

# Novell Identity Manager Driver for Exchange 5.5

3.5.1

[www.novell.com](http://www.novell.com)

---

IMPLEMENTATION GUIDE

September 28, 2007



**Novell**<sup>®</sup>

## 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. For more information on exporting Novell software, see the [Novell International Trade Services Web page \(http://www.novell.com/info/exports/\)](http://www.novell.com/info/exports/). Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 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 at [Novell Legal Patents \(http://www.novell.com/company/legal/patents/\)](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](http://www.novell.com)

*Online Documentation:* To access the online documentation for this and other Novell products, and to get updates, see [Novell Documentation \(http://www.novell.com/documentation/\)](http://www.novell.com/documentation/).

## **Novell Trademarks**

For a list of Novell trademarks, see [Trademarks \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html).

## **Third-Party Materials**

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



# Contents

<b>About This Guide</b>	<b>9</b>
<b>1 Overview</b>	<b>11</b>
1.1 What's New	11
1.2 Driver Concepts	11
1.2.1 Key Terms	11
1.2.2 Benefits	12
1.2.3 Required Skills	13
1.2.4 How the Exchange Driver Works	13
1.3 Driver Features	14
1.3.1 Local Platforms	14
1.3.2 Remote Platforms	15
1.3.3 Entitlements	15
1.3.4 Password Synchronization	15
1.3.5 Synchronizing Data	15
1.3.6 Other Features	15
<b>2 Installing the Exchange Driver</b>	<b>17</b>
2.1 Prerequisites	17
2.1.1 Software Requirements	17
2.1.2 Hardware Requirements	17
2.2 Upgrading to Identity Manager 3.5	17
2.3 Installing on Windows	17
<b>3 Upgrading the Exchange Driver</b>	<b>21</b>
3.1 Running the Normalize Exchange Associations Utility	21
3.2 Upgrading the Driver by Using Designer	22
3.3 Upgrading the Driver by Using iManager	25
3.4 Upgrading the Driver Configuration	26
<b>4 Importing an Example Configuration File</b>	<b>27</b>
4.1 Using Designer to Import	27
4.2 Using iManager to Import	28
<b>5 Configuring the Exchange Driver</b>	<b>31</b>
5.1 Configuring the Exchange Server	31
5.2 Installing a Remote Exchange Driver	33
5.3 Configuring the Driver Filter	33
5.4 Integrating the Identity Manager Driver for Exchange and the Identity Manager Driver for NT Domain	34
5.5 Managing External Recipients	37
5.6 Synchronizing Proxy-Address and Target-Address Attributes	38
5.7 Using Authoritative Bind	39
5.8 Using a Custom Bind	39

5.9	Specifying the LDAP Port . . . . .	40
<b>6</b>	<b>Activating the Exchange Driver</b>	<b>43</b>
<b>7</b>	<b>Managing the Exchange Driver</b>	<b>45</b>
7.1	Starting, Stopping, or Restarting the Exchange Driver . . . . .	45
7.2	Migrating and Resynchronizing Data . . . . .	46
7.3	Using the DirXML Command Line Utility . . . . .	46
7.4	Viewing Driver Version Information . . . . .	46
7.4.1	Viewing a Hierarchical Display of Version Information . . . . .	46
7.4.2	Viewing the Version Information As a Text File . . . . .	48
7.4.3	Saving Versioning Information . . . . .	50
7.5	Reassociating a Driver Set Object with a Server Object . . . . .	51
7.6	Changing the Driver Configuration . . . . .	52
7.7	Storing Driver Passwords Securely with Named Passwords . . . . .	52
7.7.1	Using Designer to Configure Named Passwords . . . . .	53
7.7.2	Using iManager to Configure Named Passwords . . . . .	53
7.7.3	Using Named Passwords in Driver Policies . . . . .	55
7.7.4	Using the DirXML Command Line Utility to Configure Named Passwords . . . . .	55
7.8	Adding a Driver Heartbeat . . . . .	59
<b>8</b>	<b>Synchronizing Objects</b>	<b>61</b>
8.1	What Is Synchronization? . . . . .	61
8.2	When Does Synchronization Occur? . . . . .	61
8.3	How Does the Metadirectory Engine Decide Which Object to Synchronize? . . . . .	62
8.4	How Synchronization Works . . . . .	63
8.4.1	Scenario One . . . . .	63
8.4.2	Scenario Two . . . . .	65
8.4.3	Scenario Three . . . . .	66
<b>9</b>	<b>Troubleshooting the Driver</b>	<b>69</b>
9.1	Troubleshooting Tips . . . . .	69
9.2	Driver Error Messages . . . . .	69
9.3	Troubleshooting Driver Processes . . . . .	71
9.3.1	Viewing Driver Processes . . . . .	71
<b>10</b>	<b>Backing Up the Exchange Driver</b>	<b>77</b>
10.1	Exporting the Driver in Designer . . . . .	77
10.2	Exporting the Driver in iManager . . . . .	77
<b>11</b>	<b>Security: Best Practices</b>	<b>79</b>
<b>A</b>	<b>The DirXML Command Line Utility</b>	<b>81</b>
A.1	Interactive Mode . . . . .	81
A.2	Command Line Mode . . . . .	90

<b>B</b>	<b>Properties of the Exchange Driver</b>	<b>95</b>
B.1	Identity Manager: Driver Configuration	95
B.1.1	Driver Module	96
B.1.2	Authentication	97
B.1.3	Startup Option	98
B.1.4	Driver Parameters	99
B.1.5	ECMAScript	100
B.2	Identity Manager: Global Configuration Values	100
B.3	Identity Manager: Named Passwords	101
B.4	Identity Manager: Engine Control Values	102
B.5	Identity Manager: Log Level	104
B.6	Driver Image	105
B.7	Security Equals	105
B.8	Filter	105
B.9	Edit Filter XML	106
B.10	Identity Manager: Misc.	106
B.11	Excluded Objects	107
B.12	Driver Manifest	107
B.13	Driver Cache Inspector	108
B.14	Driver Inspector	108
B.15	Server Variables	109
B.16	Driver Inspector	112





# About This Guide

This guide explains how to install, configure, and manage the Identity Manager Driver for Microsoft\* Exchange 5.5.

- ◆ Chapter 1, “Overview,” on page 11
- ◆ Chapter 2, “Installing the Exchange Driver,” on page 17
- ◆ Chapter 3, “Upgrading the Exchange Driver,” on page 21
- ◆ Chapter 4, “Importing an Example Configuration File,” on page 27
- ◆ Chapter 5, “Configuring the Exchange Driver,” on page 31
- ◆ Chapter 6, “Activating the Exchange Driver,” on page 43
- ◆ Chapter 7, “Managing the Exchange Driver,” on page 45
- ◆ Chapter 8, “Synchronizing Objects,” on page 61
- ◆ Chapter 9, “Troubleshooting the Driver,” on page 69
- ◆ Chapter 10, “Backing Up the Exchange Driver,” on page 77
- ◆ Chapter 11, “Security: Best Practices,” on page 79
- ◆ Appendix A, “The DirXML Command Line Utility,” on page 81
- ◆ Appendix B, “Properties of the Exchange Driver,” on page 95

## Audience

This document is for administrators who have expertise in Exchange and Novell® Identity Manager.

## Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Use the User Comment feature at the bottom of each page of the online documentation, or go to [www.novell.com/documentation/feedback.html](http://www.novell.com/documentation/feedback.html) and enter your comments there.

## Documentation Updates

For the most recent version of this document, see *Identity Manager Driver for Exchange* in the Identity Manager Drivers section on the [Novell Documentation Web site \(http://www.novell.com/documentation/idm35drivers/index.html\)](http://www.novell.com/documentation/idm35drivers/index.html).

## Additional Documentation

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

## Documentation Conventions

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.

# Overview

# 1

The Identity Manager Driver for Exchange is a connector that synchronizes data between an Identity Vault and Microsoft Exchange. This synchronization makes it possible for Exchange accounts to be managed in an Identity Vault. You no longer need to manage a user's Identity Vault and Exchange accounts separately.

The Identity Manager Driver for Exchange increases the efficiency of your network management by allowing you to manage an Identity Vault and Exchange accounts as a single account in a single management tool.

- ◆ [Section 1.1, “What’s New,” on page 11](#)
- ◆ [Section 1.2, “Driver Concepts,” on page 11](#)
- ◆ [Section 1.3, “Driver Features,” on page 14](#)

## 1.1 What’s New

The Identity Manager Driver for Exchange runs on Windows\* NT\*.

The driver supports only the Distribution List, Remote, and Mailbox classes.

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

## 1.2 Driver Concepts

- ◆ [Section 1.2.1, “Key Terms,” on page 11](#)
- ◆ [Section 1.2.2, “Benefits,” on page 12](#)
- ◆ [Section 1.2.3, “Required Skills,” on page 13](#)
- ◆ [Section 1.2.4, “How the Exchange Driver Works,” on page 13](#)

### 1.2.1 Key Terms

**Driver Shim.** A dynamically linked library (`Exchange55Shim.dll`) loaded directly by Identity Manager or by the Remote Loader. The shim collects the changes to be sent from Exchange to the Identity Vault, communicates changes from the Identity Vault to Exchange, and operates as the link that connects the Identity Vault and Exchange.

**Driver.** A set of policies, filters, and objects that act as the connector between the Identity Vault and the driver shim. The Identity Manager Driver for Exchange is a bidirectional synchronization connector between Microsoft Exchange and an Identity Vault. This connector uses XML to convert Exchange objects to Identity Vault objects and vice versa.

The driver enables an application to publish events from an application to the directory, enables an application to subscribe to events from the directory, and synchronizes data between the directory and applications.

To establish a connection between the Metadirectory engine and Exchange, you specify the driver's configuration and connection parameters, policies, and filter values.

**Driver Object.** A collection of channels, policies, rules, and filters that connect an application to an Identity Vault that is running Identity Manager.

Each driver performs different tasks. Policies, rules, and filters tell the driver how to manipulate the data to perform those tasks.

The Driver object displays information about the driver's configuration, policies, and filters. This object enables you to manage the driver and provide Identity Vault management of the driver shim parameters.

**Identity Vault.** A hub, with other applications and directories publishing their changes to it. The Identity Vault then sends changes to the applications and directories that have subscribed for them. This results in two main flows of data:

- ◆ The Publisher channel
- ◆ The Subscriber channel

**Publisher Channel.** Reads information from your Exchange Server and submits that information to the Identity Vault via the Metadirectory engine.

The Publisher channel uses the Poll parameter to poll the Exchange server for changes to objects. If the Identity Manager Driver for Exchange detects changes in Exchange, the data between Exchange and the Identity Vault is synchronized. If the change was caused by data sent to Exchange from the Subscriber, no synchronization is necessary.

**Subscriber Channel.** Watches for additions and modifications to Identity Vault objects and creates changes on the Exchange server via the Metadirectory engine.

The Subscriber channel synchronizes changes made in the Identity Vault with data on the Exchange server. If an associated object is changed in the Identity Vault, the Subscriber channel updates the Exchange server with the new information.

## 1.2.2 Benefits

You can use the driver to automate and maintain business processes in the following ways:

- ◆ Automatically create Identity Vault objects from Exchange objects.
- ◆ Synchronize bidirectional data between Exchange and an Identity Vault.
- ◆ Maintain accurate and consistent Identity Vault IDs.
- ◆ Enable integration between Exchange and multiple applications (for example, an Identity Vault, Lotus Notes\*, Netscape\*, SAP\*, and Active Directory\*) by using Identity Manager and an Identity Vault.
- ◆ Manage Exchange distribution lists and remote objects.

You can configure the Identity Manager Driver for Exchange to use custom business logic in the form of policies to enhance your organization's 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.

## 1.2.3 Required Skills

Implementing the driver requires expertise in Exchange and Identity Manager. This document assumes that your expertise in Exchange is equivalent to one of the following:

- ♦ An Exchange developer
- ♦ An Exchange administrator
- ♦ An application designer
- ♦ An upgrade administrator
- ♦ A database administrator

This document assumes that your expertise in Identity Manager is equivalent to an Identity Vault administrator or an Identity Manager administrator.

## 1.2.4 How the Exchange Driver Works

- ♦ [“Processing Events” on page 13](#)
- ♦ [“Policies” on page 13](#)
- ♦ [“Associations” on page 14](#)

### Processing Events

The driver supports the following events on the Publisher and Subscriber channels.

*Table 1-1 Supported Events*

Functionality	Event
Publisher	Add Modify Delete Rename
Subscriber	Add Modify Delete Rename

The driver also supports a defined query capability so that Identity Manager can query the synchronized application or directory.

### Policies

Policies control the synchronization of the driver with the Identity Vault and the application, database, or directory. Policies help Identity Manager transform an event on a channel input into a set of commands on the channel output.

You can configure policies by using the Designer and iManager plug-ins for Identity Manager. The example driver configuration includes the following set of policies:

**Table 1-2** Policies in the Sample Configuration File

Policy	Description
Placement	Operates on both the Publisher and Subscriber channels
Matching	Operates on both the Publisher and Subscriber channels
Mapping	Configured on the Driver object
Input Transform	Configured on the Driver object
Output Transform	Configured on the Driver object
Create	Operates on the Publisher and Subscriber channels
Event Transform	Operates on the Publisher channel
Command Transform	Operates on the Publisher channel

For more information about creating policies, see [Understanding Policies for Identity Manager 3.5.1](#) and [Policies in iManager for Identity Manager 3.5.1](#).

## Associations

The driver uses the Exchange DN for associations. A unique ID or unique user name is created for records relating to Exchange objects. However, Identity Manager does not need to share these same unique IDs.

The association attribute received from Exchange is unique to the Exchange application, based on each driver for Exchange that you install and enable. If other drivers are installed, they use an association specific to that application. The association attribute is multivalued. Therefore, if Identity Manager is being used to connect multiple applications, all of their associations can be stored on this attribute.

The unique ID association links an object in Exchange to its associated object in the Identity Vault. This association allows the driver to perform subsequent tasks on the appropriate object.

The Association field is stored on the Identity Vault object on the Identity Manager property page.

## 1.3 Driver Features

- ◆ [Section 1.3.1, “Local Platforms,” on page 14](#)
- ◆ [Section 1.3.2, “Remote Platforms,” on page 15](#)
- ◆ [Section 1.3.3, “Entitlements,” on page 15](#)
- ◆ [Section 1.3.4, “Password Synchronization,” on page 15](#)
- ◆ [Section 1.3.5, “Synchronizing Data,” on page 15](#)
- ◆ [Section 1.3.6, “Other Features,” on page 15](#)

### 1.3.1 Local Platforms

The Exchange driver runs on Windows NT.

## 1.3.2 Remote Platforms

The Exchange driver can run on the Remote Loader or remotely access the Exchange server.

## 1.3.3 Entitlements

The Exchange driver supports entitlements for mailboxes.

Using entitlements is a design decision. Before choosing this option during an import, see “[Creating and Using Entitlements](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

## 1.3.4 Password Synchronization

The Exchange driver does not synchronize passwords. It relies on the NT driver to do that for it.

## 1.3.5 Synchronizing Data

The Exchange driver synchronizes bidirectional data between an Identity Vault and Microsoft Exchange.

