

Novell Identity Manager Driver for Active Directory*

3.1

www.novell.com

DRIVER GUIDE

August 24, 2006



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. Please refer to 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 © 2005 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.novell.com/company/legal/patents/> and one or more additional patents or pending patent applications in the U.S. and in other countries.

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

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

Novell Trademarks

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

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

eDirectory is a trademark of Novell, Inc.,

NCP and NetWare Core Protocol are registered trademarks of Novell, Inc.,

NDS and Novell Directory Services are registered trademarks of Novell, Inc., in the United States and other countries.

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

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

Novell Certificate Server is a trademark of Novell, Inc.,

Novell Client is a registered trademark of Novell, Inc.,

Third-Party Materials

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

Contents

About This Guide	9
1 Overview	11
1.1 Key Terms	11
1.1.1 Identity Manager	11
1.1.2 Connected System	11
1.1.3 Identity Vault	11
1.1.4 Metadirectory Engine	12
1.1.5 Active Directory Driver	12
1.1.6 Driver Shim	12
1.1.7 Remote Loader	12
1.2 New Features	13
1.2.1 Driver Features	13
1.2.2 Identity Manager Features	13
1.3 Data Transfers Between Systems	13
1.3.1 Publisher and Subscriber Channels	13
1.4 Default Driver Configuration	14
1.4.1 User Object Name Mapping	14
1.4.2 Data Flow	14
2 Preparing Active Directory	19
2.1 Active Directory Prerequisites	19
2.2 Planning Your Installation	19
2.2.1 Where To Install the Active Directory Driver and Shim	19
2.3 Addressing Security Issues	21
2.3.1 Authentication Methods	22
2.3.2 Encryption	22
2.3.3 SSL Connection Between the Remote Loader and Identity Manager	25
2.4 Creating an Administrative Account	26
2.5 Becoming Familiar with Driver Features	26
2.5.1 Multivalue Attributes	26
2.5.2 Managing Account Settings Using Custom Boolean Attributes	27
2.5.3 Provisioning Exchange Mailboxes using the homeMDB Attribute	28
2.5.4 Expiring Accounts in Active Directory	28
2.5.5 Retaining eDirectory Objects When You Restore Active Directory Objects	28
3 Installing the Active Directory Driver	29
3.1 Basic Steps	29
3.2 Installing the Active Directory Driver Shim	30
3.2.1 Installing the Shim on a Metadirectory Server	30
3.2.2 Installing the Shim on a Remote Loader	33
3.3 Installing Preconfiguration Import Files	35
3.4 Installing the Active Directory Discovery Tool	36
4 Configuring the Active Directory Driver	39
4.1 Importing the Driver Configuration File in Designer	39

4.2	Importing the Driver Configuration File in iManager	39
4.3	Configuration Parameters.	40
5	Upgrading the Active Directory Driver	49
5.1	Checklist for Upgrading	49
5.2	Addressing the Login Disabled Value.	50
5.3	Upgrading the Driver Shim from DirXML 1.1a	51
5.4	Upgrading the Driver Shim from IDM 2.x	51
5.5	Applying the Overlay for Exchange Mailboxes.	52
5.5.1	Applying the Overlay in Designer.	53
5.5.2	Applying the Overlay in iManager	56
6	Managing the Active Directory Driver	59
6.1	Security Parameters	59
6.1.1	Recommended Security Configurations.	60
6.2	Managing Groups.	61
6.3	Managing Microsoft Exchange Mailboxes	62
6.4	Activating the Driver	63
7	Password Synchronization	65
7.1	Comparing Password Synchronization 1.0 and Password Synchronization Provided with Identity Manager	65
7.2	Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager	67
7.2.1	Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies.	70
7.3	New Driver Configuration and Identity Manager Password Synchronization	73
7.4	Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization	73
7.5	Setting Up Password Synchronization Filters.	77
7.5.1	Configuring Password Filters for All Domain Controllers from One Machine.	78
7.5.2	Separately Configuring Password Filters on Each Domain Controller.	81
7.6	Retrying Synchronization after a Failure	84
7.6.1	Retrying after an Add or Modify Event	84
7.6.2	Password Expiration Time	85
8	Troubleshooting	89
8.1	Changes Are Not Synchronizing from the Publisher or Subscriber	89
8.2	Using Characters Outside the Valid NT Logon Names	89
8.3	Synchronizing c, co, and countryCode Attributes.	90
8.4	Synchronizing Operational Attributes	90
8.5	Password Complexity on Windows 2003	90
8.6	Error Message LDAP_SERVER_DOWN	91
8.7	Tips on Password Synchronization	91
8.7.1	Providing Initial Passwords	92
8.8	Where to Set the SSL Parameter.	92
8.9	Active Directory Account Disabled after a User Add on the Subscriber Channel.	92
8.9.1	Account Disabled in Active Directory Users and Computers	93
8.10	Moving a Parent Mailbox to a Child Domain	93
8.11	Restoring Active Directory	93

8.12	Moving the Driver to a Different Domain Controller	94
8.13	Migrate from Active Directory	94
8.14	Setting LDAP Server Search Constraints	94
A	Changing Permissions on the CN=Deleted Objects Container	97
B	Documentation Update	101
B.1	July 31, 2006	101
B.1.1	Introduction to Policies	101
B.2	September 8, 2006	101
B.2.1	Implementing Credential Provisioning Policies with Novell SecureLogin	101
B.2.2	Configuring Credential Provisioning Policies for Novell SecureLogin	101
B.2.3	Implementing Credential Provisioning Policies with Novell SecretStore	102
B.2.4	Configuring Credential Provisioning Policies for Novell SecretStore.	102

About This Guide

This guide explains how to install, configure, and manage the Identity Manager Driver for Active Directory.

- ◆ Chapter 1, “Overview,” on page 11
- ◆ Chapter 2, “Preparing Active Directory,” on page 19
- ◆ Chapter 3, “Installing the Active Directory Driver,” on page 29
- ◆ Chapter 5, “Upgrading the Active Directory Driver,” on page 49
- ◆ Chapter 6, “Managing the Active Directory Driver,” on page 59
- ◆ Chapter 7, “Password Synchronization,” on page 65
- ◆ Chapter 8, “Troubleshooting,” on page 89
- ◆ Appendix A, “Changing Permissions on the CN=Deleted Objects Container,” on page 97

Audience

This guide is intended for Active Directory administrators, Novell® eDirectory™ 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 and enter your comments there.

Documentation Updates

For the most recent version of this document, visit the [Drivers Documentation Web site \(http://www.novell.com/documentation/lg/dirxmldrivers\)](http://www.novell.com/documentation/lg/dirxmldrivers).

Additional Documentation

For documentation on using Identity Manager and the other Identity Manager drivers, see the [Identity Manager Documentation Web site \(http://www.novell.com/documentation/lg/dirxml20\)](http://www.novell.com/documentation/lg/dirxml20).

Documentation Conventions

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

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

Overview

1

- ♦ Section 1.1, “Key Terms,” on page 11
- ♦ Section 1.2, “New Features,” on page 13
- ♦ Section 1.3, “Data Transfers Between Systems,” on page 13
- ♦ Section 1.4, “Default Driver Configuration,” on page 14

1.1 Key Terms

- ♦ Section 1.1.1, “Identity Manager,” on page 11
- ♦ Section 1.1.2, “Connected System,” on page 11
- ♦ Section 1.1.3, “Identity Vault,” on page 11
- ♦ Section 1.1.4, “Metadirectory Engine,” on page 12
- ♦ Section 1.1.5, “Active Directory Driver,” on page 12
- ♦ Section 1.1.6, “Driver Shim,” on page 12
- ♦ Section 1.1.7, “Remote Loader,” on page 12

1.1.1 Identity Manager

Novell® Identity Manager is a service that synchronizes data among servers in a set of connected systems by using a robust set of configurable policies. Identity Manager uses the Identity Vault to store shared information, and uses the Metadirectory engine for policy-based management of the information as it changes in the vault or connected system. Identity Manager runs on the server where the Identity Vault and the Metadirectory engine are located.

1.1.2 Connected System

A connected system is any system that can share data with Identity Manager through a driver. Active Directory is a connected system.

1.1.3 Identity Vault

The Identity Vault is a persistent database powered by eDirectory™ and used by Identity Manager to hold data for synchronization with a connected system. The vault can be viewed narrowly as a private data store for Identity Manager or more broadly as a metadirectory that holds enterprise-wide data. Data in the vault is available to any protocol supported by eDirectory, including NCP™ (the traditional protocol used by such utilities as ConsoleOne® and iManager), LDAP, and DSML.

Because the vault is powered by eDirectory, Identity Manager can be easily integrated into your corporate directory infrastructure by using your existing directory tree as the vault.

1.1.4 Metadirectory Engine

The Metadirectory engine is the core server that implements the event management and policies of Identity Manager. The engine runs on the Java* Virtual Machine in eDirectory.

1.1.5 Active Directory Driver

A driver implements data sharing policy for a connected system. You control the actions of the driver by using iManager to define the filters and policy. For Active Directory, a driver implements policy for a single domain.

1.1.6 Driver Shim

A driver shim is the component of a driver that converts the XML-based Identity Manager command and event language (XDS) to the protocols and API calls needed to interact with a connected system. The shim is called to execute commands on the connected system after the Output Transform has been run. Commands are usually generated on the Subscriber channel but can be generated by command write-back on the Publisher channel.

The shim also generates events from the connected system for the Input Transformation policy. A driver shim can be implemented either in Java class or as a native Windows DLL file. The shim for Active Directory is `ADDriver.dll`.

`ADDriver.dll` is implemented as a native Windows DLL file. `ADDriver` uses several different Windows APIs to integrate with Active Directory. These APIs typically require some type of login and authentication to succeed. Also, the APIs might require that the login account have certain rights and privileges within Active Directory and on the machine where `ADDriver.dll` executes.

If you use the Remote Loader, `ADDriver.dll` executes on the server where the Remote Loader is running. Otherwise, it executes on the server where the Metadirectory engine is running.

1.1.7 Remote Loader

A Remote Loader enables a driver shim to execute outside of the Metadirectory engine (perhaps remotely on a different machine). The Remote Loader is typically used when a requirement of the driver shim is not met by the Identity Manager server. For example, if the Metadirectory engine is running on Linux*, the Remote Loader is used to execute the Active Directory driver shim on a Windows server.

The Remote Loader is a service that executes the driver shim and passes information between the shim and the Metadirectory engine. When you use a Remote Loader, you install the driver shim on the server where the Remote Loader is running, not on the server where the Metadirectory engine is running. You can choose to use SSL to encrypt the connection between the Metadirectory engine and the Remote Loader.

When you use the Remote Loader with the Active Directory driver shim, two network connections exist:

- ◆ Between the domain controller and the Remote Loader
- ◆ Between Active Directory and the Active Directory driver shim

1.2 New Features

- ♦ [Section 1.2.1, “Driver Features,” on page 13](#)
- ♦ [Section 1.2.2, “Identity Manager Features,” on page 13](#)

1.2.1 Driver Features

- ♦ Platform Logon is a driver shim configuration parameter. It enables local logon for the shim. When local logon is enabled, Subscriber channel password set and password modify use platform password management API's that do not require an SSL-encrypted LDAP session.

Exchange operations using CDOEXM uses the thread identity for authentication and authorization to reduce the possibility that operations outside of the LDAP channel can fail. Please refer to [Chapter 4, “Configuring the Active Directory Driver,” on page 39](#) for details.

- ♦ The driver shim configuration parameters are updated. The driver parameters use flexible prompting which allows for a better categorization of parameters and better control of the values placed in the parameters. Parameters constrained to a set of well-known values are controlled by a drop-down list and parameters that require integer values are checked for invalid characters.
- ♦ Two driver shim configuration parameters are added to control Microsoft Exchange mailbox moves and deletes. When CDOEXM and Exchange Mailbox Move are enabled, setting a value for the homeMDB attribute on a user object that already holds an Exchange mailbox causes that mailbox to be moved to new Exchange Message Database. The shim supports intra-domain moves only: the Exchange server hosting the new message database must be in the same domain being managed by the shim.
- ♦ Increased support for Role-Based Entitlements through the User Application or through policies. See [“Creating and Using Entitlements”](#) in the *Novell Identity Manager 3.0.1 Administration Guide*.
- ♦ The driver shim includes support for extended queries (query-ex). Extended queries enable paged-results in LDAP searches. This feature allows the shim to migrate a larger dataset from Active Directory to the Identity Vault. See [Chapter 8, “Troubleshooting,” on page 89](#) for more information on migrating from Active Directory.

1.2.2 Identity Manager Features

For information about the new features in Identity Manager, see [“What's New in Identity Manager?”](#) in the *Identity Manager 3.0.1 Installation Guide*.

1.3 Data Transfers Between Systems

This sections explains how the data flows between Active Directory and the Identity Vault.

1.3.1 Publisher and Subscriber Channels

The Active Directory driver supports Publisher and Subscriber channels.

The Publisher channel does the following:

- ◆ Reads events from Active Directory for the domain hosted on the server that the driver shim is connecting to.
- ◆ Submits that information to the Identity Vault.

The Subscriber channel does the following:

- ◆ Watches for additions and modifications to the Identity Vault objects.
- ◆ Makes changes to Active Directory that reflect those changes.

You can configure the driver so that both Active Directory and the Identity Vault are allowed to update a specific attribute. In this configuration, the most recent change determines the attribute value, except in the case of merge operations that are controlled by the filters and merge authority.

1.4 Default Driver Configuration

The Active Directory driver is shipped with a default configuration file called `ActiveDirectory.xml`. When imported with Designer or iManager, this configuration file creates a driver with a set of rules and policies suitable for synchronizing with Active Directory. If your requirements for the driver are different from the default policies, you need to change them to effect the policies you want. Pay close attention to the default Matching policies. The data that you trust to match users usually is different from the default. The policies themselves are commented and you can gain a greater understanding of what they do by importing a test driver and reviewing the policies with Designer or iManager.

1.4.1 User Object Name Mapping

Management utilities for the Identity Vault such as iManager and ConsoleOne typically name user objects differently than the Users and Computers snap-in for the Microsoft* Management Console (MMC). Make sure that you understand the differences so the Matching policy and any Transformation policies you have are implemented properly.

1.4.2 Data Flow

Data can flow between Active Directory and an Identity Vault. The flow of data is controlled by the policies that are in place for the Active Directory driver.

Policies

Policies control data synchronization between Active Directory and an Identity Vault.

During the driver configuration, the Active Directory configuration file enables you to select several options that affect the default policies and filters created for you. The [Table 1-1 on page 15](#) lists these options and how they affect policies and filters that are created:

Table 1-1 Data Flow Options

Option	Description
Vault to AD	<p>Configure Data Flow establishes the initial driver filter which controls the classes and attributes that will be synchronized. The purpose of this option is to configure the driver to best express your general data flow policy. It can be changed after import to reflect specific requirements.</p> <p><i>Bidirectional</i> sets classes and attributes to synchronize on both the Publisher and Subscriber channels. A change in either the Identity Vault or Active Directory is reflected on the other side. Use this option if you want both sides to be authoritative sources of data.</p> <p><i>AD to Vault</i> sets class and attributes to synchronize on the Publisher channel only. A change in Active Directory is reflected in the Identity Vault but Identity Vault changes are ignored. Use this option if you want Active Directory to be the authoritative source of data.</p> <p><i>Vault to AD</i> sets classes and attributes to synchronize on the Subscriber channel only. A change in the Identity Vault is reflected in Active Directory but Active Directory changes are ignored. Use this option if you want the vault to be the authoritative source of data.</p>
Flat	<p>Publisher Placement controls where objects are created in the Identity Vault.</p> <p><i>Mirrored</i> places objects in the Identity Vault in the same hierarchy as they exist in Active Directory.</p> <p><i>Flat</i> places all objects in the base container in the Identity Vault specified during configuration.</p>
Mirrored Flat	<p>Subscriber Placement controls how objects are placed in Active Directory.</p> <p><i>Mirrored</i> places objects in Active Directory in the same hierarchy as they exist in the Identity Vault.</p> <p><i>Flat</i> places all objects in the base container in Active Directory specified during configuration.</p>

The [Table 1-2 on page 16](#) lists default policies and describes how selections during configuration affect the policies:

Table 1-2 *Default Policies*

Policy	Description
Create	In either the mirrored or flat hierarchy, you must define Full Name to create an Active Directory user as a user in the Identity Vault.
Matching	In a mirrored hierarchy, the Matching policy attempts to match an object in the same position in the hierarchy. In a flat hierarchy, the Matching policy attempts to match the user with an object that has the same Full Name in the base container that you specify.
Placement	In a mirrored hierarchy, the Placement policy places all objects in a hierarchy that mirrors the hierarchy of the data store sending the operation. In a flat hierarchy, the Placement policy places all objects in the base container that you specify.

Schema Mapping

The following Identity Vault user, group, and Organizational Unit attributes are mapped to Active Directory user and group attributes.

The mappings listed in the tables are default mappings. You can remap same-type attributes.

Table 1-3 *Attributes Mapped for All Classes*

eDirectory	Active Directory
CN	cn
Description	description
Facsimile Telephone Number	facsimiletelephoneNumber
Full name	displayName
Given Name	givenName
Initials	initials
Internet EMail Address	mail
L	physicalDeliveryOfficeName
Locality	locality
Login Disabled	dirxml-uACAccountDisabled
Login Expiration Time	accountExpires
Physical Delivery Office Name	I
Postal Code	PostalCode
Postal Office Box	postOfficeBox
S	st
SA	streetAddress

eDirectory	Active Directory
See Also	seeAlso
Surname	sn
Telephone Number	telephoneNumber
Title	title

eDirectory's L attribute is mapped to Active Directory's physicalDeliveryOfficeName attribute, and eDirectory's Physical Delivery Office Name attribute is mapped to Active Directory's L attribute. Because similarly named fields have the same value, mapping the attributes this way enable the attributes to work well with ConsoleOne and the Microsoft Management Console.

Table 1-4 *Attribute Mapped for Users*

eDirectory	Active Directory
CN	userPrincipalName
DirXML-ADAliasName	sAMAccountName
Login Allowed Time Map	logonHours

Table 1-5 *Mapped Organizational Unit Attributes*

eDirectory	Active Directory
Organizational Unit	organizationalUnit
OU	ou

Name Mapping Policies

The default configuration includes two name mapping policies that work together to help you reconcile different naming policies between the Identity Vault and Active Directory. When you create a user with the Active Directory Users and Computers tool (a snap-in for the Microsoft Management Console and abbreviated as MMC in this document) you see that the user full name is used as its object name. Attributes of the user object define Pre-Windows 2000 Logon Name (also known as the NT Logon Name or sAMAccountName) and the Windows 2000 Logon Name (also known as the userPrincipalName). When you create a user in the Identity Vault with iManager or ConsoleOne, the object name and the user logon name are the same.

If you create some users in Active Directory using MMC and other objects in the Identity Vault or another connected system that is synchronized with the Identity Vault, the object can look odd in the opposing console and might fail to be created in the opposing system at all.

The Full Name Mapping Policy is used to manage objects in Active Directory using the MMC conventions. When enabled, The Full Name attribute in the Identity Vault is synchronized with the object name in Active Directory.

The NT Logon Name Mapping Policy is used to manage objects in Active Directory using the Identity Vault conventions. When enabled, the Identity Vault object name is used to synchronize

both the object name and NT Logon Name in Active Directory. Objects in Active Directory are named the same as the Identity Vault and the NT Logon Name matches the Identity Vault logon name.

When both of the policies are enabled at the same time, the Active Directory object name is the Identity Vault Full Name, but the NT Logon Name matches the Identity Vault logon name.

When both policies are disabled, no special mapping is made. The object names is synchronized and there will be no special rules for creating the NT Logon Name. Because the NT Logon Name is a mandatory attribute in Active Directory, you need some method of generating one during add operations. The NT Logon Name (sAMAccountName) is mapped to the DirMXL-ADAliasName in the Identity Vault, so you could either use that attribute to control the NT Logon Name in Active Directory or build your own policy in the Subscriber Create policies to generate one. With this policy selection, users created with MMC uses the object name generated by MMC as the object name in the Identity Vault. This name might be inconvenient for login to the Vault.

Windows 2000 Logon Name Policies

The Windows 2000 Logon name (also known as the userPrincipalName or UPN) does not have a direct counterpart in the Identity Vault. UPN looks like an e-mail address (user@mycompany.com) and might in fact be the user's e-mail name. The important thing to remember when working with UPN is that it must use a domain name (the part after the @ sign) that is configured for your domain to be used successfully. You can find out what domain names are allowed by creating a user using MMC and inspecting the domain name drop-down box when adding the UPN.

The default configuration offers several choices for managing userPrincipalName. If your domain is set up so that the user's e-mail address can be used as a userPrincipalName, then one of the options to track the user's e-mail address is appropriate. You can make userPrincipalName follow either the Identity Vault or Active Directory e-mail address, depending on which side is authoritative for e-mail. If the user e-mail address is not appropriate, you can choose to have a userPrincipalName constructed from the user logon name plus a canned domain name. If more than one name can be used, update the policy after import to make the selection. If none of these options are appropriate, then you can disable the default policies and write your own.

Entitlements

Entitlements make it easier to integrate Identity Manager with Identity Manager User Application and Role-Based Services in eDirectory. When using User Application, an action such as provisioning an account in Active Directory is delayed until the proper approvals have been made. When using Role-Based Services, rights assignments are made based on attributes of a user object and not by regular group membership. Both of these services offer a challenge to Identity Manager because it is not obvious from the attributes of an object whether an approval has been granted or the user matches a role.

Entitlements standardize a method of recording this information on objects in the Identity Vault. From the driver perspective, an entitlement grants or revokes the right to something in Active Directory. You can use entitlements to grant the right to an account in Active Directory, to control group membership, and to provision Exchange mailboxes. The driver is unaware of User Application or Role-Based Entitlements. It depends on the User Application server or the Entitlements driver to grant or revoke the entitlement for a user based upon its own rules.

