

# **HP OpenView Select Identity**

## **Universal Connector**

### **Installation and Configuration Guide**

**Connector Version: 3.3**  
**Select Identity Version: 3.3**



**April 2005**

© 2005 Hewlett-Packard Development Company, L.P.

# 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

© 2005 Hewlett-Packard Development Company, L.P.

No part of this document may be copied, reproduced, or translated into another language without the prior written consent of Hewlett-Packard Company. The information contained in this material is subject to change without notice.

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](http://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

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

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

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

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

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

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

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

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

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

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

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

## Support

Please visit the HP OpenView web site at:

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

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

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

**<http://support.openview.hp.com/>**

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

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

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

To find more information about access levels, go to:

**[http://support.openview.hp.com/access\\_level.jsp](http://support.openview.hp.com/access_level.jsp)**

To register for an HP Passport ID, go to:

**<https://passport.hp.com/hpp2/newuser.do>**

# contents

<b>Chapter 1</b>	<b>Installing the Connector</b> .....	7
	System Requirements .....	8
	Deploying on the Web Application Server .....	8
<b>Chapter 2</b>	<b>Understanding the Mapping File</b> .....	12
	General Information .....	13
	User Attribute Mappings .....	16
	Entitlement Attribute Mappings .....	17
<b>Chapter 3</b>	<b>Configuring the Connector</b> .....	19
<b>Chapter 4</b>	<b>Uninstalling the Connector</b> .....	24
	On WebLogic .....	24
	On WebSphere .....	25
<b>Appendix A</b>	<b>Sample SPML Requests</b> .....	26
<b>Appendix B</b>	<b>Sample SPML-to-CVS Conversion</b> .....	36

# Installing the Connector

The Universal connector enables HP OpenView Select Identity to generate a request and write the request to a file or email message for each provisioning task performed on the Select Identity user interface. The resulting request is, by default, in SPML format, though you can specify the format of the request by passing a parameter to the connector and providing an XSL translation file. This is useful, for example, if you need to create a CSV request. The connector also works as a NULL connector, producing no output. This functionality can be used to track user attributes.

The Universal connector enables Select Identity to perform the following tasks:

- Add, update, and remove users
- Enable and disable users
- Reset user passwords
- Expire and unexpire user passwords
- Assign and unassign entitlements to and from users

The Universal connector is packaged in the following files, which are located in the `Universal connector` directory on the Select Identity Connector CD:

- `UnivConnectorSchema.jar` — contains the default attribute mappings for the connector
- `UniversalConnector.rar` — contains the connector's binary files

## System Requirements

The Universal connector is supported in the following environment:

Select Identity Version	Application Server	Database
3.3	WebLogic 8.1.4 on Windows 2003	SQL Server 2000
	WebLogic 8.1.4 on Solaris 9	Oracle 9i
	WebLogic 8.1.4 on Red Hat Enterprise Linux 3.0	Oracle 9i
	WebLogic 8.1.4 on HP-UX 11i	Oracle 9i
	WebSphere 5.1.1 on Solaris 9	DB2 8.2 (or DB2 8.1 Service Pack 7)

## Deploying on the Web Application Server

To install the Universal 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 `UniversalConnector.rar` file from the Select Identity Connector CD to the connector subdirectory.



- 3 If deploying the connector on WebLogic, complete the following steps. If deploying on WebSphere, skip to [Step 4 on page 10](#).
  - a 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.)
  - b Extract the contents of the `UnivConnectorSchema.jar` file (on the Select Identity Connector CD) to the schema subdirectory.
  - c Ensure that the `CLASSPATH` environment variable in the WebLogic server startup script references the schema subdirectory.
  - d If you wish to configure the connector to produce a file that is not formatted in SPML, such as to create a CSV file, you must provide an XSL file that formats the SPML request created by the connector and translates it to the desired format. The `UnivConnectorSchema.jar` file contains a sample XSL file called `SPML-CSV.xsl` that converts SPML to entries in a CSV file. Refer to this file for an example.