## 1.3.6 Other Features

- ♦ The AuthoritativeBind parameter lets you use an authoritative LDAP bind instead of an anonymous LDAP bind. See “[Using Authoritative Bind](#)” on page 39.
- ♦ You can use the Assoc-NT-Account attribute for queries into Exchange.
- ♦ Instead of the preferredName attribute, the DirXML-NTAccountName attribute is now used in eDirectory. See “[Integrating the Identity Manager Driver for Exchange and the Identity Manager Driver for NT Domain](#)” on page 34.

You can still use the preferredName attribute.





# Installing the Exchange Driver

# 2

- ♦ [Section 2.1, “Prerequisites,” on page 17](#)
- ♦ [Section 2.2, “Upgrading to Identity Manager 3.5,” on page 17](#)
- ♦ [Section 2.3, “Installing on Windows,” on page 17](#)

## 2.1 Prerequisites

This section lists the software and hardware requirements for running the Identity Manager Driver for Exchange.

### 2.1.1 Software Requirements

- Novell® Identity Manager 2 or later, and its prerequisites
- Windows NT 4 with the latest patches and service packs (SP6a or later)
- Exchange 5.5 with Service Pack 4 or later and the latest patches

---

**NOTE:** If the Exchange Server doesn't have the latest patches, an Entitlement Policy won't correctly assign membership in an Exchange distribution list. Also, proxy addresses are handled improperly because of a defect in Exchange.

---

### 2.1.2 Hardware Requirements

- 128 MB RAM (256 MB or more recommended)

## 2.2 Upgrading to Identity Manager 3.5

You can upgrade from DirXML® 1.1a, Identity Manager 2.x, or Identity Manager 3.x to Identity Manager 3.5. During an Identity Manager installation, you can install the Driver for Exchange 5.5 (along with other Identity Manager drivers) at the same time that the Metadirectory engine is installed. See [“Installing Identity Manager”](#) in the *Identity Manager 3.5.1 Installation Guide*.

## 2.3 Installing on Windows

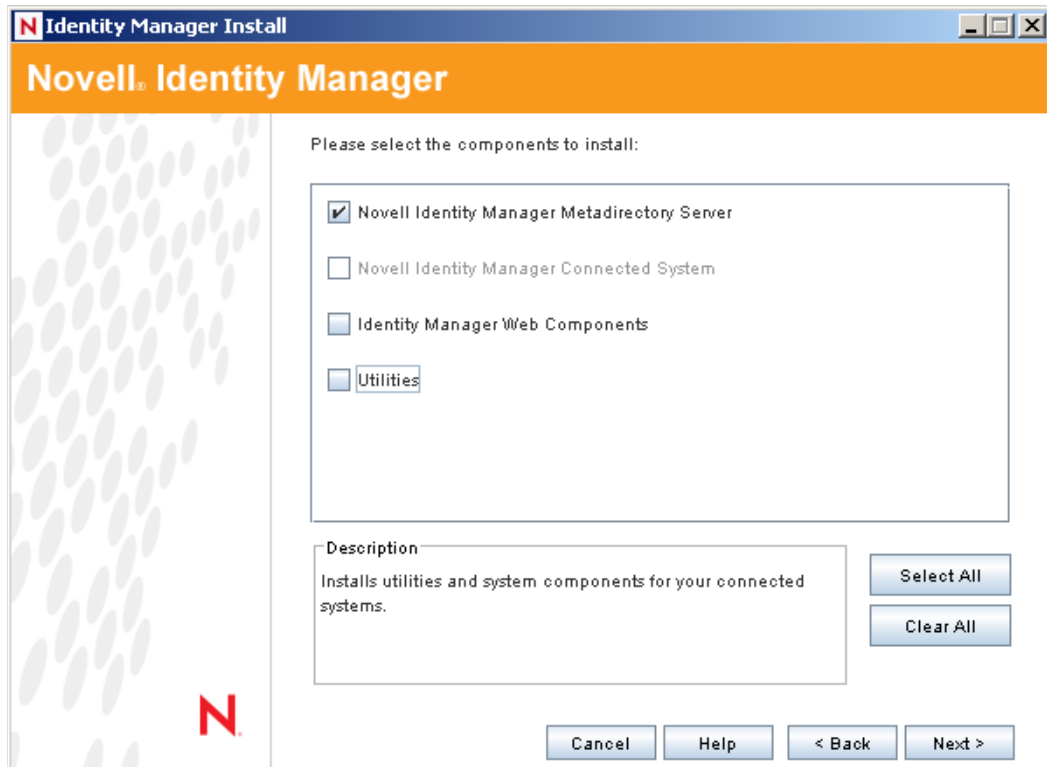
This section assumes that you have already installed the Metadirectory engine (and, most likely, other drivers) on the server and need to install only the Exchange driver. See [“Installing Identity Manager”](#) in the *Identity Manager 3.5.1 Installation Guide*.

Typically, an Identity Manager installation installs all drivers, including the Exchange driver, at the same time that the Metadirectory engine is installed. If the Exchange driver wasn't installed at that time, you can install the driver separately. The schema won't be extended during this driver install because the Identity Manager installation already extended it when the Metadirectory engine was installed.

- 1 Run the installation program from the Identity Manager 3.5.1CD or image file.

If the installation program doesn't autolaunch, you can run `\nt\install.exe`.

- 2 On the Welcome page, review information, then click *Next*.
- 3 On the License Agreement page, select a language, review the license agreement, then click *I Accept*.
- 4 On the first Identity Manager Overview page, review the information on the Identity Manager/ Metadirectory Server and a Connected System Server, then click *Next*.
- 5 In the second Identity Manager Overview page, review information on the Web-based Administration Server and utilities, then click *Next*.
- 6 On the Identity Manager Install page, select *Novell Identity Manager Metadirectory Server*, then click *Next*.



The following options are available:

- ♦ **Metadirectory Server:** Installs the Metadirectory engine and service drivers. These include Identity Manager Drivers for Active Directory\*, Avaya\*, Delimited Text, eDirectory, Exchange, GroupWise®, JDBC\*, JMS, LDAP, Linux/UNIX Settings, Lotus Notes\*, PeopleSoft, RACF, Remedy, SOAP, SAP\*, SIF\*, Top Secret, and Work Order. Selecting this option also extends the eDirectory schema.

---

**IMPORTANT:** Novell® eDirectory 8.7.3 and Security Services 2.0.4 (NMAS™ 3.1.3) with current patches must be installed before you can install this option. Install the Metadirectory Server component where you want to run the Metadirectory engine for Identity Manager. If you do not have the correct version of NMAS, you receive a warning message and you lose Identity Manager functionality.

---

- ♦ **Connected System:** Installs the Remote Loader that allows you to establish a link between the connected system and a server running the Metadirectory engine. For Windows, this option installs the following drivers: Active Directory, Avaya, Delimited

Text, eDirectory, Exchange, GroupWise, JDBC, JMS, LDAP, Linux/UNIX Settings, Lotus Notes, PeopleSoft, RACF, Remedy, SOAP, SAP, SIF, Top Secret, and Work Order.

Install the Connected System to allow application connection from an application server to an eDirectory-based server running the Metadirectory engine.

- ♦ **Web Components:** Installs driver configurations, iManager plug-ins, and application scripts and utilities.

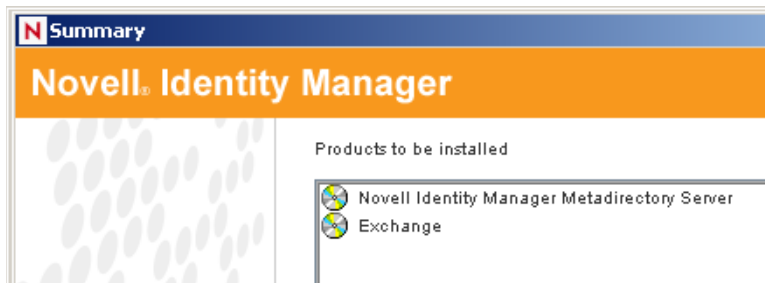
Novell iManager must be installed before you can install this option.

- ♦ **Utilities:** Installs additional scripts for the JDBC driver and utilities for other drivers. Most drivers don't have a utility connected to them. Driver utilities can include:

- ♦ SQL scripts for JDBC driver
- ♦ JMS components
- ♦ PeopleSoft components
- ♦ License Auditing tool
- ♦ Active Directory Discovery tool
- ♦ Lotus Notes Discovery tool
- ♦ SAP utilities

Another utility allows you to register the Novell Audit System components for Identity Manager. (A valid eDirectory version and a Novell Audit logging server must be installed on the tree before this utility installs.)

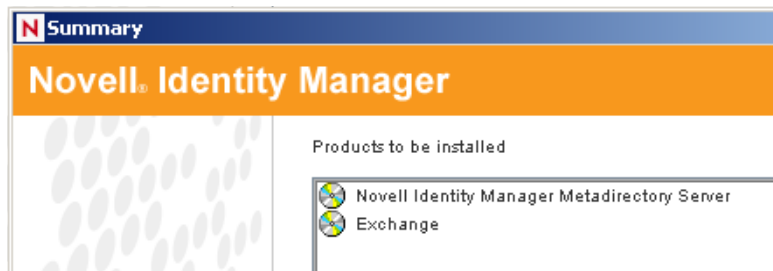
- 7 On the Select Drivers for Engine Install page, select *Exchange*, then click *Next*.



By default, all supported drivers are selected. You can install all selected drivers or you can install just the Exchange driver. Additional drivers are not viable until they are configured. To configure the driver, see [Chapter 4, "Importing an Example Configuration File," on page 27](#) and [Chapter 5, "Configuring the Exchange Driver," on page 31](#).

- 8 Review the informational message reminding you about product activation, then click *OK*. Activate the driver within 90 days of installation; otherwise, it will shut down.

**9** On the Summary page, read and verify your selections, then click *Finish*.



**10** On the Installation Complete dialog box, click *Close*.

**11** Continue by importing an example configuration file.

# Upgrading the Exchange Driver

# 3

If you are upgrading from Identity Manager 3.5.0 to Identity Manager 3.5.1, skip this section (Upgrading the Exchange Driver).

---

**NOTE:** Running an Identity Manager driver configuration with a DirXML<sup>®</sup> 1.x driver shim is not supported.

---

- ◆ [Section 3.1, “Running the Normalize Exchange Associations Utility,” on page 21](#)
- ◆ [Section 3.2, “Upgrading the Driver by Using Designer,” on page 22](#)
- ◆ [Section 3.3, “Upgrading the Driver by Using iManager,” on page 25](#)
- ◆ [Section 3.4, “Upgrading the Driver Configuration,” on page 26](#)

## 3.1 Running the Normalize Exchange Associations Utility

If you are upgrading from the DirXML<sup>®</sup> Driver 1.0 for Exchange, you need to run the Normalize Exchange Associations utility. This utility searches the Identity Vault tree and normalizes the Identity Manager Driver for Exchange associations.

---

**NOTE:** If you are upgrading from the 1.0a patch or later, you don't need to run the Normalize Exchange Associations utility.

---

- 1 Get the `changeAssocKey.zip` file from Novell<sup>®</sup> Support.
- 2 Create a temporary directory on the NT server where the Identity Manager Driver for Exchange is installed.
- 3 Expand `changeAssocKey.zip` into the directory.
- 4 Open the `run.bat` file and edit the file with these parameters:

---

Parameter	Value
Java	Driver letter and path for Java*. For example, enter C:\Novell\consoleone\1.2\jre\bin\java or D:\Novell\NDS\jre\bin\java
LDAP address:port	The IP address and port number of the Identity Vault server. Normally, this is localhost:389.
LDAP Bind ID	The LDAP authentication ID.
LDAP Bind Password	The LDAP authentication password.
Driver Name	The name of the Driver object. For example, ExchangeDriver.

---

Parameter	Value
Action	Specify one of these desired actions: <ul style="list-style-type: none"> <li>◆ 1- Lists Identity Vault objects with no association to the driver specified above.</li> <li>◆ 2- Lists Identity Vault objects with an incomplete association to the driver specified above.</li> <li>◆ 3- Lists Identity Vault objects with associations to be normalized.</li> <li>◆ 4- Modifies the associations in the Identity Vault and lists the objects.</li> </ul>

**NOTE:** We recommend that you first set the action to 3 so you can see what associations change when you set the action to 4. Then, you can set the action to 4 and run the program again. You don't cause any problems by running the program more than once. If you are concerned about the current state of the associations, you can run the utility with the action set to 1 or 2.

5 Run the `run.bat` file.

## 3.2 Upgrading the Driver by Using Designer

1 Make sure that 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.

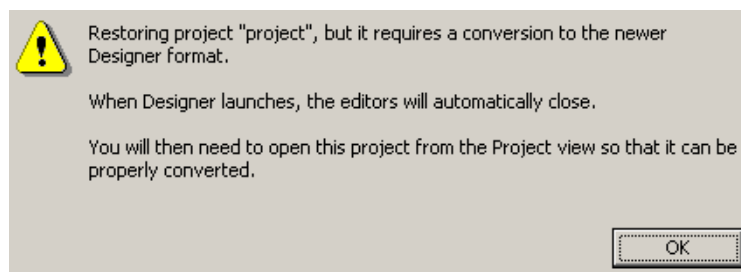
2 Back up the driver.

See [Chapter 10, "Backing Up the Exchange Driver,"](#) on page 77.

3 Install Designer version 2.1 or later, then launch Designer.

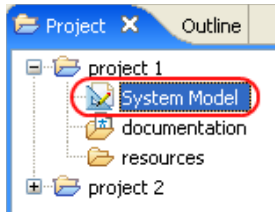
If you had a project open in Designer when you upgraded Designer, proceed to [Step 4](#). If you didn't have a project open in Designer when you upgraded Designer, skip to [Step 5](#).

4 If you had a project open when upgrading Designer, read the warning message, then click *OK*.

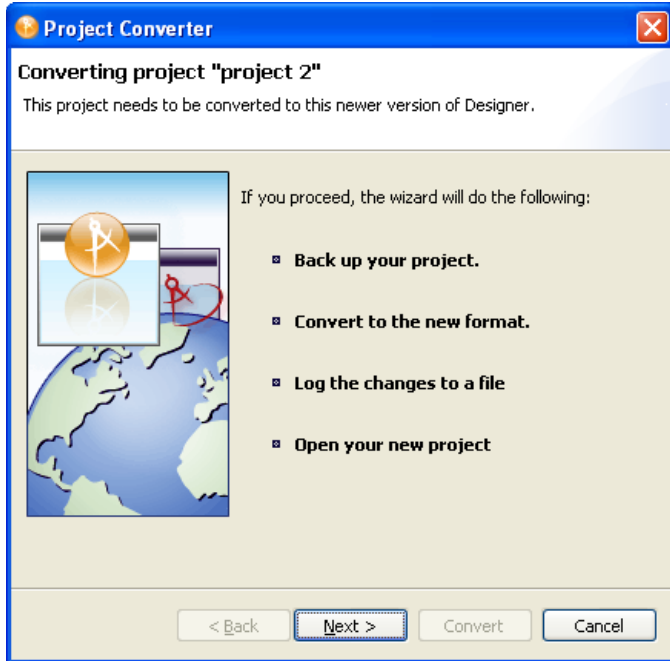


Designer closes the project to perform the upgrade.

