HP OpenView Select Identity

Connector for IBM DB2 Universal Database

Installation and Configuration Guide

Connector Version: 2.1
Select Identity Version: 3.3

April 2005

Legal Notices

Warranty

Hewlett-Packard makes no warranty of any kind with regard to this document, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

A copy of the specific warranty terms applicable to your Hewlett-Packard product can be obtained from your local Sales and Service Office.

Restricted Rights Legend

Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause in DFARS 252.227-7013.

Hewlett-Packard Company
United States of America

Rights for non-DOD U.S. Government Departments and Agencies are as set forth in FAR 52.227-19(c)(1,2).

Copyright Notices

This product includes software developed by the Apache Software Foundation (http://www.apache.org/).

Select Identity uses software from the Apache Jakarta Project including

- Commons-beanutils.
- Commons-collections.
- Commons-logging.
- Commons-digester.
- Commons-httpclient.

- Element Construction Set (ecs).
- Jakarta-poi.
- Jakarta-regexp.
- Logging Services (log4j).

Additional third party software used by Select Identity includes

- JasperReports developed by SourceForge.
- iText (for JasperReports) developed by SourceForge.
- BeanShell.
- Xalan from the Apache XML Project.
- Xerces from the Apache XML Project.
- Java API for XML Processing from the Apache XML Project.
- SOAP developed by the Apache Software Foundation.
- JavaMail from SUN Reference Implementation.
- Java Secure Socket Extension (JSSE) from SUN Reference Implementation.
- Java Cryptography Extension (JCE) from SUN Reference Implementation.
- JavaBeans Activation Framework (JAF) from SUN Reference Implementation.
- OpenSPML Toolkit from OpenSPML.org.
- JGraph developed by JGraph.
- Hibernate from Hibernate.org.
- BouncyCastle engine for keystore management, bouncycastle.org.
- Java Service Wrapper, Copyright © 1999, 2004 Tanuki Software.
- Copyright © 2001 Silver Egg Technology.
- This product includes software developed by Teodor Danciu (http://jasperreports.sourceforge.net). Portions Copyright © 2001-2004 Teodor Danciu (teodord@users.sourceforge.net). All rights reserved.
- Portions Copyright 1994-2004 Sun Microsystems, Inc. All Rights Reserved.

- This product includes software developed by the Waveset Technologies, Inc. (www.waveset.com). Portions Copyright © 2003 Waveset Technologies, Inc. 6034 West Courtyard Drive, Suite 210, Austin, Texas 78730. All rights reserved.
- Portions Copyright © 2001-2004, Gaudenz Alder. All rights reserved.

Trademark Notices

HP OpenView Select Identity is a trademark of Hewlett-Packard Development Company, L.P.

Microsoft, Windows, the Windows logo, and SQL Server are trademarks or registered trademarks of Microsoft Corporation.

Sun™ workstation, Solaris Operating Environment™ software, SPARCstation™ 20 system, Java technology, and Sun RPC are registered trademarks or trademarks of Sun Microsystems, Inc. JavaScript is a trademark of Sun Microsystems, Inc., used under license for technology invented and implemented by Netscape.

This product includes the Sun Java Runtime. This product includes code licensed from RSA Security, Inc. Some portions licensed from IBM are available at http://oss.software.ibm.com/icu4j/.

IBM, DB2 Universal Database, DB2, WebSphere, and the IBM logo are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both.

This product includes software provided by the World Wide Web Consortium. This software includes xml-apis. Copyright © 1994-2000 World Wide Web Consortium, (Massachusetts Institute of Technology, Institute National de Recherche en Informatique et en Automatique, Keio University). All Rights Reserved. http://www.w3.org/Consortium/Legal/

Intel and Pentium are trademarks or registered trademarks of Intel Corporation in the United States, other countries, or both.

AMD and the AMD logo are trademarks of Advanced Micro Devices, Inc.

BEA and WebLogic are registered trademarks of BEA Systems, Inc.

VeriSign is a registered trademark of VeriSign, Inc. Copyright © 2001 VeriSign, Inc. All rights reserved.

All other product names are the property of their respective trademark or service mark holders and are hereby acknowledged.

Support

Please visit the HP OpenView web site at:

http://www.managementsoftware.hp.com/

This web site provides contact information and details about the products, services, and support that HP OpenView offers.

You can also go directly to the support web site at:

http://support.openview.hp.com/

HP OpenView online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valuable support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track progress on support cases
- Manage a support contract
- Look up HP support contacts
- Review information about available services
- Enter discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and log in. Many also require a support contract.

To find more information about access levels, go to:

http://support.openview.hp.com/access_level.jsp

To register for an HP Passport ID, go to:

https://passport.hp.com/hpp2/newuser.do

contents

1

Installing the Connector

The IBM DB2 Universal Database connector — hereafter referred to as the DB2 connector — enables HP OpenView Select Identity to provision users on DB2 Universal Database v8.2 relational database systems. The connector also provides an agent that can send changes made to data in DB2 to Select Identity.

The following configurations are supported for this connector:

- Agent-based
  In this configuration, the connector communicates with an agent that resides on the database server; the agent uses a JDBC 2.0 compliant driver to communicate with the database. The agent can also push changes made in DB2 to the Select Identity database (this is called **reverse synchronization** and explained later).
- Agentless using a JDBC data source
  In this configuration, the connector communicates with the database directly through JDBC calls. Be sure to create or identify a JDBC data source (and underlying connection pool) on the Select Identity server that can connect to the target DB2 database.
- Agentless using a JDBC driver
  The connector communicates the database using a JDBC 2.0 compliant driver; no agent is installed on the database server.

The DB2 connector is packaged in the following files and folders, which are located on the Select Identity Connector CD:

- `IBM DB2 - Generic/Gen-DB2-Connector.rar` — The binaries for the connector
- `IBM DB2 - Generic/Agent Installers/DB2-Gen-AgentInstaller-Win.zip` — A ZIP file that contains the installation executable for the connector agent
- `IBM DB2 - Generic/Agent Installers/DB2-Gen-AgentInstaller-Unix.tar` — A TAR file that contains the installation executable for the connector agent
- `IBM DB2 - Generic/Manual Agent/DB2-Gen-Agent-Win.zip` — A ZIP file that contains agent binaries and files (for manual installation)
- `IBM DB2 - Generic/Manual Agent/DB2-Gen-Agent-Unix.tar` — A TAR file that contains agent binaries and files (for manual installation)

An Attribute Mapping Utility is also provided (deployed with the Select Identity server), which enables you to create mapping files that map database schema fields to Select Identity fields. Refer to the *HP OpenView Select Identity Attribute Mapping Utility User's Guide* for more information about the Attribute Mapping Utility.

Operations Supported by the Connector

The DB2 connector is intended for use in a wide variety of usage scenarios. Specifically, it can perform the following operations in the database schema on the DB2 system:

- Add, update, and remove users
- Retrieve user attributes
- Enable and disable users
- Verify a user's existence
- Change user passwords
- Reset user passwords
- Retrieve all entitlements

- Retrieve a list of supported user attributes
- Assign and unassign entitlements to and from users
- Add, update, and remove entitlements

This connector does not provision database system users. Rather, it provisions users into a user-defined database schema in DB2. To provision database system users, you must install and use the Admin DB2 connector.

In addition, the connector's agent can send user changes made in DB2 to Select Identity. When changes are pushed from the agent to the Select Identity server, this is referred to as **reverse synchronization**. The following reverse synchronization operations are supported:

- Change passwords stored in Select Identity based on changes to the passwords in the schema in DB2
- Add, modify, and delete users based on user additions, modifications, and deletions in the schema in DB2

When a user is added, modified, or deleted in the database, triggers capture the changes. The agent's reverse synchronization component then sends the changes to Select Identity's Web Service in SPML. If an error occurs during reverse synchronization, the agent stops the operation (without affecting the connector's operations).

Additional steps are required to configure the agent for reverse synchronization. (Note that installing and configuring the agent is mandatory in order for the connector to support reverse synchronization.)

The DB2 connector also supports custom encryption, which enables the connector to encrypt values provisioned in the schema. Refer to the *HP OpenView Select Identity Attribute Mapping Utility User's Guide* for information on how to use this feature.

System Requirements

The DB2 connector is supported in the following environment:

Select Identity Version	Application Server	Database
3.0.2	WebLogic 8.1.2 on Windows 2003	SQL Server 2000
	WebLogic 8.1.2 on Solaris 9	Oracle 9i
	WebLogic 8.1.2 on HP-UX 11i	Oracle 9i
	WebSphere 5.1.1 on Solaris 9	DB2 8.2 (or DB2 8.1 Service Pack 7)
3.3	WebLogic 8.1.4 on Windows 2003	SQL Server 2000

For Select Identity 3.0.2, the DB2 connector is supported with IBM DB2 Universal Database, version 8.2, running on Windows 2000, Windows 2003, Windows XP, and Solaris 9. For 3.3, the connector is supported with IBM DB2 Universal Database, version 8.2, running on Windows 2003 and Solaris 9. Also, for secure communication, this connector supports secure JDBC for database communication. See Configuring DB2 to Support Secure JDBC on page 33 for configuration information.