Also, you must store this file in a directory that is referenced by the `CLASSPATH` of the application server.
  - e Start the application server if it is not currently running.
  - f Log on to the WebLogic Server Console.
  - g Navigate to **My\_domain** → **Deployments** → **Connector Modules**.
  - h Click **Deploy a New Connector Module**.
  - i Locate and select the `UniversalConnector.rar` file from the list. It is stored in the connector subdirectory.
  - j Click **Target Module**.
  - k Select the **My Server** (your server instance) check box.
  - l Click **Continue**. Review your settings.
  - m Keep all default settings and click **Deploy**. The Status of Last Action column should display Success.

4 If deploying the connector on WebSphere, complete the following steps:

- a Stop the application server.
- b Extract the contents of the `UnivConnectorSchema.jar` file (on the Select Identity Connector CD) to the `WebSphere\AppServer\lib\ext` directory.
- c If you wish to configure the connector to produce a file that is not formatted in SPML, such as if you wish the connector to create a CSV file, you must provide an XSL file that formats the SPML request created by the connector and translates it to the desired format. The `UnivConnectorSchema.jar` file contains a sample XSL file called `SPML-CSV.xsl` that converts SPML to entries in a CSV file. Refer to this file for an example.

Also, you must store this file in a directory that is referenced by the CLASSPATH of the application server.

- d Start the application server.
- e Log on to the WebSphere Application Server Console.
- f Navigate to **Resources** → **Resource Adapters**.
- g Click **Install RAR**.
- h In the Server path field, enter the path to the `UniversalConnector.rar` file. It is stored in the subdirectory created in [Step 1](#).
- i Click **Next**.
- j In the Name field, enter a name for the connector.
- k Click **OK**.
- l Click the **Save** link (at the top of the page).
- m On the Save to Master Configuraton dialog, click the **Save** button.
- n Click **Resources** → **Resource Adapters**.
- o Click the new connector.
- p Click **J2C Connection Factories** in the Additional Properties table.
- q Click **New**.
- r In the Name field, enter the name of the factory for the connector. For the SQL connector, enter **eis/UniversalConnector**.

- s** Click **OK**.
  - t** Click the **Save** link.
  - u** On the Save to Master Configuraton dialog, click the **Save** button.
  - v** Restart WebSphere.
- 5** Modify the mapping file, if necessary. See [Understanding the Mapping File on page 12](#) for details.
  - 6** Modify the `TruAccess.properties` file that was installed with the Select Identity server and set the following properties:
    - `truaccess.mailSession` — The JNDI name of the mail session
    - `truaccess.sender.name` — The name that is used as the sender's name if an email is sent
    - `truaccess.sender.email` — A valid email address that is used as the “from” address if an email is sent

After installing the connector, see [Configuring the Connector on page 19](#) for information about registering and configuring the connector in Select Identity.

## Understanding the Mapping File

The Universal connector is deployed with the `UnivConnectorMapping.xml` mapping file, which lists the default attributes that are mapped. The file is created in XML, according to SPML standards, and is bundled in a JAR file called `UnivConnectorSchema.jar`. When you deploy a resource using the Resources page of the Select Identity client, you can review this file.

You can create attributes that are specific to Select Identity using the Attributes page in the Select Identity client. These attributes can be used to associate Select Identity user accounts with system resources by editing the connector mapping file described in this chapter. This process becomes necessary because, for example, a single attribute “username” can have a different name on different resources, such as “login” for UNIX, “UID” for a database, and “userID” on a Windows server.

You can edit this mapping file to change the attribute mappings. If attributes and values are not defined in this mapping file, they cannot be supported by the connector.

# General Information

The following operations can be performed in the mapping file:

- Add a new attribute mapping
- Delete an existing attribute mapping
- Modify attribute mappings

Here is an explanation of the elements in the XML mapping files provided by the Universal connector:

- **<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.

