

Novell DirXML[®] Driver for Active Directory

2.0a

www.novell.com

IMPLEMENTATION GUIDE

September 30, 2003



Novell[®]

Legal Notices

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

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

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

Copyright © 2003 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.

Patents Pending.

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

Implementation Guide
[September 30, 2003](#)

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.

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

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

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

Novell Client is a trademark of Novell, Inc.

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

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

Third-Party Trademarks

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

Contents

- About This Guide** **7**

- 1 Introducing the DirXML Driver for Active Directory** **9**
 - Driver Overview. 9
 - New Features. 9
 - Exchange 2000 Support 9
 - Schema Support 10
 - New Secure Authentication Field 10
 - Default Driver Configuration 10
 - Data Flow 10

- 2 Installing the DirXML Driver for Active Directory** **13**
 - Planning Your Installation 13
 - Installation Locations 13
 - Security Options 15
 - Prerequisites 18
 - Installing and Upgrading 18
 - Installing the Active Directory Driver (Local Install) 18
 - Installing the Active Directory Domain Driver (Remote Loader Installation) 18
 - Upgrading the Driver 19
 - Post-Installation Setup 19

- 3 Customizing the DirXML Driver for Active Directory** **27**
 - Configuring Driver Parameters 27
 - Changing Driver Authentication and Passwords 27
 - Configuring Data Synchronization 28
 - Enabling Exchange 2000 Mailboxes 28
 - Managing Login Names. 29
 - Using Rename Instead of Synchronizing a Naming Attribute. 30

- 4 Documentation Content Updates** **31**
 - November 15, 2002. 31
 - Installing the DirXML Driver for Active Directory 31
 - April 22, 2003. 32
 - Installing the DirXML Driver for Active Directory 32
 - September 30, 2003 32

About This Guide

This guide explains how to install and configure the DirXML[®] Driver for Active Directory.

The guide contains the following sections:

- ♦ **Chapter 1, “Introducing the DirXML Driver for Active Directory,” on page 9**
This section introduces new features and explains the default driver configuration.
- ♦ **Chapter 2, “Installing the DirXML Driver for Active Directory,” on page 13**
This section covers both the installation and upgrade processes as well as post-installation setup tasks.
- ♦ **Chapter 3, “Customizing the DirXML Driver for Active Directory,” on page 27**
This section explains how to customize driver parameters and data synchronization. It provides examples for common customizations.

Additional Documentation

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

Documentation Updates

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

Documentation Conventions

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

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

User Comments

We want to hear your comments and suggestions about this manual and the other documentation included with Novell DirXML. To contact us, send e-mail to proddoc@novell.com.

1

Introducing the DirXML Driver for Active Directory

This section covers the following topics:

- ◆ “[Driver Overview](#)” on page 9
- ◆ “[New Features](#)” on page 9
- ◆ “[Default Driver Configuration](#)” on page 10

Driver Overview

The DirXML[®] Driver for Active Directory* is designed to synchronize data between Novell[®] eDirectory[™] and the Microsoft* Active Directory* directory service. The synchronization is bi-directional; you determine whether information should flow to and from both directories, or whether information should flow only from one directory to the other.

In addition, this driver can be configured to synchronize Microsoft Exchange 2000 mailbox data in Active Directory.

New Features

The updated driver includes support for Exchange 2000 and for secure authentication.

Exchange 2000 Support

The driver now supports the following attributes that will enable an Active Directory object as a Microsoft Exchange 2000 mailbox:

- ◆ msExchHomeServerName
- ◆ mailNickname
- ◆ mail
- ◆ msExchMailboxSecurityDescriptor
- ◆ authOrig
- ◆ uauthOrig

For more information about using DirXML to set up Exchange 2000 mailboxes, see “[Enabling Exchange 2000 Mailboxes](#)” on page 28.

Schema Support

The preconfigured driver files ship with a partial AD schema so that you don't have to read the AD schema on driver startup. If you need more than the included partial schema, click Refresh Application Schema in the Schema Mapping rule page.

New Secure Authentication Field

The preconfigured driver file has a new configuration field that specifies the use of secure authentication for the driver. After you import the driver's preconfigured driver file, the new Use Secure Authentication field displays on the Driver Parameters tab in the Driver Properties page.

See “[Security Options](#)” on page 15 for a discussion of security for DirXML data synchronization.

Default Driver Configuration

DirXML fundamentals are explained in the *DirXML 1.1a Administration Guide* (<http://www.novell.com/documentation/lg/dirxml11a/dirxml/data/a2iii88.html>). The DirXML Driver for Active Directory *Implementation Guide* discusses implementations, additions, or exceptions specific to the Active Directory driver.

Data Flow

Publisher and Subscriber Channels

The driver supports Publisher and Subscriber channels:

- ◆ The Publisher reads events from Active Directory for the domain hosted on the server that the driver shim is running on and submits that information to eDirectory.
- ◆ The Subscriber watches for additions and modifications to eDirectory directory objects and makes changes to Active Directory that reflect those changes.

When the driver is configured so that both Active Directory and eDirectory are allowed to update a specific attribute, the most recent change will determine the attribute value.

