

# Novell Identity Manager Driver for NT Domain

3.5.1

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

---

IMPLEMENTATION GUIDE

September 28, 2007

# N

Novell®

## Legal Notices

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

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

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

Copyright © 2002-2007 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

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

Novell, Inc.  
404 Wyman Street, Suite 500  
Waltham, MA 02451  
U.S.A.  
[www.novell.com](http://www.novell.com)

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

## **Novell Trademarks**

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

## **Third-Party Materials**

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



# Contents

<b>About This Guide</b>	<b>9</b>
<b>1 Introducing the Identity Manager Driver for NT Domain</b>	<b>11</b>
1.1 New Features	11
1.1.1 Driver Features	11
1.1.2 Identity Manager Features	11
1.2 Data Flow	11
1.2.1 Policies	12
1.3 Key Driver Features	12
1.3.1 Local Platforms	13
1.3.2 Remote Platforms	13
1.3.3 Role-Based Entitlements	13
1.3.4 Password Synchronization Support	13
1.3.5 Synchronized Objects	13
<b>2 Installing the NT Domain Driver</b>	<b>15</b>
2.1 Where to Install the NT Domain Driver	15
2.1.1 Installation: Remote Loader on the PDC	15
2.1.2 Installation: All Components on the PDC	16
2.1.3 Installation: All Components on the BDC	16
2.1.4 Installation: Remote Installation on Windows and Other Platforms	17
2.2 Prerequisites	17
2.2.1 Information Needed For Installation	17
2.3 Installation	18
2.3.1 Installing the NT Domain Driver (Local Install)	18
2.3.2 Installing the NT Domain Driver (Remote Loader Installation)	18
2.3.3 Post-Installation Tasks	18
<b>3 Upgrading</b>	<b>25</b>
3.1 Upgrading the Driver Shim from DirXML 1.1a	25
3.2 Upgrading the Driver Shim from Identity Manager 2.x	26
3.3 Upgrading the Driver Configuration	26
3.3.1 Upgrading the Driver in Designer	27
3.3.2 Upgrading the Driver in iManager	29
3.4 Upgrading from Password Synchronization 1.0 to Identity Manager Password Synchronization	30
3.5 Upgrading to Support Identity Manager Password Synchronization	30
<b>4 Activating the Driver</b>	<b>31</b>
<b>5 Synchronizing Objects</b>	<b>33</b>
5.1 What Is Synchronization?	33
5.2 When Is Synchronization Done?	33
5.3 How Does the Metadirectory Engine Decide Which Object to Synchronize?	34
5.4 How Does Synchronization Work?	35

5.4.1	Scenario One . . . . .	35
5.4.2	Scenario Two . . . . .	37
5.4.3	Scenario Three . . . . .	38
<b>6</b>	<b>Managing the Driver</b>	<b>41</b>
6.1	Starting, Stopping, or Restarting the Driver . . . . .	41
6.1.1	Starting the Driver in Designer . . . . .	41
6.1.2	Starting the Driver in iManager . . . . .	41
6.1.3	Stopping the Driver in Designer . . . . .	41
6.1.4	Stopping the Driver in iManager . . . . .	41
6.1.5	Restarting the Driver in Designer . . . . .	42
6.1.6	Restarting the Driver in iManager . . . . .	42
6.2	Using the DirXML Command Line Utility . . . . .	42
6.3	Viewing Driver Versioning Information . . . . .	42
6.3.1	Viewing a Hierarchical Display of Versioning Information . . . . .	42
6.3.2	Viewing the Versioning Information as a Text File . . . . .	44
6.3.3	Saving Versioning Information . . . . .	46
6.4	Reassociating a Driver Set Object with a Server Object . . . . .	47
6.5	Changing the Driver Configuration . . . . .	48
6.6	Storing Driver Passwords Securely with Named Passwords . . . . .	48
6.6.1	Using Designer to Configure Named Passwords . . . . .	49
6.6.2	Using iManager to Configure Named Passwords . . . . .	49
6.6.3	Using Named Passwords in Driver Policies . . . . .	50
6.6.4	Using the DirXML Command Line Utility to Configure Named Passwords . . . . .	51
6.7	Adding a Driver Heartbeat . . . . .	55
<b>7</b>	<b>Backing Up the Driver</b>	<b>57</b>
7.1	Exporting the Driver in Designer . . . . .	57
7.2	Exporting the Driver in iManager . . . . .	57
<b>8</b>	<b>Customizing the NT Domain Driver</b>	<b>59</b>
8.1	Configuring Driver Parameters . . . . .	59
8.1.1	Log Level . . . . .	59
8.1.2	Polling Rate . . . . .	60
8.1.3	Password Expiration Time . . . . .	60
8.1.4	Security Options . . . . .	62
8.1.5	Startup Options . . . . .	62
8.1.6	Additional Options . . . . .	63
8.2	Configuring Data Synchronization . . . . .	63
8.2.1	Integrating the Identity Manager Driver for NT Domain and the Identity Manager Driver for Exchange . . . . .	63
8.2.2	Filtering Out Non-User Objects . . . . .	64
8.2.3	Synchronizing Group Information . . . . .	65
8.2.4	Changing the Location of User Objects by Using Placement Policies . . . . .	66
8.2.5	Changing Which Attributes Are Synchronized by Using Publisher and Subscriber Filters . . . . .	66
8.2.6	Querying GlobalGroup or LocalGroup . . . . .	69
<b>9</b>	<b>Password Synchronization</b>	<b>71</b>
9.1	Comparison of Password Synchronization 1.0 and Password Synchronization Provided with Identity Manager . . . . .	71

9.2	Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager. . . . .	73
9.2.1	Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies . . . . .	75
9.3	New Driver Configuration and Identity Manager Password Synchronization. . . . .	77
9.4	Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization . . . . .	77
9.5	Setting Up Password Synchronization Filters . . . . .	80
9.5.1	Separately Configuring Password Filters on Each Domain Controller . . . . .	81
9.5.2	Configuring Password Filters for All Domain Controllers from One Machine . . . . .	83
<b>10</b>	<b>Troubleshooting</b>	<b>87</b>
10.1	Error Messages . . . . .	87
10.2	Troubleshooting Password Synchronization . . . . .	95
10.3	Troubleshooting Driver Processes. . . . .	95
10.3.1	Viewing Driver Processes . . . . .	95
<b>11</b>	<b>Security: Best Practices</b>	<b>103</b>
<b>A</b>	<b>DirXML Command Line Utility</b>	<b>105</b>
A.1	Interactive Mode . . . . .	105
A.2	Command Line Mode . . . . .	114
<b>B</b>	<b>Properties of the Driver</b>	<b>119</b>
B.1	Driver Configuration . . . . .	119
B.1.1	Driver Module . . . . .	120
B.1.2	Driver Object Password. . . . .	120
B.1.3	Authentication . . . . .	121
B.1.4	Startup Option . . . . .	122
B.1.5	Driver Parameters . . . . .	123
B.2	Global Configuration Values . . . . .	124
B.3	Named Passwords. . . . .	125
B.4	Engine Control Values. . . . .	126
B.5	Log Level . . . . .	128
B.6	Driver Image . . . . .	129
B.7	Security Equals . . . . .	129
B.8	Filter. . . . .	130
B.9	Edit Filter XML. . . . .	130
B.10	Misc . . . . .	131
B.11	Excluded Users . . . . .	131
B.12	Driver Manifest. . . . .	132
B.13	Driver Inspector . . . . .	132
B.14	Driver Cache Inspector . . . . .	133
B.15	Inspector . . . . .	133
B.16	Server Variables . . . . .	134





# About This Guide

This guide explains how to install and configure the Identity Manager Driver for NT Domain.

The guide contains the following sections:

- ◆ [Chapter 1, “Introducing the Identity Manager Driver for NT Domain,” on page 11](#)  
This section introduces new features and explains the default driver configuration.
- ◆ [Chapter 2, “Installing the NT Domain Driver,” on page 15](#)  
This section covers the installation process as well as post-installation setup tasks.
- ◆ [Chapter 3, “Upgrading,” on page 25](#)  
This section covers the upgrade process, including important information about upgrading Password Synchronization 1.0 to Novell® Identity Manager Password Synchronization.
- ◆ [Chapter 8, “Customizing the NT Domain Driver,” on page 59](#)  
This section explains how to customize driver parameters and data synchronization. It provides examples for common customizations.
- ◆ [Chapter 9, “Password Synchronization,” on page 71](#)  
This section explains the differences between Password Synchronization 1.0 and Identity Manager Password Synchronization, and explains how to set up Identity Manager Password Synchronization. It also includes important information about upgrading Password Synchronization.
- ◆ [Chapter 10, “Troubleshooting,” on page 87](#)  
This section lists common error messages and possible causes.

## Audience

This guide is intended for Windows\* NT\* administrators and others who will implement the Identity Manager driver for NT Domains.

## Feedback

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

## Documentation Updates

For the most recent version of the *Identity Manager Driver for NT Domain*, see the [Drivers Documentation Web Site \(http://www.novell.com/documentation/idm35drivers/index.html\)](http://www.novell.com/documentation/idm35drivers/index.html).

## Additional Documentation

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

## Documentation Conventions

In 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.

# Introducing the Identity Manager Driver for NT Domain

# 1

The Identity Manager Driver for NT Domain is designed to manage and synchronize Windows NT 4 Domains with the Identity Vault. The Identity Manager Driver for NT Domain runs on the Windows NT 4 server.

The driver does the following:

- ◆ Synchronizes User objects between the Identity Vault and NT 4 Domains.
- ◆ Does a simple mapping between similar attributes.
- ◆ Can be used to migrate User objects between the Identity Vault and NT 4.

The driver does not serve as a general-purpose NT 4 Domain administration tool.

- ◆ [Section 1.1, “New Features,” on page 11](#)
- ◆ [Section 1.2, “Data Flow,” on page 11](#)

## 1.1 New Features

In this section:

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

### 1.1.1 Driver Features

The filter for Password Sync now supports 64-bit processing.

### 1.1.2 Identity Manager Features

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

## 1.2 Data Flow

This sections explains how the data flows between the NT Domain and the Identity Vault.

- ◆ The Publisher reads events from an NT Domain PDC’s registry and submits that information to the Identity Vault via the Metadirectory engine.
- ◆ The Subscriber watches for additions and modifications to the Identity Vault objects and makes changes to NT Domain that reflect those changes.

## 1.2.1 Policies

Policies are used to control data synchronization between NT Domain and the Identity Vault. The NT Domain sample driver configuration provides a set of policies, some of which are described in the table below. These policies can be customized through Novell iManager, as explained in [Chapter 8, “Customizing the NT Domain Driver,” on page 59](#).

Policy	Description
Schema Mapping	<p>Configured on the driver object.</p> <p>Maps the following eDirectory User class and properties to NT Domain Username class and attributes:</p> <ul style="list-style-type: none"><li>CN, name</li><li>Description, Comment</li><li>Full Name, FullName</li><li>Login Disabled, Disable</li><li>Password Allow Change, PasswordChange</li><li>Password Required, PasswordRequired</li><li>Login Allowed Time Map, LogonHours</li><li>Login Expiration Time, AcctExpires</li></ul>
Create	<p>Configured on the Publisher channel.</p> <p>Requires that the Surname attribute must be specified in order for a User object to be created.</p> <p>NT does not use this attribute, but eDirectory™ requires it. To satisfy the eDirectory requirement, the Create policy sets a default Surname for all users, <code>Unknown</code>, or you can specify your own when importing the driver configuration.</p>
Matching	<p>Configured on the Publisher and Subscriber channels.</p> <p>Specifies that a user in the Identity Vault is the same user as a user in NT when the value of CN is the same in both places.</p> <hr/> <p><b>NOTE:</b> Because the NT Domain APIs allow queries for only the user name attribute, this policy should not be changed.</p> <hr/>
Placement	<p>Configured on the Publisher and Subscriber channels.</p> <p>Specifies that new users are named by the value of the leafmost part of the source distinguished name and are placed in the containers you defined during driver setup. You should create these containers before you start the driver.</p>

## 1.3 Key Driver Features

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

- ◆ [Section 1.3.1, “Local Platforms,” on page 13](#)
- ◆ [Section 1.3.2, “Remote Platforms,” on page 13](#)
- ◆ [Section 1.3.3, “Role-Based Entitlements,” on page 13](#)
- ◆ [Section 1.3.4, “Password Synchronization Support,” on page 13](#)

- ◆ [Section 1.3.5, “Synchronized Objects,” on page 13](#)

### 1.3.1 Local Platforms

The NT driver is supported on Windows NT 4 with Support Pack 6a or later. For more information, see [Section 2.3.1, “Installing the NT Domain Driver \(Local Install\),” on page 18](#). To install the driver on any other platform, you must use the Remote Loader.

### 1.3.2 Remote Platforms

The NT driver shim can remotely access the NT domain by being installed on a Backup Domain Controller or any server that is a member of the domain. For this to happen, the Authoritative User needs administrative rights to the domain.

The NT driver can run remotely with the Remote Loader. The Remote Loader service is supported on Windows NT 4 with support pack 6a or later. For more information, see [Section 2.3.2, “Installing the NT Domain Driver \(Remote Loader Installation\),” on page 18](#). The Metadirectory server can run on any supported platforms for Identity Manager. For more information, see [Section 2.1, “Where to Install the NT Domain Driver,” on page 15](#).

### 1.3.3 Role-Based Entitlements

The sample driver configuration supports Role-Based Entitlements. If Role-Based Entitlements are enabled, the driver does the following actions by default:

- ◆ Add User object accounts
- ◆ Remove User object accounts
- ◆ Add Group memberships
- ◆ Remove Group memberships

### 1.3.4 Password Synchronization Support

The NT driver full supports Password Synchronization, except for the Check Password options on the Subscriber channel.

### 1.3.5 Synchronized Objects

The default configuration of the NT driver synchronizes users objects. It can synchronize group information, with modification. The driver queries for two classes: GlobalGroup and LocalGroup. Although you can't synchronize them on the Subscriber or Publisher channel, you can use the querying feature to synchronize them in an indirect way, so that the driver can use the MemberOf attribute on a user to put the user in a corresponding group in eDirectory. See [Section 8.2.3, “Synchronizing Group Information,” on page 65](#) for more information.



# Installing the NT Domain Driver

# 2

The Identity Manager Driver for NT Domain can be installed along with other Identity Manager drivers at the same time that the Metadirectory engine is installed. This method of installation is documented in “[Installing Identity Manager](#)” in the *Identity Manager 3.5.1 Installation Guide*.

The driver can also be installed separately after the Metadirectory engine is installed, by running the Identity Manager installation and selecting only the NT Domain driver.

This section covers the following installation topics:

- ♦ [Section 2.1, “Where to Install the NT Domain Driver,” on page 15](#)
- ♦ [Section 2.2, “Prerequisites,” on page 17](#)
- ♦ [Section 2.3, “Installation,” on page 18](#)

## 2.1 Where to Install the NT Domain Driver

The NT Domain driver provides synchronization for a single domain. Multiple domains require multiple Identity Manager driver installations. Consider initially setting up synchronization for a single domain and then using Identity Manager’s driver export and import functionality to expedite synchronization setup for additional domains. See the *Novell Identity Manager 3.5.1 Administration Guide* for information about driver export and import. The NT Domain driver can be installed in any of the following configurations:

- ♦ [Section 2.1.1, “Installation: Remote Loader on the PDC,” on page 15](#)
- ♦ [Section 2.1.2, “Installation: All Components on the PDC,” on page 16](#)
- ♦ [Section 2.1.3, “Installation: All Components on the BDC,” on page 16](#)
- ♦ [Section 2.1.4, “Installation: Remote Installation on Windows and Other Platforms,” on page 17](#)

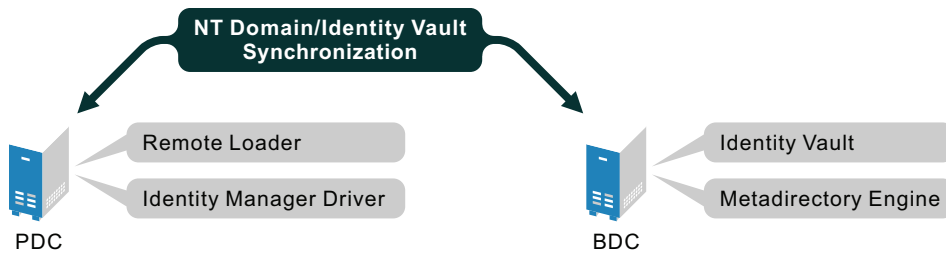
### 2.1.1 Installation: Remote Loader on the PDC

As shown in [Figure 2-1, “Installation Configuration: Remote Loader,” on page 16](#), you can install the Identity Vault and the Metadirectory engine on a Backup Domain Controller (BDC) or Member server. Then, install the NT Domain driver and the Remote Loader service on the Primary Domain Controller (PDC).

This configuration allows you to insulate the PDC, with the exception of the installation of two components that don’t require much disk space or many processing cycles.

It also allows the Identity Manager driver direct access to the PDC. From this position, the driver can manage any recovery scenarios independent of connection and API constraints.

**Figure 2-1** Installation Configuration: Remote Loader



## 2.1.2 Installation: All Components on the PDC

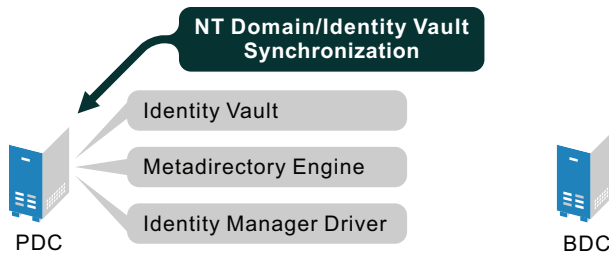
As shown in [Figure 2-2, “Installation Configuration: All Components on the PDC,”](#) on page 16, you can install the Identity Vault, the Metadirectory engine, and the NT Domain driver on the PDC.

This configuration is optimal for processing speed because all components are installed on the same computer. Additionally, it allows the Identity Manager driver direct access to the PDC. From this position, the driver can manage any recovery scenarios independent of connection and API constraints.

However, the PDC is often restricted territory. Placing the Identity Vault on the PDC might be prohibited by your corporate policy.

To set up all components on the PDC, see [“Installing the NT Domain Driver \(Local Install\)”](#) on page 18.

**Figure 2-2** Installation Configuration: All Components on the PDC



## 2.1.3 Installation: All Components on the BDC

As shown in [Figure 2-3, “Installation Configuration: All Components on the BDC,”](#) on page 17, you can install the Identity Vault, the Metadirectory engine, and the NT Domain driver on the BDC.

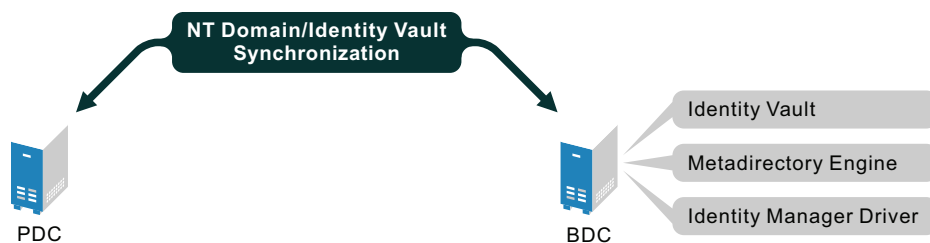
This configuration insulates the PDC completely.

However, because the driver must communicate with the PDC, this configuration can create difficulties if the driver encounters any connection or other communication problems. For this reason, the previous configurations are recommended before this configuration.

To set up all components on the BDC, see [“Installing the NT Domain Driver \(Local Install\)”](#) on page 18.



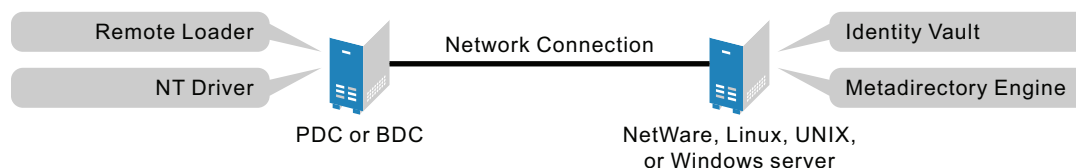
Figure 2-3 Installation Configuration: All Components on the BDC



## 2.1.4 Installation: Remote Installation on Windows and Other Platforms

As shown in [Figure 2-4](#), “Installation Configuration: Remote Loader and NT Driver on the PDC or BDC,” on [page 17](#), you can install the Remote Loader and the NT driver on the PDC or the BDC, but install the Identity Vault and the Metadirectory engine on a separate server.

Figure 2-4 Installation Configuration: Remote Loader and NT Driver on the PDC or BDC



This configuration is attractive if your Identity Vault and Metadirectory engine (Identity Manager) installations are on a platform other than one of the supported versions of Windows.

Both Scenario 1 and Scenario 4 configurations eliminate the performance impact of hosting the Identity Vault and the Metadirectory engine on the PDC.

## 2.2 Prerequisites

- Novell Identity Manager 3.5.1 and its prerequisites, as listed in “[System Requirements for Identity Manager](#)” in the *Identity Manager 3.5.1 Installation Guide*.
- Windows NT 4 with Service Pack 6a or later.
- Collect required information, as explained in “[Information Needed For Installation](#)” on [page 17](#).
- Before importing the driver configuration, create the containers that you need to specify during import. The import prompts are described in “[Driver Configuration Parameters](#)” on [page 21](#).

### 2.2.1 Information Needed For Installation

Collect the following information before installing the driver shim and importing the driver configuration:

- The name of the NT 4 PDC that the driver will be synchronizing with.
- The name of the domain you want to synchronize with.
- The eDirectory™ context where you want to synchronize the User objects.

- ❑ The name and password for an NT domain user with the rights to manipulate User objects in the domain.

When you create or import the sample driver configuration, a wizard prompts you for the information listed in [“Driver Configuration Parameters” on page 21](#).

## 2.3 Installation

In this section:

- ◆ [Section 2.3.1, “Installing the NT Domain Driver \(Local Install\),” on page 18](#)
- ◆ [Section 2.3.2, “Installing the NT Domain Driver \(Remote Loader Installation\),” on page 18](#)
- ◆ [Section 2.3.3, “Post-Installation Tasks,” on page 18](#)

### 2.3.1 Installing the NT Domain Driver (Local Install)

In a local configuration, the driver is installed on the same computer that is hosting the Metadirectory engine.

