NetIQ®

# NetIQ® Identity Manager

## Driver for LDAP Implementation Guide

**October 2019**

## Legal Notice

For information about NetIQ trademarks, see https://www.netiq.com/company/legal/.

# Contents

# About this Book and the Library

The *Identity Manager Driver for LDAP Implementation Guide* explains how to install, configure, and manage the Identity Manager Driver for LDAP.

## Intended Audience

This book provides information for individuals who are using the Identity Manager Driver for LDAP.

## Other Information in the Library

For more information about the library for Identity Manager, see the following websites:

- Identity Manager documentation website (https://www.netiq.com/documentation/identity-manager-48/)
- Identity Manager drivers documentation website (https://www.netiq.com/documentation/identity-manager-48-drivers/)

# About NetIQ Corporation

We are a global, enterprise software company, with a focus on the three persistent challenges in your environment: Change, complexity and risk—and how we can help you control them.

## Our Viewpoint

**Adapting to change and managing complexity and risk are nothing new**

In fact, of all the challenges you face, these are perhaps the most prominent variables that deny you the control you need to securely measure, monitor, and manage your physical, virtual, and cloud computing environments.

**Enabling critical business services, better and faster**

We believe that providing as much control as possible to IT organizations is the only way to enable timelier and cost effective delivery of services. Persistent pressures like change and complexity will only continue to increase as organizations continue to change and the technologies needed to manage them become inherently more complex.

## Our Philosophy

**Selling intelligent solutions, not just software**

In order to provide reliable control, we first make sure we understand the real-world scenarios in which IT organizations like yours operate—day in and day out. That's the only way we can develop practical, intelligent IT solutions that successfully yield proven, measurable results. And that's so much more rewarding than simply selling software.

**Driving your success is our passion**

We place your success at the heart of how we do business. From product inception to deployment, we understand that you need IT solutions that work well and integrate seamlessly with your existing investments; you need ongoing support and training post-deployment; and you need someone that is truly easy to work with—for a change. Ultimately, when you succeed, we all succeed.

## Our Solutions

- Identity & Access Governance
- Access Management
- Security Management
- Systems & Application Management
- Workload Management
- Service Management

# Contacting Sales Support

For questions about products, pricing, and capabilities, contact your local partner. If you cannot contact your partner, contact our Sales Support team.

| | |
|---|---|
| **Worldwide:** | www.netiq.com/about_netiq/officelocations.asp |
| **United States and Canada:** | 1-888-323-6768 |
| **Email:** | info@netiq.com |
| **Web Site:** | www.netiq.com |

# Contacting Technical Support

For specific product issues, contact our Technical Support team.

| | |
|---|---|
| **Worldwide:** | www.netiq.com/support/contactinfo.asp |
| **North and South America:** | 1-713-418-5555 |
| **Europe, Middle East, and Africa:** | +353 (0) 91-782 677 |
| **Email:** | support@netiq.com |
| **Web Site:** | www.netiq.com/support |

# Contacting Documentation Support

Our goal is to provide documentation that meets your needs. The documentation for this product is available on the NetIQ Web site in HTML and PDF formats on a page that does not require you to log in. If you have suggestions for documentation improvements, click **Add Comment** at the bottom of any page in the HTML version of the documentation posted at www.netiq.com/documentation. You can also email Documentation-Feedback@netiq.com. We value your input and look forward to hearing from you.

# Contacting the Online User Community

NetIQ Communities, the NetIQ online community, is a collaborative network connecting you to your peers and NetIQ experts. By providing more immediate information, useful links to helpful resources, and access to NetIQ experts, NetIQ Communities helps ensure you are mastering the knowledge you need to realize the full potential of IT investments upon which you rely. For more information, visit community.netiq.com.

# 1 Understanding the LDAP Driver

The Identity Manager Driver for LDAP (LDAP driver) synchronizes data between the Identity Vault and LDAP-compliant directories. The driver supports the Subscriber and Publisher channels, uses filters to control objects and attributes, and uses policies to control data.

- "Driver Concepts" on page 9
- "Support for Standard Driver Features" on page 12

## Driver Concepts

- "Synchronizing Data" on page 9
- "Publication Methods" on page 9
- "How the LDAP Driver Works" on page 10

### Synchronizing Data

The Identity Manager Driver for LDAP synchronizes data between an Identity Vault and LDAP-compliant directories. The driver can run anywhere that a Identity Manager server or Identity Manager Remote Loader is running. See "Local and Remote Platforms" on page 12.

The driver uses the Lightweight Directory Access Protocol to bidirectionally synchronize changes between an Identity Vault and the connected LDAP-compliant directory.

Because of this flexible model for communicating, the driver can synchronize with LDAP-compliant directories running on platforms that are not supported by an Identity Vault, such as HP-UX, OS/400, and OS/390.

### Publication Methods

The driver can use either of two publication methods to recognize data changes and communicate them to an Identity Vault through Identity Manager:

- "Changelog Method" on page 9
- "LDAP-Search Method" on page 10

### Changelog Method

This method is preferred when a change log is available. Change logs are found on the following:

- Critical Path InJoin Directory
- IBM SecureWay Directory
- IBM Tivoli Directory
- iPlanet Directory Server

- Isode M-Vault
- Netscape Directory Server
- Oracle Internet Directory
- Sun Java System Directory

### LDAP-Search Method

Some servers do not use the changelog mechanism. The LDAP-search method enables the LDAP driver to publish data about the LDAP server to an Identity Vault by searching for changes in predefined contexts in the LDAP directory.

The LDAP-search method synchronizes changes that occur from one poll to the next.

## How the LDAP Driver Works

Channels, filters, and policies control data flow.

- "Publisher and Subscriber Channels" on page 10
- "Filters" on page 10
- "Policies" on page 11

### Publisher and Subscriber Channels

The LDAP driver supports Publisher and Subscriber channels:

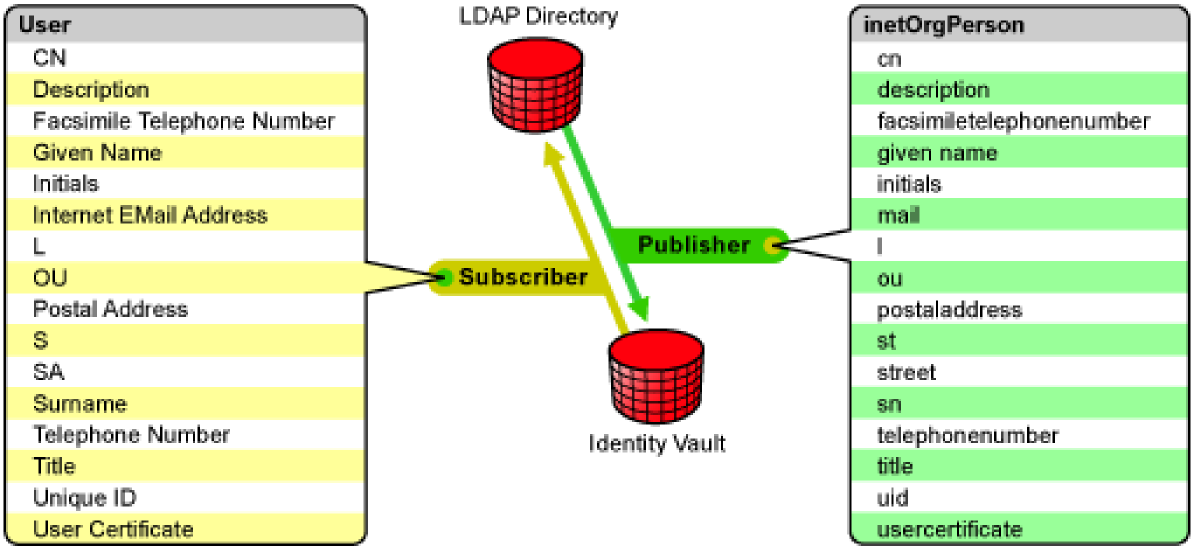
- The Publisher channel reads information from the LDAP directory change log or an LDAP search and submits that information to an Identity Vault via Identity Manager engine.

  By default, the Publisher channel checks the log every 60 seconds, processing up to 1000 entries at a time, starting with the first unprocessed entry.
- The Subscriber channel watches for additions and modifications to Identity Vault objects and issues LDAP commands that make changes to the LDAP directory.

### Filters

Identity Manager 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 1-1 LDAP Driver Filters*



## Policies

Policies are used to control data synchronization between the driver and an Identity Vault.

The following table provides information on default policies. These policies and the individual rules they contain can be customized as explained in Chapter 6, "Synchronizing Data," on page 35.

*Table 1-1 Default Policies*

| Policy | Description |
|---|---|
| Schema Mapping | Maps the Identity Vault User object and selected properties to an LDAP inetOrgPerson.<br>Maps the Identity Vault Organizational Unit to an LDAP organizationalUnit.<br>By default, more than a dozen standard properties are mapped. |
| Publisher Create | Specifies that in order for a User to be created in an Identity Vault, the cn, sn, and mail attributes must be defined. In order for an Organizational Unit to be created, the OU attribute must be defined. |
| Publisher Placement | Specifies that new User objects created in the LDAP directory are placed under a specified Identity Vault container in the same structure that mirrors the object's LDAP container structure. In other words, an Identity Vault container (defined during creation of the driver) becomes the root container in which the LDAP objects are mirrored exactly as they exist in the LDAP directory. |
| Matching | Specifies that a user object in an Identity Vault is the same object as an inetOrgPerson in the LDAP directory when the e-mail attributes match. |

| Policy | Description |
|---|---|
| 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 Organizational Unit to be created, the OU attribute must be defined. |
| Subscriber Placement | Specifies that new User objects created in the Identity Vault are placed under a specified LDAP container in the same structure that mirrors the object's Identity Vault container structure. In other words, an LDAP container (defined during creation of the driver) becomes the root container in which the Identity Vault objects are mirrored exactly as they exist in the Identity Vault. |

# Support for Standard Driver Features

The LDAP driver supports these standard driver features:

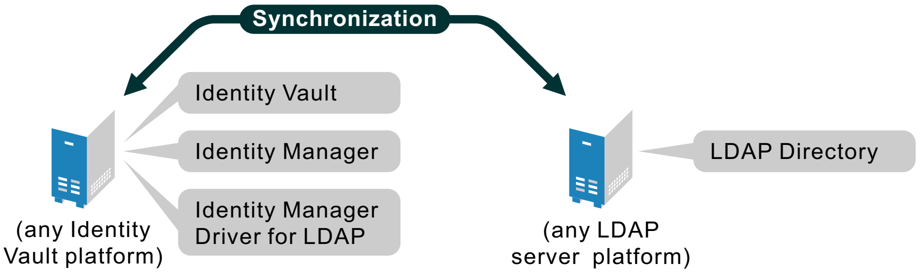
- "Local and Remote Platforms" on page 12
- "Password Synchronization" on page 13

## Local and Remote Platforms

You can install the LDAP driver locally or remotely.

