

Oracle® Fusion Middleware

Application Adapter for PeopleSoft User's Guide for Oracle
WebLogic Server

11g Release 1 (11.1.1)

E17055-01

April 2010

Oracle Fusion Middleware Application Adapter for PeopleSoft User's Guide for Oracle WebLogic Server, 11g Release 1 (11.1.1)

E17055-01

Copyright © 2009, 2010, Oracle and/or its affiliates. All rights reserved.

Primary Author: Stefan Kostial

Contributing Authors: Sunil Gopal, Marian Jones, Vikas Anand, Sunil Wadhwa, Vishal Saxena, Vimmika Dinesh

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. 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, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

Contents

Preface	vii
Audience	vii
Documentation Accessibility	vii
Related Documents	viii
Conventions	viii
1 Introduction	
Adapter Features	1-1
PeopleSoft Concepts	1-2
PeopleSoft Component Interface	1-2
PeopleSoft Application Messaging / Integration Broker	1-2
Integration with PeopleSoft	1-2
Adapter Architecture	1-3
BSE Versus Oracle Adapter J2CA Deployment	1-5
2 Configuring Oracle Application Adapter for PeopleSoft	
Starting Application Explorer	2-1
Configuring Repository Settings	2-2
Configuring the Database Repository for BSE	2-2
Configuring BSE System Settings	2-4
Configuring the Database Repository for J2CA	2-6
Password Encryption	2-8
Creating a Repository Configuration	2-8
Creating a Configuration for BSE	2-8
Creating a Configuration for J2CA	2-9
Connecting to a BSE or J2CA Configuration	2-11
Establishing a Connection (Target) for PeopleSoft	2-11
Defining a Target to PeopleSoft	2-11
Connecting to a Defined PeopleSoft Target	2-13
Managing a Target	2-13
Viewing Application System Objects	2-15
Creating XML Schemas	2-15
Generating a WSDL (J2CA Configurations Only)	2-16
Creating and Testing a Web Service (BSE Configurations Only)	2-17
Configuring an Event Adapter	2-19

Creating and Editing a Channel.....	2-19
Creating an HTTP Channel	2-20
Creating a TCP Channel.....	2-22
Creating a File Channel.....	2-25
Editing a Channel.....	2-27
Deleting a Channel	2-27
Schema Validation	2-27
3 Oracle WebLogic Server Deployment and Integration	
Adapter Integration with Oracle WebLogic Server	3-1
Deployment of Adapter	3-1
Updating Adapter Configuration.....	3-2
4 Integration With BPEL Service Components in the Oracle SOA Suite	
Overview.....	4-1
Deployment of Adapter	4-1
Configuring a New Application Server Connection.....	4-2
Designing an Outbound BPEL Process for Service Integration.....	4-6
Generating WSDL for Request/Response Service	4-7
Creating a New SOA Application for the Outbound BPEL Process	4-7
Defining a BPEL Outbound Process.....	4-10
Deploying the BPEL Outbound Process	4-28
Invoking the Input XML Document in the Oracle Enterprise Manager Console.....	4-31
Testing Outbound BPEL and Mediator Processes	4-34
Designing an Inbound BPEL Process for Event Integration.....	4-34
Generating WSDL for Event Integration	4-34
Creating a New SOA Application for the Inbound BPEL Process	4-38
Defining a BPEL Inbound Process.....	4-40
Deploying the BPEL Inbound Process	4-48
Triggering an Event in PeopleSoft.....	4-51
5 Integration With Mediator Service Components in the Oracle SOA Suite	
Configuring a New Application Server Connection.....	5-2
Configuring a Mediator Outbound Process.....	5-6
Creating a New SOA Application for the Outbound Mediator Process.....	5-7
Defining a Mediator Outbound Process	5-9
Deploying the Mediator Outbound Process	5-21
Invoking the Input XML Document in the Oracle Enterprise Manager Console.....	5-24
Configuring a Mediator Inbound Process.....	5-27
Creating a New SOA Application for the Inbound Mediator Process.....	5-27
Defining a Mediator Inbound Process	5-29
6 Troubleshooting and Error Messages	
Troubleshooting.....	6-1
BSE Error Messages	6-5
General Error Handling in BSE.....	6-5

Adapter-Specific Error Handling.....	6-5
7 Advanced User Tools	
Web Services Policy-Based Security	7-1
Configuring Web Services Policy-Based Security	7-2
Migrating Repositories.....	7-8
A Generating Component Interface APIs	
Building the PeopleSoft API Java Programs.....	A-1
Compiling the PeopleSoft API Java Programs.....	A-4
B Configuring the PeopleSoft Message Router	
Configuring the TCP/IP or HTTP Target Connector for PeopleSoft 8.4.....	B-1
Configuring the TCP/IP Target Connector.....	B-2
Configuring the Node for the TCP/IP84 Connector.....	B-3
Configuring the HTTP Target Connector.....	B-6
Configuring the HTTP Connector	B-8
Configuring the TCP/IP Handler for PeopleSoft 8.1.....	B-12
Testing Your PeopleSoft Configuration.....	B-14
C Using Component Interfaces	
Creating a Component Interface	C-1
Working With Properties	C-1
Creating a New Component Interface	C-1
Viewing or Changing Available Methods.....	C-4
Securing a Component Interface.....	C-5
Configuring Component Interface Security for PeopleSoft Version 8.1x.....	C-5
Configuring Component Interface Security for PeopleSoft Version 8.4 or Higher.....	C-8
Testing a Component Interface	C-12
D Using PeopleSoft Integration Broker	
PeopleSoft Integration Broker.....	D-1
Configuring Integration Broker in PeopleSoft 8.4.....	D-2
Configuring Application Messaging in PeopleSoft Release 8.1	D-11
Viewing the PeopleCode for a Message	D-20
Testing the Integration Broker.....	D-21
Using Outbound Synchronous Messages	D-23
Configuring Outbound Synchronous Messages	D-23
Viewing the PeopleCode for a Synchronous Message	D-25
Generating Events Using PeopleTools Tutorial.....	D-26
Configuring PeopleSoft Services.....	D-26
Triggering the Event in PeopleSoft.....	D-34
Verifying the Event Results	D-35

Glossary

Index

Preface

Oracle Fusion Middleware Application Adapter for PeopleSoft User's Guide for Oracle WebLogic Server describes how to provide connectivity and integrate with PeopleSoft systems.

Audience

Oracle Fusion Middleware Application Adapter for PeopleSoft User's Guide for Oracle WebLogic Server is intended for those who integrate with PeopleSoft systems and develop applications.

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible to all users, including users that are disabled. 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.

Access to Oracle Support

Oracle customers have access to electronic support through My Oracle Support. For information, visit <http://www.oracle.com/support/contact.html> or visit <http://www.oracle.com/accessibility/support.html> if you are hearing impaired.

Related Documents

For more information, see the following documents in the Oracle Enterprise Repository 11g Release 1 (11.1.1) documentation set:

- *Oracle Fusion Middleware Application Adapters Installation Guide for Oracle WebLogic Server*
- *Oracle Fusion Middleware Application Adapter Upgrade Guide for Oracle WebLogic Server*
- *Oracle Fusion Middleware Application Adapter Best Practices Guide for Oracle WebLogic Server*
- Oracle's Unified Method (OUM)

A wealth of additional Governance information can be found within Oracle's Unified Method (OUM). OUM can be used by Oracle employees, Oracle Partner Network Certified Partners or Certified Advantage Partners, and Clients who either participate in the OUM Customer Program or are engaged on projects where Oracle provides consulting services. OUM is a web-deployed toolkit for planning, executing and controlling software development and implementation projects.

For more information about OUM, see the OUM FAQ at

http://my.oracle.com/portal/page/myo/ROOTCORNER/KNOWLEDGEAREAS1/BUSINESS_PRACTICE/Methods/Learn_about_OUM.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.
<code>monospace</code>	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Introduction

Oracle WebLogic Server connects to a PeopleSoft system through Oracle Application Adapter for PeopleSoft, which provides connectivity and carries out interactions on a PeopleSoft system. This chapter discusses the following topics:

- [Adapter Features](#)
- [PeopleSoft Concepts](#)
- [Integration with PeopleSoft](#)
- [Adapter Architecture](#)
- [BSE Versus Oracle Adapter J2CA Deployment](#)

1.1 Adapter Features

Oracle Application Adapter for PeopleSoft provides a means to exchange real-time business data between PeopleSoft systems and other applications, databases, and external business partner systems. The adapter enables inbound and outbound processing with PeopleSoft. Oracle Application Adapter for PeopleSoft can be deployed as a J2EE Connector Architecture (J2CA) version 1.0 resource adapter. This deployment is referred to as Oracle Adapter J2CA. It can also be deployed as a Web services servlet and is referred to as Oracle Adapter Business Services Engine (BSE).

Oracle Application Adapter for PeopleSoft uses XML messages to enable non-PeopleSoft applications to communicate and exchange transactions with PeopleSoft using services and events. The roles of services and events are outlined as follows:

- **Services:** Enable applications to initiate a PeopleSoft business event.
- **Events:** Enable applications to access PeopleSoft data only when a PeopleSoft business event occurs.

To support event functionality, channels are supported. A **channel** represents configured connections to particular instances of back-end or other types of systems.

The channel is the adapter component that receives events in real time from the enterprise information system (EIS) application. The channel component can be a File reader, an HTTP listener, a TCP/IP listener, or an FTP listener. A channel is always EIS specific. The adapter supports multiple channels for a particular EIS, which enables the user to choose the optimal channel component based on deployment requirements.

Oracle Application Adapter for PeopleSoft provides:

- Synchronous and asynchronous, bidirectional message interactions for PeopleSoft component interfaces and PeopleSoft Application Messaging / Integration Broker.
- Oracle WebLogic Server Adapter Application Explorer (Application Explorer), a GUI tool that explores PeopleSoft metadata and builds XML schemas or Web services.
- XML schemas and WSDL files for the J2CA 1.0 and J2CA 1.5 resource adapter.
- Web services for BSE.

The adapter connects to the PeopleSoft Application Server by accessing APIs for the component interfaces that correspond to its supported business objects. Every component interface contains data and business logic for the business component, thus alleviating a requirement for the adapter to duplicate the processes defined within the business component.

See Also: *Oracle Application Server Adapter Concepts Guide*

1.2 PeopleSoft Concepts

PeopleSoft enables integration with other applications and systems through its component interface framework and its Integration Broker (in release 8.4) or Application Messaging (in release 8.1) facility. Oracle Application Adapter for PeopleSoft uses the PeopleSoft framework and leverages various integration access methods to provide the greatest amount of flexibility and functionality. Integration access methods supported by Oracle Application Adapter for PeopleSoft include:

- PeopleSoft Java API using component interfaces
- PeopleSoft messages using PeopleSoft Application Messaging / Integration Broker

1.2.1 PeopleSoft Component Interface

In the PeopleSoft environment, a component interface is a container for distributing PeopleSoft application data among PeopleSoft logical systems and for exchanging PeopleSoft application data with non-PeopleSoft systems.

The component interface is based on an existing business process within PeopleSoft. An example is a purchase order entry, which can be a PeopleSoft-delivered process or a user-developed process. The component interface also inherits its methods (Add, Update, and so on) and its business logic from the underlying business process.

PeopleSoft delivers generic component interfaces with each of its applications. These are called Enterprise Integration Points (EIP). Customers can also develop their own custom component interfaces, or they can modify EIP as required. Oracle Application Adapter for PeopleSoft supports both types of component interfaces.

1.2.2 PeopleSoft Application Messaging / Integration Broker

PeopleSoft Application Messaging / Integration Broker facilitates the integration of PeopleSoft XML with PeopleSoft. Oracle Application Adapter for PeopleSoft provides a handler that must be configured within the PeopleSoft application gateway using TCP/IP transport services.

1.3 Integration with PeopleSoft

Oracle Application Adapter for PeopleSoft enables you to:

- Access a PeopleSoft component using a component interface. Component methods are referred to as services.
- Receive messages from a PeopleSoft application. Messages are referred to as events.

When you access a PeopleSoft component from another application, you work with:

- Component interfaces. If a component interface does not exist, create, secure, and test one. If the component interface exists but you modified it, secure and test it. For more information, see your PeopleSoft documentation. Alternatively, you can secure and test the component interface and create the component interface API after you generate schemas or Web services.
- Component interface APIs. Create an API for the component interface.
- Schemas and Web services. Create schemas or Web services for the component methods.

See [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#) for more information.

To receive a message from PeopleSoft, you work with:

- The Integration environment. Configure and test your PeopleSoft Integration Broker (release 8.4) or Application Messaging environment (release 8.1). To ensure that the environment is properly configured, see your PeopleSoft documentation.
- Message routing. Configure TCP/IP Target Connector (release 8.4), HTTP Target Connector (release 8.4), or TCP/IP Handler (release 8.1).

See Also:

- *Oracle Application Server Adapter Concepts Guide*
- *Oracle Fusion Middleware Application Adapters Installation Guide for Oracle WebLogic Server*

1.4 Adapter Architecture

Oracle Application Adapter for PeopleSoft uses Application Explorer with one of the following components:

- Oracle WebLogic Server Adapter Business Services Engine(BSE)
- Enterprise Connector for J2EE Connector Architecture (J2CA)

Application Explorer (used to configure PeopleSoft connections and create Web services and events) can be configured to work in a Web services environment with BSE. When working in a J2CA environment, the connector uses the Common Client Interface (CCI) to provide integration services using adapters instead of Web services.

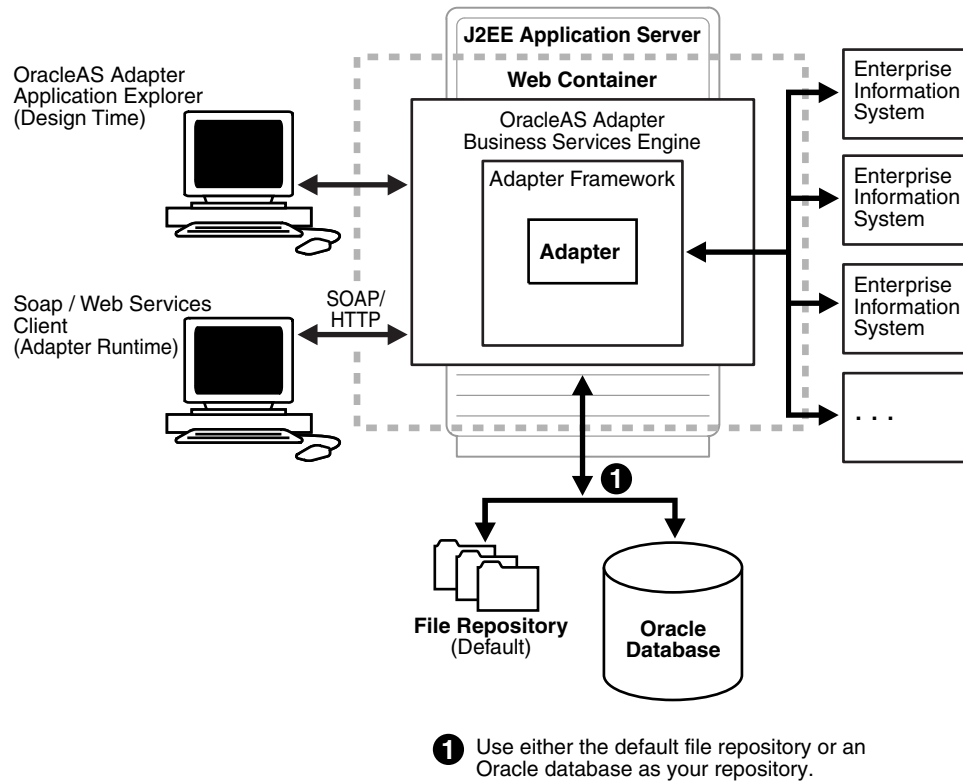
Oracle WebLogic Server Adapter Business Services Engine (BSE) Architecture

[Figure 1–1](#) shows the generic architecture for the Oracle Web service adapter for packaged applications. The adapter works with BSE, as deployed to a Web container in a J2EE application server.

Application Explorer, a design-time tool deployed along with BSE, is used to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. Metadata created while you perform these operations are stored in the repository by BSE.

BSE uses SOAP as a protocol for receiving requests from clients, interacting with the EIS, and sending responses from the EIS back to clients.

Figure 1–1 Oracle WebLogic Server Adapter Business Services Engine (BSE) Generic Architecture



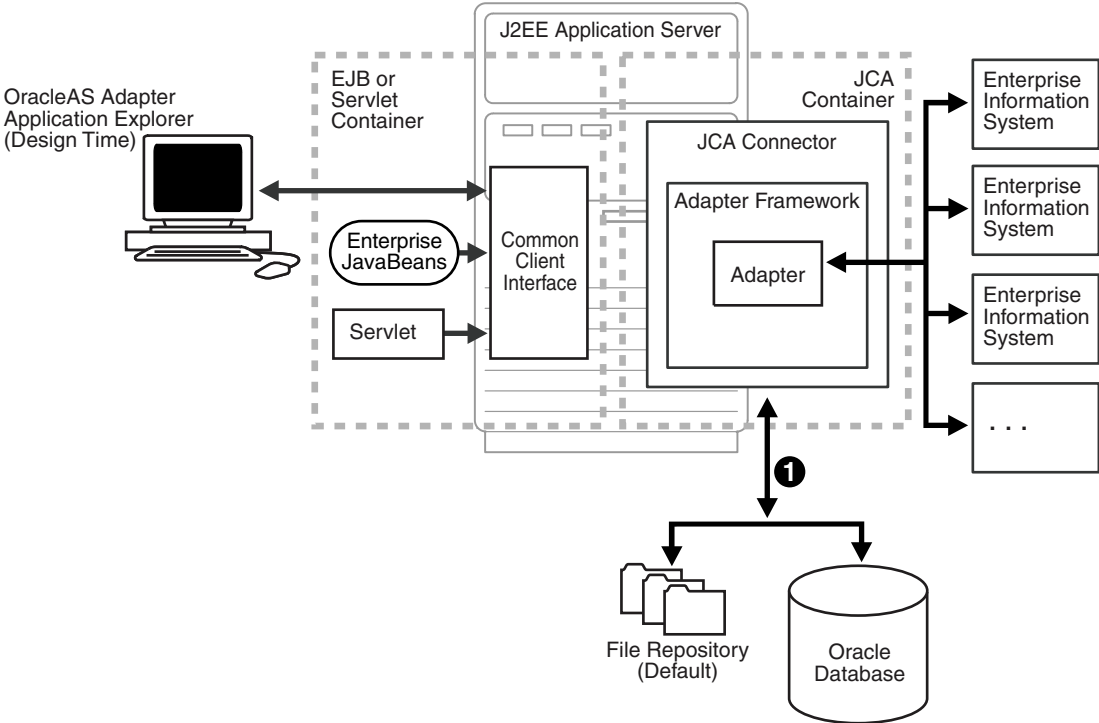
Note: Do not use a file repository for BSE in production environments.

Oracle WebLogic Server Adapter J2CA Architecture

Figure 1–2 shows the generic architecture for Oracle Adapter J2CA for packaged applications. Oracle Adapter J2CA is deployed to a standard J2CA Container and serves as a host container to the adapters. The connector is configured with a repository. The repository can be a file system or an Oracle database. It is deployed as a RAR file and has an associated deployment descriptor called `ra.xml`. You can create multiple connector factories by editing the Oracle WebLogic Server deployment descriptor `ra.xml`. See Chapter 3, "Oracle WebLogic Server Deployment and Integration" for more information.

Application Explorer, a design tool that works with the connector, is used to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. Metadata created while you perform these operations are stored in the repository by the connector.

Figure 1-2 Oracle WebLogic Server Adapter J2CA Generic Architecture



1 Use either the default file repository or an Oracle database as your repository.

1.5 BSE Versus Oracle Adapter J2CA Deployment

If you are using Oracle Application Adapter for PeopleSoft with Business Process Execution Language (BPEL) Process Manager, please note that:

- Only Oracle Adapter J2CA deployment supports inbound integration (event notification) with BPEL.
- Both Oracle Adapter J2CA and BSE deployments support outbound integration (request-response service) with BPEL.

The following three factors explain the differences between deploying the BSE and Oracle Adapter J2CA. Understanding the factors can help in selecting a deployment option.

1. BSE is the preferred deployment option because it:
 - Can be deployed in a separate instance of Oracle WebLogic Server.
 - Provides better distribution of load.
 - Provides better isolation from any errors from third party libraries.
 - Provides better capability to isolate issues for debugging purposes.
 - Conforms more closely to the Service Oriented Architecture (SOA) model for building applications.

2. Oracle Adapter J2CA provides slightly better performance.

Oracle Adapter J2CA does provide slightly better performance than BSE. However, the difference decreases as the transaction rate increases.

3. Oracle Adapter J2CA and the BSE option both provide identity propagation at run-time.

The BSE option provides the capability to pass identity using the SOAP header. For Oracle Adapter J2CA, user name and password can be passed using the connection specification of the CCI.

Configuring Oracle Application Adapter for PeopleSoft

Oracle Adapter Application Explorer (Application Explorer) enables the processing of Component Interfaces and Messages.

External applications that access PeopleSoft through the adapter use either XML schemas or Web services to pass data between the external application and the adapter. You can use Application Explorer to create the required XML schemas and Web services.

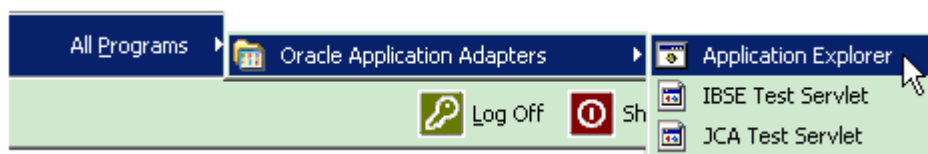
This chapter discusses the following topics:

- [Starting Application Explorer](#)
- [Configuring Repository Settings](#)
- [Creating a Repository Configuration](#)
- [Establishing a Connection \(Target\) for PeopleSoft](#)
- [Viewing Application System Objects](#)
- [Creating XML Schemas](#)
- [Generating a WSDL \(J2CA Configurations Only\)](#)
- [Creating and Testing a Web Service \(BSE Configurations Only\)](#)
- [Configuring an Event Adapter](#)

2.1 Starting Application Explorer

To start Application Explorer:

1. Ensure that Oracle WebLogic Server is started, which is where Application Explorer is deployed.
2. Start Application Explorer by clicking the Windows **Start** menu, selecting **All Programs, Oracle Application Adapters**, and clicking **Application Explorer**.



You can also start Application Explorer by executing the `ae.bat` file, which is located in the following directory:

```
C:\oracle\Middleware\home_0309\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\tools\iwae\bin\ae.bat
```

It is a good practice to create a shortcut for the **ae.bat** file on your desktop.

If you are using a UNIX or Linux platform you can start Application Explorer by executing the **iwae.sh** file.

Application Explorer starts. You are ready to define new targets to your PeopleSoft system.

2.2 Configuring Repository Settings

You need not configure BSE for a file-based repository because it is configured during the installation.

2.2.1 Configuring the Database Repository for BSE

After BSE is deployed to a managed Oracle WebLogic server (for example, `soa_server1`), you can configure it through the BSE configuration page. This configuration is required only when using a database repository with BSE.

Note: Do not use a file repository for BSE in production environments.

To configure BSE:

1. Execute the `iwse.ora` SQL script on the machine where the database is installed.

The `iwse.ora` SQL script is located in the following directory:

```
C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\etc
```

This script creates the required tables that are used to store the adapter configuration information in the database. These tables are used by Application Explorer and by adapters during design time and runtime. It is recommended that you use the same credentials to create the database repository and also in the `web.xml` file for database user credentials.

```
C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\etc>sqlplus
```

```
SQL*Plus: Release 10.1.0.2.0 - Production on Tue Dec 27 18:10:44 2005
Copyright (c) 1982, 2004, Oracle. All rights reserved.
```

```
Enter user-name: scott
Enter password: scott1
```

```
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.1.2.0 - Production
With the Partitioning, OLAP and Data Mining options
```

```
SQL>@ iwse.ora
```

2. Open the following page in your browser:

```
http://host name:port/ibse
```

Where *host name* is the system where BSE is installed and *port* is the HTTP port for a managed Oracle WebLogic server (for example, `soa_server1`).

For example,

`http://localhost:8001/ibse`

Note: If you are accessing this page for the first time, it may take longer to load.

3. Log on when prompted.

Enter the user ID and password, for example:

- User name: weblogic
- Password: welcome1

The BSE configuration page is displayed.

Property Name	Property Value
System	
Language	English ▼
Adapter Lib Directory	../../ApplicationAdapters/lib
Encoding	UTF-8 ▼
Debug Level	DEBUG ▼
Number of Async. Processors	0 ▼
Repository	
Repository Type	File System ▼
Repository Url	file://C:\oracle\Middleware\Oracle_SO
Repository Driver	
Repository User	
Repository Password	
Repository Pooling	<input type="checkbox"/>

4. The `ojdbc14.jar` file must be copied to the following directory:

`C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\lib`

5. Ensure that the Adapter Lib Directory parameter specifies the path to the lib directory, for example:

`C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\lib`

After you specify the path, adapters in the lib directory are available to BSE.

Note: The Repository URL field specifies where the file system repository is located. To use a database repository, you must enter the repository connection information. For the initial verification, use a file system repository.

6. Click **Save**.

2.2.1.1 Configuring BSE System Settings

To configure BSE system settings:

1. Display the **BSE configuration** page in a browser:

`http://host name:port/ibse/IBSEConfig`

Where *host name* is the system where BSE is installed and *port* is the port number on which BSE is listening.

Note: The server to which BSE is deployed must be running.

The BSE settings pane is displayed, as shown in the following image.

Property Name	Property Value
System	
Language	English ▾
Adapter Lib Directory	../ApplicationAdapters/lib
Encoding	UTF-8 ▾
Debug Level	DEBUG ▾
Number of Async. Processors	0 ▾

2. Configure the system settings.

The following table lists the parameters with descriptions of the information to provide.

Parameter	Description
Language	Specify the required language.
Adapter Lib Directory	Enter the full path to the directory where the adapter jar files reside.
Encoding	Only UTF-8 is supported.
Debug Level	Specify the debug level from one of the following options: <ul style="list-style-type: none"> ■ None ■ Fatal ■ Error ■ Warning ■ Info ■ Debug
Number of Async Processors	Select the number of asynchronous processors.

The following image shows all fields and check boxes for the Repository pane.

The screenshot shows a configuration window titled "Repository". It contains the following fields and controls:

- Repository Type:** A dropdown menu currently set to "File System".
- Repository Url:** A text input field containing the value "file://C:\oracle\Middleware\Oracle_SC".
- Repository Driver:** An empty text input field.
- Repository User:** An empty text input field.
- Repository Password:** An empty text input field.
- Repository Pooling:** A checkbox that is currently unchecked.
- Save:** A grey button located at the bottom right of the pane.

3. Configure the repository settings.

BSE requires a repository to store transactions and metadata required for the delivery of Web services.

See "[Configuring a File System Repository](#)" on page 2-6 for more information.

The following table lists the parameters with descriptions of the information to provide.

Parameter	Description
Repository Type	Select one of the following repositories from the list: <ul style="list-style-type: none"> ■ Oracle ■ File (Do not use for BSE in production environments.)
Repository URL	Enter the JDBC URL to use when opening a connection to the database. For example, the following repository URL format is used when connecting to Oracle: <pre>jdbc:oracle:thin:@host name:port;SID</pre>

Parameter	Description
Repository Driver	Provide the JDBC driver class to use when opening a connection to the database (optional). For example, the following repository driver format is used when connecting to Oracle: oracle.jdbc.driver.OracleDriver
Repository User	Enter a valid user ID to use when opening a connection to the database.
Repository Password	Enter a valid password that is associated with the user ID.
Repository Pooling	If selected, repository pooling is used. This option is disabled by default.

4. Click **Save**.

Configuring a File System Repository

If you do not have access to a database for the repository, you can store repository information in an XML file on your local system. However, a file system repository is less secure and efficient than a database repository. When BSE is first installed, it is automatically configured to use a file system repository.

Note: Do not use a file repository for BSE in production environments.

The default location for the repository on Windows is:

```
C:\oracle\Middleware\user_projects\domains\base_domain\servers\soa_server1\stage\ibse\ibse.war
```

On other platforms, use the corresponding location.

If you are using a file system repository, you are not required to configure any additional BSE components.

2.2.2 Configuring the Database Repository for J2CA

This section describes how to configure the database repository for J2CA.

1. Execute the `iwse.ora` SQL script on the machine where the database is installed.

The `iwse.ora` SQL script is located in the following directory:

```
C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\etc
```

This script creates the required tables that are used to store the adapter configuration information in the database. These tables are used by Application Explorer and by adapters during design time and runtime. It is recommended that you use the same credentials to create the database repository and also in the `ra.xml` file for database user credentials.

```
C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\etc>sqlplus
```

```
SQL*Plus: Release 10.1.0.2.0 - Production on Tue Dec 27 18:10:44 2005  
Copyright (c) 1982, 2004, Oracle. All rights reserved.
```

```
Enter user-name: scott  
Enter password: scott1
```

```
Connected to:
Oracle Database 11g Enterprise Edition Release 11.1.1.2.0 - Production
With the Partitioning, OLAP and Data Mining options
```

```
SQL>@ iwse.ora
```

2. Create the `jcatransport.properties` file and save it in the following directory:

```
C:\oracle\Middleware\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\config\J2CA_SampleConfig
```

Note: The `jcatransport.properties` file is required for each J2CA configuration that is created using Application Explorer. The J2CA configuration folder, for example, `J2CA_SampleConfig`, is named according to the configuration name that is specified in Application Explorer.

3. Enter values for `iwafjca.repo.url`, `iwafjca.repo.user` and `iwafjca.repo.password` fields in the newly created `jcatransport.properties` file, as shown in the following example:

```
iwafjca.repo.url=jdbc:oracle:thin:@90.0.0.51:1521:orcl
iwafjca.repo.user=scott
iwafjca.repo.password=scott1
```

The following table lists the parameters with descriptions of the information to provide.

Parameter	Description
<code>iwafjca.repo.url</code>	Specify the JDBC URL to use when opening a connection to the database. For example, the following repository URL format is used when connecting to Oracle: <code>jdbc:oracle:thin:@host name:port;SID</code>
<code>iwafjca.repo.user</code>	Specify a valid user ID to use when opening a connection to the database.
<code>iwafjca.repo.password</code>	Specify a valid password that is associated with the user ID.

4. The `ojdbc14.jar` file must be copied to the following directory:

```
C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\lib
```

5. Navigate to the following directory:

```
C:\oracle\Middleware\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\iwafjca.rar\META-INF
```

6. Open the `ra.xml` file in a text editor.
7. Provide the JDBC connection information as a value for the `IWAYRepo_URL` property.
8. Provide a valid user name for the `IWAYRepo_User` property.

9. Provide a valid password for the `IWAYRepo_Password` property.
10. Save your changes to the `ra.xml` file.

2.2.2.1 Password Encryption

When creating J2CA configurations, you can also encrypt a password using Application Explorer and use this value in the `jcatransport.properties` and `ra.xml` files for added security.

Configuring Password Encryption

To encrypt a password:

1. Open Application Explorer.
2. Click **Help** and select **Encryption**.
The Encryption dialog is displayed.
3. Type a password in the Password field and click OK.
An encrypted version of the password displays in the Encryption field.
4. Copy the password.
5. In the `jcatransport.properties` file, which is used during design time, replace the existing password with the encrypted value only if you are using a database repository.

The following is a sample of the `jcatransport.properties` file where the password is replaced:

```
iwafjca.log.level=DEBUG
iwafjca.repo.url=jdbc:oracle:thin:@172.30.166.100:1521:orcl
iwafjca.repo.user=scott
iwafjca.repo.password=ENCR (318931973183297321831293164323332123227)
```

6. In the `ra.xml` file, which is used during run time, replace the existing password with the encrypted value for the `IWayRepoPassword` element. This is applicable for file system and database repositories.
7. Restart the Oracle WebLogic Server.

2.3 Creating a Repository Configuration

Before you use Application Explorer with Oracle Application Adapter for PeopleSoft, you must create a repository configuration. You can create two kinds of repository configurations, Web services and J2CA, depending on the container to which the adapter is deployed.

During design time, the repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at run-time.

Web services and BSE refer to the same type of deployment. See "[Adapter Features](#)" on page 1-1 for more information.

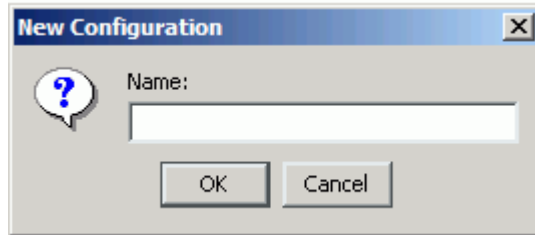
2.3.1 Creating a Configuration for BSE

To create a repository configuration for BSE using Application Explorer, you must first define a new configuration.

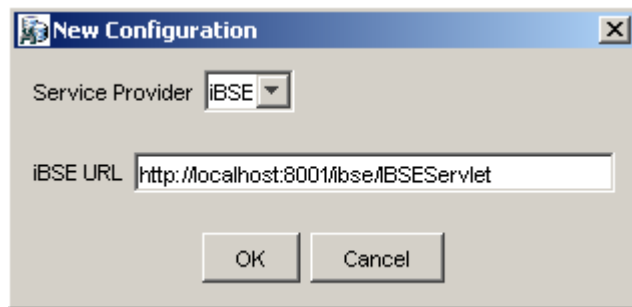
Defining a New Configuration for BSE

To define a new configuration for BSE:

1. Start the Application Explorer.
2. Right-click **Configurations** and select **New**.
The New Configuration dialog is displayed.



3. Enter a name for the new configuration (for example, myConfig) and click **OK**.
The New Configuration dialog is displayed.



4. From the **Service Provider** list, select **iBSE**.
5. In the **BSE URL** field, accept the default URL or replace it with a different URL with the following format:

```
http://host name:port/ibse/IBSEServlet
```

Where *host name* is the system where Oracle WebLogic Server resides and *port* is the HTTP port for a managed Oracle WebLogic server (for example, soa_server1).

6. Click **OK**.
A node representing the new configuration appears beneath the root Configurations node.



2.3.2 Creating a Configuration for J2CA

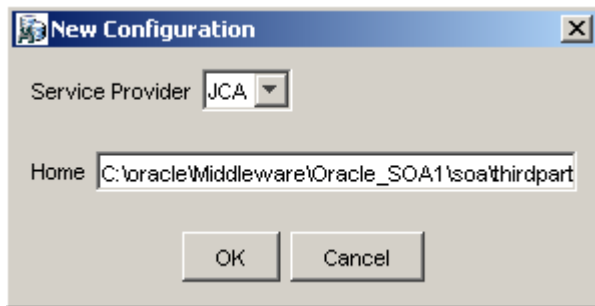
To create a configuration for Oracle Adapter J2CA using Application Explorer, you must first define a new configuration.

Defining a New Configuration for J2CA

To define a new configuration for J2CA:

1. Start the Application Explorer.

2. Right-click **Configurations** and select **New**.
The New Configuration dialog is displayed.



3. Enter a name for the new configuration (for example, myConfig) and click **OK**.
4. From the **Service Provider** list, select **JCA**.
5. Click **OK**.

A node representing the new configuration appears beneath the root Configurations node.



The Oracle Adapter J2CA configuration file is stored in:

C:\oracle\Middleware\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\config\configuration_name
Where *configuration_name* is the name of the configuration you created (for example, myConfig).

HTTP Repository Connection

J2CA users can create an HTTP repository connection, which enables them to generate and store WSDL documents remotely. Perform the following steps to create an HTTP repository connection in Application Explorer. To use the HTTP repository, ensure that the iwjaivp test tool(jca-app-adapter-test) is successfully deployed and running.

1. Start the Application Explorer.
2. Right-click the **Configurations** node in the left pane and select **New**.
The New Configuration dialog opens.
3. Type a name for the configuration and click **OK**.
4. Select **JCA** from the Service Provider list box and enter an HTTP target value in the Home field.

Use the following format for the HTTP target value:

`http://host name:port/iwafjca/JCAServlet`

For example:

`http://iwserv14:8001/iwafjca/JCAServlet`

5. Click **OK**.

The new HTTP repository connection is added to the Configurations node.

Once you connect to the remote server, you can create new Adapter targets, generate WSDL documents, and store them in the remote server.

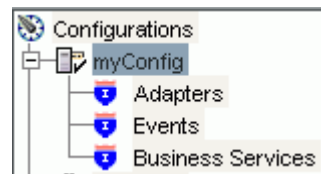
Note: When you configure an Adapter target with the J2CA HTTP repository, you are not required to restart the Oracle WebLogic Server for run time purposes.

2.3.3 Connecting to a BSE or J2CA Configuration

To connect to a new configuration:

1. Right-click the configuration to which you want to connect, for example, myConfig.
2. Select **Connect**.

Nodes appear for Adapters, Events, and Business Services (also known as Web services). The Business Services node is only available for BSE configurations. If you are connected to a J2CA configuration, the Business Services node is not shown. The following is an example of a BSE configuration named myConfig:



- Use the **Adapters** folder to create inbound interaction with PeopleSoft. For example, you use the PeopleSoft node in the Adapters folder to configure a service that updates PeopleSoft.
- Use the **Events** folder (available for J2CA configurations only) to configure listeners that listen for events in PeopleSoft.
- Use the **Business Services** folder (available for BSE configurations only) to test Web services created in the Adapters folder. You can also control security settings for the Web services by using the security features of the Business Services folder.

You are now ready to define new targets to PeopleSoft.

2.4 Establishing a Connection (Target) for PeopleSoft

Part of the application definition includes adding a target for Oracle Application Adapter for PeopleSoft. Setting up the target in Application Explorer requires information that is specific to the target.

To browse PeopleSoft business objects, you must create a target for the system you intend to use. The target serves as your connection point and is automatically saved after you create it. You must establish a connection to this system every time you start Application Explorer or after you disconnect from the system.

When you launch Application Explorer, the left pane displays (as nodes) the application systems supported by Application Explorer, based on the adapters that are installed.

2.4.1 Defining a Target to PeopleSoft

To connect to PeopleSoft for the first time, you must define a new target. Oracle Application Adapter for PeopleSoft supports PeopleSoft standard security, in addition to component interface-level security. Once connected to the PeopleSoft application

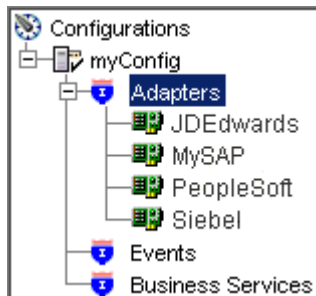
server, application security is managed by user ID, roles and privileges. For more information on PeopleSoft application security, see the appropriate PeopleSoft documentation.

When define a new target, you must restart the Oracle WebLogic Server to update the repository for run time purposes.

To define a target:

1. In the left pane, expand the **Adapters** node.

The applications systems supported by Application Explorer appear as nodes based on the adapters that are installed.



2. Right-click the **PeopleSoft** node and select **Add Target**.

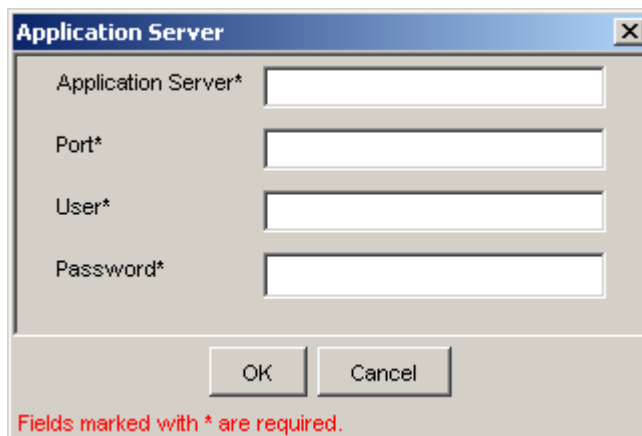
The Add Target dialog is displayed. Provide the following information:

- a. In the **Name** field, enter a descriptive name for the target, for example, **PSoftTarget**.
- b. In the **Description** field, enter a description for the target (optional).
- c. From the **Target Type** list, select **Application Server**.

This is the only possible value for target type.

3. Click **OK**.

The Application Server dialog is displayed. You must specify connection information for PeopleSoft and the application server that is hosting PeopleSoft.

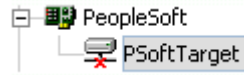


Provide the following information:

- a. In the **Application Server** field, enter the host name or IP address for the computer that is hosting the PeopleSoft application.

- b. In the **Port** field, enter the port number where the PeopleSoft application is listening.
 - c. In the **User** field, enter a valid user ID for the PeopleSoft application.
 - d. In the **Password** field, enter a valid password for the PeopleSoft application.
4. Click **OK**.

In the left pane, the new target (**PSoftTarget**) appears the PeopleSoft node.



You are ready to connect to your PeopleSoft target.

2.4.2 Connecting to a Defined PeopleSoft Target

To connect to an existing target:

1. In the left pane, expand the **Adapters** node.
2. Expand the **PeopleSoft** node.
3. Click the target name (for example, **PSoftTarget**) under the PeopleSoft node.

The Connection dialog displays the values you entered for connection parameters.

4. Verify your connection parameters.
5. Provide the correct password.
6. Right-click the target name and select **Connect**.

The x icon disappears, indicating that the node is connected. A list of PeopleSoft business objects is displayed.



2.4.3 Managing a Target

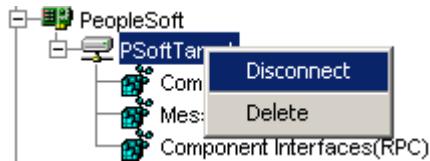
Although you can maintain multiple open connections to different transaction processing systems, it is recommended that you disconnect from connections not in use. After you disconnect, you can modify an existing target.

You can modify the connection parameters when your system properties change. You also can delete a target. The following procedures describe how to disconnect from a target, edit a target, and delete a target.

Disconnecting from a Connection to PeopleSoft

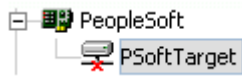
To disconnect from a connection to PeopleSoft:

1. Expand the **Adapters** node.
2. Expand the **PeopleSoft** node.
3. Right-click the target to which you are connected (for example, **PSoftTarget**), and select **Disconnect**.



Disconnecting from PeopleSoft drops the connection with PeopleSoft, but the node remains.

The x icon appears, indicating that the node is disconnected.



Modifying Connection Parameters

After you create a target for PeopleSoft using Application Explorer, you can edit any of the information that you provided previously.

When modify connection parameters for a defined target, you must restart the Oracle WebLogic Server to update the repository for run time purposes.

To edit a target:

1. Verify that the target you want to edit is disconnected.
2. Right-click the target and select **Edit**.



The Application Server dialog displays the target connection information.

3. Change the properties in the dialog as required and click **OK**.

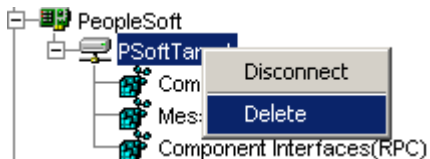
Deleting a Connection to PeopleSoft

You can delete a connection, rather than just disconnecting and closing it. When you delete the connection, the node disappears from the list of PeopleSoft connections in the left pane of Application Explorer.

When you delete a connection, you must restart the Oracle WebLogic Server to update the repository for run time purposes.

To delete a connection to PeopleSoft:

1. Locate the target you want to delete.
2. Right-click the target (for example, PsoftTarget), and select **Delete**.



The node disappears from the list of available connections.

2.5 Viewing Application System Objects

After you are connected to PeopleSoft, Application Explorer enables you to explore and browse business object metadata. For example, Application Explorer enables you to view PeopleSoft Component Interface and Message metadata stored in the PeopleSoft business object repository.

For Component Interfaces(RPC), the adapter enables Delete, Insert, Query, Update, and Find.

To view application system objects:

1. Click the icon to the left of the target name, for example, PSoftTarget.

The target expands and the available system objects are displayed.



2. To expand the desired PeopleSoft repository node, click the icon to the left of the repository name, for example, Component Interfaces.

A list of PeopleSoft Component Interfaces appears. You can now generate schemas.

Note: The Component Interfaces (RPC) node only displays the level 1 Component Interfaces.

2.6 Creating XML Schemas

After you browse the PeopleSoft business object repository, you can generate XML request and response schemas for the object you want to use with your adapter.

Creating XML Request and Response Schemas Against BSE

To create XML request and response schemas for a PeopleSoft Component Interface against a BSE implementation:

1. Expand the PeopleSoft node and then, expand the **Component Interfaces** node.
2. From the list of **Component Interfaces**, select **LOCATION**.
3. Click the **Request Schema** or **Response Schema** tab to view the request schema information.

The schema you selected appears.

After you browse the Component Interfaces and make a selection, the request and response XML schemas are automatically created for that Component Interface and stored in the repository you created.

Creating XML Request and Response Schemas Against the Oracle Adapter J2CA

To create XML request and response schemas for a PeopleSoft Component Interface against an Oracle Adapter J2CA implementation:

1. Expand the **PeopleSoft** node and then, expand the **Component Interfaces** node.

2. From the list of **Component Interfaces**, select **LOCATION**.
3. Click the **Request Schema** or **Response Schema** tab to view the request schema information.

The schema you selected appears.

After you browse the Component Interfaces and make a selection, the request and response XML schemas are automatically created for that Component Interface and stored in the repository you created.

2.7 Generating a WSDL (J2CA Configurations Only)

The Web Service Definition Language (WSDL) description of a service enables you to make the service available to other services within a host server. You use Application Explorer to create both request-response (outbound) and event notification (inbound) JCA services of the adapter.

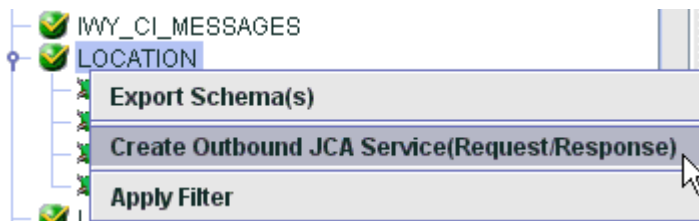
Note: The **Create Inbound JCA Service (Event)** option is only available when the selected node supports events.

Note: PeopleSoft Component Interfaces only support services. As a result, only outbound WSDL files can be generated.

To generate a WSDL file for request-response service:

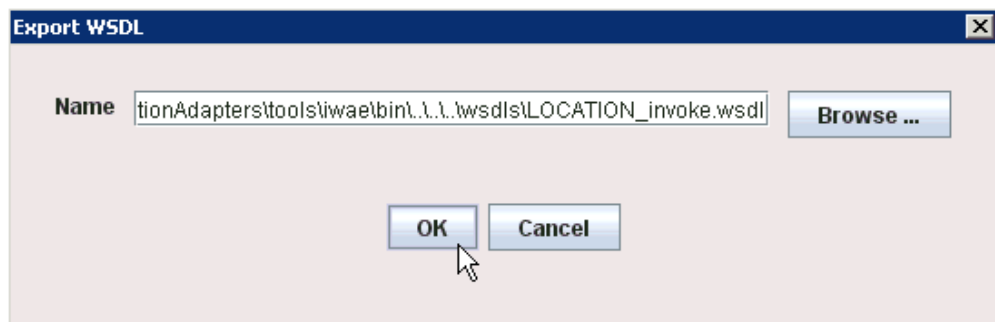
1. After you create a schema, right-click the respective object.

The following options are displayed.



2. Select **Create Outbound JCA Service (Request/Response)**.

The Export WSDL dialog is displayed.



3. Accept the default name for the file.

The **.wsdl** file extension is added automatically. By default, the names of WSDL files generated for request-response services end with **_invoke**, while those generated for event notification end with **_receive**.

4. Click **OK**.

The WSDL file is saved in the specified location.

The procedure for generating WSDL for event notification is similar to request-response. To generate WSDL for event notification, you must first create a channel for every event.

2.8 Creating and Testing a Web Service (BSE Configurations Only)

You can create Web services (also known as a **business service**) using Application Explorer. The PeopleSoft Component Interface called **LOCATION** is used as an example in the following procedure.

Note: In a J2EE Connector Architecture (J2CA) implementation of adapters, Web services are not available. When the adapters are deployed to use Oracle Adapter J2CA, the Common Client Interface provides integration services using the adapters.

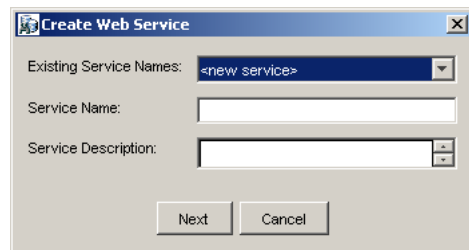
Creating a Web Service

To create a Web service, perform the following steps:

1. Expand the **PeopleSoft** node and then the **Component Interfaces** node.
2. From the list of **Component Interfaces**, select **LOCATION**.
3. Right-click the node from which you want to create a **business service** and select **Create Web Service**.



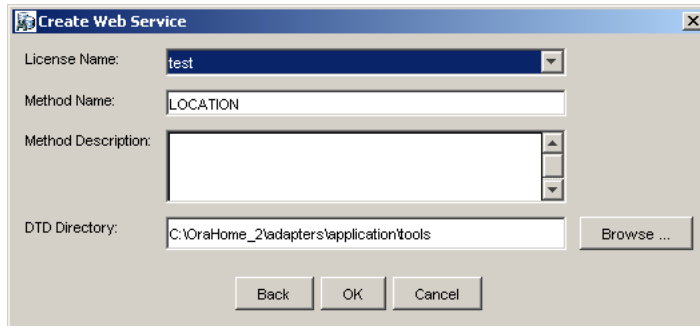
The Create Web Service dialog is displayed.



You can add the business function as a method for a new Web service or as a method for an existing one. Perform the following steps:

- a. From the **Existing Service Names** list, select either **<new service>** or an existing service.

- b. If you are creating a new service, specify a service name. This name identifies the Web service in the list of services under the **Business Services** node.
 - c. Enter a brief description for the service (optional).
4. Click **Next**.
- The license and method dialog is displayed.



Perform the following steps:

- a. In the **License Name** field, select one or more license codes to assign to the Web service. To select more than one, hold down the **Ctrl** key and click the licenses.
 - b. In the **Method Name** field, enter a descriptive name for the method.
 - c. In the **Method Description** field, enter a brief description of the method.
 - d. The **DTD Directory** field specifies a destination for your Web service. You may click **Browse** to change the default location.
5. Click **OK**.
- Application Explorer switches the view to the **Business Services** node, and the new Web service appears in the left pane.

Testing a Web Service

After a Web service is created, you can test it to ensure that it functions properly. A test tool is provided for testing the Web service.

To test a business service:

1. Click the **Business Services** node to access your Web services.
2. Expand the **Services** node.
3. Select the name of the business service you want to test.
The business service name appears as a link in the right pane.
4. In the right pane, click the named business services link.
The test option appears in a separate BSE Servlet page. If you are testing a Web service that requires XML input, an input field appears.
5. Enter the appropriate input.
6. Click **Invoke**.
The BSE Servlet page displays the results.

Identity Propagation

If you test or execute a Web service using a third party XML editor, for example XMLSPY, the Username and Password values that you specify in the SOAP header must be valid and are used to connect to PeopleSoft. The user name and password values that you provided for PeopleSoft during target creation using Application Explorer are overwritten for this Web service request. The following is a sample SOAP header that is included in the WSDL file for a Web service:

```
<SOAP-ENV:Header>
  <m:ibsinfo xmlns:m="urn:schemas-iwaysoftware-com:iwse">
    <m:service>String</m:service>
    <m:method>String</m:method>
    <m:license>String</m:license>
    <m:disposition>String</m:disposition>
    <m:Username>String</m:Username>
    <m>Password>String</m>Password>
    <m:language>String</m:language>
  </m:ibsinfo>
</SOAP-ENV:Header>
```

You can remove the `<m:disposition>` and `<m:language>` tags from the SOAP header, since they are not required.

2.9 Configuring an Event Adapter

Events are generated by activity in an application system. You can use events to trigger an action in your application. For example, PeopleSoft may generate an event when customer information is updated. If your application performs an action when this happens, your application is a consumer of this event.

Note: BSE configurations do not support events.

After you create a connection to your application system, you can add events using Application Explorer. To create an event, you must create a channel.

Note: If you are using a J2CA configuration, you must create a new channel for every event object and select this channel when you generate WSDL. Creating a channel is required for J2CA configurations only.

A **channel** represents configured connections to particular instances of back-end systems. See "[Creating and Editing a Channel](#)" on page 2-19 for more information.

2.9.1 Creating and Editing a Channel

The following procedures describe how to create a channel for your event. All defined event ports must be associated with a channel.

When you create, modify, or delete a channel, you must restart the Oracle WebLogic Server to recognize the change and update the repository for run time purposes.

Note: If using a J2CA configuration, you must create a new channel for every event and select this channel when creating an inbound service. Creating a channel is required for J2CA configurations only.

Creating a Channel Using Specific Protocols

You can create the following types of channels using Application Explorer:

- HTTP
It is recommended to use HTTP channels with PeopleTools Version 8.4.
- TCP
It is recommended to use TCP channels with PeopleTools Version 8.1.
- File
It is recommended to use File channels for testing purposes only. Do not use File channels in a production environment.

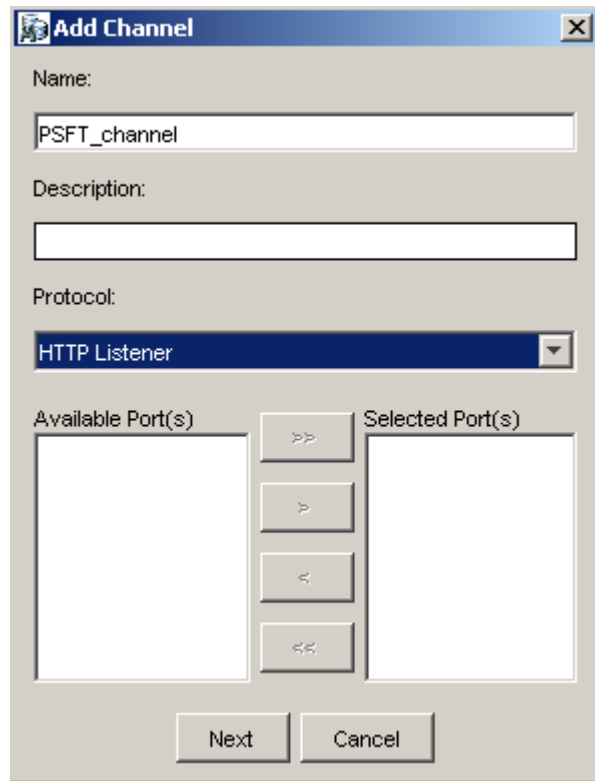
Note: Channels can be configured and started only on the system where the Oracle Application Adapter for PeopleSoft is installed. Configuring and starting a channel for a remote host is not supported.

The following procedures explain how to create these channels.

2.9.1.1 Creating an HTTP Channel

To create an HTTP Channel:

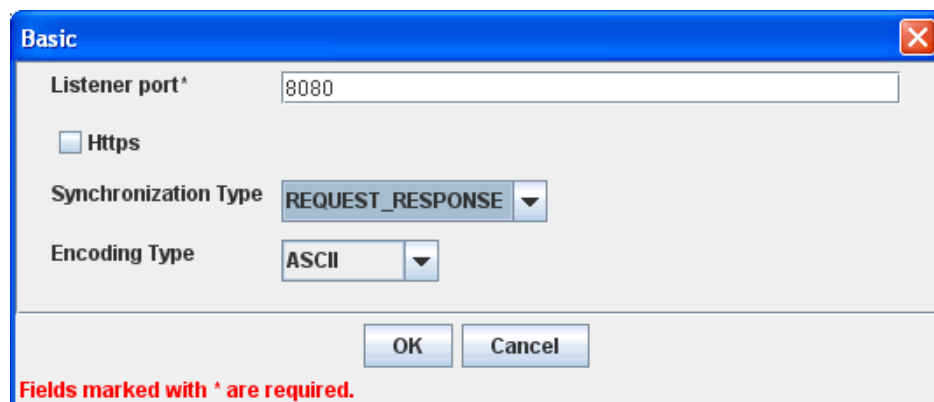
1. Click the **Events** node.
2. Expand the **PeopleSoft** node.
The ports and channels nodes appear in the left pane.
3. Right-click **Channels** and select **Add Channel**.
The Add Channel dialog is displayed.



Provide the following information:

- a. Enter a name for the channel, for example, **PSFT_channel**.
 - b. Enter a brief description.
 - c. From the **Protocol** list, select **HTTP Listener**.
4. Click **Next**.

The Basic dialog is displayed.



5. Enter the system information as specified in the following table:

Parameter	Description
Listener port	Port on which to listen for PeopleSoft event data.

Parameter	Description
Https	For a secure HTTP connection, select the Https check box. This option is currently not supported.
Synchronization Type	Choose from the following synchronization options: <ul style="list-style-type: none"> ■ REQUEST_RESPONSE ■ REQUEST_ACK <p>Important: The PeopleSoft channel does not work if the synchronization type is set to REQUEST.</p>
Encoding Type	Choose an encoding type to be used from the list. By default, ASCII is selected.

6. Click **OK**.

A summary pane is displayed, providing the channel description, channel status, and available ports. All the information is associated with the channel you created.

The channel appears under the channels node in the left pane.

An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

Note: If you are planning to integrate Oracle Application Adapter for PeopleSoft with BPEL inbound or Mediator inbound process components, do not start the channel. The channel is managed by the run-time server after the BPEL or Mediator process component is deployed. If you start the channel from Application Explorer for testing and debugging purposes, stop it before run-time (when working with BPEL or Mediator process components).

7. Right-click the channel and select **Start**.

The channel you created becomes active. The X over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select **Stop**.

2.9.1.2 Creating a TCP Channel

1. Click the **Events** node.

2. Expand the **PeopleSoft** node.

The ports and channels nodes appear in the left pane.

3. Right-click **Channels** and select **Add Channel**.

The Add Channel dialog is displayed.

Perform the following steps:

- a. Enter a name for the channel, for example, PSoftChannel.
 - b. Enter a brief description.
 - c. From the **Protocol** list, select **TCP Listener**.
4. Click **Next**.

The Tcp Listener dialog is displayed.

Provide the following information:

- a. In the **Port Number** field, specify the port number for your TCP listener (required).

- b. In the **Host/IP Binding** field, specify the host IP for your TCP listener (optional).

Note: Channels can be configured and started only on the system where the Oracle Application Adapter for PeopleSoft is installed. Configuring and starting a channel for a remote host is not supported.

- c. From the **Synchronization Type** list, select from the following synchronization type options:

REQUEST_RESPONSE

REQUEST_ACK

REQUEST

Important: The PeopleSoft channel only works with one of these synchronization types.

- d. **Is Length Prefix** check box: For PeopleSoft events that return data that is not in XML format. The TCP/IP event application must prefix the data with a 4-byte binary length field when writing data to the TCP/IP port.
 - e. **Is XML** check box: For PeopleSoft events that return data in XML format. No parser is required.
 - f. **Is Keep Alive** check box: Maintains continuous communication between the event transaction and the channel.
5. Click **OK**.

The channel appears under the Channels node in the left pane.



An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

Note: If you are planning to integrate Oracle Application Adapter for PeopleSoft with BPEL inbound or Mediator inbound process components, do not start the channel. The channel is managed by the run-time server after the BPEL or Mediator process component is deployed. If you start the channel from Application Explorer for testing and debugging purposes, stop it before run-time (when working with BPEL or Mediator process components).

6. Right-click the channel node and select **Start**.

The channel becomes active.



The X over the icon disappears.

7. To stop the channel, right-click the connected channel node and select **Stop**.
The channel becomes inactive and an X appears over the icon.

2.9.1.3 Creating a File Channel

To create a channel for the File listener:

1. Click the **Events** node.
2. In the left pane, expand the **PeopleSoft** node.
The ports and channels nodes appear.
3. Right-click **Channels** and select **Add Channel**.
The Add Channel dialog is displayed.

Perform the following steps:

- a. Enter a name for the channel, for example, **NewFileChannel**.
 - b. Enter a brief description (optional).
 - c. From the **Protocol** list, select **File Listener**.
4. Click **Next**.
 5. The File Listener dialog is displayed.

Enter the system information as follows:

- a. In the **Request** tab, enter values for the following parameters:

Parameter	Description
Polling Location	Target file system location for the PeopleSoft XML file.
File Mask	File name to be used for the output file generated by the operation.

- b. In the **Response** tab, enter values for the following parameters:

Parameter	Description
Synchronization Type	Choose from the following synchronization options: <ul style="list-style-type: none"> ■ REQUEST_RESPONSE ■ REQUEST_ACK <p>Important: The PeopleSoft channel does not work if the synchronization type is set to REQUEST.</p>
Response/Ack Directory	Target file system location for the PeopleSoft XML file.

- c. In the **Advanced** tab, enter values for the following parameters:

Parameter	Description
Error directory	Directory to which documents with errors are written.
Poll interval (msec)	Interval (in milliseconds) when to check for new input. The default is three seconds. Optional.
Processing Mode	Sequential indicates single processing of requests. Threaded indicates processing of multiple requests simultaneously.
Thread limit	If you selected threaded processing, indicate the maximum number of requests that can be processed simultaneously.

6. Click **OK**.

A summary pane is displayed, providing the channel description and channel status. All the information is associated with the channel you created.

The channel appears under the channels node in the left pane. An X over the icon indicates that the channel is currently disconnected. You must start the channel to activate your event configuration.

Note: If you are planning to integrate Oracle Application Adapter for PeopleSoft with BPEL inbound or Mediator inbound process components, do not start the channel. The channel is managed by the run-time server after the BPEL or Mediator process component is deployed. If you start the channel from Application Explorer for testing and debugging purposes, stop it before run-time (when working with BPEL or Mediator process components).

7. Right-click the channel and select **Start**.

The channel you created becomes active. The X over the icon in the left pane disappears.

8. To stop the channel, right-click the channel and select **Stop**.

2.9.1.4 Editing a Channel

You must stop the channel before you can edit it. To edit a channel:

1. In the left pane, locate the channel you want to edit.
2. Right-click the channel and select **Edit**.
The Edit channels pane is displayed.
3. Make the required changes to the channel configuration and click **Finish**.

2.9.1.5 Deleting a Channel

You must stop the channel before you can delete it. To delete a channel:

1. In the left pane, locate the channel you want to delete.
2. Right-click the channel and select **Delete**.
The channel disappears from the list in the left pane.

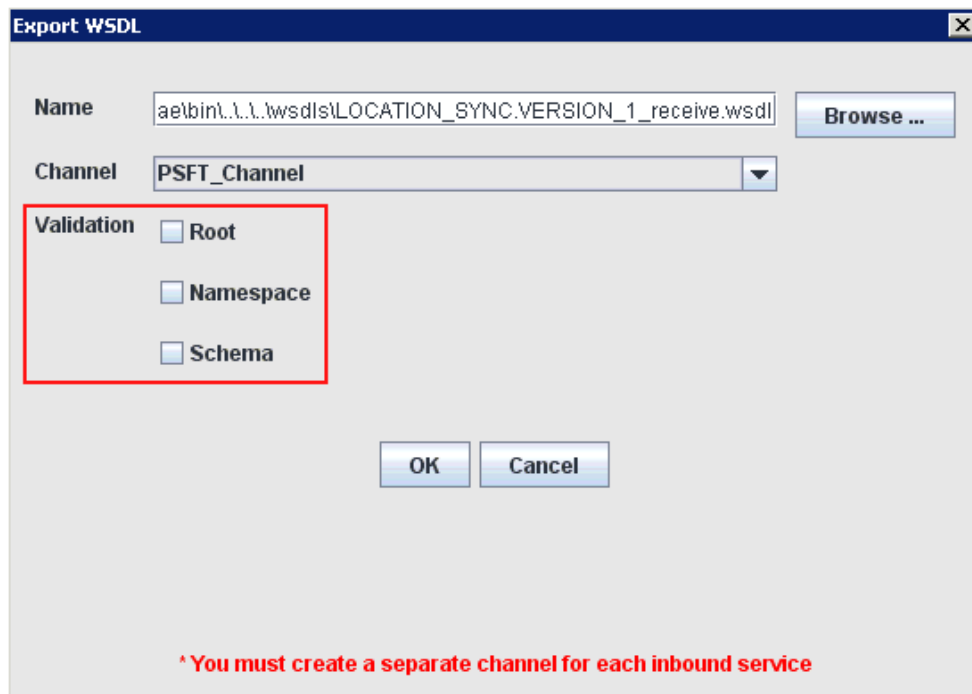
2.9.2 Schema Validation

Root validation, namespace validation, and schema validation for inbound processing (events) are supported for the Oracle Application Adapter for PeopleSoft with 11.1.1.2.0 BPEL.

To validate inbound processing using the Oracle Application Adapter for PeopleSoft, perform the following steps. This procedure uses the `LOCATION_SYNC.VERSION_1` PeopleSoft Message as an example for inbound processing.

1. Start Application Explorer.
2. Connect to the PeopleSoft target.
3. Expand the **Messages** node.
4. Verify that you have already created a channel for the PeopleSoft adapter.
5. Right-click the `LOCATION_SYNC.VERSION_1` node and select **Create Inbound JCA Service (event)**.

The Export WSDL dialog opens and includes three check boxes for Root, Namespace, and Schema validation.



- Selection of multiple validation options is allowed.
 - Root validation is used to validate the root element in the inbound XML document.
 - Namespace validation is used to validate the namespace in the inbound XML document.
 - Schema validation is used to validate the inbound XML document with the schema in the WSDL document.
 - During run time, validation is processed based on the validation options that are selected.
 - If more than one validation option is selected, during run time if the first validation option fails, the remaining validation options are not processed.
 - Root and namespace validations are considered modest levels of validation. Schema validation is a stricter validation level.
 - It is recommended to use root and namespace validation options together, unless the root element and namespace are different between the Messages in the PeopleSoft environment.
6. Generate the WSDL document and create the BPEL process.
 7. Trigger the transactions (Messages) from the PeopleTools environment.

Oracle WebLogic Server Deployment and Integration

This chapter describes Oracle WebLogic Server (OracleWLS) deployment and integration with Oracle Application Adapter for PeopleSoft.

This chapter discusses the following topics:

- [Adapter Integration with Oracle WebLogic Server](#)
- [Deployment of Adapter](#)
- [Updating Adapter Configuration](#)

See Also:

- *Oracle Application Server Adapter Concepts Guide*

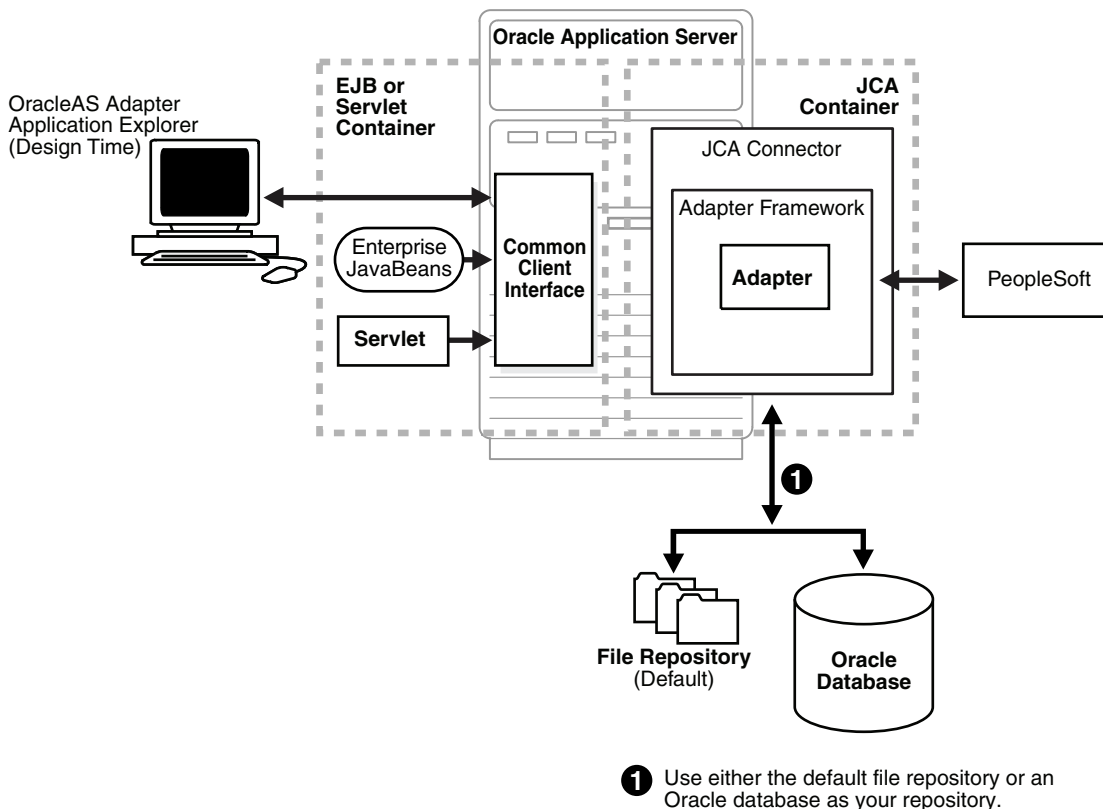
3.1 Adapter Integration with Oracle WebLogic Server

Oracle Application Adapter for PeopleSoft is deployed within an OracleWLS container during installation. All client applications run within the OracleWLS environment. In J2CA deployment, the Common Client Interface (CCI) integrates an OracleWLS client application with a resource adapter.

