

HP OpenView Select Identity

Connector for Microsoft® SQL Server (Administration)

Connector Version: 3.6

Installation and Configuration Guide

Document Release Date: July 2006
Software Release Date: July 2006



Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

© Copyright 2006 Hewlett-Packard Development Company, L.P.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>). Portions Copyright (c) 1999-2003 The Apache Software Foundation. All rights reserved.

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

This product includes software developed by Teodor Danciu (<http://jasperreports.sourceforge.net>). Portions Copyright (C) 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 (c) 2001-2004, Gaudenz Alder. All rights reserved.

Trademark Notices

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

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

JAVA™ is a US trademark of Sun Microsystems, Inc.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

Oracle® is a registered US trademark of Oracle Corporation, Redwood City, California

UNIX® is a registered trademark of The Open Group.

Support

Please visit the HP OpenView support web site at:

<http://www.hp.com/managementsoftware/support>

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

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 valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit enhancement requests online
- Download software patches
- 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://www.hp.com/managementsoftware/access_level

To register for an HP Passport ID, go to:

<http://www.managementsoftware.hp.com/passport-registration.html>

Contents

1	Documentation Map	7
2	Introduction	9
	About HP OpenView Select Identity	9
	About Connectors	9
	About SQL Server Admin Connector	9
	High-Level Architecture	10
	Overview of Installation Tasks	12
3	Installing the Connector	13
	SQL Server Admin Connector Files	13
	Planning the Installation	13
	Plan 1: Connector with the Agent	14
	Plan 2: Connector Without the Agent and with JDBC Data Source	14
	Plan 3: Connector Without the Agent and with JDBC Driver	15
	System Requirements	15
	Pre-Installation Task	17
	Enable JDBC Driver Based Communication	17
	Enable JDBC Data Source Based Communication	17
	Extracting Contents of the Schema File	17
	WebLogic	18
	WebSphere	18
	Installing the Connector RAR	18
4	Configuring the Connector with Select Identity	19
	Configuration Procedure	19
	Add a New Connector	19
	Add a New Resource	19
	Map Attributes	22
5	Installing the Agent	25
	About the Agent	25
	Installing the Agent on the Microsoft SQL Server	25
	Pre-Installation Tasks	25
	Install the Agent	26
	Installed Files	36
	Starting the Agent	37
	Modifying the Database Account and Select Identity Passwords	37

6	Uninstalling the Connector	39
	Uninstalling the Agent	39
A	Troubleshooting	41
	Connector Installation	41
	Agent and Reverse Notification Table Installation	42
	Agent Execution.....	43
B	Connector Behavior	45
	Using Entitlements to Assign User Privileges on Databases	45
	Entitlement Categories	45
	Recommended Usage of Entitlements while User Provisioning	46
	Example Use Case	46

1 Documentation Map

This chapter describes the organization of HP OpenView Select Identity connector documentation and provides necessary information on how to use the documentation set to install and configure the connectors.

[Figure 1](#) illustrates the documentation map for HP OpenView Select Identity connector. For a list of available product documentation, refer to the [Table 1](#).

Figure 1 Documentation Map

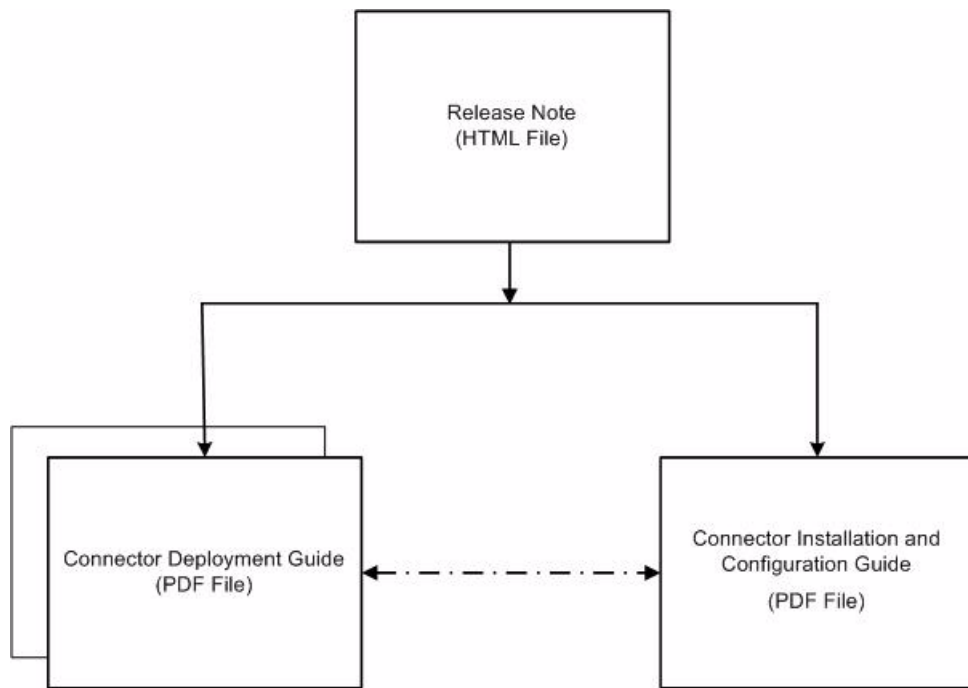


Table 1 Connector Documentation

Document Title and Filename	Contents	Location
<i>Release Note</i> SQL Server Admin Connector v3.6 Release Note.htm	This file contains necessary information on new features of the connector, enhancements, known problems or limitations, and support information.	/Docs/ subdirectory under the connector directory.
<i>Connector Deployment Guide (for Select Identity 4.0/4.01.000)</i> connector_deploy_SI4.pdf	Connector deployment guides provide detailed information on: <ul style="list-style-type: none"> • Deploying a connector on an application server. • Configuring a connector with Select Identity. Refer to these guides when you need generic information on connector installation.	/Docs/ subdirectory under the connector directory.
<i>Connector Deployment Guide (for Select Identity 3.3.1)</i> connector_deploy_SI3.3.1.pdf		
<i>Connector Installation and Configuration Guide</i> SQL Server Admin_install.pdf	Connector installation and configuration guide provides installation instructions for a specific connector. It contains resource specific configuration details.	/Docs/ subdirectory under the connector directory.

2 Introduction

This chapter gives an overview of the HP OpenView Select Identity connector for Microsoft SQL Server database administration. An HP OpenView Select Identity connector allows you to provision users and manage identities on Microsoft SQL Server. At the end of this chapter, you will be able to know about:

- The benefits of HP OpenView Select Identity.
- The role of a connector.
- The connector for Microsoft SQL Server database.

About HP OpenView Select Identity

HP OpenView Select Identity provides a new approach to identity management. It helps you manage the entire identity lifecycle of an enterprise application. By using Select Identity, you can automate the process of provisioning and managing user accounts and access privileges across platforms, applications, and corporate boundaries. Select Identity communicates with the enterprise information system through connectors, and automates the tasks of identity management. The enterprise information system, which is also referred to as **resource**, can be a database, a directory service, or an ERP package, among many others.

About Connectors

You can establish a connection between a resource and Select Identity by using a connector. A connector is resource specific. It is installed on the system where Select Identity is installed. The combination of Select Identity and connector helps you perform a set of tasks on the resource to manage identity. A connector can be **unidirectional** or **bidirectional**. A unidirectional connector helps you manage identities from Select Identity, but if any change takes place in resource, it cannot communicate that back to Select Identity. On the other hand, a bidirectional connector can reflect the changes made on resource back to Select Identity. This property of bidirectional connectors is known as **reverse synchronization**.