You should enable entitlements for the driver only if you plan to use User Application or Role-Based Entitlements with the driver.

Preparing Active Directory

2

In this section:

- ♦ [Section 2.1, “Active Directory Prerequisites,” on page 19](#)
- ♦ [Section 2.2, “Planning Your Installation,” on page 19](#)
- ♦ [Section 2.3, “Addressing Security Issues,” on page 21](#)
- ♦ [Section 2.4, “Creating an Administrative Account,” on page 26](#)
- ♦ [Section 2.5, “Becoming Familiar with Driver Features,” on page 26](#)

2.1 Active Directory Prerequisites

- Novell® Identity Manager 3.0.1 and its prerequisites, as listed in the “[Installing Identity Manager](#)” section in the *Identity Manager 3.0.1 Installation Guide*.
- Windows 2003 Server, or Windows 2000 Server with Service Pack 2 or later.
- Internet Explorer 5.5 or later on the server running the Active Directory (AD) driver and on the target domain controller.
- Active Directory domain controller DNS name or IP address, depending on the authentication method.

Also, we recommend that the server hosting the Active Directory driver be a member of the Active Directory domain. This is required to provision Exchange mailboxes and synchronize passwords. If you don't require these features, the server can be a member of any domain as long as the Simple (simple bind) authentication mode is used. To have bidirectional password synchronization function, the Negotiate authentication option must be selected.

2.2 Planning Your Installation

You can install the Active Directory driver on either the domain controller or a member server. Before you start the driver installation, determine the following:

- ♦ Where to install the Active Directory driver shim
- ♦ How to address security issues

2.2.1 Where To Install the Active Directory Driver and Shim

The Active Directory driver shim must run on one of the supported Windows platforms. However, you don't need to install the Metadirectory engine on this same machine. Using a Remote Loader, you can separate the engine and the driver shim, allowing you to balance the load on different machines or accommodate corporate directives.

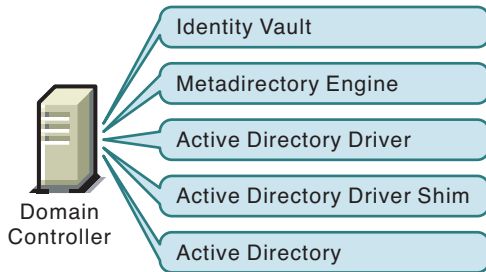
The installation scenario you select determines how the driver shim is installed. If you choose to install the driver shim on the same machine as Identity Manager (where the Metadirectory engine and the Identity Vault are located), Identity Manager calls the driver shim directly. If you choose to install the driver shim on another machine, you must use the Remote Loader.

The driver itself is installed the same way in each of the scenarios. See [Chapter 4, “Configuring the Active Directory Driver,”](#) on page 39.

Local Installation

A single Windows domain controller can host the Identity Vault, the Metadirectory engine, and the driver.

Figure 2-1 Scenario 1 - All Components are on one server



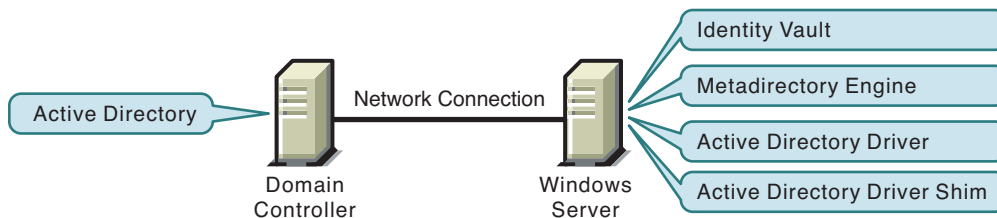
This configuration works well for organizations that want to save on hardware costs. It is also the highest-performance configuration because there is no network traffic between Identity Manager and Active Directory.

However, hosting Identity Vault and the Metadirectory engine on the domain controller increases the overall load on the controller and increases the risk that the controller might fail. Because domain controllers play a critical role in Microsoft networking, many organizations are more concerned about the speed of the domain authentication and the risks associated with a failure on the domain controller than about the cost of additional hardware.

Remote Installation on Windows Server Only

You can install the Identity Vault, the Metadirectory engine, and the driver on a separate computer from the Active Directory domain controller. This configuration leaves the domain controller free of any Identity Manager software.

Figure 2-2 Scenario 2 - Active Directory and the Driver Shim on Separate Servers

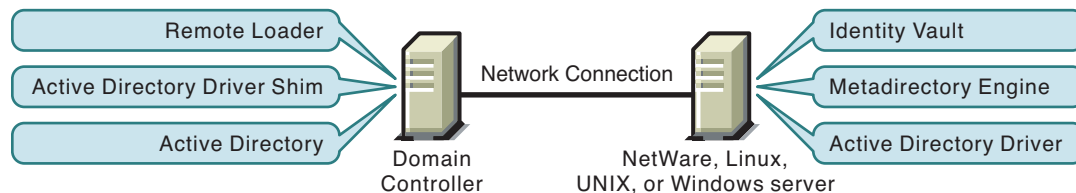


This configuration is attractive if corporate policy disallows running the driver on your domain controller.

Remote Installation on Windows and Other Platforms

You can install the Remote Loader and driver shim on the Active Directory domain controller, but install the Identity Vault and the Metadirectory engine on a separate server.

Figure 2-3 Scenario 3 - Active Directory, the Remote Loader, and Driver Shim on One Server



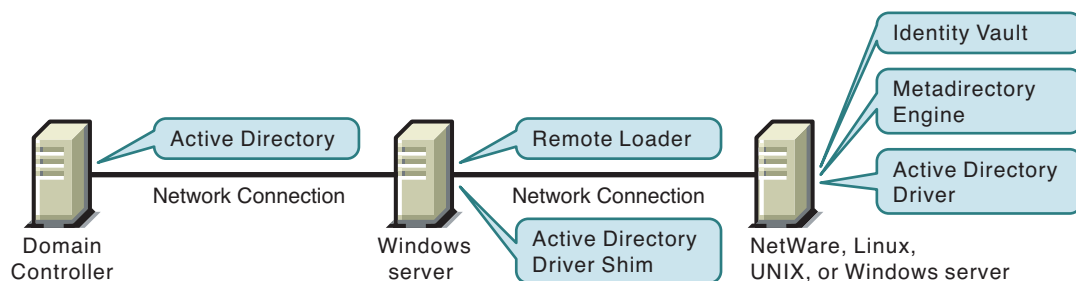
This configuration is attractive if your Identity Vault and Metadirectory engine (Identity Manager) installations are on a platform other than one of the supported versions of Windows.

Both Scenario 2 and Scenario 3 configurations eliminate the performance impact of hosting the Identity Vault and the Metadirectory engine on the domain controller.

Remote Installation on a Windows Member Server

If you have platform requirements and domain controller restrictions in place, you can use a three-server configuration.

Figure 2-4 Scenario 4 - Three-server Configuration



This configuration is more complicated to set up, but it accommodates the constraints of some organizations. In this figure, the two Windows servers are member servers of the domain.

2.3 Addressing Security Issues

The major security issues to consider are authentication, encryption, and use of the Remote Loader. If you have Windows 2003 or Windows 2000 SP3 or later, consider a security option called signing. See "Use Signing" in ["Security Parameters" on page 59](#).

A simple prescription for managing security is not possible because the security profile available from Windows varies with service pack, DNS server infrastructure, domain policy, and local policy settings on the server. The following sections explain your security choices and provide suggested configurations. When implementing your driver and when upgrading components, pay close attention to security.

2.3.1 Authentication Methods

Authentication identifies the driver shim to Active Directory and, potentially, the local machine. To authenticate to Active Directory, you can use either the Negotiate method or the Simple (simple bind) method.

Table 2-1 *Authentication Methods*

Authentication Method	Description	Advantages	Disadvantages
Negotiate	The preferred method. Uses Kerberos*, NTLM, or a pluggable authentication scheme if one is installed.	The driver can be installed on any server in the domain.	The server hosting the driver must be a member of the domain.
Simple	Used when the server hosting the driver shim is not a member of the domain.	The driver can be installed on a server that is not a member of the domain.	Some provisioning services are unavailable, such as Exchange mailbox provisioning and password synchronization.

2.3.2 Encryption

SSL encrypts data. Depending on your configuration, SSL can be used in two places:

- ◆ Between the Active Directory driver and the domain controller
- ◆ Between the Identity Vault and the Remote Loader running the Active Directory driver

Password synchronization occurs between Active Directory and the Identity Vault (eDirectory). You need to make sure that you use SSL with any communication that goes across the network.

If the Metadirectory engine, Identity Vault, the Active Directory driver, and Active Directory are on the same machine, you don't need SSL. Communication isn't going across the network.

However, if you are accessing Active Directory remotely by using an Active Directory driver shim on a member server, you need to set up SSL between the Active Directory driver shim and Active Directory. You do this by setting the SSL parameter to Yes on the driver configuration. See [Step 5 on page 25](#), in [Section](#), “[SSL Connection Between the Active Directory Driver and the Domain Controller](#),” on page 23.

If you are using the Remote Loader on the Domain Controller, you can set up SSL between the Metadirectory engine and the Remote Loader. For additional information on SSL and Remote Loaders, see “[Setting Up a Connected System](#)” in the *Novell Identity Manager 3.0.1 Administration Guide*.

The following table outlines where SSL connections can be used for each of the scenarios discussed in [Section 2.2, “Planning Your Installation,” on page 19](#):

Table 2-2 *SSL Connects*

Configuration	SSL Connections Available
Single-Server	No SSL connections are necessary.
Two-Server: Identity Manager and the Active Directory driver are on the same server	An SSL connection can be established between the Active Directory driver and the domain controller.
Dual-Server: Identity Manager is on one server but the Active Directory driver is on a separate server	An SSL connection can be established between Identity Manager and the Remote Loader running the Active Directory driver.
Three-Server	An SSL connection can be established between the Active Directory driver and the domain controller. An SSL connection can also be established between Identity Manager and the Remote Loader running the Active Directory driver.

SSL Connection Between the Active Directory Driver and the Domain Controller

To make SSL connections to an Active Directory domain controller, you must be set up to use SSL. This involves setting up a certificate authority, then creating, exporting, and importing the necessary certificates.

Setting Up a Certificate Authority

Most organizations already have a certificate authority. In this case, you need to export a valid certificate, then import it to the certificate store on your domain controller. The server hosting the driver shim must trust the root certificate authority that the issuing certificate authority of this certificate chains to.

If you do not have a certificate authority in your organization, you must establish one. Novell, Microsoft, and several other third parties provide the tools necessary to do this. Establishing a certificate authority is beyond the scope of this guide. For more information, see

- ♦ [Novell Certificate Server™ 2.5 Administration Guide \(http://www.novell.com/documentation/lg/crt252/index.html\)](http://www.novell.com/documentation/lg/crt252/index.html)
- ♦ [Securing Windows 2000 Server \(http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.msp\)](http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.msp)

Creating, Exporting, and Importing Certificates

After you have a certificate authority, for LDAP SSL to operate successfully, the LDAP server must have the appropriate server authentication certificate installed. Also, the server hosting the driver shim must trust the authority that issued those certificates. Both the server and the client must support 128-bit encryption.

- 1 Generate a certificate that meets the following Active Directory LDAP service requirements:
 - ♦ The LDAPS certificate is located in the Local Computer's Personal certificate store (programmatically known as the computer's MY certificate store).
 - ♦ A private key matching the certificate is present in the Local Computer's store and is correctly associated with the certificate.

The private key must not have strong private-key protection enabled.

- ◆ The Enhanced Key Usage extension includes the Server Authentication (1.3.6.1.5.5.7.3.1) object identifier (also known as OID).
- ◆ The Active Directory fully qualified domain name (for example, DC01.DOMAIN.COM) of the domain controller appears in one of the following places:
 - ◆ The Common Name (CN) in the Subject field.
 - ◆ The DNS entry in the Subject Alternative Name extension.
- ◆ The certificate was issued by a CA that the domain controller and the LDAPS clients trust. Trust is established by configuring the clients and the server to trust the root CA that the issuing CA chains to.

This certificate permits the LDAP service on the domain controller to listen for and automatically accept SSL connections for both LDAP and global catalog traffic.

NOTE: This information appears in the Microsoft Knowledge Base Article 321051, [How to Enable LDAP over SSL with a Third-Party Certificate Authority \(http://support.microsoft.com/default.aspx?scid=kb;en-us;321051\)](http://support.microsoft.com/default.aspx?scid=kb;en-us;321051). Consult this document for the latest requirements and additional information.

- 2 Export this certificate in one of the following standard certificate file formats supported by Windows 2000:

- ◆ Personal Information Exchange (PFX, also called PKCS #12)
- ◆ Cryptographic Message Syntax Standard (PKCS #7)
- ◆ Distinguished Encoding Rules (DER) Encoded Binary X.509
- ◆ Base64 Encoded X.509

- 3 Install this certificate on the domain controller.

The following links contain instructions for each supported platform:

- ◆ [HOW TO: Install Imported Certificates on a Web Server in Windows Server 2003 \(http://support.microsoft.com/default.aspx?scid=kb;en-us;816794\)](http://support.microsoft.com/default.aspx?scid=kb;en-us;816794)
- ◆ [HOW TO: Install Imported Certificates on a Web Server in Windows 2000 \(http://support.microsoft.com/default.aspx?scid=kb;EN-US;310178\)](http://support.microsoft.com/default.aspx?scid=kb;EN-US;310178)

Follow the instructions listed under Import the Certificate into the Local Computer Store.

- 4 Ensure that a trust relationship is established between the server hosting the driver shim and the root certificate authority that issued the certificate.

The server hosting the driver shim must trust the root certificate authority that the issuing certificate authority chains to.

For more information on establishing a trust for certificates, see the *Policies to establish trust of root certification authorities* topic in *Windows 2000 Server Help*.

- 5 In iManager, edit the driver properties and change the *Use SSL (yes/no)* option to yes.

Driver Parameters

SW3K-NDS.WM

Edit XML

Driver Settings

Polling Interval (min.)	<input type="text" value="1"/>
Authentication Method	<input type="text" value="Negotiate"/>
Use Signing (yes/no)	<input type="text" value="no"/>
Use Sealing (yes/no)	<input type="text" value="no"/>
Use SSL (yes/no)	<input type="text" value="yes"/>
Heart Beat	<input type="text" value="0"/>
Password Sync Timeout (minutes):	<input type="text" value="5"/>

- 6 Restart the driver.

When the driver restarts, an SSL connection is negotiated between the domain controller and the server running the Active Directory driver shim.

Verify the Certificate

To verify the certificate, authenticate to AD via SSL. Use the `ldifde` command line utility found on Windows servers. To use the `ldifde` command:

- 1 Open a command line prompt
- 2 Enter `ldifde -f output/input file -t 636 -b administrator domain password -s computerFullName`

Here is an example of what you would enter if your server is configured for port 636.

```
ldifde -f out.txt -t 636 -b administrator dxad.novell.com novell -s parent1.dxad3.lab.novell
```

The output is sent to the `out.txt` file. If you open the file and see the objects in Active Directory listed, you made a successful SSL connection to Active Directory and the certificate is valid.

2.3.3 SSL Connection Between the Remote Loader and Identity Manager

If you are using the Remote Loader, you need to set up SSL between the Metadirectory engine and the Remote Loader, and configure the settings between the driver and Active Directory.

For information on establishing an SSL connection between the Remote Loader and Identity Manager, see “[Setting Up Remote Loaders](#)” in the *Novell Identity Manager 3.0.1 Administration Guide*.

2.4 Creating an Administrative Account

In a test environment, use the Administrator account until you get the Active Directory driver working. Then create an administrative account, which has the proper rights (including restricted rights), that the Active Directory driver can use exclusively to authenticate to Active Directory.

Doing this keeps the Identity Manager administrative account insulated from changes to other administrative accounts. Advantages to this design are:

- ◆ You can use Active Directory auditing to track the activity of the Active Directory driver.
- ◆ You can implement a password change policy as with other accounts, then make necessary updates to the driver configuration.

This account name and password are stored in the driver configuration. Therefore, you must change this password whenever the account password changes. If you change the account password without updating the driver configuration, authentication fails the next time the driver is restarted.

At a minimum, this account must have Read and Replicating Directory Changes rights at the root of the domain for the Publisher channel to operate. You also need Write rights to any object modified by the Subscriber channel. Write rights can be restricted to the containers and attributes that are written by the Subscriber channel.

To instrument Exchange mailboxes, your Identity Manager account must have “Act as part of the Operating System” permission for the logon account.

Windows 2003 requires that you have additional rights in order to see deleted objects. See [Appendix A, “Changing Permissions on the CN=Deleted Objects Container,” on page 97](#).

2.5 Becoming Familiar with Driver Features

This section discusses driver features you should become familiar with before deploying the Active Directory driver.

- ◆ [Section 2.5.1, “Multivalued Attributes,” on page 26](#)
- ◆ [Section 2.5.2, “Managing Account Settings Using Custom Boolean Attributes,” on page 27](#)
- ◆ [Section 2.5.3, “Provisioning Exchange Mailboxes using the homeMDB Attribute,” on page 28](#)
- ◆ [Section 2.5.4, “Expiring Accounts in Active Directory,” on page 28](#)
- ◆ [Section 2.5.5, “Retaining eDirectory Objects When You Restore Active Directory Objects,” on page 28](#)

2.5.1 Multivalued Attributes

The way the Active Directory driver handles multivalued attributes has changed from version 2.

Version 2 treated multivalued attributes as single-valued on the Subscriber channel by ignoring all but the first change value in an Add or Modify operation. Version 3 of the Active Directory Driver fully supports multivalued attributes.

However, when the Active Directory driver synchronizes a multivalued attribute with a single-valued attribute, the multivalued attribute is treated as single-valued. For example, the Telephone Number attribute is single-valued in Active Directory, and multivalued in the Identity Vault. When this attribute is synchronized from Active Directory, only a single value is stored in the Identity Vault.

This creates true synchronization and mapping between the two attributes, but can result in a potential loss of data if you have multiple values in an attribute that is mapped to an attribute with a single value. In most cases, a policy can be implemented to preserve the extra values in another location if required in your environment.

2.5.2 Managing Account Settings Using Custom Boolean Attributes

The Active Directory attribute `userAccountControl` is an integer whose bits control logon account properties, such as whether logon is allowed, passwords are required, or the account is locked. Synchronizing the Boolean properties individually is problematic because each property is embedded in the integer value.

In version 2, the Active Directory driver took a shortcut that let you map `userAccountControl` to the `eDirectory Login Disabled` attribute, but didn't let you map the other property bits within the attribute.

In version 3, each bit within the `userAccountControl` attribute can be referenced individually as a Boolean value, or `userAccountControl` can be managed in-total as an integer. The driver recognizes a Boolean alias to each bit within `userAccountControl`. These alias values are included in the schema for any class that includes `userAccountControl`. The alias values are accepted on the Subscriber channel and are presented on the Publisher channel.

The advantage to this feature is that because each bit can be used as a Boolean, the bit can be enabled individually in the Publisher filter and accessed easily. You can also put `userAccountControl` into the Publisher filter to receive change notification as an integer.

The integer and alias versions of `userAccountControl` should not be mixed in a single configuration.

The following table lists available aliases and hexadecimal values. Read-only attributes cannot be set on the Subscriber channel.

Table 2-3 *Aliases and Hexadecimal Values*

Alias	Hexadecimal	Notes
<code>dirxml-uACDontExpirePassword</code>	0x10000	Read-write
<code>dirxml-uACHomedirRequired</code>	0x0008	Read-write
<code>dirxml-uACInterdomainTrustAccount</code>	0x0800	Read-only
<code>dirxml-uACNormalAccount</code>	0x0200	Read-only
<code>dirxml-uACServerTrustAccount</code>	0x2000	Read-only
<code>dirxml-uACWorkstationTrustAccount</code>	0x1000	Read-only
<code>dirxml-uACAccountDisable</code>	0x0002	Read-write
<code>dirxml-uACPasswordNotRequired</code>	0x0020	Read-write

For troubleshooting tips relating to the `userAccountControl` attribute, see [Section 8.9, “Active Directory Account Disabled after a User Add on the Subscriber Channel,”](#) on page 92.

2.5.3 Provisioning Exchange Mailboxes using the homeMDB Attribute

Options for provisioning Exchange 2000 and Exchange 2003 mailboxes have changed from version 2.

In Version 2, Exchange provisioning was accomplished by setting attributes on User objects. A Microsoft program (the Recipient Update Service) used this information to provision the Exchange database.

This method still works in version 3 of the Active Directory Driver, but a new method (CDOEXM) has been added. With CDOEXM enabled, an Exchange mailbox is provisioned by setting the homeMDB attribute. When the homeMDB attribute is set, the driver automatically sets all required attributes.

The homeMDB attribute is set during initial configuration, but you can change the setting by modifying the driver policy. For a discussion of this parameter, see [Section 4.3, “Configuration Parameters,”](#) on page 40.

2.5.4 Expiring Accounts in Active Directory

If you map the eDirectory attribute of Login Expiration Time to the Active Directory attribute of accountExpires, the account in Active Directory expire a day earlier than the time set in eDirectory.

This happens because Active Directory sets the value of the accountExpires attribute in full-day increments. The eDirectory attribute of Login Expiration Time uses a specific day and time to expire the account.

For example, if you set an account in eDirectory, to expire on July 15th, 2006, at 5:00 p.m., the last full day this account is valid in Active Directory is July 14th.

If you set the account in the Microsoft Management Console, to expire on July 15th, 2006, the eDirectory attribute of Login Expiration Time is set to expire on July 16th, 2006 at 12:00 a.m. Because the Microsoft Management Console doesn't allow for a value of time to be set, the default is 12:00 a.m.

The driver uses the most restrictive settings. You can add an additional day to the expiration time in Microsoft depending upon what your requirements are.