Deploying on the Web Application Server

To install the DB2 connector on the Select Identity server, complete these steps:

1. Create a subdirectory in the Select Identity home directory where the connector's RAR file will reside. For example, you could create the `C:\Select_Identity\connectors` folder on Windows. (A connector subdirectory may already exist.)
2. Copy the `Gen-DB2-Connector.rar` file from the Select Identity Connector CD to the connector subdirectory.

3. Copy the JDBC 2.0 compliant driver to the application server. For DB2, you must copy the JDBC driver files (`db2jcc.jar`, `db2jcc_license_cisuz.jar`, and `db2jcc_license_cu.jar`). Obtain the driver files from your database administrator.
4. Add the JDBC driver files to the application server's class path, such as by editing the `myStartWL.cmd` (on Windows) or `myStartWL.sh` (on UNIX) file.
5. If deploying the connector on WebLogic, complete the following steps.
   a. Start the application server if it is not currently running.
   b. Log on to the WebLogic Server Console.
   c. Navigate to ***My_domain*** → **Deployments** → **Connector Modules**.
   d. Click **Deploy a New Connector Module**.
   e. Locate and select the `Gen-DB2-Connector.rar` file from the list. It is stored in the connector subdirectory.
   f. Click **Target Module**.
   g. Select the **My Server** (your server instance) check box.
   h. Click **Continue**. Review your settings.
   i. Keep all default settings and click **Deploy**. The Status of Last Action column should display Success.
6. If deploying the connector on WebSphere, complete the following steps:
   a. Start the application server, if necessary.
   b. Log on to the WebSphere Application Server Console.
   c. Navigate to **Resources** → **Resource Adapters**.
   d. Click **Install RAR**.
   e. In the Server path field, enter the path to the `Gen-DB2-Connector.rar` file. It is stored in the subdirectory created in Step 1.
   f. Click **Next**.
   g. In the Name field, enter a name for the connector.
   h. Click **OK**.
   i. Click the **Save** link (at the top of the page).

j On the Save to Master Configuration dialog, click the **Save** button.

k Click **Resources** → **Resource Adapters**.

l Click the new connector.

m Click **J2C Connection Factories** in the Additional Properties table.

n Click **New**.

o In the Name field, enter the name of the factory for the connector. For the DB2 connector, enter **`eis/Gen-DB2Connector`**.

p Click **OK**.

q Click the **Save** link.

r On the Save to Master Configuration dialog, click the **Save** button.

s Restart WebSphere.

7 Create a mapping file for the schema in which the connector must provision users and entitlements. Use the Attribute Mapping Utility, as described in the *HP OpenView Select Identity Attribute Mapping Utility User's Guide*. See Understanding the Mapping Files on page 43 for more information about the file format.

8 To configure reverse synchronization on the server, you must create an XSL file based on the XML mapping file. The XSL file maps user attributes on DB2 to attributes in Select Identity. This file is generated by the Attribute Mapping Utility when the XML mapping file is created. See Understanding the Mapping Files on page 43 for more information.

After the XSL file is created, copy it to the Select Identity home directory.

Note that the agent must be installed and configured for the DB2 connector to support reverse synchronization.

9 Edit the `TruAccess.properties` file, which resides in the *`install_dir`*`/sysArchive` directory on the Select Identity server, and modify the `com.hp.ovsi.connector.schema.dir` property. Set this property to the top-level directory where the mapping files reside, which is specified in the Base Directory field on the Attribute Mapping Utility interface.

If no value was specified in the Base Directory field, the `com/trulogica/truaccess/connector/schema/spml` directory structure was created in the application server's home directory.

For example, if you entered `C:\SI_3.3\schema` in the Base Directory field, the `com/trulogica/truaccess/connector/schema/spml` directory structure was created under that directory, and the XML and XSL files were created there. In this example, the files reside in this directory:

```
C:\SI3.3\schema\com\trulogica\truaccess\connector\schema\spml
```

Thus, you would set the `com.hp.ovsi.connector.schema.dir` property as follows:

```
com.hp.ovsi.connector.schema.dir = C:/SI3.3/schema
```

After installing the connector, refer to Configuring the Connector on page 36 for information about registering and configuring this connector in Select Identity.

Installing the Agent on the Database Server

After you install the DB2 connector on the Select Identity server, you can install the agent on the database server. This is optional; the connector can provision users in DB2 without the agent. However, the agent enables you to send data back to Select Identity (reverse synchronization).

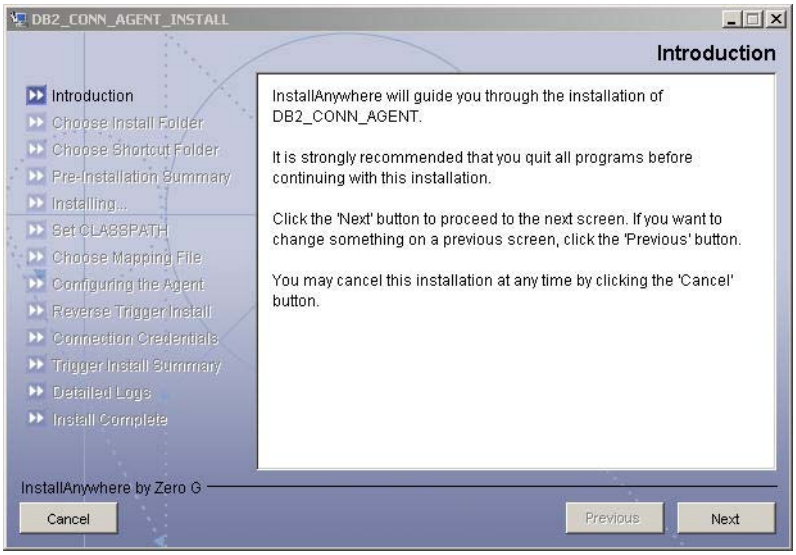
You can install the agent using the installation wizard or by manually copying files to the server.

Be sure to generate the XML and XSL mapping files described in Step 7 and Step 8 on page 13 before installing the agent. Then, copy the mapping files from the Select Identity server to the system where you will install the agent (on the database server). The agent installation requires that the mapping files are available on the local system.

Installation Using the Wizard on Windows

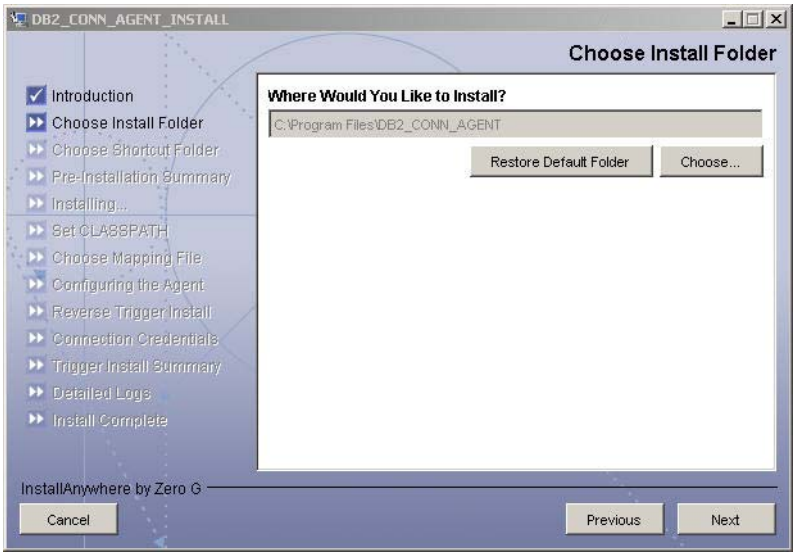
Complete the following steps to run the installation wizard, which installs the agent on Windows:

1. Extract the contents of the `DB2-Gen-AgentInstaller-Win.zip` file, which is located in the `Agent Installers` directory on the CD.
2. Run `install.exe` , which is located in the *`target_dir`*`\CDROM_Installers\Windows\Disk1\InstData\NoVM`. The following dialog displays:

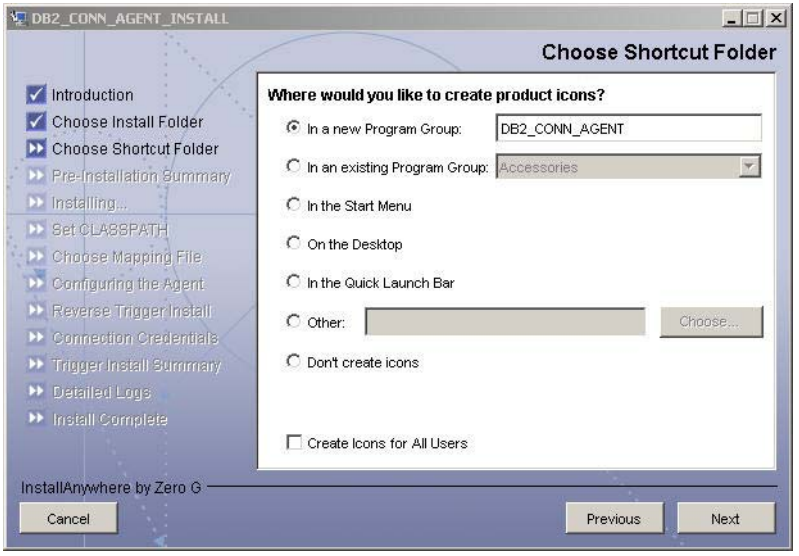


3. Click **Next** to proceed.

4 Specify an installation directory then click **Next**. By default, the agent will be installed in `C:\Program Files\DB2_CONN_AGENT`.



