AUTHORIZED DOCUMENTATION

Driver for User Management of SAP* Software Implementation Guide

Novell® Identity Manager

3.6

July 23, 2008

www.novell.com

Legal Notices

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

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

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

Novell Trademarks

Third-Party Materials

Contents

About This Guide

This manual is for Novell® Identity Manager administrators, SAP developers and administrators, and others who implement the Identity Manager 3.6 Driver for User Management of SAP Software.

The guide contains the following sections:

Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with Novell Identity Manager. Please use the User Comments feature at the bottom of each page of the online documentation, or go to http://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 3.6 Drivers Documentation Web site (http://www.novell.com/documentation/idm36drivers).

Additional Documentation

For documentation on using Identity Manager and the other drivers, see the Identity Manager 3.6 Documentation Web site (http://www.novell.com/documentation/idm36).

Documentation Conventions

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

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

Overview

1

The Identity Manager Driver for User Management of SAP Software, subsequently referred to as the SAP User driver, creates an automated link between the Identity Vault and SAP User Management systems (BASIS or Web Application Server.) This technology enables data flow within a business enterprise based on its own unique requirements, and eliminates the labor-intensive and error-prone practice of re-entering the same data into multiple databases. As User object records are added, modified, deactivated (disabled), or deleted in SAP or the Identity Vault, network tasks associated with these events can be processed automatically.

The driver allows administrators to propagate User data between SAP systems and other business applications and databases without the need for custom integration solutions. Administrators can decide what data will be shared and how data will be presented within their enterprises.

In this section:

- Section 1.1, "Supported SAP Versions," on page 11
- Section 1.2, "Driver Concepts," on page 11
- Section 1.3, "Driver Components," on page 16
- Section 1.4, "Support for Standard Driver Features," on page 16

1.1 Supported SAP Versions

The driver supports the following SAP versions:

- SAP R/3 version 4.5B or higher
- mySAP

1.2 Driver Concepts

The driver is a bidirectional synchronization product between SAP R/3 and Enterprise R/3 systems and the Identity Vault. This framework uses XML and XSLT to provide data and event transformation capabilities that convert Identity Vault data and events into SAP data and vice-versa.

The Identity Vault acts as a hub, with other 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.

- Section 1.2.1, "Publisher Channel," on page 12
- Section 1.2.2, "Subscriber Channel," on page 14
- Section 1.2.3, "Attribute Mapping from the SAP User Management Database to the Identity Vault," on page 14
- Section 1.2.4, "Associations," on page 15

1.2.1 Publisher Channel

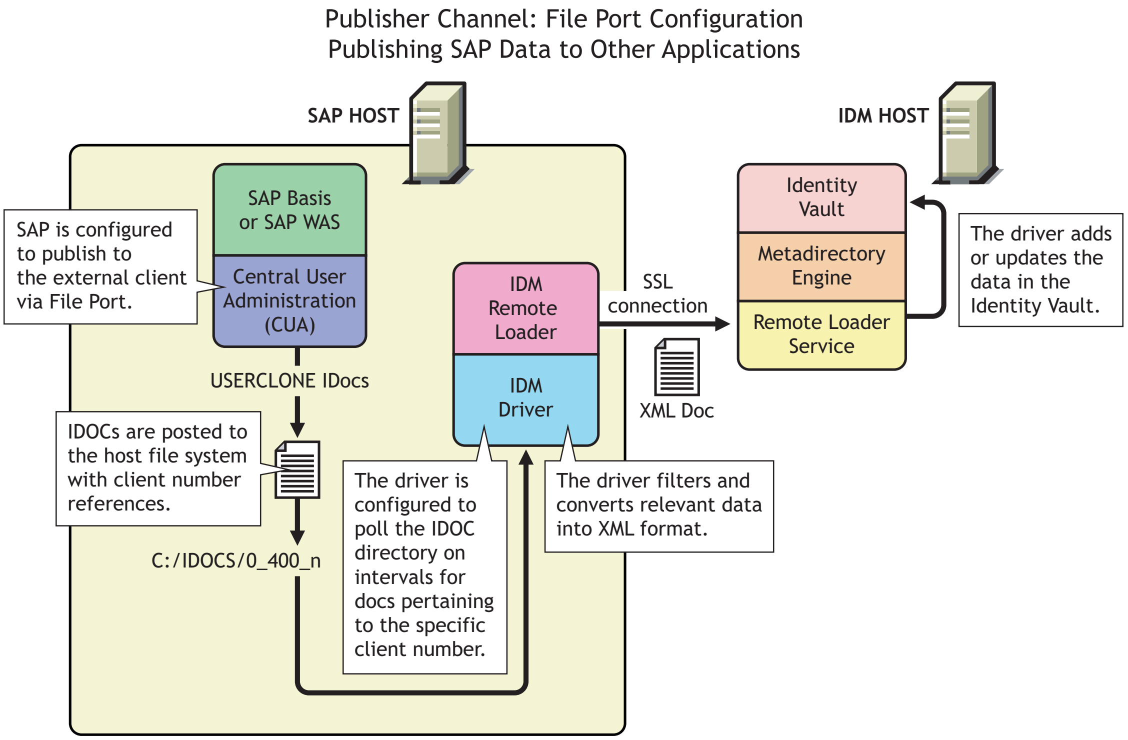
The SAP system publishes User object information in the form of USERCLONE IDocs using Application Link Enabling (ALE) and Central User Administration (CUA) technology. If desired and properly configured, the SAP system can propagate all Add, Delete, Lock, Unlock, and Modify User event data to the Identity Vault. The driver consumes the IDoc data and converts it into XML format. For more information on how the driver handles IDoc processing, refer to "IDoc Consumption by the Driver" on page 13.

The Publisher channel then submits XML-formatted documents to the Metadirectory engine for publication into the Identity Vault. By using Identity Manager and other Identity Manager drivers, the data can be shared with other business applications and directories. These other applications can add additional data, which in turn can be transferred back into the SAP User records using the standard SAP Business Application Programming Interface (BAPI).

Depending on the ALE port configuration you choose, the Publisher channel either polls the SAP database for changes via a file port or it receives the data via a TRFC connection.

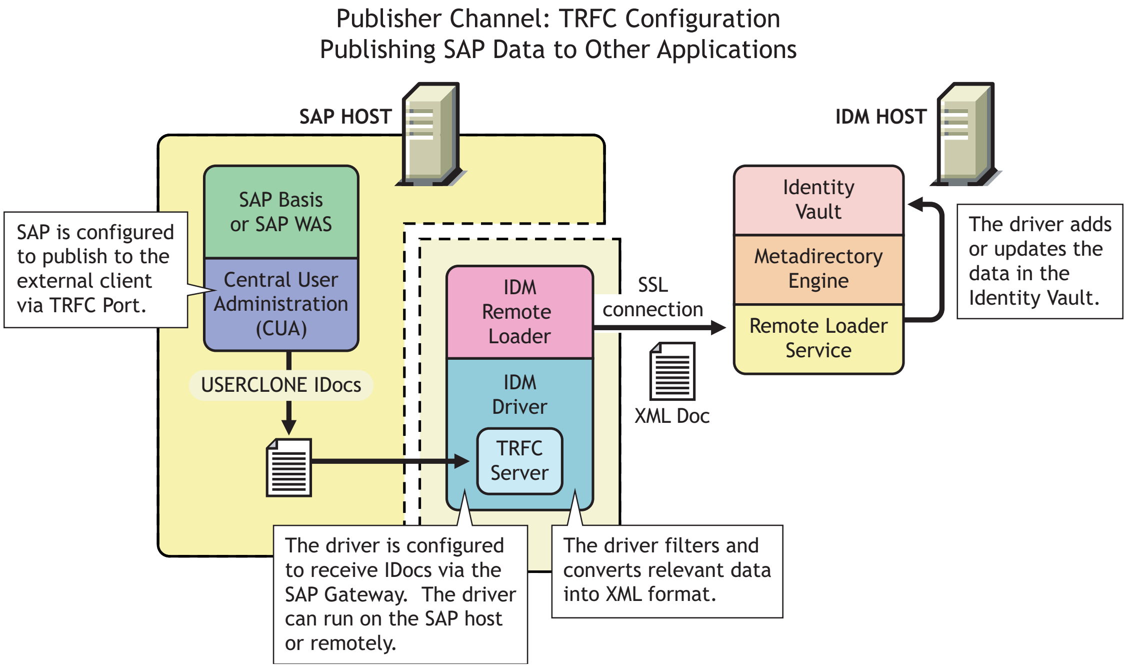
The following diagram illustrates the file port configuration. With the file port configuration, the entire IDoc is stored on the SAP host system.

Figure 1-1 *Publishing Data to the Identity Vault using the File Port Configuration*


Publisher Channel: File Port Configuration
Publishing SAP Data to Other Applications
SAP HOST
IDM HOST
SAP Basis or SAP WAS
Central User Administration (CUA)
SAP is configured to publish to the external client via File Port.
USERCLONE IDocs
IDOCs are posted to the host file system with client number references.
C:/IDOCS/0_400_n
IDM Remote Loader
IDM Driver
The driver is configured to poll the IDOC directory on intervals for docs pertaining to the specific client number.
SSL connection
XML Doc
The driver filters and converts relevant data into XML format.
Identity Vault
Metadirectory Engine
Remote Loader Service
The driver adds or updates the data in the Identity Vault.

The following diagram illustrates the TRFC port configuration. When using the TRFC configuration, a minimal "trigger" IDoc is stored on the driver host system. The driver handles the parsing of the IDoc data and uses the information to read the current User object. The driver then parses the appropriate data fields specified by the driver configuration, and provides secure transport of the data to the Identity Vault. Only data elements specifically selected by the system administrator are transported from the SAP host system to the Identity Vault.

Figure 1-2 Publishing Data to the Identity Vault using the TRFC Configuration


Publisher Channel: TRFC Configuration
Publishing SAP Data to Other Applications
SAP HOST
IDM HOST
SAP is configured to publish to the external client via TRFC Port.
SAP Basis or SAP WAS
Central User Administration (CUA)
USERCLONE IDocs
IDM Remote Loader
IDM Driver
TRFC Server
SSL connection
XML Doc
Identity Vault
Metadirectory Engine
Remote Loader Service
The driver adds or updates the data in the Identity Vault.
The driver is configured to receive IDocs via the SAP Gateway. The driver can run on the SAP host or remotely.
The driver filters and converts relevant data into XML format.

IDoc Consumption by the Driver

The driver consumes only Output IDoc files with the client number that is specified by the driver configuration, thus ensuring the privacy of other IDocs that might be generated by another driver configuration or ALE integration. Only the IDoc attributes that have been specified in the driver Publisher filter are published to the Identity Vault.

The format of a successfully published IDoc file is:

```
<(I)nput or (O)utput>_<client number>_<consecutive IDoc number>
```

For example:

```
O_300_0000000000001001
```

After the IDoc has been processed and specified attributes have been published, the filename of the IDoc file is modified to reflect the status of the publication processes. The following table lists the IDoc status and corresponding extension:

IDoc Status	Filename Extension
Processing but not published	.proc
Processed successfully and published	.done
Processed with an error or warning	.fail or .warn
Processed and retained for future-dated processing	.futr
Processed with corrupt or illegitimate data	.bad

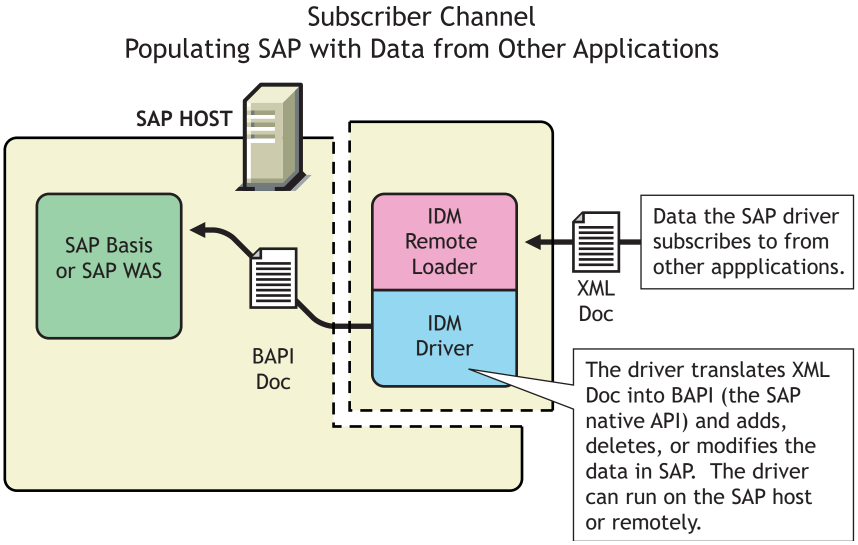
You should determine what action is required, if any, after IDoc publication is complete.

NOTE: Removing the filename extension makes the IDoc available for re-processing.

1.2.2 Subscriber Channel

The Subscriber channel receives XML-formatted Identity Vault events from the Metadirectory engine. The driver converts these documents to an appropriate data format, and updates SAP via the BAPI interface. The Identity Vault sends changes only to the applications that subscribe to receive them.

Figure 1-3 *Populating SAP with Data from other applications via the Subscriber channel*



Data the SAP driver subscribes to from other appplications.

The driver translates XML Doc into BAPI (the SAP native API) and adds, deletes, or modifies the data in SAP. The driver can run on the SAP host or remotely.

For data to flow from the Identity Vault to the SAP system, the driver uses the SAP BAPI functions. The level of functionality is based upon the R/3 release level. By default, the driver is configured to support a SAP 4.6C system using USERCLONE03 messages. (To determine the level of USERCLONE messages available on your SAP system, run transaction WE60 and specify object name USERCLONEnn.) As a SAP administrator, you can select which attributes from the infotypes can be modified.

1.2.3 Attribute Mapping from the SAP User Management Database to the Identity Vault

Schema mapping is used by Identity Manager to translate data elements as they flow between the SAP User Management database and the Identity Vault. The SAP User object schema is based on the SAP USERCLONE message type. The schema map contains all attributes of the various data infotypes of the USERCLONE message type.

Several of the USERCLONE infotypes can be instantiated multiple times on the User records. Infotypes such as ADDTEL (Telephone Number) and ACTIVITYGROUPS (Roles) are *Table* fields and can contain multiple values. Other infotypes such as ADDRESS and LOGONDATA are *Structure* fields and are instantiated only once but have multiple fields associated with them. Still other fields are *simple* field types that contain only a single data field element.

The Identity Vault (eDirectory) system administrator can configure the driver to receive any of these various data fields, and can also configure the driver to handle the data in multiple ways. The Schema Map represents the data elements that can be synchronized in the SAP system.

The map elements have the following format:

```
<Table or Structure Name>:<Field>   // Field
```

or

```
<Table Name>   // Map to entire table or structure
```

Below are a few examples of maps between SAP User attributes and Identity Vault attributes.

Identity Vault Attribute	SAP User Attribute
Given Name	ADDRESS:FIRSTNAME
Surname	ADDRESS:LASTNAME
sapRoles	ACTIVITYGROUPS:AGR_NAME
buildingName	ADDRESS:BUILDING_P
floor	ADDRESS:FLOOR_P
Internet EMail Address	ADDSMTP:E_MAIL
OU	ADDRESS:DEPARTMENT
Pager	ADDPAG:PAGER
sapAlias	ALIAS:USERALIAS
DirXML-sapLocRoles	LOCACTIVITYGROUPS

The driver can synchronize multiple-instance data (such as TELEPHONE), but it cannot guarantee the specification of a primary value. It is also possible to specify only the Table name in a schema mapping. This is useful if you want to synchronize all data fields in a Table to the Identity Vault. You must use policies to parse desired fields from the Table data. Refer to Appendix E, “Example XML Document Received from the Driver,” on page 97 to see how various formats are represented in modify events.

1.2.4 Associations

Associations are created between SAP and Identity Vault objects during the synchronization process. For the SAP User object, a unique 12-character name (per client) must be created. However, the Identity Vault and other applications do not need to share this same unique ID. Identity Manager allows the various naming policies in an organization to be applied to objects by using the DirXML-Association attribute.

The DirXML-Association attribute is multivalued. Therefore, if Identity Manager is being used to synchronize an object among multiple applications, all of the object’s unique IDs (or associations) can be stored in this attribute on the Identity Vault object.

The unique ID association links objects in SAP to their objects in the Identity Vault. When an Add or Matching event occurs, the association is made. This association allows the driver to perform subsequent tasks on the appropriate object.

The DirXML-Associations field is stored on the Identity Vault object on the Identity Manager property page.

1.3 Driver Components

This sections contains information about the following driver components.

- "Driver Configurations" on page 16
- "Driver Shim" on page 16
- "SAP User Java Connector Test Utility" on page 16

1.3.1 Driver Configurations

After you install Identity Manager and the driver, you create one or more Driver objects. Each Driver object represents an instance of the SAP User driver. The driver configuration file gets you up and running with a minimum of customization by letting you create a Driver object with preconfigured policies, filters, and driver parameters.

The driver configuration file is named `SAPUser-IDM3_6_0-V2.xml`.

1.3.2 Driver Shim

The driver shim handles communication between the SAP User database and the Metadirectory engine.

1.3.3 SAP User Java Connector Test Utility

In order to use the driver, you must download the SAP JCO and install it. The SAP User Java* Connector (JCO) Test utility enables you to check for JCO installation and configuration issues prior to configuring the driver. You can use the JCO test utility to validate correct installation of the JCO client and configuration issues prior to configuring the driver.

You can use the JCO test utility to validate correct installation of the JCO client and connectivity to the SAP host system, as well as testing for accessibility of the User Management BAPIs used by the driver. For more information, refer to Chapter 4, "Testing the SAP JCO Client Connection," on page 27.

