Novell DirXML® Driver for LDAP

1.6

www.novell.com

IMPLEMENTATION GUIDE

January 15, 2004

Novell®

Legal Notices

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

www.novell.com

DirXML Driver for LDAP Implementation Guide
January 15, 2004

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

Novell Trademarks

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.

Third-Party Trademarks

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

Contents

About This Guide

This guide explains how to install and configure the DirXML® Driver for LDAP.

The guide contains the following sections:

- Chapter 1, “Introducing the DirXML Driver for LDAP,” on page 9

  This section introduces new features and explains the default driver configuration.

- Chapter 2, “Installing the LDAP Driver,” on page 15

  This section covers both the installation and upgrade processes as well as post-installation setup tasks.

- Chapter 4, “Customizing the LDAP Driver,” on page 25

  This section explains how to customize driver parameters and data synchronization. It provides examples for common customizations.

- Chapter 5, “Troubleshooting,” on page 35

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).

Documentation Updates

For the most recent version of this document, see the DirXML Drivers Documentation Web Site (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 LDAP

This section covers the following topics:

Driver Overview

The DirXML® Driver for LDAP synchronizes data between Novell® eDirectory™ and the following LDAP-compliant directories:

- Netscape* Directory Server
- iPlanet* Directory Server
- IBM* SecureWay Directory
- Critical Path* InJoin* Directory
- Oracle* Internet Directory

This driver runs on all platforms supported by eDirectory and uses the Lightweight Directory Access Protocol to bidirectionally synchronize changes between eDirectory and the connected LDAP-compliant directory.

The driver leverages the change log mechanism in these directories to recognize data changes and communicate them to eDirectory through DirXML. Additional software and changes to the LDAP-compliant directory are not required.

Because of this flexible model for communicating, the driver can synchronize with LDAP-compliant directories running on platforms other than those supported by eDirectory, such as AIX*, HP/UX*, OS/400*, and OS/390*.

New Features

The following section contains information about the new driver features, as well as new features provided in DirXML 2.0.

Driver Features

- Instead of two sample configurations, a single sample configuration is provided that includes the option to choose between Flat placement and Mirror placement in hierarchal structures.
- An optional driver parameter has been added to let you specify preferred object classes. See "Preferred Object Classes" on page 27.

- Support for Password Synchronization 2.0 has been added.

  The driver shim works the same way, but new policies have been added to the sample driver configuration to support Password Synchronization 2.0.

  You can set or modify the LDAP password using a password from DirXML, and you can check the LDAP password to see if it matches the DirXML password.

  You could also use a style sheet to manufacture a password to be sent back to DirXML, such as a password based on the user's last name. However, LDAP does not support providing the user's actual LDAP password to DirXML.

  See the description of the different scenarios for Password Synchronization in "Implementing Password Synchronization" in the *Novell Nsure Identity Manager 2 Administration Guide*.

- Role-Based Entitlements features are supported as an option in the sample driver configuration.

  Because the LDAP protocol does not provide the ability to disable an account, only the ability to delete an account, use caution when revoking access to LDAP accounts using Entitlement Policies.

  NOTE: If disabling accounts is desired, check your LDAP application; some applications might provide the ability to disable even though it's not in the LDAP protocol. If so, you might be able to add a disable option by customizing the policies in the driver configuration and updating the interpretive variables in the driver manifest.

  For information about Role-Based Entitlements, see "Using Role-Based Entitlements" in the *Novell Nsure Identity Manager 2 Administration Guide*.

- Driver heartbeat is supported. See "Driver Heartbeat" inthe *Novell Nsure Identity Manager 2 Administration Guide*.

DirXML 2.0 Features

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

Password Management

The new password management framework includes the following benefits:

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

Policy Builder Interface and DirXML Script for Creating Policies

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

Role-Based Entitlements

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

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

Novell Nsure Audit

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

Global Configuration Values

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

GCVs allow you to specify settings for new DirXML features such as Password Synchronization, as well as settings that are specific to the function of an individual driver configuration. Some GCVs are provided with the drivers, but you can also add your own. You can refer to these values in a policy to help you customize your driver configuration.

Driver Heartbeat

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

Default Driver Configuration

DirXML fundamentals are explained in the Novell Nsure Identity Manager 2 Administration Guide. This section discusses implementations, additions, or exceptions specific to this driver.

Data Flow

Publisher and Subscriber Channels

The driver supports Publisher and Subscriber channels:

- The Publisher reads information from the LDAP directory change log and submits that information to eDirectory via the DirXML engine.