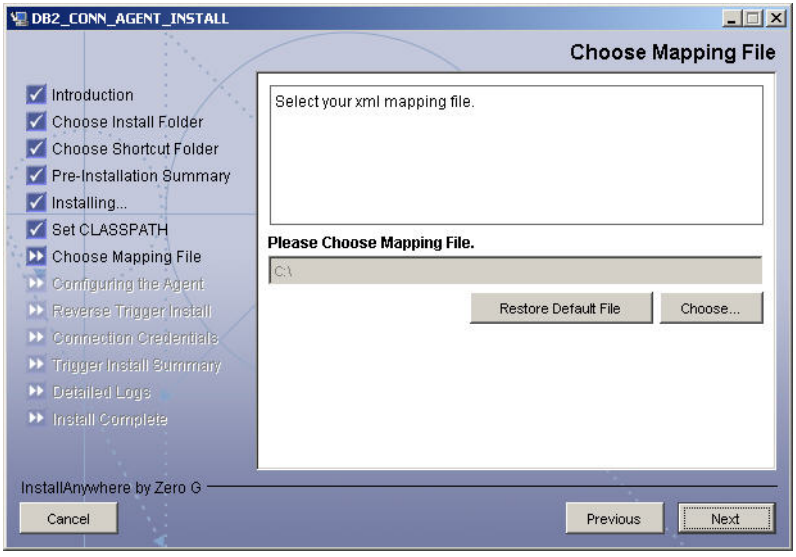
5 Select the location(s) where the product icons will be installed, then click **Next**.



6 Verify the pre-installation summary. If you wish to make changes, click **Previous** and edit the chosen options. To install the agent, click **Install**.

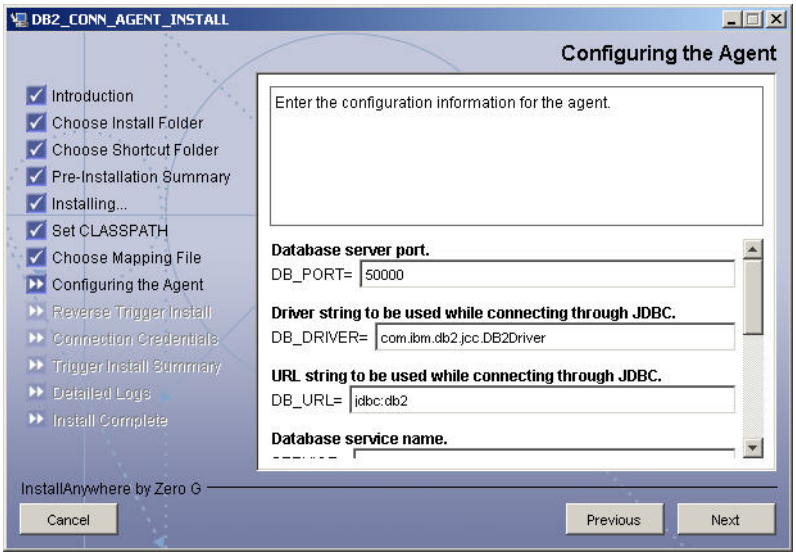
7 On the Set CLASSPATH dialog, click **Next** after you verify that the database driver files (`db2jcc.jar`, `db2jcc_license_cisuz.jar`, and `db2jcc_license_cu.jar`) are in the database server's System class path.

8 Click **Choose** to browse for and select the mapping file. This will copy the mapping file created in Step 7 on page 13 to the *`install_dir`*`/conf/com/trulogica/truaccess/connector/schema/spml` directory, where *install_dir* is the installation folder selected in Step 4 above.



Then, click **Next**.

9 On the Configuring the Agent dialog, specify the requested configuration information:



The following provides an explanation of the configuration options:

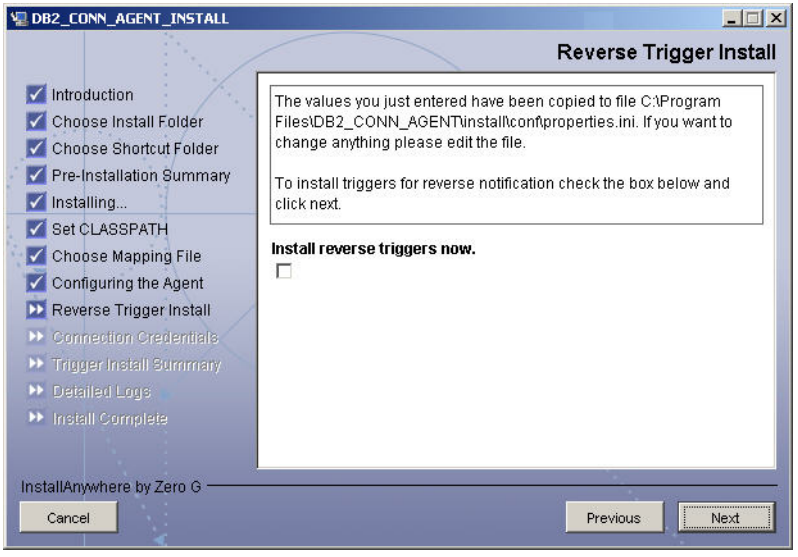
Option	Description	Example Value
DB_PORT	The port on which the database server is listening.	50000
DB_DRIVER	The JDBC driver for the database connection.	com.ibm.db2.jcc.DB2Driver
DB_URL	The JDBC URL string used for the database communication.	jdbc:db2
SERVICE	The database name.	SI_DB
CONCERO_SERVER_URL	The URL of the Select Identity Web Service.	http://*host*:*port*/lmz/webservice
PollDelay	The polling delay for reverse polling (in seconds).	10
AGENT_PORT	The port on which the agent listens for user provisioning requests from Select Identity.	5601

Option	Description	Example Value
MAPPING_FILE	The XML mapping file generated by the Attribute Mapping Utility. If you have not generated the XML file yet, you can change this value later.	Mapping.xml
SPML_DELAY	The delay (in milliseconds) between successive SPML requests sent from the agent. Increase this delay if the network or Select Identity server is performing slowly.	10000

To edit any of these values after installation, you can edit the `properties.ini` file, which resides in *`install_dir`*`\conf`.

After specifying these values, click **Next**.

10 To enable reverse synchronization, you must install the reverse triggers. (See Operations Supported by the Connector on page 9 for an explanation of reverse synchronization.) Select the **Install Triggers Now** option to install the triggers. Then, click **Next** and proceed to the next step.



If you choose not to install the reverse triggers, skip to Step 14 on page 20. (You can manually install the triggers later, if necessary. This is described in Installing the Reverse Triggers on page 30.)

11 If you selected the **Install Reverse Triggers now** option on the Reverse Trigger Install dialog, specify authentication information for the database user. Then, click **Next**. The triggers are installed for the schema specified by the mapping file.

12 Review the installation summary for the triggers. If you wish to make changes, or if the trigger installation failed, click **Previous** and edit the chosen options, such as the credentials. You can also select the **Show Logs** option to review the trigger installation log files. Then, click **Next**.

13 If you selected the **Show Logs** option, the Detailed Logs dialog is displayed. Review the log entries and click **Next**.

14 When the installation wizard completes, click **Done** on the Install Complete dialog to close the installation program.

If you configured the agent for reverse synchronization (by installing the triggers), you must edit the *install_dir*`/conf/opattributes.properties` file to configure the triggers. See Step 4 on page 30 for details.