About SQL Server Admin Connector

The connector for Microsoft SQL Server administration — hereafter referred to as the SQL Server Admin connector — enables HP OpenView Select Identity to administer the database server by provisioning database user information in system schemas. This connector supports

two type of database users, namely Microsoft SQL Server Authentication users and Windows Authentication users. The SQL Server Admin connector can perform the following operations in a database schema on the Microsoft SQL Server:

- For Microsoft SQL Server authentication users:
 - Add, update, and remove users
 - Retrieve user attributes
 - Enable and disable users
 - Verify a user's existence
 - Reset Password
 - Change Password
 - Retrieve all entitlements
 - Retrieve a list of supported user attributes
 - Grant and revoke entitlements to and from users
- For Windows authentication users:
 - Add, update, and remove users
 - Retrieve user attributes
 - Verify a user's existence
 - Enable and disable user status
 - Retrieve all entitlements
 - Retrieve a list of supported user attributes
 - Grant and revoke entitlements to and from users

The connector also provides an agent that can send changes made to the data in Microsoft SQL Server to Select Identity. The following reverse synchronization operations are supported:

- Add, modify, and delete users based on user additions, modifications, and deletions in the schema in Microsoft SQL Server
- Assign and revoke entitlements for users

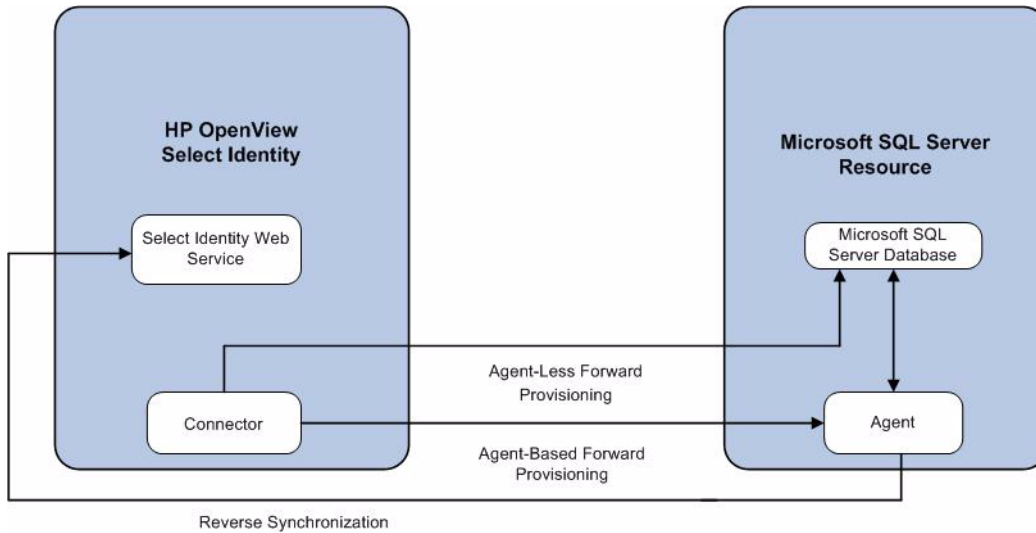


This connector can be used with Select Identity 4.01.000, 4.0, and 3.3.1.

High-Level Architecture

Figure 2 illustrates a high-level architecture of the SQL Server Admin connector. The connector supports both agent-based and agent-less mode of operation. To support reverse synchronization, you must install the connector on Select Identity server and the agent on resource system. The agent helps synchronizing the changes made on Microsoft SQL Server with Select Identity.

Figure 2 High-Level Architecture of the Connector



To perform forward provisioning operation on Microsoft SQL Server, the connector communicates either directly with the database or with the agent. The agent detects the changes on the host (Microsoft SQL Server database) resource and sends SPML notifications to Select Identity to synchronize the changes. Thus, the SQL Server Admin connector enables data to flow in both the directions, as illustrated in [Figure 2](#).

Overview of Installation Tasks

Before you start installing the connector, you must ensure that system requirements and all the installation prerequisites are met. Refer to the [Table 2](#) for an overview of installation tasks.

Table 2 Organization of Tasks

Task Number	Task Name	Reference
1	Install the connector on the Select Identity server.	See Installing the Connector on page 13.
	— Plan your installation setup.	See Planning the Installation on page 13.
	— Meet the system requirements.	See System Requirements on page 15.
	— Pre-installation task: Enable JDBC driver or JDBC data source based on your requirement.	See Pre-Installation Task on page 17.
	— Extract contents of the Schema file (the file that contains the mapping XML and XSL files).	See Extracting Contents of the Schema File on page 17.
	— Deploy the connector Resource Adapter Archive (RAR) file on an application server.	See Installing the Connector RAR on page 18.
2	Configure the connector with Select Identity	See Configuring the Connector with Select Identity on page 19.
3	Install the agent on Microsoft SQL Server database server.	See Installing the Agent on page 25.
	— Perform the pre-installation tasks.	See Pre-Installation Tasks on page 25.
	— Install the agent by using the installation wizard.	See Install the Agent on page 26.

3 Installing the Connector

This chapter elaborates the procedure to install the SQL Server Admin connector on the Select Identity server. At the end of this chapter, you will know about

- Software requirements to install the SQL Server Admin connector.
- Pre-installation tasks.
- Procedure to install the SQL Server Admin connector.

SQL Server Admin Connector Files

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

Table 3 SQL Server Admin Connector Files

Serial Number	File Name	Description
1	Admin-MSSQL2000Server-Connector.rar	The RAR file contains the binaries for the connector.
2	Admin-MSSQL2000ServerSchema.zip	The Schema file contains the mapping files (XML and XSL files) of the connector.
3	MSSQL2000Server-Admin-AgentInstaller-Win.zip	A ZIP file that contains the installation executable for the connector agent. It is located in the Agent Installers directory of the CD.

Planning the Installation

You can install the SQL Server Admin connector in three possible ways.

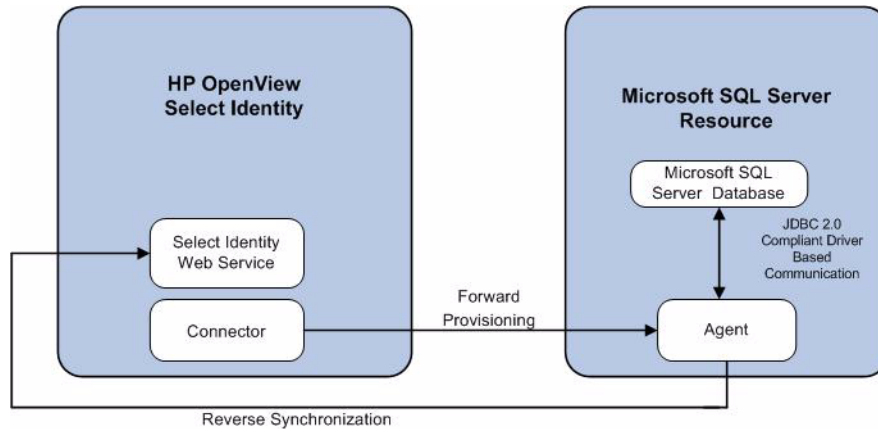
- Connector with the agent.

- Connector without the agent and with a JDBC data source.
- Connector without the agent and with a JDBC driver.

Plan 1: Connector with the Agent

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 SQL Server Admin to the Select Identity database (reverse synchronization).

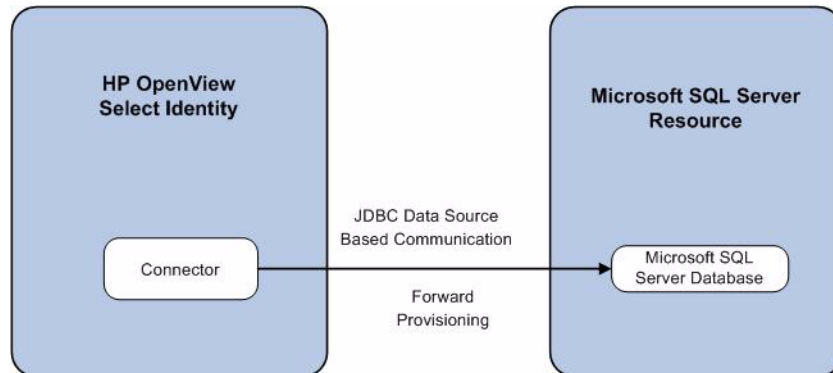
Figure 3 Connector Installed with Agent



Plan 2: Connector Without the Agent and with JDBC Data Source

In this configuration, the connector communicates with the database directly through JDBC calls. You must create or identify a JDBC data source (and underlying connection pool) on the application server hosting Select Identity and the connector that can connect to the target SQL Server Admin database. Reverse synchronization is not achieved in this configuration.

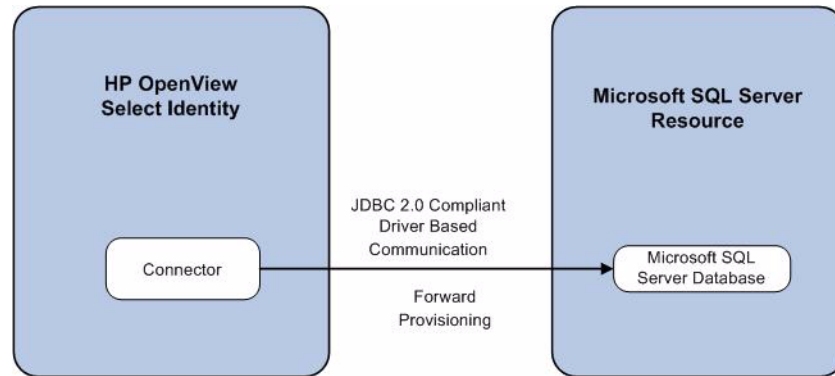
Figure 4 Connector Without Agent: JDBC Data Source Based Communication



Plan 3: Connector Without the Agent and with JDBC Driver

In this configuration, the connector communicates with the database by using a JDBC 2.0 compliant driver; no agent is installed on the database server. Reverse synchronization is not achieved in this configuration.