2.5.5 Retaining eDirectory Objects When You Restore Active Directory Objects

Any Active Directory objects that are restored through the Active Directory tools delete the associated eDirectory object when the objects are synchronized. The Active Directory driver looks for a change in the isDeleted attribute on the Active Directory object. When the driver detects a change in this attribute, a delete event is issued through the driver for the object associated with the Active Directory object.

If you don't want eDirectory objects deleted, you must add an additional policy to the Active Directory driver. Identity Manager 3.0.1 comes with a predefined rule that changes all Delete events into Remove Association events. For more information, see [“Command Transformation - Publisher Delete to Disable”](#) in the *Policy Builder and Driver Customization Guide*.

Installing the Active Directory Driver

3

- ◆ Section 3.1, “Basic Steps,” on page 29
- ◆ Section 3.2, “Installing the Active Directory Driver Shim,” on page 30
- ◆ Section 3.3, “Installing Preconfiguration Import Files,” on page 35
- ◆ Section 3.4, “Installing the Active Directory Discovery Tool,” on page 36

3.1 Basic Steps

The following figure illustrates options that you can select when installing Identity Manager.

Figure 3-1 Identity Manager Installation Options

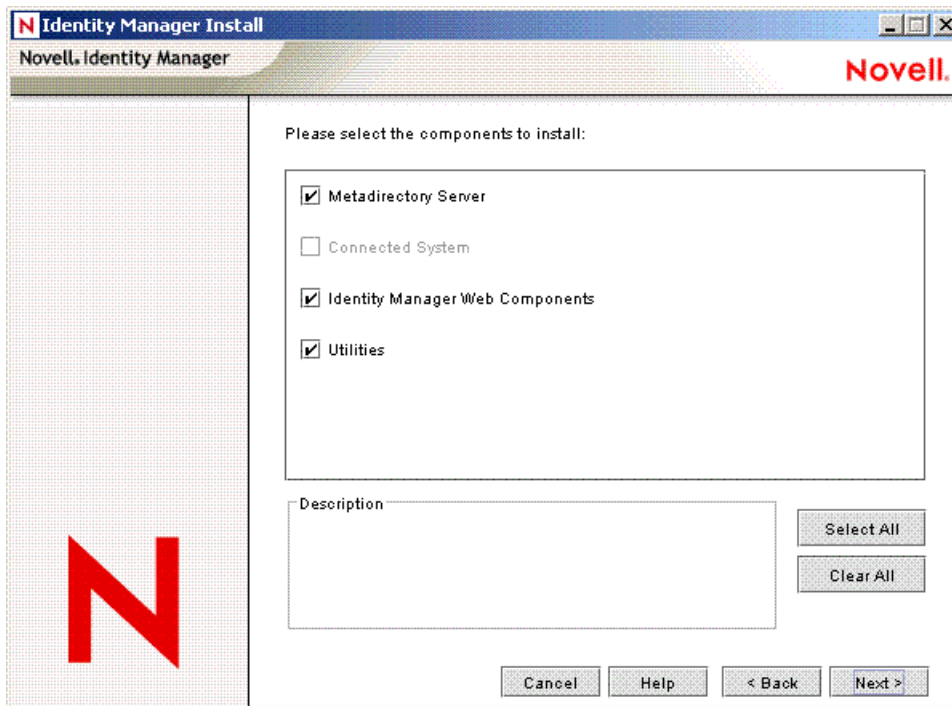


Table 3-1 Identity Manager Installation Options

Option	Description
Metadirectory Server	Installs the Metadirectory engine and Identity Manager
Connected System	Installs the Remote Loader
Identity Manager Web Components	Installs the preconfigured (example) driver configuration file

Option	Description
Utilities	Installs the Active Directory Discovery Tool

Installing the Active Directory driver shim requires three basic steps:

Table 3-2 *Installation Steps*

Step	What to Select During Installation
1. Install the Active Directory driver shim on the Metadirectory engine server or the Remote Loader server.	Select the Metadirectory Server or Identity Manager Connected System option. See Section 3.2, “Installing the Active Directory Driver Shim,” on page 30.
2. Install the preconfiguration import file for Active Directory on the iManager server.	Select the Identity Manager Web Components option. See Section 3.3, “Installing Preconfiguration Import Files,” on page 35.
3. Install the Active Directory Discovery Tool on a workstation used to configure Identity Manager.	Select the Utilities option. See Section 3.4, “Installing the Active Directory Discovery Tool,” on page 36.

Typically, you install the Active Directory driver components when you install the Metadirectory server (or Remote Loader) and Web components. However, you can install them later.

3.2 Installing the Active Directory Driver Shim

- ◆ [Section 3.2.1, “Installing the Shim on a Metadirectory Server,”](#) on page 30
- ◆ [Section 3.2.2, “Installing the Shim on a Remote Loader,”](#) on page 33

3.2.1 Installing the Shim on a Metadirectory Server

- 1 On the server where the Identity Vault and the Metadirectory engine are running, launch the Identity Manager installation.

Run the installation program from the Identity Manager CD or the download image.

- 2 In the Welcome dialog box, click *Next*, then accept the license agreement.
- 3 In the first Identity Manager Overview dialog box, review the information, then click *Next*.

The dialog box provides information on the following:

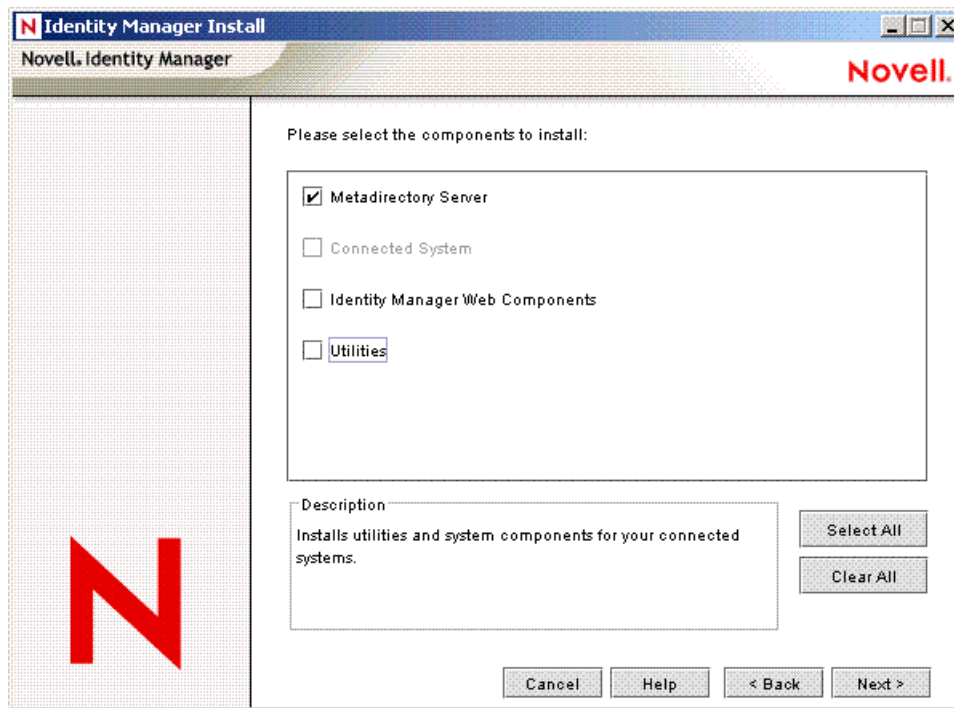
- ◆ A Metadirectory Server
- ◆ A Connected System Server

- 4 In the second Identity Manager Overview dialog box, review the information, then click *Next*.

The dialog box provides information on the following:

- ◆ A Web-based Administration Server
- ◆ Utilities

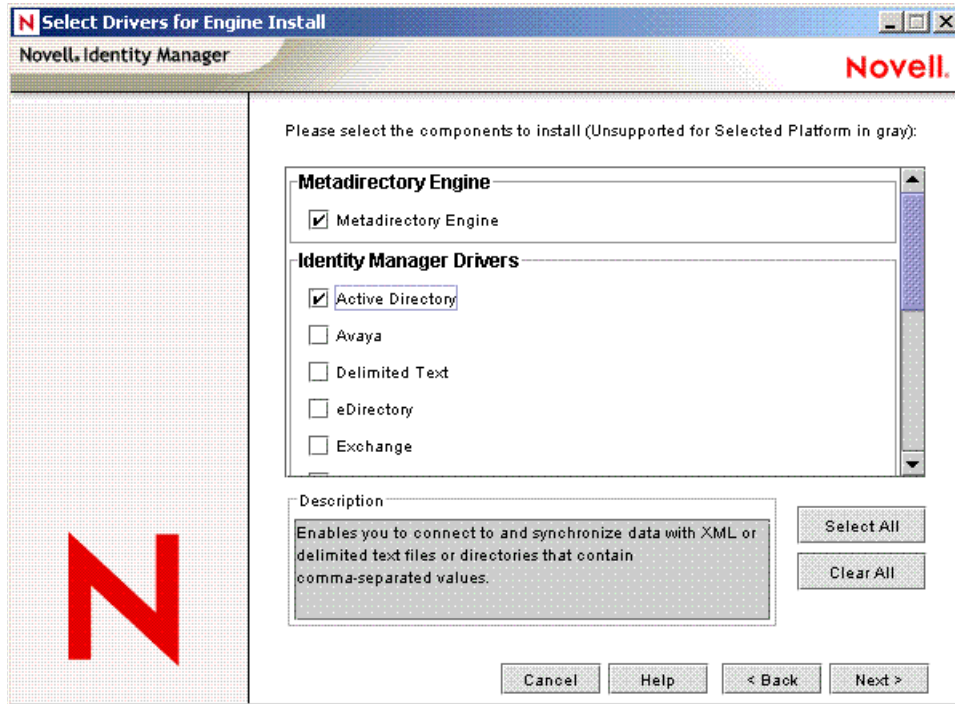
- 5 In the Please Select the Components to Install dialog box, select *Metadirectory Server*, then click *Next*.



If iManager is already installed on this machine, and if you prefer to install the iManager plug-ins and configuration files at this time, also select *Identity Manager Web Components*.

If you prefer to install the Active Directory Management tool at this time, also select *Utilities*.

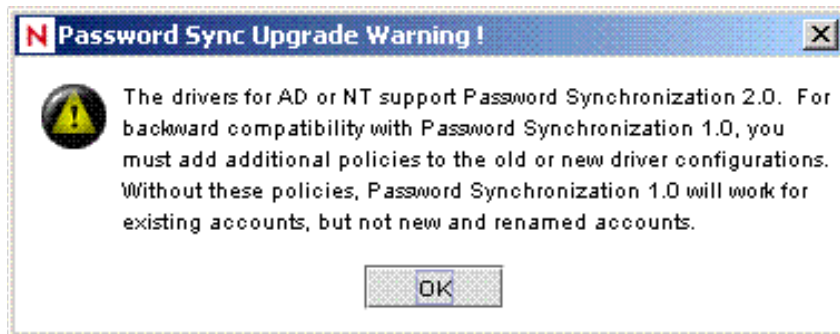
- 6 In the Select Drivers for Engine Install dialog box, select *Metadirectory Engine*, select *Active Directory*, then click *Next*.



- 7 In the Identity Manager Upgrade Warning dialog box, click *OK*.



- 8 In the Password Sync Upgrade Warning dialog box, click *OK*.



- 9 In the Schema Extension dialog box, type a username and password, then click *Next*.
- 10 Review the selected options, then click *Finish*.

3.2.2 Installing the Shim on a Remote Loader

This option enables you to install the Active Directory driver shim to run on a server that is separate from the server running the Metadirectory engine.

- 1 On the server where the Remote Loader is running, launch the Identity Manager installation.

Run the installation program from the Identity Manager CD or the download image.

- 2 In the Welcome dialog box, click *Next*, then accept the license agreement.
- 3 In the first Identity Manager Overview dialog box, review the information, then click *Next*.

The dialog box provides information on the following:

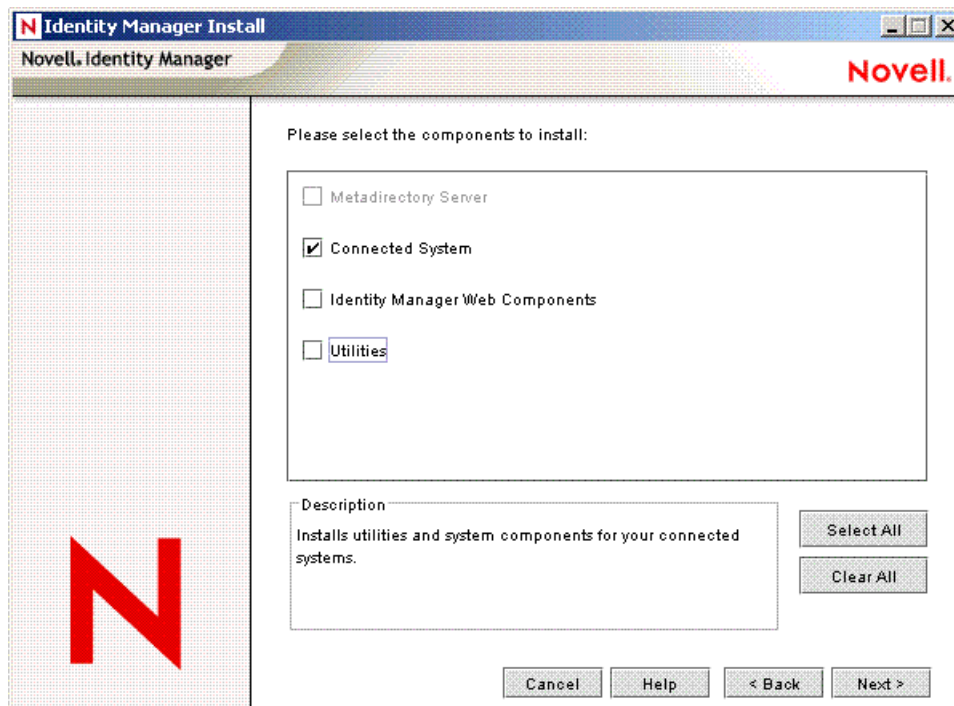
- ♦ A Metadirectory Server
- ♦ A Connected System Server

- 4 In the second Identity Manager Overview dialog box, review the information, then click *Next*.

The dialog box provides information on the following:

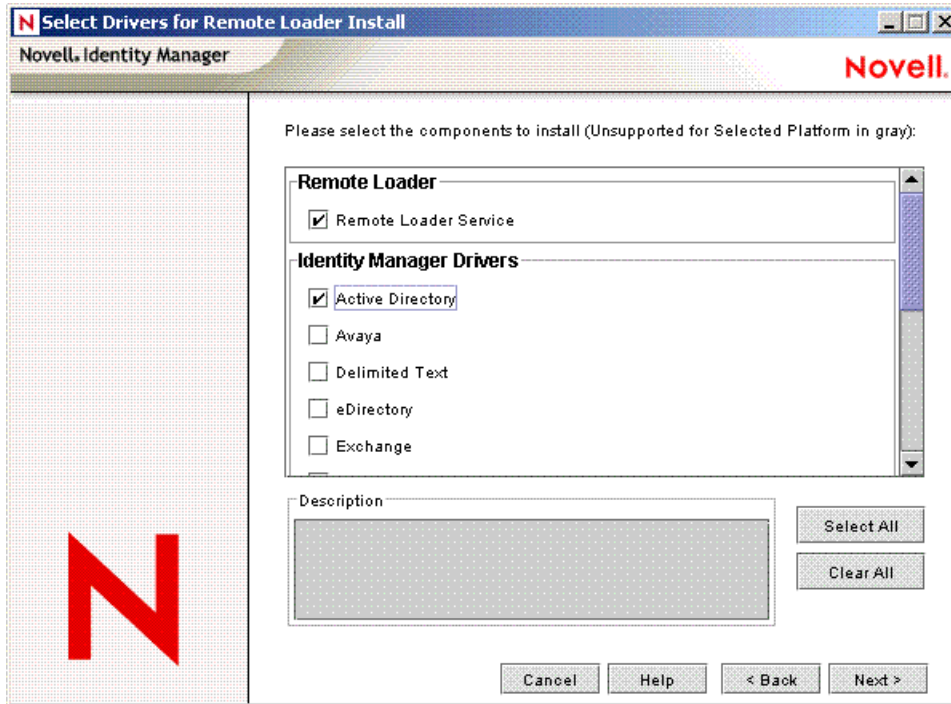
- ♦ A Web-based Administration Server
- ♦ Utilities

- 5 In the Please Select the Components to Install dialog box, deselect *Metadirectory Server* and other options, select *Identity Manager Connected System*, then click *Next*.



- 6 Specify the installation path, then click *Next*.

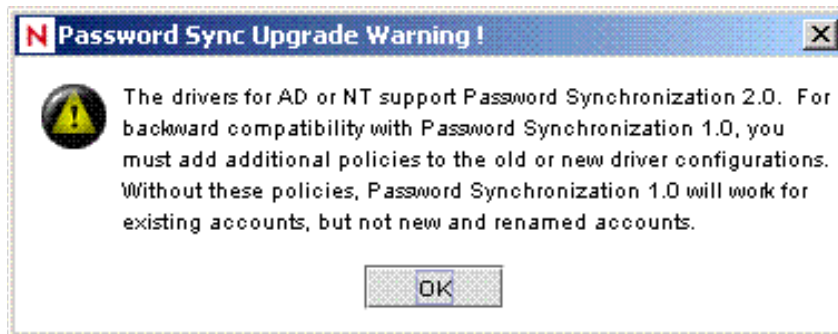
- 7 In the Select Drivers for Engine Install dialog box, select *Remote Loader Service*, select *Active Directory*, then click *Next*.



- 8 In the Identity Manager Upgrade Warning dialog box, click *OK*.



- 9 In the Password Sync Upgrade Warning dialog box, click *OK*.



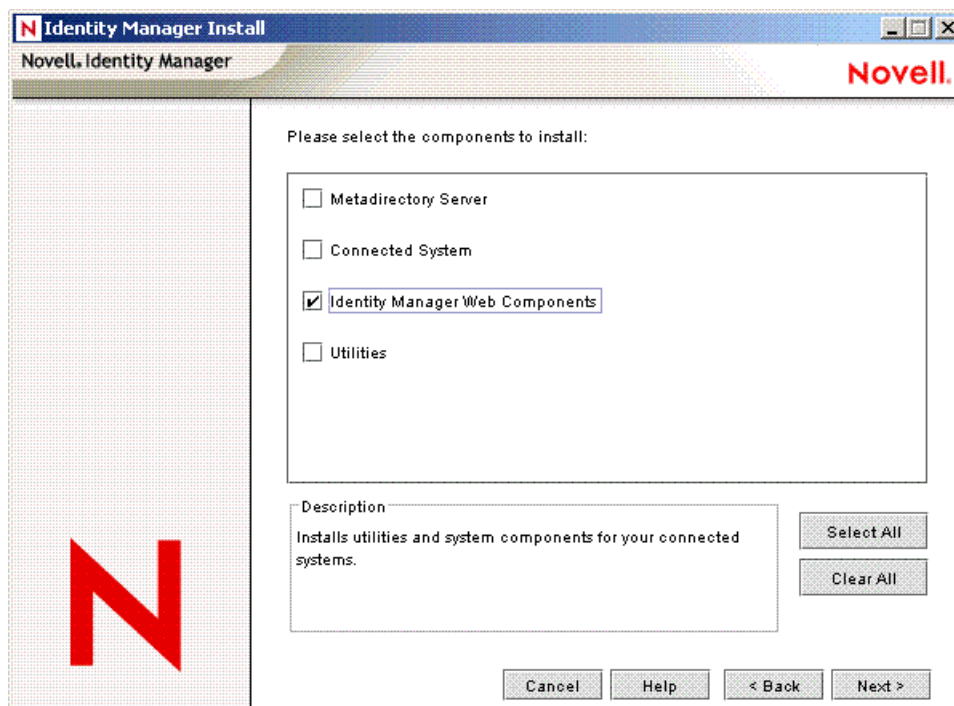
10 Review the selected options, then click *Finish*.

3.3 Installing Preconfiguration Import Files

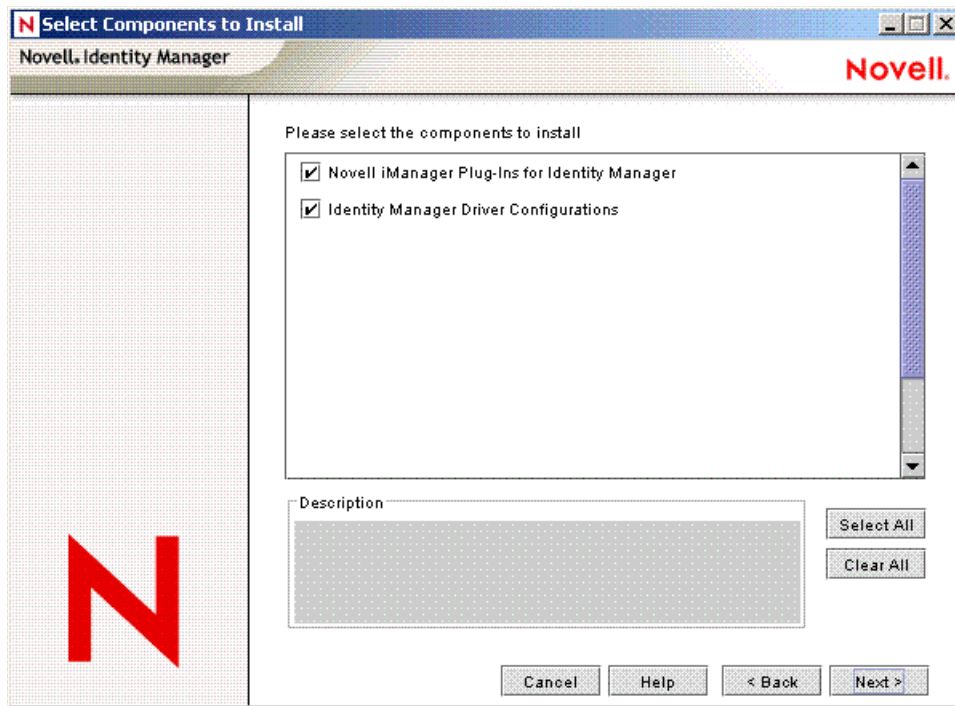
This option installs the plug-ins to Identity Manager and the preconfigured (example) driver configurations. After installing the files, you use iManager to import the Active Directory preconfigured file into a driver set and configure the driver.

