HP OpenView Select Identity

Software Version: 4.0

Connector Deployment Guide

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 valuable 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 Introduction

This chapter gives an overview of the HP OpenView Select Identity connector. An HP OpenView Select Identity connector allows you to provision users and manage identities on an enterprise information system. 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

About HP OpenView Select Identity

HP OpenView Select Identity (OVSI) provides a new approach to identity management. It helps you manage the entire identity lifecycle of an enterprise application. By using OVSI, you can automate the process of provisioning and managing user accounts and access privileges across platforms, applications, and corporate boundaries. OVSI 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 OVSI by using a connector. A connector is resource specific. It is installed on the system where OVSI is installed. The combination of OVSI 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 OVSI, but if any change takes place in resource, it cannot communicate that back to OVSI. On the other hand, a bidirectional connector can reflect the changes made on resource back to OVSI. This property of bidirectional connectors is known as **reverse synchronization**.

Features and Capabilities

A connector enables OVSI to access a resource to manage users, groups, and entitlements. OVSI can typically perform the following tasks by using a connector.

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

- Reset user passwords

The set of tasks, which can be performed by the connector on resource, varies from connector to connector.

A connector usually consists of:

- A **Resource Adapter Archive** (**RAR**) file— this file contains connector binaries.

- A **Schema** file — this contains the mapping file for the connector. A mapping file contains resource attribute information of the connector, which must be linked to OVSI attributes.

A connector may not contain a schema file. In that case, the mapping file for the connector can be generated by using attribute mapping utility of OVSI. Refer to *Appendix C: Attribute Mapping* chapter of *HP OpenView Select Identity Administrator Guide* for information on attribute mapping utility.

In addition to above two files, there could be other files packaged with the connector, such as an agent file, a script file, and so on.

To be able to use the connector, you must deploy the connector on an application server, and then on OVSI.

About Deploying a Connector

In order to use a connector with OVSI, you must deploy it on an application server, and then configure it with OVSI. The RAR file of the connector, which contains the binaries, must be deployed on an application server. You must perform the following tasks to deploy and configure a connector.

1 Extracting the Contents of Schema File

2 Deploying the Connector on Application Server

3 Configuring the Connector with OVSI

2 Extracting the Contents of Schema File

Most of the connectors contain at least one schema file. This is the file which contains the mapping information of the connector. Some of the connectors do not have a schema file packaged with it. In case of these connectors, mapping file is generated by using attribute mapping utility of OVSI. If the connector, which you are deploying, does not contain a schema file packaged with it, and you want to generate the mapping file by using attribute mapping utility, skip to the next chapter.

If the connector is packaged with a schema file that has mapping information of the connector, perform the following steps.

1   Create a schema subdirectory in the Select Identity home directory where you can store the connector's mapping files.
    For example, you can create `<OVSI_HOME_DIR>`/Schema
    where `<OVSI_HOME_DIR>` = `/opt/Select_Identity` in UNIX
    and `<OVSI_HOME_DIR>` = `C:\Select_Identity` in Windows (A schema subdirectory may already exist.)

2   Extract the contents of the schema file to the `Schema` subdirectory. Some connectors may contain more than one schema file. Refer to the connector's Installation and Configuration Guide to find out the name of the schema file, and select the right schema file.

3   Ensure that the `CLASSPATH` environment variable in the application server startup script references the `Schema` subdirectory.

3 Deploying the Connector on Application Server

To install the connector on OVSI, you must deploy the connector on the application server. To deploy the connector on an application server, perform the following steps:

1   Create a subdirectory in Select Identity home directory where you can store the connector's Resource Adapter Archive (RAR) file.
    For example, you can create `<OVSI_HOME_DIR>`/connectors
    where `<OVSI_HOME_DIR>` = /opt/Select_Identity in UNIX
    and `<OVSI_HOME_DIR>` = C:\Select_Identity in Windows (A connector subdirectory may already exist.)

2   Copy the RAR file from the Select Identity Connector CD to the connector subdirectory.

3   Perform the following steps to deploy the connector on WebLogic. If deploying on WebSphere, skip to step 4 on page 11.

    a   Start the application server in the domain for Select Identity, if it is not currently running, and log on to the WebLogic Server Console.

    b   In the left pane, expand Deployments folder, and then right click on Connector Modules, and select **Deploy a New Connector Module**.