- **<properties>**

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

- Create (CREATE)
- Read (READ)
- Update (UPDATE)
- Delete (DELETE)
- Enable (ENABLE)
- Disable (DISABLE)
- Reset password (RESET\_PASSWORD)
- Expire password (EXPIRE\_PASSWORD)
- Change password (CHANGE\_PASSWORD)

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 `PermissionException`
- bypass — the operation is not supported by the connector but will not throw any exception; the operation is simply bypassed

Here is an example:

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

- **<memberAttributes>**

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.

Each <attributeDefinitionReference> element contains the following attributes:

- Name — the name of the reference.
- Required— if this attribute is required in the provisioning (set to true or false).
- Concero:tafield — the name of the Select Identity resource attribute.
- Concero:resfield — the name of the physical resource attribute from the resource schema. If the resource does not support an explicit schema (such as UNIX), this can be a tag field that indicates a resource attribute mapping.

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

Here is an example:

```
<memberAttributes>
  <attributeDefinitionReference name="User Name"
    required="true" concero:init="true"
    concero:tafield="User Name"
    concero:resfield="userName" concero:isKey="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 Universal connector has code to interpret the mappings in one way, as follows:

- The connector attribute names are specified in square braces, like this: `[xyz]`. The value of attribute `xyz` is taken from the UserModel during provisioning.
- Composite attributes can be specified in the Universal 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. Universal connector has code to handle these composite mappings.

- **<attributeDefinition>**

Defines the properties of each object's attribute. For example, the attribute definition for the HomeDir attribute defines that it must be between zero and 100 characters in length and can contain the following letters, numbers, and characters: a-z, A-Z, 0-9, @, +, and a space.

Here is an excerpt from the `UnivConnectorMapping.xml` file:

```
<attributeDefinition name="HomeDir" description="User Home
directory" type="xsd:string" >
  <properties>
```