Install the components on the appropriate machine, as described in [Section 2.1, “Where to Install the NT Domain Driver,” on page 15](#).

For instructions, see [“Installing Identity Manager”](#) in the *Identity Manager 3.5.1 Installation Guide*.

After installation, you must set up the driver as explained in [“Post-Installation Tasks” on page 18](#).

### 2.3.2 Installing the NT Domain Driver (Remote Loader Installation)

In a remote configuration, the driver and the Remote Loader service are installed on a computer other than the one hosting the Metadirectory engine.

Install the components on the appropriate machines as described in [Section 2.1, “Where to Install the NT Domain Driver,” on page 15](#).

For instructions on installing the driver and Remote Loader, see [“Installing the Connected System Option on Windows”](#) in the *Identity Manager 3.5.1 Installation Guide* and [“Deciding Whether to Use the Remote Loader”](#) in the *Novell Identity Manager 3.5.1 Administration Guide*.

After installation, you must set up the driver as explained in [“Post-Installation Tasks” on page 18](#).

### 2.3.3 Post-Installation Tasks

Post-installation setup is not required if you are upgrading an existing driver.

If this is the first time the NT Domain driver has been used, you should complete the post-installation tasks in the following sections:

- ◆ [“Creating an Admin User” on page 19](#)
- ◆ [“Granting Rights to the Driver” on page 19](#)
- ◆ [“Importing the Driver Configuration in Designer” on page 19](#)

- ◆ “Importing the Driver Configuration File in iManager” on page 20
- ◆ “Driver Configuration Parameters” on page 21
- ◆ “Starting the Driver” on page 22
- ◆ “Migrating and Resynchronizing Data” on page 23
- ◆ “Activating the Driver” on page 23

## Creating an Admin User

The driver needs Read/Write rights to the domain. When you set up the driver, you are prompted to provide an NT account that the driver can use to access the domain. You can configure the driver to use any existing account with the appropriate rights, or to ease future management, you can create a new account to be used exclusively by the driver.

## Granting Rights to the Driver

After you complete the Identity Manager installation, you need to grant rights to the driver so that it can access the SAM keys in the registry of the server that has the domain you want to use.

Creating an Administrator equivalent gives the driver rights to read and write to the domain, but, by default, even the Administrator cannot access the registry until you explicitly assign that access.

To grant the rights:

- 1 Log in to NT as Administrator.
- 2 Run `regedt32`.
- 3 Select the `HKEY_LOCAL_MACHINE` window.
- 4 Select the *SAM key*, then on the *Security* menu, select *Permissions*.
- 5 Select the *Replace Permission on Existing Subkeys* check box.
- 6 Give Full Control permission to the administrator user you created for the driver, then click *OK*.
- 7 Click *Yes* to replace the permission on all existing subkeys within SAM.
- 8 Close the registry.

## Importing the Driver Configuration in Designer

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

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

- 1 Open a project in Designer. In the Modeler, right-click the Driver Set object and select *New > Driver*.
- 2 From the drop-down list, select *NT Domains*, then click *Run*.
- 3 Configure the driver by filling in the fields. Specify information specific to your environment. For information on the settings, see [Table 2-1 on page 21](#) for more information.
- 4 After specifying parameters, click *Finish* to import the driver.

- 5 After the driver is imported, customize and test the driver.
- 6 After the driver is fully tested, deploy the driver into the Identity Vault. See “[Deploying a Driver to an Identity Vault](#)” in the *Designer 2.1 for Identity Manager 3.5.1*.

### Importing the Driver Configuration File in iManager

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

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

Where do you want to place the new driver?

In an existing driver set

In a new driver set

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

- 3 Select how you want the driver configurations sorted:
  - ♦ All configurations
  - ♦ Identity Manager 3.5 configurations
  - ♦ Identity Manager 3.0 configurations
  - ♦ Configurations not associated with an IDM version
- 4 Select the *NT Domains* driver, then click *Next*.



- 5 Configure the driver by filling in the configuration parameters, then click *Next*. For information on the settings, see [Table 2-1 on page 21](#).
- 6 Define security equivalences using a user object that has the rights that the driver needs to have on the server
 

The tendency is to use the Admin user object for this task. However, you might want to create a DriversUser (for example) and assign security equivalence to that user. Whatever rights that the driver needs to have on the server, the DriversUser object must have the same security rights.
- 7 Identify all objects that represent administrative roles and exclude them from replication.
 

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

## Driver Configuration Parameters

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

**NOTE:** The parameters are presented on multiple screens and some parameters are only displayed if the answer to a previous prompt requires more information to properly configure the policy.

**Table 2-1** Configuration Fields for the NT Domain Driver

Import Prompt	Description
<i>Driver name</i>	The name of the driver contained in the driver configuration file is NT Domains. Specify the actual name you want to use for the driver.
<i>Domain Server</i>	The name of the server that contains the NT Domain that you want the driver to use, such as DOMAIN_SERVER. Use uppercase characters.
<i>Domain Name</i>	The name of the NT Domain that you want the driver to use, such as DOMAIN_NAME. Use uppercase characters.
<i>Authoritative User</i>	The NT Domain User the driver will use for domain authentication, such as Administrator.
<i>Authoritative Password</i>	The password for the User previously specified.  <b>IMPORTANT:</b> If you change the password in NT, you must also update the password in the driver configuration.
<i>Container</i>	The eDirectory container where the driver matches on objects to synchronize with NT, for example, Users.MyOrganization.
<i>Default Surname</i>	NT Domain Users do not have a Surname attribute. Enter a default Surname for use in the default Publisher Create policy. This can also be used as the default password (see the Publisher Command Transform, where the sample driver configuration enters the default surname).
<i>Polling Interval (milliseconds)</i>	Specify the number of milliseconds to delay before querying NT for changes.
<i>Password Sync Timeout (minutes)</i>	Specify the number of minutes for the driver to attempt to synchronize a given password. The driver does not try to synchronize the password after this interval has been exceeded. This interval should be at least twice as long as the polling interval.  See <a href="#">Section 8.1.3, "Password Expiration Time," on page 60</a> .
<i>Configure Data Flow</i>	Data flow can be configured at this time for the driver. Select the data flow that you desire.  <i>Bi-Directional</i> means that both NT and eDirectory are authoritative sources of the data synchronized between them.  <i>NT to eDirectory</i> means that NT is the authoritative source.  <i>eDirectory to NT</i> means that eDirectory is the authoritative source.

Import Prompt	Description
<i>Password Failure Notification User</i>	Password synchronization policies can send an e-mail concerning the failure of a password synchronization or password set for the associated user. This fails if that user does not have an e-mail address specified. To avoid such a failure, you can specify a default user (by DN) to which all notifications are sent.
<i>Enable Entitlements</i>	Select Yes if you are also using the Entitlements Service driver and want this driver to use Role-Based Entitlements. Otherwise, select No.  Using Role-Based Entitlements is a design decision. Select this option after you have reviewed <a href="#">"Creating and Using Entitlements"</a> in the <i>Novell Identity Manager 3.5.1 Administration Guide</i> .  The next two prompts are related to the use of Role-Based Entitlements and are displayed only if you select Yes.
<i>Action - Add Account</i>	Used only with Role-Based Entitlements.  Select what action is taken when a User account is added by Entitlements.  <i>Enable Account</i> or <i>Disable Account</i> .
<i>Action - Remove Account Entitlement</i>	Used only with Role-Based Entitlements.  Choose what action is taken when a User account is removed by Entitlements.  <i>Disable Account</i> or <i>Delete Account</i>
<i>Driver is Local/Remote</i>	Configure the driver for use with the Remote Loader service by selecting <i>Remote</i> , or select <i>Local</i> to configure the driver for local use. If <i>Local</i> is selected, the remaining prompts are not displayed.
<i>Remote Host Name and Port</i>	For remote driver configuration only.  Specify the hostname or IP address and port number where the Remote Loader Service has been installed and is running for this driver. The default port is 8090.
<i>Driver Password</i>	For remote driver configuration only.  The driver object password is used by the Remote Loader to authenticate itself to the Identity Manager server. It must be the same password that is specified in the Driver Object Password field on the Identity Manager Remote Loader.
<i>Remote Password</i>	For remote driver configuration only.  The Remote Loader password is used to control access to the Remote Loader instance. It must be the same password that is specified as the Remote Loader password on the Identity Manager Remote Loader.

## Starting the Driver

Follow the steps in the [Section 6.1, "Starting, Stopping, or Restarting the Driver,"](#) on page 41.

When the driver starts, you can open DSTrace to see the driver work its way through the registry and list every user in the domain. However, because activation is used in this release of Identity Manager, you might notice a short delay of 30 seconds or more at startup while the driver completes an activation query.

Synchronization takes place on an object-by-object basis as changes are made to individual objects. If you want to have an immediate synchronization, you must initiate that process as explained in the next section, [“Migrating and Resynchronizing Data” on page 23](#).

## Migrating and Resynchronizing Data

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

- ◆ **Migrate data from the 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 Metadirectory engine applies all of the Matching, Placement, and Create policies, as well as the Subscriber filter, to the object.
- ◆ **Migrate data into the Identity Vault:** Allows you to define the criteria Identity Manager uses to migrate objects from an application into the Identity Vault. When you migrate an object, the Metadirectory engine applies all of the Matching, Placement, and Create policies, as well as the Publisher filter, to the object. Objects are migrated into the Identity Vault using the order you specify in the Class list.
- ◆ **Synchronize:** The Metadirectory 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, follow the steps in [Section 6.1, “Starting, Stopping, or Restarting the Driver,” on page 41](#).

For a more detailed explanation of the data synchronization, see [Chapter 5, “Synchronizing Objects,” on page 33](#).

Keep the following points in mind when forcing data synchronization:

- ◆ When migrating into the Identity Vault, you can migrate either all Users or a specific User, but not a subset of Users. This constraint is imposed by the limited search capabilities of NT domains. Wildcards do not work for queries on the Publisher channel.
- ◆ When migrating a single user into the Identity Vault, specify the eDirectory user attribute mapped to the NT user name attribute (by default this is CN). Queries on other attributes are not supported by NT.
- ◆ If you have User accounts in both the Identity Vault and the domain and you want both systems to update data, synchronize data both ways.
- ◆ If the driver shuts down with an error, the driver performs a synchronization the next time it is started. In the synchronization, the driver issues a Modify command at startup for each User object found in the domain.

The Metadirectory engine accepts the Modify command if the User has an association. If the User does not have an association, the engine queries the driver for all of the attributes in the Publisher filter. The engine then creates the User.

## Activating the Driver

Activation must be completed within 90 days of installation, or the driver will not run.

For activation information, refer to [Chapter 4, “Activating the Driver,” on page 31](#).





In this section:

- ◆ [Section 3.1, “Upgrading the Driver Shim from DirXML 1.1a,” on page 25](#)
- ◆ [Section 3.2, “Upgrading the Driver Shim from Identity Manager 2.x,” on page 26](#)
- ◆ [Section 3.3, “Upgrading the Driver Configuration,” on page 26](#)
- ◆ [Section 3.4, “Upgrading from Password Synchronization 1.0 to Identity Manager Password Synchronization,” on page 30](#)
- ◆ [Section 3.5, “Upgrading to Support Identity Manager Password Synchronization,” on page 30](#)

## 3.1 Upgrading the Driver Shim from DirXML 1.1a

The driver shim replaces the previous driver shim but keeps the previous driver’s configuration. The new driver shim can run the DirXML<sup>®</sup> 1.1a configuration with no changes (unless you are using Password Synchronization 1.0).

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

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

- 2 Back up the driver. See [Chapter 7, “Backing Up the Driver,” on page 57](#) for instruction on how to back up the driver.
- 3 Install the Identity Manager 3.5.1 driver shim. You can do this at the same time that you install the Identity Manager 3.5.1 engine.

Follow the instructions in [“Installing Identity Manager”](#) in the *Identity Manager 3.5.1 Installation Guide*.

---

**WARNING:** If you have been using Password Synchronization 1.0, don’t install the upgraded Identity Manager Driver for NT Domain until you have read [Section 3.4, “Upgrading from Password Synchronization 1.0 to Identity Manager Password Synchronization,” on page 30](#) and are ready to add policies to your driver configuration to provide backward compatibility with Password Synchronization 1.0.

---

Running an Identity Manager driver shim or configuration with the DirXML 1.1a engine is not supported.

- 4 After the shim is installed, Novell<sup>®</sup> eDirectory<sup>™</sup> and the driver need to be restarted. Follow the instructions in [Section 6.1, “Starting, Stopping, or Restarting the Driver,” on page 41](#).
- 5 Activate the driver shim with your Identity Manager activation credentials.

See [“Activating the Driver” on page 31](#).

After you install the driver shim, continue with [Section 3.3, “Upgrading the Driver Configuration,” on page 26](#).

## 3.2 Upgrading the Driver Shim from Identity Manager 2.x

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

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

- 2 Back up the driver. See [Chapter 7, “Backing Up the Driver,” on page 57](#) for instruction on how to back up the driver.
- 3 Install the Identity Manager 3.5.1 driver shim. You can do this at the same time that you install the Identity Manager 3.5.1 engine.

Follow the instructions in [“Installing Identity Manager”](#) in the *Identity Manager 3.5.1 Installation Guide*.

---

**WARNING:** If you have been using Password Synchronization 1.0, don’t install the upgraded Identity Manager Driver for NT Domain until you have read [Section 3.4, “Upgrading from Password Synchronization 1.0 to Identity Manager Password Synchronization,” on page 30](#) and are ready to add policies to your driver configuration to provide backward compatibility with Password Synchronization 1.0.

---

Running an Identity Manager driver shim or configuration with the DirXML 1.x engine is not supported.

- 4 After the shim is installed, Novell® eDirectory™ and the driver need to be restarted. Follow the instructions in [Section 6.1, “Starting, Stopping, or Restarting the Driver,” on page 41](#).
- 5 Activate the driver shim with your Identity Manager activation credentials.

See [“Activating the Driver” on page 31](#).

After you install the driver shim, continue with [Section 3.3, “Upgrading the Driver Configuration,” on page 26](#).

## 3.3 Upgrading the Driver Configuration

A DirXML 1.1a driver configuration can be run with an Identity Manager driver shim and the Metadirectory engine, with no changes to the driver configuration (unless you are using Password Synchronization 1.0; see [Section 3.4, “Upgrading from Password Synchronization 1.0 to Identity Manager Password Synchronization,” on page 30](#)).

However, to edit a DirXML 1.1a driver configuration, you must either use the DirXML 1.1a iManager plug-ins or ConsoleOne®, or run the wizard that converts DirXML 1.1a configurations to Identity Manager format so you can edit the configuration using the Identity Manager iManager plug-ins. See [“Managing DirXML 1.1a Drivers in an Identity Manager Environment”](#) and [“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*.

---

**NOTE:** Running an Identity Manager driver configuration with a DirXML 1.1a driver shim is not supported.

---

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

*Understanding Policies for Identity Manager 3.5.1*. You can upgrade the driver in Designer or iManager. If you are upgrading from Identity Manager 3.5.0 to Identity Manager 3.5.1, the following information does not apply.

- ♦ [Section 3.3.1, “Upgrading the Driver in Designer,” on page 27](#)
- ♦ [Section 3.3.2, “Upgrading the Driver in iManager,” on page 29](#)

### 3.3.1 Upgrading the Driver in Designer

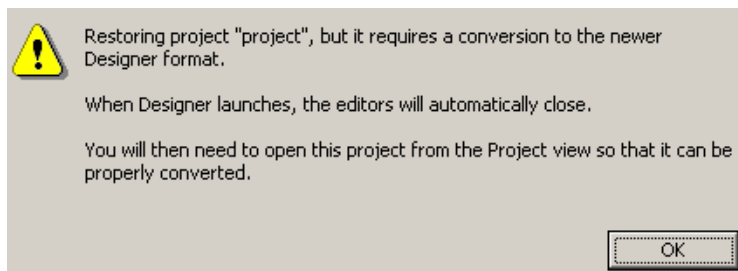
- 1 Make sure you have updated your driver with all the patches for the version you are currently running.

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

- 2 Back up the driver. See [Chapter 7, “Backing Up the Driver,” on page 57](#) for instruction on how to back up the driver.
- 3 Install Designer version 2.0 or above, then launch Designer.

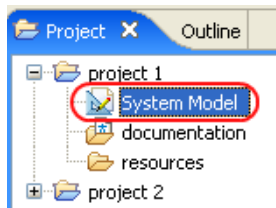
If you had a project open in Designer when you upgraded Designer, proceed to [Step 4](#). If you didn't have a project open in Designer when you upgraded Designer, skip to [Step 5](#).

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

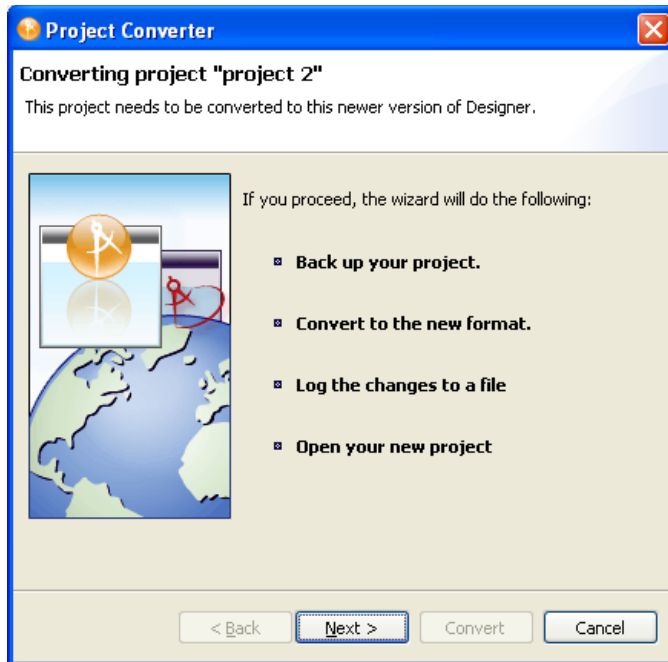


Designer closes the project to preform the upgrade.

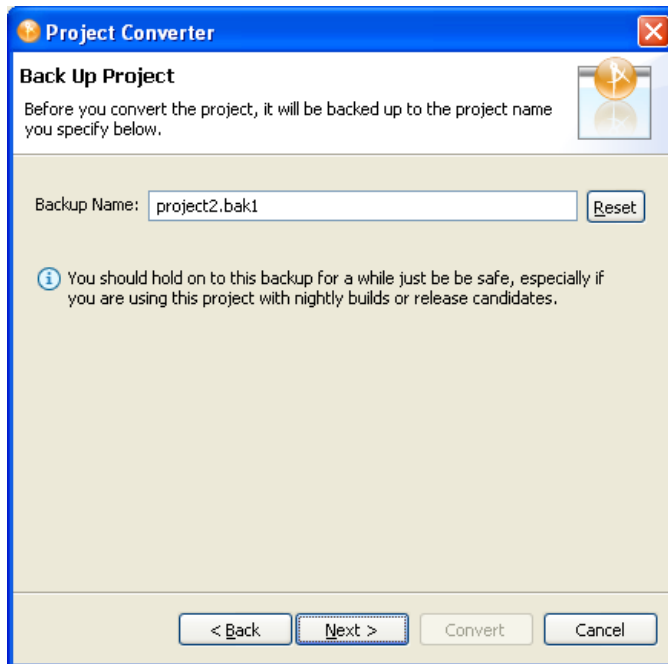
- 5 In the Project view, double-click *System Model* to open and convert the project.



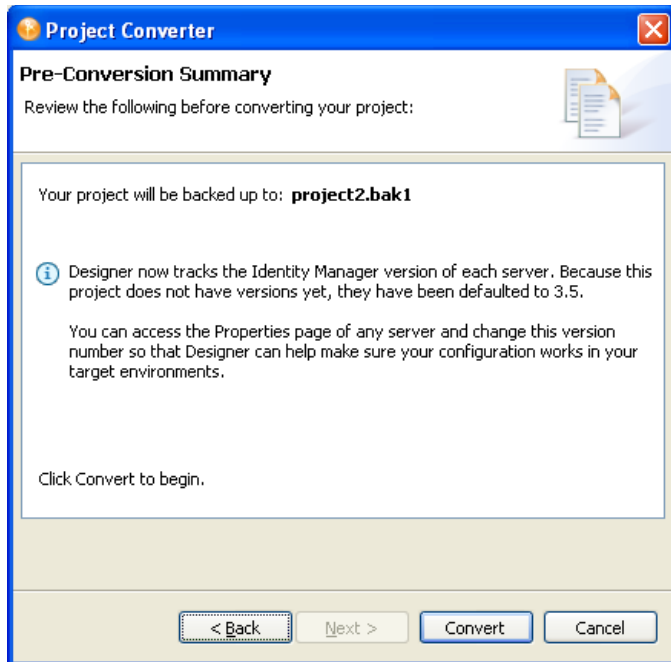
- 6 Read the Project Converter message explaining that the project is backed up, converted to the new format, changes logged to a file, and the new project is opened, then click *Next*.



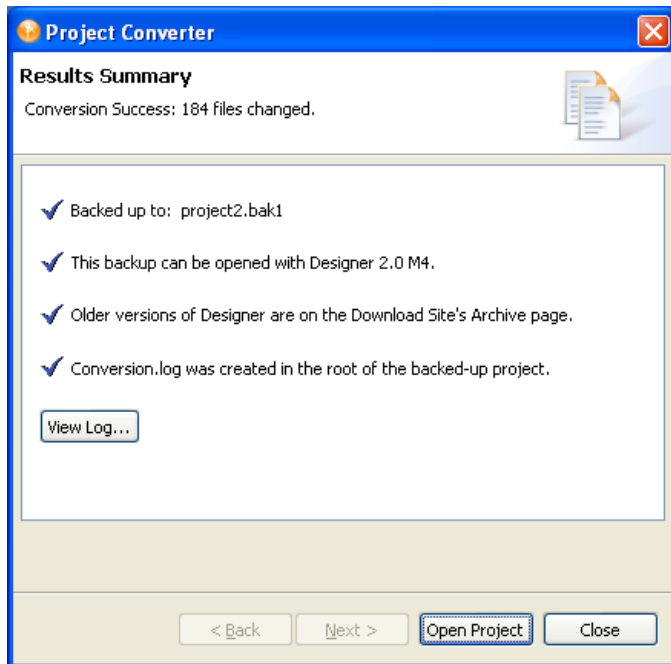
7 Specify the name of the backup project name, then click *Next*.