An installation on the same computer where an Identity Vault and the Identity Manager engine are installed is referred to as a local configuration. The following figure illustrates a local configuration:

***Figure 1-2*** *A Local Configuration*


Synchronization
Identity Vault
Identity Manager
Identity Manager Driver for LDAP
(any Identity Vault platform)
LDAP Directory
(any LDAP server platform)

If platform or policy constraints make a local configuration difficult, you can install the LDAP driver on the server hosting the target LDAP server. This installation is referred to as a remote configuration and requires the use of the Remote Loader service.

Although a remote configuration is possible, it provides little additional flexibility because of the following:

- The driver can run on any Identity Vault platform.
- The driver communicates with the LDAP server on any platform across the wire via the LDAP protocol.

See “Implementation Checklist” in the *NetIQ Identity Manager Setup Guide for Linux* or “Determine Hardware Requirements” in the *NetIQ Identity Manager Setup Guide for Windows* for information about the supported platforms for the Identity Manager server and Remote Loader.

## Password Synchronization

The LDAP driver supports password synchronization on the Subscriber channel, meaning that you can send passwords from the Identity Vault to any connected LDAP directory.

Password synchronization on the Publisher channel (LDAP directory to Identity Vault) is supported only with Sun Java System Directory Server Enterprise Edition version 6.3.x. For more information about installing the Identity Manager plug-ins for synchronizing passwords on the Subscriber channel, see “Installing the Identity Manager Plug-Ins for Password Synchronization with Sun Java System Directory” on page 15.

# 2 Installing the Driver Files

By default, the LDAP driver files are installed on the Identity Manager server at the same time as the Identity Manager engine. The installation program extends the Identity Vault's schema and installs the driver shim and the driver configuration file. It does not create the driver in the Identity Vault (see Chapter 4, "Creating a New Driver Object," on page 21) or upgrade an existing driver's configuration (see Chapter 5, "Upgrading an Existing Driver," on page 29).

The following sections explain what to do if the LDAP driver files are not on the server you want and how to install them:

- "Installing the Driver Files" on page 15
- "Installing the Identity Manager Plug-Ins for Password Synchronization with Sun Java System Directory" on page 15

## Installing the Driver Files

If you performed a custom installation and did not install the LDAP driver on the Identity Manager server, you have two options:

- Install the files on the Identity Manager server, using the instructions in Considerations for Installing Identity Manager Components  in the *NetIQ Identity Manager Setup Guide for Linux* or Planning to Install the Engine, Drivers, and Plug-ins in the *NetIQ Identity Manager Setup Guide for Windows*.
- Install the Remote Loader (required to run the driver on a non-Identity Manager server) and the driver files on a non-Identity Manager server where you want to run the driver. See Configuring the Remote Loader and Drivers in the *NetIQ Identity Manager Setup Guide for Linux* or Configuring the Remote Loader and Drivers in the *NetIQ Identity Manager Setup Guide for Windows*.

## Installing the Identity Manager Plug-Ins for Password Synchronization with Sun Java System Directory

The LDAP driver supports password synchronization on the Subscriber channel, meaning that you can send passwords from the Identity Vault to any connected LDAP directory.

Password synchronization on the Publisher channel (LDAP directory to Identity Vault) is supported only with Sun Java System Directory Server Enterprise Edition version 6.3.x. It requires you to install an Identity Manager plug-in to the Sun Java System Directory.

The following sections provide information to help you use the plug-in:

- "How the Plug-In Works" on page 16
- "Where to Find the Plug-In" on page 16
- "Installing the Plug-In" on page 16

## How the Plug-In Works

The plug-in is a post-operation plug-in. Sun Java System Directory notifies the plug-in whenever a password is set or changed. The plug-in then encrypts the password by using the Advanced Encryption Standard (AES) and stores the encrypted password on the novellDistPassword attribute. The LDAP driver can then synchronize the encrypted password to NetIQ Identity Manager. The LDAP driver decrypts the password and uses it to set the Identity Manager distribution password.

**IMPORTANT:** Only passwords that are set or modified after the plug-in is installed can be synchronized.

## Where to Find the Plug-In

The plug-in is located in the Identity Manager iso for the Windows and Linux platforms.

***Table 2-1*** *Plug-In Location*

| iso | Location | Filename |
|---|---|---|
| Identity_Manager_4.7_Windows.iso | `products\IDM\windows\setup\utilities\sun_password_plugins\win32\` | `novl-idm-pswd.dll` |
| Identity_Manager_4.7_Linux.iso | `/mnt/IDM/utilities/sun_password_plugins/linux/` | `novl-idm-pswd.so` |

## Installing the Plug-In

1. Locate the correct plug-in file. See "Where to Find the Plug-In" on page 16 for information.
2. Copy the binary plug-in file to the `lib` directory in your Sun Java System Directory installation location.

   For example, on Windows the default installation location for Sun Java System Directory is `C:\Program Files\Sun\MPS` and inside that directory is a `lib` directory. Place `novl-idm-pswd.dll` in the `lib` directory.

   On other platforms, the default installation location is often `/var/Sun/mps`. You need to locate the Sun Java System Directory installation location on your system, and put the plug-in file inside the `lib` directory.

   On Solaris SPARC computers, the Sun Java System Directory installation includes two versions of most libraries: a 32-bit version and a 64-bit version. By default, the 32-bit version is found at `/var/Sun/mps/lib.` The 64-bit version is found at `/var/Sun/mps/lib/64`.

   Both a 32-bit and a 64-bit version of the plug-in are provided. Copy both versions to their respective locations on your Solaris installation. At runtime, the Sun Java System Directory determines which version is the appropriate version to load.
3. Locate and edit the `novl-idm-pswd.ldif` or `novl-idm-pswd-win32.ldif` file. The file is located in the `sun_password_plugins` directory on your DVD image.

The .ldif file contains plug-in configuration information that you apply to the directory. It also contains two schema definitions:

- One definition is for the novellDistPassword attribute that stores the encrypted password.
- The other definition is for the novellDistPasswordUser auxiliary class that is applied to your users to allow the use of the novellDistPassword attribute.

As a convenience, the .ldif file also contains an instruction to turn on the Retro Changelog Plugin, which most customers want turned on to enable Publisher channel operations with the Identity Manager LDAP driver. If you know that the changelog is already enabled, or if you don't want to enable the changelog, you can remove the Retro Changelog Plugin section from the .ldif file.

Most users need to edit only two items in the .ldif file:

- The nsslapd-pluginPath attribute
- The nsslapd-pluginarg0 attribute

Ensure that the value of nsslapd-pluginPath is the path where you installed the plug-in. For example, if you installed the plug-in in the /var/Sun/mps/lib directory, the value should be /var/Sun/mps/lib/novl-idm-pswd.so. Set the value of nsslapd-pluginarg0 to a password that will be used to generate an AES key used to encrypt user passwords. When you create the LDAP driver, you will configure the driver with this same encryption password.

Solaris users should set the value of nsslapd-pluginPath to the path of the 32-bit version of the plug-in, even if the operating system is 64-bit. (See Step 2.) At runtime, the directory determines whether to load the 32-bit or the 64-bit version of the plug-in.

**4** Apply the novl-idm-pswd.ldif or novl-idm-pswd-win32.ldif file to the Sun directory.

To complete this step, you need to know the configuration administrator's DN and password. Typically, the DN will be "uid=admin,ou=Administrators, ou=TopologyManagement,o=NetscapeRoot". However, the password will vary. You also need to know the LDAP port used by your Sun directory.

The ldapmodify command line utility that was installed with your Sun Java System Directories can be used to apply the .ldif file. Use a command similar to the following:

```
ldapmodify -h localhost -p 389 -D
"uid=admin,ou=Administrators,ou=TopologyManagement,o=NetscapeRoot" -w
password -f novl-idm-pswd.ldif
```

**5** Restart Sun Java System Directory so that your changes take affect and the plug-in starts.

For troubleshooting, note any errors that might appear on the console.

# 3 Preparing the LDAP Server

You need to do the following to prepare the LDAP server to which you are connecting:

- Create a user account through which the LDAP driver can authenticate to the LDAP directory.
- If you plan to use changelog as the publication method (see "Publication Methods" on page 9), you need to verify that the change log mechanism of the LDAP server is enabled. This is required only for synchronization of data from the LDAP server to the Identity Vault (Publisher channel). It is not required if you plan only to push data from the Identity Vault to the LDAP directory.

The following sections provide instructions

- "Creating an LDAP User Object with Authentication Rights" on page 19
- "Enabling the Change Log" on page 20

## Creating an LDAP User Object with Authentication Rights

When you use the changelog publication method, the driver attempts to prevent loopback situations where an event that occurs on the Subscriber channel is sent back to the Identity Manager engine on the Publisher channel. However, the LDAP-search publication method relies on the Identity Manager engine to prevent loopback.

With the changelog method, one way that the driver prevents loopback from happening is to look 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, 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 can't be used to determine whether the change came from an Identity Vault or not.

In that case, all changes found in the change log are sent by the Publisher to the Identity Manager engine, and the Optimize/Modify discards unnecessary or repetitive changes.

