www.novell.com/documentation

Driver for eDirectory Implementation Guide

Identity Manager 4.0.1

April 15, 2011

Legal Notices

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

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

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

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

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

Novell Trademarks

For a list of Novell trademarks, see Trademarks (http://www.novell.com/company/legal/trademarks/tmlist.html).

Third-Party Materials

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

Contents

About This Guide

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

Audience

This guide is for Novell eDirectory and Identity Manager administrators who are using the Identity Manager Driver for eDirectory.

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Use the User Comment feature at the bottom of each page of the online documentation, or go to www.novell.com/documentation/feedback.html and enter your comments there.

Documentation Updates

For the most recent version of this document, see the Identity Manager 4.0.1 Drivers Documentation Web site (http://www.novell.com/documentation/idm401drivers/index.html).

Additional Documentation

For information on Identity Manager 4.0 and other Identity Manager 4.0 drivers, see the Identity Manager 4.0.1 Documentation Web site (http://www.novell.com/documentation/idm401).

1 Understanding the eDirectory Driver

The Identity Manager Driver for eDirectory synchronizes objects and attributes between different eDirectory trees.

This driver is unique among all other Identity Manager drivers. Because you are synchronizing data between eDirectory trees, you will always have two drivers installed, each in its own tree. The driver in one tree communicates with the driver in the other tree.

1.1 Driver Concepts

1.1.1 Key Terms

Driver: A set of policies, filters, and objects that act as the connector between an Identity Vault and the driver shim.

This software enables an application to publish events from an application to the directory, enables an application to subscribe to events from the directory, and synchronizes data between the directory and applications.

To establish a connection between the Metadirectory engine and an Identity Vault, you specify the driver’s configuration and connection parameters, policies, and filter values.

Driver object: A collection of channels, policies, rules, and filters that connect an application to an Identity Vault that is running Identity Manager.

Each driver performs different tasks. Policies, rules, and filters tell the driver how to manipulate the data to perform those tasks.

The Driver object displays information about the driver’s configuration, policies, and filters. This object enables you to manage the driver and provide eDirectory management of the driver shim parameters.

Driver shim: A Java file (`NdsToNds.jar`) loaded directly by Identity Manager. Communicates event changes to be sent from the Identity Manager Driver for eDirectory to an Identity Vault, communicates changes from the Identity Vault to the Identity Manager Driver for eDirectory, and operates as the link that connects the Identity Vault and the Identity Vault Driver object.

Identity Vault. A hub, with applications and directories publishing their changes to it. The Identity Vault then sends changes to the applications and directories that have subscribed for them. This results in two main flows of data: the Publisher channel and the Subscriber channel.

1.1.2 How the eDirectory Driver Works

Channels, filters, and policies control data flow.

Publisher and Subscriber Channels: The eDirectory driver is installed and configured in two trees. The driver's Publisher channel in TreeA communicates with the driver's Subscriber channel in TreeB. Conversely, the driver's Publisher channel in TreeB communicates with the driver's Subscriber channel in TreeA.

Filters: Identity Manager uses filters to control which objects and attributes are shared. The default filter configurations for the eDirectory driver allow objects and attributes to be shared. For a list of synchronized attributes, see Appendix B, "Synchronized Attributes," on page 43.

Policies: Identity Manager uses policies to control data synchronization between the eDirectory driver and the Identity Vaults.

The driver does not have any direct communication with eDirectory. It communicates with eDirectory via the Identity Manager engine. The driver is usually configured in pairs with an instance of the driver configured in each of the two trees being synchronized. The Subscriber of the driver instance in each tree connects to the Publisher of the driver instance in the other tree via a TCP connection. The connection is formed on demand when a driver instance needs to communicate with the other driver instance. When the connection is established, it remains intact until the connection is broken or one of the driver instances is stopped. If there is a broken connection or an attempt to connect to the other instance fails, the Subscriber channel issues a retry status to the Identity Manager engine and then attempts to reconnect when the Identity Manager engine resends the event. The port used for the connections is configurable and can be different for Subscriber to Publisher pair.

The Subscriber of each driver instance acts primarily as an event source for the Publisher of the other driver instance which in turn is the event source for the Publisher channel of the Identity Manager engine. Events provided by the Subscriber channel of one instance of the driver are passed across the TCP connection unchanged to the Publisher channel of the other instance of the driver which is then passed, mostly unchanged, to the subscriber channel of the Identity Manager engine.

Because the Subscriber channel is primarily an event source for the Publisher channel of the other instance, policies are usually not in place on the Publisher channel of either instance. The primary exception to this rule is that if any custom event filtering is to be done, it is usually more efficient to do that filtering as early in the dataflow as possible, which usually means in the Subscriber Event Transformation policy. The filters are also usually configured such that the filter of the Subscriber filter of one instance is identical (with the exception of the GUID attribute) to the Publisher filter of the other instance.

There are two different protocols used to transport the XML documents between the driver instances.

To guard against undetected loss of connectivity, such as when the network is physically unavailable or the other server crashes without gracefully closing all of its connections, a keep-alive message is periodically sent in each direction across an open connection. This causes an I/O error if the connection breaks in such a way which may be undetectable otherwise. The frequency of the keep-alive packets is configurable. You can tune it to either minimize the network traffic or minimize the amount of time it takes to detect recover from a broken connection.

The driver uses a value generated from the GUID attribute of one of the instances of the driver. The GUID attribute is required to be in the Subscriber filter for each class that is being synchronized. The GUID attribute is not present in the Publisher filter because it is usually not desirable to synchronize a value that is supposed to be globally unique. The GUID attribute is used to generate an association key only when the Publisher channel of a driver passes through an add event for an object that doesn't already have an association key.