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



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



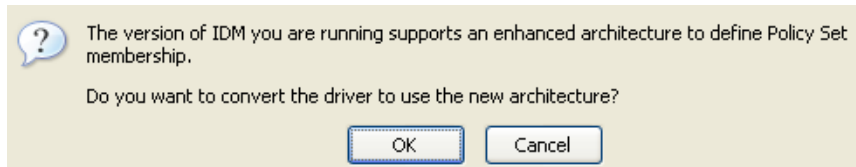
If you want to view the log file that is generated, click *View Log*.

### 3.3.2 Upgrading the Driver in iManager

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

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

- 2 Back up the driver. See [Chapter 7, “Backing Up the Driver,”](#) on page 57 for instruction on how to back up the driver.
- 3 Verify that Identity Manager 3.5.1 has been installed and you have the current plug-ins installed, then launch iManager.
- 4 Click *Identity Manager > Identity Manager Overview*.
- 5 Click *Search* to find the Driver Set object, then click the driver you want to upgrade.
- 6 Read the message that is displayed, then click *OK*.



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

## 3.4 Upgrading from Password Synchronization 1.0 to Identity Manager Password Synchronization

If you have been using Password Synchronization 1.0 with the Identity Manager Driver for NT, keep in mind the following items:

- ♦ Don't install the Identity Manager version of the driver shim until you are ready to add backward compatibility to your driver.
- ♦ Identity Manager Password Synchronization does not require the Novell Client™ to be installed on the Windows machine.

For instructions on adding backward compatibility to your driver, see [Section 9.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on page 73 in this guide.

## 3.5 Upgrading to Support Identity Manager Password Synchronization

This task is for driver objects that have not been used with Password Synchronization 1.0. It is for drivers that have existing configurations that you want to save, but you want to add support for Identity Manager Password Synchronization. See the instructions in [Section 9.4, “Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization,”](#) on page 77.

# Activating the Driver

# 4

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

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





# Synchronizing Objects

# 5

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 5.1, “What Is Synchronization?” on page 33](#)
- ♦ [Section 5.2, “When Is Synchronization Done?” on page 33](#)
- ♦ [Section 5.3, “How Does the Metadirectory Engine Decide Which Object to Synchronize?” on page 34](#)
- ♦ [Section 5.4, “How Does Synchronization Work?” on page 35](#)

## 5.1 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.

## 5.2 When Is Synchronization Done?

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

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

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

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

- ♦ The user issues a manual driver synchronization request. This corresponds to the *Resync* button in the Driver Set property page in ConsoleOne®, or to the *Synchronize* button on the iManager Identity Manager Driver Overview page.
- ♦ The Metadirectory engine encounters an error with the driver's cache and cannot recover from the cache error. The driver's cache is deleted and the engine generates object synchronization commands as detailed in [Section 5.3, "How Does the Metadirectory Engine Decide Which Object to Synchronize?"](#) on page 34.

## 5.3 How Does the Metadirectory Engine Decide Which Object to Synchronize?

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

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

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

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

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

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

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

## 5.4 How Does Synchronization Work?

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

1. Each system (the Identity Vault and the connected system) is queried for all attribute values in the appropriate filters.
  - ♦ eDirectory is queried for all values in the Subscriber filter, and for values that are marked for synchronization in Identity Manager 2.x and Identity Manager 3.x.
  - ♦ The connected system is queried for all values in the Publisher filter, and for values that are marked for synchronization in Identity Manager 2.x and Identity Manager 3.x.
2. The returned attribute values are compared and modification lists are prepared for the Identity Vault and the connected system according to [Table 5-1 on page 36](#), [Table 5-2 on page 37](#), and [Table 5-3 on page 39](#).

In the tables the following pseudo-equations are used:

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

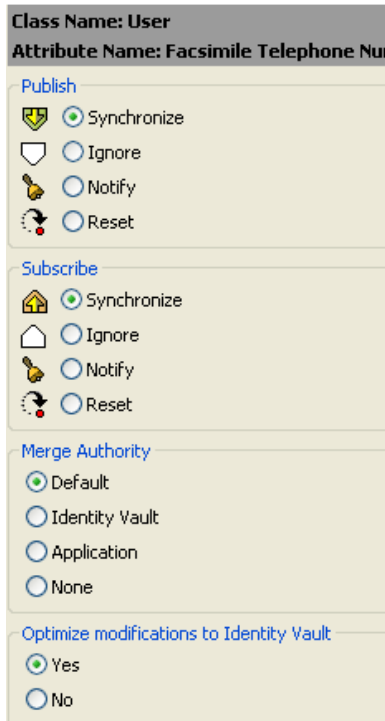
There are three different combinations of selected items in the filter, and each one creates a different output.

- ♦ [Section 5.4.1, “Scenario One,” on page 35](#)
- ♦ [Section 5.4.2, “Scenario Two,” on page 37](#)
- ♦ [Section 5.4.3, “Scenario Three,” on page 38](#)

### 5.4.1 Scenario One

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

**Figure 5-1** Scenario One



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

**Table 5-1** 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
<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

## 5.4.2 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 5-2 Scenario Two

**Class Name: User**

**Attribute Name: Description**

**Publish**

Synchronize

Ignore

Notify

Reset

**Subscribe**

Synchronize

Ignore

Notify

Reset

**Merge Authority**

Default

Identity Vault

Application

None

**Optimize modifications to Identity Vault**

Yes

No

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

Table 5-2 Output of Scenario Two

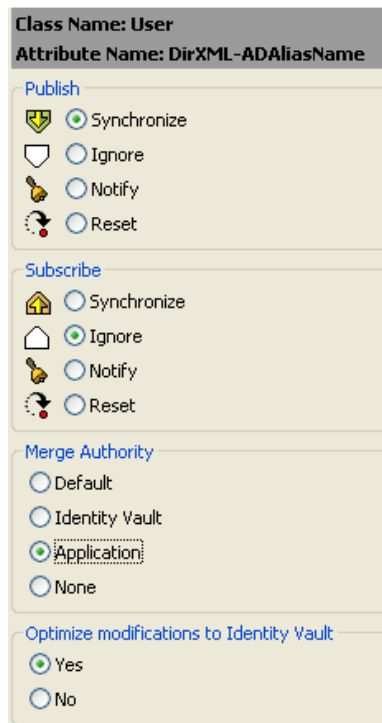
	Identity Vault single-valued empty	Identity Vault single-valued non-empty	Identity Vault multi-valued empty	Identity Vault multi-valued non-empty
<b>Application single-valued empty</b>	No change	App = Identity Vault	No change	App = Identity Vault[1]
<b>Application single-valued empty</b>	App = empty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

	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>	App = empty	App = Identity Vault	App = empty	App = Identity Vault

### 5.4.3 Scenario Three

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

**Figure 5-3** Scenario Three



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

**Table 5-3** *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





# Managing the Driver

# 6

The driver can be managed through Designer, iManager, or the DirXML<sup>®</sup> Command Line utility.

- ◆ Section 6.1, “Starting, Stopping, or Restarting the Driver,” on page 41
- ◆ Section 6.2, “Using the DirXML Command Line Utility,” on page 42
- ◆ Section 6.3, “Viewing Driver Versioning Information,” on page 42
- ◆ Section 6.4, “Reassociating a Driver Set Object with a Server Object,” on page 47
- ◆ Section 6.5, “Changing the Driver Configuration,” on page 48
- ◆ Section 6.6, “Storing Driver Passwords Securely with Named Passwords,” on page 48
- ◆ Section 6.7, “Adding a Driver Heartbeat,” on page 55

## 6.1 Starting, Stopping, or Restarting the Driver

- ◆ Section 6.1.1, “Starting the Driver in Designer,” on page 41
- ◆ Section 6.1.2, “Starting the Driver in iManager,” on page 41
- ◆ Section 6.1.3, “Stopping the Driver in Designer,” on page 41
- ◆ Section 6.1.4, “Stopping the Driver in iManager,” on page 41
- ◆ Section 6.1.5, “Restarting the Driver in Designer,” on page 42
- ◆ Section 6.1.6, “Restarting the Driver in iManager,” on page 42

### 6.1.1 Starting the Driver in Designer

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

### 6.1.2 Starting the Driver in iManager

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

### 6.1.3 Stopping the Driver in Designer

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

### 6.1.4 Stopping the Driver in iManager

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

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

### 6.1.5 Restarting the Driver in Designer

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

### 6.1.6 Restarting the Driver in iManager

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

## 6.2 Using the DirXML Command Line Utility

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

For example, you could create a shell script on Linux to check the status of the driver. See [Appendix A, “DirXML Command Line Utility,” on page 105](#) for detailed information about the DirXML Command Line utility. For daily tasks, use iManager or Designer.

## 6.3 Viewing Driver Versioning Information

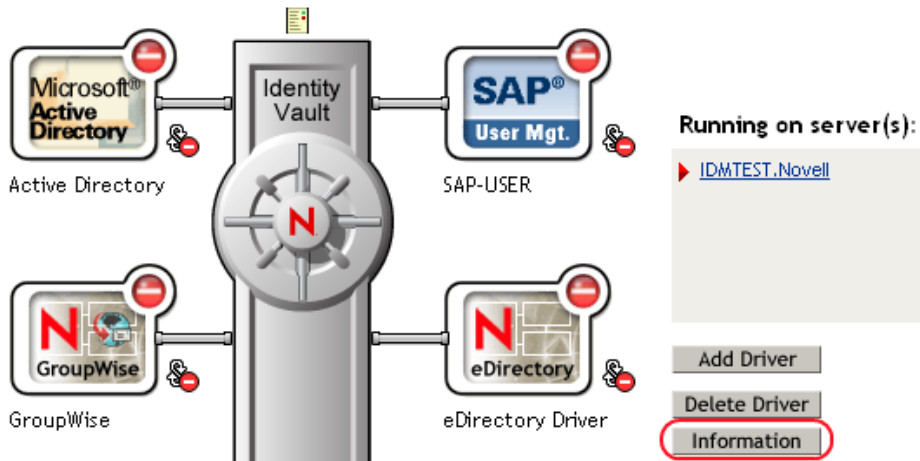
The Versioning Discovery tool only exists in iManager.

- ♦ [Section 6.3.1, “Viewing a Hierarchical Display of Versioning Information,” on page 42](#)
- ♦ [Section 6.3.2, “Viewing the Versioning Information as a Text File,” on page 44](#)
- ♦ [Section 6.3.3, “Saving Versioning Information,” on page 46](#)

### 6.3.1 Viewing a Hierarchical Display of Versioning Information

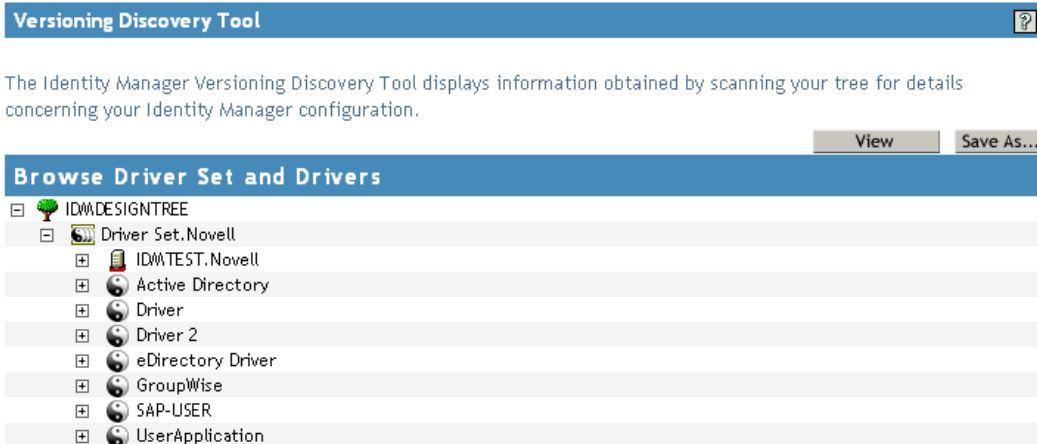
- 1 To find your Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.
- 2 In the Identity Manager Overview, click *Information*.

Driver Set: Driver Set.Novell  Activation required by: November 1, 2006 



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

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



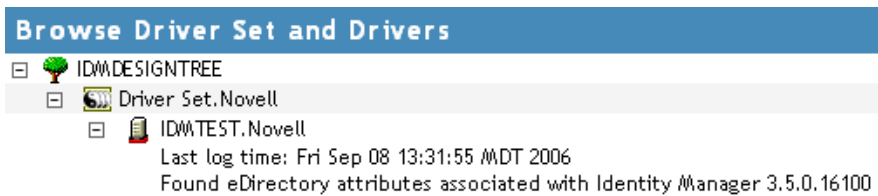
The unexpanded hierarchical view displays the following:

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

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

- ◆ Drivers

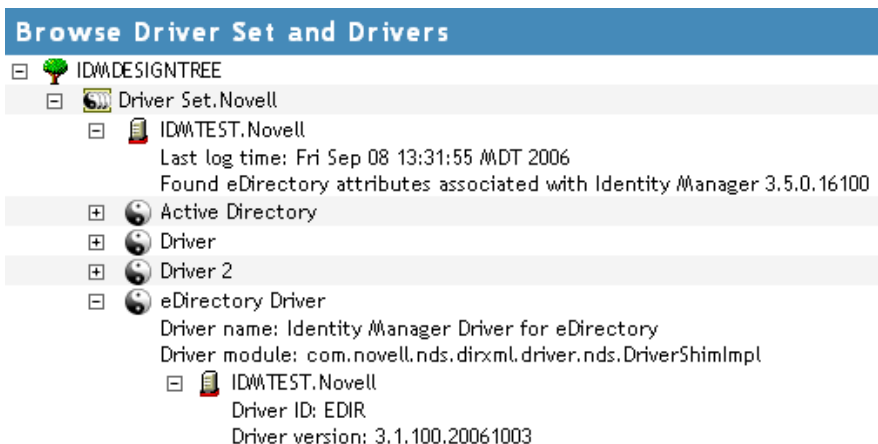
- 4 View versioning information related to servers by expanding the server icon.



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

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

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



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

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


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

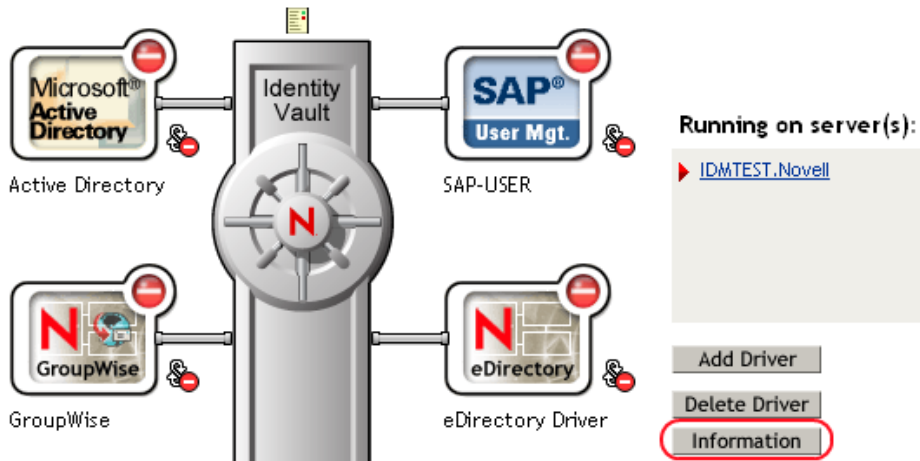
- ◆ The driver ID
- ◆ The version of the instance of the driver running on that server

### 6.3.2 Viewing the Versioning Information as a Text File

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

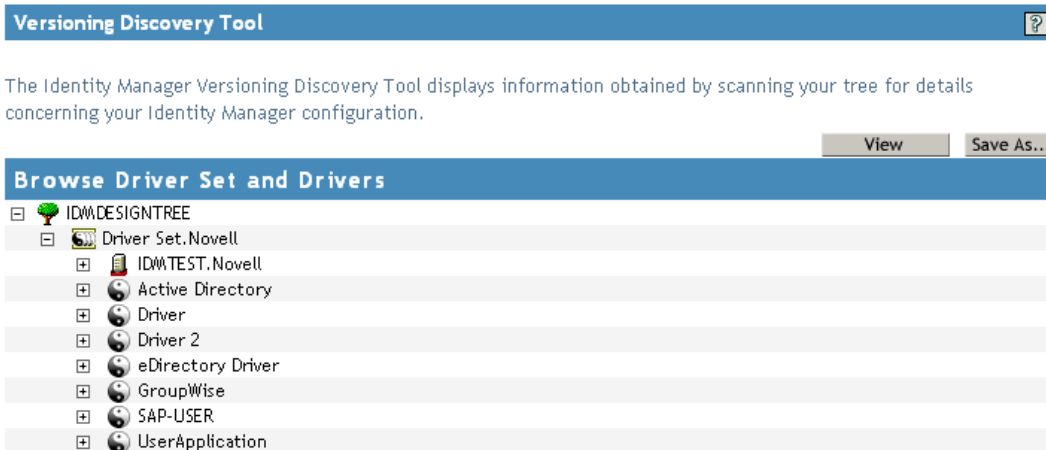
- 1** To find your Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.
- 2** In the Identity Manager Overview, click *Information*.

Driver Set: Driver Set.Novell  [Activation required by: November 1, 2006](#) 



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

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



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

## Versioning Discovery Tool - Report Viewer

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

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

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

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

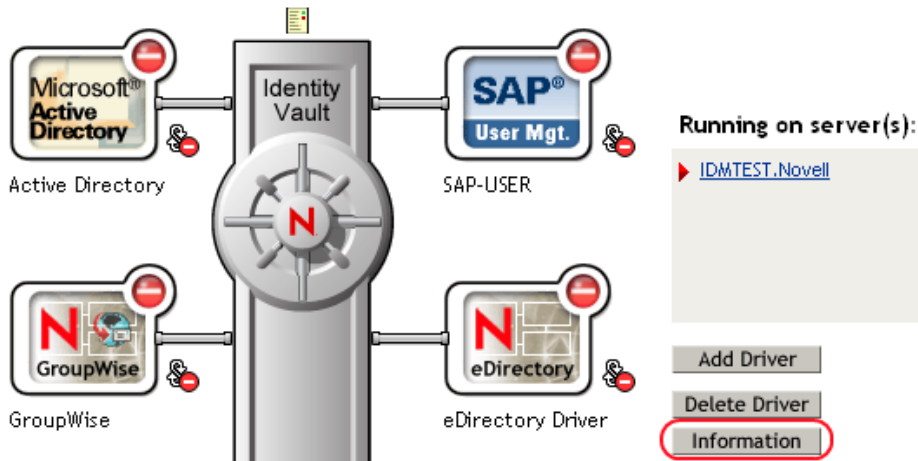
OK

### 6.3.3 Saving Versioning Information

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

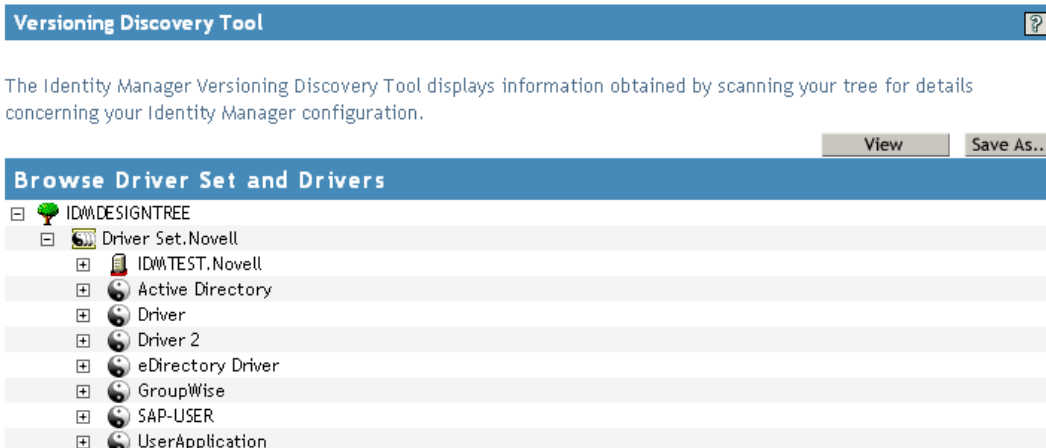
- 1 To find the Driver Set object in iManager, click *Identity Manager > Identity Manager Overview*, then click *Search*.
- 2 In the Identity Manager Overview, click *Information*.

Driver Set: Driver Set.Novell  [Activation required by: November 1, 2006](#) 



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

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



- 4 In the File Download dialog box, click *Save*.
- 5 Navigate to the desired directory, type a filename, then click *Save*.  
Identity Manager saves the data to a text file.

## 6.4 Reassociating a Driver Set Object with a Server Object

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

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

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

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

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

## 6.5 Changing the Driver Configuration

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

To change the driver configuration in iManager:

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

To change the driver configuration in Designer:

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

For a listing of all of the configuration fields, see [Appendix B, “Properties of the Driver,” on page 119](#).

## 6.6 Storing Driver Passwords Securely with Named Passwords

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

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

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

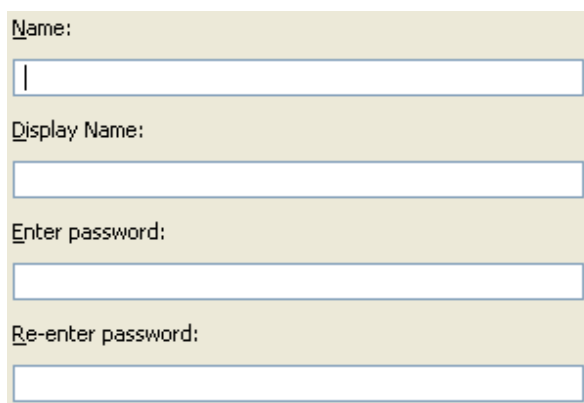


method described in this section for storing and retrieving Named Passwords can be used with any driver without making changes to the driver shim.