Rules

Rules are used to control data synchronization between the driver and eDirectory. The Active Directory driver comes with the set of preconfigured rules detailed in [Table 1](#). These rules can be customized through Novell iManager as explained in [Chapter 3, “Customizing the DirXML Driver for Active Directory,”](#) on page 27.

Table 1 Rules for Active Directory Driver

Rule	Description
Create	Configured on the driver object. Specifies that in order for an Active Directory user to be created as a user in eDirectory, Internet EMail Address and Surname attributes must be defined.

Rule	Description
Schema Map	<p>Configured on the driver object.</p> <p>Maps the following eDirectory User and Group properties to Active Directory user and group attributes:</p> <ul style="list-style-type: none"> Description, description Facsimile Telephone Number, facsimile telephoneNumber Full name, displayName Given Name, givenName Initials, initials Internet EMail Address, mail L, physicalDeliveryOfficeName Member, member Physical Delivery Office Name, l Postal Code, PostalCode Postal Office Box, postOfficeBox S, st SA, streetAddress See Also, seeAlso Surname, sn Telephone Number, telephoneNumber Title, title CN, cn Group Membership, memberOf Owner, managedBy
Matching	<p>Configured on the driver object.</p> <p>Specifies that a user in eDirectory is the same user specified in Active Directory when the value of Internet Email Address is the same in both places.</p> <p>Specifies that a group in eDirectory is the same group specified in Active Directory when the value of CN is the same in both places.</p>
Placement	<p>Configured on the Publisher and Subscriber channels.</p> <p>Specifies that new users will be named by the value of the leafmost part of the source distinguished name and be placed in the containers you defined during driver setup. You should create these containers before you start the driver.</p> <p>The default placement implements a flat tree in both eDirectory and Active Directory. If you want hierarchical placement, you'll need to modify the rule or create a style sheet.</p>

2

Installing the DirXML Driver for Active Directory

The DirXML[®] Driver for Active Directory can be installed along with other DirXML drivers at the same time that the DirXML engine is installed. This method of installation is documented in the *DirXML 1.1a Administration Guide* on the [DirXML Documentation Web site \(http://www.novell.com/documentation/lg/dirxml11a/index.html\)](http://www.novell.com/documentation/lg/dirxml11a/index.html).

The driver can also be installed separately, as explained in this section, by running the DirXML installation and selecting only the Active Directory driver.

This chapter covers the following installation topics:

- ◆ “Planning Your Installation” on page 13
- ◆ “Prerequisites” on page 18
- ◆ “Installing and Upgrading” on page 18

Planning Your Installation

You will need to determine installation location and security configuration before you start the driver installation.

Installation Locations

The driver itself must run on Windows* 2000. However, you don’t need to install the DirXML engine on this same machine. Using the Remote Loader, you can separate the engine and the driver, allowing you to balance the load on different machines or accommodate corporate directives.

The AD driver can run in any of the following three scenarios:

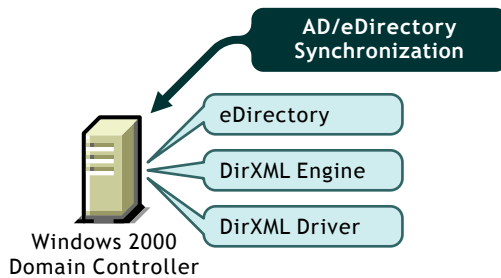
- ◆ **Single Server**

A single Windows 2000 domain controller hosts eDirectory, the DirXML engine, and the driver.

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 DirXML and Active Directory.

However, hosting eDirectory and DirXML on the domain controller increases the overall load on the controller and increases the risk that the controller may fail. Domain controllers play a critical role in Microsoft networking and 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.

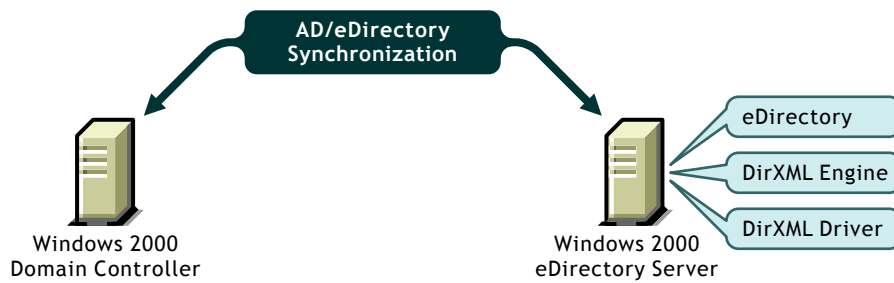
Figure 1 Single Server Installation



◆ **Dual Server**

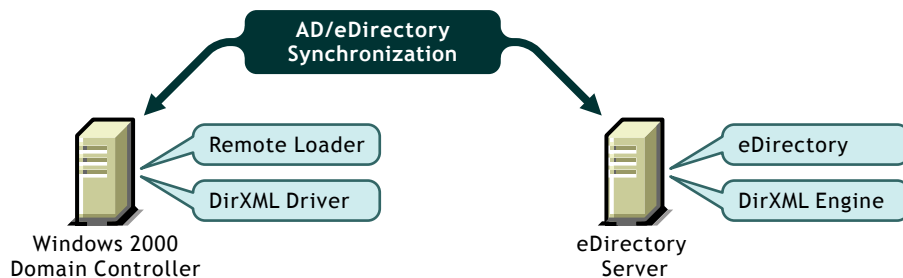
Dual server configurations can be set up in two ways. The first configuration places eDirectory, the DirXML engine, and the driver on a separate computer from the Active Directory domain controller, leaving the domain controller free of any DirXML software.