You might have already installed these files, when you installed the Metadirectory engine or Remote Loader. To install the files separately:

- 1 On the server where iManager is installed, launch the Identity Manager installation.
- 2 In the Welcome dialog box, click *Next*, then accept the license agreement.
- 3 In the two Identity Manager Overview dialog boxes, review the information, then click *Next*.
- 4 In the Please Select the Components to Install dialog box, deselect all options except *Identity Manager Web Components*, then click *Next*.



- 5 Select *Identity Manager Driver Configurations*, then click *Next*.



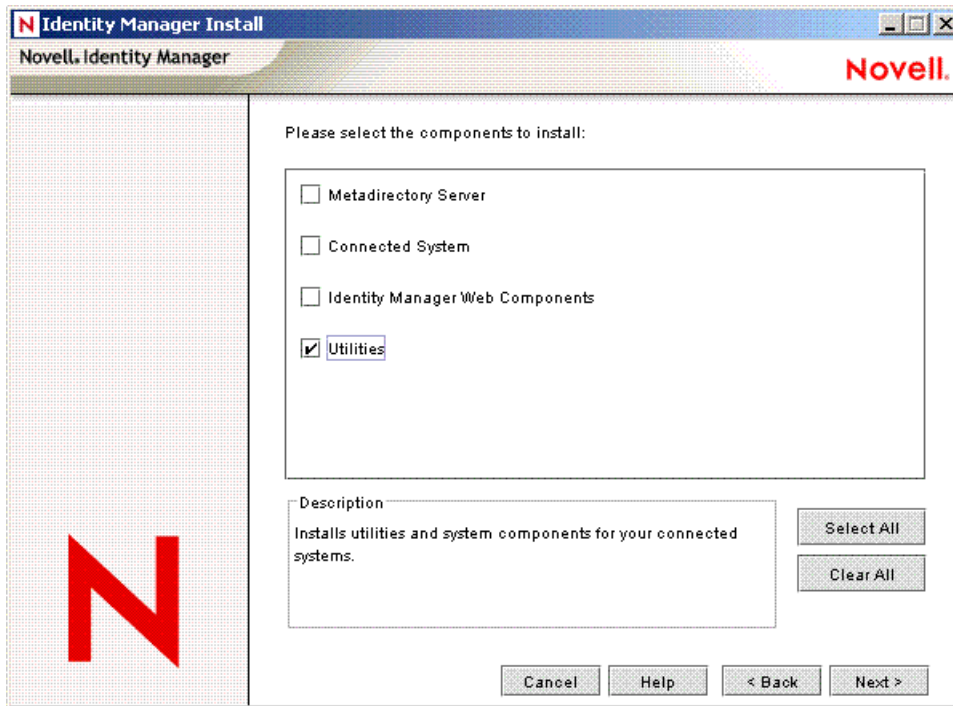
You can install the driver configuration files when you install the Novell iManager plug-ins, or you can install the files separately.

- 6 Review the selected options, then click *Finish*.

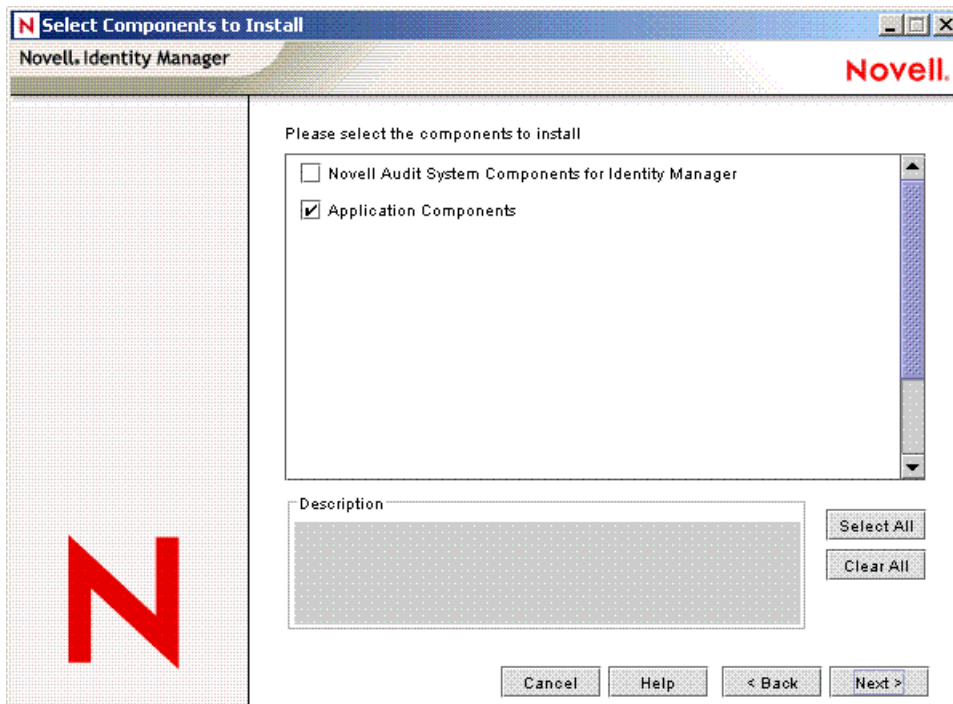
3.4 Installing the Active Directory Discovery Tool

- 1 On the workstation that you use to configure Active Directory, launch the Identity Manager installation.
- 2 In the Welcome dialog box, click *Next*, then accept the license agreement.
- 3 In the two Identity Manager Overview dialog boxes, review the information, then click *Next*.

- 4 In the Please Select the Components to Install dialog box, deselect all options except *Utilities*, then click *Next*.



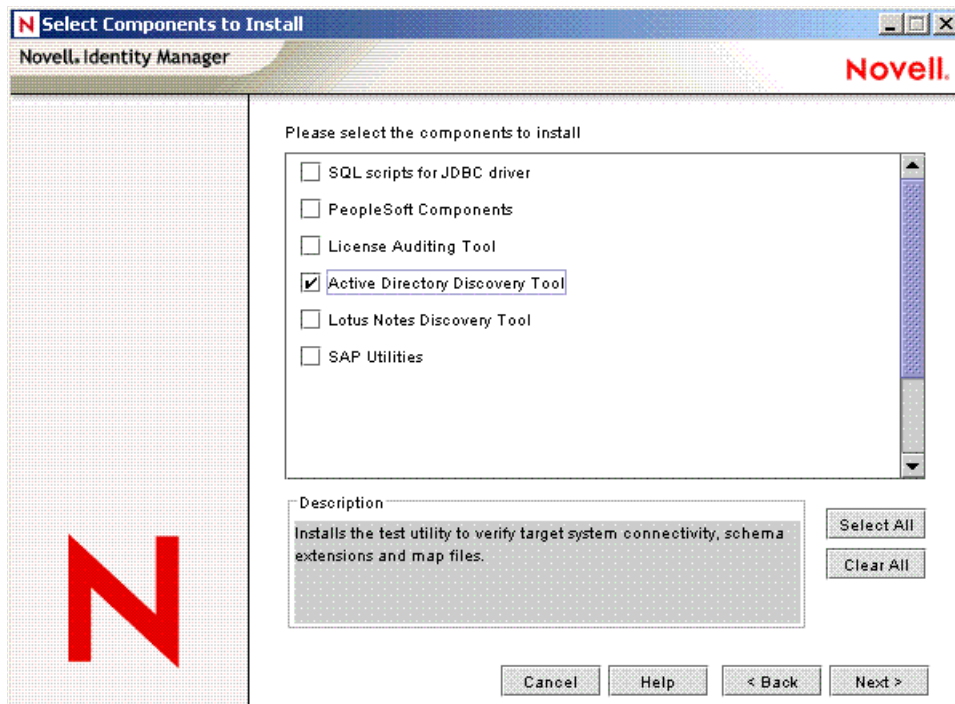
- 5 Select *Application Components*, then click *Next*.



Deselect Novell Audit System Components for Identity Manager.

- 6 Specify the installation path, then click *Next*.

7 Select only *Active Directory Discovery Tool*, then click *Next*.



8 Review the selected options, then click *Finish*.

Configuring the Active Directory Driver

4

In Novell® iManager, the Create Driver Wizard helps you import a basic driver configuration for Active Directory. This wizard creates and configures the objects needed to make the driver work properly. For details on using this wizard, see *Creating and Configuring a Driver* in the *Novell Identity Manager 3.0.1 Administration Guide*.

In this section:

- ♦ “Importing the Driver Configuration File in iManager” on page 39
- ♦ “Configuration Parameters” on page 40

4.1 Importing the Driver Configuration File in Designer

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

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

- 1 Open a project in Designer and in the modeler, right-click on the Driver Set object and select *Add Connected Application*.
- 2 From the drop-down list, select *ActiveDirectory.xml*, then click *Run*.
- 3 Click *Yes*, in the Perform Prompt Validation window. It has you fill in all of the fields to correctly configure the Active Directory driver.
- 4 Configure the driver by filling in the fields. Specify information specific to your environment. For information on the settings, see [Section 4.3, “Configuration Parameters,” on page 40](#) for more information.
- 5 After specifying parameters, click *OK* to import the driver.
- 6 After the driver is imported, customize and test the driver.
- 7 Once the driver is fully tested, deploy the driver into the Identity Vault. See “[Deploying a Driver to an Identity Vault](#)” in the *Designer for Identity Manager 3: Administration Guide*.

4.2 Importing the Driver Configuration File in iManager

The Active Directory 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 Drivers*.

- 2 Select a driver set, then click *Next*.

Where do you want to place the new drivers?

- In an existing driver set
- In a new driver set

hraun_set.DigitalAirlines

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

- 3 Select the *Active Directory* driver, then click *Next*.



- 4 Configure the driver by filling in the configuration parameters. For information on the settings, see [Section 4.3, “Configuration Parameters,” on page 40](#).

- 5 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.

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

Exclude the security-equivalence object (for example, DriversUser) that you specified in Step 2. 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.

- 7 Click Finish.

4.3 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 4-1 Configuration Parameters

Field	Description
<i>Driver name</i>	The eDirectory™ object name to be assigned to this driver. Because each Active Directory domain requires a separate driver, you should include the domain name in the driver name. When you look at the driver, you will know which domain it is associated with.

Field	Description
<i>Authentication Method</i>	<p>The method to authenticate with Active Directory.</p> <p><i>Negotiate</i> is the preferred method. Select Negotiate to use the Microsoft security package to negotiate authentication. To use Negotiate, the server hosting the driver must be a member of the domain.</p> <p>If you plan to use password synchronization and are running on a member server, you need SSL.</p> <p><i>Simple</i> uses an LDAP simple bind. If you select Simple, SSL is recommended.</p> <hr/> <p>IMPORTANT: Simple bind doesn't support password synchronization or Exchange provisioning.</p>
<i>Authentication Id</i>	<p>An Active Directory account with administrative privileges to be used by Identity Manager. The name form used depends on the selected authentication mechanism.</p> <p>For Negotiate, provide the name form required by your Active Directory authentication mechanism. For example:</p> <ul style="list-style-type: none"> ◆ Administrator - AD Logon Name ◆ Domain/Administrator - Domain qualified AD Logon Name <p>For Simple, provide an LDAP ID. For example:</p> <ul style="list-style-type: none"> ◆ cn=DirXML,cn=Users,DC=domain,dc=com
<i>Authentication Password</i>	The password for the user account specified in Authentication ID.
<i>Authentication Context</i>	<p>The name of the Active Directory domain controller to use for synchronization.</p> <p>For example, for the Negotiate authentication method, use the DNS name mycontroller.domain.com. For the Simple authentication method, you can use the IP address of your server (for example, 10.10.128.23 or the DNS name).</p> <p>If no value is specified, localhost is used.</p> <hr/> <p>NOTE: This value is stored in the Authentication Context attribute. To change this value after the initial configuration, modify this attribute as explained in "Security Parameters" on page 59.</p>
<i>Domain Name</i>	<p>The Active Directory domain managed by this driver.</p> <p>The driver requires LDAP formatted domain names dc=domain,dc=com</p>
<i>Domain DNS Name</i>	<p>The DNS name of the Active Directory domain managed by this driver.</p> <p>The driver requires DNS formatted domain names domain.com</p>

Field	Description
<i>Driver Polling Interval</i>	<p>The Identity Vault sends changes to Active Directory as they happen. However, changes to Active Directory are sent to the Identity Vault only as often as the configured polling interval. The default is 1 minute.</p> <hr/> <p>IMPORTANT: The polling interval affects system performance. A low polling interval results in frequent searches and fast updates of data. A high polling interval results in periodic bursts of traffic. Although a low polling interval has a greater overall cost, the cost is spread more evenly over time.</p> <p>If you set the interval to 0 (zero), you get a ten-second poll rate.</p>
<i>Password Sync Timeout (minutes)</i>	<p>The number of minutes the driver attempts to synchronize a password.</p> <p>Set the value large enough to handle whatever temporary backlog of passwords exists. If you are doing bulk changes, set the timeout large enough to handle all the changes. The rule of thumb is to allow one second per password. For example, to synchronize 18,000 passwords, allow 300 minutes (18,000 passwords divided by 60 seconds).</p> <p>A setting of -1 is indefinite. Although this setting can handle bulk changes, it can cause problems. For example, a password might never be able to be synchronized because the account wasn't associated. Such a password would therefore remain in the system forever. A number of similar situations could result in a large inventory of unsynchronized passwords held by the system.</p> <p>You must set the password sync timeout to at least three times the polling interval.</p>
<i>Driver is Local/Remote</i>	<p>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.</p>
<i>Remote Host Name and Port</i>	<p>Remote option only.</p> <p>The host name 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.</p> <p>This setting displays only if you set Driver is Local/Remote to Remote.</p>
<i>Driver Password</i>	<p>Remote option only.</p> <p>The Remote Loader uses the Driver Object Password to authenticate itself to the Identity Manager server. The password must be the same password that is specified as the Driver object password on the Remote Loader.</p> <p>This setting displays only if you set Driver is Local/Remote to Remote.</p>
<i>Remote Password</i>	<p>Remote option only.</p> <p>The Remote Loader password is used to control access to the Remote Loader instance. The password must be the same password that is specified as the Remote Loader password on the Remote Loader.</p> <p>This setting displays only if you set Driver is Local/Remote to Remote.</p>
<i>Import will proceed to driver policy selections</i>	<p>Remote option only.</p> <p>OK If you click the driver wizard continues on with the configuration of the policies for the driver.</p>

Field	Description
<i>Base container in eDirectory</i>	<p>Specify the base container in the Identity Vault for synchronization. This container is used in the Subscriber Matching policies to limit the Identity Vault objects being synchronized and in the Publisher Placement policies when adding objects to the Identity Vault.</p> <p>New users are placed in this container by default. Use the dot format. For example,</p> <pre>users.myorg</pre> <p>If the container doesn't exist, you must create it and make sure it is associated with the Active Directory base container before trying to add users to this container.</p>
<i>Publisher Placement</i>	<p><i>Mirrored</i> places objects hierarchically within the base container.</p> <p><i>Flat</i> places objects strictly within the base container.</p> <p>This selection builds the default Publisher Placement policies.</p> <hr/> <p>NOTE: If you select <i>Mirrored</i>, the driver assumes the structure of the eDirectory database is the same in Active Directory from the eDirectory base container. If the structure is not the same, the objects are not placed properly. Create the same structure in Active Directory that exists in eDirectory, or migrate the eDirectory containers before migrating User objects.</p>
<i>Base container in Active Directory</i>	<p>Specify the base container in Active Directory, in LDAP format. New users are placed in this container by default. For example,</p> <pre>CN=Users,DC=MyDomain,DC=com</pre> <p>If the target container doesn't exist, you must create it and make sure it is associated with the eDirectory base container before trying to add users to this container.</p> <p>If you are creating or using a container other than Users in Active Directory, the container is an OU, not a CN. For example,</p> <pre>OU=Sales,OU=South,DC=MyDomain,DC=com</pre>
<i>Active Directory Placement</i>	<p><i>Mirrored</i> places the objects hierarchically within the base container.</p> <p><i>Flat</i> places objects strictly within the base container.</p> <p>This selection builds the default Subscriber Placement policies.</p> <hr/> <p>NOTE: If you select <i>Mirrored</i>, the driver assumes the structure of the Active Directory database is the same in eDirectory from the Active Directory base container. If the structure is not the same, the objects are not placed properly. Create the same structure in eDirectory that exists in Active Directory, or migrate the Active Directory containers before migrating User objects.</p>

Field	Description
<i>Configure Data Flow</i>	<p>Configure Data Flow establishes the initial driver filter that controls the classes and attributes that will be synchronized. The purpose of this option is to configure the driver to best express your general data flow policy. It can be changed after import to reflect specific requirements.</p> <p><i>Bidirectional</i> sets classes and attributes to synchronize on both the Publisher and Subscriber channels. A change in either the Identity Vault or Active Directory is reflected on the other side. Use this option if you want both sides to be authoritative sources of data.</p> <p><i>AD to Vault</i> sets class and attributes to synchronize on the Publisher channel only. A change in Active Directory is reflected in the Identity Vault, but Identity Vault changes are ignored. Use this option if you want Active Directory to be the authoritative source of data.</p> <p><i>Vault to AD</i> sets classes and attributes to synchronize on the Subscriber channel only. A change in the Identity Vault is reflected in Active Directory, but Active Directory changes are ignored. Use this option if you want the vault to be the authoritative source of data.</p> <hr/> <p>WARNING: Delete, Move, and Rename events are independent of the filter. It does not matter which option you select, these events are processed by the driver. If you do not want these events to synchronize, you must change the default configuration of the driver.</p> <p>You can use one of the predefined policies that comes with Identity Manager 3.0.1 to change Delete events into Remove Association events. For more information, see “Command Transformation - Publisher Delete to Disable” in the Policy Builder and Driver Customization Guide.</p> <p>To block Move and Rename events, you must customize the driver.</p> <hr/>
<i>Password Failure Notification User</i>	<p>Password synchronization policies are configured to send e-mail notifications to the associated user when password updates fail. You have the option of sending a copy of the notification e-mail to another user, such as a security administrator. If you want to send a copy, enter or browse for the DN of that user. Otherwise, leave this field blank.</p>
<i>Configure Entitlements</i>	<p>The driver can be configured to use Entitlements to manage user accounts and group memberships in Active Directory and to provision Exchange mailboxes. When using Entitlements, the driver works in conjunction with external services such as the Identity Manager User Application or Role-Based Entitlements to control the conditions under which these features are provisioned or de-provisioned in Active Directory. See “Entitlements” on page 18 for more information.</p> <p>Select <i>Yes</i> if you plan to use one of these external services to control provisioning to Active Directory.</p> <p>Select <i>No</i> if you do not plan on using the Identity Manager User Application or provisioning Exchange mailboxes.</p> <hr/>

Field	Description
<i>User account policy</i>	<p>Configure Elements option only.</p> <p>User accounts in Active Directory can be controlled by synchronization or by using Entitlements with the Workflow service or Role-Based Entitlements.</p> <p><i>Entitlements</i> gives control of enabling accounts in Active Directory to the Entitlement in the Identity Vault.</p> <p><i>Implement in policy</i> uses the policies in the driver instead of Entitlements.</p>
<i>Exchange policy</i>	<p>Configure Elements option only.</p> <p>Exchange provisioning can be handled by driver policy, Entitlements, or skipped entirely. A user can be assigned a mailbox in Exchange (the user is mailbox enabled) or have information about a foreign mailbox stored in the Identity Vault record (the user is mail enabled). When using the driver policy, the decision to mailbox enable or mail enable a user, plus the Exchange message database where the account will reside, is controlled completely in the policy.</p> <p>When using <i>Entitlements</i>, an external service such as the Workflow service or Role-Based Entitlements makes these decisions and driver policy simply applies them.</p> <p><i>Implement in policy</i> uses the policies in the driver instead of Entitlements to assign Exchange mailboxes.</p> <p>When <i>None</i> is selected, the default configuration does not create Exchange mailboxes but does synchronize the Identity Vault Internet E-Mail Address with the Active Directory mail attribute.</p>
<i>Group membership policy</i>	<p>Configure Elements option only.</p> <p>Group membership in Active Directory can be controlled by synchronizing the membership list or by using Entitlements.</p> <p><i>Entitlements</i> use the Workflow service or the Role-Based Entitlements to assign group membership.</p> <p><i>Synchronize</i> uses policies to synchronize the group membership list.</p> <p><i>None</i> does not synchronize group membership information.</p>
<i>Use CDOEXM for Exchange (yes/no)</i>	<p>Exchange Policy option only.</p> <p>Exchange mailboxes can be controlled by calls into the Microsoft Exchange management system instead of regular attribute synchronization. When enabled, the driver shim intercepts changes to the Active Directory homeMDB attribute and calls into the CDOEXM (Collaboration Data Objects for Exchange Management) subsystem.</p> <p>The value you choose here is recorded in the driver shim configuration.</p> <p>Yes synchronizes Exchange mailboxes.</p> <p>No does not synchronize Exchange mailboxes.</p>