        Alternatively, at the right panel of the Server Console homepage, click on **Connector Modules** link, which is under Your Deployed Resources column of Domain Configurations section. Resource Connectors page appears. Click on **Deploy a New Connector Module** link on this page.

    c   Click the link in the Location field, locate, and select the RAR file from the list. It is stored in the connector subdirectory.

    d   Click **Target Module**.

    e   If only one server is configured, skip to next step. If more than one server is configured, the next page prompts you to select the servers on which you want to deploy the connector. Select the server instance, and then click **Continue**.

    f   Review the settings. Keep all the default settings and click **Deploy**. The Status of Last Action column should display Success.

4   If you want to deploy the connector on WebSphere, perform 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 connector's RAR file. It is stored in the subdirectory created in the beginning.

    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 Configurations dialog, click **Save**.

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. This is the pool name of the connector. Refer to respective connector's Installation and Configuration Guide to find out the specific pool name.

p   Click **OK**.

q   Click the **Save** link.

r   On the Save to Master Configurations dialog, click **Save**.

s   Restart WebSphere.

4 Configuring the Connector with OVSI

After you deploy the connector on the application server, you must configure the connector with OVSI to be able to use it.

Perform the following steps to deploy and configure the connector on OVSI:

1   Add a New Connector

2   Add a New Resource

3   Map Resource Attributes

Add a New Connector

To add the connector with OVSI, perform the following steps:

1   On HP OpenView Select Identity home page, click **Service Studio** → **Resources**. The Resources List page appears.

2   Click **Manage Connectors** on Resources List page. Manage Connectors page appears.

3   In Manage Connectors page, do the following:

— In the Connector Name text box, specify a name for the connector.

— In the Pool Name text box, enter the pool name of the connector. Exact pool name of the connector is given in the connector Installation and Configuration Guide.

— Under Mapper Available section, select **Yes** if the connector is supported by attribute mapper utility of OVSI. Otherwise, select **No**.

OVSI displays the Manage Connectors page in the following format:

Refer to *Service Studio* chapter of *HP OpenView Select Identity Administrator Guide* for more information on managing connectors.

Add a New Resource

To deploy a resource that uses the newly added connector, perform the following steps:

1   Click **Service Studio** → **Resources.** Resources List page appears.

2   Click **Add New Resource**.
    Add New Resource: Basic Information page appears.

3   Fill in the parameters given in *Table 3: Resource Configuration Parameters* in the connector's Installation and Configuration Guide.

4   Click **Next.** Add New Resource: Resource Access Information page appears.

5   Fill in the parameters given in *Table 3: Resource Configuration Parameters* in the connector's Installation and Configuration Guide.
    Some of the connectors, like database connectors, are supported by attribute mapping

utility. These connectors may not have a mapping file packaged with it. In case of such connectors, click **Edit** link next to Mapping File text box. Attribute Mapper page appears. The mapping xml file can be generated from attribute mapping page.

Refer to *Appendix C: Attribute Mapping* chapter of *HP OpenView Select Identity Administrator Guide* for more information on creating mapping file by using attribute mapping utility.

6   Click **Finish.** User Reconciliation Policy page appears.

7   Only in case of the connectors where reverse synchronization is achieved by polling, such as bidirectional LDAP based connectors, select the **Polling Enabled check** box, set the polling interval, and then click **Apply.** You must set the polling interval as mentioned in the connector's Installation and Configuration Guide. If the connector does not employ polling for reverse synchronization, skip to the next step. Refer to the connector's Installation and Configuration Guide to find out if the connector needs polling to be enabled.

8 Some of the connectors, such as bidirectional LDAP connectors, need the User Enable/ Disable Workflow External Call to be configured. In case of those connectors, modify the following fields on User Reconciliation Policy page.

  a Under the Add section, select SI Recon User Enable Disable Workflow from Reconciliation Workflow drop-down box.

  b   Under the Modify section, select SI Recon User Enable Disable Workflow from Reconciliation Workflow drop-down box.

Refer to the respective connector Installation and Configuration Guide to find out if the connector needs these fields to be modified.

Map Resource Attributes

Map the resource attributes to the OVSI attributes. To do this, perform the following steps.