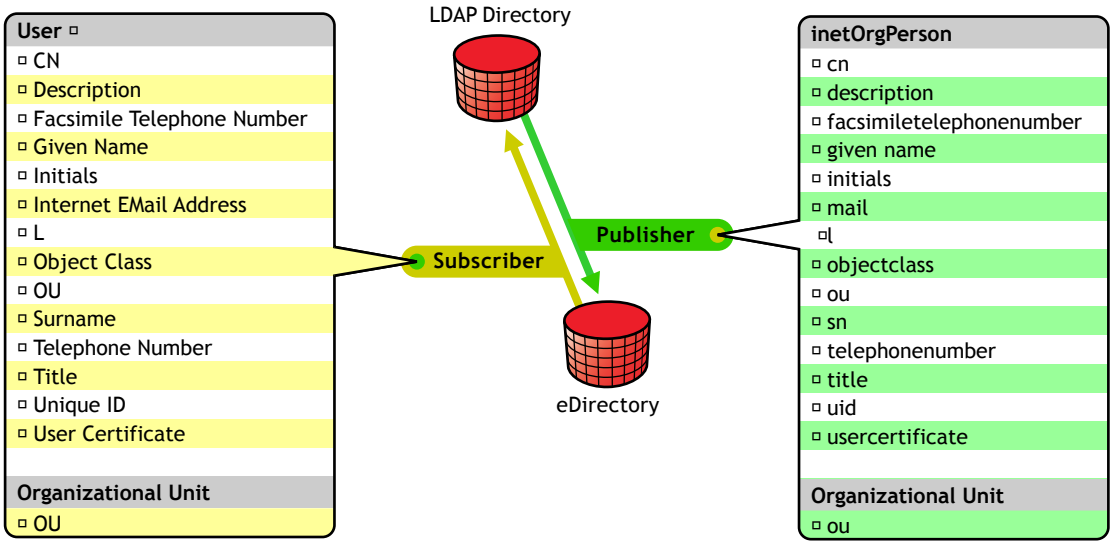
  By default, the Publisher checks the log every 20 seconds, processing up to a 1000 entries at a time, starting with the first unprocessed entry.

- The Subscriber watches for additions and modifications to eDirectory objects and issues LDAP commands that will make changes to the LDAP directory.

Filters

DirXML uses filters to control which objects and attributes are shared. The default filter configurations for the LDAP driver allow objects and attributes to be shared, as illustrated in the following figure:

Figure 1 LDAP Driver Filters



Policies

Policies are used to control data synchronization between the driver and eDirectory. The LDAP driver comes with two preconfiguration options to set up policies.

- The Flat option implements a flat structure for users in both directories.

  With this configuration, when user objects are created in one directory, they are placed in the root of the container you specified during driver setup for the other directory (the container name does not have to be the same in both eDirectory and the LDAP directory). When existing objects are updated, their context is preserved.

- The Mirror option matches the hierarchical structure in the directories.

  With this configuration, when new user objects are created in one directory, they are placed in the matching hierarchical level of the mirror container in the other directory. When existing objects are updated, their context is preserved.

Except for the Placement policy and the fact that the Flat configuration doesn't synchronize Organizational Unit objects, the policies set up for these options is identical.

Default policies are detailed in the following table. These policies and the individual rules they contain can be customized through Novell iManager as explained in Chapter 4, "Customizing the LDAP Driver," on page 25.

Policy	Description
Mapping	Maps the eDirectory User object and selected properties to an LDAP inetOrgPerson. Maps the eDirectory Organizational Unit to an LDAP organizationalUnit. By default, more than a dozen standard properties are mapped. Additionally, the driver will read the LDAP schema the first time you open the Schema Mapping policy in Novell iManager, allowing you to easily map additional properties if necessary.
Publisher Create	Specifies that in order for a User to be created in eDirectory, the cn, sn, and mail attributes must be defined. In order for an Organization Unit to be created, the ou attribute must be defined.
Publisher Placement	With the Simple placement option, new user objects created in the LDAP directory are placed in the container in eDirectory that you specify when importing the driver configuration. The user object is named with the value of cn. With the Mirror placement option, new user objects created in the LDAP directory are placed in the eDirectory container that mirrors the object's LDAP container.
Matching	Specifies that a user object in eDirectory is the same object as an inetOrgPerson in the LDAP directory when the e-mail attributes match.
Subscriber Create	Specifies that in order for a User to be created in the LDAP directory, the CN, Surname, and Internet Email Address attributes must be defined. In order for an Organization Unit to be created, the OU attribute must be defined.
Subscriber Placement	If you choose Flat placement option during the import of the driver configuration, new user objects created in eDirectory are placed in the Users\Active container in the LDAP. If you choose Mirrored placement during the import of the driver configuration, new user objects created in the eDirectory are placed in the LDAP directory container that mirrors the object's eDirectory container.

2 Installing the LDAP Driver

The DirXML® Driver for LDAP 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 Novell Nsure Identity Manager 2 Administration Guide.

The driver can also be installed separately, by running the DirXML installation and selecting only the LDAP driver.

This section covers the following installation topics:

Planning Considerations

In this section:

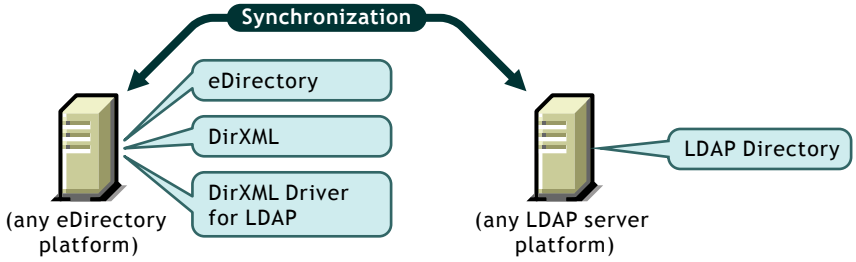
Where to Install the LDAP Driver

A DirXML driver can be installed on the same computer where Novell® eDirectory™ and the DirXML engine are installed. This installation is referred to as a local configuration. If platform or policy constraints make a local configuration difficult, a DirXML driver can be installed on the computer hosting the target application. This installation is referred to as a remote configuration.