The rights to objects are granted to the driver object by standard eDirectory rights management. Sufficient rights must be granted to the objects to perform the desired operations on the desired objects and attributes in each tree.

The connections between the two driver instances can be either clear or secured by SSL. Configuration of SSL requires the creation of a server certificate for each publisher that is signed by a certificate authority that is trusted by the Subscriber channel. For more information, see Chapter 5, "Securing Driver Communication," on page 23.

1.2 Driver Concepts

- Section 1.2.1, "Key Terms," on page 9

1.2.1 Key Terms

Driver: A set of policies, filters, and objects that act as the connector between an Identity Vault and the driver shim.

This software enables an application to publish events from an application to the directory, enables an application to subscribe to events from the directory, and synchronizes data between the directory and applications.

To establish a connection between the Metadirectory engine and an Identity Vault, you specify the driver's configuration and connection parameters, policies, and filter values.

Driver object: A collection of channels, policies, rules, and filters that connect an application to an Identity Vault that is running Identity Manager.

Each driver performs different tasks. Policies, rules, and filters tell the driver how to manipulate the data to perform those tasks.

The Driver object displays information about the driver's configuration, policies, and filters. This object enables you to manage the driver and provide eDirectory management of the driver shim parameters.

Driver shim: A Java file (`NdsToNds.jar`) loaded directly by Identity Manager. Communicates event changes to be sent from the Identity Manager Driver for eDirectory to an Identity Vault, communicates changes from the Identity Vault to the Identity Manager Driver for eDirectory, and operates as the link that connects the Identity Vault and the Identity Vault Driver object.

Identity Vault. A hub, with applications and directories publishing their changes to it. The Identity Vault then sends changes to the applications and directories that have subscribed for them. This results in two main flows of data: the Publisher channel and the Subscriber channel.

1.3 Driver Features

- Section 1.3.1, "Local Platforms," on page 10
- Section 1.3.2, "Remote Platforms," on page 10
- Section 1.3.3, "Entitlements," on page 10

- Section 1.3.4, “Password Synchronization,” on page 10
- Section 1.3.5, “Synchronizing Data,” on page 10

1.3.1 Local Platforms

The eDirectory driver runs in any Identity Manager installation. See “System Requirements” in the *Identity Manager 4.0.1 Integrated Installation Guide*.

1.3.2 Remote Platforms

The eDirectory driver supports remote connections without the Remote Loader. The driver does not use the Remote Loader because the driver in one tree communicates directly with the driver in the other tree.

1.3.3 Entitlements

The eDirectory driver example configuration does not implement any entitlements. However, the driver does support entitlements you create. For more information about entitlements, see the *Identity Manager 4.0.1 Entitlements Guide*.

1.3.4 Password Synchronization

The eDirectory driver supports password synchronization via Universal Password. If desired, you can also use the older form of password synchronization (Public/Private key pair or NDS password). For more information, see Chapter 6, “Synchronizing Passwords,” on page 27.

1.3.5 Synchronizing Data

The eDirectory driver synchronizes data between two Identity Vaults or trees. The driver can run anywhere that a Metadirectory server is running.

2 Installing the Driver Files

If you are synchronizing information between TreeA and TreeB, you must install the Metadirectory engine and eDirectory driver on eDirectory servers in both trees. Therefore, the installation must be completed twice—once for the Metadirectory engine and eDirectory driver in TreeA and once in TreeB. Use the instructions in "Installing Identity Manager" in the *Identity Manager 4.0.1 Integrated Installation Guide*.

The eDirectory servers where you install the driver must hold master or read/write replicas of the objects you want synchronized between the two trees.

The installation program extends the Identity Vault (eDirectory) schema and installs the driver shim. It does not create the driver in the Identity Vault (see Chapter 3, "Creating a New Driver," on page 13) or upgrade an existing driver's configuration (see Chapter 4, "Upgrading an Existing Driver," on page 21).

The eDirectory driver does not use the Remote Loader because the driver in one tree communicates directly with the driver in the other tree.

The eDirectory driver requires the following:

- Novell Certificate Server running on each server that hosts the driver.
- A certificate authority (CA) to support SSL encryption between drivers.

Novell Certificate Server and the certificate authority are discussed more in Chapter 5, "Securing Driver Communication," on page 23.

3 Creating a New Driver

After the eDirectory driver files are installed on the server where you want to run the driver (see Chapter 2, "Installing the Driver Files," on page 11), you can create the driver in the Identity Vault. You do so by installing the driver packages and then modifying the driver configuration to suit your environment. The following sections provide instructions:

- Section 3.1, "Creating the Driver in Designer," on page 13
- Section 3.2, "Creating the Driver in iManager," on page 19
- Section 3.3, "Activating the Driver," on page 19

3.1 Creating the Driver in Designer

You create the eDirectory driver 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.

To connect two trees, you need to complete the following procedures for the drivers that are installed in each Identity Vault.

- Section 3.1.1, "Importing the Current Driver Packages," on page 13
- Section 3.1.2, "Installing the Driver Packages," on page 14
- Section 3.1.3, "Configuring the Driver," on page 17
- Section 3.1.4, "Deploying the Driver," on page 18
- Section 3.1.5, "Starting the Driver," on page 19

3.1.1 Importing the Current Driver Packages

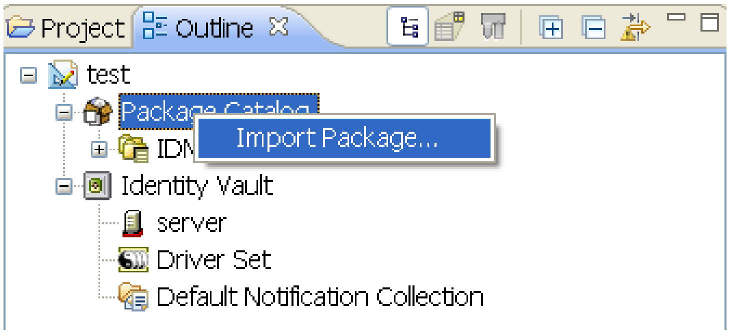
The driver packages contain the items required to create a driver, such as policies, entitlements, filters, and Schema Mapping policies. These packages are only available in Designer and can be updated often. You must have the most current version of the packages imported into the Package Catalog before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

To verify you have the most recent version of the driver packages imported into 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 eDirectory 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 Section 3.1.2, "Installing the Driver Packages," on page 14.