- ♦ [Section 6.6.1, “Using Designer to Configure Named Passwords,” on page 49](#)
- ♦ [Section 6.6.2, “Using iManager to Configure Named Passwords,” on page 49](#)
- ♦ [Section 6.6.3, “Using Named Passwords in Driver Policies,” on page 50](#)
- ♦ [Section 6.6.4, “Using the DirXML Command Line Utility to Configure Named Passwords,” on page 51](#)

## 6.6.1 Using Designer to Configure Named Passwords

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



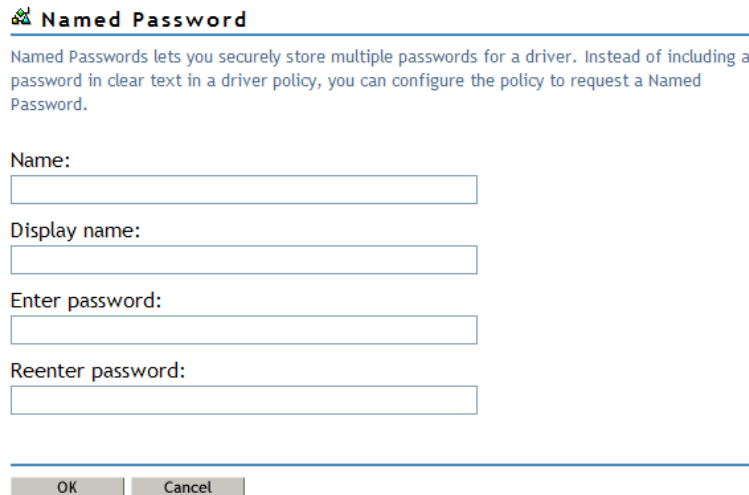
- 3 Specify the *Name* of the Named Password.
- 4 Specify the *Display name* of the Named Password.
- 5 Specify the Named Password, then re-enter the password.
- 6 Click *OK* twice.

## 6.6.2 Using iManager to Configure Named Passwords

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 In the Identity Manager Overview, click the upper right corner of the driver icon, then click *Edit properties*.
- 3 On the Modify Object page on the Identity Manager tab, click *Named Passwords*.  
The Named Passwords page appears, listing the current Named Passwords for this driver. If you have not set up any Named Passwords, the list is empty.



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



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

### 6.6.3 Using Named Passwords in Driver Policies

- ◆ “Using the Policy Builder” on page 51

- ◆ [“Using XSLT” on page 51](#)

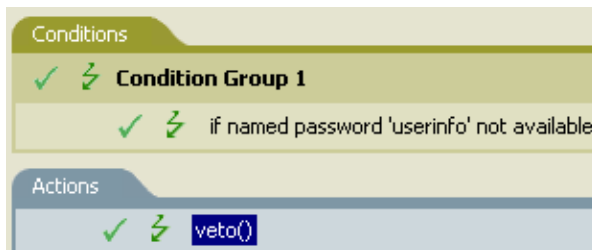
## Using the Policy Builder

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

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

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

**Figure 6-1** A Policy Using Named Passwords



## Using XSLT

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

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

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

- ◆ [“Creating a Named Password in the DirXML Command Line Utility” on page 51](#)
- ◆ [“Using the DirXML Command Line Utility to Remove a Named Password” on page 53](#)

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

- 1 Run the DirXML Command Line utility.

For information, see [Appendix A, “DirXML Command Line Utility,”](#) on page 105.

**2** Enter your username and password.

The following list of options appears.

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

**3** Enter 3 for driver operations.

A numbered list of drivers appears.

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

The following list of options appears.

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

**5** Enter 13 for password operations.

The following list of options appears.

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

- 6 Enter 5 to set a new Named Password.

The following prompt appears:

```
Enter password name:
```

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

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

```
Enter password:
```

The characters you type for the password are not displayed.

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

```
Confirm password:
```

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

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

### Using the DirXML Command Line Utility to Remove a Named Password

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

- 1 Run the DirXML Command Line utility.

For information, see [Appendix A, “DirXML Command Line Utility,” on page 105](#).

- 2 Enter your username and password.

The following list of options appears.

```
DirXML commands
```

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

```
Enter choice:
```

- 3 Enter 3 for driver operations.

A numbered list of drivers appears.

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

The following list of options appears.

```
Select a driver operation for:
```

```
driver_name
```

```
1: Start driver
2: Stop driver
3: Get driver state
4: Get driver start option
5: Set driver start option
6: Resync driver
7: Migrate from application into DirXML
8: Submit XDS command document to driver
9: Submit XDS event document to driver
```

```
10: Queue event for driver
11: Check object password
12: Initialize new driver object
13: Passwords operations
14: Cache operations
99: Exit
Enter choice:
```

**5** Enter 13 for password operations.

The following list of options appears.

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

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

The list of existing Named Passwords is displayed.

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

**7** Enter 6 to remove one or more Named Passwords.

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

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

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

```
Enter password name:
```

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

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

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

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

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

## 6.7 Adding a Driver Heartbeat

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

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

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

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

To configure the heartbeat:

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

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

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

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

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

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

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

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

After you have added the driver parameter, you can edit the time interval by using the graphical view. Another option is to create a reference to a global configuration value (GCV) for the time interval. Like other global configuration values, the driver heartbeat can be set at the driver set level instead of on each individual driver object. If a driver does not have a particular global configuration value, and the driver set object does have it, the driver inherits the value from the driver set object.





# Backing Up the Driver

# 7

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

---

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

---

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

- ♦ [Section 7.1, “Exporting the Driver in Designer,” on page 57](#)
- ♦ [Section 7.2, “Exporting the Driver in iManager,” on page 57](#)

## 7.1 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.

## 7.2 Exporting the Driver in iManager

- 1 In iManager, select *Identity Manager > Identity Manager Overview*.
- 2 Browse to and select the driver set object, then click *Search*.
- 3 Click the driver icon.
- 4 Select *Export* in the Identity Manager Driver Overview window.
- 5 Browse to and select the driver object you want to export, then click *Next*.
- 6 Select *Export all policies, linked to the configuration or not* or select *Only export policies that are linked to the configuration*, depending upon the information you want to have stored in the XML file.
- 7 Click *Next*.
- 8 Click *Save As*, then click *Save*.
- 9 Browse and select a location to save the XML file, then click *Save*.
- 10 Click *Finish*.



# Customizing the NT Domain Driver

# 8

This section covers some general categories of customization:

- ◆ [Section 8.1, “Configuring Driver Parameters,” on page 59](#)

When you change driver parameters, you are tuning driver behavior to align with your network environment. For example, you might find the default publisher polling interval to be shorter than your synchronization needs require. Making the interval longer could improve network performance while still maintaining appropriate synchronization.

- ◆ [Section 8.2, “Configuring Data Synchronization,” on page 63](#)

The real power of Novell<sup>®</sup> Identity Manager is in managing the shared data itself. This section covers some common customizations for the NT Domain driver, such as Exchange integration and local/global group resolution.

---

**NOTE:** When you customize data synchronization, you must work within the supported standards and conventions for the operating systems and accounts being synchronized. Data containing characters that are valid in one environment, but invalid in another, causes errors. Also, keep in mind that attribute names are case sensitive.

---

For information about synchronizing passwords, see [“Password Synchronization” on page 71](#).

## 8.1 Configuring Driver Parameters

Use Novell iManager to make the appropriate adjustments to any of the following properties:

In this section:

- ◆ [Section 8.1.1, “Log Level,” on page 59](#)
- ◆ [Section 8.1.2, “Polling Rate,” on page 60](#)
- ◆ [Section 8.1.3, “Password Expiration Time,” on page 60](#)
- ◆ [Section 8.1.4, “Security Options,” on page 62](#)
- ◆ [Section 8.1.5, “Startup Options,” on page 62](#)
- ◆ [Section 8.1.6, “Additional Options,” on page 63](#)

### 8.1.1 Log Level

The log level determines the kinds of errors that are sent to the Identity Manager status logs, DSTrace, and Novell Audit. For complete information about Novell Audit and Identity Manager, see [“Integrating Identity Manager with Novell Audit”](#) in the *Identity Manager 3.5.1 Logging and Reporting*.

You can set one of the following options:

- ◆ Log errors
- ◆ Log errors and warnings
- ◆ Log all messages

- ◆ Only update the last log time
- ◆ Logging off

To set the log level:

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Select the driver set containing the driver, click the driver icon to see the driver overview, then click the driver icon again to edit driver parameters.
- 3 Click the *Log Level* link at the top of the page, select a level, then click *OK*.

### 8.1.2 Polling Rate

The driver re-reads the SAM registry once each polling interval, looking for new or modified users. Setting the polling rate too fast uses all available processing cycles. The minimum polling rate is three seconds (3000 milliseconds). The recommended rate is one minute (60000 milliseconds).

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Select the driver set containing the driver, click the driver icon to see the driver overview, then click the driver icon again to edit driver parameters.
- 3 Select a polling rate from the list, then click *OK*.

### 8.1.3 Password Expiration Time

The driver and the password filter have been enhanced in the following ways to improve how password synchronization is retried after a failure:

- ◆ If a password change sent from NT is not completed successfully in the Identity Vault, the password is cached by the driver. It is not retried again until an Add or Modify event occurs for the user the password belongs to. (Previously, these saved passwords were retried at every polling interval.)

When the driver polls for changes in NT, it receives Add or Modify events for users. For each user Add or Modify event, the driver checks to see if it has a password saved for this new user. If it does, the driver sends the password to the Identity Vault as a Modify User event.

If you have set up Password Synchronization to send e-mail messages to users when password synchronization fails, this enhancement minimizes the number of e-mails a user might receive.

- ◆ A parameter named Password Expiration Time has been added. This interval lets you determine how long to save a particular user's password if synchronization is not successful on the first try. A password is saved by the driver until it is successfully changed in the Identity Vault, or until the Password Expiration Time elapses.

You are prompted to specify this interval when you import the sample driver configuration.

If no interval is specified, or if the interval field contains invalid characters, the default setting is 60 minutes. If the interval specified is less than twice the polling interval specified, the driver changes the interval to be at least twice the polling interval.

For more understanding of why these enhancements are important, review the following information.

The driver checks for changes to users in NT based on a polling interval. In contrast, the password filter is event-driven, meaning that it sends password changes from NT to the driver as soon as they

occur. After a user is created in the Identity Vault to correspond to an NT user, this immediate response for password synchronization is helpful. However, because of the differences between polling and event-driven activity, password synchronization for new users might not be immediate.

Issues such as the difference between polling and event-driven activity, and business practices such as Create policies and Password Policies, can lead to scenarios like the following. The scenarios also explain how the Password Expiration Time parameter is applicable in each case.

- ◆ A new user is created in NT with a password. The filter immediately sends the new password to the driver, but the driver has not yet received that user Add event because the event occurred between polling intervals. Because the driver has not yet created the user in the Identity Vault, the password synchronization is not successful on this first attempt. The driver caches the password.

At the next polling interval, the driver receives the Add User event for the new user, and also checks to see if it has a password cached for this new user. The driver sends the Add User event to the Identity Vault, and also sends a Modify User event to synchronize the password.

In this case, the password synchronization is delayed by only one polling interval.

The Password Expiration Time parameter does not have an effect in this situation.

- ◆ A new user is created in NT with a password, but the user information does not meet the requirements of the Create policy for the NT driver. For example, the Create policy requires a full name, and the required information is missing. Like the previous example, the filter sends the password change to the driver immediately, but on the first try the password change is not successful in the Identity Vault because the user does not exist yet. The driver caches the password.

In this case, however, even when the driver polls for changes in NT and discovers the new user, the driver cannot create the new user because the user information does not meet the requirements of the Create policy.

The new user creation and password synchronization is delayed until all the user information is added in NT to satisfy the Create policy. Then the driver adds the new user in the Identity Vault, checks to see if it has a password cached for this new user, and sends a Modify User event to synchronize the password.

The Password Expiration Time parameter affects this scenario only if the time interval elapses before the user information in NT meets the requirements of the Create policy. After the Password Expiration Time parameter elapses, the driver removes the password change from the cache. If the user later meets the requirements and is created in the Identity Vault after the Password Expiration Time has passed, this means that the driver does not have a password cached for that user and cannot synchronize a password in the Identity Vault at that time. Instead, the password is synchronized the next time it is changed in NT.

If Password Synchronization is set up for bidirectional flow of passwords, a password can also be synchronized from the Identity Vault to NT when a password change is made in the Identity Vault.

If your Create policy is restrictive, and it generally takes a couple of days for a new user's information to be completed in NT, you might want to increase the Password Expiration Time parameter interval accordingly, so that passwords are cached by the driver until the user is finally created in the Identity Vault.

- ◆ A user is created in NT with a password, but this user never meets the criteria of the Create policy for the NT driver. For example, the new user in NT has a Description that indicates the user is a contractor, and the Create policy blocks creation of user objects for contractors

because the business policy is that contract employees are not intended to have a corresponding user account in the Identity Vault. Like the previous example, the filter sends the password change immediately, but the password synchronization is not successful on the first attempt. The driver caches the password.

In this case, a corresponding user account is never created in the Identity Vault, so the driver never synchronizes the cached password. After the Password Expiration Time has passed, the driver removes the user password from its cache.

- ♦ A user with an NT account and a corresponding Identity Vault account changes his NT password. The NT password chosen by the user contains 6 characters, so it does not meet the 8-character minimum required by the Password policy the administrator created in the Identity Vault. Password Synchronization is configured to reject passwords that do not meet the policy and to send a notification e-mail to the user saying that password synchronization failed. The driver caches the password, and retries it only if a change is made to the user object in NT.

In this case, shortly after the user changes his password, he receives an e-mail stating that the password synchronization was not successful. He receives the same e-mail message each time the driver retries the password.

If the user changes his password in NT to one that complies with the Password policy, the driver successfully synchronizes the new password to the Identity Vault.

If the user does not change to a compliant password, the password synchronization is never successful. When the Password Expiration Time elapses, the driver deletes the cached password and no longer retries it.

## 8.1.4 Security Options

Creating a new user that has Read/Write rights to the domain and to the SAM registry makes Identity Manager easier to manage. This user account is used exclusively by the NT Domain Driver. This user is also a user you should exclude from synchronization because its sole purpose is to provide rights for the NT Domain Driver. After you create this user, you can assign the driver to use that user account.

To set up these security options:

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Select the driver set containing the driver, click the driver icon to see the driver overview, then click the driver icon again to edit driver parameters.
- 3 Click *Driver Configuration* at the top of the page, then enter the appropriate data in the *Authentication* fields.
- 4 Click *OK* to save the changes.

## 8.1.5 Startup Options

You can set driver startup to any of the following three options:

- ♦ **Auto Start:** When the Metadirectory engine is started, the driver starts automatically. After you have the driver configured, it is good to use this option.
- ♦ **Manual:** The driver cannot start until it is started through the status indicator on the driver icon. If an error brings the driver down, it does not restart until manually started. This option is often used during driver modification and testing cycles. The engine buffers changes to be processed when driver is started.

- ♦ **Disabled** If the driver is disabled, the Metadirectory engine does not cache events. However, upon driver startup, data changes resulting from Add or Modify (of objects with an association) events are synchronized. Data changes resulting from Delete, Rename, or Move events are not synchronized.

To set startup options:

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Select the driver set containing the driver, click the driver icon to see the driver overview, then click the driver icon again to edit driver parameters.
- 3 Click *Driver Configuration* at the top of the page, then select one of the three options listed under *Startup Options*.

## 8.1.6 Additional Options

The driver has additional parameters and global configurations values that can be changed. See [Appendix B, “Properties of the Driver,” on page 119](#) for more information about these options.

## 8.2 Configuring Data Synchronization

This section covers the following configuration topics:

- ♦ [Section 8.2.1, “Integrating the Identity Manager Driver for NT Domain and the Identity Manager Driver for Exchange,” on page 63](#)
- ♦ [Section 8.2.2, “Filtering Out Non-User Objects,” on page 64](#)
- ♦ [Section 8.2.3, “Synchronizing Group Information,” on page 65](#)
- ♦ [Section 8.2.4, “Changing the Location of User Objects by Using Placement Policies,” on page 66](#)
- ♦ [Section 8.2.5, “Changing Which Attributes Are Synchronized by Using Publisher and Subscriber Filters,” on page 66](#)
- ♦ [Section 8.2.6, “Querying GlobalGroup or LocalGroup,” on page 69](#)

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

---

**IMPORTANT:** If you are using both the NT driver and the Exchange driver, you should complete the following procedure.

---

The Identity Manager Driver for NT Domain and the Identity Manager Driver for Exchange can both create users in the domain. To avoid a conflict, a mechanism can be set up that uses Identity Manager policies to solve this problem.

The Identity Manager Driver for NT Domain has a User attribute called DirXML-NTAccountName. This attribute contains the DomainName/UserName attribute. This value is what the Exchange MailBox and Remote objects need 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 Using DirXML<sup>®</sup> Script, edit the existing Subscriber Create policy for the Exchange driver (or create a new policy) so that a new MailBox object is not created unless the DirXML-NTAccountName attribute is populated.
- 2 Verify that the DirXML-NTAccountName attribute is in both the Publisher filter on the Identity Manager Driver for NT Domain and the Subscriber filter on the Identity Manager Driver for Exchange.
- 3 Restart both drivers.

## Data Flow in the NT Domain and Exchange 5.5 Drivers

The changes outlined in [“Integrating the Identity Manager Driver for NT Domain and the Identity Manager Driver for Exchange” on page 63](#) ensure the following control flow:

1. A user is created in eDirectory.
2. The Identity Manager Driver for NT Domain is handed a Create request. 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 feeds back the name of the NT account just created to the DirXML-NTAccountName attribute.
4. The Identity Manager Driver for Exchange is now notified. It creates the mailbox and associates the mailbox with the NT account information stored in the Identity Vault.

---

**NOTE:** Although the examples use DirXML-NTAccountName as the eDirectory attribute to hold the NT account information, you can choose any attribute that works for you.

---

## 8.2.2 Filtering Out Non-User Objects

The NT registry tracks some non-user data along with user data. For example, information about workstation objects appears as User objects in the NT User Manager. This information is synchronized to the Identity Vault unless you filter it out using a style sheet. The following style sheet can be used in the Event Transformation to ensure that only real user objects are synchronized.

```
<xsl:template match="node()|@*">
  <xsl:copy>
    <xsl:apply-templates select="node()|@*" />
  </xsl:copy>
</xsl:template>

<!-- Test for Non-User user objects like workstations that have a $ in
the name -->
  <xsl:template match="add[@class-
name='User']|modify[@class-name='User']|sync[@class-name='User']">
    <xsl:choose>
      <xsl:when test="contains(@src-dn,'$')"/>
        <xsl:otherwise>
          <xsl:copy>
            <xsl:apply-templates
select="node() | @*" />
          </xsl:copy>
        </xsl:otherwise>
      </xsl:choose>
    </xsl:template>
```



```

        </xsl:otherwise>
      </xsl:choose>
    </xsl:template>
  </xsl:stylesheet>

```

## 8.2.3 Synchronizing Group Information

The driver allows you to synchronize group information in both the user attributes holding group membership information and the group objects themselves.

This functionality allows you to see which groups a user is a part of, whether you're looking at the user in the Identity Vault or in NT.

To synchronize group information:

- 1** Ensure that the groups to be synchronized exist as identically named objects in both the Identity Vault and in NT.

For example, if you want to synchronize group information for the NT global group titled Domain User, you should create a group object named Domain User in eDirectory™.

- 2** Create an Identity Manager association between the NT group and the eDirectory group.

**2a** In iManager, click *eDirectory Administration > Modify Object*.

**2b** Browse to and select the eDirectory group to be synchronized, then click *OK*.

**2c** Click the *Identity Manager* tab, then click *Add*.

The Add Association dialog box appears.

**2d** Specify the Identity Manager driver for NT in the *Integration Driver Object* field.

**2e** Specify the NT group name in the *Associated Object ID* field, using uppercase as shown in the following syntax:

```
\DOMAINNAME\GROUPNAME
```

**2f** Click *OK*.

The new association is displayed in the Associations page.

- 3** Edit the Schema Mapping policy to map the NT UserLocalGroups and UserGlobalGroups attributes to eDirectory attributes.

**3a** Click *Identity Manager > Identity Manager Overview*, then select the driver set containing the Identity Manager driver for NT.

**3b** Click the driver to display the Driver Overview page.

**3c** Double-click the Schema Map policy and map the new attributes.

You can map the NT attributes to any multivalue string attribute. UserGlobalGroups is commonly mapped to the GroupMembership attribute.

- 4** If you are publishing data from NT to the Identity Vault, double-click the Publisher filter icon and add the new attributes.

- 5** If you are subscribing to data held in the Identity Vault, double-click the Subscriber filter icon and add the new attributes.

- 6** Click *OK*.

Group information begins to synchronize when the driver is restarted and a change to user information occurs.

---

**NOTE:** If you use User Manager to change the group membership attribute values without making changes to any other data, this update does not synchronize immediately. Changes are synchronized the next time the NT user logs in or the next time user object data changes.

---

## 8.2.4 Changing the Location of User Objects by Using Placement Policies

Modify the Subscriber and Publisher Placement policies to match the eDirectory container with the NT domain name you have set up. Placement policies are created when you import the sample driver configuration file.

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Select the driver set containing the driver, then click the driver icon.  
The Driver Overview is displayed. Policies can be edited here.
- 3 Double-click the Placement policy you want to edit, then make the appropriate changes.
- 4 Click *OK*.

---

**IMPORTANT:** All Placement policies must use the slash syntax.

---

## 8.2.5 Changing Which Attributes Are Synchronized by Using Publisher and Subscriber Filters

- 1 In iManager, click *Identity Manager > Identity Manager Overview*.
- 2 Select the driver set containing the driver, then click the driver icon.  
The Driver Overview is displayed. Policies can be edited here.
- 3 Double-click the filter icon and add or remove the appropriate attributes.  
Select the eDirectory user attributes that you want to synchronize with.  
The driver supports the Domain User object. The attributes that the driver supports within the User object are the attributes that are accessible by using the USER\_INFO\_3 data structure using the NetUser APIs.  
For a list of the supported attributes, see [Table 8-1 on page 66](#).
- 4 Click *OK*.

---

**IMPORTANT:** Keep in mind that attribute names are case sensitive.

---

**Table 8-1** Supported Attributes

Driver Attribute	USER_INFO_3Name	Data Type	Description
Name	usri3_name	LPWSTR	Specifies the name of the user account. The name cannot exceed UNLEN.
(Can be set through a Create policy.)	usri3_password	LPWSTR	The password of the user. The length cannot exceed PWLEN.