3.2 Deployment of Adapter

[Figure 3–1](#) shows deployment of the J2CA Connector to the Oracle Application Server. In a run-time service scenario, an Enterprise Java Bean, servlet, or Java program client makes CCI calls to J2CA resource adapters. The adapters process the calls as requests and send them to the EIS. The EIS response is then sent back to the client.

Figure 3–1 Oracle Application Server J2CA Architecture



3.3 Updating Adapter Configuration

During the J2CA deployment of Oracle Application Adapter for PeopleSoft, OracleWLS generates a deployment descriptor called `ra.xml`, located in:

```
C:\oracle\Middleware\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\iwafjca.rar\META-INF
```

Your installation contains more than one file named `ra.xml`. The OracleWLS deployment descriptor that is described in this section is located in the directory specified above.

Note: Multiple managed connection factories are supported only for outbound processing (services).

Creating a Managed Connector Factory Object

The `ra.xml` descriptor provides OracleWLS-specific deployment information for resource adapters. For example, the `jca_sample` configuration in Application Explorer is represented in the `ra.xml` file as follows:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE connector PUBLIC "-//Sun Microsystems, Inc.//DTD Connector 1.0//EN"
'http://java.sun.com/dtd/connector_1_0.dtd'>
<connector>
  <display-name>IWAFJCA10</display-name>
  <vendor-name>IWAY Software</vendor-name>
  <spec-version>1.0</spec-version>
```

```

<eis-type>IWAF</eis-type>
<version>1.0</version>
<license>
  <license-required>>false</license-required>
</license>
<resourceadapter>

<managedconnectionfactory-class>com.ibi.afjca.spi.IWAFManagedConnectionFactory</ma
nagedconnectionfactory-class>

<connectionfactory-interface>javax.resource.cci.ConnectionFactory</connectionfacto
ry-interface>

<connectionfactory-impl-class>com.ibi.afjca.cci.IWAFConnectionFactory</connectionf
actory-impl-class>
  <connection-interface>javax.resource.cci.Connection</connection-interface>

<connection-impl-class>com.ibi.afjca.cci.IWAFConnection</connection-impl-class>
  <transaction-support>NoTransaction</transaction-support>
  <config-property>
    <config-property-name>AdapterName</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>Config</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayHome</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value>C:\oracle\Middleware\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters</config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayConfig</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value>jca_sample</config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoDriver</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoURL</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoUser</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>
  <config-property>
    <config-property-name>IWayRepoPassword</config-property-name>
    <config-property-type>java.lang.String</config-property-type>
    <config-property-value></config-property-value>
  </config-property>

```

```

<config-property>
  <config-property-name>LogLevel</config-property-name>
  <config-property-type>java.lang.String</config-property-type>
  <config-property-value>DEBUG</config-property-value>
</config-property>
<authentication-mechanism>
  <authentication-mechanism-type>BasicPassword</authentication-mechanism-type>

<credential-interface>javax.resource.spi.security.PasswordCredential</credential-interface>
  </authentication-mechanism>
  <reauthentication-support>true</reauthentication-support>
</resourceadapter>
</connector>

```

The parameters defined in the ra.xml file are described in the following table:

Parameter Name	Description
IWayHome	The base installation directory for the OracleWLS packaged application adapter.
IWayConfig	The adapter configuration name as defined in Application Explorer. For example, Oracle Application Adapter for PeopleSoft has a preconfigured <code>jca_sample</code> configuration in Application Explorer.
IWayRepoURL	The URL to use when opening a connection to the database. This is necessary only when using an Oracle database as the repository.
IWayRepoUser	User name to use when connecting to the database. This is necessary only when using an Oracle database as the repository.
IWayRepoPassword	Password. If provided, it overwrites configuration. This is necessary only when using an Oracle database as the repository.
loglevel	It overwrites the level set by the ManagedConnectorFactory property.

Creating Multiple Managed Connector Factory Objects

To establish multiple managed connector factory objects, you must edit the `weblogic-ra.xml` file and add more `<connection-instance>` nodes. This file is located in:

```

C:\oracle\Middleware\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\iwafjca.rar\META-INF

```

For example, the first `jca_configuration` in Application Explorer is represented in the `weblogic-ra.xml` file as follows:

```

<?xml version="1.0"?>
<weblogic-connector xmlns="http://www.bea.com/ns/weblogic/90">
  <enable-access-outside-app>true</enable-access-outside-app>
  <enable-global-access-to-classes>true</enable-global-access-to-classes>
  <outbound-resource-adapter>
    <default-connection-properties>
      <pool-params>
        <initial-capacity>0</initial-capacity>
      </pool-params>

```

```

        <transaction-support>LocalTransaction</transaction-support>
    </default-connection-properties>
</connection-definition-group>

<connection-factory-interface>javax.resource.cci.ConnectionFactory</connection-factory-interface>
    <connection-instance>
        <jndi-name>eis/OracleJCAAdapter/DefaultConnection</jndi-name>
    </connection-instance>
</connection-definition-group>
</outbound-resource-adapter>
</weblogic-connector>

```

To create multiple managed connector factory objects, you must add new <connection-instance> nodes in the file. For example:

```

<?xml version="1.0"?>
<weblogic-connector xmlns="http://www.bea.com/ns/weblogic/90">

    <enable-access-outside-app>true</enable-access-outside-app>
    <enable-global-access-to-classes>true</enable-global-access-to-classes>

    <outbound-resource-adapter>
        <default-connection-properties>
        <pool-params>
        <initial-capacity>0</initial-capacity>
        </pool-params>
        <transaction-support>LocalTransaction</transaction-support>
        </default-connection-properties>
    </connection-definition-group>

<connection-factory-interface>javax.resource.cci.ConnectionFactory</connection-factory-interface>
    <connection-instance>
        <jndi-name>eis/OracleJCAAdapter/DefaultConnection</jndi-name>
    </connection-instance>
    <connection-instance>
        <jndi-name>eis/OracleJCAAdapter/DefaultConnection1</jndi-name>
        <connection-properties>
        <properties>
        <property>
<name>IWayHome</name>
<value>C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters</value>
        </property>
        <property>
        <name>IWayConfig</name>
        <value>jca_sample2</value>
        </property>
        <property>
        <name>IWayRepoURL</name>
        <value></value>
        </property>
        <property>
        <name>IWayRepoUser</name>
        <value></value>
        </property>
        <property>
        <name>IWayRepoPassword</name>
        <value></value>
        </property>
        <property>

```

```

        <name>LogLevel</name>
        <value>Debug</value>
      </property>
    </properties>
  </connection-properties>
</connection-instance>
</connection-definition-group>
</outbound-resource-adapter>
</weblogic-connector>

```

If you do not specify a `<property>` element in the `<connection-instance>` section, the value is taken from the `ra.xml` file. You can specify the default properties in the `ra.xml` file and then override them as required in the `weblogic-ra.xml` file. In addition, note that the J2CA configuration (for example, `jca_sample2`) must already be created in Application Explorer.

Note: When you modify the `ra.xml` and `weblogic-ra.xml` files, the application server must be restarted. If the application server is already running, stop the application server and then restart it.

In addition, the `iwafjca.rar` file must be redeployed in the Oracle WebLogic Administration Console to activate these changes.

Modifying WSDL Files for Additional Connection Factory Values

Application Explorer generates the J2CA properties file using the default connection factory name `eis/OracleJCAAdapter/DefaultConnection`. If you created additional connection factories, the WSDLs generated for the additional configuration and connection factory should be changed to reflect the location field of the `jca:address` section in the J2CA properties file. The default J2CA properties file for the Oracle Application Adapter for PeopleSoft with a configuration of `isdsrv2_conn2` is shown in the following example.

Notice that the J2CA properties file has the following default connection factory:
`eis/OracleJCAAdapter/DefaultConnection`

```

<jca:address location="eis/OracleJCAAdapter/DefaultConnection"
  ConnectionSpec="com.ibi.afjca.cci.IWAFConnectionSpec"
  cs.AdapterName="PeopleSoft" cs.Config="isdsrv2_conn2"
  UIConnectionName="Connection1"/>

```

The connection factory value must be changed to the following:
`eis/OracleJCAAdapter/DefaultConnection1`

For example:

```

<jca:address location="eis/OracleJCAAdapter/DefaultConnection1"
  ConnectionSpec="com.ibi.afjca.cci.IWAFConnectionSpec"
  cs.AdapterName="PeopleSoft" cs.Config="isdsrv2_conn2"
  UIConnectionName="Connection1"/>

```

Note that only the value for the location field in the `jca:address` section should be modified. Do not modify any other field or section.

Integration With BPEL Service Components in the Oracle SOA Suite

Oracle Application Adapter for PeopleSoft integrates seamlessly with Oracle Business Process Execution Language (BPEL) Process Manager to facilitate Web service integration. Oracle BPEL Process Manager is based on the Service-Oriented Architecture (SOA). It consumes adapter services exposed as Web Service Definition Language (WSDL) documents.

This chapter includes the following topics:

- [Overview](#)
- [Deployment of Adapter](#)
- [Configuring a New Application Server Connection](#)
- [Designing an Outbound BPEL Process for Service Integration](#)
- [Designing an Inbound BPEL Process for Event Integration](#)

4.1 Overview

To integrate with Oracle SOA Suite, Oracle Application Adapter for PeopleSoft must be deployed in the same WLS container as Oracle BPEL Process Manager. The underlying adapter services must be exposed as WSDL files, which are generated during design time in Oracle Application Adapter Application Explorer (Application Explorer) for both request-response (outbound) and event notification (inbound) services of the adapter. See [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#) for more information.

The generated WSDL files are used to design the appropriate BPEL processes for inbound or outbound adapter services. A completed BPEL process must be successfully compiled in JDeveloper and deployed to a SOA server. Upon deployment to the SOA server, every newly built process is automatically deployed to the Oracle Enterprise Manager Console, where you run, monitor, and administer BPEL processes, and listen to adapter events.

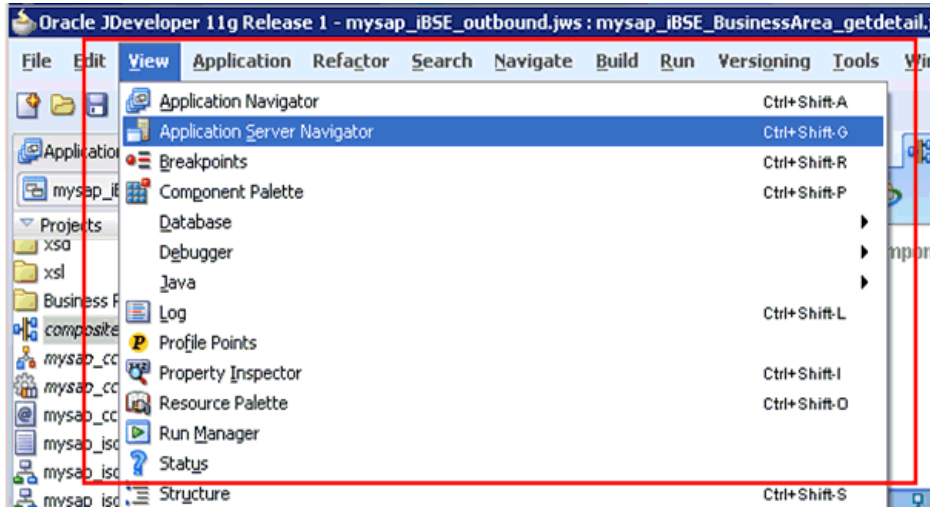
4.2 Deployment of Adapter

During installation, Oracle Application Adapter for PeopleSoft is deployed as a J2CA 1.0 resource adapter within the WLS container. The adapter must be deployed in the same WLS container as Oracle BPEL Process Manager.

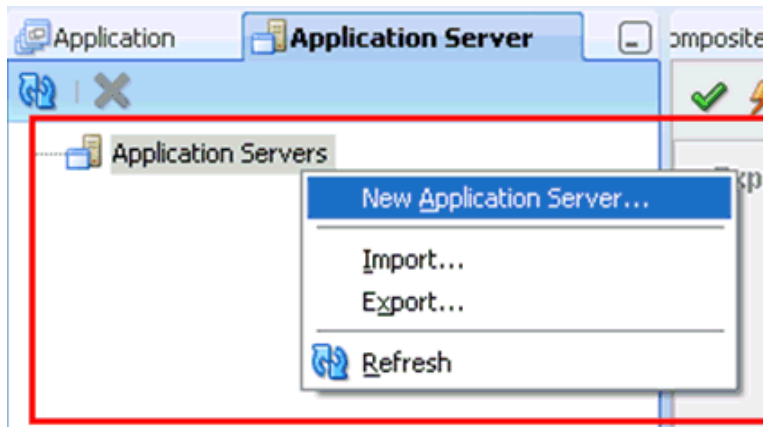
4.3 Configuring a New Application Server Connection

To configure a new Application Server connection in Oracle JDeveloper:

1. Open Oracle JDeveloper on your system.
2. From the menu bar, click **View** and select **Application Server Navigator**.

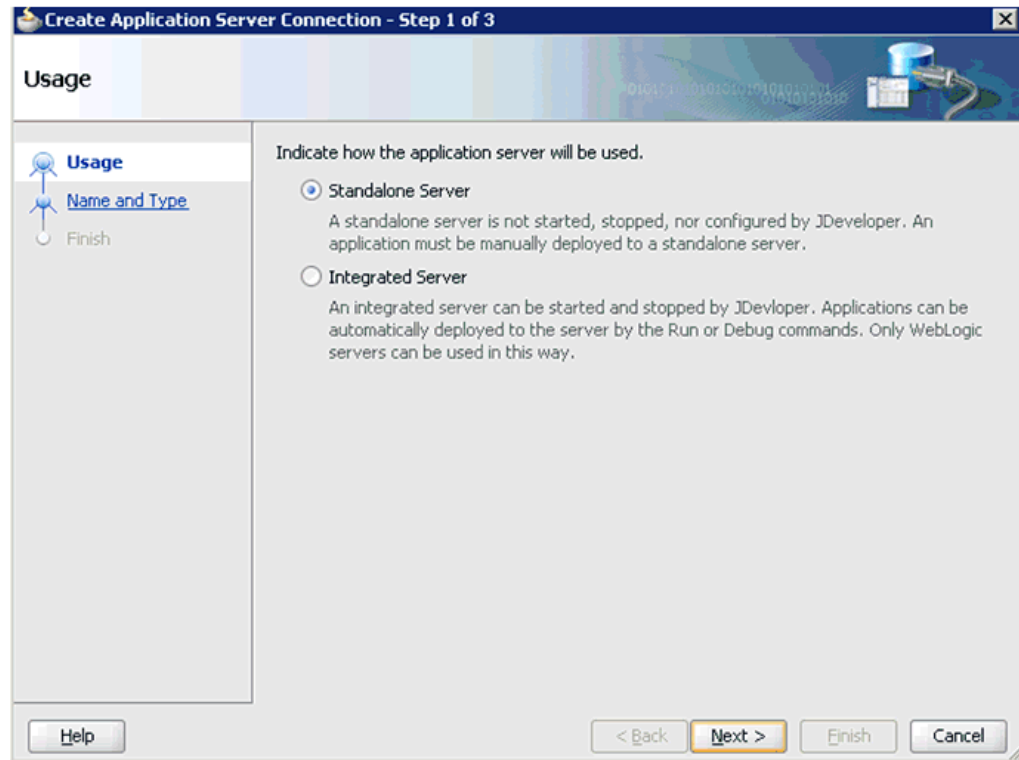


The Application Server tab is displayed.

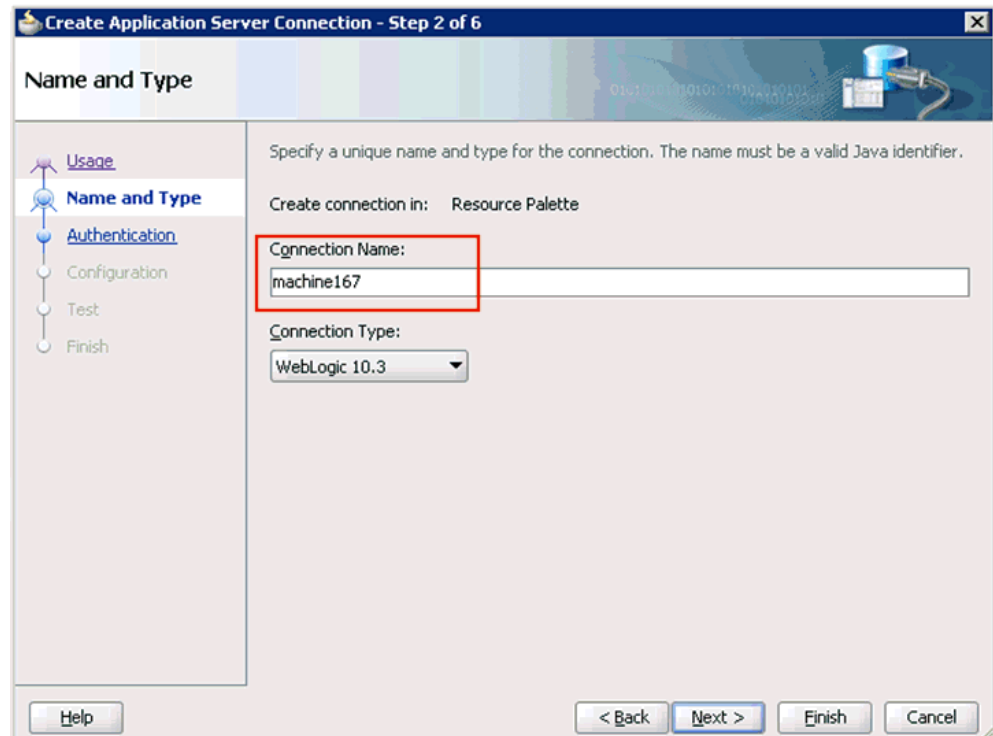


3. Right-click **Application Servers**, and then select **New Application Server**.

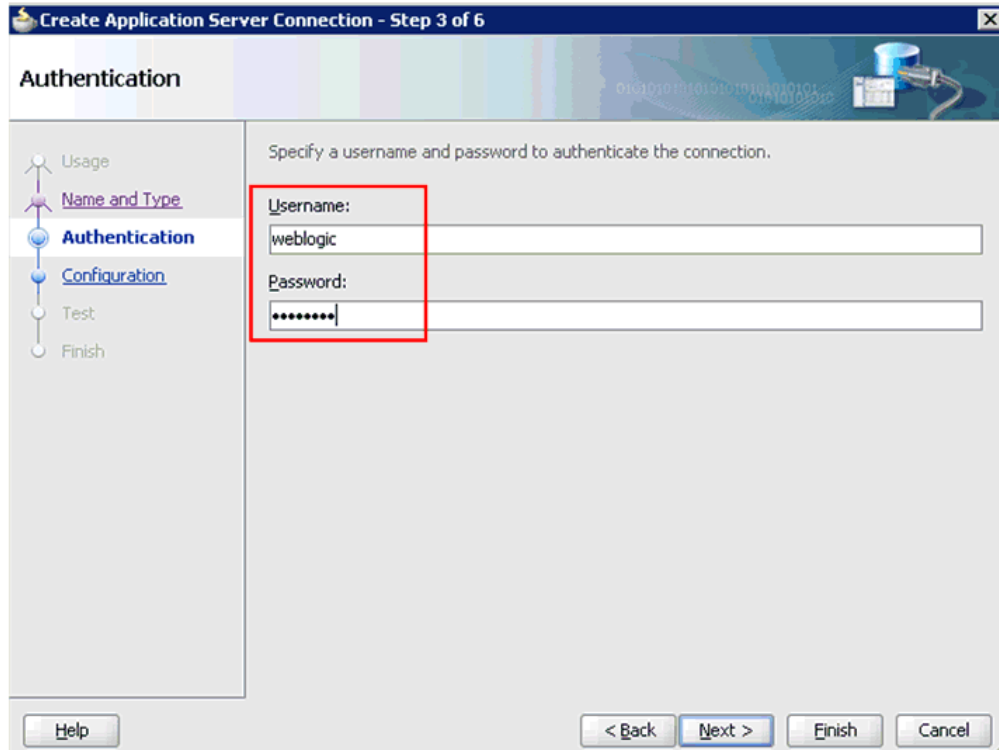
The Create Application Server Connection Wizard is displayed.



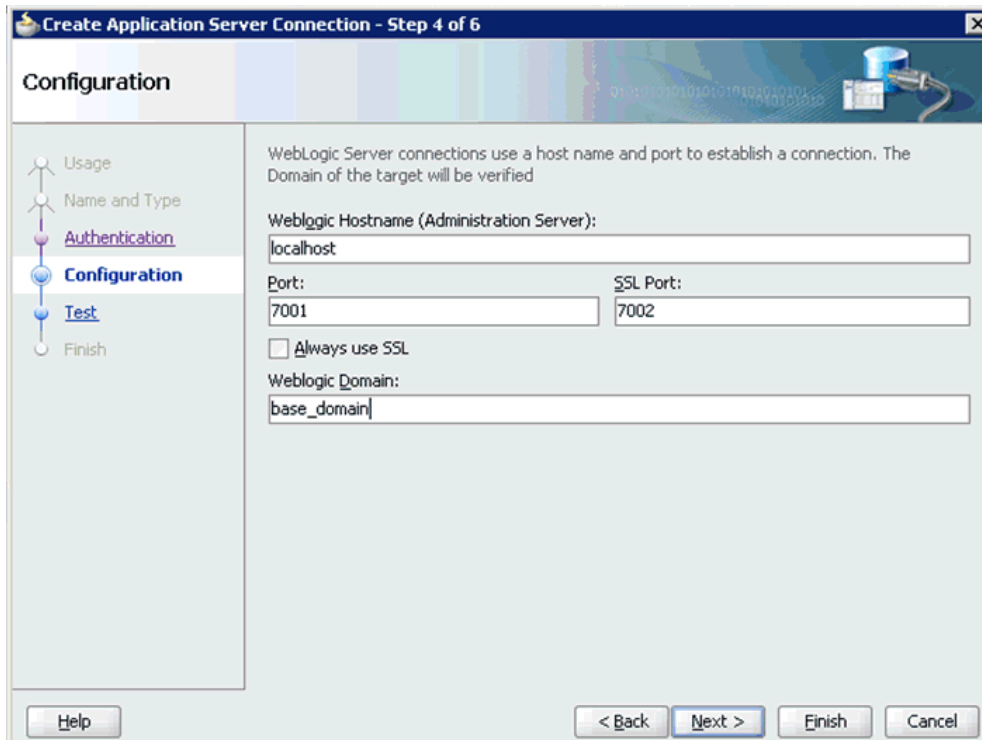
- Accept the default selection (Standalone Server) and click **Next**.
The Name and Type page is displayed.



- Specify a new name for the Application Server connection and click **Next**.
The Authentication page is displayed.

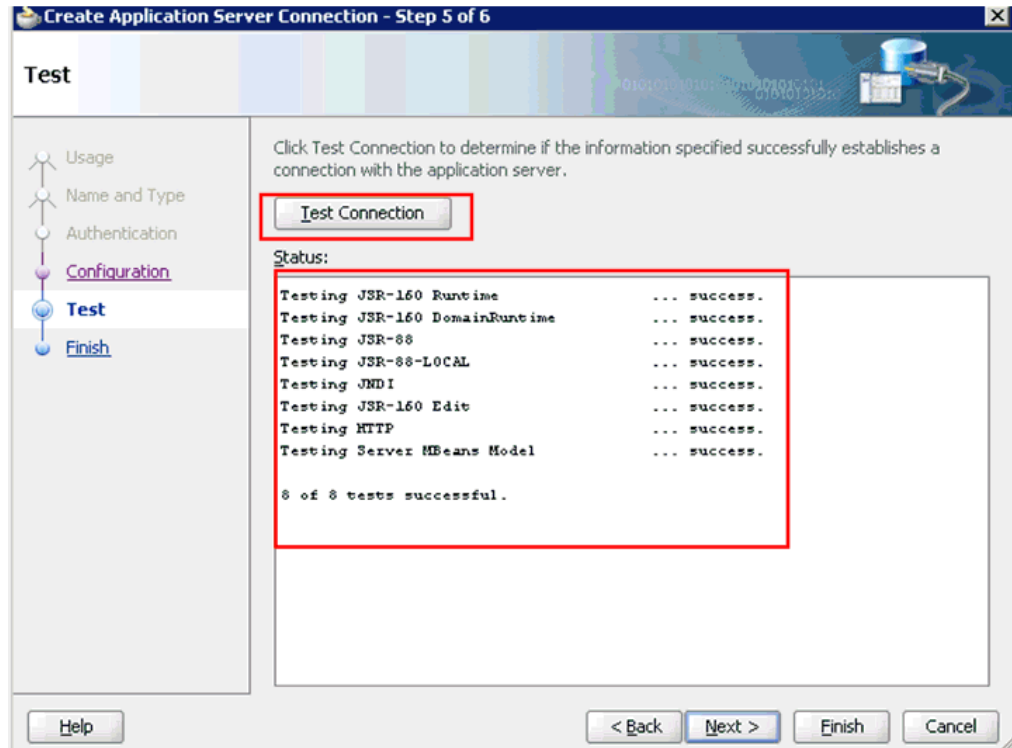


6. Specify a valid user name (for example, weblogic) and a password (for example, welcome1) for your new connection.
7. Click Next.
The Configuration page is displayed.



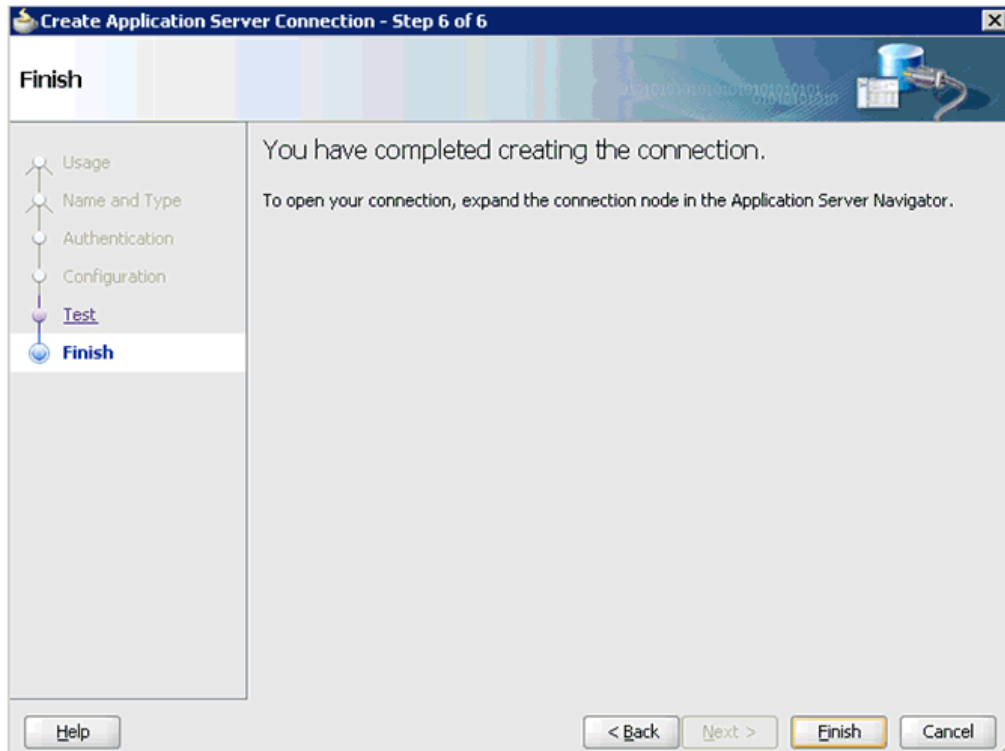
8. Specify the Oracle WebLogic host name (for example, localhost), which is the machine IP where the process needs to deploy and Oracle WebLogic domain (for example, base_domain).
9. Click **Next**.

The Test page is displayed.



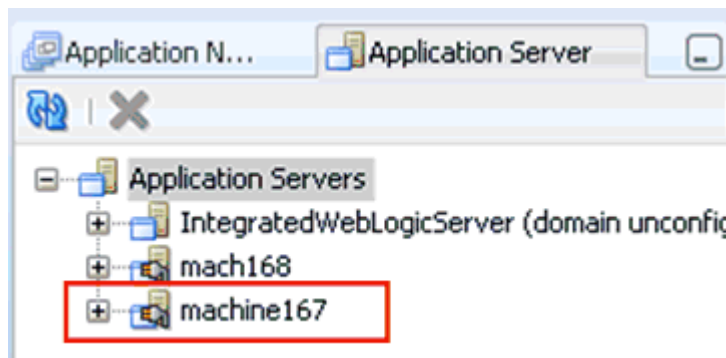
10. Click **Test Connection**.
11. Ensure that the test status is successful.
12. Click **Next**.

The Finish page is displayed.



13. Click Finish.

The new Application Server connection is listed in the left pane (Application Server tab), as shown in the following image.



4.4 Designing an Outbound BPEL Process for Service Integration

The following tools are required to complete your adapter design-time configuration:

- Oracle Adapter Application Explorer (Application Explorer)
- Oracle JDeveloper BPEL Designer (JDeveloper) or Eclipse

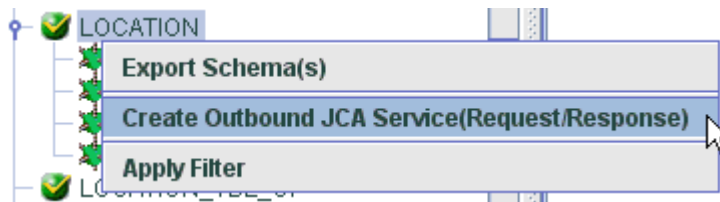
Note: The examples in this chapter demonstrate the use of JDeveloper.

Before you design a BPEL process, you must create a schema and generate the respective WSDL file using Application Explorer. See ["Generating WSDL for Request/Response Service"](#) on page 4-7 for more information.

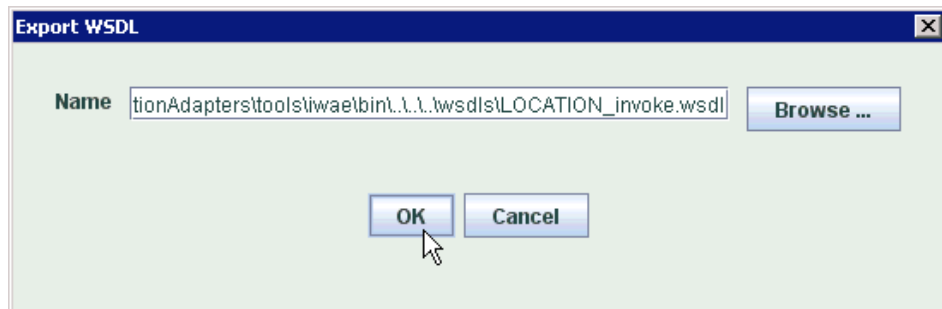
4.4.1 Generating WSDL for Request/Response Service

To generate WSDL for outbound interaction in Application Explorer:

1. Start **Application Explorer** and connect to a defined PeopleSoft target or create a new target.
See ["Defining a Target to PeopleSoft"](#) on page 2-11 for more information.
2. Expand the PeopleSoft target to which you are connected.
3. Expand **Component Interfaces** and select **LOCATION**.
4. Right-click **LOCATION**, and then select **Create Outbound JCA Service (Request/Response)**.



The Export WSDL dialog is displayed.



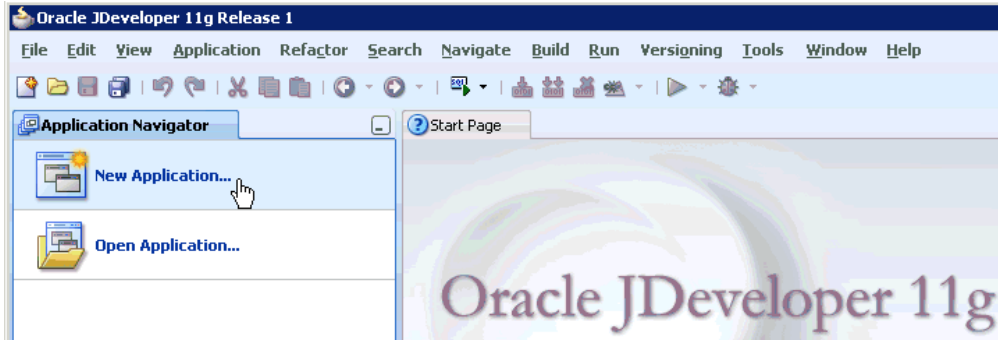
5. Specify an export location on your file system or accept the default path.
The **.wsdl** file extension is added automatically. By default, the names of WSDL files generated for request-response services end with **_invoke**.
6. Click **OK**.

You can now create a new SOA application, which is the first step that is required to define a BPEL outbound process in JDeveloper.

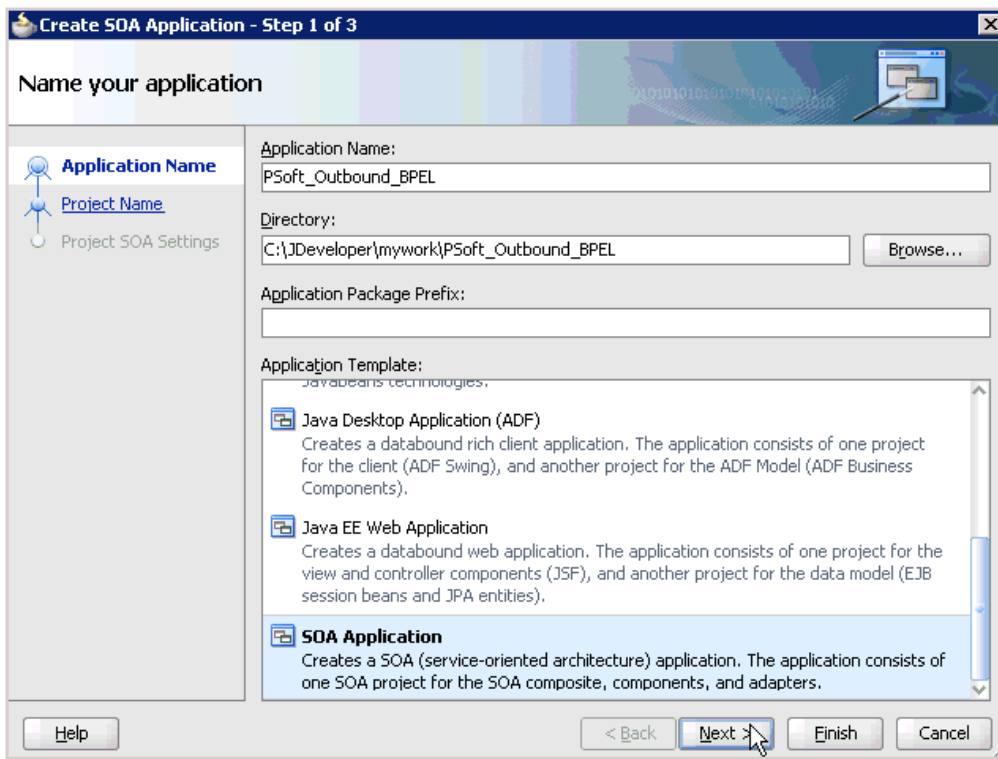
4.4.2 Creating a New SOA Application for the Outbound BPEL Process

Perform the following steps to create a new SOA application for the outbound BPEL process:

1. Open Oracle JDeveloper on your system.
2. In the Application Navigator tab, click **New Application**.

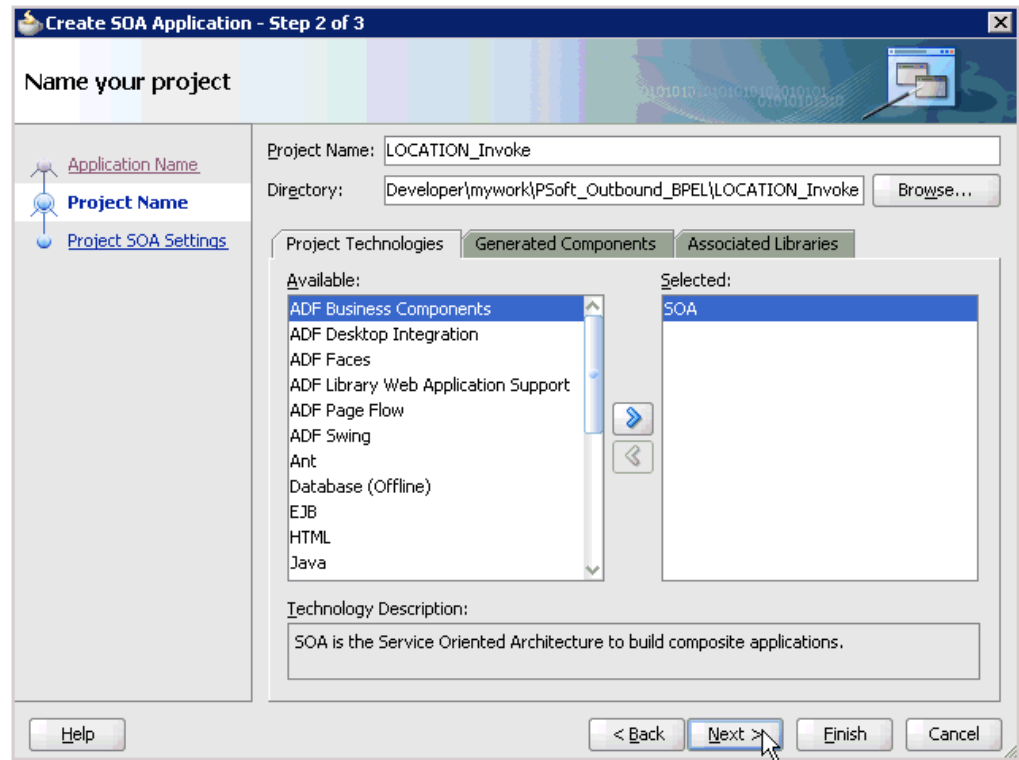


The Create SOA Application wizard is displayed.

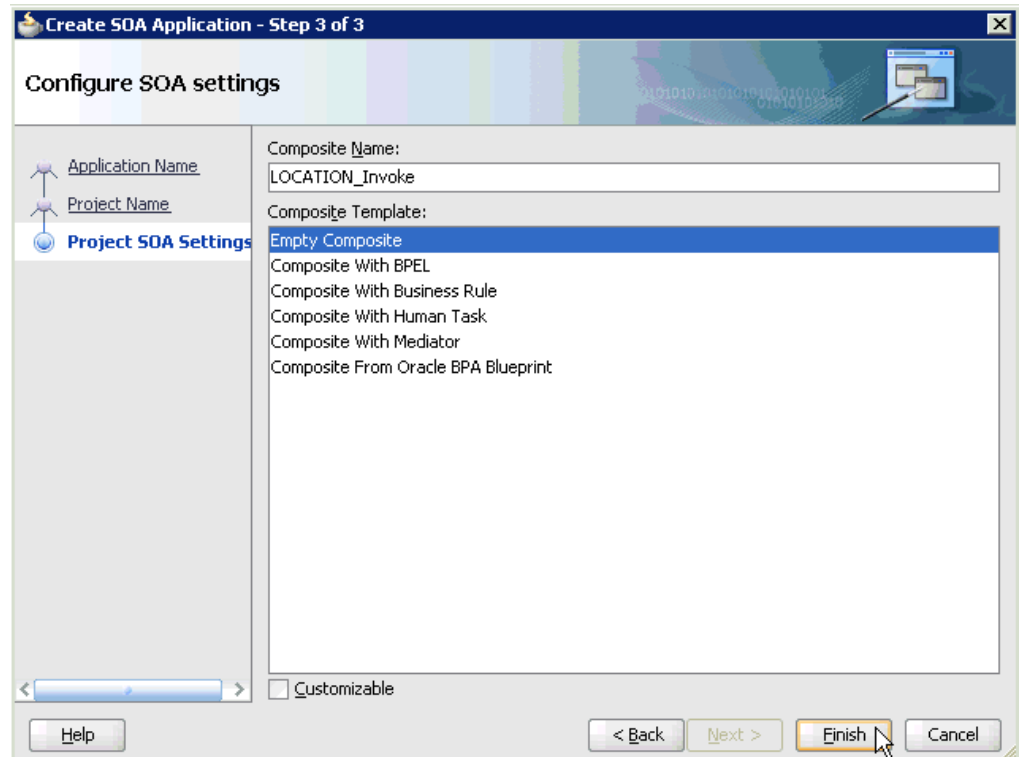


3. From the Application Template list, click **SOA Application**.
4. Enter name for the new SOA application (for example, PSoft_Outbound_BPEL) and click **Next**.

The Name your project page is displayed.

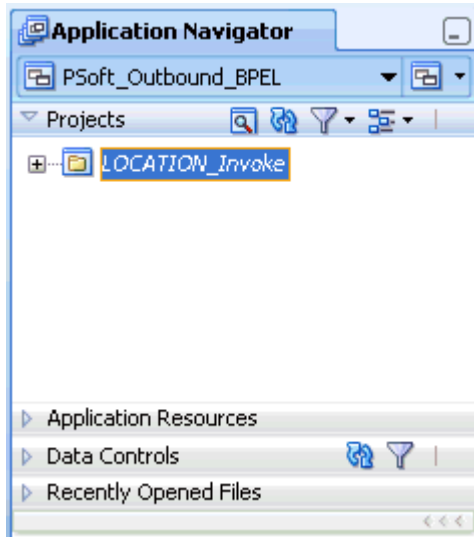


5. Enter a project name (for example, LOCATION_Invoke) and click **Next**.
The Configure SOA settings page is displayed.



6. From the Composite Template list, select **Empty Composite** and click **Finish**.

The new SOA application (PSoft_Outbound_BPEL) and associated project (LOCATION_Invoke) are added to the Application Navigator tab in the left pane.



4.4.3 Defining a BPEL Outbound Process

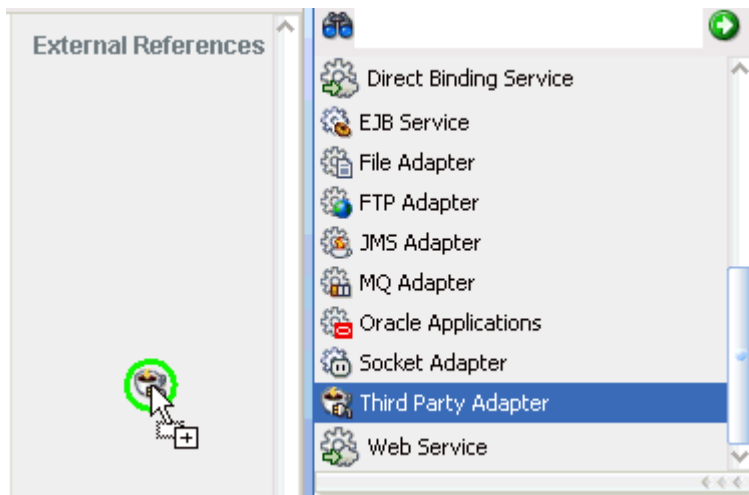
This section describes how to define a BPEL outbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Outbound BPEL Process Component

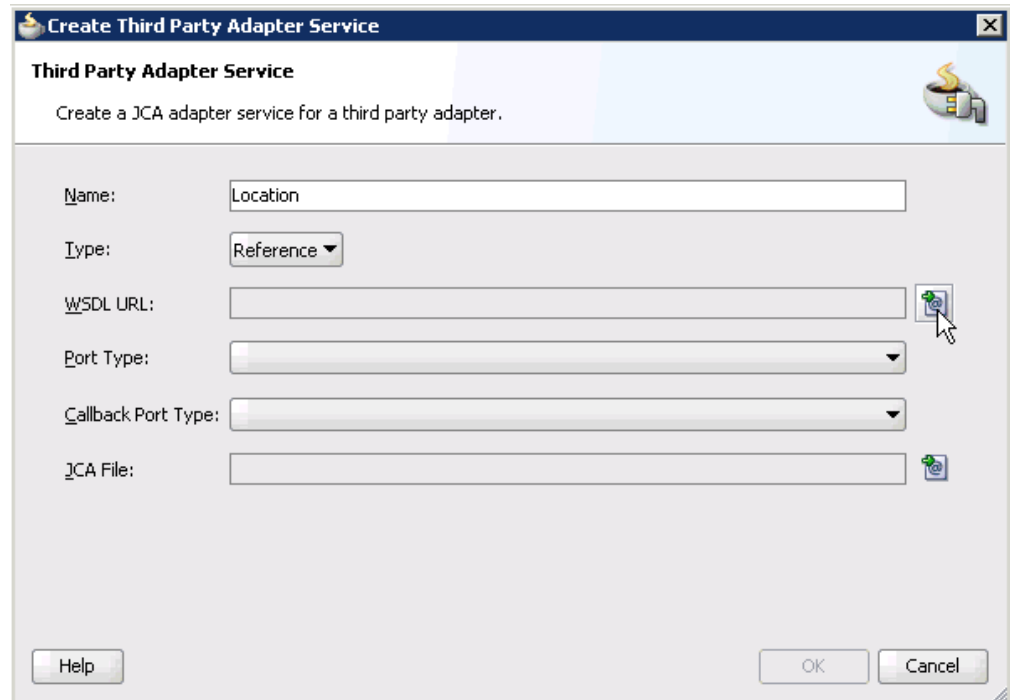
Configuring a Third Party Adapter Service Component

Perform the following steps to create a third party adapter service component:

1. Drag and drop the **Third Party Adapter** component from the Component Palette tab (Service Adapters section) to the External References pane.

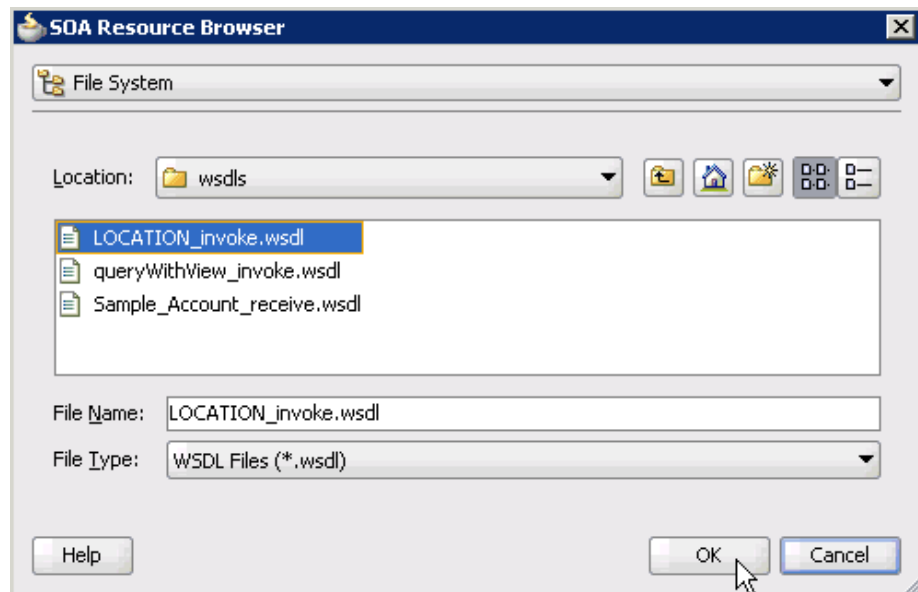


The Create Third Party Adapter Service dialog is displayed.



2. Enter a name for the new third party adapter service.
3. Ensure that **Reference** is selected from the Type list (default).
4. Click the **Find existing WSDLs** icon, which is located to the right of the WSDL URL field.

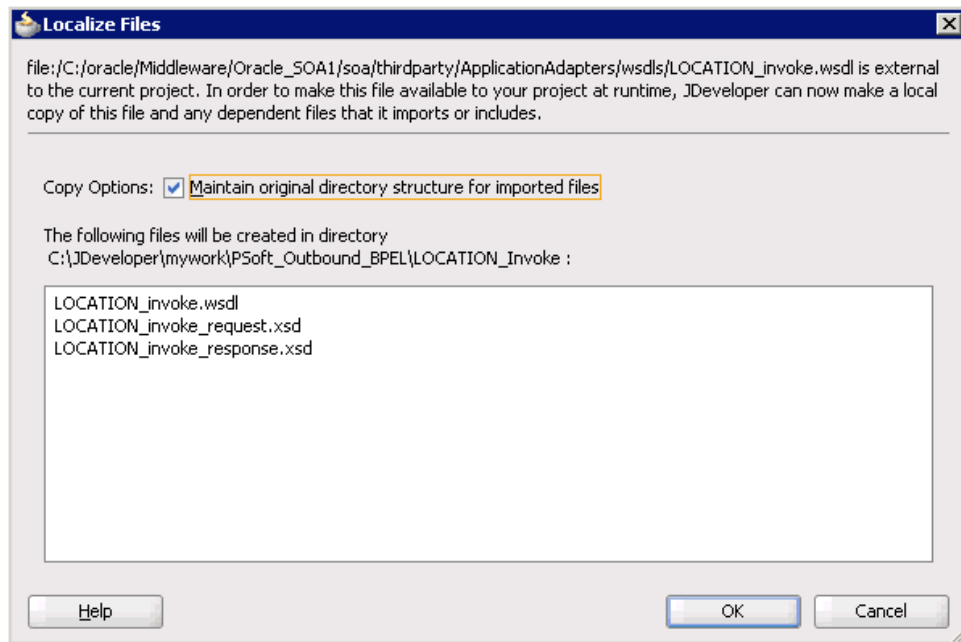
The SOA Resource Browser dialog is displayed.



5. Browse and select an outbound WSDL file (for example, LOCATION_invoke.wsdl) from the following directory:


```
C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\wsdls
```
6. Click **OK**.

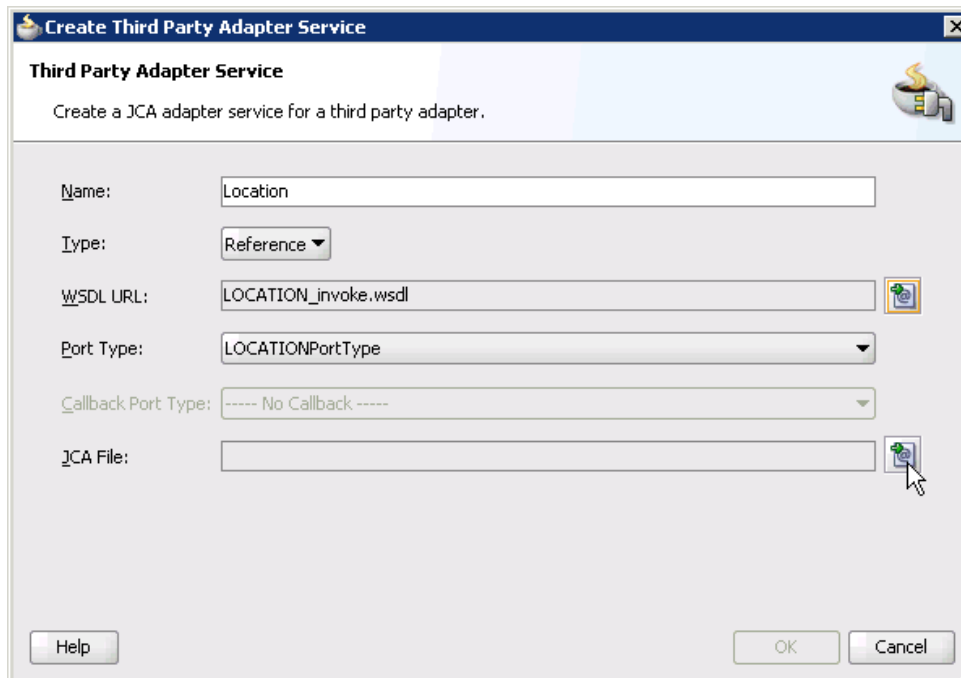
The Localize Files dialog is displayed.



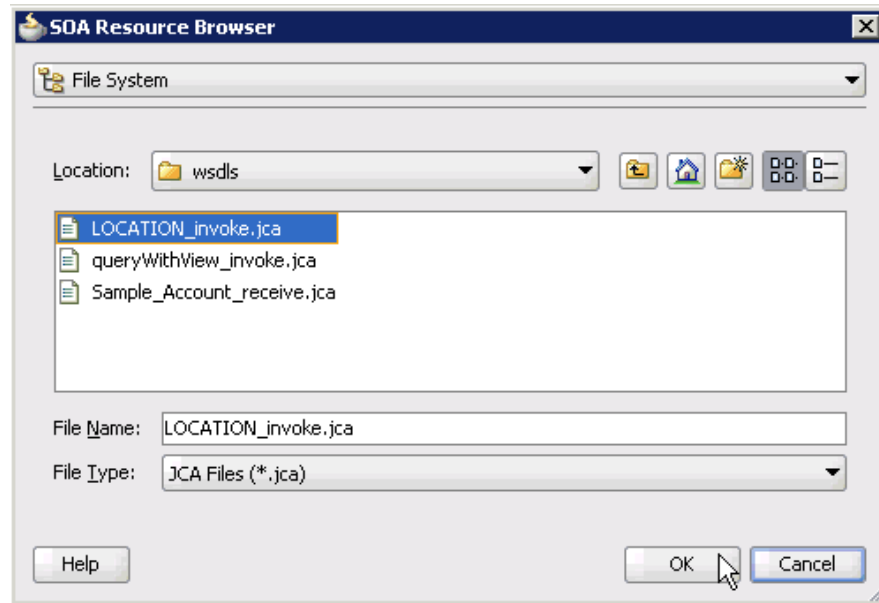
7. Click **OK**.

The outbound WSDL file and associated request and response XML schema files (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog.



8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed.

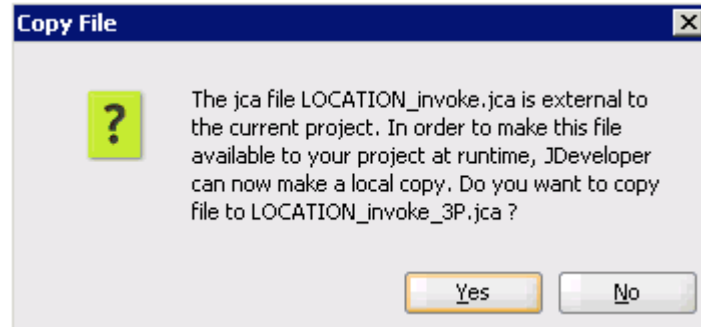


9. Browse and select the JCA properties file (for example, LOCATION_invoke.jca) from the following directory:

C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\wsdls

10. Click **OK**.

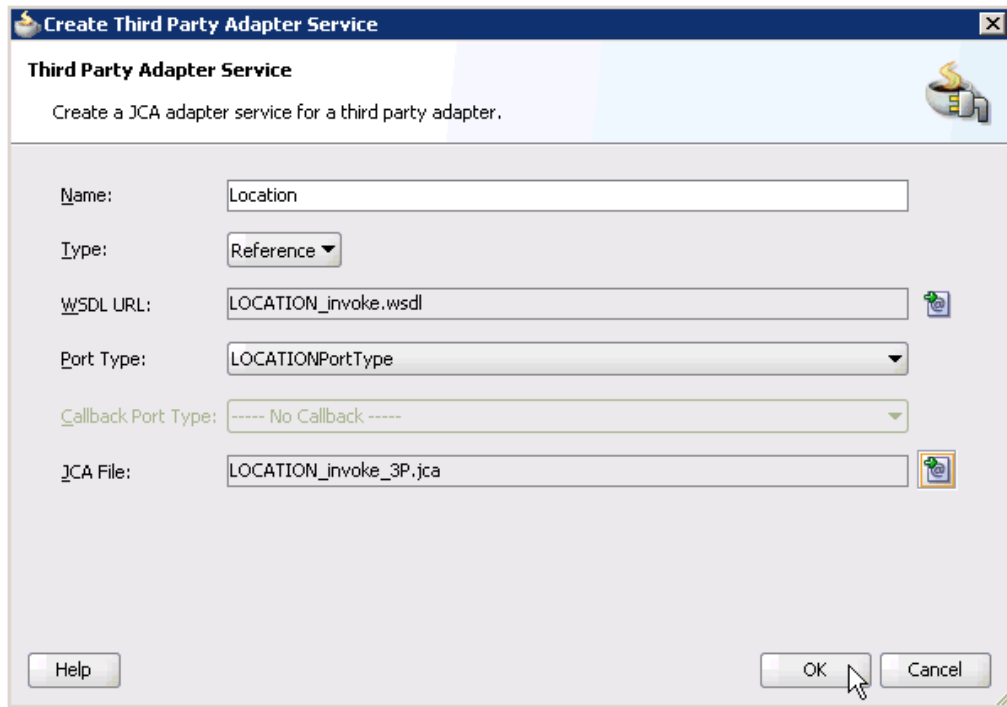
The following message is displayed.



11. Click **Yes**.

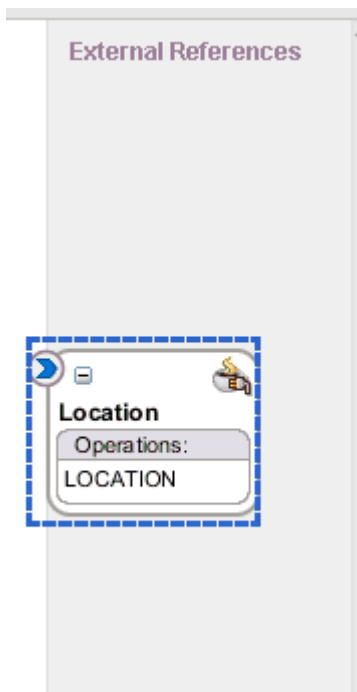
A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog.



12. Click **OK**.

The third party adapter service component (Location) is created in the External References pane, as shown in the following image.

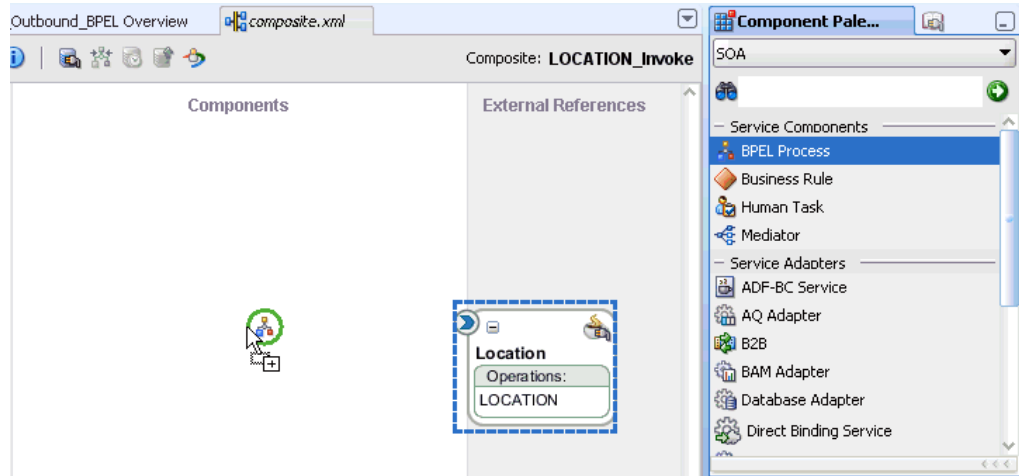


You are now ready to configure an outbound BPEL process component.

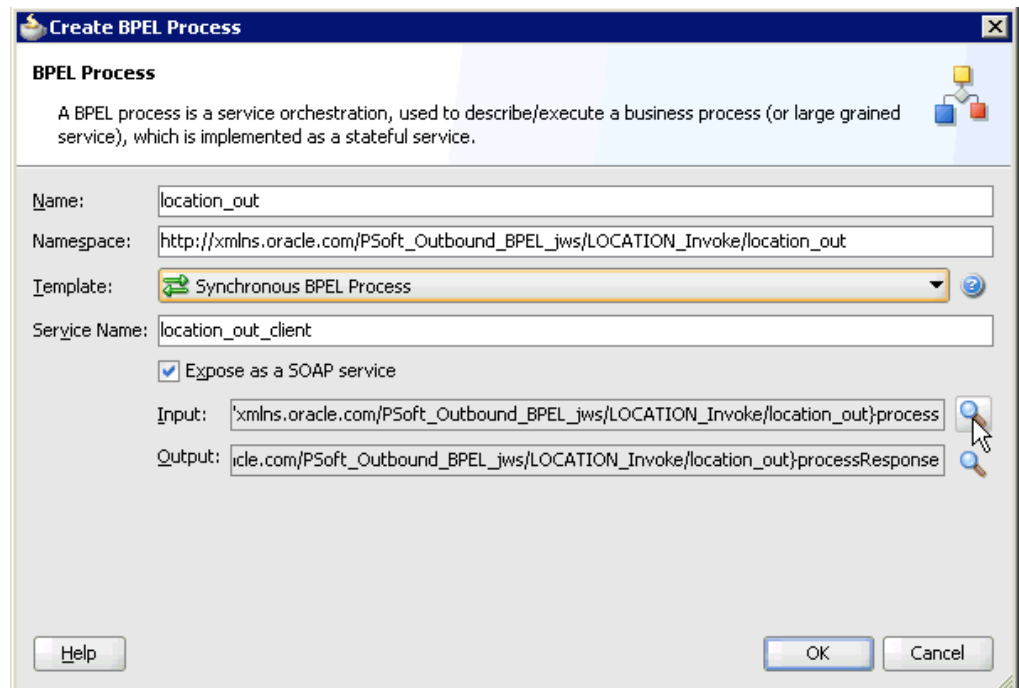
Configuring an Outbound BPEL Process Component

Perform the following steps to configure an outbound BPEL process component:

1. Drag and drop the **BPEL Process** component from the Component Palette tab (Service Components section) to the Components pane.

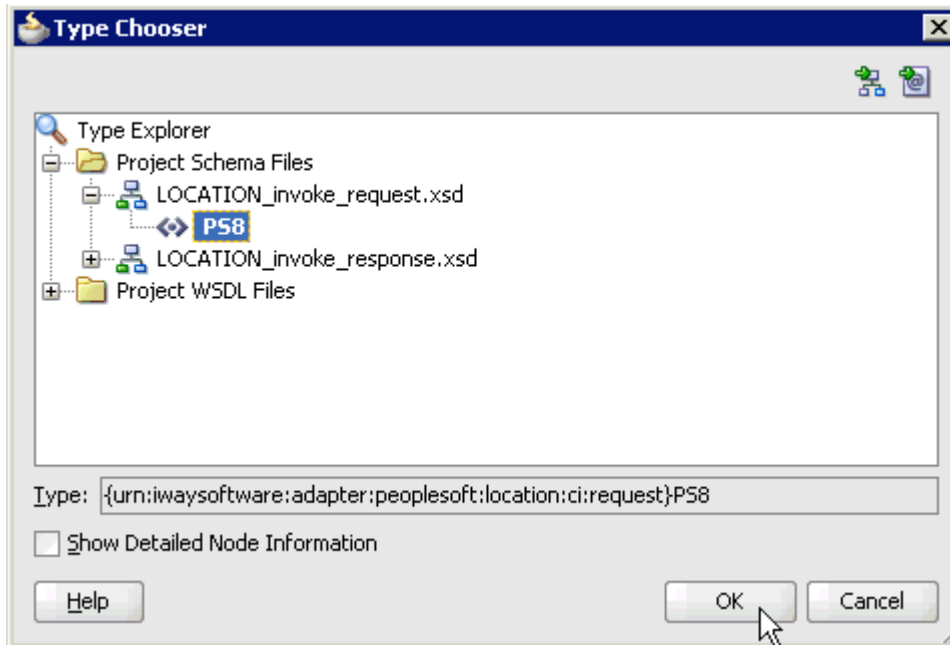


The Create BPEL Process dialog is displayed.



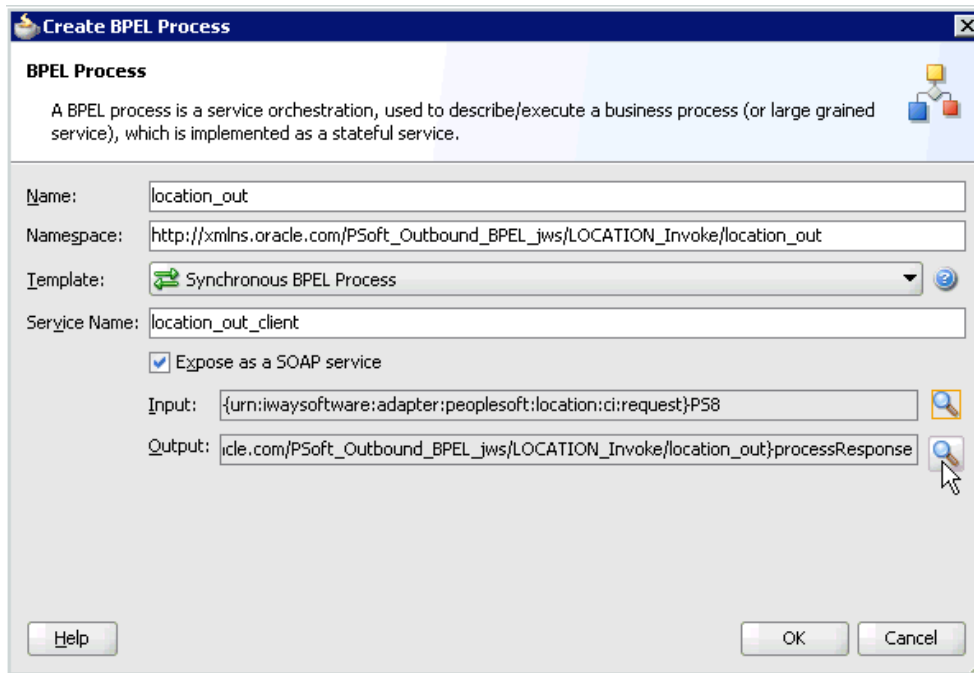
2. In the Name field, enter a name to identify the new outbound BPEL process component (for example, location_out).
3. From the Template list, select **Synchronous BPEL Process**.
4. Click the **Browse Input Elements** icon, which is located to the right of the Input field to select the associated XML request schema file.

The Type Chooser dialog is displayed.



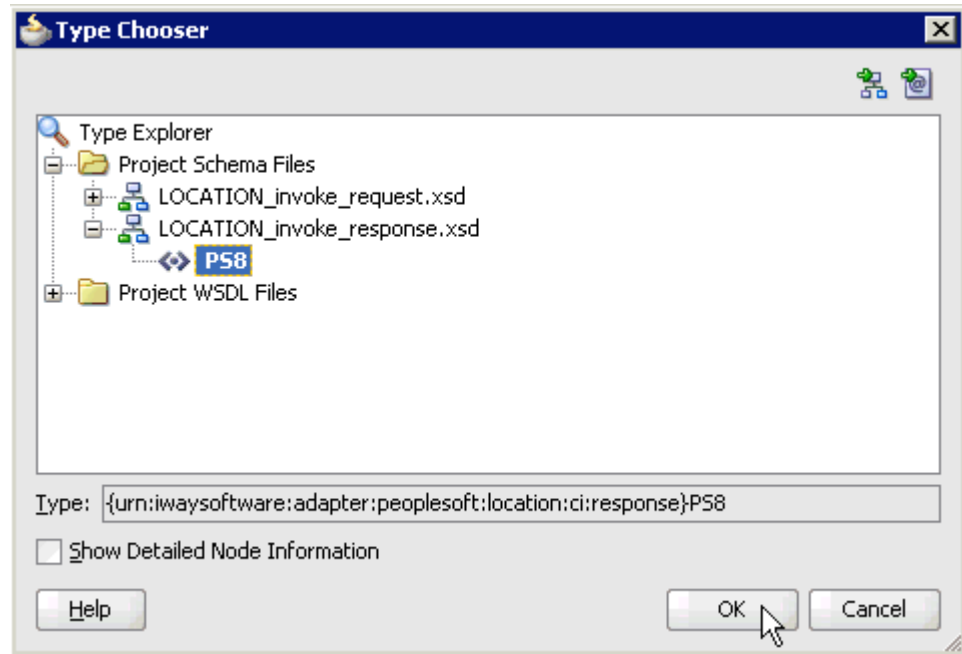
5. Expand **Project Schema Files**, **LOCATION_invoke_request.xsd**, and select **PS8**.
6. Click **OK**.

You are returned to the Create BPEL Process dialog.



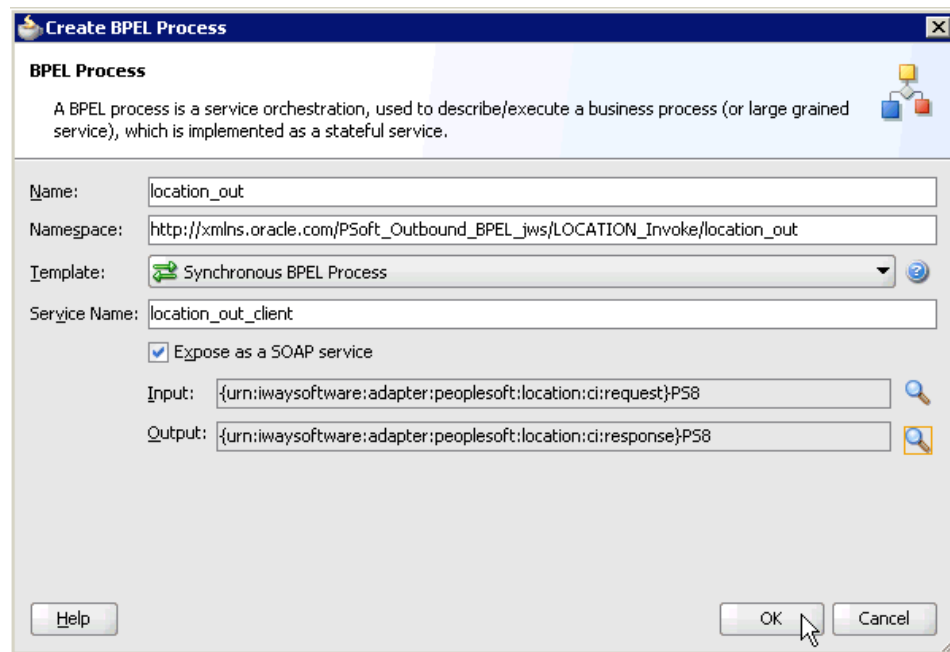
7. Click the **Browse Output Elements** icon, which is located to the right of the Output field to select the associated XML response schema file.

The Type Chooser dialog is displayed.

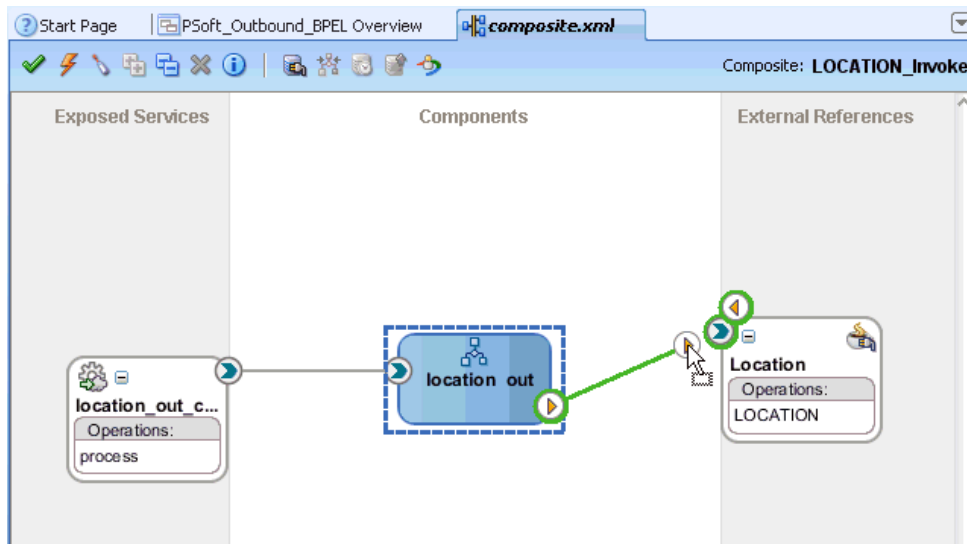


8. Expand **Project Schema Files**, **LOCATION_invoke_response.xsd**, and select **PS8**.
9. Click **OK**.

You are returned to the Create BPEL Process dialog.

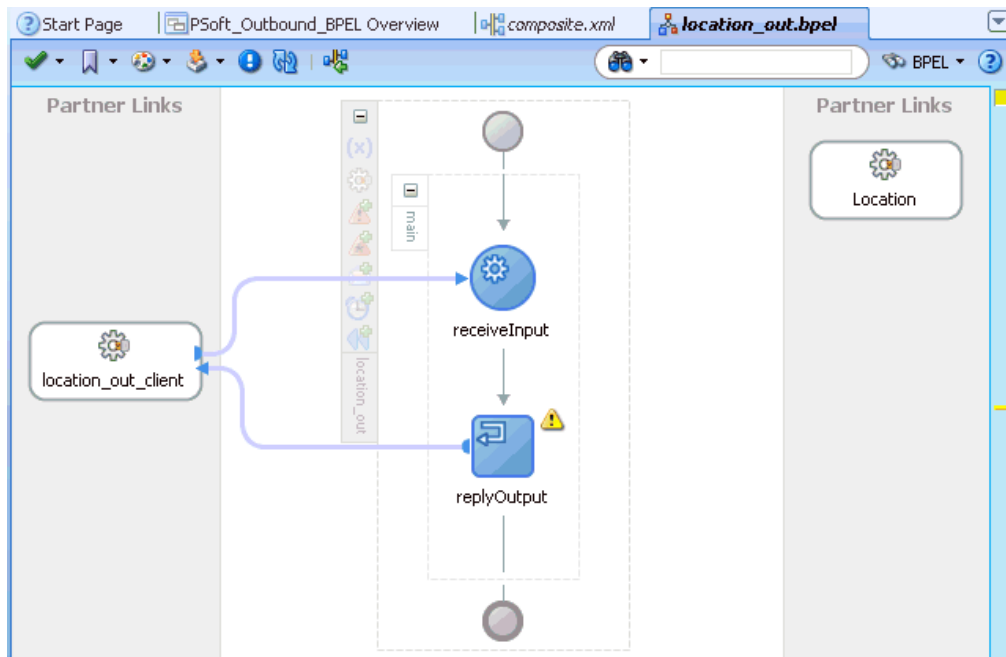


10. Click **OK**.
11. Create a connection between the outbound BPEL process component (**location_out**) and the third party adapter service component (**Location**).

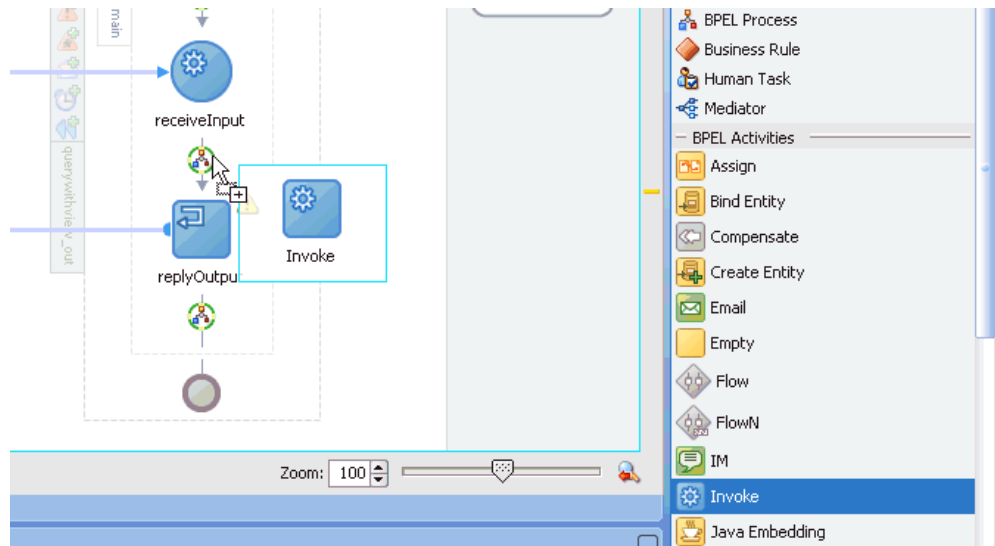


- Double-click the outbound BPEL process component (location_out) in the Components pane.

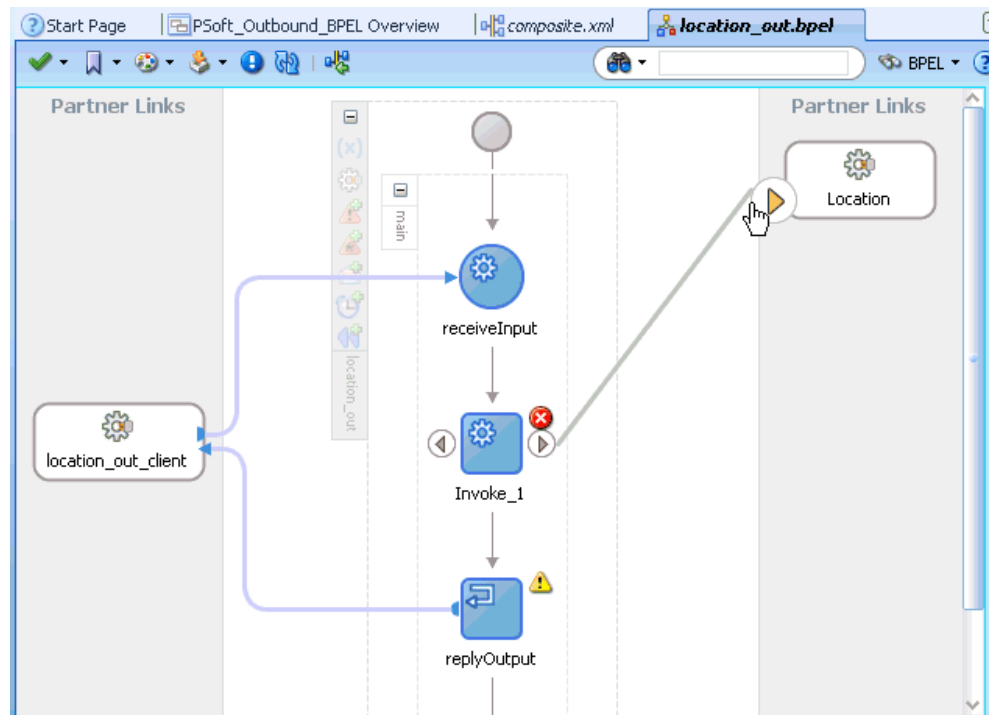
The following is displayed.



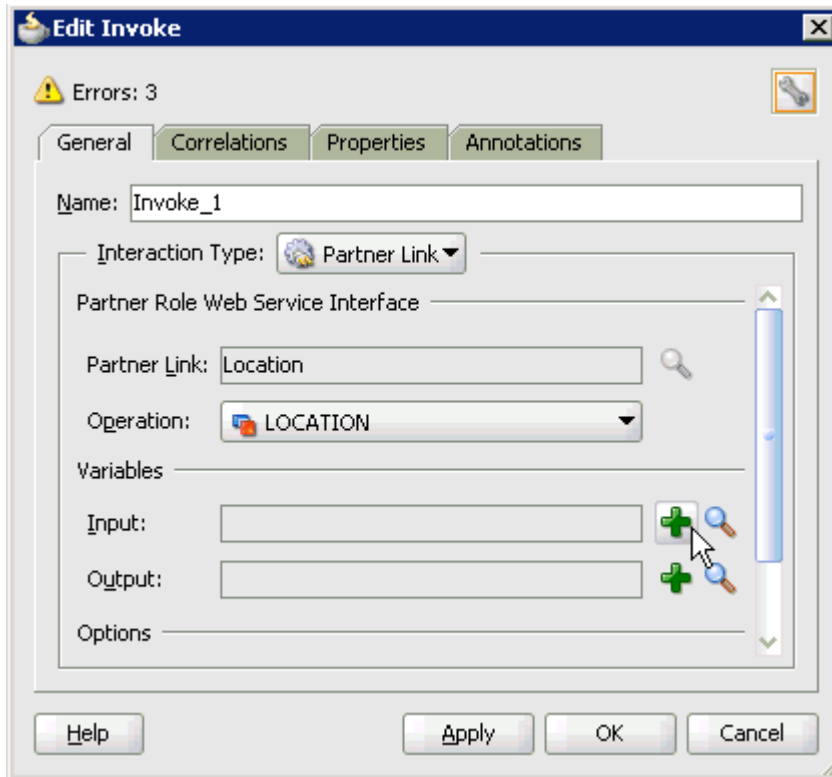
- Drag and drop the **Invoke** activity component to the Components pane and place it between the **receiveInput** activity component and the **replyOutput** activity component.



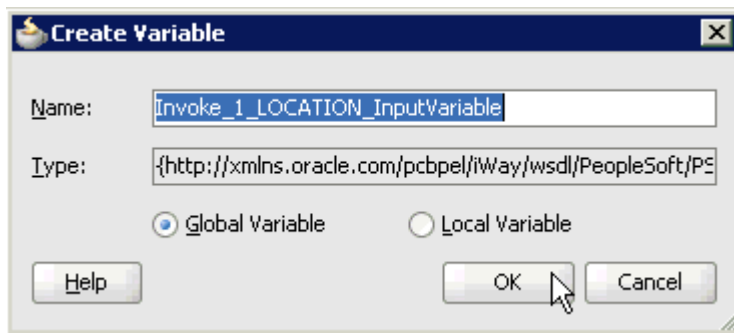
14. Create a connection between the new Invoke activity component (Invoke_1) and the third party adapter service component (Location).



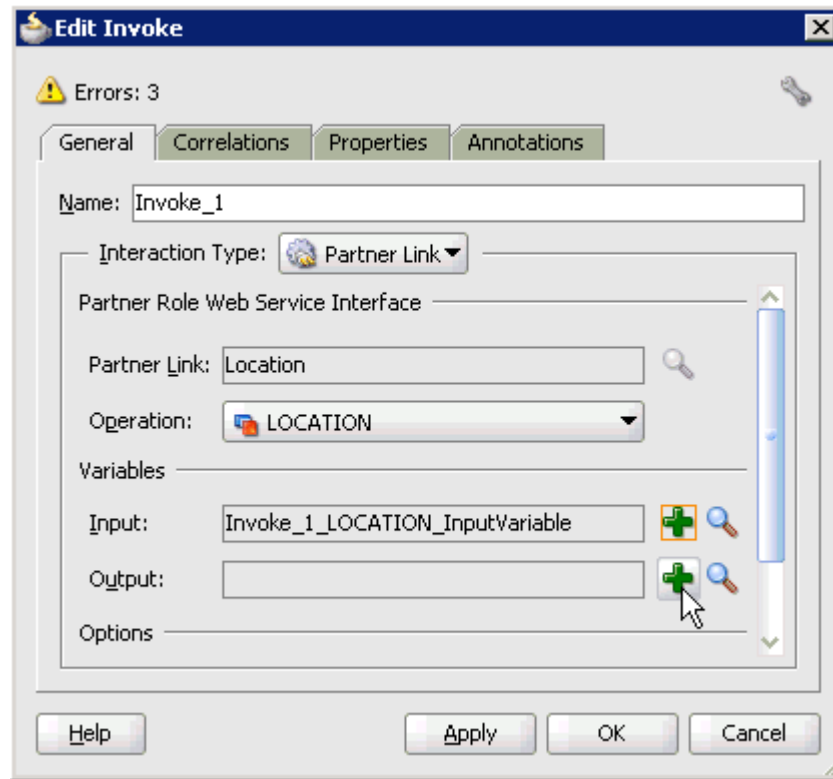
The Edit Invoke dialog is displayed.



15. Click the **Automatically Create Input Variable** icon, which is located to the right of the Input field to configure a new input variable.
The Create Variable dialog is displayed.

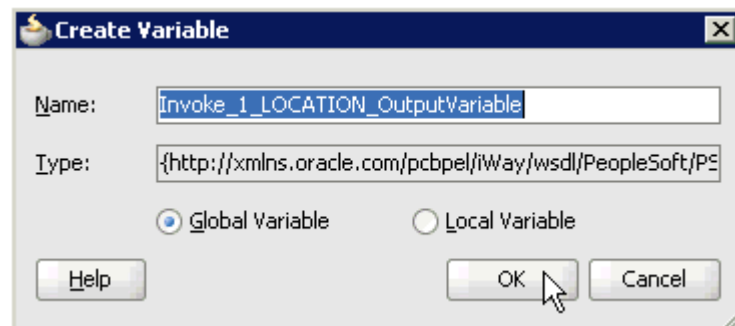


16. Accept the default values that are provided for the new input variable and click **OK**.
You are returned to the Edit Invoke dialog.



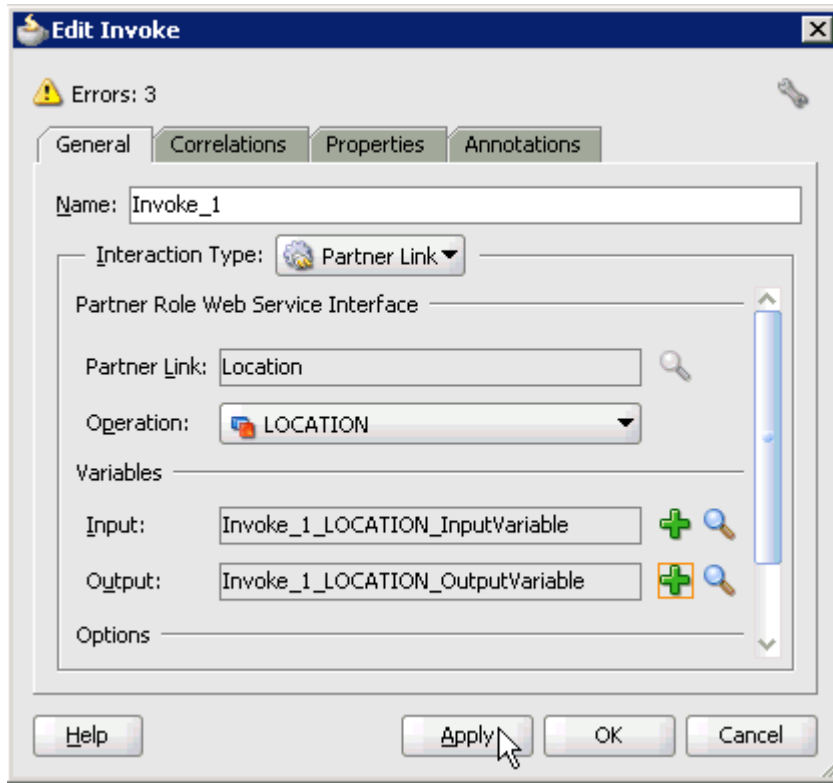
17. Click the **Automatically Create Output Variable** icon, which is located to the right of the Output field to configure a new output variable.

The Create Variable dialog is displayed.



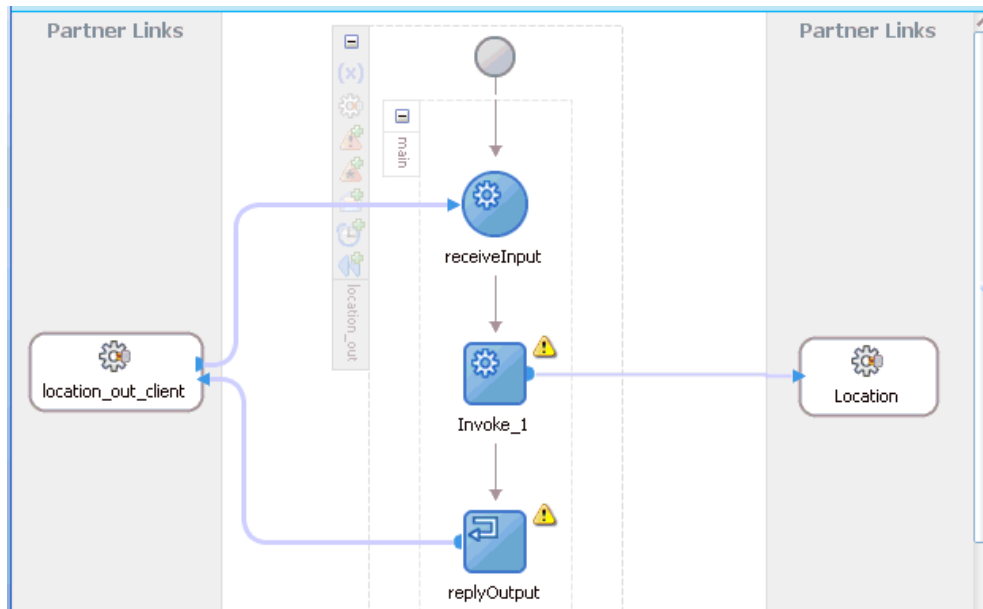
18. Accept the default values that are provided for the new output variable and click **OK**.

You are returned to the Edit Invoke dialog.

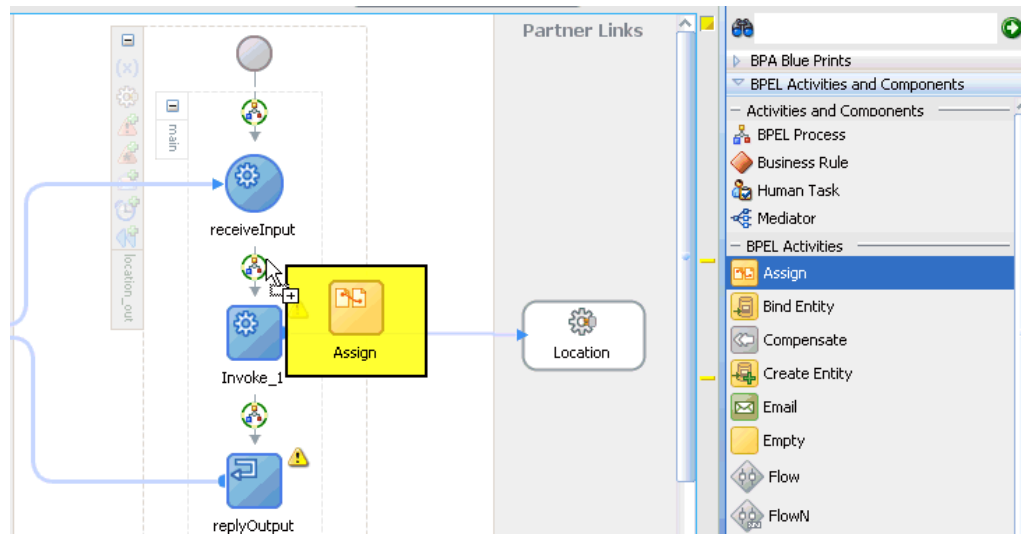


19. Click **Apply** and then **OK**.

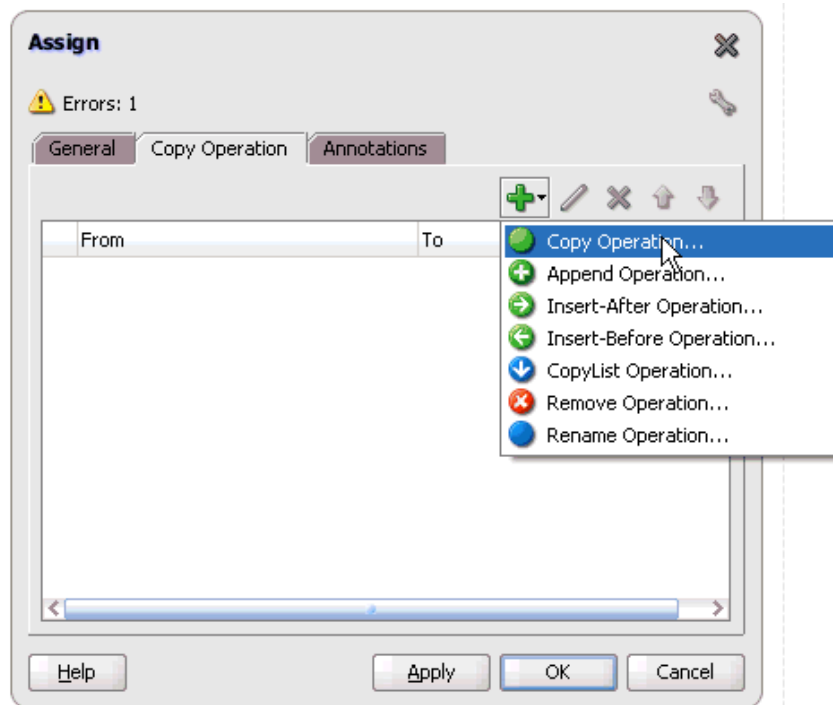
The Invoke activity component (Invoke_1) is updated accordingly.



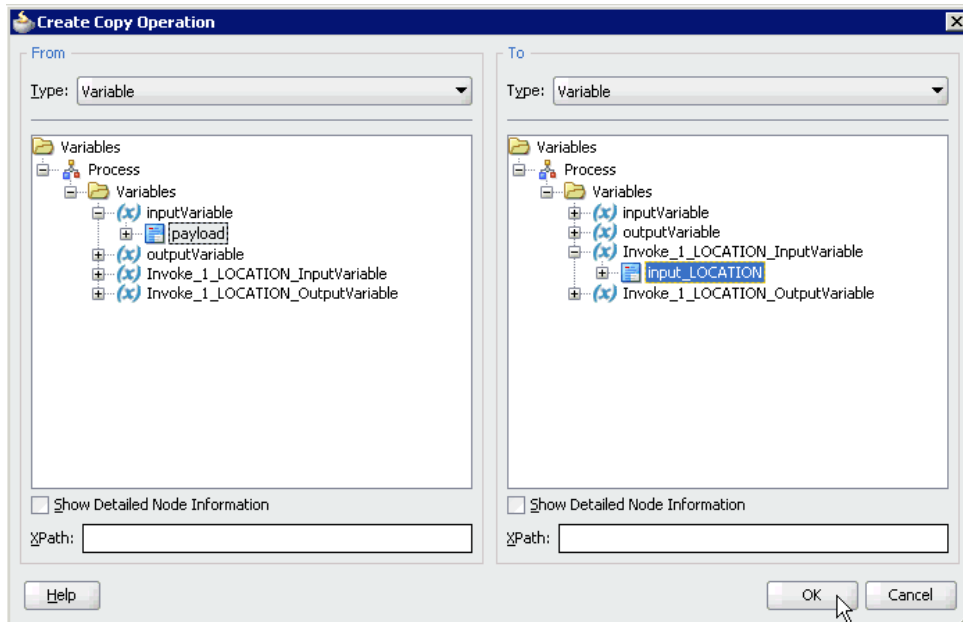
20. Drag and drop the **Assign** activity component to the Components pane and place it between the Receive activity component (receiveInput) and the Invoke activity component (Invoke_1).



21. Double-click the new Assign activity component (**Assign_1**).
The Assign dialog is displayed.

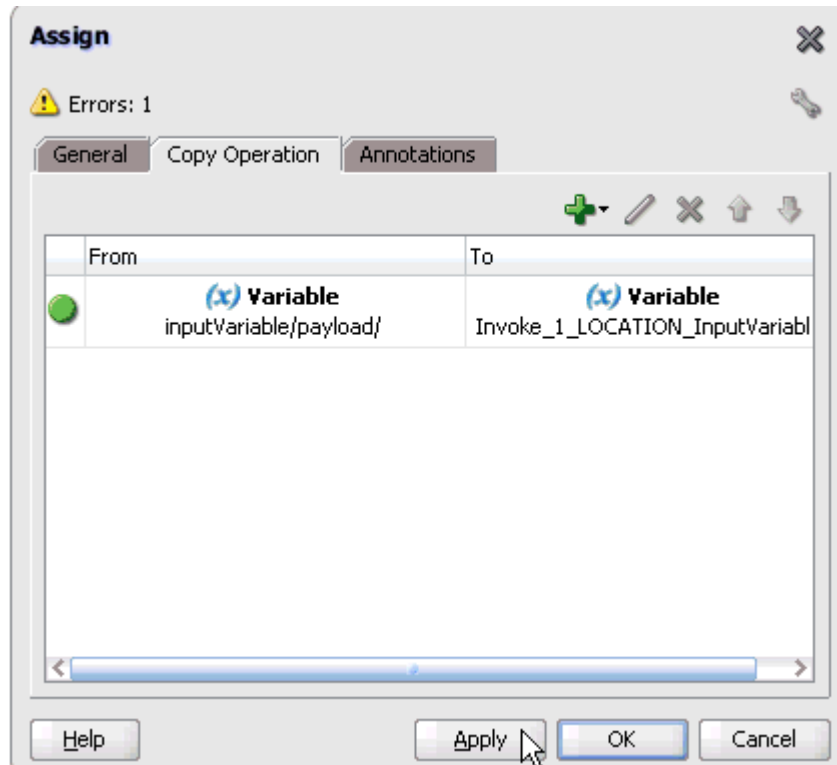


22. Click the **Copy Operation** tab.
23. Click the **Plus sign** icon and select **Copy Operation** from the list of available operations.
The Create Copy Operation dialog is displayed.



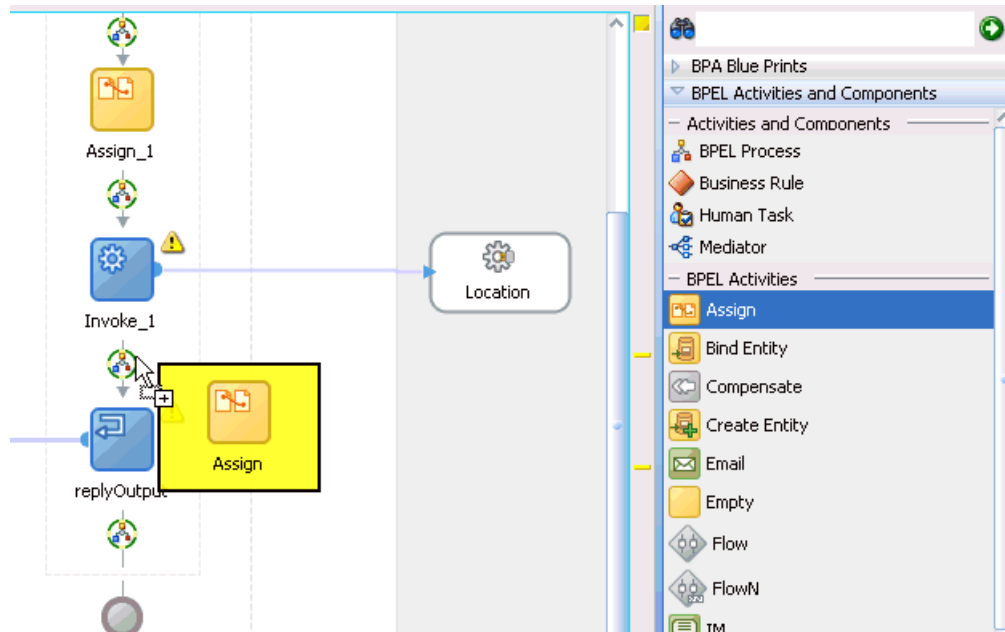
24. In the From pane, expand **Variables**, **InputVariable**, and then select **payload**.
25. In the To pane, expand **Variables**, **Invoke_1_LOCATION_InputVariable**, and then select **input_LOCATION**.
26. Click **OK**.

You are returned to the Assign dialog.

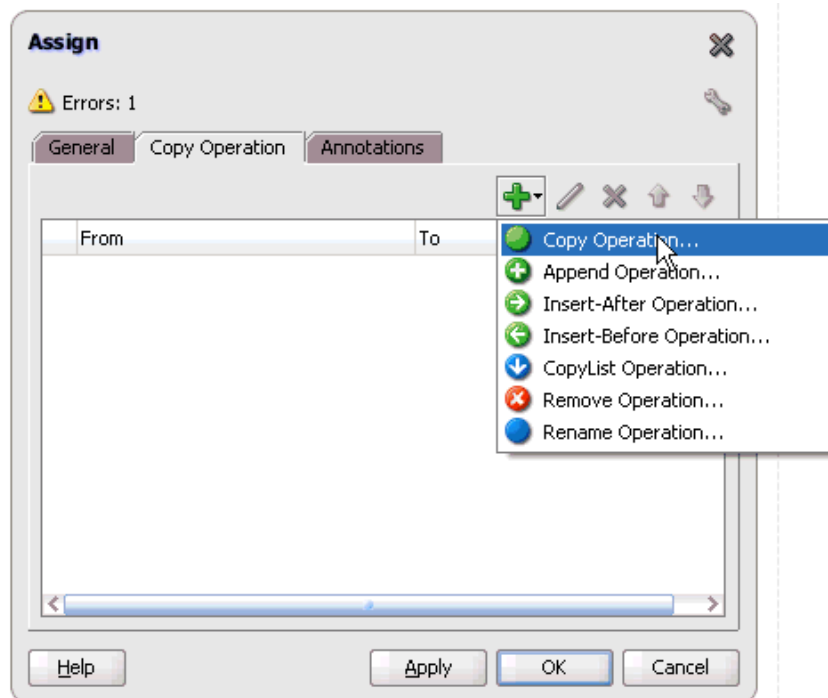


27. Click **Apply** and then **OK**.

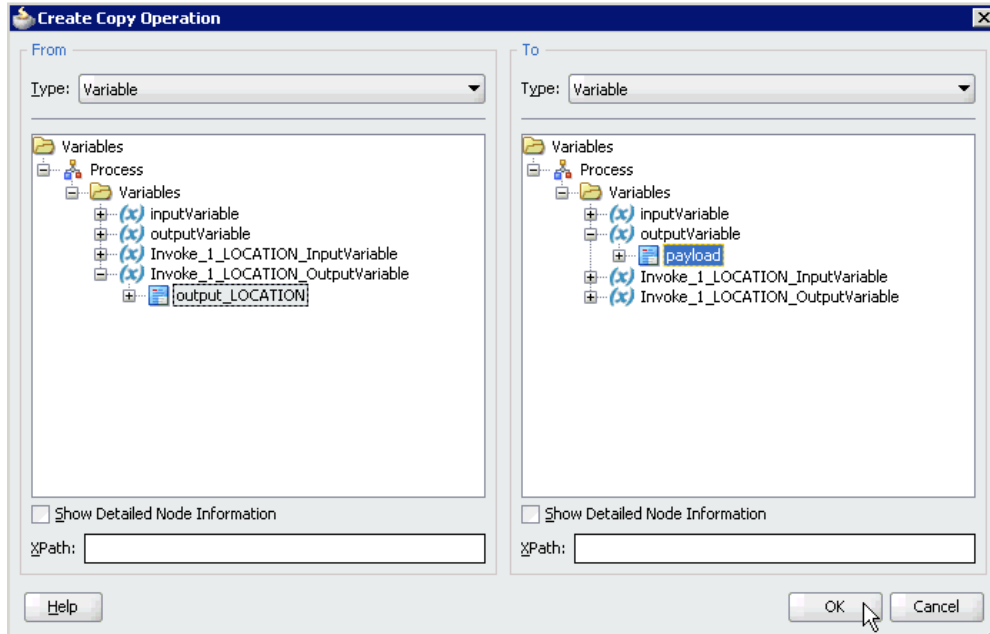
28. Drag and drop the **Assign** activity component to the Components pane and place it between the Invoke activity (Invoke_1) and the Reply activity (replyOutput).



29. Double-click the new Assign activity component (**Assign_2**).
The Assign dialog is displayed.

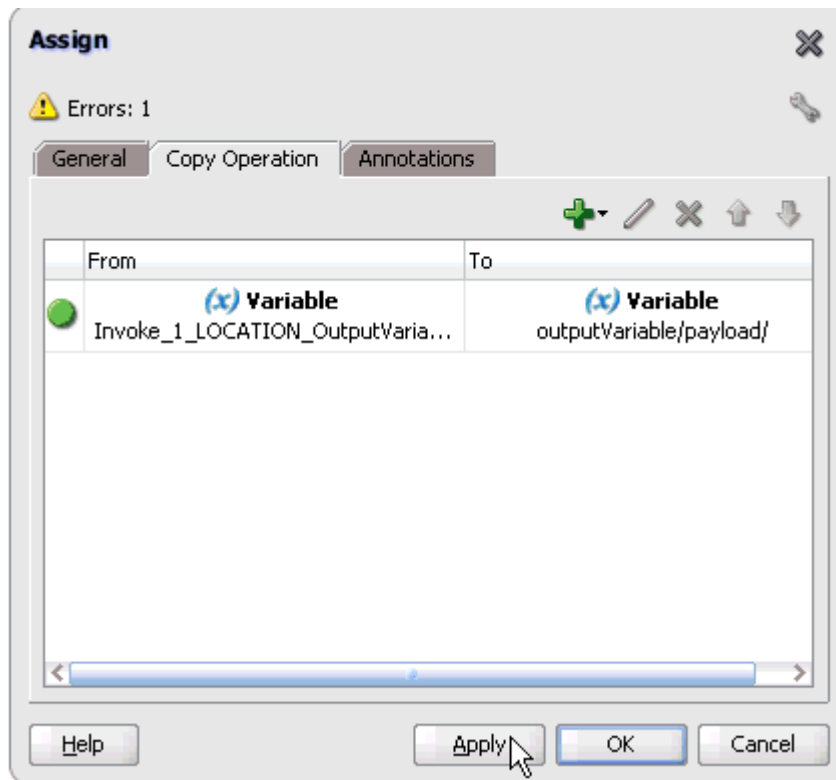


30. Click the **Copy Operation** tab.
31. Click the **Plus sign** icon and select **Copy Operation** from the list of available operations.
The Create Copy Operation dialog is displayed.



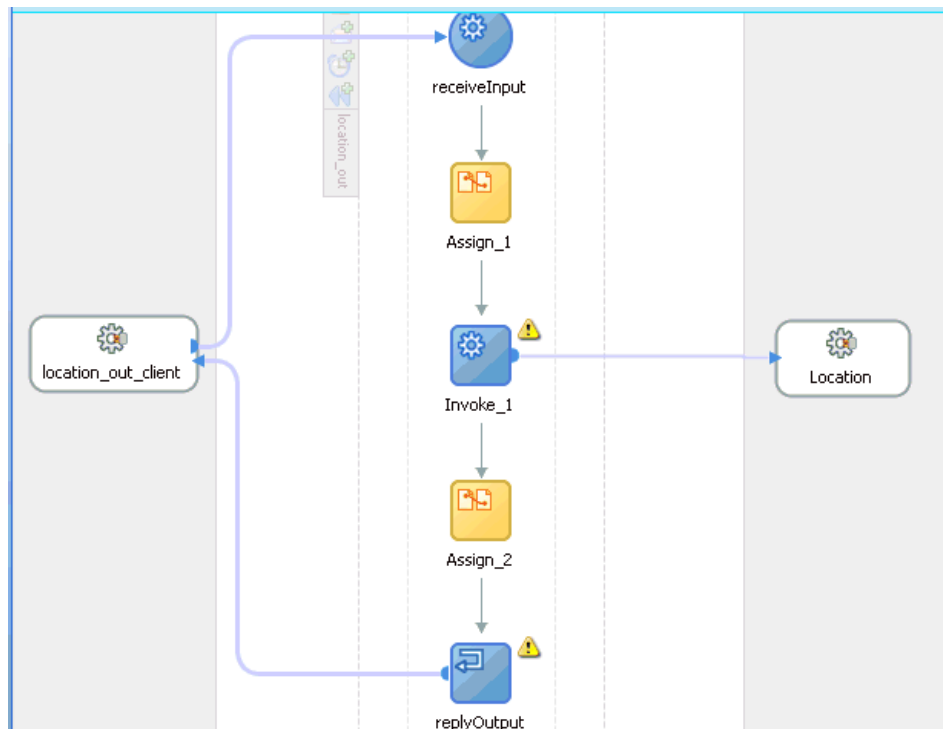
32. In the From pane, expand **Variables**, **Invoke_1_LOCATION_OutputVariable**, and then select **output_LOCATION**.
33. In the To pane, expand **Variables**, **outputVariable**, and then select **payload**.
34. Click **OK**.

You are returned to the Assign dialog.

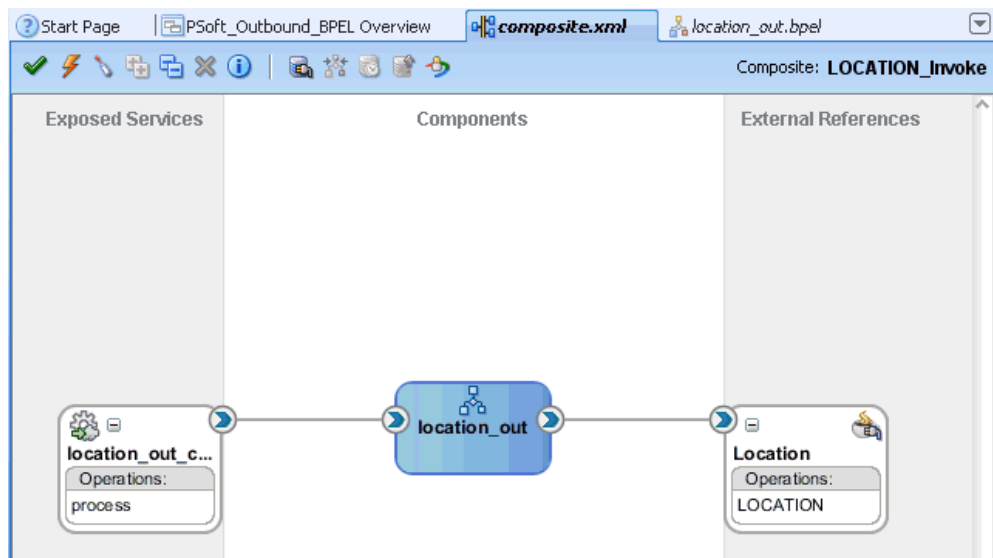


35. Click **Apply** and then **OK**.

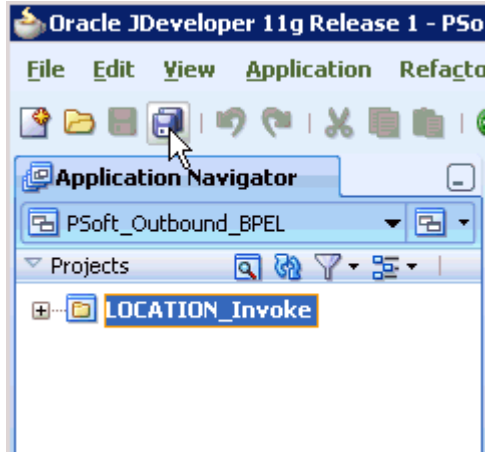
The completed activity flow is now displayed.



36. Double-click **composite.xml** in the left pane.



37. Click the **Save All** icon in the menu bar to save the new outbound BPEL process component that was configured.

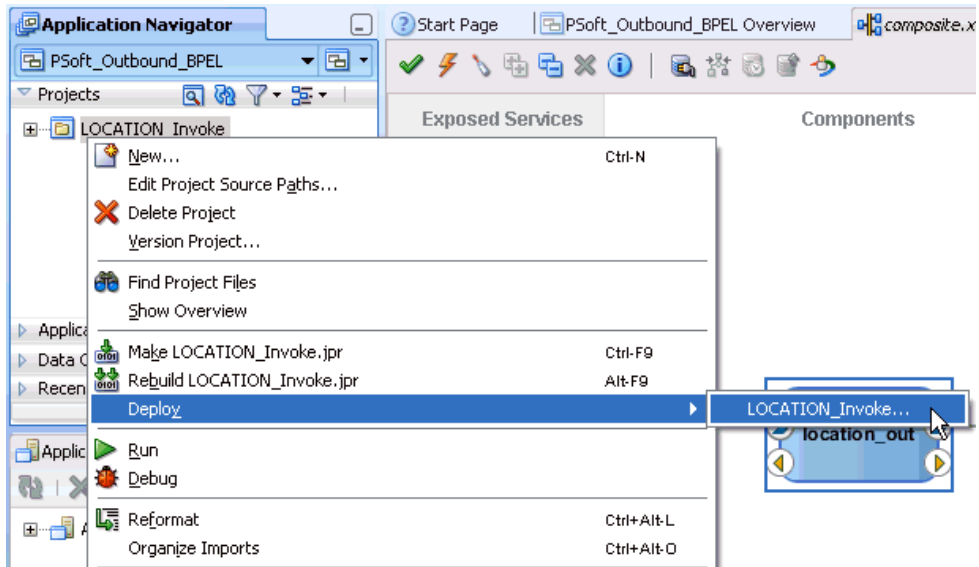


You are now ready to deploy the BPEL outbound process.

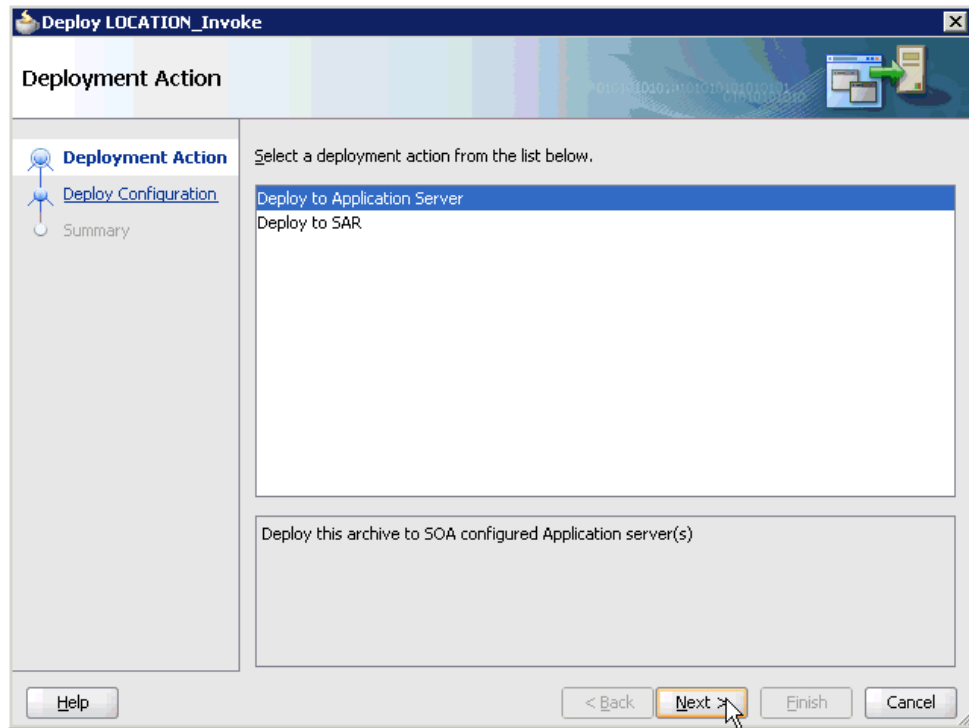
4.4.4 Deploying the BPEL Outbound Process

Perform the following steps to deploy the BPEL outbound process.

1. Right-click the project name in the left pane (for example, **LOCATION_Invoke**), select **Deploy**, and then click **LOCATION_Invoke**.



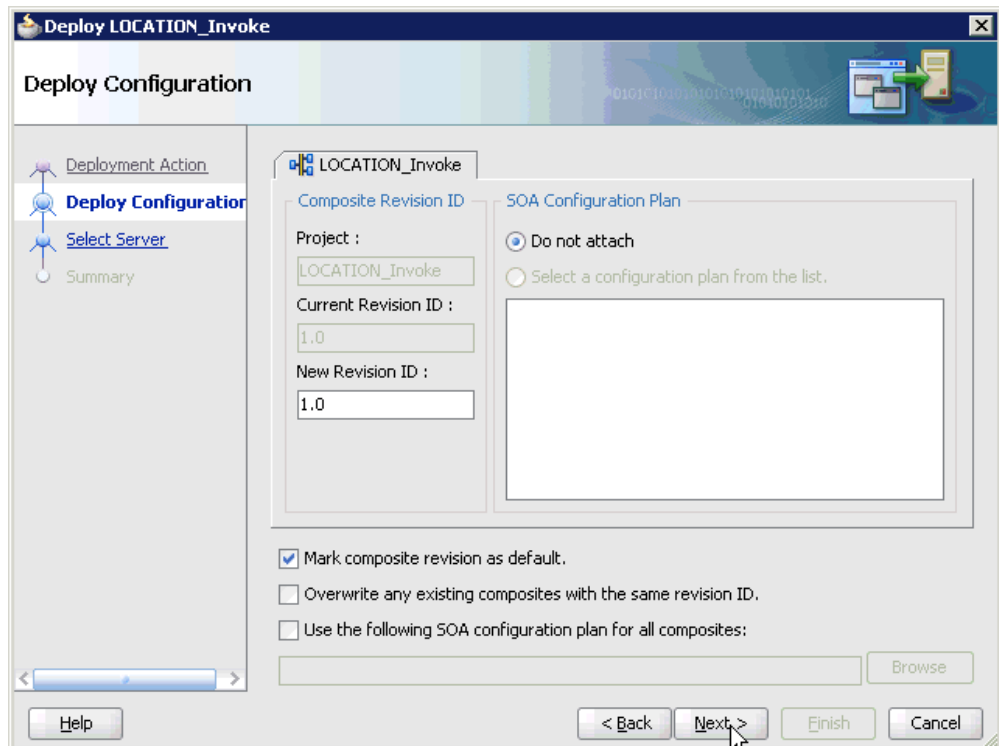
The Deployment Action page is displayed.



2. Ensure that **Deploy to Application Server** is selected.

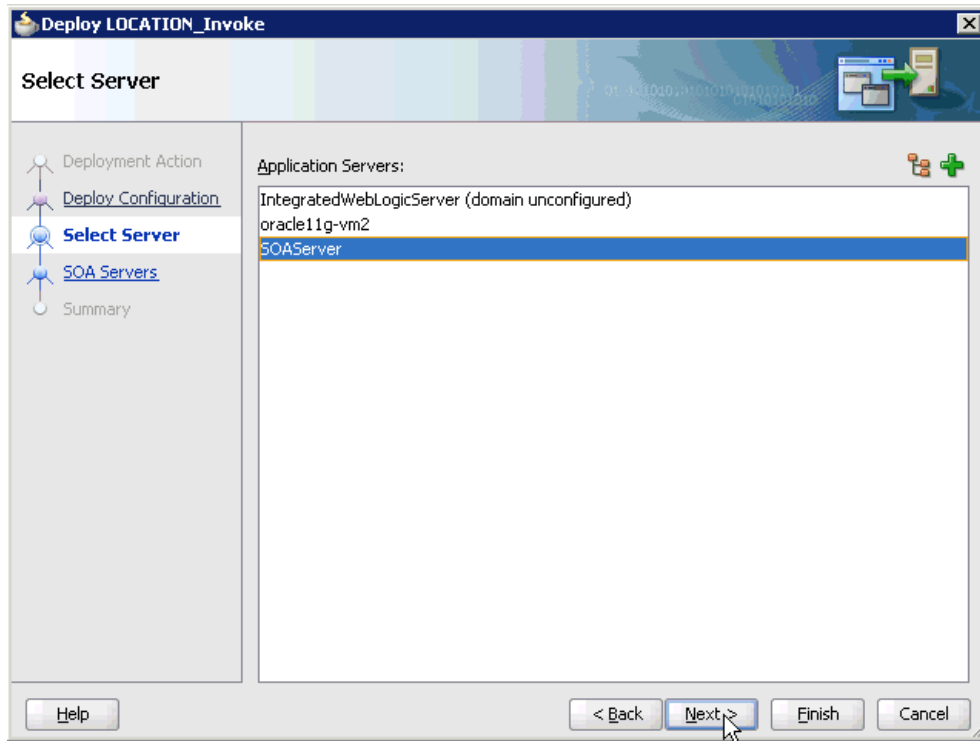
3. Click **Next**.

The Deploy Configuration page is displayed.

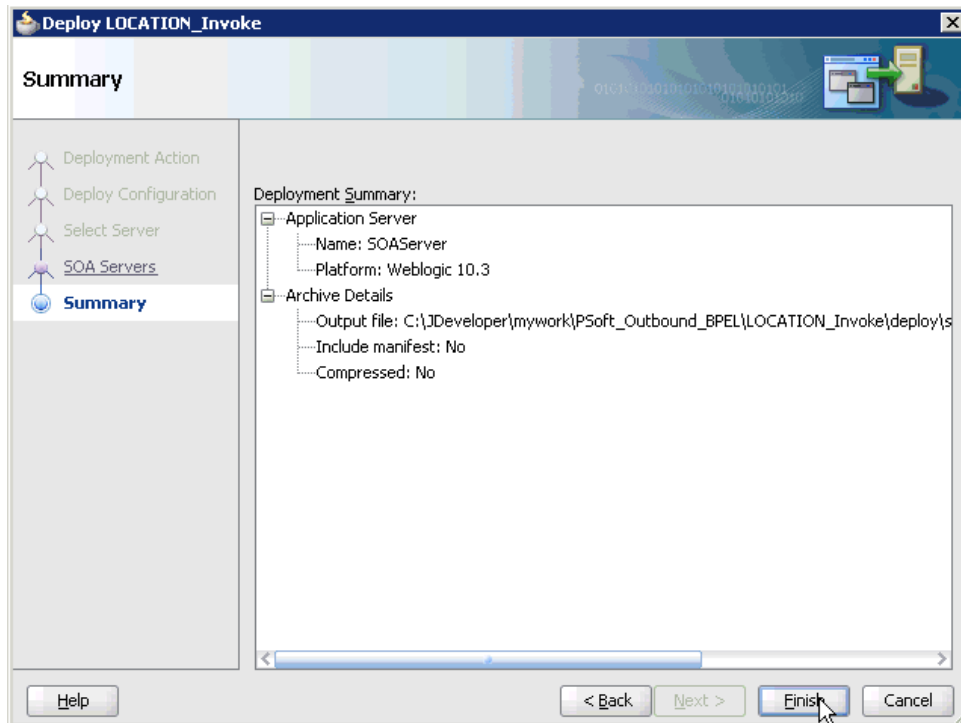


4. Leave the default values selected and click **Next**.

The Select Server page is displayed.

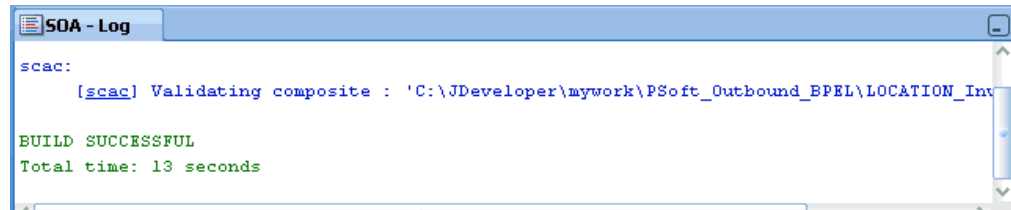


5. Select an available application server that was configured and click **Next**.
The SOA Servers dialog is displayed.
6. Select a target SOA server and click **Next**.
The Summary page is displayed.



- Review and verify all the available deployment information for your project and click **Finish**.

The process is deployed successfully.



```

SOA - Log
-----
scac:
  [_scac] Validating composite : 'C:\JDeveloper\mywork\PSoft_Outbound_BPEL\LOCATION_Inv
BUILD SUCCESSFUL
Total time: 13 seconds

```

If an Authorization Request dialog is displayed during the deployment process, provide the required user name and password and click **OK**.

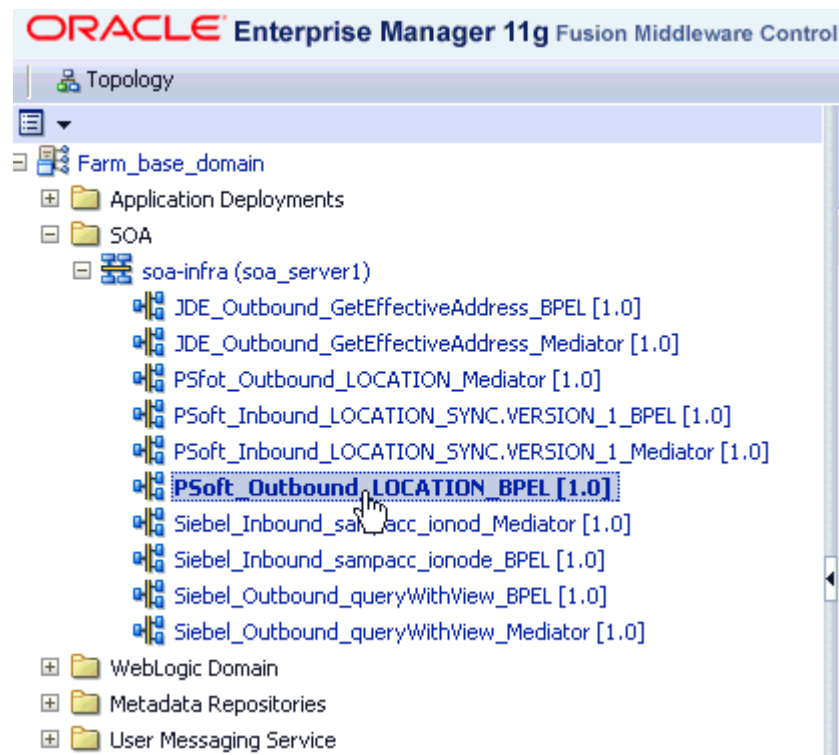
4.4.5 Invoking the Input XML Document in the Oracle Enterprise Manager Console

Perform the following steps to invoke the input XML document in the Oracle Enterprise Manager console.

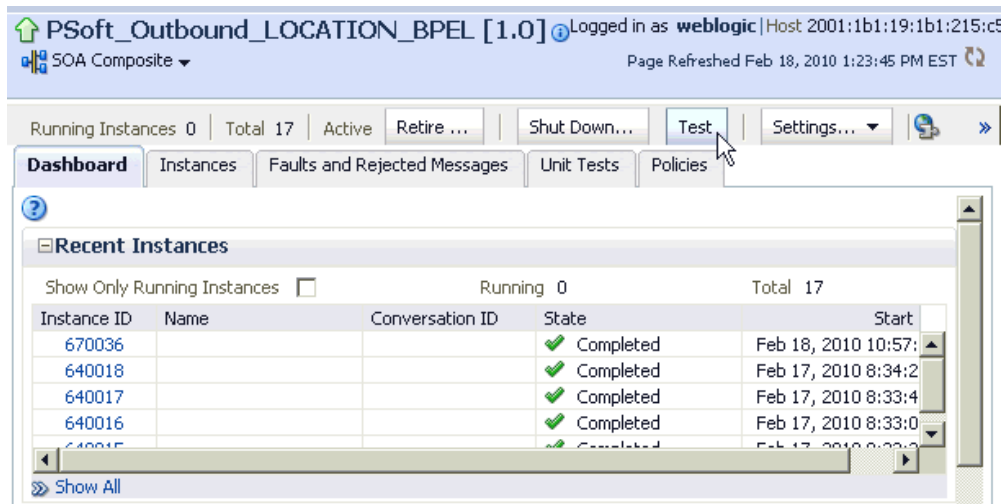
- Log in to the Oracle Enterprise Manager console by using the following URL:

<http://localhost:7001/em>

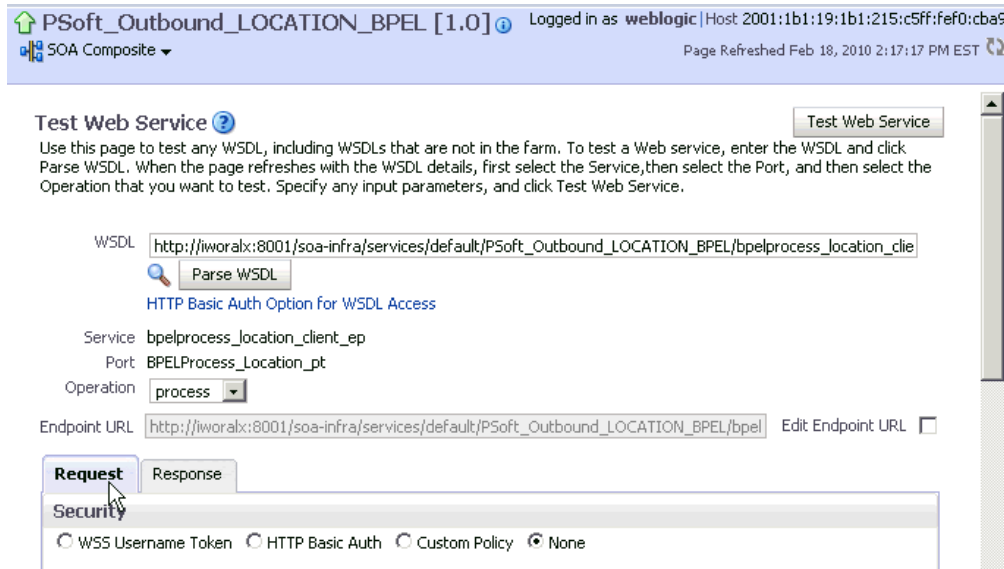
- Expand your domain in the left pane followed by the **SOA** folder.



- Select an available project (for example, PSoft_Outbound_LOCATION_BPEL).



4. Click **Test** in the right pane.
The Test Web Service page is displayed.



5. Click the **Request** tab.
6. Scroll down to the Input Arguments section.

The screenshot shows the 'Input Arguments' section of the Oracle Enterprise Manager console. It features a dropdown menu set to 'XML View' and a text area containing the following XML code:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ci="urn:iwaysoftware:adapter:peoplesoft:location:ci:request">
<ci:PS8>
  <ci:component perform="browse">LOCATION</ci:component>
  <ci:key name="SETID">SHARE</ci:key>
  <ci:key name="LOCATION">ALBERTA</ci:key>
</ci:PS8>
</soap:Body>
</soap:Envelope>
```

Below the XML view are two tabs: 'Request' (selected) and 'Response'. To the right of the console, a 'Test Web Service' button is highlighted with a red rectangular border.

7. Select **XML View** from the list in the upper-left corner.
8. Provide an appropriate input XML document in the Input Arguments area.

