

Oracle® Identity Manager

Tools Reference Guide

Release 9.0

B25939-01

May 2006

Oracle Identity Manager Tools Reference Guide, Release 9.0

B25939-01

Copyright © 1991, 2006, Oracle. All rights reserved.

Primary Authors: Vimmika Dinesh, Don Gosselin

Contributors: Jatan Rajvanshi, Raj Kuchi, Semyon Shulman

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software--Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

Oracle, JD Edwards, PeopleSoft, and Siebel are registered trademarks of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Contents

Audience.....	vii
Documentation Accessibility.....	vii
Related Documents.....	viii
Documentation Updates.....	viii
Conventions.....	viii
Online Help.....	viii
1 Introduction	
The Integration Problem	1-1
The Oracle Identity Manager Solution	1-1
About Adapters	1-2
Tabs of the Adapter Factory Form	1-3
Adapter Tasks.....	1-3
Execution Schedule.....	1-3
Resources.....	1-4
Variable List.....	1-4
Usage Lookup.....	1-4
Responses.....	1-4
2 Getting Started	
Overview	2-1
Configure Oracle Identity Manager to Reference Jar and Class Files.....	2-1
Installing Remote Manager.....	2-2
Adding the Trust Relation.....	2-2
3 Creating Adapters	
Overview	3-1
Defining Adapters	3-1
Disabling Adapters	3-3
About Adapter Variables	3-3
To Create an Adapter Variable.....	3-3
To Modify an Adapter Variable.....	3-5
To Delete an Adapter Variable.....	3-5
Creating Adapter Tasks	3-5
To Create a Java Task.....	3-7

To Create a Remote Task.....	3-11
To Create a Stored Procedure Task	3-12
To Create a Utility Task.....	3-15
To Create an Oracle Identity Manager API Task	3-17
To Create a Set Variable Task.....	3-19
To Add a Error Handler Task.....	3-20
To Create a Logic Task	3-21
Modifying Adapter Tasks	3-24
Changing the Order and Nesting of Tasks	3-24
Deleting Adapter Tasks.....	3-25
Working with Responses	3-26
To Create a Response.....	3-26
To Modify a Response	3-27
To Delete a Response.....	3-27
Scheduling Rule Generators and Entity Adapters	3-28
To Schedule Rule Generators and Entity Adapters	3-28
4 About Process Task Adapters	
Background.....	4-1
How a Process Task Adapter Works	4-2
Attaching Process Task Adapters to Process Tasks.....	4-3
Removing Process Task Adapters from Process Tasks.....	4-6
To Remove a Process Task Adapter from a Process Task.....	4-7
5 Applying Task Assignment Adapters	
Overview.....	5-1
Attaching Task Assignment Adapters to Process Tasks	5-2
To Attach a Task Assignment Adapter to a Process Task.....	5-2
Removing Task Assignment Adapters from Process Tasks	5-5
To Remove a Task Assignment Adapter from a Process Task.....	5-5
6 Understanding Rule Generators	
Overview.....	6-1
Mapping Rule Generator Adapter Variables.....	6-2
To Map a Rule Generator's Adapter Variables.....	6-2
Associating Rule Generators with Processes.....	6-5
Removing Rule Generators from Form Fields.....	6-5
To Remove a Rule Generator from a Form Field	6-5
7 Using Prepopulate Adapters	
Overview.....	7-1
Attaching Pre-Populate Adapters to Form Fields	7-2
To Attach a Prepopulate Adapter to a Form Field.....	7-2
Removing Pre-Populate Adapters from Form Fields	7-5
To Remove a Prepopulate Adapter from a Form Field	7-5

8	Managing Entities	
9	Compiling Adapters	
	Compiling Adapters	9-1
	To Compile Multiple Adapters	9-1
10	Exporting and Importing Adapters	
11	Creating and Testing a Remote Manager IT Resource	
	Post-installation Configuration	11-1
	To Create and Test a Remote Manager IT Resource	11-3
A	Tables	
	Class Files	A-1
	Naming Conventions.....	A-1
	Adapter Task Mapping Information	A-2
	Adapter Variables	A-2
	Adapter Task.....	A-2
	Literal	A-3
	Adapter References.....	A-3
	Organization Definition	A-3
	Process Definition	A-4
	User Definition	A-4
	Adapter Variable Mapping Information	A-5
	From the Variable List Tab	A-5
	Process Task Adapter Variable Mappings	A-6
	Task Assignment Adapter Variable Mappings	A-8
	Rule Generator and Entity Adapter Variable Mappings	A-10
	Pre-Populate Adapter Variable Mappings.....	A-11

Index

Preface

This preface introduces you to the *Oracle Identity Manager Tools Reference Guide* discussing the intended audience and conventions of this document. It also includes a list of related Oracle documents.

Note: This is a transitional release following Oracle's acquisition of Thor Technologies. Some parts of the product and documentation still refer to the original Thor company name and Xellerate product name and will be rebranded in future releases.

Audience

The *Oracle Identity Manager Tools Reference Guide* is intended for users of the Oracle Identity Manager Java client application. It is geared toward developers, who are responsible for creating adapters, and for Oracle Identity Manager Administrators, who attach these adapters to process tasks, forms that come packaged with Oracle Identity Manager, and user-created forms.

For further details on how to configure Oracle Identity Manager, refer to the Oracle Identity Manager Administrative and User Console Guide.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Accessibility standards will continue to evolve over time, and Oracle is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For more information, visit the Oracle Accessibility Program Web site at

<http://www.oracle.com/accessibility/>

Accessibility of Code Examples in Documentation

Screen readers may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, some screen readers may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation

This documentation may contain links to Web sites of other companies or organizations that Oracle does not own or control. Oracle neither evaluates nor makes any representations regarding the accessibility of these Web sites.

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Related Documents

This guide assumes that you have read and understood the following documents:

For more information, see the following documents in the Oracle Identity Manager documentation set:

- *Oracle Identity Manager Installation and Upgrade Guide for JBoss*
- *Oracle Identity Manager Installation and Upgrade Guide for WebLogic*
- *Oracle Identity Manager Installation and Upgrade Guide for WebSphere*
- *Oracle Identity Manager Administrative and User Console Customization Guide*
- *Oracle Identity Manager Design Console Guide*
- *Oracle Identity Manager Tools Reference Guide*
- *Oracle Identity Manager Audit Report Developer Guide*
- *Oracle Identity Manager Best Practices Guide*

Documentation Updates

Oracle is committed to delivering the best and most recent information available. For information about updates to the Oracle Identity Manager 9.0 documentation set, visit Oracle Technology Network at

<http://www.oracle.com/technology/documentation/index.html>

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
<i>italic</i>	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
monospace	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Online Help

Oracle Identity Manager comes equipped with a suite of online Help systems, designed to assist you with locating relevant information when you need it. The

various books within the Help system library are available using the Help menu (located in the main screen). To open a particular book, select it from the Help menu.

In addition, Oracle Identity Manager provides content-sensitive Help for a number of forms. The contents of the content-sensitive Help topic reflect the form that is active within the Oracle Identity Manager main screen. To launch a context-sensitive Help topic for a particular form, make sure it is active. Then, press F1.

Note: In addition to context-sensitive Help, the Oracle Identity Manager online Help system may also be invoked by pressing the F1 key.

Introduction

This chapter describes the integration problem and the solution that Oracle Identity Manager offers. It contains the following topics:

- ["The Integration Problem"](#) on page 1-1
- ["The Oracle Identity Manager Solution"](#) on page 1-1
- ["About Adapters"](#) on page 1-2
- ["Tabs of the Adapter Factory Form"](#) on page 1-3

The Integration Problem

An access rights management application cannot be successful today unless it can be integrated with other software solutions. Not only are there many resources, but also there is no single integration standard for connecting to these resources.

One way to tackle this challenge is by using the common functionality that is supported by all the integrations. To do this, you need developers who can write this code. In addition, every time an existing software resource is modified, or a new one is added, you must write more code.

The Oracle Identity Manager Solution

The Adapter Factory is a code-generation tool, provided by Oracle Identity Manager, which enables an Oracle Identity Manager administrator to create Java classes, known as adapters.

An adapter provides the following benefits:

- It extends the internal logic and functionality of Oracle Identity Manager.
- It interfaces with any software resource, by connecting to that resource using the API of the resource.
- It enables the integration between Oracle Identity Manager and an external system.
- It can be generated without manually writing code.
- It is lightweight and specific to your needs.
- It can be maintained easily because all of the definitions for the adapter are stored in a repository. This repository can be edited through a GUI.
- One Oracle Identity Manager user can retain the domain knowledge about the integration, while another user can maintain the adapter.

- It can be modified and upgraded efficiently.

About Adapters

As stated earlier, an adapter is a Java class that is created by an Oracle Identity Manager user through the Adapter Factory. In addition, an adapter is an aggregation of atomic function calls in a logical flow. These atomic steps are known as adapter tasks.

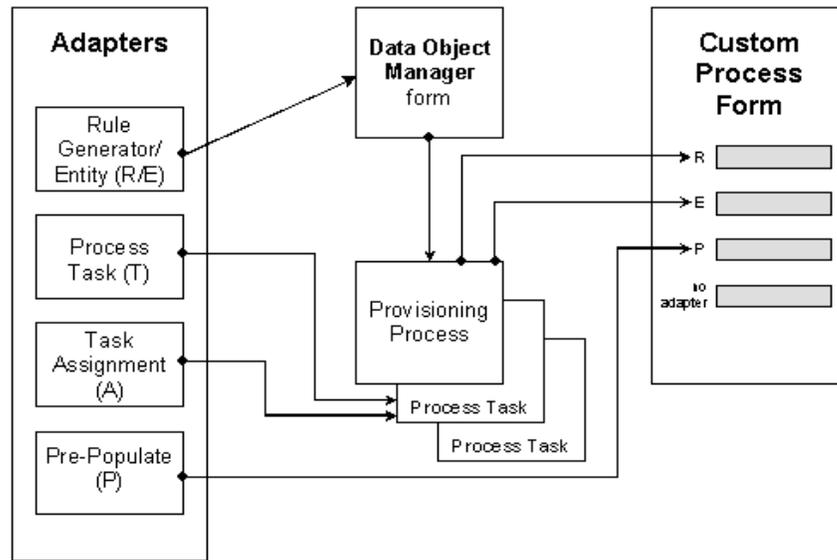
Note: Oracle Identity Manager can connect to external systems such as databases and directory servers using Java APIs for JDBC and LDAP. In addition, for all other APIs (such as C, C++, VB, COM/DCOM, and so on.), you can create a Java wrapper so that Oracle Identity Manager can communicate with the API directly.

There are five types of adapters:

- A process task adapter, which allows Oracle Identity Manager to automate the completion of a process task.
- A task assignment adapter, which enables Oracle Identity Manager to automate the assignment of a process task to a user or group.
- A rule generator, which incorporates business rules to the fields of either an Oracle Identity Manager form or a user-defined form (created using the Form Designer form), so these fields can be populated automatically and saved to the Oracle Identity Manager database.
- A pre-populate adapter, which is a specific type of rule generator adapter that can be attached to a user-created form field. The data, which is generated by a pre-populate adapter, can appear either automatically or manually. In addition, it uses criteria that enable Oracle Identity Manager to determine which pre-populate adapter will be applied to the designated form field to populate it, without saving this information to the Oracle Identity Manager database.
- An entity adapter, which is attached to an Oracle Identity Manager or user-created form field. Oracle Identity Manager triggers an entity adapter on preinsert, preupdate, predelete, postinsert, postupdate, or postdelete. Once this occurs, the field to which the adapter is attached will be populated automatically and saved to the Oracle Identity Manager database.

[Figure 1–1](#) illustrates the functionality of five different types of adapters.

Figure 1–1 Adapter Functionality



These five types of adapters will be explained in greater detail in the chapters that follow.

Note: For more information about the Form Designer form, refer to *Oracle Identity Manager Administrative and User Console Guide*.

Tabs of the Adapter Factory Form

The Adapter Factory form contains the following tabs:

- Adapter Tasks
- Execution Schedule
- Resources
- Variable List
- Usage Lookup
- Responses

Each of these tabs is covered in greater detail in the following sections.

Adapter Tasks

Within the Adapter Tasks tab, you can create and manage the atomic function calls of an adapter. These function calls are known as adapter tasks.

Execution Schedule

The Execution Schedule tab enables you to specify when you want Oracle Identity Manager to trigger a rule generator or an entity adapter. You can schedule Oracle Identity Manager to run a rule generator (Adapter Type *R*) on preinsert and/or preupdate. You can also configure Oracle Identity Manager to execute an entity adapter (Adapter Type *E*) on pre-insert, pre-update, pre-delete, post-insert, post-update, or post-delete.

Caution: Process task adapters and task assignment adapters, which are attached to process tasks, are triggered once the process task's status becomes *Pending*. So, because you do not need to specify when Oracle Identity Manager will trigger these types of adapters, Oracle Identity Manager disables the Execution Schedule tab for them.

Also, because Oracle Identity Manager always triggers prepopulate adapters on preinsert, Oracle Identity Manager disables the check boxes of this tab for prepopulate adapters.

Resources

From the Resources tab, you can:

- Click the Java APIs sub-tab to see the Java APIs that are being used by the adapter.
- Click the Other sub-tab to document a non-Java API file to the adapter, if necessary.

Variable List

From the Variable List tab, you can:

- Create, modify, and delete adapter variables.
- Set the data type and provide a description for each variable.
- Map an adapter variable to a literal or an adapter reference. You can also postpone the mapping until it is attached to a process task or a form field.

You also can resolve the value of the adapter variable at runtime, when it is attached to a process task and the process task is run. Thus, process-specific data is available to map to this variable.

Usage Lookup

For a process task or task assignment adapter, the Usage Lookup tab displays the process task to which the adapter is attached, as well as the process of which this process task is a member.

For a rule generator or entity adapter, this tab shows the Oracle Identity Manager form and associated data object to which the adapter is attached. In addition, it displays the execution schedule of the adapter, along with a sequence number that represents the order in which Oracle Identity Manager will trigger the adapter.

For a prepopulate adapter, this tab displays the user-defined form and form field to which the adapter is attached. Also, it shows the prepopulate rule that is associated with the adapter.

Note: For more information about the different types of adapters, refer to the [Chapter 3, "Creating Adapters"](#) on page 3-1.

Responses

The Responses tab is used for defining meaningful responses to the process task. These responses depend on the execution result of the adapter. The various errors messages returned by the external system can then be mapped to these responses in a way that

they make sense in the context of the process task. On attaching the adapter to a process task, the status bucket (Pending, Completed, Rejected) of the process task (and subsequently the Object status) can be set, based on the adapter response code.

Tip: Oracle Identity Manager enables the Responses tab only for process task adapters. If an adapter is a task assignment, rule generator, prepopulate, or entity adapter, then Oracle Identity Manager disables this tab.

Getting Started

This chapter describes the configuration of Oracle Identity Manager and installation of Remote Manager. It contains the following topics:

- ["Overview"](#) on page 2-1

Overview

To construct adapter tasks, you need to make sure that Oracle Identity Manager has access to the target API jar files and third-party applications to which you want to connect.

When your adapter uses Java tasks, you must configure Oracle Identity Manager so that it can find the appropriate Java APIs. To do this, you need only place the .jar files that contain these APIs, into the JavaTasks sub-folder of the <XL_HOME>/Xellerate folder path (for example, C:\oracle\Xellerate\JavaTasks). Then, you will be able to access the java classes that are associated with these Java APIs and use them in the Java task you are creating.

Sometimes, instead of directly communicating with the third party system, Oracle Identity Manager must use an Oracle Identity Manager component that acts like a proxy. This component is known as Remote Manager.

The remote manager is used for:

- Invoking non-remotable APIs through Oracle Identity Manager.
- Invoking APIs that do not support SSL over secure connections.

The following procedures will show you how to:

- Configure Oracle Identity Manager to reference jar and class files (for Java tasks).
- Configure the Remote Manager.

Note: For more information about Java tasks, refer to the ["To Create a Java Task"](#) on page 3-7. To learn more about remote managers, refer to *Oracle Identity Manager Administrative and User Console Guide*.

Configure Oracle Identity Manager to Reference Jar and Class Files

1. Open the JavaTasks sub-folder, which can be found within the <XL_HOME>/Xellerate folder path (for example, C:\oracle\Xellerate\JavaTasks).

2. Place the jar file(s) into this sub-folder. Once this occurs, these files can be used to create Java tasks within an adapter, without restarting the server.

Installing Remote Manager

To configure Remote Manager, you first need to install the Remote Manager component. Refer to *Oracle Identity Manager Installation Guide* for instructions about how to install Remote Manager.

Adding the Trust Relation

To import a trusted certificate, at the command prompt, navigate to the following location:

```
<JAVA_HOME>\jre\lib\security
```

Enter the command:

```
<JAVA_HOME>/jre/bin/keytool -import -file  
<XLREMOTE_HOME>\xlremote\config\xlserver.cert -keystore cacerts -trustcacerts -  
alias rmtrust1 -storepass changeit
```

```
Owner: CN=Customer, OU=Customer, O=Customer, L=City, ST=NY, C=US  
Issuer: CN=Customer, OU=Customer, O=Customer, L=City, ST=NY, C=US  
Serial number: 41dee35a  
Valid from: Fri Jan 07 11:30:34 PST 2005 until: Mon Jan 05 11:30:34 PST 2015  
Certificate fingerprints:  
    MD5:  B0:F2:33:C8:69:E4:25:A3:CB:59:E8:51:27:EE:5C:52  
    SHA1: 3D:6A:6D:14:33:B5:5C:19:85:CC:EE:77:7F:7F:22:1D:56:48:47:4D  
Trust this certificate? [no]:  yes
```

The certificate is added to the keystore.

Creating Adapters

This chapter describes how to create and define adapters. It contains the following topics:

- ["Overview"](#) on page 3-1
- ["Defining Adapters"](#) on page 3-1
- ["Disabling Adapters"](#) on page 3-3
- ["About Adapter Variables"](#) on page 3-3
- ["Creating Adapter Tasks"](#) on page 3-5
- ["Modifying Adapter Tasks"](#) on page 3-24
- ["Changing the Order and Nesting of Tasks"](#) on page 3-24
- ["Deleting Adapter Tasks"](#) on page 3-25
- ["Working with Responses"](#) on page 3-26
- ["Scheduling Rule Generators and Entity Adapters"](#) on page 3-28

Overview

Now that you have setup Oracle Identity Manager, you are ready to create the five types of Oracle Identity Manager adapters, which are:

- Process task adapters
- Task assignment adapters
- Rule generators
- Pre-populate adapters
- Entity adapters

Note: For a detailed description of these adapters, refer to ["About Adapters"](#) on page 1-2.