```

    <attr name="minLength">
      <value>0</value>
    </attr>
    <attr name="maxLength">
      <value>128</value>
    </attr>
    <attr name="pattern">
      <value><![CDATA[[a-zA-Z0-9@]+]]> </value>
    </attr>
  </properties>
</attributeDefinition>

```

- **<concerno:entitlementMappingDefinition>**

Defines how entitlements are mapped to users.

- **<concerno:objectStatus>**

Defines how to assign status to a user.

- **<concerno:relationshipDefinition>**

Defines how to create relationships between users.

## User Attribute Mappings

The following are the attribute mappings supported by this connector for users. These are listed in the `UnivConnectorMapping.xml` mapping file. You can add, modify, or delete attributes once you are familiar with the contents of this file. You can edit the Select Identity resource attributes; they reflect the identity information as seen in Select Identity. The resource attributes are literal attributes of user accounts generated in the output requests. See the *HP OpenView Select Identity Connector Developer Guide* for more information about attributes and mapping information.

Select Identity Resource Attribute	Request Attribute
User Name	userName
Password	password
Person Number	userId
Home Directory	directory



Select Identity Resource Attribute	Request Attribute
First Name	firstName
Last Name	lastName
Middle Name	middleName
[First Name] [Middle Name] [Last Name]	fullName
Email	email
Department	department
Title	title
Company	company
PhBus	busPhone
PhHome	homePhone
Fax Number	faxNumber
Address 1	address1
Address 2	address2
City	city
State	state
Zip	zip
Country	country
Description	description
Job Description	jobDescription

## Entitlement Attribute Mappings

The following entitlements are mapped by default in the connector's mapping file:

- Administrators
- Backup Operators

- Guests
- Network Configuration Operators
- Power Users
- Remote Desktop Users
- Replicator
- Users
- Debugger Users
- HelpServicesGroup

You can edit the XML mapping file (or create a new one) to reflect a new set of entitlements.

## Configuring the Connector

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

- 1 Register the 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:	Universal Connector
* Pool Name:	<input type="text" value="eis/UniversalConnector"/>
Mapper Available:	<input type="checkbox"/>

- 2 Deploy a resource that uses the newly created connector. On the Resources home page, click the **Deploy New Resource** button. When configuring the resource, refer to the following table for parameters specific to this connector:

Field Name	Sample Values	Description
Resource Name	Universal_connector	Name of the target resource.
Resource Type	ASCII	The connector that was deployed in <a href="#">Step 1 on page 19</a> .
Authoritative Source	No	Whether this resource is a system that is considered to be the authoritative source for user data in your environment. You must specify <b>No</b> because the connector cannot synchronize account data with the Select Identity server.
Associate to Group	Selected	Whether the system uses the concept of groups. For the Universal connector, select this option.
File	C:\tmp\Request.txt	The path to the file that will be generated. Specify a value in this field only if you wish to generate a file.
Mail To	admin@abc.com	The email address to which the request is sent. Specify a value in this field only if you wish to send an email.
Mail Subject	Provisioning Request	The subject of the email message. Specify a value in this field only if you wish to send an email.

Field Name	Sample Values	Description
XSLT File	SPML-CSV.xsl	Name of the XSL file, if you wish to create a file that is not formatted using SPML.
Mapping File	UnivConnectorMapping.xml	Location of the connector mapping file.

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, the Access Info properties for the resource will look similar to this if you configured the connector to create an SPML file (the default format):

Resource Access Information	
* Resource Name:	Resource
File:	C:\tmp\request.txt ?
Mail To:	?
Mail Subject:	?
XSLT File:	?
* Mapping File:	UnivConnectorMapping.xml ? (View)

The Access Info properties for the resource will look similar to this if you configured the connector to create an email message:

Resource Access Information	
* Resource Name:	Resource
File:	?
Mail To:	admin@hp.com ?
Mail Subject:	Provisioning Request ?
XSLT File:	?
* Mapping File:	UnivConnectorMapping.xml ? (View)

The Access Info properties for the resource will look similar to this if you configured the connector as a NULL connector (to not create an output file):

Resource Access Information	
* Resource Name:	Resource
File:	<input type="text"/> ?
Mail To:	<input type="text"/> ?
Mail Subject:	<input type="text"/> ?
XSLT File:	<input type="text"/> ?
* Mapping File:	<input type="text" value="UnivConnectorMapping.xml"/> ? ( View )

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

Refer to the “Attributes” chapter in the *HP OpenView Select Identity Administrator Guide* for more information. After you create the attributes

for the iPlanet Active Directory connector, the View Attributes page for the resource will look similar to this:

Resource Attribute	MinLength	MaxLength	Mapped To	Authoritative
A2_ENTITLEMENTS	1	255	A2_ENTITLEMENTS	<input checked="" type="checkbox"/>
A2_KEY	1	255	A2_KEY	<input checked="" type="checkbox"/>
Address 1	1	1024	Addr1	<input type="checkbox"/>
Address 2	1	1024	Addr2	<input type="checkbox"/>
City	0	255	City	<input type="checkbox"/>
Company	1	255	Company	<input type="checkbox"/>
Country	0	255	Country	<input type="checkbox"/>
Department	1	255	Department	<input type="checkbox"/>
Description	1	1024	(Select one)	<input type="checkbox"/>
Email	0	255	Email	<input type="checkbox"/>
Fax Number	1	255	(Select one)	<input type="checkbox"/>
First Name	0	255	FirstName	<input type="checkbox"/>
Home Directory	0	2048	(Select one)	<input type="checkbox"/>
Job Description	0	1024	(Select one)	<input type="checkbox"/>
Last Name	1	255	LastName	<input type="checkbox"/>
Middle Name	1	255	(Select one)	<input type="checkbox"/>
Password	1	255	Password	<input type="checkbox"/>
Person Number	1	255	(Select one)	<input type="checkbox"/>
PhBus	1	255	PhBus	<input type="checkbox"/>
PhHome	1	255	PhHome	<input type="checkbox"/>
State	0	255	State	<input type="checkbox"/>
Title	1	255	(Select one)	<input type="checkbox"/>
User Name	1	255	UserName	<input type="checkbox"/>
Zip	0	255	Zip	<input type="checkbox"/>

- 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.

# 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 using the Select Identity client Connectors pages.

## On WebLogic

Perform the following to delete a connector:

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



## On WebSphere

Complete the following steps to uninstall the connector on WebSphere:

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



## Sample SPML Requests

The following are the sample requests generated for most of the user provisioning operations invoked from the Select Identity interface. Use these samples to create an XSL file that will translate output files that are in a format other than SPML.

- **Create user**

This generates one `addRequest` that contains all attributes of the user.