Although it is possible to install the LDAP driver in a remote configuration, it provides no additional flexibility because this driver can run on any eDirectory platform and communicates with the LDAP server on any platform across the wire via the LDAP protocol.

In a local configuration, you install the LDAP driver on the same computer where eDirectory and the DirXML engine are installed, as shown in the following figure.

Figure 2 Installation Location



Information to Gather

During installation and setup, you'll be prompted for the information such as the following.

- Whether to use the Flat or Mirror option for synchronizing hierarchical structure. See "Policies" on page 12.
- The eDirectory and LDAP directory containers that you want to hold synchronized objects.
- The eDirectory user object to assign as a security equivalent for the driver and the objects to exclude from synchronization.
- The LDAP object and password used to provide driver access to the LDAP directory.

See the table in "Importing the Driver" on page 19.

Assumptions about the LDAP Data Source

If you are using the Publisher channel to send data to eDirectory about changes in the LDAP directory, you must understand the change log mechanism for your LDAP directory. The change log is the source of LDAP event information for the driver.

System Prerequisites

- Novell DirXML 2.0 or later
- The system requirements of DirXML 2.0 or later
- One of the following LDAP directories:
  - Netscape Directory Server 4.*x*
  - iPlanet Directory Server 5.0 or greater
  - IBM SecureWay Directory 3.2 or greater
  - Critical Path InJoin Directory 3.1 or greater
  - Oracle Internet Directory 2.1.1 or greater

Installation

In this section:

Installing the LDAP Driver

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

After installation, you must configure the driver as explained in “Setting Up the Driver” on page 17.

IMPORTANT: On Solaris* or Linux*, the driver package is installed by default when you install DirXML.

Setting Up the Driver

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

If this is the first time the LDAP driver has been used, you should complete the setup tasks in the following sections:

Preparing the LDAP Server

If you use the driver only to synchronize data from eDirectory to the LDAP server (on a Subscriber channel), then most LDAP servers and applications will work without any additional configuration.

However, if you require that changes be made to entries on the LDAP server synchronize back to eDirectory (on a Publisher channel), then you will need to perform at least two configuration tasks on the LDAP server before running the driver.

- Create a User object that has the necessary rights so the driver can authenticate to the LDAP server.
- Verify that the change log mechanism of the LDAP server is enabled.

  IMPORTANT: If the LDAP server doesn't have a change log mechanism, the driver cannot have a Publisher channel for that server.

Creating an LDAP User Object with Authentication Rights

The driver attempts to prevent loopback situations where an event that occurs on the Subscriber channel gets sent back to the DirXML engine on the Publisher channel.

One of the ways that it prevents this from happening is looking in the change log to see which user made the change. If the user that made the change is the same user that the driver uses to authenticate with, then the Publisher assumes that the change was made by the driver's Subscriber channel.

NOTE: If you use Critical Path InJoin Server, the change log implementation on that server is somewhat limited because it doesn't provide the DN of the object that initiated the change. Therefore, the creator/modifier DN cannot be used to determine whether the change came from eDirectory or not.

In that case, all changes found in the change log will be sent by the Publisher to the DirXML engine, and the engine itself discards unneccessary or repetitive changes.

In order to stop the Publisher channel from discarding legitimate changes, make sure the User object that the driver uses to authenticate with is not used for any other purpose.

For example, suppose you are using the Netscape Directory Server and have configured the driver to use the administrator account CN=Directory Manager. If you want to manually make a change in Netscape Directory Server and have that change synchronize, you cannot log in and make the change with CN=Directory Manager. You must use another account.

To avoid this problem:

1. Create a user account that the driver uses exclusively
2. Assign that user account rights to see the change log and to make any changes that you want the driver to be able to make

For example, you could create a user account for the driver called uid=ldriver,ou=Directory Administrators,o=provo.novell.com. You would then assign the appropriate rights to the user account by applying the following LDIF to the server using the LDAPModify tool or Novell's Import Conversion Export utility.

```
# give the new user rights to read and search the changelog

dn: cn=changelog

changetype: modify

add: aci

aci: (targetattr = "*")(version 3.0; acl "LDAP DirXML Driver"; allow
(compare,read,search) userdn = "ldap:///uid=ldriver,ou=Directory
Administrators,o=provo.novell.com"; )

-


# give the new user rights to change anything in the o=provo.novell.com
container

dn: o=provo.novell.com

changetype: modify

add: aci

aci: (targetattr = "*")(version 3.0; acl "LDAP DirXML Driver"; allow (all)
userdn = "ldap:///uid=ldriver,ou=Directory
Administrators,o=provo.novell.com"; )

-
```

Enabling the Change Log

The change log is the part of the LDAP server that enables the driver to recognize changes that require publication from the LDAP directory to eDirectory. The LDAP directories supported by this driver support the change log mechanism.

Critical Path InJoin and Oracle Internet Directory have the change log enabled by default. Unless the change log has been turned off, you do not need to perform any additional steps to enable it.