Defining Adapters

1. To define an adapter: Login to Oracle Identity Manager.
2. Open the **Adapter Factory** form.

Within the Xellerate Workspace, the **Adapter Factory** form appears.

3. In the **Adapter Name** field, enter the name of the adapter (for example, *Create Solaris User*).

Note: The Adapter name can contain special characters. However, it is not recommended as there might be runtime errors.

4. Double-click the **Adapter Type** lookup field.

The Lookup window appears, displaying the five types of Oracle Identity Manager adapters which are the following:

- Process Task
 - Rule Generator
 - Pre-populate Rule Generator
 - Entity
 - Task Assignment
5. If you want to enable the adapter to automate a process task, select **Process Task (T)**.

When you want to incorporate business rules into a Oracle Identity Manager or user-defined form field, select **Rule Generator (R)**. As an example, for the **User ID** field of a form, you can configure Oracle Identity Manager to take the initial letter of the user's first name and concatenate it with the user's last name.

You can attach a type of rule generator adapter to a user-created form field, so that it can:

- Display the data, which is generated by the adapter, automatically or manually
- Use criteria that enable Oracle Identity Manager to determine which adapter will be applied to the designated form field, select **Pre-populate (P)**.

When you want to attach the adapter to a Oracle Identity Manager or user-defined form field, and have Oracle Identity Manager trigger the adapter on preinsert, preupdate, predelete, postinsert, postupdate, or postdelete, select **Entity (E)**.

To allow the adapter to automate the allocation of a process task to a user or group, select **Task Assignment (A)**.

6. Select the type of adapter you want (for example, *Process Task (T)*). Then, click **OK**.

Note: For more information about the Form Designer form, refer to *Oracle Identity Manager Administrative and User Console Guide*.

7. In the **Description** field, type a description for the adapter (for example, *This adapter is used to create a new user for the Solaris environment*).
8. From the toolbar, click **Save**.

Your adapter is now stored in the Oracle Identity Manager database.

Disabling Adapters

When you want to deactivate an adapter, so it cannot be used with a process task or form field, check the **Disable Adapter** check box, and save the adapter.

To reactivate it, uncheck the **Disable Adapter** check box again, and save the adapter.

About Adapter Variables

Once you create an adapter, you may need to map data to the parameters of the adapter's tasks for the adapter to work. For this reason, you create placeholders to map the data at runtime. These placeholders are known as adapter variables.

Note: An adapter variable can be reused for all adapter tasks.

Once an adapter variable is not needed for the adapter to run, you can remove it from the adapter. After you have deleted the adapter variable, ensure that you recompile the adapter.

The following procedures will show you how to create, modify, and delete adapter variables.

To Create an Adapter Variable

1. Select the adapter to which you wish to add an adapter variable (for example, the *Create Solaris User* adapter).
2. Click the **Variable List** tab.
3. Click **Add**.
The Add a Variable window appears.
4. When you do not want Oracle Identity Manager to be able to change the adapter variable value, after it is activated, check the **Final** check box.
5. Within the **Variable Name** field, enter the name of the adapter variable (for example, *SolarisUserID*).

Caution: The adapter variable's name cannot contain any spaces.

6. Click the **Type** combo box, and, from the drop-down menu that appears, select the classification type of the adapter variable (for example, *String*). The available choices are the following:
 - Object
 - IT Resource
 - String
 - Boolean
 - Character
 - Byte
 - Date
 - Integer

- Float
 - Long
 - Short
 - Double
7. Within the **Description** text area, you can enter explanatory information about the adapter variable.
 8. Click the **Map To** combo box. From the drop-down menu that appears, you can map your adapter variable to one of the following:

Name	Description
Literal	This adapter variable will be mapped to a constant (or literal).
Resolve at Run time	This adapter variable's mapping occurs later, at runtime. By selecting this option, the reusability of the adapter increases.
Adapter References	This adapter variable gives access to an Oracle Identity Manager database reference or an Oracle Identity Manager data object reference.
System Date	When this adapter variable is triggered by Oracle Identity Manager, it will be mapped to the current date and time of the Server.
	Note: This option appears only when you select the Date type.

Note: When you select the Object type, a Qualifier combo box appears within the Add a Variable window. From this combo box, you can select one of the following items:

Database Reference. If you select this item, then the adapter variable will be mapped to the reference of the database that the Oracle Identity Manager is currently running against.

Data Object Reference. If you select this item, then the adapter variable will be mapped to an Oracle Identity Manager data object.

Note: If you select the IT Resource type, then a Resource Type combo box appears within the Add a Variable window. From this combo box, you can select one of the IT resource types that have been created using the IT Resource Type Definition form. By doing so, you can map the adapter variable to a parameter of this IT resource type.

9. From the Add a Variable window's toolbar, click **Save**.

The information pertaining to your adapter variable is stored in the Oracle Identity Manager database.

From this window's toolbar, click **Close**.

The Add a Variable window disappears, and the main screen is active once again. The name, classification type, mapping selection, and description of the adapter variable you created appear in the child table of the **Variable List** tab.

This adapter variable now belongs to the adapter that appears in the **Adapter Factory** form. It is saved to the Oracle Identity Manager database, and the adapter variable is ready to be used.

To Modify an Adapter Variable

1. Select the adapter that contains the adapter variable you want to edit (for example, the *Create Solaris User* adapter).
2. Click the **Variable List** tab.
3. From the list, double-click the row header of the adapter variable you want to modify.

The Edit a Variable window pops up, showing information about the adapter variable.

4. Within this window, make the necessary edits (for example, changing the adapter variable's data type from *String* to *Character*).
5. From the Edit a Variable toolbar, click **Save**.

The adapter variable's modified information is stored in the Oracle Identity Manager database.

6. From the toolbar, click **Close**.

The Edit a Variable window disappears, and the main screen is active once again. The adapter variable you modified appears within the child table of the Adapter Factory form.

Note: You need to check your data mappings and re-compile the adapter, especially if you change the adapter variable's data type.

To Delete an Adapter Variable

When an adapter variable is no longer necessary for the adapter to run, you can remove it from the adapter.

1. Select the adapter that contains an adapter variable you want to remove (for example, the *Create Solaris User* adapter).
2. Click the **Variable List** tab.
3. From the list of this tab, select the adapter variable you want to delete.
4. Click **Delete**.
5. Recompile the adapter after deleting any variable.

The adapter variable disappears from the child table. This indicates that the adapter variable has been deleted.

Creating Adapter Tasks

After you construct the adapter, and create its variables, you are ready to create the atomic function calls of an adapter. These function calls are known as adapter tasks.

Oracle Identity Manager allows you to create the following adapter tasks:

- A **Java** task, which allows an adapter to communicate with an external source by invoking Java API.

- A **Remote** task, which enables an adapter to call a method on an API. This API resides on a machine that is external to Oracle Identity Manager.

This type of task is used mostly with integrations of third-party APIs that are not network-enabled. A remote manager executes the remote API method, which is located on a remote machine. In addition, if the third-party API does not use Secure Sockets Layer (SSL), then the remote manager can be used to invoke third-party APIs over SSL-protected communication. Remote tasks can also be used with integrations of third-party APIs, which are network-enabled, but are not located on the Oracle Identity Manager Server for scalability purposes. The remote API method is still executed by a remote manager. However, because the third-party API is network-enabled, the remote manager does not have to reside on the target system.

- A **Stored Procedure** task, which allows Oracle Identity Manager to map to and execute SQL programs that are located within a particular database schema. These programs are known as **stored procedures**. Stored procedures contain information, such as SQL statements, which are pre-compiled for greater efficiency.

By incorporating a stored procedure task into an adapter, and attaching this adapter to a process task, Oracle Identity Manager can incorporate stored procedures on any Oracle or SQL Server database that is accessible on its network. This includes retrieving primitive values from stored procedures.

- A **Utility** task, which enables you to populate an adapter with methods and APIs that come packaged with Oracle Identity Manager. In addition, this type of task provides you with access to the Java Standard Library APIs.
- An **Oracle Identity Manager API** task, which enables access to Oracle Identity Manager published APIs from adapter tasks. This allows for enhanced portability of adapter code.
- A **Set Variable** task, which allows you to set a variable within an adapter.
- An **Error Handler Task**, which lets you display any errors associated with an adapter that occur at runtime. In addition, you can see the reasons for the errors, along with possible solutions.
- A **Logic** task, which lets you build a conditional statement within an adapter.

You can create the following types of logic tasks:

- FOR loops
- WHILE loops
- IF statements
- ELSE statements
- ELSE IF statements
- BREAK statements
- RETURN statements
- CONTINUE statements

For more information about the types of logic tasks you can build, refer to "[To Create a Logic Task](#)" on page 3-21.

For classification purposes, Oracle Identity Manager represents each type of adapter task by an icon. The icon, which precedes the task name, is a visual indicator of the

type of task it is. For example, "J" represents a Java task, and "LT" represents a logic task.

To see a list of these icons, click the **Adapter Tasks** tab, and click **Legend**. The Legend window appears, displaying this information.

- Java Task
- Utility Task
- Xellerate API
- Remote Task
- Stored Procedure
- Logic Task
- Set Variable Task

The procedures that follow explain how to create each type of adapter.

To Create a Java Task

Oracle Identity Manager can handshake with an external source through a Java API. To make this happen, you need to add a task to an adapter, which, when triggered by Oracle Identity Manager, will initiate communications with the external source. This type of task is called a **Java task**.

The follow procedure explains how to create a Java task.

1. Select the adapter to which you wish to add a Java task (for example, the *Update Solaris Password* adapter).
2. Click the **Adapter Tasks** tab.
3. Click **Add**.

After the Adapter Task Selection window appears, select the **Functional Task** option.

4. From the display area to the right of this option, select the **Java** item, and click **Continue**.

The Object Instance Selection window appears.

The following table will help you to understand the options in the Object Instance Selection window:

Option	Description
New Object Instance	When you click this option, you are creating a new Java object instance.
Persistent Instance	You can call the method on a persistent object by clicking this option, clicking the adjacent combo box, and selecting an object instance from the drop-down menu.
Task Return Value Instance	You can call this method on an object returned by an adapter task defined earlier by clicking this option, clicking the combo box, and selecting an adapter task from the drop down list.

Note: When the Persistent Instance option is disabled, you have not defined any persistent objects for your adapter. Similarly, if the Task Return Value Instance option is disabled, none of the tasks have "Java Object" return values associated with them.

- Click an option (for example, *New Object Instance*), and click the **Continue** button.

The Add an Adapter Factory Task window appears.

The following table lists and describes the various regions of the Add an Adapter Factory Task window:

Name	Description
Task Name	This text box displays the name of the Java task.
Persistent Instance	If this Java object is to be used again, the check box is selected, and the name of the task instance is entered in the adjacent text box.
API Source	This combo box contains a list of all jar and class files to which you have access.
Application API	This combo box contains a list of all class files to which you have access, and which belong to the jar file that has been selected from the API Source combo box.
Constructors	This text area displays all the constructors, which are available for the Java object.
Methods	This text area shows a list of all the methods, which are available for the Java object.
Application Method Parameters	This area contains the parameters of the selected constructor and method. These parameters are mapped to the adapter variables and Oracle Identity Manager components.

- In the **Task Name** field, enter the name of the task you are creating (for example, *Update Password*).
- Optional.* If you want your Java object to be reusable, then select the **Persistent Instance** check box. Then, type the name of the instance of this task in the text field, located to the right of the check box.

Caution: Make sure that name of the instance does not contain any spaces.

Note: If you want to reference a session with the target resource multiple times during the life of the adapter, and not just once, check the Persistent Instance check box.

Tip: By setting the Java object to be persistent, the next time you create a Java object, it will appear in the Persistent Instance combo box of the Object Instance Selection window. In addition, you don't have to map the constructor to all adapter tasks of the same Java object.

8. Select the API Source.

The jar files appear, which Oracle Identity Manager references from the **JavaTasks** sub-folder of the <XL_HOME>/Xellerate folder path (for example, C:\oracle\Xellerate\JavaTasks).

Note: To learn how to enable Oracle Identity Manager to use third-party jar files with a Java task, refer to "[Configure Oracle Identity Manager to Reference Jar and Class Files](#)" on page 2-1.

9. Select the Application API.

The class files appear, which belong to the jar file you selected in the API Source.

10. From the **Constructors** area, select the method that will be used to initialize the Java class you selected.**11.** From the **Methods** area, select the method that will be used with your Java task.**12.** From the toolbar, click **Save**.

The information pertaining to the Java task is stored in the Oracle Identity Manager database. You can now access the parameters of your Java task's constructors and methods. These parameters appear in the **Application Method Parameters** region of the Add an Adapter Factory Task window.

13. Click the plus icons that are displayed to the left of the **Constructor** and **Method** icons.

The parameters of the Java class's constructors and methods appear, for which you need to set mappings.

14. Select the parameter of the constructor or method for which you need to set a mapping.**15.** In the **Description** text area, you can enter a description for this mapping.**16.** Click the **Map to** combo box, and select an item that you can map to the parameter of the constructor or method (for example, *Adapter Variables*).**17.** Set the appropriate mappings.

Note: For more information about which mappings to set, refer to the "[Adapter Task Mapping Information](#)" on page A-2.

18. Click **Set**.

The parameter of the selected constructor or method now appears in blue. This signifies that it has been mapped.

Tip: To remove a parameter's mapping, right-click the appropriate parameter, and select Un-Map Parameter from the popup menu that appears.

19. Repeat Steps 15-19 for all parameters of the constructors and methods that appear in the **Application Method Parameters** region.**20.** From the Add an Adapter Factory Task window's toolbar, click **Save**.

The information pertaining to the Java task is stored in the Oracle Identity Manager database.

21. From this window's toolbar, click **Close**.

The Add an Adapter Factory Task window disappears, and the main screen is active once again. The Java task that you created (for example, *Update Password*) appears within the **Adapter Factory** form.

22. *Optional*. Repeat Steps 3-22 to create additional Java tasks for the adapter.

Tip: You can create different types of adapter tasks, and add them to the adapter.

If the adapter is logically complete and all variables on the adapter tasks are mapped, you can compile it to use with a process task or form field.

23. To compile the adapter, click **Build**.

The text in the **Compile Status** field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors. You can now attach the adapter to a process task or form field.

24. *Optional*. To see the code that Oracle Identity Manager generates, from the toolbar, click **Notes**.

The Notes window pops up, containing the code that Oracle Identity Manager generated.

Note: If, after clicking **Build**, *CODE GEN ERROR* appears in the **Compile Status** field, Oracle Identity Manager encountered one of two types of errors while validating and compiling the adapter:

Validation Error

While Oracle Identity Manager is checking the adapter to verify that it is valid, an error is found. This error can result from a parameter of an adapter task not being mapped, a parameter being mapped improperly, or an adapter task being placed out of order.

Because Oracle Identity Manager will begin to generate code for an adapter only after it is validated, if Oracle Identity Manager encounters a validation error, it will not create any code.

Java Compilation Error

Oracle Identity Manager has verified that the adapter is valid. However, while Oracle Identity Manager is compiling the adapter, an error is found. This error can result from assigning an incorrect data type to an adapter task parameter.

Because Oracle Identity Manager has validated the adapter, it will begin to generate code. However, Oracle Identity Manager will stop building code at the point of the compilation where it encounters the error.

Tip: Once you create a Java task, and add this task to an adapter, you can see the following information by accessing the Resources tab of the Adapter Factory form:

1. The jar and class files used to create the Java task.
2. The name, which represents the directory path that contains these jar and class files.

To Create a Remote Task

As described earlier in *Creating Adapter Tasks*, a **remote** task enables an adapter to invoke an API method using the Remote Manager. This API resides on a machine that is external to Oracle Identity Manager. This section explains how to create a remote task.

Note: Before creating a remote task, ensure that you define an adapter variable with a classification type of IT Resource, as well as select one of the IT resources that have been created using the IT Resource Type Definition form.

1. Select the adapter to which you wish to add a remote task.
2. Click the **Adapter Tasks** tab.
3. Click **Add**.

The Adapter Task Selection window appears.

4. Select the **Functional Task** option.
5. From the display area to the right of the button, select the **Remote** item to create a remote task. Then, click **Continue**.

The Object Instance Selection window appears.

Note: To learn more about the choices of this window, refer to "[To Create a Java Task](#)" on page 3-7.

6. Click **Continue**.

The Add an Adapter Factory Task window appears.

7. In the **Task Name** field, enter the name of the remote task you are creating.
8. *Optional.* If you want your remote task to be reusable, check the **Persistent Instance** check box. Then, type the name of the instance of this task in the text field, located to the right of the check box.

Caution: Make sure that the name of the instance does not contain any spaces.

Note: For more information about the regions of the Add an Adapter Factory Task window, refer to ["To Create a Java Task"](#) on page 3-7.

To learn how to enable Oracle Identity Manager to use third-party jar files with a Java task, refer to ["Configure Oracle Identity Manager to Reference Jar and Class Files"](#) on page 2-1.

9. From the Add an Adapter Factory Task window, select a jar file, class file, constructor, and method. Then, set the mappings for the parameters of the constructor and method.

Note: One of the input parameters will have a classification type of *IT Resource*. You must associate this parameter with an adapter variable of type *IT Resource*.

Note: For more information on which mappings to select, refer to ["Adapter Task Mapping Information"](#) on page A-2.

10. From the Add an Adapter Factory Task window's toolbar, click **Save**.

The information pertaining to the remote task is stored in the Oracle Identity Manager database.

11. From this window's toolbar, click **Close**.

The Add an Adapter Factory Task window disappears, and the main screen is active once again. The remote task that you created appears within the **Adapter Factory** form.

12. *Optional.* Repeat Steps 3-11 to create additional remote tasks for the adapter.

You are now ready to compile the adapter, so it can be used with a process task or form field.

13. To compile the adapter, click **Build**.

The text in the Compile Status field changes from Recompile to OK. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors.

You can now attach the adapter to a process task or form field, so Oracle Identity Manager can communicate with the external API.

To Create a Stored Procedure Task

Through Oracle Identity Manager, you can map to and execute SQL programs that are located within a particular database schema. These SQL programs are known as **stored procedures**. Stored procedures contain information, such as SQL statements, which are pre-compiled for greater efficiency.

For this to occur, you must add a task to an adapter, which, when triggered by Oracle Identity Manager, will incorporate stored procedures on any Oracle or SQL Server database that is accessible on its network. This includes retrieving primitive values from stored procedures.

The type of task that allows this to happen is called a **stored procedure** task. The follow procedure explains how to create a stored procedure task.

Note: The parameter values and server type for the database schema are set within the IT Resources form.

The server type of the schema must be set to Database. Otherwise, Oracle Identity Manager cannot reference the database schema during the creation and/or execution of a stored procedure task.