For example:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ci="urn:iwaysoftware:adapter:peoplesoft:location:ci:request">
<ci:PS8>
  <ci:component perform="browse">LOCATION</ci:component>
  <ci:key name="SETID">SHARE</ci:key>
  <ci:key name="LOCATION">ALBERTA</ci:key>
</ci:PS8>
</soap:Body>
</soap:Envelope>
```

9. Click **Test Web Service**.

The output response is received in the Response tab of the Oracle Enterprise Manager console.

Request **Response**

Test Status Passed

Response Time (ms) 251

XML View

Launch Message Flow Trace

```
<env:Envelope xmlns:env="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:wsa="http://www.w3.org/2005/08/addressing">
  <env:Header>
    <wsa:MessageID>urn:8AE949801CBF11DFBFDF3529C6BB7207 </wsa:MessageID>
    <wsa:ReplyTo>
      <wsa:Address>http://www.w3.org/2005/08/addressing/anonymous </wsa:Address>
    </wsa:ReplyTo>
  </env:Header>
  <env:Body>
    <PS8 xmlns:ci="urn:iwaysoftware:adapter:peoplesoft:location:ci:response"
xmlns="urn:iwaysoftware:adapter:peoplesoft:location:ci:response">
      <ci:result xmlns="">
        <ci:LOCATION_TBL row="1">
          <ci:SETID>SHARE </ci:SETID>
          <ci:LOCATION>ALBERTA </ci:LOCATION>
          <ci:EFFDT>04/25/2006 </ci:EFFDT>
        </ci:LOCATION_TBL>
      </ci:result>
    </PS8>
  </env:Body>
</env:Envelope>
```

4.4.6 Testing Outbound BPEL and Mediator Processes

When testing an outbound BPEL process or an outbound Mediator process from the Oracle Enterprise Manager console, do not use the XML envelopes that are generated by these consoles. Instead, remove them and use the XML payloads that are generated from the schemas, which conform to the WSDLs for namespace qualifications.

The Mediator data flows can be tested using the Oracle Enterprise Manager console. When creating a Mediator data flow and interactions, the Web services are created and registered with the Oracle WebLogic Server. For more information on creating a Mediator outbound process, see [Chapter 5, "Integration With Mediator Service Components in the Oracle SOA Suite"](#).

4.5 Designing an Inbound BPEL Process for Event Integration

This section demonstrates how Oracle Application Adapter for PeopleSoft integrates with PeopleSoft to receive event data. In this example, an PeopleSoft event occurs when a customer record is added to a PeopleSoft system.

The following tools are required to complete your adapter design-time configuration:

- Oracle Adapter Application Explorer (Application Explorer)
- Oracle JDeveloper BPEL Designer (JDeveloper) or Eclipse

Note: The examples in this chapter demonstrate the use of JDeveloper.

Before you design a BPEL process, you must generate the respective WSDL file using Application Explorer. See ["Generating WSDL for Event Integration"](#) on page 4-34 for more information.

4.5.1 Generating WSDL for Event Integration

You must create a separate channel for every event and select that channel when you generate WSDL for inbound interaction using Application Explorer.

Note: If two or more events share the same channel, event messages may not be delivered to the right BPEL process.

Creating a Channel

To create a channel:

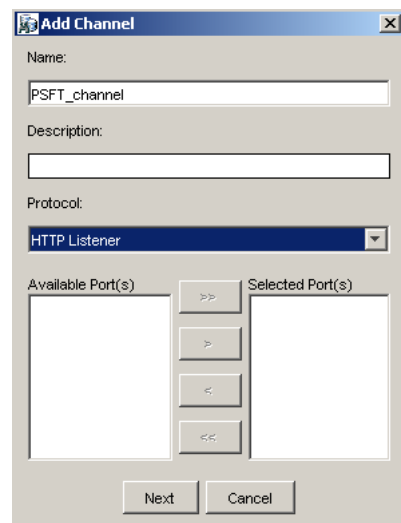
1. Click the **Events** node.
2. Expand the **PeopleSoft** node.

The ports and channels nodes appear in the left pane.



3. Right-click **Channels** and select **Add Channel**.

The Add Channel dialog is displayed.



Provide the following information:

- a. Enter a name for the channel, for example, **PSFT_Channel**.
 - b. Enter a brief description (optional).
 - c. From the Protocol list, select **HTTP Listener**.
4. Click **Next**.
 5. When the HTTP listener dialog is displayed, enter the system information as specified in the following table:

Parameter	Description
Listener port	Port on which to listen for PeopleSoft event data.

Parameter	Description
Https	For a secure HTTP connection, select the Https check box.
Synchronization Type	Choose from the following synchronization options: <ul style="list-style-type: none"> ■ REQUEST_RESPONSE ■ REQUEST_ACK <p>Important: The PeopleSoft channel does not work if the synchronization type is set to REQUEST.</p>

- Click **OK**.

The channel appears under the channels node in the left pane. An X over the icon indicates that the channel is currently disconnected.

Note: Do not start the channel, as it is managed by BPEL PM Server. If you start the channel for testing and debugging purposes, stop it before run-time.

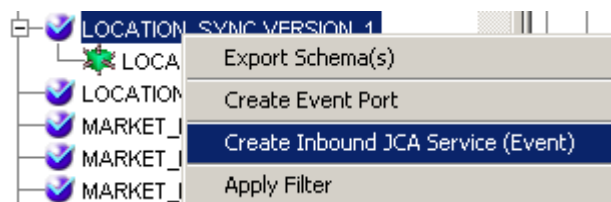
Generating WSDL for Event Notification

After you create a channel and verify that it is not started, you must generate WSDL for the event using Application Explorer.

- Start Application Explorer.
- Expand the **Adapters** node.
A list of all available adapters is displayed.
- Expand **PeopleSoft**.
- Click a target name under the **PeopleSoft** node, for example, **PSFTtarget**.
The Logon pane on the right displays the saved parameters.
- Verify your connection parameters and provide the required password.
- Right-click the target name and select **Connect**.
The x icon disappears, indicating that the target is connected.

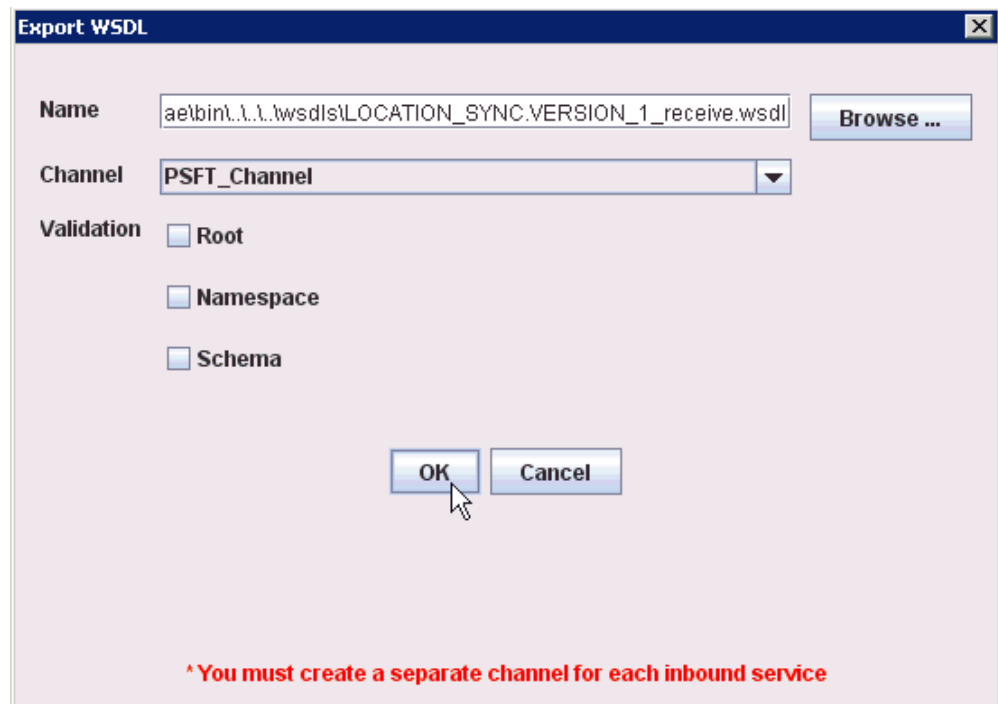


- Expand **Messages** and select **LOCATION_SYNC.VERSION_1**.
- Right-click **LOCATION_SYNC.VERSION_1**.



- Select **Create Inbound JCA Service (Event)**.

The Export WSDL dialog is displayed.



Perform the following steps:

- a. In the **Name** field, specify the name of the WSDL file.

The **.wsdl** file extension is added automatically. By default, the names of WSDL files generated for request-response services end with **_receive**.

- b. From the **Channel** list, select the channel you created for this inbound service.

Important: You must create a separate channel for every event. Verify that the channel is stopped before run-time.

- c. Three check boxes for Root, Namespace, and Schema validation are also available. Selection of multiple validation options is allowed.

- Root validation is used to validate the root element in the inbound XML document.

- Namespace validation is used to validate the namespace in the inbound XML document.

- Schema validation is used to validate the inbound XML document with the schema in the WSDL document.

During run time, validation is processed based on the validation options that are selected. If more than one validation option is selected, during run time if the first validation option fails, the remaining validation options are not processed. Root and namespace validations are considered modest levels of validation. Schema validation is a stricter validation level. It is recommended to use root and namespace validation options together, unless the root element and namespace are different between the Messages in the PeopleSoft environment.

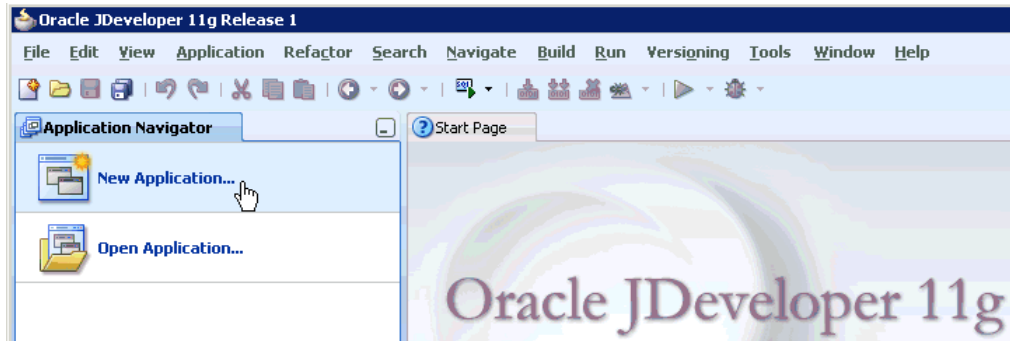
10. Click **OK**.

You can now create a new SOA application, which is the first step that is required to define a BPEL inbound process in JDeveloper.

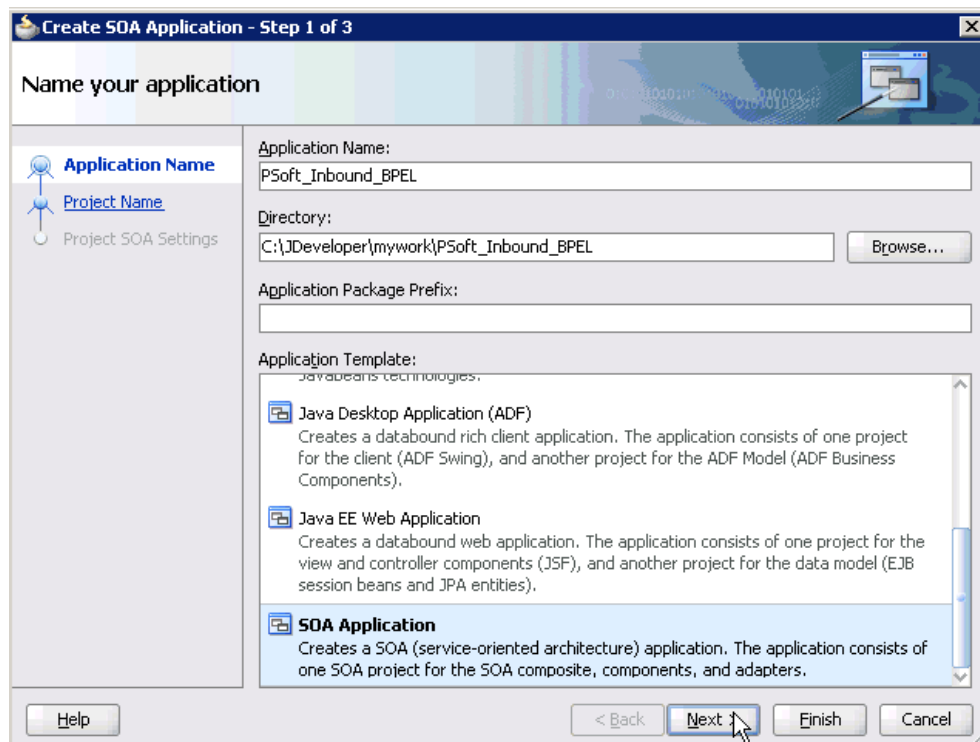
4.5.2 Creating a New SOA Application for the Inbound BPEL Process

Perform the following steps to create a new SOA application for the inbound BPEL process:

1. Open Oracle JDeveloper on your system.
2. In the Application Navigator tab, click **New Application**.

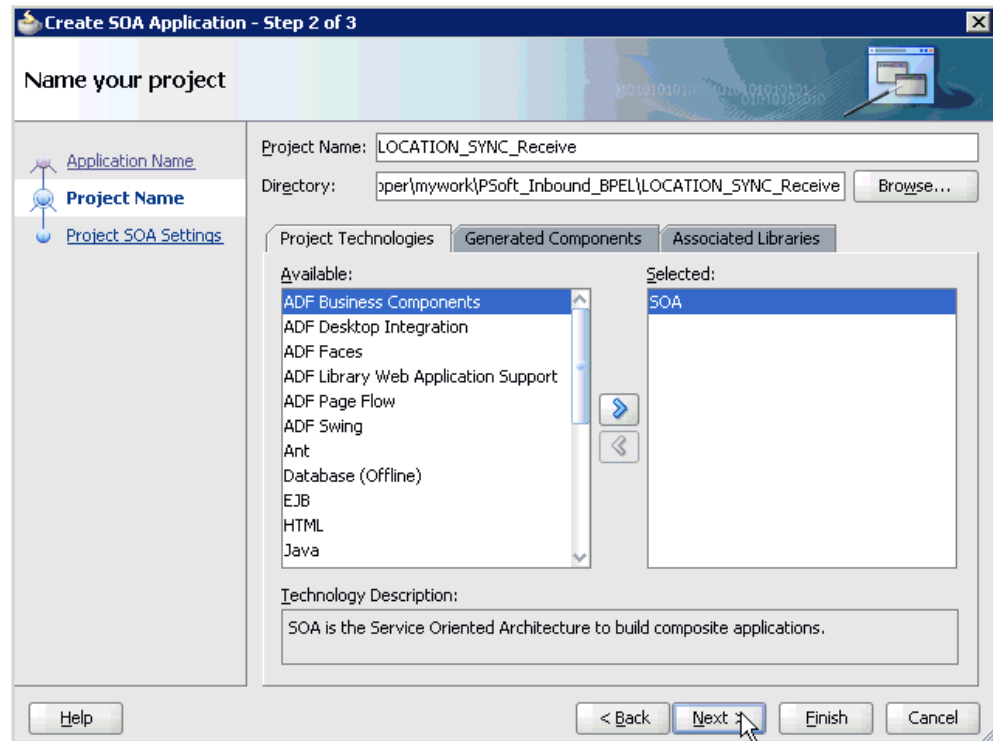


The Create SOA Application wizard is displayed.

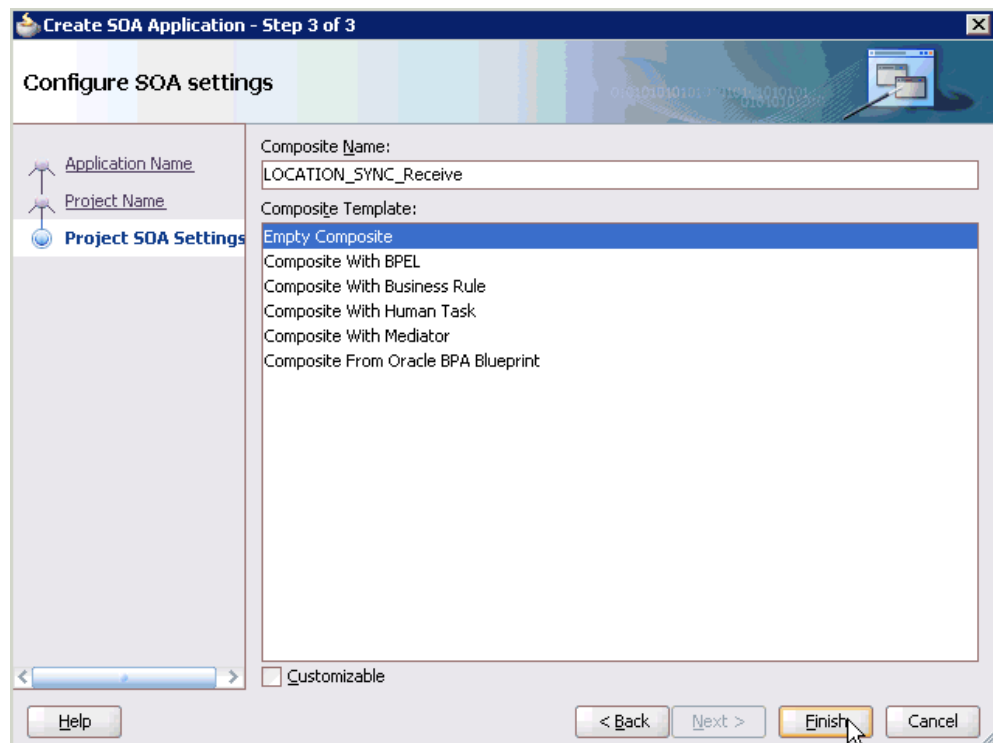


3. From the Application Template list, click **SOA Application**.
4. Enter name for the new SOA application (for example, PSoft_Inbound_BPEL) and click **Next**.

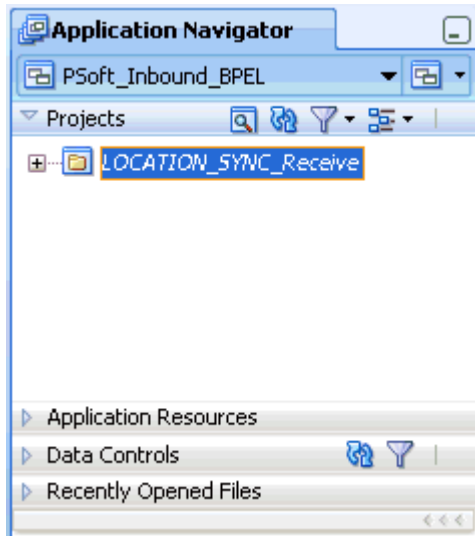
The Name your project page is displayed.



5. Enter a project name (for example, LOCATION_SYNC_Receive) and click **Next**. The Configure SOA settings page is displayed.



6. From the Composite Template list, select **Empty Composite** and click **Finish**.



The new SOA application (PSoft_Inbound_BPEL) and associated project (LOCATION_SYNC_Receive) are added to the Application Navigator tab in the left pane.

4.5.3 Defining a BPEL Inbound Process

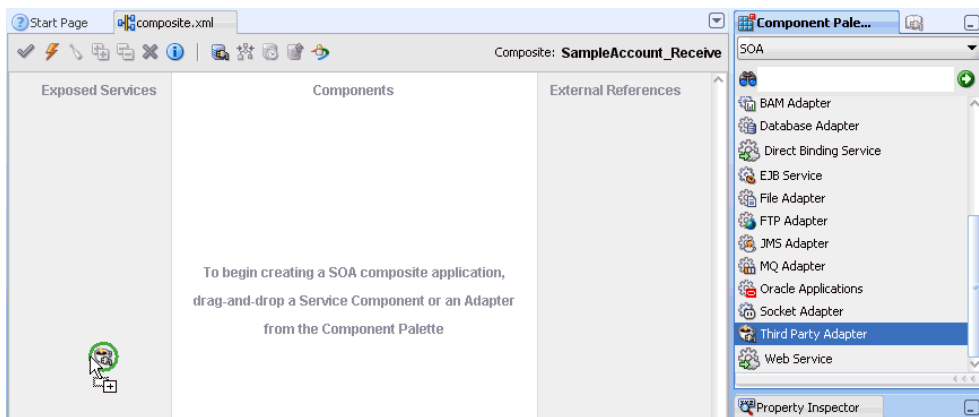
This section describes how to define a BPEL inbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Inbound BPEL Process Component

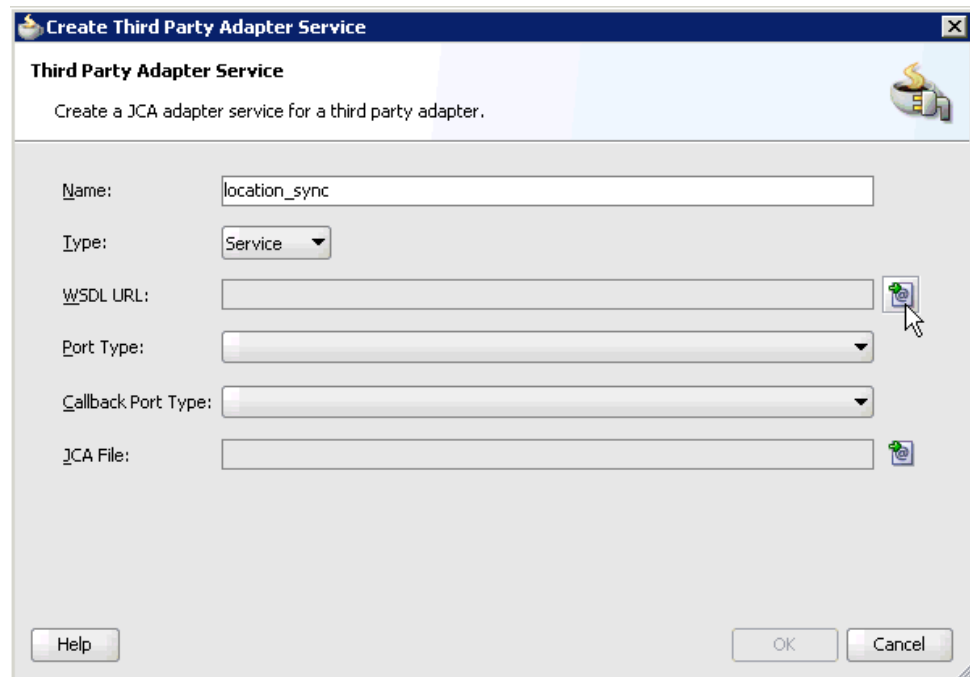
Creating a Third Party Adapter Service Component

Perform the following steps to create a third party adapter service component:

1. Drag and drop the **Third Party Adapter** component from the Component Palette tab (Service Adapters section) to the Exposed Services pane.

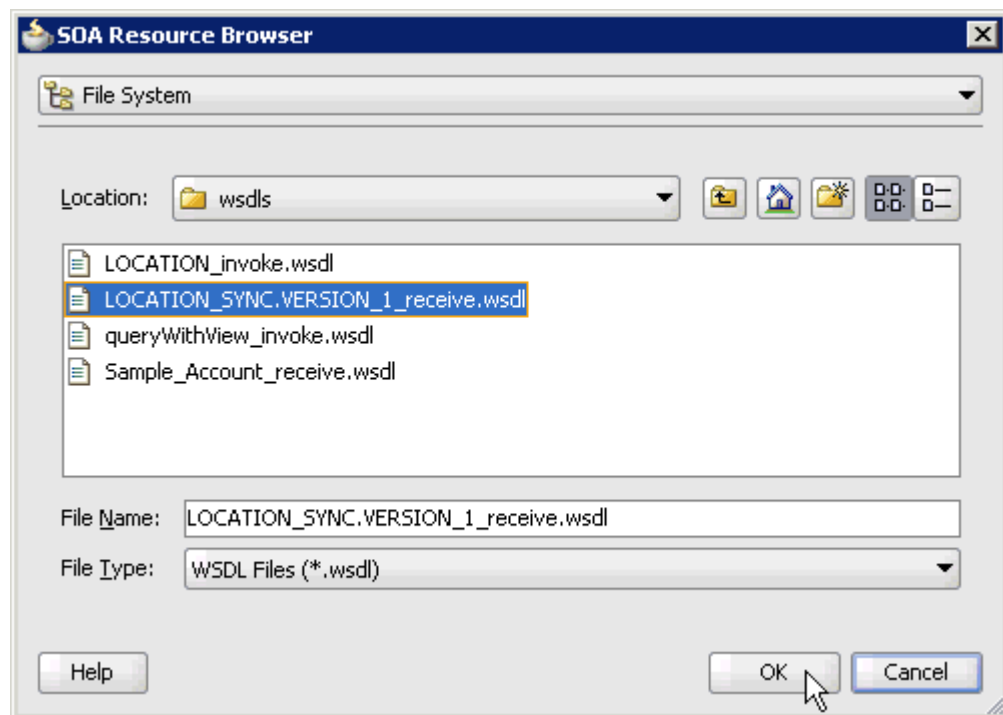


The Create Third Party Adapter Service dialog is displayed.



2. Enter a name for the third party adapter service.
3. Ensure that **Service** is selected from the Type list (default).
4. Click the **Find existing WSDLs** icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed.

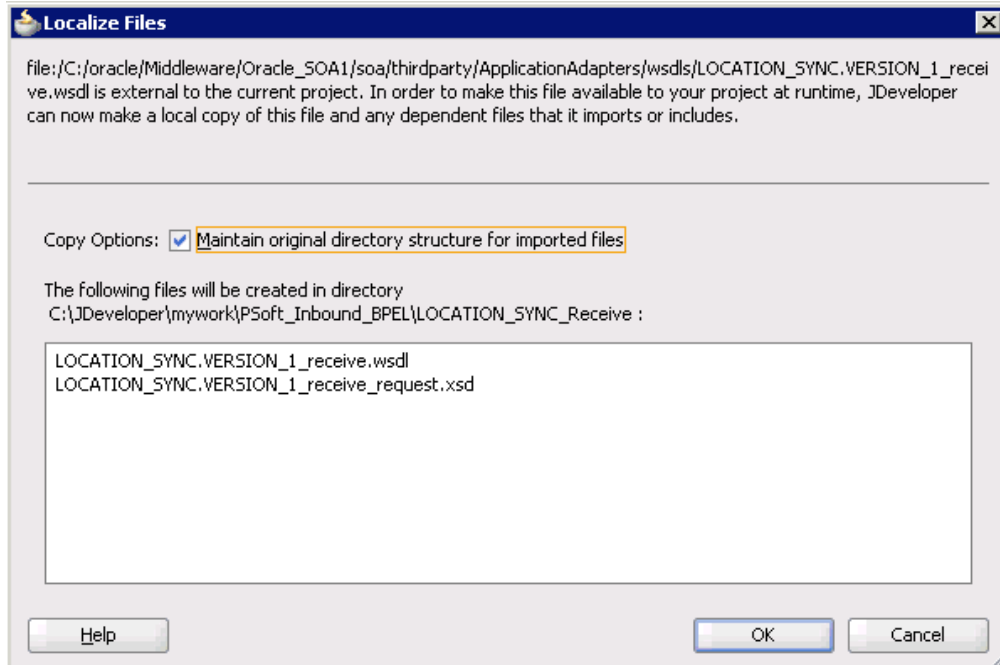


5. Browse and select an inbound WSDL file (for example, LOCATION_SYNC.VERSION_1_receive.wsdl) from the following directory:

C:\oracle\Middleware\home_GA\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\wsdls

6. Click **OK**.

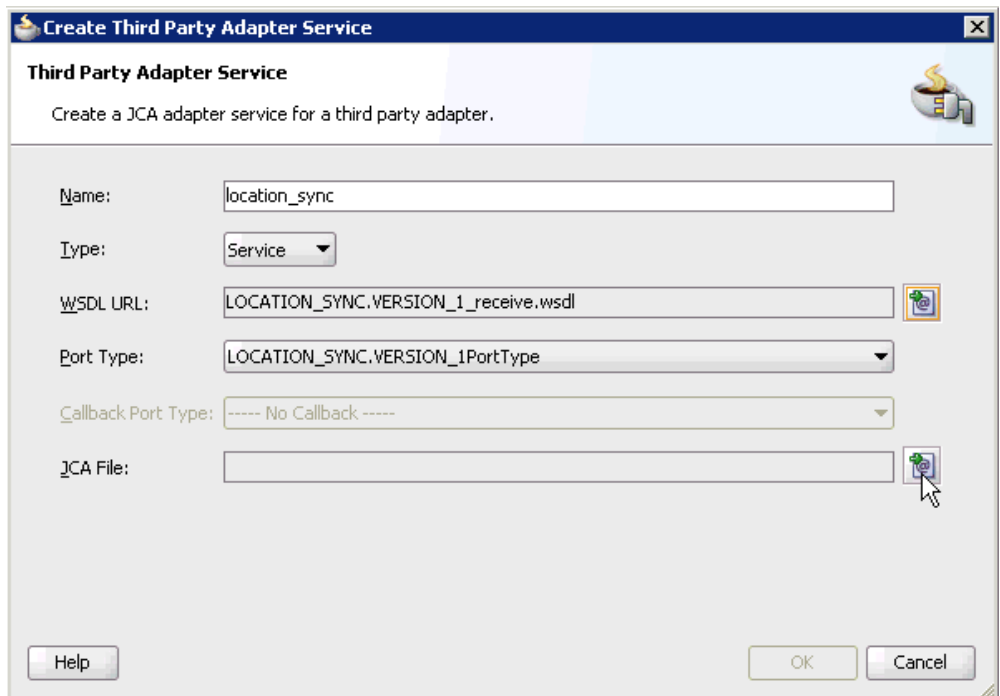
The Localize Files dialog is displayed.



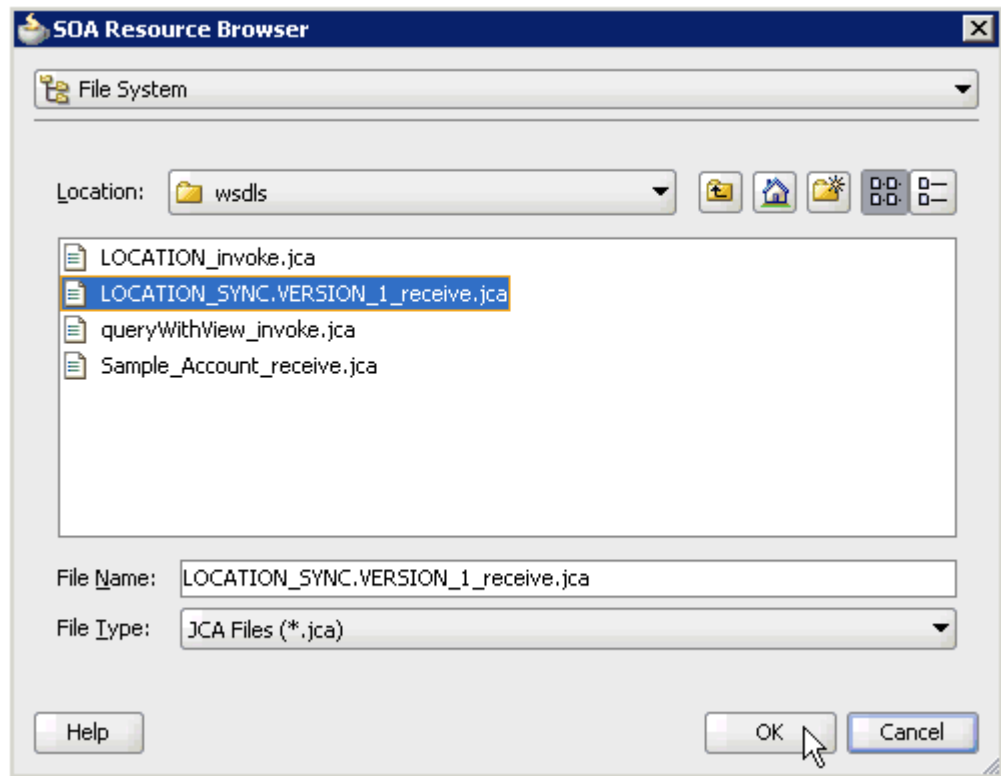
7. Click **OK**.

The inbound WSDL file and associated receive/request XML schema file (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog.



8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed.

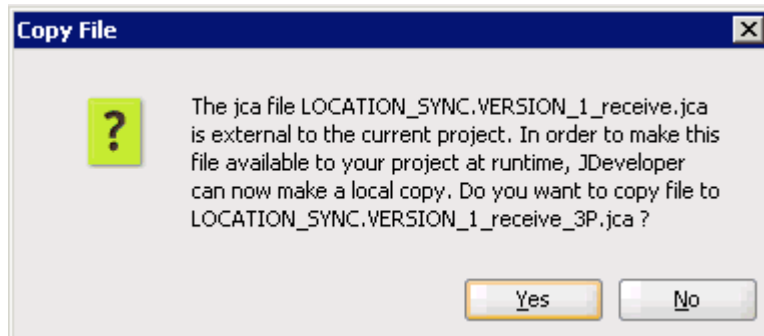


9. Browse and select the JCA properties file (for example, LOCATION_SYNC.VERSION_1_receive.jca) from the following directory:

```
C:\oracle\Middleware\home_GA\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\wsdls
```

10. Click **OK**.

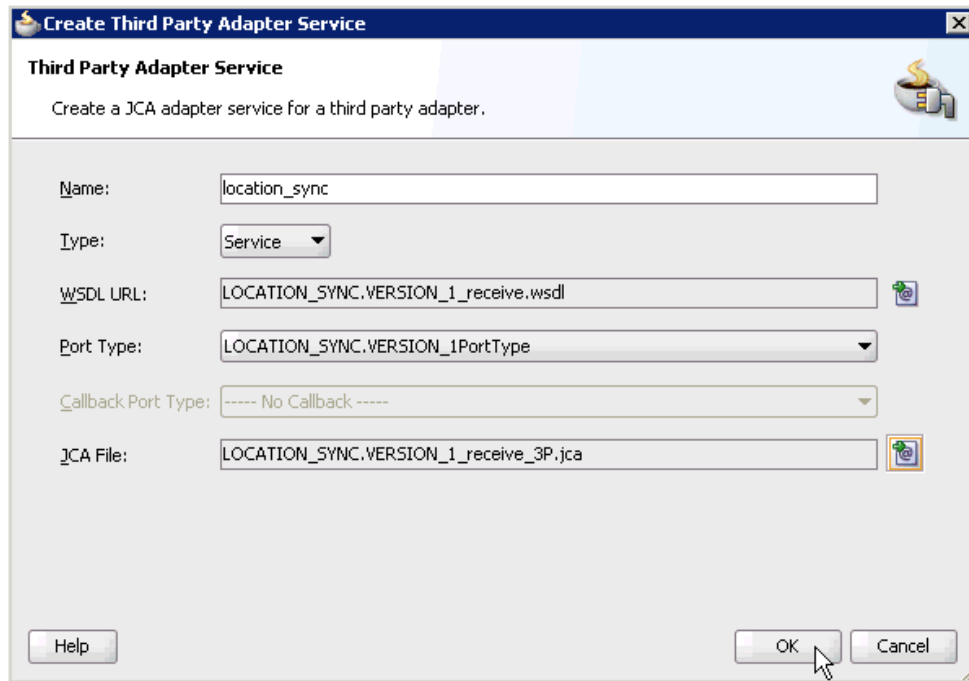
The following message is displayed.



11. Click **Yes**.

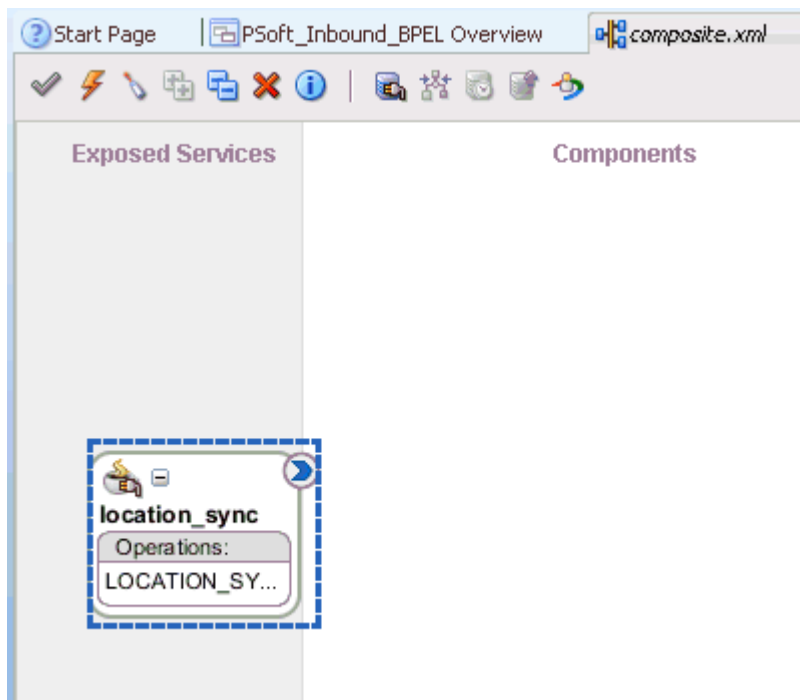
A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog.



12. Click **OK**.

The third party adapter service component (location_sync) is created in the Exposed Services pane, as shown in the following image.

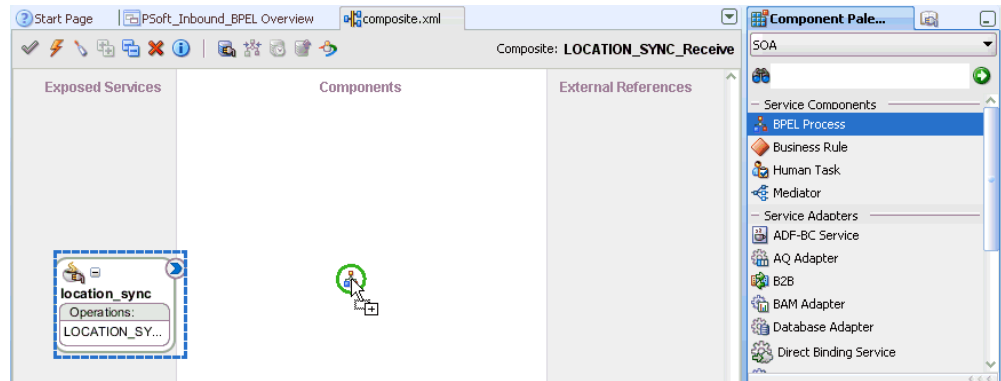


You are now ready to configure an inbound BPEL process component.

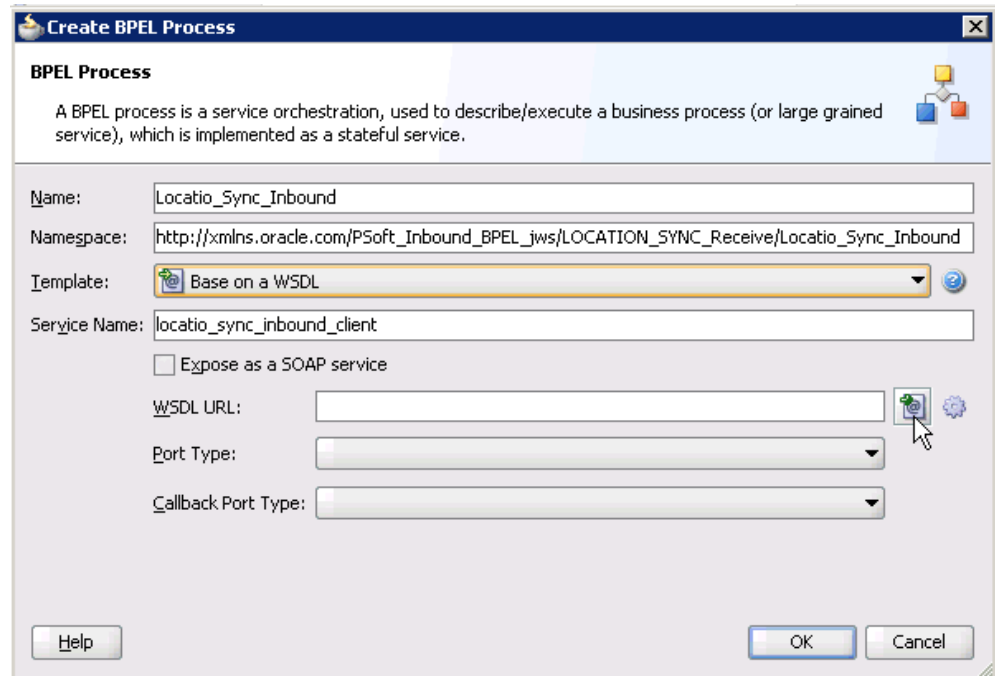
Creating an Inbound BPEL Process Component

Perform the following steps to create an inbound BPEL process component:

1. Drag and drop the **BPEL Process** component from the Component Palette tab (Service Components section) to the Components pane.

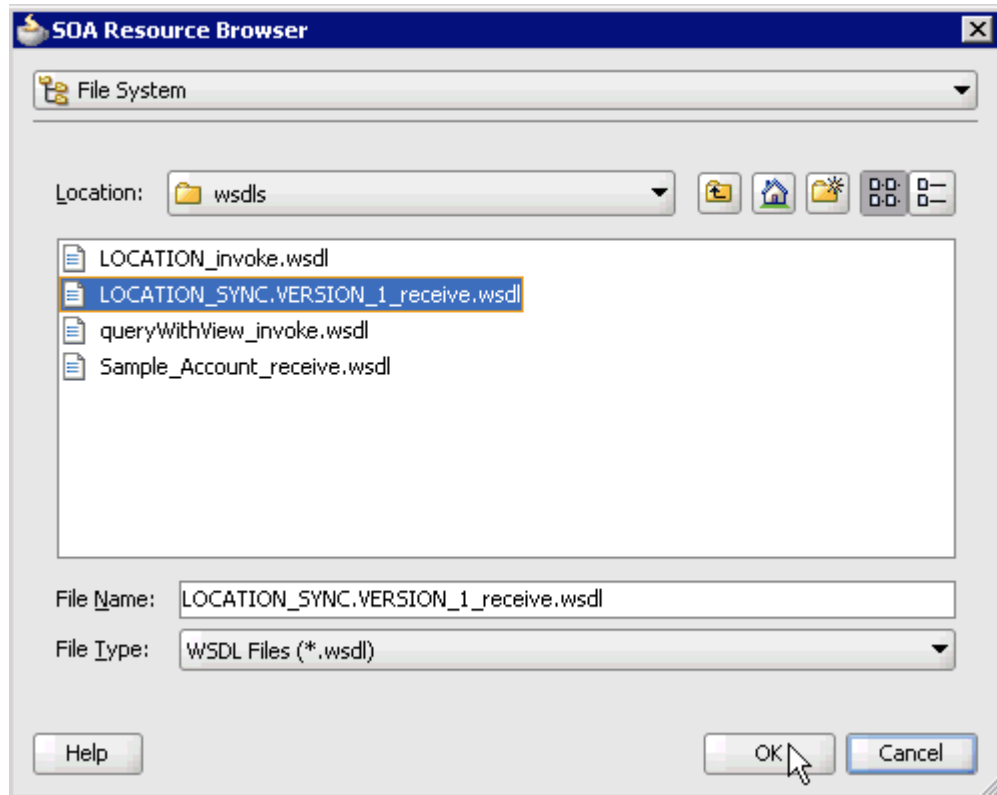


The Create BPEL Process dialog is displayed.



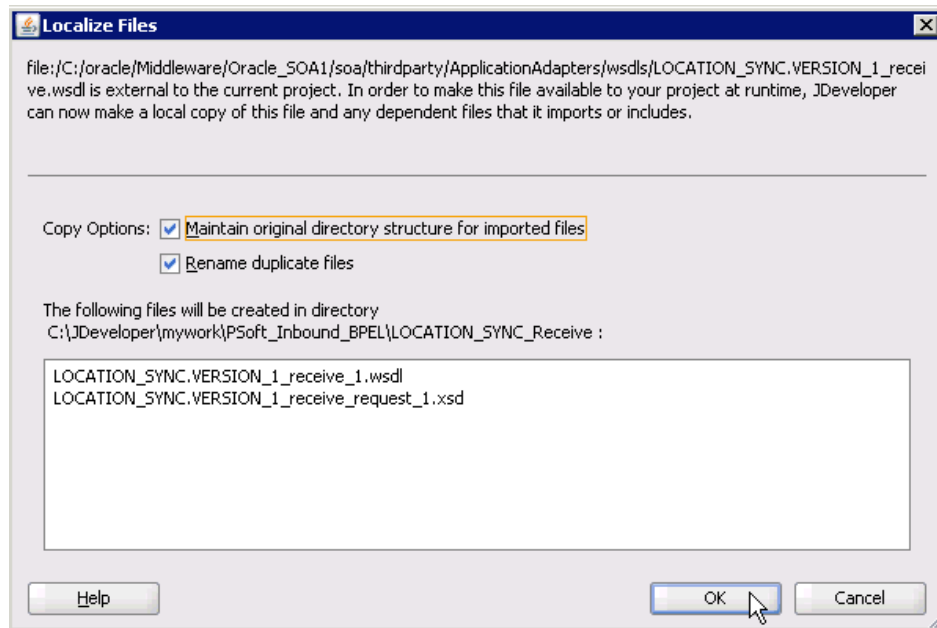
2. In the Name field, enter a name to identify the new inbound BPEL process component (for example, Location_Sync_Inbound).
3. From the Template list, select **Base on a WSDL**.
4. Uncheck the **Expose as SOAP service** check box.
5. Click the **Find existing WSDLs** icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed.



6. Browse and select an inbound WSDL file (for example, LOCATION_SYNC.VERSION_1_receive.wsdl) from the project folder.
7. Click **OK**.

The Localize Files dialog is displayed.



8. Click **OK**.

You are returned to the Create BPEL Process dialog.

Create BPEL Process

A BPEL process is a service orchestration, used to describe/execute a business process (or large grained service), which is implemented as a stateful service.

Name:

Namespace:

Template:

Service Name:

Expose as a SOAP service

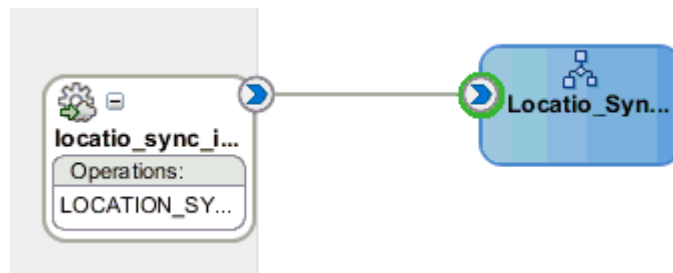
WSDL URL:

Port Type:

Callback Port Type:

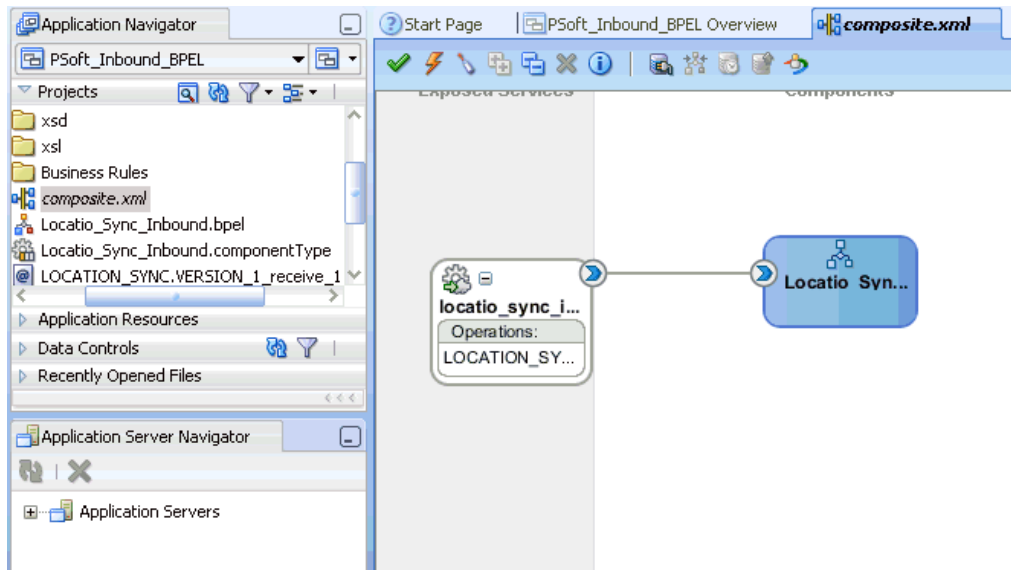
Buttons: Help, OK, Cancel

9. Click **OK**.

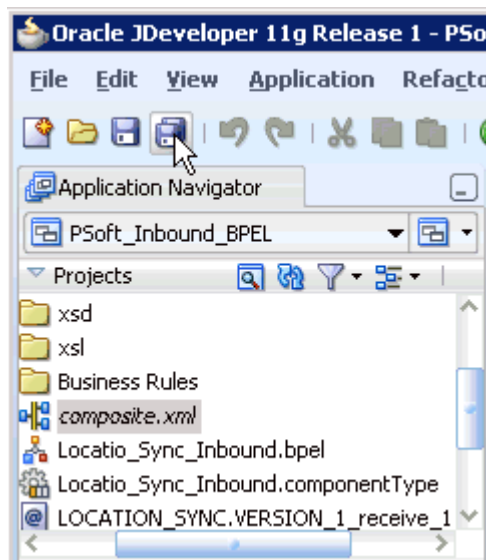


10. Create a connection between the third party adapter service component (location_sync) and the inbound BPEL process component (Location_Sync_Inbound).

11. Double-click **composite.xml** in the left pane.



12. Click the **Save All** icon in the menu bar to save the new inbound BPEL process component that was configured.

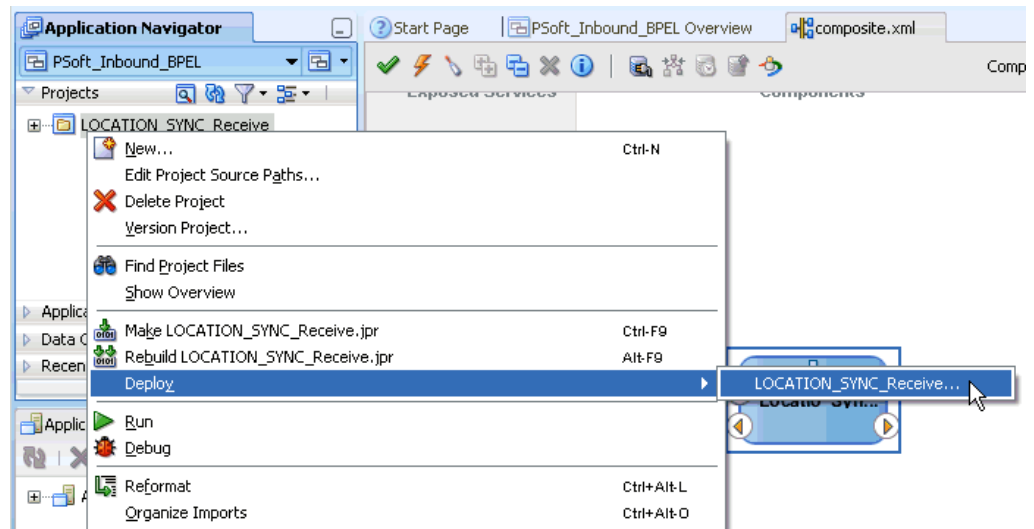


You are now ready to deploy the BPEL inbound process.

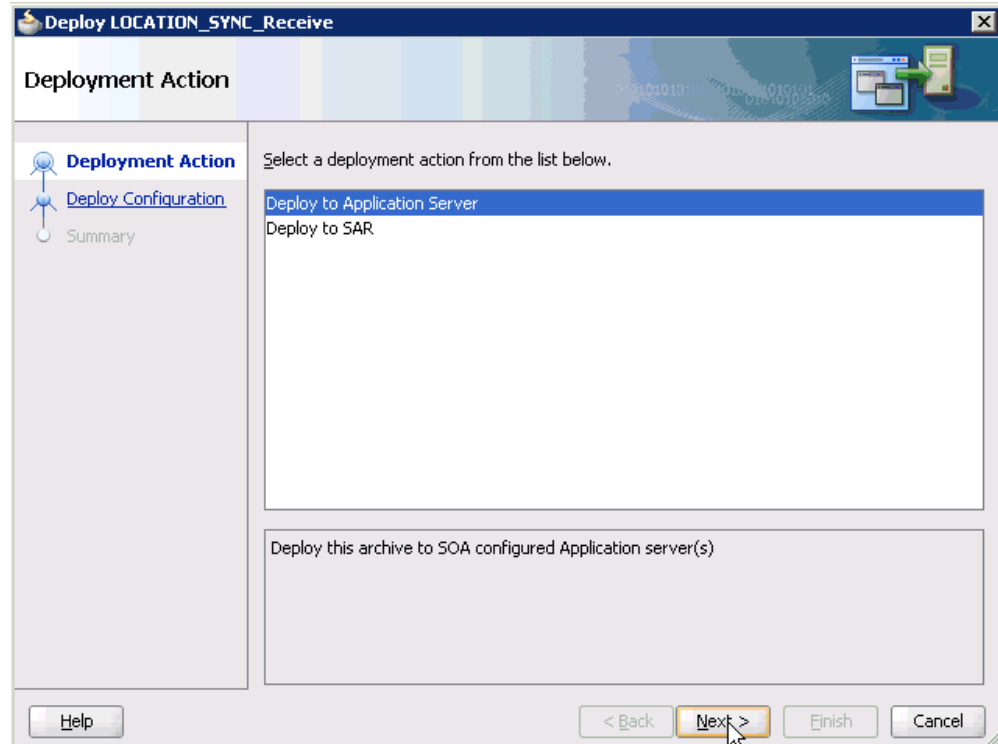
4.5.4 Deploying the BPEL Inbound Process

Perform the following steps to deploy the BPEL inbound process.

1. Right-click the project name in the left pane (for example, **LOCATION_SYNC_Receive**), select **Deploy**, and then click **LOCATION_SYNC_Receive**.

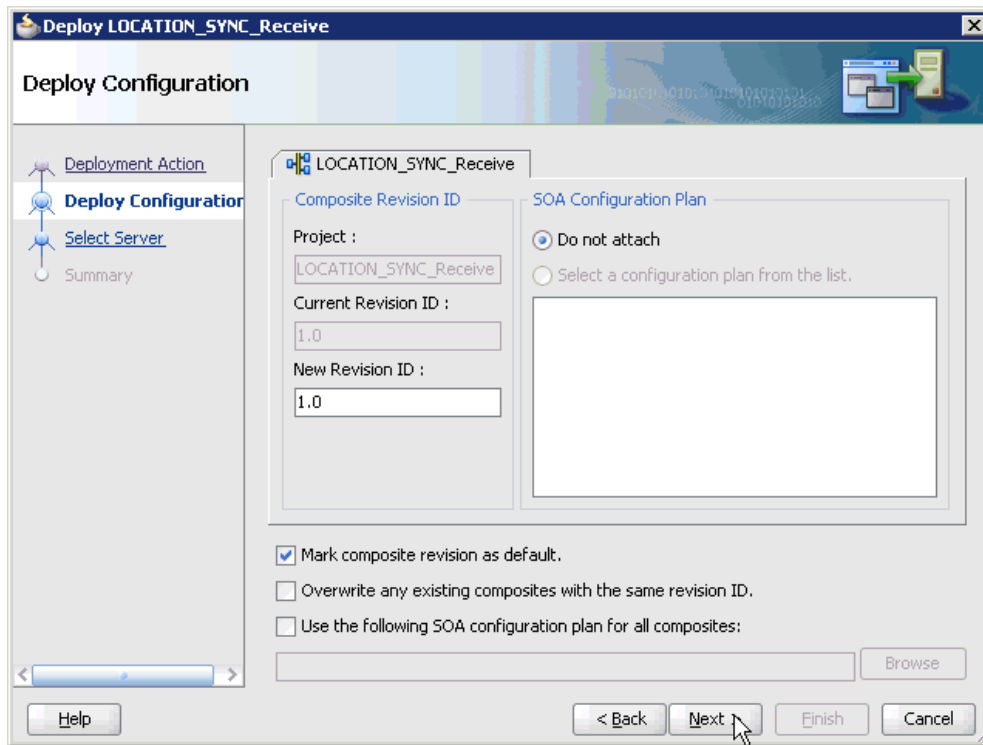


The Deployment Action page is displayed.



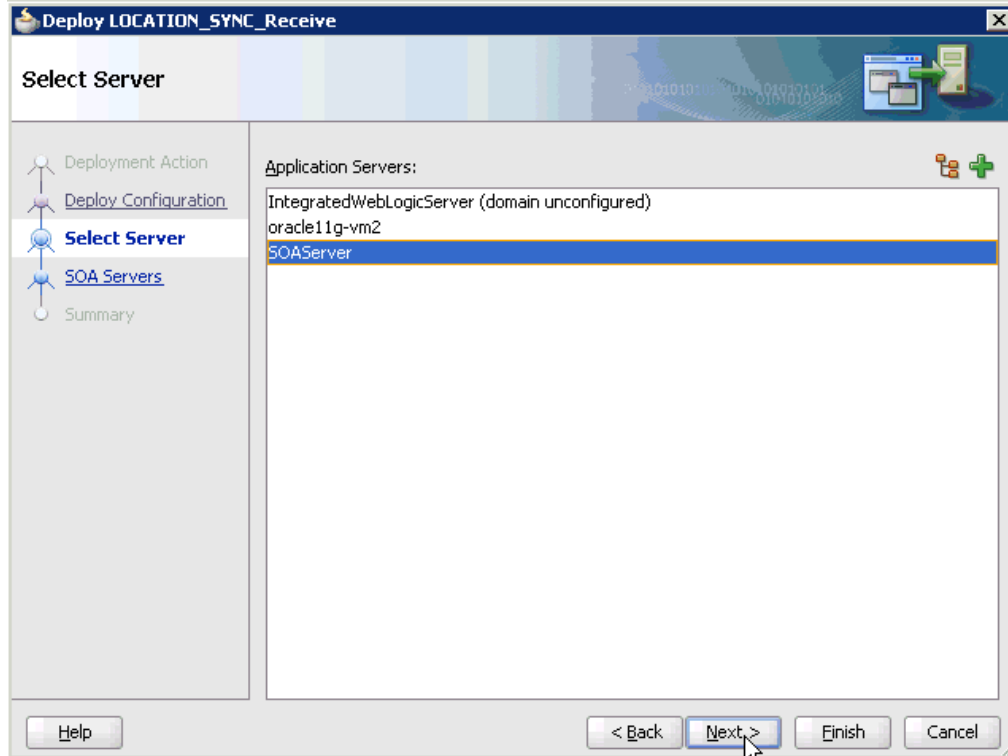
2. Ensure that **Deploy to Application Server** is selected.
3. Click **Next**.

The Deploy Configuration page is displayed.



4. Leave the default values selected and click **Next**.

The Select Server page is displayed.

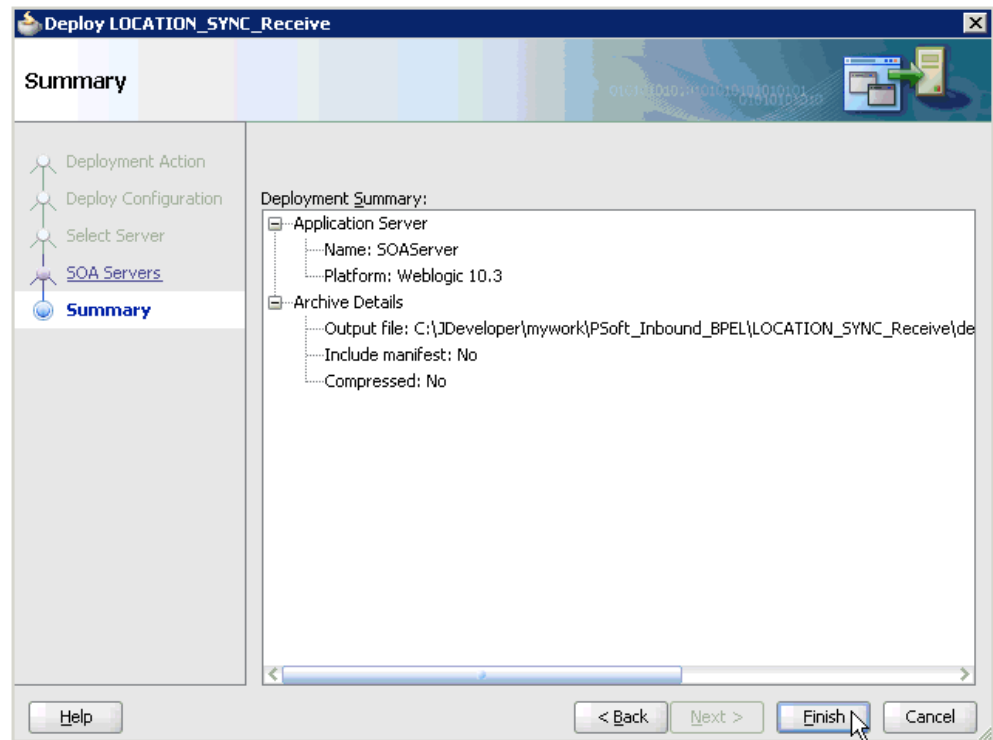


5. Select an available application server that was configured and click **Next**.

The SOA Servers dialog is displayed.

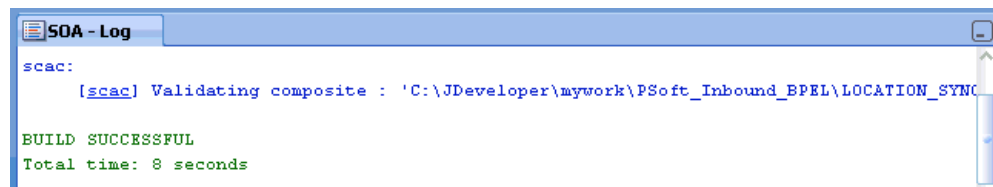
6. Select a target SOA server and click **Next**.

The Summary page is displayed.



7. Review and verify all the available deployment information for your project and click **Finish**.

The process is deployed successfully.



If an Authorization Request dialog is displayed during the deployment process, provide the required user name and password and click **OK**.

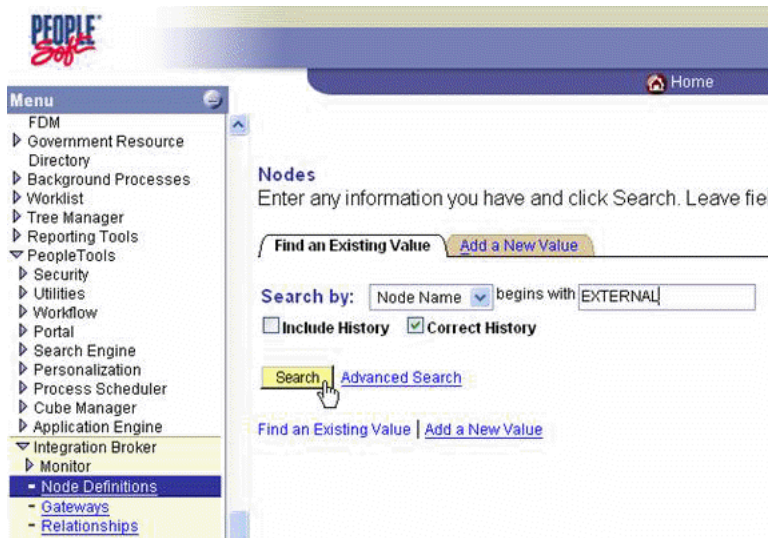
Once event messages are triggered through PeopleSoft, successful instances are received in the Oracle Enterprise Manager console.

4.5.5 Triggering an Event in PeopleSoft

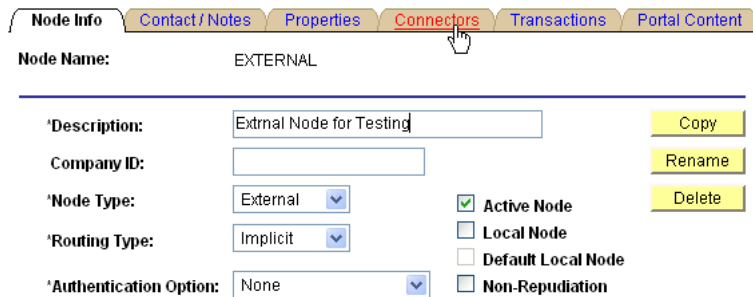
Events are generated by activity in an application system. For example, PeopleSoft may generate an event as customer information is updated in the system. The following topics describe how to trigger an event in PeopleSoft and verify the event using Oracle Application Adapter for PeopleSoft.

To trigger an event in PeopleSoft:

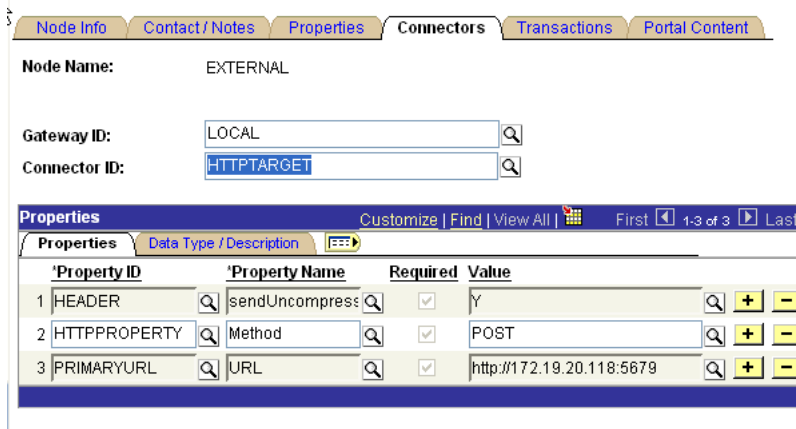
1. Log in to PeopleSoft, select **PeopleTools, Integration Broker**, and then **Node Definitions**.



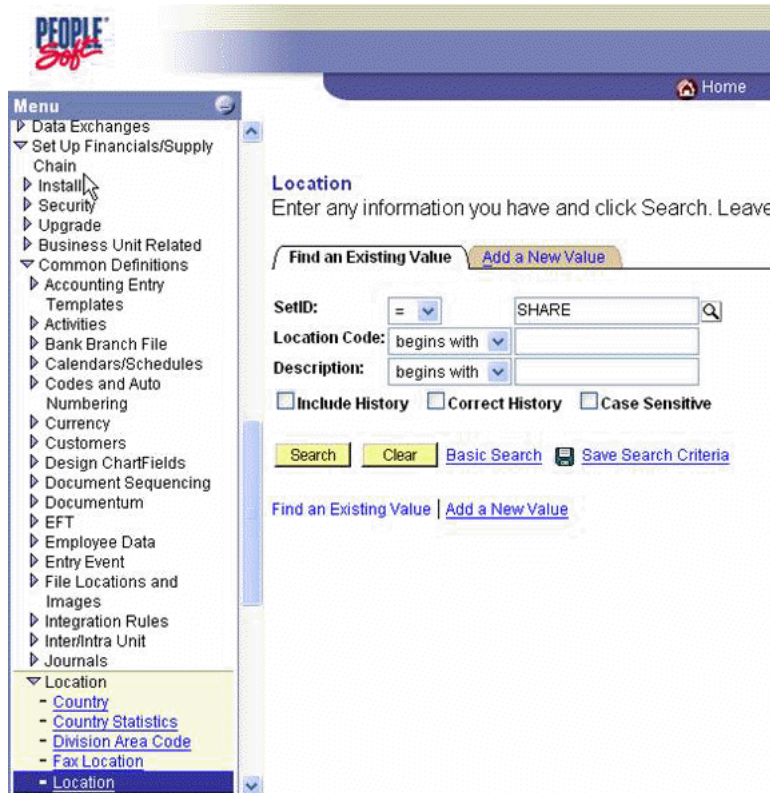
2. Enter the node name you are using, for example, EXTERNAL; then click **Search**.



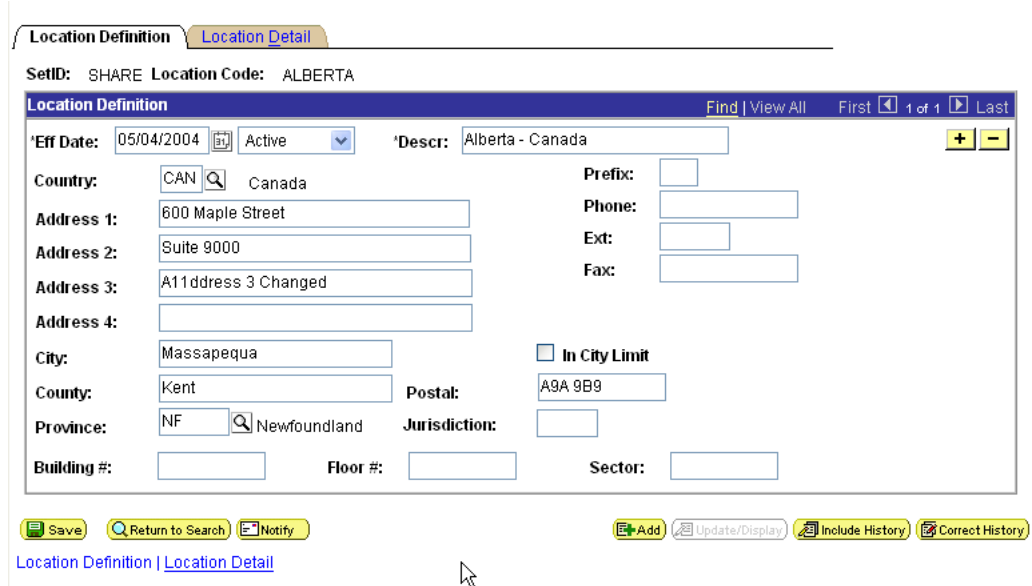
3. Click the **Connectors** tab.



4. Enter **HTTPTARGET** in the **Connector ID** field; then enter the **listener URL** and its port in the **PRIMARYURL** field.
5. To save the configuration, click **Save**.



- From **Menu** in the left pane, select **Set Up Financials/Supply Chain, Common Definitions, Location,** and then **Location**.



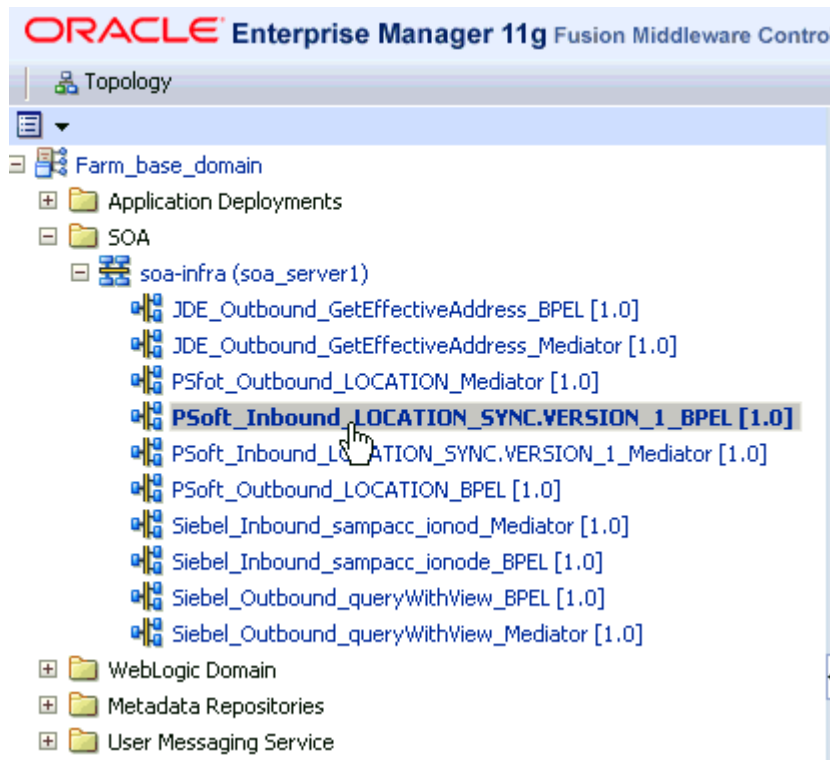
- To find the location record you want to update, click **Search**; then make the changes and click **Save**.

When the change is saved, it triggers an event and sends the event to the listener.

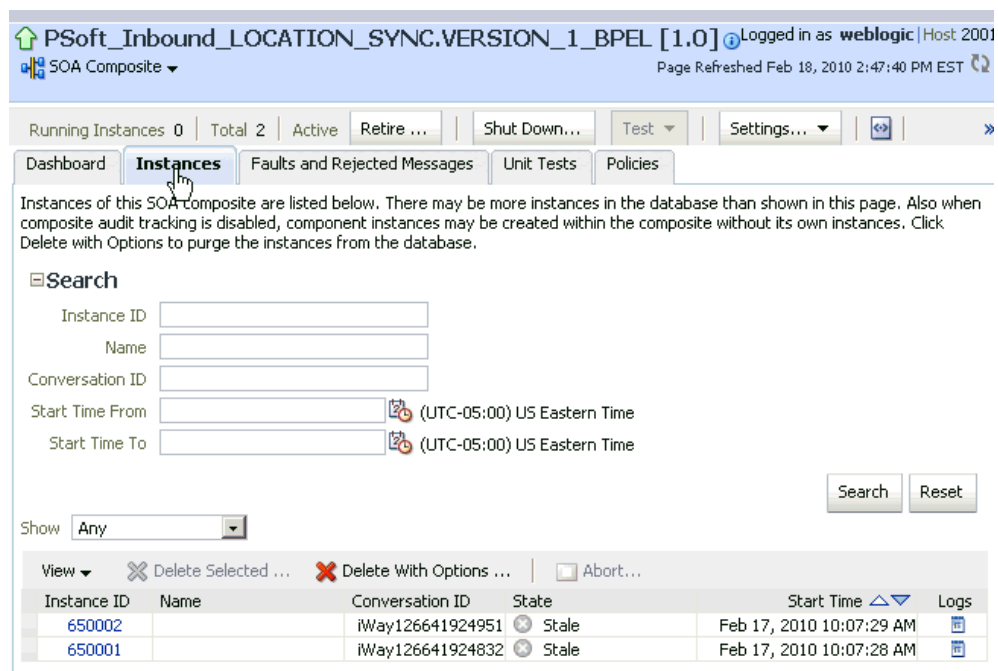
Verifying the Results

To verify your results:

1. Log in to the Oracle Enterprise Manager console by using the following URL:
<http://localhost:7001/em>
2. Expand your domain in the left pane followed by the **SOA** folder.



3. Select an available project (for example, PSoft_Inbound_LOCATION_SYNC.VERSION_1_BPEL).



4. Click the **Instances** tab in the right pane.

Recently received run-time events are displayed in the Instances tab.

Instance ID	Name	Conversation ID	State	Start Time
650002		iWay126641924951	Stale	Feb 17, 2010 10:07:29
650		iWay126641924832	Stale	Feb 17, 2010 10:07:28

5. Select a PeopleSoft instance ID.

The Flow Trace page is displayed.

Flow Trace
 This page shows the flow of the message through various composite and component instances. ECID 00001RSQN59Fw000jzwno1Bv06800009
 Started Feb 17, 2010 10:07:29 AM

Faults (0)
 Select a fault to locate it in the trace view.

Error Message	Recovery	Fault Time	Fault Location	Composite Instance
No faults found				

Sensors (0)

Trace
 Click a component instance to see its detailed audit trail.
 Show Instance IDs

Instance	Type	State	Time	Composite Instance
Service1_location_sync	Service	Stale	Feb 17, 2010 10:07:28 AM	PSoft_Inbound_LOCATION_SYNC
BPELProcess1_slocat	BPEL Component	Stale	Feb 17, 2010 10:07:30 AM	PSoft_Inbound_LOCATION_SYNC
Service1_location_sync	Service	Stale	Feb 17, 2010 10:07:29 AM	PSoft_Inbound_LOCATION_SYNC

6. Select a component instance to view its detailed audit trail.

The Instance page for the selected component is displayed.

Instance of BPELProcess1_slocat
 This page shows BPEL process instance details.

Audit Trail | Flow | Sensor Values | Faults

Expand a payload node to view the details.