Figure 2 Dual Server Configuration (1)



The second configuration places eDirectory and the DirXML engine on one computer and the driver and Remote Loader on the Active Directory domain controller.

Figure 3 Dual Server Configuration (2)



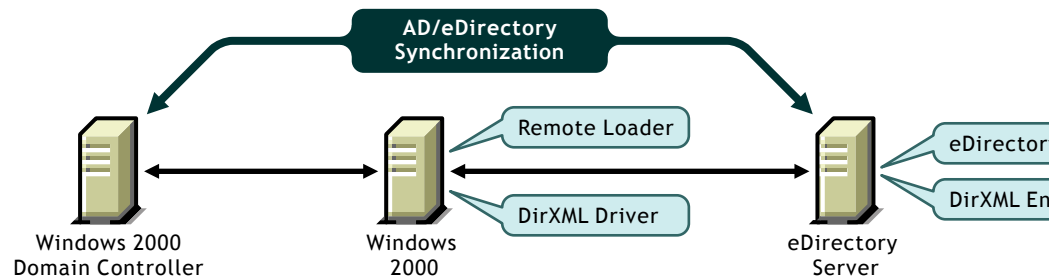
Both configurations eliminate the performance impact of hosting eDirectory and the DirXML engine on the domain controller. The first configuration is attractive if corporate policy disallows running the driver on your domain controller. The second solution is attractive if your eDirectory and DirXML installations are on a platform other than Windows 2000.

◆ **Triple Server**

A three-server configuration can be used if you have platform requirements and domain controller restrictions in place. It's more complicated to set up this configuration, but it accommodates the constraints of some organizations.

In this configuration, eDirectory and DirXML run on one computer, the Remote Loader and the driver run on a second Windows 2000 computer, and the Active Directory domain controller runs on the third computer.

Figure 4 Triple Server Configuration



Security Options

The driver can run in several security modes. The major factors to consider are authentication, encryption, and use of the DirXML Remote Loader. If you are using the Remote Loader you must consider security settings on the Remote Loader channel between DirXML and the driver plus the settings between the driver and Active Directory. If you have Windows 2000 SP3 or later, you'll want to consider a security option called signing.

A simple prescription for managing security is not possible because the security profile available from Windows 2000 varies with service pack, DNS server infrastructure, domain policy, and local policy settings on the Windows 2000 servers. The following sections explain your security choices and provide suggested configurations. Pay close attention to security when implementing your driver and when upgrading components.

Security Parameters

You can set the following parameters during installation or later, in the Driver Parameters page. Understanding how the parameters work together and work with the operating system will help you define your approach to security for DirXML data synchronization.

- ◆ **Authentication ID:** This is the account the driver uses to access domain data. Valid username formats are

Username	Format
User Principal Name	user@domain.com
Domain name	user
Fully Qualified Domain name	domain\user

If the driver is installed on the Domain Controller and LSA access to Active Directory has not been restricted, you don't have to set an Authentication ID; the driver will use its local identity for authentication.

- ◆ **Application Password:** This is the password for the Authentication ID account. Set a password whenever you use an Authentication ID.

- ◆ **Authentication Context:** This is an LDAP URL that encodes the DNS name of the Active Directory domain controller. For example: LDAP://mycontroller.mydomain.com.

To configure secure communication using SSL, add the LDAP SSL port number to the DNS server hostname (for example: LDAP://mycontroller.mydomain.com:636). Be aware that SSL will only work if you have set up a Certificate infrastructure and have imported certificates to your Windows 2000 servers. See the Microsoft documentation for Certificate Services for more detail.

If the driver is running on the Domain Controller and you don't specify an authentication context, the driver will address its connection to the local machine.

- ◆ **Use Secure Authentication:** When this option is set to Yes, the driver will negotiate Kerberos or NTLM authentication to Active Directory.

When this option is set to No, the driver uses an LDAP simple bind. Simple bind is usually unacceptable because it transmits passwords in clear text on the wire. However if you have configured SSL secure communication, then the password is sent on an SSL encrypted pipe and is secure.

- ◆ **Use SSL:** This parameter controls encryption if you connect to Active Directory using the LDAP SSL port number. By default the parameter is set to No, which means the SSL secure communication will drop out after simple bind authentication completes. Communication after authentication will use clear text.

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 will slow the general performance of your servers.

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

- ◆ **Use Signing / Sealing:** This flag enables signing and sealing of the Active Directory connection if you are not using the LDAP SSL port. Signing ensures that a malicious computer is not intercepting data. Sealing encrypts the data so that it cannot be viewed by a network monitor.

