HP OpenView Select Identity

Connector for Oracle® PeopleSoft Human Capital Management 8.8 (Peopletools 8.45)

Installation and Configuration Guide

Connector Version: 1.0
Select Identity Version: 3.3.1

August 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/). 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.

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://passport2.hp.com/hpp/newuser.do

contents

1

Installing the Connector

The Oracle PeopleSoft Human Capital Management connector — hereafter referred to as the PeopleSoft connector — enables HP OpenView Select Identity to provision users in the PeopleSoft Human Resources module of a PeopleSoft Human Capital Management (HCM) 8.8 system. The PeopleSoft connector is a two-way connector and provides an agent that can send changes made to data in PeopleSoft to Select Identity.

Specifically, the connector's agent can capture the creation of an employee in Peoplesoft Human Resources management system (HRMS) including personal and job information. The agent also captures employee modifications performs in the HRMS application. All captured data is sent through the connector to the Select Identity server.

When the employee status changes from Active to any other state, the user is disabled on the Select Identity server and the user's status is shown as S (for suspended). If the employee regains Active status, the employee is enabled in the PeopleSoft application and status is changed to A (for active).

Also, the PeopleSoft connector can maintain PeopleSoft HCM Application users. This functionality is implemented using the Component Interface functionality of PeopleSoft. In addition, the connector can capture when PeopleSoft HCM Application users are created, modified, or deleted using the the Integration Broker technology.

All connector functionality was implemented using the integration Broker capabilities of the PeopleSoft application. As much as possible, all default messages and message channels provided by the PeopleSoft application are used. The changes are captured and sent to the Select Identity Web Service, which handles requests and generates SPML requests that are forwarded to the Select Identity server. All changes are captured in the real time and sent immediately. When the Web Service receives a request from the connector, it parses the request using the XSL file provided with the connector. You can modify the XML and XSL file to add attributes. However, the defined attributes should not be removed from the files.

The Enable and Disable Employee functions are provided by using an external call, which generates disables or enables all Service request for the user in all Services. This external call is triggered by a modify user request, which uses a new workflow template that is invokedwhen the EMPL_STATUS attribute changes from A to S or vice versa.

The Username attribute is required for all users in the Select Identity. However, there is no Username attribute associated with employee information in PeopleSoft. This is resolved as follows:

- The user name is the same as the employee ID

- The user name is generated before passing the request to Select Identity

The default XSL file provided with the PeopleSoft connector is configured to generate the user name using the <FIRST_NAME><LAST_NAME>. You can edit this file to generate the user name as desired.

Password is also a required field for the user in the Select Identity. The default password can be generated or hardcoded in the XSL file so that it is passed to the Select Identity; it can be changed later.

The PeopleSoft connector is packaged in the following files, which are located in the PeopleSoft folder on the Select Identity Connector CD:

- PeopleSoftConnector.rar — contains the binaries for the connector.

- PSEmpSchema.jar — contains the following mapping files, which control how Select Identity and PeopleSoft fields are mapped to each other:

    — PeopeSoftEmp.xml — map the Select Identity fields to the PeopleSoft fields, which enables Select Identity to provision data in PeopleSoft HCM systems.

     — `PeopleSoftEmp.xsl` — maps attributes on the PeopleSoft HCM server to attributes on the Select Identity server. This file is used by the agent during reverse synchronization.

- `PSUsrSchema.jar` — contains the following mapping files:

     — `PeopleSoftUsr.xml` — maps the Select Identity user attributes to those on the Peoplesoft HCM Application server.

     — `PeopleSoftUsr.xsl` — maps attributes on the PeopleSoft HCM Application server to attributes on the Select Identity server. This file is used by the agent during reverse synchronization.

- `PSEmpSetup.jar` and `PSUsrSetup.jar` - contains the agent configuration files for employees and Application users, respectively.

- `PSEmpOpAttr.properties` and `PSUsrOpAttr.properties` — contains the operational attributes required in the SPML request for reverse synchronization for HRMS employee and user, respectively.

- `PSProperties.ini` — contains the configuration attributes

- `peoplesoftagent.war` — contains the files that comprise the PeopleSoft agent.