IBM SecureWay, Netscape Directory Server, and iPlanet Directory Server require you to enable the change log after installation. For information on enabling the change log, refer to the documentation supporting your LDAP directory.

TIP: The iPlanet change log requires you to enable the Retro Changelog Plug-in.

Importing the Driver

Import the LDAP driver configuration, following the instructions to import a driver in "Creating and Configuring a Driver ".

During import, provide the following information for the driver configuration.

Field	Description
Driver Name	The eDirectory object name to be assigned to this driver, or the existing driver for which you want to update the configuration.
Placement Type	With the Simple placement option, new user objects created in the LDAP directory are placed in the container in eDirectory that you specify when importing the driver configuration. The user object is named with the value of cn. With the Mirror placement option, new user objects created in the LDAP directory are placed in the eDirectory container that mirrors the object's LDAP container.
eDirectory Container	The container in eDirectory where new users should be created. If this container doesn't exist, you must create it before you start the driver. For the LDAPMirrorSample.xml configuration, this directory is the starting point for the driver's Placement policy. Subordinate containers should be named the same as the subordinate containers in the LDAP mirror container. For the Flat configuration, this container will house all user objects.
LDAP Container	The container in the LDAP directory where new users should be created. If this container doesn't exist, you must create it before you start the driver. For the Flat configuration, this directory is the starting point for the driver's Placement policy. Subordinate containers should be named the same as the subordinate containers in the eDirectory mirror container. For the LDAPSimplePlacementSample.xml configuration, this container will house all user objects.
LDAP Server	The hostname or IP address and port of the LDAP server.
Administrator DN	Enter the LDAP DN of the administrator account created for the LDAP driver.
Administrator Password	The password for the LDAP driver administrator account. You confirm the password by re-entering it in the next field. This is the required password for the default authenticated user, Directory Manager. If Directory Manager will be used exclusively by the LDAP driver, the default authenticated user works well. However, if this user is used for any other purpose, you will probably want to change the default after you get the driver running. See "Creating an LDAP User Object with Authentication Rights" on page 17.
Configure Data Flow	◆ Bi-directional means that both LDAP and eDirectory are authoritative sources of the data synchronized between them. ◆ LDAP to eDirectory means that LDAP is the authoritative source. ◆ eDirectory to LDAP means that eDirectory is the authoritative source.

Field	Description
Enable Role-Based Entitlements	Choose Yes or No. This is a design decision, so you should understand Role-Based Entitlements before choosing to use it. For information about Role-Based Entitlements, see "Using Role-Based Entitlements" in the Novell Nsure Identity Manager 2 Administration Guide.
Install Driver as Remote/ Local	Configure the driver for use with the Remote Loader service by selecting Remote, or select Local to configure the driver for local use. If Local is selected, skip the remaining prompts.
Remote Host Name and Port	Enter 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.
Driver Password	The Driver Object Password is used by the Remote Loader to authenticate itself to the DirXML server. It must be the same password that is specified as the Driver Object Password on the DirXML Remote Loader.
Remote Password	This password is used only in the Remote Loader configuration. It allows the Remote Loader to authenticate to the DirXML engine. The Remote Loader password is used to control access to the Remote Loader instance. It must be the same password that is specified as the Remote Loader password on the DirXML Remote Loader.

Starting the Driver

If you changed default data locations during configuration, ensure that the new locations exist before you start 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.

   The driver will process all the changes in the change log. To force an initial synchronization, see "Migrating and Resynchronizing Data" on page 20.

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 LDAP server. When you migrate an object, the DirXML engine applies all of the Matching, Placement, and Create policies, as well as the Subscriber filter, to the object.

  NOTE: When migrating data from eDirectory into the LDAP directory, you might need to change your LDAP server settings to allow migration of large numbers of objects. See "Trouble Migrating Users into LDAP" on page 35.

- **Migrate data into eDirectory:** Allows you to define the criteria DirXML uses to migrate objects from an LDAP server into Novell eDirectory. When you migrate an object, the DirXML engine applies all of the Matching, Placement, and Create policies, 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 are processed as Add events.

To use one of the options explained above, see "Migrating and Synchronizing Data" in the Novell Nsure Identity Manager 2 Administration Guide.

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 Novell DirXML Products"Novell Nsure Identity Manager 2 Administration Guide.

3 Upgrading

In this section

Preparing to Upgrade

Make sure you have reviewed all TIDs and Product Updates for the version of the driver you are using.

The new driver shim is intended to work with your existing driver configuration with no changes, but this assumes that your driver shim and configuration have the latest fixes.

Upgrading the Driver Shim

- To install the upgraded driver shim, run the DirXML installation program and select the DirXML Driver for LDAP. You can do this at the same time that you install the rest of DirXML, or you can do it after the DirXML Engine is installed. Instructions are in "Installation" in the *Novell Nsure Identity Manager 2 Administration Guide*.

  The new driver shim replaces the previous one.
- When you install the driver shim, your driver configuration is preserved, so no post-installation configuration is required until you want to take advantage of the new features included in the DirXML 2.0 sample configuration, such as Password Synchronization 2.0 features.
- After installing the driver shim, you must restart eDirectory and the driver. To restart the driver in Novell iManager, see "Starting a Driver" in the *Novell Nsure Identity Manager 2 Administration Guide*.
- After installing the driver shim, you must activate the driver. See "Activating the Driver" on page 21.