This setting is effective *only* if you are running the Windows 2000 Security Rollout Package SP1 (SRP1) or Windows 2000 SP3, with Internet Explorer 5.5 SP2 installed on both Windows 2000 servers, and will enable signing and encryption on a Kerberos or NTLM authenticated connection.

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

- ◆ **Keep Credentials:** This parameter instructs the driver to use an updated authentication method to maintain its connection to Active Directory. The updated method is important on systems that have upgraded to Windows 2000 Security Rollout Package 1 (SRP1) or Windows 2000 SP3. If you are using an earlier release of Windows and are not experiencing connection problems, set this parameter to No.

Authentication Options

The three authentication methods used by the driver are listed below. If you have installed a different security package into the Microsoft Security Service Provider Interface (SSPI) infrastructure, you will have additional options.

- ◆ **Use Process Identity:** This mode is selected when you leave the Authentication ID blank. Typically, eDirectory runs as a service and you will receive Local Service Account (LSA)

rights to Active Directory. Unless policy or local Active Directory security settings disallow access for the LSA, this level of rights works when you are running the driver on your Active Directory server. This only works if the driver is running on the Domain Controller.

- ♦ **Simple Bind:** Passes the user name and password in clear text. This option should only be used with SSL.
- ♦ **Kerberos / NTLM authentication:** NTLM is the standard Domain authentication used on Windows NT 4. It is weaker than Kerberos because key lengths might not be as long and it does not support mutual authentication. However, NTLM has been used for years with Domain authentication and is acceptable for most uses.

Kerberos is the new Windows 2000 authentication and is the preferred method for future authentications with Microsoft. It implements a third-party mutual authentication scheme and is generally stronger than NTLM. The driver is not notified which authentication scheme is being used.

Recommended Security Configurations

Using Remote Loader

Because authentication is dependent on several parameters such as the Windows 2000 support pack, your DNS infrastructure, and policy and registry settings, the most reliable means of authentication is to install the driver on the computer hosting Active Directory and then use the DirXML Remote Loader to connect to the DirXML engine, as illustrated in [Figure 3, “Dual Server Configuration \(2\),” on page 14](#). With this configuration, you will be most successful if you set the driver parameters as follows.

Authentication Context: Blank
Authentication ID: User principal name
Password: Password for the specified Authentication ID
Use Secure Authentication: Yes
Use SSL: No
Signing: No
Sealing: No
Keep Credentials: Yes

Insulating the Domain Controller

If you do not want to run the driver on your Active Directory domain controller, as shown in [Figure 2, “Dual Server Configuration \(1\),” on page 14](#) and in [Figure 4, “Triple Server Configuration,” on page 15](#), set the driver parameters as follows:

Authentication Context: `LDAP://hostname`
Authentication ID: User principal name
Password: Password for the specified Authentication ID
Use Secure Authentication: Yes
Use SSL: No
Signing: No, unless you have installed the most recent Windows 2000 support pack and Internet Explorer 5.5 SP2 on both servers.
Sealing: No, unless you have installed the most recent Windows 2000 support pack and Internet Explorer 5.5 SP2 on both servers.
Keep Credentials: Yes

Using SSL

SSL is an attractive option if you have already built up a Certificate Services infrastructure and have imported the proper certificates. It is not necessary to install Certificate Services just for DirXML data synchronization because either of the previous options provides secure communication.

Authentication Context: LDAP://*hostname*:636

Authentication ID: User principal name

Password: Password for the specified Authentication ID

Use Secure Authentication: No

Use SSL: Yes or No, depending on level of encryption you want. Choose Yes for both secure authentication and communication, No for secure authentication only.

Signing: No

Sealing: No

Keep Credentials: Yes

Prerequisites

- ◆ Novell DirXML 1.1 or higher
- ◆ Windows 2000 Professional or Server with Service Pack 2
- ◆ The server must be a member of the AD domain
- ◆ Internet Explorer 5.5 or later
- ◆ Active Directory domain controller DNS name
- ◆ Novell iManager or ConsoleOne®

For information about setting up iManager, see the *DirXML 1.1a Administration Guide* (<http://www.novell.com/documentation/lg/dirxml11a/dirxml/data/a2iii88.html>).

NOTE: DirXML management using ConsoleOne is explained in the *DirXML 1.1 documentation* (<http://www.novell.com/documentation/lg/dirxml11/index.html>).

Installing and Upgrading

Installing the Active Directory Driver (Local Install)

In a local configuration, the driver is installed on the same computer that is hosting the DirXML engine. To install the driver locally, run the DirXML 1.1 installation program and select DirXML Engine and Drivers > DirXML Driver for Active Directory.

After installation, you must set up the driver as explained in “[Post-Installation Setup](#)” on page 19.

Installing the Active Directory Domain Driver (Remote Loader Installation)

In a remote configuration, the driver and the Remote Loader service are installed on a computer other than the one hosting the DirXML engine. To set up a remote configuration:

- 1** Insert the DirXML CD and click Next at the Welcome screen.
- 2** At the License page, click I Accept.