Installation Using the Wizard on UNIX

Complete the following steps to run the installation wizard, which installs the agent on UNIX:

- Extract the contents of the `DB2-Gen-AgentInstaller-Unix.tar` file, which is located in the `Agent Installers` directory on the CD, to a directory that will server as the agent's home directory. (Use **`tar xvf`** to extract the contents of the TAR file.) This will create the required directory structure in the `DB2-Gen-AgentInstaller-Unix` subdirectory of the home directory.

15 Set the JAVA_HOME_14 environment variable to the directory where the JDK 1.4 is installed. Also, add the JVM to the system PATH variable.

16 Add the JDBC 2.0 compliant driver files (`db2jcc.jar`, `db2jcc_license_cisuz.jar`, and `db2jcc_license_cu.jar`) for the database to the CLASSPATH.

17 Start the wizard by running the following command:

***agent_home*/DB2-Gen-AgentInstaller-Unix/install.bin**

The following displays:

```
===============================================================
Extracting the installation resources from the installer
archive...
Configuring the installer for this system's environment...

Launching installer...

Preparing CONSOLE Mode Installation...
===============================================================
(created with InstallAnywhere by Zero G)
---------------------------------------------------------------

Choose Install Folder
---------------------

Where would you like to install?

Default Install Folder: /DB2_CONN_AGENT

ENTER AN ABSOLUTE PATH, OR PRESS <ENTER> TO ACCEPT THE DEFAULT:
```

18 Specify the installation location of the agent. Enter a path and press ENTER, or simply press ENTER to accept default path. The following displays:

```
===============================================================
Choose Link Location
--------------------

Where would you like to create links?

->1- Default: /
2- In your home folder
3- Choose another location...
4- Don't create links

ENTER THE NUMBER OF AN OPTION ABOVE, OR PRESS <ENTER> TO ACCEPT
THE DEFAULT:
```

19 Select where you would like the agent shortcut location to be created. Select the number of the desired option as shown and press ENTER, or simply press ENTER to accept the default.The following displays:

```
===============================================================
Pre-Installation Summary
------------------------

Please Review the Following Before Continuing:

Product Name:
DB2_CONN_AGENT

Install Folder:
/install_dir

Link Folder:
/

Disk Space Information (for Installation Target):
Required:  xxx bytes
Available: yyy bytes

PRESS <ENTER> TO CONTINUE:
```

20 Verify the pre-installation summary and press ENTER. The following displays:

```
===============================================================
Installing...
-------------

[===============|===============|==============|==============]
[---------------|---------------|--------------|--------------]

===============================================================
Configuring the Agent
---------------------

Enter the port number where database server listens. Hit <ENTER>
to accept default.

Enter DB_PORT :  (DEFAULT: 50000 ):
```

21 Specify the database port number and press ENTER to continue. Or, simply press ENTER to accept the default. The following displays:

```
===============================================================
Configuring the Agent
---------------------

Enter the driver string (Driver string to be used by Java
program to connect to the database).

Enter the database driver : (DEFAULT:
com.ibm.db2.jcc.DB2Driver):
```

22 Enter the database driver and press ENTER to continue. Or, simply press ENTER to accept the default. The following displays:

```
===============================================================
Configuring the Agent
---------------------

Enter the driver URL (URL to be used by Java program to connect
to the database).

Enter the database URL :  (DEFAULT: jdbc:db2):
```

23 Enter the JDBC URL and press ENTER to continue. Or, simply press ENTER to accept the default. The following displays:

```
================================================================
Configuring the Agent
---------------------

Enter the database service name.

Enter the service name :  (DEFAULT: ): OPENVIEW
```

24 Enter the database name and press ENTER to continue. Or, press ENTER to accept the default. The following displays:

```
================================================================
Configuring the Agent
---------------------

Enter the URL where spml is to be sent by reverse sync.

Enter the concero server URL :  (DEFAULT: ): http://
localhost:7001/lmz/webservice
```

25 Enter the URL of the Select Identity Web Service, which is where SPML requests are sent and press ENTER to continue. Or, press ENTER to accept the default. The following displays:

```
================================================================
Configuring the Agent
---------------------

Enter the interval (in seconds) at which polling is desired for
reverse sync.

Enter poll delay :  (DEFAULT: 10):
```

26 Enter the polling interval (in seconds) that is used by the agent to check for changes on the resource that must be sent to the Select Identity server (during reverse synchronization), then press ENTER. Or, simply press ENTER to accept the default. The following displays:

```
================================================================
Configuring the Agent
---------------------

Enter the port where the agent should listen.

Enter the agent port :  (DEFAULT: ):6000
```

27 Enter the listening port number for the agent and press ENTER. Or, simply press ENTER to accept the default and continue. The following displays:

```
============================================================
Configuring the Agent
---------------------

Enter the time (in milli seconds) for which the agent should
wait before sending SPML.

Enter the spml delay :  (DEFAULT: 100):
```

28 Specify the delay (in milliseconds) that the agent will wait before sending SPML requests to the Select Identity server and press ENTER. Or, simply press ENTER to accept the default and continue. The following displays:

```
============================================================
Configuring the Agent
---------------------

Enter Directory Path of the XML mapping file. (e.g. If the
mapping file is "/osd5/trulogica/xxx.xml", enter "/osd5/
trulogica/" including slashes)

Directory Path of the XML mapping file.  (DEFAULT: ): /opt/
```

29 Enter the path to the mapping file (include trailing slashes but do not include the file name) and press ENTER. Or, simply press ENTER to accept the default. The following displays:

```
============================================================
Configuring the Agent
---------------------

Enter name of the XML mapping file. (Enter extension also.)

Name of the XML mapping file. (Enter extension also.) (DEFAULT:
) : xxx.xml
```

30 Enter the name of the mapping file and press ENTER. Or, simply press ENTER to accept the default. The following displays:

```
============================================================
XML Mapping Path
-----------------

This is your directory path of Mapping File

"/opt/"

This is the Mapping File

"xxx.xml"

Is it Correct Path ?(Y/N) (DEFAULT: Y): y
```

31 Press ENTER to accept the default (y) or enter **n** and press ENTER to change the values. If you enter **y**, the following displays:

```
============================================================
Reverse Notification Tables Install
-----------------------------------

The values you just entered have been copied to file /
DB2_CONN_AGENT\conf\properties.ini. If you want to change
anything please edit the file.

Do you want to install reverse notification triggers now? (Y/N)
(DEFAULT: Y)    : y
```

32 To enable reverse synchronization, you must install the reverse triggers. (See Operations Supported by the Connector on page 9 for an explanation of reverse synchronization.) Enter **y** and press ENTER to install the triggers (or simply press ENTER to accept the default), or enter **n** and press ENTER to bypass this installation.

If you enter **y**, the following displays:

```
============================================================
Connection Credentials
----------------------

Enter user name.

Enter user name :  (DEFAULT: ): TEST
```

33 To specify credentials to install the triggers in the database, specify a user name and press ENTER. Or, leave this prompt blank and press ENTER. The following displays:

```
============================================================
Connection Credentials
----------------------

This installation requires a password to continue.

Enter password : : password
```

34 If you specified a user name, enter a password and press ENTER. The following displays:

```
============================================================
command
--------

calling the command "//DB2_CONN_AGENT/Setup.sh" -userName
"TEST" -password password

PRESS <ENTER> TO CONTINUE: y
```

35 Press ENTER to continue. The trigger pre-installation summary displays, indicating whether the installation of the triggers succeeded or failed:

```
============================================================
Reverse Trigger Install Summary
-------------------------------------

Reverse Trigger Install SUCCESS. Please see the logs for
details.

PRESS <ENTER> TO CONTINUE: y
```

36 Press ENTER to continue. The following displays:

```
============================================================
View Logs
---------

Do you want to see detailed logs? (Y/N) (DEFAULT: Y): y
```

37 If you wish to view the installation log file, enter **y** and press ENTER. Otherwise, enter **n** and press ENTER.

38 To exit the installation wizard, press ENTER.

Manual Installation

Instead of using the installation wizard, you can install the agent files and reverse triggers manually. The following sections describe how to do this.

Installing the Agent

Complete the following steps to manually copy the agent files to the target server:

1. *On Windows:*
   Extract the contents of the `DB2-Gen-Agent-Win.zip` file, which resides in the `Manual Agent` subdirectory on the CD, to a target location for the agent on the DB2 system. The extracted files will reside in the `DB2-Gen-Agent-Win` directory.

   On UNIX:
   Extract the contents of the `DB2-Gen-Agent-Unix.tar` file, which resides in the `Manual Agent` subdirectory on the CD, to a target location for the agent on the DB2 system. (Use **`tar xvf`** for extracting the contents of the TAR file.) The extracted files will reside in the `DB2-Gen-Agent-Unix` directory.

2. Copy the mapping file created in Step 7 on page 13 to the *`agent_home`*`/conf/com/trulogica/truaccess/connector/schema/spml` directory.