3.1.2 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 *eDirectory Base*, then click *Next*.

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

Default Configuration: These packages contain the default configuration information for the eDirectory driver. Always leave this option selected.

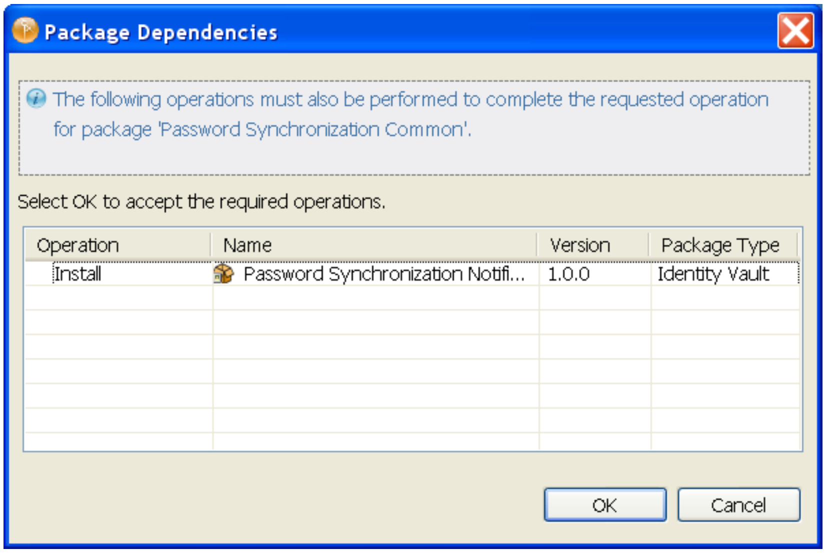
Password Synchronization: These packages contain the policies required to enable password synchronization. Leave this option selected if you want to synchronize passwords between the Identity Vaults.

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

Account Tracking: This group of packages contain the policies that enable account tracking information for reports. If you are using the Identity Reporting Module, verify that this option is selected. For more information, see the *Identity Reporting Module Guide*.

5 After selecting the optional packages, click *Next*.

6 (Conditional) If there are package dependencies for the packages you selected to install, you must install these dependencies to install the selected packages. Click *OK* to install the Password Synchronization Notification package dependency.

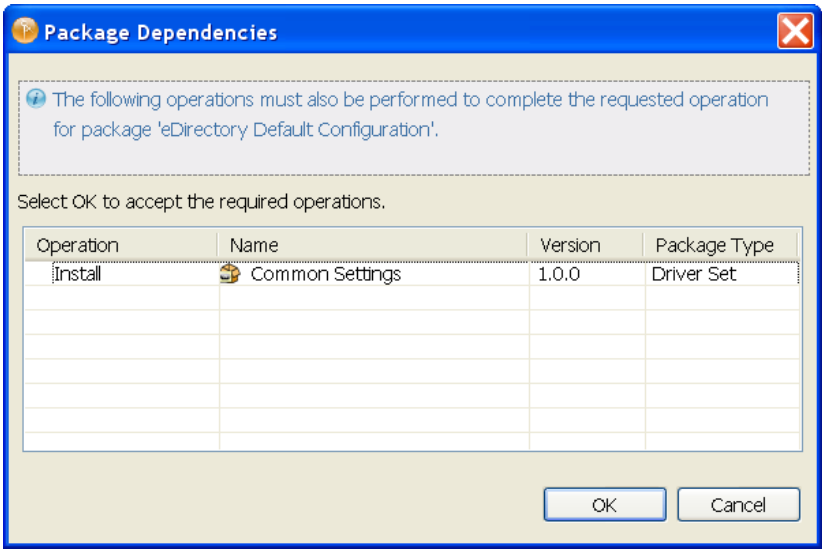


The following operations must also be performed to complete the requested operation for package 'Password Synchronization Common'.

Select OK to accept the required operations.

Operation	Name	Version	Package Type
Install	Password Synchronization Notifi...	1.0.0	Identity Vault

7 (Conditional) Click *OK* to install the Common Settings package, if you have not installed any other packages into the selected driver set.



The following operations must also be performed to complete the requested operation for package 'eDirectory Default Configuration'.

Select OK to accept the required operations.

Operation	Name	Version	Package Type
Install	Common Settings	1.0.0	Driver Set

8 Click *OK* to install the Advanced Java Class package if you have not installed any other packages into the selected driver set.

9 (Conditional) Fill in the following fields on the Common Settings page:

The Common Settings page is displayed only if the Common Settings package is installed as a dependency.

User Container: Select the Identity Vault container where the users are added if they don't already exist in the Identity Vault. This value becomes the default value 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 groups are added if they don't already exist in the Identity Vault. This value becomes the default value 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.

10 Click *Next*.

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

12 Fill in the following field to configure the driver:

Remote Tree Address and Port: Specify the hostname or IP address, and port of the server in the remote Identity Vault.