1 Click **Service Studio** → **Resources**. The Resources List page appears.

2 Select the newly created resource from the Resource List, and then click **Modify**. Basic Information page appears.

3 Click **Resource Attribute Mapping** link on left pane. Resource Attribute Mapping page appears.

4 Map each resource attribute to Select Identity Attribute by using drop-down list.

While mapping the resource attributes on OVSI, refer to *Table4* in the connector's Installation and Configuration Guide for connector specific mapping information.

➤ If the connector supports reverse synchronization for that particular attribute, you must check **Sync In** checkbox to reflect the changes on that attribute made on the resource end to OVSI.
Attributes updated at OVSI are set to **Sync Out** to reflect the change at resource end.

In order to use the connector, you must associate the newly added resource to a service. Refer to *Service Studio* chapter of *HP OpenView Select Identity Administrator Guide* for information on service.

Configuring User Enable/Disable Workflow External Call

Some of the connectors, such as PeopleSoft connector or bidirectional LDAP based connectors, require user enable/ disable workflow external call to be modified. refer to the respective connector's Installation and Configuration Guide to find out if this is required.

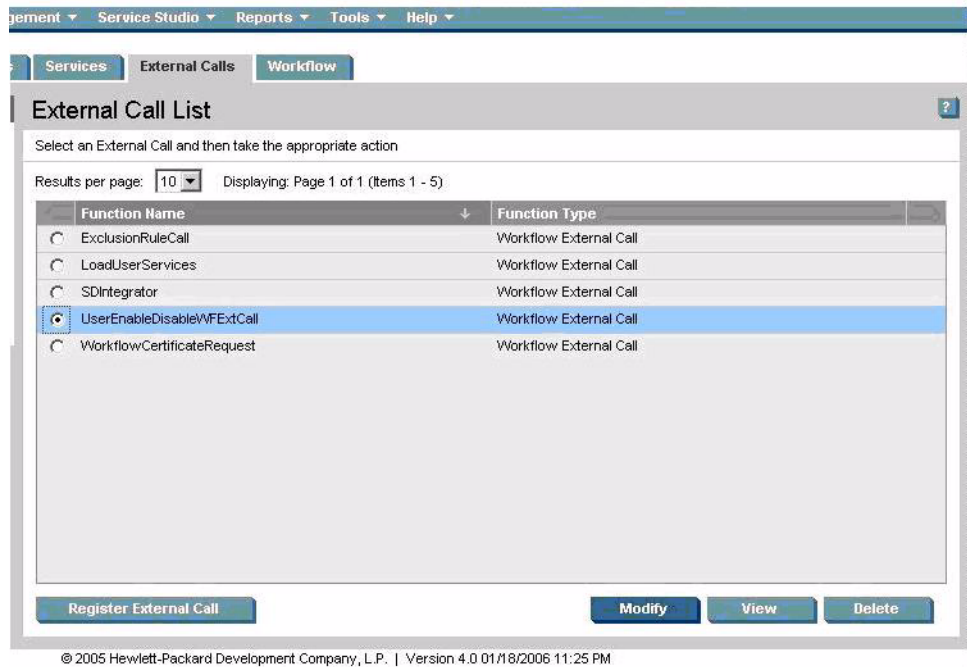Perform the following steps to configure user enable/disable workflow external call.

1 On OVSI home page, click **Service Studio** → **Resources**. External Call List page appears.

2   On left pane, select **WorkflowExternalCall** from Type drop-down box, and then click **Search**.
    On right pane, a list of workflow external calls appear.

3   On right pane, select **UserEnableDisableWFExtCall** radio button, and then click **Modify.**
    UserEnableDisableWFExtCall: Basic Information page appears.

4   Enter the description, and then click **Apply.**

5   Click **Parameters** link on left pane. UserEnableDisableWFExtCall: Set Parameters page appears.

6   You must enter the following parameters and add, one after another.

    a   `AttributeName`

    b   `EnableValue`

    c   `DisableValue`

    d   `UserName`

    e   `Password`

    f   `Url`

7   Click **OK**.

Refer to *Table 5: UserEnableDisableWFExtCall Parameters* in the connector's Installation and Configuration Guide for exact value of the parameters.