Field	Description
<i>Allow CDOEXM Exchange mailbox move (yes/no)</i>	<p>Exchange Policy option only.</p> <p>When enabled, the driver shim intercepts modifications to the Active Directory homeMDB attribute and calls into CDOEXM to move the mailbox to the new message data store.</p> <p>Yes moves the Exchange mailbox.</p> <p>No does not move the Exchange mailbox.</p>
<i>Allow CDOEXM Exchange mailbox delete (yes/no)</i>	<p>Exchange Policy option only.</p> <p>When enabled, the driver shim intercepts removal for the Active Directory homeMDB attribute and calls into CDOEXM to delete the mailbox.</p> <p>Yes allows the Exchange mailbox to be deleted.</p> <p>No does not allow the Exchange mailbox to be deleted.</p>
<i>Default Exchange MDB</i>	<p>Exchange Policy > Implement in policy option only.</p> <p>Enter the default Exchange Message Database (MDB). For example,</p> <pre>[CN=Mailbox Store (CONTROLLER),CN=First Storage Group,CN=InformationStore,CN=CONTROLLER,CN=Servers,CN=First Administrative Group,CN=Administrative Groups,CN=Domain,CN=Microsoft Exchange,CN=Services,CN=Configuration,DC=Domain,DC=com]</pre> <p>The driver can be updated to manage additional MDBs after the import is complete.</p>
<i>When account entitlement revoked</i>	<p>Exchange Policy option only.</p> <p>Allows you to choose what action is taken when a User account is removed by Entitlements.</p> <p><i>Disable Account</i></p> <p><i>Delete Account</i></p>
<i>Name mapping policy selection</i>	<p>The driver maps the Identity Vault Full Name attribute to the Active Directory object name and maps the Active Directory Pre-windows 2000 logon name to the Identity Vault user name.</p> <p>You can accept the full policy or manually select parts of the policy. If the policy does not meet your needs, you can modify the policies after import by editing the NameMap policies in the Subscriber and Publisher Command Transformation policies after the import completes.</p> <p><i>Accept</i> uses the full policy.</p> <p><i>Manual</i> allows you to use part of the policy.</p>

Field	Description
<i>Full Name Mapping</i>	<p>Name mapping policy selection > Manual option only.</p> <p><i>Yes</i> allows the driver to keep the Identity Vault Full Name attribute synchronized with the Active Directory object name and display name.</p> <p><i>No</i> does not keep the Identity Vault Full Name attribute synchronized with the Active Directory object name and display name.</p> <p>This policy is useful when creating user accounts in Active Directory using the Microsoft Management Console Users and Computers snap-in.</p>
<i>Logon Name Mapping</i>	<p>Name mapping policy selection > Manual option only.</p> <p><i>Yes</i> allows the driver to keep the Identity Vault object name synchronized with the Active Directory Pre-Windows 2000 Logon Name (also known as the NT Logon Name and the sAMAccountName).</p> <p><i>No</i> does not keep the Identity Vault object name synchronized with the Active Directory Pre-Windows 2000 Logon Name.</p>
<i>Import will proceed to Windows 2000 logon name policy selections</i>	<p>Name mapping policy selection > Manual option only.</p> <p>OK</p>
<i>User Principal Name Mapping</i>	<p>Allows you to choose a method for managing the Active Directory Windows 2000 Logon Name (also known as the userPrincipalName). userPrincipalName takes the form of an e-mail address, as in <code>usere@domain.com</code>. Although the shim can place any value into userPrincipalName, it will not be useful as a logon name unless the domain is configured to accept the domain name used with the name.</p> <p><i>Follow Active Directory e-mail address</i> sets userPrincipalName to the value of the Active Directory mail attribute. This option is useful when you want the user's e-mail address to be used for authentication and Active Directory is authoritative for e-mail addresses.</p> <p><i>Follow Identity Vault e-mail address</i> sets userPrincipalName to the value of the Identity Vault e-mail address attribute. This option is useful when you want the user's e-mail address to be used for authentication and the Identity Vault is authoritative for e-mail addresses.</p> <p><i>Follow Identity Vault name</i> is useful when you want to generate userPrincipalName from the user logon name plus a hard-coded string defined in the policy.</p> <p><i>None</i> is useful when you do not want to control userPrincipalName or want to implement your own policy.</p>

Upgrading the Active Directory Driver

5

- ♦ [Section 5.1, “Checklist for Upgrading,” on page 49](#)
- ♦ [Section 5.2, “Addressing the Login Disabled Value,” on page 50](#)

5.1 Checklist for Upgrading

To upgrade the Active Directory driver, use the following checklist. If you are not an expert with Identity Manager, you might want to engage a capable consultant.

- ❑ To use Password Synchronization 2.0, add the driver manifest and password policies.

See [Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization](http://www.novell.com/documentation/dirxml20/index.html?page=/documentation/dirxml20/admin/data/bo16oyy.html) (<http://www.novell.com/documentation/dirxml20/index.html?page=/documentation/dirxml20/admin/data/bo16oyy.html>).

- ❑ For continued use of Password Synchronization 1.0, add legacy polices to the existing driver configuration.

See [Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager](http://www.novell.com/documentation/dirxmldrivers/index.html?page=/documentation/dirxmldrivers/ad/data/bnwjt02.html) (<http://www.novell.com/documentation/dirxmldrivers/index.html?page=/documentation/dirxmldrivers/ad/data/bnwjt02.html>).

- ❑ Remove the structured formatting of the sAMAccountName in the existing driver’s style sheets.

sAMAccountName was a structured attribute in the DirXML[®] 1.1a Active Directory 2.0 driver. In the new Active Directory 3.0 driver, it is a string.

Old format:

```
<value type="structured">
  <component name="nameSpace">0</component>
  <component association-ref="XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX"
name="volume"/>
  <component name="path">jsmith</component>
</value>
```

New format:

```
<add-attr attr-name="sAMAccountName">
  <value type="string">jsmith</value>
</add-attr>
```

- ❑ Upgrade driver configuration parameters.

We recommend the use of the following settings by default:

```
<?xml version="1.0"?>
<driver-config name="Active Directory Driver">
  <driver-options>
    <pollingInterval display-name="Polling Interval (min.)">
1</pollingInterval>
    <auth-method display-name="Authentication Method">
Negotiate</auth-method>
```



```

        <signing display-name="Use Signing (yes/no)" id=">
no</signing>
        <sealing display-name="Use Sealing (yes/no)">
no</sealing>
        <use-ssl display-name="Use SSL (yes/no)">
no</use-ssl>
        <pub-heartbeat-interval display-name="Heart Beat">
0</pub-heartbeat-interval>
        <pub-password-expire-time display-name="Password Sync
Timeout
(minutes) : ">60</pub-password-expire-time>
        <use-CDOEXM display-name="Use CDOEXM for Exchange (yes/no)">
no</use-CDOEXM>
        <cdoexm-move display-name="Allow CDOEXM Exchange mailbox move
(yes/no)">yes</cdoexm-move>
        <cdoexm-delete display-name="Allow CDOEXM Exchange mailbox
delete (yes/no)">yes</cdoexm-delete>
        </driver-options>
</driver-config>

```

- ❑ Convert the authentication ID to either the Sam Account Name (for example, jsmith) or the domain name/account name format (for example, *domain/jsmith*).
- ❑ Change the mapping of the Login Disabled attribute from userAccountControl to dirxml-uACAccountDisable.
- ❑ If you are provisioning Exchange accounts, change the driver parameter for CDOEXM to Yes, then remove the following four hard-coded attributes from your existing driver configuration style sheets:
 - ◆ msExchHomeServerName
 - ◆ legacyExchangeDN
 - ◆ homeMTA
 - ◆ msExchMailboxSecurityDescriptor
- ❑ If you are upgrading from Identity Manager 2.x and you have Exchange provisioning enabled, an overlay has to be applied to the driver. Identity Manager 3.0.1 controls moves and deletes with the Exchange mailboxes. For this to function on an upgraded driver, the overlay has to be applied. See [Section 5.5, “Applying the Overlay for Exchange Mailboxes,” on page 52](#) for information on how to apply the overlay.

5.2 Addressing the Login Disabled Value

eDirectory™ treats a lack of Login Disabled = true as being the same as Login Disabled = false. Therefore, if you install the version 3 Active Directory driver as a new installation (not an upgrade), and if a Login Disabled = false value isn’t present, a default policy on the Creation Rule synthesizes that value.

Upgrading from the version 2 driver to the version 3 driver doesn’t get this policy by default.

5.3 Upgrading the Driver Shim from DirXML 1.1a

The upgrade replaces the previous driver shim with the new driver shim but keeps the previous driver's configuration. The new driver shim can run the DirXML 1.1a configuration with no changes (unless you are using Password Synchronization 1.0).

If you continue to use Password Synchronization 1.0, you don't need to upgrade the driver shim. The DirXML 1.1a driver shim runs on the Identity Manager 3.0.1 engine, but the Identity Manager 3.0.1 driver shim cannot run on a DirXML 1.1a engine.

If you choose to not upgrade the driver shim, make sure during the installation of the Identity Manager 3.0.1 engine, that you deselect the Active Directory driver. If it is selected, the driver shim is upgraded.

To upgrade the driver shim:

- 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 Install the Identity Manager 3.0.1 driver shim. You can do this at the same time that you install the Identity Manager 3.0.1 engine.

Follow the instructions in the “[Installing Identity Manager](#)” section in the *Identity Manager 3.0.1 Installation Guide*.

WARNING: If you have been using Password Synchronization 1.0, don't install the upgraded Identity Manager Driver for AD until you have read [Section 7.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on [page 67](#) and are ready to add policies to your driver configuration to provide backward compatibility with Password Synchronization 1.0.

Running an Identity Manager 2.0 or 3.0 driver shim or configuration with the DirXML 1.1a engine is not supported.

- 3 After the shim is installed, Novell eDirectory and the driver need to be restarted.
 - 3a In iManager, click *Identity Manager > Identity Manager Overview*.
 - 3b Browse to the Driver Set where the driver exists, then click *Search*.
 - 3c Click the upper-right corner of the driver icon, then click *Restart driver*.
- 4 Activate the driver shim with your Identity Manager activation credentials.

See [Section 6.4, “Activating the Driver,”](#) on [page 63](#).

After you install the driver shim, continue with [Chapter 4, “Configuring the Active Directory Driver,”](#) on [page 39](#).

5.4 Upgrading the Driver Shim from IDM 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 Install the Identity Manager 3.0.1 driver shim. You can do this at the same time that you install the Identity Manager 3.0.1 engine.

Follow the instructions in “[Installing Identity Manager](#)” section found in the *Identity Manager 3.0.1 Installation Guide*.

WARNING: If you have been using Password Synchronization 1.0, don’t install the upgraded Identity Manager Driver for AD until you have read [Section 7.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on [page 67](#) 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.

- 3 After the shim is installed, Novell eDirectory and the driver need to be restarted. Follow the instructions in “[Starting, Stopping, or Restarting a Driver](#)” in the *Novell Identity Manager 3.0.1 Administration Guide*.
- 4 Activate the driver shim with your Identity Manager activation credentials.
See [Section 6.4, “Activating the Driver,”](#) on [page 63](#).

After you install the driver shim, continue with [Chapter 4, “Configuring the Active Directory Driver,”](#) on [page 39](#).

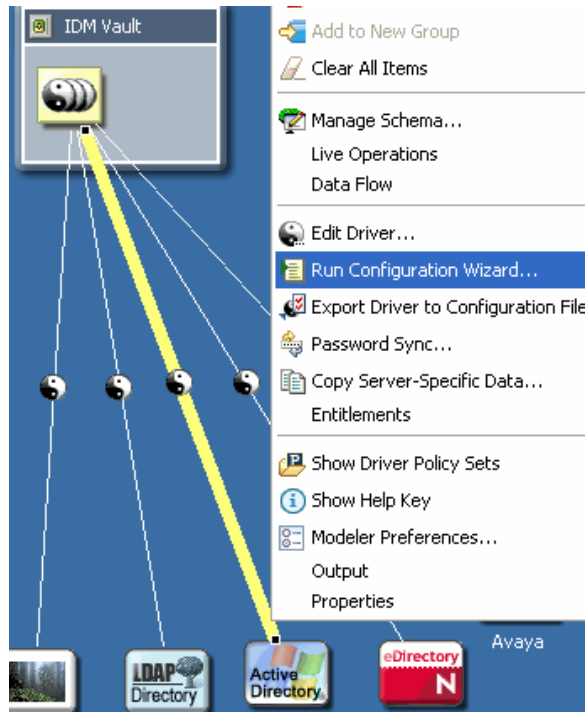
5.5 Applying the Overlay for Exchange Mailboxes

If you have upgraded from Identity Manager 2.x to Identity Manager 3.0.1, the AD driver overlay needs to be applied if Exchange provisioning is enabled on the driver. The overlay allows the driver to control deletes and moves with the Exchange mailboxes.

- ♦ [Section 5.5.1, “Applying the Overlay in Designer,”](#) on [page 53](#)
- ♦ [Section 5.5.2, “Applying the Overlay in iManager,”](#) on [page 56](#)

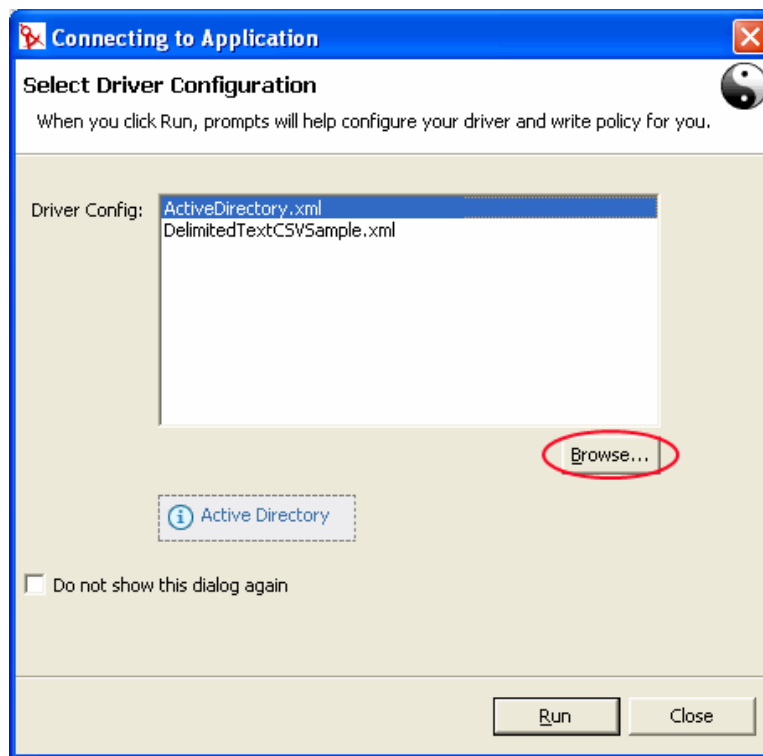
5.5.1 Applying the Overlay in Designer

- 1 In the modeler, right-click on the AD driver connector icon, then click *Run Configuration Wizard*.

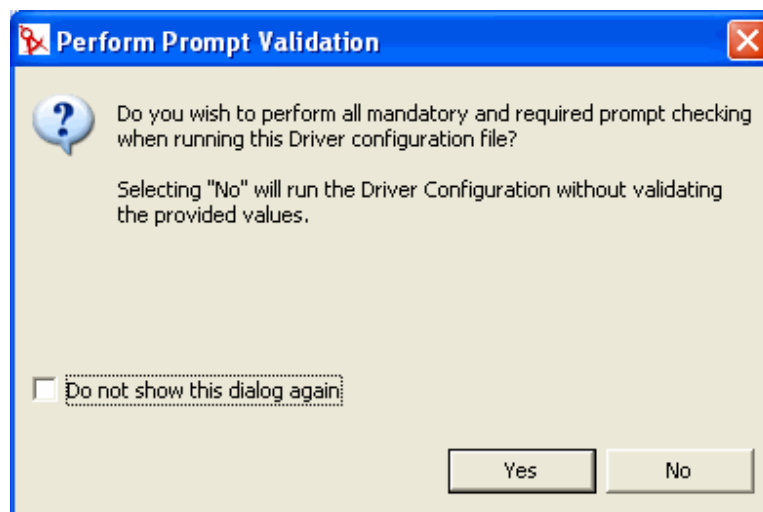


- 2 Select *Browse* and browse to the file `ActiveDirectoryUpdate.xml`, then click *Open*.

The file is located in the following plug-in
eclipse\plugins\com.novell.designer.idm_x.x.x\defs\ActiveDirectoryUpdate.xml.

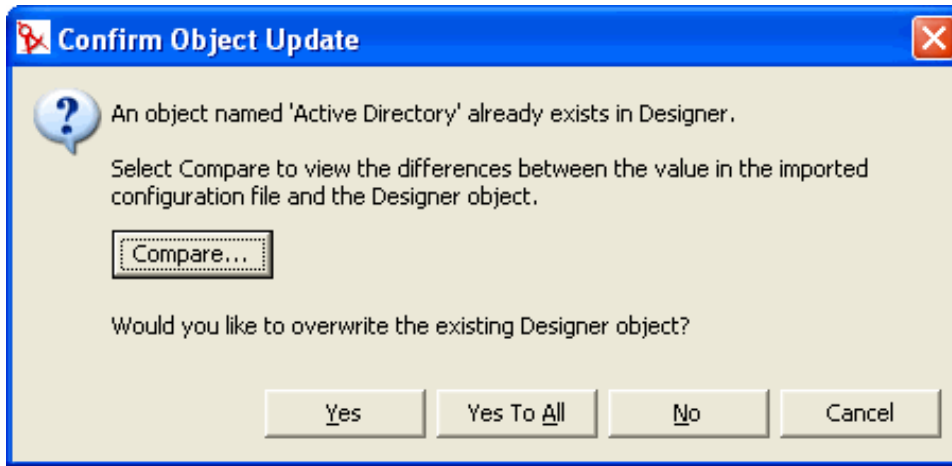


- 3 Select *ActiveDirectoryUpdate.xml*, then click *Run*.
- 4 Select *Yes* or *No* if you desire Designer to validate the information entered in the prompts.



- 5 Enter in the information specific to your environment, then click *OK*. See [Table 5-1 on page 55](#) for a description of the fields.

- 6 In the Confirm Object Update window, select *Compare* to view the differences between the values in the imported configuration file and the Designer object, then click *Close*.



- 7 If the changes are correct, select *Yes* to overwrite the existing Designer object. If you do not want to have the driver updated, select *No*.

Table 5-1 *Overlay Configuration Parameters in Designer*

Parameter	Description
<i>Driver name</i>	This is the driver that needs to be updated with the new parameters. Enter in the driver name or browse to and select the driver.
<i>Update driver</i>	It updates the driver with the parameters. Select <i>Yes</i> if you want the driver updated. Select <i>No</i> if you do not want to update the driver.
<i>homeMDB controls Exchange move</i>	<p>Allow a change to the user HomeMDB attribute to result in a move on the user's Exchange mailbox when using CDOEXM. The Exchange Message Database, where the user's mailbox is moved to, must be in the same domain as the old Exchange Message Database.</p> <p>If <i>Yes</i> is selected, when a User object is moved in eDirectory, the move is reflected in Active Directory and Exchange as well.</p> <p>If <i>No</i> is selected, when a User object is moved in eDirectory it is reflected in Active Directory, but not in Exchange.</p>
<i>homeMDB controls Exchange delete</i>	<p>Allow removal of the user homeMDB attribute to result in a delete of the user's Exchange mailbox when using CDOEXM.</p> <p>If <i>Yes</i> is selected, when an eDirectory User object is deleted the associated Active Directory User object and Exchange accounts are deleted.</p> <p>If <i>No</i> is selected, when an eDirectory User object is deleted the associated Active Directory User object is deleted, but the Exchange account is left intact.</p>

Parameter	Description
<i>Logon and impersonate</i>	<p>Allows the driver authentication account for CDOEXM and Password Set support to logon in different manners.</p> <p>If <i>No</i> is selected, the driver performs a network logon only.</p> <p>If <i>Yes</i> is selected, the driver performs a local logon. The authentication account must be an Active Directory account with administrative privileges.</p>

5.5.2 Applying the Overlay in iManager

There are two different ways to update the driver through iManager. It can be updated in the Identity Manager Overview or through Identity Manager Utilities.

Identity Manager Overview

- 1 In iManager select *Identity Manager > Identity Manager Overview*.
- 2 Select *Search* to find the Driver Set object where the Active Directory driver is stored.
- 3 Select *Add Driver* in the Identity Manager Overview screen.
- 4 Browse to and select the Driver Set object where the Active Directory driver is stored, then click *Next*.
- 5 Select *Import a driver configuration from the server (.XML file)*.
- 6 From the drop-down menu select *ActiveDirectoryUpdate.xml*, then click *Next*.
- 7 Enter in the information specific to your environment, then click *Next*. See [Table 5-2 on page 56](#) for a description of the fields.
- 8 Select *Update that driver (including the driver's image)* to update the driver, or select *Select a different driver*, then click *Next*.
- 9 View the summary of changes, then click *Finish*.

Table 5-2 *Overlay Configuration Parameters in iManager*

Parameter	Description
<i>Driver name</i>	This is the driver that needs to updated with the new parameters.
<i>Existing drivers</i>	From the drop-down menu, select the name of the updated AD driver with Exchange provisioning enabled. Once the driver name is selected, the Drive name field is automatically populated.
<i>Update driver</i>	It updates the driver with the parameters. Select <i>Yes</i> if you want the driver updated. Select <i>No</i> if you do not want to update the driver.