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 the Netscape Directory Server and have that change synchronize, you can't 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, at the VMP company, you create a user account for the driver called uid=ldriver,ou=Directory Administrators,o=lansing.vmp.com. You then assign the appropriate rights to the user account by applying the following LDIF to the server by using the LDAPModify tool or the Novell 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=lansing.vmp.com"; )
   -
   # give the new user rights to change anything in the o=lansing.vmp.com
   container
   dn: o=lansing.vmp.com
   changetype: modify
   add: aci
   aci: (targetattr = "*")(version 3.0; acl "LDAP DirXML Driver"; allow
   (all)
   userdn = "ldap:///uid=ldriver,ou=Directory
   Administrators,o=lansing.vmp.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 an Identity Vault. The LDAP directories supported by this driver support the changelog mechanism.

Critical Path InJoin and Oracle Internet Directory have the change log enabled by default. Unless the change log has been turned off, you don’t 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. See Step 3 on page 16 for instructions.

# 4 Creating a New Driver Object

After the LDAP driver files are installed on the server where you want to run the driver (see Chapter 2, “Installing the Driver Files,” on page 15), you can create the driver in the Identity Vault.

- “Creating the Driver Object in Designer” on page 21
- “Activating the Driver” on page 27
- “Adding Packages to an Existing Driver” on page 28

## Creating the Driver Object in Designer

You create the LDAP driver object by installing the driver packages and then modifying the configuration to suit your environment. After you create and configure the driver, you need to deploy it to the Identity Vault and start it.

- “Importing the Current Driver Packages” on page 21
- “Installing the Driver Packages” on page 22
- “Configuring the Driver Object” on page 25
- “Deploying the Driver Object” on page 26
- “Starting the Driver” on page 27

### Importing the Current Driver Packages

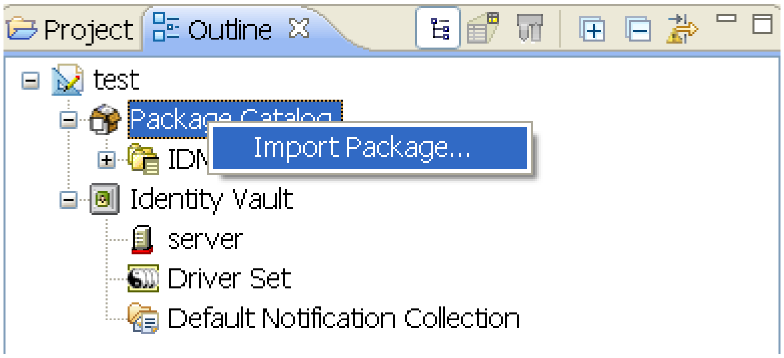
The driver packages contain the items required to create a driver, such as policies, entitlements, filters, and Schema Mapping policies. These packages are only available in Designer and can be updated after they are initially installed. You must have the most current version of the packages in the Package Catalog before you can create a new driver object.

To verify that you have the most recent version of the driver packages in the Package Catalog:

1. Open Designer.
2. In the toolbar, click **Help** > **Check for Package Updates**.
3. Click **OK** to update the packages

   or

   Click **OK** if the packages are up-to-date.
4. In the Outline view, right-click the Package Catalog.
5. Click **Import Package**.



6 Select any LDAP driver packages

or

Click **Select All** to import all of the packages displayed.

By default, only the base packages are displayed. Deselect **Show Base Packages Only** to display all packages.

7 Click **OK** to import the selected packages, then click **OK** in the successfully imported packages message.

8 After the current packages are imported, continue with “Installing the Driver Packages” on page 22.

## Installing the Driver Packages

After you have imported the current driver packages into the Package Catalog, you can install the driver packages to create a new driver.

1 In Designer, open your project.

2 In the Modeler, right-click the driver set where you want to create the driver, then click **New** > **Driver**.

3 Select **LDAP Base**, then click **Next**.

4 Select the optional features to install for the LDAP driver. All options are selected by default. The options are:

**Default Configuration:** This package contains the default configuration information for the LDAP driver. Always leave this option selected.

**Password Synchronization:** This packages contains the policies that enable the LDAP driver to synchronize passwords. If you want to synchronize passwords, verify that this option is selected. For more information, see the *NetIQ Identity Manager Password Management Guide*.

**Data Collection:** These packages contain the policies that enable the driver to collect data for reports. If you are using Identity Reporting, verify that this option is selected. For more information, see the *Administrator Guide to NetIQ Identity Reporting*.

**Account Tracking:** These packages contain the policies that enables account tracking information for reports. If you are using Identity Reporting, verify that this option is selected. For more information, see the *Administrator Guide to NetIQ Identity Reporting*.

5 Click **Next**.

6 (Conditional) If there are package dependencies for the packages you selected to install, you must install them to install the selected package. Click **OK** to install the package dependencies that are listed.

7 (Conditional) If more than one type of package dependency must be installed, you are presented with these packages separately. Continue to click **OK** to install any additional package dependencies.

8 (Conditional) The Common Settings page is displayed only if the Common Settings package is installed as a dependency. On the Install Common Settings page, fill in the following fields:

**User Container:** Select the Identity Vault container where the LDAP accounts will be added if they don't already exist in the vault. This value becomes the default for all drivers in the driver set.

If you want a unique location for this driver, set the value for all drivers on this page. After the driver is created, change the value on the driver's Global Configuration Values page.

**Group Container:** Select the Identity Vault container where the LDAP accounts will be added if they don't already exist in the vault. This value becomes the default for all drivers in the driver set.

If you want a unique location for this driver, set the value for all drivers on this page. After the driver is created, change the value on the driver's Global Configuration Values page.

9 Click **Next**.

10 On the Driver Information page, specify a name for the driver, then click **Next**.

11 On the Application Authentication page, fill in the following fields:

**Authentication ID:** Specify the authentication ID for the driver in LDAP format.

**Connection Information:** Specify the connection information for the driver to connect to the LDAP directory.

**Password:** Specify the password for the authentication ID.

12 Click **Next**.

13 Fill in the following fields for Remote Loader information:

**Connect To Remote Loader:** Select **Yes** or **No** to determine if the driver will use the Remote Loader. For more information, see Configuring the Remote Loader and Drivers in the *NetIQ Identity Manager Setup Guide for Linux* or Configuring the Remote Loader and Drivers in the *NetIQ Identity Manager Setup Guide for Windows*.

If you select **No**, skip to Step 14. If you select **Yes**, use the following information to complete the configuration of the Remote Loader.

**Host Name:** Specify the IP address or DNS name of the server where the Remote Loader is installed and running.

**Port:** Specify the port number for this driver. Each driver connects to the Remote Loader on a separate port. The default value is 8090.

**Remote Loader Password:** Specify a password to control access to the Remote Loader. It must be the same password that is specified as the Remote Loader password on the Remote Loader.

**Driver Password:** Specify a password for the driver to authenticate to the Identity Manager server. It must be the same password that is specified as the Driver Object Password on the Remote Loader.

14 Click **Next**.

15 On the Account Status Support page, fill in the following fields:

**LDAP Server Type:** Select the LDAP server type this driver connects to. Based on the type, the appropriate attribute is modified in the LDAP directory to disable the account. The options are:

- eDirectory
- openLDAP
- iPlanet | SunOne | OID

**nsManageDisabledRole DN:** This option is displayed only if you select **iPlanet | SunOne |OID.** This is the DN of the role used to disable users in the connected LDAP system. If a user is enabled in the Identity Vault, this role must first be removed from the inetOrgPerson object before the attribute that disables the account can be cleared. By default, this role should be name `cn=nManagedDisabledRole`, plus the name of your directory servers' root DN. For example `cn=nManagedDisabledRole,dc=example,dc=com`.

16 Click **Next**.

17 On the Synchronization Settings page, fill in the following fields:

**Subscriber Channel Placement Type:** Select the desired form of placement for the Subscriber channel. This option determines the Subscriber channel Placement policies.

- **mirrored:** Places objects hierarchically within the base container.
- **flat:** Places objects only in the base container.

**LDAP Directory Base Container:** Specify the container where user objects reside in the LDAP directory. If you are using a flat Placement rule, this is the container where the users are placed. If you are using a mirrored Placement rule, this is the root container. For example, ou=people,dc=example,dc=com.

**Publisher Channel Placement Type:** Select the desired form of placement for the Publisher channel. This option determines the Publisher channel Placement policies.

- **mirrored:** Places object hierarchically within the base container.
- **flat:** Places objects only in the base container.

18 Click **Next**.

19 (Conditional) This page is displayed only if you selected to install the Data Collection packages and the Account Tracking packages. On the Install LDAP Managed System Information page, fill in the following fields to define your LDAP system:

**Name:** Specify a descriptive name for this LDAP system. The name is displayed in reports.

**Description:** Specify a brief description for this LDAP system. The description is displayed in reports.

**Location:** Specify the physical location of this LDAP system. The location is displayed in reports.

**Vendor:** Specify the vendor of LDAP system. This information is displayed in reports.

**Version:** Specify the version of this LDAP system. The version is displayed in the reports.

20 Click **Next**.

21 (Conditional) This page is displayed only if you selected to install the Data Collection packages and the Account Tracking packages. On the Install LDAP Managed System Information page, fill in the following fields to define the classification of the LDAP system:

**Classification:** Select the classification of the LDAP system. This information is displayed in the reports. You options are:

- Mission-Critical
- Vital
- Not-Critical
- Other

  If you select **Other**, you must specify a custom classification for the LDAP system.

**Environment:** Select the type of environment the LDAP system provides. The options are:

- Development
- Test
- Staging
- Production
- Other

  If you select **Other**, you must specify a custom classification for the LDAP system.

22 Click **Next**.

23 (Conditional) This page is displayed only if you selected to install the Data Collection packages and the Account Tracking packages. On the Install LDAP Managed System Information page, fill in the following fields to define the ownership of the LDAP system:

**Business Owner:** Select a user object in the Identity Vault that is the business owner of the LDAP system. This can only be a user object, not a role, group, or container.

**Application Owner:** Select a user object in the Identity Vault that is the application owner of the LDAP system. This can only be a user object, not a role, group, or container.