Upgrading the Driver Configuration

Installing the driver shim does not change your existing configuration. Your existing configuration will continue to work with the new driver shim with no changes.

However, if you want to take advantage of the new features, you must upgrade your driver configuration, either by replacing your driver configuration with the new sample configuration, or by converting your existing configuration to DirXML 2.0 format and adding policies to it.

- To replace your existing configuration, import the new sample configuration for your existing driver objects.
- To convert an existing driver configuration so you can edit it with the new DirXML plugins, see "Upgrading a Driver Configuration from DirXML 1.x to DirXML 2.0 Format" in the *Novell Nsure Identity Manager 2 Administration Guide*..
- To add Password Synchronization 2.0 functionality to an existing driver configuration, see "Upgrading Existing Driver Configurations to Support Password Synchronization 2.0 " the *Novell Nsure Identity Manager 2 Administration Guide*.

4 Customizing the LDAP Driver

The LDAP driver includes sample configurations that you can use as a starting point for your deployment. However, most DirXML® deployments will require you to make changes to these samples.

This section covers the following customization topics:

- “Configuring the Driver Parameters” on page 25
- “Configuring Data Synchronization” on page 27
- “Configuring SSL Connections” on page 30

NOTE: When you customize data synchronization, you must work within the supported standards and conventions for the operating systems and accounts being synchronized. Data containing characters that are valid in one environment, but invalid in another, will cause errors.

Configuring the Driver Parameters

Adjusting the driver’s operating parameters allows you to tune 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.

Controlling Data Flow from the LDAP Directory to eDirectory (Publisher Settings)

Use the Publisher channel settings to control the following aspects of data exchange:

- “Poll Rate in Seconds” on page 25
- “Change Log Entries to Process on Startup” on page 26
- “Maximum Batch Size for Change Log Processing” on page 26
- “Prevent Loopback” on page 26
- “Preferred Object Classes” on page 27

Poll Rate in Seconds

This is the interval at which the driver will check the LDAP server's change log. When new changes are found, they are applied to Novell eDirectory™.

The recommended polling rate is 120 seconds.

Change Log Entries to Process on Startup

This parameter specifies where in the change log the Publisher looks for change entries.

- 1-All: The Publisher will attempt to process all of the changes found in the change log. It will continue until all changes have been processed. It will process new changes according to the poll rate.
- 2-None: The Publisher will not process any of the changes from the change log when the driver starts running. It will process new changes according to the poll rate.
- 3- Previously Unprocessed: This setting is the default. If this is the first time the driver has been run, it behaves like 1-All, processing all changes in the change log.

  If the driver has been run before, this setting causes the Publisher to process only changes that are new since the last time the driver was running. Thereafter, it will process new changes according to the poll rate.

Maximum Batch Size for Change Log Processing

When the Publisher processes new entries from the LDAP change log, it will ask for them in batches of this size. If there are fewer than this number of change log entries, all of them will be processed immediately. If there are more than this number, they will be processed in consecutive batches of this size.

Prevent Loopback

This is an advanced parameter and is not present in the sample configuration because you will seldom need to change the default behavior. The default behavior for the Publisher channel is to avoid sending changes that were made by the Subscriber channel. The way the Publisher channel detects subscriber channel changes is by looking in the LDAP change log at the creatorsName or modifiersName to see whether the authenticated entry that made the change is the same entry that the driver uses to authenticate to the LDAP server. If the entry is the same, then the Publisher channel assumes that this change was made by the driver's Subscriber channel and will not synchronize the change.

If you are certain that you want to allow this type of loopback to occur (for example, if you don't have a Subscriber channel configured for this driver and you want to be able to use the same DN and password as other processes use to make changes with), then you can set the parameter by editing the driver parameter XML.

To edit the driver parameter:

1. In iManager, click DirXML Management > Overview.
2. Find the driver in its driver set.
3. Click the driver to open the Driver Overview page, then click the driver again to open the Modify Object page.
4. Scroll to the bottom of the Driver Configuration parameters and click Edit XML.
5. In the Driver Parameters XML, find the line that contains </publisher-options> and add the following line immediately above it:

   ```
   <prevent-loopback display-name="Prevent loopback">no</prevent-loopback>
   ```
6. Click OK, click Apply, then restart the driver for this parameter to function.

Preferred Object Classes

An optional driver parameter has been added to let you specify preferred object classes on the Publisher channel.

DirXML requires that objects be identified using a single object class. However, many LDAP servers and applications can list multiple object classes for a single object. By default, when the DirXML Driver for LDAP finds an object on the LDAP server or application that has been added, deleted or modified, it sends the event to the DirXML engine and identifies it using the object class that has the most levels of inheritance in the schema definition.

For example, if a user object in LDAP is identified with the object classes of inetorgperson, organizationalperson, person, and top, then by default the driver will use inetorgperson as the object class it reports to the DirXML engine because inetorgperson has the most levels of inheritance in the schema (inheriting from organizationalperson, which inherits from person, which inherits from top).