System Requirements

The PeopleSoft connector is supported in the following environment:

Select Identity Version	Application Server	Database
3.3.1	WebLogic 8.1.4 on Windows 2003	SQL Server 2000
	WebLogic 8.1.4 on Red Hat Enterprise Linux 3.0	Oracle 9i
	WebLogic 8.1.4 on Solaris 9	Oracle 9i

This connector supports PeopleSoft Human Capital Management, Human Resouces module, version 8.8 (People Tools version 8.4.5).

Deploying on the Web Application Server

To install the PeopleSoft 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 `PeopleSoftConnector.rar` file from the Select Identity Connector CD to the connector subdirectory.

3   Create a schema subdirectory in the Select Identity home directory where the connector's mapping file(s) will reside. For example, you could create the `C:\Select_Identity\schema` folder. (This subdirectory may already exist.)

4   Extract the contents of the `PSEmpSchema.jar` file on the Select Identity Connector CD to the schema subdirectory. (If you copied the JAR file to the local system, be sure to remove it after extracting its contents.)

5   Ensure that the CLASSPATH environment variable in the WebLogic server startup script references the schema subdirectory.

6   Start the application server if it is not currently running.

7   Log on to the WebLogic Server Console.

8   Navigate to *My_domain* →**Deployments** →**Connector Modules**.

9   Click **Deploy a New Connector Module**.

10   Locate and select the `PeopleSoftConnector.rar` file from the list. It is stored in the connector subdirectory.

11   Click **Target Module**.

12   Select the **My Server** (your server instance) check box.

13   Click **Continue**. Review your settings.

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

15   Navigate to *My_domain* →**Deployments** →**Web Application Modules**.

16   Click **Deploy a New Web Application Module**.

17 Locate and select the `PeopleSoftAgent.war` file from the list. It is stored in the connector subdirectory.

18 Click `Target Module`.

19 Select the **My Server** (your server instance) check box.

20 Click **Continue**. Review your settings.

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

22 Modify the mapping files, if necessary. These files are described in detail in Understanding the Mapping Files on page 33.

23 *For PeopleSoft Employees*
Extract the files from the `PSEmpSchema.jar` and `PSEmpSetup.jar` files to the Select Identity home directory.

Because the attributes in the `PeopleSoftEmp.xsl` file are based on those in the `PeopleSoftEmp.xml` file, you must modify the `PeopleSoftEmp.xsl` file to reflect changes made to these files.

24 *For PeopleSoft Application User*
Extract the files from the `PSUsrSchema.jar` and `PSUsrSetup.jar` files to the Select Identity home directory.

Because the attributes in the `PeopleSoftUser.xsl` file are based on those in the `PeopleSoftUser.xml` file, you must modify the `PeopleSoftUser.xsl` file to reflect changes made to these files.

25 *For PeopleSoft Employees*
Modify the `PSEmpOpAttr.properties` file to include the corresponding mapping file name and the resource ID:

`urn:trulogica:concero:2.0#resourceType=PeopleSoftEmp`

`urn:trulogica:concero:2.0#resourceId=<PeopleSoft Empl res ID>`

Then, modify `PSProperties.ini` to include the Select Identity URL. The SPML request needs to be changed, as in this example:

`CONCERO_SERVER_URL=http://localhost:7001/lmz/webservice`

26 *For PeopleSoft Application User*
Modify the `PSUsrOpAttr.properties` file to include the corresponding mapping file name and the resource ID:

`urn:trulogica:concero:2.0#resourceType=PeopleSoftUsr`

`urn:trulogica:concero:2.0#resourceId=<PeopleSoft User res ID>`

Then, modify `PSProperties.ini` to include the Select Identity URL. The SPML request needs to be changed, as in this example:

```
CONCERO_SERVER_URL=http://localhost:7001/lmz/webservice
```

Modify this line:

```
PEOPLESOFT_SI_USR_ADMIN=<PeopleSoft admin user for SI's use>
```

where as the Peoplesoft admin user is the user that Select Identity will use to provision users in the PeopleSoft HCM Application module. This attribute is introduced to prevent cyclic blocks.