```

<process>
  <sequence>
    receiveInput
      Feb 17, 2010 10:07:30 AM Received "inputVariable" call from partner "bpeprocess1_slocat_client"
        <payload>
          <inputVariable>
            <part name="event_LOCATION_SYNC.VERSION_1">
              <m:LOCATION_SYNC>
                <m:FieldTypes>
                  <m:LOCATION_TBL class="R">
                    <m:SETID type="CHAR"/>
                    <m:LOCATION type="CHAR"/>
                    <m:EFFDT type="DATE"/>
                    <m:EFF_STATUS type="CHAR"/>
                    <m:DESCR type="CHAR"/>
                    <m:DESCR_AC type="CHAR"/>
                    <m:DESCRSHORT type="CHAR"/>
                    <m:BUILDING type="CHAR"/>
                    <m:FLOOR type="CHAR"/>
                    <m:SECTOR type="CHAR"/>
                    <m:JURISDICTION type="CHAR"/>
                    <m:ATTN_TO type="CHAR"/>
                    <m:COUNTRY type="CHAR"/>
                    <m:ADDRESS1 type="CHAR"/>
                    <m:ADDRESS2 type="CHAR"/>
                    <m:ADDRESS3 type="CHAR"/>
                    <m:ADDRESS4 type="CHAR"/>
                    <m:CITY type="CHAR"/>
                    <m:NUM1 type="CHAR"/>
                    <m:NUM2 type="CHAR"/>
                </m:FieldTypes>
              </m:LOCATION_SYNC>
            </part>
          </inputVariable>
        </payload>
      </receiveInput>
    </sequence>
  </process>
    
```

7. Click the Audit Trail tab to view the event message.

The message received from the PeopleSoft system is displayed in the Audit Trail tab.

Integration With Mediator Service Components in the Oracle SOA Suite

This chapter contains the following examples:

- [Configuring a New Application Server Connection](#)
- [Configuring a Mediator Outbound Process](#)
- [Configuring a Mediator Inbound Process](#)

The scenarios shown in this chapter require the following prerequisites.

Prerequisites

The following are installation and configuration requirements:

- Oracle Application Adapter for PeopleSoft must be installed on Oracle WebLogic Server.
- PeopleSoft must be configured for inbound and outbound processing..

See Also: *Oracle Fusion Middleware Application Adapters Installation Guide for Oracle WebLogic Server*

The examples in this chapter present the configuration steps necessary for demonstrating service and event integration with PeopleSoft. Prior to using this material, you must be familiar with the following:

- How to configure Oracle Application Adapter for PeopleSoft for services and events. For more information, see [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#).
- How to configure a new Application Server and Integration Server connection in Oracle JDeveloper. For more information, see [Chapter 4, "Integration With BPEL Service Components in the Oracle SOA Suite"](#).

Overview of InterConnect Integration

Mediator provides a comprehensive application integration framework. Oracle Application Adapter for PeopleSoft used with Mediator enables you to seamlessly integrate enterprise software, eliminating the need to write custom code. Functional modeling, as opposed to custom coding solutions, allows for software reuse and reduces the complexity and management challenges that arise over the software lifecycle. This integration model consists of two components—high-level integration logic and low-level platform services.

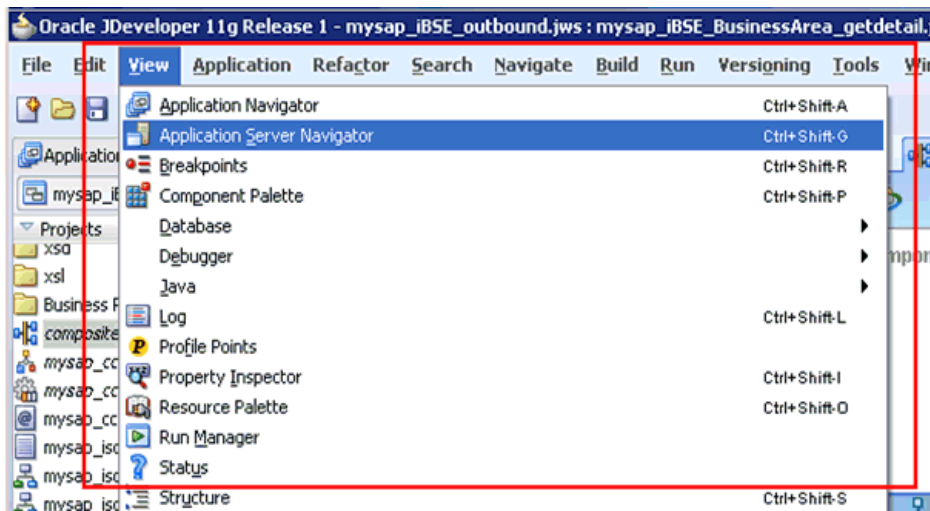
Adapter integration with Oracle WebLogic Server, Mediator is a two-step process:

1. **Design Time:** Oracle Application Adapter for PeopleSoft is configured in Application Explorer for services and events, as described in [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#). Integration logic is modeled in iStudio. Metadata are stored in repositories.
2. **Runtime:** The underlying platform treats this metadata as run-time instructions to enable the communication between participating applications.

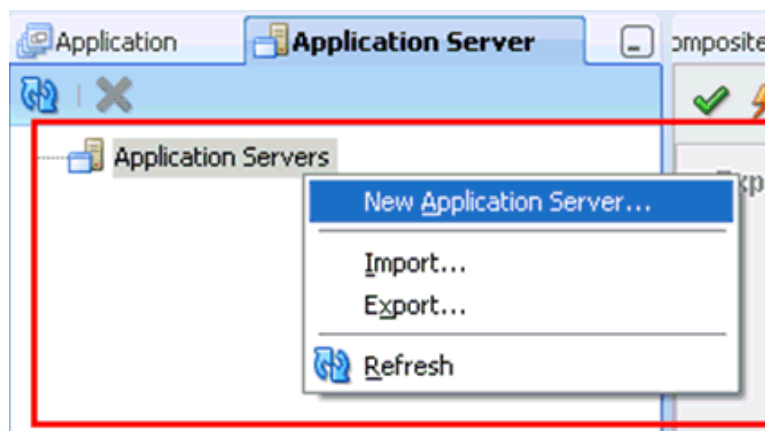
5.1 Configuring a New Application Server Connection

To configure a new Application Server connection in Oracle JDeveloper:

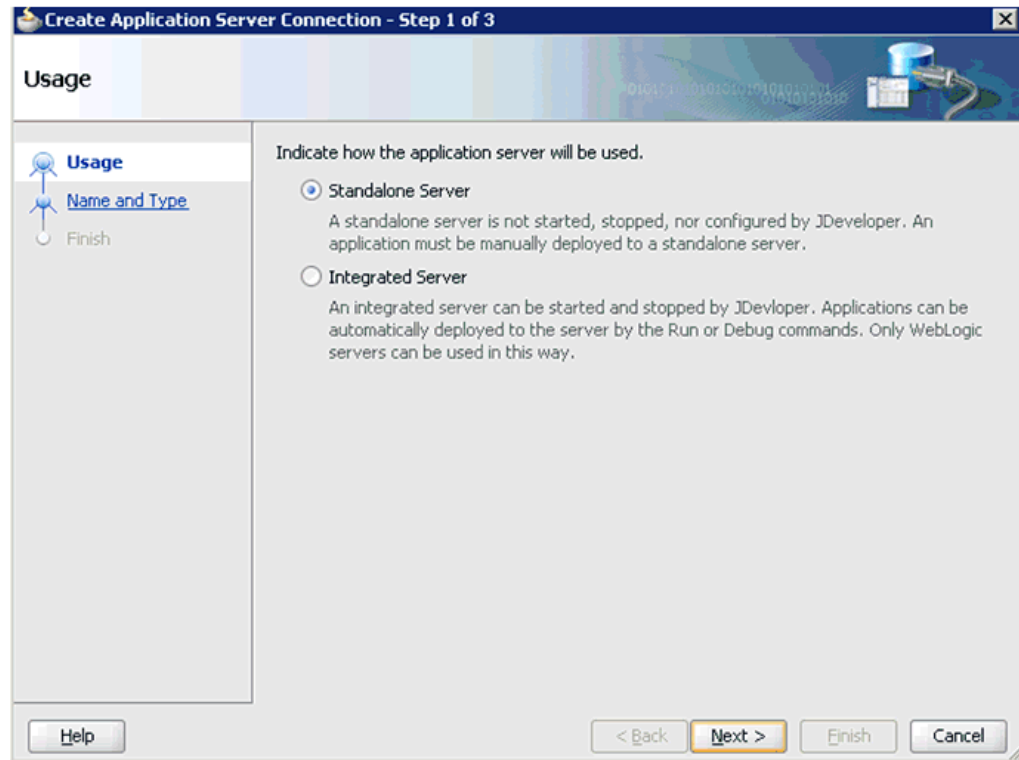
1. Open Oracle JDeveloper on your system.
2. From the menu bar, click **View** and select **Application Server Navigator**.



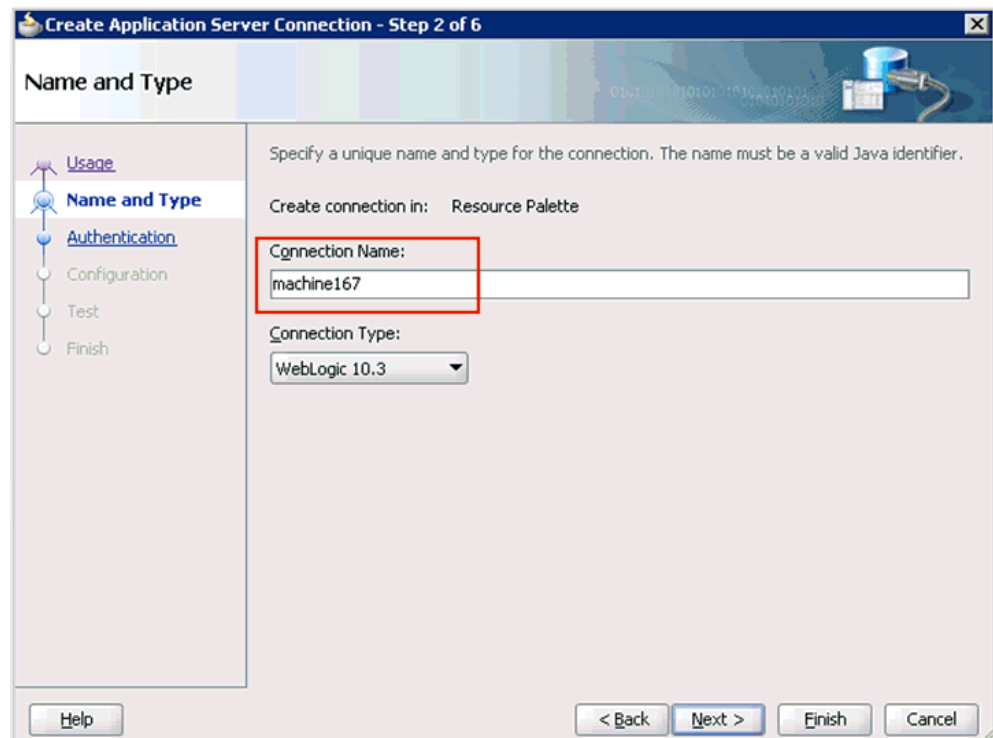
The Application Server tab is displayed.



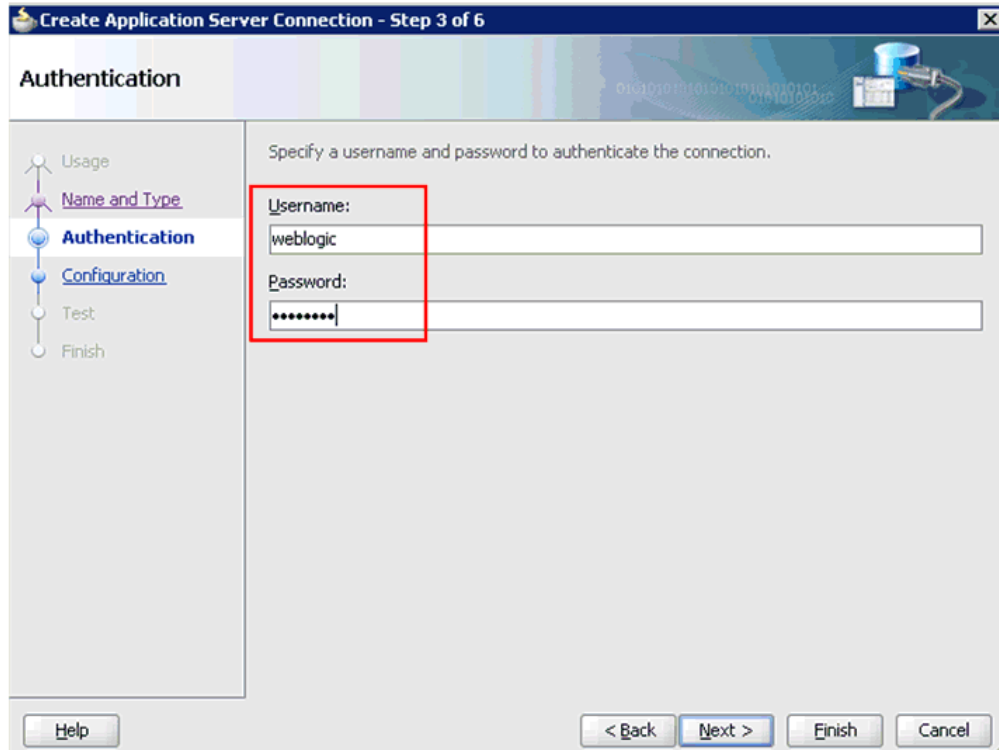
3. Right-click **Application Servers**, and then select **New Application Server**.
The Create Application Server Connection Wizard is displayed.



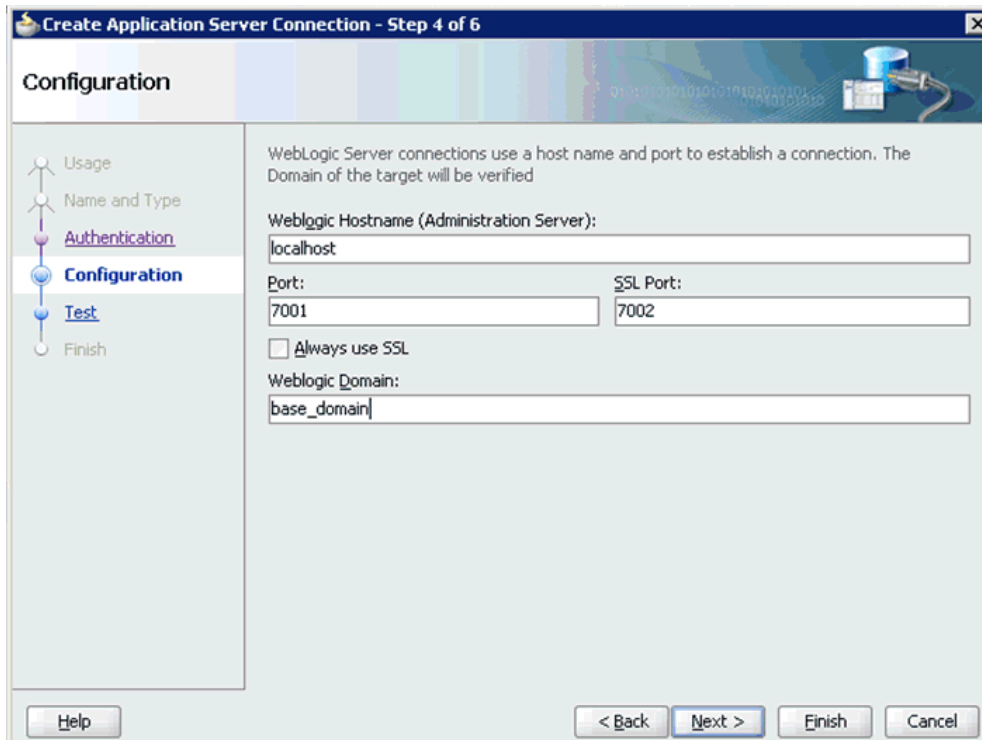
4. Accept the default selection (Standalone Server) and click **Next**.
The Name and Type page is displayed.



5. Specify a new name for the Application Server connection and click **Next**.
The Authentication page is displayed.

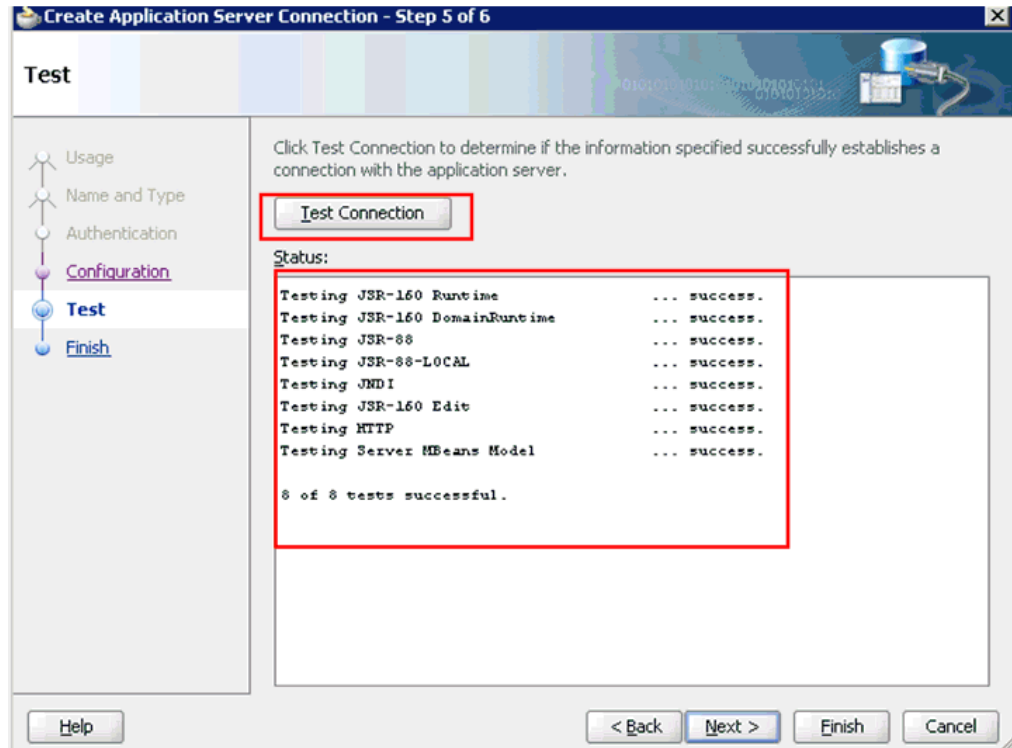


6. Specify a valid user name (for example, weblogic) and a password (for example, welcome1) for your new connection.
7. Click Next.
The Configuration page is displayed.



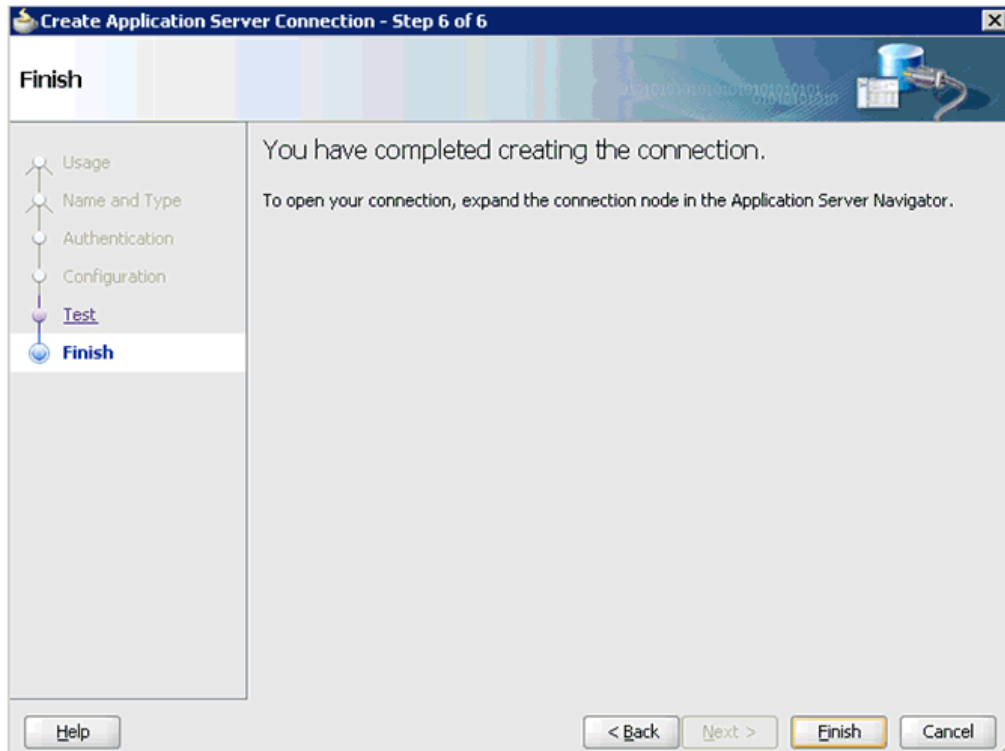
8. Specify the Oracle WebLogic host name (for example, localhost), which is the machine IP where the process needs to deploy and Oracle WebLogic domain (for example, base_domain).
9. Click **Next**.

The Test page is displayed.



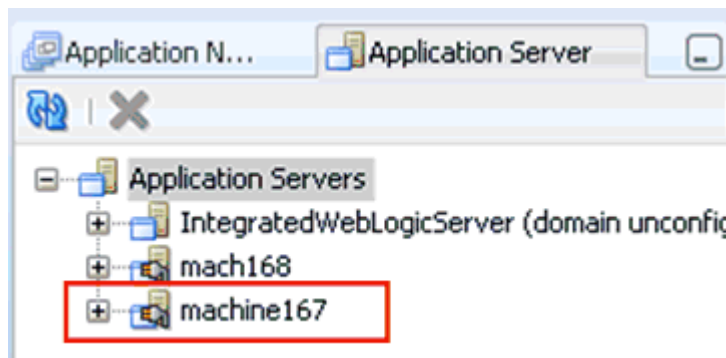
10. Click **Test Connection**.
11. Ensure that the test status is successful.
12. Click **Next**.

The Finish page is displayed.



13. Click Finish.

The new Application Server connection is listed in the left pane (Application Server tab), as shown in the following image.



5.2 Configuring a Mediator Outbound Process

The following example describes how to configure a Mediator outbound process to your PeopleSoft system, using a Mediator project in Oracle JDeveloper.

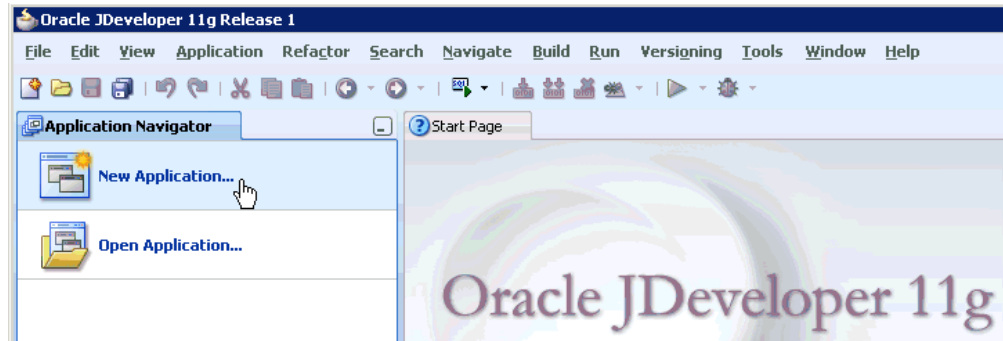
Prerequisites

Before you design a Mediator outbound process, you must generate the respective WSDL file using Application Explorer. See ["Generating WSDL for Request/Response Service"](#) on page 4-7 for more information.

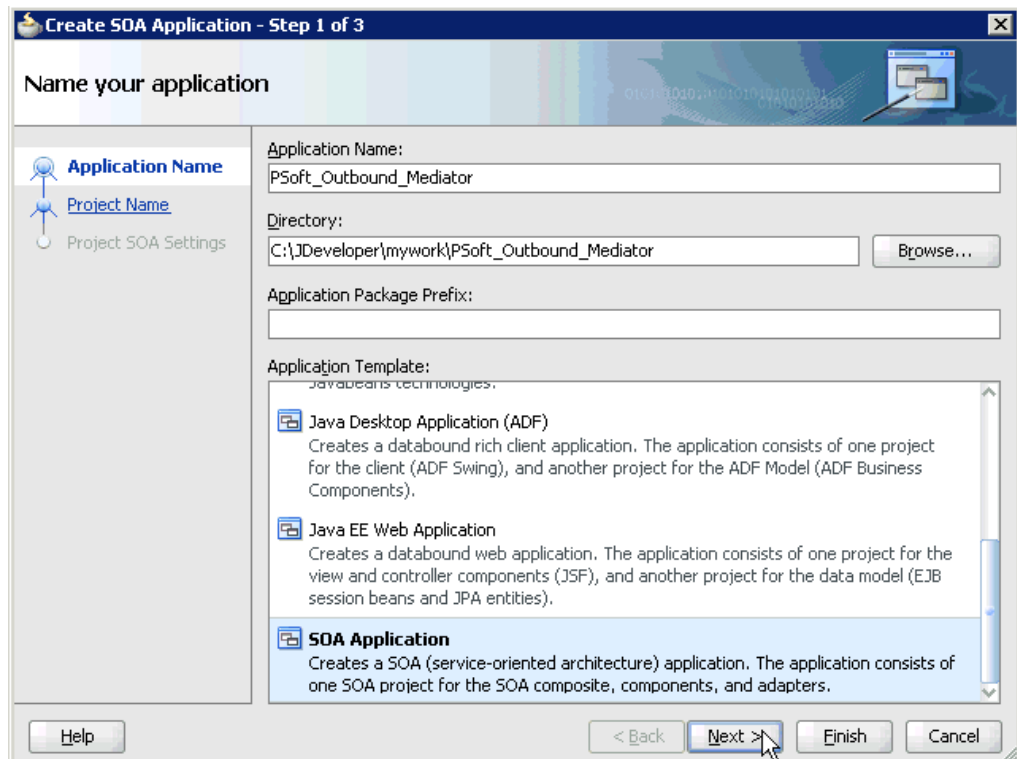
5.2.1 Creating a New SOA Application for the Outbound Mediator Process

Perform the following steps to create a new SOA application for the outbound Mediator process:

1. Open Oracle JDeveloper on your system.
2. In the Application Navigator tab, click **New Application**.

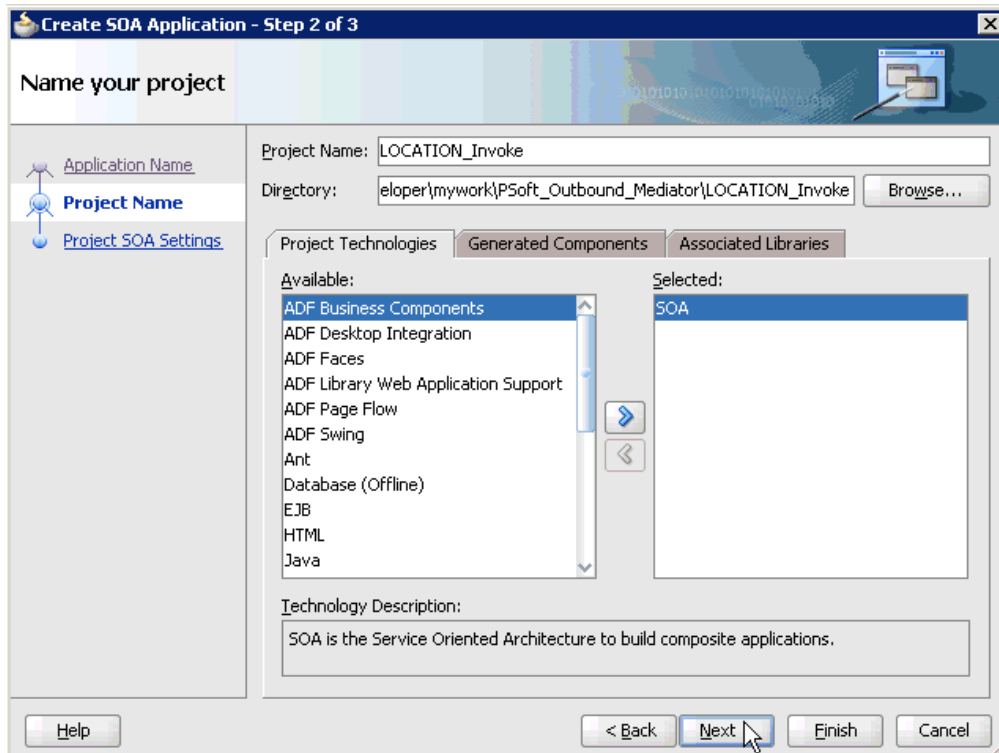


The Create SOA Application wizard is displayed.

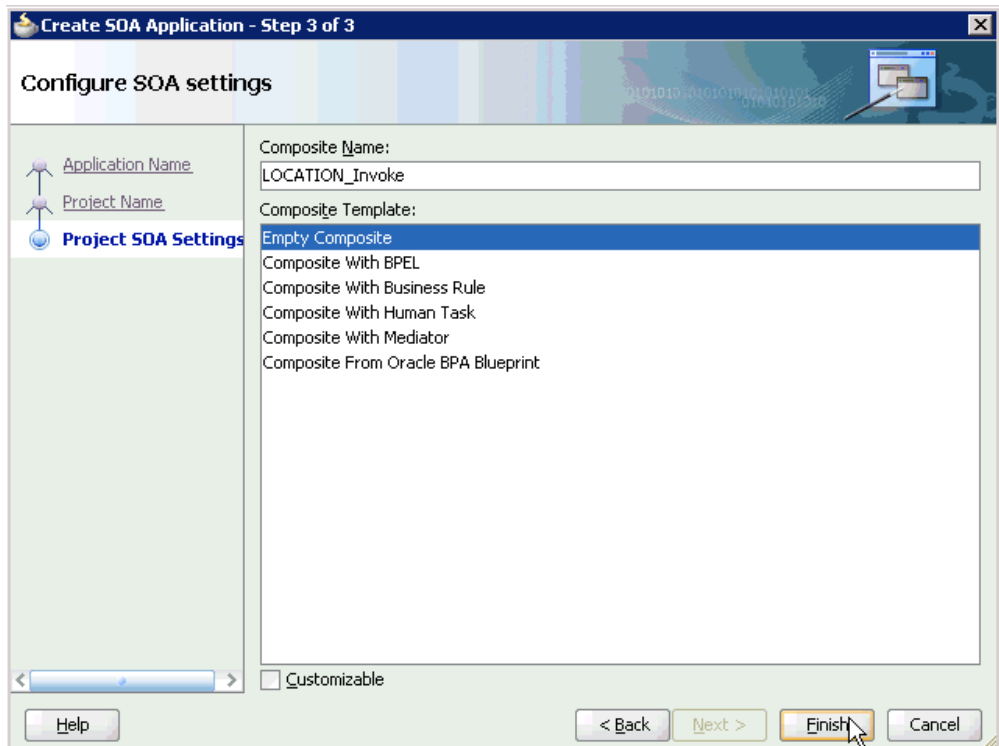


3. From the Application Template list, click **SOA Application**.
4. Enter name for the new SOA application (for example, PSoft_Outbound_Mediator) and click **Next**.

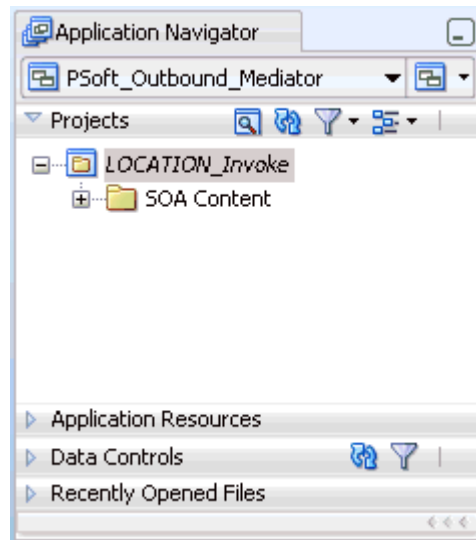
The Name your project page is displayed.



5. Enter a project name (for example, LOCATION_Invoke) and click **Next**.
The Configure SOA settings page is displayed.



6. From the Composite Template list, select **Empty Composite** and click **Finish**.



The new SOA application (PSoft_Outbound_Mediator) and associated project (LOCATION_Invoke) are added to the Application Navigator tab in the left pane.

5.2.2 Defining a Mediator Outbound Process

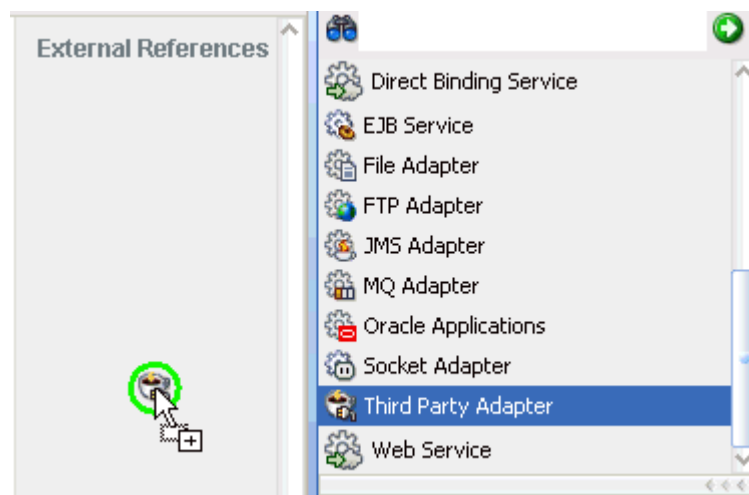
This section describes how to define a Mediator outbound process, which consists of the following stages:

1. Configuring a Third Party Adapter Service Component
2. Configuring an Outbound Mediator Process Component
3. Configuring the Routing Rules

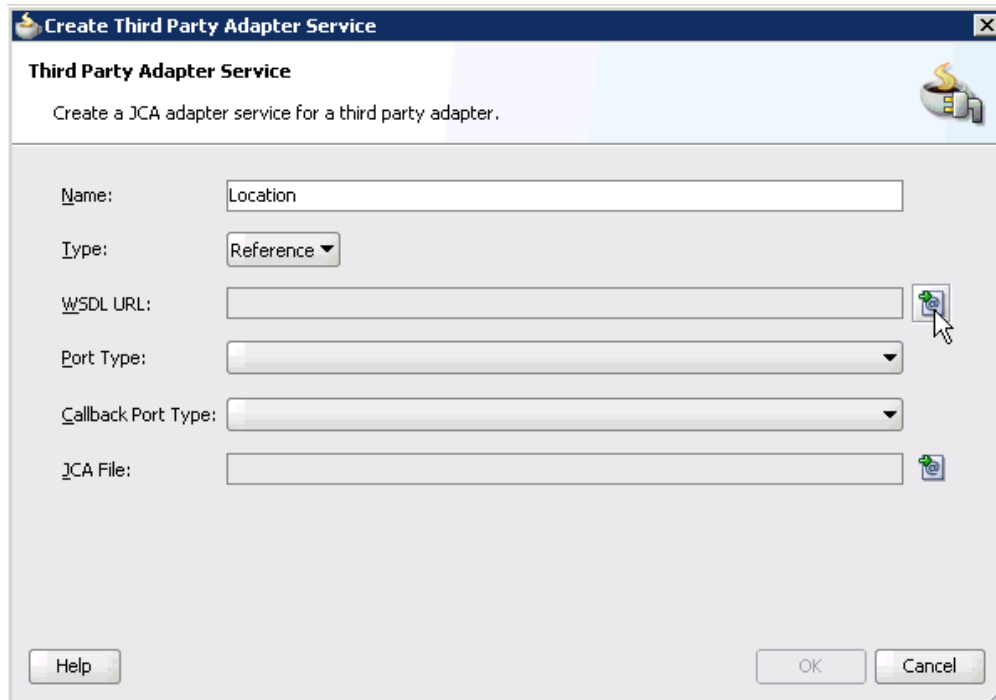
Configuring a Third Party Adapter Service Component

Perform the following steps to create a third party adapter service component:

1. Drag and drop the **Third Party Adapter** component from the Component Palette tab (Service Adapters section) to the External References pane.

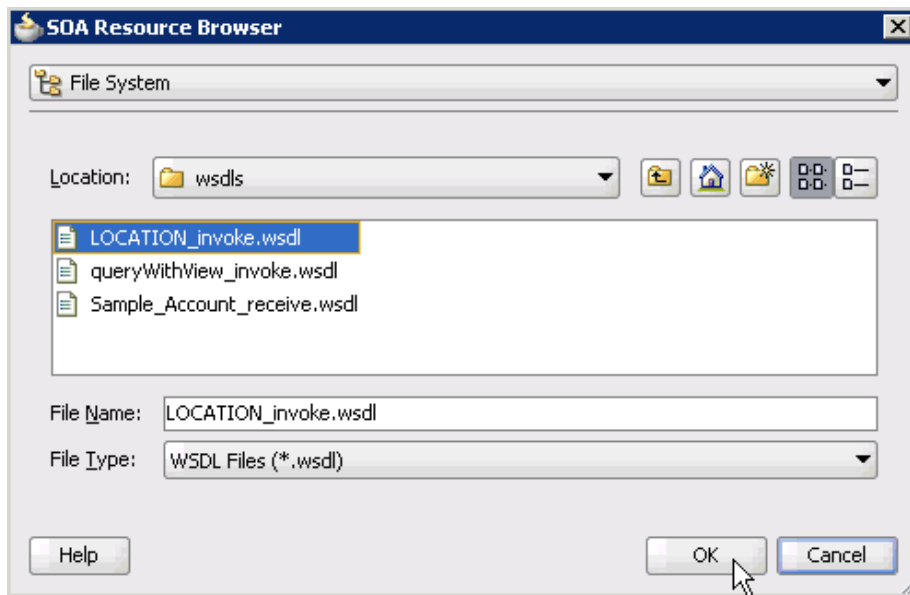


The Create Third Party Adapter Service dialog is displayed.



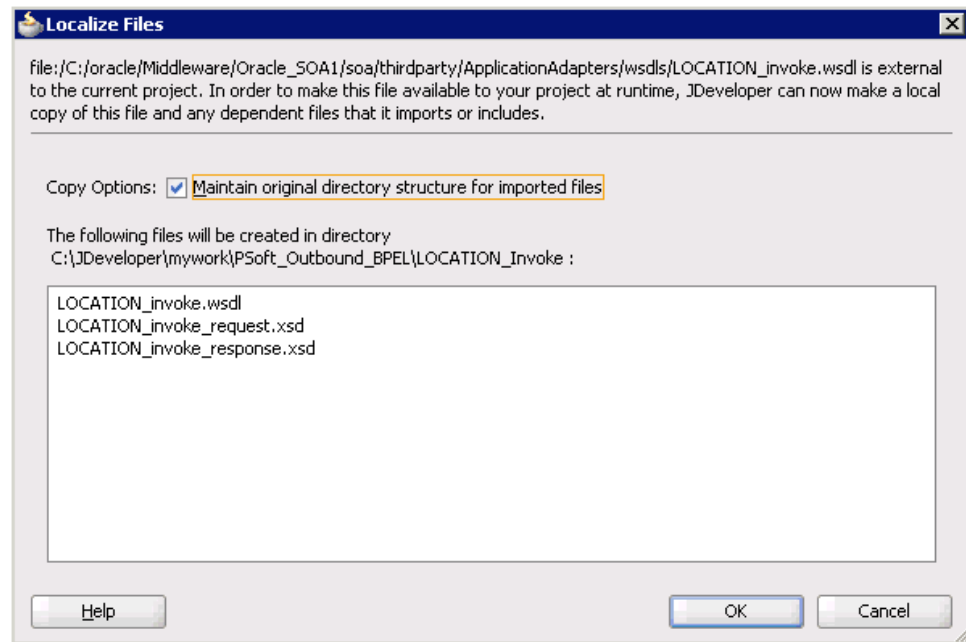
2. Enter a name for the new third party adapter service.
3. Ensure that **Reference** is selected from the Type list (default).
4. Click the **Find existing WSDLs** icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed.



5. Browse and select an outbound WSDL file (for example, LOCATION_invoke.wSDL) from the following directory:
 C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\wsdls
6. Click **OK**.

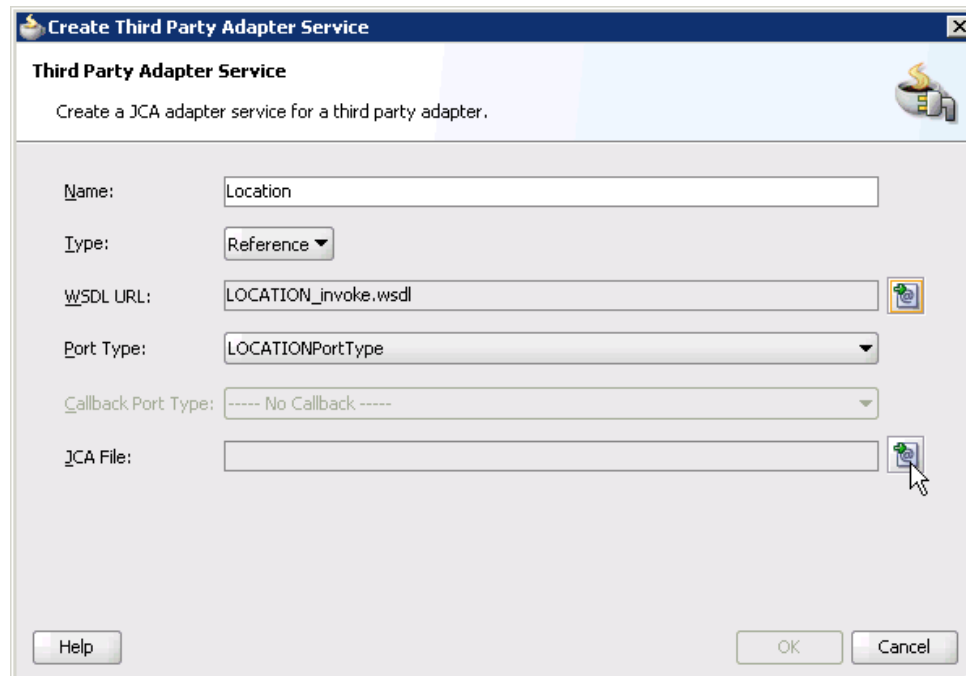
The Localize Files dialog is displayed.



7. Click **OK**.

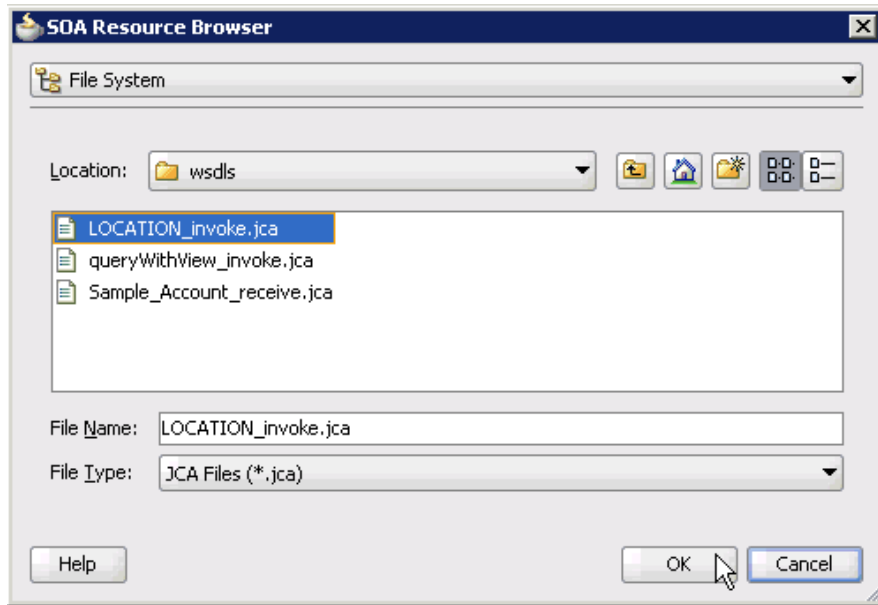
The outbound WSDL file and associated request and response XML schema files (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog.



8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field.

The SOA Resource Browser dialog is displayed.

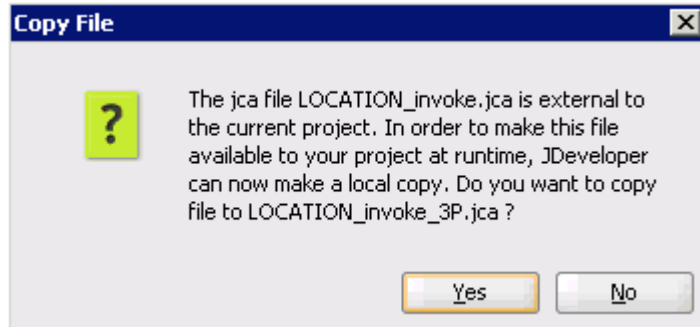


- 9. Browse and select the JCA properties file (for example, LOCATION_invoke.jca) from the following directory:

C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\wsdls

- 10. Click **OK**.

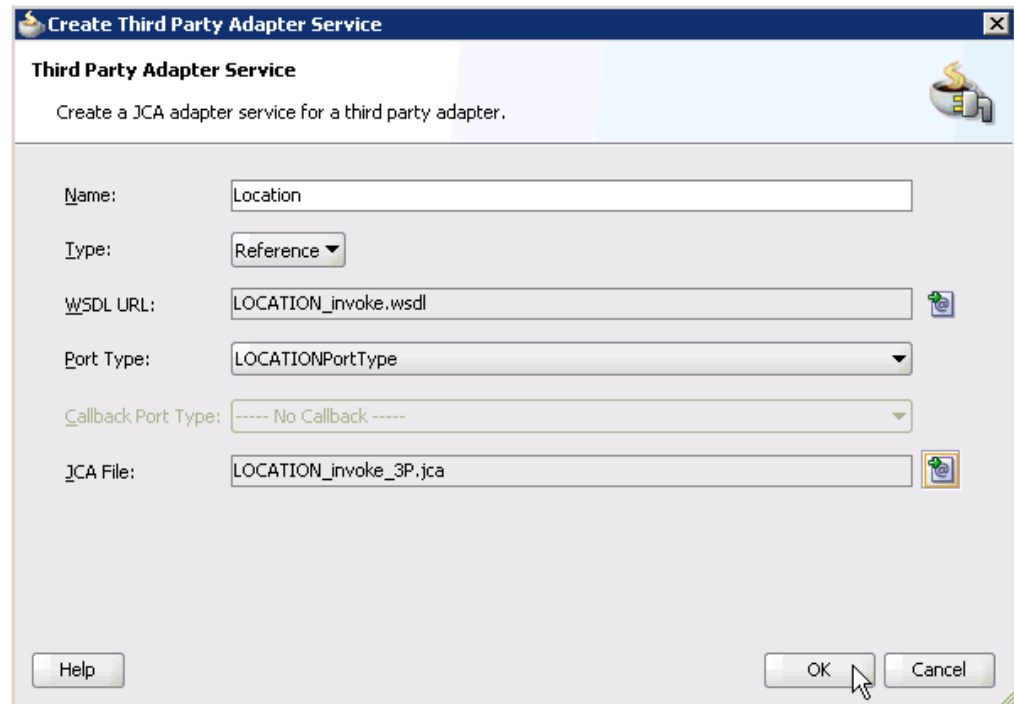
The following message is displayed.



- 11. Click **Yes**.

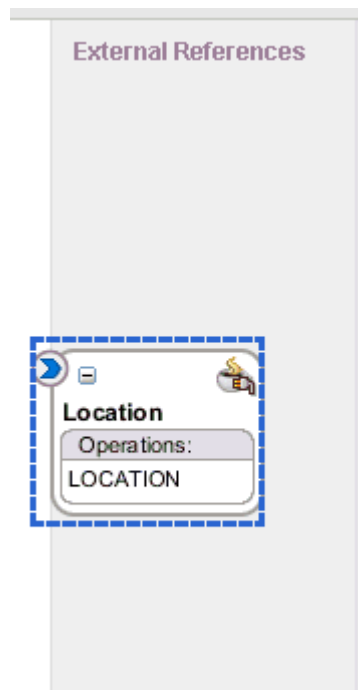
A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog.



12. Click OK.

The third party adapter service component (Location) is created in the External References pane, as shown in the following image.

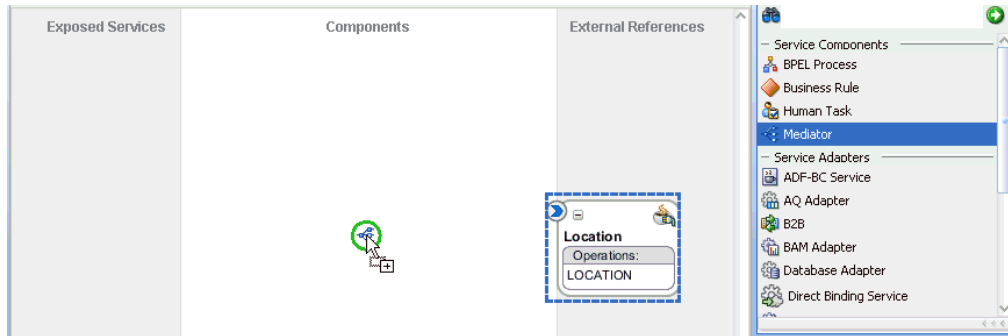


You are now ready to configure an outbound Mediator process component.

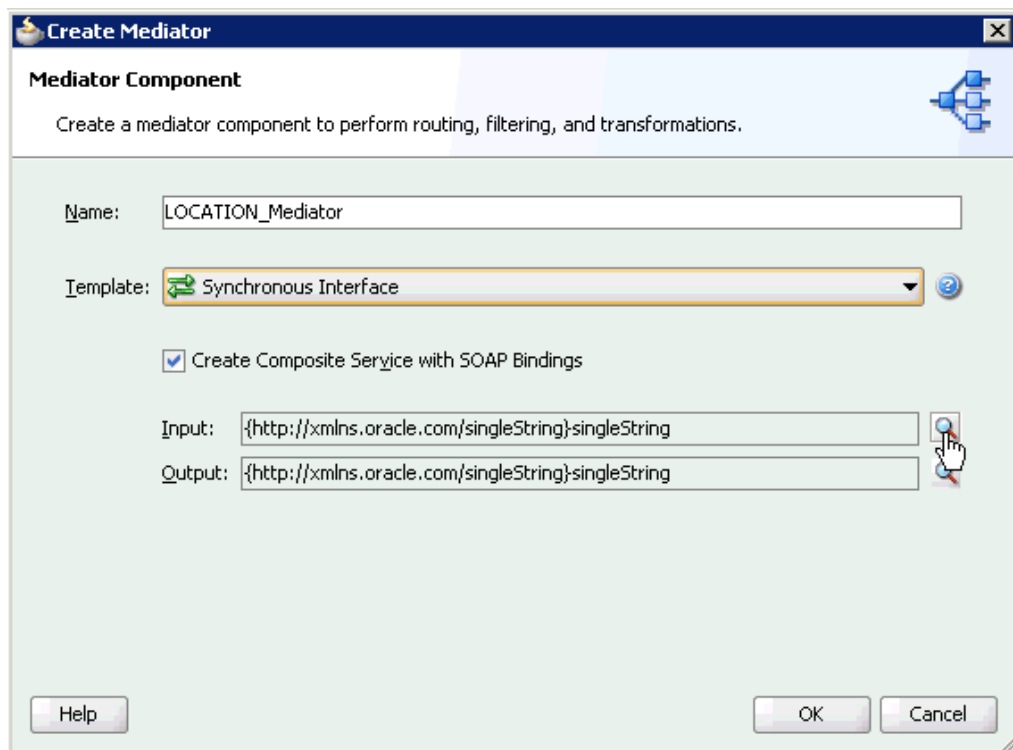
Configuring an Outbound Mediator Process Component

Perform the following steps to configure an outbound Mediator process component:

1. Drag and drop the **Mediator** component from the Component Palette tab (Service Components section) to the Components pane.

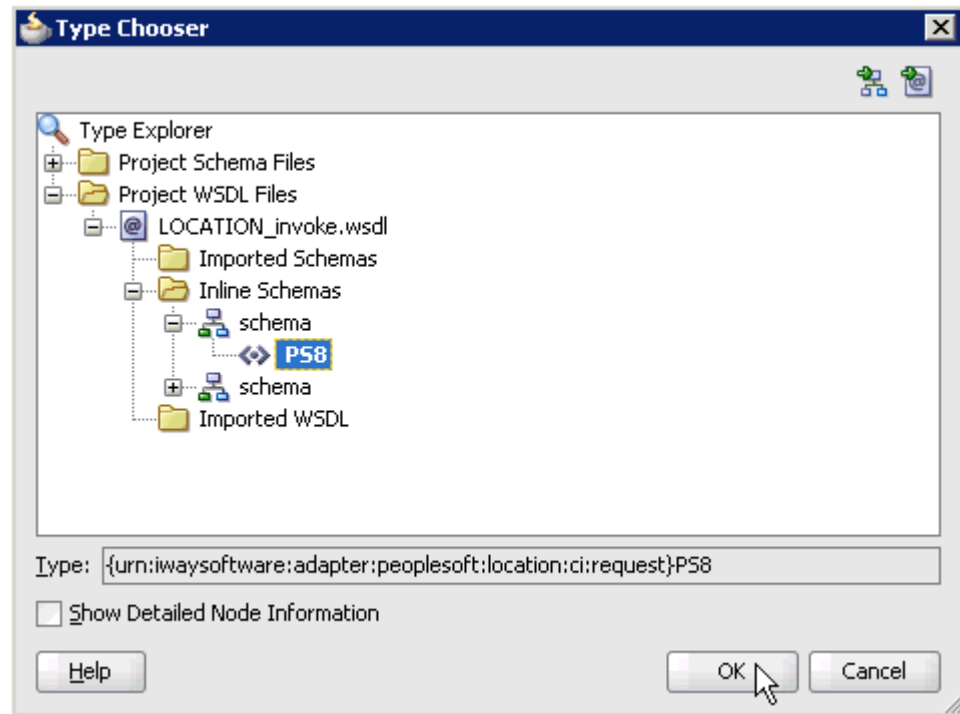


The Create Mediator dialog is displayed.



2. In the Name field, enter a name to identify the new outbound Mediator process component (for example, LOCATION_Mediator).
3. From the Template list, select **Synchronous Interface**.
4. Click the **Browse Input Elements** icon, which is located to the right of the Input field to select the associated XML request schema file.

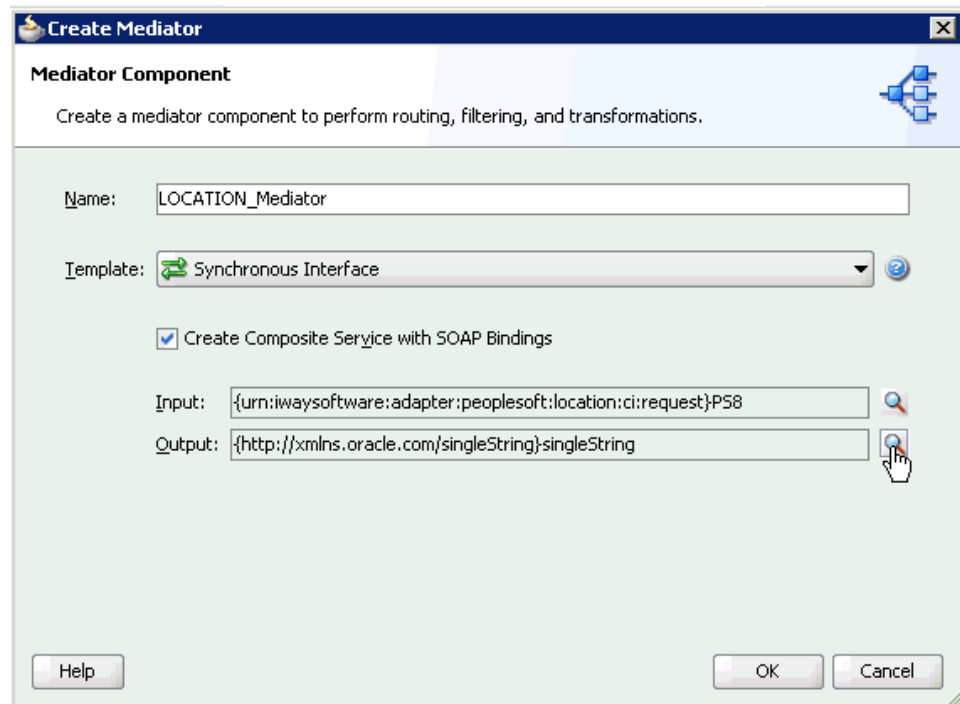
The Type Chooser dialog is displayed.



5. Expand **Project WSDL Files**, **LOCATION_invoke.wSDL**, **Inline Schemas**, **schema**, and select **PS8**.

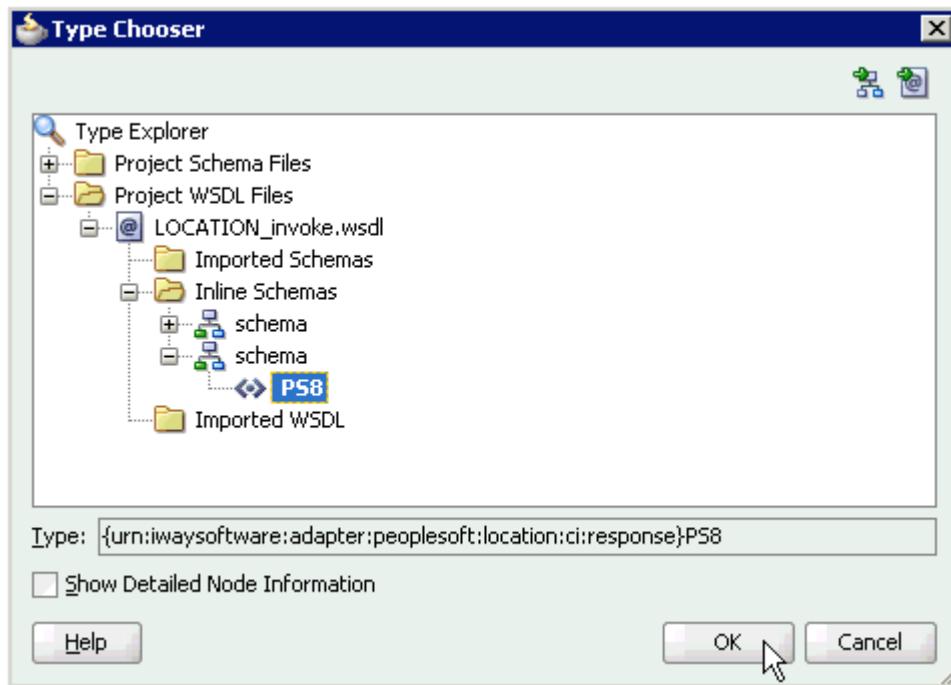
6. Click **OK**.

You are returned to the Create Mediator dialog.



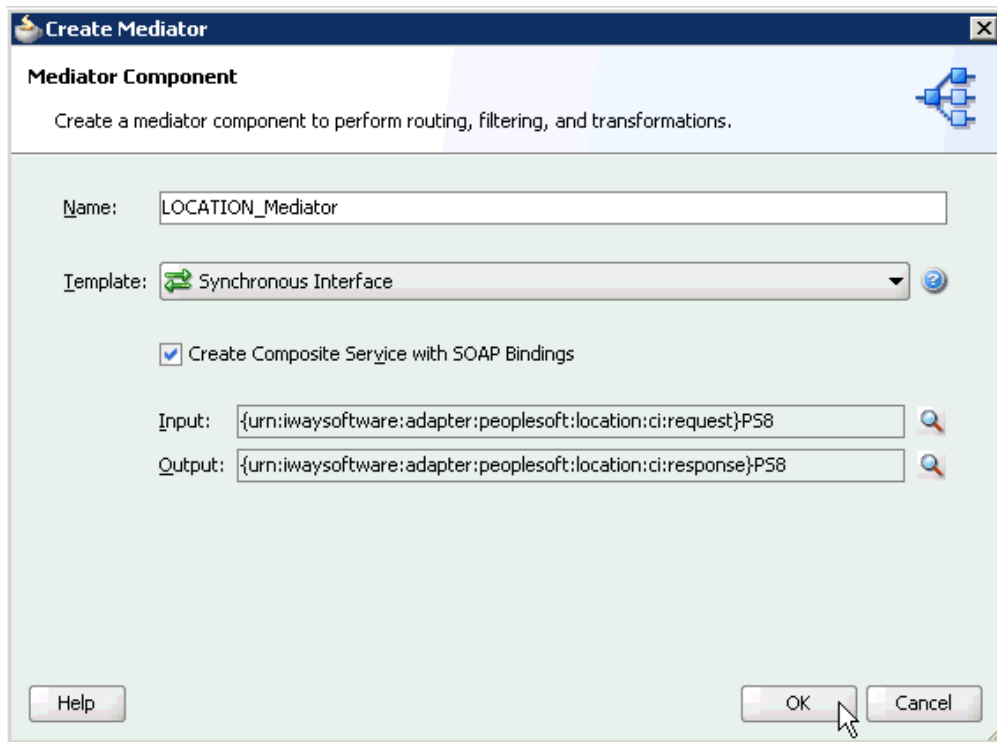
7. Click the **Browse Output Elements** icon, which is located to the right of the Output field to select the associated XML response schema file.

The Type Chooser dialog is displayed.



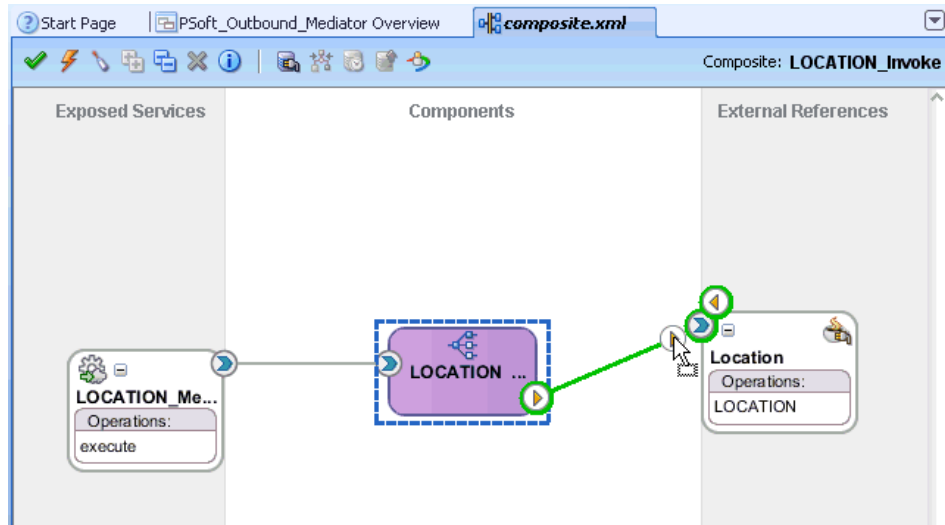
8. Expand **Project WSDL Files**, **LOCATION_invoke.wsdl**, **Inline Schemas**, **schema**, and select **PS8**.
9. Click **OK**.

You are returned to the Create Mediator dialog.



10. Click **OK**.

11. Create a connection between the outbound Mediator process component (LOCATION_Mediator) and the third party adapter service component (Location).

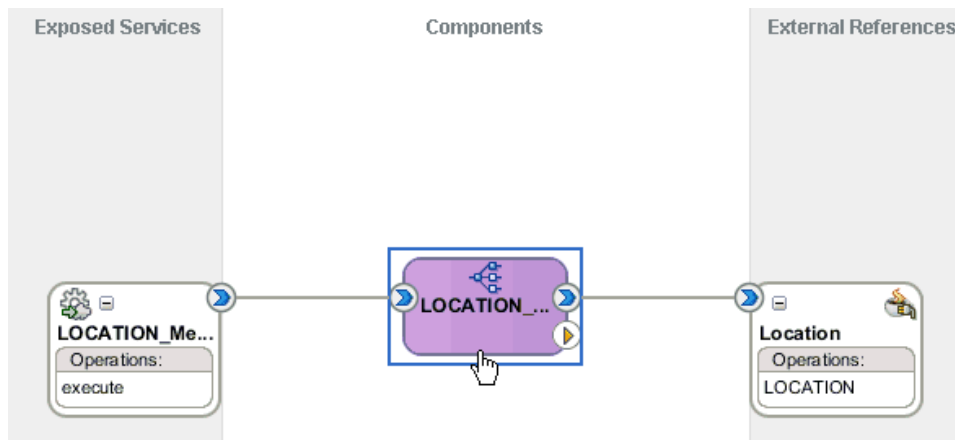


You are now ready to configure the routing rules.

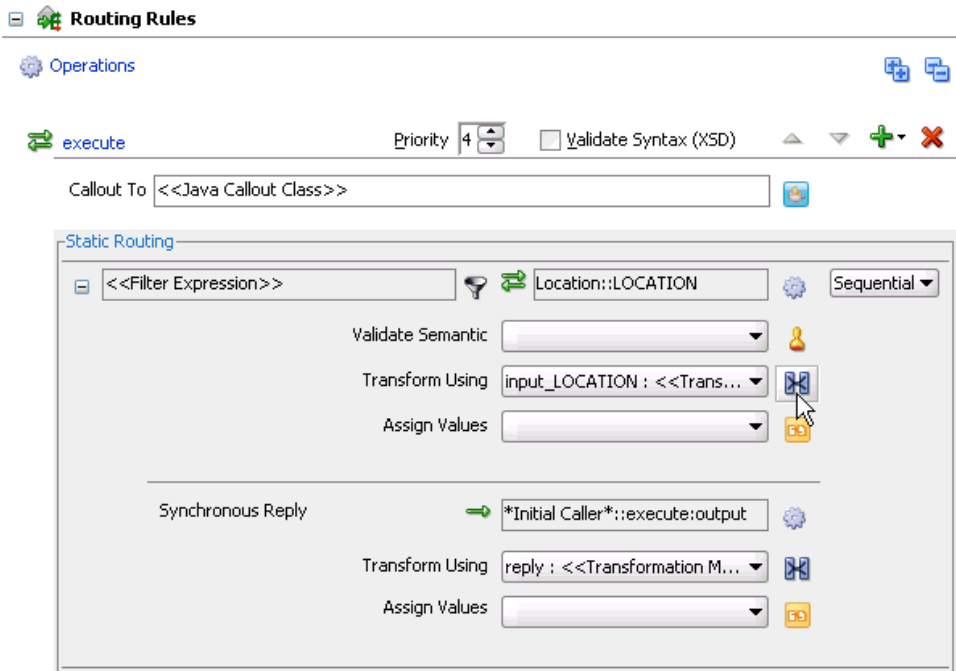
Configuring the Routing Rules

Perform the following steps to configure routing rules for the Mediator outbound process component:

1. Double-click the outbound Mediator process component (LOCATION_Mediator) in the Components pane.

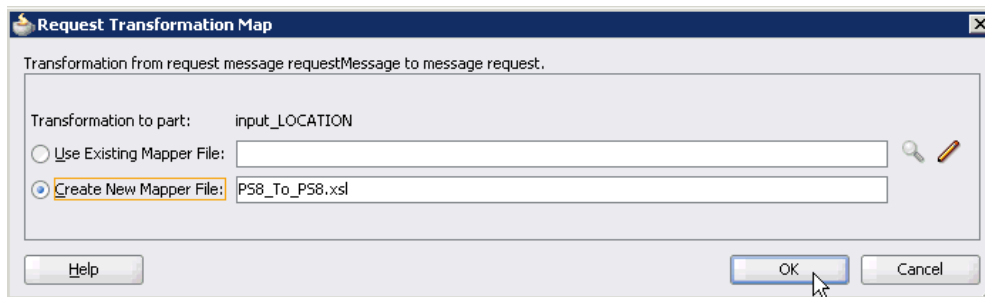


The Routing Rules dialog is displayed.



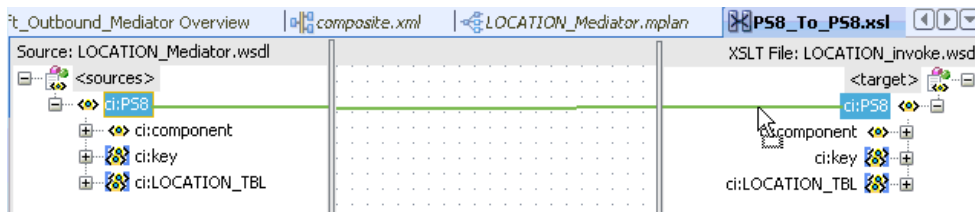
- In the <<Filter Expression>> area, click the **Select an existing mapper file or create a new one** icon to the right of the Transform Using field.

The Request Transformation Map dialog is displayed.



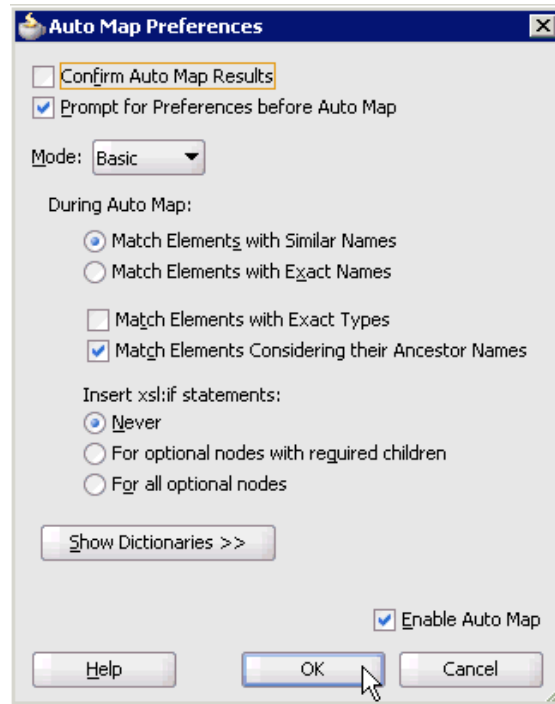
- Select the **Create New Mapper File** option and click **OK**.

The PS8_To_PS8.xsl tab is displayed.

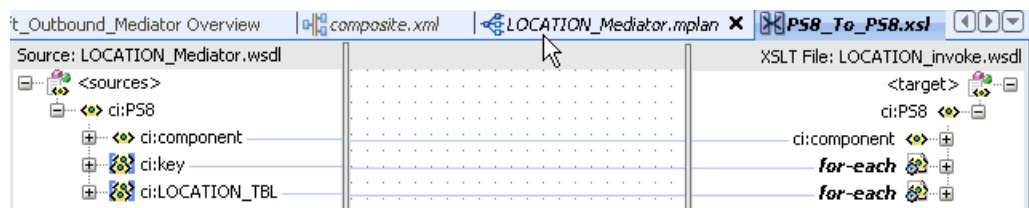


- Map the **ci:PS8** source element to the **ci:PS8** target element.

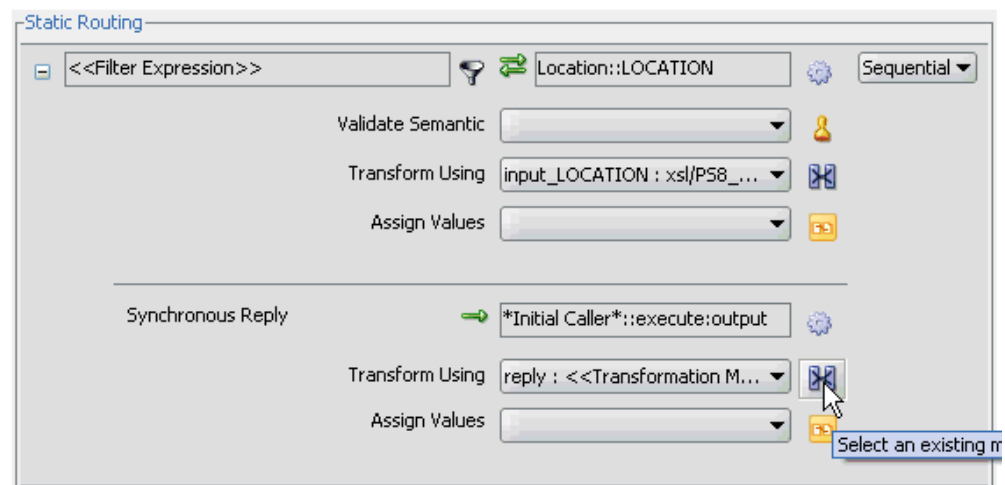
The Auto Map Preferences dialog is displayed.



5. Retain the default values and click **OK**.
6. Click the **LOCATION_Mediator.mplan** tab.

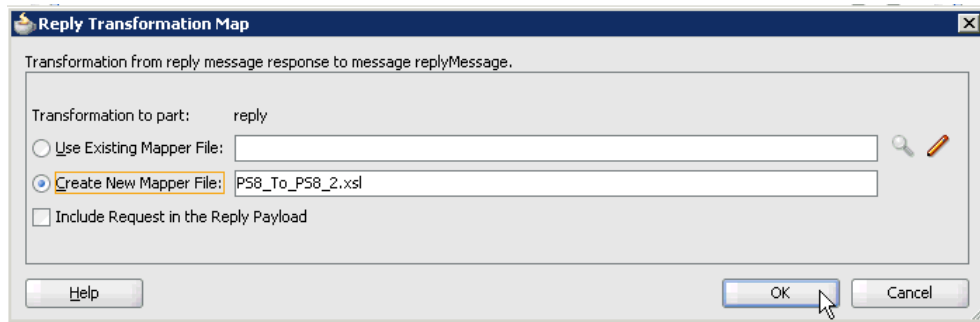


You are returned to the Routing Rules dialog.



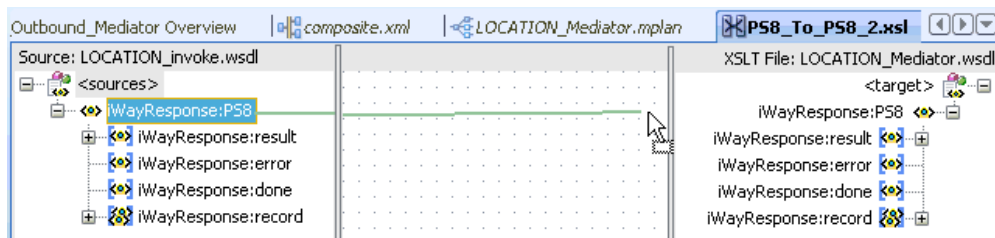
7. In the Synchronous Reply area, click the **Select an existing mapper file or create a new one** icon to the right of the Transform Using field.

The Reply Transformation Map dialog is displayed.



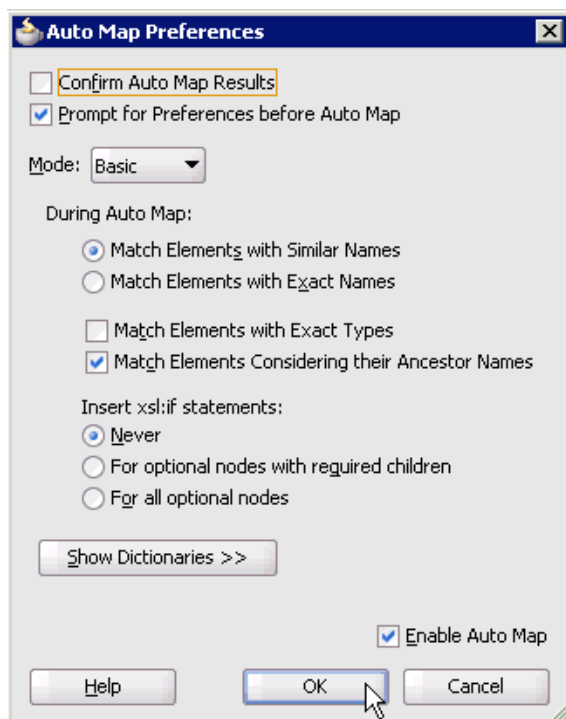
8. Select the **Create New Mapper File** option and click **OK**.

The PS8_To_PS8_2.xsl tab is displayed.



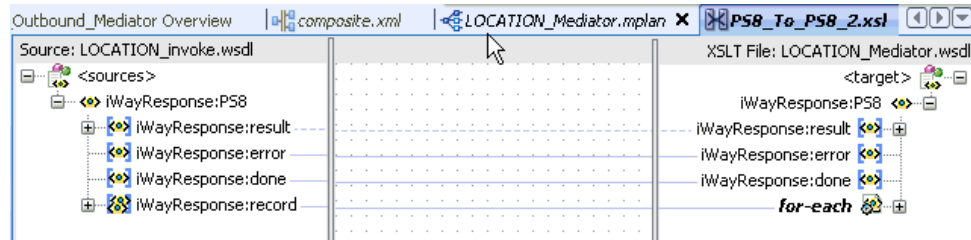
9. Map the **iWayResponse:PS8** source element to the **iWayResponse:PS8** target element.

The Auto Map Preferences dialog is displayed.

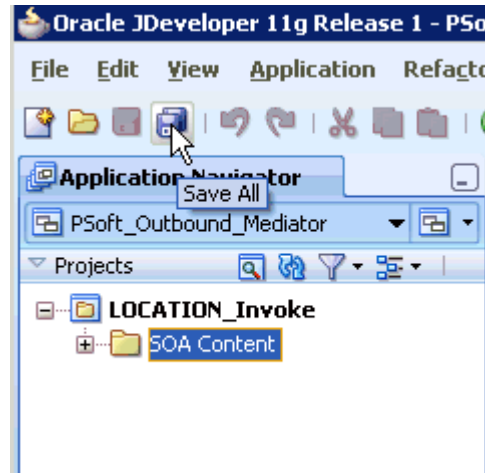


10. Retain the default values and click **OK**.

The mapping is complete, as shown in the following image.



11. Click the **Save All** icon in the menu bar to save the new outbound Mediator process component that was configured.

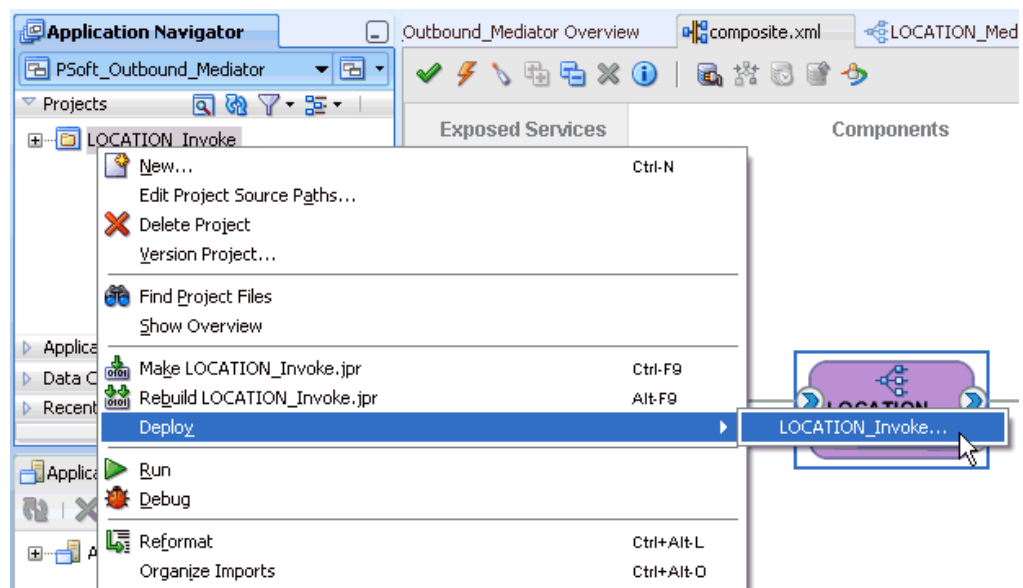


You are now ready to deploy the Mediator outbound process.

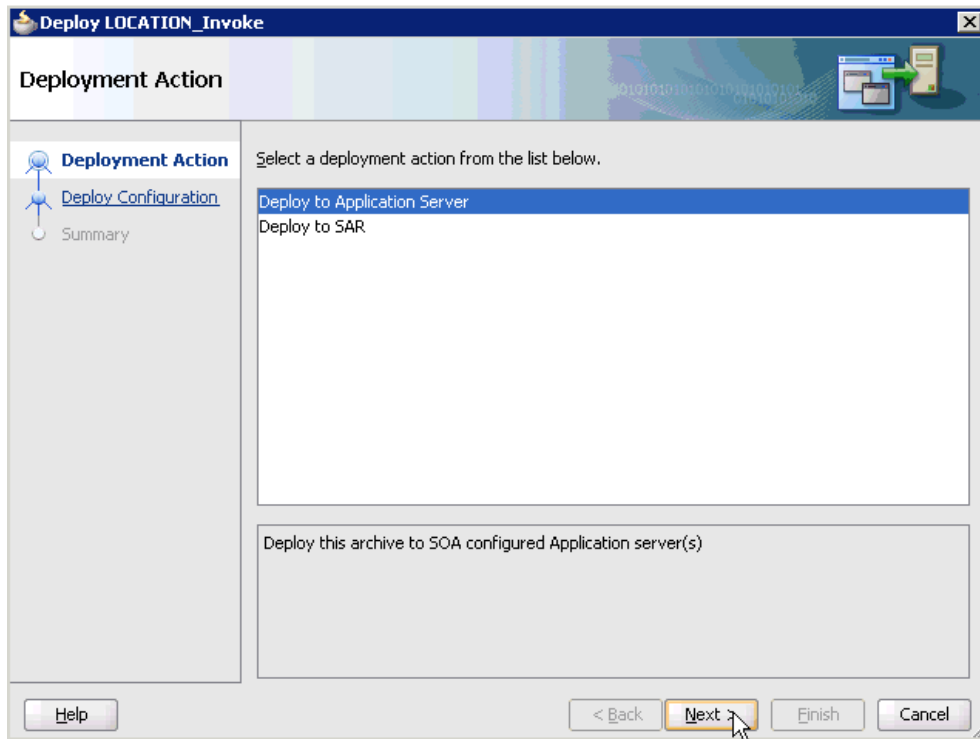
5.2.3 Deploying the Mediator Outbound Process

Perform the following steps to deploy the Mediator outbound process.

1. Right-click the project name in the left pane (for example, **LOCATION_Invoke**), select **Deploy**, and then click **LOCATION_Invoke**.



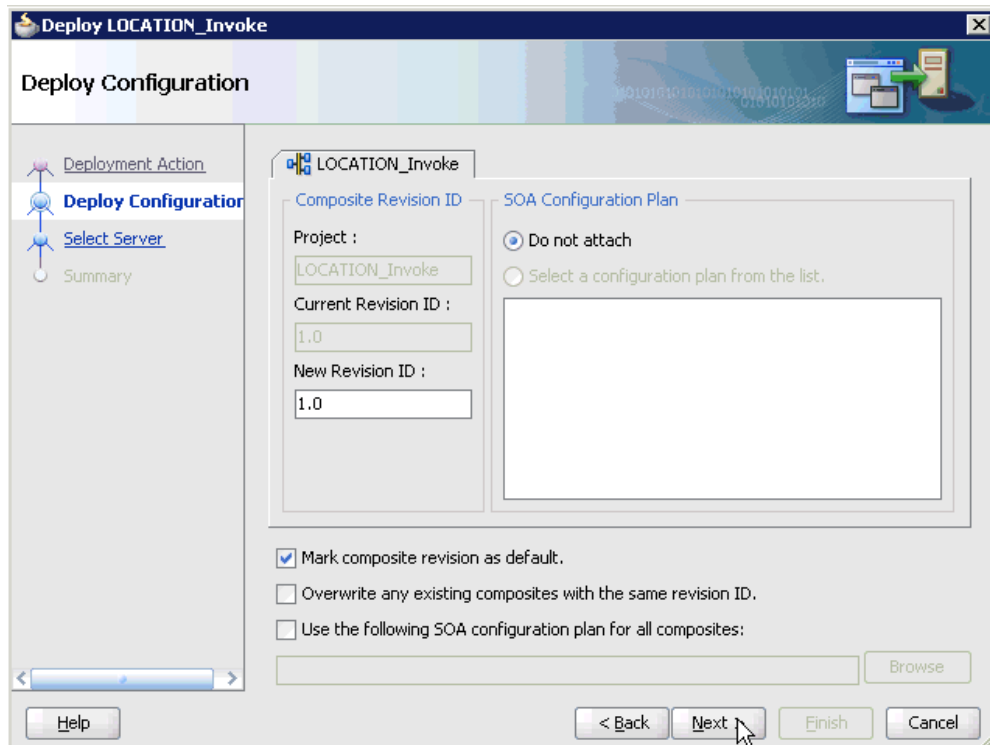
The Deployment Action page is displayed.



2. Ensure that **Deploy to Application Server** is selected.

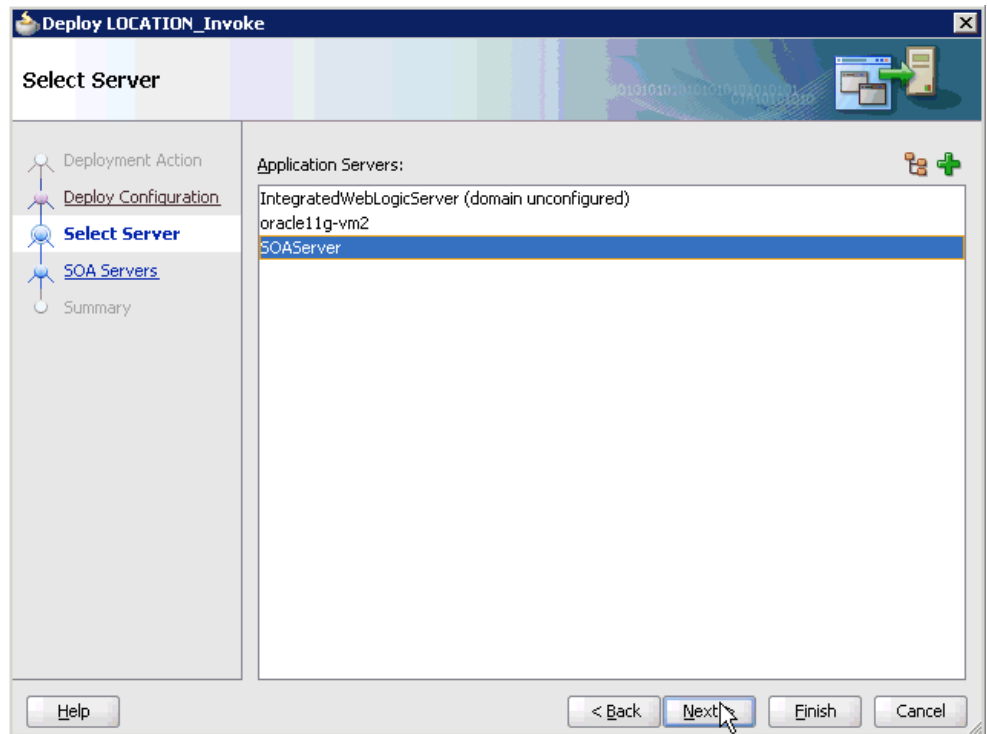
3. Click **Next**.

The Deploy Configuration page is displayed.



4. Leave the default values selected and click **Next**.

The Select Server page is displayed.

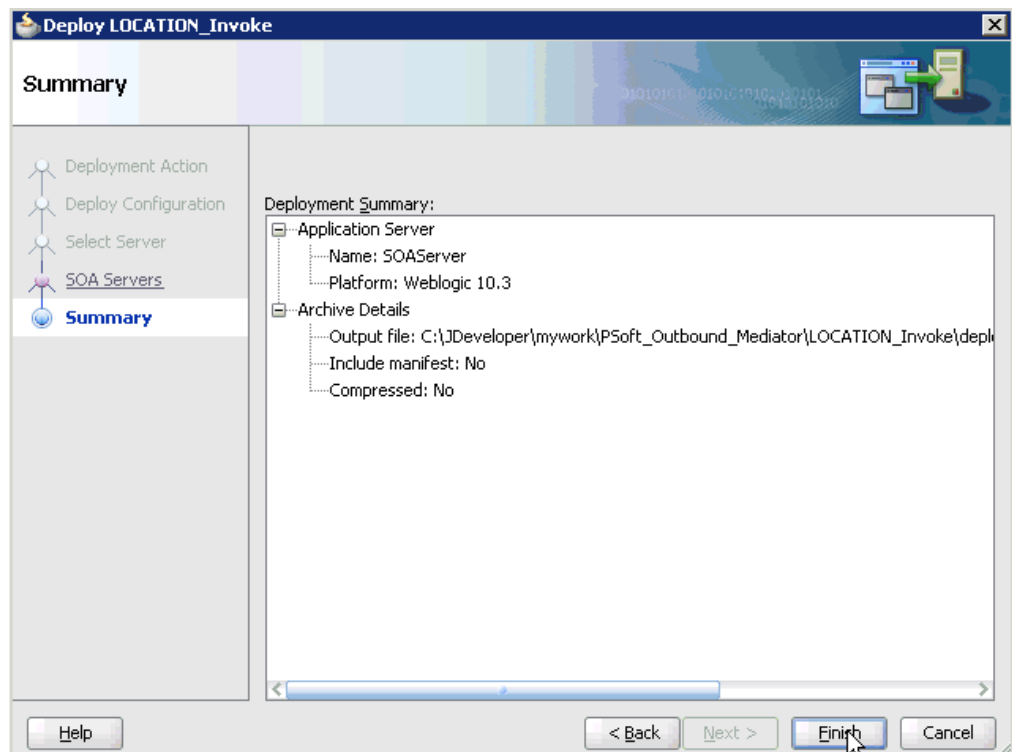


5. Select an available application server that was configured and click **Next**.

The SOA Servers dialog is displayed.

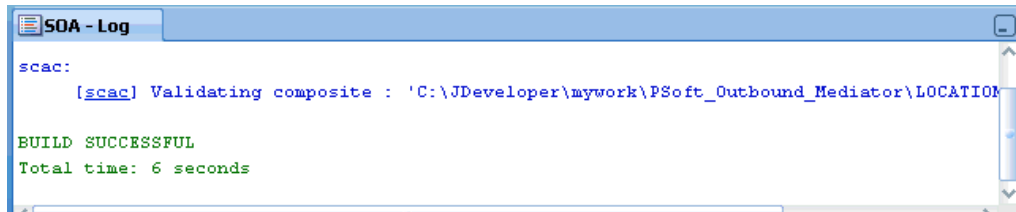
6. Select a target SOA server and click **Next**.

The Summary page is displayed.



- Review and verify all the available deployment information for your project and click **Finish**.

The process is deployed successfully.

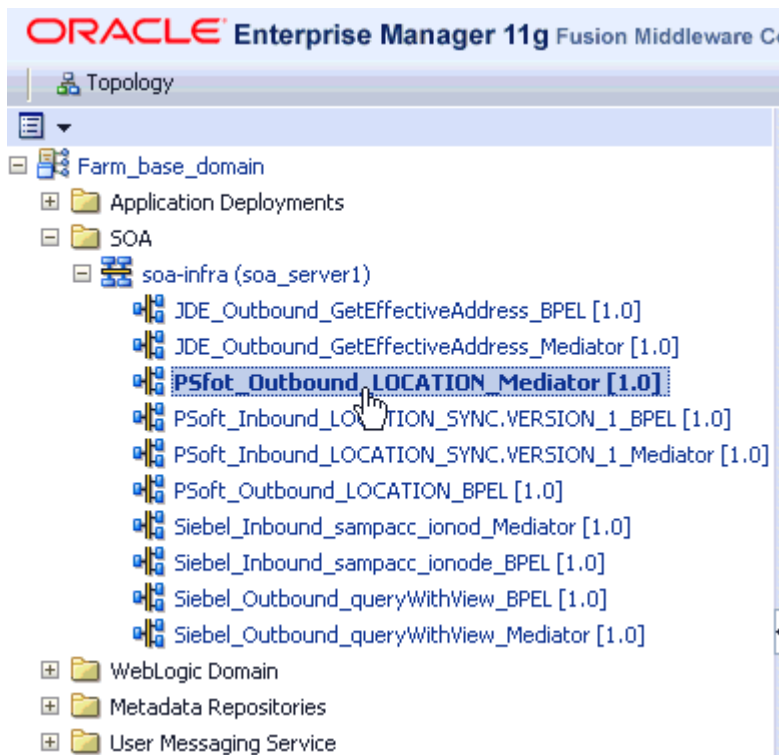


If an Authorization Request dialog is displayed during the deployment process, provide the required user name and password and click **OK**.

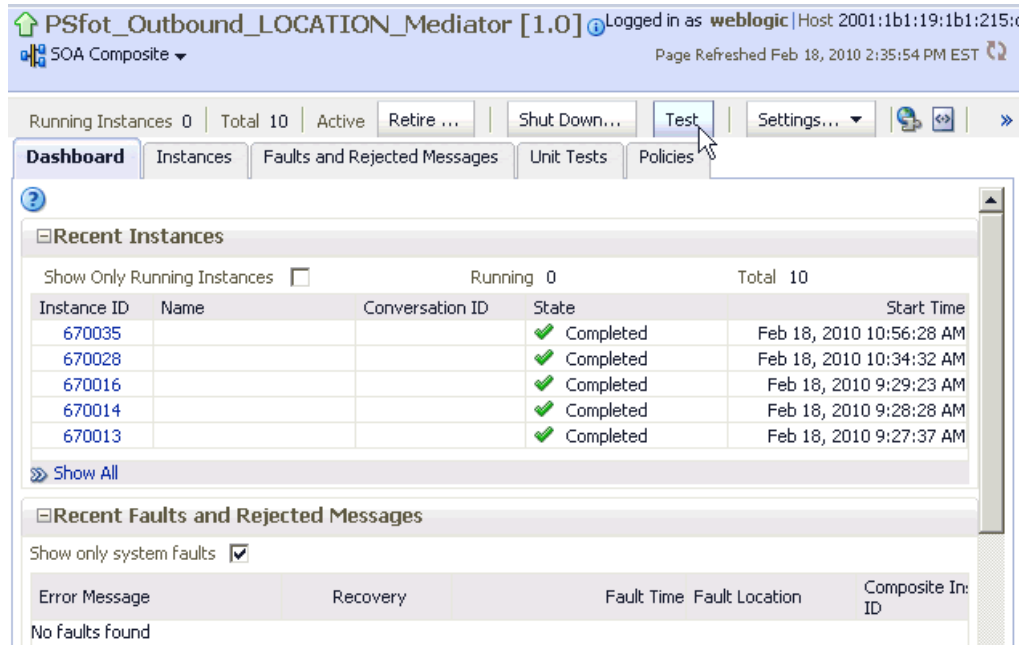
5.2.4 Invoking the Input XML Document in the Oracle Enterprise Manager Console

Perform the following steps to invoke the input XML document in the Oracle Enterprise Manager console.

- Logon to the Oracle Enterprise Manager console by using the following URL:
<http://localhost:7001/em>
- Expand your domain in the left pane followed by the **SOA** folder.

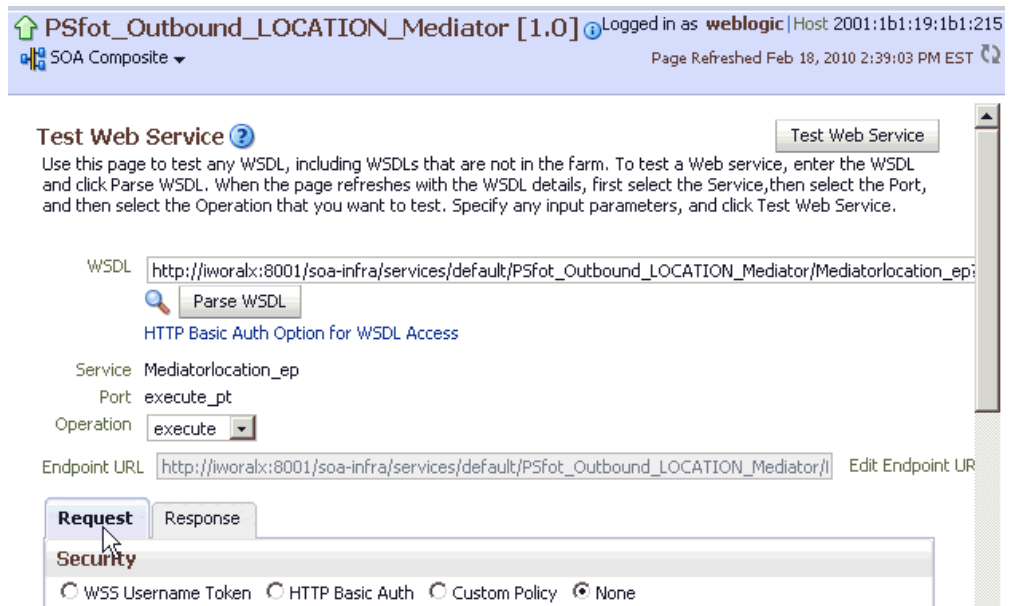


- Select an available project (for example, PSoft_Outbound_LOCATION_Mediator).

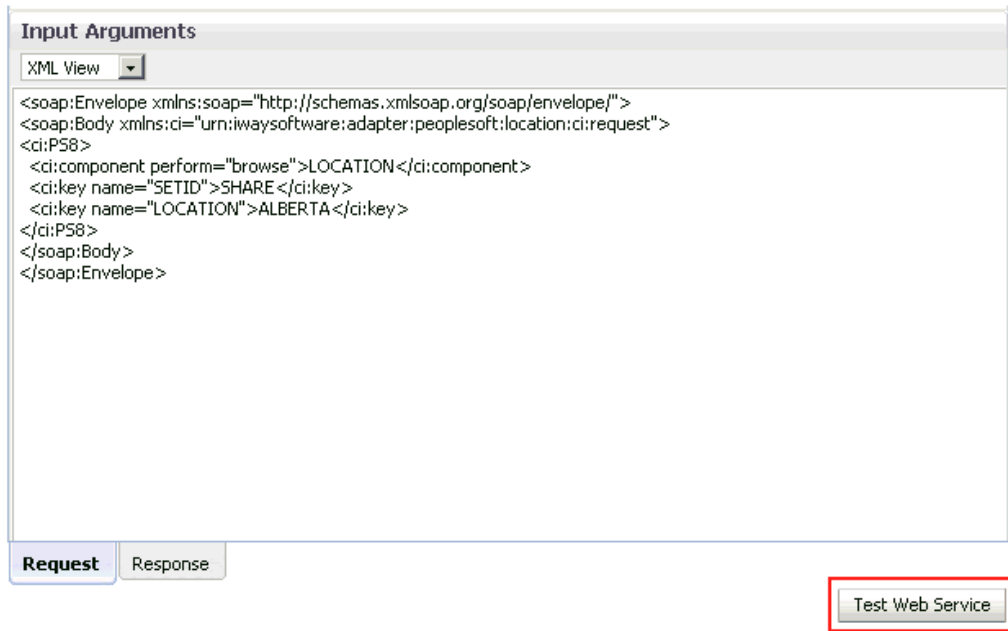


- Click **Test** in the right pane.

The Test Web Service page is displayed.



- Click the **Request** tab.
- Scroll down to the Input Arguments section.



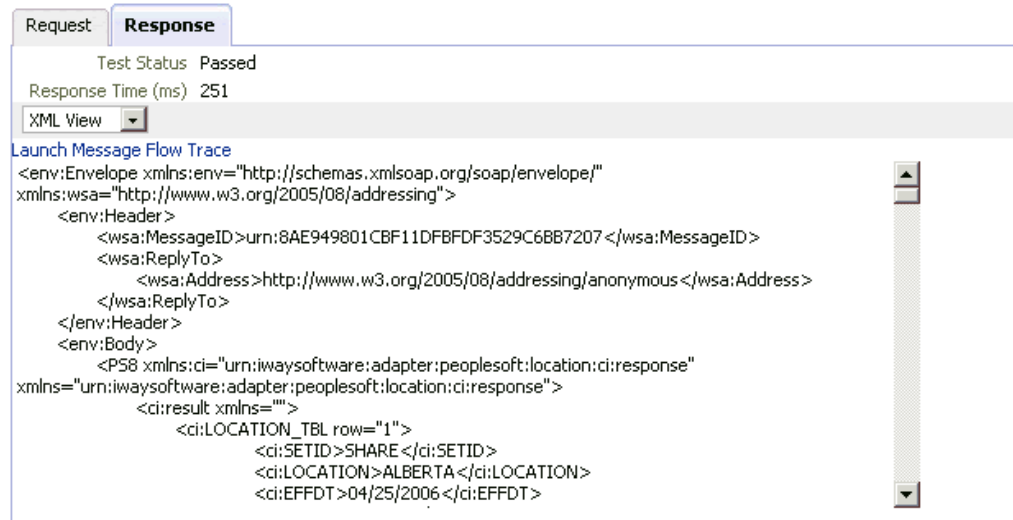
7. Select **XML View** from the list in the upper-left corner.
8. Provide an appropriate input XML document in the Input Arguments area.

For example:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body xmlns:ci="urn:iwaysoftware:adapter:peoplesoft:location:ci:request">
<ci:PS8>
  <ci:component perform="browse">LOCATION</ci:component>
  <ci:key name="SETID">SHARE</ci:key>
  <ci:key name="LOCATION">ALBERTA</ci:key>
</ci:PS8>
</soap:Body>
</soap:Envelope>
```