24 Click **Next**.

25 (Conditional) This page is displayed only if you selected to install the Account Tracking packages. On the Install LDAP Account Tracking page, fill in the following field:

**Realm:** Specify the name of the realm, security domain, or namespace where the account name is unique.

26 Click **Next**.

27 Review the summary of tasks that will be completed to create the driver, then click **Finish**.

28 After you have installed the driver, you must change the configuration for your environment. Proceed to “Configuring the Driver Object” on page 25.

## Configuring the Driver Object

There are many settings that can help you customize and optimize the driver. The settings are divided into categories such as Driver Configuration, Engine Control Values, and Global Configuration Values (GCVs). Although it is important for you to understand all of the settings, your first priority should be to review the Driver Parameters located on the Driver Configuration page. The Driver Parameters let you configure the LDAP directory type, publication method, and other parameters associated with the Publisher channel.

To access the Driver Properties page:

1. Open your project.
2. In the Modeler, right-click the driver icon or the driver line, then select **Properties**.
3. Click **Apply**.
4. Modify any other settings as necessary.

   You should review the set of default policies and rules provided by the basic driver configuration. Although these policies and rules are suitable for synchronizing with LDAP directory, your synchronization requirements for the driver might differ from the default policies. If this is the case, you need to change them to carry out the policies you want. The default policies and rules are discussed in "How the LDAP Driver Works" on page 10.

   - **Configure the driver filter:** Modify the driver filter to include the object classes and attributes you want synchronized between the Identity Vault and the LDAP directory. For instructions, see Chapter 6, "Synchronizing Data," on page 35.
   - **Configure policies:** Modify the policies as needed. For information about the default configuration policies, see "Policies" on page 11.
   - **Secure the driver connection:** Using a secure connection (SSL) between the Identity Vault and the LDAP directory is strongly recommended. To set up this secure connection, see Chapter 7, "Configuring SSL Connections," on page 41.
   - **Configure password synchronization:** The basic driver configuration is set up to support password synchronization through Universal Password. If you don't want this setup, see "Configuring Password Flow" in the *NetIQ Identity Manager Password Management Guide*.
5. Click **OK** when finished.

After completing the configuration tasks, continue with the next section, Deploying the Driver Object.

## Deploying the Driver Object

After the driver object is created in Designer, it must be deployed into the Identity Vault.

1. In Designer, open your project.
2. In the Modeler, right-click the driver icon or the driver line, then select **Live > Deploy**.
3. If you are authenticated to the Identity Vault, skip to Step 5; otherwise, specify the following information:

   **Host:** Specify the IP address or DNS name of the server hosting the Identity Vault.

   **Username:** Specify the DN of the user object used to authenticate to the Identity Vault.

   **Password:** Specify the user's password.
4. Click **OK**.
5. Read through the deployment summary, then click **Deploy**.
6. Read the successful message, then click **OK**.

7 Click **Define Security Equivalence** to assign rights to the driver.

The driver requires rights to objects within the Identity Vault. The Admin user object is most often used to supply these rights. However, you might want to create a DriversUser (for example) and assign security equivalence to that user. Whatever rights that the driver needs to have on the server, the DriversUser object must have the same security rights.

**7a** Click **Add**, then browse to and select the object with the correct rights.

**7b** Click **OK** twice.

8 Click **Exclude Administrative Roles** to exclude users that should not be synchronized.

You should exclude any administrative User objects (for example, Admin and DriversUser) from synchronization.

**8a** Click **Add**, then browse to and select the user object you want to exclude.

**8b** Click **OK**.

**8c** Repeat Step 8a and Step 8b for each object you want to exclude.

**8d** Click **OK**.

9 Click **OK**.

## Starting the Driver

When a driver is created, it is stopped by default. To make the driver work, you must start the driver and cause events to occur. Identity Manager is an event-driven system, so after the driver is started, it won’t do anything until an event occurs.

To start the driver:

1 In Designer, open your project.

2 In the Modeler, right-click the driver icon or the driver line, then select **Live > Start Driver**.

For information about management tasks with the driver, see Chapter 8, “Managing the Driver,” on page 43.

# Activating the Driver

The Identity Manager driver for LDAP does not need a separate activation. If you create the driver in a driver set where you have already activated the Identity Manager server and service drivers, the driver inherits the activation from the driver set.

If you create the driver in a driver set that has not been previously activated, the driver will run in the evaluation mode for 90 days. You must activate the driver during the evaluation period; otherwise, the driver will be disabled. If you try to run the driver, the trace displays an error message indicating that you need to reactivate the driver to use it. For information on activation, refer to Activating Identity Manager in the *NetIQ Identity Manager Overview and Planning Guide*.

## Adding Packages to an Existing Driver

You can add new functionality to an existing driver by adding new packages to it.

1 Right-click the driver, then click **Properties**.

2 Click **Packages**, then upgrade the already installed LDAP Base package.

  2a Select the package from the list of packages, then click the **Select Operation** cell.

  2b Click **Upgrade** from the drop-down list, then click **Apply**.

  2c Click **OK** to close the Package Management page.

  You can upgrade the Password Synchronization package in a similar way.

3 Click the **Add Packages** icon ✚.

4 Select the packages to install.

5 (Optional) If you want to see all available packages for the driver, clear the **Show only applicable package versions** option, if you want to see all available packages for the driver, then click **OK**.

  This option is only displayed on drivers. By default, only the packages that can be installed on the selected driver are displayed.

6 Click **Apply** to install all of the packages listed with the Install operation.

7 (Conditional) Fill in the fields with appropriate information to install the package you selected for the driver, then click **Next**.

8 Read the summary of the installation, then click **Finish**.

9 Click **OK** to close the Package Management page after you have reviewed the installed packages.

10 Modify the driver configuration settings. See "Configuring the Driver Object" on page 25.

11 Deploy the driver. See "Deploying the Driver Object" on page 26.

12 Start the driver. See "Starting the Driver" on page 27.

13 Repeat Step 1 through Step 9 for each driver where you want to add the new packages.

# 5 Upgrading an Existing Driver

The following sections provide information to help you upgrade an existing driver:

- "Supported Upgrade Paths" on page 29
- "What's New" on page 29
- "Working with MapDB 3.0.5" on page 29
- "Upgrading the Driver" on page 30

## Supported Upgrade Paths

You can upgrade from any 3.*x* version of the LDAP driver. Upgrading a pre-3.*x* version of the driver directly to version 4.0 or later is not supported.

## What's New

### What's New in Version 4.1.1.2

This version of the driver does not provide any new features.

### What's New in Version 4.1.1.1

This version of the driver does not provide any new features.

### What's New in Version 4.1.0

Identity Manager 4.7 provides support for MapDB 3.0.5. To ensure that your driver works correctly with Identity Manager 4.7 engine, see "Working with MapDB 3.0.5" on page 29.

## Working with MapDB 3.0.5

NetIQ recommends that you review the following sections before upgrading your driver to work with Identity Manager 4.7 engine:

- "Understanding Identity Manager 4.7 Engine Support for Driver Versions" on page 30
- "Manually Removing the MapDB Cache Files" on page 30

## Understanding Identity Manager 4.7 Engine Support for Driver Versions

- Drivers shipped with Identity Manager 4.7 are compatible with Identity Manager 4.7 Engine or Remote Loader. You must perform the following actions to complete the driver upgrade:
    1. Upgrade the Identity Manager Engine.
    2. (Conditional) Upgrade the Remote Loader.
    3. Upgrade the driver.
    4. Manually remove the MapDB state cache files from the Identity Vault's DIB directory. For more information, see "Manually Removing the MapDB Cache Files" on page 30.
- Drivers shipped before Identity Manager 4.7 are not compatible with Identity Manager 4.7 Engine or Remote Loader.
- Drivers shipped with Identity Manager 4.7 are not backward compatible with Identity Manager 4.6.x Engine or Remote Loader.
- Drivers shipped with Identity Manager 4.7 are not backward compatible with Identity Manager 4.5.x Engine or Remote Loader.

## Manually Removing the MapDB Cache Files

The Identity Manager engine upgrade process removes the existing MapDB driver work cache files (`dx*`) from the Identity Vault's DIB directory (`/var/opt/novell/eDirectory/data/dib` or `C:\Novell\NDS\DIBFiles`). You must manually remove the existing MapDB state cache files for the driver after upgrading the driver. The MapDB state cache files for the LDAP driver are represented in the following format:

```
ldap_<driver instance guid>_*
```

where ***** is the name of the state cache file for the driver. For example, `ldap_<driver instance guid>_b` or `ldap_<driver instance guid>_b.p`

This action ensures that your driver works correctly with Identity Manager 4.7 engine.

# Upgrading the Driver

The driver upgrade process involves updating the driver files, upgrading the installed driver packages, and updating the Sun-One plug-in.

This section provides general instructions for updating a driver. For information about updating the driver to a specific version, search for that driver patch in the NetIQ Patch Finder Download Page and follow the instructions from the Readme file accompanying the driver patch release.

## Upgrading the Installed Packages

1 Download the latest available packages.

To configure Designer to automatically read the package updates when a new version of a package is available, click **Windows > Preferences > NetIQ > Package Manager > Online Updates** in Designer. For more information about managing packages, see the *NetIQ Designer for Identity Manager Administration Guide*.

2 Upgrade the installed packages.

2a Open the project containing the driver.

2b Right-click the driver for which you want to upgrade an installed package, then click **Driver > Properties**.

2c Click **Packages**.

If there is a newer version of a package, there is check mark displayed in the Upgrades column.

2d Click **Select Operation** for the package that indicates there is an upgrade available.

2e From the drop-down list, click **Upgrade**.

2f Select the version that you want to upgrade to, then click **OK**.

**NOTE:** Designer lists all versions available for upgrade.

2g Click **Apply**.

2h (Conditional) Fill in the fields with appropriate information to upgrade the package, then click **Next**.

Depending on which package you selected to upgrade, you must fill in the required information to upgrade the package.

2i Read the summary of the packages that will be installed, then click **Finish**.

2j Review the upgraded package, then click **OK** to close the Package Management page.