In addition, Oracle Identity Manager uses values, which are represented by parameters (for example, Database Name or *URL*), to connect to the schema. As a result, the stored procedures, contained within the schema, can be executed.

1. Select the adapter to which you wish to add a stored procedure task (for example, the *Update User ID* adapter).
2. Click the **Adapter Tasks** tab.
3. Click **Add**.

The Adapter Task Selection window appears.

4. Select the **Functional Task** option.
5. From the display area to the right of the option, select **Stored Procedure**, and click **Continue**.

The Add an Adapter Factory Task window appears.

The following table lists and describes the regions of the Add an Adapter Factory Task window:

Name	Description
Task Name	This text box displays the name of the stored procedure task.
Description	This text area displays explanatory information about the stored procedure task.
Database	This combo box contains the list of databases, which are defined in the IT Resources form. Important: Only those IT resources with a Server Type of Database will appear in the Database combo box.
Schema	This combo box lists the schemas, which are associated with the database that appears in the Database combo box.
Procedure	This combo box lists the stored procedures, which reside within the database schema that is displayed in the Schema combo box.

Name	Description
Connection Status	<p>This region displays the status of the connection between Oracle Identity Manager and the database that contains the target stored procedure.</p> <p>When Oracle Identity Manager can connect to the database, Connection Established is displayed in the Connection Status region.</p> <p>Note: If Oracle Identity Manager cannot connect, Connection Failed appears in the display area. In addition, the Notes button of the Add an Adapter Factory Task window is enabled. By clicking this button, you can see why a connection could not be established.</p> <pre data-bbox="696 562 1268 667"> Exception Type: java.lang.ClassNotFoundException: com.sybase.jdbc2.jdbc.SybDriver </pre> <p>In the preceding example, Oracle Identity Manager could not connect to the designated database because it could not find a particular Java class.</p>
Parameters	<p>This area contains parameters that can be mapped to the stored procedure. These parameters appear after you select a database, schema, and stored procedure, and save this information to the Oracle Identity Manager database.</p>

6. In the **Task Name** field, enter the name of the stored procedure task you are creating (for example, *Update ID*).
7. In the **Description** text area, you can enter a description for this stored procedure task.
8. Click the **Database** combo box.

The databases appear, which are defined in the IT Resources form.

Note: If Oracle Identity Manager cannot connect to the database you selected, *Connection Failed* appears in the display area. In addition, the Notes button of the Add an Adapter Factory Task window is enabled. By clicking this button, you can see why a connection could not be established..

Tip: Schemas and stored procedures appear only after you select a database to which Oracle Identity Manager can connect. Based on this selection, related schemas and stored procedures appear in the corresponding combo boxes.

Note: For more information on using the IT Resources form to define the settings for databases that contain stored procedures, refer to *Oracle Identity Manager Administrative and User Console Guide*.

9. Click the **Schema** combo box.

The schemas appear, which are associated with the database you selected.

10. Click the **Procedure** combo box.

The stored procedures are displayed, which reside within the database schema that you selected from the **Schema** combo box.

11. From the Add an Adapter Factory Task window's toolbar, click **Save**.

The information pertaining to the stored procedure task is saved into the Oracle Identity Manager database. You can now set the mappings for the parameter(s) of the stored procedure. These parameters appear in the **Parameters** region of the Add an Adapter Factory Task window.

Note: Oracle Identity Manager automatically maps the database and schema of the selected stored procedure. However, if you need to override these mappings, Oracle Identity Manager enables you to do so.

Note: For more information on which mappings to select, refer to "[Adapter Task Mapping Information](#)" on page A-2.

12. From the Add an Adapter Factory Task window's toolbar, click **Save**.

The mappings that you have set for the parameter(s) of the stored procedure task are stored in the Oracle Identity Manager database.

13. From this window's toolbar, click **Close**.

The Add an Adapter Factory Task window disappears, and the main screen is active once again. The stored procedure task you created (for example, *Update ID*) appears within the Adapter Factory form.

14. *Optional.* Repeat Steps 3-13 to create **additional stored procedure tasks** for the adapter.

You are now ready to compile the adapter, so it can be used with a process task or form field.

15. To compile the adapter, click **Build**.

The text in the **Compile Status** field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors. You can now attach the adapter to a process task or form field, so Oracle Identity Manager can map to and execute the stored procedure you selected.

To Create a Utility Task

The Adapter Factory is shipped with a library of utility classes and methods, which increase the efficiency of developing adapters.

These utility classes and methods are contained within the **xlUtils.jar**, **xlIntegration.jar**, and **rt.jar** files. A Java task you create by using a class or method from one of these jar files is called a **utility task**.

Note: To see a list of the class files that comprise the xlUtils.jar, xlIntegration.jar, and rt.jar files, along with descriptions of the functions they perform, refer to "[Class Files](#)" on page A-1.

1. Select the adapter to which you wish to add a utility task (for example, the *Update Solaris User Group* adapter).
2. Click the **Adapter Tasks** tab.
3. Click **Add**.
The Adapter Task Selection window appears.
4. Select the **Utility Task** option.
5. From the display area to the right of the option, select **Utility**, and click **Continue**.
The Object Instance Selection window appears.

Note: To learn more about the choices of this window, refer to "[To Create a Java Task](#)" on page 3-7.

6. Click **Continue**. The Add an Adapter Factory Task window appears
7. In the **Task Name** field, enter the name of the utility task you are creating (for example, *Update User Group*).
8. *Optional.* If you want your utility task to be reusable, click the **Persistent Instance** check box. Then, type the name of the instance of this task in the text field, located to the right of the check box.

Caution: Make sure that name of the instance does not contain any spaces.

Note: For more information on the regions of the Add an Adapter Factory Task window, refer to "[To Create a Java Task](#)" on page 3-7.

To learn how to enable Oracle Identity Manager to use third-party jar files with a Java task, refer to "[Configure Oracle Identity Manager to Reference Jar and Class Files](#)" on page 2-1.

9. Click the **Application API** combo box.
The class files appear, which belong to the xlUtils.jar, xlIntegration.jar, and rt.jar files.

Note: The xlUtils.jar, xlIntegration.jar, and rt.jar files contain all of the class files that you can use for a utility task. Therefore, you do not need to access the API Source combo box.

10. From the Add an Adapter Factory Task window, select a constructor and method. Then, set the mappings for the parameters of the constructor and method.

Note: For more information on which mappings to select, refer to "[Adapter Task Mapping Information](#)" on page A-2.

11. From the Add an Adapter Factory Task window's toolbar, click **Save**.
The information pertaining to the utility task is stored in the Oracle Identity Manager database.
12. From this window's toolbar, click **Close**.
The Add an Adapter Factory Task window disappears, and the main screen is active once again. The utility task that you created (for example, *Update User Group*) appears within the Adapter Factory form.
13. *Optional*. Repeat Steps 3-12 to create additional utility tasks for the adapter.
You are now ready to compile the adapter, so it can be used with a process task or form field.
14. To compile the adapter, click **Build**.
The text in the Compile Status field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors. You can now attach the adapter to a process task or form field.

To Create an Oracle Identity Manager API Task

The "Oracle Identity Manager API task" allows Adapter tasks to call Oracle Identity Manager's published APIs for greater portability of the adapter code. This is better than accessing Oracle Identity Manager data directly through hardcoded SQL statements. This type of task is called an Oracle Identity Manager API task.

The Adapter Factory is shipped with a library of utility classes and methods, which increase the efficiency of developing adapters that contain Oracle Identity Manager API tasks. These utility classes and methods are contained within the xlAPI.jar file.

Note: To see a list of the class files that comprise the xlAPI.jar file, along with descriptions of the functions they perform, refer to "[Class Files](#)" on page A-1.

Now that you understand Oracle Identity Manager API tasks, you will learn how to create this type of adapter task.

1. Select the adapter to which you wish to add an Oracle Identity Manager API task (for example, the *Get User's Password* adapter).
2. Click the **Adapter Tasks** tab.
3. Click **Add**.
The Adapter Task Selection window appears.
4. Select the **Utility Task** option.
5. From the display area to the right of the option, select **Xellerate API**, and click **Continue**.

The Object Instance Selection window appears.

Note: To learn more about this window, refer to "[To Create a Java Task](#)" on page 3-7.

6. Click **Continue**.

The Add an Adapter Factory Task window appears.

7. In the Task Name field, enter the name of the Oracle Identity Manager API task you are creating (for example, *Retrieve Password*).
8. *Optional.* If you want your Oracle Identity Manager API task to be reusable, click the Persistent Instance check box. Then, type the name of the instance of this task in the text field, located to the right of the check box.

Tip: Ensure that name of the instance does not contain any spaces.

Note: For more information on the regions of the Add an Adapter Factory Task window, refer to ["To Create a Java Task"](#) on page 3-7.

To learn how to enable Oracle Identity Manager to use third-party jar files with a Java task, refer to ["Configure Oracle Identity Manager to Reference Jar and Class Files"](#) on page 2-1.

9. Click the **Application API** combo box.

The class files appear, which belong to the **xlAPI.jar** file.

Note: The xlAPI.jar file contains all of the class files that you can use for an Xellerate API task. Therefore, you do not need to access the API Source combo box.

10. From the Add an Adapter Factory Task window, select a class file, constructor, and method. Then, set the mappings for the parameters of the constructor and method.

Note: For more information on which mappings to select, refer to ["Adapter Task Mapping Information"](#) on page A-2.

11. From the Add an Adapter Factory Task window's toolbar, click **Save**.

The information pertaining to the Oracle Identity Manager API task is stored in the Oracle Identity Manager database.

12. From this window's toolbar, click **Close**.

The Add an Adapter Factory Task window disappears, and the main screen is active once again. The Oracle Identity Manager API task that you created (for example, *Retrieve Password*) appears within the **Adapter Factory** form.

13. *Optional.* Repeat Steps 3-12 to create additional Oracle Identity Manager API tasks for the adapter.

You are now ready to compile the adapter, so it can be used with a process task or form field.

14. To compile the adapter, click **Build**.

The text in the **Compile Status** field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors.

You can now attach the adapter to a process task or form field, so Oracle Identity Manager can communicate with a third-party application.

To Create a Set Variable Task

Sometimes, for an adapter to accomplish its required objective, you need to reassign the value of an adapter variable to another adapter variable, a different type of adapter task, or a constant (or literal). The task that enables you to reallocate an adapter variable's value is known as a **set variable** task.

Note: For more information on adapter variables, refer to "[About Adapter Variables](#)" on page 3-3.

For example, you can create a set variable task that will set the adapter variable's return value to equal the output of an adapter task (*UserName*) if the User ID's length is fewer than 11 characters.

1. Select the adapter to which you wish to add a set variable task (for example, the *Check the Solaris User ID adapter*).
2. Click the **Adapter Tasks** tab.
3. Click **Add**.

The Adapter Task Selection window appears.

4. Select the **Logic Task** option.
5. From the display area, select **SET VARIABLE**, and click **Continue**.

The Add Set Variable Task Parameters window appears.

6. From the **Variable Name** combo box, select the adapter variable that has a value you want to reassign (for example, *Adapter return value*).
7. From the **Operand Type** combo box, select the type of operand that will provide the value for the variable.

Tip: You can reassign an adapter variable's value to another adapter variable, a different type of adapter task, or a literal.

Use the following table to understand the various types of operands:

Operand Name	Description
Variable	<p>If you select this operand type, adapter variables appear in the Operand Qualifier combo box. From this combo box, select the specific adapter variable that will provide the reassigned value.</p> <p>Note: The only adapter variables that will appear in the Operand Qualifier combo box will be those variables that have the same data type as the adapter variable that is displayed within the Variable Name combo box.</p>

Operand Name	Description
Adapter Task	<p>By selecting this operand type, adapter tasks are displayed in the Operand Qualifier combo box. From this combo box, select the particular adapter task that will provide the reallocated value.</p> <p>Note: The only adapter tasks that will appear in the Operand Qualifier combo box will be those tasks that have the same data type as the adapter variable that is displayed within the Variable Name combo box.</p>
Literal	<p>When you select this operand type, types of literals appear in the Operand Qualifier combo box. From this combo box, select the type of literal that will provide the reallocated value. Then, type the specific literal into the text box that appears underneath the combo box.</p>

The following task sets the adapter variable's return value to be equal to the *UserName* adapter variable.

- From the toolbar in the Add Set Variable Task Parameters window click **Save**.
The set variable task you created is stored in the Oracle Identity Manager database.
- From the Add Set Variable Task Parameters window toolbar, click **Close**.
The Add Set Variable Task Parameters window disappears, and the main screen is active once again. The set variable task that you created (for example, *Set Adapter return value = UserName*) appears within the Adapter Factory form.
- Optional.* Repeat Steps 3-9 to create additional set variable tasks for the adapter.
You are now ready to compile the adapter, so it can be used with a process task or form field.
- To compile the adapter, click **Build**.
The text in the Compile Status field changes from *Recompile* to *OK*. Oracle Identity Manager compiled the adapter and found no errors. You can attach the adapter to a process task or form field.

To Add a Error Handler Task

- An adapter task can return errors. When this occurs, the process task or form field to which the adapter is attached gets rejected.
You can attach your own customizable error messages, which will be displayed to the user. These messages are known as error handler tasks.
For example, you can attach an error handler task to an adapter that will display an error message when the length of a User ID is greater than 10 characters.
- Select the adapter to which you wish to add an error handler task (for example, the *Check the Solaris User ID* adapter).
- Click the **Adapter Tasks** tab.
- Click **Add**.
The Adapter Task Selection window appears.
- Select the **Logic Task** option.
- From the display area, select **Handle Error**, and click **Continue**.

The Add an Adapter Factory Task window appears.

7. Double-click this window's Lookup field.

The Lookup window appears, displaying the error handler tasks you can add to the adapter.

Note: The only error handler tasks that appear in this Lookup window are the ones that begin with ADAPTER (such as *ADAPTER.USERIDLENERR*).

Note: If you do not see the error handler task that you want to incorporate into the adapter, you can create one by accessing the Error Message Definition form.

For more information on the Error Message Definition form, refer to *Oracle Identity Manager Administrative and User Console Guide*.

8. Select the error handler task you want (for example, *ADAPTER.USERIDLENERR*).
9. Click **OK**.

The Lookup window disappears, and the Add an Adapter Factory Task window is active. In addition, the error handler task you selected appears in the text box of this window.

10. From the Add an Adapter Factory Task window's toolbar, click **Save**.

The error handler task you incorporated into the adapter is stored in the Oracle Identity Manager database.

11. From this window's toolbar, click **Close**.

The Add an Adapter Factory Task window disappears, and the main screen is active once again. The error handler task you added (for example, *Handle Error.ADAPTER.USERIDLENERR*) appears within the child table of the **Adapter Factory** form.

12. *Optional.* Repeat Steps 3-10 to create additional error handler tasks for the adapter.

If the adapter is logically complete and all variables on the adapter tasks are mapped, you can compile it to use with a process task or form field.

13. To compile the adapter, click **Build**.

The text in the Compile Status field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors. You can now attach the adapter to a process task or form field.

To Create a Logic Task

While defining the adapter, you may need to add conditional statements to the adapter to control its logic flow. These conditional statements are known as logic tasks.

For example, you can create a logic task that will trigger an action if the length of a User ID is greater than 10 characters.

1. Select the adapter to which you wish to add a logic task (for example, the *Check the Solaris User ID* adapter).

2. Click the **Adapter Tasks** tab.
3. Click **Add**.
The Adapter Task Selection window appears.
4. Select the **Logic Task** option.
5. From the display area, select the type of logic task you want to create. Then, click **Continue**.

Note: If you select a conditional expression, and click Continue, one of the following actions occurs:

Oracle Identity Manager adds the conditional statement to the adapter directly; or

A secondary window appears, containing fields pertaining to the conditional expression, which you need to configure.

To see what happens when you select a particular conditional statement, refer to the following table.

Conditional Statement	Statement Is Added to the Adapter Directly	Secondary Window Appears
FOR		X
WHILE		X
IF		X
ELSE	X	
ELSE IF		X
BREAK	X	
RETURN	X	
CONTINUE	X	

When you select the **IF** conditional statement, for example, the following secondary window appears:

The following table will help you understand the various regions of the Add Adapter Factory Logic Task Parameters window:

Name	Description
Operand Type	These combo boxes contain types of operands, such as adapter tasks and adapter variables.
Comparator Combo Box	From this combo box, you can set the relationship between two operands (for example, <, =, >).
Operand Qualifier	These combo boxes contain the qualifiers for the operands.
Literal Text Box	When you select the Literal operand type, enter the specific literal into this text box.

Note: By selecting the FOR conditional statement, an Add Adapter Factory Logic Task Parameters window is displayed. However, it contains different text and combo boxes.

For this conditional expression, use the following table to understand the various regions of this Add Adapter Factory Logic Task Parameters window:

Name	Description
Operand Type	These combo boxes contain types of operands, such as adapter tasks and adapter variables.
Comparator Combo Box	From this combo box, you can set the relationship between two operands (for example, <, =, >).
Operand Qualifier	These combo boxes contain the qualifiers for the operands.
Increment Combo Box	Within this area, you can set whether the initial value will increase or decrease, and by how much.

Note: If you select the ELSE, BREAK, RETURN, or CONTINUE conditional expressions, proceed to Step 8.

Set the parameters for your conditional expression.

This logic task will check to see if the length of the User ID is greater than 10 characters.

- From the Add Adapter Factory Logic Task Parameters window's toolbar, click **Save**.

The logic task you created is stored in the Oracle Identity Manager database.

- From this window's toolbar, click **Close**.

The Add Adapter Factory Logic Task Parameters window disappears, and the main screen is active once again. The logic task that you created [for example, *If (Check ID Length > 10)*] appears within the Adapter Factory form.

- Optional.* Repeat Steps 3-8 to create additional logic tasks for the adapter.

Caution: All adapter tasks that need to be executed for a condition of a logic task should be nested properly under that logic task.

Note: For more information on nesting tasks, refer to "[Changing the Order and Nesting of Tasks](#)" on page 3-24.

You are now ready to compile the adapter, so it can be used with a process task or form field.

- To compile the adapter, click **Build**.

The text in the Compile Status field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors. You can now attach the adapter to a process task or form field.

Modifying Adapter Tasks

The following procedure will show you how to edit an adapter task, in case you need to make changes to it. To modify an adapter task

1. Select the adapter that contains the adapter task you wish to edit (for example, the *Update Solaris User Group* adapter).
2. Click the **Adapter Tasks** tab.
3. Double-click the adapter task that you want to modify.

The Edit Adapter Factory Task Parameters window appears, displaying information that relates to the adapter task you selected. Within this window, make the necessary modifications.