3. Modify the `properties.ini` file, which resides in the *`agent_home`*`/conf` subdirectory, to specify parameters for the agent. The parameters are listed in the following table.

Parameter	Sample Values	Description
PORT	50000	The port on which the database server is listening.
DB_DRIVER	com.ibm.db2.jcc.DB2Driver	JDBC driver for the database connection.
DB_URL	jdbc:db2	JDBC URL string used for the database communication.
SERVICE	SI_DB	Database name.

Parameter	Sample Values	Description
CHECK_LOGIN	true	The Login Check flag.
MAX_LOGIN_RETRIES	3	The number of times the agent will attempt to log in to the database.
CONCERO_SERVER_URL	http://*host*:*port*/lmz/webservice	URL of the Select Identity Web Service.
PollDelay	10	The polling delay for reverse polling (in seconds).
AGENT_PORT	5601	The port on which the agent listens for user provisioning requests from Select Identity.
MAPPING_FILE	Mapping.xml	The XML mapping file generated by the Attribute Mapping Utility.
SPML_Delay	10000	The delay (in milliseconds) between successive SPML requests sent from the agent. Increase this delay if the network or Select Identity server is performing slowly.

4 Copy the DB2 JDBC driver files (`db2jcc.jar`, `db2jcc_license_cisuz.jar`, and `db2jcc_license_cu.jar`) to the system CLASSPATH. Obtain these files from the DB2 system, the Select Identity server, or your system or database administrator.

See Installing the Reverse Triggers on page 30 for steps to configure reverse synchronization. See :Starting the Agent on page 33 for information about starting the agent.

Installing the Reverse Triggers

Perform these steps if you want to synchronize changes made to users in DB2 with Select Identify. Reverse synchronization relies on triggers configured on the database. When you start the agent, reverse synchronization is enabled.

1. Copy the XML mapping file created in Step 8 on page 13 to the *agent_home*/conf/com/trulogica/truaccess/connector/schema/spml directory.
2. Edit the properties.ini file, which resides in the *agent_home*/conf subdirectory, to specify parameters for reverse synchronization. See Step 3 on page 28 for details on this file.
3. Run the *agent_home*/setup.cmd file (on Windows) or setup.sh file (on UNIX) from the command line. This installs triggers as specified by the mapping file and creates audit tables (SID_TAB, Table_Audit, and Column_Audit). If the tables exist, table creation fails, indicating the error.
4. Modify the opattributes.properties file, which resides in the *agent_home*/conf/ subdirectory and provides operational attributes that are sent to the Select Identity server during reverse synchronization requests. The file must contain the following:

Parameter	Sample Values	Description
urn:oasis:names:tc:SPML:1:0# UserIDAndOrDomainName	Sisa	User ID of the administrative user on Select Identity
urn:trulogica:concero:2.0#password	Abc123	Password of the administrative user
urn:trulogica:concero:2.0# reverseSync	true	Set to **true** if you want to enable reverse synchronization

Parameter	Sample Values	Description
urn:trulogica:concero:2.0# resourceType	GenDB2_ Schema1	The name of the XSL file (without the .xsl extension) that is used during reverse synchronization
urn:trulogica:concero:2.0# resourceId	SQL	The name of the Select Identity resource that is created for the DB2 connector.

If you wish to delete the triggers, complete the steps in Manually Removing the Agent on page 55. These steps assume that `properties.ini` is configured as mentioned in Installing the Agent on the Database Server on page 14.

Installed Files

The following provides a listing of the directories and files installed for the agent:

Directories and Files	Description
agent_home/	Contains the following files: • `AddToStartupGroup.cmd/sh` — Adds icons to startup group; this file is present only if the agent was installed using the wizard • `CopyFile.cmd/sh` — Used by agent to copy files; this file is present only if the agent was installed using the wizard • `DelFile.cmd/sh` — Used by agent to delete files; this file is present only if the agent was installed using the wizard • `setup.cmd/sh` — Installs the reverse triggers • `sqlapp.cmd/sh` — Used by agent to communicate with the database • `SQLConnectorConsole.cmd/sh` — Starts the agent • `uninstall.cmd/sh` — Uninstalls triggers
agent_home/conf/	Contains the following files: • `properties.ini` — Provides configuration settings for the agent • `opAttributes.properties` — Provides configuration settings for reverse synchronization • `log4j.properties` — Provides settings for logging.
agent_home/conf/com/	Contains the `trulogica/truaccess/connectior/schema/spml` directory structure where the XML mapping file is stored
agent_home/lib/	Contains JAR files used by the agent.

Directories and Files	Description
agent_home/logs	Contains log files produced by the agent.
agent_home/Uninstall_DB2_CONN_AGENT/	Contains files for uninstalling the agent. This subdirectory is created only if the agent is installed using the installation wizard.

:Starting the Agent

To start the agent, run `SQLConnectorConsole.cmd` (on Windows) or `SQLConnectorConsole.sh` (on UNIX), which resides in the agent's home directory. This program logs in to the database server using the user name and password of a user who has administrative privileges on the database.

If you wish, you can provide the following parameters to the command:

username — The user name of the user who has administrative privileges on the database.

password — The specified user's password.

Here is an example you can use on Windows:

***agent_home*/SQLConnectorConsole.cmd -userName si -password abc123**

If you start the agent before or without configuring reverse synchronization (the reverse triggers), a message is displayed stating that reverse notification is disabled.

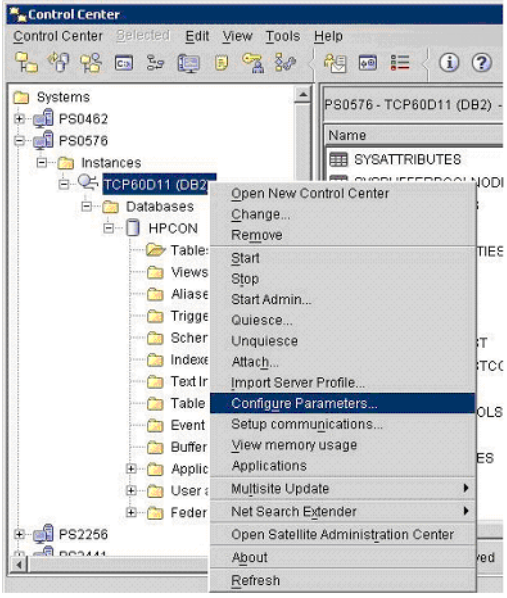
Configuring DB2 to Support Secure JDBC

Complete the following steps to configure the DB2 server to support secure JDBC:

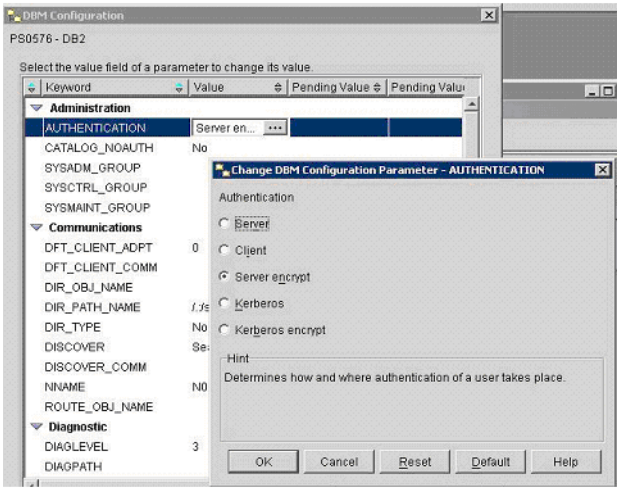
1 Update the `java.security` file, which resides in *JAVA_DIR*`\jre\lib\security`, by adding the following line:

security.provider.2=com.ibm.crypto.provider.IBMJCE

2. Ensure that the `ibmjceprovider.jar` and `ibmpkcs.jar` files are added to the *`JAVA_DIR`*`\jre\lib\etc` directory.
3. Launch the Control Center on the DB2 server.
4. Right-click on ***DB2_server*** → **Instances** → ***instance*** and select **Configure Parameters...** from the menu:



5 In the configuration window, change the authentication to **Server encrypt**.



6 Restart the DB2 server.

2

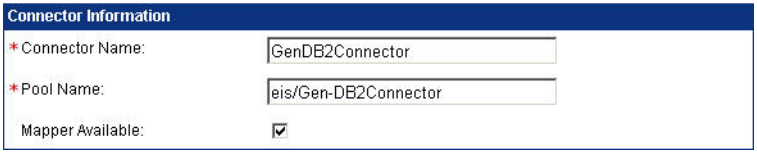
Configuring the Connector

After you deploy the connector on the application server, you must configure Select Identity to use the connector by deploying it in the Select Identity client. The following provides an overview of the procedures you must complete in order to deploy your connector. It also provides connector-specific information you must provide when configuring Select Identity to use the connector.