1.4 Support for Standard Driver Features

The following sections provide information about how the SAP User driver supports these standard driver features:

- Section 1.4.1, "Local Platforms," on page 17
- Section 1.4.2, "Remote Platforms," on page 17
- Section 1.4.3, "Entitlements," on page 17

1.4.1 Local Platforms

A local installation is an installation of the driver on the same server as the Metadirectory engine, Identity Vault, and SAP User application. Both systems that the driver needs to communicate with (Metadirectory engine and SAP User application) are local to the driver.

The SAP User driver can be installed on the same operating systems supported by the Metadirectory server. For information about the operating systems supported by the Metadirectory server, see "Metadirectory Server" in "System Requirements" in the *Identity Manager 3.6 Installation Guide*.

1.4.2 Remote Platforms

The SAP User driver must reside on the same server as the SAP User application. If you don't want to install the Metadirectory engine and Identity Vault (eDirectory) on the SAP server, you can use the Remote Loader service to run the driver on the SAP server while having the Metadirectory engine and Identity Vault on another server.

The SAP User driver can be installed on the same operating systems supported by the Remote Loader. For information about the operating systems supported by the Remote Loader, see "Remote Loader" in "System Requirements" in the *Identity Manager 3.6 Installation Guide*.

1.4.3 Entitlements

The SAP User driver does not have entitlement functionality defined with the default configuration file. The driver does support entitlements, if there are policies created for the driver to consume.

Installing the Driver Files

2

By default, the SAP User driver files are installed on the Metadirectory server at the same time as the Metadirectory 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 5, "Creating a New Driver," on page 35) or upgrade an existing driver's configuration (see Chapter 6, "Upgrading an Existing Driver," on page 43).

The SAP User driver must be located on the same server as the SAP User application. If the driver is not on that server, you have the following options:

- Install the Metadirectory server (Metadirectory engine and drivers) to the SAP server. This requires eDirectory to be installed on the server. See the instructions in "Installing the Metadirectory Server" in the *Identity Manager 3.6 Installation Guide*.
- Install the Remote Loader (required to run the driver on a non-Metadirectory server) and the SAP User driver files to the SAP server. This assumes that you already have a Metadirectory server installed on another server in your environment. See "Installing the Remote Loader" in the *Identity Manager 3.6 Installation Guide*.

As part of the installation, select the Utilities option and install the SAP Utilities. This installs the SAP Java Connector Test utility that you can use to ensure that the driver has connection to the SAP system. If you've already installed the driver files but did not install the SAP Utilities, you can run the installation program again to install only the SAP Utilities.

Installing the SAP Java Connector Client

The server where the SAP driver is installed must have the SAP Java Connector (JCO) client technology version 1.1*x* or 2.*x* to provide the driver with connectivity to the SAP system.

This JCO client is available to SAP customers and developer partners through SAP, and is provided for most popular server operating systems. You can download the JCO from the SAP Connectors site (http://service.sap.com/connectors).

Configuring the SAP System

3

You must configure the SAP system parameters to enable Application Link Enabling (ALE) and Central User Administration (CUA) processing of USERCLONE IDocs if you want to publish real-time changes of SAP User data to the Identity Vault. Before you continue, make sure you have sufficient rights to configure the distribution model and to distribute user data via ALE.

Novell® follows SAP's general guidelines for configuring BAPI (Business Application and Programming Interface) and ALE technologies for this integration solution. For information about BAPI, see Appendix C, "Business Application Programming Interfaces (BAPIs)," on page 87. For information about ALE, see Appendix B, "Application Link Enabling (ALE)," on page 83.

Complete the steps in the following sections in the order listed. The instructions are for SAP version 4.6C. If you are using a previous version of SAP, the configuration process is the same; however, the SAP interface will be different.

- Section 3.1, "Defining Sending and Receiving Systems," on page 21
- Section 3.2, "Creating a Distribution Model," on page 22
- Section 3.3, "Creating a Port Definition," on page 23
- Section 3.4, "Generating Partner Profiles," on page 24
- Section 3.5, "Activating Central User Administration," on page 26
- Section 3.6, "Creating a Communication (CPIC) User," on page 26

3.1 Defining Sending and Receiving Systems

The sending and receiving systems must be defined for messaging. In order to distribute data between systems you must first define both the sending and receiving systems as unique logical systems.

For this particular solution, we recommend defining two logical systems. One logical system represents the driver and acts as the *receiver* system. The other logical system represents the SAP system and acts as the *sender* system. Because only one of these clients is used as a data source (that is, the client/logical system where SAP User data is stored and "actions" occur), there is no need to assign a client to the receiving logical system.

NOTE: Depending on your current SAP environment, you might not need to create a logical system. You might only need to modify an existing Distribution Model by adding the USERCLONE message type to a previously configured Model View. For more information, see "Creating a Distribution Model" on page 22.

It is important, however, that you follow SAP's recommendations for logical systems and configuring your ALE network. The following instructions assume that you are creating new logical systems and a new model view.

3.1.1 Creating a Logical System

1 In SAP, type transaction code `BD54`.

2. Click *New Entries*.
3. Type an easily identifiable name to represent the SAP *sender* system. SAP recommends the following format for logical systems representing R/3 clients: *systemID*CLNT*client number* (such as ADMCLNT100).
4. Type a description for the logical system (such as Central System for SAP User Distribution).
5. Add a second logical system name to represent the Identity Manager external *receiver* system (such as DRVCLNT100).
6. Type a description for the logical system (such as Identity Manager User Management Integration).
7. Save your entries.

3.1.2 Assigning a Client to the Logical System

1. In SAP, type transaction code `SCC4`.
2. Click *Table View* > *Display* > Change to switch from display to change mode.
3. Select the client from which you want User information distributed (such as 100).
4. Click *Goto* > *Details* > *Client Details*.
5. In the Logical System field, browse to the *sender* logical system you want to assign to this client (such as ADMCLNT100).
6. Save your entry.

3.2 Creating a Distribution Model

The distribution model contains essential information about message flow. The model view defines the systems that will communicate with each other and the messages that will flow between them. The distribution model forms the basis of distribution and controls it directly.

To create a distribution model:

1. Verify that you are logged on to the sending system/client.
2. In SAP, type transaction code `BD64`. Ensure that you are in Change mode (click *Table View* > *Display* > *Change*.)
3. Click *Edit* > *Model View* > *Create*.
4. Type the short text to describe the distribution model (such as Client 100 Distribution to Identity Manager).
5. Type the technical name for the model (such as SAP2IDM).
6. Accept the default Start and End dates or specify valid values. Click the check mark icon to save your entry.
7. Select the view you created, then click *Add BAPI*.
8. In the Sender/Client field, type the name of the *sender* logical system (such as ADMCLNT100).
9. In the Receiver/Client field, add the name of the *receiver* logical system (such as DRVCLNT100).
10. In the *Obj. Name/Interface* field, add the USER object name.

NOTE: Ensure that you add the USER object name with all capital letters.

11. In the *Method* field, add Clone.
12. Click the check mark icon to save the BAPI.
13. Select the SAP2IDM model view.
14. Click *Add BAPI*.
15. Define the sender (logical system ADMCLNT100).
16. Define the receiver (logical system DRVCLNT100).
17. In the *Obj. Name/Interface* field, add the UserCompany object name.
18. In the *Method* field, add Clone.
19. Click the check mark icon to save your BAPI entries.
20. Save the Distribution Model entries.

3.3 Creating a Port Definition

The port is the communication channel to which IDocs are sent. The port describes the technical link between the sending and receiving systems.

The driver can be configured to support a connection via a TRFC port or to consume IDocs distributed via a File port. The default driver configuration assumes that you use the TRFC port configuration.

- Section 3.3.1, “Creating a TRFC Port Definition,” on page 23
- Section 3.3.2, “Creating a File Port Definition,” on page 24

3.3.1 Creating a TRFC Port Definition

Complete the following two tasks to create a TRFC port definition:

- “Creating the RFC Destination” on page 23
- “Creating the TRFC Port Definition” on page 24

Creating the RFC Destination

If you are distributing data to multiple drivers, each driver must have a unique RFC destination and program ID.

1. In SAP, type transaction code `SM59`.
2. Click the *Create* icon.
3. Name the RFC destination (use the driver’s logical system name, for example, DRVCLNT100.)
4. Select *T* as the connection type (for a TCP/IP connection.)
5. Add a description for the destination (such as JCO Server in IDM User Driver.)
6. Save your entry.
7. Select the option for *Registration* or *Registered Server Program*. Type the program ID to be used for the driver. In the default driver configuration, this value is set to *IDMUser100*.

8 (Conditional) If the SAP server is configured to use a Unicode database, complete the following steps:

   8a Select the *Special Options* tab.

   8b Select *Unicode*.

9 Save your entry.

Creating the TRFC Port Definition

If you are distributing data to multiple drivers, each driver must have a unique TRFC port.

1 In SAP, type transaction code *WE21*.

2 Select *Transactional RFC*, then click the *Create* icon.

3 Select *Own Port Option Name*.

   3a Type a port name (such as IDMPORT).

   3b Type a description for the port definition (such as Port to IDM User Driver).

   3c Select a version (such as IDoc record types SAP release 4.X)

   3d Specify the RFC destination. This is the name of the RFC destination representing the driver (such as DRVCLNT100.)

4 Save your entry.

3.3.2 Creating a File Port Definition

If you are distributing data to multiple drivers, each driver must have a unique file port.

1 In SAP, type transaction code `WE21`.

2 Select *File*, then click the *Create* icon.

   2a Type a port name (such as IDMFILE).

   2b Type a port description (such as File Port to IDM User Driver).

   2c Select a version (such as SAP release 4.X).

3 Define the outbound file:

   3a Select the physical directory. This is the directory where you want IDocs placed. You might need to create this directory.

   Type the directory where the outbound files are written, for example: `\\sapdev\nov\sys\global\sapndsconnector`.

   3b Type the function module. This names the IDoc file in a specific format. Use the following: `EDI_PATH_CREATE_CLIENT_DOCNUM`.

4 Save your changes.

You do not need to configure the other three tabs for the port properties (outbound:trigger, inbound file, and status file).

3.4 Generating Partner Profiles

The system automatically generates a partner profile or you can manually maintain the profile.

NOTE: If you are using an existing distribution model and partner profile, you do not need to generate a partner profile. Instead, you can modify it to include the USERCLONE BAPI.

3.4.1 Generating a Profile

1. In SAP, type transaction code `BD82`.
2. Select the *Model View*. This should be the Model View previously created in "Creating a Distribution Model" on page 22.
3. Ensure that the *Transfer IDoc Immediately* and *Trigger Immediately* option buttons are selected.
4. Click the *Execute* icon.

   When the status screen appears, ignore any red error or warning messages related to the driver's logical system.

3.4.2 Modifying the Port Definition

The port definition might have been generated incorrectly. For your system to work properly, you might need to modify the port definition.

1. In SAP, type transaction code `WE20`.
2. Select *Partner Type LS*.
3. Select your *receiver* logical system (such as DRVCLNT100).
4. Click the *Create Outbound Parameter* icon, then select message type *USERCLONE*.
5. Modify the receiver port so it is the *file* or *TRFC port name* you created earlier (such as IDMPORT or IDMFILE).
6. Under *Output Mode*, select *Transfer IDoc Immediately* to send IDocs immediately after they are created.
7. In the IDoc Type section, select the *Basic type* and the appropriate *USERCLONE*:
   - For SAP 4.5, select USERCLONE01
   - For SAP 4.6a, select USERCLONE02
   - For SAP 4.6c, select USERCLONE03
   - For SAP 6.10, select USERCLONE04
   - For SAP 6.20 or greater, select USERCLONE05
8. Save your entries.

   NOTE: The following procedures are only necessary if you want to distribute company address data.

9. Click the *Create Outbound Parameter* icon, then select message type *CCLONE*.
10. Modify the receiver port so it is the *file* or *TRFC port name* you created earlier (such as IDMPORT or IDMFILE.)
11. (Conditional) If you are using a TRFC port, modify the packet size. Select Packet Size = 1.
12. Under *Output Mode*, select *Transfer IDoc Immediately* to send IDocs immediately after they are created.

13. In the *IDoc type* section, select *Basic type* and the appropriate *CCLONE*. (For all SAP versions, select *CCLONE01*.)
14. Save your entries.

3.5 Activating Central User Administration

Central User Administration (CUA) is the process that activates the distribution model.

1. In SAP, type transaction code `SCUA`.
2. In the *Maintain System Landscape* dialog box, select the distribution *Model View* previously created (such as SAP2IDM).
3. Save your entry.

   You might see a message stating "Unable to distribute the system landscape to system IDMDRV." This is an informative message and is not an error or issue of concern.

On some versions of SAP, all systems in the distribution, including the Identity Manager driver, must be accessible during this step. If a TRFC port is being used for the driver Publisher channel, the driver should be running to ensure connectivity and completion of the CUA configuration.

3.6 Creating a Communication (CPIC) User

Users are client-independent. For each client that will be using the driver, a system user with CPIC access must be created.

1. In SAP, type transaction code `SU01`.
2. From *User Maintenance*, enter a username in the *User* dialog box (such as IDM_CPIC), then click the *Create* icon.
3. Click the *Address tab*, then type data in the last name fields (Last_IDM).
4. Click the *Logon Data tab*, then define the *initial password* and set the user type to *CPIC* (Communication).
5. Click the *Profiles* tab, then add the *S_A.CPIC profile*. The driver must also have sufficient rights to perform required operations, which might include *SAP_ALL* and *SAP_NEW* depending on your company's system security policy.

   NOTE: We recommend using the most restrictive rights possible.

6. Click the *Systems* tab. Add the *logical name* of the *sender* system (such as ADMCLNT100). This enables the CPIC user to authenticate to the client system.
7. Click *Save*.

NOTE: Initially, you can create a dialog user to test your SAP system configuration. If there are processing problems, you can analyze the dialog user in the debugger. You should also log into the SAP system once to set this user's password. After the system is tested and works properly, you should switch to a CPIC user for security measures.

Testing the SAP JCO Client Connection

4

The driver uses the SAP Java Connector (JCO) and Business Application Programming Interface (BAPI) technologies to connect to and integrate data with the Identity Vault. The SAP JCO is a SAP client that creates service connections to a SAP R/3 system. After the driver is connected to the R/3 system, it calls methods on business objects within the R/3 system via BAPI.

The SAP Java Connector Test utility enables you to check for JCO installation and configuration issues. Use the JCO Test utility to validate installation and connectivity to the SAP JCO client, as well as testing for accessibility to the BAPIs used by the driver.

Ensure that you are using JDK/JRE version 1.3.1 or later.

The following sections apply to JCO versions 1.1.*x* and 2.*x*:

In this section:

- "About the Utility" on page 27
- "Running and Evaluating the Test" on page 28
- "Understanding Test Error Messages" on page 30

4.1 About the Utility

The JCO Test utility completes the following checks:

- Ensures that the `jco.jar` or `sapjco.jar` file, which contains the exported JCO interface, is present.
- Ensures that the JCO native support libraries are properly installed.
- Ensures that connection parameters to the SAP target system are correct.
- Ensures that the authentication parameters to the SAP target system are correct.
- Ensures that the selected language code is valid.
- Ensures that the BAPIs used by the driver are present as expected for the version of the SAP target system.

4.1.1 Utility Prerequisites

Before you run the JCO Test utility, you must install the SAP JCO client for the desired platform. The JCO can only be obtained from the SAP Service Marketplace Web site (http://www.sap-ag.de/services). The download is free to any SAP software customer or development partner, but you are required to log in.

In order to configure the driver, you must first download the SAP JCO and install it. For installation instructions, refer to the documentation accompanying the SAP JCO.

Follow the installation instructions for your platform. Each installation requires you to set one or two environment variables, such as CLASSPATH for the `sapjco.jar` file location. For the UNIX* platforms, set either the LD_LIBRARY_PATH or LIBPATH variables for the location of native support libraries. Ensure that these variables are set in the shell environment to run this test and for the subsequent use of the Identity Manager Driver for User Management of SAP Software.

You must also make sure that you have your PATH environment variable set to include the path to your Java executable file. For Win32 platforms, the environment variables are set via the System configuration in the Control Panel. On UNIX systems, edit the appropriate .profile or .bash_profile to include and export these path variables.

4.1.2 Components

The JCO Test utility consists of the `UserJCOTest.class` file. The format of an execution batch or script file varies, depending on the platform on which the JCO client has been installed.

The basic content of the file includes a path to the Java executable (or just java if your PATH is appropriately configured), and the name of the `UserJCOTest.class` file. A sample UNIX script file and Win32 batch file are listed below, where `sapjco.jar` is in the executable directory of the `UserJCOTest.class` file and the batch file:

```
Win32 jcotest.bat file
java -classpath %CLASSPATH%;. UserJCOTest

Unix jcotest file
java UserJCOTest
```

You must use proper slash notation when specifying pathnames, and you must use the proper classpath delimiter for the platform. You must also remember that the name of the `sapjco.jar` file is case-sensitive on UNIX platforms and that the name of the test class, UserJCOTest, must be specified with proper case for any platform.

4.1.3 Running and Evaluating the Test

Running the Test

To run the JCO Test utility on a Win32 platform:

1. From Windows Explorer, double-click `UserJCOTest.bat`.
   or
   From a command prompt, run the `UserJCOTest.bat` script.

To run the JCO Test utility on a UNIX platform:

1. From your preferred shell, run the `userjcotest` script file.

NOTE: It is possible that when you run the test program, an error message appears before any test output is displayed. This indicates an improper installation of the JCO client components. The error messages are documented for each platform in “Understanding Test Error Messages” on page 30.