4. From the Edit Adapter Factory Task Parameters window's toolbar, click **Save**.

The information you modified is stored in the Oracle Identity Manager database.

5. From this window's toolbar, click **Close**.

The Edit Adapter Factory Task Parameters window disappears. The main screen is active again. The modified task appears within the child table of the **Adapter Factory** form. You must re-compile the adapter, so it can be used with a process task or form field.

6. To re-compile the adapter, click **Build**.

The text in the **Compile Status** field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors. You can now attach the adapter to a process task or form field.

Caution: You cannot modify the API call inside a Java, Xellerate API, or Utility task. The adapter task has to be deleted and re-created.

In addition, if *CODE GEN ERROR* appears in the Compile Status field, Oracle Identity Manager encountered errors while compiling the adapter. Rectify the errors, if necessary re-do the adapter task modifications, and compile the adapter again.

Changing the Order and Nesting of Tasks

If you add multiple tasks to an adapter, you may need to either change the order in which the tasks are executed, or place one task inside of another task for the adapter to work.

The following procedure will show you how to change the order and nesting of tasks.

Caution: You should not change the order and nesting of adapter tasks unless you understand the mapping dependencies of the adapter tasks.

To change the order and nesting of tasks:

1. Select the adapter that contains tasks of which you want to change the order and/or nest (for example, the *Check the Solaris User ID* adapter).
2. Click the **Adapter Tasks** tab.

The tasks appear, which belong to the current adapter.

In this example, the following changes must occur:

- The error handler task must be nested inside of the *IF (Check ID Length > 10)* logic task.
- The set variable task has to be nested inside of the **ELSE** logic task.
- The **IF** logic task needs to precede the **ELSE** logic task.

Therefore, you must first reorganize the logic tasks. Then, you must nest the error handler task and set variable task inside of the **IF** and **ELSE** logic tasks, respectively. To reorganize tasks:

3. Select the task that needs to run before another task, and click the **Up** arrow button. The selected task will switch places with the task that precedes it.

or

Select the task that must be executed after another task, and click the **Down** arrow button. The highlighted task will now appear below the task that previously followed it.

To nest tasks/remove task nestings:

4. Select the task that needs to be placed inside of another task, and click the **Right** arrow button. The selected task will be nested inside of the task that appears above it.

or

Select the task that no longer needs to be nested inside of another task, and click the **Left** arrow button. The highlighted task will not be nested inside of the task that is displayed above it.

5. From the toolbar, click **Save**.

The order and nesting of the adapter's tasks is stored in the Oracle Identity Manager database. If the adapter is logically complete and all variables on the adapter tasks are mapped, you can compile it to use with a process task or form field.

6. To compile the adapter, click **Build**.

The text in the Compile Status field changes from *Recompile* to *OK*. This indicates that Oracle Identity Manager compiled the adapter and did not find any errors. You can now attach the adapter to a process task or form field.

Caution: If you see *CODE GEN ERROR* in the Compile Status field, Oracle Identity Manager found errors while compiling the adapter. Rectify the errors, if necessary re-do the adapter task modifications, and compile the adapter again.

Deleting Adapter Tasks

When an adapter task is no longer necessary for the adapter to run, you must remove it from the adapter. To delete an adapter task:

1. Select the adapter that contains the task you wish to remove (for example, the *Update Solaris User Group* adapter).
2. Click the **Adapter Tasks** tab.
3. Select the task that you want to remove (for example, the *CONTINUE* logic task).
4. Click **Delete**.

The selected task is deleted and disappears from the child table.

5. From the toolbar, click **Save**.
6. Recompile the adapter.

Caution: While deleting adapter tasks, ensure that the logic of the adapter is consistent and maintained.

Working with Responses

Adapters can have different outcomes, called **responses**. Based on these responses, adapters can trigger other process tasks.

For example, if the adapter returns a *True* response, the process task's status can be set automatically to *Completed*. However, if the adapter returns a *False* response, the process task's status can be set automatically to *Rejected*, and another process task can be triggered.

These responses can be added, modified, or removed on the **Responses** tab of the Adapter Factory form.

The following procedures will show you how to create, modify, and delete responses.

Tip: Responses are used only with process task adapters, because these adapters are attached to process tasks. Rule generators, pre-populate adapters, and entity adapters are not connected to processes. In addition, task assignment adapters are not associated with responses. Therefore, if the active adapter is a task assignment adapter, rule generator, pre-populate adapter, or entity adapter, Oracle Identity Manager disables the Responses tab.

To Create a Response

1. Select the adapter to which you want to add responses (for example, the *Create Solaris User* adapter).
2. Click the **Responses** tab.
3. Click **Add**.
An empty row is inserted into the **Responses** tab, as shown below:
4. Click the field that appears within the **Code Name** column.
5. Enter a code, which represents a response type that can be generated (for example, *True*).
6. Click the field that appears within the **Description** column.
7. Enter a description for this response (for example, *The user was created successfully*).
8. Double-click the field that appears within the **Status** column.

The Lookup popup window appears, containing the different status levels that you can associate with the response.

Note: For more information on Oracle Identity Manager's status levels, refer to [Chapter 4, "About Process Task Adapters"](#) on page 4-1.

9. Click the desired status level (for example, *Completed (C)*). Then, click **OK**.

The Lookup window disappears, and the **Responses** tab is active once again.

10. Create another response, by clicking the **Add** button, and entering *False* and *The user was not created successfully.* into the Code Name and Description fields, respectively. Then, access the Lookup window, and assign the *Rejected (R)* status level to this response.
11. From the toolbar, click **Save**.

The responses that you created for this adapter have been stored in the Oracle Identity Manager database. Once you attach this adapter to a process task, these responses will appear in the Responses tab of the Editing Task window of the Process Definition form.

To Modify a Response

The following procedure demonstrates how to edit a response.

1. Select the adapter that contains the response you want to edit (for example, the *Create Solaris User* adapter).
2. Click the **Responses** tab.
The **Responses** tab appears.
3. Double-click the field of the response, which contains information that you want to modify.
 - a. If the field is a text field, Oracle Identity Manager enables it. You can now edit the contents within this field.
 - b. When the field is a Lookup field, the Lookup popup window appears, containing the different status levels that you can associate with the response. Click the desired **status level**, then click **OK**.

For example, double-click the **Status** column of the *False* response, select the *Suspended (S)* status level, and click **OK**.

4. From the toolbar, click **Save**.

The information that you modified for the response is stored in the Oracle Identity Manager database.

To Delete a Response

When a response is no longer necessary, you can delete it from the adapter.

1. Select the adapter, which contains a response that you want to remove.
2. Click the **Responses** tab.
3. Select the response that you want to delete.
4. Click **Delete**.

The response disappears. This indicates that Oracle Identity Manager has deleted the response.

Scheduling Rule Generators and Entity Adapters

Oracle Identity Manager triggers a process task adapter or a task assignment adapter automatically if it is attached to a process task, and the process task's status is *Pending*. In addition, Oracle Identity Manager always triggers pre-populate adapters on pre-insert. Therefore, you do not need to schedule when process task adapters, task assignment adapters, or pre-populate adapters will be executed.

On the other hand, a rule generator and an entity adapter are attached to a form field. The only way that Oracle Identity Manager will be able to execute the rule generator or entity adapter is for you to specify when it will be triggered. You do this through the Execution Schedule tab.

Using this tab, you can determine that Oracle Identity Manager will trigger the rule generator or entity adapter on pre-insert or pre-update. In addition, you can also schedule an entity adapter to be executed on pre-delete, post-insert, post-update, and post-delete.

This procedure demonstrates how to configure Oracle Identity Manager to trigger a rule generator or entity adapter.

To Schedule Rule Generators and Entity Adapters

1. Select the rule generator or entity adapter that you want Oracle Identity Manager to trigger (for example, *Solaris User ID Generator*).

Note: When you work with process task adapters or pre-populate adapters, you do not need to use the Execution Schedule tab. As a result, this tab, as well as its contents, are disabled.

2. Click the Execution Schedule tab.

The contents of the Execution Schedule tab appear.

The following table will help you understand the various check boxes of the **Execution Schedule** tab:

Name	Description
Pre-Insert	By clicking this check box, Oracle Identity Manager can trigger the rule generator or entity adapter before the record is inserted into the database.
Pre-Update	When you click this check box, Oracle Identity Manager can trigger the rule generator or entity adapter before the record is updated in the database.
Pre-Delete	By clicking this check box, Oracle Identity Manager can trigger the entity adapter before the record is deleted from the database.
Post-Insert	When you click this check box, Oracle Identity Manager can trigger the entity adapter after the record is inserted into the database.

Name	Description
Post-Update	By clicking this check box, Oracle Identity Manager can trigger the entity adapter after the record is updated in the database.
Post-Delete	When you click this check box, Oracle Identity Manager can trigger the entity adapter after the record is deleted from the database.

Note: By clicking the check boxes of the Execution Schedule tab, you are defining the times when Oracle Identity Manager *can* trigger the rule generator or entity adapter. The Data Object Manager form allows you to specify when Oracle Identity Manager *will* trigger the rule generator or entity adapter.

For more information on the Data Object Manager form, refer to ["Mapping Rule Generator Adapter Variables"](#) on page 6-2.

3. Enable the desired check box(es). Then, from the toolbar, click **Save**.

The criteria you set for Oracle Identity Manager to execute the rule generator or entity adapter is stored in the Oracle Identity Manager database.

About Process Task Adapters

This chapter describes the process task adapters. It contains the following topics:

- ["Background"](#) on page 4-1
- ["How a Process Task Adapter Works"](#) on page 4-2
- ["Attaching Process Task Adapters to Process Tasks"](#) on page 4-3
- ["Removing Process Task Adapters from Process Tasks"](#) on page 4-6

Background

A process task adapter is java code created through the adapter factory. It enables the Oracle Identity Manager to automatically execute process tasks in provisioning processes.

Each process and process task has a status, which indicates the stage of its completion. The statuses for a process or process task appear in the table below, in order of importance:

Task Status	Description
X	This process/process task has been stopped. Its status cannot change anymore
S	Suspended: This process/process task has been put on hold temporarily.
R	Rejected: This process/process task has not been completed successfully or has not been approved. The status of rejected process tasks can only be changed to <i>Cancelled</i> or <i>Unsuccessfully Completed</i> .
P	Pending: This process/process task is in the progress of being completed. All preceding tasks and processes, respectively, have been completed.
PX	Pending Cancellation: This process task will be cancelled, but this task has to be completed first before it will be cancelled.
C	Completed: This process/process task has been finished successfully.
MC	Manually Completed: This process task has been completed successfully by an Oracle Identity Manager user (i.e., manually).
UC	Unsuccessfully Completed: This process task has been set to <i>Completed</i> . However, it had been rejected before.

Task Status	Description
W	Waiting: This process/process task cannot be completed until all preceding process tasks or processes are completed.

The status level of a process represents the most important status level of its process tasks, which must be completed for the process to be completed. Suppose a process has three process tasks, each process task has a different status level (*Completed*, *Waiting*, and *Rejected*), and all three process tasks need to be completed for the process to be completed. Since the highest task status level is *Rejected*, the status level of the process is also *Rejected*.

A process task can be managed in one of the following ways:

- It can be handled manually, using the Object Process Console tab of the Organizations or Users forms, or the Oracle Identity Manager Web Application.
- An Oracle Identity Manager process can be configured so that one (or more) of its tasks is triggered automatically once it achieves a status of *Pending*.

The Java code that enables Oracle Identity Manager to automatically execute a process task is referred to as a process task adapter. The Oracle Identity Manager tool that allows developers to create and manage process task adapters is known as the Adapter Factory.

Note: For more information on handling process tasks manually, refer to the *Oracle Identity Manager Administrative and User Console Guide*.

How a Process Task Adapter Works

Once you create a process task adapter, you attach it to the appropriate process task using the Integration tab of the Process Definition form. From this tab, you can also map any variables of the adapter to their proper locations, which were designated as either *Resolve at Runtime* or as an adapter return variable.

For the example below, the adapter named *adpSOLARISPASSWORDUPDATED* is connected to the *Password Updated* task of the *Solaris* process.

After you attach an adapter to a process task, for the adapter to be functional, it may need data from fields of other forms. For this example, the *adpSOLARISPASSWORDUPDATED* adapter cannot work unless it obtains the following information:

- The user's Oracle Identity Manager ID and password.
- The user's Solaris ID and password.
- The IP address where Solaris is located.

Therefore, it must get this information from the *UserID*, *Passwd*, *SolarisUserID*, *SolarisUserPasswd*, and *ServerAddress* adapter variables, respectively. These five variables are created using the Adapter Factory form. The "Y" that precedes each adapter variable signifies that it has been mapped correctly.

The form that enables you to create process-specific fields, which will be used by a process to obtain the information it needs, is called the Form Designer. When you create these fields, Oracle Identity Manager stores them into a table. Then, by associating this table with a process (through the Table Name lookup field of the

Process Definition form), the adapter, which you attach to a task of this process, will use the table to retrieve the appropriate data.

If you need to modify this table, you can do so through the Form Designer form.

The `UD_SOLARIS` table contains two fields: `UD_SOLARIS_USERID` and `UD_SOLARIS_PASSWD`. By accessing this record of the Form Designer form, you can edit the fields of the table.

Note: For more information on using the Form Designer form to create and modify process-specific fields, refer to the *Oracle Identity Manager Administrative and User Console Guide*.

Once you attach the process task adapter to a dependent process task, and the status of this process task is *Pending* (the status of the previous process task is *Completed*), Oracle Identity Manager will trigger the adapter automatically. When the process task is an independent task, Oracle Identity Manager will execute the adapter as soon as the process is requested.

The result of the adapter being executed represents the state of the process task. When the adapter is finished successfully, the process task to which this adapter is attached will have a status of *Completed*.

On the other hand, if the adapter cannot perform its designated function, the process task to which this adapter is attached will have a status of *Rejected*. By discovering the cause of the error, you can modify the process task and/or adapter so it can run successfully.

Note: To determine why a process task may have failed:

Find the process task. When the process task has not yet been provisioned to the target user or organization, it is located in the To Do List or Pending Approvals. To find the task:

1. Log in as the user.
 2. Select the To Do List link or the Pending Approvals links in the left side of the window.
-
-

Attaching Process Task Adapters to Process Tasks

In the previous chapter, you learned how to create a process task adapter. You must attach it to a process task to execute that process task automatically.

To connect an adapter to a process task, access the Integration tab (from the Process Definition form). From this tab, you can also map any adapter variables to their proper locations.

The following procedure shows you how to attach a process task adapter to a process task:

1. Open the Process Definition form, which is located in the Process Management folder.

Within the Oracle Identity Manager Workspace, the Process Definition form appears.

2. Select the process, which contains a task to which you want to attach an adapter. The selected process, along with its tasks, appears in the Process Definition form. For this example, the Solaris process has been selected.
3. Double-click the row header of the task to which you want to attach an adapter. The Editing Task window appears, containing information about the task (for example, the *Password Updated* process task).
4. Click the **Integration** tab.
The **Integration** tab appears.
5. From this tab, click **Add**.
The Handler Selection window appears.
6. To access Oracle Identity Manager's adapters, click the **Adapter** option.
The adapters appear, which you can attach to the process task.
7. From this region, select the adapter that you want to attach to the process task (for example, the *adpSOLARISPASSWORDUPDATED* adapter).

Tip: For classification purposes, the first three letters of each adapter's name are *adp*. For classification purposes, the first three letters of each adapter's name are *adp*.

8. From the Handler Selection window's toolbar, click **Save**.
A dialog box appears, stating that the adapter was successfully added to the process task.
9. Click **OK**.
The dialog box disappears, and the **Integration** tab is now active. This tab now displays the following:
 - The name of the adapter that is attached to the process task;
 - The status of the adapter; and
 - The names, descriptions, and mapping statuses of the adapter's variables.

Note: An adapter can have one of three mapping statuses:

Ready. This adapter has been successfully compiled, and all of its variables have been mapped correctly.

Mapping Incomplete. This adapter has been successfully compiled, but at least one of its variables have been not mapped correctly.

Adapter Unavailable. After this adapter had been compiled successfully, it was modified, and now needs to be recompiled.

Note: If an adapter does not have any mappable variables, the Adapter Variables region is empty. In addition, the Status field will display either *Ready* or *Adapter Unavailable*, depending on whether the adapter needs to be recompiled.

Note: A mappable adapter variable either has been designated as *Resolve at Runtime* or it is an adapter return variable.

Note: Once you attach the adapter to the process task, any responses that you defined for the adapter appear in the Responses tab of the Editing Task window.

10. Set the mappings for each variable that appears in the Adapter Variables region of the Integration tab. To do so, double-click the row header of the variable you want to map (for example, *SolarisUserID*).

The Data Mapping for Variable window appears.

The following table will help you understand the various fields of the Data Mapping for Variable window:

Field Name	Description
Variable Name	This field displays the name of the adapter variable for which you are setting a mapping (for example, <i>SolarisUserID</i>).
Data Type	This field shows the data type of the adapter variable (for example, <i>String</i> is the data type for the <i>SolarisUserID</i> variable).
Map To	This field contains the types of mappings that you can set for the adapter variable (for example, <i>IT Resources</i>). When you map the adapter variable to a location or a contact, Oracle Identity Manager enables the adjacent combo box. From this combo box, select the specific type of location or contact to which you are mapping the adapter variable. In addition, if you map the adapter variable to a custom process form, and this form contains child table(s), Oracle Identity Manager enables the adjacent combo box. From this combo box, select the child table to which you are mapping the adapter variable. If you are not mapping the adapter variable to a location, contact, or child table of a custom process form, this combo box is disabled.
Qualifier	This field contains the qualifiers for the mapping you selected in the Map to combo box (for example, <i>IT Asset</i>).
IT Asset Type	This field enables you to select a specific IT Resource (for example, <i>Solaris</i>) when you map an adapter variable to an IT Resource, and this variable's data type is <i>String</i> . If you are not mapping the adapter variable to an IT Resource, or the variable's data type is not <i>String</i> , this field does not appear.
IT Asset Property	This field enables you to select a specific field that will receive the results of the mapping (for example, <i>User Name</i>), when you map an adapter variable to an IT Resource, and this variable's data type is <i>String</i> . If you are not mapping the adapter variable to an IT Resource, or the variable's data type is not <i>String</i> , this field does not appear. Important: The IT Asset Type and IT Asset Property fields are included within this window for backward compatibility. The preferred way is to create an adapter variable with a data type of <i>IT Resource</i> , in which case these fields will not appear.