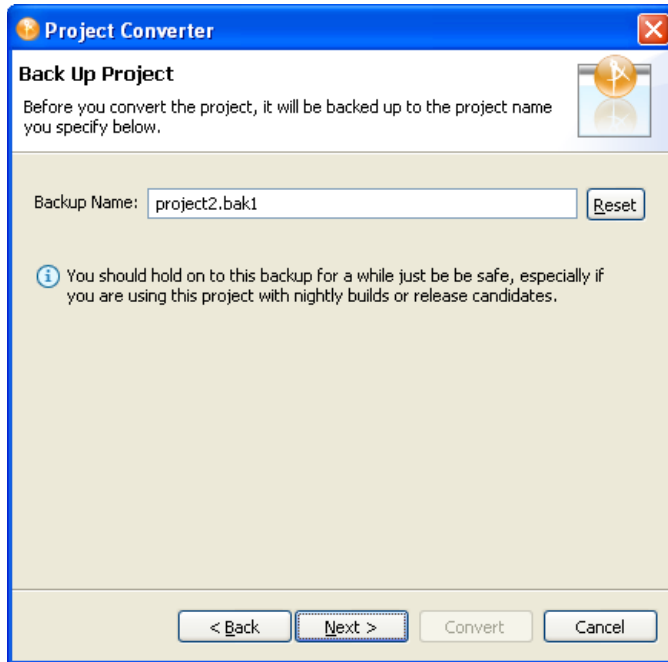
5 To open and convert the project, double-click *System Model* in the Project view.



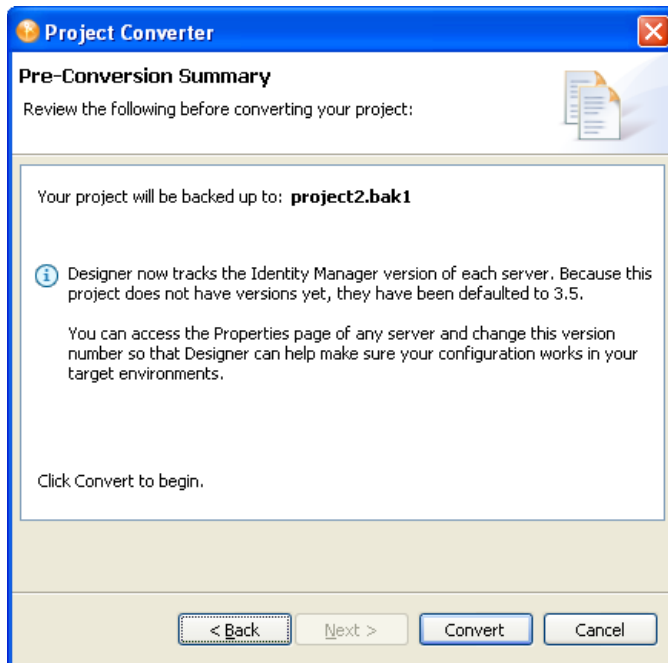
6 Read the tasks listed in the Project Converter message, then click *Next*.



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

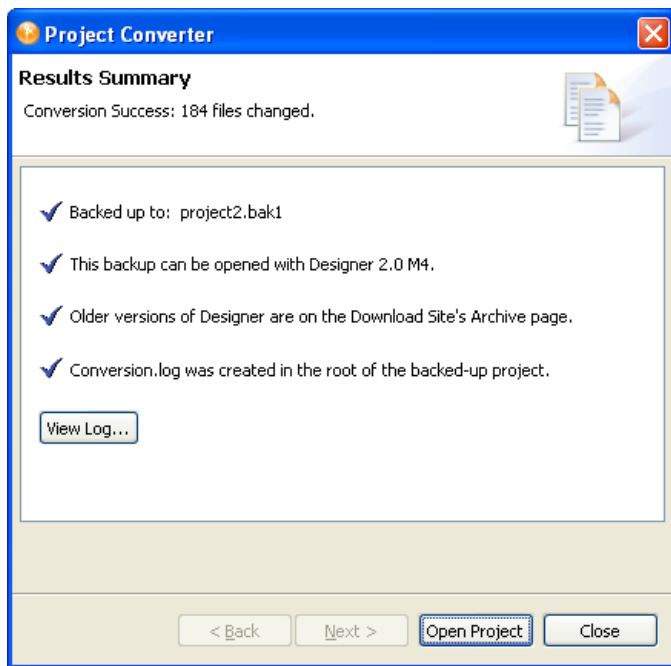


8 Read the project conversion summary, then click *Convert*.





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



To view the log file that is generated, click *View Log*.

### 3.3 Upgrading the Driver by Using iManager

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

To help minimize upgrade issues, we recommend that you complete this step on all drivers.

2 Back up the driver.

See [Chapter 10, “Backing Up the Exchange Driver,”](#) on page 77.

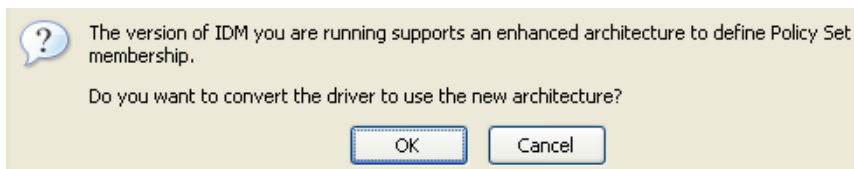
3 Verify that Identity Manager 3.5.1 has been installed and that you have the current plug-ins installed.

4 Launch iManager.

5 Click *Identity Manager > Identity Manager Overview*.

6 Click *Search* to find the Driver Set object, then click the driver that you want to upgrade.

7 Read the message that is displayed, then click *OK*.



**IMPORTANT:** The example configuration file for the updated driver changed for the Identity Manager 3.0 release. If your current configuration meets your requirements, you don't need to

import this example configuration. If you *do* import the new sample configuration, you see an additional driver for Delimited Text with a new name, a new Identity Vault container specified in the placement rule, and new rule names.

---

## 3.4 Upgrading the Driver Configuration

You can run a DirXML 1.x driver configuration with an Identity Manager 3.5.1 driver shim and the new Metadirectory engine, with no changes to the driver configuration.

However, you might want to edit a DirXML 1.x driver configuration. To do this, do one of the following:

- ◆ Use the DirXML 1.x iManager plug-ins.

See “[Managing DirXML 1.1a Drivers in an Identity Manager Environment](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

- ◆ Run the wizard that converts DirXML 1.x configurations to Identity Manager format so that you can use the iManager plug-ins for Identity Manager.

See “[Upgrading a Driver Configuration from DirXML 1.1a to Identity Manager 3.5.1 Format](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

# Importing an Example Configuration File

# 4

The example Exchange 5.5 driver configuration creates and configures the objects needed to make the driver work properly.

**Scenario:** You create a driver set and driver object in the lab. After configuring the driver, you save the configuration to a file. To save time and keep the same settings that worked well in the lab, you import the driver's configuration file from the lab environment into your production environment.

- ◆ [Section 4.1, “Using Designer to Import,” on page 27](#)
- ◆ [Section 4.2, “Using iManager to Import,” on page 28](#)

## 4.1 Using Designer to Import

You can import the basic driver configuration file for Exchange by using Designer. This basic file creates and configures the objects and policies needed to make the driver work properly.

The following procedure explains one of several ways to import the sample configuration file:

- 1 Open a project in Designer.
- 2 In the Modeler, right-click the driver set, then select *New > Driver*.
- 3 From the drop-down list, select *Exchange 5.5*, then click *Run*.
- 4 Configure the driver by filling in the fields.  
Specify information specific to your environment. For information on settings, see [Table 4-1 on page 27](#). To continue through the settings, Click *Next*.
- 5 Click *Finish* to import the driver.
- 6 Deploy the driver into the Identity Vault.

**Table 4-1** Settings for the Exchange Driver

Field	Description
<i>Driver Name</i>	The object name to be assigned to this driver, or the existing driver for which you want to update the configuration.
<i>Domain Name</i>	Specifies the NT Domain that the driver connects with.
<i>IP address of Exchange Server</i>	Specifies the IP address or host name of the Exchange Server. The driver makes LDAP queries to this server.
<i>Authoritative Bind</i>	Specifies whether to bind authoritatively or anonymously. The default is authoritative (Yes). See <a href="#">Section 5.7, “Using Authoritative Bind,” on page 39</a> .
<i>Exchange Server Name</i>	Specifies the server that contains the Exchange Post Office. The driver connects to this Exchange Post Office.

Field	Description
<i>Exchange Site Organization</i>	Specifies the organization that the driver administers.
<i>Exchange Site</i>	Specifies the site that the driver administers.
<i>Polling Frequency (in seconds)</i>	Specifies how long the driver suspends processing between each Exchange connection.
<i>Authoritative User</i>	Specifies the NT Domain user that the driver uses to authenticate to the NT Domain.
<i>User Password</i>	Specifies the password that the authoritative user uses to authenticate to the NT domain.
<i>Re-enter the Password</i>	Verifies the user password.
<i>eDir Users Container</i>	Specifies the top-level container where Groups synchronized from Exchange are placed.
<i>Configure Data Flow</i>	<p><b>Bidirectional:</b> Both Exchange and eDirectory™ are authoritative sources of the data synchronized between them.</p> <p><b>Exchange to eDirectory:</b> Exchange is the authoritative source.</p> <p><b>eDirectory to Exchange:</b> eDirectory is the authoritative source.</p>
<i>Password Failure Notification User</i>	Sends an e-mail notification to a specified user when password updates fail.
<i>Enable Entitlements</i>	<p>Specifies whether the driver implements entitlements. Because this is a design decision, you should understand entitlements before choosing to use it.</p> <p>For information about entitlements, see “<a href="#">Creating and Using Entitlements</a>” in the <i>Novell Identity Manager 3.5.1 Administration Guide</i>.</p>
<i>Driver is Local/Remote</i>	Specifies whether the driver runs locally or on a Remote Loader.

## 4.2 Using iManager to Import

1 In iManager, select *Identity Manager Utilities > Import Configurations*.

2 Select whether to place the configuration file in a new or existing driver set.

Select *In an Existing Driver Set* for the following situations:

- ◆ The driver should be logically grouped with the other drivers in the tree.
- ◆ The server can handle the additional traffic that the new driver would generate.
- ◆ You want to update or customize an existing driver.

For example, you can point the driver to a different container but keep all the rules that you have set up.

**3** In the Import Configurations dialog box, select the Exchange 5.5 driver, then click *Next*.



**4** Scroll to the parameter sections and provide required information.

Refer to the descriptions provided in the interface and in [Table 4-1](#).

**5** Define security equivalences.

The tendency is to assign Admin. However, you might want to create a DriversUser (for example) and assign security equivalence to that user.

**6** 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 5](#). If you delete the security-equivalence object, you have removed the rights from the driver. Therefore, the driver can't make changes to the Identity Vault.

**7** (Conditional) If you are re-creating or updating a driver, select *Update Everything about That Driver*, then click *Next*.

**8** On the Summary page, review options, then click *Finish*.

If you need to make changes, click *Back*.



# Configuring the Exchange Driver

# 5

- ♦ Section 5.1, “Configuring the Exchange Server,” on page 31
- ♦ Section 5.2, “Installing a Remote Exchange Driver,” on page 33
- ♦ Section 5.3, “Configuring the Driver Filter,” on page 33
- ♦ Section 5.4, “Integrating the Identity Manager Driver for Exchange and the Identity Manager Driver for NT Domain,” on page 34
- ♦ Section 5.5, “Managing External Recipients,” on page 37
- ♦ Section 5.6, “Synchronizing Proxy-Address and Target-Address Attributes,” on page 38
- ♦ Section 5.7, “Using Authoritative Bind,” on page 39
- ♦ Section 5.8, “Using a Custom Bind,” on page 39
- ♦ Section 5.9, “Specifying the LDAP Port,” on page 40

## 5.1 Configuring the Exchange Server

This section contains information on configuring the Exchange server for use with the Identity Manager Driver for Exchange. You should already be familiar with Exchange administration and deployment.

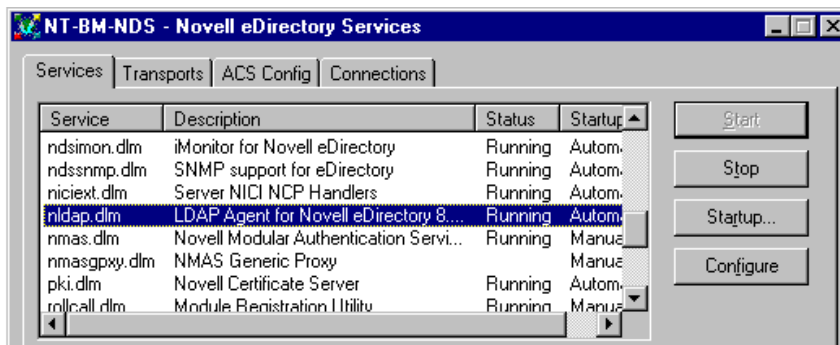
Before you proceed, you must have the following information about your setup:

- ♦ The name of the Exchange Server that the driver will be synchronizing with.
- ♦ The name of the Exchange site you want to administer.
- ♦ The IP address or hostname of the Exchange server.
- ♦ The name of the Exchange service account and its password.

If the Exchange server is running on the same computer as the Identity Vault, unload the LDAP server or reconfigure it to run on a different port.

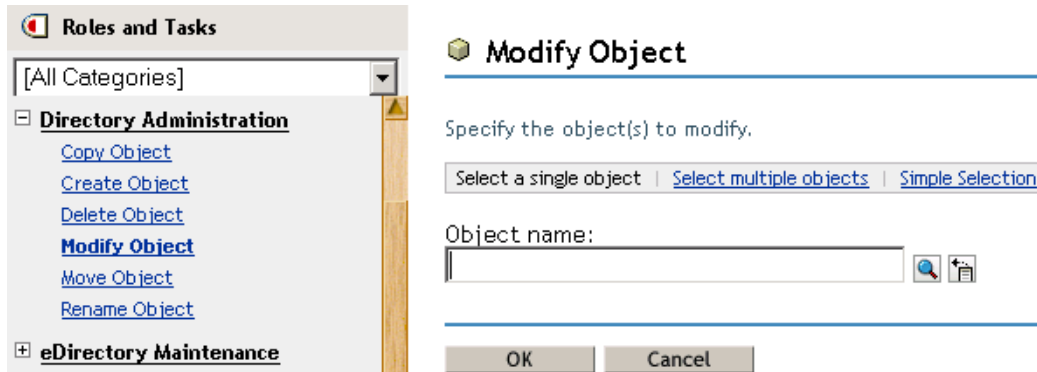
To unload LDAP:

- 1 In the *Control Panel*, double-click *Novell eDirectory*.
- 2 Scroll to and select `ldap.dlm`, then click *Stop*.



To reconfigure LDAP to run on a different port:

- 1 In Novell iManager, select *eDirectory Administration > Modify Object*, then click *OK*.



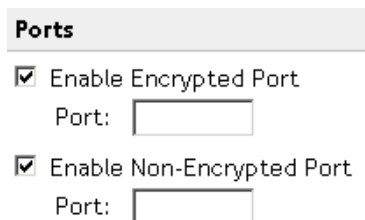
- 2 Navigate to and select the LDAP Server object, then click *OK*.



- 3 Select *General > Connections*.



- 4 Scroll to the *Ports* section.



- 5 Change *Enable Non-Encrypted Port* to a value other than 389, then click *OK*.

If another LDAP service is already using port 389, change the Exchange server's LDAP port number so that it doesn't conflict with the other service. If you change the Exchange server's



port number, also change the LDAP port that the Identity Manager Driver for Exchange looks at. See [Section 5.9, “Specifying the LDAP Port,”](#) on page 40.