> It is assumed that the user created for Select Identity will be used to log in to the PeopleSoft application using the PeopleSoft connector. If changes are made directly in the Peoplesoft application using the same login information, the changes will not be synchronized back to Select Identity and will be ignored (for cyclic requests).

After installing the connector, refer to Configuring the Connector on page 22 for information about registering and configuring this connector in Select Identity.

Configuring the PeopleSoft Integration Broker

This sections describes how to configure the PeopleSoft Integration Broker to enable it to capture employee and Application user changes that are made on the PeopleSoft resource.

Prerequisites

Ensure that the following prerequisites are met before proceeding:

- Obtain a user who has permission on the Integration Broker (People Tools) and access to the Portal

- Ensure that the Default Gateway (Local) is configured with all default connectors

- Make sure the Integration Broker is configured and all related processes are running

- *For PeopleSoft HRMS employees only*
  Verify that the PERSON_DATA channel status is Active and Running

- *For a PeopleSoft Application user only*
  Verify that the USER_PROFILE and DELETE_USER_PROFILE channel status are Active and Running

- Verify that the user has permission on the PERSON_DATA channel and the following:

  — *For a PeopleSoft HRMS employee*
  Verify that the user has permission on the PERSON_BASIC_SYNC and WORKFORCE_SYNC messages

  — *For a PeopleSoft Application user*
  Verify that the user has permission on the USER_PROFILE and DELETE_USER_PROFILE messages

  > For each message, check the message properties and ensure that the status is active (by checking the Status checkbox).
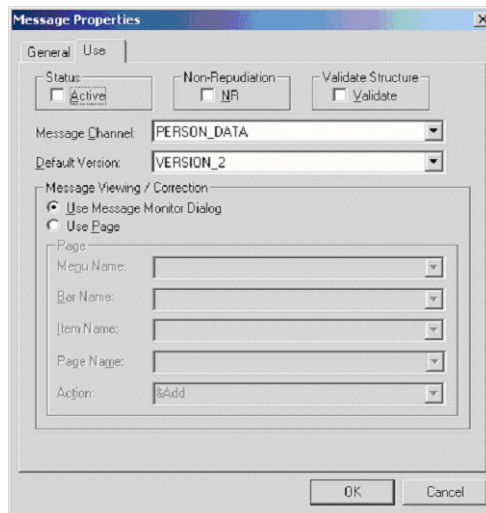
Configuration Steps

Complete the following steps using the web client to configure the Integration Broker:

1 Create a node with the following properties and options set:

a   Click the **Connector** tab and enter the following values:

For a PeopleSoft HRMS employee
Enter the following for the PRIMARY URL value:
http://ob-stlpool2.asiapacific.hpqcorp.net:7001/
PeopleSoftAgent/PeopleSoftEmpServlet?&siusername=
<siusername>&siuserpasswd=<encrypted siuserpassword>

To encrypt the password, run encode.bat (on Windows) or encode.sh (on UNIX), which is provided in the weblogic/keystore subdirectory in the Select Identity home directory. This utility prompts you for the password to encrypt and will generate the encrypted password. Be sure to copy the entire encrypted password in the field, as shown here:

For a PeopleSoft HCM Application user

Enter the following for the PRIMARY URL value:

`http://ob-stlpool2.asiapacific.hpqcorp.net:7001/`
`PeopleSoftAgent/PeopleSoftUsrServlet?&siusername=`
`<siusername>&siuserpasswd=<encrypted siuserpassword>`

> Again, to encrypt the password, run encode.bat (on Windows) or
> encode.sh (on UNIX), which is provided in the weblogic/keystore
> subdirectory in the Select Identity home directory. This utility prompts you
> for the password to encrypt and will generate the encrypted password. Be
> sure to copy the entire encrypted password in the field, as shown above.

2 *For a PeopleSoft HRMS employee*
Create a node transaction for employee personal information. Specify
these properties:

Also, make sure the transaction is active.

3 *For a PeopleSoft HRMS employee*
Create a node transaction for employee job information. Specify the
following properties:

Make sure the transaction is active.

4 *For a PeopleSoft HCM Application user*
Create a node transaction for user's personal information. Specify these
properties:

Also, make sure the transaction is active.