Evaluating the Test

If the JCO client is installed properly, the following output is displayed:

```
**The SAP JCO client installation has been verified to be correct.

Version of the JCO-library: version information

Input SAP Server Connection Information
----------------------------------------
```

You then receive a series of prompts for connection and authentication information. All data must be provided unless a default value, identified by [] delimiters, is provided. Failure to fill in a response value to each prompt ends the test. Enter the following when prompted:

- Application server name or IP address
- System number [00]
- Client number
- User
- User password
- Language code [EN]

The values you provide are the same values that could be used to authenticate via the SAPGUI client. Based on the validity of the input, the test either displays error messages with solution suggestions or runs to completion. At the end of the test, a status message displays. If the test indicates full functionality as required by the driver, the following status message appears (it describes valid values that can be used as the configuration parameters for the driver):

```
**All expected platform support is verified correct.

JCO Test Summary
----------------
Full JCO/BAPI Functionality has been verified.
The following parameters may be used for driver configuration

Authentication ID: Username
Authentication Context: SAP Host Name/IP Address
Application Password: User password
SAP System Number: System Number
SAP User Client Number: Client Number
SAP User Language: Language Code
```

If the test indicates that the functionality required by the driver is not available, the following status message is displayed:

```
**There are <number> required BAPI functions NOT supported on this platform.

JCO Test Summary
----------------
JCO/BAPI functionality issues have been detected that will prevent proper driver
functionality.
```

Post-Test Procedures

After the JCO Test utility has successfully passed all tests, you can then begin to configure the driver. Make sure that the `sapjco.jar` file is copied to the location where the `sapusershim.jar` file has been installed.

On UNIX systems, ensure that the environment variables used for the successful completion of the User JCO Test are also in the environment of the driver. If these conditions are met, there should be no driver errors that are related to the JCO.

4.1.4 Understanding Test Error Messages

Use the information in this section to analyze error messages that might display during the User JCO Test. Some errors are applicable to all platforms, and other errors are platform-specific.

The test has been run on the platforms listed below. Other UNIX platforms supported by the JCO are configured in a similar manner and errors generated by improper JCO installation and configuration should be similar to the errors described below. Because of periodic modifications of the JCO, messages might not be exactly as shown.

- "General Errors" on page 30
- "Errors on Win32 Systems" on page 31
- "Errors on IBM-AIX Systems" on page 31
- "Errors on Solaris Systems" on page 32
- "Errors on HP-UX Systems" on page 32

General Errors

Error Message	Problem
Error connecting to SAP host: com.sap.mw.jco.JCO$Exception: (102) RFC_ERROR_COMMUNICATION: Connect to SAP gateway failed .... Check values of Application Server Name/IP Address and System Number	This indicates that one or both of the values entered for the Application Server Name or IP Address and System Number are incorrect. Verify that these values are consistent with the information found in the Properties page of the SAP Logon dialog box used to connect to the SAP R/3 system.
Error authenticating to SAP host: com.sap.mw.jco.JCO$Exception: (103) RFC_ERROR_LOGON_FAILURE: You are not authorized to logon to the target system (error code 1).	The authentication credentials are not valid. Verify that the values for Client Number, User, and User Password are correct.
Error connecting to SAP host: com.sap.mw.jco.JCO$Exception: (101) RFC_ERROR_PROGRAM: Language '<value>' not availableCheck value of Language Code	The language code selected is not valid or is not installed on the SAP R/3 system.

Errors on Win32 Systems

Error Message	Problem
'userjcotest' is not recognized as an internal or external command, operable program, or batch file.	The `userjcotest.bat` batch file is not present.
Exception in thread “main” java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$AbapExceptionor Exception in thread “main” java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$Exception	The `sapjco.jar` file is not in the location specified in the `userjcotest.bat` batch file.
Exception while initializing JCO client.java.lang. UnsatisfiedLinkError: no jRFC12 in java.library.path. Verify proper installation of JCO Native support libraries packaged with JCO client.	The `jRFC12.dll` file that shipped with the JCO client is not installed or is installed in an incorrect location. The default location for `jRFC12.dll` and `libRfc32.dll` is `/winnt/system32`.
Exception while initializing JCO client.java.lang. UnsatisfiedLinkError: C:\WINNT\system32\jrfc12.dll: Can’t find dependent libraries. Verify proper installation of JCO Native support libraries packaged with JCO client.	The `librfc32.dll` file shipped with the JCO client is not installed or is installed in an incorrect location. The default location for `jRFC12.dll` and `libRfc32.dll` is `/winnt/system32`.

Errors on IBM-AIX Systems

Error Message	Problem
ksh: userjcotest: not found.	The `userjcotest` script file is not present in the directory.
Exception in thread “main” java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$AbapExceptionorException in thread “main” java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$Exception	The `sapjco.jar` file is not in the location specified in the jcotest script file or the case specified for `sapjco.jar` does not match the actual filename.
Exception while initializing JCO client.java.lang.UnsatisfiedLinkError: no jRFC12 (libjRFC12.a or .so) in java.library.pathVerify proper installation of JCO Native support libraries packaged with JCO client.	The `libjRFC12.so` file that shipped with the JCO client is not installed or is installed in an incorrect location. You must configure a LIBPATH environment variable to specify the location in which the file resides.
Exception while initializing JCO client.java.lang.UnsatisfiedLinkError: <path>/libjRFC12.so: A file or directory in the path name does not exist. Verify proper installation of JCO Native support libraries packaged with JCO client.	The `librfccm.so` file shipped with the JCO client is not installed or is installed in an incorrect location. You must copy the file to the same location as `libjRFC12.so` or configure the LIBPATH environment variable to specify the location in which the file resides.

Errors on Solaris Systems

Error Message	Problem
ksh: userjcotest: not found.orbash: userjcotest: command not found	The `userjcotest` script file is not present in the directory.
Exception in thread "main" java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$AbapExceptionorException in thread "main" java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$Exception	The `sapjco.jar` file is not in the location specified in the `jcotest` script file or the case specified for `sapjco.jar` does not match the actual filename.
Exception while initializing JCO client.java.lang.UnsatisfiedLinkError: no jRFC12 in java.library.pathVerify proper installation of JCO Native support libraries packaged with JCO client.	The `libjRFC12.so` shipped with the JCO client is not installed or is installed in an incorrect location. You must configure a LD_LIBRARY_PATH environment variable to specify the location in which the file resides.
Exception while initializing JCO client.java.lang.UnsatisfiedLinkError: <path>/libjRFC12.so: ld.so.1: <search-path>: fatal: librfccm.so: open failed: No such file or directoryVerify proper installation of JCO Native support libraries packaged with JCO client.	The `librfccm.so` file shipped with the JCO client is not installed or installed in incorrect location. You must copy the file to the same location as `libjRFC12.so` or configure the LD_LIBRARY_PATH environment variable to specify the location in which the file resides.

Errors on HP-UX Systems

Error Message	Problem
ksh: userjcotest: not found.orbash: userjcotest: command not found	The `userjcotest` script file is not present in the directory.
Exception in thread "main" java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$AbapExceptionorException in thread "main" java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$Exception	The `sapjco.jar` file is not in the location specified in the `jcotest` script file or the case specified for `sapjco.jar` does not match the actual filename.
Exception while initializing JCO client.java.lang.ExceptionInInitializerError: JCO.classInitialize(): Could not load middleware layer 'com.sap.mw.jco.rfc.MiddlewareRFCno sapjcorfc in java.library.pathVerify proper installation of JCO Native support libraries packaged with JCO client.	The `libjRFC12.sl` file shipped with the JCO client is not installed or is installed in an incorrect location. You must configure a SHLIB_PATH environment variable to specify the location in which the file resides.
Exception while initializing JCO client.java.lang.ExceptionInInitializerError: JCO.classInitialize(): Could not load middleware layer 'com.sap.mw.jco.rfc.MiddlewareRFC . . .Verify proper installation of JCO Native support libraries packaged with JCO client.	The `librfccm.sl` file shipped with the JCO client is not installed or is installed in an incorrect location. You must copy the file to the same location as `libjRFC12.sl` or configure the SHLIB_PATH environment variable to specify the location in which the file resides.

Errors on Linux Systems

Error Message	Problem
ksh: userjcotest: not found.orbash: jcotest: command not found	The `userjcotest` script file is not present in the directory.
Exception in thread “main” java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$AbapExceptionorException in thread “main” java.lang.NoClassDefFoundError: com/sap/mw/jco/JCO$Exception	The `sapjco.jar` file is not in the location specified in the `jcotest` script file or the case specified for `sapjco.jar` does not match the actual filename.
Exception while initializing JCO client.java.lang.ExceptionInInitializerError: JCO.classInitialize(): Could not load middleware layer ’com.sap.mw.jco.rfc.MiddlewareRFCno jRFC12 in java.library.pathVerify proper installation of JCO Native support libraries packaged with JCO client.	The `libjRFC12.so` file shipped with the JCO client is not installed or is installed in an incorrect location. You must configure a LD_LIBRARY_PATH environment variable to specify the location in which the file resides
Exception while initializing JCO client.java.lang.ExceptionInInitializerError: JCO.classInitialize(): Could not load middleware layer ’com.sap.mw.jco.rfc.MiddlewareRFC<path>/libjRFC12.so: librfccm.so: cannot open shared object file: No such file or directoryVerify proper installation of JCO Native support libraries packaged with JCO client.	The `librfccm.so` file shipped with the JCO client is not installed or is installed in an incorrect location. You must copy the file to the same location as `libjRFC12.so` or configure the LD_LIBRARY_PATH environment variable to specify the location in which the file resides.

Creating a New Driver

5

After the SAP User driver files are installed on the server where you want to run the driver (see Chapter 2, “Installing the Driver Files,” on page 19), you can create the driver in the Identity Vault. You do so by importing the basic driver configuration file and then modifying the driver configuration to suit your environment. The following sections provide instructions:

- Section 5.1, “Creating an SAP User Account,” on page 35
- Section 5.2, “Creating the Driver in Designer,” on page 35
- Section 5.3, “Creating the Driver in iManager,” on page 38
- Section 5.4, “Activating the Driver,” on page 41

5.1 Creating an SAP User Account

The driver requires an administrative account for access to the SAP User system. You can use an existing administrative account; however, we recommend that you create an administrative account exclusively for the driver.

5.2 Creating the Driver in Designer

You create the driver by importing the driver’s basic configuration file and then modifying the configuration to suit your environment. After you’ve created and configured the driver, you need to deploy it to the Identity Vault and start it.

- Section 5.2.1, “Importing the Driver Configuration File,” on page 35
- Section 5.2.2, “Configuring the Driver,” on page 36
- Section 5.2.3, “Deploying the Driver,” on page 37
- Section 5.2.4, “Starting the Driver,” on page 38

5.2.1 Importing the Driver Configuration File

1. In Designer, open your project.
2. In the Modeler, right-click the driver set where you want to create the driver, then select *New > Driver* to display the Driver Configuration Wizard.
3. In the Driver Configuration list, select *SAP User Management*, then click *Run*.
4. On the Import Information Requested page, fill in the following fields:

   Driver Name: Specify a name that is unique within the driver set.

   SAP System Number: Specify the SAP system number on the SAP application server.

   SAP User Client Number: Specify the client number to be used on the SAP application server.

   Publisher IDoc Directory: Specify the file system location where the SAP USer IDoc files are placed by the SAP ALE system (for FILE port) or by the driver (for TRFC port).

User Container: Select the Identity Vault container where any new users created from delimited text file information will be placed. This value becomes the default for all drivers in the driver set. If you don't want to change this value for all drivers, leave this field alone and change the value on the driver's Global Configuration Values page after you've finished importing the driver.

Driver is Local/Remote: Select *Local* if this driver will run on the Metadirectory server without using the Remote Loader service. Select *Remote* if you want the driver to use the Remote Loader service, either locally on the Metadirectory server or remotely on another server.

SAP User ID: Enter the ID of the User this driver will use for SAP Logon.

SAP User Password: Enter the User password this driver will use for SAP Logon.

SAP Application Server: Specify the Host Name or IP Adress to the appropriate SAP Application Server.

5 (Conditional) If you chose to run the driver remotely, click *Next*, then fill in the fields listed below. Otherwise, skip to Step 6.

Remote Host Name and Port: Specify the host name or IP address of the server where the driver's Remote Loader service is running.

Driver Password: Specify the driver object password that is defined in the Remote Loader service. The Remote Loader requires this password to authenticate to the Metadirectory server.

Remote Password: Specify the Remote Loader's password (as defined on the Remote Loader service). The Metadirectory engine (or Remote Loader shim) requires this password to authenticate to the Remote Loader

6 Click *Next* to import the driver configuration.

At this point, the driver is created from the basic configuration file. To ensure that the driver works the way you want it to for your environment, you must review and modify the driver's default configuration settings.

7 To review or modify the default configuration settings, click *Configure*, then continue with the next section, Configuring the Driver.

or

To skip the configuration settings at this time, click *Close*. When you are ready to configure the settings, continue with the next section, Configuring the Driver.

5.2.2 Configuring the Driver

After importing the driver configuration file, you need to configure the driver before it can run. You should complete the following tasks to configure the driver:

- **Ensure that the driver can authenticate to the SAP HR system:** Make sure that you've established an SAP User administrative account for the driver (see Section 5.1, "Creating an SAP User Account," on page 35) and the correct authentication information, including the User ID and password, is defined for the driver parameters (see Section A.1.3, "Authentication," on page 76).
- **Configure the driver parameters:** 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 configure the driver parameters located on the Driver Configuration page. For information about the driver parameters, see Section A.1.5, “Driver Parameters,” on page 78.

- **Customize the driver policies and filter:** Modify the driver policies and filter to implement your business policies. For instructions, see Chapter 7, “Customizing the Driver,” on page 45.
- **Configure the driver for use in a Central User Administration Environment:** If you want to integrate the driver into a Central User Administration (CUA) environment, see Chapter 8, “Using the Driver in a Central User Administration Environment,” on page 59.

Continue with the next section, Deploying the Driver.

5.2.3 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 the deployment summary, then click *Deploy*.

6 Read the 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*.

5.2.4 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*.
3. Continue with Section 5.4, "Activating the Driver," on page 41.

5.3 Creating the Driver in iManager

You create the driver by importing the driver's basic configuration file and then modifying the configuration to suit your environment. After you've created and configured the driver, you need to start it.

- Section 5.3.1, "Importing the Driver Configuration File," on page 38
- Section 5.3.2, "Configuring the Driver," on page 40
- Section 5.3.3, "Starting the Driver," on page 41

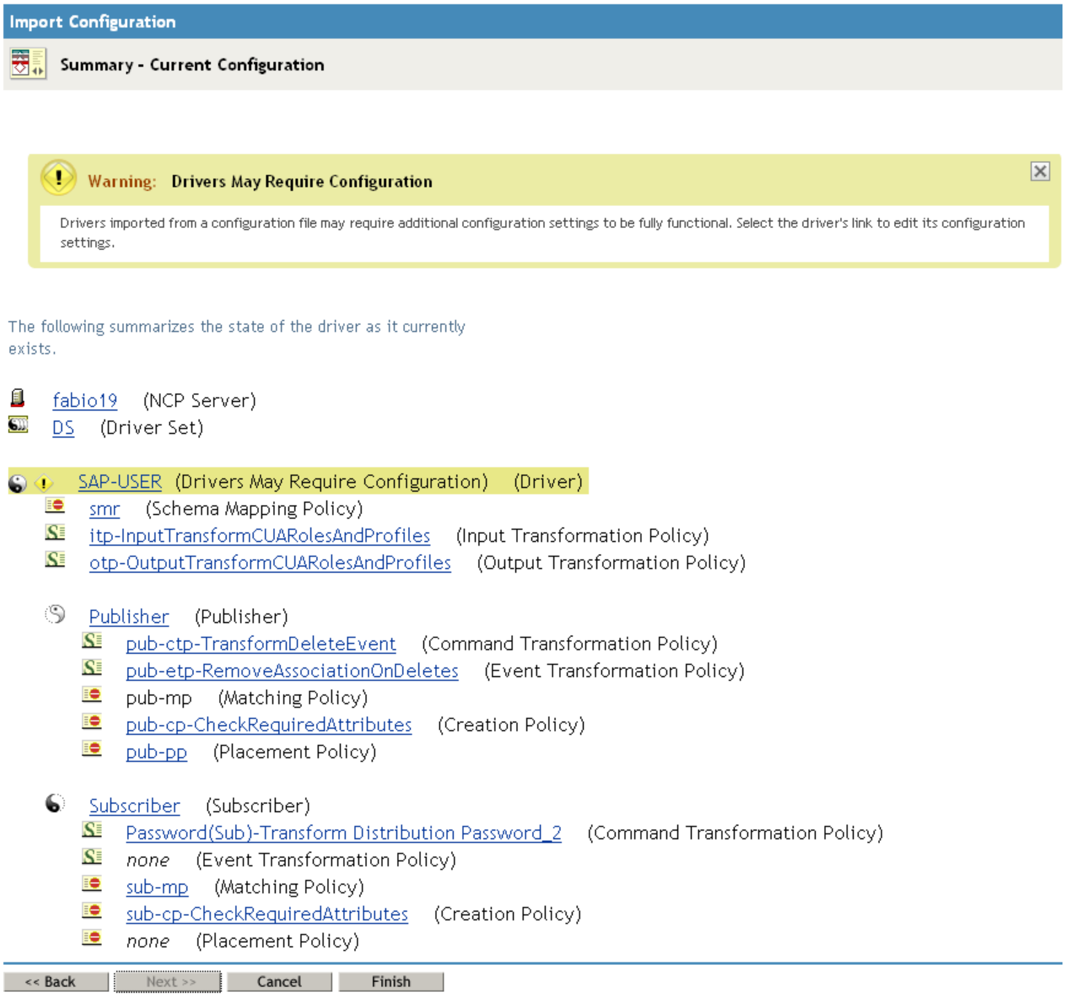
5.3.1 Importing the Driver Configuration File

1. In iManager, click to display the Identity Manager Administration page.
2. In the Administration list, click *Import Configuration* to launch the Import Configuration wizard.
3. Follow the wizard prompts, filling in the requested information (described below) until you reach the Summary page.

Prompt	Description
Where do you want to place the new driver?	You can add the driver to an existing driver set, or you can create a new driver set and add the driver to the new set. If you choose to create a new driver set, you are prompted to specify the name, context, and server for the driver set.
Import a configuration into this driver set	Use the default option, *Import a configuration from the server (.XML file)*. In the *Show* field, select *Identity Manager 3.6 configurations*. In the *Configurations* field, select the SAPUser file.
Driver name	Type a name for the driver. The name must be unique within the driver set.
SAP System Number	Specify the SAP system number on the SAP application server.
SAP User Client Number	Specify the client number to be used on the SAP application server.