```
<addRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concerro:2.0#date'>
      <value>Wed Mar 16 16:57:43 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <attributes>
    <attr name='password'>
      <value>abcd1234</value>
    </attr>
    <attr name='userName'>
      <value>user031601</value>
    </attr>
```

```

    <attr name='company'>
      <value>HP</value>
    </attr>
  </attributes>
</addRequest>

```

- **Update to user attributes**  
This generates one `modifyRequest`.

```

<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concerro:2.0#date'>
      <value>Wed Mar 16 18:55:29 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <modifications>
    <modification name='password' operation='replace'>
      <value>abcd1234</value>
    </modification>
    <modification name='busPhone' operation='replace'>
      <value></value>
    </modification>
    <modification name='address1' operation='replace'>
      <value></value>
    </modification>
    <modification name='homePhone' operation='replace'>
      <value></value>
    </modification>
    <modification name='firstName' operation='replace'>
      <value></value>
    </modification>
    <modification name='lastName' operation='replace'>
      <value></value>
    </modification>
    <modification name='company' operation='replace'>
      <value>HP</value>
    </modification>
    <modification name='address2' operation='replace'>
      <value></value>
    </modification>
    <modification name='country' operation='replace'>

```

```

        <value></value>
    </modification>
    <modification name='state' operation='replace'>
        <value></value>
    </modification>
    <modification name='email' operation='replace'>
        <value>user031601@hp.com</value>
    </modification>
    <modification name='zip' operation='replace'>
        <value></value>
    </modification>
    <modification name='city' operation='replace'>
        <value></value>
    </modification>
</modifications>
</modifyRequest>

```

- **Reset password**

This generates one `extendedRequest` with the

`urn:trulogica:concero:2.0#resetPassword` operation and the `newPassword` attribute to which the new password value is assigned.

```

<extendedRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concero:2.0#date'>
      <value>Wed Mar 16 18:56:57 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <providerIdentifier
    providerIDType='urn:oasis:names:tc:SPML:1:0#URN'>
    <providerID>urn:trulogica:concero:2.0#connector</providerID>
  </providerIdentifier>
  <operationIdentifier
    operationIDType='urn:oasis:names:tc:SPML:1:0#URN'>
    <operationID>urn:trulogica:concero:2.0#resetPassword
    </operationID>
  </operationIdentifier>
  <attributes>
    <attr name='urn:trulogica:concero:2.0#newPassword'>
      <value>abc123</value>
    </attr>
  </attributes>
</extendedRequest>

```

```

    </attr>
  </attributes>
</extendedRequest>

```

- **Disable all services**

This generates one `extendedRequest` that includes the `urn:trulogica:concero:2.0#disable` operation.