9. Click **Test Web Service**.

The output response is received in the Response tab of the Oracle Enterprise Manager console.



5.3 Configuring a Mediator Inbound Process

The following example describes how to configure a Mediator inbound process to your PeopleSoft system, using a Mediator project in Oracle JDeveloper.

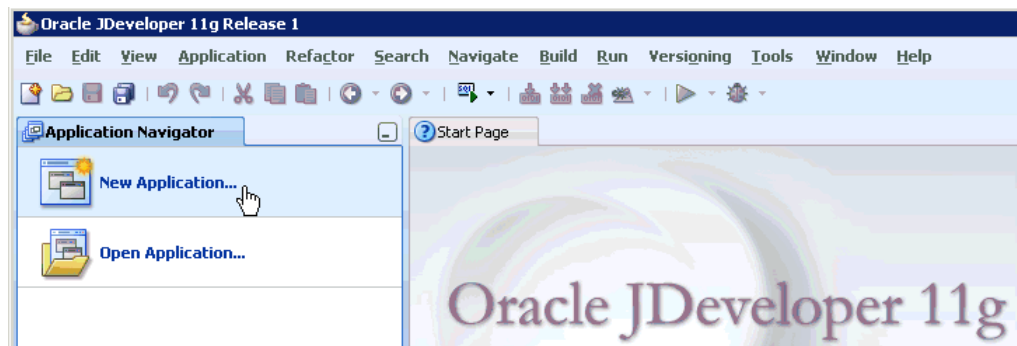
Prerequisites

Before you design a Mediator inbound process, you must generate the respective WSDL file using Application Explorer. See ["Generating WSDL for Event Integration"](#) on page 4-34 for more information.

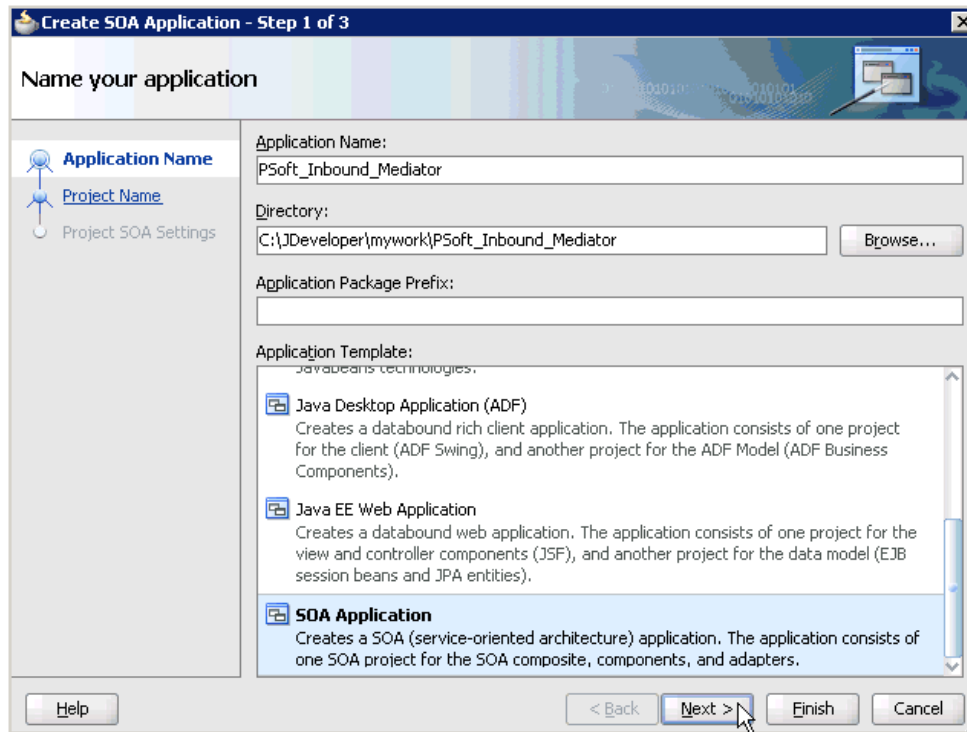
5.3.1 Creating a New SOA Application for the Inbound Mediator Process

Perform the following steps to create a new SOA application for the inbound Mediator process:

1. Open Oracle JDeveloper on your system.
2. In the Application Navigator tab, click **New Application**.

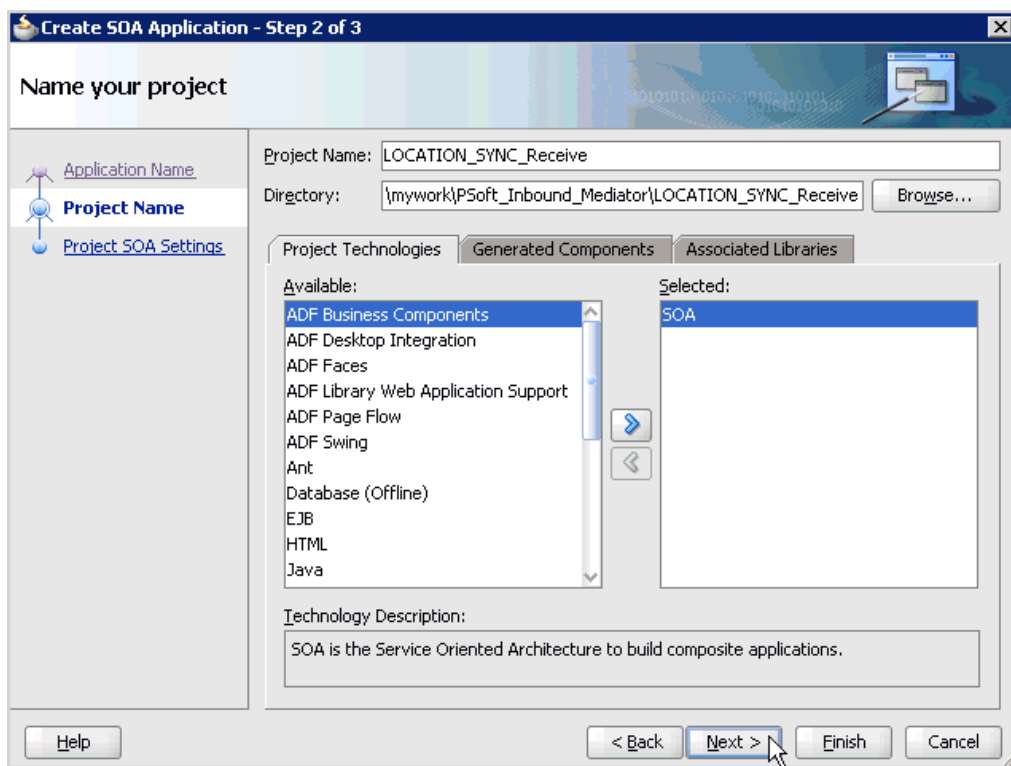


The Create SOA Application wizard is displayed.



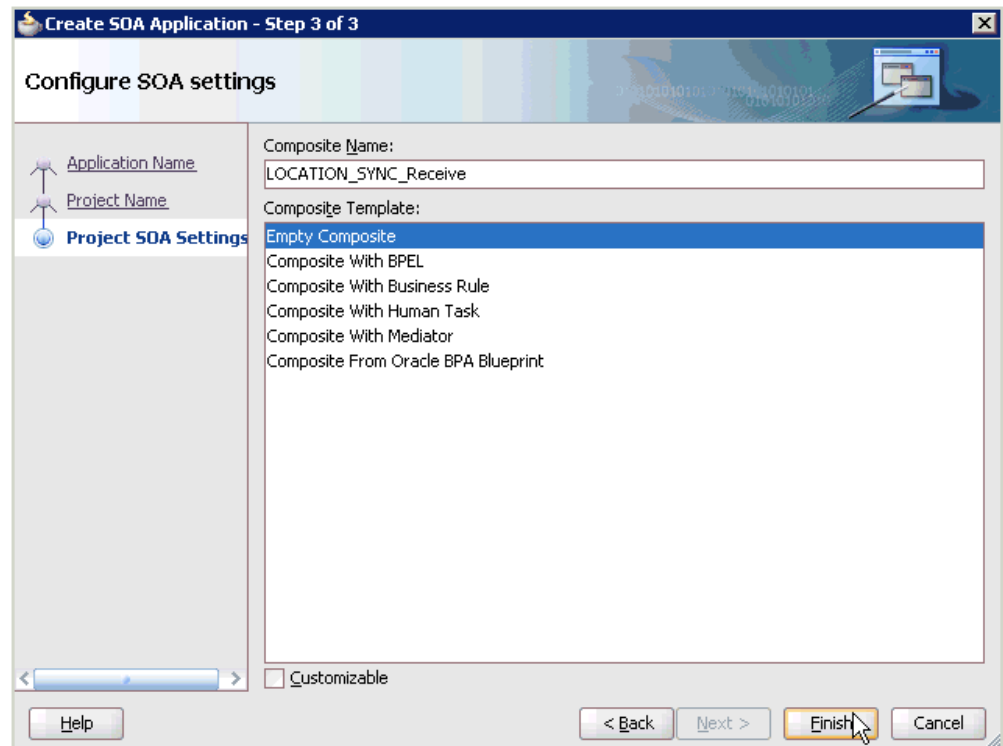
3. From the Application Template list, click **SOA Application**.
4. Enter name for the new SOA application (for example, PSoft_Inbound_Mediator) and click **Next**.

The Name your project page is displayed.

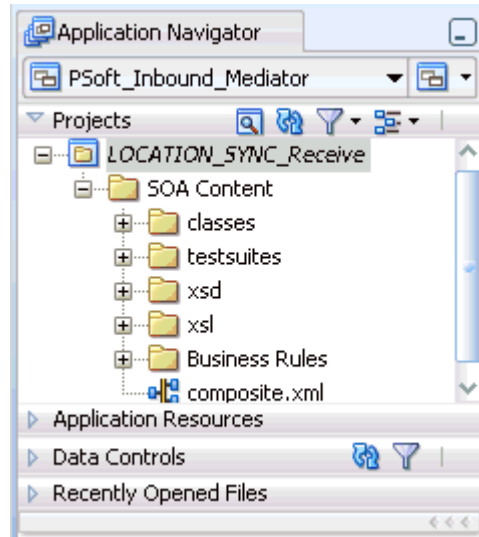


5. Enter a project name (for example, LOCATION_SYNC_Receive) and click **Next**.

The Configure SOA settings page is displayed.



- From the Composite Template list, select **Empty Composite** and click **Finish**.



The new SOA application (PSoft_Inbound_Mediator) and associated project (LOCATION_SYNC_Receive) are added to the Application Navigator tab in the left pane.

5.3.2 Defining a Mediator Inbound Process

This section describes how to define a Mediator inbound process, which consists of the following stages:

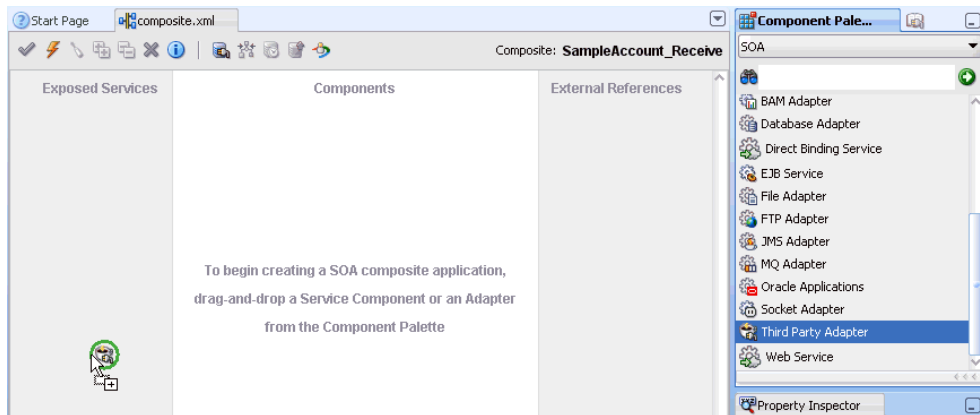
- Configuring a Third Party Adapter Service Component

2. Configuring an Inbound Mediator Process Component With a File Adapter
3. Configuring the Routing Rules

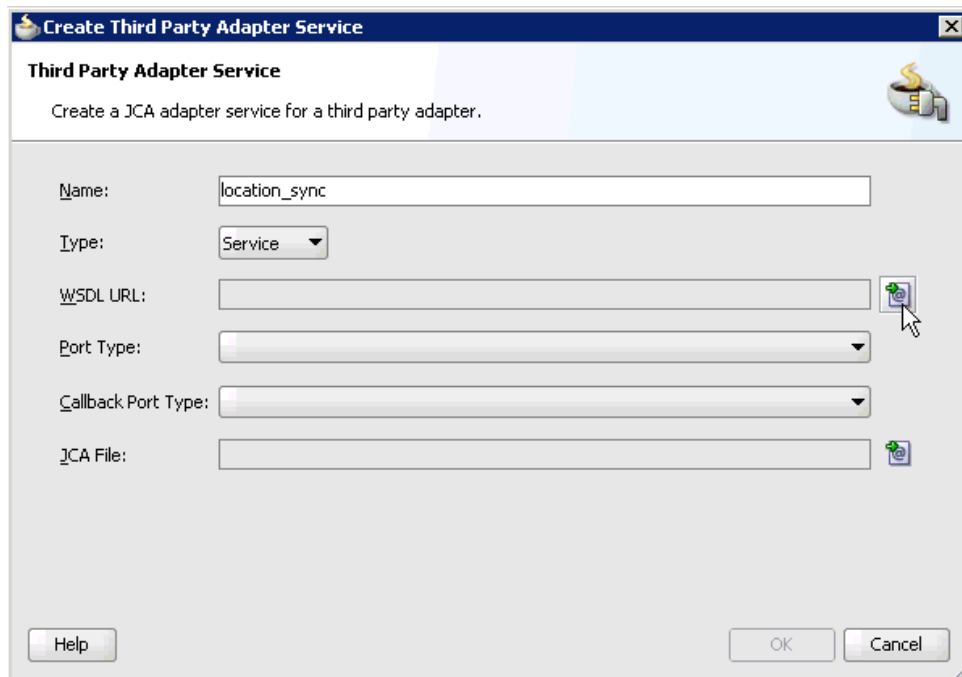
Configuring a Third Party Adapter Service Component

Perform the following steps to create a third party adapter service component:

1. Drag and drop the **Third Party Adapter** component from the Component Palette tab (Service Adapters section) to the Exposed Services pane.

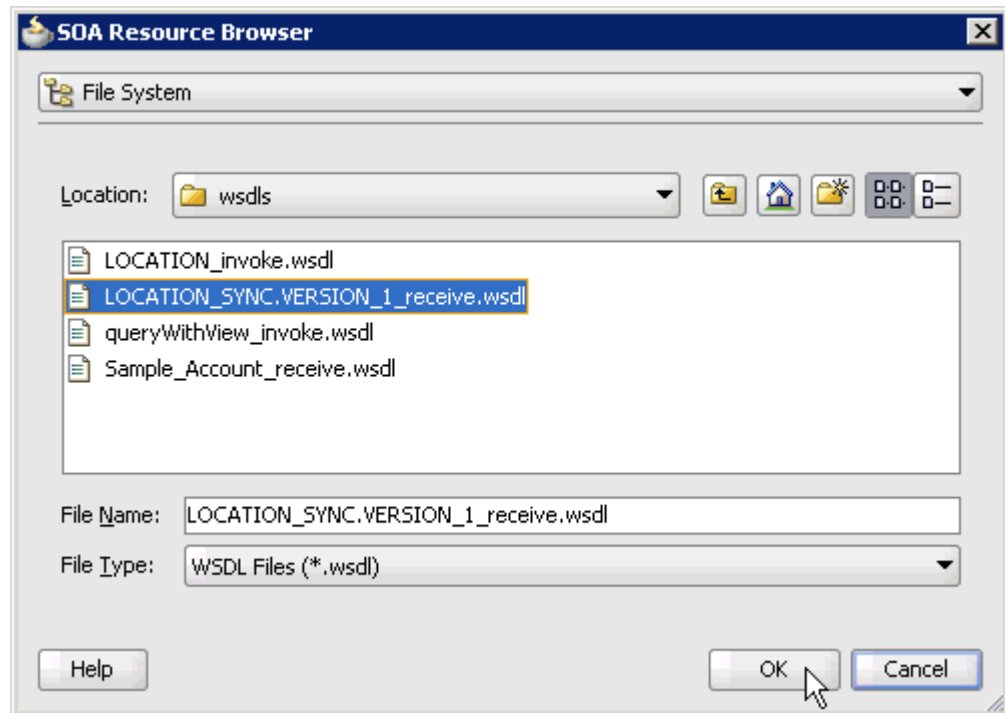


The Create Third Party Adapter Service dialog is displayed.



2. Enter a name for the third party adapter service.
3. Ensure that **Service** is selected from the Type list (default).
4. Click the **Find existing WSDLs** icon, which is located to the right of the WSDL URL field.

The SOA Resource Browser dialog is displayed.

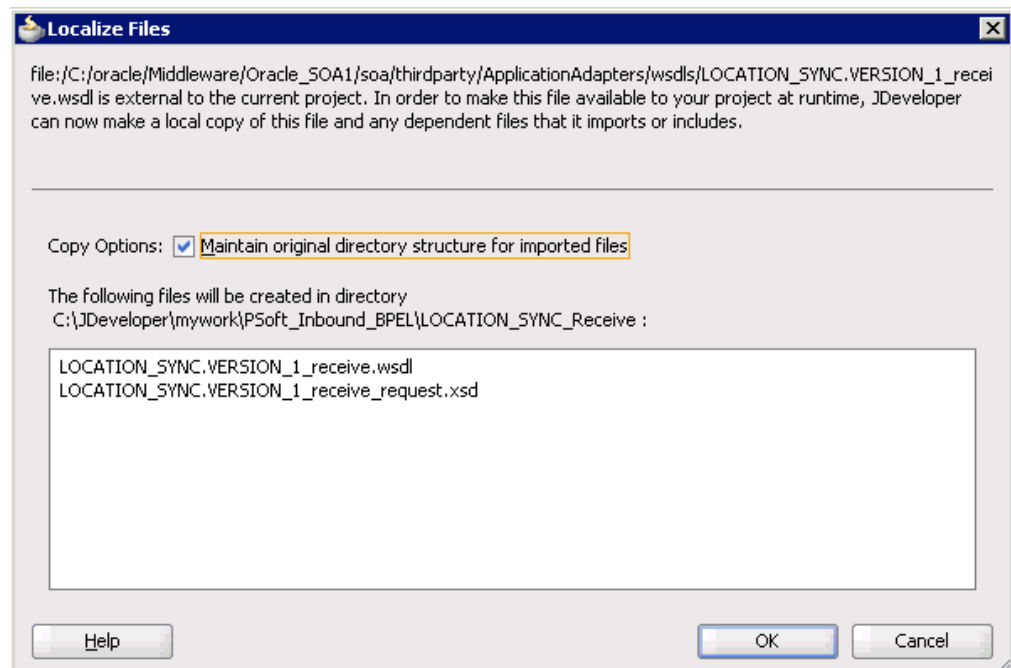


5. Browse and select an inbound WSDL file (for example, LOCATION_SYNC.VERSION_1_receive.wsd) from the following directory:

C:\oracle\Middleware\home_GA\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\wsdls

6. Click **OK**.

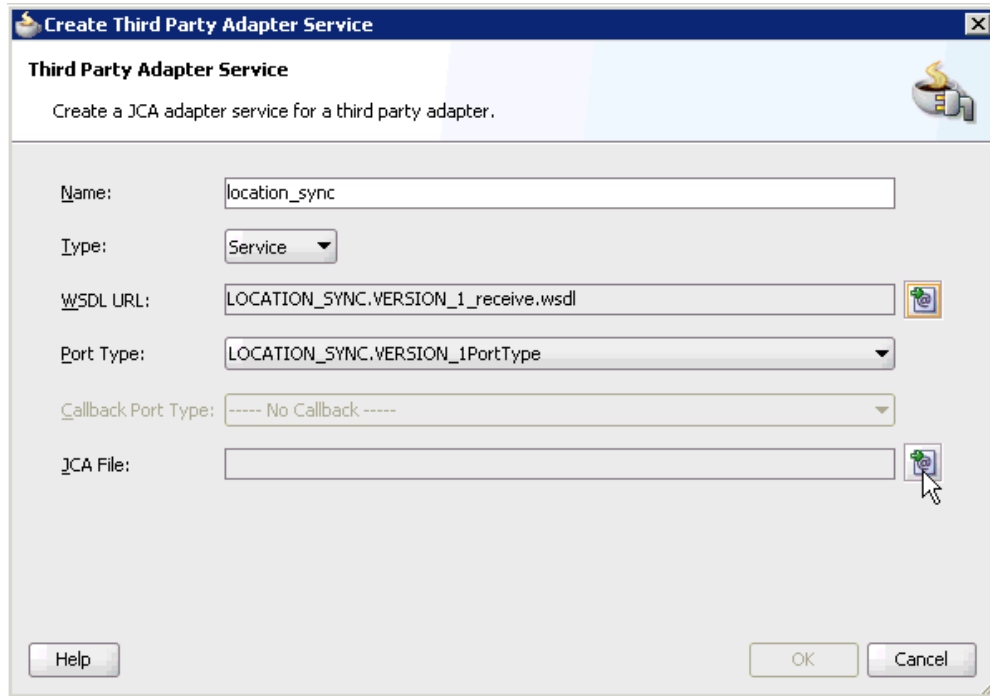
The Localize Files dialog is displayed.



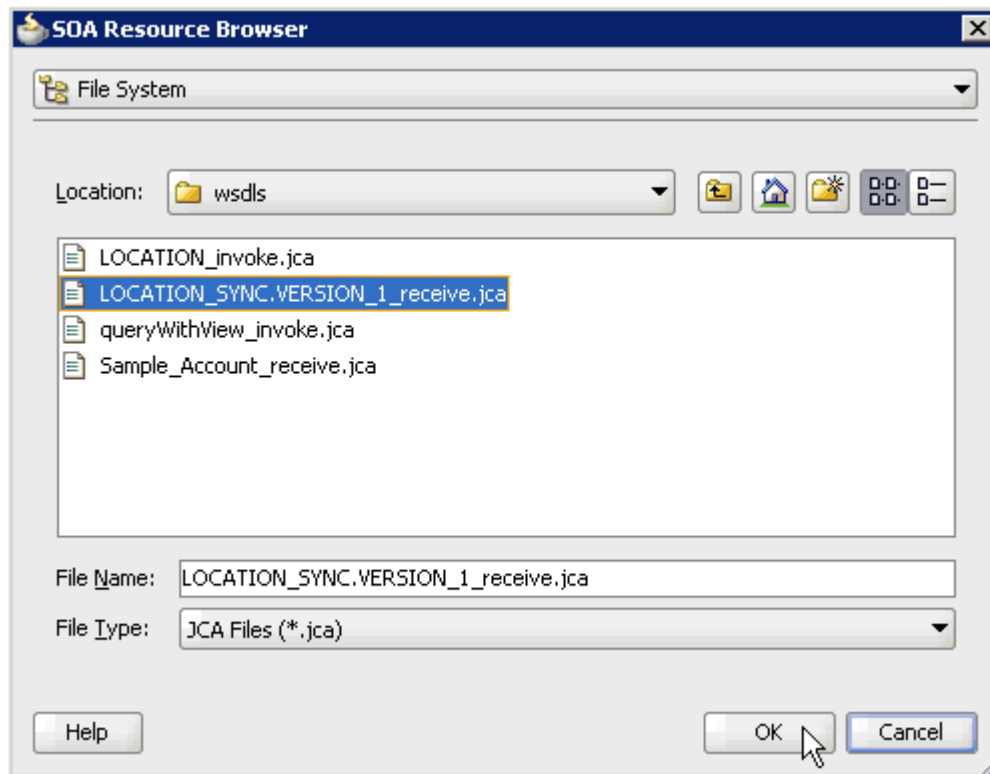
7. Click **OK**.

The inbound WSDL file and associated receive/request XML schema file (.xsd) are imported to the project folder that has been created.

You are returned to the Create Third Party Adapter Service dialog.



8. Click the **Find JCA Files** icon, which is located to the right of the JCA File field. The SOA Resource Browser dialog is displayed.

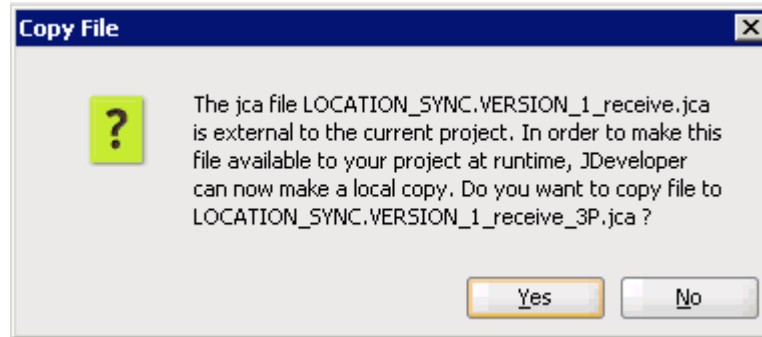


9. Browse and select the JCA properties file (for example, LOCATION_SYNC.VERSION_1_receive.jca) from the following directory:

```
C:\oracle\Middleware\home_GA\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\wsdls
```

10. Click **OK**.

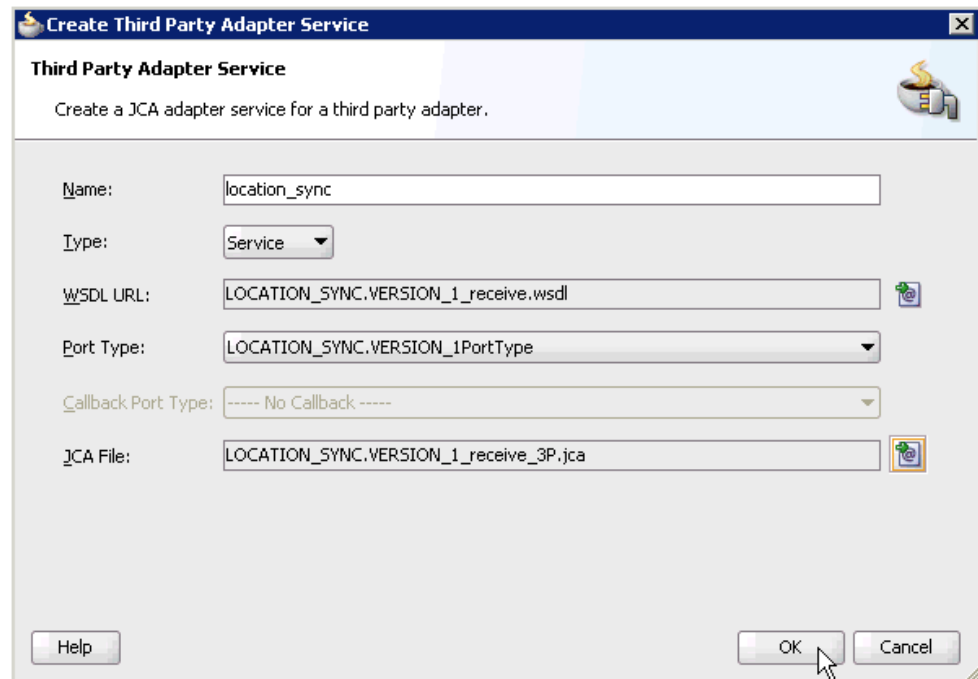
The following message is displayed.



11. Click **Yes**.

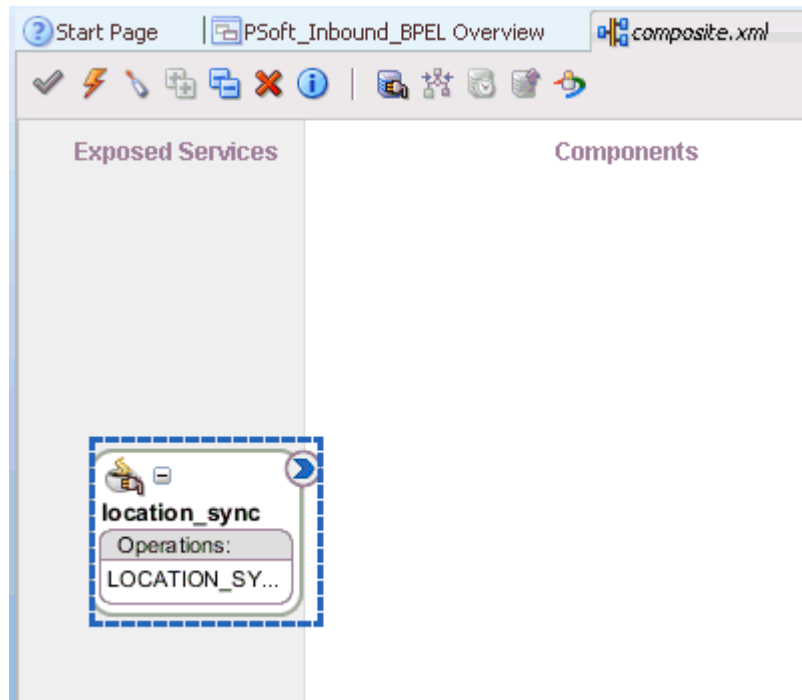
A copy of the JCA properties file is made in the project folder.

You are returned to the Create Third Party Adapter Service dialog.



12. Click **OK**.

The third party adapter service component (location_sync) is created in the Exposed Services pane, as shown in the following image.

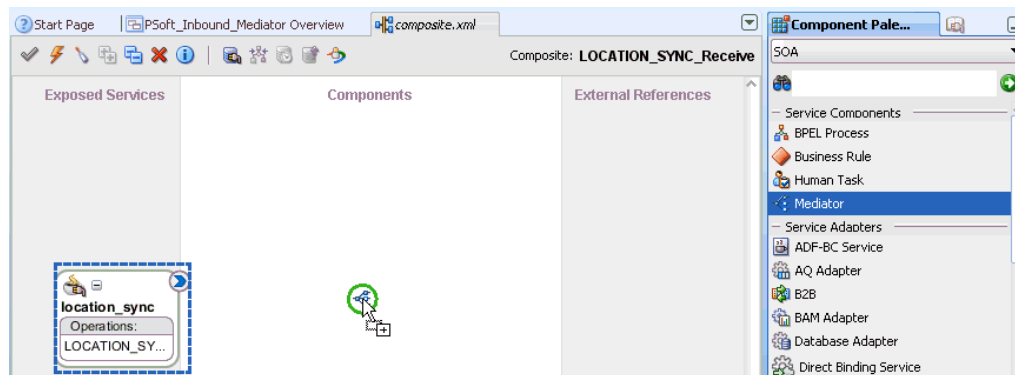


You are now ready to configure an inbound Mediator process component.