Field Name	Description
Literal Value	<p>When you map the adapter variable to a literal, use this field to specify the specific literal value.</p> <p>If you are not mapping the adapter variable to a literal, this field does not appear.</p>
Old Value	<p>By selecting this check box, you map the adapter variable to the value that was originally in the selected Qualifier field before modification.</p> <p>Process task adapters associated with process tasks are conditionally triggered when some field on the process form gets changed. If you click the Old Value checkbox, and the process task is marked Conditional, then the value that is passed to the adapter is the previous value of the field, before it got modified. This is useful in cases of fields that accept passwords. For example, if you want to disallow setting the password to the same value, you can use the old value for comparison.</p> <p>If you are not mapping the adapter variable to a field that belongs to a child table of a custom process form, this check box is disabled.</p>

11. Complete the Map To, Qualifier, IT Asset Type, IT Asset Property, Literal Value, and Old Value fields.

Note: For more information on which mappings to select, refer to [Tables, "Adapter Variable Mapping Information"](#).

12. From this window's toolbar, click Save. Then, click Close.

The Data Mapping for Variable window disappears. The **Integration** tab is active again.

13. From the Editing Task window's toolbar, click Save.

The contents in the **Status** field change from *Mapping Incomplete* to *Ready*. In addition, the mapping statuses for the adapter's variables change from *No (N)* to *Yes (Y)*.

14. From this window's toolbar, click Close.

The Editing Task window disappears, and the main screen is active once again. The adapter you added to the *Password Updated* task (*adpSOLARISPASSWORDUPDATED*) appears in the **Process Definition** form.

This signifies that the *adpSOLARISPASSWORDUPDATED* process task adapter was attached to the *Password Updated* process task.

Tip: Once you attach a process task adapter to a process task, a quick way to see the process and task to which it is connected is by accessing the Usage Lookup tab of the Adapter Factory form.

Removing Process Task Adapters from Process Tasks

If a process task adapter is no longer necessary for Oracle Identity Manager to complete the process task automatically, or when you wish to attach a different adapter to a process task, you must first remove the adapter that is attached to the process task.

This procedure will show you how to remove a process task adapter from a process task.

To Remove a Process Task Adapter from a Process Task

1. Open the Process Definition form.

Within the Xellerate Workspace, the **Process Definition** form appears.

2. Select the process, which contains a task from which you want to remove an adapter (for example, the *Solaris* process).

The selected process, along with its tasks, appears in the **Process Definition** form.

3. Double-click the row header of the process task from which you want to remove the adapter (for example, the *Password Updated* task).

The Editing Task window appears, containing information about the process task. Click the Integration tab.

4. Click the **Integration** tab.

The **Integration** tab appears, displaying information about the adapter that is attached to the process task.

5. From this tab, click **Remove**.

A dialog box appears, asking if you want to remove the adapter from the process task.

6. Click **OK**.

A dialog box appears, signifying that the adapter has been removed from the process task.

7. Click **OK**.

The contents of the adapter no longer appear in the **Integration** tab.

8. From this window's toolbar, click **Close**.

The Editing Task window disappears, and the main screen is active once again. The adapter that was once linked to the *Password Updated* task (*adpSOLARISPASSWORDUPDATED*) no longer appears in the child table of the **Process Definition** form.

This signifies that you have removed the adapter from the process task.

Applying Task Assignment Adapters

This chapter describes the process of applying task assignment adapters. It contains the following topics:

- ["Overview"](#) on page 5-1
- ["Attaching Task Assignment Adapters to Process Tasks"](#) on page 5-2
- ["Removing Task Assignment Adapters from Process Tasks"](#) on page 5-5

Overview

For a process task that needs to be completed manually, you can configure Oracle Identity Manager to automate the assignment of the task to either a specific user or to a user who belongs to a particular user group.

The Java code that enables Oracle Identity Manager to assign a process task to a user or group automatically is known as a task assignment adapter.

For efficiency purposes, a task assignment adapter can be used repeatedly for different process tasks. In addition, multiple task assignment adapters can be designated to be associated with a particular task. Therefore, Oracle Identity Manager selects the task assignment adapter for the process task. This occurs through task assignment rules.

Task assignment rules are criteria. These standards enable Oracle Identity Manager to select the task assignment adapter that is to be used to allocate the process task to the target user or group.

Note: In other words, the task assignment rule allows Oracle Identity Manager to decide whether to assign a process task to a user or group. The task assignment adapter enables Oracle Identity Manager to determine which user or group will be the recipient of the process task.

Each task assignment adapter has a task assignment rule associated with it. In addition, every rule has a priority number, which indicates the order in which Oracle Identity Manager will trigger it.

For this example, Oracle Identity Manager will trigger the Associate Adapter with User rule first (because it has the highest priority). If the condition of this rule is TRUE, then it is successful. As a result, Oracle Identity Manager will associate the related task assignment adapter (the Assign Task to User adapter) with the Get Solaris UUID process task.

On the other hand, when the condition of a rule is FALSE, then the rule has failed. Oracle Identity Manager will then trigger the rule with the next highest priority. If this rule is successful, then Oracle Identity Manager will assign the designated adapter to the target process task.

So, in this example, if the Associate Adapter with User rule fails, then Oracle Identity Manager will trigger the Associate Adapter with Group rule. If this rule is successful, then Oracle Identity Manager will attach the related task assignment adapter (the Assign Task to Group adapter) to the Get Solaris UUID process task.

After assigning a rule to a task assignment adapter, if this type of adapter contains adapter variables, then you must map these variables to their proper locations. Otherwise, the adapter will not be functional.

Finally, when a task assignment adapter either becomes invalid, or is no longer necessary for Oracle Identity Manager to allocate the process task to a user or group, you must remove the adapter from the task.

The following sections demonstrate how to attach a task assignment adapter to a process task, and remove a task assignment adapter from a process task.

Attaching Task Assignment Adapters to Process Tasks

In the [Chapter 3, "Creating Adapters"](#), you learned how to create a task assignment adapter. You must attach it to a process task so that Oracle Identity Manager can automate the assignment of the task to a user or group.

To connect a task assignment adapter to a process task, access the Assignment tab (from the Process Definition form). From this tab, you can also map any adapter variables to their proper locations.

This procedure will show you how to attach a task assignment adapter to a process task.

To Attach a Task Assignment Adapter to a Process Task

1. Open the Process Definition form, which is located in the Process Management folder.

Within the Oracle Identity Manager workspace, the Process Definition form appears.

2. Select the process, which contains a task to which you want to attach an adapter.

The selected process, along with its tasks, appears in the Process Definition form.

3. Double-click the row header of the task to which you want to attach a task assignment adapter.

The Editing Task window appears, containing information about the task (for example, the Get Solaris UUID process task).

4. Click the **Assignment** tab. The Assignment dialog box is displayed.

5. From this tab, click **Add**.

A blank row appears within the Assignment tab.

The following table lists the relevant fields of the Assignment tab:

Field Name	Description
Priority	From this field, set the priority number for the associated task assignment rule.
Rule	From this Lookup field, select the rule that will determine if the associated adapter will be used to automate the assignment of the process task to a user or group.
Target Type	From this Lookup field, specify whether the task is to be assigned to an Oracle Identity Manager user or group.
Adapter	From this Lookup field, select the adapter that is to be associated with the designated task assignment rule.
Adapter Status	This field displays the mapping status of the adapter's variables. To learn more about the various mapping statuses for an adapter, refer to the " Attaching Process Task Adapters to Process Tasks " on page 4-3.

6. Double-click the **Priority** field. From this field, set the priority number for the associated task assignment rule.
7. Double-click the **Rule** lookup field. From the Lookup dialog box that is displayed, select the rule that will determine if the associated adapter will be used to automate the assignment of the process task to a user or group.
8. Double-click the **Target Type** lookup field. From the Lookup dialog box that is displayed, specify whether the task is to be assigned to an Oracle Identity Manager user or group.
9. Double-click the **Adapter** lookup field. From the Lookup dialog box that is displayed, specify the task assignment adapter that is to be associated with the rule you selected in Step 7 of this procedure.
10. From the toolbar that is displayed within the Assignment tab, click **Save**.

The mapping status of the task assignment adapter variables is displayed within the Adapter Status field. Use the following table to decide which action to perform, based on the adapter's mapping status.

Mapping Status	Action
Ready	The adapter does not have any variables that need to be mapped. In other words, none of the adapter variables are return variables or have been designated as Resolve at Runtime. So, proceed to Step 14 of this procedure.
Mapping Incomplete	At least one of the adapter's variable needs to be mapped. So, proceed to Step 11 of this procedure.
Adapter Unavailable	After the adapter had been compiled successfully, it was modified. As a result, you must recompile the adapter.

Note: To learn more about the various mapping statuses for an adapter, refer to the "[Attaching Process Task Adapters to Process Tasks](#)" on page 5-2.

11. From this tab, click **Map**.

The Adapter Variables window appears. It displays the following information:

- The name of the task assignment adapter that is attached to the process task;
 - The status of the adapter; and
 - The mapping statuses, names, and descriptions of the adapter’s variables.
12. Set the mappings for each variable that appears in the **Adapter Variables** region of this window. To do so, double-click the row header of the variable you want to map (for example, UUID).

The Edit Data Mapping for Variable dialog box is displayed.

The following table lists the various fields of the Edit Data Mapping for Variable dialog box is displayed:

Field Name	Description
Variable Name	This field displays the name of the adapter variable for which you are setting a mapping (for example, UUID).
Data Type	This field shows the data type of the adapter variable (for example, <i>String</i> is the data type for the UUID variable).
Map To	<p>This field contains the types of mappings that you can set for the adapter variable (for example, IT Resources).</p> <p>When you map the adapter variable to a location or a contact, Oracle Identity Manager enables the adjacent combo box. From this combo box, select the specific type of location or contact to which you are mapping the adapter variable.</p> <p>In addition, if you map the adapter variable to a custom process form, and this form contains child table(s), then Oracle Identity Manager enables the adjacent combo box. From this combo box, select the child table to which you are mapping the adapter variable.</p> <p>If you are not mapping the adapter variable to a location, contact, or child table of a custom process form, this combo box is disabled.</p>
Qualifier	This field contains the qualifiers for the mapping you selected in the Map To combo box (for example, IT Asset).
IT Asset Type	<p>This field enables you to select a specific IT Resource (for example, <i>Solaris</i>) when you map an adapter variable to an IT Resource, and this variable’s data type is String.</p> <p>If you are not mapping the adapter variable to an IT Resource, or the variable’s data type is not String, this field does not appear.</p>
IT Asset Property	<p>This field enables you to select a specific field that will receive the results of the mapping (for example, Unique ID), when you map an adapter variable to an IT Resource, and this variable’s data type is <i>String</i>.</p> <p>If you are not mapping the adapter variable to an IT Resource, or if the variable’s data type is not String, then this field does not appear.</p> <p>Important: The IT Asset Type and IT Asset Property fields are included within this window for backward compatibility. The preferred way is to create an adapter variable with a data type of IT Resource, in which case these fields will not appear.</p>
Literal Value	<p>When you map the adapter variable to a literal, use this field to specify the specific literal value.</p> <p>If you are not mapping the adapter variable to a literal, then this field does not appear.</p>

Field Name	Description
Old Value	<p>By selecting this check box, you map the adapter variable to the value that was originally in the selected Qualifier field before modification.</p> <p>Process task adapters associated with process tasks are conditionally triggered when some field on the process form is changed. If you click the Old Value checkbox, and the process task is marked Conditional, then the value that is passed to the adapter is the previous value of the field. This is useful in cases of fields that accept passwords.</p> <p>For example, if you want to disallow setting the password to the same value, you can use the old value for comparison.</p> <p>If you are not mapping the adapter variable to a field that belongs to a child table of a custom process form, this check box is disabled.</p>

13. Complete the Map To, Qualifier, IT Asset Type, IT Asset Property, Literal Value, and Old Value fields.

Note: For more information on which mappings to select, refer to the ["Adapter Variable Mapping Information"](#) on page A-5.

14. From this window's toolbar, click **Save**. Then, click **Close**.

The Edit Data Mapping for Variable window disappears. The Adapter Variables dialog box is active again.

The contents in the Status field change from Mapping Incomplete to Ready. In addition, the mapping statuses for the adapter's variables change from No (N) to Yes (Y).

15. Click **Save**. Then, click **Close**

The Adapter Variable dialog box disappears, and the Assignment tab is active once again.

The adapter that you assigned to the process task (for example, *Assign Solaris Task*) now has a status of Ready.

16. From the toolbar that appears within the Assignment tab, click **Save** and **Close**

The Assignment tab disappears, and the main screen is active once again. This signifies that the task assignment adapter is attached to the process task.

Note: Once you attach a task assignment adapter to a process task, a quick way to see the process and the task to which it is connected is by accessing the Usage Lookup tab of the Adapter Factory form.

Removing Task Assignment Adapters from Process Tasks

When a task assignment adapter either becomes invalid, or is no longer necessary for Oracle Identity Manager to allocate the process task to a user or group, you must remove the adapter from the task.

To Remove a Task Assignment Adapter from a Process Task

To detach a task assignment adapter from a process task, perform the following tasks:

1. Open the Process Definition form.

Within the Xellerate Workspace, the Process Definition form appears.

2. Select the process, which contains a task from which you want to remove an adapter (for example, the Solaris 8 process)

The selected process, along with its tasks, appears in the Process Definition form.

3. Double-click the row header of the process task from which you want to remove the adapter (for example, the Get Solaris UUID task).

The Editing Task dialog box is displayed, containing information about the process task.

4. Click the **Assignment** tab.

The Assignment tab appears, displaying information about the adapter that is attached to the process task.

5. Highlight the row, containing the adapter that you want to remove from the process task.

6. Click **Delete**. The adapter no longer appears within the Assignment tab.

7. Click **Save**. Then, click **Close**.

The Assignment tab disappears, and the Main Screen is active once again. This signifies that the task assignment adapter is removed from the process task.

Understanding Rule Generators

This chapter describes the rule generators. It contains the following topics:

- ["Overview"](#) on page 6-1
- ["Mapping Rule Generator Adapter Variables"](#) on page 6-2
- ["Associating Rule Generators with Processes"](#) on page 6-5
- ["Removing Rule Generators from Form Fields"](#) on page 6-5

Overview

To perform field validations and enter default values into forms, which either come packaged with Oracle Identity Manager or are created by Oracle Identity Manager users, certain business rules must be applied. As an example, for the Users form, you may want Oracle Identity Manager to generate the User ID automatically by concatenating the user's first name and last name.

To do this, you must create a specific type of adapter, which is designed to modify a form's field value. You can do this by using the Adapter Factory form. Then, when you attach this adapter to the form, Oracle Identity Manager will automatically update the field value for all records of that form, and save this information to the Oracle Identity Manager database.

This type of adapter, which can generate, modify, or verify the value of a form's field automatically, is called a rule generator. Oracle Identity Manager triggers a rule generator on preinsert and preupdate.

If you create a rule generator, and it contains adapter variables, then you must map these adapter variables to their proper locations. Otherwise, the adapter will not be functional.

Also, you can attach this adapter to a provisioning process for the adapter to work. Then, once the process is provisioned to a target user or organization, Oracle Identity Manager will trigger the associated rule generator.

On occasion, a rule generator, which has been assigned to a provisioning process, may no longer be needed for the process to be completed. If this happens, then you need to remove the rule generator from the provisioning process. Similarly, after you attach one rule generator to a form field, you may need to connect a different rule generator to that form field. When this occurs, you must first remove the rule generator that is currently attached to the form field.

The following sections will show you how to:

- Map the adapter variables of a rule generator.

- Associate rule generators with provisioning processes.
- Remove a rule generator from a form field.

Mapping Rule Generator Adapter Variables

In the [Example 3, "Creating Adapters"](#), you learned how to create a rule generator. Now, you must map the adapter variables of the rule generator to their proper locations. This will ensure that adapter will work.

To map these adapter variables, access the Data Object Manager form (from the Development Tools/Business Rule Definition folder).

This procedure will show you how to map the adapter variables for a rule generator.

To Map a Rule Generator's Adapter Variables

1. Open the Data Object Manager form. Within the Xellerate Workspace, the Data Object Manager form is displayed.

The following table lists and describes the various regions of the Data Object Manager form:

Name	Description
Form Description Field	From this Lookup field, select the form that contains the field to which you are attaching the rule generator.
Data Object Field	This field displays the name of the data object, which is represented by the selected form.
Attach Handlers Tab	This tab displays: <ul style="list-style-type: none"> The rule generators that are attached to the selected form. The execution schedule of the rule generators associated with this form. The order in which Oracle Identity Manager will run the rule generators. Insert, update, and delete permissions for user groups.
Map Adapters Tab	This tab displays: <ul style="list-style-type: none"> The names of the rule generators that are associated with the form; The status of these adapters. The names, descriptions, and mapping statuses of the rule generators' adapter variables. <p>Note: The Map Adapters tab is disabled until an adapter is assigned to the current data object. For more information about assigning an adapter to a data object, refer to <i>Oracle Identity Manager Administrative and User Console Guide</i>.</p>

2. Double-click the **Form Description** field. A Lookup dialog box is displayed, displaying the forms to which you can attach rule generators.
3. Select the form you want (for example, *Solaris*). Then, click **OK**.
4. From the toolbar, click **Save**.

The selected form, the form's data object, and the rule generator adapters associated with the form appear. In addition, Oracle Identity Manager enables the Map Adapters tab.

For this example, the Solaris form has been selected. Its data object (*Thor.CarrierBase.tcUD_SOLARIS*) appears, along with the four rule generator adapters associated with it (*adpCONVERTTOLOWERCASE*, *adpSOLARISHMDSTRINGGEN*, *adpSETSOLARISASSET*, and *adpSETPASSWORDFROMMAIN*). Oracle Identity Manager will trigger these four rule generators on preinsert.

Based on the sequence numbers of these adapters, Oracle Identity Manager will trigger the *adpCONVERTTOLOWERCASE* adapter first, followed by the *adpSOLARISHMDSTRINGGEN*, *adpSETSOLARISASSET*, and *adpSETPASSWORDFROMMAIN* adapters, respectively.

Tip: To change the order sequence of a rule generator:

1. Click Assign. The Event Handlers dialog box is displayed.
2. Select the rule generator from.
3. Click the up arrow and down arrow buttons to modify the order of the rule generator.

For these rule generators to work properly, you must map the adapter variables to their proper locations.

5. Click the **Map Adapters** tab.
6. From the Name combo box, select the rule generator, which has adapter variables that need to be mapped (for example, the *adpCONVERTTOLOWERCASE* rule generator).

The Map Adapters tab now displays the following:

- The name of the rule generator that is to be attached to the form.
- The status of the rule generator.
- The names, descriptions, and mapping statuses of the rule generator's adapter variables.

Note: To learn more about the various mapping statuses for an adapter, refer to the ["Attaching Process Task Adapters to Process Tasks"](#) on page 4-3.