Driver Attribute	USER_INFO_3Name	Data Type	Description
PasswordAge	usri3_password_age	DWORD	Read-only. Specifies the number of seconds elapsed since the password was last changed.
PrivilegeLevel	usri3_priv	DWORD	Specifies the privilege level of the user: Guest, User, or Administrator.
HomeDirectory	usri3_home_dir	LPWSTR	Points to a Unicode* string that contains the path of the home directory of the user. The string can be null. The string cannot exceed PATHLEN. The Subscriber, on an Add event, creates the folder specified by the path as a Shared to Everyone folder, if it does not already exist.
Comment	usri3_comment	LPWSTR	Points to a Unicode string that contains a comment. The string can be null. The comment cannot exceed 1024 characters in length.
Flags	usri3_flags	DWORD	Contains values that determine several features. See USER_INFO_3 documentation.
LogonDisable	usri3_flags	LPWSTR TRUE or FALSE	Represents a bit in the usri_flags that is the UF_ACCOUNTDISABLE. The user's account is disabled.
PasswordChange	usri3_flags	LPWSTR TRUE or FALSE	Represents a bit in the usri_flags that is the UF_PASSWD_CANT_CHANGE. The user cannot change the password if this value is TRUE.
PasswordRequired	usri3_flags	LPWSTR TRUE or FALSE	Represents a bit in the usri_flags that is the PASSWD_NOTREQ. No password is required.
ScriptPath	usri3_script_path	LPWSTR	Points to a Unicode string specifying the path of the user's logon script. The string can be null. The string cannot exceed PATHLEN.
AuthorizationFlags	usri3_auth_flags	DWORD	Read-only. Specifies an unsigned long integer that contains values that specify the user's privileges.
FullName	usri3_full_name	LPWSTR	Points to a Unicode string that contains the full name of the user. This string can be null or up to 1024 characters in length.
UserComment	usri3_usr_comment	LPWSTR	Points to a Unicode string that contains a user comment. This string can be null or up to 1024 characters in length.
AppParams	usri3_parms	LPWSTR	Read-only. A Unicode string used by Microsoft* products.

Driver Attribute	USER_INFO_3Name	Data Type	Description
Workstations	usri3_workstations	LPWSTR	Points to a Unicode string that contains the names of the workstations from which the user can log on. This string can be null or up to 1024 characters in length.
LastLogon	usri3_last_logon	DWORD	Read-only. Specifies when the last logon occurred. The value is stored as the number of seconds elapsed since 00:00:00, January 1, 1970.
LastLogoff	usri3_last_logoff	DWORD	Specifies when the last logoff occurred. The value is stored as the number of seconds elapsed since 00:00:00, January 1, 1970.
AccExpires	usri3_acct_expires	DWORD	Specifies when the account expires. The value is stored as the number of seconds elapsed since 00:00:00, January 1, 1970. A value of TIMEQ_FOREVER indicates that the account never expires. The driver will map this to what eDirectory is looking for.
MaxStorage	usri3_max_storage	DWORD	Specifies the maximum amount of disk space the user can use. Use USER_MAXSTORAGE_UNLIMITED to use all available disk space.
UnitsPerWeek	usri3_units_per_week	DWORD	Read-only. Specifies the number of equal-length time units into which the week is divided.
LogonHours	usri3_logon_hours	PWORD	The driver maps this to an octet string that specifies an account's allowed login time periods for each day of the week to a precision of one-half hour.
BadPasswordCnt	usri3_bad_pw_count	DWORD	Read-only. Counts the number of times the user tried to log in to the account using the incorrect password.
NumLogons	usri3_num_logons	DWORD	Read-only. Counts the number of successful times the user logged in to this account.
LogonServer	usri3_logon_server	LPWSTR	Read-only. Points to a Unicode string that contains the name of the server to which login requests are sent.
CountryCode	usri3_country_code	DWORD	Specifies the country code for the user's language of choice.
CodePage	usri3_code_page	DWORD	Specifies the code page for the user's language of choice.
UserID	usri3_user_id	DWORD	Read-only. Specifies the relative ID (RID) of the user.
PrimaryGroupID	usri3_primary_group_id	DWORD	Specifies the relative ID (RID) of the primary global group of the user.

Driver Attribute	USER_INFO_3Name	Data Type	Description
Profile	usri3_profile	LPWSTR	Specifies a path to the user's profile. This value can be a null string, a local absolute path, or a UNC path. The length of the string cannot exceed PATHLEN.
HomeDirDrive	usri3_home_dir_drive	LPWSTR	Specifies the drive letter assigned to the user's home directory for login purposes.
PasswordExpired	usri3_password_expired	DWORD	Determines whether the password of the user has expired. Use zero if the password has not expired and non-zero if it has expired.  Although this attribute is supported, keep in mind that the eDirectory attribute named Password Expiration Time is used to expire a password by setting a date and time that is previous to the current date, instead of by setting a zero or non-zero value.  This means that these attributes are not easily mapped to each other.

The driver also supports the UserGlobalGroups and UserLocalGroups that are accessible through the NetUserGroup API.

The following table lists the supported attributes:

**Table 8-2** Supported Group Attributes

Driver Attribute	Data Type	Description
UserGlobalGroups	LPWSTR	A multivalued attribute that contains the global groups the user is a member of.
UserLocalGroups	LPWSTR	A multivalued attribute that contains the global groups the user is a member of.

## 8.2.6 Querying GlobalGroup or LocalGroup

You can query for GlobalGroup or LocalGroup objects, although you can't synchronize them on the Subscriber or Publisher channel.

The query supports the following attributes.

- ◆ **GlobalGroup:** Name, Comment, MemberOf
- ◆ **LocalGroup:** Name, Comment

A query is successful if the SearchClass is GlobalGroup or LocalGroup and any of the following are true:

- ◆ The query includes all of the attributes.
- ◆ The query includes some of the attributes.
- ◆ The query includes none of the attributes.

This feature could be used to synchronize GlobalGroups or LocalGroups in an indirect way. For example, you could use a style sheet to configure the driver to query for them when you are migrating users, and create corresponding Group objects in eDirectory. Doing this would allow the MemberOf attribute for an NT user to work for making a user a member of matching groups in eDirectory (this aspect would work without an additional style sheet). To keep the GlobalGroups and LocalGroups mirrored in eDirectory using this method, you would need to periodically migrate again as new groups are added or removed from NT.

In the sample driver configuration, this feature is used if you choose the Role-Based Entitlements option, to allow you to assign a user to a GlobalGroup or LocalGroup in NT as an entitlement. (Using Role-Based Entitlements is a design decision. Choose this option only after you have reviewed “[Creating and Using Entitlements](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.)

# Password Synchronization

# 9

This section assumes that you are familiar with the information in “[Password Synchronization across Connected Systems](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*. The information in this section is specific to this driver.

---

**IMPORTANT:** If you have used Password Synchronization 1.0 previously, don’t install the new driver shim until you have read [Section 9.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on page 73 and understand the implications. If you install the driver shim, you need to add backward compatibility for Password Synchronization 1.0 to your driver policies at the same time, even if you are not planning to immediately use the Password Synchronization provided with Identity Manager.

---

In this section:

- ◆ [Section 9.1, “Comparison of Password Synchronization 1.0 and Password Synchronization Provided with Identity Manager,”](#) on page 71
- ◆ [Section 9.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on page 73
- ◆ [Section 9.3, “New Driver Configuration and Identity Manager Password Synchronization,”](#) on page 77
- ◆ [Section 9.4, “Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization,”](#) on page 77
- ◆ [Section 9.5, “Setting Up Password Synchronization Filters,”](#) on page 80
- ◆ [Section 10.2, “Troubleshooting Password Synchronization,”](#) on page 95

## 9.1 Comparison of Password Synchronization 1.0 and Password Synchronization Provided with Identity Manager

**Table 9-1** *Password Synchronization 1.0 Versus Password Synchronization with Identity Manager*

	<b>Password Synchronization 1.0</b>	<b>Password Synchronization with Identity Manager 2</b>
Product delivery	A product separate from Identity Manager.	A feature included with Identity Manager, not sold as a separate product.

	Password Synchronization 1.0	Password Synchronization with Identity Manager 2
Platforms	<ul style="list-style-type: none"> <li>◆ Active Directory*</li> <li>◆ NT Domain</li> </ul>	<p>Full bidirectional password synchronization is supported on these platforms:</p> <ul style="list-style-type: none"> <li>◆ Active Directory</li> <li>◆ eDirectory™</li> <li>◆ NIS</li> <li>◆ NT Domain</li> </ul> <p>These connected systems support publishing user passwords to Identity Manager. Because Universal Password and Distribution Password are reversible, Identity Manager can distribute passwords to connected systems.</p> <p>Any connected system that supports the Subscriber password element can subscribe to passwords from Identity Manager.</p> <p>See <a href="#">“Password Synchronization across Connected Systems”</a> in the <i>Novell Identity Manager 3.5.1 Administration Guide</i>.</p>
Password used in eDirectory	eDirectory Password (non-reversible)	Universal Password (reversible), or Distribution Password (also reversible). The eDirectory password can also be kept synchronized, if desired. For example scenarios, see <a href="#">“Implementing Password Synchronization”</a> in the <i>Novell Identity Manager 3.5.1 Administration Guide</i> .
Main functionality for Windows connected systems	To send passwords to Identity Manager so the eDirectory password is synchronized with the Windows password. Because the eDirectory password was not reversible, passwords were not sent back to NT or AD.	To provide bidirectional password synchronization. Because Universal Password and Distribution Password are reversible, passwords can be synchronized in both directions.
LDAP changes	Not supported.	Supported
Novell Client™	Required.	Not required.
nadLoginName attribute	Used for keeping passwords updated.	Not used.



	Password Synchronization 1.0	Password Synchronization with Identity Manager 2
The component that contains the password synchronization functionality	The Identity Manager driver contained the functionality for updating nadLoginName.	Policies in the driver configuration provide the password synchronization functionality. The driver simply carries out the tasks given by the Metadirectory engine, which come from logic in the policies.  The driver manifest, global configuration values, and driver filter settings must also support password synchronization. These are included in the sample driver configurations, or can be added to an existing driver. See <a href="#">Section 9.4, “Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization,”</a> on page 77.
Agents	A separate piece of software.	No agents are installed; instead, the functionality is now part of the driver.

## 9.2 Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager

If you are currently using Password Synchronization 1.0, complete the instructions in this section to upgrade. If you are running Identity Manager 2.x, and are using Universal Password, these procedure is not needed.

**IMPORTANT:** Do not install the identity Manager driver shim until you have reviewed these instructions.

With the exception of one step, these instructions are the same for both NT and AD, so both drivers are mentioned throughout.

To upgrade from Password Synchronization 1.0 to Password Synchronization provided with Identity Manager:

- 1 Make sure your environment is ready to use Universal Password, including upgrading the Novell Client if you are using it in your environment. See [“Preparing to Use Identity Manager Password Synchronization and Universal Password”](#) in the *Novell Identity Manager 3.5.1 Administration Guide*.

Identity Manager Password Synchronization does not require the Novell Client to be installed on Windows machines.

- 2 If you are running DirXML<sup>®</sup> 1.1a, install the Identity Manager 3.5.1 driver shim and immediately complete [Step 3](#).

Use the installation program as described in [“Installing Identity Manager”](#) in the *Identity Manager 3.5.1 Installation Guide*, and select only the Identity Manager Driver for NT Domain.

- 3 Create backward compatibility with Password Synchronization 1.0, by adding a new policy to the driver configuration as described in [Section 9.2.1, “Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies,”](#) on page 75.

A DirXML 1.1a driver shim updates the nadLoginName attribute. The Identity Manager driver shim does not, so you must add policies to the driver configuration to update nadLoginName.

This allows Password Synchronization 1.0 to function as usual when you install the driver shim, so no password changes are missed while you finish deploying Identity Manager Password Synchronization.

---

**IMPORTANT:** If you don't do this, Password Synchronization 1.0 continues to update existing users, but any new or renamed users are not synchronized until you deploy Identity Manager Password Synchronization.

---

When you complete this step, you have the new driver shim and the policies for backward compatibility, so your driver is supporting Password Synchronization 1.0.

If you can't complete the rest of this procedure right away, you can continue to use Password Synchronization 1.0 until you are ready to finish deploying Identity Manager Password Synchronization.

- 4 Add support for Identity Manager Password Synchronization to each driver that you want to participate in password synchronization, by either upgrading an existing configuration or replacing an existing configuration:

**Upgrade existing configuration:** Upgrade your existing DirXML 1.1a driver configuration by converting it to Identity Manager format and adding the policies needed for Identity Manager Password Synchronization:

- ◆ Convert the driver to Identity Manager format using a wizard. 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*.
- ◆ Add policies to support Identity Manager Password Synchronization. You can use an “overlay” configuration file to add the policies, driver manifest, and GCVs, all at once. You must also add an attribute to the Filter. For instructions, see [Section 9.4, “Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization,”](#) on page 77.

**Replace the existing configuration with Identity Manager configuration, and add backward compatibility again:** The Identity Manager sample driver configuration contains the policies, driver manifest, GCVs, and filter settings to support Identity Manager Password Synchronization. See “[Driver Configuration Parameters](#)” on page 21 for information on importing the new driver configuration.

- ◆ If you choose to replace your existing configuration, make sure you add backward compatibility again, as described in [Section 9.2.1, “Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies,”](#) on page 75. The Identity Manager sample driver configuration does not contain those policies.
  - ◆ Make sure nadLoginName attribute is set to Publish and Subscribe in the filter for NT, and Publish for AD, as it was in your previous driver configuration.
- 5 Install new Password Synchronization filters and configure them if you want the connected system to provide user passwords to Identity Manager. See [Section 9.5, “Setting Up Password Synchronization Filters,”](#) on page 80.
  - 6 Turn on Universal Password for eDirectory user accounts by creating Password Policies with Universal Password enabled.

See “[Managing Password Synchronization](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

We recommend that you assign Password Policies as high up in the tree as possible, to simplify administration.

**7** Set up the scenario for Password Synchronization that you want to use, using the Password Policies and the Password Synchronization settings for the driver.

See “[Implementing Password Synchronization](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

**8** Test synchronization.

**9** After Identity Manager Password Synchronization is working, remove Password Synchronization 1.0:

**9a** Turn off Password Synchronization 1.0 by using Add/Remove Programs to remove the agent.

**9b** In the filter for the driver, change the nadLoginName attribute to Ignore.

**9c** Remove the backward compatibility policies that are updating nadLoginName from the driver configuration.

**9d** If desired, you can also remove the nadLoginName attribute from users after Identity Manager Password Synchronization is working, because it is no longer needed.

## 9.2.1 Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies

Password Synchronization 1.0 relies on the driver shims updating an attribute named nadLoginName. This is the attribute that indicates whether a user’s password should be synchronized. If a new user was added or the user’s name was changed, the nadLoginName attribute was added or updated to match.

The driver shims in Identity Manager no longer update this attribute because it is not necessary for Identity Manager Password Synchronization. So, after you install the new driver shim, the nadLoginName attribute is not being updated. This means that Password Synchronization 1.0 no longer receives notice of new or renamed users unless you add backward compatibility to your driver configuration.

For a smooth transition from Password Synchronization 1.0 to Identity Manager Password Synchronization, you need backward compatibility with Password Synchronization 1.0.

To create backward compatibility with Password Synchronization 1.0, you must add policies that update the nadLoginName attribute.

These policies must be added for both AD and NT drivers, and they must be added regardless of whether you are updating your existing driver configurations, or replacing them with new configurations that ship with Identity Manager. The Identity Manager sample driver configurations for AD and NT do not include them by default.

Three policies are necessary, one each for the Subscriber Output Transformation, Publisher Input Transformation, and Publisher Command Transformation. These policies are provided with Identity Manager in a configuration file named Password Synchronization 1.0 Policies for AD and NT. The following procedure explains how to import the new policies and add them to a driver configuration.

**1** In iManager, click *Identity Manager Utilities > Import Drivers*.

The Import Driver Wizard opens.

**2** Select the driver set where your existing AD or NT driver resides.

**3** In the list of driver configurations that appears, scroll to the bottom and select *Legacy Password Synchronization 1.0 Policies: Backwards Compatibility for AD and NT*.

It is listed under the *Additional Policies* heading.

**4** Complete the import prompts:

**4a** Select your existing AD or NT driver.

Selecting the existing driver allows you to add the three policies that are necessary. The import process creates three new policy objects, which you must then insert in the appropriate place in the driver configuration.

**4b** Specify whether the driver is an AD or NT driver.

The policies imported have minor differences depending on which system is chosen.

**4c** Browse for and select the `nadDomain` object associated with the driver you want to update.

It can normally be found under the driver object.

**4d** (AD only) Specify the name of the eDirectory attribute mapped to the AD attribute `sAMAccountName`.

You can find this information in the Schema Mapping policy in the driver configuration.

**5** Click *Next*.

Because you chose an existing driver, a page appears asking you to decide how you want the driver to be updated. In this case, you just want to update selected policies.

**6** Select *Update Only Selected Policies in That Driver*, and select the check boxes for all three policies listed.

**7** Click *Next*, then click *Finish* to complete the wizard.

At this point, the three new policies have been created as policy objects under the driver object, but are not yet part of the driver configuration. To link them in, you must manually insert each of them at the right point in the driver configuration on the Subscriber and Publisher channels.

**8** Insert each of the three new policies into the correct place on your existing driver configuration. If there are multiple policies for any of these parts of the driver configuration, make sure these new policies are listed last.

Policy Object Name	Where to Insert It
<b>For an NT driver, use the following:</b>	
PassSync(Pub)-Command Transform Policies	Command Transformation Policies on the Publisher channel
PassSync(Pub)-Input Transform Policies	Input Transformation Policies on the Publisher channel
PassSync(Sub)-Command Transform Policies	Command Transformation Policies on the Subscriber channel
<b>For an Active Directory driver, use the following:</b>	
PassSync(Pub)-Command Transform Policies	Command Transformation Policies on the Publisher channel
PassSync(Pub)-Input Transform Policies	Input Transformation Policies on the Publisher channel

Policy Object Name	Where to Insert It
PassSync(Sub)-Output Transform Policies	Output Transformation Policies on the Subscriber channel

Use the following procedure. Repeat these steps for each policy.

- 8a** Click *Identity Manager > Identity Manager Overview*. Select the driver set for the driver you are updating.
  - 8b** Click the driver you just updated.  
A page opens showing a graphical representation of the driver configuration.
  - 8c** Click the icon for the place where you need to add one of the three new policies.
  - 8d** Click *Insert* to add the new policy. In the Insert page that appears, click *Use an Existing Policy*, browse for and select the new policy object, then click *OK*.
  - 8e** If you have more than one policy in the list for any of the three new policies, use the arrow buttons to move the new policy down so it is last in the list.
- 9** Repeat this procedure for all your AD and NT Domain drivers.

After you have completed this procedure, the driver configurations for your AD and NT Domain drivers are backward compatible with Password Synchronization 1.0. This means Password Synchronization can continue to function as it did before, allowing you to upgrade to Identity Manager Password Synchronization at your convenience.

## 9.3 New Driver Configuration and Identity Manager Password Synchronization

If you are not using Password Synchronization 1.0, and you are creating a new driver or replacing an existing driver's configuration with the Identity Manager configuration, follow the instructions in [“Creating and Configuring a Driver”](#) in the *Novell Identity Manager 3.5.1 Administration Guide*.

In addition, do the following:

- ♦ Install new Password Synchronization filters and configure them if you want the connected system to provide user passwords to Identity Manager. See [Section 9.5, “Setting Up Password Synchronization Filters,”](#) on page 80.
- ♦ Set up the scenario for Password Synchronization that you want to use, using the Password Policies and the Password Synchronization settings for the driver. See [“Implementing Password Synchronization”](#) in the *Novell Identity Manager 3.5.1 Administration Guide*.

## 9.4 Upgrading Existing Driver Configurations to Support Identity Manager Password Synchronization

This section explains the process for adding support for Identity Manager Password Synchronization to existing driver configurations.

---

**IMPORTANT:** If a driver is being used with Password Synchronization 1.0, you should complete this section only as part of [Section 9.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on page 73, not alone.

---

The following is an overview of the tasks you must complete, using the procedure in this section:

- ◆ Add driver manifest, global configuration values, and password synchronization policies to the driver configuration. For a list of the policies you add, see [“Policies Required in the Driver Configuration”](#) in the *Novell Identity Manager 3.5.1 Administration Guide*.
- ◆ Change the Filter to allow the `nspmDistributionPassword` attribute to be synchronized.

### Prerequisites

- ❑ Make sure you have converted your existing driver to Identity Manager format, as described in [“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*.
- ❑ Create a backup of your existing driver by exporting the driver.
- ❑ Make sure you have installed the new driver shim. Some password synchronization features such as Check Password Status won’t work without the Identity Manager driver shim.

### Procedure

- 1 In iManager, click *Identity Manager Utilities > Import Drivers*.

The Import Driver Wizard opens.

- 2 Select the driver set where your existing driver resides.
- 3 In the list of driver configurations that appears, select *Password Synchronization 2.0 Policies* (it is listed under *Additional Policies*), then click *Next*.

A list of import prompts appears.

- 4 Select your existing driver to update.
- 5 Answer three prompts about the capabilities of the driver and the connected system.
  - ◆ Whether the connected system can provide passwords to Identity Manager.
  - ◆ Whether the connected system can accept passwords from Identity Manager
  - ◆ Whether the connected system can check a password to see if it matches the password in Identity Manager.

If you are uncertain which answers to give, check the settings that are provided with the Identity Manager sample configurations for your driver type. You could also create a temporary driver with the Identity Manager driver configurations, and view the settings in the driver manifest for that driver.

- 6 Click *Next*, then select to update everything about the driver.

This option gives you the driver manifest, global configuration values (GCVs), and password policies necessary for password synchronization.

The driver manifest and GCVs overwrite any values that already exist, but because these kinds of driver parameters are new in Identity Manager, there should be no existing values to overwrite.

The password policies don't overwrite any existing policy objects; they are simply added to the driver object.

---

**NOTE:** If you do have driver manifest or GCV values that you want to save, choose the option named *Update only Selected Policies* for that driver, and select the check boxes for all the policies. This option imports the password policies but does not change the driver manifest or GCVs.