Prompt	Description
Publisher IDoc Directory	Specify the file system location where the SAP User IDoc files are placed by the SAP Ale system (for FILE port) or by the driver (for TRFC port).
User Container	Select the Identity Vault container where any new users created from delimited text file information will be placed. This value becomes the default for all drivers in the driver set. If you don't want to change this value for all drivers, leave this field alone and change the value on the driver's Global Configuration Values page after you've finished importing the driver.
Driver is Local/Remote	Select *Local* if this driver will run on the Metadirectory server without using the Remote Loader service. Select *Remote* if you want the driver to use the Remote Loader service, either locally on the Metadirectory server or remotely on another server.
Remote Host Name and Port	This applies only if the driver is running remotely. Specify the host name or IP address of the server where the driver's Remote Loader service is running.
Driver Password	This applies only if the driver is running remotely. Specify the driver object password that is defined in the Remote Loader service. The Remote Loader requires this password to authenticate to the Metadirectory server.
Remote Password	This applies only if the driver is running remotely. Specify the Remote Loader's password (as defined on the Remote Loader service). The Metadirectory engine (or Remote Loader shim) requires this password to authenticate to the Remote Loader
SAP User ID	Enter the ID of the User this driver will use for the SAP Logon.
SAP User Password	Enter the User password this driver will use for the SAP Logon.
SAP Application Server	Enter the Host Name or IP Address for connecting to the appropriate SAP Application Server.
Define Security Equivalences	The driver requires rights to objects within the Identity Vault and to the input and output directories on the server. 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.
Exclude Administrative Roles	You should exclude any administrative User objects (for example, Admin and DriversUser) from synchronization.

When you finish providing the information required by the wizard, a Summary page, similar to the following is displayed.



Warning: Drivers May Require Configuration

Drivers imported from a configuration file may require additional configuration settings to be fully functional. Select the driver's link to edit its configuration settings.

The following summarizes the state of the driver as it currently exists.

fabio19 (NCP Server)
DS (Driver Set)

SAP-USER (Drivers May Require Configuration) (Driver)
- smr (Schema Mapping Policy)
- itp-InputTransformCUARolesAndProfiles (Input Transformation Policy)
- otp-OutputTransformCUARolesAndProfiles (Output Transformation Policy)
- Publisher (Publisher)
  - pub-ctp-TransformDeleteEvent (Command Transformation Policy)
  - pub-etp-RemoveAssociationOnDeletes (Event Transformation Policy)
  - pub-mp (Matching Policy)
  - pub-cp-CheckRequiredAttributes (Creation Policy)
  - pub-pp (Placement Policy)
- Subscriber (Subscriber)
  - Password(Sub)-Transform Distribution Password_2 (Command Transformation Policy)
  - *none* (Event Transformation Policy)
  - sub-mp (Matching Policy)
  - sub-cp-CheckRequiredAttributes (Creation Policy)
  - *none* (Placement Policy)

At this point, the driver is created from the basic configuration file. To ensure that the driver works the way you want it to for your environment, you must review and modify the driver's default configuration settings.

4 To modify the default configuration settings, click the linked driver name, then continue with the next section, Configuring the Driver.

or

To skip the configuration settings at this time, click *Finish*. When you are ready to configure the settings, continue with the next section, Configuring the Driver.

5.3.2 Configuring the Driver

After importing the driver configuration file, you need to configure the driver before it can run. You should complete the following tasks to configure the driver:

- **Ensure that the driver can authenticate to the SAP HR system:** Make sure that you've established an SAP User administrative account for the driver (see Section 5.1, "Creating an SAP User Account," on page 35) and the correct authentication information, including the User ID and password, is defined for the driver parameters (see Section A.1.3, "Authentication," on page 76).

- **Configure the driver parameters:** 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 configure the driver parameters located on the Driver Configuration page. For information about the driver parameters, see Section A.1.5, "Driver Parameters," on page 78.
- **Customize the driver policies and filter:** Modify the driver policies and filter to implement your business policies. For instructions, see Chapter 7, "Customizing the Driver," on page 45.
- **Configure the driver for use in a Central User Administration Environment:** If you want to integrate the driver into a Central User Administration (CUA) environment, see Chapter 8, "Using the Driver in a Central User Administration Environment," on page 59.

Continue with the next section, Starting the Driver.

5.3.3 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 iManager, click to display the Identity Manager Administration page.
2. Click *Identity Manager Overview*.
3. Browse to and select the driver set object that contains the driver you want to start.
4. Click the driver set name to access the Driver Set Overview page.
5. Click the upper right corner of the driver, then click *Start driver*.
6. Continue with Section 5.4, "Activating the Driver," on page 41.

5.4 Activating the Driver

If you created the driver in a driver set where you've 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 3.6 Installation Guide*.

Upgrading an Existing Driver

6

The following sections provide information to help you upgrade an existing driver to version 3.6:

- Section 6.1, "Supported Upgrade Paths," on page 43
- Section 6.2, "What's New in Version 3.6," on page 43
- Section 6.3, "Upgrade Procedure," on page 43

6.1 Supported Upgrade Paths

You can upgrade from any 3.*x* version of the SAP User driver. Upgrading a pre-3.*x* version of the driver directly to version 3.6 is not supported.

6.2 What's New in Version 3.6

Version 3.6 of the driver does not include any new features that are not already in the 3.5.1 version. However, the process for creating a new driver has changed. For detailed information, see Chapter 5, "Creating a New Driver," on page 35.

6.3 Upgrade Procedure

The process for upgrading the SAP User driver is the same as for other Identity Manager drivers. For detailed instructions, see "Upgrading" in the *Identity Manager 3.6 Installation Guide*.

Customizing the Driver

7

The policies and filter included in the default driver configuration provide bidirectional creation, deletion, and modification of User information between the Identity Vault and the SAP system. The driver is configured to synchronize more information from the Identity Vault to SAP (Subscriber channel) than from SAP to the Identity Vault (Publisher channel).

The following sections explain how the default driver configuration uses policies and the filter. You can use this overview as a basis to create your own policies and filters for specific business implementations.

7.1 Modifying Policies and the Filter

You must modify policies and the filter to work with your specific business environment. We recommend that you make modifications in this order:

1. Modify the Filter (publish and subscribe options) to include additional attributes you want synchronized.
2. Modify the Mapping policy to include all attributes specified in the Subscriber and Publisher channel filters.
3. Modify the InputTransform policy
4. Modify the OutputTransform policy
5. Modify the Publisher policies
6. Modify the Subscriber policies

Refer to the following sections for information:

7.1.1 Filter Publish Options

Setting attributes in the filter to *publish* specifies which classes and attributes are published from the SAP system to the Identity Vault.

The default driver configuration publishes the following User class attributes in the filter.

Class	Attributes
User	DirXML-sapLocRoles
	DirXML-sapLocProfiles
	Given Name
	Surname
	sapProfiles
	sapRoles
	sapUsername

7.1.2 Filter Subscribe Options

Setting attributes in the filter to *subscribe* specifies which classes and attributes are synchronized from the Identity Vault to the SAP system.

The default driver configuration subscribes to the following User class attributes in the filter:

Class	Attributes
User	buildingName costCenter firstPrefix floor Full Name Given Name Initials Internet Email Address Login Disabled OU pager sapGroups sapProfiles sapRoles Surname Telephone Number Title

7.1.3 Schema Mapping Policy

The Schema Mapping policy is referenced by the driver object and applies to both the Subscriber and Publisher channel. The purpose of the Schema Mapping policy is to map schema names (particularly attribute names and class names) between the Identity Vault and the SAP User database. Any modification or removal of existing entries in the Schema Mapping policy could destroy the default configuration and policies processing behavior. Adding new attribute mappings is discretionary.

NOTE: The Application Schema definition in the default driver configuration is from a SAP R/3 version 4.7 system with Web Application Server version 6.40. If the target SAP system is a different version, the actual User object schema might be different. Refresh application schema using the iManager Schema Mapping editor to obtain the actual schema of the target server.

The following class mapping is included with the default driver configuration:

Identity Vault Class	SAP Class	SAP Description
User	US	USER

The User class is configured to synchronize bidirectionally between SAP and the Identity Vault. A change made in one system will transfer to the other system.

All attributes in the Publisher and Subscriber filters should be mapped unless they are used only for policy processing.

SAP User field values can be arranged in three types:

- Simple fields: These values are not grouped with other fields. The syntax in the schema map is <field name>.
- Structure fields: These values are grouped with other pieces of data that describe a larger collection of single-instance data. The syntax for these fields in the schema map is <structure name>:<field name>. For example, ADDRESS:TELEPHONE.
- Table fields: These values are similar to Structure fields, but there can be multiple instances of the structured data. The syntax for these fields in the schema map is <table name>:<field name>. For example, ADDTEL:TELEPHONE.

The following table includes common attribute mappings for the User class and their descriptions, assuming that only the primary piece of structure communication data is required (such as ADDTEL:TELEPHONE). If fields of a table are to be mapped, you should specify only the Table name in the mapping (such as LOCACTIVITYGROUPS). If you do this, the driver generates all table field values in structured format. For more information, see Appendix F, "Structured Format Examples," on page 99. On the Publisher channel, the structured data must be transformed to string format.

The default mappings for the driver are as follows:

Identity Vault Attribute	SAP User Field Description	SAP User Field(s)
DirXML-sapLocRoles	Role for specified CUA logical system	LOCACTIVITYGROUPS:SUBSYSTEM LOCACTIVITYGROUPS:AGR_NAME
DirXML-sapLocProfiles	Profile for specified CUA logical system	LOCPROFILES:SUBSYSTEM LOCPROFILES:PROFILE
DirXML-sapVClass	License type classification	UCLASS:LIC_TYPE
DirXML-sapLocVClass	License type classification for specified CUA Logical System	UCLASSSYS:RCVSYSTEM UCLASSSYS:LIC_TYPE
birthName	Name of person at birth	ADDRESS:BIRTH_NAME
buildingName	Building (number or code)	ADDRESS:BUILDING_P
commType	Communication type (key) (Central address management)	ADDRESS:COMM_TYPE
company	Company address, cross-system key	COMPANY:COMPANY
costCenter	Cost center	DEFAULTS:KOSTL
Facsimile Telephone Number	Fax number: dialing code+number	ADDFAX:FAX
firstPrefix	Name prefix	ADDRESS:PREFIX1

Identity Vault Attribute	SAP User Field Description	SAP User Field(s)
floor	Floor in building	ADDRESS:FLOOR_P
Full Name	Complete personal name	ADDRESS:FULLNAME
Given Name	First name	ADDRESS:FIRSTNAME
inHouseMail	Int. mail postal code	ADDRESS:INHOUSE_ML
Initials	Middle Initial or personal initials	ADDRESS:INITIALS
InitialsSig	Short name for correspondence	ADDRESS:INITS_SIG
Internet EMail Address	Internet mail (SMTP) address	ADDSMPT:E_MAIL
Login Disabled	Lock User account	LOCKUSER
		The LOCKUSER attribute does not actually exist in SAP. This pseudo-attribute is used by the driver to determine when to call USER_LOCK and USER_UNLOCK BAPI functions.
middleName	Middle name or second forename of a person	ADDRESS:MIDDLENAME
nickname	Nickname or name used	ADDRESS:NICKNAME
OU	Department	ADDRESS:DEPARTMENT
pager	Pager number	ADDPAG:PAGER
personalTitle	Title text	ADDRESS:TITLE_P
roomNumber	Room or apartment number	ADDRESS:ROOM_NO_P
sapAlias	Internet user alias	ALIAS:USERALIAS
sapCATT	CATT: Test status	DEFAULTS:CATTKENNZ
sapClass	User group in user master maintenance	LOGONDATA:CLASS
sapDateFormat	Date format	DEFAULTS:DATFM
sapDecimalFormat	Decimal Notation	DEFAULTS:DCPFM
sapGroups	User group in user master maintenance	GROUPS:USERGROUP
sapLoginLanguage	Language	DEFAULTS:LANGU
sapParameters	Get/Set parameter ID and parameter values	PARAMETER:PAR10
sapPrintParam1	Print parameter 1	DEFAULTS:SPLG
sapPrintParam2	Print parameter 2	DEFAULTS:SPDB
sapPrintParam3	Print parameter 3	DEFAULTS:SPDA
sapProfiles	Profile name	PROFILES:BAPIPROF
sapRefUser	User name in user master record	REF_USER:REF_USER

Identity Vault Attribute	SAP User Field Description	SAP User Field(s)
sapRoles	Role Name	ACTIVITYGROUPS:AGR_NAME
sapSncGuiFlag	Unsecure communication permitted flag	SNC:GUIFLAG
sapSncName	Secure network communication printable name	SNC:PNAME
sapSpool	Spool: Output device	DEFAULTS:SPLD
sapStartMenu	Start Menu	DEFAULTS:START_MENU
sapTimeZone	Time zone	LOGONDATA:TZONE
sapUsername	User Name	USERNAME:BAPIBNAME
sapUserType	User Type	LOGONDATA:USTYP
sapValidFrom	User valid from	LOGONDATA:GLTGV
sapValidTo	User valid to	LOGONDATA:GLTGB
secondName	Second surname of a person	LOGONDATA: SECONDNAME
secondPrefix	Name prefix	ADDRESS:PREFIX2
Surname	Last name	ADDRESS:LASTNAME
Telephone Number	Telephone no.: dialing code+number	ADDTEL:TELEPHONE
telexNumber	Telex Number	ADDTLX:TELEX_NO
Title	Function	ADDRESS:FUNCTION
titleAcademic1	Academic title: written form	ADDRESS:TITLE_ACA1
titleAcademic2	Academic title: written form	ADDRESS:TITLE_ACA2

7.1.4 Input Transform Policy

You modify the Input Transform policy to implement your specific business rules. The Input Transform policy is applied to affect a transformation of the data received from the driver shim.

The policy is applied as the first step of processing an XML document received from the driver shim. The Input Transform policy converts the syntax of the SAP attributes into the syntax for the Identity Vault.

The default driver configuration includes two rules that perform the following functions:

- Transforming LOCACTIVITYGROUPS from structured format to string format.
- Transforming LOCPROFILES from structured format to string format.

7.1.5 Output Transform Policy

You modify the Output Transform policy to implement your specific business rules. The Output Transformation policy is referenced by the driver object and applies to both the Subscriber channel and to the Publisher channel. The purpose of the Output Transformation policy is to perform any final transformation necessary on XML documents sent to the driver by Identity Manager.

The default driver configuration:

- Transforms LOCACTIVITYGROUPS from string format to structured format.
- Transforms LOCPROFILES from string format to structured format.
- Adds the driver's LOCACTIVITYGROUPS attribute to Modify events with the from-merge attribute set.
- Transforms the pseudo-attribute LOCKUSER value from a true/false format to a 1/0 format.
- Transforms ADDFAX:FAX values from structured format to string format.
- Adds USERNAME:BAPIBNAME to the Queries style sheet (invokes the driver's wildcard search functionality; see Appendix H, "Using Wildcard Search Capabilities," on page 103.)

7.1.6 Publisher Placement Policy

The Publisher Placement policy is applied to an Add Object event document to determine the placement of the new object in the hierarchical structure of the Identity Vault.

The Placement policy places all User objects in an Identity Vault container that you specify during installation. You can also modify this location by using the Publisher User Placement Global Configuration Variable (GCV.)

The default driver configuration:

- Appends <remove-association> to Delete events; it's used in conjunction with the Publisher Command Transformation policy.

7.1.7 Publisher Matching Policy

The Publisher Matching policy is applied to a Modify Object event document. Matching policies establish links between an existing entry in the Identity Vault and an existing entry in the SAP system. The Matching policy attempts to find an existing object that matches the object generating the event by the criteria specified in the policy.

The default driver checks for matches based on the sapUsername attribute. A fallback policy is also provided that checks for matches on the Given Name and Surname attributes.

7.1.8 Publisher Create Policy

The Publisher Create policy is applied when a new object is to be added to the Identity Vault. The default driver configuration:

- Creates a User object (Surname and Given Name attributes are required)

- Generates a unique CN based on Given Name and Surname attributes
- Sets the initial account password on creation. Allows an administrator or user to reset or change passwords.

7.1.9 Subscriber Matching Policy

The Subscriber Matching policy is applied to a Modify Object event document. Matching policies establish links between an existing entry in the Identity Vault and an existing entry in the SAP system. The Matching policy attempts to find an existing object that matches the object generating the event by the criteria specified in the policy.

The default driver checks for matches based on the values of the Given Name, Surname, and sapUsername attributes.

If you do not have an association in your query, the SAP system performs a full table scan of the user table. This might cause a long delay in receiving a reply from the matching query.

If the specified user name is known in SAP, adding an association value reduces the query to a single object. You can use the following Output Transformation policy to add the association.

```
<rule>
<description>Add association value to matching queries</description>
<conditions>
<and>
<if-operation op="equal">query</if-operation>
<if-xpath op="not-true">association</if-xpath>
<if-xpath
op="true">search-attr[@attr-name="USERNAME:BAPIBNAME"]/value</if-xpath>
</and>
</conditions>
<actions>
<do-append-xml-element expression="." name="association"/>
<do-append-xml-text expression="association">
<arg-string>
<token-text xml:space="preserve">USd</token-text>
<token-upper-case>
<token-xpath
expression='search-attr[@attr-name="USERNAME:BAPIBNAME"]/value/text()'/>
</token-upper-case>
</arg-string>
</do-append-xml-text>
</actions>
</rule>
```

7.1.10 Subscriber Create Policy

The Subscriber Create policy is applied when you want to add a new object to the Identity Vault. The default driver configuration:

- Ensures that the Surname and Given Name attributes are present.
- Generates an unique CN based on the Given name and Surname attributes.
- Appends the sapUserType attribute with a value of A.
- Sets the initial password (the driver can also set and manage persistent passwords in the SAP system.)