```

<extendedRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concero:2.0#date'>
      <value>Wed Mar 16 17:27:22 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <providerIdentifier
    providerIDType='urn:oasis:names:tc:SPML:1:0#URN'>
    <providerID>urn:trulogica:concero:2.0#connector
    </providerID>
  </providerIdentifier>
  <operationIdentifier
    operationIDType='urn:oasis:names:tc:SPML:1:0#URN'>
    <operationID>urn:trulogica:concero:2.0#disable
    </operationID>
  </operationIdentifier>
</extendedRequest>

```

- **Enable all services**

This generates one `extendedRequest` that includes the `urn:trulogica:concero:2.0#enable` operation.

```

<extendedRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concero:2.0#date'>
      <value>Wed Mar 16 17:28:21 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <providerIdentifier

```

```

    providerIDType='urn:oasis:names:tc:SPML:1:0#URN'>
      <providerID>urn:trulogica:concero:2.0#connector</providerID>
    </providerIdentifier>
    <operationIdentifier
      operationIDType='urn:oasis:names:tc:SPML:1:0#URN'>
        <operationID>urn:trulogica:concero:2.0#enable</operationID>
      </operationIdentifier>
    </extendedRequest>

```

- **Modify user entitlements**

This generates one `modifyRequest` for user attributes followed by a `modifyRequest` for each of the entitlements that was changed.

```

<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concero:2.0#date'>
      <value>Wed Mar 16 17:31:40 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <modifications>
    <modification name='password' operation='replace'>
      <value>qwer1234</value>
    </modification>
    <modification name='busPhone' operation='replace'>
      <value></value>
    </modification>
    <modification name='address1' operation='replace'>
      <value></value>
    </modification>
    <modification name='homePhone' operation='replace'>
      <value></value>
    </modification>
    <modification name='firstName' operation='replace'>
      <value></value>
    </modification>
    <modification name='lastName' operation='replace'>
      <value></value>
    </modification>
    <modification name='company' operation='replace'>
      <value>HP</value>
    </modification>

```

```

    <modification name='address2' operation='replace'>
      <value></value>
    </modification>
    <modification name='country' operation='replace'>
      <value></value>
    </modification>
    <modification name='state' operation='replace'>
      <value></value>
    </modification>
    <modification name='email' operation='replace'>
      <value></value>
    </modification>
    <modification name='zip' operation='replace'>
      <value></value>
    </modification>
    <modification name='city' operation='replace'>
      <value></value>
    </modification>
  </modifications>
</modifyRequest>

```

- **Removal of Administrators entitlement**  
This generates one modifyRequest that deletes the entitlement.

```

<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concerro:2.0#date'>
      <value>Wed Mar 16 17:31:49 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <modifications>
    <modification name='urn:trulogica:concerro:2.0#groups'
operation='delete'>
      <value>Administrators</value>
    </modification>
  </modifications>
</modifyRequest>

```

- **Addition of Backup Operators entitlement**

This generates one modifyRequest that adds the entitlement.

```
<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concerro:2.0#date'>
      <value>Wed Mar 16 17:31:50 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
  <modifications>
    <modification name='urn:trulogica:concerro:2.0#groups'
      operation='add'>
      <value>Backup Operators</value>
    </modification>
  </modifications>
</modifyRequest>
```

- **Delete user (terminate user)**

This generates a deleteRequest.

```
<deleteRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concerro:2.0#date'>
      <value>Wed Mar 16 17:54:49 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user031601</id>
  </identifier>
</deleteRequest>
```

- **Add user with Debugger Users and Guests entitlements**

This generates one addRequest followed by two modifyRequests, one for each entitlement added.

