valid DN to be created.

Message Agent Relationships Not Functioning

If data does not show up in the attributes, or isn't getting posted into PeopleSoft, or data is missing, you should begin looking at the message agent relationships. You should:

- ◆ Verify that the API is getting the data from the PeopleSoft buffer. (Use the Message Agent Test program and follow all four steps in the driver process to test all of the APIs.)

For the delivered Message Agent Activities, the following represents the process flow of testing the Message Agent Scripts with the Message Agent Test Program. The purpose of this test is to completely validate and walk through the process that the driver uses to interface with PeopleSoft and retrieve transactions and update the PeopleSoft environment. Each of the applications that these Message Activities are connected with should be tested online through the use of PeopleTools prior to testing the Message Agent scripts:

- 1** Ensure that you only have one active/available transaction in the worklist ready for processing.
- 2** Execute the DIRXML_TRANS01.ist script to retrieve the transaction in the worklist.
- 3** Verify that key elements of the transaction are returned and that they are unique.
- 4** Execute the second script, DIRXML_SCHEMA01 replacing the key element values with the values retrieved from Step 2 above from the execution of DIRXML_TRANS01.ist script. Also, change the INSTSTATUS to 1 in order to represent updating the PeopleSoft transaction to a selected state.
- 5** Validate that all of the data elements appear as desired.

If elements are missing, there is probably an issue with the data being retrieved on one of the pages in the application. You should check the ...DATA1X, ...DATA2X, and ...DATA3X pages in the associated application. Sometimes, even though the data shows up on the page, it doesn't appear in the results when running the script because there is a problem in the related display definition.

Another reason why the data might not appear is that the process direction is not set to output on the Field Mapping within the Activity definition.

- 6** Repeat steps 4 and 5 by executing the script again for the same transaction while changing the status to 2 (worked) and 4 (error).

The driver, when processing a transaction, will always set the transaction to a status of 1 (selected). Once the transaction is processed it will either set the transaction to 2 (worked) or 4 (error). If an error occurs, the comment field will also be updated. Be sure to test all possible updates and verify online that the update occurred as desired.

- 7** Execute the DIRXML_SCHEMA01_UPDATE script for a particular object. The default would be for a particular employee by specifying the EMPLID of the employee. The NDS ID and NDS CONTEXT fields are updated in PeopleSoft on the DIRXML_TRANS01 record when the driver complete processing of a transaction retrieved on the publisher channel. If this fails, the driver cannot complete the update of the status that occurs in the prior steps. The update of these fields is for documentation purposes only or to make them available elsewhere in PeopleSoft, or for using in an e-mail generated by a PeopleSoft workflow, etc. This functionality can be turned off by using an override parameter defined on the properties parameter page in the driver configuration.

NOTE: You should execute the script multiple times by replacing the contents of the Description and the Email ID fields and/or any additional fields that may have been added to the DIRXML_TRANS01 table and this definition that are set to be updated by the driver on the subscriber channel, i.e., from eDirectory to PeopleSoft.

- 8** Execute the DIRXML_SCHEMA01_QUERY script to validate that the data elements for a particular object are accessible.

These elements should all be set to output except for the key input value. The elements listed here on the field mapping should be consistent with those listed on the DIRXML_SCHEMA01 definition. The only difference should be the key input fields. On this Activity, there is normally only one key input field for the object being accessed. This would equate to EMPLID for the delivered configuration against the HR database.

Once all of the Message Agent Test Scripts have been tested completely with validation of all processes that the driver is configured to do, there should be no issues regarding the driver accessing PeopleSoft through the Message Agent. Typical other problems include:

- ◆ Connectivity IP address and Port for the application server
- ◆ ID and password
- ◆ Correct naming of all activities in the parameters for the driver.

Remember the Three Basic Test Phases:

1. Test all of the processes manually online using the PeopleSoft applications as configured.
2. Test all of the processes using the four test scripts with the Message Agent Test program.
3. Test the driver connecting to the Activities through the Message Agent.