13 Click *Next*.

14 Fill in the following fields on the eDirectory Default Configuration page:

eDirectory Publisher Placement type: Select how the objects are placed in the remote Identity Vault and the local Identity Vault. The options are:

- **Mirrored:** Mirrors the structure between the remote Identity Vault and the local Identity Vault.

  If you choose this option, use the same option for configuring both eDirectory trees you are synchronizing.

  This option in the driver configuration synchronizes User, Group, Organization, Country, and Organizational Unit objects. It also mirrors the structure of a subtree in the other tree.

- **Flat:** All of the objects are placed into a single container.

  This option synchronizes User and Group objects and places all users in one container and all groups in another container.

  This option is typically used in conjunction with the Department option (or a similar configuration) in the other tree.

  This option doesn't create the containers that hold the users and groups. You must create those manually.

- **Department:** Users are placed in containers named after the department.

  This option synchronizes User and Group objects and places all users and groups in a container based on the *Department* field in your management console.

  This configuration is typically used in conjunction with the Flat option (or a similar configuration) in the other tree.

  This option doesn't create the containers for each department. You must create those manually. They must be the same as the container specified during import.

Remote Tree Base User Container: Specify the source container of the user objects in the remote Identity Vault.

Remote Tree Base Groups Container: Specify the source container of the group objects in the remote Identity Vault.

15 Click *Next*.

16 (Conditional) Fill in the following fields on the eDirectory Managed System Information page. This page is displayed only if you selected to install the Data Collection and Account Tracking groups of packages.

Name: Specify a descriptive name for this Identity Vault. The name is displayed in the reports.

Description: Specify a brief description of the this Identity Vault. The description is displayed in the reports.

Location: Specify the physical location of this Identity Vault. The location is displayed in the reports.

Vendor: Select Novell as the vendor of this system. The vendor information is displayed in the reports.

Version: Specify the version of this Identity Vault. The version is displayed in the reports.

17 Click *Next*.

18 (Conditional) Fill in the following fields to define the ownership of this Identity Vault. This page is displayed only if you selected to install the Data Collection and Account Tracking groups of packages.

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

19 Click *Next*.

20 (Conditional) Fill in the following fields to define the classification of the Identity Vault. This page is only displayed if you selected to install the Data Collection and Account Tracking groups of packages.

Classification: Select the classification of the Identity Vault. This information is displayed in the reports. The options are:

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

  If you select *Other*, you must specify a custom classification for the Identity Vault.

Environment: Select the type of environment the Identity Vault provides. The options are:

- Development
- Test
- Staging
- Production
- Other

  If you select *Other*, you must specify a custom classification for the Identity Vault.

21 Click *Next*.

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

23 After the driver packages are installed, there is additional configuration required for the eDirectory driver. Continue to Section 3.1.3, "Configuring the Driver," on page 17 to configure the driver.

3.1.3 Configuring the Driver

After installing the driver packages, the eDirectory driver will run. However, the basic configuration might not meet the requirements for your environment. You should complete the following tasks to configure the driver:

- **Secure the driver connection:** eDirectory drivers communicate via SSL using digital certificates for authentication. You need to set up this secure connection. See Chapter 5, "Securing Driver Communication," on page 23.

- **Configure the driver filter:** Modify the driver filter to include the object classes and attributes you want synchronized between the two eDirectory trees. For information about the classes and attributes include in the filter for the basic configuration, see Appendix B, “Synchronized Attributes,” on page 43.
- **Configure policies:** Modify the policies as needed. Policies should generally be placed only on the Publisher channel, not on the Subscriber channel. The Matching and Placement policies cannot operate correctly on the Subscriber channel because the Subscriber channel is acting primarily as a source of events for the Publisher channel of the other tree.

  You might consider placing an Event Transform or Create Policy on the Subscriber channel to prevent sending unnecessary data across the channel. See “Using Scope Filtering to Manage Users on Different Servers ” in the *Identity Manager 4.0.1 Framework Installation Guide*.
- **Configure password synchronization:** The basic driver configuration is set up to support bidirectional password synchronization through Universal Password. If you don’t want this setup, see Chapter 6, “Synchronizing Passwords,” on page 27.

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

3.1.4 Deploying the Driver

After a driver 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*.

3.1.5 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 7, "Managing the Driver," on page 29.

3.2 Creating the Driver in iManager

Drivers are created with packages, and iManager does not support packages. In order to create or modify drivers, you must use Designer. See Section 3.1, "Creating the Driver in Designer," on page 13.

3.3 Activating the Driver

If you created the driver in a driver set where you have already activated the Metadirectory engine and service drivers, the driver inherits the activation. If you created the driver in a driver set that has not been activated, you must activate the driver within 90 days. Otherwise, the driver stops working.

For information on activation, refer to "Activating Novell Identity Manager Products" in the *Identity Manager 4.0.1 Integrated Installation Guide*.

4 Upgrading an Existing Driver

The following sections provide information to help you upgrade an existing driver to version 4.0.1:

- Section 4.1, “Supported Upgrade Paths,” on page 21
- Section 4.2, “What’s New in Version 4.0.1,” on page 21
- Section 4.3, “Upgrade Procedure,” on page 21

4.1 Supported Upgrade Paths

You can upgrade from any 3.*x* version of the eDirectory driver. Upgrading a pre-3.*x* version of the driver directly to version 4.0.1 is not supported.

4.2 What’s New in Version 4.0.1

Version 4.0.1 of the driver does not include any new features.

From 4.0 version of the eDirectory driver, driver content is delivered in packages instead of through a driver configuration file.

4.3 Upgrade Procedure

You can upgrade one eDirectory driver and then upgrade the second driver. It is not a requirement to upgrade both drivers at the same time, but is it recommended that you upgrade both drivers within a short amount of time.