Figure 5 Connector Without Agent: JDBC 2.0 compliant Driver Based Communication



System Requirements

The SQL Server Admin connector is supported in the following environment:

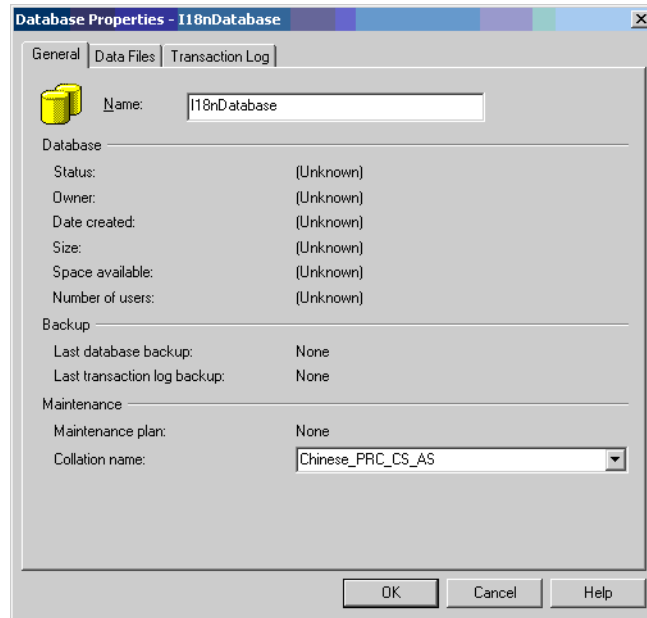
Table 4 Platform Matrix for SQL Server Admin Connector

Select Identity Version	Application Server	Database
3.3	WebLogic 8.1.4 on Windows 2003	Microsoft SQL Server 2000
	WebLogic 8.1.4 on Solaris 9	Oracle 9i
	WebLogic 8.1.4 on HP-UX 11i	Oracle 9i
3.3.1	WebLogic 8.1.4 on Windows 2003	Microsoft SQL Server 2000
	WebLogic 8.1.4 on Solaris 9	Oracle 9i
	WebLogic 8.1.4 on HP-UX 11i	Oracle 9i
	WebSphere 5.1.1 on HP-UX 11i	Oracle 9i
	WebSphere 5.1.1 on Windows 2003	Oracle 9i
4.0/4.01.000	SQL Server Admin connector is supported on all the platform configurations of Select Identity 4.0/4.01.000.	

The SQL Server Admin Connector is supported for Microsoft SQL Server 2000 on Windows 2003, Windows 2000 and Windows XP.

SQL Server Admin connector is internationalized and able to operate with languages that are supported by the Java Unicode specification. If you want to use the connector on non-English platforms, make sure that the following prerequisites are met:

- The Select Identity server should be configured for internationalization. Refer to the *HP OpenView Select Identity Installation Guide* for more information.
- SQL Server Admin connector can support internationalization if the Collation Name is set appropriately when the database is created. For Microsoft SQL Server 2000, the Collation Name is set by default to the Local Language type:



The SQL Server Admin connector supports Microsoft SQL Server 2000 resource with the following three types of environments:

Environment 1

Microsoft SQL Server installed in English environment with master and all other databases in English (default) collation.

Environment 2

Microsoft SQL Server installed in local language environments with master and all other databases in local language collation.

Environment 3

Microsoft SQL Server installed in English environment with master database as English (default) collation and some of the other databases as local language collation. Note that only one collation type (apart from English) is supported. For example, master database can be in English (SQL_Latin1-General_CP1_CI_AS) collation name and the SQL Server can contain one (or more) Databases which may have Chinese-PRC_CI_AS collation name.

Pre-Installation Task

Before you start installing, you must enable the communication mode between the connector and Microsoft SQL Server database according to your installation plan.

Enable JDBC Driver Based Communication

To enable a JDBC 2.0 compliant driver based communication (for Plan1 or Plan3), you must copy the files `msbase.jar`, `mssqlserver.jar`, and `msutil.jar` on the Select Identity server. Perform the following steps to enable JDBC driver based communication:

- 1 Obtain the files `msbase.jar`, `mssqlserver.jar`, and `msutil.jar`.
- 2 For Select Identity on WebLogic:
 - a Copy the files to a location on the Select Identity server.
 - b Add the file to the application server's CLASSPATH. To add the files to the application server's CLASSPATH:
 - Edit the startup script `myStartWL.cmd` for WebLogic on Windows.
 - Edit the startup script `myStartWL.sh` for WebLogic on UNIX.
- 3 For Select Identity on WebSphere, copy the `msbase.jar`, `mssqlserver.jar`, and `msutil.jar` files to `%WAS_HOME%/lib/ext/` where `%WAS_HOME%` is a location like `D:\WebSphere\AppServer`.

Enable JDBC Data Source Based Communication

To enable a JDBC data source based communication between the connector and the Microsoft SQL Server database (Plan2), you must create a new or use an existing JDBC data source and an underlying connection pool on the application server that hosts Select Identity. This data source is not the Select Identity data source but different connection pool and data source that point to the target database where the connector has to provision users.

While creating a new JDBC data source on WebLogic, you must do the following:

- Cancel the selection Honor Global Transactions.
- Select the option Emulate Two-Phase Commit for non-XA Driver.



This configuration enables the newly created data source to co-exist with the Select Identity data source.

Extracting Contents of the Schema File

The Schema file (`Admin-MSSQL2000ServerSchema.zip`) contains the mapping information of the connector. Extract contents of this file to a location on the Select Identity server. You will obtain an XML file (`AdminServerSQL2000.xml`) and an XSL file (`AdminServerSQL2000.xsl`). Perform one of the following procedures depending on the application server ([WebLogic](#) or [WebSphere](#)) on which the connector will be deployed.

WebLogic

- 1 Identify a directory that is available in WebLogic CLASSPATH.
- 2 Place the XSL file under this directory.
- 3 Place the XML file in the path `com\trulogica\truaccess\connector\schema\spml` under this directory.

WebSphere

- 1 `<WebSphere_Install_Dir>/AppServer/lib/ext` is the default directory in WebSphere CLASSPATH. Place the XSL file directly under it.
- 2 Place the XML file in the path `com\trulogica\truaccess\connector\schema\spml` under the `<WebSphere_Install_Dir>/AppServer/lib/ext` directory.

Installing the Connector RAR

To install the RAR file of the connector (`Admin-MSSQL2000Server-Connector.rar`) on the Select Identity server, you must copy the file to a local subdirectory on the Select Identity server, and then deploy on the application server. Refer to the *HP OpenView Select Identity Connector Deployment Guide* for detailed information on deploying a RAR file on an application server.



While deploying the RAR on WebSphere, enter the JNDI Pool Name as `eis/Admin-MSSQL2000Server`.

4 Configuring the Connector with Select Identity

This chapter describes the procedure to configure the SQL Server Admin connector with Select Identity. At the end of this chapter, you will know the procedure to configure the SQL Server Admin connector with Select Identity.

Configuration Procedure

After you deploy the connector RAR on application server, you must configure the connector with Select Identity. Perform the following steps to configure the SQL Server Admin connector with Select Identity.

- 1 Add a New Connector
- 2 Add a New Resource
- 3 Map Attributes

Add a New Connector

Add a new connector in Select Identity by using the user interface. While adding the connector, do the following:

- In the Connector Name text box, specify a name for the connector.
- In the Pool Name text box, enter `eis/Admin-MSSQL2000Server`.
- Select **Yes** for the Mapper Available section.

Refer to the *HP OpenView Select Identity Connector Deployment Guide* for detailed information on adding a new connector in Select Identity.

Add a New Resource

Add a new resource in Select Identity that uses the newly added connector. Refer to the *HP OpenView Select Identity Connector Deployment Guide* for detailed instructions on adding a resource in Select Identity.