Parameter	Description
<i>homeMDB controls Exchange move</i>	<p>Allow a change to the user HomeMDB attribute to result in a move on the user's Exchange mailbox when using CDOEXM. The Exchange Message Database, where the user's mailbox is move to, must be in the same domain as the old Exchange Message Database.</p> <p>If <i>Yes</i> is selected, when a User object is moved in eDirectory, the move is reflected in Active Directory and Exchange as well.</p> <p>If <i>No</i> is selected, when a User object is moved in eDirectory it is reflected in Active Directory, but not in Exchange.</p>
<i>homeMDB controls Exchange delete</i>	<p>Allow removal of the user homeMDB attribute to result in a a delete of the user's Exchange mailbox when using CDOEXM.</p> <p>If <i>Yes</i> is selected, when an eDirectory User object is deleted the associated Active Directory User object and Exchange accounts are deleted.</p> <p>If <i>No</i> is selected, when an eDirectory User object is deleted the associated Active Directory User object is deleted, but the Exchange account is left in tact.</p>
<i>Logon and impersonate</i>	<p>Allows the driver authentication account for CDOEXM and Password Set support to logon in different manners.</p> <p>If <i>No</i> is selected, the driver performs a network logon only.</p> <p>If <i>Yes</i> is selected, the driver performs a local logon. The authentication account must be an Active Directory account with administrative privileges.</p>

Identity Manager Utilities

- 1 In iManager select *Identity Manager Utilities > Import Drivers*.
- 2 Browse to and select the Driver Set object where the Active Directory driver is stored, then click *Next*.
- 3 Under Additional Policies, select *AD Driver shim configuration update from IDM2 to IDM 3*, then click *Next*.



AD Driver shim configuration update from IDM2 to IDM 3

- 4 Enter in the information specific to your environment, then click *Next*. See [Table 5-2 on page 56](#) for a description of the fields.
- 5 Select *Update that driver (including the driver's image)* to update the driver, or select *Select a different driver*, then click *Next*.
- 6 View the summary of changes, then click *Finish*.

Managing the Active Directory Driver

6

- ◆ [Section 6.1, “Security Parameters,” on page 59](#)
- ◆ [Section 6.2, “Managing Groups,” on page 61](#)
- ◆ [Section 6.4, “Activating the Driver,” on page 63](#)

6.1 Security Parameters

During installation, the driver gathers the necessary information and creates default security policies and parameters. Before you begin customizing your Active Directory driver, you should become familiar with the following:

- ◆ Default policies and parameters
- ◆ The topics discussed in [Chapter 8, “Troubleshooting,” on page 89](#), so you can decide whether any of these issues apply to your environment

Understanding how the parameters work together and work with the operating system helps you define your approach to security for Identity Manager data synchronization.

- ◆ **Authentication ID:** The account that the driver uses to access domain data.

Table 6-1 *Authentication ID*

Format	Username	Method
Domain name	user	Negotiate
Fully Qualified Domain name	domain\user	Negotiate
Distinguished name	cn=DirXML,cn=Users,DC=domain,dc=com	Simple

- ◆ **Authentication Context:** The context used to access domain data.

Table 6-2 *Authentication Context*

Format	Example	Method
The DNS name of the Active Domain controller	mycontroller.mydomain.com	Negotiate
The DNS name of the Active Domain controller, or the IP address of your LDAP server	mycontroller.mydomain.com 137.65.134.83	Simple

- ◆ **Application Password:** The password for the Authentication ID account.

- ◆ **Use Signing:** This parameter is for use between the Active Directory driver and Active Directory, but not between the Metadirectory engine and the Remote Loader. Signing ensures that a malicious computer is not intercepting data. This flag enables signing of the Active Directory connection if you are not using the LDAP SSL port.

This setting requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers. This enables signing on a Kerberos or NTLM v2 authenticated connection.

Like SSL, this parameter is not available on initial import. You set it through the Driver Parameters page after installation is complete.

- ◆ **Use Sealing:** This parameter is for use between the Active Directory driver and Active Directory, but not between the Metadirectory engine and the Remote Loader. Sealing encrypts the data so that it cannot be viewed by a network monitor. This flag enables sealing of the Active Directory connection if you are not using the LDAP SSL port.

This setting requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers. This setting enables encryption on a Kerberos or NTLM v2 authenticated connection.

Like SSL, this parameter is not available on initial import. You set it through the Driver Parameters page after installation is complete.

- ◆ **Use SSL:** This parameter is for use between the Active Directory driver and Active Directory. This parameter controls encryption if you connect to Active Directory by using the LDAP SSL port. This parameter applies to both the Negotiate and Simple authentication methods.

By default the parameter is set to No. If you set this value to Yes, the SSL pipe is encrypted for the entire conversation. An encrypted pipe is preferred because the driver typically synchronizes sensitive information. However, encryption slows the general performance of your servers.

This parameter is configurable through the Driver Parameters page after the driver has been imported.

6.1.1 Recommended Security Configurations

Using the Identity Manager Remote Loader

Table 6-3 *Recommended Settings*

Parameter	Description
<i>Authentication ID</i>	The domain logon name, for example Administrator.
<i>Authentication Context</i>	The DNS name of the domain controller. If you don't want to run the driver on your Active Directory domain controller, use <i>hostname</i> for the Negotiate method but use <i>hostname</i> or the IP address for the Simple method.
<i>Application Password</i>	The password used for the authentication account.
<i>Remote Loader Password</i>	The password for the Remote Loader service.
<i>Authentication Method</i>	Negotiate.

Parameter	Description
<i>Use Signing</i>	No. Requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers.
<i>Use Sealing</i>	No. Requires Windows 2003 or Windows 2000 with the most recent support pack, and Internet Explorer 5.5 SP2 or later on both servers.
<i>Use SSL</i>	Yes. SSL is required to perform Subscriber password check, set, and modify when the driver shim isn't running on the domain controller.

Using SSL

SSL is recommended if you have selected the Simple authentication mechanism because Simple authentication passes passwords in clear text.

Table 6-4 *SSL Parameters*

Parameter	Description
<i>Authentication ID</i>	LDAP format Authentication ID
<i>Authentication Context</i>	IP address of domain controller
<i>Password</i>	The password for the specified Authentication ID
<i>Use Signing</i>	No
<i>Use Sealing</i>	No
<i>Use SSL</i>	Yes

6.2 Managing Groups

The Active Directory group class defines two types of groups and three scopes for membership in the group. Type and scope are controlled by the `groupType` attribute which can be set via Identity Manager policy when a group is created in Active Directory and changed by modifying the attribute.

A group holds a collection of object references. The Distribution Group type gives no special rights or privileges to its members and is commonly used as a distribution list for Exchange. The Security Group type is a security principal. Its members receive the rights and privileges of the group. Security Groups have a pre-Windows 2000 logon name (`samAccountName`) and a Security Identifier (SID) that can be used in Security Descriptor (SD) Access Control Lists (ACL) on other objects to grant or deny rights and privileges to its members.

Group scope controls whether an object from a foreign domain can be a member of the group and also whether the group itself can be a member of another group. The three scopes are Domain Local, Global, and Universal. How these scopes behave, or whether the scope is valid at all, depend on whether Active Directory is operating in Windows 2000 Mixed, Windows 2000 Native or Windows 2003 mode.

In general, Domain Local groups can hold references to objects anywhere in the forest but can be assigned permissions only within the domain. Global groups are the opposite. They can only hold references to objects within the domain but can be assigned permissions throughout the forest. Universal groups can hold references and can be assigned permissions throughout the forest. But

Universal groups come with their own restrictions and performance issues. Groups should be created and used in conformance with Microsoft recommendations.

The groupType attribute is a 32-bit integer whose bits define type and scope. Groups can have only a single scope at any given time.

Table 6-5 GroupType Attribute

GroupType Attribute	Scope	Bits That Define Type and Scope
GROUP_TYPE_GLOBAL_GROUP	Distribution	0x00000002
GROUP_TYPE_DOMAIN_LOCAL_GROUP	Distribution	0x00000004
GROUP_TYPE_UNIVERSAL_GROUP	Distribution	0x00000008
GROUP_TYPE_SECURITY_ENABLED	Security	0x80000000

6.3 Managing Microsoft Exchange Mailboxes

The Active Directory driver can be configured to create, move and delete Microsoft Exchange mailboxes for users in Active Directory. Mailboxes are managed by setting and removing the value for the homeMDB attribute on the user object. This attribute holds the Distinguished Name of the Exchange Private Message Database (MDB) where the mailbox resides. The driver manages mailboxes on Exchange servers that are in the same domain as the driver only.

There are several different ways to manage Exchange mailboxes. The default configuration manages mailboxes through policy decisions made in the Subscriber Command Transformation policy. When a user meets the given conditions, a mailbox is created, moved or removed. The import file gives you three choices for mailbox management:

- ◆ Entitlements
- ◆ Policies
- ◆ Do not Manage Exchange Mailboxes

When using the entitlement method for provisioning, a user is granted or denied a mailbox based on the entitlement set on the user in the Identity Vault. The entitlement holds the Distinguished Name of the MDB and a state value which tells the driver whether the entitlement is granted or revoked. The entitlement itself is managed by the User Application or the Role-Based Entitlements driver. In either case, the external tool grants (or revokes) the right to the mailbox, the Subscriber Command Transformation policy translates that right into an add-value or remove-value on the homeMDB attribute and the driver shim translates the change to homeMDB into the proper calls to the Exchange management system.

If you are using entitlements and have multiple MDB's in your organization, you use the User Application to decide which MDB is to be assigned to a given user. The [Identity Manager Accessory Portlet Reference Guide \(http://www.novell.com/documentation/idm\)](http://www.novell.com/documentation/idm) contains the documentation on how to configure multiple MDBs. The role of the Identity Manager driver is to respond to the entitlements placed on the user object, not to put them there. If you are using the User Application, you are given a list of Exchange MDB's to choose from as the workflow item flows through the approval process. If you are using Role-Based entitlements, the MDB is assigned to the group that holds the role for the user.

When using the policy-based method for provisioning, the Subscriber Command Transformation policy uses information about the state of the user object in the Identity Vault to assign the MDB. The driver shim translates the change into the proper calls to the Exchange management system. The default policy uses a simple rule for assigning the mailbox. It assumes that there is only one MDB and that all users that have made it this far through the policy chain should be assigned to that MDB. Because the rules for assigning different MDB's varies widely from company to company, the default configuration does not attempt to establish a "right way" of doing it. You implement your own policies simply by changing the default assignment rules. You use DirXML Script if statements to define the conditions for mailbox assignments and the do-set-dest-attribute command for the homeMDB attribute to effect the the change. You can get a list of Exchange MDBs using the `ADManager.exe` tool or by your own means.

When not managing Exchange mailboxes, the driver will synchronize the user's e-mail address and mail nickname.

There are other ways to manage the Exchange mailbox. For instance, you could extend the schema of the Identity Vault to hold the homeMDB information and use basic data sychronization to assign the mailbox to the user in Active Directory. In this case, you would use your own tool to make assignments in the Identity Vault.

The default policy works well for simple mailbox assignment to a single MDB. If you want the policy to reflect more complex rules demanded in your environment, the policy has to be changed.

6.4 Activating the Driver

Activate the driver within 90 days of installation. After the 90-day trial period has expired, the driver won't start without the proper activation credential. Events that occur when the driver isn't activated are processed upon activation and subsequent start of the driver.

For activation information, refer to "[Activating Novell Identity Manager Products](#)" in the *Identity Manager 3.0.1 Installation Guide*.

Password Synchronization

7

This section assumes that you are familiar with the information in “[Password Synchronization across Connected Systems](#)” in the *Novell Identity Manager 3.0.1 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 “[Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager](#)” on page 67 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 use the Password Synchronization provided with Identity Manager right away.

In this section:

- ◆ [Section 7.1, “Comparing Password Synchronization 1.0 and Password Synchronization Provided with Identity Manager,”](#) on page 65
- ◆ [Section 7.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on page 67
- ◆ [Section 7.3, “New Driver Configuration and Identity Manager Password Synchronization,”](#) on page 73
- ◆ [Section 7.4, “Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization,”](#) on page 73
- ◆ [Section 7.5, “Setting Up Password Synchronization Filters,”](#) on page 77
- ◆ [Section 7.6, “Retrying Synchronization after a Failure,”](#) on page 84

For information on troubleshooting password synchronization, see “[Tips on Password Synchronization](#)” on page 91.

7.1 Comparing Password Synchronization 1.0 and Password Synchronization Provided with Identity Manager

Table 7-1 Differences Between the Different Versions of Password Synchronization

Functionality	In Password Synchronization 1.0	In Password Synchronization with Identity Manager
Product delivery	A product separate from Identity Manager.	A feature included with Identity Manager, not sold as a separate product.

Functionality	In Password Synchronization 1.0	In Password Synchronization with Identity Manager
Platforms	<ul style="list-style-type: none"> ◆ Active Directory ◆ NT Domain 	<p>Full bidirectional password synchronization is supported on these platforms:</p> <ul style="list-style-type: none"> ◆ Active Directory ◆ eDirectory™ ◆ NIS ◆ NT Domain <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 “Password Synchronization across Connected Systems” in the <i>Novell Identity Manager 3.0.1 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 “Implementing Password Synchronization” in the <i>Novell Identity Manager 3.0.1 Administration Guide</i> .
Main functionality for Windows connected systems	To provide bidirectional password synchronization so that the eDirectory password is synchronized with the Windows password. However, each workstation requires the Novell® Client™.	To provide bidirectional password synchronization. Because Universal Password and Distribution Password are reversible, passwords can be synchronized in both directions. Accomplished within the Identity Manager Publisher and Subscriber channels.
LDAP password changes	Not supported.	Supported.
Novell Client	Required.	Not required.
nadLoginName attribute	Used for keeping passwords updated.	Not used.

Functionality	In Password Synchronization 1.0	In Password Synchronization with Identity Manager
The component that contains the password synchronization functionality	The Identity Manager driver contained the functionality for updating nadLoginName.	<p>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.</p> <p>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 Section 7.4, “Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization,” on page 73.</p>
Agents	A separate piece of software.	No agents are installed; instead, the functionality is now part of the driver.

7.2 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.

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

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.

See “[Preparing to Use Identity Manager Password Synchronization and Universal Password](#)” in the *Novell Identity Manager 3.0.1 Administration Guide*.

Enabling Universal Password doesn’t automatically cause password changes in both systems. Universal Password synchronization starts working only after users change their passwords.

Scenario: Universal Password. At DigitalAirlines, network administrator Sandy enables Universal Passwords. User Markus logs in and changes his password. The Universal Password for Markus is set on both systems. However, user Marie logs in but doesn’t change her password. She continues to log in by using her unchanged password. Universal Password functionality for Marie isn’t set until she changes her password.

- 2 Install the Identity Manager 3.0.1 driver shim to replace the DirXML[®] 1.1a driver shim, and immediately complete [Step 3](#).

NOTE: If you are running Identity Manager 2.0, and are using Universal Password, you do not have upgrade Password Synchronization.

Use the installation program as described in the “[Installing Identity Manager](#)” chapter in the *Identity Manager 3.0.1 Installation Guide*, and select only the Identity Manager Driver for Active Directory.

- 3 Create backward compatibility with Password Synchronization 1.0, by adding a new policy to the driver configuration as described in “[Creating Backward Compatibility with Password Synchronization 1.0 by Adding Policies](#)” on page 70.

A DirXML 1.1a driver shim updates the nadLoginName attribute, but the Identity Manager Identity Manager driver shim doesn’t. Therefore, 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 create backward compatibility, Password Synchronization 1.0 continues to update existing users, but any new or renamed users can’t be synchronized until you deploy Identity Manager Password Synchronization.

After you complete this step, you have the Identity Manager 3.0.1 driver shim and the policies for backward compatibility. Therefore, 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 you want to participate in password synchronization.

Either upgrade an existing configuration or replace 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 Existing Driver Configurations to Support Password Synchronization](#)” in the *Novell Identity Manager 3.0.1 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 “[Upgrading Existing Driver Configurations to Support Password Synchronization](#)” in the *Novell Identity Manager 3.0.1 Administration Guide*.

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 the instructions in [Chapter 4, “Configuring the Active Directory Driver,” on page 39](#) of this driver guide 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 “[Creating Backward Compatibility with Password](#)”

Synchronization 1.0 by Adding Policies” on page 70. The Identity Manager sample driver configuration does not contain those policies.

- ♦ Make sure the nadLoginName attribute is set to Publish, 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 7.5, “Setting Up Password Synchronization Filters,” on page 77.**

6 Set up SSL, if necessary.

For instructions, see **Section 2.3, “Addressing Security Issues,” on page 21.**

The ability of the driver to set a password in Active Directory (Subscriber channel) requires a secure connection provided by one of the following conditions:

- ♦ The machine running the driver is the same machine as the domain controller.
- ♦ The machine running the driver is in the same domain as the domain controller.
- ♦ The machine not in the domain requires the Simple method and SSL set up between it and the domain controller. Bidirectional password synchronization is available only when using the Negotiate authentication mechanism.

Refer to Microsoft documentation for instructions, such as [Configuring Digital Certificates on Domain Controllers \(http://support.microsoft.com/default.aspx?scid=kb;en-us;Q195724\)](http://support.microsoft.com/default.aspx?scid=kb;en-us;Q195724).

7 Turn on Universal Password for Identity Vault user accounts by creating Password Policies with Universal Password enabled.

See **“Managing Password Synchronization”** in the *Novell Identity Manager 3.0.1 Administration Guide*.

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

8 Using the Password Policies and the Password Synchronization settings for the driver, set up the scenario that you want to use for Password Synchronization.

See **“Implementing Password Synchronization”** in the *Novell Identity Manager 3.0.1 Administration Guide*.

9 Test password synchronization.

10 After Identity Manager Password Synchronization is working, remove Password Synchronization 1.0.

10a Using Add/Remove Programs, turn off Password Synchronization 1.0 by removing the agent.

10b In the filter for the driver, change the nadLoginName attribute to Ignore.

10c Remove the backward compatibility policies that are updating nadLoginName from the driver configuration.

10d If desired, you can also remove the nadLoginName attribute from users after Identity Manager Password Synchronization is working, because it is no longer needed.

7.2.1 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 attribute 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 the Identity Manager no longer update this attribute because it is not necessary for Identity Manager Password Synchronization. Therefore, 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.

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

These policies 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 Active Directory do not include the policies 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 Active Directory. 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 Drivers Wizard opens.

- 2** Select the driver set where your existing Active Directory driver resides, then click *Next*.
- 3** In the list of driver configurations that appears, scroll to the Additional Policies section and select *Legacy Password Synchronization 1.0 Policies: Backwards Compatibility for AD and NT*, then click *Next*.

- 4** Complete the import prompts:

- 4a** Select your existing Active Directory 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 Active Directory 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** (Active Directory only) Specify the name of the eDirectory™ attribute mapped to the Active Directory attribute `sAMAccountName`.

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

NOTE: If the sAMAccountName is not mapped to any eDirectory attribute, map sAMAccountName to DirXML-ADAlias name.

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 they aren't 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 any of these parts of the driver configuration has multiple policies, make sure these new policies are listed last.

Table 7-2 Policies

Policy Object Name	Where To Insert It
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)-Output Transform Policies	Output Transformation Policies on the Subscriber channel 

Repeat steps 8a through 8f for each policy.

8a Click *Identity Manager > Identity Manager Overview*.

8b Select the driver set for the driver you are updating.



8c Click the driver you just updated.

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

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

8e Click *Insert* to add the new policy.

In the Insert page that appears, click *Use an Existing Policy*, browse for the new policy object, then click *OK*.

8f 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 steps 1 through 9 for all your Active Directory drivers.

If the sAMAccountName needs to be mapped to the DirXML-ADAliasName in the Publisher channel Schema Mapping policy, then follow this procedure.

WARNING: If the sAMAccountName is mapped to another attribute, following this procedure invalidates your policies. The policies stop synchronizing passwords. Make sure you enter in the proper attribute in [Step 4d on page 70](#).

- 1 In iManager, select *Identity Manager > Identity Manager Overview*.
- 2 Browse to and select the Driver Set object that contains the Active Directory driver, then click *Search*.
- 3 Click the driver icon, then click the *Schema Mapping Policies* icon for the Publisher channel.



- 4 Click *Edit*.
- 5 Select the User class, then click *Attributes*.

Driver DN: ADExchange.Driver Set.Novell

eDirectory Classes	Application Classes	
User	user	Remove
Group	group	Attributes...
Organizational Unit	organizationalUnit	
Organization	organization	
Locality	locality	
[Anything]	<No Unmapped Classes>	Add

- 6 Click the drop-down list under eDirectory Attributes, then browse to and select DirXML-ADAliasName.
- 7 Click the drop-down list under Application Attributes, then browse to and select sAMAccountName.

eDirectory Class: User
Application Class: user

eDirectory Attributes	Application Attributes	
nspmDistributionPassword	nspmDistributionPassword	Remove
DirXML-ADAliasName	sAMAccountName	Add

- 8 Click *Add*, then click *OK*.
- 9 Select the Group class, then click *Attributes*.

10 Repeat steps 6 through 8 for the Group class.

11 Click *OK* twice.

After you have completed this procedure, the driver configurations for your Active Directory drivers are backward compatible with Password Synchronization 1.0. This means that Password Synchronization continues to function as it did before, allowing you to upgrade to Identity Manager Password Synchronization at your convenience.

7.3 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 “[Configuring and Synchronizing a New Driver](#)” in the *Novell Identity Manager 3.0.1 Administration Guide*.

In addition, do the following:

- ◆ Set up SSL, if necessary. See [Section 2.3, “Addressing Security Issues,”](#) on page 21.

The ability of the driver to set a password in Active Directory (Subscriber channel) requires a secure connection provided by one of the following conditions:

- ◆ The machine running the driver is the same machine as the domain controller.
- ◆ The machine running the driver is in the same domain as the domain controller.
- ◆ The machine not in the domain requires the Simple method and SSL set up between it and the domain controller. Bidirectional password synchronization is available only when using the negotiate authentication mechanism.

Refer to Microsoft documentation for instructions, such as [Enabling Secure Sockets Layer for SharePoint Portal Server 2003 \(http://office.microsoft.com/en-us/assistance/HA011648191033.aspx\)](http://office.microsoft.com/en-us/assistance/HA011648191033.aspx).

- ◆ Install new Password Synchronization filters and configure them if you want the connected system to provide user passwords to Identity Manager. See [Section 7.5, “Setting Up Password Synchronization Filters,”](#) on page 77.
- ◆ 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 3.0.1 Administration Guide*.