For detailed information, see the Upgrading Installed Packages in the *NetIQ Designer for Identity Manager Administration Guide*.

## Applying the Driver Patch

The driver patch updates the driver files. You can install the patch as a `root` or `non-root` user.

### Prerequisites

Before installing the patch, complete the following steps:

1 Take a back-up of the current driver configuration.

2 (Conditional) If the driver is running with the Identity Manager engine, stop the Identity Vault and the driver instance.

3 (Conditional) If the driver is running with a Remote Loader instance, stop the Remote Loader instance and the driver instance.

4 In a browser, navigate to the NetIQ Patch Finder Download Page.

5. Under **Patches**, click **Search Patches**.
6. Specify **Identity Manager *nn* LDAP Driver nn** in the search box.
7. Download and unzip the contents of the patch file to a temporary location on your server.

   For example, `IDM45_LDAP_410.zip`.

## Applying the Patch as a Root User

In a root installation, the driver patch installs the driver files RPMs in the default locations on Linux. On Windows, you need to manually copy the files to the default locations.

1. Ensure that you have completed the prerequisites for installing the patch. For more information, see “Prerequisites” on page 31.
2. Update the driver files:
   - **Linux:** Log in to your server as `root` and run the following command in a command prompt:

     ```
     rpm -Uvh <Driver Patch File Temporary Location>/linux/novell-DXMLldap.rpm
     ```

     For example, `rpm -Uvh <IDM45_LDAP_410.tar.gz>/linux/`novell-DXMLldap.rpm
   - **Windows:** Navigate to the <*`Extracted Driver Patch File Temporary Location`*>`\windows` folder and copy the following files to `<IdentityManager installation>\NDS\lib` or `<IdentityManager installation>\RemoteLoader\<architecture>\lib` folder.
     - `LDAPShim.jar`
     - `LDAPUtil.jar`
3. (Conditional) Copy the `mapdb.jar` file to `<IdentityManager installation>\NDS\lib` or `<IdentityManager installation>\RemoteLoader\<architecture>\lib` folder.

   This step is only required for driver version 4.0.2 if not performed in the previous driver patch updates.
4. (Conditional) Replace the existing cache (JDBM) with the new cache (mapDB).

   This step is only required for driver version 4.0.2 if not performed in the previous driver updates.

   You need to resynchronize the changes made on the connected system while the driver is stopped for upgrade. This is because a new Cache (DBfiles) is being regenerated and previous cache (JDBM) will not be referenced after upgrade. Therefore, changes during this period must be resynchronized.
5. (Conditional) If the driver is running locally, start the Identity Vault and the driver instance.
6. (Conditional) If the driver is running with a Remote Loader instance, start the Remote Loader and the driver instance.

When JDBM is replaced with mapDB, a new set of DB files are created by mapDB and the existing cache is no longer referenced in the upgraded driver. Therefore, you must set **Synchronize subsequent changes** flag instead of **Synchronize Everything** in the upgraded driver. When this flag is set, the driver takes relatively lesser time to synchronize the objects when you have a large number of objects in the connected system.

**NOTE:** To install the driver on Solaris, call NetIQ Support.

### Applying the Patch as a Non-Root User

1. Verify that <*non-root eDirectory location*>/rpm directory exists and contains the _db.000 file.

   The _db.000 file is created during a non-root installation of the Identity Manager engine. Absence of this file in the directory might indicate that Identity Manager is not properly installed and it might be necessary to re-install it to place the file in the directory.

2. To set the root directory to non-root eDirectory location, enter the following command in the command prompt:

   ROOTDIR=<*non-root eDirectory location*>

   This will set the environmental variables to the directory where eDirectory is installed as a non-root user.

3. To install the driver files, enter the following command:

   ```
   rpm --dbpath $ROOTDIR/rpm -Uvh --relocate=/usr=$ROOTDIR/opt/novell/
   eDirectory --relocate=/etc=$ROOTDIR/etc --relocate=/opt/novell/
   eDirectory=$ROOTDIR/opt/novell/eDirectory --relocate=/opt/novell/
   dirxml=$ROOTDIR/opt/novell/dirxml --relocate=/var=$ROOTDIR/var --
   badreloc --nodeps --replacefiles <rpm-location>
   ```

   For example, to install the LDAP driver RPM, use this command:

   ```
   rpm --dbpath $ROOTDIR/rpm -Uvh --relocate=/usr=$ROOTDIR/opt/novell/
   eDirectory --relocate=/etc=$ROOTDIR/etc --relocate=/opt/novell/
   eDirectory=$ROOTDIR/opt/novell/eDirectory --relocate=/opt/novell/
   dirxml=$ROOTDIR/opt/novell/dirxml --relocate=/var=$ROOTDIR/var --
   badreloc --nodeps --replacefiles /home/user/novell-DXMLLDAP.rpm
   ```

## Installing the SunOne Plug-In

**NOTE:** If your plug-in version matches the version shipped with the driver patch file version, skip this section.

1. Locate the correct plug-in binary file in the extracted driver patch directory.

   The file is located in directories representing the supported platforms under the \sunpasswordplugin4002 directory. For example, if your Sun directory runs on Linux, look in the sunpasswordplugin4002\Linux directory. The name of the plug-in file is novl-idm-pswd.so on all platforms. On Windows, the file is named as novl-idm-pswd.dll.

2. Copy the binary plug-in file to the correct place in your Sun Java System Directory installation.

   For example, the default installation location for Sun Java System Directory on Windows is C:\Program Files\Sun\MPS\lib directory. Copy the novl-idm-pswd.dll to the lib directory. On other platforms, the default installation location is often /var/Sun/mps. Locate the directory on your server and copy the plug-in file to the lib directory.

**NOTE:** On Solaris SPARC computers, the Sun Java System Directory installation includes both 32-bit and 64-bit versions of the library. The default 32-bit version of the library is located in `/var/Sun/mps/lib` and the 64-bit version is located in `/var/Sun/mps/lib/64`. Both 32-bit and a 64-bit versions of the plug-in are included in the driver patch file. You should copy both versions to their respective locations in your Solaris installation directory. At runtime, the Sun Java System Directory appropriately determines which version to load.

**3** After making the changes, restart the Sun directory.

# 6 Synchronizing Data

The following sections provide information to help you control which classes and attributes are synchronized between your Identity Vault and the connected LDAP directory. Not only can you choose which classes and attributes are synchronized, but you can also determine which direction they flow (Identity Vault to LDAP, LDAP to Identity Vault, or both).

---

**IMPORTANT:** When Active Directory Domain Services is used as a connected LDAP server and LDAP driver is used to synchronize user objects from Identity Vault to Active Directory Domain Services, you must set the **dsHeuristics** attribute to `000000001` for restricting the **userPassword** visibility in the LDAP search. For more information, see the Microsoft documentation.

---

## Determining Which Objects Are Synchronized

Identity Manager uses the driver filter, located on both the Publisher and Subscriber channels, to control which objects are synchronized and to define the authoritative data source for these objects.

The following steps provide instructions for editing the filter in Identity Console. For information about editing the filter in Designer, see "Controlling the Flow of Objects with the Filter" in the *NetIQ Identity Manager - Using Designer to Create Policies*.

1 In Identity Console, open the LDAP driver Overview page:

   1a Click **IDM Administration**.

   1b If the driver set is not listed on the **Driver Sets** tab, use the **Search In** field to search for and display the driver set.

   1c Click the LDAP driver icon to display its Overview page.

2 Go to **Data Transformation and Synchronization** tab.

3 Click the Publisher or Subscriber filter icon and make the appropriate changes.

   For every object and attribute selected in the filter, the Schema Mapping policy must have a corresponding entry unless the class or attribute names are the same in both directories (see "Defining Schema Mapping" on page 36). Before mapping an attribute, verify that a corresponding attribute actually exists in the target directory.

# 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 directory 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 you map 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.
- Before mapping a directory attribute to an LDAP server attribute, verify that an LDAP server attribute actually exists. For example, the Full Name attribute is defined for a User object on an Identity Vault, but fullname doesn't exist in an inetOrgPerson object on Netscape.
- Always map attributes to attributes of the same type. For example, map strings attributes to strings attributes, octet attributes to binary attributes, or telenumber attributes to telenumber attributes.
- Map multivalue attributes to multivalue attributes.

The driver doesn't provide data conversion between different attribute types or conversions from multivalue to single-value attributes. The driver also doesn't understand structured attributes except for Facsimile Telephone Number and Postal Address.

Identity Manager is flexible about the syntax that it accepts from the Publisher:

- **Accepting Non-Structured/Non-Octet Syntax:** Identity Manager accepts 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 the Identity Vault is looking for a numeric value, the actual data should be a number.
- **Coercing the Data to Octet:** When Identity Manager is expecting octet data and gets another non-octet/non-structured type, Identity Manager coerces the data to octet by serializing the string value to UTF-8.
- **Coercing the Data to a String:** When Identity Manager is passed octet data and another non-structured type is expected, Identity Manager coerces the data to a string by decoding the Base64 data. Identity Manager next tries 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 applies the same rules as Accepting Non-Structured/Non-Octet Syntax.
- **FaxNumber:** For faxNumber, if a non-structured type is passed in, Accepting Non-Structured/Non-Octet Syntax and Coercing the Data to a String are applied to the data to get the phone number portion of the fax number. The other fields are defaulted.
- **State:** State. For state, False, No, F, N (in either uppercase or lowercase), 0 and " " (empty string) are interpreted as False, and any other value is interpreted as True.

The following steps provide instructions for modifying the Schema Mapping Policy in Identity Console. For information about using Designer, see "Defining Schema Map Policies" in the *NetIQ Identity Manager - Using Designer to Create Policies* guide.

1. In Identity Console, open the LDAP driver Overview page:
   - 1a Click **IDM Administration**.
   - 1b If the driver set is not listed on the **Driver Sets** tab, use the **Search In** field to search for and display the driver set.
   - 1c Click the LDAP driver icon to display its Overview page.
2. Go to **Data Transformation and Synchronization** tab.
3. Click the schema mapping icon on the Publisher or Subscriber channel.
4. Click the policy to display the editing page.
5. Edit the policy as appropriate for your setup.