Refer to the following table while entering the parameters in the Basic Information and the Access Information pages:

Table 5 Resource Configuration Parameters

Field Name	Sample Values	Description	Comment
Resource Name	Admin-SQL2000	The name of the resource.	
Connector Name	AdminSQL	The newly deployed connector.	Known as Resource Type in Select Identity 3.3.1.
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		Whether the system uses the concept of groups. For this connector, select this option.	Applicable only to Select Identity 3.3.1.
Server Name	HP0111	Host name or IP address of the database server. You must specify this parameter if the agent was installed.	Leave the field empty if you configure the connector without agent for a JDBC data source based communication.
Server Port	1433	Port on which the database server is listening. You must specify this parameter if the agent was installed.	Leave the field empty if you configure the connector without agent for a JDBC data source based communication
Username	sa	The login name of the database administrative user. You must specify this parameter if the agent was installed.	Leave the field empty if you configure the connector without agent for a JDBC data source based communication
Password	P4ssword	Password of the database administrative user. You must specify this parameter if the agent was installed.	Leave the field empty if you configure the connector without agent for a JDBC data source based communication

Table 5 Resource Configuration Parameters

Field Name	Sample Values	Description	Comment
Agent Port	5601	The port where the agent listens for incoming connections. You must specify this parameter if the agent was installed.	Leave the field empty if you install the connector without agent
SQL URL	jdbc:microsoft:sqlserver	URL to use to communicate with the database over a JDBC connection. You must specify this parameter if the agent was installed.	Leave the field empty if you configure the connector without agent for a JDBC data source based communication
Database / Service Name	testDB	This parameter can have two types of values: <ul style="list-style-type: none"> • If the Microsoft SQL server has all the databases (including master) in same language collation, leave his parameter blank. • If the Microsoft SQL server has some databases in local language collation (with master database in English collation), enter the name of the database which has the local language collation. For example, if db1 has Chinese_PRC_CI_AS collation, then this can be mentioned here. This parameter is used to make the connector compatible with heterogeneous collation types on the SQL Server. 	Leave the field empty if you configure the connector without agent for a JDBC data source based communication
Database Driver String	com.microsoft.jdbc.sqlserver.SQLServerDriver	Name of the JDBC driver to connect to the database. You must specify this parameter if the agent was installed.	Leave the field empty if you configure the connector without agent for a JDBC data source based communication
Mapping File	AdminServerSQL2000.xml	The XML mapping file, which must reside in <code>install/conf/com/truologica/truaccess/connector/schema/spml</code> directory in order for the Select Identity server to find it.	

Table 5 Resource Configuration Parameters

Field Name	Sample Values	Description	Comment
JDBC Datasource String		JNDI data source name that was created or identified on the Select Identity server that can connect to the target Microsoft SQL Server database. Specify a value for this property if the agent was not installed.	Leave the field empty if you configure the connector for JDBC driver based communication (with or without agent).
Encryption Specification Algo		Encryption algorithm specification string.	
Encryption Algorithm		Name of the encryption algorithm.	
Encryption Specification Level		Encryption level specification string. Specify this parameter if you wish to use secure communication with Microsoft SQL Server.	
Encryption Level		Encryption level. Specify this parameter if you wish to use secure communication with Microsoft SQL Server.	

*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.

Map Attributes

After successfully adding a resource for SQL Server Admin connector, you must map the resource attributes to Select Identity attributes. Add new attributes to Select Identity if necessary. Refer to the *HP OpenView Select Identity Connector Deployment Guide* for more information on mapping and creating attributes. While mapping the attributes, refer to the following table for SQL Server Admin attribute mapping information.

Table 6 SQL Server Admin Mapping Information

Attribute Name	Description	Sample Value
Dbname	This attribute holds the default database that is assigned to the user who gets created on SQL Server.	Tempdb
External	This attribute specifies whether this user is authenticated using SQL Server authentication or Windows Authentication	True if the user is a Windows user (Windows Authentication) False if the user is authenticated using SQL Server authentication.
Language	The language used by the user on the database server. Please consult the database administrator for valid values.	Valid Language name on the SQL Server.
Password	The password of the user on the database	
Userid	The username of the user on the database. This should be of the form '<DomainName>\<Username>' for 'Windows Authentication' user.	

After mapping the attributes, you can use the connector to create a service, or you can associate the connector with an existing service. Refer to the *Service Studio* chapter of the *HP OpenView Select Identity Administrator Guide* for information on Select Identity services.

5 Installing the Agent

This chapter gives an overview of the agent for SQL Server Admin connector and the procedure to install the agent on an Microsoft SQL Server. At the end of the chapter, you will be able to know about:

- The role of an agent.
- The procedure to install the agent.

About the Agent

The SQL Server Admin connector agent performs forward provisioning operations on the resource and sends back any changes made on Microsoft SQL Server database to Select Identity web service in the form of SPML requests and sends back any changes made on Microsoft SQL Server database to Select Identity web service in the form of SPML requests. The connector is packaged with agent installer.

When a user is added, modified, or deleted in the database, the agent captures the change from reverse notification table. The agent's reverse synchronization component then sends the change 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). In order to achieve reverse synchronization, you must install and configure the agent.

The SQL Server Admin agent supports secure channel of communication to Select Identity web service by using HTTPS. You must configure the application server with Secure Socket Layer (SSL). You configure the agent to enable secure communication between agent and Select Identity in reverse synchronization. The agent automatically imports the certificate from Select Identity and initializes secure communication.

Installing the Agent on the Microsoft SQL Server

After you install the SQL Server Admin connector on the Select Identity server, you can install the agent on the database server depending on your installation plan. If you do not need reverse synchronization ([Plan 2](#) and [Plan 3](#)), you can skip this chapter. However, agent installation is mandatory if you need reverse synchronization ([Plan 1](#)). The agent enables you to send data back to Select Identity.

Pre-Installation Tasks

Before you start installing the agent on Microsoft SQL Server database, make sure the following prerequisites are met:

- Copy the mapping files to the resource (Microsoft SQL Server) system as the agent installation requires the mapping files to be available on the local system.
- Copy the database driver files `msbase.jar`, `mssqlserver.jar`, and `msutil.jar`. to the Microsoft SQL Server database system and they must be in the database server's CLASSPATH.
- Make sure that Java 1.4.2 (or above) is installed on the system and the `JAVA_HOME` environment variable is set. Also, `%JAVA_HOME%\bin` is specified in the `PATH` system variable.

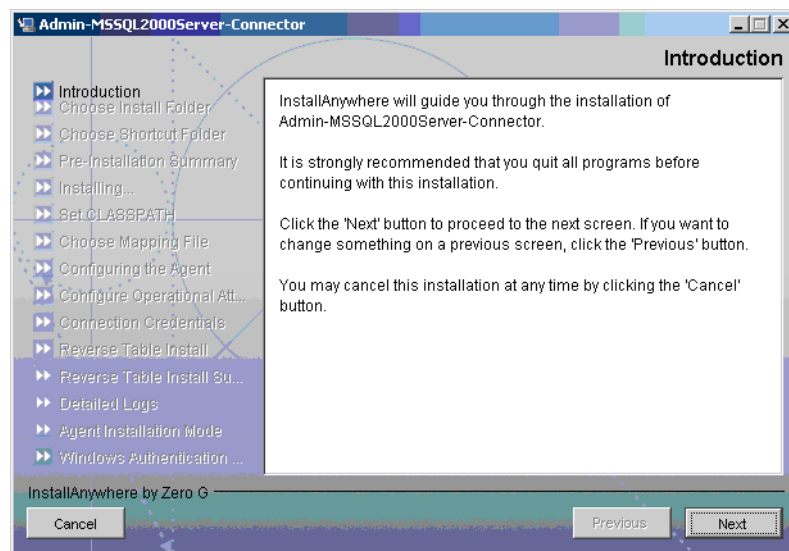


Also, you can pass the `LAX_VM` argument to point the wizard directly to the correct `java.exe` executable. For example: `install.exe LAX_VM c:\java14\bin\java.exe`

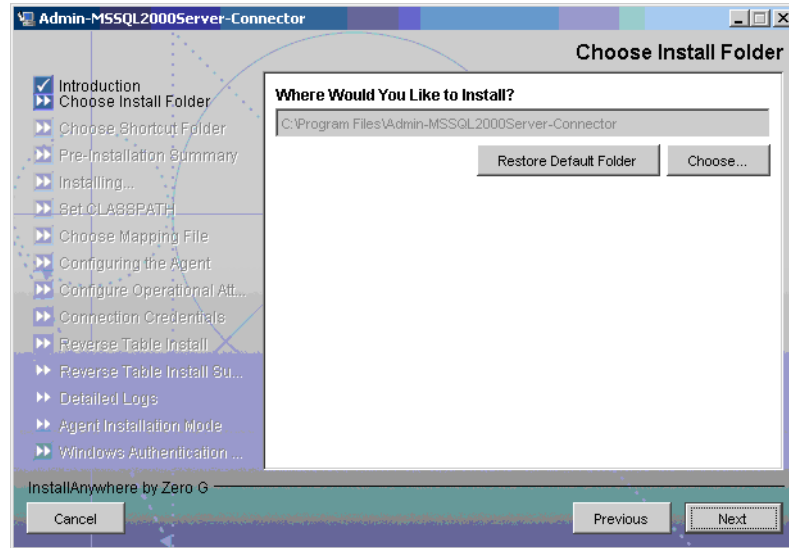
Install the Agent

You can install the agent by using the installation wizard. The wizard is packaged in the file `MSSQL2000Server-Admin-AgentInstaller-Win.zip` for installation on Windows. Perform the following steps to run the installation wizard and install the agent:

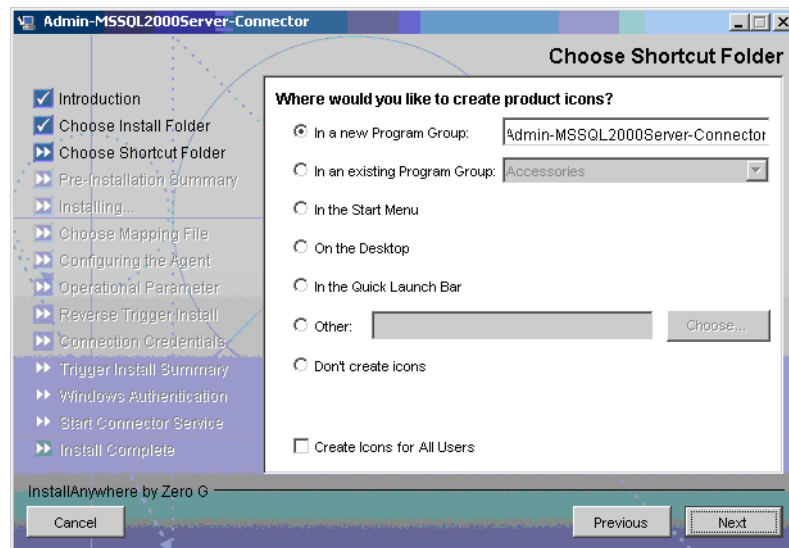
- 1 Extract the contents of the `MSSQL2000Server-Admin-AgentInstaller-Win.zip` file, which is located in the `Agent Installers` directory on the CD.
- 2 Run `Admin-MSSQL2000Server-Connector-Installer.exe`, which is located in the `target_dir\CDROM_Installers\Windows\Disk1\InstData\NoVM`. The **Introduction** screen appears:



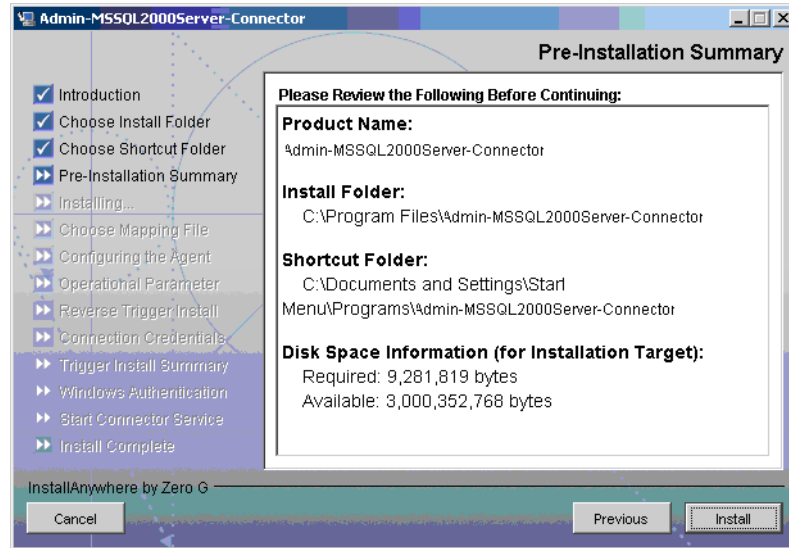
- 3 Click **Next**. The **Choose Install Folder** screen appears.



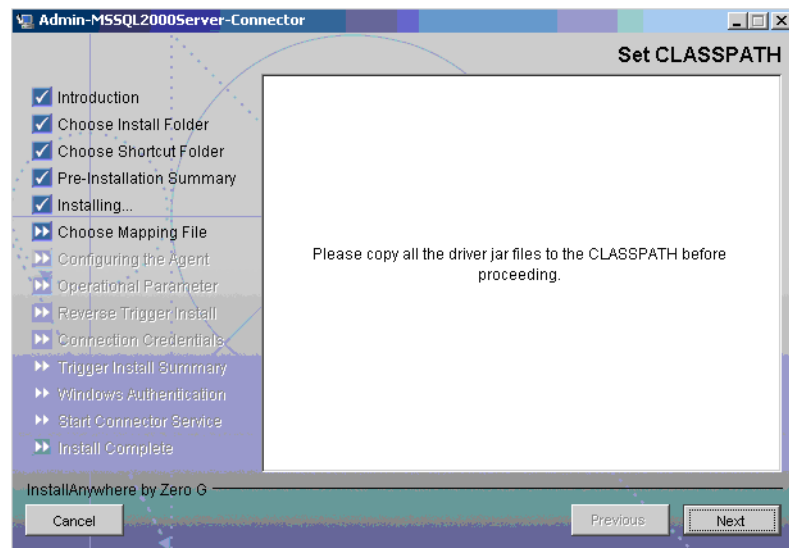
- 4 Keep the default location or click **Choose** to change the location, specify an installation directory, and then click **Next**. The Choose Shortcut Folder screen appears.



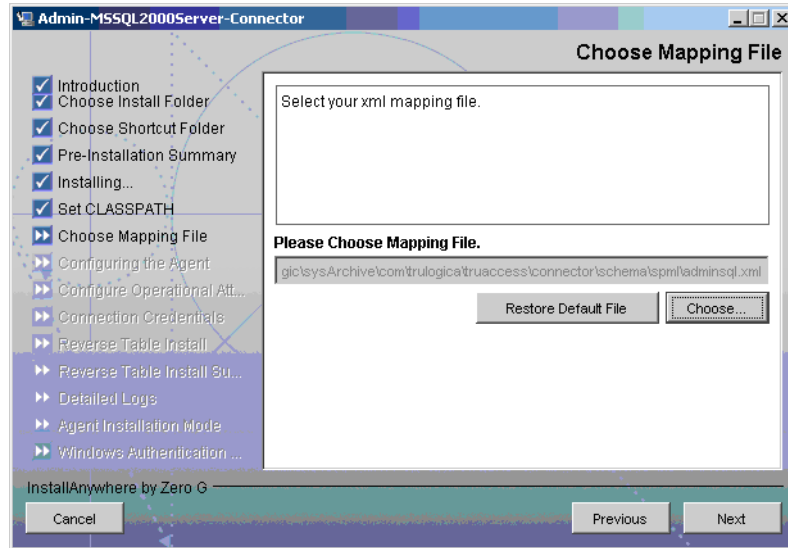
- 5 Select the location(s) where the product icons will be installed, and then click **Next**. The Pre-Installation screen appears.



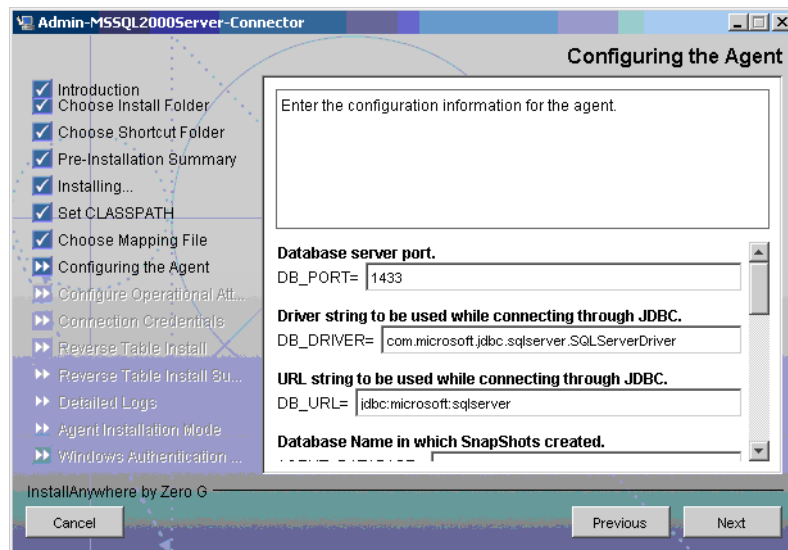
- 6 Review the pre-installation summary, click **Previous** if you want to change any setting, and then click **Install** to begin installation. During the course of installation, the Set CLASSPATH screen appears.



- 7 Verify that the database driver files (`msbase.jar`, `mssqlserver.jar`, and `msutil.jar`) are in the database server's system classpath, and then click **Next**. The Choose Mapping File screen appears.




- 8 Click **Choose** to browse for and select the mapping file. This will copy the mapping file to the `<install_dir>/conf/com/trulogica/truaccess/connector/schema/spml` directory, where `<install_dir>` is the installation folder.
- 9 Click **Next**. The Configuring the Agent screen appears.