- Sets a default sapRoles value of SAP_ESSUSER.
- Sets a default sapProfiles value of SAP_NEW.
- Adds the following sample DirXML-sapLocRole values: DRVCLNT100:, ADMCLNT100:SAP_EMPLOYEE, and ADMCLNT500:SAP_ESSUSER.
- Adds the following sample DirXML-sapLocProfiles values: DRVCLNT100:, ADMCLNT100:SAP_ALL, and ADMCLNT500:SAP_NEW.

7.2 Adding the Organizational Role Class

The SAP User driver can be queried for ACTIVITYGROUP objects and all other PDOBJECTS in the SAP User database so that they may be synchronized into the Identity Vault, and used by the administrator through a browse interface. To do this, the default class mapping must be manually changed to the following:

Identity Vault Class	SAP User Field Description	SAP User Field(s)
Organizational Role	PDOBJECT	Organizational Role

The following sections explain what you need to do to allow support for querying of the Organizational Role class:

- Section 7.2.1, "Editing the Global Configuration Values," on page 53
- Section 7.2.2, "Adding a New Placement Rule," on page 54
- Section 7.2.3, "Modifying the XSLT," on page 55
- Section 7.2.4, "Adding the Organizational Role Class to the Driver Filter," on page 55
- Section 7.2.5, "Migrating Data into the Identity Vault," on page 56

7.2.1 Editing the Global Configuration Values

To edit the Global Configuration Values (GCV), follow these steps:

1. In iManager, browse to the driver, and click the upper right corner of the driver icon.
2. Select the *Edit Properties* link.

   The Driver Configuration window is displayed.
3. Click the *Global Config Values* tab.

   A list of the existing GCV values is displayed.
4. Click the *Edit XML* tab to open the XML Editor window.
5. Select the *Enable XML Editing* checkbox and add the following XML code:

```
<definition display-name="Organizational Role Placement"
dn-space="dirxml" dn-type="slash" name="sap-pdobject-placement" type="dn">
    <description>
      The name of the Organizational Role object under which
published SAP Organizational Roles will be placed.
     </description>
<value> </value>
</definition>
```

6 Click *Apply* and *OK* to save the changes.

The updated GCV is now displayed in the list.

7 Browse and select the container in the Identity Vault where you want to place the Organizational Role.

8 Click *Apply and OK.*

7.2.2 Adding a New Placement Rule

A new rule is required in the placement policy, to place the Organizational Role object in. Follow these steps to create the new rule:

1 In iManager, click on the driver icon.

The Identity Manager Overview screen is displayed.

2 In the publisher channel, click on the Placement Policies icon.

The Publisher Placement policy window is displayed.

3 Click the existing default publisher placement policy.

The Policy Rules screen is displayed.

4 Click the `Edit XML` tab.

The XML Editor window is displayed.

5 Select the *Enable XML Editing* checkbox and add the following XML code:

```
<rule>
    <description>Organizational Role Placement</description>
    <conditions>
       <or>
         <if-class-name op="equal">
            Organizational Role
         </if-class-name>
       </or>
       <or>
        <if-op-attr name="CN" op="available"/>
       </or>
    </conditions>
    <actions>
      <do-set-op-dest-dn>
        <arg-dn>
        <token-global-variable name="sap-pdobject-placement"/>
        <token-text xml:space="preserve">\</token-text>
        <token-escape-for-dest-dn>
        <token-op-attr name="CN"/>
        </token-escape-for-dest-dn>
        </arg-dn>
      </do-set-op-dest-dn>
     </actions>
</rule>
```

6 Click *Apply* and *OK,* to save the changes.

7 Click *Close* to close the Publisher Placement Policy window.

7.2.3 Modifying the XSLT

The XSLT file must be modified so that it triggers events only for the USER class. To modify the XSLT file, follow these steps:

1. From the Identity Manager Driver Overview page, click on the Creation Policies icon on the publisher channel of the driver.

   The Publisher Creation Policy window is displayed.
2. Click the *Generate User Name Style Sheet* link.

   The XML Editor window is displayed.
3. Search for the following XML code:
   `<xsl:template match="add">`

   Replace it with the following code:

   ```
   <xsl:template match="add[@class-name='User']">
   ```
4. Click *Apply* and *OK* to save the changes .
5. Click *Close,* to close the Publisher Placement Policy window.

7.2.4 Adding the Organizational Role Class to the Driver Filter

To add the Organizational Role class, and to change the default class mapping, follow these steps:

1. From the Identity Manager Driver Overview page, click the ‘Driver Filter’ icon in the publisher channel.
2. Click the *Add Class* tab.

   A pop-up window is displayed.
3. Click the *Show All Classes* link.

   A list of the available classes is displayed in alphabetical order.
4. Scroll down to the class Organizational Role, and click on it.
5. In the *Application Name* field on the right, browse and select the SAP User class PDOBJECT that will be mapped to Organizational Role.
6. Click *Apply* to confirm the mapping.
7. From the filter window, select Organizational Role, and click on the *Add Attribute* tab.

   A list of the available attributes is displayed.
8. Select the *CN* attribute and click *OK.*
9. In the*Application Name* field on the right, browse and select the SAP attribute *OBJECTS:EXT_OBJ_ID*
10. Select Organizational Role again and click the *Add Attribute* tab.
11. Select the *Description* attribute and click *OK*.
12. In the Application Name field on your right, browse and select the *OBJECTS:LONG_TEXT* attribute.
13. Click *Apply.*
14. In the Filter window, select the Organizational Role class.

15 In the text field on the right, delete PDOBJECT and replace it with AG.

16 Click *Apply* to save the changes.

17 Click *Organizational Role* and select the *Synchronize* option in the publisher channel.

18 Click the *CN* attribute and select the *Synchronize* option in the publisher channel.

19 Click the *Description* attribute and select the *Synchronize* option in the publisher channel.

20 Click *Apply* and *OK* to save the changes, and close the Filter window.

7.2.5 Migrating Data into the Identity Vault

To migrate ACTIVITYGROUP objects into the Identity Vault, ensure that the driver is running and follow these steps:

1 From the Identity Manager Driver Overview window, click *Migrate* > *Migrate into Identity Vault*.

The Migrate Data into the Identity Vault window is displayed.

2 To migrate a single ACTIVITYGROUP object, follow these steps:

  2a Click the *Edit List* tab.

  The Edit Migration Criteria dialog box is displayed.

  2b Select the Organizational Role class from the list on the left side of the window.

  2c Select the *CN* attribute and click *OK*.

  The Attribute Value dialog box is displayed.

  2d Enter a valid value for the *CN* attribute and click *OK*..

  Example of a valid attribute: SAP_ESSUSER

  2e Click *OK* to confirm the entered value, and close the dialog box.

  2f Click *OK* again in the Migrate Data into the Identity Vault window, to start the migration.

  You will see that the *Success* box is now checked, indicating that migration has started.

3 To migrate all ACTIVITYGROUP objects, follow these steps:

  3a Click the *Edit List* tab.

  The Edit Migration Criteria dialog box is displayed.

  3b Select the Organizational Role class from the list, and click *OK*.

  3c Click *OK* again in the Migrate Data into the Identity Vault window, to start the migration.

To verify that the objects you selected have been migrated successfully, you can browse to the container that you specified in the Organizational Role placement policy. Successful migration can also be verified by looking at the DSTRACE window.

7.3 Obtaining Company Address Data for User Objects

There are several attributes of the SAP User object that are associated with the Company Address object assigned to the User. These attributes, by default, are never populated in BAPI or IDoc distributions of User data from the SAP application server. These fields also cannot be read from the User object in SAP. Company Address data is maintained in a table of related records of the ADDRESSORG type. The driver can retrieve this data from the ADDRESSORG table if desired.

The driver parameter to publish Company Address data <nsap-use-addressorg> is set to 1 by default. Setting the value to 1 retrieves the data from the ADDRESSORG table if attributes in the table exist in the Publisher filter, or if the attributes are in <read-attr> elements of a query document. Although this data can be retrieved from the SAP system, ADDRESSORG data cannot be added, modified, or removed from the SAP system via the driver. If the value of this parameter is set to 0, the company address fields are retrieved from the User object itself. As previously mentioned, by default, these fields won't contain any data.

To fully implement the address retrieval functionality, you must configure the driver to receive events when the ADDRESSORG table is modified. By receiving these events, the driver obtains a list of all User objects assigned to the modified ADDRESSORG table and issues modify events with the changed data for each affected user.

To generate ADDRESSORG modify events, you need to modify the ALE distribution model on the SAP application server to include the distribution of the Company Clone (CCLONE) BAPI. Refer to "Creating a Distribution Model" on page 22 and "Modifying the Port Definition" on page 25 for more information.

The following User object fields might be affected by this functionality.

NAME
NAME_2
NAME_3
NAME_4
C_O_NAME
CITY
CITY_NO
DISTRICT
DISTRICT_NO
POSTL_COD1
POSTL_COD2
POSTL_COD3
PO_BOX
PO_BOX_CIT
PBOXCIT_NO
HOUSE_NO2
STR_SUPPL1
STR_SUPPL2
STR_SUPPL3
BUILDING
DISTRICT
FLOOR
ROOM_NO
COUNTRY
COUNTRYIOS
LOCATION
LANGU_ISO
REGION
SORT1
TIME_ZONE

DELIV_DIS	TAXJURCODE
TRANSPZONE	STR_ABBR
STREET	HOUSE_NO
STREET_NO	

Using the Driver in a Central User Administration Environment

8

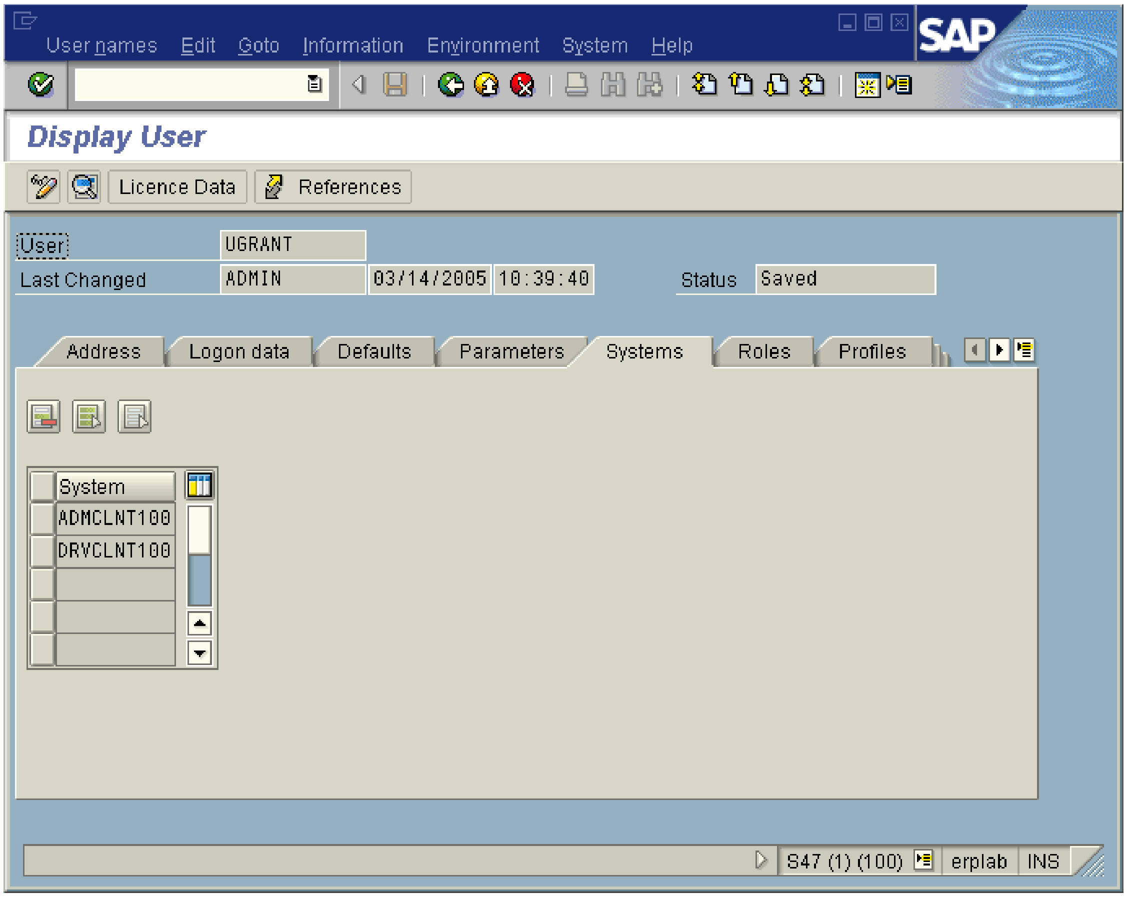
The following sections provide information about integrating the driver into a Central User Administration (CUA) environment. It is not intended to be a CUA configuration or administration guide. Refer to the SAP documentation and SAP help, support, and tips Web sites and journals for authoritative sources of standard CUA information.

8.1 Overview

The driver is designed to perform User management and synchronization with any SAP Application Server. However, the most value can be derived from the driver when it is used in a CUA environment. CUA is the standard User data distribution technology provided by SAP. It is used to distribute data between logical systems on one or more application servers. In a typical CUA landscape, there is one logical system designated as the "Central" system. The Central system administrator has the capability to distribute User account information and access rights to the other "Child" logical systems in the landscape. There are many variations, however, of the flow of User account information, including configurations where the Child systems can locally administrate some of the User account information and distribute it back to the Central system. This information focuses primarily on using the driver in a basic CUA landscape where User account information is distributed one-way from the Central system to the Child logical systems.

The User Maintenance transaction in SAP is SU01. The major difference between the maintenance options in a CUA environment and a non-CUA environment is the existence of the *Systems* tab. The entries under this tab indicate which logical systems to which the User account information should be distributed. The following illustration shows a User that is distributed to logical systems ADMCLNT100 and DRVCLNT100.

Figure 8-1 User distribution to logical systems ADMCLNT100 and DRVCLNT100



User UGRANT

Last Changed ADMIN 03/14/2005 10:39:40

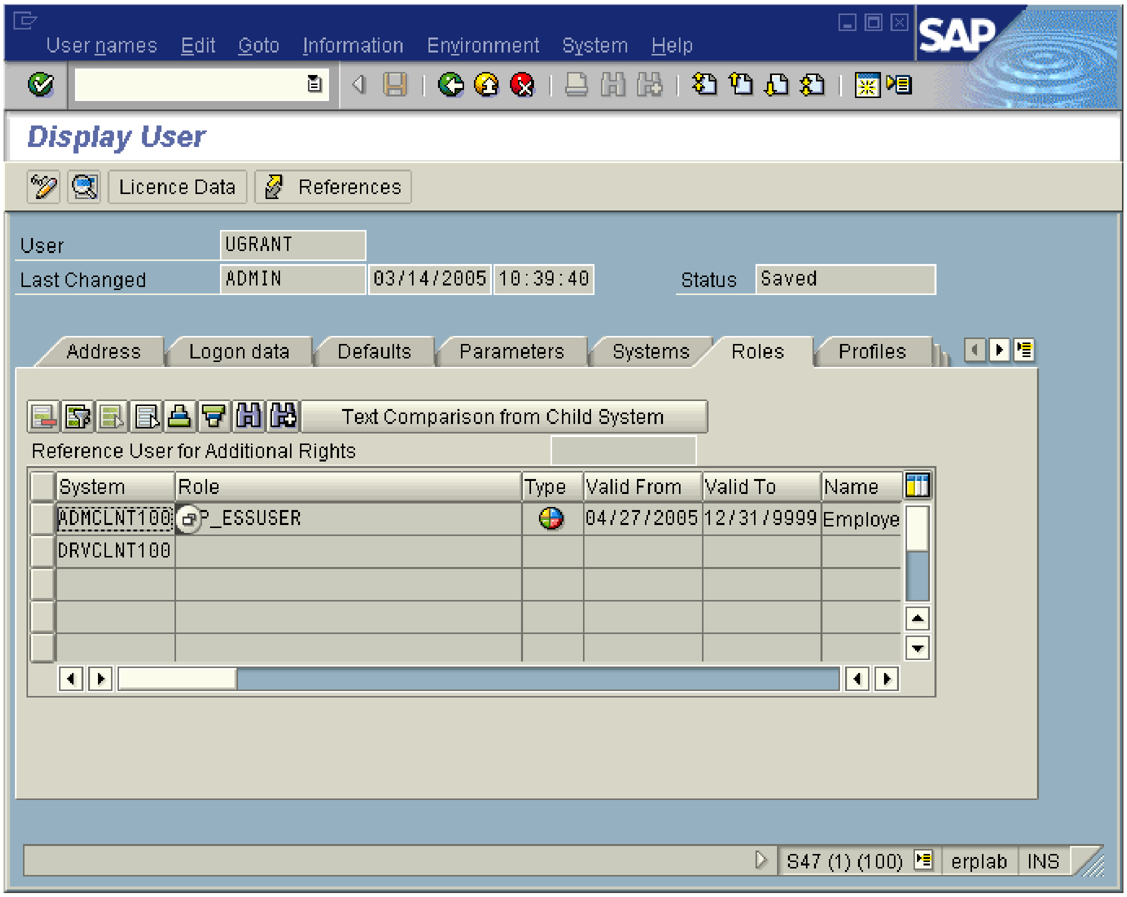
Status Saved

System
ADMCLNT100
DRVCLNT100

S47 (1) (100) erplab INS

Another difference can be seen when the Central system has been configured to maintain Role and Profile information on a *Global* level, which means the Central system administrator can set Role and Profile values for all logical systems in the CUA landscape. When the Global level is selected (via transaction SCUM), the Roles and Profiles for a User account are displayed with the logical system to which they are assigned. The following illustration shows a User assigned the default SAP Employee Self-Service role on logical system ADMCLNT100.