5 *For a PeopleSoft HCM Application user*
Create a node transaction for deleting a user's information. Specify the
following properties:

Also, make sure the transaction is active.

6   *For a PeopleSoft HCM Application user*
Generate the `pstools.properties` file for the PeopleSoft HCM
installation. To do so, open the configuration manager in your Peoplesoft
Installation as shown below:
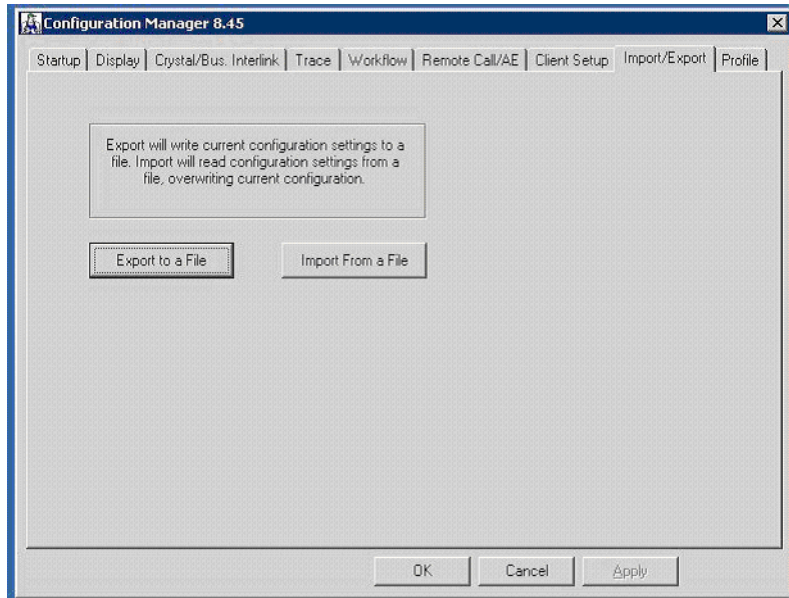
Export the configuration to a file and save the file as
`pstools.properties`. Place this file in the Select Identity home
directory where other configuration file for the connector are kept.

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 is an overview of the procedures you must complete to deploy your connector. It also provides connector-specific information you must provide when configuring Select Identity to use the connector.

⊘ Make sure the `commons-beanutils.jar` and `commons-collections-2.0.jar` files reside in the *WebLogic_home*/jdk/jre/lib/ext directory (typically `C:\bea\jdk142_05\jre\lib\ext` on Windows systems). The connector will not work if these JAR files are not present.

1 Register the 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*.

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

Connector Information	
* Connector Name:	PeopleSoftConnector
* Pool Name:	eis/PeopleSoftConnector
Mapper Available:	☐

2 Deploy a resource that uses the newly created connector. On the
Resources home page, click the **Deploy New Resource** button. Enter these
values:

Field Name	Sample Values	Description
Resource Name	PeopleSoftConnector	Name given to the resource. If you enabled reverse synchronization, this must be the same as the value provided for the urn:trulogica:concero:2.0#resourceId attribute on the agent console.
Resource Type	PeopleSoftHCM	The connector that was deployed in Step 1 on page 22.
Authoritative Source	No	Whether this resource is a system that is considered to be the authoritative source for user data in your environment. Specify **No** if the connector is not enabled for reverse synchronization. Specify **Yes** if you want to add users through 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 the PeopleSoft connector, select this option.
Host Name	Peoplesoft.hp.com	The NETBIOS name or IP address of the PeopleSoft server.
Server Port	9000	The Jolt port of the PeopleSoft server.
UserName	PeopleSoftuser	The user name on the Peoplesoft server, which has access as described earlier.
User Password	password	The user's password.

Field Name	Sample Values	Description
MappingFile	PeopleSoftEmp.xml	The XML mapping file used to map resource attributes to Select Identity attributes.
Symbolic ID	Sa1	The symbolic ID that will be assigned to the PeopleSoft user and the user that is created should have the same value. Note: This is relevent for PeopleSoft user only.
isEmployee	Yes/No	Whether the user is an employee. Specify **Yes** for PeopleSoft employee and **No** for PeopleSoft Application user.