- 3** At the Components dialog box, select DirXML Remote Loader Service, then click Next.
- 4** Accept the default installation path for the Remote Loader, then click Next.
- 5** Mark the following items, then click Next.
 - ◆ DirXML Remote Loader Service
 - ◆ DirXML Driver for Active Directory
- 6** Review the Product Summary, then click Finish to install the Remote Loader files.
- 7** When prompted, create a shortcut.
- 8** Run the DirXML Remote Loader Configuration Wizard from your desktop.
- 9** At the Welcome page, click Next.
- 10** Keep the default Command Port number, then click Next.
- 11** Keep the default Configuration File Name, then click Next.
- 12** In the DirXML Driver dialog box, mark Native, then click Next.
- 13** In the Connection to DirXML dialog box, leave the default Port settings.
Record the port number for later use during driver setup.
- 14** Set Trace Level to 3 so that you'll get minimal tracking data for troubleshooting, specify a location and filename for trace file, then click Next.
- 15** Mark Install the Remote Loader Instance as a Service, then click Next.
- 16** Set Remote Loader and Driver Object passwords.
Record the passwords for later use during driver setup.
- 17** Review the summary, then click Finish.
- 18** When prompted, start the service.
Continue with driver setup as explained in [“Post-Installation Setup” on page 19](#).

Upgrading the Driver

To upgrade your DirXML Driver for Active Directory, run the DirXML installation program and select DirXML Engine and Drivers > DirXML Driver for Active Directory. You can do this at the same time that you install the engine or you can do it after the engine is installed.

The new driver replaces the previous driver but keeps the previous driver's configuration. Simply confirm the settings read by the driver import file.

No post-installation configuration is required if you are upgrading the driver. However, you must restart the driver. To do this in Novell iManager, select DirXML Management > Overview. Then click the driver status indicator in the upper right corner of the driver icon and click Start Driver.

Post-Installation Setup

Setup is not required if you are upgrading an existing driver.

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

- ◆ [“Creating an Admin User” on page 20](#)

- ◆ “Setting Up the Driver” on page 21
- ◆ “Configuring the Driver for Remote Loader” on page 22
- ◆ “Starting the Driver” on page 23
- ◆ “Migrating and Resynchronizing Data” on page 23
- ◆ “Activating the Driver” on page 25

Creating an Admin User

We recommend that you create a user with Admin privileges to be used exclusively by the driver to authenticate into Active Directory. Doing this keeps the DirXML Admin account insulated from changes to other Admin accounts.

To create an Admin User:

- 1** Click Start > Programs > Administrative Tools > Active Directory Users and Computers.
- 2** From Active Directory Users and Computers, select the container where you want to add the user, then click Create a New User.
- 3** Enter the Fullname, which is the AD object name, and enter the User logon name, which is the AD authentication name.

Figure 5 Creating an Active Directory User for the Driver

The screenshot shows the 'New Object - User' dialog box. At the top, it says 'Create in: mercury.com/Users'. Below that, there are several input fields:

- First name: Novell
- Initials: (empty)
- Last name: Dirxml
- Full name: Novell Dirxml
- User logon name: novelldirxml
- Domain dropdown: @mercury.com
- User logon name (pre-Windows 2000): MERCURY\novelldirxml

 At the bottom, there are three buttons: '< Back', 'Next >', and 'Cancel'.

Record the logon name plus the domain for later use during driver setup. For example, record novelldirxml@mercury.com.

- 4** Click Next, then set the password for the new user. Mark Password Never Expires so that a password policy won't disable the driver unexpectedly.
- 5** Click Next, review the summary, then click Finish.
- 6** In the Tree view, select Builtin > Administrator's properties > Members > Add.

- 7** Select the full name of the user you created, click Add, click OK, then click OK again.
- 8** Close the Active Directory Users and Computers window.
- 9** In the Administrative Tools window, select Domain Controller Security Policy.
- 10** In Tree View, expand Security Settings > Local Policies > User Rights Assignment.
- 11** Select Log On As a Service > Security > Add > Browse.
- 12** Select the user you created, click Add, click OK, click OK, then click OK again.
- 13** Close the Domain Controller Security Policy.
- 14** Reboot the system.

Setting Up the Driver

The Create Driver Wizard will help you import a preconfigured Active Directory driver file. This file will create and configure the objects needed to make the driver work properly.

- 1** In Novell iManager, select DirXML Management > Create Driver.
- 2** Select a driver set.
If you place this driver in a new driver set, you must specify a driver set name, context, and associated server.
- 3** Mark Import a Preconfigured Driver from the Server, then select the ADDriver.xml file.
The preconfigured driver file is installed on the Web server when you set up iManager.
- 4** During the import you will be prompted to enter the following information:

Field	Description
Driver Name	This is 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 your driver name.
Authentication ID	Use the User Principal Name or fully qualified NT domain name. If you don't specify an Authentication User, you will use the local identity to negotiate data access.
Password	Enter the password for the user account specified in Authentication ID.
Authentication Server	Leave Server blank to use the local machine. If you qualify the LDAP URL with port 636, you're specifying LDAP over SSL. This only works if you have a Certification Authority and certificates installed on the local machine.
Domain Name (in LDAP format)	The driver requires LDAP formatted domain names dc=mydomain,dc=com
Domain DNS Name (DNS format)	The driver requires DNS formatted domain names mydomain.com