Figure 8-2 *A User with the default SAP Employee Self-Service role on logical system ADMCLNT100*



User	UGRANT
Last Changed	ADMIN 03/14/2005 10:39:40
Status	Saved

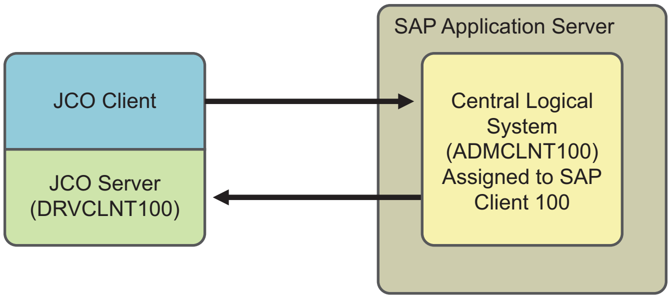
Reference User for Additional Rights

System	Role	Type	Valid From	Valid To	Name
ADMCLNT100	P_ESSUSER		04/27/2005	12/31/9999	Employe
DRVCLNT100					

8.2 Configuring the Driver as a CUA Child System

The driver’s Publisher channel functionality requires that the driver be configured as a Child logical system in a CUA environment. The configuration documentation describes a configuration as illustrated below.

Figure 8-3 *CUA child system configuration*


SAP Application Server
JCO Client
JCO Server (DRVCLNT100)
Central Logical System (ADMCLNT100) Assigned to SAP Client 100

In this configuration, the driver acts as an administrative client to perform User administration, such as User account creation, password set, and role administration, etc., in the CUA Central logical system ADMCLNT100. The Central system is configured to distribute the User account information to the CUA Child logical system DRVCLNT100 that represents the driver. As can be seen from the diagram, the driver acts as both a SAP Client and a Server to obtain full bidirectional synchronization functionality.

After the systems are configured for synchronization, you must set the data attributes that will trigger synchronization. In order to synchronize a User object, you must create a User in SAP Client 100, allow the user to login, and establish synchronization back to the driver.

- Surname and Password are required attributes for User creation
- Set ADMCLNT100 in the *Systems* tab to allow new User to login to Client 100.
- Set DRVCLNT100 in the *Systems* tab to establish data distribution back to the driver.

Setting attributes and passwords has been part of the driver functionality since its creation. As of version 1.0.5, you can now set the *Systems* tab on the Central system using BAPIs for setting Local ActivityGroups (Roles) and Local Profiles. These BAPIs allow the driver to set specified Roles and Profiles on specified logical systems in the CUA landscape. Because there are two component parameters required for each Local Role and Local Profile, the default configuration use a colon ":" delimited string syntax for the Identity Vault values. The form for these values is <Logical System Name>:<Role or Profile Name>. These values are transformed to and from the SAP structured syntax by the default InputTransform and OutputTransform policies.

If you want to set the *Systems* tab for a logical system without setting a Local Role or Local Profile (this should always be done for the driver where SAP Roles and Profiles have no meaning), the string value should be set without the *Role or Profile Name* component.

A new field named FORCE_SYSTEM_ASSIGNMENT is available in newer versions of SAP in the BAPI_USER_CREATE1 function. The driver tries to use this for the *Systems* tab assignment on the Connected SAP System.

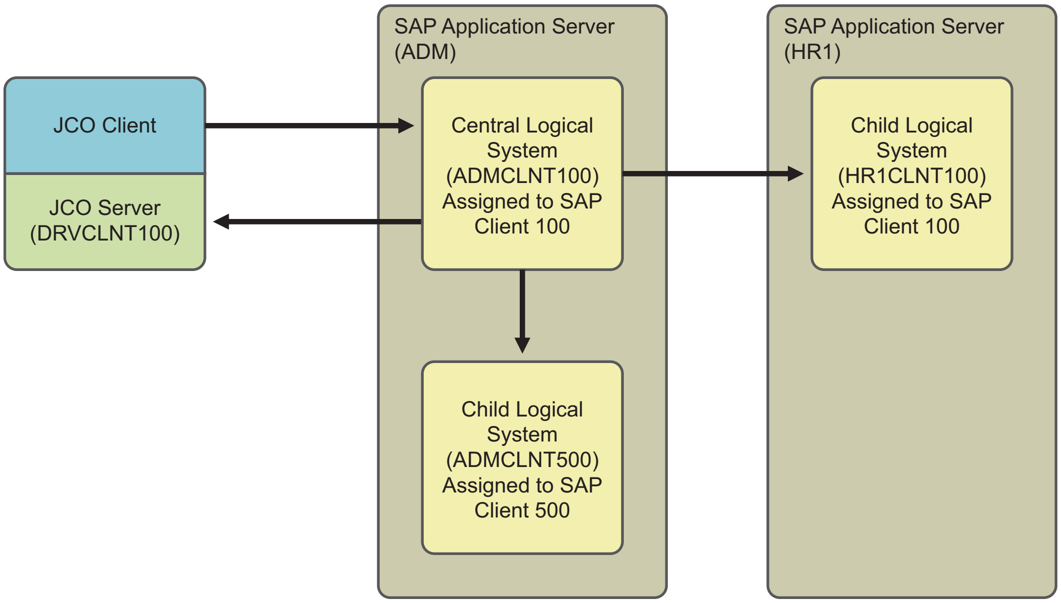
The following example shows a Create style sheet template for the setting of only the *Systems* tab for logical systems ADMCLNT100 and DRVCLNT100. Note that the attr-name used is DirXML-sapLocRoles. For this purpose, the DirXML-sapLocProfiles attribute could also be used. (In Identity Manager 3, this policy is implemented through Policy Builder.)

```
<xsl:template name="add-systems-tab">
<!--
  Sample CUA distribution settings.
  - Central SAP system is ADMCLNT100
  - Driver’s logical system is DRVCLNT100
          ->
    <add-attr attr-name="DirXML-sapLocRoles">
         <!--
     In a CUA environment, set driver’s LS name with a blank role.
is allows
     the driver to receive events from SAP.
     -->
         <value>
            <xsl:value-of select="’DRVCLNT100:’"/>
         </value>
       <!--
         Setting the target LS name with a blank CUA role allows the
         User object to log on to the target child system but receive no rights
         -->
         <value>
              <xsl:value-of select="’ADMCLNT100:’"/>
         </value>
    </add-attr>
</xsl:template>
```

8.3 Using the Driver to Provision a CUA Landscape

The previous example showed a simple CUA environment where the Central system only distributed User data only to the driver’s logical system. This is not a typical environment. In most CUA environments, a Central system distributes data to SAP Child logical systems on multiple application servers. A small example of a typical CUA landscape looks more like this:

Figure 8-4 *A central system distributing data to SAP child logical systems on multiple application servers*


SAP Application Server (ADM)
SAP Application Server (HR1)
JCO Client
JCO Server (DRVCLNT100)
Central Logical System (ADMCLNT100) Assigned to SAP Client 100
Child Logical System (HR1CLNT100) Assigned to SAP Client 100
Child Logical System (ADMCLNT500) Assigned to SAP Client 500

As with the previous example, the driver can set the distribution of User account information to the additional CUA Child systems by setting the *Systems* tab for them. However, the real power of the driver is realized when you use access controls to the various SAP clients based on the driver’s policies. For example, all employees can receive employee Self-Service rights on the HR system, but an employee identified as an HR Administrator could also be granted rights to the HR administration functions. The following example shows a Create Stylesheet template for the setting of the *Systems* tab for logical system ADMCLNT100 and DRVCLNT100, setting the SAP_ESSUSER Role on logical system HR1CLNT100, and setting the SAP_ALL Profile on logical system ADMCLNT500. (In Identity Manager 3, this policy is implemented through Policy Builder.)

```
<xsl:template name="add-cua-auths">
     <!--
     Sample CUA distribution settings.
     - Central SAP system is ADMCLNT100
     - Child SAP systems are: ADMCLNT500 and HR1CLNT100
     - Driver’s logical system is DRVCLNT100
     -->
        <add-attr attr-name="DirXML-sapLocRoles">
```

```
           <!--
         In a CUA environment, set driver’s LS name with a
                blank role. This allows the driver to receive events
                from SAP.
        -->
             <value>
                    <xsl:value-of select="’DRVCLNT100:’"/>
             </value>
             <!--
             Setting the target LS name with a blank CUA role
                     allows the User object to log on to the target
                     child system but receive no rights.
             -->
            <value>
                    <xsl:value-of select="’ADMCLNT100:’"/>
            </value>
            <!--
             The third value shows how to set a ’real’ CUA role
                    for a child logical system.  This causes
                    distribution from the Central system to the child
                    system and sets the Employee Self-Service role.
            -->
            <value>
                   <xsl:value-of
select="’HRCLNT100:SAP_ESSUSER’"/>
            </value>
      </add-attr>
  <!--
  Example of setting a ’real’ CUA profile.
  -->
      <add-attr attr-name="DirXML-sapLocProfiles">
             <value>
                    <xsl:value-of select="’ADMCLNT500:SAP_ALL’"/>
             </value>
      </add-attr>
</xsl:template>
```

8.4 User Classification Settings (Licensing)

Beginning with version SAP R/3 version 4.7, SAP added the ability to set licensing information on User records. In a CUA environment, this information is set using table UCLASSSYS. There can be a maximum of 1 license type set for each client system in the CUA landscape. The primary data field for licensing in the LIC_TYPE field. This is a two-character code indicating the type of license utilized by the SAP User. Because the license is a system dependent value, you must also set the RCVSYSTEM field to a valid logical system name. You can only set a license value for logical systems specified in the *Systems* tab of the User record. It is not necessary or possible to set license values for the driver’s logical system. The following example shows a Create Stylesheet template for the setting of a sample Employee license value for a User of logical system ADMCLNT100. (In Identity Manager 3, this policy is implemented through Policy Builder.)

```
<xsl:template name="add-license">
     <!--
     - Sample Setting of User Classification (License) Table UCLASSSYS
     - Central SAP system is ADMCLNT100, License Type = 54
     -->
   <add-attr attr-name="DirXML-sapLocUClass">
```

```
            <value>
                   <xsl:value-of select="’ADMCLNT100:52’"/>
            </value>
     </add-attr>
</xsl:template>
```

NOTE: The data sent to the driver must be in a structured format. The default Input Transformation and Output Transformation policies handle the required syntax conversions of UCLASSSYS similar to the way they handle LOCPROFILES and LOCACTIVITYGROUPS.

8.5 Important CUA Integration Notes

- The BAPIs utilized to perform the CUA integration are documented as being available for SAP version 4.0A in the SAP system documentation. Novell® has successfully tested this functionality for SAP R/3 version 4.6C and later. This includes all versions of SAP Web Application Server. For 4.6C systems, the BAPIs are not documented by SAP in the system documentation and support might not be available.
- Password distribution to the CUA Central system can be performed for all initial set and reset operations. However, passwords provisioned to CUA Child systems from the Central system can only be initially set. Password change/reset operations cannot be distributed to Child systems. This is a SAP-designed restriction and is not a limitation of the methodology used by the driver. SAP has determined that setting a single password across systems via CUA violates client system administrative authority and security, so they recommend the use of Single Sign-On (SSO) products to perform this task. Refer to SAP’s documentation related to Password Change for more explicit information on this restriction.
- User Classification (Licensing) table entries may only be made to systems listed in the *Systems* tab on the User record. If Central Licensing values are to be set while adding Users to the CUA Central System, make sure all targeted client systems are also available by setting a DirXML-sapLocRoles or DirXML-sapLocProfiles value for them in the Add event.

Managing the Driver

9

As you work with the SAP User 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

Because these tasks, as well as several others, are common to all Identity Manager drivers, they are included in one reference, the *Identity Manager 3.6 Common Driver Administration Guide*.

Troubleshooting the Driver

10

The following sections contain potential problems and error codes you might encounter while configuring or using the driver.

10.1 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 3.6 Common Driver Administration Guide*.

10.2 Driver Errors

You might see the following driver errors in the DSTrace utility. An explanation of the error is given along with recommended solutions.

- Section 10.2.16, “JCO Installation and Configuration Errors,” on page 73
- Section 10.2.17, “Error When Mapping Drives to the IDoc Directory,” on page 73

10.2.1 java.lang.ClassNotFoundException:com.novell.nds.dirxml.driver.sapusershim.SAPDriver Shim

This is a fatal error that occurs when `sapusershim.jar` is not installed properly. Ensure that the file is in the proper location for either a local or Remote Loader configuration.

This error also occurs when the class name for the `sapusershim.jar` is incorrect. You should ensure that the Java class name is set on the Driver Module tab in a local installation and that the -class parameter is set in a Remote Loader configuration. See Section A.1.1, “Driver Module,” on page 75.

The class name is `com.novell.nds.dirxml.driver.sapusershim.SAPDriverShim`.

10.2.2 com/sap/mw/jco/JCO

This error occurs when the SAP Java Connector `jco.jar` file or `sapjco.jar` or the JCO native support libraries are not present or are improperly located.

Make sure the proper platform version of `jco.jar` or `sapjco.jar` is located in the same directory as `sapusershim.jar.`

Also check the JCO native support libraries to make sure they are present and properly configured. Use the JCO installation instructions for the appropriate platform.

10.2.3 no jRFC12 in java.library.path

This error occurs when the SAP Java Connector (JCO) native RFC12 support library is not present or is located improperly. Make sure the JCO native support libraries are present and configured properly. Use the JCO installation instructions for the appropriate platform.

10.2.4 /usr/jdk1.3.1/lib/sparc/libjRFC12.so:<classpath info>:fatal librfccm.so:open failed: No such file or directory

This error occurs when the SAP Java Connector (JCO) native RFC support library `librfccm.so` is not present or is improperly located. This sample error is from a Solaris* system.

Make sure the JCO native support libraries are present and properly configured. Follow the JCO installation instructions for the appropriate platform.

10.2.5 com.novell.nds.dirxml.engine.VRDException

This error occurs when the SAP Java Connector (JCO) components cannot be located.This error generally occurs if the driver or Remote Loader has not been restarted after the JCO has been configured. Restart Novell® eDirectory™ if you are using a local configuration or restart the remote loader for a remote configuration.

10.2.6 Error connecting to SAP host

This error occurs when the SAP authentication or connection information is not configured properly. Ensure that the values for Authentication and Driver Parameters are correct for authentication to the SAP host system.

10.2.7 nsap-pub-directory parameter is not a directory

This error occurs when the Publisher IDoc Directory parameter in the Publisher Settings of the Driver Parameters does not specify a valid file system location. Ensure that this parameter specifies the directory on the SAP system configured in the SAP ALE subsystem for IDoc file output.

10.2.8 No connection to remote loader

This error occurs when the Remote Loader connection parameter information is incorrect. Configure the proper connection information for the remote connection to the system where the Remote Loader is running.

10.2.9 Authentication handshake failed, Remote Loader message: "Invalid loader password."

This error occurs when the Remote Loader password configured on the remote system does not match the Remote Loader password on the Driver object.

Set matching passwords for both remote loaders. In iManager, ensure that both the application password and Remote Loader passwords are set at the same time.

10.2.10 Authentication handshake failed: Received invalid driver object password

This error occurs when the driver password configured on the remote system does not match the Driver object password on the Driver object. To correct this, you should set both Driver object passwords identically.

10.2.11 IDoc File or IDoc TRFC Documents Not Generated when a SAP User Is Created or Modified

You should ensure that the ALE and CUA processes are configured properly, and that you have correctly entered the data.

User data is distributed to the driver only if CUA has been properly configured and if the logical system representing the driver has been selected for distribution under the Systems tab in the SAP User Maintenance dialog box.

10.2.12 Users Created in SAP Cannot Log On to the SAP System (CUA in Use)

When creating users in the CUA central system, you must associate User objects with the client systems to which they authenticate. This occurs in the default policies when you set a value for the driver's logical system in the DirXML-sapLocRoles or DirXML-sapLocProfiles attribute.

10.2.13 The Driver Does Not Recognize IDocs in the Directory

You should first test the ALE and CUA interface. Refer to your SAP documentation for more information.

If the IDoc interface fails:

- Using transaction WE21, ensure that the file port is configured properly. You should validate the path to the directory and make sure the Transfer IDoc Immediately radio button is selected.
- Using transaction WE20, ensure that the appropriate file port is selected in the Partner Profile. Also, verify that it is on the outbound parameters of the receiving system.

If the IDoc interface succeeds:

- Ensure that the correct distribution model has been selected using transaction SCUA.
- Ensure that the proper User field data distribution is configured using transaction SCUM.

10.2.14 IDocs Are Not Written to the Driver (TRFC Port Configuration)

You should first test the ALE and CUA interface. Refer to your SAP documentation for more information.

If the IDoc distribution succeeds but data is not received:

- Verify that the driver is configured to receive data from the correct SAP Gateway.
- Verify that the driver Program ID is unique.
- Using transaction WE21, verify that the SAP port configuration is configured to distribute to the logical system representing the driver.

If the IDoc interface succeeds:

- Ensure that the correct distribution model has been selected using transaction SCUA.
- Ensure that the proper User field data distribution is configured using transaction SCUM.

10.2.15 The Driver Does Not Authenticate to SAP

You should first ensure that you have configured all of the driver parameters and that the proper passwords have been entered. If the SAP system is the central system of a CUA configuration, make sure the User object used for authentication is properly associated with the client logical system. See "Users Created in SAP Cannot Log On to the SAP System (CUA in Use)" on page 72.

If you are running the driver remotely, make sure that the Remote Loader has been started before you start the driver.

10.2.16 JCO Installation and Configuration Errors

For detailed instructions on using the JCO Test utility and analyzing error messages, refer to "Testing the SAP JCO Client Connection" on page 27.