The process for upgrading the eDirectory driver is the same as for other Identity Manager drivers. For detailed instructions, see “Upgrading Drivers to Packages” in the *Identity Manager 4.0.1 Upgrade and Migration Guide*.

5 Securing Driver Communication

To provide security while transmitting information between two Identity Vaults, you must configure the eDirectory driver to communicate with the destination eDirectory driver through an SSL connection.

In addition, you can provide additional security by requiring the two eDirectory drivers to authenticate to one another. Although this is optional, it is strongly recommended.

The following sections explain how to set up SSL and configure driver authentication:

- Section 5.1, "Configuring Secure Data Transfers," on page 23
- Section 5.2, "Configuring Authentication Between Drivers," on page 25

5.1 Configuring Secure Data Transfers

All eDirectory driver communication is secured through SSL. To configure your eDirectory drivers to handle secure data transfers, run the NDS-to-NDS Driver Certificate Wizard in iManager.

You can also configure your eDirectory drivers through Designer. For information about configuring eDirectory drivers through Designer, see Designer 4.0.1 for Identity Manager 4.0.1 Administration Guide (http://www.novell.com/documentation/idm401/designer_admin/data/appedir2certifs.html)

- Section 5.1.1, "Understanding Secure Connections via the eDirectory Driver," on page 23
- Section 5.1.2, "Setting Up a KMO," on page 24

5.1.1 Understanding Secure Connections via the eDirectory Driver

The following items can help you understand how secure connections are established when using the eDirectory driver:

- The driver uses SSL sockets to provide authentication and a secure connection. SSL uses digital certificates to allow the parties to an SSL connection to authenticate one another. Identity Manager in turn uses Novell Certificate Server certificates for secure management of sensitive data.
- To use the driver, you must have the Novell Certificate Server running in each tree. We recommend that you use the certificate authority from one of the trees containing the driver to issue the certificates used for SSL. If your tree does not have a certificate authority, you need to create one. You can use an external certificate authority. For information about Novell Certificate Server, see the Novell Certificate Server 3.3 Documentation Web site (http://www.novell.com/documentation/crt33/).
- The Novell implementation of SSL that the driver uses is based on Novell Secure Authentication Services (SAS) and NTLS for eDirectory. These must be installed and configured on the server where the driver runs. eDirectory usually does this automatically.

- To configure driver security, it is necessary to create and reference certificates in the eDirectory trees that will be connected using the driver. Certificate objects in eDirectory are called Key Material Objects (KMOs) because they securely contain both the certificate data (including the public key) and the private key associated with the certificate.

  A minimum of two KMOs (one KMO per tree) must be created for use with the eDirectory drivers. This section explains using a single KMO per tree.

  The NDS-to-NDS Driver Certificate Wizard sets up the KMOs.

5.1.2 Setting Up a KMO

To configure your Identity Vault system to handle secure Identity Manager data transfers:

1. Find out the tree name or IP address of the destination server.
2. Launch iManager and authenticate to your first tree.
3. Click to display the Identity Manager Administration page.
4. In the *Administration* list, click *NDS-to-NDS Driver Certificates* to launch the wizard.
5. At the Welcome page, enter the requested information for the first tree.

   Default values are provided by using objects in the tree that you authenticated to when you launched iManager. You must enter or confirm the following information:

   Driver DN: Specify the distinguished name of the eDirectory driver (for example, eDirectoryDriver.DriverSet1.Services.Novell).

   Tree: Verify the name of the current tree; if it is not correct, enter the correct name.

   Username: Specify the username for an account with Admin privileges in the current tree (for example, Admin).

   Password: Specify the password for the user.

   Context: Specify the user's context (for example Services.Novell).
6. Click *Next*.

   The wizard uses the information you entered to authenticate to the first tree, verify the driver DN, and verify that the driver is associated with a server.
7. Specify the requested information for the second tree:

   Driver DN: Specify the distinguished name of the eDirectory driver (for example, eDirectoryDriver.DriverSet2.Novell).

   Tree: Specify the name of the second tree.

   Username: Specify the username for an account with Admin privileges in the second tree (for example, Admin).

   Password: Specify the password for the user.

   Context: Specify the user's context (for example Users.Novell).
8. Click *Next*.

   The wizard uses the information you entered to authenticate to the second tree, verify the driver DN, and verify that the driver is associated with a server.
9. Review the information on the Summary Page, then click *Finish*.

   If KMOs already existed for these trees, the wizard deletes them and then does the following:

   - Exports the trusted root of the CA in the first tree.
   - Creates KMO objects.

- Issues a certificate signing request.
- Places certificate key pair names in the drivers' Authentication IDs (see Section A.1.3, "Authentication," on page 34).

5.2 Configuring Authentication Between Drivers

In addition to providing the mandatory certificates needed to use SSL, you can set up additional security by configuring the Subscriber channel on one eDirectory driver to authenticate to the Publisher channel on the other driver.

Set a driver object password and application password on each driver. Make sure the driver object password of the first driver matches the application password of the second driver, and that the driver object password of the second driver matches the application password of the first driver. For example:

Table 5-1 *Driver Object and Application Passwords*

	Driver Object Password	Application Password
Driver 1	Provo	Cambridge
Driver 2	Cambridge	Provo

For information about setting the passwords, see Section A.1.2, "Driver Object Password (iManager Only)," on page 34 and Section A.1.3, "Authentication," on page 34.

6 Synchronizing Passwords

To use the eDirectory driver to set up password synchronization between the two Identity Vaults, follow the instructions in the *Identity Manager 4.0.1 Password Management Guide*.

The following list contains information that is specific to setting up password synchronization with the eDirectory driver. Use it to supplement the information in the *Password Management Guide*.