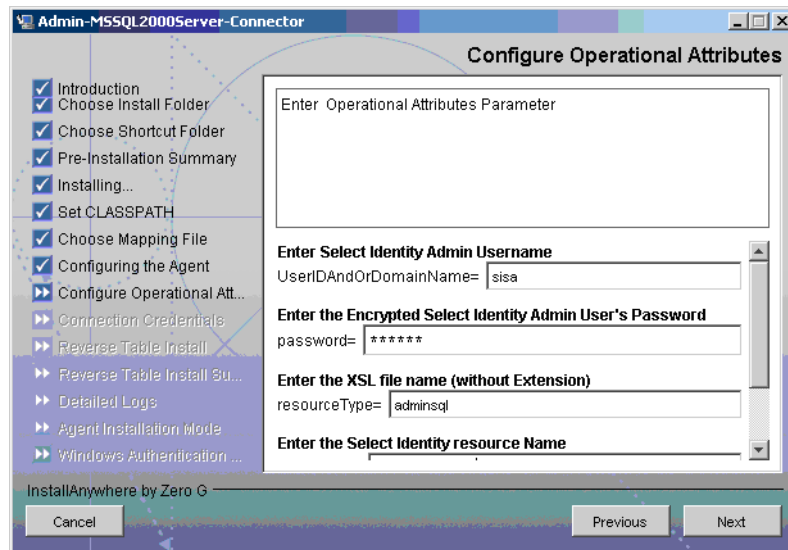
- 10 On the Configuring the Agent dialog, specify the configuration parameters, which are explained in the table below:

Parameter	Description	Example Value
DB_PORT	The port on which the database server is listening.	1433
DB_DRIVER	The JDBC driver for the database connection.	com.microsoft.jdbc.sqlserver. SQLServerDriver
DB_URL	The JDBC URL string used for the database communication.	jdbc:microsoft:sqlserver
AGENT_DATABASE	The database name in which the SNAPSHOT tables of agent will be stored. When the agent is being deployed on the SQL Server that is installed as described in Environment 3 on page 16, this parameter should be the name of the Database which has other than the 'English' collation. For example, if 'db1' has 'Chinese_PRC_CI_AS' collation, then this can be mentioned here. The reason for providing such a database is the notification tables created in it should be capable of holding data of both English and the other non-English language.	
SERVER_SECURE	Whether communication between the agent and Select Identity must be secure. By default, non-secure communication is used.	Select this check box if you want to establish a secure communication (HTTPS).
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
MAPPING_FILE	The XML mapping file.	AdminServerSQL2000.xml

Parameter	Description	Example Value
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
NO_OF_RETRIES	The number of times the agent will retry sending SPML requests in case of failure.	10
DELAY_BETWEEN_RETRIES	The delay (in milliseconds) between each retry.	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**. The Configure Operational Attributes screen appears.

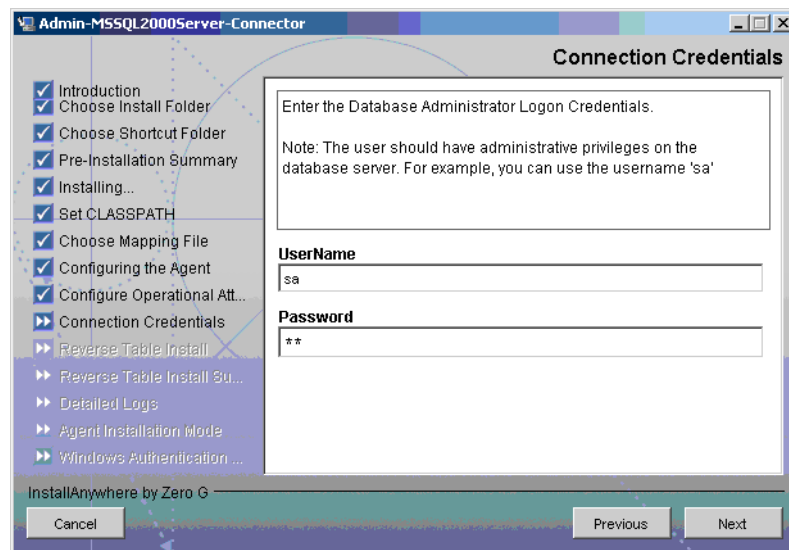


- 11 Provide the operational attributes that are sent to the Select Identity server during reverse synchronization requests. The table below gives a description of the attributes:

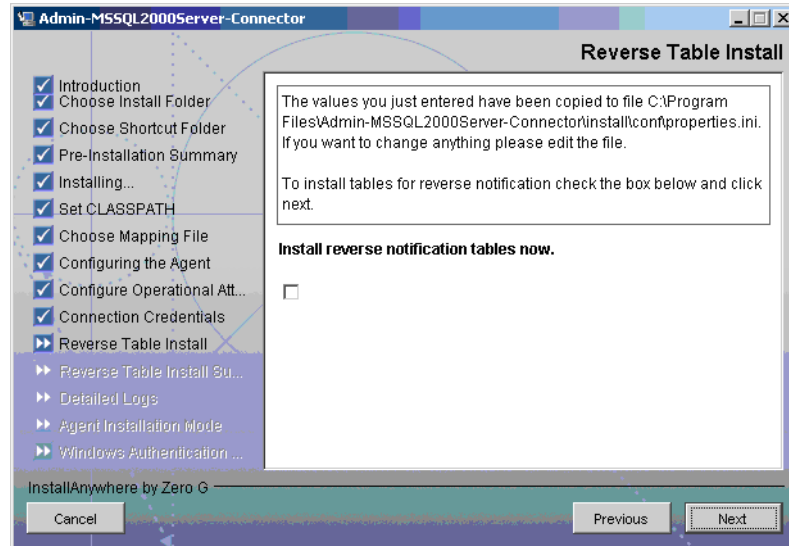
Attribute	Description
UserIDAndOrDomainName	User ID of the administrative user on Select Identity. For example, sisa.
password	Password of the administrative user.
reverseSync	Select this check box to enable reverse synchronization.
resourceType	The name of the XSL file (without the .xsl extension) that is used during reverse synchronization. For example, adminMicrosoft SQL Server.
resourceId	The name of the Select Identity resource that is created for the SQL Server Admin connector. For example, AdminMicrosoft SQL Server-Resource.

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

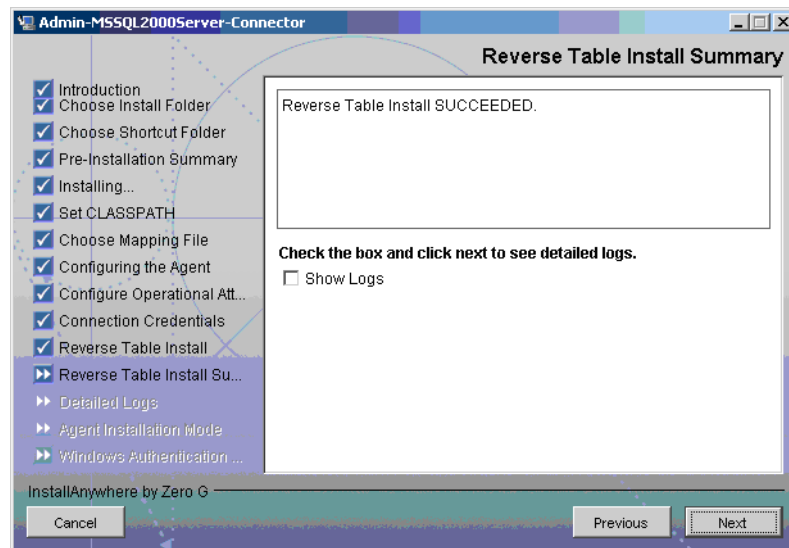
After entering the attributes, click **Next**. The Connection Credentials screen appears.



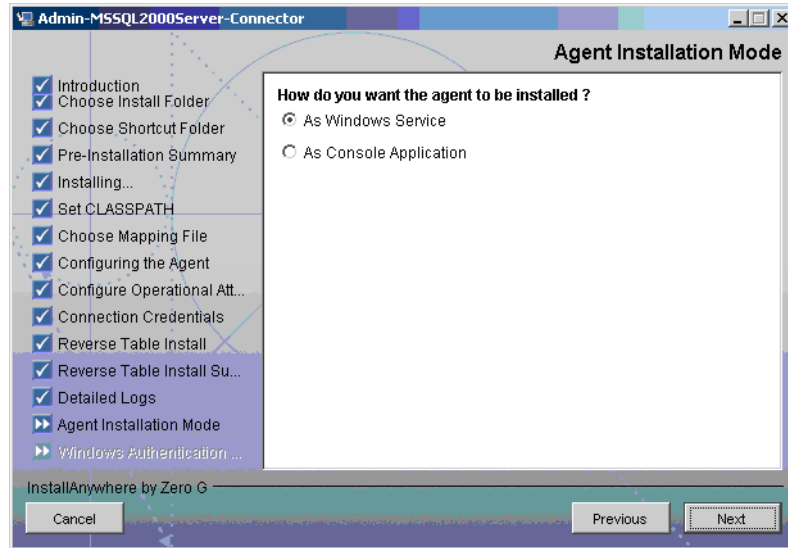
- Enter Username/password for the Microsoft SQL Server database user with which the agent can connect to the Database. This user should have admin privileges, and then click **Next**. The Reverse Table Install screen appears.