Complete the steps in this procedure as described in the "Resources" chapter of the *HP OpenView Select Identity Administrator Guide.* After you deploy the resource for the PeopleSoft connector, the Basic Info page of the resource properties will look similar to this:

The Additional Info page will look similar to this:

The Access Info page will look similar to this (for a PeopleSoft HRMS employee):

Resource Access Information	
* Resource Name:	PeopleSoftConnector
* Host Name:	sint2.india.hp.com
* Server Port:	9000
* User Name:	PS
* User Password:	**
* Mapping File:	PeopleSoftEmp.xml
* Symbolic ID:	sa1
* isEmployee:	Yes

For a PeopleSoft HCM Application user, the Access Info pages will look similar to this:

Resource Access Information	
* Resource Name:	PeopleSoftUser
* Host Name:	15.70.182.125
* Server Port:	9000
* User Name:	PS
* User Password:	**
* Mapping File:	PeopleSoftUser.xml
* Symbolic ID:	sa1
* isEmployee:	No

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.

For a PeopleSoft HRMS employee, the View Attributes page for the resource will look similar to this:

Resource Name=PeopleSoftConnector				
◁◁ ◁ Page 1 of 1 ▷ ▷▷				Total Records:38
Name	**Min Length**	**Max Length**	**Attribute Mapped To**	**Authorative**
Addr1	1	128	Addr1	N
Addr2	1	128	Addr2	N
BASE_LANGUAGE_CD	1	64	Language	N
BIRTHCOUNTRY	1	64	Country	N
BIRTHDATE	1	64	Birthdate	N
BusAddr1	1	128	BusAttr1	N
BusAddr2	1	128	BusAddr2	N
BusCity	1	64	BusCity	N
BusCountry	1	64	BusCountry	N
BUSINESS_UNIT	1	64	CostCenter	N
BusZip	1	64	BusZip	N
Cell	1	64	PhMobile	N
City	1	64	City	N
COMPANY	1	128	Company	N
Country	1	64		
DEPTID	1	64	DEPTID	N
Effective Date	1	64	Effective Date	N
EMAIL_ADDR	1	64	Email	N
EMPL_STATUS	1	64	EMPL_STATUS	N
EMPL_STATUS_REASON	1	64	EMP_STATUS_REASON	N
EMPLID	1	64	EmployeeID	N
Fax	1	64	Fax	N

For a PeopleSoft HCM Application user, the View Attributes page for the resource will look similar to this:

Name	Min Length	Max Length	Attribute Mapped To	Authorative
AlternateUserID	1	64	AlternateUserID	N
BbEmail	1	64	BbEmail	N
BusEmail	1	64	BusEmail	N
CurrencyCode	1	64	CurrencyCode	N
Description	1	64	Description	N
EFFDT_FROM	1	64	EFFDT_FROM	N
EFFDT_TO	1	64	EFFDT_TO	N
EMAILID	1	64	Email	N
EMAILTYPE	1	64	EMAILTYPE	N
EMPLID	1	64	EmployeeID	N
HomeEmail	1	64	HomeEmail	N
Language	1	64	Language	N
LASTUPDOPRID	1	64	LASTUPDOPRID	N
NavigatorHomePage	1	64	NavigatorHomePage	N
Password	1	64	Password	N
PeopleSoftUser_ENTITLEMENTS	1	255	PeopleSoftUser_ENTITLEMENTS	Y
PeopleSoftUser_KEY	1	255	PeopleSoftUser_KEY	Y
PrimaryEmail	1	64	PrimaryEmail	N
PrimaryPermissionList	1	64	PrimaryPermissionList	N
ProcessProfile	1	64	ProcessProfile	N
RowSecurityPermissionList	1	64	RowSecurityPermissionList	N
SupervisingUserID	1	64	SupervisingUserID	N
UserDescription	1	64	UserDescription	N
UserIDAlias	1	64	UserIDAlias	N
UserName	1	64	UserName	N

Resource Name=PeopleSoftUser — Page 1 of 1 — Total Records: 26

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.

Keep the following in mind when creating the Service:

— 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 synchronization 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.