- Universal Password is the standard method to synchronize passwords with Identity Manager. The eDirectory driver's policies and filters (in the basic configuration file) are set up to support this method. However, you can use the older method of synchronizing passwords through the NDS password. This method is also known as synchronizing the public key and private key. If you choose to use the NDS password method, make sure you follow the instructions in "Scenario 1: Using NDS Password to Synchronize between Two Identity Vaults" in the *Identity Manager 4.0.1 Password Management Guide*.
- If you decide to enforce password policies in multiple trees, make sure that the Advanced Password Rules in the password policies are compatible in each tree, so that password synchronization can be successful.

  If you enforce incompatible password policies in multiple eDirectory trees, and choose to reset a password back to the distribution password if it does not comply (with the option *If password does not comply, enforce Password Policy on the connected system by resetting user's password to the Distribution Password*), you could encounter a loop in which each Identity Vault server tries to change a noncompliant password.

  Information about password policies is in "Managing Passwords by Using Password Policies" in the *Password Management Administration Guide* (http://www.novell.com/documentation/password_management33/pwm_administration/data/bookinfo.html).
- The Check Password Status task in iManager does not work for a connected system if the Password policy has Universal Password enabled and does not have the setting selected for synchronizing Universal Password with NDS Password.

  The Check Password Status task lets you see whether a user's password in Identity Manager is synchronized with the password on connected systems.

  If you are using the eDirectory driver and the password policy for a user specifies in the *Configuration Options* tab that the NDS Password should not be updated when the Universal Password is updated, then the Check Password Status task for that user always shows that the password is not synchronized. The password status is shown as not synchronized, even if the Identity Manager Distribution Password and the Universal Password on the connected system are in fact the same.

  This is because the Identity Vault check-password functionality is checking the NDS Password at this time, instead of going through NMAS to refer to the Universal Password.

  By default, the NDS Password is updated when the Universal Password is updated in the password policy. If you select this option, Check Password Status should be accurate for the connected system.

7 Managing the Driver

As you work with the eDirectory 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 *Identity Manager 4.0.1 Common Driver Administration Guide*.

8 Troubleshooting

Refer to the following sections for information about troubleshooting problems you might encounter with the eDirectory driver:

- Section 8.1, "Troubleshooting Driver Processes," on page 31
- Section 8.2, "Synchronizing eDirectory Objects in a Linux High Availability Setup," on page 31

8.1 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 *Identity Manager 4.0.1 Common Driver Administration Guide*.

8.2 Synchronizing eDirectory Objects in a Linux High Availability Setup

To start the user synchronization immediately after a failover in the Linux High Availability cluster, change the eDirectory driver configuration:

1. Set the *Receive timeout in minutes* option in Publisher options to a smaller value.
2. Delete the port number from the *Authentication Context* and specify two different ports in the Subscriber and Publisher settings.
3. In the Subscriber settings, go to the *Advanced options*, select the *Socket local bind* option and specify the IP address in the *Local bind address for the subscriber socket* option.

   This is the IP address where eDirectory is listening. You must specify the IP address if there are multiple IP addresses in the high availability setup.
4. Specify the same IP address that you specified in the *Local bind address for the subscriber socket* option in the Publisher settings.

A Driver Properties

This section provides information about the Driver Configuration and Global Configuration Values properties for the eDirectory driver. These are the only unique properties for the eDirectory driver. 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 *Identity Manager 4.0.1 Common Driver Administration Guide* for information about the common properties.

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

- Section A.1, "Driver Configuration," on page 33
- Section A.2, "Global Configuration Values," on page 37

A.1 Driver Configuration

In iManager:

1 Click to display the Identity Manager Administration page.

2 Open the driver set that contains the driver whose properties you want to edit:

2a In the *Administration* list, click *Identity Manager Overview*.

2b 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.

2c Click the driver set to open the Driver Set Overview page.

3 Locate the driver icon, then click the upper right corner of the driver icon to display the *Actions* menu.

4 Click *Edit Properties* to display the driver's properties page.

In Designer:

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

The Driver Configuration options are divided into the following sections:

- Section A.1.1, "Driver Module," on page 34
- Section A.1.2, "Driver Object Password (iManager Only)," on page 34
- Section A.1.3, "Authentication," on page 34
- Section A.1.4, "Startup Option," on page 35
- Section A.1.5, "Driver Parameters," on page 36
- Section A.1.6, "ECMAScript," on page 37
- Section A.1.7, "Global Configurations," on page 37

A.1.1 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.

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

Connect to Remote Loader: The Remote Loader is not used with the eDirectory driver. However, Designer includes two suboptions, one (Driver Object Password) of which is required to set up authentication between two eDirectory drivers. If you use a driver object password, you need to select the *Connect to Remote Loader* option, set the password, click *Apply* to save the password, then select the *Java* option again.

- **Remote Loader Client Configuration for Documentation:** This option is not used with the eDirectory driver.
- **Driver Object Password:** Specifies a password for the eDirectory driver. This password must match the "Application Password:" on page 35 set for the destination eDirectory driver.

A.1.2 Driver Object Password (iManager Only)

The driver object password is used to enable the eDirectory driver's Subscriber channel to authenticate to the Publisher channel of the destination eDirectory driver. This authentication, although it is optional, provides an extra layer of security between the two drivers.

In Designer, this setting is located under the Connect to Remote Loader: option.

For additional information about setting up authentication between the two drivers, see Chapter 5, "Securing Driver Communication," on page 23.

Driver Object Password: Specifies a password for the eDirectory driver. This password must match the "Application Password:" on page 35 set for the destination eDirectory driver.

A.1.3 Authentication

The Authentication section stores the information required to authenticate to the connected system. For the eDirectory driver, it stores the information required to authenticate to the connected eDirectory driver and tree.