Set the mappings for each variable that appears in the Adapter Variables region of the Map Adapters tab. To do so:

7. Double-click the row header of the variable you want to map (for example, *Data*). The Data Mapping for Variable dialog box is displayed.

The following table describes the various fields of the Data Mapping for Variable dialog box:

Field Name	Description
Variable Name	This field displays the name of the adapter variable for which you are setting a mapping (for example, <i>Data</i>).

Field Name	Description
Data Type	This field shows the data type of the adapter variable (for example, <i>String</i> is the data type for the Data adapter variable).
Map To	<p>This field contains the source and target locations of the mappings you can set for the adapter variable (for example, <i>User Definition</i>).</p> <p>When you map the adapter variable to a location or a contact, Oracle Identity Manager enables the adjacent combo box. From this combo box, select the specific type of location or contact to which you are mapping the adapter variable.</p> <p>If you are not mapping the adapter variable to a location or contact, this combo box is disabled.</p>
Qualifier	This field contains the qualifiers for the mapping you selected in the Map To combo box (for example, <i>User Login</i>).
IT Asset Type	<p>This field enables you to select a specific IT Resource (for example, <i>Solaris</i>) when you map an adapter variable to an IT Resource, and this variable's data type is <i>String</i>.</p> <p>If you are not mapping the adapter variable to an IT Resource, or the variable's data type is not <i>String</i>, this field does not appear.</p>
IT Asset Property	<p>This field enables you to select a specific field that will receive the results of the mapping (for example, <i>User Name</i>), when you map an adapter variable to an IT Resource, and this variable's data type is <i>String</i>.</p> <p>If you are not mapping the adapter variable to an IT Resource, or the variable's data type is not <i>String</i>, this field does not appear.</p> <p>Important: The IT Asset Type and IT Asset Property fields are included within this window for backward compatibility. The preferred way is to create an adapter variable with a data type of IT Resource, in which case these fields will not appear.</p>
Literal Value	<p>When you map the adapter variable to a literal, type the name of the specific literal in this text box (for example, <i>IBM</i>).</p> <p>If you are not mapping the adapter variable to a literal, this field does not appear.</p>

Complete the Map To, Qualifier, IT Asset Type, IT Asset Property, and Literal Value fields.

Note: For more information on which mappings to select, refer to the "[Adapter Variable Mapping Information](#)" on page A-5.

8. Click *Save*. Then, click **Close**

The Data Mapping for Variable window disappears. The Map Adapters tab is active again.

9. From the main screen's toolbar, click *Save*.

Repeat Steps 7 and 8 for all adapter variables that need to be mapped.

The contents in the Status field change from Mapping Incomplete to Ready. In addition, the mapping statuses for the adapter's variables change from *No (N)* to *Yes (Y)*.

This signifies that all the adapter variables for the rule generator adapter have been mapped correctly. You are now ready to attach this rule generator to a provisioning process, so it can be triggered, once the process is provisioned to a target user or organization.

Tip: Once you map all the adapter variables for a rule generator that is associated with a form, a quick way to see the form to which it is attached, as well as the execution schedule of the rule generator, is by accessing the Usage Lookup tab of the Adapter Factory form.

Once the rule generator is assigned to a process, and the process is provisioned, the rule generator will be executed by Oracle Identity Manager.

Associating Rule Generators with Processes

Once you map the adapter variables of a rule generator to their proper locations, you must attach it to a provisioning process. Then, once the process is provisioned to a target user or organization, Oracle Identity Manager will trigger the associated rule generator.

Similarly, when a rule generator, which has been assigned to a provisioning process, is no longer needed for the process to be completed, you must remove the rule generator from the provisioning process.

To assign a rule generator to a provisioning process or remove a rule generator from a provisioning process, access the Event Handlers/Adapters tab of the Process Definition form. This form can be found in the Process Management folder.

Removing Rule Generators from Form Fields

Sometimes, after you attach one rule generator to a form field, you may need to connect a different rule generator to that form field. When this occurs, you must first remove the rule generator that is currently attached to the form field.

This procedure will show you how to remove a rule generator from a form field.

Caution: If you remove a rule generator from a form, and if the class name of the form's data object matches the table name of a provisioning process, then you will not be able to assign the rule generator to that provisioning process.

For example, suppose the `adpCONVERTTOLOWERCASE` rule generator is removed from the Solaris form. If the class name of the form's associated data object is `UD_SOLARIS`, then the rule generator cannot be assigned to any provisioning process with a table name of `UD_SOLARIS`.

To Remove a Rule Generator from a Form Field

To remove a rule generator from a form field, perform the following steps:

1. Open the Data Object Manager form.
2. Select the form that contains a rule generator you want to remove.
3. The selected form, along with its rule generators, appear in the Data Object Manager form.

4. Click the rule generator that you want to remove from the form field.
5. Click **Delete**.

The selected rule generator no longer appears in the Data Object Manager form. This indicates that you have removed the rule generator from the form field.

Caution: If you attempt to remove a rule generator from a form field, and if an error box appears, then the adapter has already been associated with a provisioning process. First, detach the rule generator from the process. Then, you can remove it from the form field.

Using Prepopulate Adapters

This chapter describes the prepopulate adapters. It contains the following tasks:

- ["Overview"](#) on page 7-1
- ["Attaching Pre-Populate Adapters to Form Fields"](#) on page 7-2
- ["Removing Pre-Populate Adapters from Form Fields"](#) on page 7-5

Overview

Sometimes, a user-created form contains two types of fields: fields that need to be populated by Oracle Identity Manager; and fields, into which an Oracle Identity Manager user must enter data. When the information that the user types into a field is contingent upon the data that appears in a system-generated field, Oracle Identity Manager must first populate this field. Then, once the form is displayed, an Oracle Identity Manager user can view the system-generated data, and, as a result, enter information into the appropriate fields.

For this to happen, you must create a specific type of rule generator. Then, by attaching it to the field that is designated to be system-generated, Oracle Identity Manager will automatically populate this field with the appropriate information, without saving this information to the Oracle Identity Manager database.

This type of rule generator is known as a prepopulate adapter. The data, which is generated by a pre-populate adapter, can appear automatically or manually entering it. Oracle Identity Manager displays this information automatically when the Auto-Pre-populate check box is checked for a provisioning process. When this check box is unchecked, an Oracle Identity Manager user must manually generate the displaying of the data that is generated by the prepopulate adapter. To do this, the administrator must click the Pre-Populate button on the form section of the Direct Provisioning wizard in the Web client, while provisioning the form to a user.

For efficiency purposes, a pre-populate adapter can be used repeatedly for different form fields. In addition, multiple pre-populate adapters can be designated to be associated with a particular field. As a result, Oracle Identity Manager must know which pre-populate adapter it needs to select for the form field. This occurs using the prepopulate rules.

Prepopulate rules are criteria. These standards enable Oracle Identity Manager to select one pre-populate adapter, which is associated with a form field, once this pre-populate adapter is assigned to the field.

Each pre-populate adapter has a pre-populate rule associated with it. In addition, every rule has a priority number, which indicates the order in which Oracle Identity Manager will trigger it.

For this example, Oracle Identity Manager will trigger the Rule for Uppercase User ID rule first (because it has the highest priority). If the condition of this rule is TRUE, then it is successful. As a result, Oracle Identity Manager will attach the related pre-populate adapter (the Display Uppercase Letters for User ID adapter) to the User ID field.

On the other hand, when the condition of a rule is FALSE, then the rule has failed. Oracle Identity Manager will then trigger the rule with the next highest priority. If this rule is successful, then Oracle Identity Manager will attach the associated adapter to the designated field.

So, in this example, if the Rule for Uppercase User ID rule fails, Oracle Identity Manager will trigger the Rule for Lowercase User ID rule. If this rule is successful, Oracle Identity Manager will attach the related prepopulate adapter (the Display Lowercase Letters for User ID adapter) to the User ID field.

After assigning a rule to a prepopulate adapter, if this type of adapter contains adapter variables, then you must map these adapter variables to their proper locations. Otherwise, the adapter will not be functional.

Finally, when a prepopulate adapter, which has been associated with a field, is no longer valid, you must remove the adapter from the field.

The following sections will show you how to attach a prepopulate adapter to a form field, and remove a prepopulate adapter from a form field.

Attaching Pre-Populate Adapters to Form Fields

To attach a prepopulate adapter to a form field, perform the following steps:

1. Select the field to which a prepopulate adapter will be attached.
2. Select the rule that will determine if the adapter will be used to populate the designated field with information.
3. Select the adapter that will be associated with the designated field.
4. Set the priority number of the selected rule.
5. Map the adapter variables of the prepopulate adapter to their proper locations.

The following procedure demonstrates how to complete these steps, and, as a result, attach a prepopulate adapter to a form field.

Note: To attach a prepopulate adapter to a form field, you must ensure the following:

- The form is not in an active state. Otherwise, create a new form version.
 - After attaching the adapter, you must activate the form to be able to use it.
-
-

To Attach a Prepopulate Adapter to a Form Field

To attach a prepopulate adapter to a Form field, perform the following steps:

1. Open the Form Designer form.
2. Query for the form to which you want to attach a pre-populate adapter (for example, Solaris).

3. Click the **Pre-Populate tab.**

The prepopulate adapter, which have already been attached to the form you queried, appear within this tab.

Note: If no adapters have been attached to a form field, then the Prepopulate tab will be empty.

4. Click Add.

The Pre-Populate Adapters dialog box is displayed.

The following table lists and describes the various regions of the Pre-Populate Adapters dialog box:

Name	Description
Field Name	This combo box contains a list of all of the form fields to which a prepopulate adapter can be attached.
Rule	From this Lookup field, select the rule that will determine if the associated adapter will be used to populate the designated form field with information.
Adapter	From this Lookup field, select the adapter that will be associated with the designated field.
Order	From this text box, set the priority number of the selected rule.
Adapter Status	This field displays the mapping status of the adapter variables. To learn more about the various mapping statuses for an adapter, refer to the " Attaching Process Task Adapters to Process Tasks " on page 4-3.
Adapter Variables	This area displays the following: Mapped. The mapping statuses of the adapter's variables. "Y" indicates that an adapter variable has been mapped properly; "N" indicates that this variable has not been mapped correctly. Name. The names of the adapter variables. Mapped to. The form fields to which the variables are mapped If an adapter variable is not yet mapped, then the corresponding cell in this column will be empty.

- 5.** From the **Field Name** combo box, select the form field to which the prepopulate adapter will be attached. (for example, User ID).
- 6.** Double-click the **Rule** lookup field. From the Lookup dialog box that is displayed, select the rule that will determine if the associated adapter will be used to populate the designated form field with information (for example, Rule for Lowercase User ID).
- 7.** Double-click the **Adapter** lookup field. From the Lookup dialog box that is displayed, choose the adapter that will be associated with the field you selected in Step 5 (for example, Display Lowercase Letters for User ID).
- 8.** In the **Order** field, enter the priority number of the rule you selected in Step 6 (for example, 2).
- 9.** From the Pre-Populate Adapters window's toolbar, click **Save**

10. Mapping Incomplete appears within the Adapter Status field. This signifies that the adapter you selected contains variables that have not been mapped correctly. These variables need to be mapped to their proper locations. Otherwise, the adapter will not work.
11. Set the mappings for each variable that appears in the Adapter Variables region of the Pre-Populate Adapters window. To do so, double-click the row header of the variable you want to map (for example, UserID).

The Map Adapter Variables window is displayed.

The following table describes the various fields of the Map Adapter Variables window:

Field Name	Description
Variable Name	This field displays the name of the adapter variable for which you are setting a mapping (for example, UserID).
Data Type	This field shows the data type of the adapter variable (for example, <i>String</i> is the data type for the UserID adapter variable).
Map To	<p>This field contains the types of mappings that you can set for the adapter variable (for example, Process Data).</p> <p>When you map the adapter variable to a location or a contact, Oracle Identity Manager enables the adjacent combo box. From this combo box, select the specific type of location or contact to which you are mapping the adapter variable.</p> <p>If you are not mapping the adapter variable to a location or contact, this combo box is disabled.</p>
Qualifier	This field contains the qualifiers for the mapping you selected in the Map to combo box (for example, User ID).
IT Asset Type	<p>This field enables you to select a specific IT Resource (for example, Solaris) when you map an adapter variable to an IT Resource, and this variable's data type is String.</p> <p>If you are not mapping the adapter variable to an IT Resource, or the variable's data type is not String, this field does not appear.</p>
IT Asset Property	<p>This field enables you to select a specific field that will receive the results of the mapping (for example, <i>User Name</i>), when you map an adapter variable to an IT Resource, and this variable's data type is String.</p> <p>If you are not mapping the adapter variable to an IT Resource, or the variable's data type is not String, then this field does not appear.</p> <p>Important: The IT Asset Type and IT Asset Property fields are included within this window for backward compatibility. The preferred way is to create an adapter variable with a data type of IT Resource, in which case these fields will not appear.</p>
Literal Value	<p>When you map the adapter variable to a literal, use this field to specify the specific literal value.</p> <p>If you are not mapping the adapter variable to a literal, this field does not appear.</p>

12. Complete the Map To, Qualifier, IT Asset Type, IT Asset Property, and Literal Value fields.

Note: For more information on which mappings to select, refer to the "[Adapter Variable Mapping Information](#)" on page A-5.

13. From the Map Adapter Variable's toolbar, click **Save**. Then, click **Close**.

The Map Adapter Variables window disappears. The Pre-Populate Adapters window is active again.

The text in the Adapter Status field changes from Mapping Incomplete to Ready. In addition, the mapping statuses for the adapter's variables change from No (N) to Yes (Y).

14. From the Pre-Populate Adapters window's toolbar, click **Close**.

The Pre-Populate Adapters window disappears, and the Form Designer form is active again. The pre-populate adapter, which you attached to the User ID form field (Display Lowercase Letters for User ID), appears in the Pre-Populate tab of the Results of 1Q Sales 2003 form.

Once a process, which references this form, is provisioned to a target user or organization, the form will appear. Oracle Identity Manager will then check to see if the pre-populate rule, which has the highest priority, is valid. If so, Oracle Identity Manager will assign the associated pre-populate adapter to the designated field (User ID), and execute it. At this point, one of the following actions occur:

- If the Auto-Pre-populate check box was selected for the provisioning process, Oracle Identity Manager will display the data that is generated by the pre-populate adapter automatically.
- If the Auto-Pre-populate check box was unchecked, an Oracle Identity Manager user must manually trigger the displaying of the data that is generated by the pre-populate adapter. To do this, the administrator must click the Pre-Populate button on the form section of the direct provisioning wizard in the Web client, while provisioning the form to a user.

Tip: Once you allocate a pre-populate adapter to a form field, and assign a pre-populate rule to the adapter, a quick way to see the association among the adapter, the form field, and the rule is by accessing the Usage Lookup tab of the Adapter Factory form.

Removing Pre-Populate Adapters from Form Fields

If a prepopulate adapter, which has been associated with a form field, is no longer valid, then you must remove the adapter from the field.

The following procedure will show you how to remove a pre-populate adapter from a form field.

Note: Before removing the pre-populate adapter from a form field, you must create a new version of the form before removing the adapter.

To Remove a Prepopulate Adapter from a Form Field

To remove a prepopulate adapter from a Form Field, perform the following steps:

1. Highlight the prepopulate adapter that you want to remove.

2. Click **Delete**. The pre-populate adapter is removed from the form field. It cannot be triggered when the form is launched.
3. After removing the adapter, you must activate the form.

Managing Entities

Similar to rule generator adapters, entity adapters are also responsible for generating, modifying, or verifying the value of a form's field automatically, and saving this information to the Oracle Identity Manager database. However, the differences between rule generators and entity adapters are as follows:

1. **Execution schedule.** Entity adapters can be triggered by Oracle Identity Manager on preinsert, preupdate, predelete, postinsert, postupdate, and postdelete. A rule generator adapter can be executed only on preinsert and preupdate.
2. **Manual field value modification.** The adapter populates the form field to which an entity adapter is attached. An Oracle Identity Manager user should not edit this value because the entity adapter will overwrite this modification. As a result, the modification will not be saved to the database.

Similarly, the adapter also populates the form field to which a rule generator adapter is attached. However, an Oracle Identity Manager user can edit this value because this modification will take precedence over the value that the rule generator adapter generates. Because of this, the modification will be saved to the database.

3. **Background color of form field.** If a rule generator is attached to a form field, then the field will appear in a particular background color (for example, pink). This is a visual indicator that the field has a rule generator attached to it. On the other hand, when an entity adapter is attached to a form field, the field will not have a distinct background color.

Note: For more information about mapping the variables of an entity adapters with Oracle Identity Manager forms and/or provisioning processes, refer to the procedures in the [Chapter 6, "Understanding Rule Generators"](#) on page 6-1.

Compiling Adapters

When you create multiple adapters, and need to compile them, you can do so in one of two ways. First, you can access each adapter, and compile it individually. However, this method is unproductive.

A more-efficient approach is to either select the adapters that you want to compile, and compile them in one shot, or, when necessary, to compile all adapters that exist in the Oracle Identity Manager database, by clicking a button.

Compiling Adapters

The Adapter Manager form is located in the Development Tools folder. It is used to compile multiple adapters simultaneously.

You will now learn how to compile several adapters at the same time.

To Compile Multiple Adapters

To compile multiple adapters, perform the following steps:

1. Open the Adapter Manager form.
2. To compile every adapter that resides within the Oracle Identity Manager database, select the **Compile All** option.

To compile multiple adapters, highlight those adapters you wish to compile. Then, select the **Compile Selected** option.

To compile all adapters that do not have an OK status, select the **Compile Previously Failed** option.

3. Click the **Start** button.

Oracle Identity Manager will compile the adapters that match the criteria you specified in Step 2.

Tip: Oracle Identity Manager allows you to look at the record of any adapter that appears within the Adapter Manager form. By doing so, you can see detailed information about the adapter.

To view an adapter's record, first, highlight the desired adapter. Then, either double-click its row header, or right-click the adapter, and select the Launch Adapter command from the menu that appears.

Exporting and Importing Adapters

This chapter describes the exporting and importing adapters.

Sometimes, when you create an adapter for one database, you may want to use it with another database. One approach is to manually reconstruct the adapter for that database. However, besides this method being unproductive in terms of time, money, and resources, the developer may inadvertently introduce an error while re-creating the adapter. As a result, once the process with which this adapter is associated is provisioned to the target user or organization, the adapter will not be able to perform its designated function, and Oracle Identity Manager will not be able to provision the corresponding resource to that user or organization.

An alternate method, which is more efficient, and eliminates the chance for error, is to:

1. Create a bug-free adapter and apply it properly within one database. The Deployment Manager, internally, builds an *.xml file, which contains this adapter and all of its internal variable mappings.