10.2.17 Error When Mapping Drives to the IDoc Directory

You might see the following error in DSTrace if the IDoc directory parameter specifies an invalid local file system container or if it specifies a mapped drive on a remote system.

```
*** NDS Trace Utility - BEGIN Logging *** Fri Sep 13 15:45:59 2005

DirXML Log Event -------------------
    Driver = \FLIBBLE_TREE\n\Driver Set\SAP-UM
    Channel = publisher
    Status = fatal
    Message = <description>SAP Document Poller initialization failed:
com.novell.nds.dirxml.driver.sapusershim.SAPDocumentPollerInitFailure: Specified
Publisher IDoc Directory is invalid.</description>

*** NDS Trace Utility - END   Logging *** Fri Sep 13 15:46:31 2005
```

This error occurs because the Windows operating system service controls the rights of the local system, not the rights of a user. Thus, the local Windows system does not have rights to access any file resources outside of its own system, including the IDoc directory.

Driver Properties

A

This section provides information about the Driver Configuration and Global Configuration Values properties for the SAP User driver. These are the only unique properties for this 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 3.6 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, it is marked with an icon.

- Section A.1, "Driver Configuration," on page 75
- Section A.2, "Global Configuration Values," on page 80

A.1 Driver Configuration

In Designer:

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

In iManager:

1. In iManager, click to display the Identity Manager Administration page.
2. Open the driver set that contains the driver whose properties you want to edit. To do so:
   - **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 Sentinel 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.

   By default, the properties page opens with the Driver Configuration tab displayed.

The Driver Configuration options are divided into the following sections:

- Section A.1.1, "Driver Module," on page 75
- Section A.1.2, "Driver Object Password (iManager Only)," on page 76
- Section A.1.3, "Authentication," on page 76
- Section A.1.4, "Startup Option," on page 77
- Section A.1.5, "Driver Parameters," on page 78

A.1.1 Driver Module

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

Table A-1 *Driver Modules*

Option	Description
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.sapusershim.SAPDriverShim`
Native	This option is not used with the SAP User driver.
Connect to Remote Loader	Used when the driver is connecting remotely to the connected system. Designer includes two suboptions: ◆ *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. ◆ *Remote Loader Client Configuration for Documentation*: Includes information on the Remote Loader client configuration when Designer generates documentation for the Delimited Text driver.

A.1.2 Driver Object Password (iManager Only)

Table A-2 *Driver Object Password*

Option	Description
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 or the remote driver does not run. This password is used by the Remote Loader to authenticate itself to the remote driver shim.

A.1.3 Authentication

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

Table A-3 *Authentication Options*

Option	Description
Authentication ID	Specify an SAP account that the driver can use to authenticate to the SAP system. Example: `SAPUser`

Option	Description
Authentication Context or *Connection Information*	Specify the IP address or name of the SAP server the driver should communicate with.
Remote Loader Connection Parameters or *Host name* Port *KMO* *Other parameters*	Used only if the driver is connecting to the application through the Remote Loader. The parameter to enter is `hostname=xxx.xxx.xxx.xxx port=xxxx kmo=certificatename`, when the host name is the IP address of the application server running the Remote Loader server and the port is the port the remote loader is listening on. The default port for the Remote Loader is 8090. The `kmo` entry is optional. It is only used when there is an SSL connection between the Remote Loader and the Metadirectory engine. Example: `hostname=10.0.0.1 port=8090 kmo=IDMCertificate`
Driver Cache Limit (kilobytes) or *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.
Application Password or *Set Password*	Specify the password for the user object listed in the *Authentication ID* field.
Remote Loader Password or *Set 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.

A.1.4 Startup Option

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

Table A-4 *Startup Options*

Option	Description
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 only applies 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 presented by category:

- Table A-5, "Driver Settings," on page 78
- Table A-6, "Subscriber Settings," on page 79
- Table A-7, "Publisher Settings," on page 79

Table A-5 *Driver Settings*

Parameter	Description
SAP System Number	Specify the SAP system number of the SAP application server. This is referred to as the *System Number* in the SAP logon properties. The default value is 00.
SAP User Client Number	Specify the client number to be used on the SAP application server. This is referred to as the *Client* in the SAP logon screen.
SAP User Language	Specify the language code this driver will use for the SAP session. This is referred to as the *Language* in the SAP logon screen.
Character Set Encoding	The code for the character set to translate IDoc byte-string data into Unicode* strings. An empty value causes the driver to use the host JVM* default.
Publish all Communication Table Values	Set this to *Publish Primary* if only the primary value of Communicate tables should be synchronized. or Set this to *Publish All* if all values should be synchronized.
Publish Company Address Data	By default, an SAP User record does not include Company Address information. That data is kept in a related table. Use this parameter to specify if you want the driver to retrieve the data from the appropriate company record. Regardless of the option you specify, Company Address information cannot be updated in SAP. Set this to *Include Company Address* to populate User Company Address information for the Publisher and Subscriber channel queries. or Set this to *Ignore Company Address* if you do not want this functionality. For additional information, see Section 7.3, "Obtaining Company Address Data for User Objects," on page 57.

***Table A-6** Subscriber Settings*

Parameter	Description
Communication Table Comments	The communication table comment is a text comment the driver adds to all Communication table entries added by the Subscriber channel. This is a useful method for determining where an entry originated from when viewing values via the SAP GUI. Leaving this field blank provides no comment to the table entries.
Require User to Change Set Passwords	This parameter specifies the methodology used by the driver to set User account passwords. Passwords can be set by the driver's administrative User account or by the affected User's account (this sets a password on new accounts or modifies passwords for existing Users.) Select *Change Required* if passwords must be changed immediately at the user's next login. or Select *No Change Required* if you do not want user's to change passwords immediately at login.
Password Set Method (Conditional)	If you select the *No Change Required* option above, you should specify a Password Set Method: *Administrator Set* or *User Set.* **Administrator Set:** Passwords are set by the driver's administrative User account. This method is deprecated and does not comply with SAP security best practices. The method works only for SAP systems that are version 4.6c or older. **User Set:** Passwords are supplied by the affected users. The following parameters must be set if you select User Set: ◆ **Default Reset Password:** This parameter specifies a default password reset value. It is set during password changes if the User-supplied password is not accepted by the SAP server. There is an 8-character size limit for this value. (SAP 7.0 does not require an 8-character size limit on passwords.) ◆ **Force Passwords to Uppercase:** This option defines if passwords are forced to uppercase. Uppercase is the default value, however, SAP 7.0 allows for mixed-case passwords.

***Table A-7** Publisher Settings*

Parameter	Description
Publisher Channel Enabled	Select whether or not you want to enable the driver's Publisher channel.
Publisher Channel Port Type	Set to TRFC if the driver will instantiate a JCO Server to receive data distribution broadcasts from the SAP ALE system. Set to FILE if the driver will consume text file IDocs distributed by the SAP ALE system.
Poll Interval (seconds)	Specify how often the Publisher channel polls for unprocessed IDocs. The default value is 10 seconds.

Parameter	Description
Future-dated Event Handling Option	The behavior of this option is based on the values of the User record's Logon Data "Valid From" date (LOGONDATA:GLTGV) when IDocs are processed by the Publisher channel. This field does not need to be in the Publisher filter for this processing to occur. Choose one of the following options: ◆ **Publish Immediately:** Indicates that all attributes are processed by the driver when the IDoc is available. No future-dated processing is performed. ◆ **Publish on Future Date:** Indicates that only attributes that have a current or past time stamp are processed by the driver when the IDoc is available. Future-dated infotype attributes are cached in a `.futr` file to be processed at a future date. ◆ **Publish Immediately and on Future Date:** Indicates that the driver blends the first two options. All attributes with a current or past time stamp are processed at the time the IDoc is available. All future-dated infotype attributes are cached in a `.futr` file to be processed at a future date. ◆ **Publish Immediately and Daily through Future Date:** Indicates that the driver processes all events at the time the IDoc is made available. All future-dated infotype attributes are cached in a `.futr` file to be processed again on the next calendar day. This continues until the attributes are sent for a final time on the future date.
Publisher IDoc Directory	Specify the file system location where the SAP User IDoc files are placed by the SAP ALE system (FILE port configuration) or by the driver (TRFC configuration.) This setting is only used if the Publisher channel is enabled.
Publisher Heartbeat Interval	Specify how many minutes of inactivity can 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.

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 SAP HR driver includes several predefined GCV's. 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 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*.

Table A-8 *Global Configuration Values*

Option	Description
Application accepts passwords from Identity Manager	If *True*, allows passwords to flow from the Identity Vault to the SAP system. In Designer, you must click the icon next to an option to edit it. This displays the Password Synchronization Options dialog that better shows 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 better shows the relationship between the different GCVs. For more information about how to use the Password Management GCVs, see "Configuring Password Flow " in the *Identity Manager 3.6 Password Management Guide*.
Identity Manager accepts passwords from application	If *True*, allows passwords to flow from the SAP system to the Identity Vault.
Publish passwords to NDS password	Use the password from the SAP system to set the non-reversible NDS® password in the Identity Vault.
Publish passwords to Distribution Password	Use the password from the SAP 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 Identity Vault 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 SAP system using the Distribution Password from the Identity Vault.

Option	Description
Notify the user of password synchronization failure via e-mail	If True, notify the user by e-mail of any password synchronization failures.
Connected System or Driver Name	The name of the connected system, application or Identity Manager driver. This value is used by the e-mail notification templates.

Application Link Enabling (ALE)

B

Application Link Enabling (ALE) technology enables communication between SAP and external systems such as the Identity Vault (eDirectory). ALE is comprised of various components. If you want to distribute User modification data automatically from the SAP system to the Identity Vault, you must configure the ALE and CUA systems. If your integration requires only reading and writing data to the SAP system, this configuration is not necessary.

When configuring the SAP system to enable the driver, you should consider the following ALE components and their relationship to the driver:

- Section B.1, “Clients and Logical Systems,” on page 83
- Section B.2, “Message Type,” on page 83
- Section B.3, “IDoc Type,” on page 84
- Section B.4, “Distribution Model,” on page 84
- Section B.5, “Partner Profiles,” on page 84
- Section B.6, “Port,” on page 84
- Section B.7, “Port Definition,” on page 84
- Section B.8, “File Port,” on page 84
- Section B.9, “TRFC Port,” on page 85
- Section B.10, “CUA,” on page 85

Refer to “Configuring the SAP System” on page 21 for instructions on how to configure these SAP system parameters.

B.1 Clients and Logical Systems

In the SAP configuration for the driver, a logical system is a representation of either a SAP system or an external system. The logical system is used to distribute data to and from SAP. To use ALE, every SAP system needs to have a base logical system associated with a client. There is a one-to-one relationship between the client and the logical system.

The driver uses an outbound ALE interface. In an outbound ALE interface, the base logical system becomes the *sender* for outbound messages and the *receiver* of inbound messages. A SAP user is likely logged into the base logical system/client when making changes to the database (for example, modifying User profiles or logon preferences). A logical system/client must also be defined for the receiving client. This logical system acts as the receiver of outbound messages.

B.2 Message Type

A message type represents the type of data that is exchanged between the two systems. For this driver, the USERCLONE message type is used. A message type characterizes data being sent across the systems and relates to the structure of the data, also known as an IDoc type (for example, USERCLONE03).

B.3 IDoc Type

Intermediate Document (IDoc) Type represents the structure of the data associated with a message type. ALE technology uses IDocs to exchange data between logical systems. An IDoc is an object with the data of a specific message type in it. IDocs consist of three record types:

- The control record
- The data record
- The status record

The control record contains information about the IDoc, such as what IDoc type it is, the message type, the sending and receiving systems, or the direction.

The data record contains the application data. Data records consist of several fields that describe the content of the specific object.

The status record contains data on the state of the processing of the IDoc.

B.4 Distribution Model

The distribution model is a tool that stores information about the flow of message types between systems. A distribution model must be configured when setting up the driver. After the two logical systems have been defined and you have a general understanding of message types and IDocs, you can configure your distribution model.

The distribution model determines what message types can be sent from a logical system to another logical system.

B.5 Partner Profiles

Partner profiles specify the components used in an outbound process. Some of these components include the IDoc type, message type, IDoc size, mode, and the person to be notified in case of errors.

B.6 Port

A port is the communication link between the two logical systems.

B.7 Port Definition

A port definition is used in an outbound process to define how documents are transferred to the destination system.

B.8 File Port

A file port can be used in the integration solution. IDocs are transferred to a file in a specified file system location accessible by the SAP host system.

B.9 TRFC Port

A Transactional Remote Function Call (TRFC) can be used in the integration solution. IDocs are transferred to a specified application process (such as the driver) via the SAP Gateway.

B.10 CUA

Central User Administration (CUA) is a process provided by SAP to distribute and manage User object data between a Central SAP logical system and one or more Client logical systems. The client logical systems might be SAP or external systems. The base technology used for the CUA is ALE.

Business Application Programming Interfaces (BAPIs)

C

Table C-1 contains a list of the BAPIs used by the driver.

***Table C-1** Driver BAPIs*

BAPI Name	Description
BAPI_PDOTYPES_GET_DETAILEDLIST	Used to obtain lists and minimal detailed information for SAP USER objects and other specified business object types.
BAPI_USER_ACTGROUPS_ASSIGN	Used to assign the Activity Groups (Roles) to SAP USER objects in a non-CUA landscape.
BAPI_USER_ACTGROUPS_DELETE	Used to delete the Activity Groups (Roles) from SAP USER objects in a non-CUA landscape.
BAPI_USER_PROFILES_ASSIGN	Used to assign Profiles to SAP USER objects in a non-CUA landscape.
BAPI_USER_PROFILES_DELETE	Used to delete Profiles from SAP USER objects in a non-CUA landscape.
BAPI_USER_CHANGE	Used to modify SAP USER object attributes (fields, structures, and general tables) and non-persistent passwords.
BAPI_USER_CREATE1	Used to create a new SAP USER object.
BAPI_USER_DELETE	Used to delete an SAP USER object.
BAPI_USER_GETDETAIL	Used to read the current data field values, structures, and general table attributes of an SAP USER object.
BAPI_ADDRESSORG_GETDETAIL	Used to read the Company Address attributes of an SAP USER object.
BAPI_USER_LOCK	Used to lock an SAP USER object account. On a CUA Central system this is a global lock. On a CUA Child system or on a non-CUA system, this is a local lock.
BAPI_USER_UNLOCK	Used to unlock an SAP USER object account. On a CUA Central system this is a global lock. On a CUA Child system or on a non-CUA system this is a local lock.

BAPI Name	Description
SUSR_BAPI_USER_LOCK	Used to set granular locks on an SAP USER object account. The granular lock types available are `LOCK_LOCAL` and `LOCK_GLOBAL`. By default, this BAPI is not a Remote-Enabled Module and is thus not accessible to the driver. An ABAP developer must modify the Processing Type attribute of the BAPI on the target SAP system. This is accomplished via the SAP Function Builder (Transaction SE37).
SUSR_BAPI_USER_UNLOCK	Used to clear granular locks on an SAP USER object account. The granular lock types available are `LOCK_LOCAL`, `LOCK_GLOBAL`, and `LOCK_WRONG_LOGON`. By default, this BAPI is not a Remote-Enabled Module and is thus not accessible to the driver. An ABAP developer must modify the Processing Type attribute of the BAPI on the target SAP system. This is accomplished via the SAP Function Builder (Transaction SE37).
SUSR_USER_CHANGE_PASSWORD_RFC	Used to set a persistent password for an SAP USER object.
BAPI_USER_LOCACTGROUPS_ASSIGN	Used to assign client-specific Activity Groups (Roles) to SAP USER objects in a CUA landscape.
BAPI_USER_LOCACTGROUPS_READ	Used to read the current client-specific Activity Groups (Roles) assignments of SAP USER objects in a CUA landscape.
BAPI_USER_LOCACTGROUPS_DELETE	Used to delete the client-specific Activity Groups (Roles) assignments from SAP USER objects in a CUA landscape.
BAPI_USER_LOCPROFILES_ASSIGN	Used to assign client-specific Profiles to SAP USER objects in a CUA landscape.
BAPI_USER_LOCPROFILES_READ	Used to read the current client-specific Profile assignments of SAP USER objects in a CUA landscape.
BAPI_USER_LOCPROFILES_DELETE	Used to delete the client-specific Profile assignments from SAP USER objects in a CUA landscape.
BAPI_USER_CLONE	Sent from the SAP ALE subsystem to communicate SAP-initiated changes of USER objects to the driver Publisher channel.
BAPI_COMPANY_CLONE	Sent from the SAP ALE subsystem to communicate SAP-initiated changes of Company Address information to the driver Publisher channel.

Configuration and Deployment Notes

D

The following information can be valuable when modifying the driver configuration or when trying to understand SAP system behavior. Many of these notes relate to data value restrictions on the User record. You should investigate the system configuration thoroughly, because some values might have been modified or extended by the SAP administrator.

D.1 SAP Object Types

The following SAP object types of interest might be referenced in <query> operations to SAP.