7.4 Upgrading an Existing Driver Configuration to Support Identity Manager Password Synchronization

IMPORTANT: If a driver is being used with Password Synchronization 1.0, you should complete this section only as part of [Section 7.2, “Upgrading Password Synchronization 1.0 to Password Synchronization Provided with Identity Manager,”](#) on page 67, 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 3.0.1 Administration Guide*.
- ◆ Change the filter to allow Subscriber notify and Publisher ignore on the nspmDistributionPassword attribute.

Prerequisites

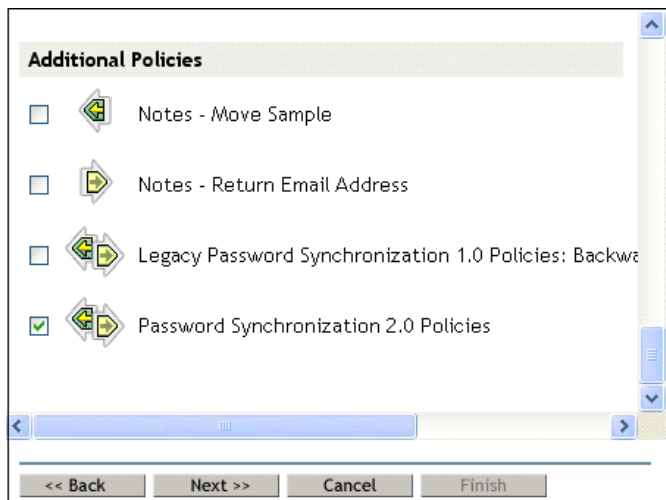
- ❑ Make sure you have converted your existing driver to Identity Manager format, as described in “[Upgrading a Driver Configuration from DirXML 1.1a to Identity Manager Format](#)” in the *Novell Identity Manager 3.0.1 Administration Guide*.
- ❑ Create a backup of your existing driver using the Export Drivers Wizard.
- ❑ 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*.

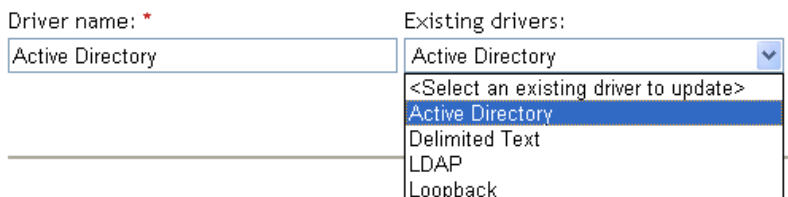
The Import Drivers Wizard opens.

- 2 Select the driver set where your existing driver resides, then click *Next*.

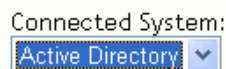


- 3** In the list of driver configurations that appears, select *Password Synchronization 2.0 Policies*, then click *Next*.

The name of the driver contained in the driver configuration file is "Choose an existing driver to update". Enter the actual name you want to use for the driver.



- 4** Select *Active Directory* from the drop-down list.



- 5** Select *Active Directory* as the connected system, then click *Next*.
- 6** Answer yes to 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.
- 7** 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.

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 doesn't change the driver manifest or GCVs.

- 8** 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. However, the new policies aren't 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.

- 9** Insert each of the new policies into the correct place in your existing driver configuration.

If a policy set has multiple policies, 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 3.0.1 Administration Guide*.

Repeat steps 9a through 9e for each policy.

9a Click *Identity Manager > Identity Manager Overview*, then select the driver set for the driver you are updating.



9b Click the driver you just updated.

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

9c Click the icon for the place where you need to add one of the new policies.

9d Click *Insert* to add the new policy.

In the Insert page that appears, click *Use an Existing Policy*, browse for the new policy object, then click *OK*.

9e 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 3.0.1 Administration Guide*.

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

Enable Notify on the Subscriber channel only. Set the Publisher channel to Ignore.

11 Set up SSL, if necessary.

Instructions are contained in [Section 2.3, “Addressing Security Issues,”](#) on page 21.

The ability of the driver to set a password in Active Directory (Subscriber channel) requires a secure connection provided by one of the following conditions:

- ♦ The machine running the driver is the same machine as the domain controller.
- ♦ The machine running the driver is in the same domain as the domain controller.
- ♦ The machine not in the domain requires the Simple method and SSL set up between it and the domain controller. Bidirectional password synchronization is available only when using the Negotiate authentication mechanism.

Refer to Microsoft documentation for instructions, such as [Configuring Digital Certificates on Domain Controllers](http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/prodtech/win2000/secwin2k/a0701.asp) (<http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/prodtech/win2000/secwin2k/a0701.asp>).

12 Install new Password Synchronization filters and configure them if you want the connected system to provide user passwords to Identity Manager. See [Section 7.5, “Setting Up Password Synchronization Filters,”](#) on page 77.

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.

13 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 3.0.1 Administration Guide*.

14 Repeat steps 1 through 14 for all the drivers that you want to participate in password synchronization.

7.5 Setting Up Password Synchronization Filters

The driver needs to be configured to run on only one Windows machine.

However, after you install and configure the driver, do the following on each of the other domain controllers:

- 1 Install a password filter (`pwfilter.dll` file).
- 2 Configure the registry 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 that users make by using Windows clients, encrypts the changes, 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 3.0.1 Administration Guide*.

To simplify your setup and administration of password filters, a 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 are willing to allow remote access to the registry on your domain controllers:

- ♦ **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 reboot a domain controller remotely.

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 7.5.1, “Configuring Password Filters for All Domain Controllers from One Machine,”](#) on page 78.

- ♦ **If you don’t allow remote access to the registry:** Set up the password filters on each domain controller separately. To do this, 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 7.5.2, “Separately Configuring Password Filters on Each Domain Controller,”](#) on page 81.

7.5.1 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.

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

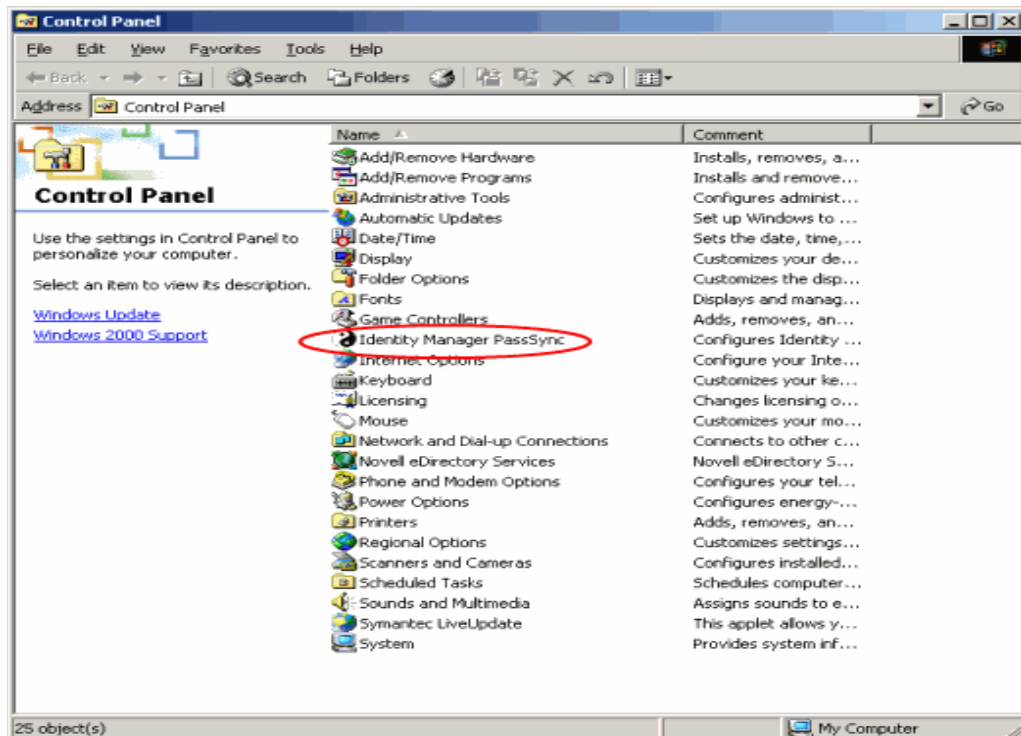
- 1 Confirm that port 135 (the RPC endpoint mapper) is accessible on the domain controllers and on the machine where the Identity Manager Driver for Active Directory is configured to run.

If you are using NetBIOS over TCP, you also need these ports:

- ♦ 137: NetBIOS name service
- ♦ 138: NetBIOS datagram service
- ♦ 139: NetBIOS session service

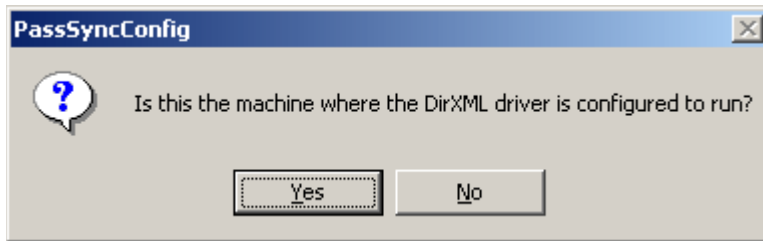
A firewall could prevent the ports from being accessible remotely.

- 2 At the computer where the driver is installed, click *Start > Settings > Control Panel*.



- 3 Double-click *Identity Manager PassSync*.

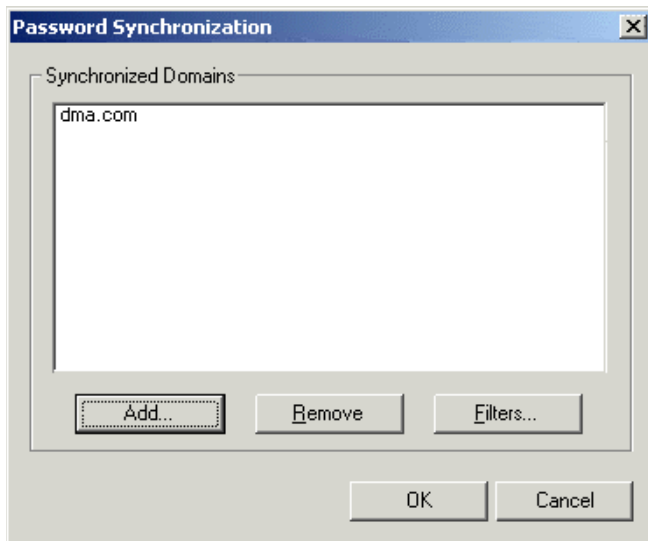
The first time you open the utility, it asks whether this is the machine where the Identity Manager driver is installed.



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

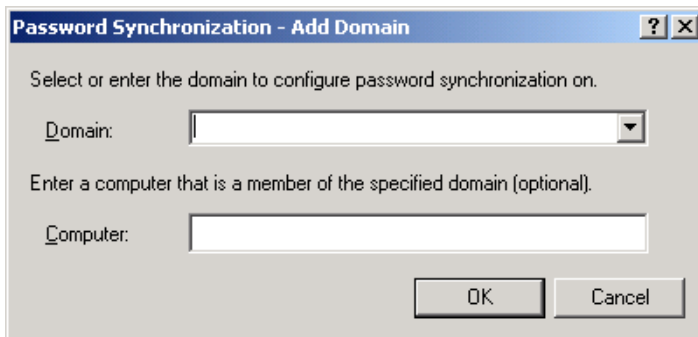
4 Click *Yes*.

A list appears, labeled Synchronized Domains.



5 To add a domain that you want to participate in password synchronization, click *Add*.

The Add Domain dialog box appears.

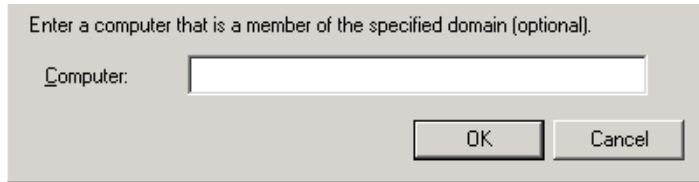


- 6 Specify or select the domain name that you want to add.



The drop-down list displays known domains.

- 7 (Optional) Specify a computer in the domain.



If you leave the Computer edit box blank, PassSync queries the local machine. Therefore, if you are running PassSync on a domain controller, you don't need to enter a name. PassSync queries the local machine (in this case, a domain controller) and gets (from the database) the list of all domain controllers in the domain.

If you aren't installing on a domain controller, enter the name of a computer that is in the domain and that can get to a domain controller.

If you receive an error message indicating that PassSync can't locate a domain, enter a different name.

- 8 Decide whether to use the domain's DNS name.

The DNS name provides more advanced authentication and the ability to more reliably discover domains in bigger installations. However, the choice depends on your environment.

- 9 Log in with administrator rights.

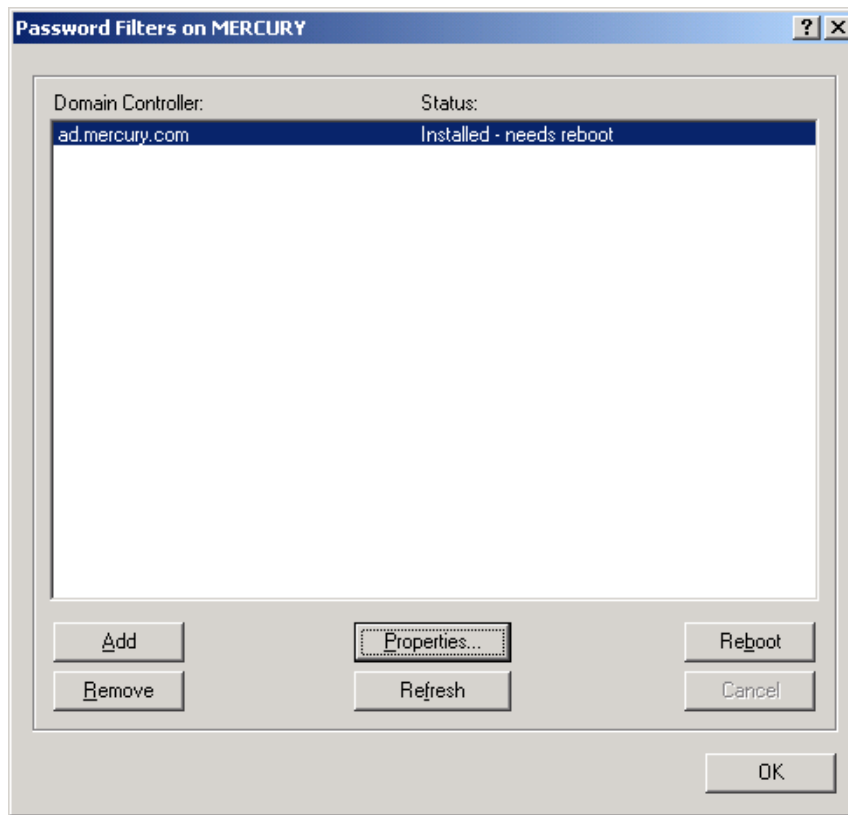
The Identity Manager PassSync utility discovers all the domain controllers for that domain, and installs `pwfilter.dll` on each domain controller. It also updates the registry on the computer where you are running the drivers, and on each domain controller. This might take a few minutes.

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.

- 10 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.



11 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.

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

13 To add more domains, click *OK* to return to the list of domains, and repeat **Step 6** through **Step 12**.

7.5.2 Separately Configuring Password Filters on Each Domain Controller

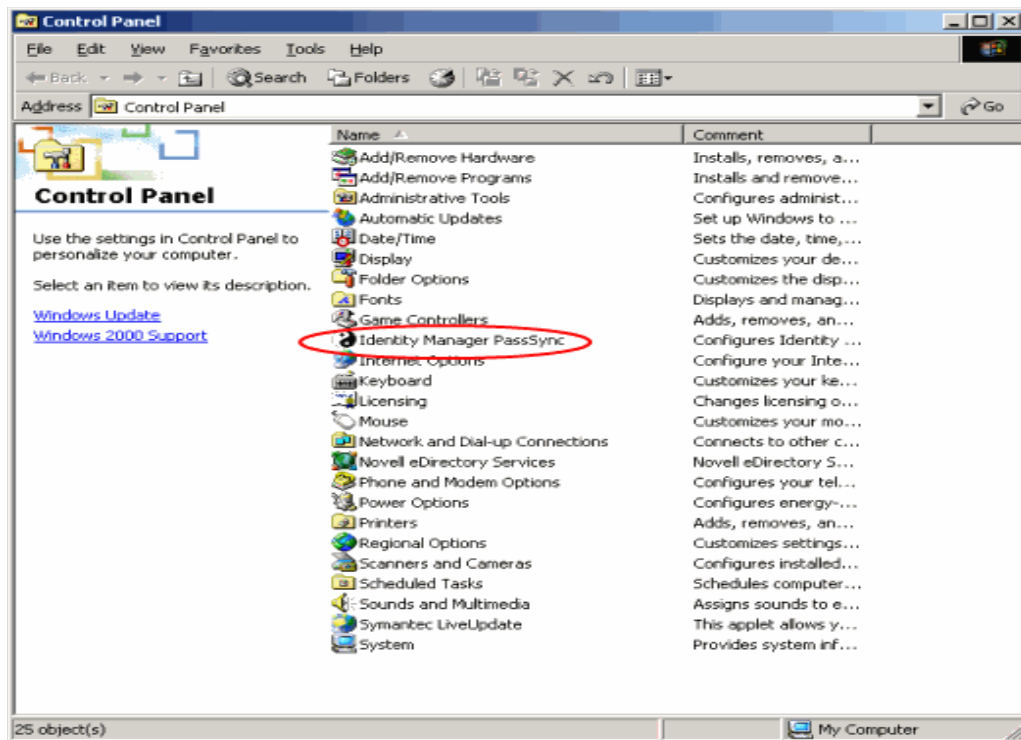
The procedure described in this section explains how to install and configure the password filter on each domain controller, one at a time.

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 Active Directory.

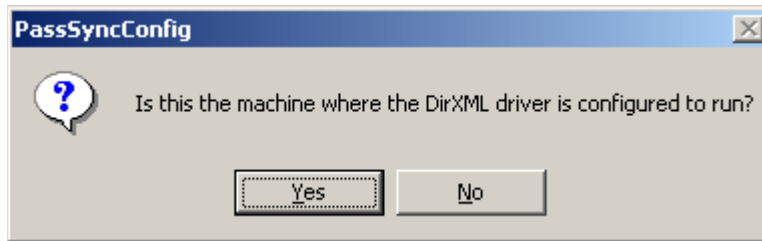
Because setting up the filter requires rebooting the domain controller, you might want to perform this procedure after hours, or reboot only one domain controller at a time. If a domain has more than one domain controller, keep in mind that each domain controller where you want Password Synchronization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

- 1 Confirm that these ports are available on both the domain controller and the machine where the Identity Manager Driver for Active Directory 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 Active Directory.
Installing the driver installs the Identity Manager PassSync utility.
- 3 Click *Start > Settings > Control Panel*, then locate the Identity Manager PassSync utility.



- 4 Double-click *Identity Manager PassSync*.

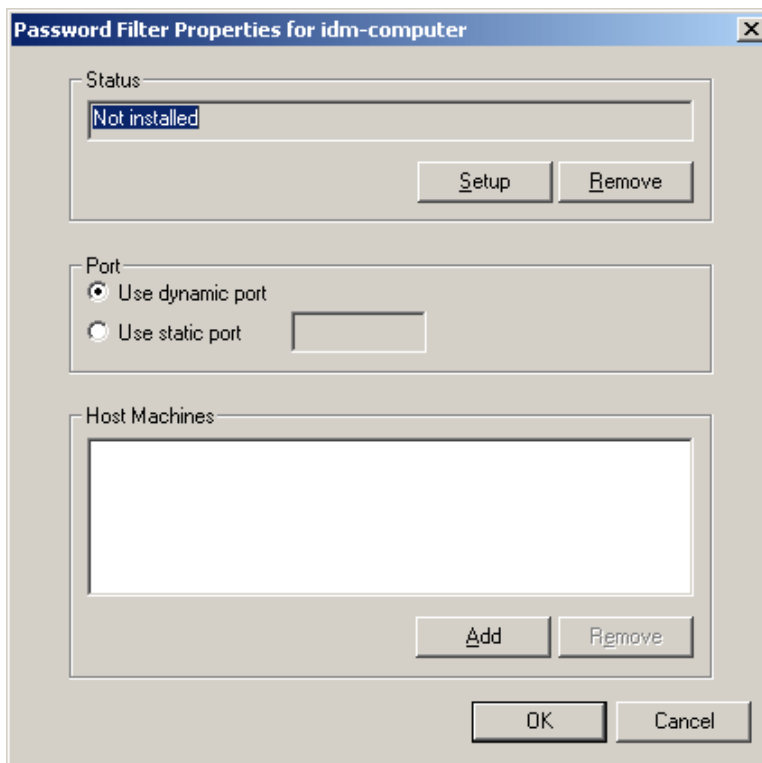
The first time you open the utility, it asks whether this is the machine where the Identity Manager driver is installed.



5 Click *No*.

After you complete the configuration, you are not shown this prompt again unless you remove the password filter by 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.

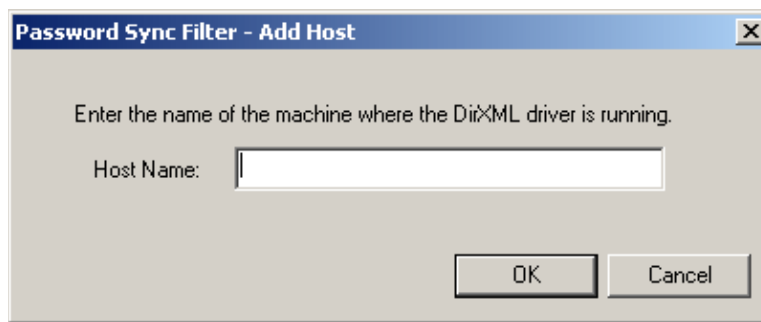


6 Click the *Setup* button to install the password filter, `pwfilter.dll`.

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

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

- In the Password Filter Properties dialog box, click *OK*.
- 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.
- 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.
- Repeat [Step 2](#) through [Step 11](#) for each domain controller that you want to participate in Password Synchronization.
- When the status says Running for all the domain controllers, test Password Synchronization to confirm that it is working.