5 Modify the UserEnableDisableWFExtCall workflow external call, which disables a user across all Services to which he or she is assigned if the employee is disabled on the PeopleSoft resource. To do so, select **UserEnableDisableWFExtCal** on the External Calls home page then modify the call:

Home > External Calls > **Modify Call** : **UserEnableDisableWFExtCall**

Modify the information as desired for the external call. Click "Save & Continue" when finished.

Basic Information	
*External Call Name:	UserEnableDisableWFExtCall
Description:	Function to enable disable user depending on a specific attribue value
*Classname:	com.trulogica.truaccess.externalcall.workflow.Use
Classpath: (Separated by ;)	
*Call Type:	Workflow External Call
*Number of Parameters:	6

Provide the parameter values as follows:

Home > External Calls > **Modify Call** : **UserEnableDisableWFExtCall**

Modify parameters as desired for the external call. Click "Submit" when finished.

Basic Information		
External Call Name: UserEnableDisableWFExtCall		
Parameter Name	Parameter Value	Sensitive
1. AttributeName	EMPL_STATUS	☐
2. EnableValue	A	☐
3. DisableValue	S	☐
4. UserName	sisa	☐
5. Password	••••••	☑
6. Url	http://localhost:7001/lmz/webservice	☐

6 Modify the `TruAccess.properties` file to include the new workflow process that is executed for modify requests containing disabled user information. Insert the following entry:

```
truaccess.fixedtemplate.recon_modify.<ResourceName>==SI\ User\
Enable\ Disable\ Workflow
```

Replace <ResourceName> with that of the the PeopleSoft connector that was created in Step 2 on page 23.

7 Run the `PeopleSoftEmployeeStatus.sql` SQL script provided with the connector binaries to create a table in the Select Identity database. This table provides values that will be used to decrypt the status code for the employee as it changes.

8   Create an attribute called EMP_STATUS_REASON with the following properties:

Attribute Information	
✱Attribute Name	EMP_STATUS_REASON
✱Identity Object Type	User
✱Primitive Type	String
✱Attribute Type	Normal
✱Storage Type	normal
Description	
Default Help Text	
✱Multi Value	No
✱Min Length	1
✱Max Length	64
Value Pattern	
✱Self-Service Permission	Updateable
✱Default Display Name	EMP_STATUS_REASON
Default Display Mask	0
Default Display Length	0
Value Constraint Type	Dynamic
Value Constraint Function	Search Table
Value Generation Function	
Value Validation Function	

Provide the following arguments for the attribute:

Attribute Information	
Attribue Name:	EMP_STATUS_REASON
Argument Information: Constraint Function: Search Table	
valuefield:	ACTION
query:	select ACTION, DESCR from PS_ACTN_REASON_TBL
poolname:	jdbc/TruAccess

Then, map the attribute to the EMPL_STATUS_REASON attribute on the PeopleSoft resource.

9   Create the following attributes in Select Identity and map them to the
    attributes with similar names on the PeopleSoft resource:

Select Identity Resource Attribute	PeopleSoft Employee Attribute
UserName	UserName
Password	Password
BASE_LANGUAGE_CD	BASE_LANGUAGE_CD
EMPLID	EMPLID
EMAIL_ADDR	EMAIL_ADDR
FIRST_NAME	FIRST_NAME
MIDDLE_NAME	MIDDLE_NAME
LAST_NAME	LAST_NAME
NAME_PREFIX	NAME_PREFIX
PhBus	PhBus
PhRes	PhRes
Fax	Fax
Effective Date	EFFDT
BusCity	BusCity
BusAddr1	BusAddr1
BusAddr2	BusAddr2
BusZip	BusZip
BusCountry	BusCountry
Addr1	Addr1
Addr2	Addr2
BIRTHCOUNTRY	BIRTHCOUNTRY
BIRTHDATE	BIRTHDATE
ORIG_HIRE_DT	ORIG_HIRE_DT
EMPL_STATUS	EMPL_STATUS

Select Identity Resource Attribute	PeopleSoft Employee Attribute
EMPL_STATUS_REASON	EMPL_STATUS_REASON
SUPERVISOR_ID	SUPERVISOR_ID
DEPTID	DEPTID
BUSINESS_UNIT	BUSINESS_UNIT
LOCATION	LOCATION
COMPANY	COMPANY
JOBCODE	JOBCODE
State	State
Zip	Zip
City	City
Cell	Cell
Country	Country