If you want to change the default behavior of the driver, you can add the optional driver Publisher parameter named preferredObjectClasses. The value of this parameter can be either one LDAP object class, or a list of LDAP object classes separated by spaces.

When this parameter is present, the DirXML Driver for LDAP examines each object being presented on the Publisher channel to see if it contains one of the object classes in the list. It looks for them in the order they appear in the preferredObjectClasses parameter. If it finds that one of the listed object classes matches one of the values of the objectclass attribute on the LDAP object, it uses that object class as the one it reports to the DirXML engine. If none of the object classes match, it resorts to its default behavior for reporting the primary object class.

To add the optional driver Publisher parameter preferredObjectClasses, do the following:

1. In iManager, navigate to the DirXML Driver Overview page for the LDAP driver.
2. Click the LDAP driver icon to access the Modify Object page for that driver.
3. If necessary, scroll to the Driver Parameters section.
4. Click the Edit XML button for that section.
5. On the Driver Parameters (XML) page that opens, check the Enable XML Editing checkbox.
6. Below the <publisher-options> open tag (but before the closing tag) insert the following XML element. Replace the example of inetorgperson with your list of preferred object classes.

   ```
   <preferredObjectClasses display-name="Preferred object
   classes">inetorgperson</preferredObjectClasses>
   ```
7. Click OK to save and close the Driver Parameters (XML) page.
8. Click OK to save and close the Modify Object page for the driver.
9. If the driver was running, restart it.

Configuring Data Synchronization

Determining Which Objects Are Synchronized

DirXML uses filters on the Publisher and Subscriber channels to control which objects are synchronized and to define the authoritative data source for these objects.

The default filters are illustrated in “Filters” on page 12. Use the following procedures to make changes to the default.

Editing the Publisher and Subscriber Filters

1. In iManager, click DirXML Management > Overview.
2. Locate the driver in its driver set.
3. Click the driver to open the Driver Overview Page.
4. Click the Publisher or Subscriber Filter icon and make the appropriate changes.

   The Publisher filter must include the eDirectory mandatory attributes. The Subscriber filter must include the LDAP server required attributes.

   For every object and attribute selected in the filter, there must be a corresponding entry in the Mapping policy unless the class or attribute names are the same in both directories. Verify that a corresponding attribute actually exists in the target directory before mapping it.

Defining Schema Mapping

Different LDAP servers have different schemas. When the driver is first started, it queries the server for the specific schema.

You must be familiar with the characteristics of eDirectory attributes and the LDAP server attributes. The driver handles all LDAP attribute types (cis, ces, tel, dn, int, bin). It also handles the eDirectory Facsimile Telephone Number.

When mapping attributes, follow these guidelines:

- Verify that every class and attribute specified in the Subscriber and Publisher policies is mapped in the Mapping policy unless the class or attribute names are the same in both directories.
- Verify that an LDAP server attribute actually exists before mapping an eDirectory attribute to it. For example, the Full Name attribute is defined for a User object on eDirectory but fullname does not exist in an inetOrgPerson object on Netscape.
- Always map attributes to attributes of the same type, for example, strings attributes to strings attributes, octet attributes to binary attributes, telenumber attributes to telenumber attributes.
- Map multi-valued attributes to multi-valued attributes.

The driver does not provide data conversion between different attribute types or conversions from multi-valued to single-valued attributes. The driver also does not understand structured attributes except for Facsimile Telephone Number and Postal Address.

DirXML is flexible on the syntax that it accepts coming in from the Publisher, notably:

1. DirXML will accept any non-structured/non-octet syntax for any other non-structured/non-octet syntax as long as the actual data can be coerced to the appropriate type (that is, if eDirectory is looking for a numeric value, the actual data should be a number).
2. When DirXML is expecting octet data and gets another non-octet/non-structured type, it will coerce the data to octet by serializing the string value to UTF-8.
3. When DirXML is passed octet data and another non-structured type is expected, it will coerce the data to a string by decoding the Base64 data and then try to interpret the result as a UTF-8 encoded string (or the platform's default character encoding if it is not a valid UTF-8 string) and then apply the same rules as 1.
4. For faxNumber, if a non-structured type is passed in, 1) and 3) are applied to the data to get the phone number portion of the fax number, and the other fields are defaulted.

5. For state, False, No, F, N (in either upper or lowercase), 0 and "" (empty string) are interpreted as False, and any other value is interpreted as True.
6. For emailAddress, if a non-structured type is passed in, 1) and 3) are applied to the data to get the address, and the type is defaulted to 3 (SMTP).

To configure the Schema Mapping policy:

1. In iManager, click DirXML Management > Overview.
2. Locate the driver in its driver set.
3. Click the driver to open the Driver Overview page.
4. Click the schema mapping icon on the Publisher or Subscriber channel.
5. Edit the policy as appropriate for your setup.

Defining Object Placement

We recommend following the Netscape naming rules for objects in Netscape Directory Server. A brief explanation of naming rules is included here for your convenience.