Authentication information for server: Displays or specifies the server that the driver is associated with.

Authentication ID: This ID is used by the driver to authenticate to the destination eDirectory driver. The ID is automatically generated and stored in this field when you run the NDS-to-NDS Driver Certificates Wizard. Authentication ID is used for establishing the secure connection. Format of the data in the Authentication ID field is the name of the local KMO object to use for the connection. This KMO object should be present in the local directory tree. For example, if eDirectory driver is configured between Server1 and Server2:

- Authentication ID on server1 is : eDirectory(Server1)
- Authentication ID on Server2 is : eDirectory(Server2)

  Both KMOs eDirectory(Server1) and eDirectory(Server2) should be signed by the same certificate authority (CA).

For information, see Chapter 5, “Securing Driver Communication,” on page 23.

Authentication Context: Specify the hostname or IP address of the destination server as well as the decimal port number (for example, 187.168.1.1:8196).

You can specify a separate port for Subscriber and Publisher channels by specifying a second port number following a second colon. If a second port number is specified, the Publisher channel uses the second port number rather than using the same port number as the Subscriber channel (for example, 255.255.255.255:2000:2001).

If your server has multiple IP addresses, you can specify the IP address you want the Publisher channel to use. This requires specifying the remote IP address, the Subscriber channel port, the local IP address, and the Publisher channel port. For example. 137.65.134.81:2000:137.65.134.83:2000 specifies that the Subscriber channel communicates with the remote tree on 137.65.134.81, port 2000, and that the Publisher channel listens on 137:65.134.83, port 2000.

If you see `java.net.ConnectException: Connection Refused`, no port connection is available in the other eDirectory tree. This error might be caused by one of the following:

- The driver in the other eDirectory tree is not running.
- The driver is running but is configured to use a different port.

Remote Loader Connection Parameters: The eDirectory driver does not support the use of the Remote Loader. These options do not apply.

Application Password: The application password, when used in conjuction with the driver object password, enables the eDirectory driver’s Subscriber channel to authenticate to the Publisher channel of the destination eDirectory driver. This authentication, although it is optional, provides an extra layer of security between the two drivers.

This password be the same as the driver object password for the destination eDirectory driver.

For more information, see Chapter 5, “Securing Driver Communication,” on page 23.

Remote Loader Password: The eDirectory driver does not support the use of the Remote Loader. These options do not apply.

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.

A.1.4 Startup Option

The Startup Option section enables 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 iManager.

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.

A.1.5 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 the following categories:

- "Driver Options" on page 36
- "Subscriber Options" on page 36
- "Publisher Options" on page 36

Driver Options

SSL type: Specifies whether to use a Key Material Object (KMO) for SSL or use a Java keystore file. For more information, click the *Information* icon.

Subscriber Options

Address or host name of remote publisher: Specifies the IP address or DNS name of the server hosting the remote eDirectory driver that the local subscriber connects to.

TCP port of remote publisher: If the remote publisher options specify a TCP port, this must be set to *specify* and the value from the remote Publisher channel entered into the *Port number* field. (These two fields must match what is set in the remote Publisher channel's options, which have corresponding fields).

Port number: Specifies the port number that the remote publisher is configured to run on. Displays only if you select *specify* in the *TCP port of remote publisher* field.

Advanced options: Displays additional fields when you select *show*.

Socket local bind: The *local bind* fields specify which IP address the Subscriber channel's socket will be bound to. This is generally only useful if the server has more than one IP address and it is important to bind to a particular address because of firewall settings.

Local bind address for subscriber socket: The *local bind* fields specify which IP address the Subscriber channel's socket will be bound to. This is generally only useful if the server has more than one IP address and it is important to bind to a particular address because of firewall settings.

Receive timeout in minutes: In order to detect a lost TCP/IP connection, the eDir-to-eDir driver periodically sends small packets. This value determines how long after entering a receive-wait condition the Subscriber channel waits until sending a keep-alive packet to determine if the TCP/IP connection has been lost. Generally, do not change this value except under instruction from Novell.

The default value for the Subscriber channel is one minute.

Publisher Options

Publisher heartbeat interval: Specifies how often you want the driver to send a status message along the Publisher channel when there has not been any traffic during the interval time.

Local bind address for publisher socket: Specifies which IP address the Subscriber channel's socket will be bound to. This is generally only useful if the server has more than one IP address and it is important to bind to a particular address because of firewall settings. This setting applies to the local publisher's "server" socket on which the local publisher listens for connections from the remote Subscriber channel.

Receive timeout in minutes: In order to detect a lost TCP/IP connection, the eDirectory driver periodically sends small packets. This value determines how long after entering a receive-wait condition the Publisher channel waits until sending a keep-alive packet to determine if the TCP/IP connection has been lost. Generally, do not change this value except under instruction from Novell.

The default value for the Publisher channel is ten minutes.

A.1.6 ECMAScript

Enables you to add ECMAScript resource files. The resources extend the driver's functionality when Identity Manager starts the driver.

A.1.7 Global Configurations

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.

A.2 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 eDirectory driver includes several GCVs that are created from information supplied during importing the driver configuration file (see Chapter 3, "Creating a New Driver," on page 13) and one that is not.

The driver also includes the GCVs that are used with password synchronization. In Designer, you must click the icon next to a password synchronization GCV to edit it. This displays the Password Synchronization Options dialog box that has a better view of the relationship between the different settings. In iManager, you should edit the password synchronization settings on the *Server Variables* tab rather than under the GCVs. The Server Variables page has a better view of the relationship between the different GCVs.

You can add your own GCVs if you discover you need additional ones as you implement policies in the driver.