The Deployment Manager also exports this file into a designated location.

2. Retrieve the file from the location, and import it into the target database.

You can use Deployment Manager to transfer adapters from one database to another. Refer to the *Oracle Identity Manager Administrative and User Console Guide* for instructions about how to use Deployment Manager.

Note: Any adapter that you import into a target database must be recompiled within that database. Also, you should copy the third-party jars files, after migrating the adapter. Otherwise, the adapter will not work.



Creating and Testing a Remote Manager IT Resource

This chapter describes the tasks for creating and testing a Remote Manager IT Resource. It contains the following topics:

- [Post-installation Configuration](#)
- [To Create and Test a Remote Manager IT Resource](#)

Remote Manager is an Oracle Identity Manager component that acts like a proxy in directly communicating with a third-party system. The Remote Manager is used to invoke non-remotable APIs through Oracle Identity Manager and APIs that do not support SSL over secure connections.

After installing the Remote Manager and establishing the trust relation between the Oracle Identity Manager Server and the Remote Manager (trusting the certificate), you can create an IT Resource for the Remote Manager and then test it.

Post-installation Configuration

After installing the Remote Manager, you need to ensure that the certificate is trusted between the application server and the Remote Manager. To do so, first open the Remote Manager form in the Xellerate Administration folder of the Design Console (Java client). The Remote Manager form will show all Remote Managers that are connected but not necessarily "trusted".

Perform the following steps to ensure that the trust relation between the application server and the Remote Manager is established through the certificate. In this procedure, the JBoss Application Server is used as an example. The keytool utility is used to import/export the certificates.

1. Using a command prompt, open the directory path and use the keytool utility to list the certificate fingerprints:

```
<XLREMOTE_HOME>\xlremote>
```

2. Enter the command:

```
<JAVA_HOME>\bin\keytool -list -keystore config\xlkeystore
```

3. Enter the default password for xellerate keystore: xellerate

```
Your keystore contains 1 entry
    xell, Jan 7, 2005, keyEntry,
    Certificate fingerprint (MD5):
```

B0:F2:33:C8:69:E4:25:A3:CB:59:E8:51:27:EE:5C:52

The certificate fingerprint is in marked in bold. Compare this to the list of certificates in the keystore.

4. Open the Java SDK folder used for the application server. Again, enter the path and use the keytool to list the certificates in the keystore:

```
<JAVA_HOME>\jre\lib\security\cacerts
```

5. Enter the command to see the list of trusted certificates:

```
<JAVA_HOME>\bin\keytool -keystore cacerts -storepass changeit
-storetype jks -provider <provider_name>
```

The output showing the keystore entries are as follows:

```
Your keystore contains 25 entries
equifaxsecureebusinessca1, Jul 23, 2003, trustedCertEntry,
Certificate fingerprint (MD5): 64:9C:EF:2E:44:FC:C6:8F:52:07:D0:51:73:8F:CB:3D
verisignclass4ca, Jun 29, 1998, trustedCertEntry,
Certificate fingerprint (MD5): 1B:D1:AD:17:8B:7F:22:13:24:F5:26:E2:5D:4E:B9:10
entrustglobalclientca, Jan 9, 2003, trustedCertEntry,
Certificate fingerprint (MD5): 9A:77:19:18:ED:96:CF:DF:1B:B7:0E:F5:8D:B9:88:2E
gtecybertrustglobalca, May 10, 2002, trustedCertEntry,
Certificate fingerprint (MD5): CA:3D:D3:68:F1:03:5C:D0:32:FA:B8:2B:59:E8:5A:DB
entrustgsslca, Jan 9, 2003, trustedCertEntry,
Certificate fingerprint (MD5): 9D:66:6A:CC:FF:D5:F5:43:B4:BF:8C:16:D1:2B:A8:99
verisignclass1ca, Jun 29, 1998, trustedCertEntry,
Certificate fingerprint (MD5): 51:86:E8:1F:BC:B1:C3:71:B5:18:10:DB:5F:DC:F6:20
thawtepersonalbasicca, Feb 12, 1999, trustedCertEntry,
Certificate fingerprint (MD5): E6:0B:D2:C9:CA:2D:88:DB:1A:71:0E:4B:78:EB:02:41
entrustsslca, Jan 9, 2003, trustedCertEntry,
Certificate fingerprint (MD5): DF:F2:80:73:CC:F1:E6:61:73:FC:F5:42:E9:C5:7C:EE
thawtepersonalfreemailca, Feb 12, 1999, trustedCertEntry,
Certificate fingerprint (MD5): 1E:74:C3:86:3C:0C:35:C5:3E:C2:7F:EF:3C:AA:3C:D9
verisignclass3ca, Oct 24, 2003, trustedCertEntry,
Certificate fingerprint (MD5): 10:FC:63:5D:F6:26:3E:0D:F3:25:BE:5F:79:CD:67:67
gtecybertrustca, May 10, 2002, trustedCertEntry,
Certificate fingerprint (MD5): C4:D7:F0:B2:A3:C5:7D:61:67:F0:04:CD:43:D3:BA:58
thawteserverca, Feb 12, 1999, trustedCertEntry,
Certificate fingerprint (MD5): C5:70:C4:A2:ED:53:78:0C:C8:10:53:81:64:CB:D0:1D
equifaxsecureca, Jul 23, 2003, trustedCertEntry,
Certificate fingerprint (MD5): 67:CB:9D:C0:13:24:8A:82:9B:B2:17:1E:D1:1B:EC:D4
thawtepersonalpremiumca, Feb 12, 1999, trustedCertEntry,
Certificate fingerprint (MD5): 3A:B2:DE:22:9A:20:93:49:F9:ED:C8:D2:8A:E7:68:0D
thawtepremiumserverca, Feb 12, 1999, trustedCertEntry,
Certificate fingerprint (MD5): 06:9F:69:79:16:66:90:02:1B:8C:8C:A2:C3:07:6F:3A
entrust2048ca, Jan 9, 2003, trustedCertEntry,
Certificate fingerprint (MD5): BA:21:EA:20:D6:DD:DB:8F:C1:57:8B:40:AD:A1:FC:FC
verisignserverca, Jun 29, 1998, trustedCertEntry,
Certificate fingerprint (MD5): 74:7B:82:03:43:F0:00:9E:6B:B3:EC:47:BF:85:A5:93
entrustclientca, Jan 9, 2003, trustedCertEntry,
Certificate fingerprint (MD5): 0C:41:2F:13:5B:A0:54:F5:96:66:2D:7E:CD:0E:03:F4
baltimorecybertrustca, May 10, 2002, trustedCertEntry,
Certificate fingerprint (MD5): AC:B6:94:A5:9C:17:E0:D7:91:52:9B:B1:97:06:A6:E4
geotrustglobalca, Jul 23, 2003, trustedCertEntry,
Certificate fingerprint (MD5): F7:75:AB:29:FB:51:4E:B7:77:5E:FF:05:3C:99:8E:F5
gtecybertrust5ca, May 10, 2002, trustedCertEntry,
Certificate fingerprint (MD5): 7D:6C:86:E4:FC:4D:D1:0B:00:BA:22:BB:4E:7C:6A:8E
equifaxsecureglobalebusinessca1, Jul 23, 2003, trustedCertEntry,
Certificate fingerprint (MD5): 8F:5D:77:06:27:C4:98:3C:5B:93:78:E7:D7:7D:9B:CC
baltimorecodesigningca, May 10, 2002, trustedCertEntry,
```

Certificate fingerprint (MD5): **90:F5:28:49:56:D1:5D:2C:B0:53:D4:4B:EF:6F:90:22**
 equifaxsecureebusinessca2, Jul 23, 2003, trustedCertEntry,
 Certificate fingerprint (MD5): **AA:BF:BF:64:97:DA:98:1D:6F:C6:08:3A:95:70:33:CA**
 verisignclass2ca, Oct 24, 2003, trustedCertEntry,
 Certificate fingerprint (MD5): **B3:9C:25:B1:C3:2E:32:53:80:15:30:9D:4D:02:77:3E**

For clarity, the certificate fingerprints are highlighted in bold. The certificate fingerprint that is required is Certificate fingerprint (MD5).

B0:F2:33:C8:69:E4:25:A3:CB:59:E8:51:27:EE:5C:52) is not in the trusted certificates. So, you will need to import the certificate.

To Create and Test a Remote Manager IT Resource

To create and test a Remote Manager IT resource, perform the following steps:

1. In the Oracle Identity Manager Design Console, open the Resource Object form.
2. Create a Resource object. In this example, the following parameters are set:
 - The name is *MyObj*
 - The option, *Order for User* is enabled
 - The Type is *Application*
 - The following checkboxes are enabled:
 - Allowed Multiple
 - Auto Save
 - Self Request Allowed
 - Allow All
 - Auto Launch
3. Create an IT Resource Type Definition for Resource Object. Open the IT Resource Type Definition form. In this example, the following parameters are set:
 - The Server Type name is *MyObjServer*
 - The Password field name is encrypted
4. Create an IT Resource for the Remote Manager. In this example, the following parameters are set:
 - The name of the IT Resource is *remote*
 - The name of the Type is *Remote Manager*

Ensure that the IT resource has the proper URL and service name, and that the Remote Manager is installed at the location indicated by the URL.

Note: Check to see if the name itself is not present in the URL. For example, the Remote Manager is composed of the service name and URL, as such:

service name: *RManager* url: *rmi://w2kevandanwkstn:12346*

5. Create an instance of the *MyObjServer* IT Resource Type created previously. Open the IT Resource Information Form. In the Remote Manager field, ensure that the Remote Manager created in the Step 4 is selected.

6. After saving the information in the IT Resources Information form, you can provide any additional details required for that IT resource. In this example, the user name and password are entered.
7. Create the .jar files that contains code and will be executed in the Remote Manager. For this you need to perform the following:
8. Copy these .jar files to the JavaTasks folder in <xlremote_home>/JavaTasks and <XL_HOME>/JavaTasks.

The following lines of code tests the settings you have made so far:

```
package testme;
import java.io.PrintStream;
public class test
{
    public test ()
    {
    }
    public static int addme(int i, int j)
    {
        /*6*/System.out.println(i + "+" + j + "=" + (i + j));
        /*7*/return i + j;
    }
    public static void main(String args[])
    (
        /* 11*/addme(5, 10);
    )
}
```

9. Create an adapter that will be run in the Remote Manager. Open the Adapter Factory form. In this example, the following parameters are set:
 - The Adapter Name is *remotetest*
 - The Adapter Type is *Process Task*

For this example, you need to create three variables for this adapter (based on example code in the .jar file). Click **Add**. The Java code takes two integers as arguments and the IT resource as the third variable.
10. In the first variable, the following parameters are set:
 - The Variable name is *var1*
 - The Variable type is *Integer*
 - The Map To option is set to *Resolve at Runtime*
11. Create the second variable in the same way you did the first. The following parameters are set:
 - The Variable name is *var2*
 - The Variable type is *Integer*
 - The Map To option is set to *Resolve at Runtime*
12. Create the third variable for IT Resource. The parameters are set as follows:
 - The Variable name is *ITRes*
 - The Variable type is *ITResource*
 - The Map To option is set to *Resolve at Runtime*
 - The Resource Type is *MyObjServer*

Note: The Resource Type field must be the same "ITResource Type" created in Step 5 and not Remote Manager.

13. Add a New Remote Java Task. In the Adapter Factory Form, click **Add**. Ensure that the Functional Task option is active. Select the **Remote** option. Click **Continue**.
14. The Object Instance Selection dialog box is displayed. Create a new Object Instance. Ensure that the New Object Instance option is active. Click **Continue**.
15. The Remote window is displayed. In this example, the following parameters are set:
 - The Task Name is *remote*
 - The API Source references the .jar file in the JavaTask folder
 - The Application API is *Testme.test*
 - The Constructor is set to *0 public testme.test ()*
 - The Method is set to *testme.test.addme (int, int)*After clicking the Save icon, the IT Resource is automatically added as an argument. The Application Method Parameters are ready for mapping.
16. Begin mapping the parameters by highlighting the first item in the Parameter Data Mapping list. This output parameter is an integer. The following mapping is set:
 - The Map To pull-down option is *Adapter Variables*
 - The Name pull-down option is *Return variable*
17. Click **Set**.
18. Highlight the second parameter to map. This input parameter is an integer. The following mapping is set:
 - The Map To pull-down option is *Adapter Variables*
 - The Name pull-down option is *var1*
19. Click **Set**
20. Highlight the third parameter to map. This input parameter is an integer. The following mapping is set:
 - The Map To pull-down option is *Adapter Variables*
 - The Name pull-down option is *var2*
21. Click **Set**
22. Highlight the final parameter to map. Map this ITResource to the variable passed as input to the adapter. The following mapping is set:
 - The Map To pull-down option is *Adapter Variables*
 - The Name pull-down option is *ITRes*
23. Click **Set**.
24. Click **Set**. Then click **Save**. The Adapter Factory form is displayed.
25. Compile the adapter by clicking **Build**.

To invoke the adapter, you need to create a provisioning process that calls this adapter as one task. To do this:

1. Open the Process Definition Form. In this example, the following parameters are set:
 - The Name field is *MyObjProv*
 - The Type field is *Provisioning*
 - The Object name is *MyObj*

The following checkboxes are enabled:

 - Default Process
 - Auto Pre-populate
 - Auto Save Form
2. Click the **Save** icon. The provisioning tasks automatically appear in the Tasks tab.
3. Click **Add** to create a new task. In this example, the parameters are set:
 - The Task Name field is *Call Remote Adapter*
 - The Task Description field explains the task's function
4. Click the **Save** icon. Then click the **Integration** tab. Next, click **Add** to add an adapter to this task. The Handler Type window is displayed.
5. Enable the **Adapter** option and select the adapter to be executed.
6. Click the **Save** icon. In the Integration tab, the adapter name appears in the Name field. The Status field shows that the Mapping is incomplete. The Adapter Variables pane shows the variables are not mapped.
7. Select the first variable, Adapter return value, then click **Map**. The Edit Data Mapping for Variable window is displayed. The parameters are set to:
 - The Data Type is automatically set to *Object*
 - The Map To pull-down option is *Response Code*
8. Select the second variable, *var1* then click **Map**. The **Edit Data Mapping for Variable** window appears. The parameters are set to:
 - The Data Type is automatically set to *Integer*
 - The Map To pull-down option is *Literal*
 - The Qualifier pull-down option is *Integer*
 - The Literal Value is *10*
9. Select the third variable, *var2*, then click **Map**. The Edit Data Mapping for Variable window is displayed. The parameters are set to:
 - The Data Type is automatically set to *Integer*
 - The Map To pull-down option is *Literal*
 - The Qualifier pull-down option is *Integer*
 - The Literal Value is *20*
10. Select the fourth variable, *ITRes*, and then click **Map**. The Edit Data Mapping for Variable window is displayed. The parameters are set to:
 - The Data Type is automatically set to *IT Resource (MyObjServer)*

- The Map To pull-down option is *IT Resource*
 - The Qualifier pull-down option is *MyObjServerInstance*
11. Click the **Responses** tab of the Editing Task window. Click **Add** to add the possible responses from the adapter. In this example, the only possible response is 30.
 12. Click the **Task to Object Status Mapping** tab. In the Completed category, set **Object Status** to Provisioned.
 13. At this point, you are ready to directly provision a user with the newly created resource to test the execution of the remote task. However, you must first ensure that the Remote Manager is running. Open the Remote Manager Form and verify that the service is available.
 14. Launch the **Oracle Identity Manager Admin Console** (Web client) and login as Administrator. Navigate to **Users, Manage** and select a user to provision this resource (*MyObj*). The User Detail page appears with the selected user. In the View Additional Details About This User pull-down option, select **Resource Profile**.
 15. The User Detail, Resource Profile page is displayed. Click **Provision New Resource** and select the newly created resource (*MyObj*).
 16. The Provision Resource to User wizard is displayed. Click **Continue** to complete the provisioning process.
 17. Continue with the provisioning process until you come to the **Resource Successfully Provisioned** page is displayed.
 18. Check the Remote Manager Log file to see if the code is executed. [Example 11-1](#) shows the console where the Remote Manager has executed the code:

Example 11-1 Code Example

```
<JBoss_HOME>\xlremote><JAVA_HOME>/bin/java -cp
.\lib\xlRemoteManager.jar;.\lib\xerces.jar;.\lib\xlDataObjects
.jar;.\lib\log4
j-1.2.8.jar;.\lib\xlUtils.jar;.\lib\xlLogger.jar;.\lib\xlCrypto.
jar -Dxellerate.remote.manager.rmi.over.ssl=true
-Dxellerate.remote
.manager.ssl.port=12345
-Dxellerate.remote.manager.ssl.SSLContextAlgorithm=SSL
-Dxellerate.remote.manager.ssl.KeyManagerFactory=SunX
509 -Dxellerate.remote.manager.ssl.KeyStore=<JBoss_
HOME>\xlremote/config/.xlkeystore
-Dxellerate.remote.manager.ssl.KeyStorePasswor
d=xellerate -Dxellerate.remote.manager.ssl.KeyStoreType=JKS
-Dxellerate.remote.manager.binding.port=12346
com.thortech.xl.remotemana
ger.RemoteManager RManager
log4j:WARN No appenders could be found for logger
(XELLERATE.REMOTEMANAGER) .
log4j:WARN Please initialize the log4j system properly.
```

```
Thu, 2005 Feb 24, 15:10:58.775 WARN XL.SysOutErrLogFile
property was not specified or has no value. System.out and
System.err st
```

```
reams will be redirected to the file System.Out.Err.log in
<JBOSS_HOME>\xlremote\log
```

```
Thu, 2005 Feb 24, 15:10:58.775 WARN Unable to
open/create/write to null Check if the file name specified as
the value of XL.SysO
```

```
utErrLogFile property is valid and if it has the necessary
permissions.
```

```
Thu, 2005 Feb 24, 15:10:58.775 WARN Unable to redirect
System.out and System.err. Check if the file name/path is
correct
```

```
Watching for a file that doesn't exist Will report when the file
is found.
```

```
SSL Enabled : trueBinding Remote Manager to:
rmi://w2kevandanwkstn:12346/RManager ...DONE10+20=30
```

Note: The last line of this log file shows that the two input integers are added to equal 30. This indicates that the code executed correctly and that the resource object was provisioned.

This appendix describes in detail about the tables. It contains the following topics:

- ["Class Files"](#) on page A-1
- ["Naming Conventions"](#) on page A-1
- ["Adapter Task Mapping Information"](#) on page A-2
- ["Adapter Variable Mapping Information"](#) on page A-5

Class Files

To learn more about the class files that contains the xlUtils.jar, xlAPI.jar, xlIntegration.jar, and rt.jar files, refer to the *Java API Documentation*.