The directory contains entries that represent people. These person entries must have names. In other words, you must decide what the relative distinguished name (RDN) will be for each person entry. The DN must be a unique, easily recognizable, permanent value. We recommend that you use the uid attribute to specify a unique value associated with the person. An example DN for a person entry is:

uid=jsmith,o=novell

The directory will also contain entries that represent many things other than people (for example, groups, devices, servers, network information, or other data). We recommend that you use the cn attribute in the RDN. Therefore, if you are naming a group entry, name it as follows:

cn=administrators,ou=groups,o=novell

The directory will also contain branch points or containers. You need to decide what attributes you will use to identify the branch points. Attribute names have a meaning, so use the attribute name with the type of entry it is representing. The Netscape recommended attributes are defined as follows:

Attribute Name	Definition
c	Country name
o	Organization name
ou	Organizational Unit
st	State
l	Locality
dc	Domain Component

A Subscriber Placement Policy specifies the naming attribute for a classname. The example below is for the User classname. The <placement> statement specifies that uid is used as the naming attribute.

```
<placement-rule>
   <match-class class-name="User"/>
   <match-path prefix="\Novell-Tree\Novell\Users"/>
   <placement>uid=<copy-name/>,ou=People,o=Netscape</
 placement>
</placement-rule>
```

The Subscriber Placement policy below specifies that ou is used as the naming attribute for class-name Organizational Unit.

```
<placement-rule>
   <match-class class-name="Organizational Unit"/>
   <match-path prefix="\Novell-Tree\Novell\Users"/>
   <placement>ou=<copy-name/>,ou=People,o=Netscape</placement>
</placement-rule>
```

Configuring Placement Policies

1. In iManager, click DirXML Management > Overview.
2. Locate the driver in its driver set.
3. Click the driver to open the Driver Overview Page.
4. Click the Publisher or Subscriber Placement policy icon and make the appropriate changes.

Working with eDirectory Groups

Group attributes are different in eDirectory and Netscape Directory Server, so some special processing is required by the driver. On the Publisher channel, special processing takes place when the driver sees the attribute *uniquemember* in the classname *groupofuniquenames*.

The driver also sets the attribute Equivalent To Me in the eDirectory Group. The attribute Equivalent To Me must be included in the Publisher filter. The attribute Equivalent To Me need not be in the Schema Mapping policy because the eDirectory attribute name is used. There is no equivalent attribute name in Netscape Directory Server. No special processing is required on the Subscriber channel.

Configuring SSL Connections

The driver uses the LDAP protocol to communicate with the LDAP server. Most LDAP servers allow non-encrypted connections (also called clear text connections). Additionally, when configured correctly, some LDAP servers allow SSL-encrypted connections. SSL connections encrypt all traffic on the TCP/IP socket using a public/private key pair. The actual LDAP protocol doesn't change, but the communication channel performs the encryption.

The procedure for enabling SSL connections differs slightly from one LDAP server to another. This document covers the process for enabling SSL connections when using Netscape Directory Server 4.12.

- “Step 6: Importing the Trusted Root Certificate” on page 33
- “Step 7: Adjusting Driver Settings” on page 34

If you are using another LDAP server, the procedure will be similar.

Step 1: Generating a Server Certificate

You first need to install a server certificate. The LDAP server itself can generate a certificate, but the certificate must then be signed by a CA that is trusted by the server. One way to get the certificate signed is to use the CA that comes with eDirectory.

To generate a certificate request:

1. In the navigation tree in Netscape Console, select the server the driver will communicate with.
2. Click Open Server.
3. Click Tasks > Certificate Setup Wizard.
4. Provide information to request a certificate.

   Depending on the certificates or tokens that might already be installed on the host system, you might see some or all of the following fields:

   Select a Token (Cryptographic Device): Select Internal (Software).

   Is the Server Certificate Already Requested and Ready to Install? Select No.

   If a trust database doesn't already exist for this host, one is generated for you.

   A trust database is a key pair and certificate database installed on the local host. When you use an internal token, the trust database is the database into which you install the key and certificate.
5. Enter and confirm the password.

   The password must contain at least eight characters, and at least one of them must be numeric. This password helps secure access to the new key database you're creating.
6. Continue providing information as prompted > click Next.
7. After a trust database is created, click Next.
8. Enter the requested information > click Next.
9. Enter the password for the token you selected earlier > click Next.

   The Certificate Setup Wizard generates a certificate request for your server. When you see the page, you can send the certificate request to the certification authority.

Step 2: Sending the Certificate Request

1. Copy the server certificate request into Notepad or another text editor.
2. Save the file as CSR.TXT.

   Your certificate request e-mail should look like the following:

```
-----BEGIN NEW CERTIFICATE REQUEST-----

                .
```

.

.

```
-----END NEW CERTIFICATE REQUEST----
```

3. In iManager, select Novell Certificate Server > Issue Certificate.
4. In the Filename field, browse to CSR.TXT > click Next.
5. Select Organizational Certificate Authority.
6. Specify SSL as the key type > click Next.
7. Specify the certificate parameters > click Next > click Finish.
8. Save the certificate in Base64 format as CERT.B64 to a local disk or diskette.

Step 3: Installing the Certificate

1. In the navigation tree in Netscape Console, select the server the driver will be connecting to.
2. Click Open.
3. Click Tasks > Certificate Setup Wizard.
4. Start the wizard and indicate you are ready to install the certificate.
5. When prompted, provide the following information:

   Select a Token (Cryptographic Device): Select Internal (Software).