```
<addRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concerro:2.0#date'>
      <value>Mon Mar 21 15:10:25 CST 2005</value>
```



```

        </attr>
    </operationalAttributes>
    <identifier
        type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
        <id>user032102</id>
    </identifier>
    <attributes>
        <attr name='userName'>
            <value>sk032102</value>
        </attr>
        <attr name='company'>
            <value>HP</value>
        </attr>
    </attributes>
</addRequest>

<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
    execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
    <operationalAttributes>
        <attr name='urn:trulogica:concerro:2.0#date'>
            <value>Mon Mar 21 15:10:26 CST 2005</value>
        </attr>
    </operationalAttributes>
    <identifier
        type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
        <id>user032102</id>
    </identifier>
    <modifications>
        <modification name='urn:trulogica:concerro:2.0#groups'
            operation='add'>
            <value>Debugger Users</value>
        </modification>
    </modifications>
</modifyRequest>

<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
    execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
    <operationalAttributes>
        <attr name='urn:trulogica:concerro:2.0#date'>
            <value>Mon Mar 21 15:10:26 CST 2005</value>
        </attr>
    </operationalAttributes>
    <identifier
        type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
        <id>user032102</id>
    </identifier>

```

```

<modifications>
  <modification name='urn:trulogica:concero:2.0#groups'
    operation='add'>
    <value>Guests</value>
  </modification>
</modifications>
</modifyRequest>

```

- **Delete Service membership**

This generates one `modifyRequest` that deletes the entitlement for each of the user's entitlements, followed by one `deleteRequest`.

```

<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concero:2.0#date'>
      <value>Mon Mar 21 15:10:26 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user032102</id>
  </identifier>
  <modifications>
    <modification name='urn:trulogica:concero:2.0#groups'
      operation='delete'>
      <value>Debugger Users</value>
    </modification>
  </modifications>
</modifyRequest>

<modifyRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concero:2.0#date'>
      <value>Mon Mar 21 15:10:26 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user032102</id>
  </identifier>
  <modifications>
    <modification name='urn:trulogica:concero:2.0#groups'
      operation='delete'>
      <value>Guests</value>
    </modification>
  </modifications>
</modifyRequest>

```

```
        </modification>
    </modifications>
</modifyRequest>

<deleteRequest xmlns='urn:oasis:names:tc:SPML:1:0'
  execution='urn:oasis:names:tc:SPML:1:0#asynchronous'>
  <operationalAttributes>
    <attr name='urn:trulogica:concero:2.0#date'>
      <value>Mon Mar 21 15:11:58 CST 2005</value>
    </attr>
  </operationalAttributes>
  <identifier
    type='urn:oasis:names:tc:SPML:1:0#UserIDAndOrDomainName'>
    <id>user032102</id>
  </identifier>
</deleteRequest>
```



## Sample SPML-to-CVS Conversion

The UnivConnectorSchema.jar file provided with the connector contains a sample XSL file called SPML-CSV.xsl that converts SPML requests to entries in a CSV file. The following is the output of this XSL file for the operation requests generated by the connector.

- Add a new user with two entitlements:

```
userAdd user032201 password="abc123",userName="user032201",
country="USA",fullName="user Test032201",
email="user032201@hp.com",state="Plano",firstName="user",
city="TX",department="OVSI",company="HP",lastName="Test032201"

entAdd user032201 "Remote Desktop Users"
entAdd user032201 "Users"
```

- Modify user attributes, remove one entitlement, and add a new entitlement:

```
userModify user032201 password="abc123",busPhone="",
address1="",homePhone="",firstName="user",
department="HP OVSI",lastName="Changed",company="HP",
address2="",country="USA",fullName="user Changed",
state="Plano",email="user032201@hp.com",zip="75024",city="TX"

entAdd user032201 "Network Configuration Operators"
entDelete user032201 "Users"
```

- **Disable Service membership:**

```
entDelete user032201 "Network Configuration Operators"  
entDelete user032201 "Remote Desktop Users"
```

- **Enable Service membership:**

```
entAdd user032201 "Network Configuration Operators"  
entAdd user032201 "Remote Desktop Users"
```

- **Disable all services:**

```
userDisable user032201
```

- **Enable all Services:**

```
userEnable user032201
```

- **Delete Service membership:**

```
entDelete user032201 "Network Configuration Operators"  
entDelete user032201 "Remote Desktop Users"  
userDelete user032201
```