# Netscape Directory Server Configuration

The following sections apply only when connecting to Netscape Directory Server:

- "Defining Object Placement in Netscape Directory Server" on page 37
- "Working with eDirectory Groups and Netscape" on page 39
- "LDAP Driver Configured to Use eDirectory as Connected Server" on page 39

## Defining Object Placement in Netscape Directory Server

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) is 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 also contains 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 also contains branch points or containers. You need to decide what attributes to use to identify the branch points. Because attribute names have a meaning, 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 following example 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 following Subscriber Placement 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>
```

To configure a placement policy:

1. In Identity Console, click **IDM Administrator**.
2. Locate the driver in its driver set.
3. Open the Identity Manager Driver Overview page by clicking the driver.
4. Go to **Data Transformation and Synchronization** tab.
5. Click the Publisher or Subscriber Placement policy icon, then make the appropriate changes.
6. Click **Close**.

### Working with eDirectory Groups and Netscape

Because group attributes are different in an Identity Vault and Netscape Directory Server, 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 Equivalent To Me attribute in the eDirectory Group. The Equivalent To Me attribute must be included in the Publisher filter. The Equivalent To Me attribute does not need to 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.

### LDAP Driver Configured to Use eDirectory as Connected Server

When a Member is added to a group in Identity Vault and the data is synchronized with the connected system using an LDAP driver, by default in the LDAP driver schema, **Member** is mapped to **uniqueMember**. However, if **eDirectory** is selected as **LDAP Server Type** in driver GCV under **Account status** then, the newly added member information is removed from the Identity vault and the connected system. To prevent this, the schema mapping needs to be updated from “**Member** to **uniqueMember**” to “**Member** to **Member**” on user object class.

## Migrating and Resynchronizing Data

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

- **Migrate Data from the Identity Vault:** Allows you to select containers or objects you want to migrate from an Identity Vault to an LDAP server. When you migrate an object, the Identity Manager engine applies all of the Matching, Placement, and Create policies, as well as the Subscriber filter, to the object.

  **NOTE:** When you migrate data from an Identity Vault into the LDAP directory, you might need to change your LDAP server settings to allow migration of large numbers of objects. See “Migrating Users into an Identity Vault” on page 45.

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

**1** In Identity Console, open the LDAP driver Overview page:

**1a** Click **IDM Administration**.

**1b** If the driver set is not listed on the **Driver Sets** tab, use the **Search In** field to search for and display the driver set.

**1c** Click the LDAP driver icon to display its Overview page.

2 Go to **Data Transformation and Synchronization** > **Migrate into IDVault** tab.

3 Click the migrate button .

# 7 Configuring SSL Connections

The driver uses the LDAP protocol to communicate with the LDAP server. Most LDAP servers allow non-encrypted (clear-text) connections. Additionally, when configured correctly, some LDAP servers allow SSL-encrypted connections. SSL connections encrypt all traffic on the TCP/IP socket by 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 procedure for enabling SSL connection with Sun Java System Directory Server. If you are using another LDAP server, use the methods described in the associated documentation.

1 Start the instance of the Sun Java System Directory Server.

2 Obtain a certificate for the instance and store it in the key database.

The certificate can be issued by a certificate authority (CA) or it can be self-signed. The certificate includes a server certificate and a private key. For information about different methods of obtaining a certificate, see Sun Java System Directory Server documentation. Remember the secure SSL port on the server. The default port is 636.

3 Obtain the signer certificate.

**NOTE:** If the certificate is issued by a CA, the server certificate includes a signer certificate. If the certificate is self-signed, the server certificate acts as the signer certificate. For more information, see Using Certificates and Keys in the Sun ONE Web Server Administrator's Guide.

4 Copy the signer certificate to a temporary directory on the computer where the LDAP driver is installed with which you want to enable the SSL communication.

5 Import the trusted root certificate into a certificate store (also called a keystore) that the driver can use.

**5a** If the certificate is in pkcs12 or pfx format, enter the following at the command line and proceed to Step 5c.

```
keytool -importkeystore -srckeystore <srcfile> -srcstoretype PKCS12
-destkeystore .keystore -alias <keyAlias>
```

**5b** If the certificate is in base 64 or der format, perform the following:

**5b1** Use the keytool utility. The utility is found in the `jre/bin` directory.

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