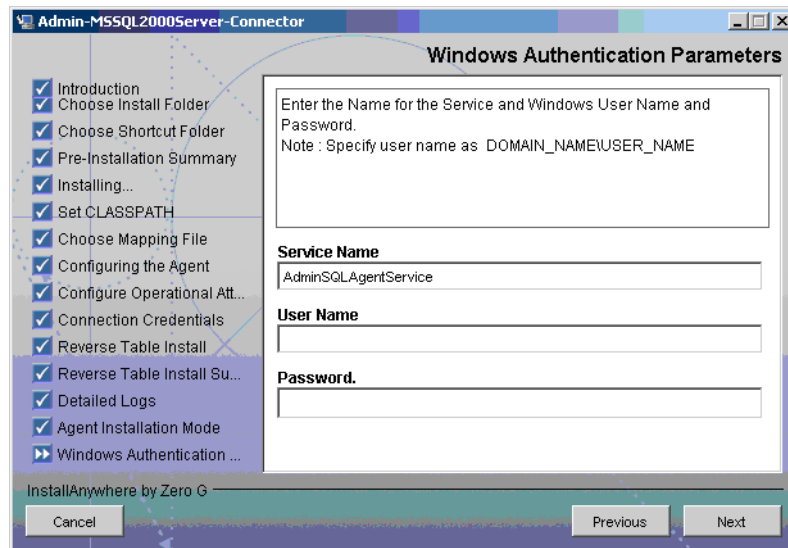
- 13 To enable reverse synchronization, select the Install reverse notification tables now check box, and then click **Next**. The Reverse Table Install Summary table appears.



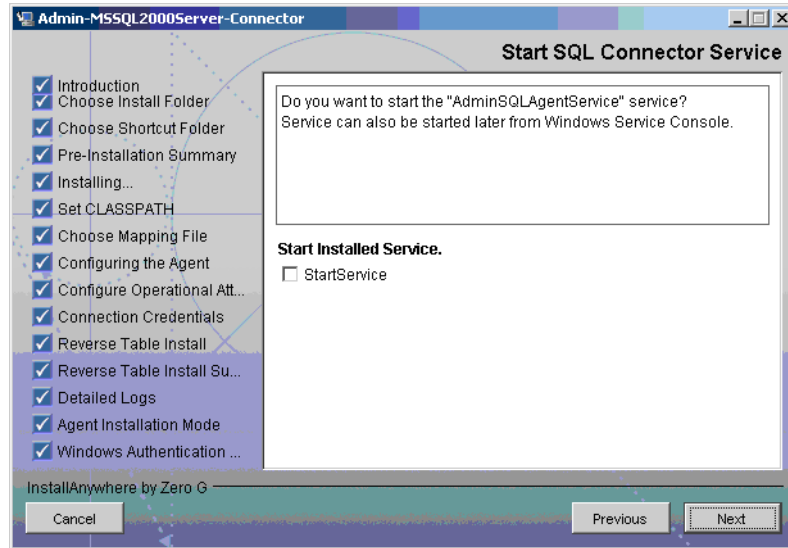
- 14 Select the ShowLog check box to view the detailed log, and then click **Next**. The Detailed Logs screen appears.
- 15 Click **Next**. The Agent Installation Mode screen appears.



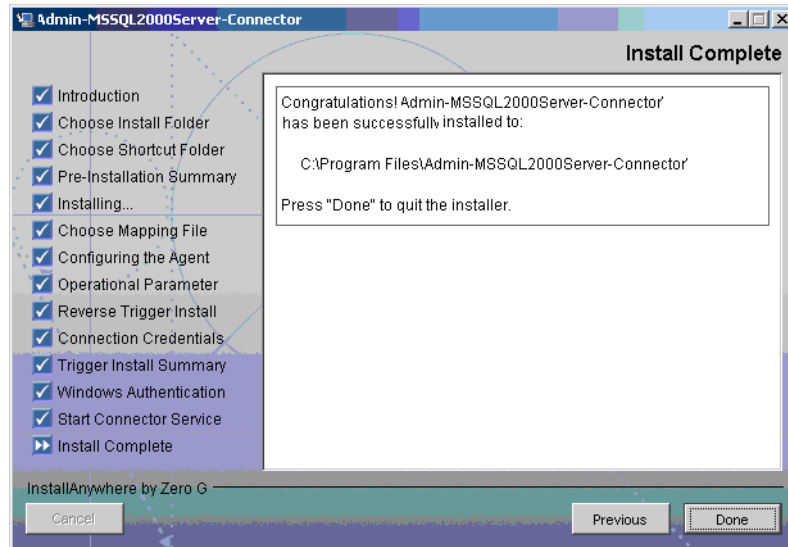
- 16 In the Agent Installation Mode screen, perform one of the following:
 - To run the agent as a Windows service, select the As Windows Service radio button, and then click **Next**. The Windows Authentication Parameters screen appears (step 17).
 - To run the agent as a console application, select the As Console Application radio button, and then click **Next**. The Install Complete screen appears (step 20).
- 17 The Windows Authentication Parameters screen displays the fields to enter the Windows username and password with administrative privilege, and the agent service name.



- 18 Type the agent service name and the administrative username and password for the Windows, and then click **Next**. The Start SQL Connector Service appears.



- 19 Select the StartService check box to start the agent service immediately after installation and **Next**. The Install Complete screen appears.



- 20 In the Install Complete screen, click **Done**.

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: <ul style="list-style-type: none">• AddToStartupGroup.cmd — Adds icons to startup group.• CopyFile.cmd — Used by agent to copy files.• DelFile.cmd — Used by agent to delete files.• AdminSetup.cmd — Installs the reverse notification tables.• sqlapp.jar — Agent library JAR.• SQLConnectorConsole.cmd — Starts the agent.• AdminUninstall.cmd — Uninstalls reverse notification table.• passwordEncrypt.cmd — Utility to populate Properties.ini and opAttributes.properties file with encrypted password.• PortTest.cmd — The utility to check the availability of the port number mentioned in Properties.ini for agent.• LogonTest.cmd — Utility to check the database connectivity.
agent_home/conf/	Contains the following files: <ul style="list-style-type: none">• 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/connector/schema/spml directory structure where the XML mapping file is stored.
agent_home/lib/	Contains JAR files used by the agent.
agent_home/logs	Contains log files produced by the agent.
agent_home/Uninstall_Admin-MSSQL2000Server-Connector/	Contains files for uninstalling the agent.

Starting the Agent

To start the agent, run `SQLConnectorConsole.cmd`, 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.

You can use the following example:

```
<agent_home>/SQLConnectorConsole.cmd
```

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

Modifying the Database Account and Select Identity Passwords

After the agent is installed, if you change the database account password or the Select Identity administrative password, you must update the agent with the change.

Perform the following steps on Microsoft SQL Server machine to update password change to the agent.

- To update the change in database password, run the following command on the Windows command prompt:

```
<install_dir>\passwordEncrypt.cmd -r <db-password>
```

where *<install_dir>* is the location of the agent and *<db-password>* is the new database password.

- To update the change in Select Identity administrative password, run the following command on the Windows command prompt:

```
<install_dir>\passwordEncrypt.cmd -s <ovsi-password>
```

where *<install_dir>* is the location of the agent and *<ovsi-password>* is the new Select Identity password.

6 Uninstalling the Connector

If you want to uninstall SQL Server Admin connector from Select Identity, perform the following steps:

- Remove all resource dependencies.
- Delete the connector from Select Identity.
- Delete the connector from application server.
- Uninstall the agent.

See *HP OpenView Select Identity Connector Deployment Guide* for more information on deleting the connector from application server and Select Identity.

Uninstalling the Agent

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

- 1 Select **Programs** → **Admin-MSSQL2000Server-Connector** → **Uninstall Agent** from the Start menu. The wizard appears.
- 2 Click **Next** on the introductory screen.
- 3 Provide the database credentials to uninstall the reverse tables, if they were installed.
- 4 Click **Uninstall**.
- 5 Click **Continue** when the pop-up dialog indicates that the reverse notification tables were successfully uninstalled.
- 6 Click **Done** on the Uninstall Complete screen to close the wizard.

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 an 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.
Solutions: Add the Create operation to the Entity/Attribute by using the attribute mapping utility. Refer to *Appendix C* of the *HP OpenView Select Identity Administrator Guide* for details on how to add create operations for an entity.

Agent and Reverse Notification Table Installation

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

- 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.

- The following error message is displayed:

```
Can't create view dbo.DBA_USERS Message received from the database: There is already an object named ... Cannot proceed.
```

Possible Cause: You are attempting to reinstall the agent without removing previously installed database tables.