3

Understanding the Mapping Files

The PeopleSoft connector is deployed with the following mapping files for PeopleSoft HRMS employees:

- `PeopleSoftEmp.xml`
- `PeopleSoftEmp.xsl`

and the following files for PeopleSoft HCM Application users:

- `PeopleSoftUsr.xml`
- `PeopleSoftUsr.xsl`

The XML file maps Select Identity user attributes to attributes on the PeopleSoft HCM server, which enables Select Identity to send account additions and modifications to PeopleSoft. If you configured the agent to support reverse synchronization, you must also configure the `PeopleSoftEmp.xsl` or `PeopleSoftUsr.xsl` file, which provides a reverse mapping of the Select Identity and resource fields mapped in the XML file.

When you deploy a resource through the Select Identity Resources pages, you can review the XML files. You can create attributes that are specific to Select Identity through the Attributes pages on the Select Identity client. These attributes can be used to associate Select Identity user accounts with system resources by mapping them to the connector mapping file described in this chapter.

The mapping files do not need to be edited unless you want to map additional attributes to your resource. If attributes and values are not defined in this mapping file, they cannot be saved to the resource through Select Identity. The files are created in XML, according to SPML standards, and are bundled in a JAR files called `PSEmpSchema.jar` and `PSUsrSchema.jar`.

XML Mapping Information

The following sections provide the mapping supported for PeopleSoft. The Select Identity resource attributes are editable. They reflect the identity information as seen in Select Identity. The physical resource attributes are literal attributes of user accounts on PeopleSoft. These attributes cannot be changed.

For PeopleSoft HRMS Employees

The following are the attribute mappings supported for PeopleSoft HRMS employees. These are listed in the `PeopleSoftEmp.xml` mapping file. You can add, modify, and delete attributes once you are familiar with the contents of this file.

Select Identity Attribute	PeopleSoft Resource Attribute
UserName	UserName
Password	Password
BASE_LANGUAGE_CD	BASE_LANGUAGE_CD
EMPLID	EMPLID
EMAIL_ADDR	EMAIL_ADDR
FIRST_NAME	FIRST_NAME
MIDDLE_NAME	MIDDLE_NAME
LAST_NAME	LAST_NAME
NAME_PREFIX	NAME_PREFIX
PhBus	PhBus

Select Identity Attribute	PeopleSoft Resource Attribute
PhRes	PhRes
Fax	Fax
Effective Date	EFFDT
BusCity	BusCity
BusAddr1	BusAddr1
BusAddr2	BusAddr2
BusZip	BusZip
BusCountry	BusCountry
Addr1	Addr1
Addr2	Addr2
BIRTHCOUNTRY	BIRTHCOUNTRY
BIRTHDATE	BIRTHDATE
ORIG_HIRE_DT	ORIG_HIRE_DT
EMPL_STATUS	EMPL_STATUS
EMPL_STATUS_REASON	EMPL_STATUS_REASON
SUPERVISOR_ID	SUPERVISOR_ID
DEPTID	DEPTID
BUSINESS_UNIT	BUSINESS_UNIT
LOCATION	LOCATION
COMPANY	COMPANY
JOBCODE	JOBCODE
State	State
Zip	Zip
City	City
Cell	Cell
Country	Country

For PeopleSoft HCM Application Users

The following are the attribute mappings supported for PeopleSoft HCM Application users. These are listed in the `PeopleSoftUsr.xml` mapping file. You can add, modify, and delete attributes once you are familiar with the contents of this file.