Field	Description
Polling Interval	<p>eDirectory sends changes to Active Directory as they happen. However, changes to Active Directory are sent to eDirectory only as often as the configured polling interval. The default is 15 minutes.</p> <p>IMPORTANT: The polling interval affects system performance. We recommend that you set the interval to 1 minute. You can then set it to a higher interval if it affects system performance.</p>
Secure Authentication	<p>When this option is set to Yes, the driver selects a negotiation for Kerberos or NTLM authentication between Windows 2000 servers. When this option is set to No, the driver selects an LDAP simple bind.</p> <p>Review “Security Options” on page 15 to be sure you are setting up security that meets your needs.</p>
Base container in eDirectory	<p>Specify the container using slash format, for example</p> <p>acmetree\least\users</p> <p>If this container doesn't exist, you must create it before you start the driver.</p>
Base container in AD	<p>Specify the container using LDAP comma-delimited names, for example</p> <p>CN=Users,DC=MyDomain,DC=com</p> <p>Make sure you know whether your organization uses CN or UI for naming.</p> <p>If the target container doesn't exist, you must create it before you start the driver.</p>

5 When the import is finished, click Yes to define security equivalence on the imported driver.

5a Click Add, then select an object with Admin rights.

5b Click Apply, then click Close.

6 Click Yes to specify excluded users:

6a Click Add, then select any users you want to exclude (such as the admin user).

6b Click Apply, then click Close.

7 Click Finish.

The DirXML objects necessary for synchronization with Active Directory have now been created.

Configuring the Driver for Remote Loader

If you installed the driver, along with the Remote Loader service, on a computer other than the server hosting the DirXML engine, you need to complete the following steps to connect the driver and the Remote Loader.

1 In iManager, click DirXML Management > Overview.

2 Locate the driver in its driver set.

3 Click the driver icon to open the Driver Overview page.

4 Click the driver icon again to open the Modify Object page.

- 5** Fill in the following fields with information specific to your environment:
 - ◆ Driver Object Password
 - ◆ Remote Loader Connection Promontories
 - ◆ Remote Loader Password
- 6** Click Apply.

Starting the Driver

- 1** In iManager, select DirXML Management > Overview.
- 2** Locate the driver in its driver set.
- 3** Click the driver status indicator in the upper right corner of the driver icon and click Start Driver.

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

Migrating and Resynchronizing Data

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

- ◆ **Migrate data from eDirectory:** Allows you to select containers or objects you want to migrate from eDirectory to an application. When you migrate an object, the DirXML engine applies all of the Matching, Placement, and Create rules, as well as the Subscriber filter, to the object.
- ◆ **Migrate data into eDirectory:** Allows you to define the criteria DirXML uses to migrate objects from an application into Novell eDirectory. When you migrate an object, the DirXML engine applies all of the Matching, Placement, and Create rules, as well as the Publisher filter, to the object. Objects are migrated into eDirectory using the order you specify in the Class list.
- ◆ **Synchronize:** DirXML looks in the Subscriber class filter and processes all objects for those classes. Associated objects will be merged. Unassociated objects will be processed as Add events.

To use one of the options explained above:

- 1** In iManager, select DirXML Management > Overview.
- 2** Locate the driver set containing the Active Directory driver, then double-click the driver icon.
- 3** Click the appropriate migration button.

If you will be migrating more than 1000 objects into eDirectory, you should adjust LDAP policies as explained in the following section, [“Migrating a Large Number of Objects” on page 23](#).

Migrating a Large Number of Objects

Active Directory defines several policies to control the impact of LDAP operations on the system as a whole. Under normal operations, the Active Directory driver operates within the default settings for these policies. However, you might find that some policy settings keep the Migrate into eDirectory feature of DirXML from working properly. This can happen if you are migrating

a large number of objects (typically greater than 1000) and the query operation DirXML uses to find objects exceeds one of the policies.

You can solve the problem in two ways:

- ◆ **Reduce the number of objects migrated in a single operation.** Do this by restricting the selection criteria. Consider migrating each class type separately (containers first, user objects second, and groups third). If that is not sufficient, then limit the scope of each migration to a specific container in Active Directory. You can even limit the scope by using wildcards in an attribute name (for instance, migrate Surname 'A*', 'B*', and etc....), but this can quickly become burdensome.
- ◆ **Change Active Directory policies for the migration.** After migration is complete, you can return policies to the best value determined by your organization. The following policies affect migration:
 - ◆ **MaxTempTableSize** (default: 10,000 records) This policy sets the maximum number of objects that DirXML will process on a migration. Set this policy to a number greater than the number of objects you expect to migrate.
 - ◆ **MaxPageSize** (default: 1000 records) This policy is the most likely to interfere with migration. The value must be higher than the maximum number of objects you expect to migrate.
 - ◆ **MaxResultSetSize** (default: 262144 bytes) This policy defines the maximum payload that Active Directory returns on a given query. Migrate into eDirectory requests only a small amount of information for any given object, but the total space used increases with the number of objects returned in the query. Generally, this value should be increased in proportion to the increase in MaxPageSize. A good rule of thumb is to use 300 times the MaxPageSize value.
 - ◆ **MaxQueryDuration** (default: 120 seconds) This policy is unlikely to be a problem unless you are migrating 25,000 or more objects or the domain controller is heavily used.