To access the driver's GCVs in iManager:

1 Click to display the Identity Manager Administration page.

2 Open the driver set that contains the driver whose properties you want to edit.

2a In the *Administration* list, click *Identity Manager Overview*.

2b 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.

2c Click the driver set to open the Driver Set Overview page.

3 Locate the driver icon, click the upper right corner of the driver icon to display the *Actions* menu, then click *Edit Properties*.

or

To add a GCV to the driver set, click *Driver Set*, then click *Edit Driver Set properties*.

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 divided into categories:

- Section A.2.1, “Default Configuration,” on page 38
- Section A.2.2, “Password Synchronization,” on page 39
- Section A.2.3, “Account Tracking,” on page 39
- Section A.2.4, “Managed System Information,” on page 40

A.2.1 Default Configuration

The following GCVs define control the default configuration of the eDirectory driver:

eDirectory Publisher Placement type: Controls how the objects are placed in the remote Identity Vault and the local Identity Vault. The options are:

- **Mirrored:** Mirrors the structure between the remote Identity Vault and the local Identity Vault.

  If you choose this option, use the same option for configuring both eDirectory trees you are synchronizing.

  This option in the driver configuration synchronizes User, Group, Organization, Country, and Organizational Unit objects. It also mirrors the structure of a subtree in the other tree.
- **Flat:** All of the objects are placed into a single container.

  This option synchronizes User and Group objects and places all users in one container and all groups in another container.

  This option is typically used in conjunction with the Department option (or a similar configuration) in the other tree.

  This option doesn’t create the containers that hold the users and groups. You must create those manually.
- **Department:** Users are placed in containers named after the department.

  This option synchronizes User and Group objects and places all users and groups in a container based on the *Department* field in your management console.

  This configuration is typically used in conjunction with the Flat option (or a similar configuration) in the other tree.

  This option doesn’t create the containers for each department. You must create those manually. They must be the same as the container specified during import.

Remote Tree Base User Container: Specify the source container of the user objects in the remote Identity Vault.

Remote Tree Base Groups Container: Specify the source container of the group objects in the remote Identity Vault.

A.2.2 Password Synchronization

The following GCVs control password synchronization for the eDirectory driver. For more information, see the *Identity Manager 4.0.1 Password Management Guide*.

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 iManager, you should edit the Password Management Options on the *Server Variables* tab rather than under the GCVs. The Server Variables page has a better view of the relationship between the different GCVs.

Connected System Name or Driver Name: Specify the name of the driver. The e-mail notification template uses this value to identify 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, attempt 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, notify the user by e-mail of any password synchronization failures.

A.2.3 Account Tracking

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

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.

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

Identifiers: Add the account identifier attributes. Attribute 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 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 the policies assume when an object is published to the Identity Vault and the status attribute is not set in the application.

A.2.4 Managed System Information

These settings help the Identity Reporting Module function to generate reports. There are different sections in the *Managed System Information* tab.

- "General Information" on page 40
- "System Ownership" on page 40
- "System Classification" on page 40
- "Connection and Miscellaneous Information" on page 41

General Information

Name: Specify a descriptive name for this Identity Vault. This name is displayed in the reports.

Description: Specify a brief description of this Identity Vault. This description is displayed in the reports.

Location: Specify the physical location of this Identity Vault. This location is displayed in the reports.

Vendor: Select Novell as the vendor of the Identity Vault. This information is displayed in the reports.

Version: Specify the version of this Identity Vault. This version information is displayed in the reports.

System Ownership

Business Owner: Browse to and select the business owner in the Identity Vault for this Identity Vault. You must select a user object, not a role, group, or container.

Application Owner: Browse to and select the application owner in the Identity Vault for this Identity Vault. You must select a user object, not a role, group, or container.

System Classification

Classification: Select the classification of the Identity Vault. This information is displayed in the reports. The options are:

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

  If you select *Other*, you must specify a custom classification for the Identity Vault.

Environment: Select the type of environment the Identity Vault provides. The options are:

- Development
- Test

- Staging
- Production
- Other

  If you select *Other*, you must specify a custom classification for the Identity Vault.

Connection and Miscellaneous Information

Connection and miscellaneous information: This options is always set to *hide*, so that you don't make changes to these options. These options are system options that are necessary for reporting to work. If you make any changes, reporting stops working.

B Synchronized Attributes

The filter for the basic driver configuration synchronizes the following attributes:

Table B-1 *eDirectory Driver Attributes That Are Synchronized*

- accessCardNumber
- ACL
- assistant
- assistantPhone
- businessCategory
- city
- CN
- co
- company
- costCenter
- costCenterDescription
- departmentNumber
- Description
- destinationIndicator
- directReports
- EMail Address
- employeeStatus
- employeeType
- Equivalent To Me
- Facsimile Telephone Number
- Full Name
- Generational Qualifier
- Given Name
- Group Membership
- GUID
- Higher Privileges
- Initials
- instantMessagingID
- internationaliSDNNumber
- Internet EMail Address
- jackNumber
- jobCode
- L
- Language
- Mailbox ID
- Mailbox Location
- mailstop
- manager
- managerWorkforceID
- mobile
- NSCP:employeeNumber
- otherPhoneNumber
- O
- OU
- pager
- personalTitle
- photo
- Physical Delivery Office Name
- Postal Address
- Postal Code
- Postal Office Box
- preferredDeliveryMethod
- preferredName
- Private Key
- Public Key
- registeredAddress
- roomNumber
- S
- SA
- Security Equals
- Security Flags
- See Also
- siteLocation
- Surname
- Telephone Number
- teletexTerminalIdentifier
- telexNumber
- Timezone
- Title
- tollFreePhoneNumber
- UID
- uniqueID
- vehicleInformation
- workforceID
- x121Address
- x500UniqueIdentifier