Configuring Connector on Non-English Platforms

If you install the connector, which is internationalized, on non-English platform, you will have the following limitations while configuring the connector:

- When entering user attributes to provision (in OVSI), you cannot enter local language characters for the following attributes

    — UserName

    — Password

    — Email

- The attribute names on the resource cannot contain non-English characters. Thus, you cannot include non-English characters in the mapping file.

- Non-English entitlements are not supported by the connector.

- All configuration and property file names must be in English.

- The exception messages from the resource are in English.

- The log messages are in English.

- The OVSI resource name, which is included in the reverse synchronization configuration of the agent, must be in English.

Reverse synchronization of local language characters is supported if the connector is internationalized. While provisioning users on the LDAP resource, you can enter local language characters as input data. These characters are reconciled with OVSI through SPML communication. However, the following user attributes must contain English characters:

- UserName
- Password
- Email

5 Uninstalling the Connector

To uninstall a connector from OVSI, perform the following:

1 Delete the connector from OVSI home page.

2 Delete the connector from application server.

Deleting the Connector from OVSI

Perform the following steps to delete a connector from OVSI.

1 Click **Service Studio** → **Resources** on OVSI homepage. Resource List page appears.

2 Click **Manage Connectors.** Manage Connectors page appears.

3 Select the connector, which you want to delete.

4 Click **Delete**.

Deleting the Connector from WebLogic

Perform the following to delete a connector from WebLogic:

1 Log on to the WebLogic Server Console.

2 Expand the Deployments folder on the left pane, and then double click on **Connector Modules**

   Alternatively, at the right panel of the Server Console homepage, click on **Connector Modules** link, which is under Your Deployed Resources column of Domain Configurations section.

3 The right hand pane of the console displays a table showing all the deployed connectors. Click the delete icon next to the connector that you want to uninstall.

4 Click **Yes** to confirm the deletion.

5 Click **Continue**.

Deleting the Connector from WebSphere

Perform the following steps to uninstall the connector from 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 Configuration dialog, click the **Save** button.

A Mapping Files

User profile in a resource has a number of attributes, for example, username, first name, last name, and so on. You must map these resource attributes to the OVSI attributes. Every connector is associated with a mapping file, which contains resource-specific attributes. While mapping resource attributes, OVSI fetches the attributes from the connector's mapping file, and displays under Resource Attribute column. The Attribute column displays the drop-down boxes, which list all the OVSI attributes.

A connector may have more than one mapping file. Mapping files are usually XML files or properties files.

In addition to the mapping file, if you configure the connector for reverse synchronization, you must have an XSL transformation file. The mapping file(s) and the transformation file for the connector are usually bundled with Schema file. Some of the connectors do not have any schema file bundled. In that case, you must generate the XML mapping file and XSL transformation by using attribute mapping utility of OVSI.

If reverse synchronization is configured, OVSI server receives SPML requests that contain the attribute changes. The names of the attributes in the SPML request are defined by the resource. To transform the attribute names to Select Identity attribute names, the request is parsed by Select Identity by using the XSL file.

XML Mapping File

An XML mapping file typically contains the following elements.

- **<Schema>** element □— It is the first element in the mapping file. Entire content of the mapping file is contained in this tag. The schema element in the mapping file for TAM connector is shown below.

```
<Schema
  xmlns="urn:oasis:names:tc:SPML:1:0"
  xmlns:spml="urn:oasis:names:tc:SPML:1:0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:dsml="urn:oasis:names:tc:DSML:2:0:core"
xmlns:saml="urn:oasis:names:tc:SAML:1.0:assertion"
  xmlns:concero="http://www.trulogica.com/concero/v21"
xsi:schemaLocation="urn:oasis:names:tc:SPML:1:0 file://C:/SPML/
cs-pstc-spml-schema-1.0.xml"
  majorVersion="1.0" minorVersion="1.0" >
```

-)**<providerID>** and **<schemaID>** elements — These elements provide standard elements for header information. These elements in the mapping file for TAM connector is shown below.

```
<providerID providerIDType="urn:oasis:names:tc:SPML:1:0#URN" >trulogica's
URN from oasis</providerID>

 <schemaID
schemaIDType="urn:oasis:names:tc:SPML:1:0#GenericString">TivoliAccessMana
ger</schemaID>
```

- **<objectClassDefinition>** element — This element defines the actions that can be performed on the specified object   and the Select Identity-to-resource field mappings for the object. This element consists of the elements:

  — **<properties>** element — It defines the operations that are supported on the object. This can be used to control the operations that are performed through Select Identity. The operation is assigned as the name of the <attr> element and access to the operation is assigned to a corresponding <value> element.