Naming Conventions

This table lists both the Oracle Identity Manager actions, and the conditions and results of these actions.

Action	Condition	Result
A user is deleted.	Oracle Identity Manager checks each process for any tasks that display the Revoke selection in the Task Effect combo box.	If so, then the condition for this task has been met (the user has been revoked), and Oracle Identity Manager inserts the task into the existing process. If the task has an adapter attached to it, it will be executed.
A user is disabled.	Oracle Identity Manager checks each process for any tasks that display the Disable selection in the Task Effect combo box.	If so, then the condition for this task has been met (the user has been disabled), and Oracle Identity Manager inserts the task into the existing process. If the task has an adapter attached to it, it will be executed.
A user is enabled.	Oracle Identity Manager checks each process for any tasks that display the Enable selection in the Task Effect combo box.	If so, then the condition for this task has been met (the user has been enabled), and Oracle Identity Manager inserts the task into the existing process. If the task has an adapter attached to it, it will be executed.

Action	Condition	Result
A user's password has been modified on the Users form	Oracle Identity Manager checks each process to see if it has a Change User Password task.	If so, then the condition for this task has been met (the user's password has been modified), and Oracle Identity Manager inserts the task into all existing processes, which have that task defined. If the task has an adapter attached to it, then it will be executed.
The data fields of an application process form have been modified.	Oracle Identity Manager checks each process to see if it has a task that starts with the <field label>Updated naming convention (for example, HomeDirectory Updated).	The condition for this task is met (the process task begins with the <field label>updated naming convention). Oracle Identity Manager inserts the task into all existing processes, which have that task defined. If the task has an adapter attached to it, then it will be executed.
A user's profile information has been moved to a different organization.	Oracle Identity Manager checks each process to see if it has a task that begins with the words Move User.	The condition for this task is met (the user's profile information has been moved to a different organization). Oracle Identity Manager inserts the task into the existing process. If the task has an adapter attached to it, then it will be executed.

Adapter Task Mapping Information

The following section lists the mappings that you can set for the parameters of an adapter task.

Adapter Variables

The following table lists and describes the items of the Map To and Name combo boxes to which you can map the parameters of an adapter variable for an adapter task.

Map To Combo Box	Name Combo Box	Description
Adapter Variables	A list of adapter variables appears	<p>You can map the parameter to the adapter variables that you created for this adapter.</p> <p>Note: When the adapter variable's classification type is Object, it cannot be used with process task adapters.</p> <p>Note: If the adapter variable's classification type is IT Resource, an Attribute combo box appears. From this combo box, select the attribute of the IT resource to which you wish to map the parameter.</p>

Adapter Task

The following table lists and describes the items of the Map To, Name, and Output combo boxes to which you can map the parameters of an adapter task.

Map To Combo Box	Name Combo Box	Output combo Box	Description
Adapter Task	A list of adapter tasks appears.	A list of output variables pertaining to the selected adapter task appears.	You can map the parameter to the adapter tasks that you created for this adapter.

Literal

The following table lists and describes the items of the Map To and Type combo boxes, as well as the Value text box, to which you can map the parameters of a constant (or literal) for an adapter task.

Map To Combo Box	Type Combo Box	Value Text Box	Description
Literal	String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	Enter the value of the literal into this text box.	You can map the parameter to a String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, or Double data type, respectively.

Adapter References

The following table lists and describes the items of the Map To and Type combo boxes to which you can map the parameters of an adapter reference for an adapter task.

Map To Combo Box	Type Combo Box	Description
Adapter References	Event Handler Name or Database Reference	You can map the parameter to the active adapter.

Organization Definition

The following table lists and describes the items of the Map To and Field combo boxes to which you can map the parameters of an organization definition for an adapter task.

Map To combo box	Field Combo Box	Description
Organization Definition	Organization Name	You can map the parameter to the Organization Name field of the Organizations form.
	Organization Type	You can map the parameter to the Type field of the Organizations form.
	Organization ID	You can map the parameter to the Organization # field of the Organizations form.
	Organization Parent	You can map the parameter to the Parent Organization field of the Organizations form.
	Organization Status	You can map the parameter to the Status field of the Organizations form.

Map To combo box	Field Combo Box	Description
	Organization Parent ID	You can map the parameter to the parent_key field in the ACT database table.
	Any fields that appear in the User Defined Fields tab of the Organizations form.	You can map the parameter to the selected user-defined field.

Process Definition

The following table lists and describes the items of the Map To and Field combo boxes to which you can map the parameters of a process definition for an adapter task.

Map To Combo Box	Field Combo Box	Description
Process Definition	Name	You can map the parameter to the Name field of the Process Definition form.
	Type	You can map the parameter to the Type field of the Process Definition form.

User Definition

The following table lists and describes the items of the Map To and Field combo boxes to which you can map the parameters of a user definition for an adapter task.

Map To Combo Box	Field Combo Box	Description
User Definition	User Key	You can map the parameter to a key, representing a unique record of the Users form.
	First Name	You can map the parameter to the First Name field of the Users form.
	Middle Initial	You can map the parameter to the Middle Name field of the Users form.
	Last Name	You can map the parameter to the Last Name field of the Users form.
	User Login	You can map the parameter to the User ID field of the Users form.
	Password	You can map the parameter to the New Password and Confirm Password fields of the Users form.
	Type	You can map the parameter to the Xellerate Type field of the Users form.
	User Status	You can map the parameter to the Status field of the Users form.
	Role	You can map the parameter to the Role field of the Users form.
	Identity	You can map the parameter to the Identity field of the Users form.
Disabled	You can map the parameter to the Disable User check box of the Users form.	

Map To Combo Box	Field Combo Box	Description
User Definition	Organization	You can map the parameter to the Organization field of the Users form.
	Manager	You can map the parameter to the Manager field of the Users form.
	Start Date	You can map the parameter to the Start Date field of the Users form.
	End Date	You can map the parameter to the End Date field of the Users form.
	Email	You can map the parameter to the Email field of the Users form.
	Any fields that appear in the User Defined Fields tab of the Users form.	You can map the parameter to the selected user-defined field.

Adapter Variable Mapping Information

When an adapter variable is not the adapter return variable, or it is not designated as Resolve at Runtime, it should be mapped within the Variable List tab of the Adapter Factory form.

On the other hand, if the adapter variable is classified as an adapter return variable, or the adapter variable is set to Resolve at Runtime, it needs to be mapped at another location within Oracle Identity Manager. This location is contingent upon the adapter's type. For example, the variables of a process task adapter will be mapped at a different place than the variables of a pre-populate adapter.

To learn where the variables of a particular type of adapter need to be mapped, consult the table below.

Adapter Type	Location
Process Task	The Integration tab of the Editing Task window.
Task Assignment	The Assignment tab of the Editing Task window.
Rule Generator	The Map Adapters tab of the Data Object Manager form.
Pre-Populate	The Pre-Populate tab of the Form Designer form.
Entity	The Map Adapters tab of the Data Object Manager form.

Now that you know where the variables for each type of adapter need to be mapped, the following sections list the mappings that you can set for the parameters of these adapter variables.

From the Variable List Tab

Variable Type	Map To	Qualifier/Resource Type
Object	Adapter References	Database References Data Object References

Variable Type	Map To	Qualifier/Resource Type
	Set at runtime (for Task Assignment adapters only)	Database References Data Object References
IT Resource	Resolve at Runtime	The IT Resource types that appear in the Table view of the IT Resources Type Definition form
String, Character, Byte, Integer, Float, Long, Short, Double	Literal	If you are mapping the adapter variable to a literal, a Literal Value text box appears below the Resource Type combo box. Within this text box, enter the value of this literal.
	Resolve at Runtime	N/A
	Adapter References	Event Handler Name Note: If the data type of the adapter variable is not String, Adapter References cannot be selected from the Map To combo box.
Boolean	Literal	Boolean. If you select this resource type, two Literal Value options appear below the Resource Type combo box: True and False. Select the option that corresponds to the value of the adapter variable.
	Resolve at Runtime	N/A
Date	Literal	If you are mapping the adapter variable to a literal, a Literal Value Lookup field appears below the Resource Type combo box. Double-click this Lookup field. From the Date & Time window that appears, select the date and time that will be the value of this literal.
	Resolve at Runtime	N/A
	System Date	N/A Note: Since this variable's value will reflect Oracle Identity Manager's date and time, you do not need to map it.

Process Task Adapter Variable Mappings

Variable Type	Map To	Qualifier/Description
Object (Adapter Return Variable)	Process Data	You can map the parameter to a field of either the associated custom process form, or a child table that belongs to this form.
	Response Code	N/A

Variable Type	Map To	Qualifier/Description
	Task Information	<p>Note. You can map the parameter to the Note tab of the Task List form.</p> <p>Reason. You can map the parameter to the Error Details window. To access this window, double-click a task that appears within the Task List form.</p>
	Process Definition	<p>Name. You can map the parameter to the Name field of the Process Definition form.</p> <p>Type. You can map the parameter to the Type Lookup field of the Process Definition form.</p>
Object (Adapter Return Variable)	Organization Definition	<p>The fields of the Organizations form to which you can map the adapter variable.</p> <p>Note: Since the data type of the adapter variable is Object, you cannot select Organization ID and Organization Parent ID from the Qualifier combo box.</p>
	User Definition	The fields of the Users form to which you can map the adapter variable.
IT Resource	IT Resource	You can map the parameter to an IT resource. This IT resource is a member of the IT resource type that appears in parenthesis from within the Data Type field.
	Process Data	<p>You can map the parameter to a field of the associated process-specific form.</p> <p>Note: The only field names that will appear in this combo box are ones with a data type of IT Resource Lookup Field.</p>
String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	Process Data	You can map the parameter to a field of either the associated custom process form, or a child table that belongs to this form.
	Task Information	<p>Note. You can map the parameter to the Note tab of the Task List form.</p> <p>Reason. You can map the parameter to the Error Details window. To access this window, double-click a task that appears within the Task List form.</p>
	Process Definition	<p>Name. You can map the parameter to the Name field of the Process Definition form.</p> <p>Type. You can map the parameter to the Type Lookup field of the Process Definition form.</p>
	Organization Definition	The fields of the Organizations form to which you can map the adapter variable.

Variable Type	Map To	Qualifier/Description
String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	User Definition	The fields of the Users form to which you can map the adapter variable.
	Literal	<p>If you are mapping the adapter variable to a literal, and the variable's data type is String, Character, Byte, Integer, Float, Long, Short, or Double, a Literal Value text box appears below the Qualifier combo box. Within the text box, enter the value of this literal.</p> <p>When you are mapping the adapter variable to a literal, and the variable's data type is Boolean, two Literal Value options appear below the Qualifier combo box: True and False. Select the option that corresponds to the value of the adapter variable.</p> <p>If you are mapping the adapter variable to a literal, and the variable's data type is Date, a Literal Value Lookup field appears below the Qualifier combo box. Double-click this Lookup field. From the Date & Time window that appears, select the date and time that will be the value of this literal.</p>
String	IT Resources	<p>If you are mapping the adapter variable to an IT Resource, three combo boxes appear below the Map To combo box: Qualifier, IT Asset Type, and IT Asset Property. From these combo boxes, select the qualifier for the mapping, the specific name of the IT resource, and the field of the IT resource that will receive the results of the mapping.</p> <p>Note: If the data type of the adapter variable is not String, IT Resources cannot be selected from the Map To combo box.</p>

Task Assignment Adapter Variable Mappings

Variable Type	Map To	Qualifier/Description
IT Resource	Object Data	You can map the parameter to an IT resource's instance key. This IT resource is a member of the IT resource type that appears in parenthesis from within the Data Type field.
	IT Resource	You can map the parameter to an IT resource.

Variable Type	Map To	Qualifier/Description
Object (Adapter Return Value)	Object Data	You can map the parameter to a field of either the associated custom resource object form, or a child table that belongs to this form.
	Response Code	N/A
	Task Information	The fields of the Task List form to which you can map the adapter variable.
	Process Definition	The fields of the Process Definition form to which you can map the adapter variable.
	Organization Definition	The fields of the Organizations form to which you can map the adapter variable.
String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	User Definition	The fields of the Users form to which you can map the adapter variable.
	Object Data	You can map the parameter to a resource object's instance key.
	Task Information	The fields of the Task List form to which you can map the adapter variable.
	Process Definition	The fields of the Process Definition form to which you can map the adapter variable.
String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	Organization Definition	The fields of the Organizations form to which you can map the adapter variable.
	User Definition	The fields of the Users form to which you can map the adapter variable.
String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	Request Info	Request ID. You can map the parameter to the Request ID field of the Requests form. Request Action. You can map the parameter to the Request Action field of the Requests form. Request Priority. You can map the parameter to the Request Priority field of the Requests form.
	Request Target User	The fields of the Users form to which you can map the adapter variable.
	Request Target Organization	The fields of the Organizations form to which you can map the adapter variable.
	Requester Info	The fields of the Users form to which you can map the adapter variable.

Variable Type	Map To	Qualifier/Description
	Literal	<p>If you are mapping the adapter variable to a literal, a Literal Value text box appears below the Qualifier combo box. Within the text box, enter the value of this literal.</p> <p>Note: If the data type of the adapter variable is Boolean, two options appear in place of the text box: True and False. Select the option that reflects the value of the adapter variable.</p> <p>Note: If the data type of the adapter variable is Object, Literal cannot be selected from the Map To combo box.</p>
String	IT Resources	<p>Resource Instance. You can map the parameter to an IT resource's instance key. This IT resource is a member of the IT resource type that appears in parenthesis from within the Data Type field.</p> <p>IT Asset Type. You can map the parameter to an IT resource type.</p>
String	IT Resources	<p>IT Asset Property. You can map this parameter to one of the properties that comprise the selected IT resource type.</p>

Rule Generator and Entity Adapter Variable Mappings

Variable Type	Map To	Qualifier/Description
Object (Adapter Return Variable), IT Resource, String, Boolean, Character, Byte, Date, Integer, Float, Long, Short	Literal	<p>If you are mapping the adapter variable to a literal, a Literal Value text box appears below the Qualifier combo box. Within the text box, enter the value of this literal.</p> <p>Note: If the data type of the adapter variable is Object, Literal cannot be selected from the Map To combo box.</p>
	Entity Field	<p>You can map the adapter variable to a field of the associated process form. The name of this form is displayed in the Form Description field of the Data Object Manager form.</p>
	Contact	<p>The fields of the Contact Information tab to which you can map the adapter variable.</p>
	Location	<p>The fields of the Location form to which you can map the adapter variable.</p>

Variable Type	Map To	Qualifier/Description
	Organization Definition	The fields of the Organizations form to which you can map the adapter variable. Note: If the data type of the adapter variable is not Object, you cannot select Organization ID and Organization Parent ID from the Qualifier combo box.
	User Definition	The fields of the Users form to which you can map the adapter variable.

Pre-Populate Adapter Variable Mappings

Variable Type	Map To	Qualifier/Description
IT Resource	IT Resource	You can map the parameter to an IT resource. This IT resource is a member of the IT resource type that appears in parenthesis from within the Data Type field.
	Process Data	You can map the parameter to a field of the associated process-specific form. Note: The only field names that will appear in this combo box are ones with a data type of IT Resource Lookup Field.
String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	Process Data	You can map the parameter to a field of the associated process-specific form.
	Organization Definition	The fields of the Organizations form to which you can map the adapter variable.
	User Definition	The fields of the Users form to which you can map the adapter variable.
	Location	The fields of the Location form to which you can map the adapter variable.
	Contact	The fields of the Contact Information tab to which you can map the adapter variable.

Variable Type	Map To	Qualifier/Description
String, Boolean, Character, Byte, Date, Integer, Float, Long, Short, Double	Literal	<p>If you are mapping the adapter variable to a literal, and the variable's data type is String, Character, Byte, Integer, Float, Long, Short, or Double, a Literal Value text box appears below the Qualifier combo box. Within the text box, enter the value of this literal.</p> <p>When you are mapping the adapter variable to a literal, and the variable's data type is Boolean, two Literal Value options appear below the Qualifier combo box: True and False. Select the option that corresponds to the value of the adapter variable.</p> <p>If you are mapping the adapter variable to a literal, and the variable's data type is Date, a Literal Value Lookup field appears below the Qualifier combo box. Double-click this Lookup field. From the Date & Time window that appears, select the date and time that will be the value of this literal.</p>
String	IT Resources	<p>If you are mapping the adapter variable to an IT Resource, three combo boxes appear below the Map To combo box: Qualifier, IT Asset Type, and IT Asset Property. From these combo boxes, select the qualifier for the mapping, the specific name of the IT resource, and the field of the IT resource that will receive the results of the mapping.</p> <p>Note: If the data type of the adapter variable is not String, IT Resources cannot be selected from the Map To combo box</p>

Index

A

Adapter Variables, 3-3
Adapters, 1-2, 3-1
Adding the Trust Relation, 2-2
Attaching Pre-Populate Adapters, 7-2
Attaching Process Task Adapters to Process Tasks, 4-3
Attaching Task Assignment Adapters to Process Tasks, 5-2

C

Compiling Adapters, 9-1
Create a Java Task, 3-7
Create a Logic Task, 3-21
Create a Remote Task, 3-11
Create a Response, 3-26
Create a Set Variable Task, 3-19
Create a Stored Procedure Task, 3-12
Create a Utility Task, 3-15
Create an Oracle Identity Manager API Task, 3-17
Creating Adapter Tasks, 3-5

D

Delete a Response, 3-27
Delete an Adapter Variable, 3-5
Deleting Adapter Tasks, 3-25
Disabling Adapters, 3-3

E

Entity Adapters, 3-28
Execution Schedule, 1-3
Exporting and Importing Adapters, 10-1

H

How a Process Task Adapter Works, 4-2

I

Installing the Remote Manager, 2-2
Integration Problem, 1-1

M

Managing Entities, 8-1
Mapping Rule Generator Adapter Variables, 6-2
Modify a Response, 3-27
Modify an Adapter Variable, 3-5
Modifying Adapter Tasks, 3-24

O

Oracle Identity Manager Solution, 1-1

P

Post-install Configuration, 11-1
Pre-Populate Adapters, 7-1

R

Removing Process Task Adapters from Process Tasks, 4-6
Removing Rule Generators from Form Fields, 6-5
Removing Task Assignment Adapters from Process Tasks, 5-5
Resources, 1-4
Responses, 1-4

S

Scheduling Rule Generators, 3-28

T

Tables, A-1
Tabs of the Adapter Factory Form, 1-3

U

Usage Lookup, 1-4

V

Variable List, 1-4

W

Working with Responses, 3-26