You can use the Microsoft ntdsutil.exe utility to change policies.

- 1** Log on as an administrative user in Active Directory.
- 2** Run ntdsutil.exe by clicking Start > Run, then entering **NTDSUTIL**.
- 3** At the NTDSUTIL prompt, enter **LDAP policies**.
- 4** Enter **connections** to set up a connection to your domain controller.
- 5** Enter **Connect to server server-dns-name**, replacing *server-dns-name* with the DNS name of your domain controller.
- 6** Enter **q** to return to the LDAP Policy prompt.
- 7** Enter **Show values** to get your current policy settings.
Write down these settings.
- 8** Modify policies as necessary to accommodate your migration plans.
- 9** Enter **Commit changes** to activate the new policy.
- 10** Enter **q** to exit NTDSUTIL.

You are now ready to migrate objects. After you have completed the migration, you can use the same procedure to set your policies back to their original values.

Further instructions for managing Active Directory policies with ntdsutil.exe are found in [Microsoft Knowledge Base Article 315071 \(http://support.microsoft.com/default.aspx?scid=kb%3Ben-us%3B315071\)](http://support.microsoft.com/default.aspx?scid=kb%3Ben-us%3B315071).

Activating the Driver

DirXML and DirXML drivers must be activated within 90 days of installation, or they will shut down. At any time during the 90 days, or afterward, you can choose to activate DirXML products to a fully licensed state.

To activate your driver, you should:

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

For more information about completing these tasks, refer to [Activating Your DirXML Product \(http://www.novell.com/documentation/lg/dirxml11a/index.html\)](http://www.novell.com/documentation/lg/dirxml11a/index.html).

3

Customizing the DirXML Driver for Active Directory

The DirXML[®] Driver for Active Directory driver includes a sample configuration that you can use as a starting point for your deployment.

Most DirXML deployments require you to make changes to the sample configuration. For example, if you need only one-way data synchronization, or if the attributes you're synchronizing are different from those provided in the sample, you'll need to customize the driver.

This section covers the following customization topics:

- ◆ “Configuring Driver Parameters” on page 27
- ◆ “Configuring Data Synchronization” on page 28

Configuring Driver Parameters

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

Changing Driver Authentication and Passwords

Driver authentication gives the driver administrative privileges necessary to make data updates. During the driver installation, you were prompted to specify an Authentication ID and password. Consider using a different account than Administrator to ease password change management.

NOTE: No administrative equivalent is needed if the Active Directory driver is running on the same computer where Active Directory is installed and if local security context rights are allowed.

- 1** In iManager, click DirXML Management > Overview.
- 2** Locate the driver in its driver set.
- 3** Click the driver icon to display the Driver Overview page.
- 4** Click the driver icon again to display the Modify Object page.
- 5** Find the Authentication field and enter the following authentication information:
 - ◆ The user principal name in the Authentication ID field (for example, enter Administrator@domain.com).
 - ◆ The domain controller you are synchronizing with in the Authentication Context field (for example, enter LDAP://Domain controller.domain.com).

You can choose not to enter the authentication ID and context if you are running the ADDRIVER.DLL shim on the same machine that hosts your Active Directory domain.

- ◆ Authentication User password

You need to change the password in this field each time your administrator password changes.

- 6 Click Apply.

Configuring Data Synchronization

The real power of DirXML is in managing the shared data itself. This section covers some common customizations for the Active Directory driver, including

- ◆ “Enabling Exchange 2000 Mailboxes” on page 28
- ◆ “Managing Login Names” on page 29
- ◆ “Using Rename Instead of Synchronizing a Naming Attribute” on page 30

Enabling Exchange 2000 Mailboxes

You can enable Exchange 2000 mailboxes for an Active Directory object. You do this by adding several attributes to define which Exchange Server will host the mailbox, what the mailbox name will be, and who has rights to view the mailbox. You can also configure the mailbox to restrict receipt to a known set of e-mail senders, or to automatically reject e-mail from others. The driver supports the attribute syntaxes involved, but does not automatically create the attributes for you. You must create these attributes in your own style sheets.

Exchange Attributes

You can generate Exchange attributes for Add Events on the Subscriber channel. This section describes the Exchange attributes supported by the Active Directory DirXML Driver.