Select Identity Attribute	PeopleSoft Resource Attribute	Description
UserName	OPRID	Must be unique
Password	Password	
Language	BASE_LANGUAGE_CD	
UserDescription	UserDescription	
PrimaryPermissionList	PrimaryPermissionList	
RowSecurityPermission-List	RowSecurityPermission-List	
NavigatorHomePage	NavigatorHomePage	
ProcessProfile	ProcessProfile	
CurrencyCode	CurrencyCode	
UserIDAlias	UserIDAlias	Must be unique
AlternateUserID	AlternateUserID	
SupervisingUserID	SupervisingUserID	
EMPLID	EMPLID	
EMAILID	EMAILID	Any other type of email address created in the PeopleSoft HCM Application other than pre-defined email types

Select Identity Attribute	PeopleSoft Resource Attribute	Description
EMAILTYPE	EMAILTYPE	Works in tandem with EMAILID; if a value is assigned to EMAILTYPE such as HOME, BUS, WORK, or BB, EMAILTYPE is set to the email value of the corresponding type.
PrimaryEmail	PrimaryEmail	Takes the values HOME, BUS, WORK, and BB and makes the corresponding email address the primary email address for the user; by default, it sets the value of EMAILID to OTH email address and makes it primary
BusEmail	BusEmail	Business email address
WorkEmail	WorkEmail	Work email address
HomeEmail	HomeEmail	Home email address
BbEmail	BbEmail	BlackBerry email address
EFFDT_FROM	EFFDT_FROM	
EFFDT_TO	EFFDT_TO	

Elements in the XML Mapping Files

Here is an explanation of the format of the XML mapping files.

- **<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 Informix.

  - **<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)

    — 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 and will
   throw a permission exception

— bypass — the operation is not supported by the connector but will
   not throw an exception; the operation is simply bypassed

Here is an example:

```
<objectClassDefinition name="User" description="User">
  <properties>
  <attr name="GETCHILDREN">
    <value>true</value>
  </attr>
  <attr name="DELETE">
    <value>true</value>
  </attr>
  <attr name="EXPIREPASSWORD">
    <value>false</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. In general, the attribute assigned to tafield should be
the same as the physical resource attribute, or at least the
connector attribute. For example, it is recommended to have the
following:

```
<attributeDefinitionReference name="FirstName"
required="false" concero:tafield="[givenname]"
concero:resfield="UserName" concero:init="true"
concero:isMulti="true"/>
```

instead of this:

```
<attributeDefinitionReference name="FirstName"
required="false" concero:tafield="[FirstName]"
concero:resfield="UserName" concero:init="true"
concero:isMulti="true"/>
```

— 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.

Also, the attribute name may be case-sensitive; for example, if the
attribute is defined in all uppercase letters on the resource, be
sure to specify it in all uppercase letters here.

— 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.

Note that for a key field mapping where isKey="true" and tafield
is not assigned the UserName attribute, UserName should not be
used in any other mapping. That is, UserName can be assigned to
tafield only in cases where it is mapped to the key field in the
resource. Example:

```
<attributeDefinitionReference name="UserName"
required="true" concero:tafield="[UserName]"
concero:resfield="uid" concero:isKey="true"
concero:init="true"/>
```

— 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 name="ATTR_UserName"
  required="true" concero:tafield="UserName"
  concero:resfield="[x_user_name] [USER_NAME] [] [VARCHAR]"
  concero:isKey="true" concero:init="true"/>
...
```

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 Informix 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 Informix 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 Informix 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 name="ATTR_ResponsibilityKey"
  description="Responsibility Key" type="xsd:string" >
    <properties>
      <attr name="minLength">
        <value>1</value>
      </attr>
      <attr name="maxLength">
        <value>128</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.

Elements in the XSL Reverse Mapping File

If the agent is installed on the resource and you wish to enable reverse synchronization, you must configure the PeopleSoftEmp.xsl file to map all attributes that are specified in the XML mapping file.

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 XSL code translates all of the resource attribute names to lowercase for easier comparison. Then, the resource attribute (employeeid in this example) is mapped to the Select Identity attribute EmployeeID.

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

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 the PeopleSoft connector (`PeopleSoftConnector.rar`) from the Select Identity server:

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 Agent

Perform the following to delete the PeopleSoft agent
(`PeopleSoftAgent.war`) from the Select Identity server:

1   Log on to the WebLogic Server Console.

2   Navigate to *My_Domain* →**Deployments** →**Web Application Modules**.

3   Click the delete icon next to the web application
    (`PeopleSoftAgent.war`) that you want to uninstall.

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

5   Click **Continue**.