Solution: Uninstall the agent as documented in [Uninstalling the Connector](#) on page 39. This removes previously installed tables. Then, run the agent installation wizard again.

- The following error message is displayed:

```
Exception occurred while starting reverse. Error message receive: Io exception: Connection refused(DESCRIPTION=(TMP=) (VSNNUM=135295488) (ERR=12505) (ERROR_S TACK=(ERROR=(CODE=12505) (EMFI=4))))Error in logon. Can not proceed.
```

Possible Cause: The wrong database service name was entered.

Solution: Verify the database service name in the `properties.ini` for correctness and ensure that the case of the name is correct (the name is case-sensitive).

- The agent installation wizard fails to start and displays an error message.

Possible Cause: The JVM is not in the System Path environment variable or Java 1.4 is not available.

Solution: Add the Java 1.4 to the System Path.

- While deploying the reverse synchronization tables, the installation stops and displays an exception.

Possible Cause: A version of Java that is older than 1.4 is the default JDK in use.

Solution: Set the `JAVA_HOME` variable to the path of Java version 1.4.

- An error message appears displaying `JZ006 Exception`. Cannot connect to database.

Possible Cause: The hostname may not be resolved to proper IP address of the database server.

Solution: Provide the proper IP address.

- During agent installation, an error message appears displaying `Invalid Login credentials` even if correct values are provided for database Username and Password.

Possible Causes:

- `JAVA_HOME` environment variable is not set or not set properly.

- JDBC driver jars are not placed in system CLASSPATH.
- commons-logging.jar is present in the <JAVA_HOME>\jre\lib\ext folder.

Solutions:

- Set the JAVA_HOME upto the path from where the bin folder with java.exe is accessible.
- Update the system CLASSPATH with the paths of SQL Server JDBC jars.
- Place the log4j-1.2.8.jar along with commons-logging.jar in the same path.
- While registering the agent as a service, the Windows account name given is not accepted.

Possible Causes:

- The complete Windows account name (<DOMAIN_NAME>\<USER_NAME>) is not given.
- The local account is given in the form localhost\administrator.

Solutions:

- Enter the user name along with the domain name as the installer needs the complete windows username (with Domain Name) for registering agent as a service.
- The account name with localhost is not supported. Instead, you can prefix the machine name for local accounts. For example: sqlmachine1\Administrator.
- An error message appears displaying CREATE VIEW permission denied in database while deploying the agent.

Possible Causes: The database user account used does not have all the necessary privileges to the database.

Solutions: Select a user with proper privileges to install the agent.

- While deploying the agent, an error message appears displaying
Class Not Found Exception caught: driverName =
"com.microsoft.jdbc.sqlserver.SQLServerDriver Can not establish connection to the DB.

Possible Cause: The JDBC driver files are not in system CLASSPATH of Microsoft SQL Server machine.

Solution: Place the mssqlserver.jar, msbase.jar, and msutil.jar files in the system CLASSPATH.

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.

- 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 the class path.

- The following error is displayed:

```
SQLException occurred while adding element into SNAPSHOT_TAB. Message  
received from the database: table or view does not exist
```

Possible Cause: The agent is installed without the reverse notification tables.

Solution: Install the tables by running the installation, then run the agent.

- Error message appears displaying The system cannot find the path specified while running/starting the agent.

Possible Cause: The agent is not able to find JAVA in system PATH.

Solutions: Make sure that JAVA_HOME variable is set on database server machine and JAVA is available in system PATH.

B Connector Behavior

For forward provisioning, the following behaviors are observed:

- There is no STATUS attribute for a user on SQL Server, where the information of whether the user is enabled or disabled can be stored. Hence, the Enable All Services and Disable All Services request returns success.
- The user is always linked to the public role and this membership cannot be changed. Hence, this role is not shown in Select Identity and operations cannot be performed on it.
- The Dbname attribute of a user in the mapping file is the default database of the user. If no value is provided, master is assigned as the default database.

Using Entitlements to Assign User Privileges on Databases

This section describes how the connector uses the entitlements assignment/ un-assignment feature of Select Identity in order to provision a user across databases within a single Microsoft SQL Server 2000 instance. The following roles and privileges can be assigned to users using this connector:

Entitlement Categories

- 1 Database – If only public access to a database is required, the search criteria can be set to "Begins With - DATABASE^<Database Name>". For example, "Begins With - DATABASE^" returns all the databases available on the SQL Server. In order to provide a user with a specific database privilege, or public access, select that privilege from the entitlement list and add it to the user.
- 2 Server Roles – These server roles begin with the string SERVERROLE. They can be uniquely retrieved by setting the search criteria on SI to "Begins With - SERVERROLE^". In order to assign a user a specific server role, select that role from the entitlement list and add it.
- 3 Database Roles – The database roles associated with every database are displayed in the entitlement list as strings in the format "ROLE^<Database Name>^<Role Name>". In order to retrieve all the roles of a particular database, the search criteria on SI has to be set to "Begins With - ROLE^<Database Name>". For example, "Begins With - ROLE^model" retrieves the database roles associated with the model database, as illustrated in the figure below. In order to assign a user a specific database role, select that role from the entitlement list and add it.
- 4 Database Privileges – The database privileges associated with a database on the server are displayed in the entitlement list as strings in the format "ENTITLEMENT^<Database Name>^<Database Privilege>". Public access on a database is represented by the string "DATABASE^<Database Name>". In order to

retrieve all the database privileges, the search criteria on SI has to be set to "Begins With - ENTITLEMENT^<Database Name>". For example, "Begins With - ENTITLEMENT^model" retrieves all the privileges on the database model.



In the absence of any search criteria, the entitlement search from SI will return all the above categories for all the databases in the SQL Server. This would typically be a large list and would span multiple pages. If the entitlements to be assigned are known before hand, it is recommended that appropriate search strings are used in order to filter the entitlement list.

Recommended Usage of Entitlements while User Provisioning

Note the following points regarding entitlements, their behavior, and relationship between the different categories for the Admin SQL Connector.

- The SERVERROLE type entitlement is not dependent on other two type of entitlements. This entitlement refers to the Server Roles available for a SQL Server Login User.
- DATABASE and ROLE/ENTITLEMENT type entitlements are mutually dependant. Hence it is recommended that user takes precaution while working with the DATABASE and ROLE/ENTITLEMENT entitlement types.
- As a user should be assigned to a particular database before being assigned the ROLES/ENTITLEMENT of that particular Database, its recommended that user be assigned the DATABASE type entitlement of a database before the ROLE/ENTITLEMENT type entitlements of that Database are assigned to the user. Connector however implicitly assigns the DATABASE type entitlement in case an attempt of assigning a ROLE/ENTITLEMENT type entitlement of a database is case DATABASE type entitlement of particular database is not assigned to the user. This implicit assignment of DATABASE entitlement is captured as a reverse sync request and an SPML is sent. But in case REVERSE SYNC is not enabled, this change will not be propagated to Select Identity leading to inconsistency.
- When a User is unlinked from a DATABASE entitlement, the user is deleted from that particular database on SQL Server. So, all ROLES/ENTITLEMENT assigned to the user on that Database will also be automatically deleted. This ROLES/ENTITLEMENT deletion for a user are captured as reverse sync requests and SPMLs are sent to Select Identity to reflect this change. In case, Reverse Sync is not enabled the implicit ROLES/ENTITLEMENT deletion occurring on SQL Server are not captured leading to inconsistency. Hence it is recommended that ROLE/ENTITLEMENT type entitlements are unlinked before DATABASE type entitlement.

Example Use Case

A Database user by name **dbuser1** has to be created and assigned to two databases temp1 and temp2. **dbuser1** should be assigned db_owner ROLE on both the databases (temp1 and temp2) and assigned db_securityadmin on temp2.

To do this on Select Identity, following events will occur in the given sequence.

- 1 Resource and Service will be created for this SQL Server.
- 2 During Add User action, the retrieval of entitlements will give a list of databases and roles of each database. So SI ADMIN needs to filter these out.

- 3 Type the keyword database in Search String of the Get Entitlements page on Select Identity. List of databases appears, which the user can be assigned to. You will see DATABASE^temp1 and DATABASE^temp2 in the list. These should be selected.
- 4 ROLES of databases cannot be linked to the user till you have assigned him to the database. So the user should be linked to Database-temp1 and Database-temp2 before the entitlements related to the ROLES of each database are assigned to the user.
- 5 Search for temp2 as the Search String. This will retrieve all the ROLES of temp2 database. Select db_owner and -db-securityadmin which are displayed as ROLE^temp2^db_owner and ROLE^temp2^db_securityadmin. These can be linked to the user.
- 6 Search for temp1 as the Search String. This will retrieve all the ROLES of temp1 database. Select db_owner is displayed as ROLE^temp1^db_owner. This can be linked to the user.