   Is the Server Certificate Already Requested and Ready to Install? Select Yes.
6. Click Next.
7. In the Install Certificate for field, select This Server.
8. In the Password field, enter the password you used to set up the trust database > click Next.
9. In the Certificate Is Located in This File field, enter the absolute path to the certificate, for example, A: \CERT.B64.
10. After the certificate is generated, click Add.
11. After the certificate is successfully installed, click Done.

Step 4: Activating SSL in Netscape Directory Server 4.12

After you install the certificate, complete the following to activate SSL:

1. In the navigation tree in Netscape Console, select the server you want to use SSL encryption with.
2. Click Open > Configuration > Encryption.
3. Enter the following information:

   Enable SSL: Select this option.

   Cipher Family: Check RSA.

   Token to Use: Select Internal (Software).

   Certificate to Use: Select Server-Cert.

Client Authentication: Select Allow Client Authentication because the driver does not support client authentication.

4. Click Save.
5. Click Tasks > restart the server for the changes to take effect.

Step 5: Exporting the Trusted Root from the eDirectory Tree

1. In iManager, select eDirectory Administration > Modify Object.
2. Browse to the Certificate Authority (CA) object > click OK.
3. Click the Certificates tab.
4. Click Export.
5. Click No at the prompt that says "Do you want to export the private key with the certificate?"
6. Click Next.
7. In the Filename field, type in a filename (for example, PublicKeyCert) > select Base64 as the format.
8. Click Export.

Step 6: Importing the Trusted Root Certificate

You need to import the trusted root certificate into the LDAP server's trust database and the client's certificate store.

Importing into the LDAP Server's Trust Database

You need to import the trusted root certificate into the LDAP server's trust database. Because the server certificate was signed by eDirectory's CA, the trust database needs to be configured to trust the eDirectory CA.

To import the trusted root certificate into the LDAP server's trust database:

1. In the Netscape Console, click Tasks > Certificate Setup Wizard > Next.
2. In Select a Token, accept the default for Internal (Software).
3. In Is the Server Certificate Already Requested and Ready to Install, select Yes.
4. Click Next > Next.
5. In Install Certificate For, select Trusted Certificate Authority.
6. Click Next.
7. Select The Certificate Is Located in This File > enter the full path to the .b64 file containing the trusted root certificate.
8. Click Next.
9. Verify the information on the screen > click Add.
10. Click Done.

Importing into the Client's Certificate Store

You need to import the trusted root certificate into a certificate store (also called a key store) that the driver can use.

To import the trusted root certificate into the client's certificate store:

1. Use the KeyTool class found in rt.jar.

   For example, if your public key certificate is saved as PublicKeyCert.b64 on a diskette and you want to import it into a new certificate store file named .keystore in the current directory, type the following at the command line:

   ```
   java sun.security.tools.KeyTool -import -alias TrustedRoot -file
   a:\PublicKeyCert.b64

    -keystore .keystore -storepass keystorepass
   ```

2. When you are asked to trust this certificate, enter Yes > click Enter.
3. Copy the .keystore file to any directory on the same file system that has the eDirectory files.
4. In iManager, click DirXML Management > Overview. Search for drivers.
5. Click the LDAP Driver object, then click it again in the next page that appears.
6. In the Keystore Path parameter, enter the complete path to the .keystore file.

Step 7: Adjusting Driver Settings

The following table lists the driver's settings and its default values in the sample configurations.

Parameter	Sample Configuration Value
Use SSL for LDAP Connections	no
SSL Port	636
Keystore Path (for SSL Certs)	[blank]

Use SSL for LDAP Connections

The value for this parameter should be either Yes or No. It indicates whether or not SSL connections should be used when communicating with the LDAP server. To use SSL, you must also correctly configure the LDAP server.

For more information, refer to "Configuring SSL Connections" on page 30.

SSL Port

This parameter is ignored unless Use SSL for LDAP Connections is set to Yes. It indicates which port the LDAP server uses for secure connections.

Keystore Path (for SSL Certs)

When Use SSL for LDAP Connections is set to Yes, this parameter value should be the complete path to the keystore file that contains the trusted root certificate of the Certificate Authority (CA) that signed the server certificate.

For more information about creating the keystore file, refer to "Importing into the Client's Certificate Store" on page 34.

5 Troubleshooting

This section contains troubleshooting tips.

Trouble Migrating Users into LDAP

Some LDAP servers have settings that limit the number of entries that can be returned by an LDAP query. For example, iPlanet Directory Server 5.1 has a default limit of 2000 objects.

When migrating user data from eDirectory into LDAP, the driver makes an LDAP query to the server, and returns the objects that match the criteria (such as objectclass=User).

A limit on the number of entries that can be returned on an LDAP query can cause a migration to stop before it is complete, even though the DirXML driver continues to run normally otherwise.

To fix this, change the limit.

For example, in iPlanet do the following:

1. Go to the Configuration tab > Database settings.
2. Raise the look-through limit on the LDBM plug-in tab from default of 5000 to an appropriate number. (This is the number of records the query is allowed to look at while fulfilling the query.)
3. Go to the Configuration tab > directory server settings > performance tab and raise the Size limit according to the number of user accounts you need to migrate. (This is the actual number of records the query is allowed to return.)

   After these setting have been adjusted, the migration should complete correctly.