Table D-1 *SAP Object Types*

USER	Object Type: US
Activity Groups	Object Type: AG
Standard Roles	Object Type: AC
Company	Object Type: U
User Groups	Object Type: UG

D.2 User Types: LOGONDATA:USTYP

- A - Dialog
- C - Communication (CPIC)
- B - System (BDC)
- S - Service
- L - Reference

D.3 Output Controller Options

***Table D-2** Output Controller Options*

G - Output immediately	DEFAULTS: SPDB
H - Don't output immediately	DEFAULTS: SPDB
D - Delete after output	DEFAULTS: SPDA
K - Don't delete after output	DEFAULTS: SPDA

D.4 Communication Types: ADDCOMREM:COMM TYPE

- INT - EMail Address type (SMTP)
- LET - Letter (Standard Post)
- PAG - Pager
- FAX - Facsimile
- PRT - Printer
- RML - Remote Mail
- TEL - Telephone
- TLX - Telex
- TTX - Teletex
- SSF - Secure Store and Forward

D.5 Date Formats: DEFAULTS:DATAFM

1. DD.MM.YYYY
2. MM/DD/YYYY
3. MM-DD-YYYY
4. YYYY.MM.DD
5. YYYY/MM/DD
6. YYYY-MM-DD

D.6 Decimal Formats: DEFAULTS:DCPFM

- "X" - The decimal divider is a dot, and the thousands divider is a comma (NN,NNN.NN)
- "Y" - The decimal divider is a comma, and the thousands divider is a blank (NNN NNN,NN)
- " " - The decimal divider is a comma, and the thousands divider is a dot (NN.NNN,NN)

D.7 Computer Aided Test (CATT): DEFAULTS:CATTKENNZ

- “X” - CATT: Test status set
- “ ” - CATT: Test status not set
- “.” - CATT: CATT status set

D.8 Communication Comment Type to Table Mappings

Table D-3 *Communication Comment Type to Table Mappings*

Table: ADDTEL	Comment Type: TEL	Key Field: TELEPHONE
Table: ADDFAX	Comment Type: FAX	Key Field: FAX
Table: ADDPAG	Comment Type: PAG	Key Field: PAGER
Table: ADDSMTP	Comment Type: INT	Key Field: E_MAIL
Table: ADDTTX	Comment Type: TTX	Key Field: TELETEX
Table: ADDPRT	Comment Type: PRT	Key Field: PRINT_DEST
Table: ADDTLX	Comment Type: TLX	Key Field: TELEX_NO
Table: ADDRML	Comment Type: RML	Key Field: R_MAIL
Table: ADDURI	Comment Type: URI	Key Field: URI

D.9 Language Codes

Language	Two-Letter Code	One-Letter Code
Afrikaans	AF	a
Arabic	AR	A
Bulgarian	BG	W
Czech	CS	C
Danish	DA	K
German	DE	D
Greek	EL	G
English	EN	E
Spanish	ES	S
Estonian	ET	9
Finnish	FI	U

Language	Two-Letter Code	One-Letter Code
French	FR	F
Hebrew	HE	B
Croatian	HR	6
Hungarian	HU	H
Indonesian	ID	i
Italian	IT	I
Japanese	JA	J
Korean	KO	3
Lithuanian	LT	X
Latvian	LV	Y
Malaysian	MS	7
Dutch	NL	N
Norwegian	NO	O
Polish	PL	L
Portuguese	PT	P
Romanian	RO	4
Russian	RU	R
Slovak	SK	Q
Slovene	SL	5
Serbian	SR	0 (zero)
Swedish	SV	V
Thai	TH	2
Turkish	TR	T
Ukrainian	UK	8
Customer Reserve	Z1	Z
Chinese Traditional	ZF	M
Chinese	ZH	1

D.10 Configuration Parameters

Comment text for configuration parameters is limited to a maximum length of 50 bytes.

D.11 Design Comments and Notes

When specifying either USER or COMPANY names in BAPI calls, the name field must be in all-caps format, even if the naming field is not specified as such.

NOTE: The ADMIN_SET mode is deprecated prior to R/3 4.7. You need to USER_SET mode with SAP 4.7 and above.

In BAPI_USER_CHANGE (ADDRESS table)

- The COMM-TYPE attribute in SAP has defined, acceptable values. Invalid input generates an exception and an error message stating, "The communication type <commType> is not defined." Valid fields are the abbreviations for the supported communication types on the SAP Host.
- The TITLE_ACA1 and TITLE_ACA2 attributes have predefined, acceptable values. Invalid input results in the value in SAP being set to a null string (" ").
- The PREFIX1 and PREFIX2 attributes have predefined, acceptable values. Invalid input results in the value in SAP being set to a null string (" ").
- The TEL1_NUMBR is linked to the primary, or Standard, Telephone number in the Telephone communication table.

In BAPI_USER_CHANGE (ADDFAX table)

- The Facsimile Telephone Number attribute in the Identity Vault is a structured attribute. An output transformation converts it to a single attribute format.

In BAPI_USER_CHANGE (ADDTEL table)

- Must have a CONSNUMBER (either the number of the one you wish to change or a new, non-000 number.)
- The STD_NO field must be set to X if you are synchronizing a single field or if the number is the only number present.
- The primary data field is TELEPHONE.

In BAPI_USER_CHANGE (ADDTLX table)

- By default, this table is mapped to the Organizational Person; telexNumber attribute. This syntax is OCTET_STRING, which is encoded by Identity Manager into Base64 string encoding. A Java function is provided in the driver `sapusershim.jar` file that can decode this into the proper string format in the Output Transformation prior to submission to SAP. If you are using the driver on a remote system, place the driver shim in the same file system container with the Identity Manager library in the Input Transformation for the Publisher channel.
- The primary data field is TELEX_NO.
- Other rules apply as described for the ADDTEL table.

In BAPI_USER_CHANGE (ADDFAX table)

- The primary data field is FAX.
- Other rules apply as described for the ADDTEL table.

In BAPI_USER_CHANGE (GROUPS table)

- The USERGROUP is the only field in this table.

In BAPI_USER_CHANGE (ALIAS structure)

- The USERALIAS is the only field in this table.
- The SAP system guarantees that alias names are unique among all users. If an alias value is already assigned to another user, the modification fails.

In BAPI_USER_CHANGE (REF_USER structure)

- The REF_USER is the only field in this table.
- The value specified as REF_USER must be an existing User object on the SAP client, and the Reference User's type flag must be set to Reference (User Type L)

In BAPI_USER_CHANGE (DEFAULTS structure)

- The SPDB field can only be populated with a G (GO or Output Immediately), or an H (Hold output), or a null string " ", which sets the value to H. All other values generate an error message. This field is case sensitive.
- The SPDA field can only be populated with a D (Delete after print), or a K (Keep), or a null string " ", which sets the value to K. All other values generate an error message. This field is case sensitive.
- The KOSTL (Cost center) field is automatically truncated to 8 bytes by the SAP system.
- The SPLG field does not appear to be utilized at all. Any value is accepted but does not relate to any attribute shown in the SAP GUI.
- The START_MENU field can be set to any value up to 30 characters whether or not a valid menu exists for the value being set.
- The SPLD (Output Controller) field accepts only a null string value (" ") or a valid output device that is available via the SAP GUI drop-down list for this field. Invalid selections return an error.
- The LANGU field must be set to one of the one-letter language codes defined in Section D.9, "Language Codes," on page 91 or to a null string (" "). The null string defaults to the language of the SAP system default language. This field is case sensitive. Non-defined fields result in an error.

In BAPI_USER_CHANGE (LOGONDATA structure)

- The USTYP field only accepts the valid User Types defined in Section D.2, "User Types: LOGONDATA:USTYP," on page 89 or a null string (" "). Other input generates an exception and error message stating "Invalid user type<type>."

- The TZONE field accepts only valid, selectable fields from the SAP GUI drop-down list. Invalid input generates an exception and an error message stating "Invalid time zone." The Time Zone setting is displayed under the Defaults tab in the SAP client Display User dialog box.
- The CLASS field represents the User's User Group for Authorization Check setting. Only fields that are selectable from the SAP GUI drop-down list are accepted. Invalid input generates an exception and error message stating "User group <class> does not exist."
- The GLTGV (Validity Begin Date) and GLTGB (Validity End Date) values exist as a set of data.
- The Begin Date must always be less than the End date.
- Invalid date input generates an exception and an error message stating "Invalid time interval: Begin date after end date."

In BAPI_USER_CHANGE (GROUPS table)

- Only valid groups that exist in the SAP User Groups table can be added to a user. Invalid input generates an exception and an error message stating "User group<name> does not exist."

In BAPI_USER_CHANGE (ADDCOMREM table)

- The LANGU and LANGU_ISO fields are set with the driver's language parameter value.

Example XML Document Received from the Driver

E

The following example is a typical XML document received from the default driver configuration.

```
<nds dtdversion=”1.0” ndsversion=”8.5”>
       <source>
               <product build="20050509_1030" instance="SAP-USER-REMOTE-46C"
version="1.0">Identity
     Manager Driver for User Management of SAP Software</product>
<contact>Novell, Inc.</contact>
       </source>
       <input xmlns:sapshim="http://www.novell.com/dirxml/drivers/sapusershim">
              <modify class-name="US" event-id="O_001_0000000000216097" src-
dn="SSAMPLE"
          timestamp="20030509">
                     <association>USdJSMITH</association>
                   <modify-attr attr-name="PROFILES:BAPIPROF">
                            <remove-all-values/>
                     </modify-attr>
                     <modify-attr attr-name="USERNAME:BAPIBNAME">
                            <remove-all-values/>
                     </modify-attr>
                 <modify-attr attr-name="ACTIVITYGROUPS:AGR_NAME">
                            <remove-all-values/>
                     </modify-attr>
                     <modify-attr attr-name="PROFILES:BAPIPROF">
                            <add-value>
                                   <value>SAP_ALL</value>
                                   <value>SAP_NEW</value>
                            </add-value>
                     </modify-attr>
                       <modify-attr attr-name="USERNAME:BAPIBNAME">
                              <add-value>
                                    <value>JSMITH</value>
                              </add-value>
                       </modify-attr>
                       <modify-attr attr-name="ACTIVITYGROUPS:AGR_NAME">
                              <add-value>
                                      <value>SAP_EMPLOYEE</value>
                              </add-value>
                        </modify-attr>
                  </modify>
          </input>
</nds>
```

Some characteristics to note:

- All XML documents received from the SAP system are translated into <modify> documents. This translation occurs because it is not possible to determine whether the object described by the document has been modified or is new. Additional modification or translation of the document is accomplished through policies and the Metadirectory engine.

- The <modify> element contains the classname of the object described in the SAP namespace (that is, US=User). The event-id attribute contains the IDoc number from which the data is derived. The src-dn attribute contains the SAP Object name value. The timestamp attribute contains the date that the IDoc was processed by the driver.
- The <association> element data always contains the format "USd*SAPobjectID*". User names in SAP are always uppercase.
- The <modify-attr> element contains the attr-name described in SAP format (Structure or Table name:Attribute Name).
- Because multivalue attributes cannot be consistently mapped across systems, the <remove-all-values> element is used prior to all <add-value> tags on Publisher channel documents. This instructs the Metadirectory engine to remove all existing values for the attribute prior to assigning the new values. If this functionality is not desired, one of the policies may be used to modify the document.
- All values are in a string format.
- All values for DirXML-locSapRoles and DirXML-locSapProfiles require that you set two fields in SAP. In order to map from a single string value to a structured format, default policies use a colon ":" delimiter in the Identity Vault values (such as ADMCLNT100:SAP_ESSUSER), which are then transformed to (or from) the SAP structured format. "Schema Mapping Policy" on page 47 indicates the structure components to set for these values.

Structured Format Examples

F

```
// Single value field
//
<modify-attr attr-name="LOCKUSER">
     <add-value>
          <value>1</value>
     </add-value>
</modify-attr>
//
// Single field from Structure
//
<modify-attr attr-name="ADDRESS:E_MAIL">
     <add-value>
          <value>UGRANT@uniongenerals.org</value>
     </add-value>
</modify-attr>
//
// Single field, multi-values from Table
//
<modify-attr attr-name="ACTIVITYGROUPS:AGR_NAME">
     <add-value>
          <value>SAP_ESSUSER</value>
          <value>SAP_EMPLOYEE</value>
     </add-value>
</modify-attr>
//
// All fields, multi-values from Table
//
<modify-attr attr-name="LOCACTIVITYGROUPS">
     <add-value>
          <value type="structured">
                 <component name="FROM_DAT">2005-04-01</component>
                 <component name="TO_DAT">9999-12-31</component>
                 <component name="ORG_FLAG"></component>
                 <component name="AGR_NAME">SAP_ESSUSER</component>
                 <component name="SUBSYSTEM">ADMCLNT500</component>
                 <component name="AGR_TEXT"></component>
          </value>
          <value type="structured">
                 <component name="FROM_DAT">2005-04-01</component>
                 <component name="TO_DAT">9999-12-31</component>
                 <component name="ORG_FLAG"></component>
                 <component name="AGR_NAME">SAP_EMPLOYEE</component>
                 <component name="SUBSYSTEM">ADMCLNT100</component>
                 <component name="AGR_TEXT"></component>
          </value>
     </add-value>
</modify-attr>
```

Setting and Clearing Granular Locks

G

This functionality is available for SAP systems that support the concept of granular locks via the `SUSR_BAPI_USER_LOCK` and `SUSR_BAPI_USER_UNLOCK` functions. These locks relate to the account locking mechanisms that are available from the Central System of an SAP Central User Administration (CUA) environment.

This functionality is only available through the SAP User Management driver if the BAPI functions are configured to be a Remote-Enabled Module. This is done via an attribute setting in the SAP Function Builder transaction (SE37) and must be performed by an authorized administrator.

The driver can set or clear the supported lock types by using two pseudo-attributes called `SETGRANULARLOCKS` and `CLEARGRANULARLOCKS`.

The supported lock types for `SETGRANULARLOCKS` are:

- `LOCK_LOCAL`
- `LOCK_GLOBAL`

The supported lock types for `CLEARGRANULARLOCKS` are:

- `LOCK_LOCAL`
- `LOCK_GLOBAL`
- `LOCK_WRONG_LOGON`

To set or clear a particular lock, simply use a value of `X` or `x` for the desired lock type value. Any unspecified lock type sets to a value of ‘ ’, which implies the lock type is not set or cleared.

NOTE: It is not valid to use these pseudo-attributes in a `<remove-value>` element.

G.1 Examples

```
//
// Example - Set Local Lock on User
//
<modify-attr attr-name="SETGRANULARLOCKS">
  <add-value>
    <value type="structured">
      <component name="LOCK_LOCAL">X</component>
    </value>
  </add-value>
</modify-attr>

//
// Example - Set Local and Global Locks on User
//
<modify-attr attr-name="SETGRANULARLOCKS">
  <add-value>
    <value type="structured">
```

```
      <component name="LOCK_LOCAL">X</component>
      <component name="LOCK_GLOBAL">X</component>
    </value>
  </add-value>
</modify-attr>

//
// Example - Clear Local and Wrong Logon Locks on User
//
<modify-attr attr-name="CLEARGRANULARLOCKS">
  <add-value>
    <value type="structured">
      <component name="LOCK_LOCAL">X</component>
      <component name="LOCK_WRONG_LOGON">X</component>
    </value>
  </add-value>
</modify-attr>
```

Using Wildcard Search Capabilities

H

Releases of this driver prior to version 1.0.5 had issues related to the implementation of the default Subscriber Matching policy. This policy issues a query to the SAP server for matches of the “Given Name” and “Surname” attributes (mapped to ADDRESS:FIRSTNAME and ADDRESS:LASTNAME) prior to processing the creation of a new User object. The following XDS query illustrates the output of this policy.

```
<nds dtdversion="1.1" ndsversion="8.6">
   <source>
      <product version="1.1.2">DirXML</product>
         <contact>Novell, Inc.</contact>
   </source>
   <input>
      <query class-name="US" event-id="0">
         <search-class class-name="US"/>
         <search-attr attr-name="ADDRESS:LASTNAME">
            <value timestamp="1114621366#3" type="string">Smith</value>
         </search-attr>
         <search-attr attr-name="ADDRESS:FIRSTNAME">
            <value timestamp="1114621375#1" type="string">Joe</value>
         </search-attr>
         <read-attr/>
      </query>
   </input>
</nds>
```

This is a problem because SAP does not provide the capability to Search for a User account based on attribute values. Therefore, the driver needs to obtain a list of all User objects, then read each object, compare their FIRSTNAME and LASTNAME attributes to the search values, and return a list of matching objects. In a database with hundreds or thousands of User objects, this process takes a very long time.

To alleviate this problem, starting with version 1.0.5, the driver now has the capability to use a wildcard syntax for queries that contain the User name field (USERNAME:BAPIBNAME). This allows you to write policies that take advantage of the known account naming policies of the SAP system to reduce the number of objects that need to be read and compared during matching operations.

For example, the default Subscriber Create rule uses the first initial of the Given Name attribute value appended with the Surname attribute value to create a proposed account name. A new User with Given Name “John” and Surname “Smith” generates a proposed SAP User account name of JSMITH. Any duplicates of this proposed name are appended with numeric values (ie. JSMITH1, JSMITH2, etc.) The default Output Transformation policy now contains a template that takes advantage of the USERNAME:BAPIBNAME wildcard capabilities of the driver and appends this additional search attribute to the query. When the driver receives a query containing a USERNAME:BAPIBNAME search attribute, it determines if the value is a wildcard or a literal value. Any value that is contained within single-quote characters is evaluated for wildcard syntax. If the single quote characters do not exist, the driver attempts to read the specified User object.

The supported variations of the wildcard syntax are:

- “Starts-with” syntax (ie. JSmith*) - This restricts attribute matching to User account names starting with JSMITH.
- “Ends-with” syntax (ie. *ith) - This restricts attribute matching to User account names ending with ITH.
- “Contains” syntax (ie. *SMIT*) - This restricts attribute matching to User account names containing SMIT.

When the list of objects to be matched has been restricted, the remaining search attributes are used to determine a match.

The output from the default Output Transform policy converts the Matching Rule query shown above to the following query. It should be noted that this policy will only be applied to queries that do not already contain a USERNAME:BAPIBNAME search attribute.

```
<nds dtdversion="1.1" ndsversion="8.6">
   <source>
      <product version="1.1.2">DirXML</product>
      <contact>Novell, Inc.</contact>
   </source>
   <input>
      <query class-name="US" event-id="0">
         <search-class class-name="US"/>
         <search-attr attr-name="ADDRESS:LASTNAME">
            <value timestamp="1114621366#3" type="string">Smith</value>
         </search-attr>
         <search-attr attr-name="ADDRESS:FIRSTNAME">
            <value timestamp="1114621375#1" type="string">Joe</value>
         </search-attr>
         <read-attr/>
         <search-attr attr-name="USERNAME:BAPIBNAME">
            <value>’JSmith*’</value>
         </search-attr>
      </query>
   </input>
</nds>
```

With this query, the driver searches only User objects whose name starts with JSMITH for matching ADDRESS:LASTNAME value “Smith” and matching ADDRESS:FIRSTNAME value “Joe.”