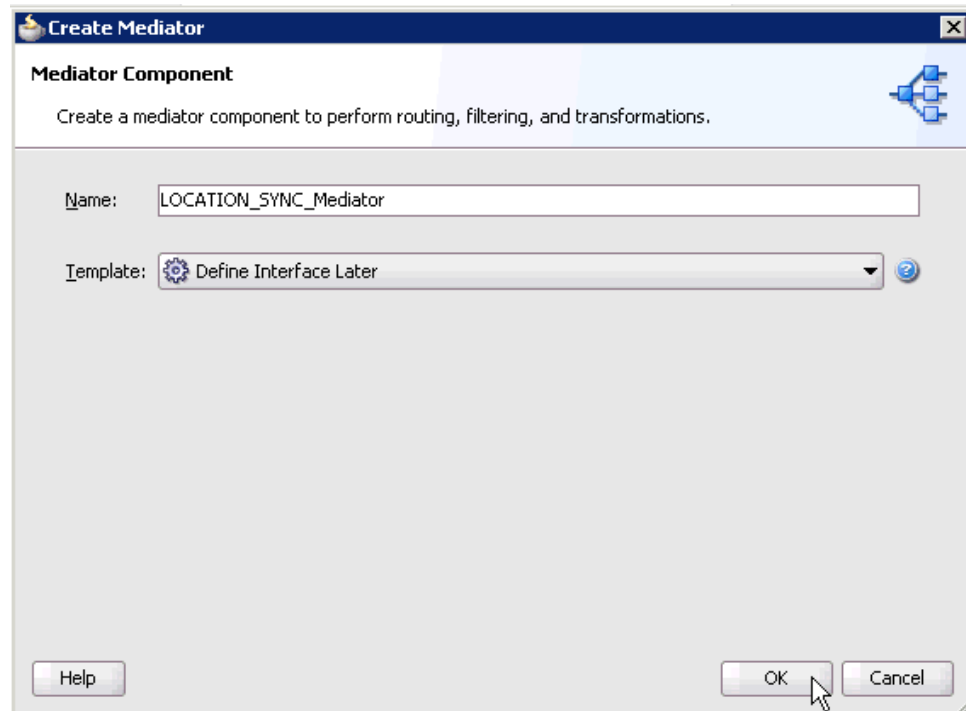
Configuring an Inbound Mediator Process Component With a File Adapter

Perform the following steps to configure an inbound Mediator process component with a File adapter.

1. Drag and drop the **Mediator** component from the Component Palette tab (Service Components section) to the Components pane.

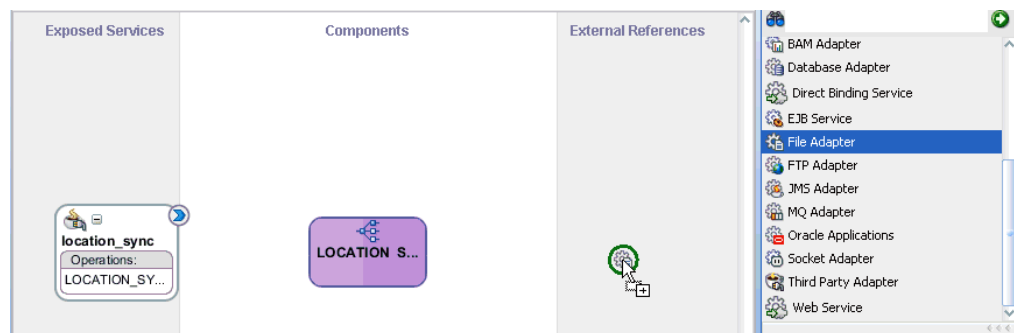


The Create Mediator dialog is displayed.



2. In the Name field, enter a name to identify the new inbound Mediator process component (for example, LOCATION_SYNC_Mediator).
3. From the Template list, select **Define Interface Later**.
4. Click the **OK**.

The new Mediator process component is added to the Components pane, as shown in the following image.



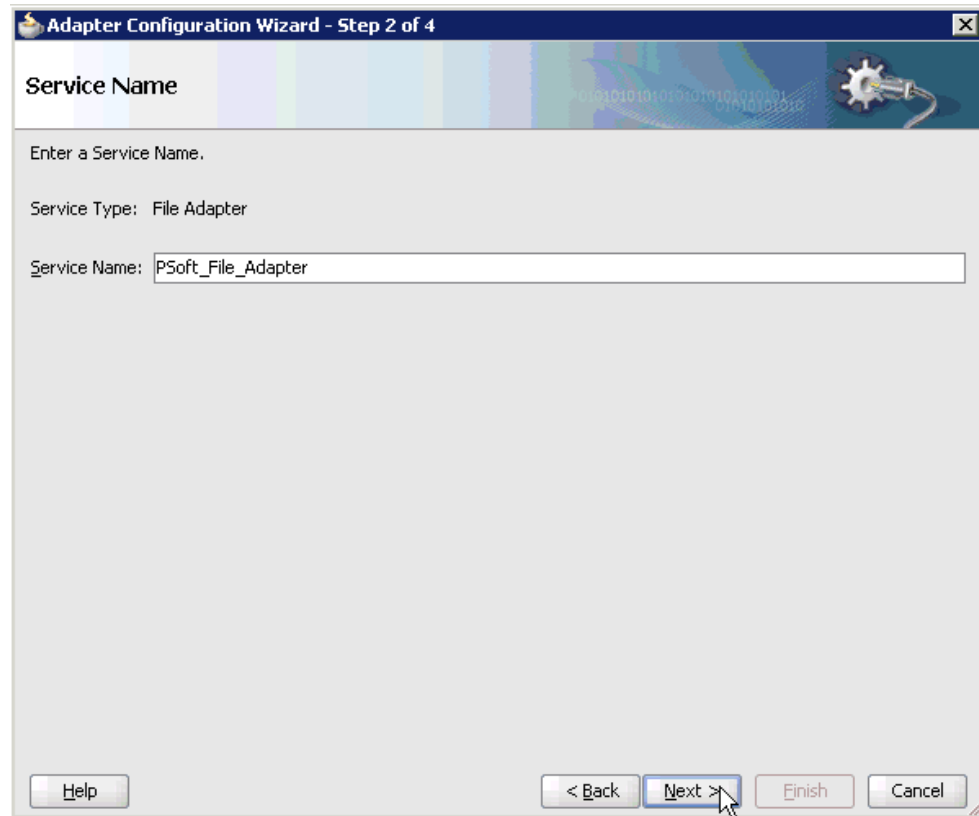
5. Drag and drop the **File Adapter** component from the Service Adapters pane to the External References pane.

The Adapter Configuration Wizard is displayed, showing the Welcome page.

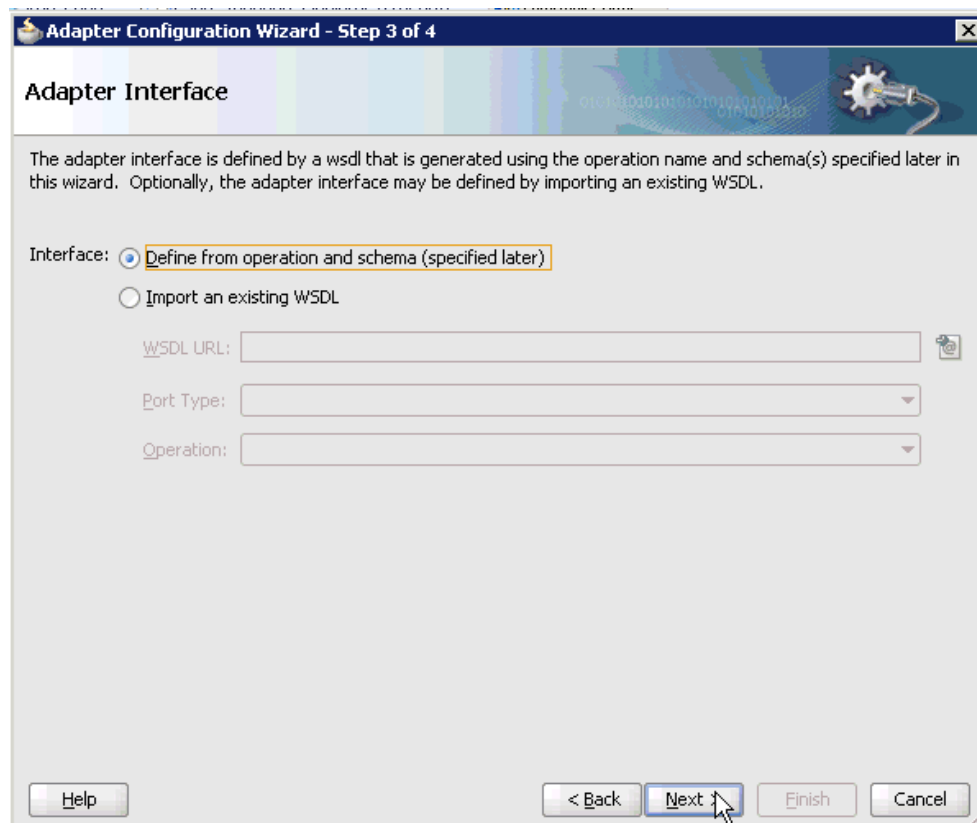


6. Click **Next**.

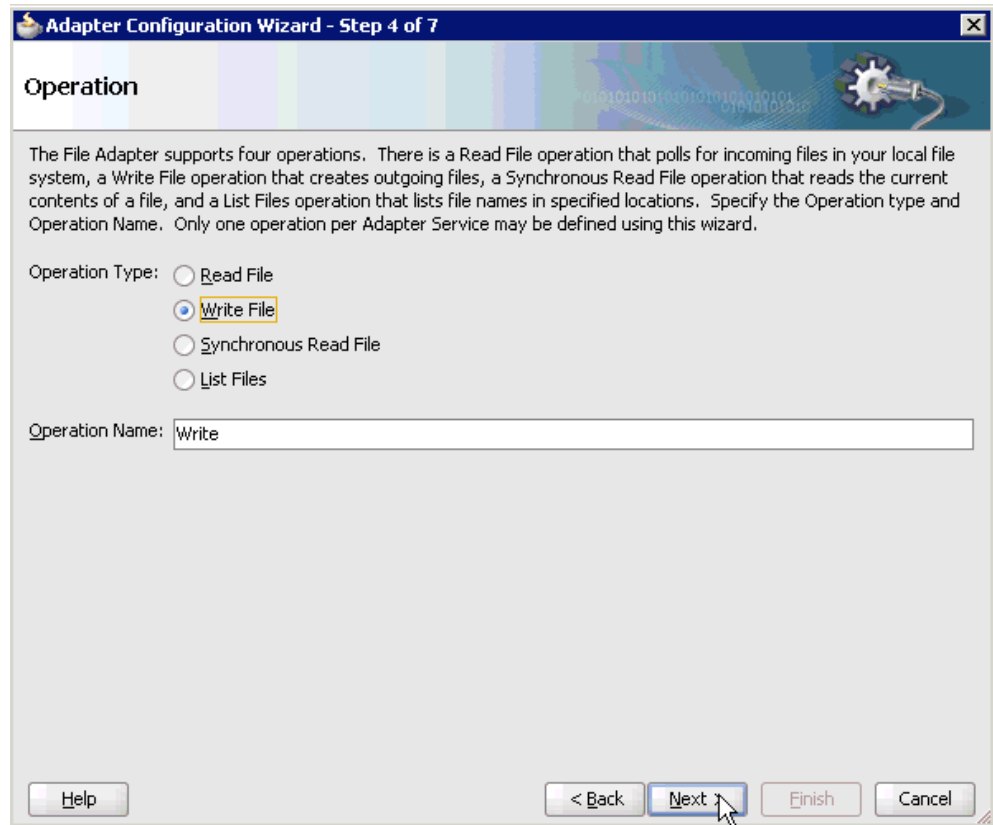
The Service Name page is displayed.



7. Type a name for the new File adapter in the Service Name field and click **Next**. The Adapter Interface page is displayed.

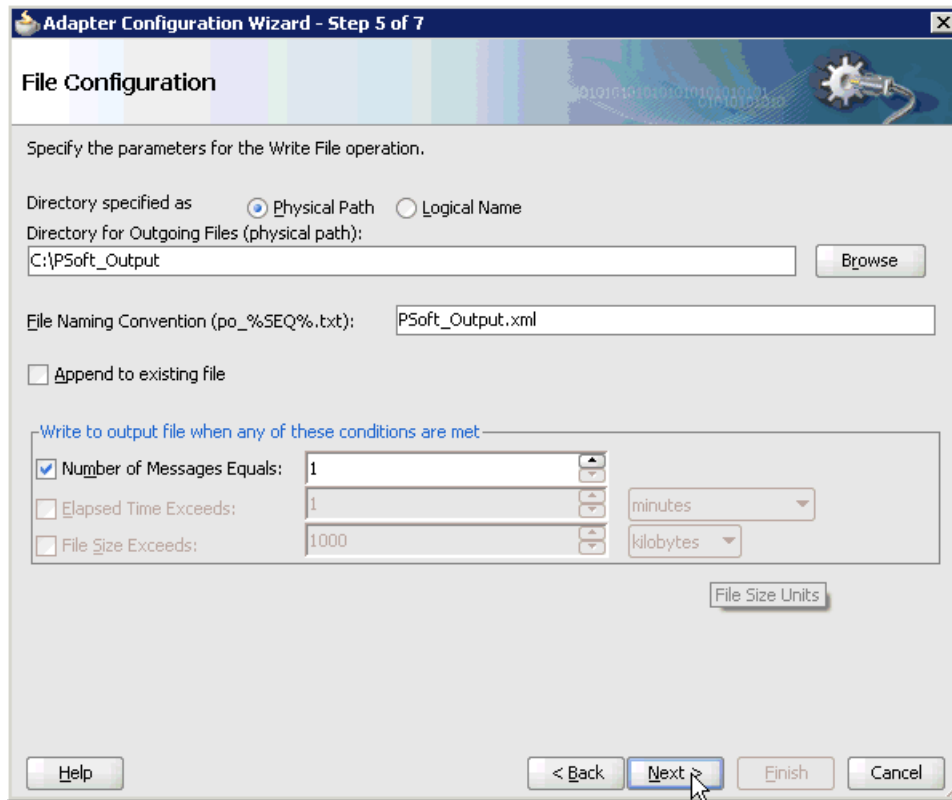


8. Ensure that the **Define from operation and schema (specified later)** option is selected.
9. Click **Next**.
The Operation page is displayed.



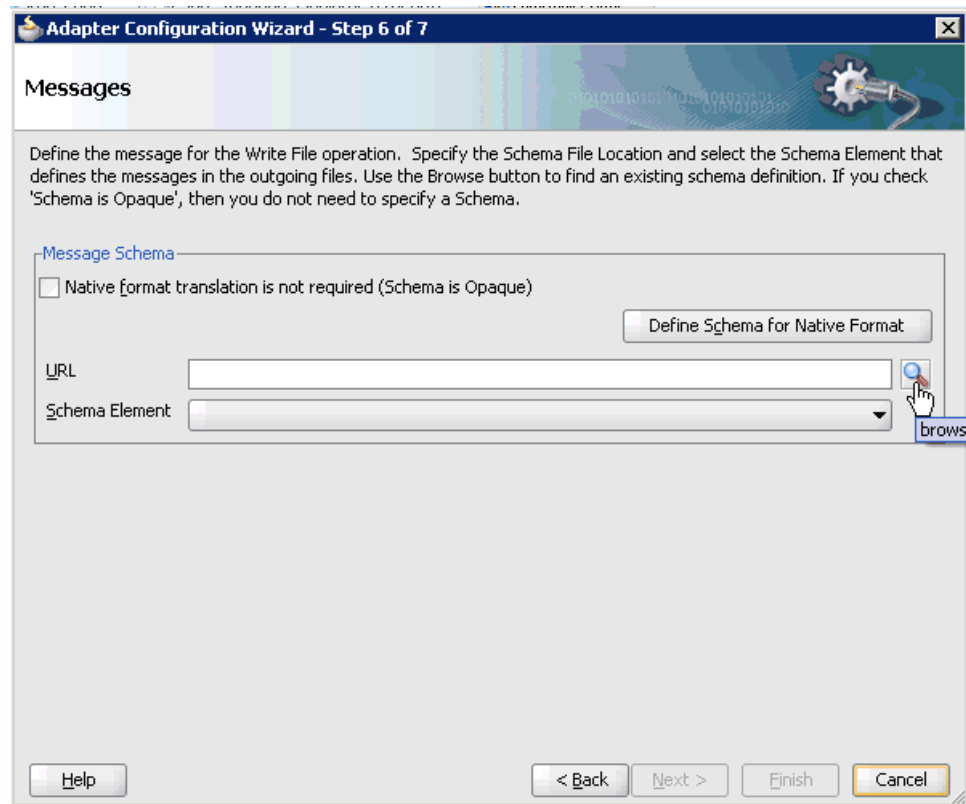
10. Select **Write File** from the list of Operation Type options and specify an Operation Name (for example, Write).
11. Click **Next**.

The File Configuration page is displayed.

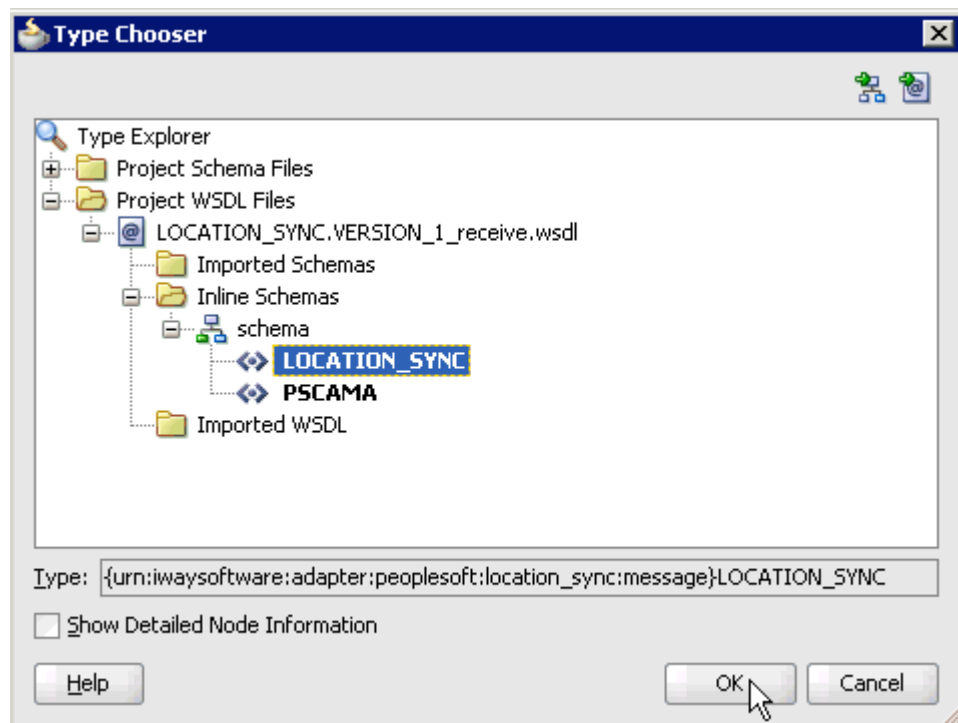


12. Specify a location on your file system where the output file is written.
13. In the File Naming Convention field, specify a name for the output file.
14. Click **Next**.

The Messages page is displayed.



15. Click **browse for schema file**, which is located to the right of the URL field. The Type Chooser dialog is displayed.

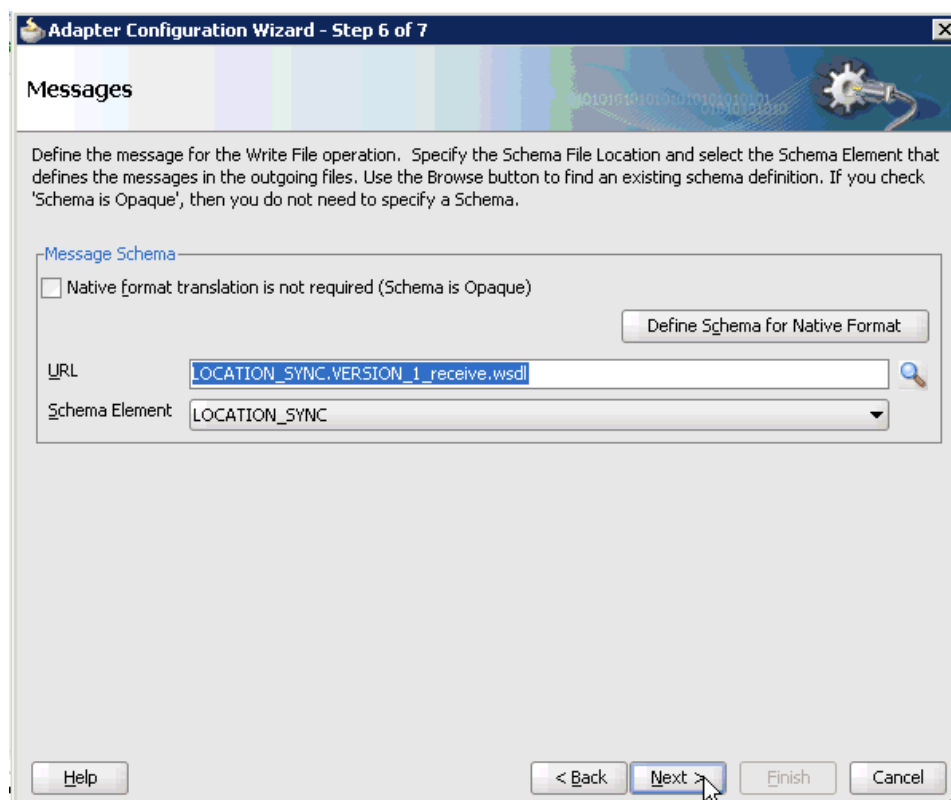


16. Expand **Project WSDL Files**, **LOCATION_SYNC.VERSION_1_receive.wSDL**, **Inline Schemas**, and **schema**.

17. Select the available schema (for example, LOCATION_SYNC)

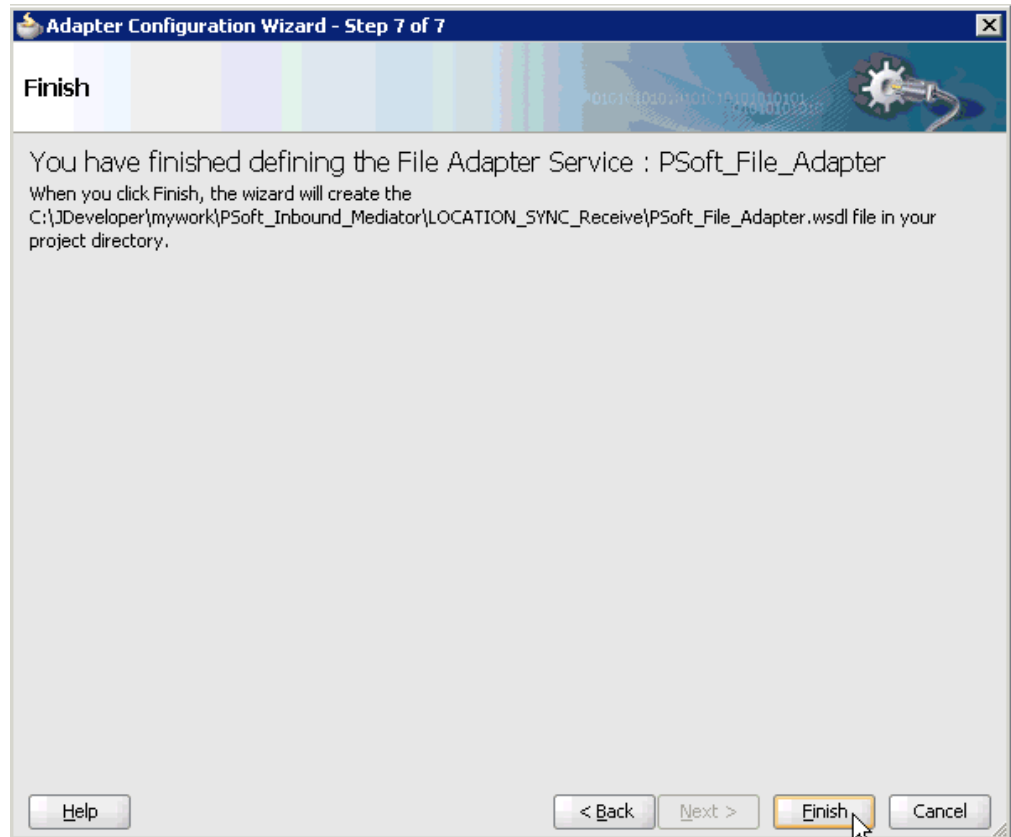
18. Click OK.

You are returned to the Messages page.



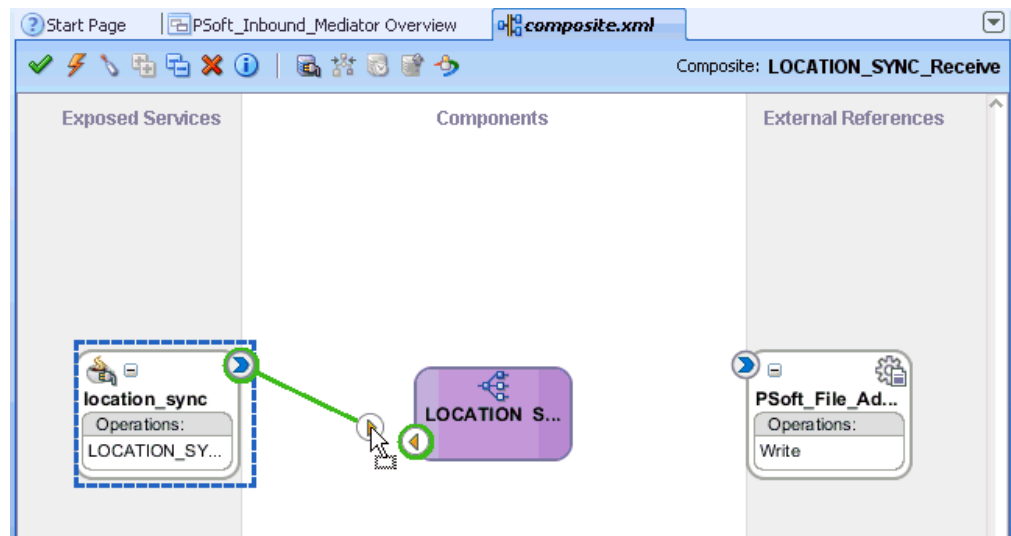
19. Click Next.

The Finish page is displayed.

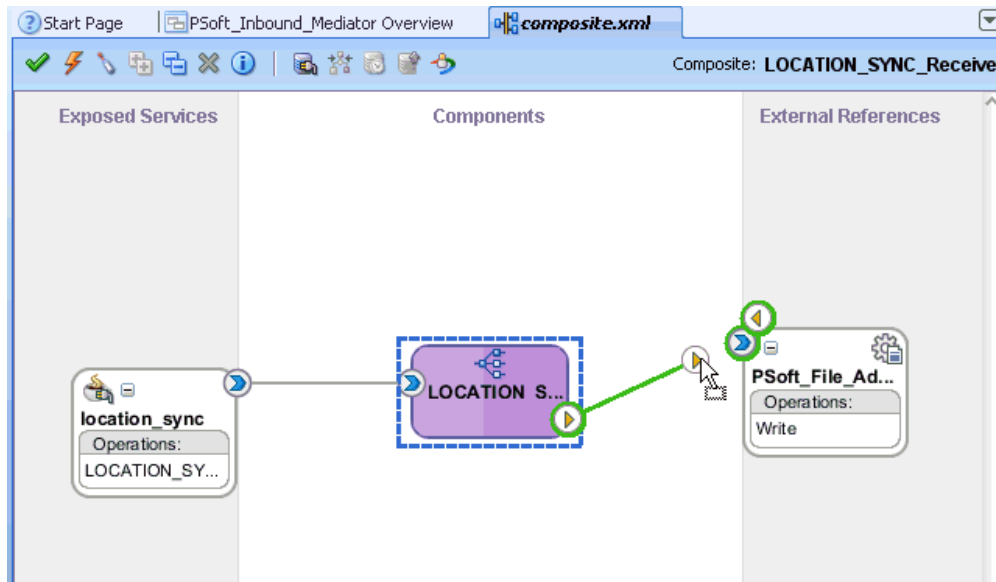


20. Click **Finish**.

21. Create a connection between the inbound Mediator process component and the third party adapter service component.



22. Create a connection between the inbound Mediator process component and the File adapter component.

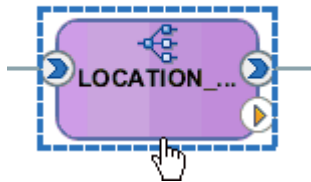


You are now ready to configure the routing rules.

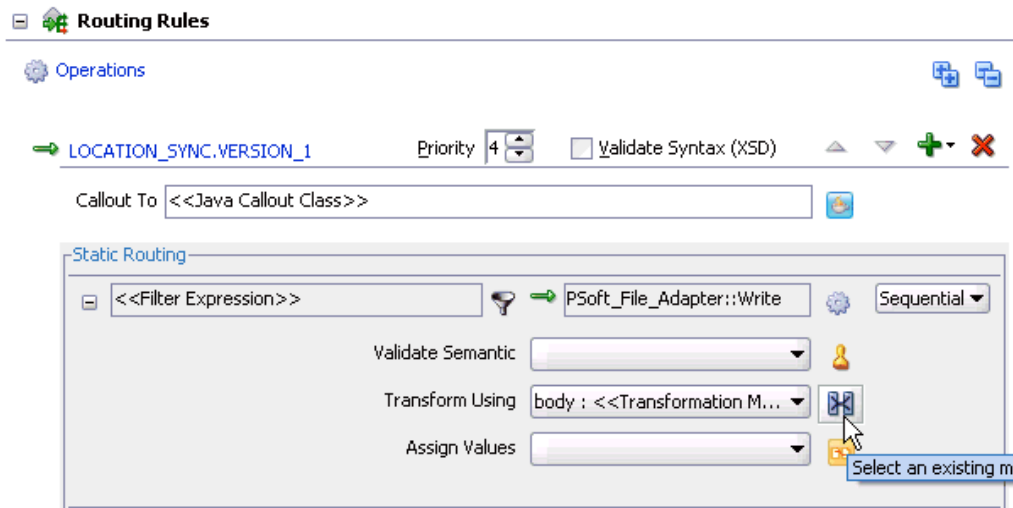
Configuring the Routing Rules

Perform the following steps to configure routing rules for the Mediator inbound process component:

1. Double-click the inbound Mediator process component in the Components pane.

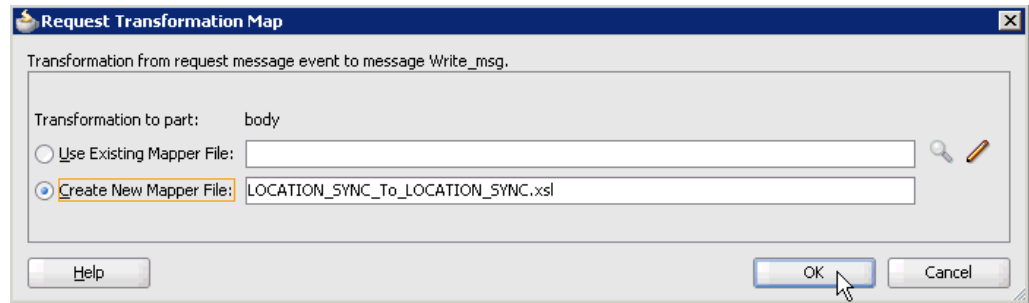


The Routing Rules dialog is displayed.



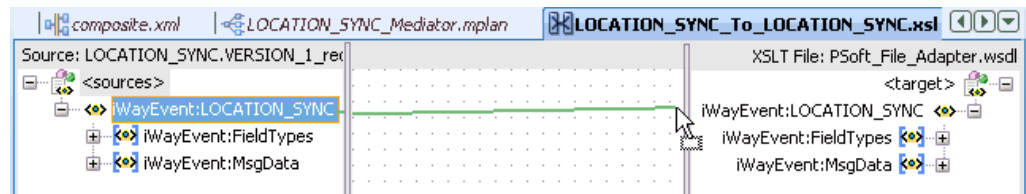
- In the <<Filter Expression>> area, click the **Select an existing mapper file or create a new one** icon to the right of the Transform Using field.

The Request Transformation Map dialog is displayed.



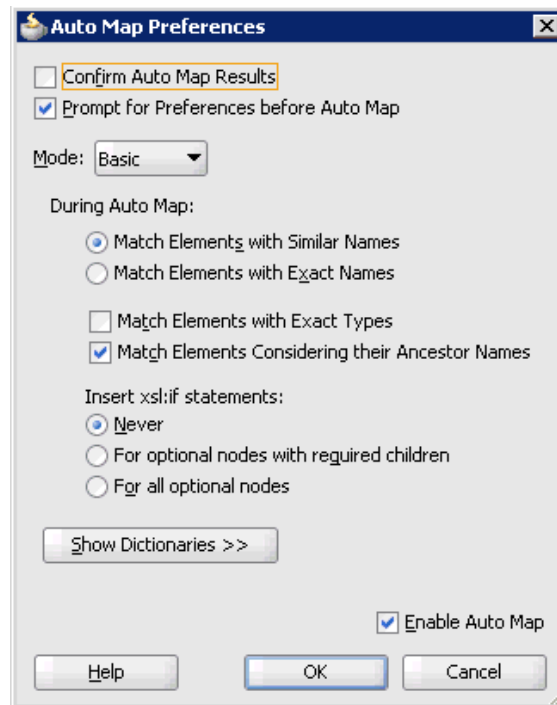
- Select the **Create New Mapper File** option and click **OK**.

The LOCATION_SYNC_To_LOCATION_SYNC.xsl tab is displayed.



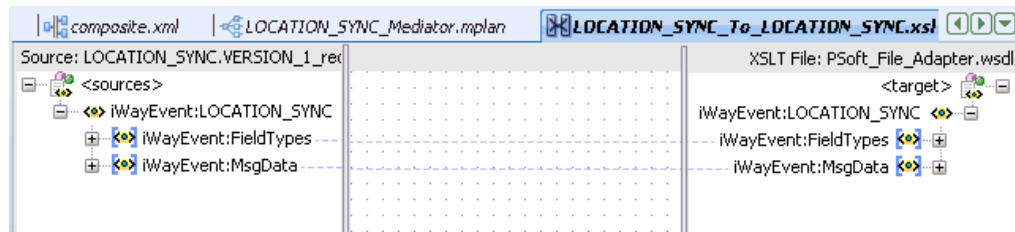
- Map the **iWayEvent:LOCATION_SYNC** source element to the **iWayEvent:LOCATION_SYNC** target element.

The Auto Map Preferences dialog is displayed.

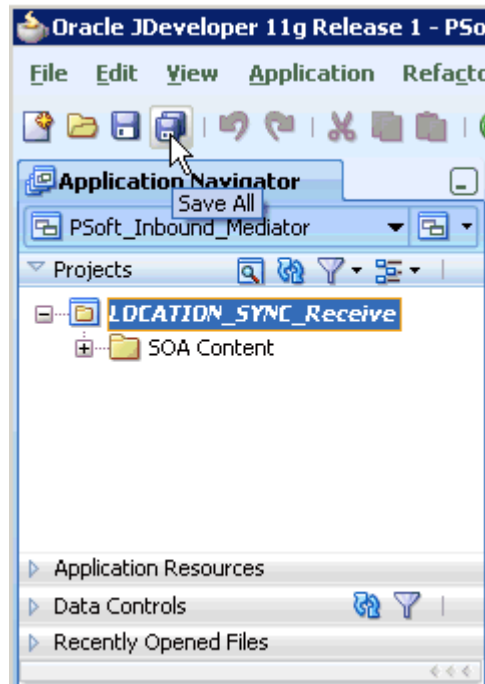


- Retain the default values and click **OK**.

The mapping is now complete.



6. Click the **Save All** icon in the menu bar to save the new inbound Mediator process component that was configured.



You are now ready to deploy the Mediator inbound process. You can follow the same procedure that is described in ["Deploying the BPEL Inbound Process"](#) on page 4-48.

Once event messages are triggered through PeopleSoft, output XML is received in the location that was specified for the File adapter component. For more information on triggering events in PeopleSoft, see ["Triggering an Event in PeopleSoft"](#) on page 4-51.

Troubleshooting and Error Messages

This chapter explains the limitations and workarounds when connecting to PeopleSoft. The following topics are discussed:

- [Troubleshooting](#)
- [BSE Error Messages](#)

The adapter-specific errors listed in this chapter can arise whether using the adapter with an Oracle Adapter J2CA or with an Oracle Adapter Business Services Engine (BSE) configuration.

6.1 Troubleshooting

This topic provides troubleshooting information for PeopleSoft, separated into four categories:

- Application Explorer
- PeopleSoft
- Oracle Adapter J2CA
- Oracle Adapter Business Services Engine (BSE)

Note: Log file information that can be relevant in troubleshooting can be found in the following locations:

- The Oracle Adapter J2CA trace information can be found under the following directory:

```
C:\oracle\Middleware\Oracle_  
SOA1\soa\thirdparty\ApplicationAdapters\config\config_name\log
```

- BSE trace information can be found under the following directory:

```
C:\oracle\Middleware\user_projects\domains\base_  
domain\servers\soa_server1\stage\ibse\ibse.war\ibselogs
```

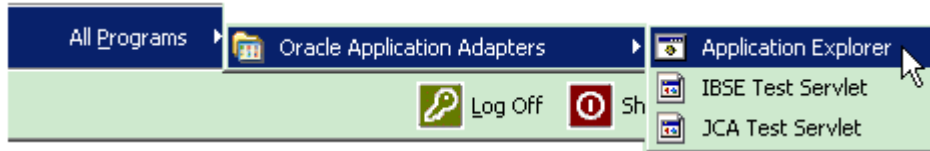
- The log file for Application Explorer can be found under the following directory:

```
C:\oracle\Middleware\Oracle_  
SOA1\soa\thirdparty\ApplicationAdapters\tools\iwae\bin
```

Application Explorer

To use Application Explorer on **Windows** for debugging or testing purposes:

1. Ensure that Oracle WebLogic Server is started, which is where Application Explorer is deployed.
2. Start Application Explorer by clicking the Windows **Start** menu, selecting **All Programs, Oracle Application Adapters**, and clicking **Application Explorer**.



You can also start Application Explorer by executing the **ae.bat** file, which is located in the following directory:

```
C:\oracle\Middleware\home_0309\Oracle_
SOA1\soa\thirdparty\ApplicationAdapters\tools\iwae\bin\ae.bat
```

It is a good practice to create a shortcut for the **ae.bat** file on your desktop.

If you are using a UNIX or Linux platform you can start Application Explorer by executing the **iwae.sh** file.

Error	Solution
<p>Cannot connect to Oracle Application Adapter for PeopleSoft from Application Explorer. The following error message appears:</p> <p>Problem activating adapter</p>	<p>Ensure that:</p> <ul style="list-style-type: none"> ■ PeopleSoft is running. ■ The PeopleSoft user ID and password are correct. ■ The port number is correct. ■ The custom component interface is properly installed.
<p>The following error message appears:</p> <p>java.lang.IllegalStateException: java.lang.Exception: Error Logon to PeopleSoft System</p>	<p>You have provided invalid connection information for PeopleSoft or the wrong <code>psjoe.jar</code> is in the <code>lib</code> directory.</p> <p>The <code>psjoe.jar</code> file version is specific to the PeopleTools release.</p>
<p>PeopleSoft does not appear in the Application Explorer Adapter node list.</p>	<p>Ensure that the PeopleSoft JAR files, <code>iwpsci84.jar</code> (or <code>iwpsci81.jar</code>) and <code>psjoe.jar</code>, are added to the <code>lib</code> directory.</p>
<p>Logon failure error at run-time.</p>	<p>If the password for connecting to your PeopleSoft system is not specified when creating a target or with the Edit option in Application Explorer, you will be unable to connect to PeopleSoft. The connection password is not saved in <code>repository.xml</code>. Update the password using the Edit option in Application Explorer, then restart the application server.</p>
<p>The following error message appears:</p> <p>Jolt Session Pool cannot provide a connection to the appserver. This appears to be because there is no available application server domain. [Fri Aug 27 13:06:27 EDT 2004] bea.jolt.ServiceException: Invalid Session</p>	<p>The host name or port number for PeopleSoft is incorrect.</p>

Error	Solution
Properties are not displayed for a component interface.	You are using the wrong <code>iwpsci8x.jar</code> file.
Cannot generate schemas.	If the error message “Index: -1, Size:0” appears, or if you can log on to Application Explorer but you cannot see any Component Interfaces or Messages, then you may have both the <code>iwpsci81.jar</code> and <code>iwpsci84.jar</code> files in your <code>lib</code> directory. Stop your server, remove the unrequired jar file, and restart the server.
The following exception occurs when you start Application Explorer by activating <code>ae.bat</code> (not <code>iaexplorer.exe</code>): <pre>java.lang.ClassNotFoundException: org.bouncycastle.jce.provider.Boun cyCastleProvider</pre>	This is a benign exception. It does not affect adapter functionality. Download BouncyCastle files from: ftp://ftp.bouncycastle.org/pub
Unable to start Application Explorer in a Solaris environment. The following exception is thrown in the console: <pre>javax.resource.ResourceException: IWAFFManagedConnectionFactory: License violation.at com.ibi.afjca.spi.IWAFFManagedConne ctionFactory.createConnectionFacto ry(IWAFFManagedConnectionFactory.ja va:98)at com.iwaysoftware.iwae.common.JCATr ansport.getConnectionFactory(JCATr ansport.java:133) at com.iwaysoftware.iwae.common.JCATr ansport.initJCA(JCATransport.java: 69)at com.iwaysoftware.iwae.common.JCATr ansport.<init>(JCATransport.java:6 2)at com.iwaysoftware.iwae.common.Adapt erClient.<init>(AdapterClient.java :85)at com.ibi.bse.ConfigWorker.run(Config Worker.java:41) at java.lang.Thread.run(Thread.java:5 34)</pre> <p>Could not create the connection factory.</p>	JAVACMD is not set on the user system. Before starting Application Explorer, export JAVACMD as follows: JAVACMD= <code>/<jdk_home>/bin/java</code> , where <code><jdk_home></code> is the directory where JDK is installed on your system.

PeopleSoft

Error	Solution
Services are not working properly when using the PeopleSoft Component Interface testing tool in three-tier mode.	To test properly using the Component Interface testing tool: <ol style="list-style-type: none"> 1. Open Application Designer. 2. Select the Component Interface. 3. Use the test tool. <p>If service works in test tool, then review the XML and check for redundant fields in XML.</p>

Error	Solution
<p>The following error message appears:</p> <pre>Jolt Session Pool cannot provide a connection to the appserver. This appears to be because there is no available application server domain. [Fri Aug 27 13:06:27 EDT 2004] bea.jolt.ServiceException: Invalid Session</pre>	<p>The host name or port number for PeopleSoft is incorrect.</p>
<p>Component Interfaces and Messages do not appear in the adapter tree.</p>	<p>The project is not installed properly on the PeopleSoft system.</p>
<p>Return error code -1 is received from PeopleSoft at run-time, for example:</p> <pre><LOCATIONProcessResponse xmlns="http://xmlns.oracle.com/LOCATION"> <error xmlns="">-1</error> </LOCATIONProcessResponse></pre>	<p>You are either using the incorrect version of <code>psjoe.jar</code>, or have both the <code>iwpsci81.jar</code> and <code>iwpsci84.jar</code> files in your <code>lib</code> directory. In the second case, you must delete the unused JAR file, and then restart the server.</p> <p>The <code>psjoe.jar</code> file version is specific to the PeopleTools release.</p>
<p>Pstools.properties file has not been initialized.</p>	<p>This file is required for PeopleSoft 8.1. If you are using PeopleSoft 8.1, you should add this file. If you are not using PeopleSoft 8.1 and this error message still appears, ignore the message.</p>
<p>The following error message appears:</p> <pre>Cannot find Component Interface {CI name}</pre>	<p>The reason may be either of the following:</p> <ul style="list-style-type: none"> ■ The Java API for the selected component interface is not found in the API JAR file. Please check the Java API for the class file for the CI. If not found, please add the class file for the CI. ■ The component interface name is mentioned incorrectly in the request document.
<p>The following error message appears:</p> <pre>Not Authorized (90,6) Failed to execute PSSession request</pre>	<p>The component interface does not have the necessary access to perform the operation. Change the permission settings in the PeopleSoft > Security > Permission list for the component interface.</p>
<p>The following error message appears:</p> <pre>Must also provide values for keys {keyname}</pre>	<p>The reason may be any of the following:</p> <ul style="list-style-type: none"> ■ The request XML document does not have the element for the mandatory key. Please include the keyname and the value in the request document. ■ The Key field name is mentioned incorrectly in the request document. ■ The Perform operation is mentioned incorrectly in the request XML document.

Oracle Adapter J2CA

Error	Solution
In Application Explorer, the following error message appears when you attempt to connect to an Oracle Adapter J2CA configuration: Could not initialize JCA	In the Details tab in the right pane, ensure that the directory specified in the Home field points to the correct directory, for example: C:\oracle\Middleware\Oracle_ SOA1\soa\thirdparty\ApplicationAdapters \tools\iwaeb\bin\..\..\..\

6.2 BSE Error Messages

This topic discusses the different types of errors that can occur when processing Web services through Oracle WebLogic Server Adapter Business Services Engine (BSE).

6.2.1 General Error Handling in BSE

BSE serves as both a SOAP gateway into the adapter framework and as the engine for some of the adapters. In both design time and run-time, various conditions can cause errors in BSE when Web services that use adapters run. Some of these conditions and resulting errors are exposed the same way, regardless of the specific adapter; others are exposed differently, based on the adapter being used. This topic explains what you can expect when you encounter some of the more common error conditions on an adapter-specific basis.

Usually, the SOAP gateway (**agent**) inside BSE passes a SOAP request message to the adapter required for the Web service. If an error occurs, how it is exposed depends on the adapter and the API or interfaces that the adapter uses. The APIs are generated from PeopleTools and are specific to the PeopleTools release. A few scenarios cause the SOAP gateway to generate a SOAP fault. In general, anytime the SOAP agent inside BSE receives an invalid SOAP request, a SOAP fault element is generated in the SOAP response. The SOAP fault element contains fault string and fault code elements. The fault code contains a description of the SOAP agent error.

The following SOAP response document results when BSE receives an invalid SOAP request:

```
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SOAP-ENV:Fault>
      <faultcode>SOAP-ENV:Client</faultcode>
      <faultstring>Parameter node is missing</faultstring>
    </SOAP-ENV:Fault>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

In this example, BSE did not receive an element in the SOAP request message that is mandatory for the WSDL for this Web service.

6.2.2 Adapter-Specific Error Handling

When an adapter raises an exception during run-time, the SOAP agent in BSE produces a SOAP fault element in the generated SOAP response. The SOAP fault element contains fault code and fault string elements. The fault string contains the native error description from the adapter target system. Since adapters use the target

system interfaces and APIs, whether an exception is raised depends on how the target systems interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in BSE, and that request is invalid based on the WSDL for that service, the adapter may raise an exception yielding a SOAP fault.

While it is almost impossible to anticipate every error condition that an adapter may encounter, the following is a description of how adapters handle common error conditions and how they are then exposed to the Web services consumer application.

Oracle Application Adapter for PeopleSoft Invalid SOAP Request

If the PeopleSoft agent receives a SOAP request message that does not conform to the WSDL for the Web service being executed, then the following SOAP response is generated.

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
  <SOAP-ENV:Body>
    <m:CARRIERResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
xmlns="urn:schemas-iwaysoftware-com:iwse"
cid="2A3CB42703EB20203F91951B89F3C5AF">
      <PS8>
        <error>Cannot find Component Interface {VARRIER} (91,2)Initialization
failed (90,7)Not Authorized (90,6)Failed to execute PSSession request Cannot
find Component Interface {VARRIER} (91,2)</error>
      </PS8>
    </m:CARRIERResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Empty Result From PeopleSoft Request

If Oracle Application Adapter for PeopleSoft executes a component interface as a Web service using input parameters passed in the SOAP request that do not match records in PeopleSoft, then the following SOAP response is generated.

```
<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
  <SOAP-ENV:Body>
    <m:CARRIERResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
xmlns="urn:schemas-iwaysoftware-com:iwse" cid="2A3CB42703EB20203F91951B89F3C5AF">
      <PS8>
        <error>No rows exist for the specified keys. {CARRIER} (91,50)Failed to
execute PSBusComp request</error>
      </PS8>
    </m:CARRIERResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Failure to Connect to PeopleSoft

If Oracle Application Adapter for PeopleSoft cannot connect to PeopleSoft, then the following SOAP response is generated.

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
```

```

    <faultcode>SOAP-ENV:Server</faultcode>
    <faultstring>java.lang.Exception: Error Logon to PeopleSoft
    System<faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Invalid SOAP Request

If Oracle WebLogic Server Application Adapter receives a SOAP request message that does not conform to the WSDL for the Web services being executed, then the following SOAP response is generated.

```

<?xml version="1.0" encoding="ISO-8859-1"
  ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
    <faultcode>SOAP-ENV:Server</faultcode>
    <faultstring>RPC server connection failed: Connection refused:
connect</faultstring>
  </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Empty Result From Oracle WebLogic Server Application Adapter Request

If the adapter executes a SOAP request using input parameters passed that do not match records in the target system, then the following SOAP response is generated.

Note: The condition for this adapter does not yield a SOAP fault.

```

<SOAP-ENV:Envelope xmlns:xsi="http://www.w3.org/1999/XMLSchema-instance"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="http://www.w3.org/1999/XMLSchema">
  <SOAP-ENV:Body>
    <m:RunDBQueryResponse xmlns:m="urn:schemas-iwaysoftware-com:iwse"
      xmlns="urn:schemas-iwaysoftware-com:iwse"
      cid="2A3CB42703EB20203F91951B89F3C5AF">
      <RunDBQueryResult run="1" />
    </m:RunDBQueryResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

Advanced User Tools

This chapter includes the following topics:

- [Web Services Policy-Based Security](#)
- [Migrating Repositories](#)

7.1 Web Services Policy-Based Security

Oracle Adapter Application Explorer (Application Explorer) provides a security model called Web services policy-based security. The following topics describe how the feature works and how to configure it.

Web services provide a layer of abstraction between the back-end business logic and the user or application running the Web service. Easy application integration is enabled. However, the issue of controlling the use and implementation of critical and sensitive business logic that is run as a Web service is raised.

Application Explorer controls the use of Web services that use adapters, using a feature called policy-based security. This feature enables an administrator to apply policies to business services (Web services) to deny or permit their execution.

A policy is a set of privileges dealing with the execution of a business service that can be applied to an existing or new business service. When you set specific rights or privileges inside a policy, you do not have to re-create privileges for every business service that has security concerns in common with other business services. Instead, you reuse a policy on multiple business services.

The goal of the feature is to secure requests at both the transport and the SOAP request level transmitted on the wire. Some of the policies do not deal with security issues directly, but do affect the run-time behavior of the Web services to which they have been applied.

The BSE administrator creates an "instance" of a policy type, names it, associates individual users or groups (a collection of users), and then applies that policy to one or more business services.

You can assign a policy to a business service, or to a method within a business service. If a policy is only applied to a method, other methods in that business service will not be governed by it. However, if a policy is applied to the business service, all methods are governed by it. At run-time, the user ID and password that are sent to BSE in the SOAP request message are checked against the list of users for all policies applied to that specific business service. The policy type that is supported is Resource Execution, which dictates who can or cannot execute the business service.

When a policy is not applied, the default value for a business service is to "grant all". For example, anybody can execute the business service, until the Resource Execution

policy is associated to the business service. At that time, only those granted execution permissions, or users not part of the group that has been denied execution permissions, have access to the business service.

7.1.1 Configuring Web Services Policy-Based Security

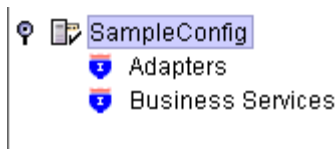
The following procedures describe how to configure Web services policy-based security.

Creating and Associating a User with a Policy

Before you create instances of policies, you must have a minimum of one user or one group to associate to an instance. You can create users and groups using Application Explorer.

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#) for information on creating a new configuration.
3. Select **Connect**.

Nodes appear for Adapters and Business Services (also known as Web services).

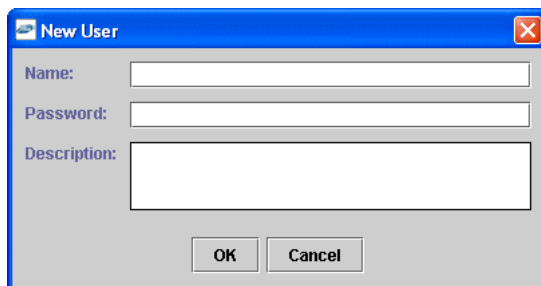


- a. Expand the **Business Services** node.
- b. Expand the **Configuration** node.
- c. Expand the **Security** node.
- d. Expand the **Users and Groups** node.



4. Right-click **Users** and click **New User**.

The New User dialog is displayed.



- a. In the **Name** field, enter a user ID.
- b. In the **Password** field, enter the password associated with the user ID.
- c. In the **Description** field, enter a description of the user (optional).

5. Click **OK**.



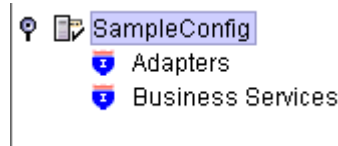
The new user is added under the Users node.

Creating a Group to Use With a Policy

To create a group to use with a policy:

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#) for information on creating a new configuration.
3. Select **Connect**.

Nodes appear for Adapters and Business Services (also known as Web services).

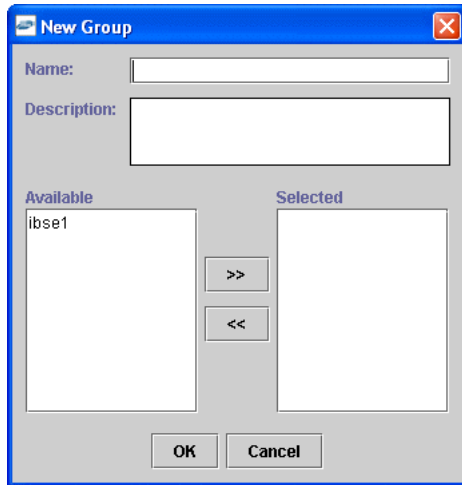


- a. Expand the **Business Services** node.
- b. Expand the **Configuration** node.
- c. Expand the **Security** node.
- d. Expand the **Users and Groups** node.



4. Right-click **Groups** and select **New Group**.

The New Group dialog is displayed.



- a. In the **Name** field, enter a name for the group.
 - b. In the **Description** field, enter a description for the group (optional).
 - c. From the available list of users in the left pane, select one or more users and add them to the **Selected** list by clicking the double right-facing arrow.
5. When you have selected at least one user, click **OK**.

The following shows the new group added under the **Groups** node.



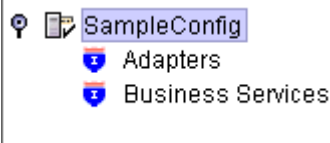
Creating an Execution Policy

An execution policy governs who can execute the business services to which the policy is applied.

To create an execution policy:

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#) for information on creating a new configuration.
3. Select **Connect**.

Nodes appear for Adapters and Business Services (also known as Web services).

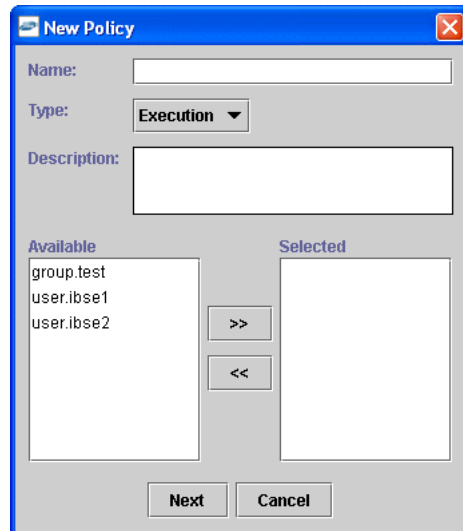


- a. Expand the **Business Services** node.
- b. Expand the **Configuration** node.
- c. Expand the **Security** node.
- d. Expand the **Policies** node.



4. Right-click **Policies** and select **New Policy**.

The New policy dialog is displayed.



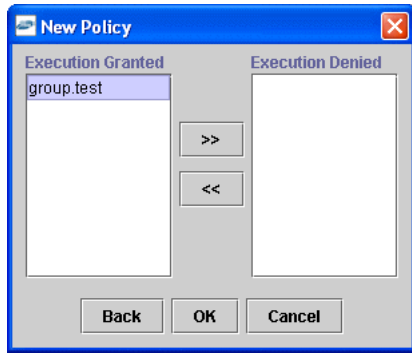
Provide the following information:

- a. In the **Name** field, enter a name for the policy.
- b. From the **Type** list, select **Execution**.
- c. In the **Description** field, enter a description for the policy (optional).
- d. From the available list of users in the left pane, select one or more users and add them to the **Selected** list by clicking the double right-facing arrow.

Note: This user ID is checked against the value in the user ID element of the SOAP header sent to BSE in a SOAP request.

5. When you have selected at least one user, click **OK**.
6. Click **Next**.

The New Policy permissions dialog is displayed.



7. To grant permission to a user or group to execute a business service, select the user or group and move them into the **Execution Granted** list by selecting the double left-facing arrow.
8. To deny permission to a user or group to execute a business service, select the user or group and move them into the Execution Denied list by selecting the double right-facing arrow.
9. Click **OK**.

The following pane summarizes your configuration.

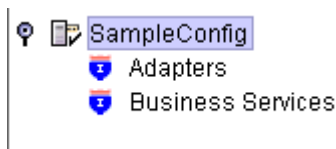
- **Name** test
- **Type** Execution
- **Description**
- **User and Group Restrictions**
 - group.test Execution Granted

Using the IP and Domain Restrictions Policy Type

You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to BSE and therefore need not be applied to individual Web services. You need not create a policy; however, you must enable the Security Policy option in Application Explorer.

1. Start Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See [Chapter 2, "Configuring Oracle Application Adapter for PeopleSoft"](#) for information on creating a new configuration.
3. Select **Connect**.

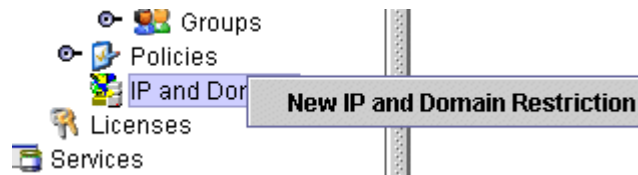
Nodes appear for Adapters and Business Services (also known as Web services).



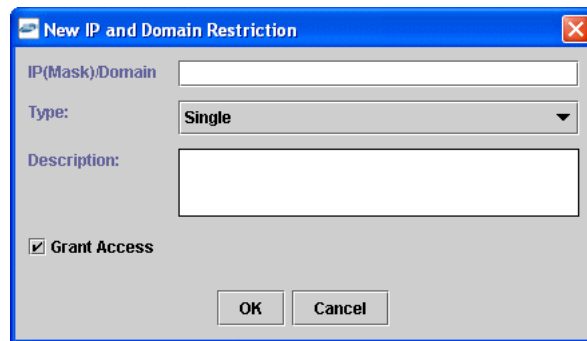
Perform the following steps:

- a. Expand the **Business Services** node.

- b. Expand the **Configuration** node.
 - c. Expand the **Security** node.
4. Right-click **IP and Domain** and select **New IP and Domain Restriction**.



The New IP and Domain Restriction dialog is displayed.



Perform the following steps:

- a. In the **IP(Mask)/Domain** field, enter the IP or domain name using the following guidelines.
 - If you select **Single** (Computer) from the **Type** list, you must provide the IP address for that computer. If you only know the DNS name for the computer, click **DNS Lookup** to obtain the IP Address based on the DNS name.
 - If you select **Group** (of Computers), you must provide the IP address and subnet mask for the computer group.
 - If you select **Domain**, you must provide the domain name.
 - b. From the **Type** list, select the type of restriction.
 - c. In the **Description** field, enter a description (optional).
 - d. To grant access, select the **Grant Access** check box.
5. Click **OK**.

The new domain is added under the IP and Domain node.

The following pane summarizes your configuration.

- **IP Address (Mask) /Domain** www.yahoo.com
- **Type** Domain
- **Access** Denied
- **Description**

7.2 Migrating Repositories

During design time, the Oracle repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. The information in the repository is also referenced at run-time. For management purposes, you can migrate BSE and J2CA repositories that are configured for Oracle to new destinations without affecting your existing configuration. For example, you may want to migrate a repository from a test environment to a production environment.

Migrating a BSE Repository

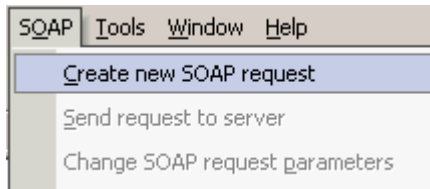
To migrate a BSE repository:

1. Copy the BSE control service URL, for example:

```
http://localhost:8001/ibse/IBSEServlet/admin/iwcontrol.ibs
```

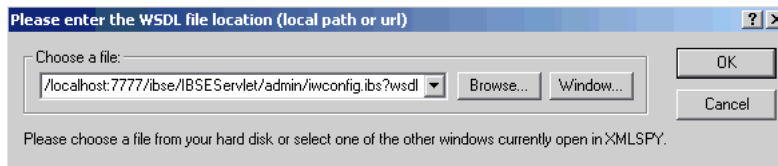
2. Open a third party XML editor, for example, XMLSPY.
3. From the menu bar, click **SOAP**.

A list of options appears.



4. Select **Create new SOAP request**.

The WSDL file location dialog is displayed.



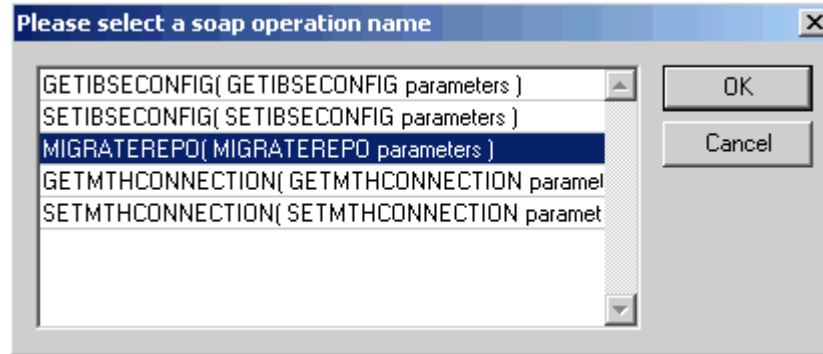
Perform the following steps:

- a. In the **Choose a file** field, paste the BSE control service URL.
- b. Append **?wsdl** to the URL, for example:

```
http://localhost:8001/ibse/IBSEServlet/admin/iwcontrol.ibs?wsdl
```

5. Click **OK**.

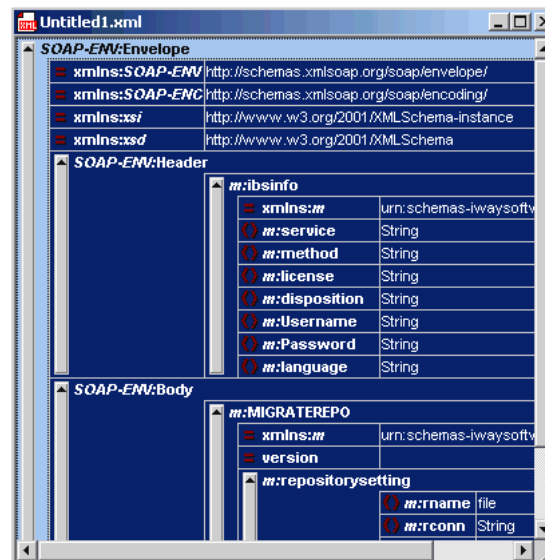
The soap operation name dialog is displayed, listing the available control methods.



6. Select the **MIGRATEREPO(MIGRATEREPO parameters)** control method and click **OK**.

Note: The **MIGRATEREPO(MIGRATEREPO parameters)** control method is available from the BSE administration console. This control method migrates all Web services to the new (empty) repository. You can choose to migrate select Web services only.

The following window is displayed. It shows the structure of the SOAP envelope.



7. Locate the **Text view** icon in the toolbar.



8. To display the structure of the SOAP envelope as text, click the **Text view** icon.

The `<SOAP-ENV:Header>` tag is not required and can be deleted from the SOAP envelope.

9. Locate the following section:

```
<m:MIGRATEREPO xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config"
version="">
```

```

<m:repositorysetting>
<m:rname>oracle</m:rname>
<m:rconn>String</m:rconn>
<m:rdriver>String</m:rdriver>
<m:ruser>String</m:ruser>
<m:rpwd>String</m:rpwd>
</m:repositorysetting>
<m:serviceName>String</m:serviceName>
</m:MIGRATEREPO>
    
```

Perform the following steps:

- a. For the **<m:rconn>** tag, replace the String placeholder with a repository URL where you want to migrate your existing BSE repository.

The Oracle repository URL has the following format:

```
jdbc:oracle:thin:@[host]:[port]:[sid]
```

- b. For the **<m:rdriver>** tag, replace the String placeholder with the location of your Oracle driver.
- c. For the **<m:ruser>** tag, replace the String placeholder with a valid user name to access the Oracle repository.
- d. For the **<m:rpwd>** tag, replace the String placeholder with a valid password to access the Oracle repository.

10. Perform one of the following migration options.

- If you want to migrate a single Web service from the current BSE repository, enter the Web service name in the **<m:serviceName>** tag, for example:

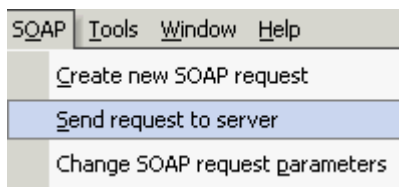
```
<m:serviceName>PeopleSoftService1</m:serviceName>
```

- If you want to migrate multiple Web services from the current BSE repository, duplicate the **<m:serviceName>** tag for each Web service, for example:

```
<m:serviceName>PeopleSoftService1</m:serviceName>
<m:serviceName>PeopleSoftService2</m:serviceName>
```

- If you want to migrate all Web services from the current BSE repository, remove the **<m:serviceName>** tag.

11. From the menu bar, click **SOAP** and select **Send request to server**, as shown in the following image.



Your BSE repository and any Web services you specified are now migrated to the new Oracle repository URL you specified.

Migrating a J2CA Repository

To migrate a J2CA repository:

1. Navigate to the location of your J2CA configuration directory where the repository schemas and other information is stored, for example:

C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\config\JCA_
CONFIG

Where *JCA_CONFIG* is the name of your J2CA configuration.

2. Locate and copy the `repository.xml` file.
3. Place this file in a new J2CA configuration directory to migrate the existing repository.

Your J2CA repository is migrated to the new J2CA configuration directory.

Generating Component Interface APIs

This section describes how to generate component interface APIs to use with Oracle Application Adapter for PeopleSoft.