1 Register the DB2 connector with Select Identity by clicking the **Deploy New Connector** button on the Connectors home page. Complete this procedure as described in the "Connectors" chapter of the *HP OpenView Select Identity Administrator Guide*.

Be sure to select the **Mapper Available** check box, to enable users to edit the connector's mapping file using the Attribute Mapping Utility.

After you deploy the connector, the connector properties will look similar to this:



2 Deploy a resource that uses the newly created connector. On the Resources home page, click the **Deploy New Resource** button. The resource configuration depends on how the connector and agent were installed and configured:

— Using a JDBC data source, an agent is not installed:
In this configuration, the connector performs operations on the database directly through JDBC calls. You must specify the JDBC data source and mapping file when configuring the resource.

— Using a JDBC driver, an agent is not installed:
The connector uses the JDBC driver to communicate with the database. You must specify all parameters except the agent port and JDBC data source.

— Using a JDBC driver, an agent installed:
If the agent is installed and a JDBC driver is used to communicate with the database, you must specify all parameters except the JDBC data source.

Complete the steps in this procedure as described in the "Resources" chapter of the *HP OpenView Select Identity Administrator Guide*.When configuring the resource, refer to the following table for parameters specific to this connector:

Copy or move the XML and XSL files to the proper locations. For example, if `C:\si3.3\weblogic\sysarchive` is a folder in the WebLogic CLASSPATH, the XSL should reside in `C:\si3.3\weblogic\sysarchive` and the XML should reside in `C:\si3.3\weblogic\sysarchive\com\trulogica\truaccess\connector\schema\spml`.

Field Name	Sample Values	Description
Resource Name	Gen-DB2	The name of the resource.
Resource Type	SQL	The connector that was deployed in Step 1 on page 36.

Field Name	Sample Values	Description
Authoritative Source*	No	Whether this resource is a system that is considered to be the authoritative source for user data in your environment. Specify **Yes** if the connector is enabled for reverse synchronization. If the resource is not authoritative, the resource can only modify user entitlements during reverse synchronization.
Associate to Group	Selected	Whether the system uses the concept of groups. For this connector, select this option.
Server Name	Ps0111	Host name or IP address of the database server. You must specify this parameter if the agent was installed.
Server Port	50000	Port on which the database server is listening. You must specify this parameter if the agent was installed.
Username	sa	The login name of the database administrative user. You must specify this parameter if the agent was installed.
Password	p4ssword	Password of the database administrative user. You must specify this parameter if the agent was installed.
Agent Port	5601	The port where the agent listens for incoming connections. You must specify this parameter if the agent was installed.

Field Name	Sample Values	Description
SQL URL	jdbc:db2	URL to use to communicate with the database over a JDBC connection. You must specify this parameter if the agent was installed.
Database / Service Name	testDB	The database name in which to provision users. You must specify this parameter if the agent was installed.
Database Driver String	com.ibm.db2.jcc.DB2Driver	Name of the JDBC driver to connect to the database. You must specify this parameter if the agent was installed.
Mapping File	Mapping.xml	Mapping file containing the mappings generated by the Attribute Mapping Utility. The mapping file must reside in the `install/conf/com/trulogica/truaccess/connector/schema/spml` directory in order for the Select Identity server to find it.
JDBC Datasource String	Jdbc/SQLDataSource	JNDI data source name that was created or identified on the Select Identity server that can connect to the target DB2 database. Specify a value for this property if the agent was not installed.
Encryption Specification Algo	kerberosServerPrincipal	Encryption algorithm specification string. Specify this parameter if you wish to use secure communication with DB2.
Encryption Algorithm	kdcsrv1.sj.ibm.com	Name of the encryption algorithm. Specify this parameter if you wish to use secure communication with DB2.

Field Name	Sample Values	Description
Encryption Specification Level	securityMechanism	Encryption level specification string. Specify this parameter if you wish to use secure communication with DB2.
Encryption Level	3, 7, or 9	Encryption level. Specify 3 if a user ID and password is used as the security mechanism. Specify 7 if a user ID is used. Specify 9 if a user ID and encrypted password is used. Specify this parameter if you wish to use secure communication with DB2.

* Instead of creating an authoritative resource, you can create authoritative attributes (in the next step) for the attributes that will be synchronized. Entitlements are authoritative by default in a non-authoritative resource but other attributes are not.

After you deploy the resource for the DB2 connector, the Access Info page of the resource properties will look similar to this (this is the configuration for an agentless connector):

Resource Access Information	
* Resource Name:	DB2Resource
Server Name:	sisun4
Server Port:	50001
Username:	db2inst2
Password:	********
Agent Port:	
SQL URL:	jdbc:db2
DataBase/Service Name:	TEST
Database Driver String:	com.ibm.db2.jcc.DB2Driver
* Mapping File:	db21user0401.xml
JDBC Datasource String:	JDBC/DataSource
Encryption Specification Algo:	
Encryption Algorithm:	
Encryption Specification Level:	
Encryption Level:	

3 Create attributes that link Select Identity to the connector. For each mapping in the connector's mapping file, create an attribute using the Attributes capability on the Select Identity client.

Refer to the "Attributes" chapter in the *HP OpenView Select Identity Administrator Guide* for more information. After you create the attributes for the DB2 connector, the View Attributes page for the resource will look similar to this:

(Resource Name=DB2Resource)

Page 1 of 1 Total Records:6

Name	Min Length	Max Length	Attribute Mapped To	Authorative
Address	0	255		
DB2Resource_ENTITLEMENTS	1	255	DB2Resource_ENTITLEMENTS	Y
DB2Resource_KEY	1	255	DB2Resource_KEY	Y
Email	0	255		
Password	0	255		
UserID	0	255		

The attributes in the snapshot are sample values based on the sample XML file given above.

4 Create a Service that will use the newly created resource. To do so, click the **Deploy New Service** button on the Services home page. Complete this procedure as described in "Services" of the *HP OpenView Select Identity Administrator Guide*. You will reference your new resource created in Step 2 while creating this service.

If you are enabling reverse synchronization, configure the Service as follows:

— When selecting the Business Relationship, choose the ReconciliationDefaultProcess workflow for the RECONCILIATION:Add Service and RECONCILIATION:Delete Service Membership request events. For RECONCILIATION:Add Service, use the user addition view.

— In the user addition view, specify mandatory attributes that are guaranteed to be passed by the reverse synchronization request when adding a user. If you specify a mandatory attribute that is not passed by the resource, the user will be created in Select Identity but reverse synchronization will not succeed.

— When specifying the context, obtain the value from the add request issued by the resource. For example, if the context is Country and the value is US, the <addRequest> element in the reverse synchroniation request should have an attribute called country and a value of US. If the context attribute is not present in the add user request, the user will be created in Select Identity but will not be assigned to a Service.

3

Understanding the Mapping Files

To enable the connector to provision users and entitlements in the schema on the DB2 resource, you must create an XML mapping file. If you configured the agent to support reverse synchronization, you must also provide an XSL file that provides a reverse mapping of the Select Identity and resource fields mapped in the XML file.

To generate these mapping files, you must use the Attribute Mapping Utility. Only attributes that are defined in the mapping files are provisioned by Select Identity. Refer to the *HP OpenView Select Identity Attribute Mapping Utility User's Guide* for detailed information on how to generate a mapping file using that utility.

This chapter provides an explanation of the XML and XSL mapping files. The following sections are provided:

Refer to the `IBM DB2 - Generic/SampleXML` subdirectory on the Select Identity Connector CD for sample files.

Elements in the XML Mapping File

Here is an explanation of the format of the XML mapping file. For a sample mapping file, see the `SampleXML` subdirectory in the `IBM DB2 - Generic` directory on the Select Identity Connector CD.

If the database schema changes and you need of a new set of mapping files, you can load the existing XML file in the Attribute Mapping Utility and modify it as needed. Using a text editor to edit the XML and XSL files is error-prone.

- **<Schema>**, **<providerID>**, and **<schemaID>**

  Provides standard elements for header information.

- **<objectClassDefinition>**

  Defines the actions that can be performed on the specified object as defined by that name attribute (in the <properties> element block) and the Select Identity-to-resource field mappings for the object (in the <memberAttributes> block). For example, the object class definition for users defines that users can be created, read, updated, deleted, reset, and expired in DB2.