7.6 Retrying Synchronization after a Failure

The driver and the password filter have been enhanced to improve how password synchronization is retried after a failure.

7.6.1 Retrying after an Add or Modify Event

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

When the driver polls for changes in Active Directory, the driver 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 that a user might receive.

7.6.2 Password Expiration Time

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

You are prompted to specify an expiration time when you import the sample driver configuration. If you don't specify a time, or if the time (interval field) contains invalid characters, the default setting is 60 minutes. If the time specified is less than three times the polling interval specified, the driver changes the time to be three times the polling interval.

Set the value large enough to handle whatever temporary backlog of passwords exists. If you are doing bulk changes, set the timeout large enough to handle all the changes. The rule of thumb is to allow one second per password. For example, to synchronize 18,000 passwords, allow 300 minutes (18,000 passwords divided by 60 seconds).

A setting of -1 is indefinite. Although this setting can handle bulk changes, it can cause problems. For example, a password might never be synchronized because the account wasn't associated. Such a password would therefore remain in the system forever. A number of similar situations could result in a large inventory of unsynchronized passwords held by the system.

Scenarios Relating to Password Expiration Time

On the Publisher channel, password synchronization might occur before the Add event. The driver retries immediately following the Add event.

Scenario: No Effect

A new user with a password is created in Active Directory. The filter immediately sends the new password to the driver. However, the driver hasn't 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. The driver 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 doesn't have an effect in this situation.

Scenario: Increasing the Expiration Time

A new user with a password is created in Active Directory. However, the user information doesn't meet the requirements of the Create policy for the Active Directory driver.

For example, perhaps the Create rule requires a full name, and the required information is missing. Like the No Effect example, the filter sends the password change to the driver immediately.

However, on the first try the password change is not successful in the Identity Vault because the user doesn't exist yet. The driver caches the password.

In this case, however, even when the driver polls for changes in Active Directory and discovers the new user, the driver can't create the new user because the user information doesn't meet the Create policy's requirements.

Creating the new user and synchronizing the password are delayed until all the user information is added in Active Directory 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 Active Directory meets the requirements of the Create policy. If the Add event comes in after the password has expired and the driver doesn't have the password cached for that user, synchronization can't occur. Because the driver doesn't have a cached password, the driver uses the default password in the password policy.

After the user changes the password in either Active Directory or the Identity Vault, that password is synchronized.

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

If your Create policy is restrictive, and it generally takes longer than a day for a new user's information to be completed in Active Directory, you might want to increase the Password Expiration Time parameter interval accordingly. The driver can then cache the passwords until the user is finally created in the Identity Vault.

Scenario: Never Meeting Requirements

A user with a password is created in Active Directory. However, this user never meets the criteria of the Create policy for the Active Directory driver.

For example, perhaps the new user in Active Directory 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 immediately sends the password change, but the password synchronization isn't successful on the first attempt. The driver caches the password.

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

Scenario: E-Mail Notifications

Markus has an Active Directory account and a corresponding Identity Vault account. He changes his Active Directory password, which contains six characters. However, the password doesn't meet the eight-character minimum required by the Password Policy that the administrator created in eDirectory. Password Synchronization is configured to reject passwords that do not meet the policy and to send a notification e-mail to Markus saying that password synchronization failed. The driver caches the password and retries it only if a change is made to the User object in Active Directory.

In this case, shortly after changing a password, Markus receives an e-mail stating that the password synchronization wasn't successful. Markus receives the same e-mail message each time the driver retries the password.

If Markus changes the password in Active Directory to one that complies with the Password Policy, the driver synchronizes the new password to the Identity Vault successfully.

If Markus doesn't 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.

- ◆ [Section 8.1, “Changes Are Not Synchronizing from the Publisher or Subscriber,” on page 89](#)
- ◆ [Section 8.2, “Using Characters Outside the Valid NT Logon Names,” on page 89](#)
- ◆ [Section 8.3, “Synchronizing c, co, and countryCode Attributes,” on page 90](#)
- ◆ [Section 8.4, “Synchronizing Operational Attributes,” on page 90](#)
- ◆ [Section 8.5, “Password Complexity on Windows 2003,” on page 90](#)
- ◆ [Section 8.6, “Error Message LDAP_SERVER_DOWN,” on page 91](#)
- ◆ [Section 8.7, “Tips on Password Synchronization,” on page 91](#)
- ◆ [Section 8.8, “Where to Set the SSL Parameter,” on page 92](#)
- ◆ [Section 8.9, “Active Directory Account Disabled after a User Add on the Subscriber Channel,” on page 92](#)
- ◆ [Section 8.10, “Moving a Parent Mailbox to a Child Domain,” on page 93](#)
- ◆ [Section 8.11, “Restoring Active Directory,” on page 93](#)
- ◆ [Section 8.12, “Moving the Driver to a Different Domain Controller,” on page 94](#)
- ◆ [Section 8.13, “Migrate from Active Directory,” on page 94](#)
- ◆ [Section 8.14, “Setting LDAP Server Search Constraints,” on page 94](#)

8.1 Changes Are Not Synchronizing from the Publisher or Subscriber

To synchronize changes in Active Directory, the account used by the Identity Manager driver must have the proper rights set up. For information on the necessary rights, see [Section 2.4, “Creating an Administrative Account,” on page 26](#).

If you use the default policies, you must also meet the requirements for the Create, Match, and Placement policies. For information on default policy requirements, see [“Policies” on page 14](#).

The attribute dirxml-uACLockout is not synchronized on the Publisher channel.

8.2 Using Characters Outside the Valid NT Logon Names

The default Subscriber creation policy generates an NT Logon Name (also known as the sAMAccountName and the Pre-Windows 2000 Logon Name) based on the Relative-Distinguished Name of the account in the Identity Vault. The NT Logon name uses a subset of the ASCII character set. The default policy strips any character outside of the valid range before creating an object in Active Directory.

If the policy doesn't satisfy the business rules of your company, you can change the policy after import. Businesses that use Identity Vault account names outside of the traditional ASCII character set should pay particular attention to this policy.

8.3 Synchronizing c, co, and countryCode Attributes

When you use the Active Directory management console to select a country for a user, three attributes are set:

Table 8-1 *Attributes for Country*

Attribute	Description
c	Contains a two-character country code as defined by the ISO.
co	Contains a longer name for the country.
countryCode	Contains a numeric value (also defined by the ISO) that represents the country.

Because the ISO-defined numeric country codes are intended for use by applications that can't handle alphabetic characters, by default the schema in the Identity Vault includes c and co but not countryCode.

Identity Manager is capable of mapping c and co. It can also map countryCode if you add a similar attribute to the eDirectory™ schema.

Active Directory's management console tries to keep all three of these attributes synchronized, so that when you set the country in the console, all three attributes have appropriate values. Some administrators might want a similar behavior when the attribute is set through Identity Manager. For example, you might want to configure the driver so that even though only c is in the Filter, co and countryCode are also set when a change for c is sent on the Subscriber channel.

8.4 Synchronizing Operational Attributes

Operation attributes are attributes that are maintained by an LDAP server that contains special operational information. Operation attributes are read-only. They can't be synchronized or changed.

8.5 Password Complexity on Windows 2003

Passwords must meet criteria that the password policies specify.

Complexities and requirements in Windows 2000/2003 password policies are different from complexities and requirements in eDirectory.

If you plan to use Password Synchronization, create and use passwords that match the rules of complexity in both Active Directory and eDirectory™. Otherwise, the passwords fail.

TIP: Make the password policies for both systems as similar to each other as you can. In a lab environment, disable strong-password functionality on Windows 2003 servers before installing the Active Directory driver. After the Active Directory driver is working properly, make sure that passwords used in eDirectory and Active Directory satisfy the rules of complexity for both systems. Then re-enable strong-password functionality on the Windows 2003 server.

For troubleshooting tips, see [TID 10083320 \(http://support.novell.com/cgi-bin/search/searchtid.cgi?/10083320.htm\)](http://support.novell.com/cgi-bin/search/searchtid.cgi?/10083320.htm).

8.6 Error Message LDAP_SERVER_DOWN

The error code LDAP_SERVER_DOWN usually means that the driver can't open the LDAP port on the Active Directory domain controller configured for synchronization. This can happen for several reasons.

- ◆ The server named in the driver authentication context is incorrect. The authentication context should hold the DNS name or the IP address of the domain controller you use for synchronization. If you leave the parameter empty, the driver attempts to connect to the machine that is running the driver shim (either the same server that is running IDM, or the server hosting the Remote Loader).
- ◆ You are using an IP address for authentication context, and you have disabled non-Kerberos authentication to Active Directory. Kerberos requires a DNS name for authentication context. The driver shim can authenticate only using the pre-Windows 2000 Logon method or simple bind. If you have disabled NTLM, NTLM2, and simple bind on your network, you might receive the LDAP_SERVER_DOWN message.
- ◆ You have configured the driver to use an SSL connection to Active Directory. This message means that something is wrong with the certificate that you imported to the driver shim server (or no certificate was imported at all).

8.7 Tips on Password Synchronization

We recommend that you use a secure connection when you are synchronizing passwords. Vulnerable connections are between the following:

- ◆ The Metadirectory engine and the Remote Loader
- ◆ The Remote Loader and Active Directory
This is true only when you run the Remote Loader remotely from the domain controller that you're connecting to.
- ◆ The Metadirectory engine and Active Directory when you aren't using the Remote Loader
This is true only if the domain controller isn't local to this machine.

You can create a secure connection by doing one or more of the following:

- ◆ Configure SSL between the Metadirectory engine and the Remote Loader
- ◆ Run the Remote Loader on the domain controller
- ◆ Configure SSL between the driver shim and Active Directory
This doesn't apply if you are running the driver on the domain controller that you're connecting to.

For password synchronization to work when the driver shim isn't running on the domain controller, you must have SSL configured.

8.7.1 Providing Initial Passwords

If you see an error about a password not complying when a user is initially created, you need to check your password policies.

For example, perhaps you want the Active Directory driver to provide the initial password for a user when the Active Directory driver creates a User object in the Identity Vault. When a user is created, the driver shim creates the user and then sets the password.

Because adding the user and setting the password are done separately, the new user in this example receives the default password, even if only momentarily. The password is soon updated because the Active Directory driver sends it immediately after adding the user.

If the default password doesn't comply with the eDirectory Password Policy for the user, an error is displayed. For example, if a default password that was created by using the user's surname is too short to comply with the Password Policy, you might see a -216 error saying that the password is too short. However, the situation is soon rectified if the Active Directory 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 following:

- ♦ Change the policy on the Publisher channel that creates default passwords, so that default passwords conform to the Password Policies (created by using the Manage Password Policies option in Password Management) that have been defined for your organization in the Identity Vault. When the initial password comes from the authoritative application, it replaces the default password.

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

- ♦ 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.

These measures are especially important if the initial password doesn't come with the add event but instead comes in a subsequent event.

8.8 Where to Set the SSL Parameter

The SSL parameter in the driver configuration is for SSL between the Active Directory driver and Active Directory. It isn't for SSL between the Metadirectory engine and the Remote Loader. See [“Encryption” on page 22](#).

8.9 Active Directory Account Disabled after a User Add on the Subscriber Channel

The default configuration maps the Identity Vault Logon Disabled attribute to the dirxml-uACAccountDisable bit of the userAccountControl attribute in Active Directory. A Subscriber add operation might set Logon Disabled to false (account enabled), but the Publisher loopback of the add operation reports that Logon Disabled is true (account disabled).

Additionally, inspecting the object in Active Directory might show that the account is disabled. This happens in part because of the way that the driver creates objects in Active Directory and in part because of a mismatch of policies between the driver and Active Directory itself.

8.9.1 Account Disabled in Active Directory Users and Computers

If the account remains disabled in Active Directory after the provisioning cycle completes, you might have a mismatch between policies configured for the driver and policies enforced by Active Directory.

Take for example a Password Required policy. If a user add operation contains an invalid password (or no password at all), the account created in Active Directory should be disabled. But Active Directory might set the `dirxml-uACPasswordNotRequired` bit in `userAccountControl` without the driver's knowledge.

Interestingly, this causes the logon enable action of the add operation to fail if the add operation does not include a policy for `dirxml-uACPasswordNotRequired`. Therefore, the account stays disabled.

Later (perhaps almost immediately because of a merge operation), the driver might attempt to enable the account again by setting Logon Disabled to false. If you want to override the Active Directory policy and ensure that accounts always require a password, you should set `dirxml-uACPasswordNotRequired` to false whenever Logon Disabled changes on the Subscriber channel.

8.10 Moving a Parent Mailbox to a Child Domain

If you move a parent mailbox to a mailbox store in a child domain by changing a user's `homeMDB` attribute, the driver fails the move. The error code returned is `0x80072030`.

This error occurs on inter-domain moves. Moving an Exchange parent mailbox to a child domain isn't supported.

8.11 Restoring Active Directory

When you need to restore some or all of Active Directory, the driver might pick up interim events and perform unwanted actions on the Identity Vault. To restore safely, temporarily disable the driver during the restore operation and then bring the Identity Vault back into synchronization with Active Directory.

- 1 Disable the driver.
- 2 Delete the `Dirxml-DriverStorage` attribute on the driver object in the Identity Vault.
- 3 Restore Active Directory.
- 4 Set the Active Directory driver to Manual or Automatic startup.
- 5 Start the driver.
- 6 Re-migrate to find unassociated objects.

8.12 Moving the Driver to a Different Domain Controller

You can configure the driver to synchronize against a different domain controller by changing the driver Authentication Context parameter. When you restart the driver, the state information that the driver uses to track changes in Active Directory is invalid, and Active Directory might replay a large number of old events to bring the state back to the current time.

You can avoid this replay by removing the driver state information while updating the Authentication Context:

- 1 Stop the driver.
- 2 Delete the Dirxml-DriverStorage attribute on the Driver object in the Identity Vault.
- 3 Update the Authentication Context parameter.
- 4 Start the driver.
This causes a resync of associated objects in the Identity Vault.
- 5 Re-migrate to find unassociated objects in Active Directory.

8.13 Migrate from Active Directory

When migrating from Active Directory to the Identity Vault you need to be concerned about object containment, DN references and search limits on the Active Directory server. The general strategy for dealing with containment is to migrate containers first, objects that may be members of groups (including user objects) second and groups last. If you have a moderately large number of objects to migrate you will need to adjust your strategy to handle the LDAP search constraints configured on the Active Directory server. You can change the constraints on the LDAP server or adjust your migration to get only a subset of objects each time (for instance, migrating container by container or objects starting with 'A', 'B', and etc...).

8.14 Setting LDAP Server Search Constraints

Following is a terminal session showing you how to use NTDSUTIL.EXE to change the LDAP search parameters on your domain controller. You need only change these settings on the domain controller being used for IDM synchronization for the duration of the migration. Write down the current configuration values and run NTDSUTIL.EXE after migration completes to restore the original values. NTDSUTIL.EXE can be run on any member server.

- 1 At a command prompt type ntdsutil.
- 2 Type LDAP Policies press Enter.
- 3 Type Connections press Enter.
- 4 Type Connect to domain *domain_name* press Enter.
- 5 Type Connect to server *server_name* press Enter.
- 6 Type Quit press Enter.
- 7 Type Show Values press Enter.

```
C:\>ntdsutil
ntdsutil: LDAP Policies
ldap policy: Connections
```

```
server connections: Connect to domain raptor
Binding to \\raptor1.raptor.lab ...
Connected to \\raptor1.raptor.lab using credentials of locally logged
on user.
server connections: Connect to server raptor1
Disconnecting from \\raptor1.raptor.lab...
Binding to raptor1 ...
Connected to raptor1 using credentials of locally logged on user.
server connections: Quit
ldap policy: Show Values
```

Policy	Current (New)
MaxPoolThreads	4
MaxDatagramRecv	4096
MaxReceiveBuffer	10485760
InitRecvTimeout	120
MaxConnections	5000
MaxConnIdleTime	900
MaxPageSize	1000
MaxQueryDuration	120
MaxTempTableSize	10000
MaxResultSetSize	262144
MaxNotificationPerConn	5
MaxValRange	1500

```
ldap policy: set MaxQueryDuration to 1200
ldap policy: set MaxResultSetSize to 6000000
ldap policy: Commit Changes
ldap policy: Quit
ntdsutil: Quit
Disconnecting from raptor1...
C:\>
```


Changing Permissions on the CN=Deleted Objects Container

A

When an Active Directory object is deleted, a small portion of the object remains for a specified time so that other domain controllers that are replicating changes become aware of the deletion. By default, only the System account and members of the Administrators group can view the contents of this container. This section describes how to modify the permissions on the CN=Deleted Objects container.

Changing permissions on the Deleted Objects container might be necessary if you have enterprise applications or services that bind to Active Directory with a non-System or non-Admin account and poll for directory changes.

This process requires `dsacals.exe` from the Active Directory Application Mode (ADAM) package. This version is an upgrade from the one in the Windows Server 2003 Support Tools and now supports the required capabilities. The ADAM Administration Tools are supported on Windows XP Professional, Windows Server 2003 Standard Edition, Windows Server 2003 Enterprise Edition, and Windows Server 2003 Datacenter Edition.

To get and install the ADAM Administration Tools:

- 1 From the [ADAM Web page \(http://www.microsoft.com/downloads/details.aspx?FamilyId=9688F8B9-1034-4EF6-A3E5-2A2A57B5C8E4&displaylang=en\)](http://www.microsoft.com/downloads/details.aspx?FamilyId=9688F8B9-1034-4EF6-A3E5-2A2A57B5C8E4&displaylang=en), download the ADAM retail package.
- 2 Double-click the downloaded file and provide a directory to extract the archive into.
- 3 Launch the Active Directory Application Mode Setup Wizard by double-clicking `adamsetup.exe`, then click *Next*.
- 4 Review and accept the license terms, then click *Next*.
- 5 Select ADAM administration tools only, then click *Next*.
- 6 Review the selections, then click *Next*.
- 7 When Setup has concluded, click *Finish*.

After ADAM Administration Tools is installed, modify the permissions on the CN=Deleted Objects container:

- 1 Log in with a user account that is a member of the Domain Admins group.
- 2 Click *Start > All Programs > ADAM > ADAM Tools Command Prompt*.
- 3 In the Command Prompt, enter the following command:

```
dsacals "CN=Deleted Objects,DC=Contoso,DC=com" /takeownership
```

Substitute the distinguished name of the Deleted Objects container for your own domain.

Each domain in the forest will have its own Deleted Objects container.

The following output should be displayed:

```
Owner: Contoso\Domain Admins
Group: NT AUTHORITY\SYSTEM
Access list:
```



```
{This object is protected from inheriting permissions from
the parent}
```

```
Allow BUILTIN\Administrators  SPECIAL ACCESS
                                LIST CONTENTS
                                READ PROPERTY
Allow NT AUTHORITY\SYSTEM      SPECIAL ACCESS
                                DELETE
                                READ PERMISSONS
                                WRITE PERMISSIONS
                                CHANGE OWNERSHIP
                                CREATE CHILD
                                DELETE CHILD
                                LIST CONTENTS
                                WRITE SELF
                                WRITE PROPERTY
                                READ PROPERTY
```

The command completed successfully

- 4** To grant a security principal permission to view the objects in the CN=Deleted Objects container, enter the following command:

```
dsacls "CN=Deleted Objects,DC=Contoso,DC=com" /g
CONTOSO\JaneDoe:LCRP
```

In this example, the user CONTOSO\JaneDoe has been granted List Contents and Read Property permissions on the container. These permissions are sufficient to allow the user to view the contents of the Deleted Objects container. However, these permissions don't allow the user to make any changes to objects in that container. These permissions are equivalent to the default permissions granted to the Administrators group. By default, only the System account has permission to modify objects in the Deleted Objects container.

The following output should be displayed:

```
Owner: CONTOSO\Domain Admins
Group: NT AUTHORITY\SYSTEM
Access list:
{This object is protected from inheriting permissions from
the parent}
Allow BUILTIN\Administrators  SPECIAL ACCESS
                                LIST CONTENTS
                                READ PROPERTY
Allow NT AUTHORITY\SYSTEM      SPECIAL ACCESS
                                DELETE
                                READ PERMISSONS
                                WRITE PERMISSIONS
                                CHANGE OWNERSHIP
                                CREATE CHILD
                                DELETE CHILD
                                LIST CONTENTS
                                WRITE SELF
                                WRITE PROPERTY
                                READ PROPERTY
Allow CONTOSO\JaneDoe         SPECIAL ACCESS
                                LIST CONTENTS
                                READ PROPERTY
```

The command completed successfully.

The user CONTOSO\JaneDoe now has permissions to view deleted objects in the CONTOSO domain.

Documentation Update

B

The documentation was updated on the following dates:

- ◆ Section B.1, “October 3, 2006,” on page 101

B.1 October 3, 2006

Updates were made to the following sections. The changes are explained below.

B.1.1 Preparing Active Directory

The following updates were made in this section:

Location	Change
“SSL Connection Between the Active Directory Driver and the Domain Controller” on page 23	Left the link to step 5, but changed the link to Creating, Exporting, and Importing Certificates instead of to SSL Connection Between the Remote Loader and Identity Manager.
“SSL Connection Between the Active Directory Driver and the Domain Controller” on page 23	Fixed the broken link to Microsoft’s Web site. The link is now http://www.microsoft.com/technet/security/prodtech/windows2000/secwin2k/swin2kad.mspx .
“Creating, Exporting, and Importing Certificates” on page 23	Fixed the file name to be Idifde instead of Idife.