## 5.2 Installing a Remote Exchange Driver

The driver doesn't need to run on the same machine as the Exchange Server. However, when running remotely, the driver can run only on an NT server or member server that belongs to the same domain as the Exchange server domain. This restriction is a Microsoft-imposed NT credential restriction.

The NT server where you install the driver needs to have three Microsoft .dll files installed before the driver can run: `libxds.dll`, `exchmem.dll`, and `expsrv.dll`. The files are installed by the Exchange Administrator program. You can install Exchange Administrator from the *Microsoft Exchange Server CD*.

A remote driver doesn't create NT accounts when a new Exchange mailbox is created. This is also because of restrictions imposed by the Microsoft DAPI API that the driver uses.

For instructions on installing the Remote Loader, see [“Deciding Whether to Use the Remote Loader”](#) in the *Novell Identity Manager 3.5.1 Administration Guide*.

## 5.3 Configuring the Driver Filter

Modify the filter on the Publisher and Subscriber channels to include object classes and attributes that you want Identity Manager to process.

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Locate and select the driver set that contains the Exchange driver, then click *Search*.
- 3 Click the Exchange driver's icon to display the Identity Manager Driver Overview page.
- 4 Click the driver filter icon.

### Identity Manager Driver Overview

Driver: Exchange 5\_5.snati\_drset.novell

Overview

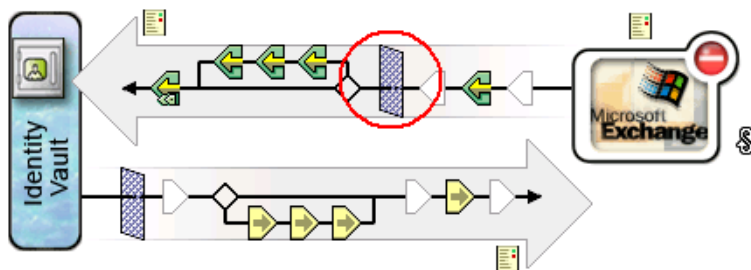
Advanced

Jobs

Export...

Migrate ▾

Synchronize...



- 5 (Optional) Add classes that you want Identity Manager to process.

The Exchange driver supports the Distribution List, Remote, and Mailbox classes.

## 6 Enable synchronization.

As the following figure illustrates with red Xs, when you add a class, the Publisher and Subscriber channels aren't enabled.



To enable a channel, click the channel icon, then click *Synchronize*.



## 7 Save changes by clicking *OK*.

Mail-nickname is the Alias attribute on the General page in the Exchange Administrator. It is the Exchange attribute name that the driver supports, but it does not map to any existing Identity Vault attributes. Based on your organization's needs, you can map this Exchange attribute to existing or new Identity Vault attributes (after extending the schema) by modifying the Schema Mapping policy. Make sure that the syntax for any maps you add is valid. You can also handle this in a style sheet.

## 5.4 Integrating the Identity Manager Driver for Exchange and the Identity Manager Driver for NT Domain

---

**IMPORTANT:** If you are using the Identity Manager Driver for NT Domain and the Identity Manager Driver for Exchange, edit the default policy or create a new one to resolve an account issue between the two drivers. This policy prevents the Exchange driver from attempting to create an NT Domain account before the NT Domain driver creates the account.

---

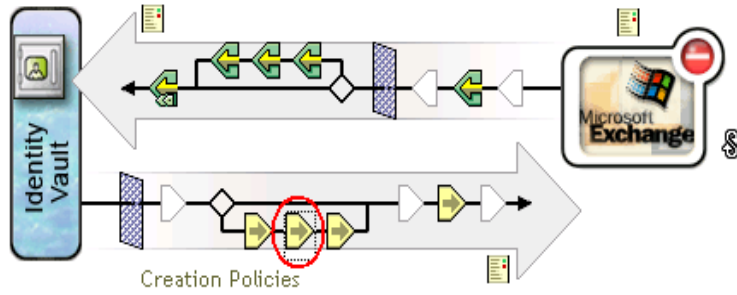
The Identity Manager Driver for NT Domain has a User attribute called DirXML-NTAccountName. This attribute contains the DomainName/UserName value. The Exchange MailBox object needs the value to associate to a domain account. For that association to occur correctly, the value in DirXML-NTAccountName needs to be put in the MailBox attribute Assoc-NT-Account. Keep in mind that attribute names are case sensitive.

### 1 Create a policy so that a new MailBox object isn't created unless the DirXML-NTAccountName attribute is populated.

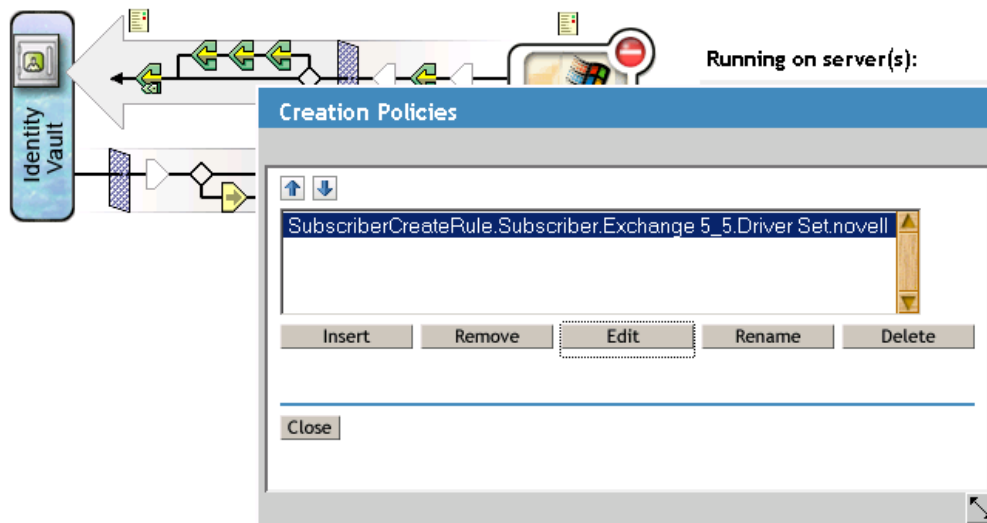
**1a** In iManager, select *Identity Manager > Identity Manager Overview*.

**1b** Search for a driver set, then double-click the Exchange 5.5 driver.

1c Select the Creation Policies object on the Subscriber channel.



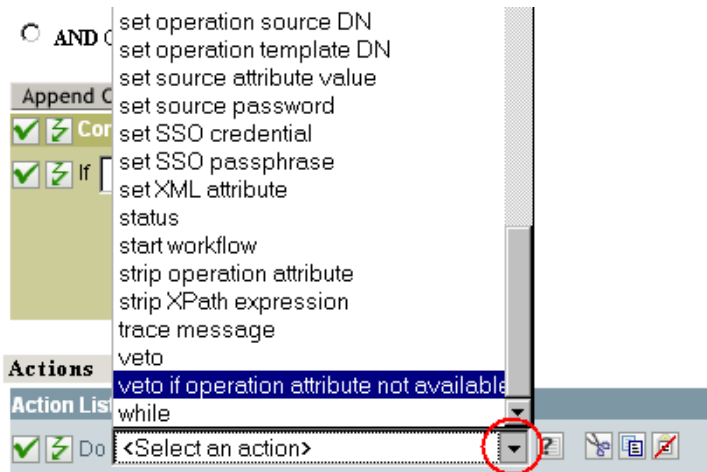
1d In the Creation Policies dialog box, click *Edit*.



1e Click *Required Attributes*.

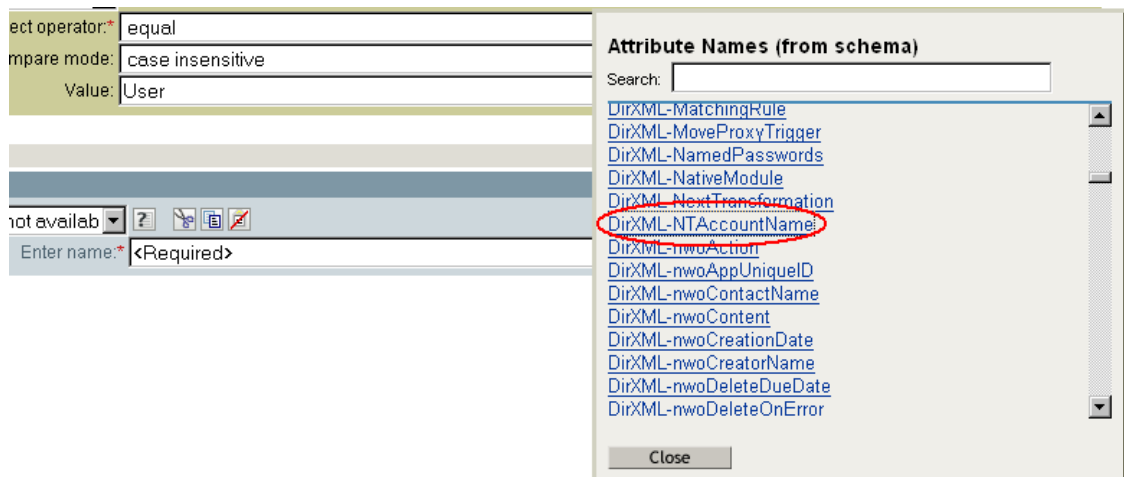


- 1f In the *Actions* section, click the drop-down list, then select *veto if operation attribute not available*("Given Name").



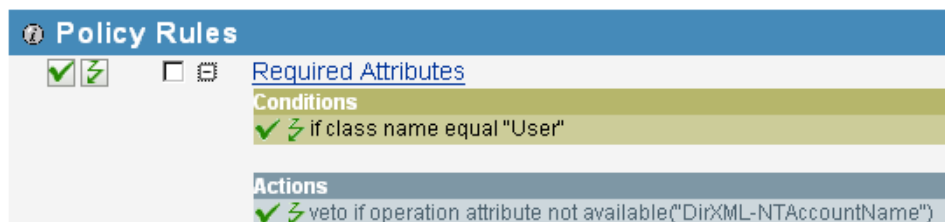
- 1g Click the Browse button by the *Enter Name* field, then select *DirXML-NTAccountName* from the drop-down list.

**NOTE:** This example uses the DirXML-NTAccountName as the attribute to hold the NT account information, but you can choose any attribute that works for you.



- 1h Click *OK*.

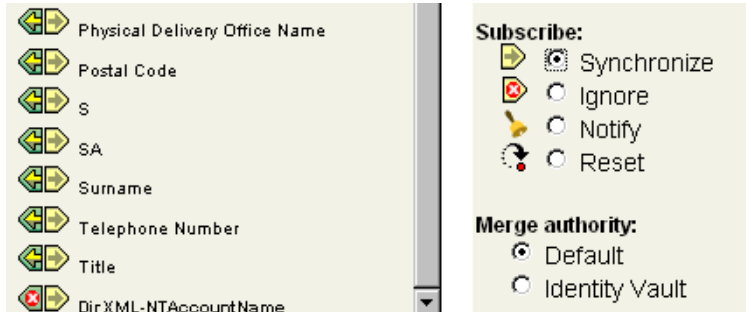
As the following expanded Required Attributes section illustrates, the action is placed in the Required Attributes section.



2 Verify that the DirXML-NTAccountName attribute is in the following locations:

- ♦ The Publisher filter on the Identity Manager Driver for NT Domains
- ♦ The Subscriber filter on the Identity Manager Driver for Exchange

3 Synchronize the Subscriber channel.



4 Restart both drivers.

After you have made these changes to the drivers, the following control flow occurs when you create a user in an Identity Vault:

1. The Identity Manager Driver for NT Domain is handed a create request.
2. The Identity Manager Driver for Exchange Create event is vetoed because of the absence of the DirXML-NTAccountName attribute.
3. The Identity Manager Driver for NT Domain creates the NT account and publishes the name of the NT account just created to the DirXML-NTAccountName attribute.
4. The Identity Manager Driver for Exchange is notified. It creates the mailbox and associates the mailbox with the NT account information stored in the Identity Vault.

## 5.5 Managing External Recipients

Microsoft Exchange directories let you create special objects called External Recipients. Think of these objects as address book entries that represent recipients in external messaging systems. You can modify the Schema Mapping policy so that you can map a remote object to a User object or any other desired Identity Vault object. For example:

```
<class-name>
  <nds-name>User</nds-name>
  <app-name>remote</app-name>
</class-name>
```

If you decide to make this change, you should also add the Internet EMail Address attribute as a required attribute to the Create policy as shown in the following example:

```
<create-rules>
  <create-rule class-name="User">
    <required-attr attr-name="Given Name"/>
  </create-rule>
</create-rules>
```

An Internet EMail Address attribute is required to create an External Recipient object in the Exchange directory. Failure to add the Internet EMail Address attribute results in an error when you try to create an External Recipient.

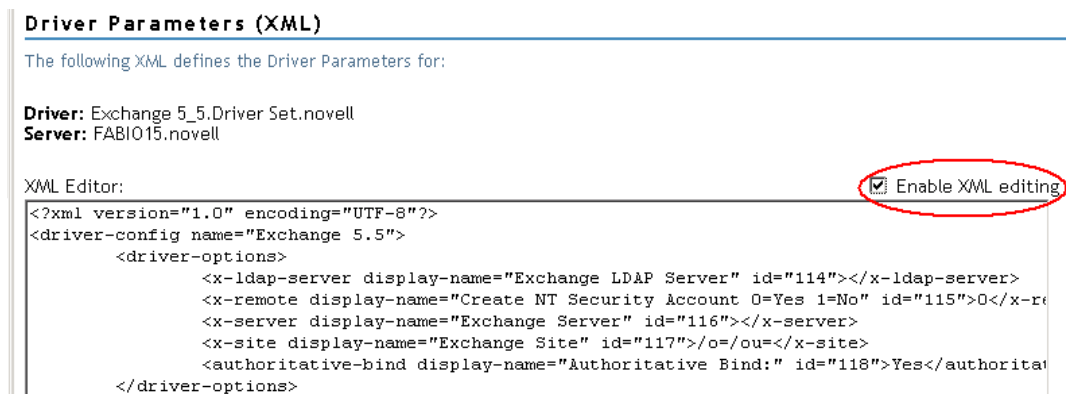
## 5.6 Synchronizing Proxy-Address and Target-Address Attributes

To synchronize all the e-mail values of the multivalued Proxy-Address and Target-Address attributes, add the `<proxyFlg/>` tag to the driver parameters.

- 1 In iManager, click *eDirectory Administration > Modify Object*.
- 2 Locate and select the Identity Manager Driver for Exchange object (for example, EXCHANGE\_5\_5), then click *OK*.
- 3 Locate the *Driver Parameters* section, then click *Edit XML*.



- 4 Select *Enable XML Editing* so that you can edit the script.



- 5 Add the `<proxyFlg/>` tag.

This tag can go anywhere between the `<driver-options>` tags. The updated parameters could look similar to the following example:

```
<driver-options>
  <x-ldap-server display-name="IP address of Exchange Server (for LDAP queries) :">167.55.135.28</x-ldap-server>
  <x-remote display-name="Remote Exchange Driver? (1=true; 0=false) :">0</x-remote>
  <x-server display-name="Exchange Server Name:">DHEAD</x-server>
  <x-site display-name="Exchange Site:">/o=Novell/ou=DOMAINLIMA</x-site>
  <proxyFlg/>
</driver-options>
```

- 6 Deselect *Enable XML Editing*, then return to the *Driver Parameters* section.
- 7 Click *OK*.