The way you generate Exchange mailbox attributes and which style sheets you use depends on your specific deployment rules. The attributes defined in this section are in the application namespace. The DirXML Schema Mapping rule will convert attributes you specify between the application namespace and the eDirectory namespace and will pass through any other attributes unchanged.

msExchHomeServerName

This is the name of the Exchange server that will host the mailbox. The server name is presented in a legacy Exchange format as a fully-qualified distinguished name in slash format. For example:

```
/O=Hyperion/OU=First Administrative Group/CN=Configuration/CN=servers/CN=TIMS-DELL
```

To find the legacy name:

- 1 Run the following command to get your domain name and other root information:

```
ldifde -f domain.txt -d "" -r "(objectClass=*)" -  
p Base -l "defaultNamingContext,configurationNamingContext,  
rootDomainNamingContext,dnsHostName" -s myserver.mydomain.com -a  
administrator@mydomain.com mypassword
```

You should see a result similar to this:

```
dn:  
changetype: add
```

```
defaultNamingContext: DC=td,DC=provo,DC=novell,DC=com
configurationNamingContext:
CN=Configuration,DC=td,DC=provo,DC=novell,DC=com
rootDomainNamingContext: DC=td,DC=provo,DC=novell,DC=com
dnsHostName: tims-dell.td.provo.novell.com
```

- 2 Replace the path in the -d option with the configurationNamingContext listed in the results.

For example:

```
ldifde -f exchservers.txt -d "cn=configuration,dc=mydomain,dc=com" -r
(objectClass=msExchExchangeServer) -p Subtree -l
"legacyExchangeDN" -s myserver.mydomain.com -a
administrator@mydomain.com mypassword
```

You should see a result similar to:

```
dn: CN=TIMS-DELL,CN=Servers,CN=First Administrative
Group,CN=Administrative Groups,CN=Hyperion,CN=Microsoft
Exchange,CN=Services,CN=Configuration,DC=td,DC=provo,DC=novell,DC=com
changetype: add
legacyExchangeDN:
/o=Hyperion/ou=First Administrative Group/cn=Configuration/cn=Servers/
cn=TIMS-DELL
```

- 3 Use the legacyExchangeDN information for the msExchHomeServerName attribute.

mailNickname

This attribute is the local name for the user. You can choose the user's CN.

mail

This attribute is the user's full name, including the domain, such as user@domain.com.

msExchMailboxSecurityDescriptor

This attribute controls who has access to the mailbox. This is a standard Active Directory Access Control List. The current ADDRIVER.DLL does not handle the ACL syntax in general, but has been updated to handle this attribute specifically. You can treat this attribute as a "state" (Boolean) type and the DLL will generate an ACL that grants the proper rights to the user.

authOrig

This attribute is a list of senders that have the right to send mail to this mailbox. If you set this attribute, senders not in this list will be rejected.

unauthOrig

This attribute is a list of senders that are not allowed to send mail to this mailbox.

Managing Login Names

In eDirectory, your login name is your fully distinguished name, such as jsmith.sales.ny.acme. In Active Directory, you have two login names: your NT domain name, such as jsmith, and your user principal name, such as jsmith@acme.com. You can log in with either name. In Active Directory, you are also identified by your Relative Distinguished Name (RDN), which is generated by the application that was used to create your account.

The default configuration for the Active Directory driver will synchronize RDN values. This can be a problem if you use the Microsoft Management Console (MMC) to create or manage accounts in Active Directory. MMC generates RDNs in the form Lastname, Firstname. The potential difficulty is that your eDirectory login name will change to *lastname\,firstname.context*. To avoid assigning this difficult login name, create a publisher event transform that strips rename events.

Using Rename Instead of Synchronizing a Naming Attribute

The naming attribute for an object (CN on User, for example) shouldn't be in the driver's filter. Directly synchronizing a naming attribute is not allowed by AD. However, the AD driver supports rename commands from DirXML, so you may use that method instead if necessary.

4

Documentation Content Updates

This section contains information on documentation content changes that have been made in the *Implementation Guide* for the DirXML Driver for Active Directory. The information will help you to keep current on updates to the documentation.

The information is grouped according to the date the documentation updates were published. Within a dated section, the changes are alphabetically listed by the names of the main table of contents sections for Policy and Distribution Services.

The documentation is provided on the Web in two formats: HTML and PDF. The HTML and PDF documentation are both kept up-to-date with the documentation changes listed in this section.

If you need to know whether a copy of the PDF documentation you are using is the most recent, the PDF document contains the date it was published in the Legal Notices section immediately following the title page.

The documentation was updated on the following dates:

- ♦ [November 15, 2002](#)
- ♦ [April 22, 2003](#)
- ♦ [September 30, 2003](#)

November 15, 2002

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

- ♦ [Installing the DirXML Driver for Active Directory](#)

Installing the DirXML Driver for Active Directory

The following updates were made in this section:

Location	Change
"Security Parameters" on page 15	Added information about use of local identity.
"Creating an Admin User" on page 20	The procedure was incomplete. Added steps to complete it.
"Migrating and Resynchronizing Data" on page 23	Corrected explanation of Synchronize. Synchronize will update objects regardless of association.

April 22, 2003

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

- ◆ [Installing the DirXML Driver for Active Directory](#)

Installing the DirXML Driver for Active Directory

The following updates were made in this section:

Location	Change
“Migrating a Large Number of Objects” on page 23	Explained Active Directory policy changes that might be required when you migrate more than 1000 objects into eDirectory.

September 30, 2003

A new section was added, [“Using Rename Instead of Synchronizing a Naming Attribute” on page 30](#).