  - **<properties>**
    Defines the operations that are supported on the object. This can be used to control the operations that are performed through Select Identity. The following operations can be controlled:

    — Create (CREATE)

    — Read (READ)

    — Update (UPDATE)

    — Delete (DELETE)

    — Enable (ENABLE)

    — Disable (DISABLE)

    — Reset password (RESET_PASSWORD)

    — Expire password (EXPIRE_PASSWORD)

    — Change password (CHANGE_PASSWORD)

    — Assign entitlements (LINK)

— Unassign entitlements (UNLINK)

— Retrieve entitlements (GETALL)

The operation is assigned as the name of the <attr> element and access to the operation is assigned to a corresponding <value> element. You can set the values as follows:

— true — the operation is supported by the connector

— false — the operation is not supported by the connector

— bypass — the operation is not supported by the connector

Here is an example:

```
<objectClassDefinition description="" name="User">
   <properties>
    <attr name="GETCHILDREN">
      <value>true</value>
    </attr>
    <attr name="DELETE">
      <value>true</value>
    </attr>
    <attr name="EXPIREPASSWORD">
      <value>true</value>
    </attr>
    <attr name="GETALL">
      <value>true</value>
    </attr>
...
```

- **<memberAttributes>**
  Defines the attribute mappings. This element contains <attributeDefinitionReference> elements that describe the mapping for each attribute. Each <attributeDefinitionReference> can be followed by an <attributeDefinition> element that specifies details such as minimum length, maximum length, and so on.

  Each <attributeDefinitionReference> element contains the following attributes:

  — Name — the name of the reference.

  — Required— if this attribute is required in the provisioning (set to true or false).

  — Concero:tafield — the name of the Select Identity resource attribute.

— Concero:resfield — the name of the physical resource attribute from the resource schema. If the resource does not support an explicit schema (such as UNIX), this can be a tag field that indicates a resource attribute mapping.

— Concero:isKey — An optional attribute that, when set to true, specifies that this is the key field to identify the object on the resource. Only one <attributeDefinitionReference> can be specified where isKey="true". This key field does not need to be the same as the key field of the identity object in Select Identity.

— Concero:init — An optional attribute that identifies that the attribute is initialized with the value of the attribute passed in from Select Identity.

Here is an example:

```
<memberAttributes>
  <attributeDefinitionReference
   concero:resfield="schema=SISQL1_ADMIN,
   table=USERINFO_1,column=ADDRESS" concero:tafield=""
   encrypt="false" encryptionAlgorithm="" fk="" iTK="false"
   isKey="false" isPassword="false"
   name="schemaSISQL1_ADMINtableUSERINFO_1columnADDRESS"
   required="false" supportedOperations="UNLINK,LINK,
   GETPARENT,GETCHILDREN,GETALL,GETAATTRIBUTES,
   RESETPASSWORD,CHANGEPASSWORD,EXPIREPASSWORD,
   CREATE,DELETE,ENABLED,UPDATE" type="string"/>
...
```

The interpretation of the mapping between the connector field (as specified by the Concero:tafield attribute) and the resource field (as specified by the Concero:resfield attribute) is determined by the connector. The DB2 connector has code to interpret the mappings in one way, as follows:

— The connector attribute names are specified in tafield. The value of attribute xyz is taken from the UserModel during provisioning.

— Composite attributes can be specified in the DB2 connector mapping file. To do this, specify [attr1] xxxx [attr2] as the connector attribute. This specifies that the value of the attr1 and attr2 attributes should be combined with the string xxxx to form a mapping for the specified resource field. The DB2 connector has code to handle these composite mappings.

You must specify static text (strings) in composite attributes withn brackets ({ }). Also, if no string separates two connector attributes, you must add a space that is within brackets, like this: attr1{ }attr2.

- **<attributeDefinition>**

  Defines the properties of each object's attribute. For example, the attribute definition for the Directory attribute defines that it must be between one and 50 characters in length and can contain the following letters, numbers, and characters: a-z, A-Z, 0-9, @, +, and a space.

  Here is an example:

```
<attributeDefinition
description="schemaSISQL1_ADMINtableUSERINFO_1columnADDRESS"
name="schemaSISQL1_ADMINtableUSERINFO_1columnADDRESS"
type="xsd:string">
  <properties>
    <attr name="minLength">
      <value>0</value>
    </attr>
    <attr name="maxLength">
      <value>255</value>
    </attr>
    <attr name="defaultValue">
      <value>null</value>
    </attr>
    <attr name="pattern">
      <value><![CDATA[[a-zA-Z0-9@]+]]></value>
    </attr>
  </properties>
</attributeDefinition>
```

- **<concero:entitlementMappingDefinition>**

  Defines how entitlements are mapped to users.

- **<concero:objectStatus>**

  Defines how to assign status to a user.

- **<concero:relationshipDefinition>**

  Defines how to create relationships between users.

Refer to the `SampleXML` subdirectory in the `IBM DB2 - Generic` directory on the Select Identity Connector CD for a sample XML file for this connector.

Elements in the XSL Reverse Mapping File

If the agent is installed on the resource and you wish to enable reverse synchronization, you must create an XSL file to map all attributes that are specified in the XML mapping file. The Attribute Mapping Utility generates a corresponding XSL file when it generates the XML file. Use the examples in this section to customize the XSL files for a different schema. See the `SampleXML` subdirectory in the `IBM DB2 - Generic` directory on the Select Identity Connector CD for a full sample.

If the database schema changes and you need of a new set of mapping files, you can load the existing XML file in the Attribute Mapping Utility and modify it as needed. Using a text editor to edit the XML and XSL files is error-prone.

Also, note that the elements in the XSL file are case sensitive.

You must define the user's ID field on the resource and in Select Identity. In the following example, RES_USERID is the user ID resource attribute for the user on the resource. The RES_PASSWORD is the corresponding password attribute on the resource. The following provides an example for setting these attributes:

```
<xsl:variable name="RES_USERID"
select="'schema=dbo,table=USERINFO_1,column=USERID'"/>
```

```
<xsl:variable name="RES_PASSWORD"
select="'schema=dbo,table=USERINFO_1,column=PASSWORD'"/>
```

Note that if you generated the XSL file using the Attribute Mapping Utility, the RES_USERID and RES_PASSWORD attributes are the only ones you must edit in this file. If you are creating the XSL file using a text editor, you must edit additional attributes, as described below.

SI_USERID is the Select Identity attribute for the user ID, and SI_PASSWORD is the Select Identity attribute for the password. The following shows how to set these attributes:

```
<xsl:variable name="SI_USERID" select="'username'"/>
```

```
<xsl:variable name="SI_PASSWORD" select="'Password'"/>
```

For each resource attribute, you must define a corresponding Select Identity attribute, which defines the attribute in Select Identity to which the resource attribute is mapped. The following example defines the RES_ATTR0 resource attribute and the SI_ATTR0 attribute in Select Identity:

```
<xsl:variable name="RES_ATTR0" select="'xxxxxxxxxx'"/>

<xsl:variable name="SI_ATTR0" select="'xxxxxxxxxx'"/>
```

Then, define the resource attribute, such as in this example for RES_ATTR0:

```
<xsl:when test="$ATTRNAME = $RES_ATTR0">
  <xsl:call-template name="AttributeBuilder">
    <xsl:with-param name="DSMLELEMENT" select="$DSMLELEMENT"/>
    <xsl:with-param name="ATTRNAME" select="$SI_ATTR0"/>
    <xsl:with-param name="ATTRVALUE" select="$ATTRVALUE"/>
    <xsl:with-param name="MODIFYFLAG" select="$MODIFYFLAG"/>
  </xsl:call-template>
</xsl:when>
```

Refer to the `SampleXML` subdirectory in the `IBM DB2 - Generic` directory on the Select Identity Connector CD for a sample XSL file for this connector.

4

Uninstalling the Connector

If you need to uninstall a connector from Select Identity, make sure that the following are performed:

- All resource dependencies are removed.
- The connector is deleted through the Connectors home page on the Select Identity client.

Uninstalling the Connector from WebLogic

Perform the following to delete a connector:

1. Log on to the WebLogic Server Console.
2. Navigate to ***My_Domain*** → **Deployments** → **Connector Modules**.
3. Click the delete icon next to the connector that you want to uninstall.
4. Click **Yes** to confirm the deletion.
5. Click **Continue**.

Uninstalling the Connector from WebSphere

Complete the following steps to uninstall the connector on WebSphere:

1. Log on to the WebSphere Application Server Console.
2. Navigate to **Resources** → **Resource Adapters**.
3. Select the connector to uninstall.
4. Click **Delete**.
5. Click the **Save** link (at the top of the page).
6. On the Save to Master Configuraton dialog, click the **Save** button.