## 5.7 Using Authoritative Bind

Whenever a query happens with a scope of subordinate or subtree, the driver uses LDAP. In the past, only an anonymous bind was possible. When using an anonymous bind, the driver can't see attributes that are hidden in Exchange.

The new `AuthoritativeBind` parameter lets you use an authoritative LDAP bind instead of an anonymous LDAP bind. This option is one of the prompts when you import the sample driver configuration.

We recommend that you use authoritative bind only in cases where you need to see hidden attributes, such as when you want to do matching based on a hidden attribute.

Keep in mind that when you use authoritative bind, hidden attributes, such as `NT4AccountName`, are seen in the trace. After using authoritative bind for a specific purpose such as migrating users, if you no longer need to use authoritative bind, you could change the driver parameters back to using anonymous bind.

## 5.8 Using a Custom Bind

You might need to bind to LDAP by using a custom bind. For example, to find hidden objects in Exchange, you need to bind as user admin.

To use a custom bind:

- 1 In iManager, select `Directory Administration > Modify Object`.
- 2 Locate and select the Identity Manager Driver for Exchange object (for example, `EXCHANGE_5_5`), then click *OK*.
- 3 Locate the *Driver Parameters* section at the bottom of the Driver Configuration page, then click *Edit XML*.

**Driver Parameters**

---

S3K-NDS.Vmp

[Edit XML](#)

**Driver Settings**

Exchange LDAP Server	<input type="text"/>
Create NT Security Account 0=Yes 1=No	<input type="text" value="0"/>
Exchange Server	<input type="text"/>
Exchange Site	<input "="" type="text" value="/o=/ou="/>
Authoritative Bind:	<input type="text" value="Yes"/>

- 4 Click *Enable XML Editing*.
- 5 In the `<driver-options>` section, add the tags and string that specify a custom bind.

For example, type

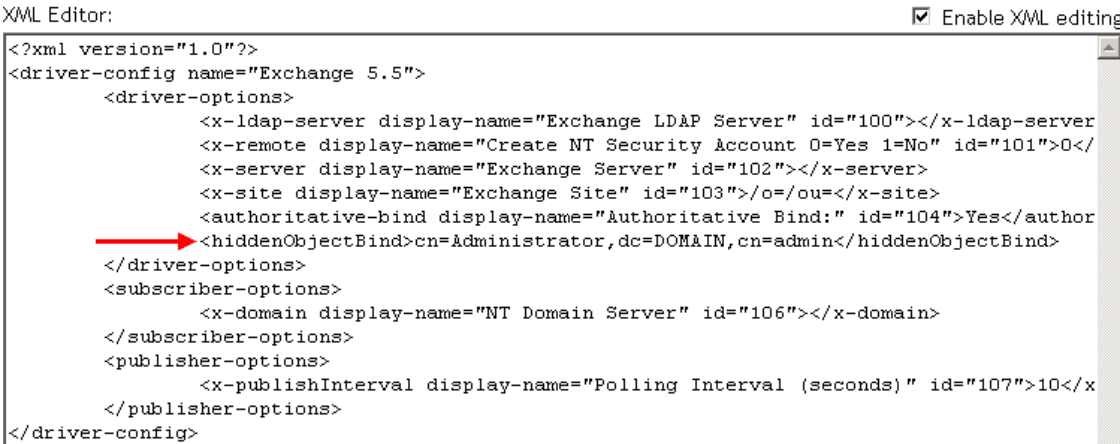
```
<hiddenObjectBind>cn=Administrator,dc=DOMAIN,cn=admin<hiddenObjectBind/>
```

This example uses the following, which you need to customize in your configuration:

String	Description
Administrator	A user with administrative rights
DOMAIN	The name of your domain

The following figure illustrates these tags and string:

**Driver:** Exchange 5\_5.hraun\_set.Vmp  
**Server:** 53K-NDS.Vmp



XML Editor:  Enable XML editing

```
<?xml version="1.0"?>
<driver-config name="Exchange 5.5">
  <driver-options>
    <x-ldap-server display-name="Exchange LDAP Server" id="100"></x-ldap-server
    <x-remote display-name="Create NT Security Account 0=Yes 1=No" id="101">0</
    <x-server display-name="Exchange Server" id="102"></x-server>
    <x-site display-name="Exchange Site" id="103">/o=/ou=</x-site>
    <authoritative-bind display-name="Authoritative Bind:" id="104">Yes</author
    <hiddenObjectBind>cn=Administrator,dc=DOMAIN,cn=admin</hiddenObjectBind>
  </driver-options>
  <subscriber-options>
    <x-domain display-name="NT Domain Server" id="106"></x-domain>
  </subscriber-options>
  <publisher-options>
    <x-publishInterval display-name="Polling Interval (seconds)" id="107">10</x
  </publisher-options>
</driver-config>
```

6 Click *OK* twice.

The Exchange driver then uses the string in the tag as the user for the bind.

Also, an additional value to not allow deleted objects is placed in the search filter.

If this custom tag is present, it overrides the authoritative bind tag. If it isn't present, the authoritative bind takes precedence. If neither tag is present, an anonymous bind is used.

## 5.9 Specifying the LDAP Port

If you changed the Exchange Server's port number from the default 389 value, you need to configure the Identity Manager Driver for Exchange, so that it looks at the port that the server uses.

- 1 In iManager, click *eDirectory Administration > Modify Object*.
- 2 Locate and select the Identity Manager Driver for Exchange object (for example, EXCHANGE 5\_5), then click *OK*.
- 3 Locate the *Driver Parameters* section at the bottom of the Driver Configuration page, then click *Edit XML*.
- 4 Click *Enable XML Editing*.



- 5 In the <driver-options> section, add the tags (<ldap-port></ldap-port> and value (for example, 391) that specify the different port number.

**Driver:** Exchange 5\_5.hraun\_set.Vmp  
**Server:** 53K-NDS.Vmp

XML Editor:  Enable XML editing

```
<?xml version="1.0"?>
<driver-config name="Exchange 5.5">
  <driver-options>
    <x-ldap-server display-name="Exchange LDAP Server" id="100"></x-ldap-server
    <x-remote display-name="Create NT Security Account 0=Yes 1=No" id="101">0</
    <x-server display-name="Exchange Server" id="102"></x-server>
    <x-site display-name="Exchange Site" id="103">/o=/ou=</x-site>
    <authoritative-bind display-name="Authoritative Bind:" id="104">Yes</author
    <hiddenObjectBind id="105">cn=Administrator,dc=DOMAIN,cn=admin</hiddenObjec
    <ldap-port>391</ldap-port>
  </driver-options>
```

- 6 Click *OK* twice.



# Activating the Exchange Driver

# 6

Activate the driver within 90 days of installation. Otherwise, the driver will stop working.

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



# Managing the Exchange Driver

# 7

- ♦ Section 7.1, “Starting, Stopping, or Restarting the Exchange Driver,” on page 45
- ♦ Section 7.2, “Migrating and Resynchronizing Data,” on page 46
- ♦ Section 7.3, “Using the DirXML Command Line Utility,” on page 46
- ♦ Section 7.4, “Viewing Driver Version Information,” on page 46
- ♦ Section 7.5, “Reassociating a Driver Set Object with a Server Object,” on page 51
- ♦ Section 7.6, “Changing the Driver Configuration,” on page 52
- ♦ Section 7.7, “Storing Driver Passwords Securely with Named Passwords,” on page 52
- ♦ Section 7.8, “Adding a Driver Heartbeat,” on page 59

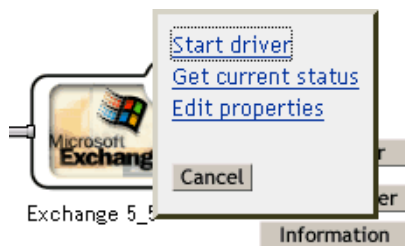
## 7.1 Starting, Stopping, or Restarting the Exchange Driver

In Designer:

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

In iManager:

- 1 If you changed default data locations during configuration, ensure that the new locations exist before you start the driver.
- 2 Click *Identity Manager > Identity Manager Overview*.
- 3 Select the driver set where the driver exists.
- 4 Click the driver status indicator in the upper right corner of the driver icon, then click *Start driver, Stop driver, or Restart driver*.



If a change log is available, the driver processes all the changes in the change log. To force an initial synchronization, see “[Migrating and Resynchronizing Data](#)” on page 46.

## 7.2 Migrating and Resynchronizing Data

Identity Manager synchronizes data when the data changes. If you want to synchronize all data immediately, you can choose from the following options:

- ♦ **Migrate Data from Identity Vault:** Allows you to select containers or objects you want to migrate from the Identity Vault to an application. When you migrate an object, the Identity Manager engine applies all of the Matching, Placement, and Create policies, as well as the Subscriber filter, to the object.
- ♦ **Migrate Data into Identity Vault:** Allows you to define the criteria that the Identity Manager engine uses to migrate objects from an application into Novell® eDirectory. When you migrate an object, the Identity Manager engine applies all of the Matching, Placement, and Create policies, as well as the Publisher filter, to the object. Objects are migrated into eDirectory by using the order you specify in the Class list.
- ♦ **Synchronize:** The Identity Manager engine looks in the Subscriber class filter and processes all objects for those classes. Associated objects are merged. Unassociated objects are processed as Add events.

To use one of the options explained above:

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Browse to and select the driver set where the driver exists, then click *Search*.
- 3 Click the driver icon.
- 4 Click the appropriate migration button.

For more information, see [Chapter 8, “Synchronizing Objects,” on page 61](#).

## 7.3 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, “The DirXML Command Line Utility,” on page 81](#) for information about the DirXML Command Line utility. For daily tasks, use iManager or Designer.

## 7.4 Viewing Driver Version Information

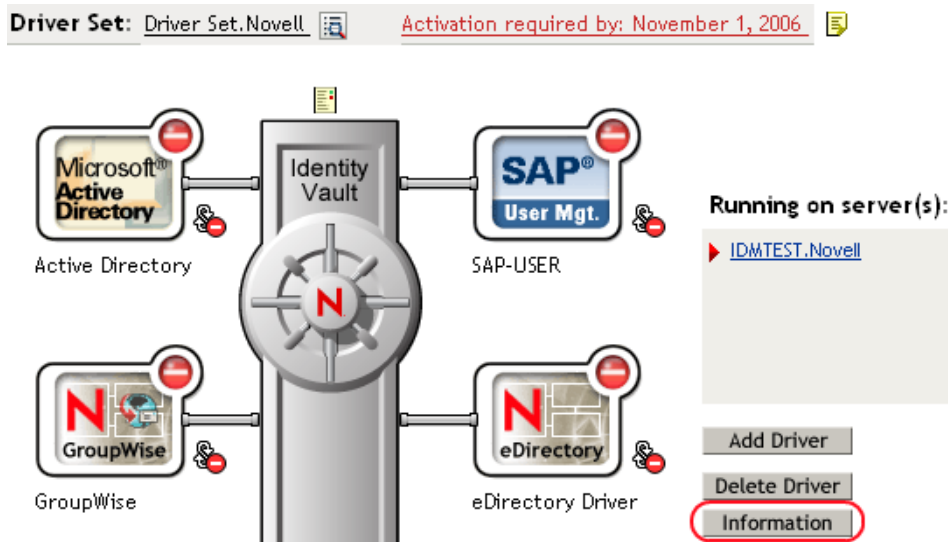
To view information on versions of Identity Manager and versions of drivers, use the Versioning Discovery tool. This tool exists only in iManager.

- ♦ [Section 7.4.1, “Viewing a Hierarchical Display of Version Information,” on page 46](#)
- ♦ [Section 7.4.2, “Viewing the Version Information As a Text File,” on page 48](#)
- ♦ [Section 7.4.3, “Saving Versioning Information,” on page 50](#)

### 7.4.1 Viewing a Hierarchical Display of Version Information

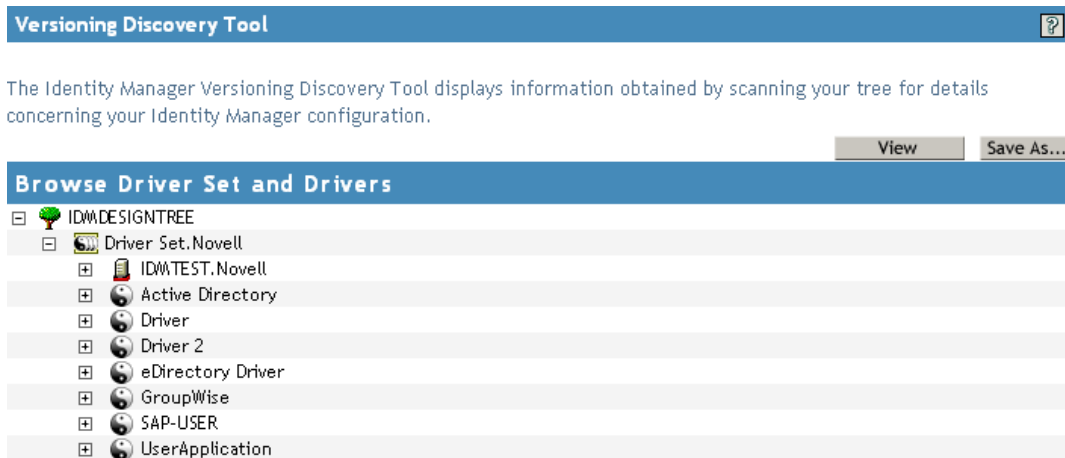
- 1 To find your Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.

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 version information.



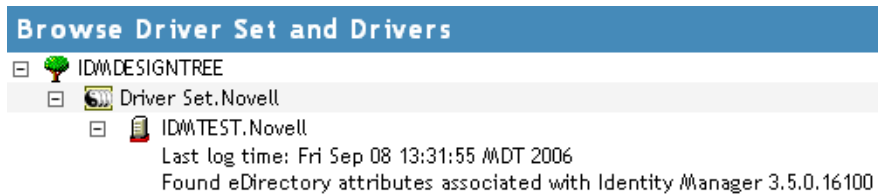
The unexpanded hierarchical view displays the following:

- ◆ 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

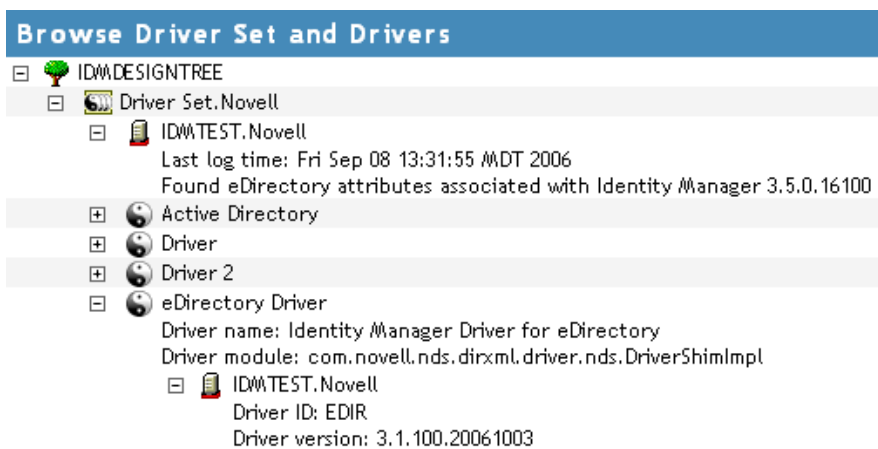
- 4 View version information related to servers by expanding the server icon.