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

The <objectClassDefinition> element in the mapping file for TAM connector is shown below.

```
<objectClassDefinition name="User" description="TAM User">
    <properties>
      <attr name="CREATE">
        <value>true</value>
      </attr>
      <attr name="READ">
        <value>true</value>
      </attr>
      <attr name="UPDATE">
        <value>true</value>
```

```
        </attr>
          <attr name="DELETE">
            <value>true</value>
          </attr>
          <attr name="RESET_PASSWORD">
            <value>true</value>
          </attr>
          <attr name="EXPIRE_PASSWORD">
            <value>false</value>
          </attr>
</properties>
    <memberAttributes>
        <!-- This is the Key for the user in TAM and the Directory store -->
        <attributeDefinitionReference name="UserDn" required="true"
          concero:tafield="[First Name] [Last Name]-[GUID]"
concero:resfield="cn"
concero:isDn="true"/>
        <!-- This is Concero UserId, Also user can login into TAM using this
-->    <attributeDefinitionReference name="User Name" required="true"
concero:tafield="User Name" concero:resfield="uid" concero:isKey="true"/>
<attributeDefinitionReference name="Password" required="true"
concero:tafield="Password" concero:resfield="password"/>
<attributeDefinitionReference name="First Name" required="true"
concero:tafield="First Name" concero:resfield="fname" />
<attributeDefinitionReference name="Last Name" required="true"
concero:tafield="Last Name" concero:resfield="lname"/>
    <attributeDefinitionReference name="Description" required="false"
concero:tafield="Description" concero:resfield="description"/>
    </memberAttributes>
</objectClassDefinition>
```

concero:tafield **attribute of** <attributeDefinitionReference> **element specifies the**
name of the Select Identity attribute to which the resource attribute needs to be mapped.

concero:resfield **attribute of** <attributeDefinitionReference> **element specifies**
the name of the attribute from the resource schema, which has to be mapped to a Select
Identity attribute. If the resource does not support an explicit schema (such as UNIX),
this can be a tag field that indicates a resource attribute mapping.

- **<attributeDefinition> element** — It 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. An example of this element in the
  mapping file of TAM connector is shown below.

```
<attributeDefinition name="User Name" description="userId"
type="xsd:string" >

  <properties>
      <attr name="minLength">
        <value>1</value>
      </attr>
      <attr name="maxLength">
        <value>100</value>
      </attr>
      <attr name="pattern">
        <value><![CDATA[[a-zA-Z0-9@]+]]> </value>
```

```
            </attr>
          </properties>
        </attributeDefinition>
```

Properties Mapping File

Some of the connectors, such as Domino connector or agent based Active Directory connector, use properties file to hold mapping information, instead of an XML file. The mapping information in properties file is given in the following format.

SI Attribute | Resource Attribute

For example, in `aduser.properties` file of agent based Active Directory connector, the mapping information is given as

`Email | mail` where `Email` is an SI attribute and `mail` is a resource attribute.

Attributes can be concatenated. The attribute names and the separators must

not contain the | delimiter. For concatenation, the format is as follows:

[*SI Attribute*]<separator>[*SI Attribute*]|*Resource Attribute*

For example, in `aduser.properties` file,

`[City] [addr1]|street`

XSL Transformation File

If the value of a resource attribute, which has been mapped to SI, is changed at the resource end, the change can be reflected to SI by reverse synchronization.

The Select Identity server receives SPML requests that contain the attribute changes, which are parsed by SI by using the XSL file. In Attribute name mappings section of the XSL file, mapping relationship between the resource attribute and Select Identity attributes are defined.

Resource side attribute is represented as:
`<xsl:variable name="RES_ATTR0" select="'`*xxxxxxxxxx*`'"/>`

The mapped attribute for `ATTR0` in SI is represented as:

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

For example, in `domino.xsl`, the XSL file for Domino connector, `ATTR0` is defined as

```
<xsl:variable name="RES_ATTR0" select="'fullname'"/>
<xsl:variable name="SI_ATTR0" select="'SIUSERKEY'"/>
```