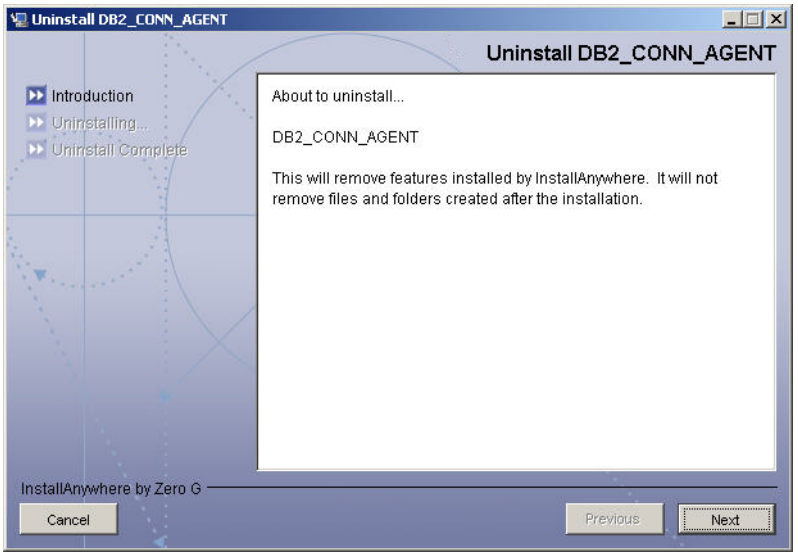
Uninstalling the Agent

The following sections describe how to remove the agent, which you can do using a wizard or manually.

Using the Wizard to Remove the Agent on Windows

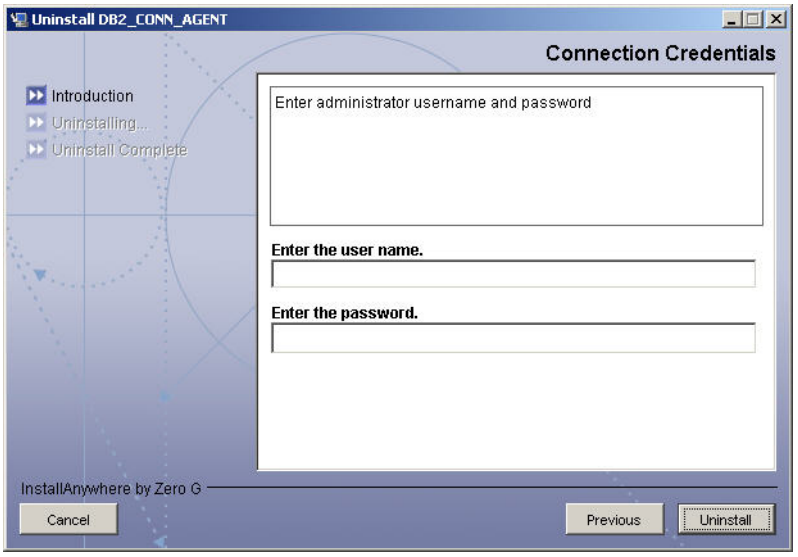
Perform the following steps to delete the agent on the Windows server:

1. Select **Programs** → DB2_CONN_AGENT → **Uninstall Agent** from the Start menu. The wizard displays.



2. Click **Next** on the introductory dialog.

3 Provide the database credentials to uninstall the reverse triggers, if they were installed. Then, click **Uninstall**.



4 Click **Continue** when the pop-up dialog indicates that the triggers were successfully uninstalled.

5 Click **Done** on the Uninstall Complete dialog to close the wizard.

Using the Wizard to Remove the Agent on UNIX

Perform the following steps to delete the agent on the UNIX server:

1 Start the wizard by running the following command:

***agent_home*/Uninstall_DB2_CONN_AGENT/**
Uninstall_DB2_CONN_AGENT

The following displays:

```
===============================================================
Preparing CONSOLE Mode Installation...

===============================================================
(created with InstallAnywhere by Zero G)
---------------------------------------
```

```
===============================================================
Uninstall DB2_CONN_AGENT
-------------------------------

About to uninstall...

DB2_CONN_AGENT

This will remove features installed by InstallAnywhere. It will
not remove files and folders created after the installation.

PRESS <ENTER> TO CONTINUE:
```

2 Press ENTER to continue. The following displays:

```
===============================================================
Get User Input
--------------

Enter requested information

Enter user name :  (DEFAULT: ): TEST
```

3 Enter the database user name and press ENTER. The following displays:

```
===============================================================
Get User Input
--------------

Enter requested information

Enter password :  (DEFAULT: ): password
```

4 Enter the user's password and press ENTER. The installer removes the reverse triggers (if installed) and displays a success or failure message, as follows:

```
===============================================================
Executed the command
--------------------

"/DB2_CONN_AGENT/Uninstall.sh" -userName "TEST" -password
"password"

Reverse Notification Trigger Uninstall Summary

Reverse Trigger Uninstall SUCCEEDED.
```

5 To view the log file, select the Show Logs and press ENTER.

6 Press ENTER to exit the wizard.

Manually Removing the Agent

Perform the following steps to manually remove the agent:

1. Make sure that the *agent_home*\conf\properties.ini file retains the same values used during the installation of the reverse triggers.
2. Make sure that the mapping file during the installation of the agent is available in the *agent_home*\conf\com\trulogica\truacess\connector\schema\spml folder.
3. Run the uninstall.cmd file (on Windows) or uninstall.sh file (on UNIX).
4. Provide the database login credentials when prompted.
5. Delete the agent files and directory structure, if you wish.

A

Troubleshooting

This appendix describes common problems encountered during the installation and use of the connector and its agent.

Connector Installation

This section lists the common problems encountered during installation and use of the connector.

- After redeploying the connector, Select Identity does not display the current connector information.

  Possible Cause: The application is using a cached connector file.

  Solution: Restart the application server.

- Select Identity does not display the most current mapping file information.

  Possible Cause: The application server is using a cached mapping file.

  Solution: Restart the application server.

- The mapping file of a existing resource is changed and, when you attempt to modify the resource to add a new mapping file, the following error displays:

  ```
  Application cannot be modified at this time
  ```

Possible Cause: Major differences may exist between the old and new mapping files.

Solutions:

— Create a new resource with the new mapping file.

— Unmap all attributes in the current resource and modify the resource to reference the new mapping file. You cannot use this second solution, however, if users were provisioned using this resource.

- Select Identity can successfully add a user but the new user is not shown in the resource's database table.

  Possible Causes:

  — The mapping file lacks the Create operation for the Key attribute.

  — The Create operation for the User entity is not added in the XML file.

  — The XML parser files may be missing from the `BEA_HOME/jdk_1.4.1/jre/lib/endorsed` folder (on WebLogic).

  — A database exception occurred.

  Solutions:

  — Add the Create operation or add the relevant JARs to the path. Refer to the *HP OpenView Select Identity Attribute Mapping Utility User's Guide* for details on how to add create operations for an entity.

  — If a database exception occurred, refer to the logs for details of the exception. Common exceptions include size mismatches for columns and foreign key constraint violations. Refer to the database documentation for more information on the database exceptions.

Agent and Trigger Installation

This section lists the common problems encountered while installing and configuring reverse synchronization.

- An error message similar to one of the following is displayed while installing the agent:

  ```
  Object already exists
  ```

  ```
  Table_Audit (or Column_Audit) already exists
  ```

  Possible Cause: Triggers or audit tables exist, possibly from a prior attempt to install and configure the agent.

Solution: Run `uninstall.cmd` (on Windows) or `uninstall.sh` (on UNIX), which removes the triggers from the database. Verify that the Table_Audit, Column_Audit, and SID_TAB tables were removed from the database. If removal was not successful, delete the tables manually before installing the agent triggers.

- A NullPointerException occurs

  Possible Cause: The specified mapping file is not available in the class path.

  Solution: Make sure that the file is placed in the `Install/conf` directory. Ensure the name of the file specified in `properties.ini` is spelled correctly. Note that it is case sensitive. Also, check the format of the mapping file.

Agent Execution

This section lists the common problems encountered while running the agent.

- An exception similar to the following is displayed:

  ```
  java.net.BindException: Address in use: JVM_Bind
  ```

  Possible Cause: The listening port on the agent's system is in use, possibly by another invocation of the agent.

  Solution: Stop the older invocation and run the agent again.

- An error message similar to the following is displayed:

  `Invalid Object` *`schema.tableName`*

  Possible Cause: The schema specified in the mapping file is incorrect.

  Solution: Check the mapping file. For more information on the format of mapping file, see the *HP OpenView Select Identity Attribute Mapping Utility User's Guide*.

- An error message similar to the following is displayed:

  ```
  Invalid Object Table_Audit or Column_Audit
  ```

  Possible Cause: Audit tables are deleted or moved, or they are inaccessible to the triggers. If a trigger fails, the operation that caused the trigger is also rolled back.

  Solution: Make sure that the audit tables (Table_Audit, Column_Audit) are available. If that does not work and the connector's operations are

failing, triggers and audit tables can be uninstalled, though this will cause reverse synchronization to stop.

- The agent console shows a Log4jFactory exception when started.

  Possible Cause: The agent cannot find the `log4j-1.2.8.jar` in the classpath.

  Solution: Add the JAR to `BEA_HOME/jdk1_4_05/jre/lib/ext` (on WebLogic).