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 version 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

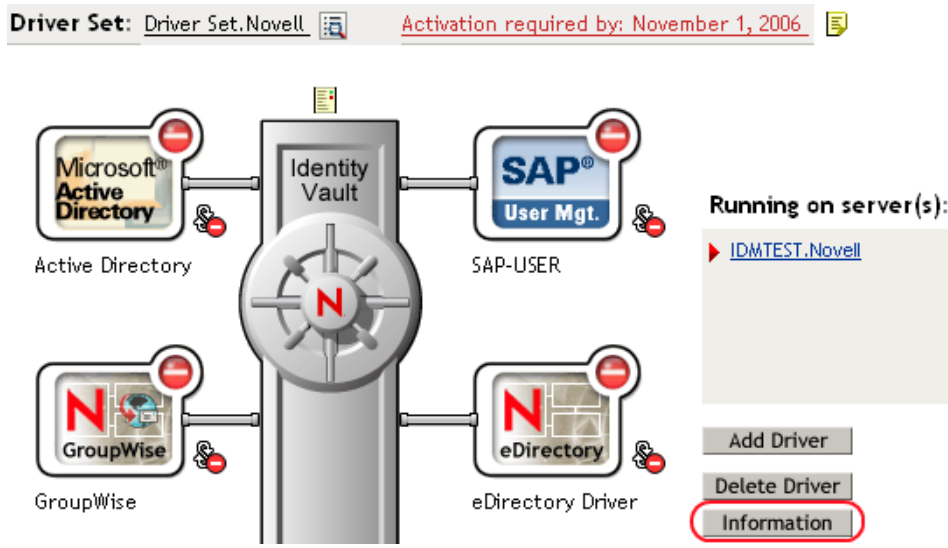
## 7.4.2 Viewing the Version Information As a Text File

Identity Manager publishes version 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*.

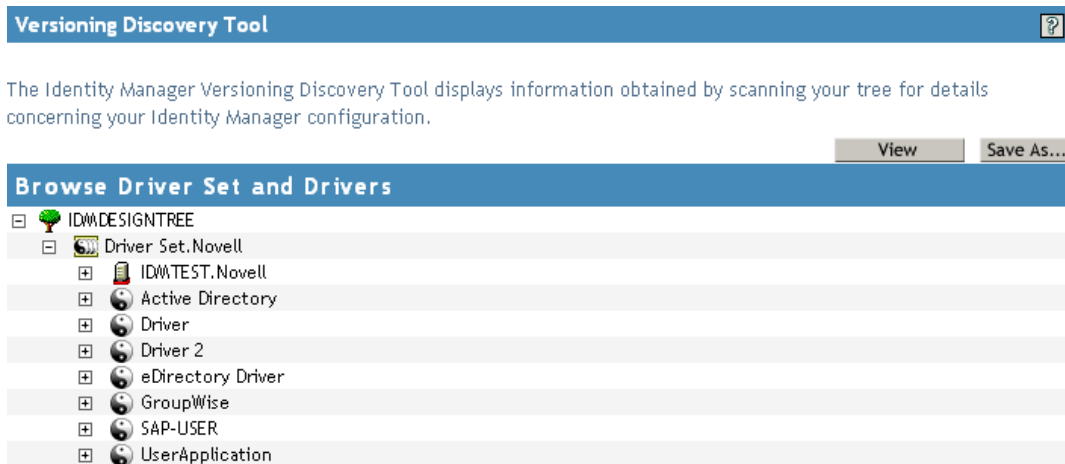


2 In the Identity Manager Overview, click *Information*.



You can also select *Identity Manager Utilities > Versions 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:  IDMDSIGNTREE
  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 Excha
  Driver module:  addriver.dll
  Driver Set running on Identity Vault:  IDMTEST.Novell
  Didn't find any DirXML-DriverVersion attributes associated w:
  This may mean the Metadirectory engine is older than
  It does not indicate anything about the version of t
Driver:  Driver.Driver Set.Novell
  Driver name:  Identity Manager Driver for Peoplesoft
  Driver module:  NPSShim.dll
  Driver Set running on Identity Vault:  IDMTEST.Novell
```

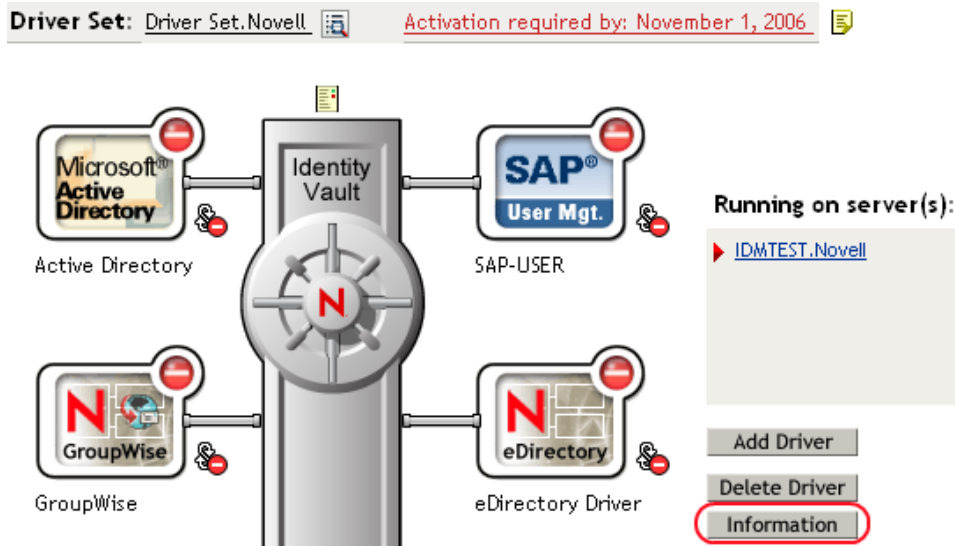
OK

### 7.4.3 Saving Versioning Information

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

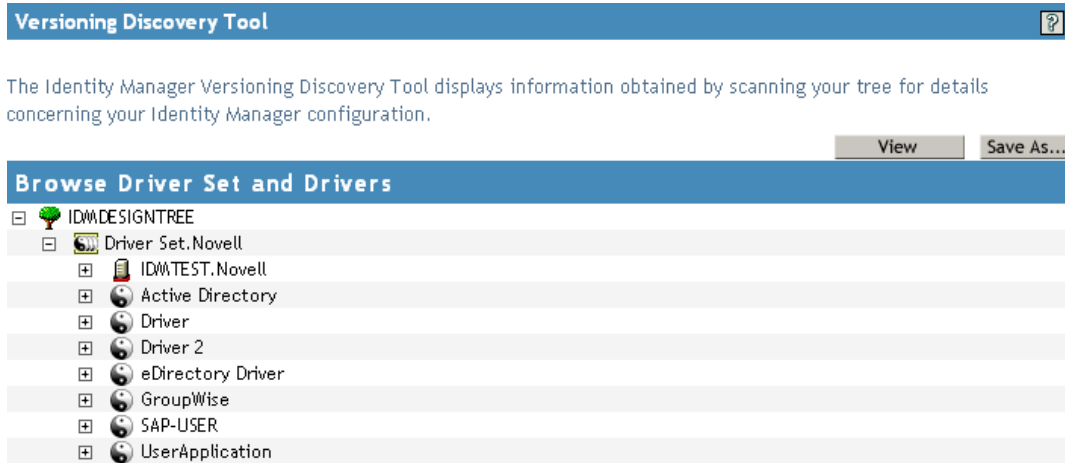
- 1 To find the Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.

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



You can also select *Identity Manager Utilities > Versions 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.

## 7.5 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 object 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 on 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*.

## 7.6 Changing the Driver Configuration

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

To change the driver configuration in Designer:

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

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*.

For a list of all of the configuration fields, see [Appendix B, “Properties of the Exchange Driver,” on page 95](#).

## 7.7 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.

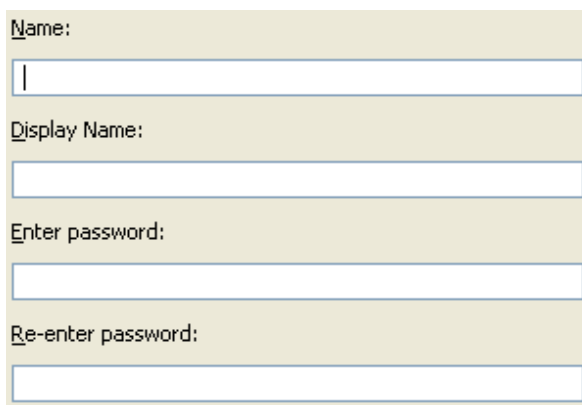
To use a named password in a driver policy, you refer to it by the name of the password, instead of using the actual password, and the Metadirectory engine sends the password to the driver. The

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 7.7.1, “Using Designer to Configure Named Passwords,” on page 53](#)
- ♦ [Section 7.7.2, “Using iManager to Configure Named Passwords,” on page 53](#)
- ♦ [Section 7.7.3, “Using Named Passwords in Driver Policies,” on page 55](#)
- ♦ [Section 7.7.4, “Using the DirXML Command Line Utility to Configure Named Passwords,” on page 55](#)

## 7.7.1 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:

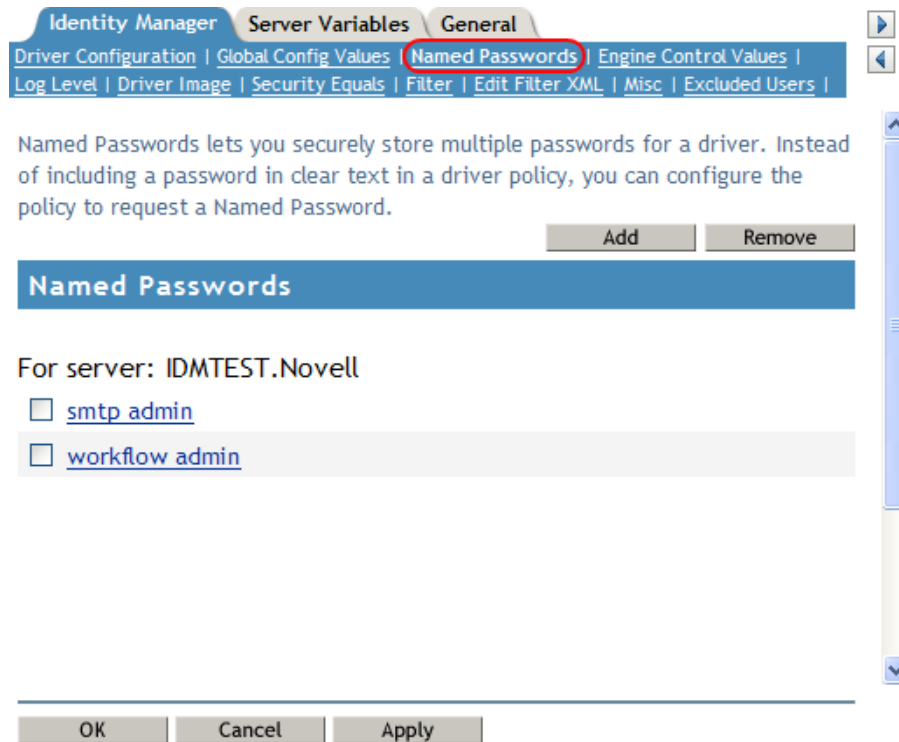
Re-enter password:

- 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.

## 7.7.2 Using iManager to Configure Named Passwords

- 1 Click *Identity Manager > Identity Manager Overview*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 In the Identity Manager Overview, click the upper right corner of the driver icon, then click *Edit properties*.
- 4 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.



- 5 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:

- 6 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.

7 Click *OK* to restart the driver and have the changes take effect.

To remove a Named Password, select the password name, then click *Remove*. The password is removed without prompting you to confirm the action.

### 7.7.3 Using Named Passwords in Driver Policies

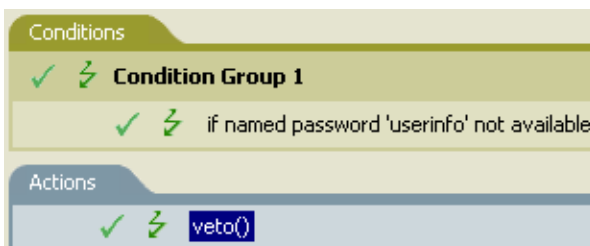
- ♦ “Making a Call to a Named Password” on page 55
- ♦ “Referencing a Named Password” on page 55

#### Making a Call to a Named Password

- 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, the named password 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 7-1** A Policy Using Named Passwords



#### Referencing a Named Password

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/>
```

### 7.7.4 Using the DirXML Command Line Utility to Configure Named Passwords

- ♦ “Creating a Named Password in the DirXML Command Line Utility” on page 56
- ♦ “Removing a Named Password by Using the DirXML Command Line Utility” on page 57

## Creating a Named Password in the DirXML Command Line Utility

- 1 Run the DirXML Command Line utility.

For information, see [Appendix A, “The DirXML Command Line Utility,”](#) on page 81.

- 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, use the 99 option twice to exit the menu and quit the DirXML Command Line Utility.

## Removing a Named Password by Using the DirXML Command Line Utility

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, “The DirXML Command Line Utility,” on page 81](#).

- 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, use the 99 option twice to exit the menu and quit the DirXML Command Line utility.

## 7.8 Adding a Driver Heartbeat

The driver heartbeat is a feature of the Identity Manager drivers that ship with Identity Manager 2 and later. Using it 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 no communication occurs 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. Configuration is then complete.

The value of the interval cannot be less than 1. A value of 0 means that 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 similar to 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, then 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.

# Synchronizing Objects

# 8

This section explains driver and object synchronization in DirXML<sup>®</sup> 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 61](#)
- ♦ [Section 8.2, “When Does Synchronization Occur?” on page 61](#)
- ♦ [Section 8.3, “How Does the Metadirectory Engine Decide Which Object to Synchronize?” on page 62](#)
- ♦ [Section 8.4, “How Synchronization Works,” on page 63](#)

## 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 Does Synchronization Occur?

The Metadirectory engine synchronizes objects or merges them in the following circumstances:

- ♦ When a `<sync>` event element is submitted on the Subscriber or Publisher channel.
- ♦ When 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.
- ♦ When 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. The engine submits these <sync> events by using the Subscriber thread, but processes them by using the Publisher channel filter and policies.
- ♦ When an <add> event (real or synthetic) is submitted on a channel, and the channel Matching policy finds a matching object in the target system.
- ♦ When 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.
- ♦ When 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<sup>®</sup>, 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 explained in [“How Does the Metadirectory Engine Decide Which Object to Synchronize?” on page 62](#).

## 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. DirXML<sup>®</sup> 1.1a has 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 following:
  - ♦ The driver cache for each unique object found that has an entry modification time stamp greater than or equal to the starting filter time
  - ♦ All objects and classes that are in the Subscriber filter channel in the driver being synchronized

## 8.4 How Synchronization Works

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 64](#), [Table 8-2 on page 66](#), and [Table 8-1 on page 64](#).

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 sides receives the union of the values of the left and right sides.

Identity Manager has three different combinations of selected items in the filter, and each one creates a different output.