```
keytool -import -alias TrustedRoot -file A:\PublicKeyCert.b64 -
keystore .keystore -storepass keystorepass
```

**5b2** When you are asked to trust this certificate, select **Yes**, then click **Enter**.

**5c** Copy the `.keystore` file to any directory on the same file system that has the Identity Vault files.

**5d** In Identity Console, select **IDM Administrator** and search for drivers.

**5e** Click the LDAP Driver icon, then go to **Configuration** tab.

**5f** Click the **Driver Parameters** drop down.

**5g** Change the **Use SSL** option as **Yes**, enter the complete path to the `keystore` file.

**6** Enable the driver's SSL parameter and adjust the other SSL parameters as needed. For information, see "Driver Parameters" on page 51.

# 8 Managing the Driver

As you work with the LDAP driver, there are a variety of management tasks you might need to perform, including the following:

- Starting, stopping, and restarting the driver
- Viewing driver version information
- Using Named Passwords to securely store passwords associated with the driver
- Monitoring the driver’s health status
- Backing up the driver
- Inspecting the driver’s cache files
- Viewing the driver’s statistics
- Using the DirXML Command Line utility to perform management tasks through scripts
- Securing the driver and its information
- Synchronizing objects
- Migrating and resynchronizing data
- Activating the driver

Because these tasks, as well as several others, are common to all Identity Manager drivers, they are included in one reference, the *NetIQ Identity Manager Driver Administration Guide*.

# 9 Troubleshooting

## Troubleshooting Driver Processes

Viewing driver processes is necessary to analyze unexpected behavior. To view the driver processing events, use DSTrace. You should only use it during testing and troubleshooting the driver. Running DSTrace while the drivers are in production increases the utilization on the Identity Manager server and can cause events to process very slowly. For more information, see "Viewing Identity Manager Processes" in the *NetIQ Identity Manager Driver Administration Guide*.

## Working with Case-Sensitive DNs

The LDAP driver stores the Fully Qualified Distinguished Name (FQDN) of an object in lowercase as an association value in the connected LDAP server. If you have configured the LDAP server to enforce case sensitivity for DNs, many Subscriber operations such as modify, query, or migrate a user will fail. Therefore, NetIQ does not recommend you to enforce case sensitivity for DNs in the LDAP server.

## Migrating Users into an Identity Vault

Some LDAP servers have settings that limit the number of entries that an LDAP query can return. For example, iPlanet Directory Server 5.1 has a default limit of 5000 objects.

When you migrate user data from LDAP into an Identity Vault, 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 Identity Manager driver continues to run normally.

To fix this, change the limit. For example, do the following in iPlanet:

1. Go to the **Configuration** tab, then select **Database** settings.
2. Raise the look-through limit on the LDBM plug-in tab from the 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, select **Directory Server Settings**, select the **Performance** tab, then raise the size limit according to the number of user accounts you need to migrate.

   This is the actual number of records that the query is allowed to return.

   After these settings have been adjusted, the migration should finish correctly.

# OutOfMemoryError

If you use the LDAP-Search method and the driver shuts down with a java.lang.OutOfMemoryError:

1. Try setting or increasing the DHOST_JVM_INITIAL_HEAP and **DHOST_JVM_MAX_HEAP** environment variables.
2. Restart the driver.
3. Monitor the driver to make sure that the variables provide enough memory.

For more information, see "Configuring Java Environment Parameters" in the *NetIQ Identity Manager Driver Administration Guide*.

# LDAP v3 Compatibility

The LDAP driver works with most LDAP v3 compatible LDAP servers. The driver is written to the LDAP specification (IETF RFCs 4510-4519). To increase compatibility with some LDAP servers that don't fully meet the RFC 2251 requirements, we have added workarounds to the LDAP driver.

One compatibility issue that cannot be ignored or worked around is the RFC 4511 section 4.1.1 requirement that servers allow Message ID values up to 2,147,483,647 (integer values using four bytes).

Oracle Internet Directory version 2.1.1.0.0 (which is part of Oracle 8i) allows only Message ID values up to 32,767 (integer values using two bytes). Therefore, it can't function properly with the LDAP driver.

If you need compatibility with Oracle Internet Directory, Novell recommends upgrading to version 9.2.0.1.0 (included with Oracle 9i) or later.

# Synchronizing Data after Remote Loader Failover on Linux High Availability Cluster

To synchronize data after the Remote Loader is restarted on the Linux High Availability Cluster with LDAPsearch as the publication method, change the **Search Results to Synchronize on First Startup** option to **Synchronize Everything** to capture the events that occur after the LDAP driver starts.

If you set **Synchronize Everything** option for migrating a large number of objects, it might cause a performance issue when you start the driver for the first time after you create the driver. Set the driver option to **Synchronize only subsequent changes** when you start the driver for the first time, change it to **Synchronize Everything**, then restart the driver.

**NOTE:** If your LDAP server supports changelog, use it as a publication method instead of LDAPsearch for better performance and consistent search results.

## Character Change in pub-state-dir

If you specify a value with upper case characters for `pub-state-dir`, the upper case character is converted into lower case when `.zoomdb` files are created. This resulting in the incorrect creation of a different directory structure. The recommendation is to leave the value blank and that would create the `.zoomdb` files correctly in the `dib` folder.

## Frequently Asked Questions

**Question:** Does the LDAP-search method retrieve everything every time, or does it just retrieve updates since the last poll?

**Answer:** The LDAP-search method synchronizes updates from one poll to the next.

**Question:** If I have a choice between using the LDAP-search method or the changelog method, should I use the LDAP Search method?

**Answer:** Use the changelog method because it has performance advantages and is the recommended method.

# A Driver Properties

This section provides information about the Driver Configuration and Global Configuration Values properties for the LDAP driver. These are the only unique properties for drivers. All other driver properties (Named Password, Engine Control Values, Log Level, and so forth) are common to all drivers. Refer to “Driver Properties” in the *NetIQ Identity Manager Driver Administration Guide* for information about the common properties.

- “Driver Configuration” on page 49
- “Global Configuration Values” on page 55

## Driver Configuration

In Identity Console:

1. Click the **IDM Administration** tile.
2. Select the driver set that contains the driver whose properties you want to edit.
3. Click the driver icon to display the driver’s properties page.

In Designer:

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

The Driver Configuration options are divided into the following sections:

- “Driver Module” on page 49
- “Driver Object Password (Identity Console Only)” on page 50
- “Authentication” on page 50
- “Startup Option” on page 51
- “Driver Parameters” on page 51
- “ECMAScript” on page 54
- “Global Configuration” on page 55

### Driver Module

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

**Java:** Used to specify the name of the Java class that is instantiated for the shim component of the driver. This class can be located in the `classes` directory as a class file, or in the `lib` directory as a `.jar` file. If this option is selected, the driver is running locally.

The name of the Java class is: `com.novell.nds.dirxml.driver.ldap.LDAPDriverShim`

**Native:** This option is not used with the driver.

**Connect to Remote Loader:** Used when the driver is connecting remotely to the connected system. Designer includes two suboptions:

- **Remote Loader Configuration for Documentation:** Includes information on the Remote Loader client configuration when Designer generates documentation for the driver.
- **Driver Object Password:** Specifies a password for the Driver object. If you are using the Remote Loader, you must enter a password on this page. Otherwise, the remote driver does not run. The Remote Loader uses this password to authenticate itself to the remote driver shim.

## Driver Object Password (Identity Console Only)

**Driver Object Password:** Use this option to set a password for the driver object. If you are using the Remote Loader, you must enter a password on this page. Otherwise, the remote driver does not run. The Remote Loader uses this password to authenticate itself to the remote driver shim.

## Authentication

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

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

**Authentication ID:** Specifies the DN of the LDAP account that the driver will use for authentication. For example: `Administrator`

**Authentication Context:** Specify the IP address or name of the LDAP server.

**Remote Loader Connection Parameter:** Used only if the driver is connecting to the application through the Remote Loader.

In Identity Console, enter `hostname=xxx.xxx.xxx.xxx port=xxxx secureprotocol=TLS version enforceSuiteB=`*`true/false`* `kmo=certificatename.`

- `hostname` specifies the IP address of the Remote Loader server.
- `port` specifies the TCP/IP port on which the Remote Loader listens for connections from the remote interface shim. The default port for the Remote Loader is 8090.
- `secureprotocol` specifies the version of the TLS protocol that the Remote Loader uses to connect to the Identity Manager engine. Identity Manager supports TLSv1, TLS v1_1, and TLSv1_2 versions only.
- `enforceSuiteB` specifies whether the Remote Loader uses Suite B for communicating with the Identity Manager engine. To use Suite B, specify `enforceSuiteB=true.` The communication supports only TLS version 1.2 version. Communication is not established if the connection has non-Suite B authentication algorithms.
- The kmo entry is optional. Use it only when an SSL connection exists between the Remote Loader and the Identity Manager engine.

  For example: `hostname=10.0.0.1 port=8090 kmo=IDMCertificate`

**Application Password:** Specify the password for the user object listed in the **Authentication ID** field.

**Remote Loader Password:** Used only if the driver is connecting to the application through the Remote Loader. The password is used to control access to the Remote Loader instance. It must be the same password specified during the configuration of the Remote Loader on the connected system.

**Cache limit (KB):** Specify the maximum event cache file size (in KB). If it is set to zero, the file size is unlimited. Click **Unlimited** to set the file size to Unlimited in Designer.

## Startup Option

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

**Auto start:** The driver starts every time the Identity Manager server is started.

**Manual:** The driver does not start when the Identity Manager server is started. The driver must be started through Designer or Identity Console.

**Disabled:** The driver has a cache file that stores all of the events. When the driver is set to **Disabled**, this file is deleted and no new events are stored in the file until the driver state is changed to **Manual** or **Auto Start**.

**Do not automatically synchronize the driver:** This option applies only if the driver is deployed and was previously disabled. If this is not selected, the driver re-synchronizes the next time it is started.

## Driver Parameters

The Driver Parameters section lets you configure the driver-specific parameters. When you change driver parameters, you tune driver behavior to align with your network environment. The parameters are divided into different categories:

- "Driver Options" on page 51
- "Subscriber Options" on page 52
- "Publisher Options" on page 52

### Driver Options

**LDAP Directory Type:** When Isode M-Vault is the target LDAP directory, set the LDAP Directory Type to **M-Vault**. Otherwise, use the **LDAPv3** setting.

**Enforce Matching Parenthesis in Schema Elements:** Select whether the driver enforces matching parenthesis in the LDAP schema objectclass and attribute type definitions. If you choose **No**, the driver ignores the parenthesis syntax infractions in the schema definitions.

**Additional Allowable Schema Name Characters:** Specify extra characters to allow in LDAP objectclass and attribute type names, even when those characters are specifically disallowed by RFC 2252. Some LDAP servers don't always follow the specifications.

**Use SSL:** Select **Yes** to use SSL to secure communication between the driver and the LDAP server. If you use SSL, fill in the following parameters:

- **Keystore Path for SSL Certs:** Specify the full path to the keystore file containing the SSL certificates.

- **Use Mutual Authentication:** Select Yes if you want the driver to use SSL mutual authentication (both client and server), or select No for server authentication only. If you select Yes, you must have the appropriate certificates configured in your keystore.

**Key Alias:** The alias created when importing the public key certificate into the keystore. Typically, you only need to specify the alias when using mutual authentication.

**Keystore Password:** Specify the password used to access the keystore file that contains the SSL certificates.

**Maximum number of operations for a single bind:** Specify the number of LDAP operations after which the driver reconnects to the LDAP server. Change the default value to a large value if the driver does frequent binds.

## Subscriber Options

**LDAP Server Supports Binary Attribute Option:** Most LDAP servers support the use of the binary attribute option as defined in RFC 2251 section 4.1.5.1. If you don't know whether the LDAP server supports the binary attribute option, select Yes.

**Ignore empty components for Postal Address:** The LDAP driver uses a space " " as the value between the Postal Address separator ($) for missing values of Postal Address components. Selecting yes turns on this parameter discards the trailing components that have no value. The default setting is no.

## Publisher Options

**Polling Interval in Seconds:** Specify the interval at which the driver checks the LDAP server for changes. When new changes are found, they are applied to the Identity Vault.

**Enable Paged Search:** Select Yes to enable the paged search of the target LDAP directory objects. The default setting is No.

**Temporary File Directory:** Specify a directory on the local file system (the one where the driver is running) where temporary state files can be written. If you don't specify a path, the driver uses the default driver path:

- **Metadirectory server:** Defaults to the eDirectory DIB file directory.
- **Remote Loader server:** Defaults to the root of the Remote Loader directory.

These files help maintain driver consistency even when the driver is shut down. They also help prevent memory shortages during extensive data searches.

**Heartbeat interval in minutes:** Specify how many minutes of inactivity should elapse before this channel sends a heartbeat document. In practice, more than the number of minutes specified can elapse. That is, this parameter defines a lower bound.

**Publication Method:** Select whether you want to use LDAP Search or changelog as the publication method. The changelog method is the recommended method for LDAP directories that support it. For more information, see "Publication Methods" on page 9.

If you select Changelog, fill in the following fields:

**Changelog Entries to Process on Startup:** Select how the LDAP driver processes entries at startup.

- **All:** The Publisher attempts to process all of the changes found in the change log. The Publisher continues until all changes have been processed. It processes new changes according to the poll rate.
- **None:** When the driver starts running, the Publisher doesn't process any previously existing entries. It processes new changes according to the poll rate.
- **Previously Unprocessed:** This setting is the default. If this is the first time the driver has been run, it behaves like the All option, processing all new changes.

  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 processes new changes according to the poll rate.

**Maximum Batch Size for Changelog Processing:** When the Publisher channel processes new entries from the LDAP change log, the Publisher asks for the entries in batches of this size (the default is 1000). If there are fewer than this number of change log entries, all of them are processed immediately. If there are more than this number, they are processed in consecutive batches of this size.

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

For example, a user object in LDAP is identified with the object classes of inetorgperson, organizationalperson, person, and top. Inetorgperson has the most levels of inheritance in the schema (inheriting from organizationalperson, which inherits from person, which inherits from top). By default, the driver uses inetorgperson as the object class it reports to the Identity Manager engine.

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 LDAP driver 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 Identity Manager engine. If none of the object classes match, it resorts to its default behavior for reporting the primary object class.

**Prevent Loopback:** The Prevent Loopback parameter is used only with the changelog publication method. The LDAP-search method doesn't prevent loopback, other than the loopback prevention built into the Identity Manager engine.

The default behavior for the Publisher channel is to avoid sending changes that the Subscriber channel makes. The Publisher channel detects Subscriber channel changes by looking in the LDAP change log at the creatorsName or modifiersName attribute 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, the Publisher channel assumes that this change was made by the driver's Subscriber channel and doesn't synchronize the change.

If you select LDAP Search, fill in the following fields:

**LDAP Directory Base Container:** Specify the container where user objects reside in the LDAP directory. If you are using a flat Placement rule, this is the container where the users are placed. If you are using a mirrored Placement rule, this is the root container.

**Search Scope:** Indicates the depth of the polling searches. This parameter defaults to search the entire subtree that the LDAP base-dn points to.

**Class Processing Order:** Use this parameter to order certain events when referential attributes are an issue. The value of the parameter is a list of class names from the LDAP server, separated by spaces. For example, to make sure that new users are created before they are added to groups, make sure that interorgperson comes before groupofuniquenames.

The driver defines a special class name, `others`, to mean all classes other than those explicitly listed.

The default value for this parameter is `others groupofuniquenames`.

**Search Results to Synchronize on First Startup:** This parameter defines whether the initial search results are synchronized, or only subsequent changes are synchronized.

**LDAP search filters to filter on individual attributes:** Specify the LDAP search filters to filter the individual attributes for different classes which are in Driver filter. If you don't specify this option, the search is done based only on the objectclasses in the Driver filter like `objectclass=inetorgperson`. If there are n classes in the Driver filter, you can specify a maximum of n LDAP search filters separated by space. Each search filter is for its corresponding class in the driver filter. The following is an example of a search filter:

```
(&(objectclass=inetorgperson)(cn=test))
```

**Search Results to Synchronize on First Startup:** The first time the driver starts, it performs the defined LDAP search. This setting defines whether the initial search results are synchronized, or only the subsequent changes are synchronized.

If you select No Publisher, there are no additional fields.

**Use Sun Password Plugin:** Select Yes if you have installed and configured the Novell Identity Manager Password plugin on Sun Java System Directory and you want to use it to synchronize to the Identity manager distribution password.

## ECMAScript

Displays an ordered list of ECMAScript resource files. The files contain extension functions for the driver that Identity Manager loads when the driver starts. You can add additional files, remove existing files, or change the order the files are executed.

## Global Configuration

Displays an ordered list of Global Configuration objects. The objects contain extension GCV definitions for the driver that Identity Manager loads when the driver is started. You can add or remove the Global Configuration objects, and you can change the order in which the objects are executed.

# Global Configuration Values

Global configuration values (GCVs) are values that can be used by the driver to control functionality. GCVs are defined on the driver or on the driver set. Driver set GCVs can be used by all drivers in the driver set. Driver GCVs can be used only by the driver on which they are defined.

The LDAP driver includes many GCVs. You can also add your own if you discover you need additional ones as you implement policies in the driver.

To access the driver's GCVs in Identity Console:

1. Click the **IDM Administration** tile.
2. Select the driver set that contains the driver whose properties you want to edit.
3. Locate the driver icon, then click the driver icon to display the driver's properties page.
4. Click **Global Config Values** drop down to display the GCV page.

To access the driver's GCVs in Designer:

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

   or

   To add a GCV to the driver set, right-click the driver set icon, then click **Properties > GCVs**.

The global configuration values are organized as follows:

- "Driver Parameters" on page 55
- "Entitlements" on page 56
- "Password Synchronization" on page 58
- "Account Status Support" on page 58
- "Account Tracking" on page 59

## Driver Parameters

**Subscriber Channel Placement Type:** Select the desired form of placement for the Subscriber channel. This option determines the Subscriber channel Placement policies.

- **mirrored:** Places objects hierarchically within the base container.
- **flat:** Places objects only in the base container.

**LDAP Directory Base Container:** Specify the container where user objects reside in the LDAP directory. If you are using a flat Placement rule, this is the container where the users are placed. If you are using a mirrored Placement rule, this is the root container. For example, ou=people,dc=example,dc=com.

**Publisher Channel Placement Type:** Select the desired form of placement for the Publisher channel. This option determines the Publisher channel Placement policies.

- **mirrored:** Places object hierarchically within the base container.
- **flat:** Places objects only in the base container.

# Entitlements

Entitlements act like an On/Off switch to control account access. When the driver is enabled for entitlements, accounts are created and removed or disabled only when the account entitlement is granted or revoked from users. For more information, see the *NetIQ Identity Manager Entitlements Guide*.

There are multiple sections in the Entitlements tab. Depending on which packages you installed, different options are enabled or displayed.

- "Entitlements" on page 56
- "Data Collection" on page 57
- "Role Mapping" on page 57
- "Resource Mapping" on page 57
- "Parameter Format" on page 57
- "Entitlement Extensions" on page 57

## Entitlements

**Use Entitlements to Control LDAP Accounts?:** Select **True** to enable the driver to manage LDAP accounts based on the driver's defined entitlements.

Select **False** to disable management of LDAP accounts based on the entitlements.

**Account action on Entitlement Revoke:** Select the action to take when an LDAP User Account entitlement is revoked. The options are:

- Do Nothing
- Disable User
- Delete User

**Use Group Entitlement:** Select **True** to enable the driver to manage LDAP groups based on the driver's defined entitlements.

Select **False** to disable management of LDAP groups based on the entitlements.

**Advanced settings:** Select **show** to display the entitlement options that allow or deny additional functionality like data collection and others. These settings should rarely be changed.

## Data Collection

Data collection enables Identity Reporting to gather information to generate reports. For more information, see the *Administrator Guide to NetIQ Identity Reporting*.

**Enable data collection:** Select **Yes** to enable data collection for the driver through Data Collection Service by the Managed System Gateway driver. If you are not going to run reports on data collected by this driver, select **No**.

**Allow data collection from user accounts:** Select **Yes** to allow data collection by Data Collection Service through the Managed System Gateway driver for the user accounts.

**Allow data collection from groups:** Select Yes to allow data collection by Data Collection Service through the Managed System Gateway driver for groups.

### Role Mapping

Identity Applications allow you to map business roles with IT roles.

**Enable role mapping:** Select Yes to make this driver visible to Identity Applications.

**Allow mapping of user accounts:** Select Yes if you want to allow mapping of user accounts in Identity Applications. An account is required before a role, profile, or license can be granted through Identity Applications.

**Allow mapping of groups:** Select Yes if you want to allow mapping of groups in Identity Applications.

### Resource Mapping

Identity Applications allow you to map resources to users. For more information, see the *NetIQ Identity Manager - User's Guide to the Identity Applications*.

**Enables resource mapping:** Select Yes to make this driver visible to Identity Applications.

**Allow mapping of user accounts:** Select Yes if you want to allow mapping of user accounts in Identity Applications. An account is required before a role, profile, or license can be granted.

**Allow mapping of groups:** Select Yes if you want to allow mapping of groups in Identity Applications.

### Parameter Format

**Format for Account entitlement:** Select the parameter format the entitlement agent must use when granting this entitlement. The options are **Identity Manager 4** or **Legacy**.

**Format for Group entitlement:** Select the parameter format the entitlement agent must use when granting this entitlement. The options are **Identity Manager 4** or **Legacy**.

### Entitlement Extensions

**User account extensions:** The content of this field is added below the entitlement elements in the EntitlementConfiguration resource object.

**Group extensions:** The content of this field is added below the entitlement elements in the EntitlementConfiguration resource object.

## Password Synchronization

These GCVs enable password synchronization between the Identity Vault and the LDAP system.

In Designer, you must click the icon next to a GCV to edit it. This displays the Password Synchronization Options dialog box for a better view of the relationship between the different GCVs.

In Identity Console, to edit the Password management options go to **Configuration > Global Configuration Values**, and then edit it in your Password synchronization policy tab.

For more information about how to use the Password Management GCVs, see "Configuring Password Flow" in the *NetIQ Identity Manager Password Management Guide*.

**Connected System or Driver Name:** Specify the name of the LDAP system or the driver name. This valued is used by the e-mail notification template to identity the source of the notification message.

**Application accepts passwords from Identity Manager:** If **True**, allows passwords to flow from the Identity Manager data store to the connected system.

**Identity Manager accepts passwords from application:** If **True**, allows passwords to flow from the connected system to Identity Manager.

**Publish passwords to NDS password:** Use the password from the connected system to set the non-reversible NDS password in eDirectory.

**Publish passwords to Distribution Password:** Use the password from the connected system to set the NMAS Distribution Password used for Identity Manager password synchronization.

**Require password policy validation before publishing passwords:** If **True**, applies NMAS password policies during publish password operations. The password is not written to the data store if it does not comply.

**Reset user's external system password to the Identity Manager password on failure:** If **True**, on a publish Distribution Password failure, attempts to reset the password in the connected system by using the Distribution Password from the Identity Manager data store.

**Notify the user of password synchronization failure via e-mail:** If **True**, notifies the user by e-mail of any password synchronization failures.

## Account Status Support

**LDAP Server Type:** Select the LDAP server type this driver connects to. Based on the type, the appropriate attribute is modified in the LDAP directory to disable the account. The options are:

- eDirectory
- openLDAP
- iPlanet | SunOne | OID

**nsManageDisabledRole DN:** This options is only displayed if you select **iPlanet | SunOne |OID.** This is the DN of the role used to disable users in the connected LDAP system. If a user is enabled in the Identity Vault, this role must first be removed from the inetOrgPerson object before the attribute that disables the account can be cleared. By default, this role should be name cn=nManagedDisabledRole, plus the name of your directory servers' root DN. For example cn=nManagedDisabledRole,dc=example,dc=com.

## Account Tracking

Account tracking is part of Identity Reporting. For more information, see the *Administrator Guide to NetIQ Identity Reporting*.

**Enable account tracking:** Set this to **True** to enable account tracking policies. Set it to **False** if you do not want to execute account tracking policies.

**Realm:** Specify the name of the realm, security domain, or namespace in which the account name is unique.

**Advanced settings** Select **show** to display the account tracking settings. Changing these settings might result in malfunction of the Account Tracking feature. Only change these settings if you know exactly what you are doing.

**Identifiers:** Add the account identifier attributes. Attribute names must be in the application namespace.

**Object Class:** Add the object class to track. Class names must be in the application namespace.

**Status attribute:** Name of the attribute in the application namespace to represent the account status.

**Status active value:** Value of the status attribute that represents an active state.

**Status inactive value:** Value of the status attribute that represents an inactive state.

**Subscription default status:** Select the default status that the policies assume when an object is subscribed to the application and the status attribute is not set in the Identity Vault.

**Publication default status:** Select the default status that the policies assume when an object is published to the Identity Vault and the status attribute is not set in the application.

# B Trace Levels

The driver supports the following trace levels:

| Level | Description |
|---|---|
| 1 | Minimal Tracing |
| 2 | Previous level and LDAP Server information, LDAP operations in the Subscriber channel, Changelog Information |
| 3 | Previous level and LDAPSearch messages in the Publisher channel and all the remaining messages |
| 4 | Previous level and LDAPSearch messages in the Publisher channel and all the remaining messages |
| 5 | Previous level and password synchronization messages in the Publisher channel |

For information about setting driver trace levels, see “Viewing Identity Manager Processes” in the *NetIQ Identity Manager Driver Administration Guide*.