---

- 7 Click *Next*, then click *Finish* to complete the wizard.

At this point, the new policies have been created as policy objects under the driver object, but are not yet part of the driver configuration. To link them in, you must manually insert each of them at the right point in the driver configuration on the Subscriber and Publisher channels.

- 8 Insert each of the new policies into the correct place in your existing driver configuration. If there are multiple policies in a policy set, make sure these password synchronization policies are listed last.

The list of the policies and where to insert them is in “[Policies Required in the Driver Configuration](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

Use the following procedure. Repeat these steps for each policy.

- 8a Click *Identity Manager > Identity Manager Overview*. Select the driver set for the driver you are updating.

- 8b Click the driver you just updated.

A page opens showing a graphical representation of the driver configuration.

- 8c Click the icon for the place where you need to add one of the new policies.

- 8d Click *Insert* to add the new policy. In the Insert page that appears, click *Use an Existing Policy*, browse for and select the new policy object, then click *OK*.

- 8e If you have more than one policy in the list for any of the new policies, use the arrow buttons to move the new policies to the correct location in the list. Make sure the policies are in the order listed in “[Policies Required in the Driver Configuration](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

- 9 Change the filter for the driver to allow the `nspmDistributionPassword` attribute to be synchronized.

- 10 Install new Password Synchronization filters and configure them if you want the connected system to provide user passwords to Identity Manager. See [Section 9.5, “Setting Up Password Synchronization Filters,”](#) on page 80.

At this point, the driver has the new driver shim, Identity Manager format, and the other pieces that are necessary to support password synchronization: driver manifest, GCVs, password synchronization policies, and filters. Now you can specify how you want passwords to flow to and from connected systems, using the Password Synchronization interface in iManager.

- 11 Set up the scenario for Password Synchronization that you want to use, using the Password Policies and the Password Synchronization settings for the driver. See “[Implementing Password Synchronization](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

- 12 Repeat this procedure for all the drivers that you want to participate in password synchronization.

## 9.5 Setting Up Password Synchronization Filters

The driver needs to be configured to run on only one Windows machine. However, after you install the driver, each of the other domain controllers needs a password filter (`pwfilter.dll` file) installed and the registry configured to capture passwords so that passwords can be sent to Identity Manager.

The password filter is automatically started when the domain controller is started. The filter captures password changes made by users through Windows clients, encrypts them, and sends them to the driver to update the Identity Manager data store.

---

**NOTE:** For information about configuring Password Synchronization, see “[Implementing Password Synchronization](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

---

To simplify your setup and administration of password filters, an Identity Manager PassSync utility is added to the Control Panel when the driver is installed. This utility gives you two choices for setting up the password filters, depending on whether you want to allow remote access to the registry on your domain controllers:

- ♦ **If you don’t allow remote access to the registry:** Set up the password filters on each domain controller separately. To do this, you go to each domain controller, install the driver files so you have the Identity Manager PassSync utility, and use the utility on each machine to install the password filter and update the registry.

See [Section 9.5.1, “Separately Configuring Password Filters on Each Domain Controller,”](#) on page 81.

- ♦ **If you allow remote access to the registry:** From the single machine where you plan to run the driver, configure the password filter for all the domain controllers, using the Identity Manager PassSync utility.

This method lets you configure all the domain controllers from one place.

If you configure all the domain controllers from one machine, the Identity Manager PassSync utility provides the following features to help you during setup:

- ♦ Lets you specify which domain you want to participate in password synchronization.
- ♦ Automatically discovers all the domain controllers for the domain.
- ♦ Lets you remotely install the `pwfilter.dll` on each domain controller.
- ♦ Automatically updates the registry on the machine where the driver is running and on each domain controller.
- ♦ Lets you view the status of the filter on each domain controller.
- ♦ Lets you remotely reboot a domain controller. This is necessary when you first add a domain for password synchronization, because the filter that captures password changes is a `.dll` file that starts when the domain controller is started.

See [Section 9.5.2, “Configuring Password Filters for All Domain Controllers from One Machine,”](#) on page 83.



## 9.5.1 Separately Configuring Password Filters on Each Domain Controller

This procedure explains how to install and configure the password filter on each domain controller, one at a time.

Use this method if you don't want to allow remote access to the registry.

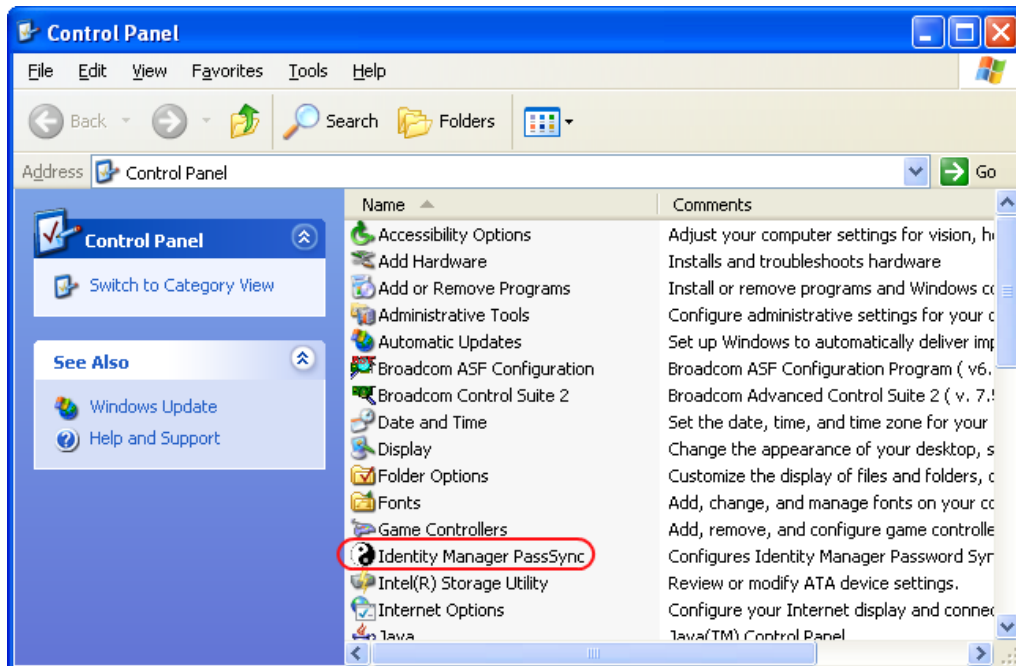
In this procedure, you install the driver so that you have the Identity Manager PassSync utility, then you use the utility to install the `pwfilter.dll` file, specify the port to use, and specify which host machine is running the Identity Manager Driver for NT.

Setting up the filter requires rebooting the domain controller, so you might want to perform this procedure after hours, or reboot only one domain controller at a time. If there is more than one domain controller in the domain, keep in mind that each domain controller where you want Password Synchronization to function must have the filter installed and must be rebooted.

- 1 Confirm that the following ports are available on both the domain controller and the machine where the Identity Manager Driver for NT is configured to run:
  - ♦ 135: The RPC endpoint mapper
  - ♦ 137: NetBIOS name service
  - ♦ 138: NetBIOS datagram service
  - ♦ 139: NetBIOS session service
- 2 On the domain controller, use the Identity Manager Installation to install only the Identity Manager Driver for NT.

Installing the driver installs the Identity Manager PassSync utility.

- 3 Click *Start > Settings > Control Panel*.



- 4 Double-click *Identity Manager PassSync*.

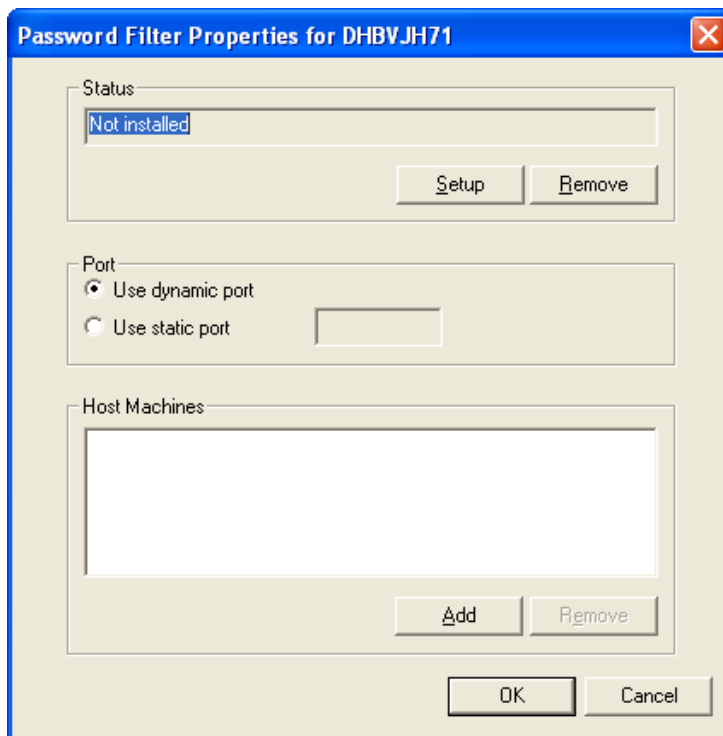
The first time you open the utility, it asks whether this is the machine where the Identity Manager driver is installed.



**5** Click *No*.

After you complete the configuration, you are not shown this prompt again unless you remove the password filter using the Remove button in the Password Filter Properties dialog box.

After you click *No*, the Password Filter Properties dialog box appears, with a status message indicating that the password filter is not yet set up on this domain controller.

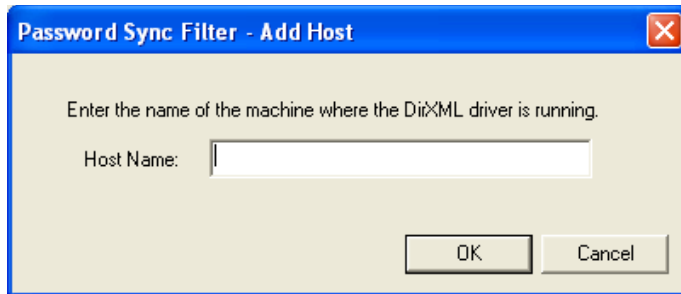


**6** Click the *Setup* button to install the password filter, `pwfilter.dll`.

**7** For the *Port* setting, specify whether to use a dynamic port or a static port.

Use the static port option only if you have decided to configure your remote procedure call (RPC) for the domain controller differently than the default.

**8** Specify the location of the Identity Manager driver, click the *Add* button, then specify the *Host Name* of the machine that is running the Identity Manager driver in the Password Sync Filter - Add Host dialog box. Click *OK*.



This step is necessary so that the password filter knows where to send the password changes. The password filter captures password changes, and must send them to the Identity Manager driver to update the Identity Manager data store.

- 9 In the Password Filter Properties dialog box, click *OK*.
- 10 Reboot the domain controller to complete the installation of the password filter.

You can choose to reboot at a time that makes sense for your environment. Just keep in mind that password synchronization won't be fully functional until every domain controller has the password filter installed and has been rebooted.

After the installation is complete and the domain controller is rebooted, the password filter is loaded automatically whenever the domain controller starts up.
- 11 Check the status for the password filter again by clicking *Start > Settings > Control Panel*, and double-clicking the Identity Manager PassSync utility. Confirm that the status says *Running*.
- 12 Repeat **Step 2** through **Step 11** for each domain controller that you want to participate in Password Synchronization.
- 13 When the status says *Running* for all the domain controllers, test Password Synchronization to confirm that it is working.

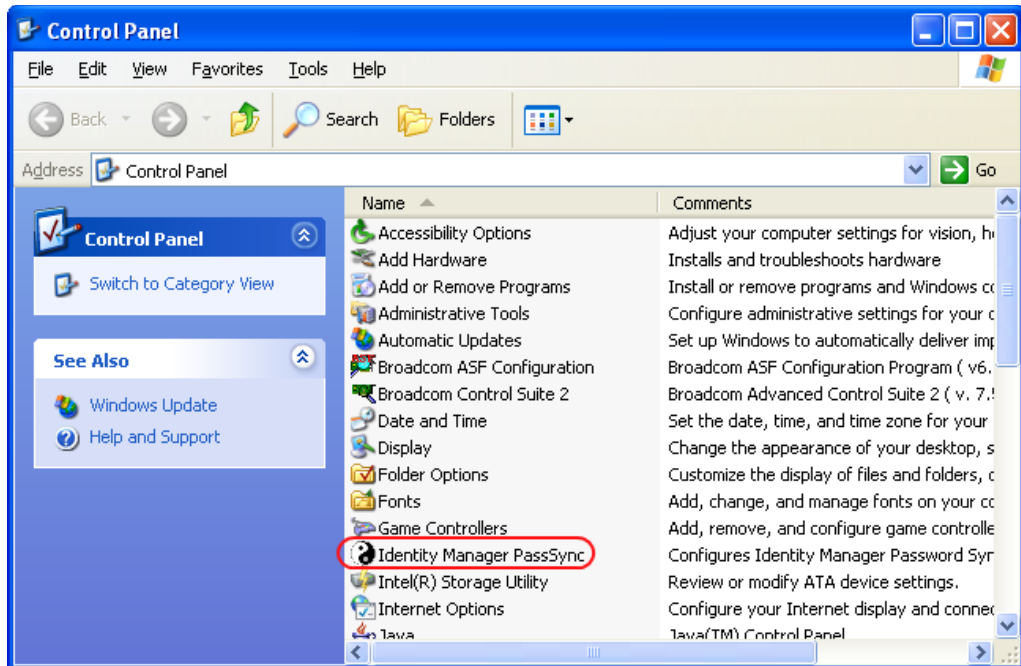
## 9.5.2 Configuring Password Filters for All Domain Controllers from One Machine

This procedure explains how to install and configure the password filter on each domain controller, all from the same machine where you are running the driver.

Use this method if you allow remote access to the registry.

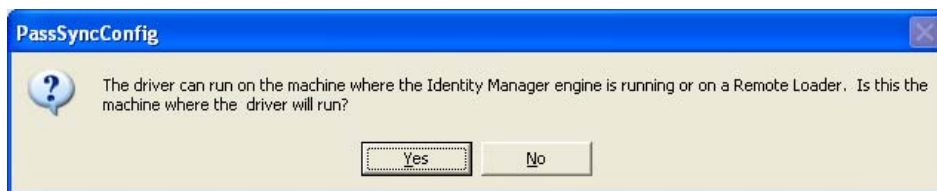
Setting up the filter requires rebooting the domain controller, so you might want to perform this procedure after hours, or reboot only one domain controller at a time. If there is more than one domain controller in the domain, keep in mind that each domain controller where you want Password Synchronization to function must have the filter installed and must be rebooted.

- 1 Confirm that these ports are available on the domain controllers and on the machine where the Identity Manager Driver for NT is configured to run:
  - ♦ 135: The RPC endpoint mapper
  - ♦ 137: NetBIOS name service
  - ♦ 138: NetBIOS datagram service
  - ♦ 139: NetBIOS session service
- 2 At the computer where the driver is installed, click *Start > Settings > Control Panel*.



**3** Double-click *Identity Manager PassSync*.

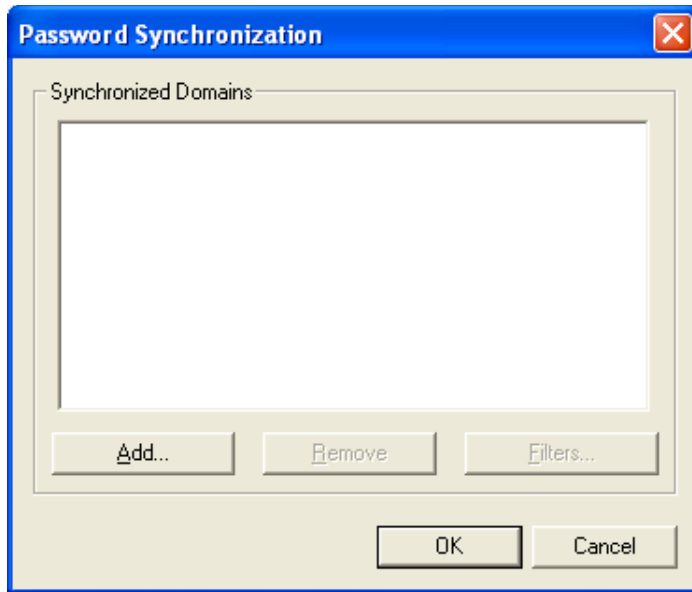
The first time you open the utility, it asks whether this is the machine where the Identity Manager driver is installed.



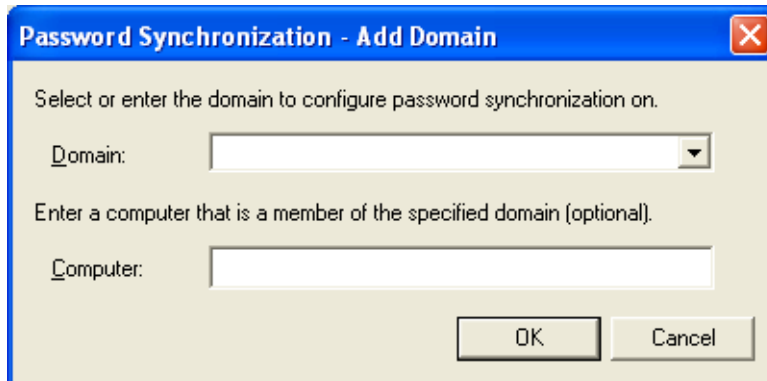
After you complete the configuration, you are not shown this prompt again unless you remove this domain from the list.

**4** Click *Yes*.

A list appears, labeled *Synchronized Domains*.



- 5 To add a domain you want to participate in password synchronization, click *Add* and specify the domain name.



- 6 Log in with administrator rights.

The Identity Manager PassSync utility discovers all the domain controllers for that domain, and installs `pwfilter.dll` on each domain controller. It also updates the registry on the computer where you are running the drivers, and on each domain controller. This might take a few minutes.

The `pwfilter.dll` doesn't capture password changes until the domain controller has been rebooted. The Identity Manager PassSync utility lets you see a list of all the domain controllers and the status of the filter on them. It also lets you reboot the domain controller from inside the utility.

- 7 Click the name of the domain in the list, then click *Filters*.

The utility displays the names of all the domain controllers and the status of the filter on each of them.

The status for each domain controller should indicate that it needs rebooting. However, it might take a few minutes for the utility to complete its automated task, and in the meantime the status might say Unknown.

**8** Reboot each domain controller.

You can choose to reboot them at a time that makes sense for your environment. Just keep in mind that password synchronization won't be fully functional until every domain controller has been rebooted.

**9** When the status for the domain controllers says Running, test password synchronization to confirm that it is working.

**10** To add more domains, click *OK* to return to the list of domains, and repeat **Step 5** through **Step 9**.

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

- ♦ [Section 10.1, “Error Messages,” on page 87](#)
- ♦ [Section 10.2, “Troubleshooting Password Synchronization,” on page 95](#)
- ♦ [Section 10.3, “Troubleshooting Driver Processes,” on page 95](#)

## 10.1 Error Messages

The following section identifies common error messages and the possible causes.

### **Error Wrong Destination DN**

Source: The status log or DSTrace screen.

Explanation: The destination DN sent to the driver was wrong or not present.

Possible Cause: This can occur when a User object was changed in a container not covered in the Subscriber Placement policy.

Action: Look at changing the Placement policy to include the correct containers.

### **Error Password Length is too long. Password not set**

Source: The status log or DSTrace screen.

Explanation: The password sent to the driver was too long and the driver was unable to set the password.

Action: Shorten the password.

### **Error Failed to attach to the registry = error #**

Source: The status log or DSTrace screen.

Explanation: The driver was unable to attach to the system registry. The error was fatal, so the driver will shut down.

Possible Cause: Check the error code to see possible causes.

### **Error Failed to attach to the registry retrying = error #**

Source: The status log or DSTrace log.

Explanation: The driver was unable to attach to the system registry but the error suggested to try again later.

Possible Cause: Check the error number to see possible causes.

**Error Unable to logon as User %S to Domain %S error code = error #**

Source: The status log or DSTrace screen.

Explanation: The driver was unable to log in as the user in the domain specified.

Possible Cause: Check the error code to see possible causes.

**Error: Missing Poll Rate parameter**

Source: The status log or DSTrace screen.

Explanation: The poll rate in the driver parameters has not been set.

Action: Add the poll rate to the driver parameters.

**Error: Missing Publisher State parameter**

Source: The status log or DSTrace screen.

Explanation: This is the first time the driver has been run.

**Returning an error to DS**

Source: The status log or DSTrace screen.

Explanation: An error has occurred and the driver is returning the error.

**LogonUser = error #**

Source: The status log or DSTrace screen.

Explanation: The driver has tried to log in as the user specified in the driver parameters.

Possible Cause: Check the error number to see possible reasons why logon failed.

**ImpersonateLoggedOnUser = error #**

Source: The status log or DSTrace screen.

Explanation: The driver has tried to impersonate the user.

Possible Cause: Check the error number to see possible reason why the impersonation failed.

**Failed MKDIR directory path = error #**

Source: The status log or DSTrace screen.

Explanation: The driver attempted to create a directory. MKDIR failed to create a directory path and returned the error #.

Possible Cause: Check the error number for the reason for the failure.

**Failed SharDir directory path**

Source: The status log or DSTrace screen.

Explanation: The driver attempted to share the directory path with the Everyone account but failed.



**ERROR ADD failed, NERR\_PasswordTooShort**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: The password is too short

**ERROR ADD failed, NERR\_InvalidComputer**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: The computer is invalid.

**ERROR ADD failed, NERR\_ERROR\_ACCESS\_DENIED**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: Access is denied.

**ERROR ADD failed, NERR\_NotPrimary**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: This is not the primary domain controller.

**ERROR ADD failed, NERR\_GroupExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: The group already exists.

**ERROR ADD failed, NERR\_UserExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: The user already exists.

**ERROR ADD failed, NERR\_ServiceCtlBusy**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: The service is busy.

**ERROR ADD failed, ERROR\_INVALID\_PARAMETER**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a user to the domain.

Possible Cause: There is an invalid parameter on the driver.

**ERROR ADD failed. = error #, username**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to add a username to the domain.

Possible Cause: It failed because of the error number stated.

**ERROR MOIDY failed, NERR\_PasswordTooShort**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: The password is too short

**ERROR MODIFY failed, NERR\_InvalidComputer**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: The computer is invalid.