- ♦ [Section 8.4.1, “Scenario One,” on page 63](#)
- ♦ [Section 8.4.2, “Scenario Two,” on page 65](#)
- ♦ [Section 8.4.3, “Scenario Three,” on page 66](#)

### 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

**Class Name: User**  
**Attribute Name: Facsimile Telephone Num**

**Publish**

Synchronize  
 Ignore  
 Notify  
 Reset

---

**Subscribe**

Synchronize  
 Ignore  
 Notify  
 Reset

---

**Merge Authority**

Default  
 Identity Vault  
 Application  
 None

---

**Optimize modifications to Identity Vault**

Yes  
 No

Table 8-1 on page 64 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 the following:

- ◆ 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.
- ◆ 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
<b>Application single-valued empty</b>	No change	App = Identity Vault	No change	App = Identity Vault[1]
<b>Application single-valued non-empty</b>	Identity Vault = App	App = Identity Vault	Identity Vault = App	Identity Vault + = App
<b>Application multi-valued empty</b>	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
<b>Application multi-valued non-empty</b>	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

Table 8-2 on page 66 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 the following:

- ◆ Whether the attribute comes from the Identity Vault or the Application
- ◆ If the attribute is single-valued or multi-valued
- ◆ 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
<b>Application single-valued empty</b>	No change	App = Identity Vault	No change	App = Identity Vault[1]
<b>Application single-valued empty</b>	App = empty	App = Identity Vault	Identity Vault = App	App = Identity Vault[1]
<b>Application multi-valued empty</b>	No change	App = Identity Vault	No change	App = Identity Vault
<b>Application multi-valued non-empty</b>	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

**Class Name: User**

**Attribute Name: DirXML-ADAliasName**

**Publish**

Synchronize

Ignore

Notify

Reset

**Subscribe**

Synchronize

Ignore

Notify

Reset

**Merge Authority**

Default

Identity Vault

Application

None

**Optimize modifications to Identity Vault**

Yes

No

Table 8-3 on page 67 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 the following:

- ◆ Whether the attribute comes from the Identity Vault or the Application
- ◆ If the attribute is single-valued or multi-valued
- ◆ If the attribute is empty or non-empty

**Table 8-3** *Output of Scenario Three*

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



# Troubleshooting the Driver

# 9

This section provides information on the following:

- ♦ [Section 9.1, “Troubleshooting Tips,” on page 69](#)
- ♦ [Section 9.2, “Driver Error Messages,” on page 69](#)
- ♦ [Section 9.3, “Troubleshooting Driver Processes,” on page 71](#)

## 9.1 Troubleshooting Tips

- ♦ Exchange Directory names used in the Placement policy are case sensitive.
- ♦ If Exchange and the Identity Vault are running on the same machine, you must disable the Identity Vault LDAP server or change the port assignment from port 389.
- ♦ If you encounter syntax errors in XML, verify the syntax by using Identity Manager and a Web browser.
- ♦ The authentication credentials you specify should be for an NT account/domain that has rights to the Exchange Directory.
- ♦ CN and Object-Class should not be in the filter.
- ♦ If there is an invalid attribute in the filter (for example, one for which Schema Mapping is not defined), the following happens:
  - ♦ At each polling loop, the Publisher fails to synchronize the class that contains the invalid attribute, and gives an error indicating an unsupported attribute.
  - ♦ The driver is still able to start.
  - ♦ The Subscriber channel still functions correctly unless the invalid attribute is referenced. A task that references an invalid attribute gives an error and is not successful.

The sample driver configuration supports the standard attributes in Exchange, so this issue could occur only for custom attributes you have added in Exchange.

- ♦ The Identity Manager Driver for Exchange supports the following classes:
  - ♦ Distribution List
  - ♦ Remote
  - ♦ Mailbox

## 9.2 Driver Error Messages

The following is a list of error messages the driver might return:

- ♦ `USN Cache Initialized from disk`

This is an informational message printed at the beginning of the driver initialization. The message indicates that the driver’s last state was read from the disk.

- ♦ `USN Cache could not be initialized. Most likely reason: Insufficient memory.`

The previous driver state was not initialized correctly. This means that another process manipulated the registry or the driver configuration file. This might happen if you accidentally deleted one of the driver configuration files. This results in a loss of event data.

- ◆ DAPIStart() failed. Please check the Event Log for details. Returned error code =

This usually suggests that the Exchange Server is down or could not be reached. The NT Application Event Log should contain a more detailed description of the error.

- ◆ DAPIStart() encountered non fatal error. Please check the Event Log for details. Returned error code =

This error message is returned when Exchange returned a warning. The returned warning is logged in the NT Application Event Log.

- ◆ Call to Import function failed. Likely cause- bad XML or too little memory.
- ◆ Subscriber Import attempt failed. Please check the NT Event Log for details. Returned error code =

An attempt to write to the Event log did not succeed.

- ◆ Subscriber Import attempt encountered non critical error. Please check the NT Event Log for details. Returned error code =
- ◆ A bad XML document was sent to the driver. Critical error in the Subscriber XML document.
- ◆ A bad XML document was sent to the driver. The subscriber could not process the input node.
- ◆ A bad XML document was sent to the driver. An unsupported operation type was received.
- ◆ A bad XML document was sent to the driver. The operation node could not be processed.

- ◆ Bad Subscriber filter or Subscriber filter contains an unsupported attribute. Please check the Subscriber filter.

An unsupported attribute was added to the Subscriber filter. Verify the Subscriber filter with the list of supported attributes in the ATTRIBUTES.TXT file (located in the NT\DIRXML\DRIVERS\EXCHANGE\RULES directory).

- ◆ Could not log in to Exchange with the specified credentials. Driver will not start.

The authentication credentials supplied were incorrect. The credentials specified should be for an NT account/domain that has rights to the Exchange directory.

- ◆ NT Event handles could not be created. The System could be low on memory
- ◆ An expected initialization parameter was missing from the parameter list.

One or more of the initialization parameters was missing. Try restarting. If the problem persists, try retyping the initialization parameters.

- ◆ Could not allocate memory.
- ◆ Failed to initialize the base of the USN cache. Driver will not start.

## 9.3 Troubleshooting Driver Processes

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

### 9.3.1 Viewing Driver Processes

To see the driver processes in DSTrace, values are added to the Driver Set object and the Driver object. You can do this in Designer or iManager.

- ◆ “Adding Trace Levels in Designer” on page 71
- ◆ “Adding Trace Levels in iManager” on page 72
- ◆ “Capturing Driver Processes to a File” on page 73

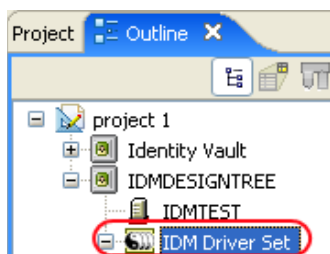
#### Adding Trace Levels in Designer

You can add trace levels to the Driver Set object or to each Driver object.

- ◆ “Driver Set” on page 71
- ◆ “Driver” on page 72

#### Driver Set

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



- 2 Right-click, 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. 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.

Parameter	Description
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.
Trace file size limit	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 no disk space remains.

If you set the trace level on the Driver Set object, all drivers appear in the DSTrace logs.

## Driver

- 1 In an open project in Designer, select the Driver object in the *Outline* view.
- 2 Right-click, select *Properties*, then click *8. Trace*.
- 3 Set the parameters for tracing, then click *OK*.

Parameter	Description
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. To see password synchronization information, set the trace level to 5.  if you select <i>Use setting from Driver Set</i> , the value is taken from the Driver Set object.
Trace file	Specify a filename and location for where the Identity Manager information is written for the selected driver.  if you select <i>Use setting from Driver Set</i> , the value is taken from the Driver Set object.
Trace file size limit	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 no disk space remains.  If you select <i>Use setting from Driver Set</i> , the value is taken from the Driver Set object.
Trace name	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.

If you set the parameters only on the Driver object, only information for that driver appears in the DSTrace log.

## Adding Trace Levels in iManager

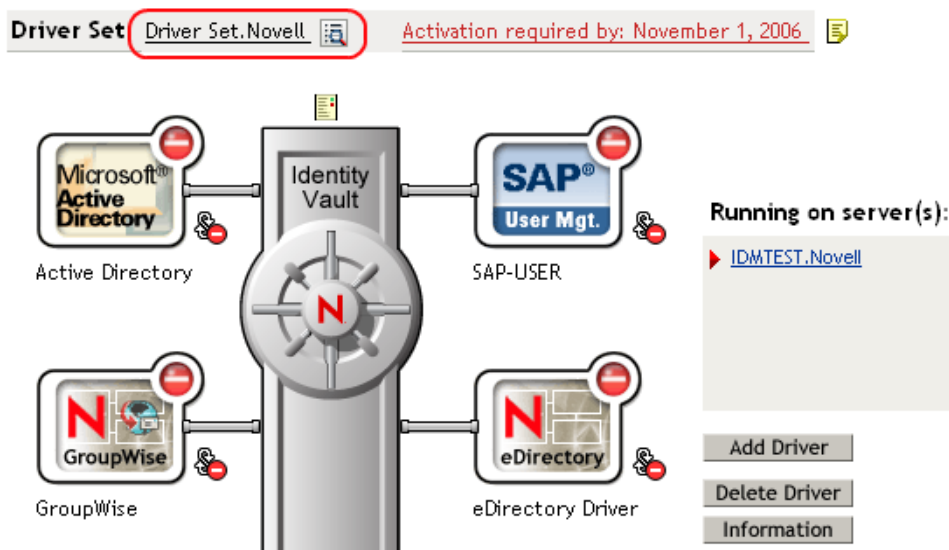
You can add trace levels to the Driver Set object or to each Driver object.

- ◆ “Driver Set” on page 73
- ◆ “Driver” on page 73



## 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*.

## 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*.

---

**NOTE:** 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 74

- ◆ “Windows” on page 74
- ◆ “UNIX” on page 75
- ◆ “iMonitor” on page 75
- ◆ “Remote Loader” on page 76

## 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`s 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, select *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.
- ♦ `ndsmonitor` 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 the time stamp.

If you need a copy of the HTML file, the default location is:

- ♦ NetWare: `sys:\system\ndsimon\dstrace*.htm`
- ♦ Windows: `Drive_letter:\novell\nds\ndsimon\dstrace\*.htm`
- ♦ UNIX: `/var/nds/dstrace/*.htm`

## Remote Loader

You can capture the events that occur on the machine running the Remote Loader service.

- 1 Launch the Remote Loader Console by clicking the icon.
- 2 Select the driver instance, then click *Edit*.
- 3 Set the *Trace Level* to 3 or above.
- 4 Specify a location and file for the trace file.
- 5 Specify the amount of disk space that the file is allowed.
- 6 Click *OK* twice to save the changes.

You can also enable tracing from the command line by using the following switches. For more information, see “[Configuring the Remote Loader](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

**Table 9-1** Command Line Tracing Switches

Option	Short Name	Parameter	Description
-trace	-t	integer	<p>Specifies the trace level. This is used only when hosting an application shim. Trace levels correspond to those used on the Identity Manager server.</p> <p>Example: <code>-trace 3</code> or <code>-t3</code></p>
-tracefile	-tf	filename	<p>Specify a file to write trace messages to. Trace messages are written to the file if the trace level is greater than zero. Trace messages are written to the file even if the trace window is not open.</p> <p>Example: <code>-tracefile c:\temp\trace.txt</code> or <code>-tf c:\temp\trace.txt</code></p>
-tracefilemax	-tfm	size	<p>Specifies the approximate maximum size that trace file data can occupy on disk. If you specify this option, Identity Manager creates a trace file with the name specified by using the tracefile option and up to 9 additional “roll-over” files. The roll-over files are named by 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 Exchange Driver

# 10

You can use Designer for Identity Manager or iManager to create an XML file of the driver. The file contains all of the information that you entered into the driver during configuration. If the driver becomes corrupted, you can restore the configuration information by importing the exported file.

---

**IMPORTANT:** If the driver has been deleted, all of the associations on the objects are purged. When you import the XML file, the migration process creates new associations.

---

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

- ♦ [Section 10.1, “Exporting the Driver in Designer,” on page 77](#)
- ♦ [Section 10.2, “Exporting the Driver in iManager,” on page 77](#)

## 10.1 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.

## 10.2 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 that 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

# 11

To secure the driver and the information it is synchronizing, see “[Security: Best Practices](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.





# The DirXML Command Line Utility

# A

The DirXML<sup>®</sup> Command Line utility allows you to use a command line interface to manage the driver. You can create scripts that have the commands to manage the driver.

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<sup>®</sup>: sys:\system\dxcmd.ncf
- ♦ UNIX: /usr/bin/dxcmd

Either of the following methods enable you to use the DirXML Command Line utility:

- ♦ [Section A.1, “Interactive Mode,” on page 81](#)
- ♦ [Section A.2, “Command Line Mode,” on page 90](#)

## 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 that you want to perform.  
[Table A-1 on page 82](#) contains the list of options and what functionality is available.
- 5 To quit the utility, enter 99.

**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 more than one driver exists, 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 more than one driver exists, 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 more than one driver exists, each driver is listed with a number. Enter the number of the driver to see the operations available. See <a href="#">Table A-2 on page 83</a> 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"><li>◆ 1: Associate driver set with server</li><li>◆ 2: Disassociate driver set from server</li><li>◆ 99: Exit</li></ul>
5: <i>Log events operations</i>	Lists the operations available for logging events through Novell® Audit. See <a href="#">Table A-5 on page 87</a> for a description of these options.
6: <i>Get DirXML version</i>	Lists the installed version of Identity Manager.
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"><li>◆ 0 - Driver is stopped</li><li>◆ 1 - Driver is starting</li><li>◆ 2 - Driver is running</li><li>◆ 3 - Driver is stopping</li></ul>
4: <i>Get driver start option</i>	Lists the current driver start option. <ul style="list-style-type: none"><li>◆ 1 - Disabled</li><li>◆ 2 - Manual</li><li>◆ 3 - Auto</li></ul>
5: <i>Set driver start option</i>	Changes the start option of the driver. <ul style="list-style-type: none"><li>◆ 1 - Disabled</li><li>◆ 2 - Manual</li><li>◆ 3 - Auto</li><li>◆ 99 - Exit</li></ul>
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 <i>Yes</i>, 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 <i>No</i>, 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 <a href="http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsstd/query.html">Novell <code>nds.dtd</code></a> (<a href="http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsstd/query.html">http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsstd/query.html</a>).</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>Nine Password options are available. See <a href="#">Table A-3 on page 85</a> for a description of these options.</p>
14: <i>Cache operations</i>	<p>Five Cache operations are available. See <a href="#">Table A-4 on page 86</a> for a descriptions 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 <a href="#">Section 7.7, "Storing Driver Passwords Securely with Named Passwords,"</a> on page 52 for more information.  Lists four prompts: <ul style="list-style-type: none"> <li>◆ <i>Enter password name:</i></li> <li>◆ <i>Enter password description:</i></li> <li>◆ <i>Enter password:</i></li> <li>◆ <i>Confirm password:</i></li> </ul>

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: <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"> <li>◆ Driver Object password</li> <li>◆ Application password</li> <li>◆ Remote loader password</li> </ul> <p>The dxcmd utility enables you to set the Application password and the Remote Loader password. You cannot set the Driver Object password with this utility. It displays whether the password has been set.</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"> <li>◆ <i>Enter option token (default=0):</i></li> <li>◆ <i>Enter maximum transactions records to return (default=1):</i></li> <li>◆ <i>Enter name of file for response:</i></li> </ul>
4: <i>Delete cached transactions</i>	<p>Deletes the transactions stored in cache.</p> <ul style="list-style-type: none"> <li>◆ <i>Enter position token (default=0):</i></li> <li>◆ <i>Enter event-id value of first transaction record to delete (optional):</i></li> <li>◆ <i>Enter number of transaction records to delete (default=1):</i></li> </ul>
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. You can select 49 items to log. See <a href="#">Table A-6 on page 88</a> 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 log event options.
3: <i>Set driver log events</i>	<p>Allows you to log driver events through Novell Audit. You can select 49 items to log. See <a href="#">Table A-6 on page 88</a> 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>

<b>Operation</b>	<b>Description</b>
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*

<b>Options</b>
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** Job Scheduler Operations

Option	Description
1: Get available job definitions	Allows you to select an existing job. <ul style="list-style-type: none"><li>◆ Enter the job number:</li><li>◆ Do you want to filter the job definitions by containment? Enter Yes or No</li><li>◆ Enter name of the file for response:</li></ul> Examples: NetWare: sys:\files\user.log Windows: c:\files\user.log Linux: /files/user.log
2: Operations on specific job object	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 90](#) 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
<b>Configuration</b>	
-user <user name>	Specify the name of a user with administrative rights to the drivers you want to test.
-host <name or IP address>	Specify the IP address of the server where the driver is installed.
-password <user password>	Specify the password of the user specified above.
-port <port number>	Specify a port number, if the default port is not used.
-q <quiet mode>	Displays very little information when a command is executed.
-v <verbose mode>	Displays detailed information when a command is executed.
-s <stdout>	Writes the results of the <code>dxcmd</code> command to <code>stdout</code> .

Option	Description
-? <show this message>	Displays the help menu.
-help <show this message>	Displays the help menu.
<b>Actions</b>	
-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> ( <a href="http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy_dtd/data/dtdndsoverview.html#dtdndsoverview">http://www.novell.com/documentation/idm35/index.html?page=/documentation/idm35/policy_dtd/data/dtdndsoverview.html#dtdndsoverview</a> ).
-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 is 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 <a href="#">Table A-6 on page 88</a> 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>	Stops 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 is executed successfully, it returns a zero. If the command 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 93](#) 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	Returns 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 Exchange Driver

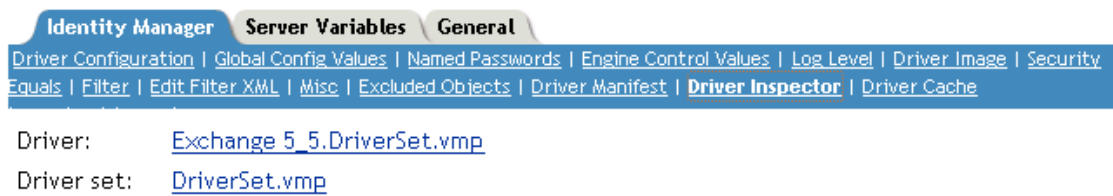
# B

This section is a reference for all of the fields on the driver as displayed in iManager and Designer. Sometimes fields are displayed differently in iManager than in Designer.

The information is presented from the viewpoint of iManager. If a field is different in Designer for Identity Manager, it is marked with a Designer  icon.

The following figure illustrates property pages in iManager:

**Figure B-1** Property Pages in iManager



- ◆ [Section B.1, “Identity Manager: Driver Configuration,” on page 95](#)
- ◆ [Section B.2, “Identity Manager: Global Configuration Values,” on page 100](#)
- ◆ [Section B.3, “Identity Manager: Named Passwords,” on page 101](#)
- ◆ [Section B.4, “Identity Manager: Engine Control Values,” on page 102](#)
- ◆ [Section B.5, “Identity Manager: Log Level,” on page 104](#)
- ◆ [Section B.6, “Driver Image,” on page 105](#)
- ◆ [Section B.7, “Security Equals,” on page 105](#)
- ◆ [Section B.8, “Filter,” on page 105](#)
- ◆ [Section B.9, “Edit Filter XML,” on page 106](#)
- ◆ [Section B.10, “Identity Manager: Misc,” on page 106](#)
- ◆ [Section B.11, “Excluded Objects,” on page 107](#)
- ◆ [Section B.12, “Driver Manifest,” on page 107](#)
- ◆ [Section B.13, “Driver Cache Inspector,” on page 108](#)
- ◆ [Section B.14, “Driver Inspector,” on page 108](#)
- ◆ [Section B.15, “Server Variables,” on page 109](#)
- ◆ [Section B.16, “Driver Inspector,” on page 112](#)

## B.1 Identity Manager: Driver Configuration

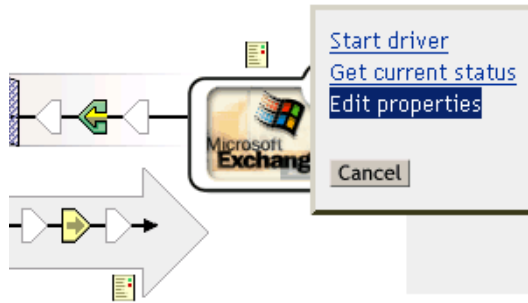
In Designer:

- 1 Open a project in the Modeler, then right-click the driver line.
- 2 Click *Properties > 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 Select the Exchange 5\_5 driver.
- 3 Click the driver's status indicator, in the upper right corner of the driver icon, then select *Edit Properties*.

**Figure B-2** Options in the Status Indicator



- 4 Click *Driver Configuration*.

To configure the Exchange driver, set parameters on the following:

- ♦ [Section B.1.1, “Driver Module,” on page 96](#)
- ♦ [Section B.1.2, “Authentication,” on page 97](#)
- ♦ [Section B.1.3, “Startup Option,” on page 98](#)
- ♦ [Section B.1.4, “Driver Parameters,” on page 99](#)
- ♦ [Section B.1.5, “ECMAScript,” on page 100](#)

## B.1.1 Driver Module

The driver module changes the driver from running locally to running remotely or the reverse.

In Designer:



- 1 Open a project in the Modeler.
- 2 Right-click the driver icon or driver line, then select *Properties > Driver Configuration*.
- 3 Select the *Driver Module* tab.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 Click the upper right corner of the driver icon.
- 4 Click *Edit Properties > Driver Configuration > Driver Module*.



**Table B-1** Settings: Driver Module

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>Driver Object Password: Set 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. Otherwise, the remote driver does not run. The Remote Loader uses this password to authenticate itself to the remote driver shim.
 <i>Remote Loader Client Configuration for Documentation: Include in documentation</i>	 Includes information on the Remote Loader client configuration when Designer generates documentation on the driver.

## B.1.2 Authentication

The authentication section stores the information required to authenticate to the connected system.

In Designer:











- 1 Open a project in the Modeler.
- 2 Right-click the driver icon or driver line, then select *Properties > Driver Configuration*.
- 3 Click *Authentication*.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 Click the upper right corner of the driver icon.
- 4 Click *Edit Properties > Driver Configuration > Authentication*.

**Table B-2** Settings: Authentication

Option	Description
Authentication information for server	Displays or specifies the IP address or server name that the driver is associated with

Option	Description
<i>Authentication DN</i> or  <i>Authentication ID</i>	Specifies the DN of the LDAP account that the driver will use for authentication.  Example: Administrator
<i>Authentication Context</i> or  <i>Connection Information</i>	Specifies the IP address or name of the server the application shim should communicate with.
<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> <code>kmo=certificatename</code> , 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 <code>kmo</code> entry is optional. It is used only when an SSL connection exists between the Remote Loader and the Metadirectory engine.  Example: <code>hostname=10.0.0.1 port=8090</code> <code>kmo=IDMCertificate</code>
<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>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>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.1.3 Startup Option

The Startup Option allows you to set the driver state when the Identity Manager server is started.

In Designer:


- 1 Open a project in the Modeler.
- 2 Right-click the driver icon or driver line, then select *Properties > Driver Configuration*.
- 3 Click *Startup Option*.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*.
- 2 Click *Search* to search for the driver set that is associated with the driver.

- 3 Click the upper right corner of the driver icon.
- 4 Click *Edit Properties > Driver Configuration > Startup Option*.


**Table B-3** Settings: Startup Option

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 applies only if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it starts.

## B.1.4 Driver Parameters

In Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver icon or driver line, then select *Properties > Driver Configuration*.
- 3 Click *Driver Parameters*.

Parameter	Description
 <b>Driver parameters for server</b>	Displays or specifies the server name or IP address of the server whose driver parameters you want to modify.
<b>Edit XML</b>	Opens an editor so that you can edit the driver's configuration file.
<b>Driver Options</b>	
<i>Exchange LDAP Server</i>	Specifies the IP address of the Exchange LDAP server.
<i>Create NT Security Account</i> 0=Yes 1=No	Specifies whether to create an NT account when a new mailbox is created. Usually, this is Yes. If you do not want a security account associated with the mailbox account, specify No.
<i>Exchange Server</i>	Specifies the IP address or host name of the Exchange Server. The driver makes LDAP queries to this server.
<i>Exchange Site</i>	Specifies the site that the driver administers.
<i>Authoritative Bind</i>	Specifies whether to bind authoritatively or anonymously. The default is authoritative (Yes). See <a href="#">Section 5.7, "Using Authoritative Bind," on page 39</a> .
<b>Subscriber Options</b>	
<i>NT Domain Server</i>	Specifies the IP address or host name of the Exchange Server. The driver makes LDAP queries to this server.

Parameter	Description
<b>Publisher Options</b>	
<i>Polling Rate (in Seconds)</i>	Specifies how long the driver suspends processing between each Exchange connection.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 Click the upper right corner of the driver icon.
- 4 Click *Edit Properties > Driver Configuration > Driver Parameters*.

### B.1.5 ECMAScript

Enables you to add ECMAScript resource files. The resources extend the driver's functionality when Identity Manager starts the driver.

## B.2 Identity Manager: Global Configuration Values

Global configuration values (GCVs) enable 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 as with 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 Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver line, then select *Properties > Global Configuration Values*.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 Click the upper right corner of the driver icon.
- 4 Click *Edit Properties > Global Config Values*.

**Table B-4** Settings: Password Configuration

Option	Description
<i>Application accepts passwords from Identity Manager</i>	If <i>True</i> , allows passwords to flow from the Identity Manager data store to the connected system.
<i>Identity Manager accepts passwords from application</i>	If <i>True</i> , allows passwords to flow from the connected system to Identity Manager.
<i>Publish passwords to NDS password</i>	Use the password from the connected system to set the non-reversible NDS <sup>®</sup> password in eDirectory.
<i>Publish passwords to Distribution Password</i>	Use the password from the connected system to set the NMAST <sup>™</sup> Distribution Password used for Identity Manager password synchronization.
<i>Require password policy validation before publishing passwords</i>	If <i>True</i> , applies NMAST 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 by 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. The e-mail notification templates use this value.

## B.3 Identity Manager: Named Passwords

Identity Manager enables 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 configured Named Passwords, see [Section 7.7, “Storing Driver Passwords Securely with Named Passwords,” on page 52.](#)

In Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver line, then select *Properties > Named Passwords*.

In iManager:

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

## B.4 Identity Manager: 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 Designer:

- 1 In the Modeler, right-click a 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, the Engine Control Values display in the large pane.

In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 Click the upper right corner of the driver icon.
- 4 Click *Edit Properties > Engine Control Values*.

**Table B-5** Settings: 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 <i>True</i> 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 are presented to the Subscriber channel with type qualifiers. For example, CN=. A <i>True</i> 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.)

Option	Description
<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 backwards-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 backwards-compatibility with existing DirXML<sup>®</sup> 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> <hr/> <p><b>NOTE:</b> This control does not limit the number of application objects that can be migrated; it merely limits the batch size.</p>
<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<sup>™</sup> 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 <i>False</i> 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 <i>True</i> 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>

Option	Description
<i>Enable password synchronization status reporting</i>	This control determines whether the Identity Manager engine reports the status of Subscriber channel password change events.  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.

## B.5 Identity Manager: Log Level

Each driver set and each 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.) To track additional message types, change the log level.

Novell® recommends that you use Novell Audit instead of setting the log levels. See the *Identity Manager 3.5.1 Logging and Reporting* guide.


In Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver line, then select *Properties > Driver Log Level*.

In iManager:

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

**Table B-6** *Settings: Log Level*

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 Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver line, then select *Properties > iManager Icon*.

In iManager:

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

## 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*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 Click the upper right corner of the driver icon.
- 4 Click *Edit Properties > Security Equals*.

Designer does not list the users the driver is security equals to.

## B.8 Filter

Launches the Filter editor. You can edit the Filter from this tab.

In Designer:

- 1 In an open project, click the *Outline* tab (Outline view).
- 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.

In iManager:

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

## B.9 Edit Filter XML

Allows you to edit the filter directly in XML instead of using the Filter editor.

In Designer:

- 1 In an open project, click the *Outline* tab (Outline view).
- 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.

In iManager:

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

## B.10 Identity Manager: 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 affects only 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 Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver line, then select *Properties > Trace*.

In iManager:

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

**Table B-7** *Settings: Misc*

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.
<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 Objects

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*.
- 2 Click *Search* to search for the driver set that is associated with the driver.
- 3 Click the upper right corner of the driver icon.
- 4 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 Designer:

- 1 Open a project in the Modeler.
- 2 Right-click the driver line, then select *Properties > Driver Manifest*.

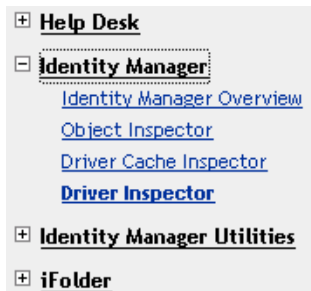
In iManager:

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

- 4 Click *Edit Properties > Driver Manifest*.

## B.13 Driver Cache Inspector

Figure B-3 The Link to the 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 allows 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 events stored in the cache file.

## B.14 Driver Inspector

The Driver Inspector page displays information about 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:** Enables you to re-read all of the objects associated with the driver and refresh the displayed information.
- ◆ **Actions:** Enables 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 the driver's associated objects that have a Disabled state.
  - ◆ **Filter for Manual Associations:** Displays all the driver's associated objects that have a Manual state.
  - ◆ **Filter for Migrate Associations:** Displays all the driver's associated objects that have a Migrate state.
  - ◆ **Filter for Pending Associations:** Displays all the driver's associated objects that have a Pending state.
  - ◆ **Filter for Processed Associations:** Displays all the driver's associated objects that have a Processed state.
  - ◆ **Filter for Undefined Associations:** Displays all the driver's associated objects 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.15 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 <i>&lt;password&gt;</i> 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 <i>&lt;password&gt;</i> 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>
<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>

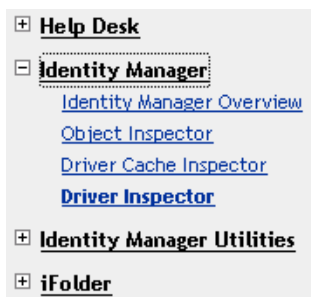
---

Option	Description
<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>
<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>
<i>Always accept password; ignore Password Policies</i>	<p><b>NOTE:</b> 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> <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>

Option	Description
<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> <p><b>NOTE:</b> To set up e-mail notification, select <i>Passwords &gt; Edit EMail Templates</i>.</p>

## B.16 Driver Inspector

Figure B-4 The Link to the Driver Inspector



The Driver Inspector displays information about the connected system without directly accessing the system. Designer does not have this option.