**ERROR MODIFY failed, NERR\_ERROR\_ACCESS\_DENIED**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: Access is denied.

**ERROR MODIFY failed, NERR\_NotPrimary**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: This is not the primary domain controller.

**ERROR MODIFY failed, NERR\_GroupExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: The group already exists.

**ERROR MODIFY failed, NERR\_UserExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: The user already exists.

**ERROR MODIFY failed, NERR\_ServiceCtlBusy**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: The service is busy.

**ERROR MODIFY failed, ERROR\_INVALID\_PARAMETER**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a user in the domain.

Possible Cause: There is an invalid parameter on the driver.

**ERROR MODIFY failed. = error #, username**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to modify a username in the domain.

Possible Cause: It failed because of the error number stated.

**ERROR RENAME failed, NERR\_PasswordTooShort**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: The password is too short

**ERROR RENAME failed, NERR\_InvalidComputer**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: The computer is invalid.

**ERROR RENAME failed, NERR\_ERROR\_ACCESS\_DENIED**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: Access is denied.

**ERROR RENAME failed, NERR\_NotPrimary**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: This is not the primary domain controller.

**ERROR RENAME failed, NERR\_GroupExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: The group already exists.

**ERROR RENAME failed, NERR\_UserExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: The user already exists.

**ERROR RENAME failed, NERR\_ServiceCtlBusy**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: The service is busy.

**ERROR RENAME failed, ERROR\_INVALID\_PARAMETER**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a user in the domain.

Possible Cause: There is an invalid parameter on the driver.

**ERROR RENAME failed. = error #, username**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to rename a username in the domain.

Possible Cause: It failed because of the error number stated.

**ERROR GETINFO failed, NERR\_UserNotFound**

Source: The status log or DSTrace screen.

Explanation: A query was requested and failed because the user was not found.

**ERROR GETINFO failed, ERROR\_ACCESS\_DENIED**

Source: The status log or DSTrace screen.

Explanation: A query was requested and failed because access was denied.

**ERROR GETINFO failed, NERR\_InvalidComputer**

Source: The status log or DSTrace screen.

Explanation: A query was requested and failed because the computer is invalid.

**ERROR GETINFO failed**

Source: The status log or DSTrace screen.

Explanation: A query was requested and failed.

**ERROR DELETE failed, NERR\_PasswordTooShort**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: The password is too short

**ERROR DELETE failed, NERR\_InvalidComputer**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: The computer is invalid.

**ERROR DELETE failed, NERR\_ERROR\_ACCESS\_DENIED**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: Access is denied.

**ERROR DELETE failed, NERR\_NotPrimary**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: This is not the primary domain controller.

**ERROR DELETE failed, NERR\_GroupExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: The group already exists.

**ERROR DELETE failed, NERR\_UserExists**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: The user already exists.

**ERROR DELETE failed, NERR\_ServiceCtlBusy**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: The service is busy.

**ERROR DELETE failed, ERROR\_INVALID\_PARAMETER**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user in the domain.

Possible Cause: There is an invalid parameter on the driver.

**ERROR DELETE failed**

Source: The status log or DSTrace screen.

Explanation: The Subscriber has attempted to delete a user from the domain.

Possible Cause: Check the error message for a possible cause.

**HeapReAlloc error!**

Source: The status log or DSTrace screen.

Explanation: Not enough memory.

Action: Allocate more memory to Java.

Action: Add more memory to the server.

#### **LookupAccountName error! error #**

Source: The status log or DSTrace screen.

Explanation: LookupAccountName was not successful because of error #.

Possible Cause: Check the error number for a possible cause.

#### **SetSecurityDescriptorDacl error! error #**

Source: The status log or DSTrace screen.

Explanation: SetSecurityDescriptorDacl was not successful because of error #.

Possible Cause: Check the error number for a possible cause.

#### **NetShareAdd error! error #**

Source: The status key or DSTrace screen.

Explanation: NetShareAdd was not successful because of error #.

Possible Cause: Check the error number for a possible cause.

Action: What can be done to resolve the problem.

#### **Publisher Error NO MEMORY**

Source: The status log or DSTrace screen.

Explanation: The Publisher channel ran out of memory.

Action: Allocate more memory to Java.

Action: Add more memory to the server.

#### **Error out of memory**

Source: The status log or DSTrace screen.

Explanation: The Publisher channel ran out of memory.

Action: Allocate more memory to Java.

Action: Add more memory to the server.

#### **Unable to process Nt4 User data**

Source: The status log or DSTrace screen.

Explanation: This error occurs when the Subscriber channel was unable to complete a request to the NT domain.

## 10.2 Troubleshooting Password Synchronization

- ♦ If you see an error about a password not complying when a user is initially created, but the password is set correctly in eDirectory, this might be an issue with the default password in the driver policy not conforming to the Password policy that applies to that user.

For example, perhaps you want the NT driver to provide the initial password for a user when it creates a new user object in eDirectory to match a user in NT. The sample configuration for the NT driver sends the initial password as a separate operation from adding the user, and the sample configuration also includes a policy that provides a default password for a user, based on the user's surname, if no password is provided by NT. Because adding the user and setting the password are done separately, in this case a new user always receives the default password, even if only momentarily, and it is soon updated because the NT driver sends the password immediately after adding the user. If the default password does not comply with the eDirectory Password Policy for the user, an error is displayed. For example, if a default password created from the user's surname is too short to comply with the Password policy, you might see a -216 error saying password is too short. However, the situation is soon rectified if the NT driver then sends an initial password that does comply.

Regardless of the driver you are using, if you want a connected system that is creating user objects to provide the initial password, consider doing one of the actions in the list below. These measures are especially important if the initial password does not come with the Add event and instead comes in a subsequent event.

- ♦ Change the policy on the Publisher channel that creates the default password, so that the default password conforms to the Password policies (created through *Password Management > Manage Password Policies*) that have been defined for your organization in eDirectory. When the initial password comes from the authoritative application, it replaces the default password.

This option is preferable because we recommend that a default password policy exists in order to maintain a high level of security within the system.

or

- ♦ Remove the policy on the Publisher channel that creates default password. In the sample configuration, this policy is provided in the Command Transformation policy set. Adding a user without a password is allowed in eDirectory. The assumption for this option is that the password for the newly created user object eventually comes through the Publisher channel, so the user object exists without a password only for a short time.

## 10.3 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.

### 10.3.1 Viewing Driver Processes

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

- ♦ [“Adding Trace Levels in Designer” on page 96](#)

- ◆ “Adding Trace Levels in iManager” on page 97
- ◆ “Capturing Driver Processes to a File” on page 98

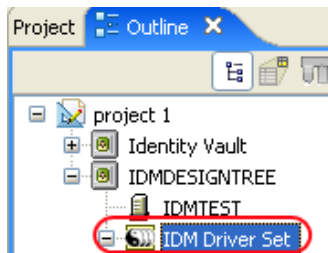
## Adding Trace Levels in Designer

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

- ◆ “Driver Set” on page 96
- ◆ “Driver” on page 97

### Driver Set

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



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

Parameter	Description
Driver trace level	As the driver object trace level increases, the amount of information displayed in DSTrace increases.  Trace level 1 shows errors, but not the cause of the errors. If you want to see password synchronization information, set the trace level to 5.
XSL trace level	DSTrace displays XSL events. Set this trace level only when troubleshooting XSL style sheets. If you do not want to see XSL information, set the level to zero.
Java debug port	Allows developers to attach a Java* debugger.
Java trace file	When a value is set in this field, all Java information for the driver set object is written to a file. The value for this field is the path for that file.  As long as the file is specified, Java information is written to this file. If you do not need to debug Java, leave this field blank.
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 there is no disk space left.
	<b>NOTE:</b> The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.

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

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

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

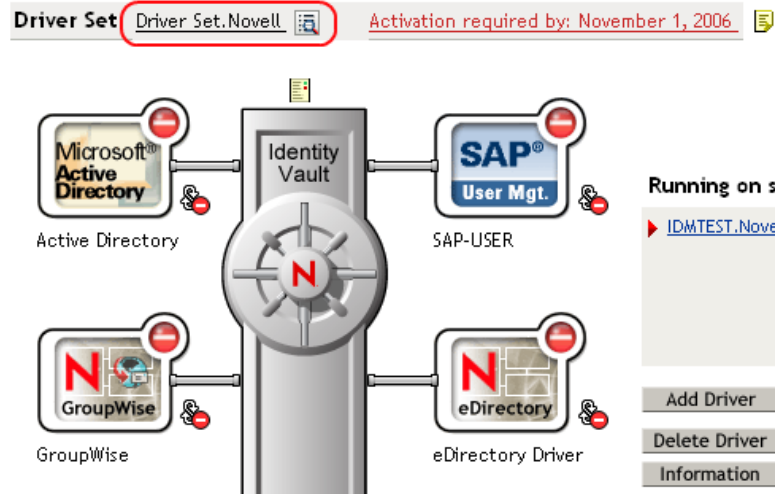
## Adding Trace Levels in iManager

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

- ♦ [“Driver Set” on page 97](#)
- ♦ [“Driver” on page 98](#)

## Driver Set

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



- 4 Select the *Misc* tab for the driver set object.
- 5 Set the parameters for tracing, then click *OK*.  
See “[Driver trace level](#)” on page 96 for the parameters.

#### Driver

- 1 In iManager, select *Identity Manager > Identity Manager Overview*.
- 2 Browse to the driver set object where the driver object resides, then click *Search*.
- 3 Click the upper right corner of the driver object, then click *Edit properties*.
- 4 Select the *Misc* tab for the driver object.
- 5 Set the parameters for tracing, then click *OK*.  
See “[Trace level](#)” on page 97 for the parameters.

The option *Use setting from Driver Set* does not exist in iManager.

#### Capturing Driver Processes to a File

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

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

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

- ◆ “[NetWare](#)” on page 99
- ◆ “[Windows](#)” on page 99
- ◆ “[UNIX](#)” on page 99
- ◆ “[iMonitor](#)” on page 100
- ◆ “[Remote Loader](#)” on page 100

## 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* > *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 DTrace information from a Web browser. It does not matter where Identity Manager is running. The following files run iMonitor:

- ♦ `ndsimon.nlm` runs on NetWare<sup>®</sup>.
- ♦ `ndsimon.dlm` runs on Windows.
- ♦ `ndsimonitor` runs on UNIX\*.

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

Port 8008 is the default.

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

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 by 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 switches in [Table 10-1](#). For more information, see “[Configuring the Remote Loader](#)” in the *Novell Identity Manager 3.5.1 Administration Guide*.

**Table 10-1** *Command Line Tracing Switches*

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



# Security: Best Practices

# 11

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





# DirXML Command Line Utility

# A

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

The utility and scripts are installed on all platforms during the Identity Manager installation. The utility is installed to the following locations:

- ♦ Windows: \Novell\Nds\dxcmd.bat
- ♦ NetWare<sup>®</sup>: sys:\system\dxcmd.ncf
- ♦ UNIX: /usr/bin/dxcmd

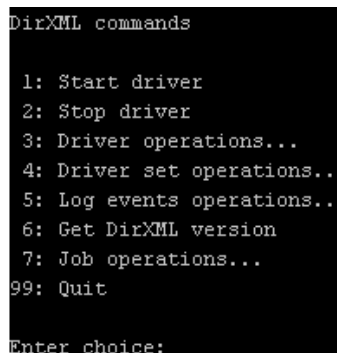
There are two different methods for using the DirXML Command Line utility:

- ♦ [Section A.1, “Interactive Mode,” on page 105](#)
- ♦ [Section A.2, “Command Line Mode,” on page 114](#)

## A.1 Interactive Mode

The interactive mode provides a text interface to control and use the DirXML Command Line utility.

- 1 At the console, enter dxcmd.
- 2 Enter the name of a user with sufficient rights to the Identity Manager objects, such as admin.novell.
- 3 Enter the user’s password.



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

- 4 Enter the number of the command you want to perform.  
[Table A-1 on page 106](#) contains the list of options and what functionality is available.
- 5 Enter 99 to quit the utility.

**NOTE:** If you are running eDirectory<sup>™</sup> 8.8 on UNIX or Linux, you must specify the `-host` and `-port` parameters. For example, `dxcmd -host 10.0.0.1 -port 524`. If the parameters are not specified, a jclient error occurs:

```
novell.jclient.JCException: connect (to address) 111 UNKNOWN ERROR
```

By default, eDirectory 8.8 is not listening to localhost. The DirXML Command Line utility needs to resolve the server IP address or hostname and the port to be able to authenticate.

**Table A-1** *Interactive Mode Options*

Option	Description
1: <i>Start Driver</i>	Starts the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to start the driver.
2: <i>Stop Driver</i>	Stops the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to stop the driver.
3: <i>Driver operations</i>	Lists the operations available for the driver. If there is more than one driver, each driver is listed with a number. Enter the number of the driver to see the operations available. See <a href="#">Table A-2 on page 107</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 111</a> for a description of these options.
6: <i>Get DirXML version</i>	Lists the version of the Identity Manager installed.
7: <i>Job operations</i>	Manages jobs created for Identity Manager.
99: <i>Quit</i>	Exits the DirXML Command Line utility

**Figure A-1** *Driver Options*

```
Select a driver operation for:
Active Directory.Driver Set.Novell.IDMDESIGNTREE.

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

Enter choice:
```

**Table A-2** *Driver Options*

Options	Description
1: <i>Start driver</i>	Starts the driver.
2: <i>Stop driver</i>	Stops the driver.
3: <i>Get driver state</i>	Lists the state of the driver. <ul style="list-style-type: none"><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 Yes, specify the date and time you want the resynchronization to occur: <i>Enter a date/time (format 9/27/05 3:27 PM)</i>.</p> <p>If you enter No, the resynchronization occurs immediately.</p>
7: <i>Migrate from application into DirXML</i>	<p>Processes an XML document that contains a query command: <i>Enter filename of XDS query document:</i></p> <p>Create the XML document that contains a query command by using the <a href="http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsstdtd/query.html">Novell nds.dtd (http://developer.novell.com/ndk/doc/dirxml/dirxmlbk/ref/ndsstdtd/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 and event to the driver queue</p> <p><i>Enter filename of XDS event document:</i></p> <p>Examples:</p> <p>NetWare: <code>sys:\files\add.xml</code></p> <p>Windows: <code>c:\files\add.xml</code></p> <p>Linux: <code>/files/add.xml</code></p>
11: <i>Check object password</i>	<p>Validates that an object's password in the connected system is associated with a driver. It matches the object's eDirectory password (Distribution Password, used with Universal Password).</p> <p><i>Enter user name:</i></p>
12: <i>Initialize new driver object</i>	<p>Performs an internal initialization of data on a new Driver object. This is only for testing purposes.</p>
13: <i>Password operations</i>	<p>There are nine Password options. See <a href="#">Table A-3 on page 109</a> for a description of these options.</p>
14: <i>Cache operations</i>	<p>There are five Cache operations. See <a href="#">Table A-4 on page 110</a> for a description of these options.</p>

Options	Description
99: <i>Exit</i>	Exits the driver options.

**Figure A-2** Password Operations

```
Select a password operation

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

Enter choice:
```

**Table A-3** Password Operations

Operation	Description
1: <i>Set shim password</i>	Sets the application password. This is the password of the user account you are using to authenticate into the connected system with.
2: <i>Clear shim password</i>	Clears the application password.
3: <i>Set Remote Loader password</i>	The Remote Loader password is used to control access to the Remote Loader instance.  Enter the Remote Loader password, then confirm the password by typing it again.
4: <i>Clear Remote Loader password</i>	Clears the Remote Loader password so no Remote Loader password is set on the Driver object.
5: <i>Set named password</i>	Allows you to store a password or other pieces of security information on the driver. See <a href="#">Section 6.6, "Storing Driver Passwords Securely with Named Passwords,"</a> on page 48 for more information.  There are four prompts to fill in: <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.</p> <p><i>Do you want to clear all named passwords? (yes/no).</i></p> <p>If you enter Yes, all Named Passwords are cleared. If you enter No, you are prompted to specify the password name that you want to clear.</p>
7: <i>List named passwords</i>	<p>Lists all Named Passwords that are stored on the driver object. It lists the password name and the password description.</p>
8: <i>Get password state</i>	<p>Lists if a password is set for:</p> <ul style="list-style-type: none"> <li>◆ Driver Object password</li> <li>◆ Application password</li> <li>◆ Remote loader password</li> </ul> <p>The dxcmd utility allows you to set the Application password and the Remote Loader password. You cannot set the Driver Object password with this utility. It shows if the password has been set or not.</p>
99: <i>Exit</i>	<p>Exits the current menu and takes you back to the Driver options.</p>

**Figure A-3** *Cache Operations*

```

Enter choice: 14

Select a cache operation

1: Get driver cache limit
2: Set driver cache limit
3: View cached transactions
4: Delete cached transactions
99: Exit

Enter choice:

```

**Table A-4** *Cache Operations*

Operation	Description
1: <i>Get driver cache limit</i>	<p>Displays the current cache limit that is set for the driver.</p>
2: <i>Set driver cache limit</i>	<p>Sets the driver cache limit in kilobytes. A value of 0 is unlimited.</p>

Operation	Description
3: <i>View cached transactions</i>	<p>A text file is created with the events that are stored in cache. You can select the number of transactions to view.</p> <ul style="list-style-type: none"> <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. There are 49 items you can select to log. See <a href="#">Table A-6 on page 112</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 of the log event options.
3: <i>Set driver log events</i>	<p>Allows you to log driver events through Novell Audit. There are 49 items to select to log. See <a href="#">Table A-6 on page 112</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*

---

**Options**

---

- 1: Status success
  - 2: Status retry
  - 3: Status warning
  - 4: Status error
  - 5: Status fatal
  - 6: Status other
  - 7: Query elements
  - 8: Add elements
  - 9: Remove elements
  - 10: Modify elements
  - 11: Rename elements
  - 12: Move elements
  - 13: Add-association elements
  - 14: Remove-association elements
  - 15: Query-schema elements
  - 16: Check-password elements
  - 17: Check-object-password elements
  - 18: Modify-password elements
  - 19: Sync elements
  - 20: Pre-transformed XDS document from shim
  - 21: Post input transformation XDS document
  - 22: Post output transformation XDS document
  - 23: Post event transformation XDS document
  - 24: Post placement transformation XDS document
  - 25: Post create transformation XDS document
  - 26: Post mapping transformation <inbound> XDS document
  - 27: Post mapping transformation <outbound> XDS document
-



---

**Options**

---

- 28: Post matching transformation XDS document
  - 29: Post command transformation XDS document
  - 30: Post-filtered XDS document <Publisher>
  - 31: User agent XDS command document
  - 32: Driver resync request
  - 33: Driver migrate from application
  - 34: Driver start
  - 35: Driver stop
  - 36: Password sync
  - 37: Password request
  - 38: Engine error
  - 39: Engine warning
  - 40: Add attribute
  - 41: Clear attribute
  - 42: Add value
  - 43: Remove value
  - 44: Merge entire
  - 45: Get named password
  - 46: Reset Attributes
  - 47: Add Value - Add Entry
  - 48: Set SSO Credential
  - 49: Clear SSO Credential
  - 50: Set SSO Passphrase
  - 51: User defined IDs
  - 99: Accept checked items
-

**Table A-7** Enter Table Title Here

Options	Description
1: <i>Get available job definitions</i>	Allows you to select an existing job.  <i>Enter the job number:</i>  <i>Do you want to filter the job definitions by containment? Enter Yes or No</i>  <i>Enter name of the file for response:</i>  Examples:  NetWare: <code>sys:\files\user.log</code>  Windows: <code>c:\files\user.log</code>  Linux: <code>/files/user.log</code>
2: <i>Operations on specific job object</i>	Allows you to perform operations for a specific job.

## A.2 Command Line Mode

The command line mode allows you to use script or batch files. [Table A-8 on page 114](#) 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>	
<code>-user &lt;user name&gt;</code>	Specify the name of a user with administrative rights to the drivers you want to test.
<code>-host &lt;name or IP address&gt;</code>	Specify the IP address of the server where the driver is installed.
<code>-password &lt;user password&gt;</code>	Specify the password of the user specified above.
<code>-port &lt;port number&gt;</code>	Specify a port number, if the default port is not used.
<code>-q &lt;quiet mode&gt;</code>	Displays very little information when a command is executed.
<code>-v &lt;verbose mode&gt;</code>	Displays detailed information when a command is executed.

Option	Description
-s <stdout>	Writes the results of the <code>dxcmd</code> command to <code>stdout</code> .
-? <show this message>	Displays the help menu.
-help <show this message>	Displays the help menu.
<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 gets processed after anything that might be in the cache at the time of the submission. The submission won't fail if the driver isn't running.</p>
-setlogevents <dn> <integer ...>	<p>Sets Novell Audit log events on the driver. The integer is the option of the item to log. See <a href="#">Table A-6 on page 112</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>	Aborts the specified job.
-getjobrunningstate <job dn>	Returns the specified job's running state.
-getjobenabledstate <job dn>	Returns the specified job's enabled state.
-getjobnextruntime <job dn>	Returns the specified job's next run time.
-updatejob <job dn>	Updates the specified job.
-clearallnamedpasswords <driver dn>	Clears all Named Passwords set on a specific driver.

If a command line is executed successfully, it returns a zero. If the command line returns anything other than zero, it is an error. For example 0 means success, and -641 means invalid operation. -641 is an eDirectory error code. [Table A-9 on page 117](#) contains other values for specific command line options.

**Table A-9** *Command Line Option Values*

Command Line Option	Values
-getstate	0- stopped 1- starting 2- running 3- shutting down 11- get schema Anything else that is returned is an error.
-getstartoption	0- disabled 1- manual 2- auto Anything else that is returned is an error.
-getcachelimit	0- unlimited Anything else that is returned is an error.
-getjobrunningstate	0- stopped 1- running Anything else that is returned is an error.
-getjobenabledstate	0- disabled 1- enabled 2- configuration error Anything else that is returned is an error.

---


Command Line Option	Values
-getjobnextruntime	Return is the next scheduled time for the job in eDirectory time format (number of seconds since 00:00:00 Jan 1, 1970 UTC).

---

# Properties of the Driver

# B

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

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

- ◆ [Section B.1, “Driver Configuration,” on page 119](#)
- ◆ [Section B.2, “Global Configuration Values,” on page 124](#)
- ◆ [Section B.3, “Named Passwords,” on page 125](#)
- ◆ [Section B.4, “Engine Control Values,” on page 126](#)
- ◆ [Section B.5, “Log Level,” on page 128](#)
- ◆ [Section B.6, “Driver Image,” on page 129](#)
- ◆ [Section B.7, “Security Equals,” on page 129](#)
- ◆ [Section B.8, “Filter,” on page 130](#)
- ◆ [Section B.9, “Edit Filter XML,” on page 130](#)
- ◆ [Section B.10, “Misc,” on page 131](#)
- ◆ [Section B.11, “Excluded Users,” on page 131](#)
- ◆ [Section B.12, “Driver Manifest,” on page 132](#)
- ◆ [Section B.13, “Driver Inspector,” on page 132](#)
- ◆ [Section B.14, “Driver Cache Inspector,” on page 133](#)
- ◆ [Section B.15, “Inspector,” on page 133](#)
- ◆ [Section B.16, “Server Variables,” on page 134](#)

## B.1 Driver Configuration

In iManager:

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

In Designer:

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

There are different sections under *Driver Configuration*. In this document, each section is listed in a table. The table contains a description of the fields, and the default value or an example of the value that should be specified in the field.

## B.1.1 Driver Module

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



In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Driver Module*.  
See [Table B-1](#) for a list of the driver module options.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Select the *Driver Module* tab.  
See [Table B-1](#) for a list of the driver module options.

**Table B-1** *Driver Modules*

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.  The NT driver always uses <i>Native</i> to connect to the NT Domain natively through <code>NtDomainShim.dll</code> .
<i>Connect to Remote Loader</i>	Used when the driver is connecting remotely to the connected system.
 <i>Remote Loader Client Configuration for Documentation</i>	 Includes the Remote Loader client configuration information in the driver documentation that is generated by Designer.

## B.1.2 Driver Object Password

In iManager:

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



See [Table B-2](#) for more information.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and click *Properties > Driver Configuration*.
- 2 Click *Driver Module > Connect to Remote Loader > Driver Object Password > Set Password*.  
See [Table B-2](#) for more information.

**Table B-2** *Driver Object Password*

Option	Description
<i>Driver Object Password</i>	Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page or the remote driver does not run. This password is used by the Remote Loader to authenticate itself to the remote driver shim.

### B.1.3 Authentication

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


In iManager:










- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Authentication*.  
See [Table B-3](#) for a list of the authentication options.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Click *Authentication*.  
See [Table B-3](#) for a list of the authentication options.

**Table B-3** *Authentication Options*

Option	Description
<i>Authentication ID</i>	Specify a user application ID. This ID is used to pass Identity Vault subscription information to the application.
or	
 <i>User ID</i>	Example: Administrator

Option	Description
<p><i>Authentication Context</i></p> <p>or</p> <p> <i>Connection Information</i></p>	Specify the IP address or name of the server the application shim should communicate with.
<p><i>Remote Loader Connection Parameters</i></p> <p>or</p> <p> <i>Host name</i></p> <p> <i>Port</i></p> <p> <i>KMO</i></p> <p> <i>Other parameters</i></p>	<p>Used only if the driver is connecting to the application through the Remote Loader. The parameter to enter is</p> <pre>hostname=xxx.xxx.xxx.xxx port=xxxx</pre> <p><i>kmo=certificatename</i>, when the host name is the IP address of the application server running the Remote Loader server and the port is the port the Remote Loader is listening on. The default port for the Remote Loader is 8090.</p> <p>The <i>kmo</i> entry is optional. It is only used when there is an SSL connection between the Remote Loader and the Metadirectory engine.</p> <p>Example: <code>hostname=10.0.0.1 port=8090</code>  <code>kmo=IDMCertificate</code></p>
<p><i>Driver Cache Limit (kilobytes)</i></p> <p>or</p> <p> <i>Cache limit (KB)</i></p>	<p>Specify the maximum event cache file size (in KB). If it is set to zero, the file size is unlimited.</p> <p> Click <i>Unlimited</i> to set the file size to unlimited in Designer.</p>
<p><i>Application Password</i></p> <p>or</p> <p> <i>Set Password</i></p>	Specify the password for the user object listed in the <i>Authentication ID</i> field.
<p><i>Remote Loader Password</i></p> <p>or</p> <p> <i>Set Password</i></p>	Used only if the driver is connecting to the application through the Remote Loader. The password is used to control access to the Remote Loader instance. It must be the same password specified during the configuration of the Remote Loader on the connected system.

## B.1.4 Startup Option

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


In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Startup Option*.  
See [Table B-4](#) for a list of the startup options.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Click *Startup Option*.  
See [Table B-4](#) for a list of the startup options.

**Table B-4** Startup Options

Option	Description
<i>Auto start</i>	The driver starts every time the Identity Manager server is started.
<i>Manual</i>	The driver does not start when the Identity Manager server is started. The driver must be started through Designer or iManager.
<i>Disabled</i>	The driver has a cache file that stores all of the events. When the driver is set to Disabled, this file is deleted and no new events are stored in the file until the driver state is changed to Manual or Auto Start.
 <i>Do not automatically synchronize the driver</i>	This option only applies if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it is started.

## B.1.5 Driver Parameters



In iManager:


- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Driver Configuration > Driver Parameters*.  
See [Table B-5](#) for a list of the driver parameters.

In Designer:

- 1 Open a project in the modeler, then right-click the driver line and select *Properties > Driver Configuration*.
- 2 Click *Driver Parameters*.  
See [Table B-5](#) for a list of the driver parameters.

**Table B-5** Driver Parameters

Option	Description
<i>Driver Settings &gt; Server Name of NT Domain</i>	The name of the NT Domain server. It must be in uppercase.
or	
 <i>Driver options &gt; Server Name of NT Domain</i>	
<i>Publisher Settings &gt; Poll Interval (milliseconds)</i>	How often the driver checks the NT Domain for events.
or	
 <i>Publisher options &gt; Poll Interval (milliseconds)</i>	

Option	Description
<i>Publisher Settings &gt; Password Sync Timeout (minutes)</i>	How long the driver tries to synchronize the password, if a problem occurs. After the default time of 10 minutes, the driver stops trying to synchronize the password change.
or	
 <i>Publisher options &gt; Password Sync Timeout (minutes)</i>	

## B.2 Global Configuration Values

Global configuration values (GCVs) allow you to specify settings for the Identity Manager features such as password synchronization and driver heartbeat, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own.

**IMPORTANT:** Password synchronization settings are GCVs, but it's best to edit them in the graphical interface provided on the Server Variables page for the driver, instead of the GCV page. The Server Variables page that shows Password Synchronization settings is accessible as a tab like other driver parameters, or by clicking *Password Management > Password Synchronization*, searching for the driver, and clicking the driver name. The page contains online help for each Password Synchronization setting.

In iManager:

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

In Designer:

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

There are global configuration values for the driver configuration and there are global configuration values for password synchronization, as listed in [Table B-6](#) and [Table B-7](#).

**Table B-6** *Driver Configuration*

Option	Description
<i>Action - Add Account Entitlement</i>	When an account is added, it allows you to select whether the account is enabled or disabled.
<i>Action - Remove Account Entitlement</i>	When an account is removed, it allows you to disable the account or delete the account.

**Table B-7** Password Synchronization

Option	Description
<i>Application accepts passwords from Identity Manager data store</i>	If True, allows passwords to flow from the Identity Manager data store to the connected system.
<i>Identity Manager accepts passwords from application</i>	If True, 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 <sup>™</sup> .
<i>Publish passwords to Distribution Password</i>	Use the password from the connected system to set the NMAS <sup>™</sup> Distribution Password used for Identity Manager password synchronization.
<i>Require password policy validation before publishing passwords</i>	If True, applies NMAS password policies during publish password operations. The password is not written to the data store if it does not comply.
<i>Reset user's external system password to the Identity Manager password on failure</i>	If True, on a publish Distribution Password failure, attempt to reset the password in the connected system using the Distribution Password from the Identity Manager data store.
<i>Notify the user of password synchronization failure via e-mail</i>	If True, notify the user by e-mail of any password synchronization failures.
<i>Connected System or Driver Name</i>	The name of the connected system, application or Identity Manager driver. This value is used by the e-mail notification templates.

## B.3 Named Passwords

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

You can also use the Named Passwords feature to store other pieces of information securely, such as a user name. To configured Named Passwords, see [Section 6.6, “Storing Driver Passwords Securely with Named Passwords,” on page 48](#).

In iManager:

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

In Designer:

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

## B.4 Engine Control Values

The engine control values are a means through which certain default behaviors of the Metadirectory engine can be changed. The values can only be accessed if a server is associated with the Driver Set object.

In iManager:

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

See [Table B-8](#) for a list of the engine control values.

In Designer:

- 1 In the Modeler, right-click the driver line.
- 2 Select *Properties > Engine Control Values*.
- 3 Click the tooltip icon to the right of the *Engine Control For Server* field. If a server is associated with the Identity Vault, and if you are authenticated, the engine control values display in the large pane.

See [Table B-8](#) for a list of the engine control values.

**Table B-8** *Engine Control Values*

Option	Description
<i>Subscriber channel retry interval in seconds</i>	The Subscriber channel retry interval controls how frequently the Metadirectory engine retries the processing of a cached transaction after the application shim's Subscriber object returns a retry status.
<i>Qualified form for DN-syntax attribute values</i>	The qualified specification for DN-syntax attribute values controls whether values for DN-syntax attribute values are presented in unqualified slash form or qualified slash form. A True setting means the values are presented in qualified form.
<i>Qualified form from rename events</i>	The qualified form for rename events controls whether the new-name portion of rename events coming from the Identity Vault is presented to the Subscriber channel with type qualifiers. For example, CN=. A True setting means the names are presented in qualified form.
<i>Maximum eDirectory replication wait time in seconds</i>	The maximum eDirectory™ replication wait time controls the maximum time that the Metadirectory engine waits for a particular change to replicate between the local replica and a remote replica. This only affects operations where the Metadirectory engine is required to contact a remote eDirectory server in the same tree to perform an operation and might need to wait until some change has replicated to or from the remote server before the operation can be completed (for example, object moves when the Identity Manager server does not hold the master replica of the moved object; file system rights operations for Users created from a template.)

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 backward-compatible mode causes the XSLT processor to use one or more behaviors that are not XPath 1.0 and XSLT 1.0 standards-compliant. This is done in the interest of backward-compatibility with existing DirXML<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>

Option	Description
<i>Use password event values</i>	<p>This control determines the source of the value reported for the nspmDistributionPassword attribute for Subscriber channel Add and Modify events.</p> <p>Setting the control to False means that the current value of nspmDistributionPassword is obtained and reported as the value of the attribute event. This means that only the current password value is available. This is the default behavior.</p> <p>Setting the control to True means that the value recorded with the eDirectory event is decrypted and is reported as the value of the attribute event. This means that both the old password value (if it exists) and the replacement password value at the time of the event are available. This is useful for synchronizing passwords to certain applications that require the old password to enable setting a new password.</p>
<i>Enable password synchronization status reporting</i>	<p>This control determines whether the Identity Manager engine reports the status of Subscriber channel password change events.</p> <p>Reporting the status of Subscriber channel password change events allows applications such as the Identity Manager User Application to monitor the synchronization progress of a password change that should be synchronized to the connected application.</p>

## B.5 Log Level

Every driver set and driver has a log level field where you can define the level of errors that should be tracked. The level you indicate here determines which messages are available to the logs. By default, the log level is set to track error messages (this also includes fatal messages). Change the log level if you want to track additional message types.

Novell® recommends that you use Novell Audit instead of setting the log levels. See “[Integrating Identity Manager with Novell Audit](#)” in the *Identity Manager 3.5.1 Logging and Reporting*.

In iManager:

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

See [Table B-9](#) for a list of the driver log levels.


In Designer:

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

See [Table B-9](#) for a list of the driver log levels.



**Table B-9** *Driver Log Levels*

Option	Description
<i>Use log settings from the DriverSet</i>	If this is selected, the driver logs events as the options are set on the Driver Set object.
<i>Log errors</i>	Logs just errors
<i>Log errors and warnings</i>	Logs errors and warnings
<i>Log specific events</i>	Logs the events that are selected. Click the  icon to see a list of the events.
<i>Update only the last log time</i>	Updates the last log time.
<i>Turn logging off</i>	Turns logging off for the driver.
<i>Turn off logging to DriverSet, Subscriber and Publisher logs</i>	If selected, turns all logging off for this driver on the Driver Set object, Subscriber channel, and the Publisher channel.
<i>Maximum number of entries in the log (50-500)</i>	Number of entries in the log. The default value is 50.

## B.6 Driver Image

Allows you to change the image associated with the driver. You can browse and select a different image from the default image.

The image associated with a driver is used by the Identity Manager Overview plug-in when showing the graphical representation of your Identity Manager configuration. Although storing an image is optional, it makes the overview display more intuitive.

---

**NOTE:** The driver image is maintained when a driver configuration is exported.

---

In iManager:

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

In Designer:

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

## B.7 Security Equals

Use the Security page to view or change the list of objects that the driver is explicitly security equivalent to. This object effectively has all rights of the listed objects.

If you add or delete an object in the list, the system automatically adds or deletes this object in that object's "Security Equal to Me" property. You don't need to add the [Public] trustee or the parent

containers of this object to the list, because this object is already implicitly security equivalent to them.

In iManager:

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

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

## B.8 Filter

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

In iManager:

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

The Filter editor is accessed through the outline view in Designer.

- 1 In an open project, click the *Outline* tab.
- 2 Select the driver you want to manage the filter for, then click the plus sign to the left.
- 3 Double-click the *Filter* icon to launch the Filter Editor.

## B.9 Edit Filter XML

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

In iManager:

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

You can edit the Filter in XML through the Filter Editor.

- 1 In an open project, click the *Outline* tab.
- 2 Select the driver you want to manage the filter for, then click the plus sign to the left.
- 3 Double-click the *Filter* icon and to launch the Filter Editor, then click *XML Source* at the bottom of the Filter Editor.

## B.10 Misc

Allows you to add a trace level to your driver. With the trace level set, DSTrace displays the Identity Manager events as the Metadirectory engine processes the events. The trace level only affects the driver it is set for. Use the trace level for troubleshooting issues with the driver when the driver is deployed. DSTrace displays the output of the specified trace level.


In iManager:

- 1 Click *Identity Manager > Identity Manager Overview*, then click *Search* to search for the driver set that is associated with the driver.
- 2 Browse to the driver, then click the upper right corner of the driver icon.
- 3 Click *Edit Properties > Misc*.  
See [Table B-10](#) for a list of the driver trace options.

In Designer:

- 1 Open a project in the Modeler, then right-click the driver line and select *Properties > Trace*.  
See [Table B-10](#) for a list of the driver trace options.

**Table B-10** *Driver Trace Options*

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.  <b>NOTE:</b> The trace file is created in multiple files. Identity Manager automatically divides the maximum file size by ten and creates ten separate files. The combined size of these files equals the maximum trace file size.
<i>Trace name</i>	Driver trace messages are prepended with the value entered in this field.
 <i>Use setting from Driver Set</i>	This option is only available in Designer. It allows the driver to use the same setting that is set on the Driver Set object.

## B.11 Excluded Users

Use this page to create a list of users or resources that are not replicated to the application. Novell recommends that you add all objects that represent an administrative role to this list (for example, the Admin object).

In iManager:

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

Designer does not list the excluded users.

## B.12 Driver Manifest

The driver manifest is like a resumé for the driver. It states what the driver supports, and includes a few configuration settings. The driver manifest is created by default when the Driver object is imported. A network administrator usually does not need to edit the driver manifest.

In iManager:

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

In Designer:

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

## B.13 Driver Inspector

The Driver Inspector page displays information about all of the objects associated with the driver.

- ♦ **Driver:** A link to run the *Driver Overview* on the driver that is being inspected.
- ♦ **Driver Set:** A link to run the *Driver Set Overview* of the driver set that holds the driver.
- ♦ **Delete:** Deletes the associations of the selected objects.
- ♦ **Refresh:** Select this option to re-read all of the objects associated with the driver and refresh the displayed information.
- ♦ **Actions:** Allows you to perform actions on the objects associated with the driver. Click *Actions* to expand the menu, which includes:
  - ♦ **Show All Associations:** Displays all objects associated with the driver.
  - ♦ **Filter for Disabled Associations:** Displays all objects associated with the driver that have a Disabled state.
  - ♦ **Filter for Manual Associations:** Displays all objects associated with the driver that have a Manual state.
  - ♦ **Filter for Migrate Associations:** Displays all objects associated with the driver that have a Migrate state.
  - ♦ **Filter for Pending Associations:** Displays all objects associated with the driver that have a Pending state.

- ♦ **Filter for Processed Associations:** Displays all objects associated with the driver that have a Processed state.
- ♦ **Filter for Undefined Associations:** Displays all objects associated with the driver that have an Undefined state.
- ♦ **Association Summary:** Displays the state of all objects associated with the driver.
- ♦ **Object DN:** Displays the DN of the associated objects.
- ♦ **State:** Displays the association state of the object.
- ♦ **Object ID:** Displays the value of the association.

## B.14 Driver Cache Inspector

The Driver Cache Inspector page uses a table format to display information about the cache file that stores events while the driver is stopped.

- ♦ **Driver:** A link to run the *Driver Overview* on the driver that is associated with this cache file.
- ♦ **Driver Set:** A link to run the *Driver Set Overview* on the driver set that holds the driver.
- ♦ **Driver's cache on:** Lists the server object that contains this instance of the cache file.
- ♦ **Start/Stop Driver icons:** Displays the current state of the driver and allows you to start or stop the driver.
- ♦ **Edit icon:** Allows you to edit the properties of the currently selected Server object.
- ♦ **Delete:** Deletes the selected items from the cache file.
- ♦ **Refresh:** Select this option 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: Allows you to specify a desired number.
- ♦ **Actions:** Allows 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:** Allows you to access a specified entry in the cache file. Specify the entry number, then click *OK*.
  - ♦ **Cache Summary:** Summarizes all of the events stored in the cache file.

## B.15 Inspector

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

## B.16 Server Variables

This page lets you enable and disable password synchronization and the associated options for the selected driver.

When setting up password synchronization, consider both the settings on this page for an individual driver and the Universal Password Configuration options in your password policies.

This page lets you control which password Identity Manager updates directly, either the Universal Password for an Identity Vault, or the Distribution Password used for password synchronization by Identity Manager.

However, Novell Modular Authentication Service (NMAS™) controls whether the various passwords inside the Identity Vault are synchronized with each other. Password Policies are enforced by NMAS, and they include settings for synchronizing Universal Password, NDS® Password, Distribution Password, and Simple Password.

To change these settings in iManager:

- 1 In iManager, select *Passwords > Password Policies*.
- 2 Select a password policy, then click *Edit*.
- 3 Select *Universal Password*.

This option is available from a drop-down list or a tab, depending on your version of iManager and your browser.

- 4 Select *Configuration Options*, make changes, then click *OK*.

---

**NOTE:** Enabling or disabling options on this page corresponds to values of True or False for certain global configuration values (GCVs) used for password synchronization in the driver parameters. Novell recommends that you edit them here in the graphical interface, instead of on the GCVs page. This interface helps ensure that you don't set conflicting values for the password synchronization GCVs.

---

Option	Description
<i>Identity Manager accepts password (Publisher Channel)</i>	<p>If this option is enabled, Identity Manager allows passwords to flow from the connected system driver into the Identity Vault data store.</p> <p>Disabling this option means that no <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>

---

Option	Description
<i>Use Distribution Password for password synchronization</i>	<p>To use this setting, you must have a version of eDirectory that supports Universal Password, regardless of whether you have enabled Universal Password in your password policies.</p> <p>If this option is enabled, a password value coming from the connected system is written to the Distribution Password. The Distribution Password is reversible, which means that it can be retrieved from the Identity Vault data store for password synchronization. It is used by Identity Manager for bidirectional password synchronization with connected systems. For Identity Manager to distribute passwords to connected systems, this option must be enabled.</p> <p>NMAS and Password policies control whether the Distribution Password is synchronized with other passwords in the Identity Vault. By default, the Distribution Password is the same as the Universal Password in the Identity Vault.</p> <p>If the password in the Identity Vault is to be independent of password synchronization, so that Identity Manager is a conduit only for synchronizing passwords among connected systems, change this default setting. In the Universal Password Configuration Options in a Password policy, disable <i>Synchronize Universal Password with Distribution Password</i>. This use of Identity Manager password synchronization is also referred to as "tunneling."</p>
<i>Accept password only if it complies with user's Password Policy</i>	<p>To use this setting, users must have a Password policy assigned that has Universal Password enabled, and Advanced Password Rules enabled and configured.</p> <p>If this option is chosen, Identity Manager does not write a password from this connected system to the Distribution Password in the Identity Manager data store or publish it to connected systems unless the password complies with the user's Password policy.</p> <p>By using the notification option that is also on this page, you can inform users when a password is not set because it is not compliant.</p>

Option	Description
<i>If password does not comply, ignore Password Policy on the connected system by resetting user's password to the Distribution Password</i>	<p data-bbox="722 258 1421 432">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 data-bbox="722 457 1421 512">Keep in mind that if you do not select this option, user passwords can become out-of-sync on connected systems.</p> <p data-bbox="722 537 1421 764">By using the notification option that is also on this page, you can inform users when a password is not set or reset. Notification is especially helpful for this option. If the user changes to a password that is allowed by the connected system but rejected by Identity Manager because of the Password policy, the user won't know that the password has been reset until the user receives a notification or tries to log in to the connected system with the old password.</p> <hr/> <p data-bbox="722 806 1421 919"><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>
<i>Always accept password; ignore Password Policies</i>	<p data-bbox="722 957 1421 1129">If you select this option, Identity Manager does not enforce the user's Password policy for this connected system. Identity Manager writes the password from this connected system to the Distribution Password in the Identity Vault data store, and distributes it to other connected systems, even if the password does not comply with the user's Password policy.</p>
<i>Application accepts passwords (Subscriber Channel)</i>	<p data-bbox="722 1157 1421 1331">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 data-bbox="722 1356 1421 1465">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 data-bbox="722 1491 1421 1717">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>



---

Option	Description
<i>Notify the user of password synchronization failure via-email</i>	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.

---

**NOTE:** To set up e-mail notification, select *Passwords > Edit EMail Templates*.

---