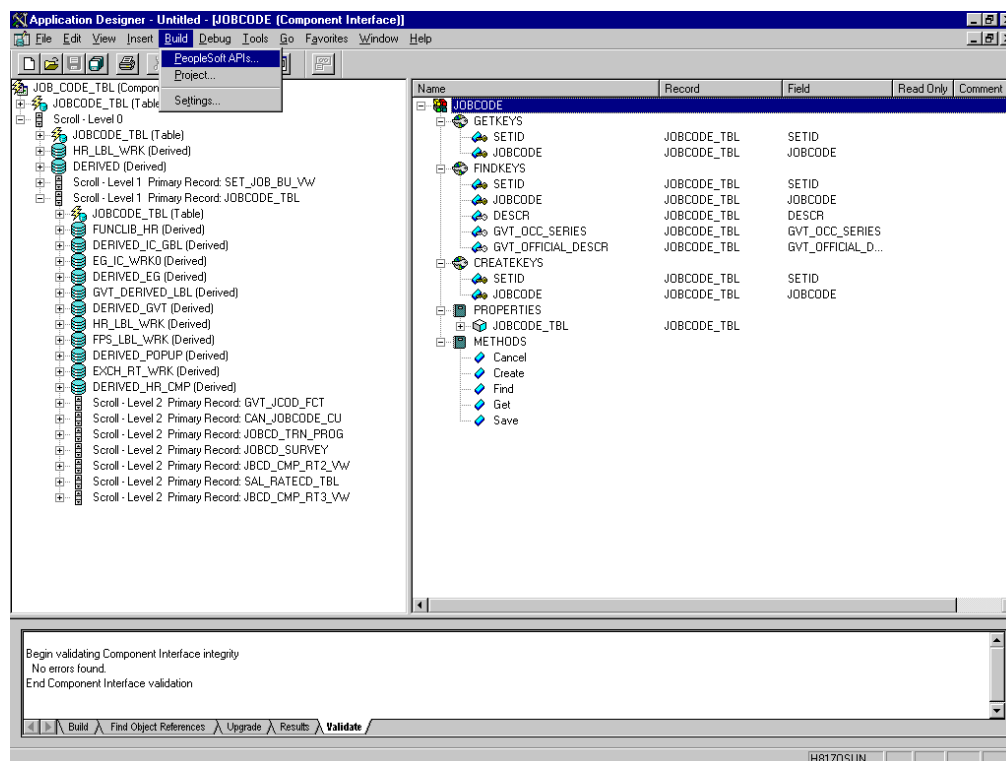
A.1 Building the PeopleSoft API Java Programs

Whether you are using an Enterprise Integration Point (EIP) supplied by PeopleSoft or a customized component interface, you must create a PeopleSoft API to enable communications with the PeopleSoft application. The API is a collection of Java class files that reside on the client system and mediate between the client application layer and PeopleSoft.

Before using your component interface, you must apply security to it and test it.

To build a PeopleSoft API Java program:

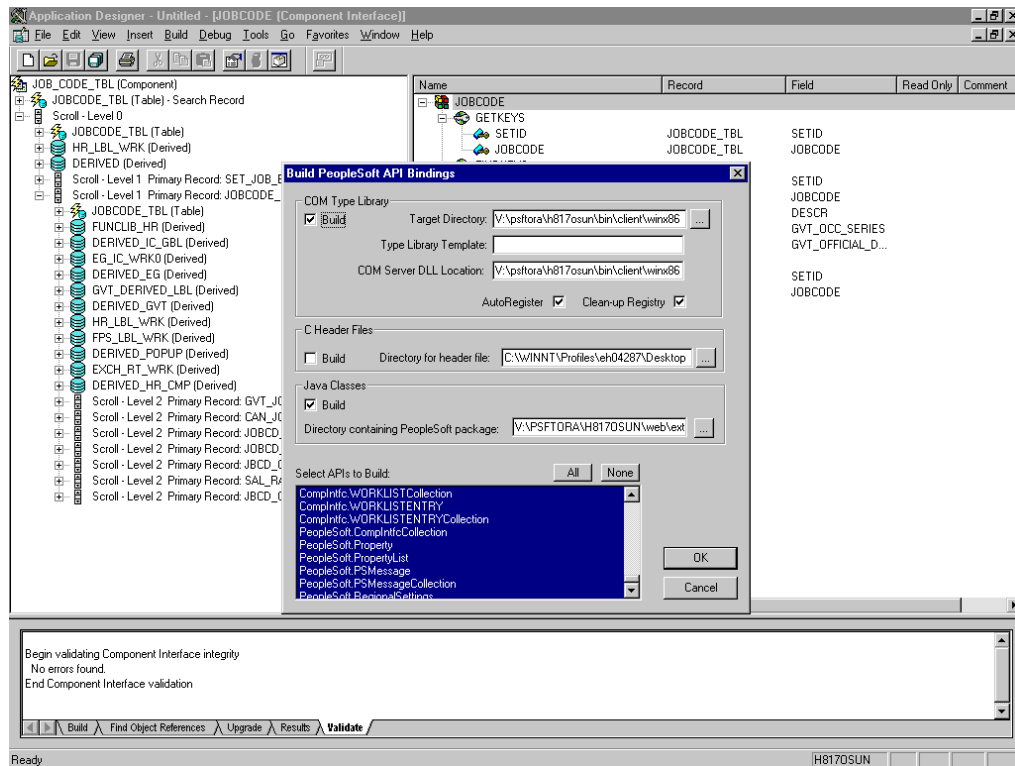
1. Open the PeopleSoft Application Designer.



2. From the PeopleSoft Application Designer, open a component interface.

- Click the right pane and select **PeopleSoft APIs** from the **Build** menu.

The Build PeopleSoft API Bindings dialog prompts you for the types of bindings to create.

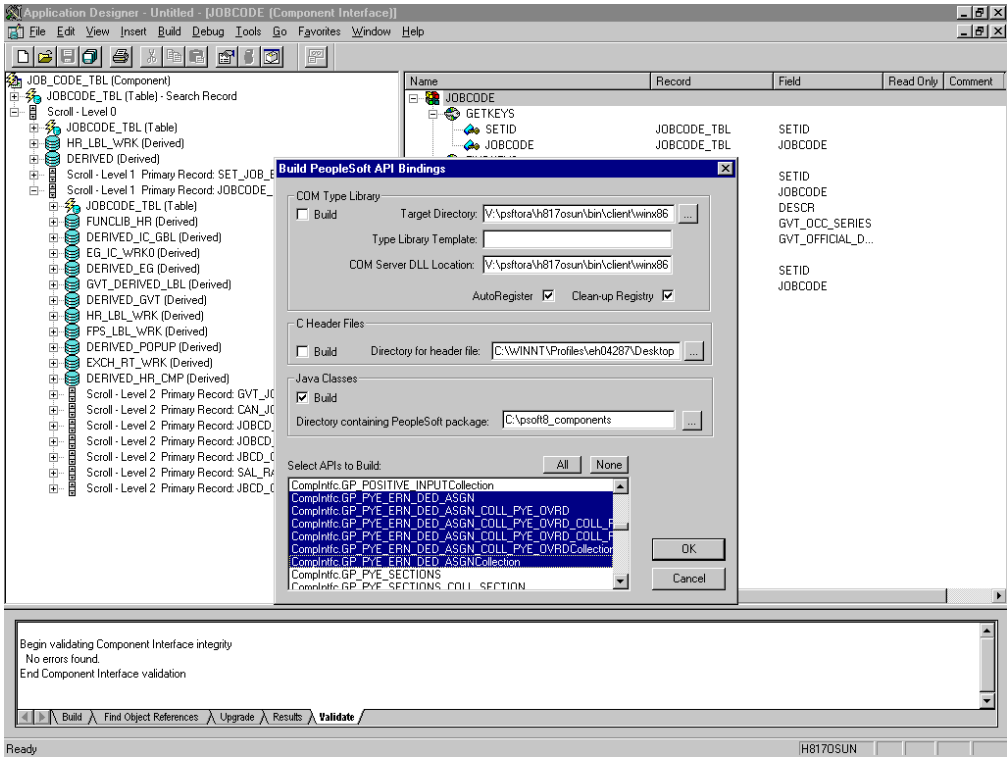


- Because you are creating Java files, ensure you deselect **COM Type Library Build**.
- Ensure that **Java Classes Build** is selected, and then select a directory on your local system where the Java files are to be placed, for example, `c:\psft8_components`.
 - To build all API files, select the default, **All** (potentially a large number), and click **OK**.

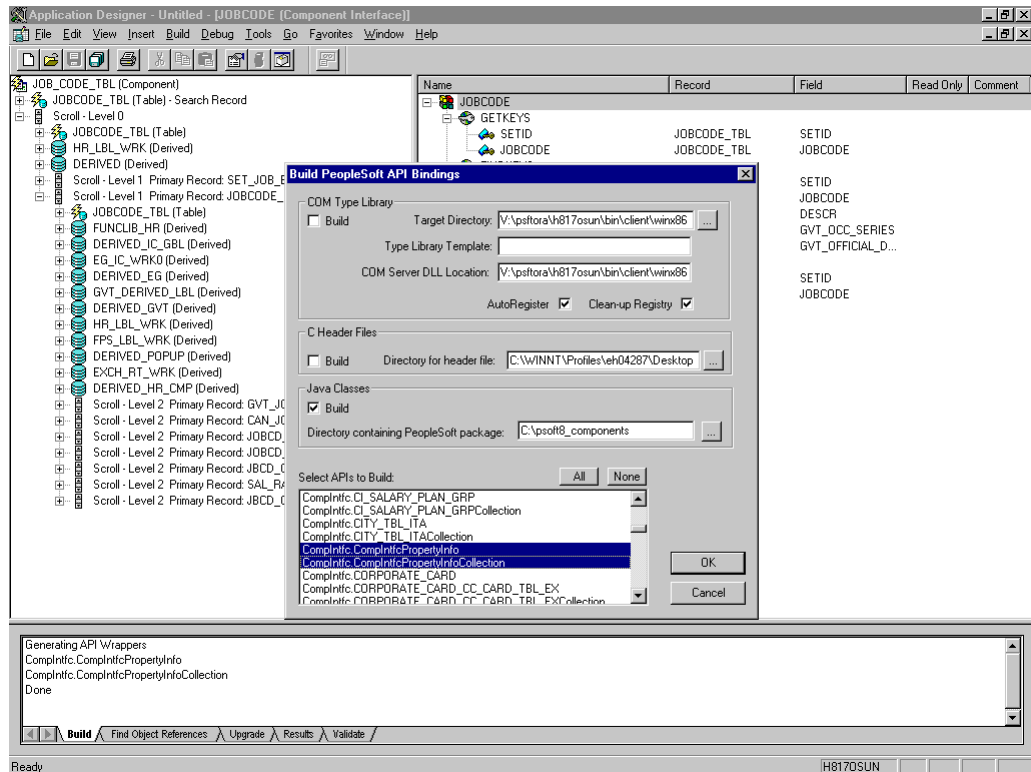
PeopleSoft generates the files. This takes a few minutes. After the process is complete, a message appears in the output window.

You are now ready to compile the Java files. See "[Compiling the PeopleSoft API Java Programs](#)" on page A-4 for more information.

The following pane illustrates the `GP_PYE_ERN_DED_ASGN` component interface from the HR 8.1 application.



- To create APIs for a specific component interface or interfaces, click **None**.
This clears the selected APIs, so you can select the appropriate APIs for your component interface. These APIs begin with the name of your component interface. There may be fewer than five, or more than 50 APIs, for a particular component interface.
The following dialog displays the following APIs, including generic component interface properties.



- a. In addition to the APIs for the selected component interface, you also must generate the API files for the following generic component interface properties:

CompIntfcPropertyInfo

CompIntfcPropertyInfoCollection

You may select these items in the same step as the component interface build, or you may select them separately.

- b. Click **OK**.

PeopleSoft generates the files. This takes a few minutes. After the process is complete, a message appears in the output window. You are now ready to compile the Java files. See "[Compiling the PeopleSoft API Java Programs](#)" on page A-4 for more information.

A.1.1 Compiling the PeopleSoft API Java Programs

PeopleSoft places the Java programs to compile in the directory called *psoft8_components\PeopleSoft\Generated\CompIntfc*.

Where *psoft8_components* is the directory specified during the build process.

If you chose to generate all APIs, the systems creates a second directory, *psoft8_components\PeopleSoft\Generated\PeopleSoft*. You are not required to access it.

The process for compiling the PeopleSoft API Java programs depends on whether you are compiling on the system where you installed Application Explorer or on another system.

Note: There are two Java programs for every API file that you selected when you built the Java programs. See "[Building the PeopleSoft API Java Programs](#)" on page A-1 for more information.

Before you compile the Java programs, you require the PeopleSoft Java Object Adapter, the `psjoa.jar` file that resides on your PeopleSoft Application Server under the `PS_HOME\Web\psjoa` directory. This is the file that you placed in the `adapter lib` directory during installation.

If you are compiling on the same system where you installed Application Explorer

Point to the `psjoa.jar` file or copy it to the directory where you placed the Java API files, for example, `c:\psoft8_components`.

If you are compiling on a system other than the one where you installed Application Explorer

Perform the following steps:

1. Obtain a copy of the `psjoa.jar` file from the PeopleSoft Application Server. Ensure that the `psjoa.jar` file is in the Java class path before you compile the programs.
2. Compile the Java programs and ensure that you include the `\PeopleSoft\Generated\CompIntfc` path.

The path is case-sensitive.

The following Windows batch file, run from the `psoft8_components` directory, properly compiles the Java APIs. The code assumes that `psjoa.jar` was placed in `psoft8_components`.

```
@echo off
set JAVA_HOME=<my-java-home>
set PATH=%JAVA_HOME%\bin;%PATH%
set CLASSPATH=%JAVA_HOME%\lib\tools.jar;psjoa.jar;%CLASSPATH%
javac -classpath %CLASSPATH% .\PeopleSoft\Generated\CompIntfc\*.java
```

Where `<my-java-home>` is the fully qualified path name of your Java home directory.

This code places the class files in the same directory with the Java files, but you can choose a different location depending on your site requirements.

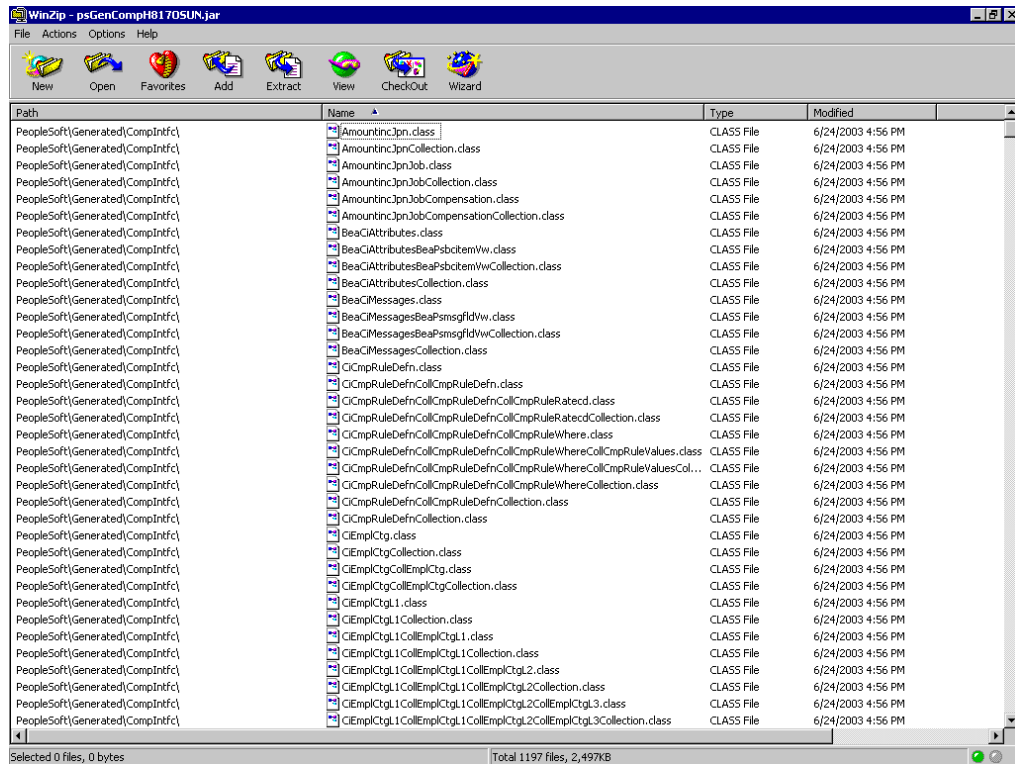
3. Compress the class files into a JAR file.

The following Windows batch file, if run from the `psoft8_components` directory, creates a correct JAR file:

```
@echo off
set JAVA_HOME= my-java-home
set PATH=%JAVA_HOME%\bin;%PATH%
set CLASSPATH=%JAVA_HOME%\lib\tools.jar;%CLASSPATH%
jar cvf my-jar-file.jar .\PeopleSoft\Generated\CompIntfc\*.class
```

Where appropriate, substitutions are made for `my-java-home` and `my-jar-file`.

4. To verify that your JAR file is correct, open it with the WinZip application.



If the JAR file does not use the case-sensitive
 PeopleSoft\Generated\CompIntfc\ path, you must go back and correct it.

- Place the JAR file in the adapters common lib directory, which enables the Oracle Application Adapter for PeopleSoft to communicate with the PeopleSoft component interface.

C:\oracle\Middleware\Oracle_SOA1\soa\thirdparty\ApplicationAdapters\lib

Note: If you run on UNIX, perform the compile and JAR steps on Windows and then move the file to your UNIX system. The JAR file is binary. If you use an FTP-based tool to move your JAR file from Windows to UNIX, the file format must be set to binary.

Configuring the PeopleSoft Message Router

This section describes how to configure and test a TCP/IP or HTTP target connector and a TCP/IP handler for PeopleSoft.

The following configuration topics assume you are familiar with PeopleSoft Integration Broker (in release 8.4) or Application Messaging (in release 8.1). If not, see [Appendix D, "Using PeopleSoft Integration Broker"](#) for more information. For a complete description *before* you work with Oracle Application Adapter for PeopleSoft, see your PeopleSoft documentation.

Note: In PeopleSoft release 8.1, the messaging architecture is called Application Messaging and includes Application Messaging Gateway. In release 8.4, the messaging architecture is called Integration Broker, which includes Integration Gateway. When discussing release-independent issues, this section uses release 8.4 terminology. When discussing release-specific issues, it uses release-specific terminology.

B.1 Configuring the TCP/IP or HTTP Target Connector for PeopleSoft 8.4

The procedures in this topic assume that your Integration Broker environment is configured and tested. See [Appendix D, "Using PeopleSoft Integration Broker"](#) for more information.

1. Configure the gateway for the TCP/IP Target Connector or HTTP Target Connector. See [Configuring the TCP/IP Target Connector](#) on page B-2 for more information.

Note: This step is optional when configuring the HTTP Connector. The HTTP Target Connector is supplied with your PeopleSoft application, and no special configuration steps are required. If you choose, you may configure default connection values on the Gateway. You can override these values when you configure the node.

2. Configure the node. See [Configuring the Node for the TCP/IP84 Connector](#) on page B-3 for more information.

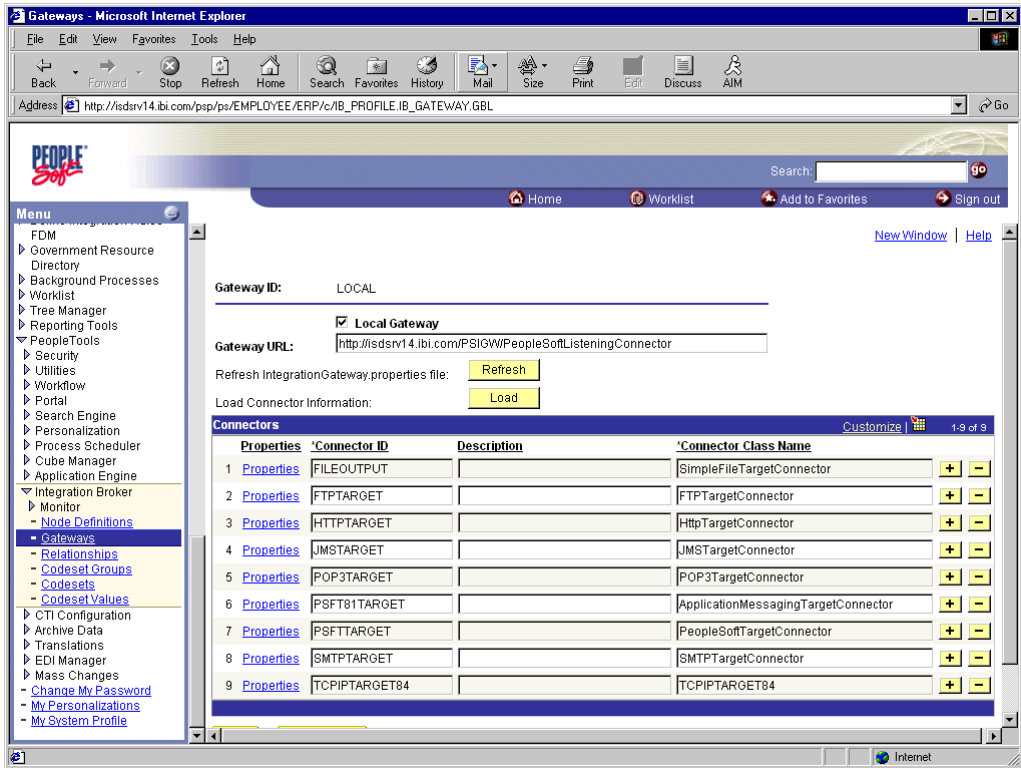
Note: Starting with release 8.4, the Integration Broker is delivered with an HTTP Outbound Connector. See [Configuring the HTTP Connector](#) on page B-8 for more information.

B.1.1 Configuring the TCP/IP Target Connector

To configure the gateway for the TCP/IP Target Connector:

1. In a Web browser, open your PeopleSoft release 8.4 application.
2. In the menu pane, expand **PeopleTools**, then expand **Integration Broker**, and click **Gateways**.
3. Open the **LOCAL** Gateway ID.

A pane similar to the following Gateway ID pane is displayed.



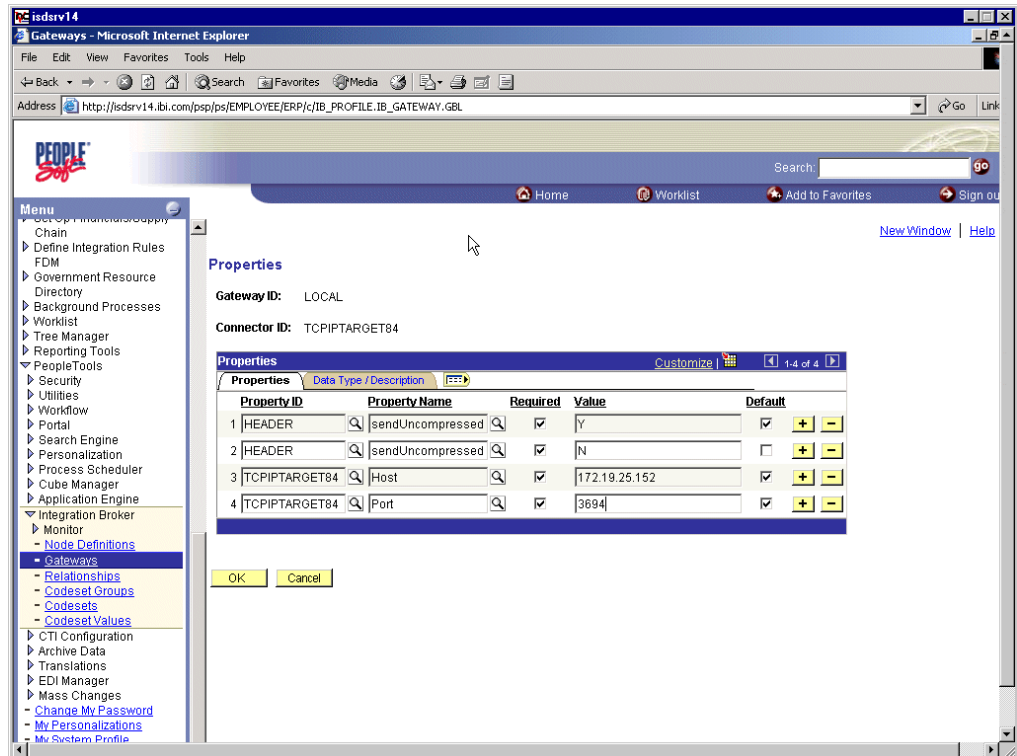
4. If you do not see the TCPIPTARGET84 Connector ID, click **Load** and scroll to locate **TCPIPTARGET84** in the list.

If TCPIPTARGET84 still does not appear, the connector class file was not installed in the Integration Gateway.

Perform the following steps:

- a. Click the **Properties** URL for TCPIPTARGET84.

The Properties pane for TCPIPTARGET84 is displayed. Default values appear for the host and the port. For complex business situations, you can override this setting on the individual node.



- b. Enter the values for the host and the port for the system on which your PeopleSoft XML listener is listening for incoming messages.
5. Click **OK**.
The Gateway window is displayed.
6. Scroll to the bottom of the window and click **Save**.
You have finished configuring the gateway for the TCP/IP Target Connector.

B.1.2 Configuring the Node for the TCP/IP84 Connector

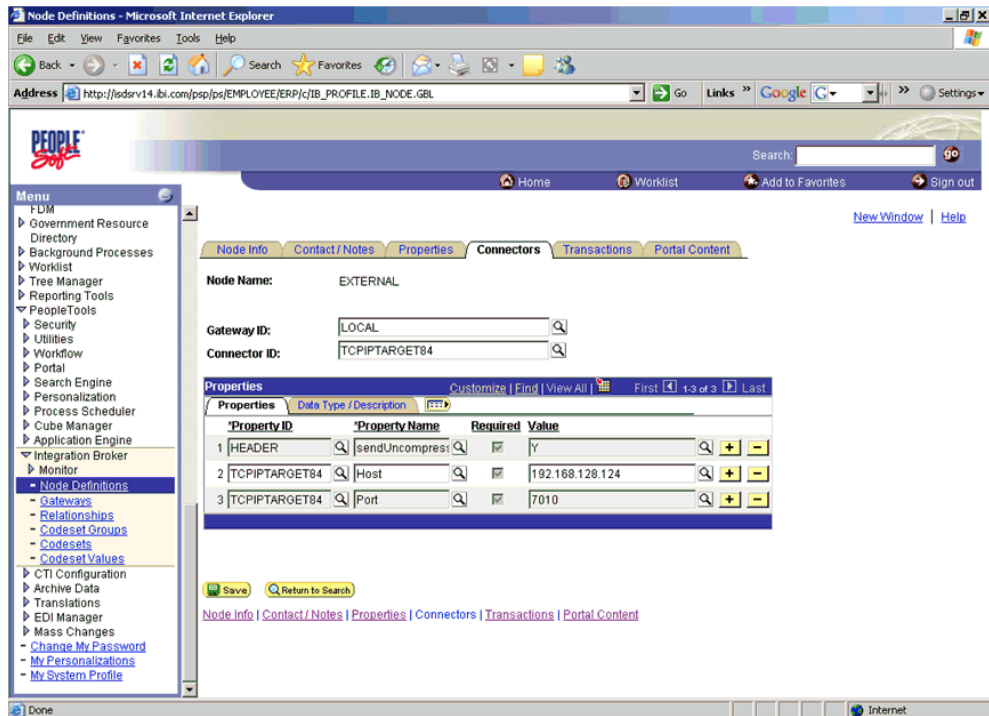
To configure the node for the TCP/IP84 Connector:

1. In the Menu pane, select **PeopleTools**, **Integration Broker**, and then click **Node Definitions**.
2. Select the node that you want to configure.

Note: This procedure uses a node called EXTERNAL. For more information about creating and using nodes, see [Appendix D, "Using PeopleSoft Integration Broker"](#) or your PeopleSoft documentation.

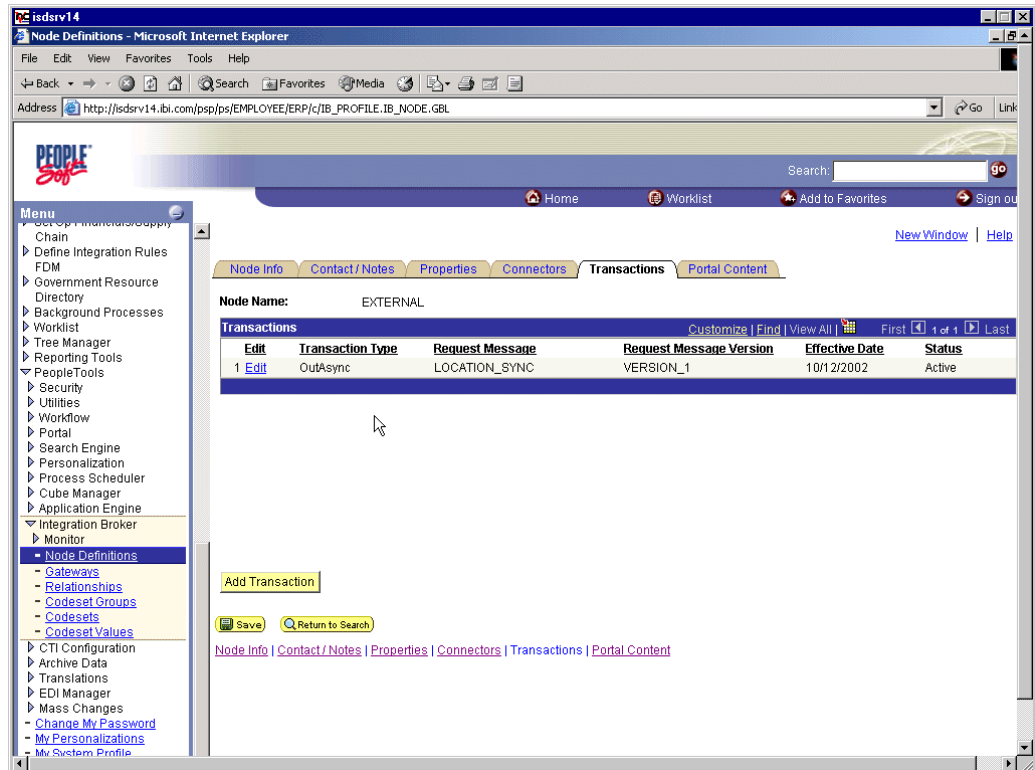
Perform the following steps:

- a. Select **External** from the **Node Type** list.
 - b. Select **Implicit** from the **Routing Type** list.
3. Select the **Connectors** tab.



Perform the following steps:

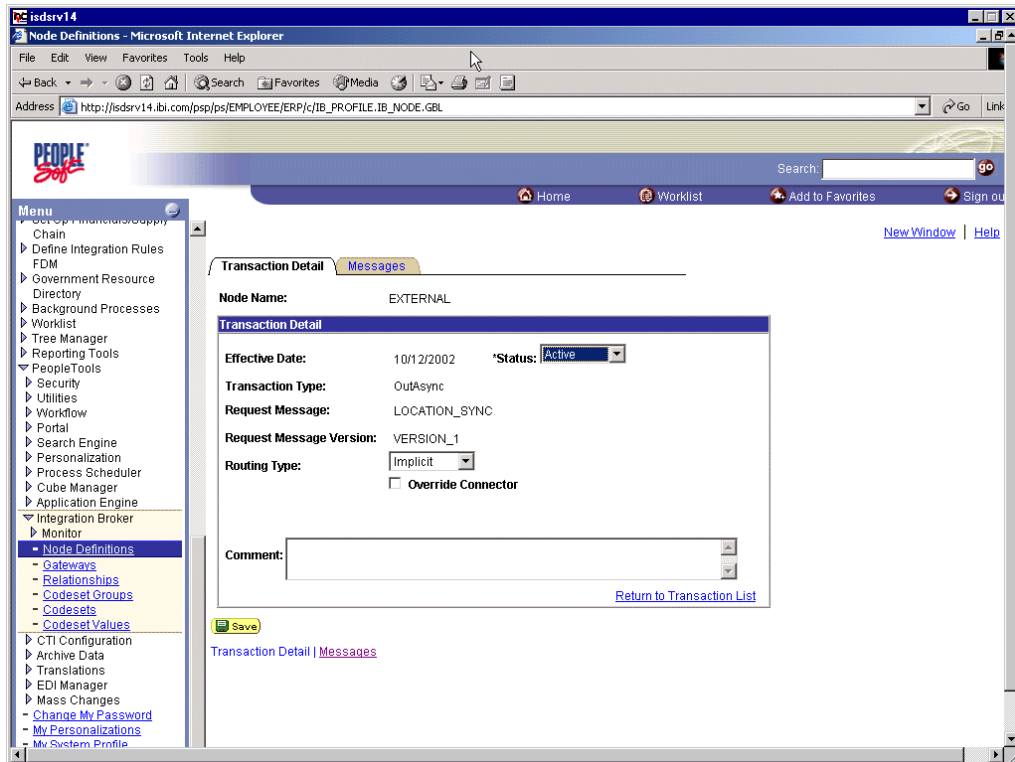
- a. Select **TCPIPTARGET84** as the Connector ID.
Default values appear for the host and the port.
- b. Enter the values for the host and the port for the system and port that route XML to Oracle WebLogic Server. You can accept or override the default values for individual nodes.
- c. Click **Save**.
4. If you are warned that you are changing the connector, click **OK**.
5. Select the **Transactions** tab.



Perform the following steps:

- a. If there are no transactions, click **Add Transaction** to add the message with which you are working. In this procedure, the node is already configured with the LOCATION_SYNC message.
- b. To view transaction details for the LOCATION_SYNC message, click **Edit**.

The Transaction Detail tab appears



- c. Add the message with which you are working.
 - d. Verify that the **Routing Type** is **Implicit**.
6. Click **Save**.
 7. Return to the **Transactions** tab.

Perform the following steps:

- a. Edit additional transactions by clicking **Edit** and navigating to the **Transaction Detail** tab.
 - b. In the **Transaction Detail** tab, select **Inactive** from the **Status** list.
- Inactive status is for initial testing only. After you test your configuration, you may change the status to Active and have as many nodes and transactions as required to satisfy your business requirements.
8. Click **Save**.

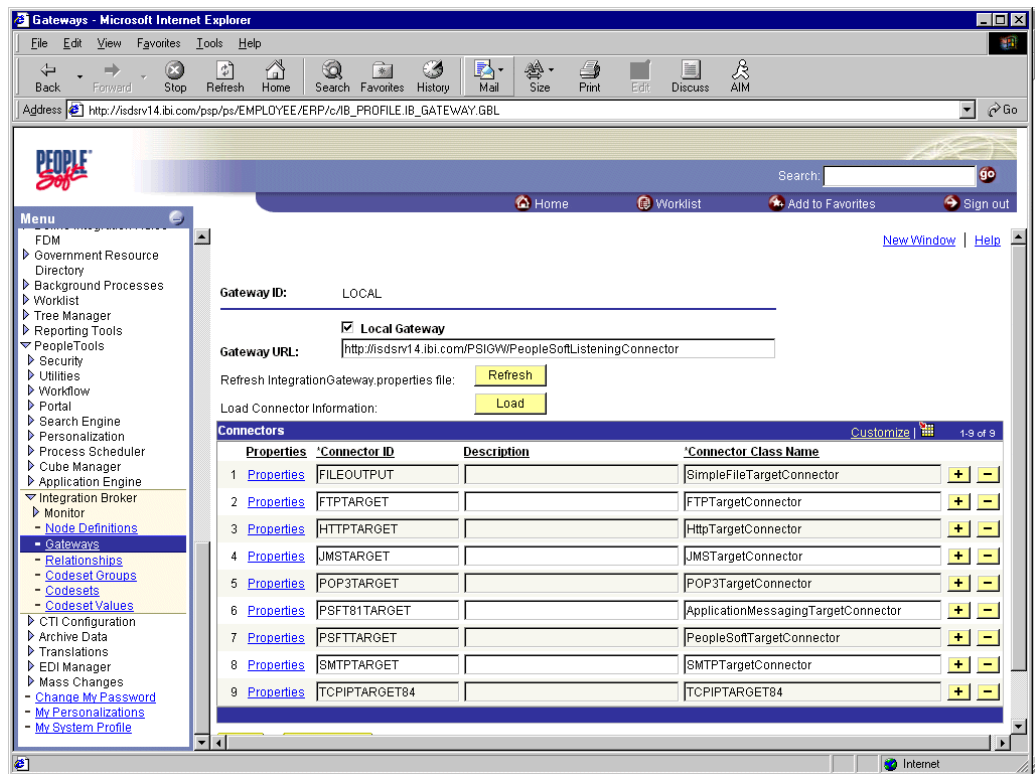
You can now send XML messages to your PeopleSoft XML listener.

B.1.2.1 Configuring the HTTP Target Connector

The HTTP Target Connector is supplied with your PeopleSoft application, and no special configuration steps are required. If you choose, you may configure default connection values on the Gateway. You can override these values when you configure the node. To configure the gateway for the HTTP Target Connector:

1. In a Web browser, open your PeopleSoft 8.4 application.
2. In the Menu pane, expand **PeopleTools**, then expand **Integration Broker**, and click **Gateways**.
3. Open the **LOCAL Gateway ID**.

A pane similar to the following Gateway ID pane is displayed.

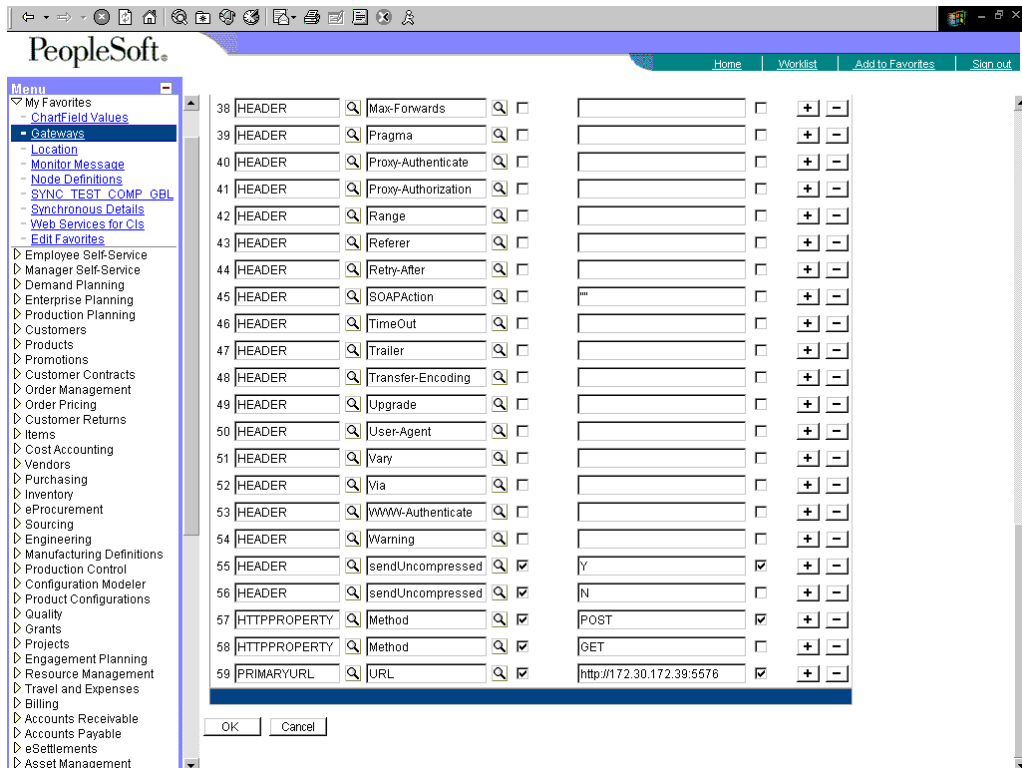


4. If you do not see the **HTTPTARGET** Connector ID, click **Load**.

If it does not appear, your Gateway was not installed properly. Check with your PeopleSoft system administrator.

5. Click the **Properties** URL for **HTTPTARGET**.

The Properties pane for **HTTPTARGET** displays default values:



6. Scroll to the bottom and enter a value for the **PRIMARYURL**.

This is the default HTTP address (system and port) on which your PeopleSoft XML listener is listening for incoming messages.

Note: For complex business situations, you can override this setting on the individual node.

7. Click **OK**.

The Gateway window is displayed.

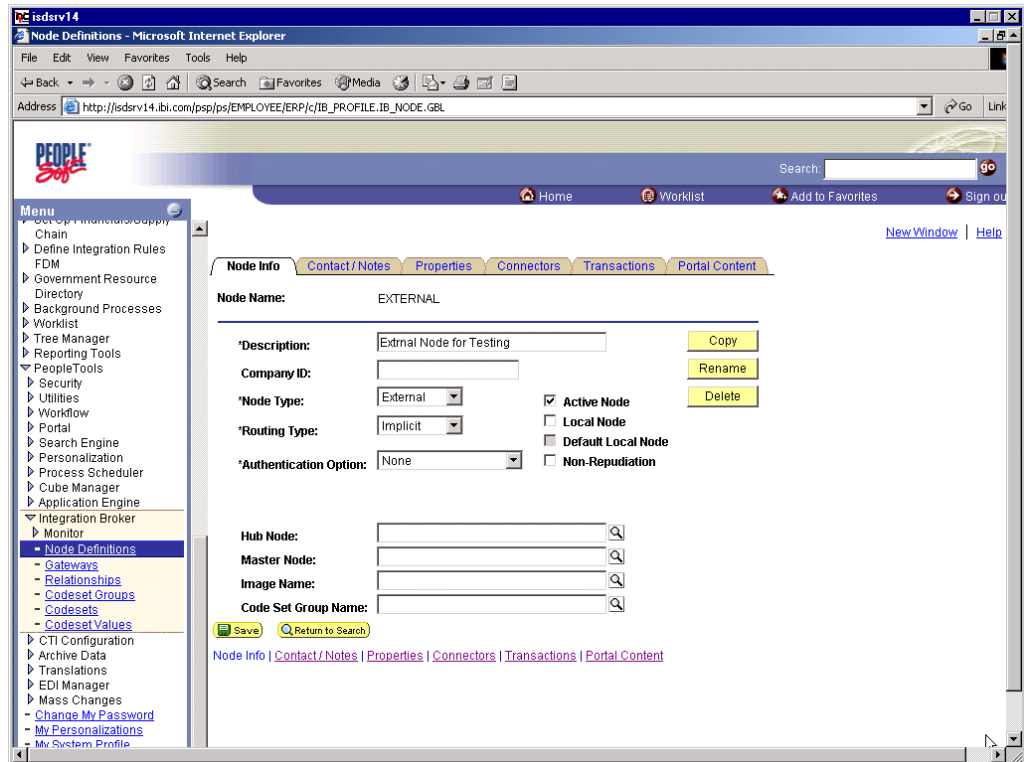
8. Scroll to the bottom of the window and click **Save**.

You have finished configuring the gateway for the HTTP Target Connector.

B.1.2.2 Configuring the HTTP Connector

Starting with release 8.4, the Integration Broker is delivered with an HTTP Outbound Connector. This connector can be used in place of the TCP/IP84 connector for sending messages to Oracle WebLogic Server.

To configure the node to use the HTTP Connector:

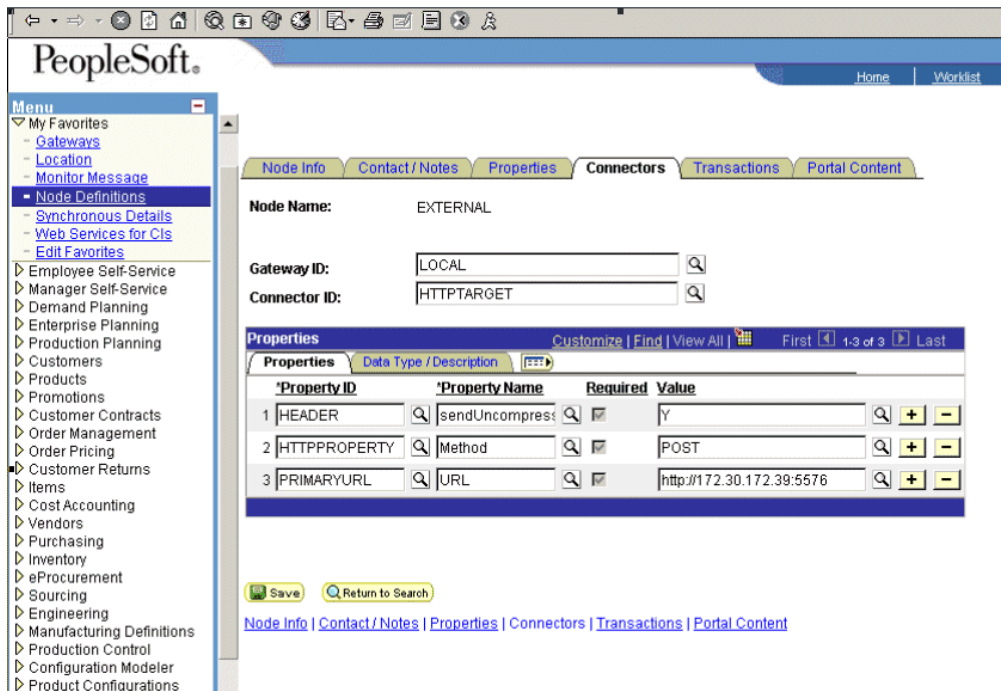


1. In the Menu pane, expand **PeopleTools**, **Integration Broker**, and then click **Node Definitions**.
2. Select the node that you want to configure.

Note: This procedure uses a node called EXTERNAL. For more information about creating and using nodes, see [Appendix D, "Using PeopleSoft Integration Broker"](#) or your PeopleSoft documentation.

Perform the following steps:

- a. From the **Node Type** list, select **External**.
- b. From the **Routing Type** list, select **Implicit**.



3. Select the **Connectors** tab.

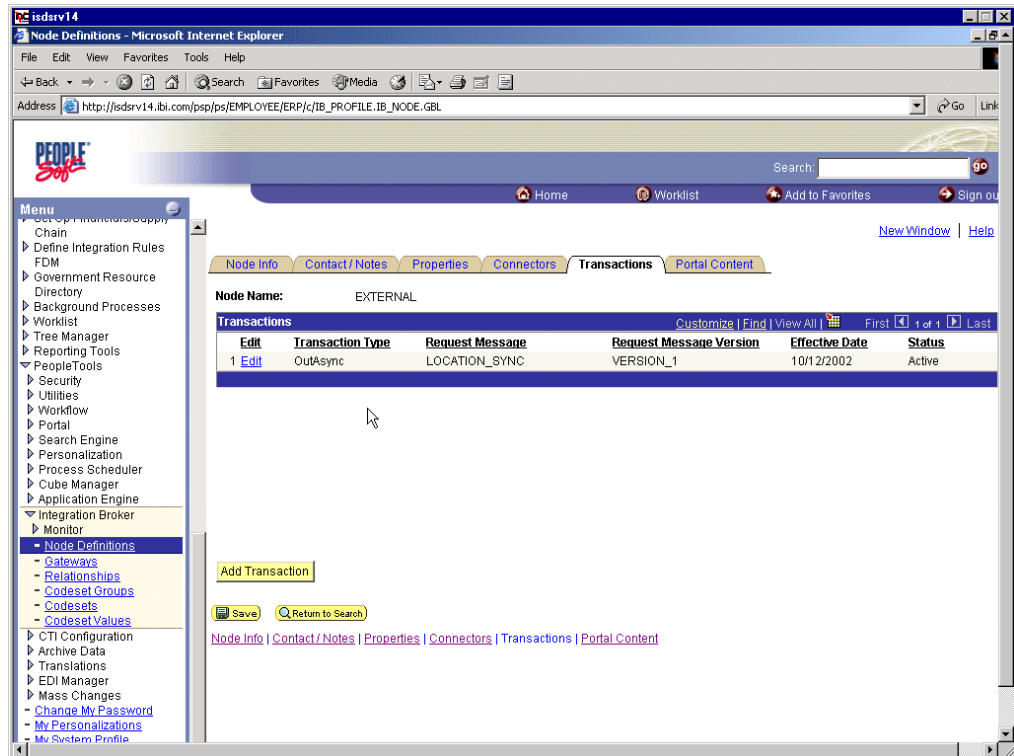
Perform the following steps:

- a. Change the Connector ID to **HTTPTARGET**.
- b. Enter a value for each property based on the following:

Property ID	Property Name	Value
HEADER	sendUncompressed	Y
HTTTPROPERTY	Method	POST
PRIMARYURL	URL	URL and the port of the HTTP listener

Note: For complex business situations you can configure multiple nodes and multiple listeners.

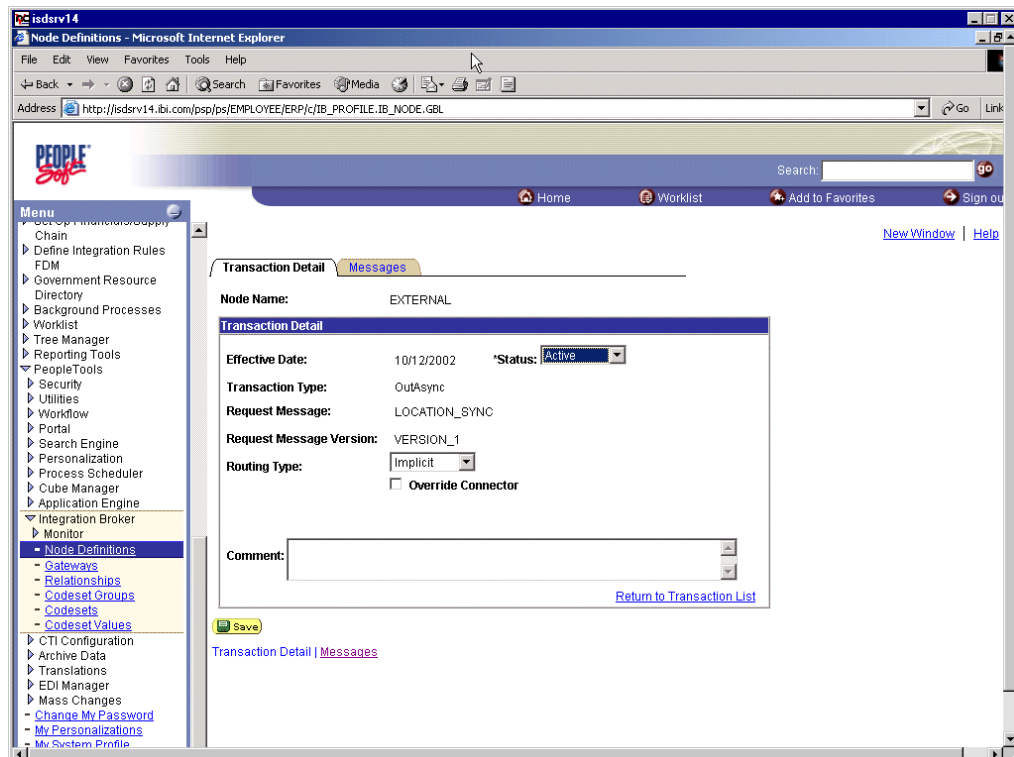
- 4. Click **Save**.
- 5. If you are warned that you are changing the Connector, click **OK**.
- 6. Select the **Transactions** tab.
The following pane is displayed.



7. If there are no transactions, click **Add Transaction**.

In this procedure, the node is already configured with the LOCATION_SYNC message.

The Transaction Detail tab appears.



You can add the message with which you are working.

- a. Verify that the **Routing Type** is **Implicit**.
 - b. Click **Save**.
8. Return to the Transaction List. Perform the following steps:
 - a. If there are other transactions, edit them.
 - b. Set the status to **Inactive**.

Inactive status is for initial testing only. After you test your configuration, you may change the status to Active and have as many nodes and transactions as required to satisfy your business requirements.

9. Click **Save** on the Transaction List.

You can now send XML messages to your PeopleSoft XML listener.

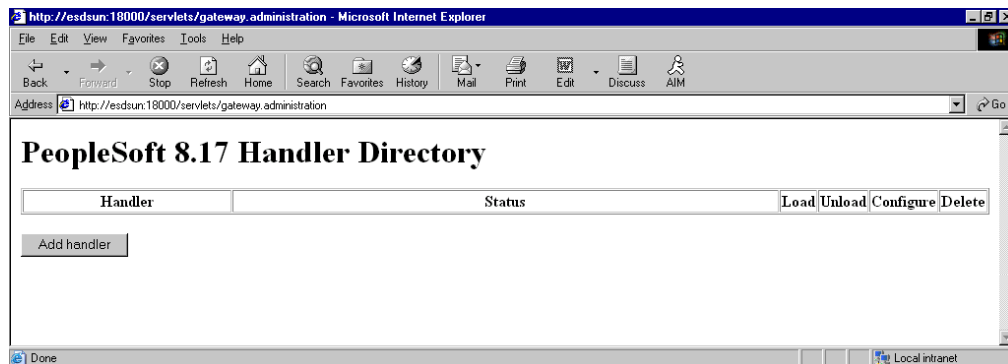
B.1.3 Configuring the TCP/IP Handler for PeopleSoft 8.1

The following procedure assumes that your Application Messaging environment is properly configured and tested. See [Appendix D, "Using PeopleSoft Integration Broker"](#) for more information.

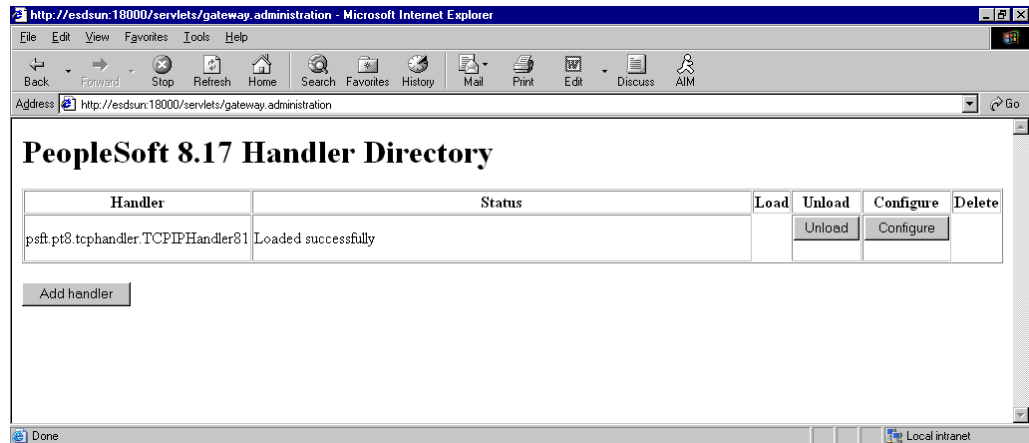
To configure the TCP/IP Handler for PeopleSoft 8.1 to send messages to Oracle WebLogic Server:

1. In a Web browser, launch the **PeopleSoft 8.1 Gateway Configuration** servlet interface.
2. If the Simple File Handler is currently loaded, unload and delete it before proceeding.

You must see an empty Handler directory.



3. Click **Add handler**.



Perform the following steps:

- a. Enter the full path of TCPIPHandler81 (case-sensitive):

psft.pt8.tcphandler.TCPIPHandler81

- b. Click **Save**.

4. Click **Load**.

The PeopleSoft Handler Directory window is displayed.

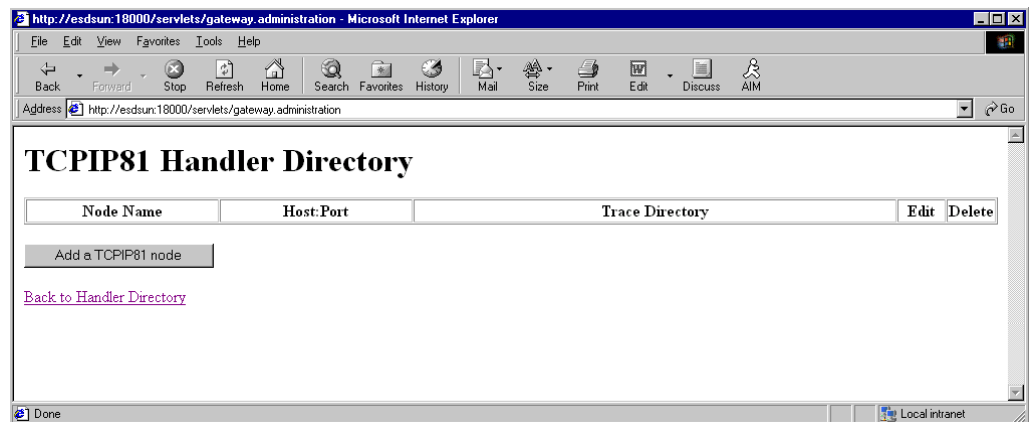
5. Click **Configure**.

The TCPIP81 Handler Directory window is displayed.

6. Click **Add a TCPIP81 node**.

Note: The screens illustrating this procedure show a node named EXTERNAL. For more information about creating and using nodes, see [Appendix D, "Using PeopleSoft Integration Broker"](#) or your PeopleSoft documentation.

The Add TCPIP81 Handler window is displayed.



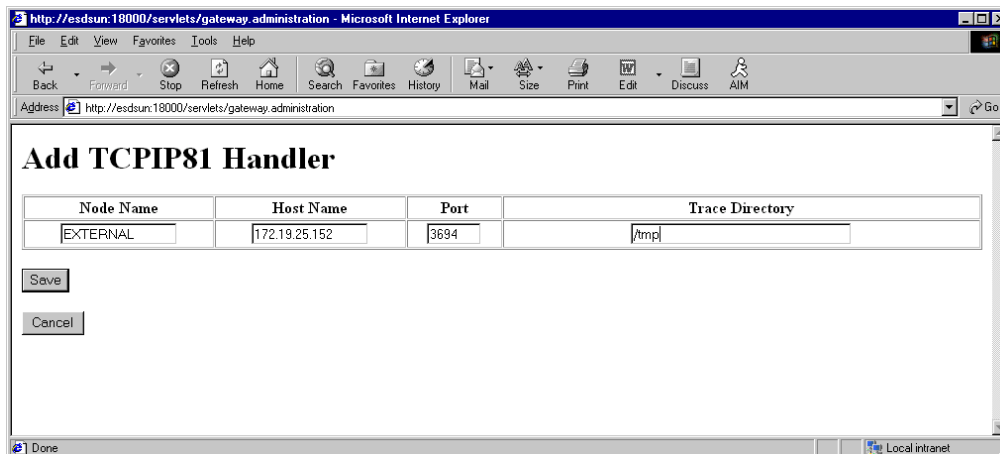
7. Enter the requested values based on the information in the following table.

Field	Value Example	Description
Node Name	EXTERNAL	The name of the TCP/IP node.
Host Name	172.19.25.152	The system on which your PeopleSoft XML listener is listening for incoming messages.
Port	3694	The port on which your PeopleSoft XML listener is listening for incoming messages.
Trace Directory	/tmp	The directory where a trace file is created when errors occur in message delivery.

The system does not validate your entries.

8. Click Save.

The TCP/IP81 Handler Directory window is displayed.



9. For your changes to take effect, click Back to Handler Directory to return to the PeopleSoft 8.1 Handler Directory window.

10. Click Unload and re-Load TCPIPHandler81.

You can now send messages from PeopleSoft to Oracle Application Adapter for PeopleSoft.

B.1.4 Testing Your PeopleSoft Configuration

PeopleSoft 8.1 and 8.4 provide a ping node mechanism for testing your configuration. The mechanism functions identically in both versions.

Test your configuration to ensure that:

- Oracle WebLogic Server is up and running.
- The server name and port number for PeopleSoft and Oracle WebLogic Server match.

- The default page for HTTP exists.

To test a PeopleSoft configuration:

1. In a Web browser, open your PeopleSoft application.
2. Navigate to the message monitoring menu.

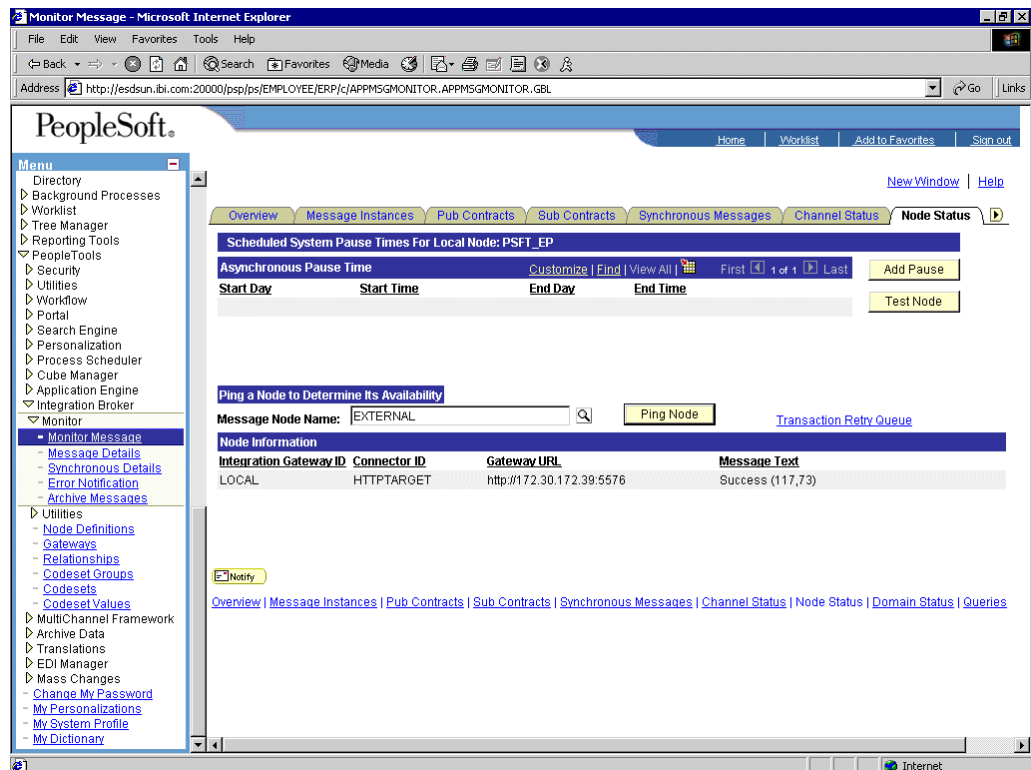
For PeopleSoft 8.4:

- a. In the menu pane, expand **PeopleTools**, **Integration Broker**, and **Monitor**.
- b. Select **Monitor Message**.

For PeopleSoft 8.1:

- a. In the menu pane, expand **Home**, **PeopleTools**, **Application Message Monitor**, and **Use**.
- b. Select **Application Message Monitor**.

3. Click the **Node Status** tab.



Perform the following steps:

- a. From the Message Node Name list, select your node.
- b. Click **Ping Node**.

If you properly configured both PeopleSoft and Oracle WebLogic Server, you receive a Success message.

An error indicates a configuration problem. For more information, see the Integration Broker error log.

Using Component Interfaces

This appendix describes how to create new and modify existing component interfaces for use with Oracle Application Adapter for PeopleSoft. It also describes how to apply security to those component interfaces and how to test them.

You can:

- Use component interfaces supplied by PeopleSoft with your application.
- Component interfaces also are known as Enterprise Integration Points (EIP).
- Modify an existing component interface.
- Create a new component interface.

Before using your component interface you must apply security to it and test it.

After securing and testing a component interface, you must generate its API, see [Appendix A, "Generating Component Interface APIs"](#) for more information.

Note: This section is intended as a helpful supplement; it is not a substitute for PeopleSoft documentation. For complete and up-to-date information about PeopleSoft component interfaces, see the PeopleSoft Online Library for your PeopleSoft system.

C.1 Creating a Component Interface

You create component interfaces using the PeopleSoft Application Designer. For more information about Application Designer, see your PeopleSoft documentation.

C.1.1 Working With Properties

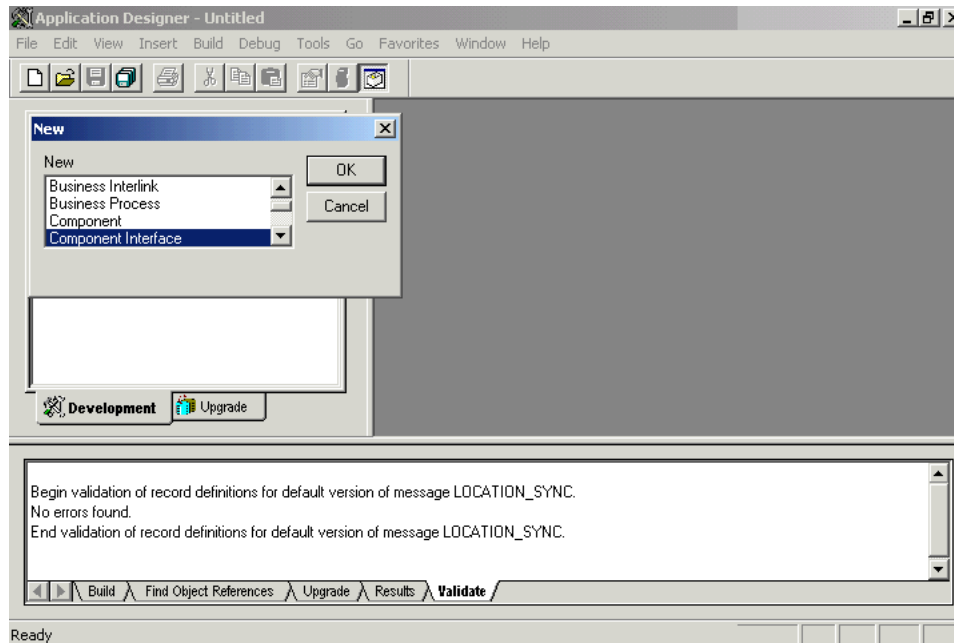
You can add properties from the records in the component view. You can delete a property in the component interface that you do not want to expose. You can rename properties by clicking the property and then clicking again until you can enter a new name. If you rename a property, it can be referenced in the component interface only by the new name, not by the underlying component name.

Properties may have various icons adjacent to them. For example, EMPLID has an icon indicating that it is a key field from the underlying record. NAME has an icon indicating that it is an alternate key field from the underlying record. For a complete list of property icons, see the PeopleBooks documentation.

C.1.1.1 Creating a New Component Interface

To create a component interface:

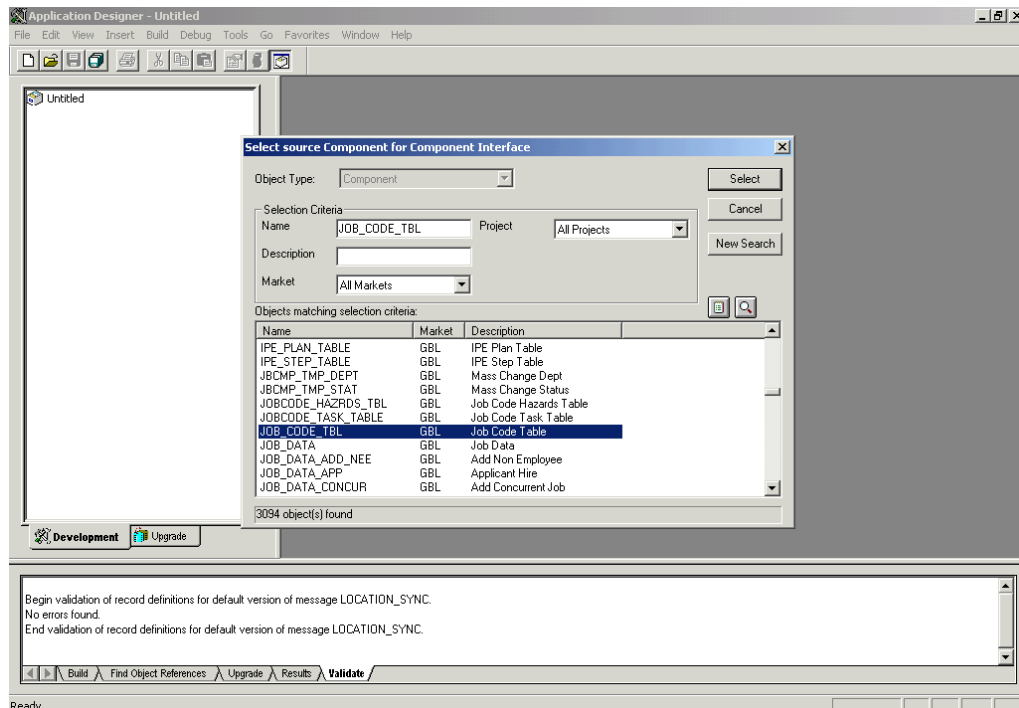
1. Open the PeopleSoft Application Designer.
 2. Select **New** from the **File** menu.
- The New dialog is displayed.



Perform the following steps:

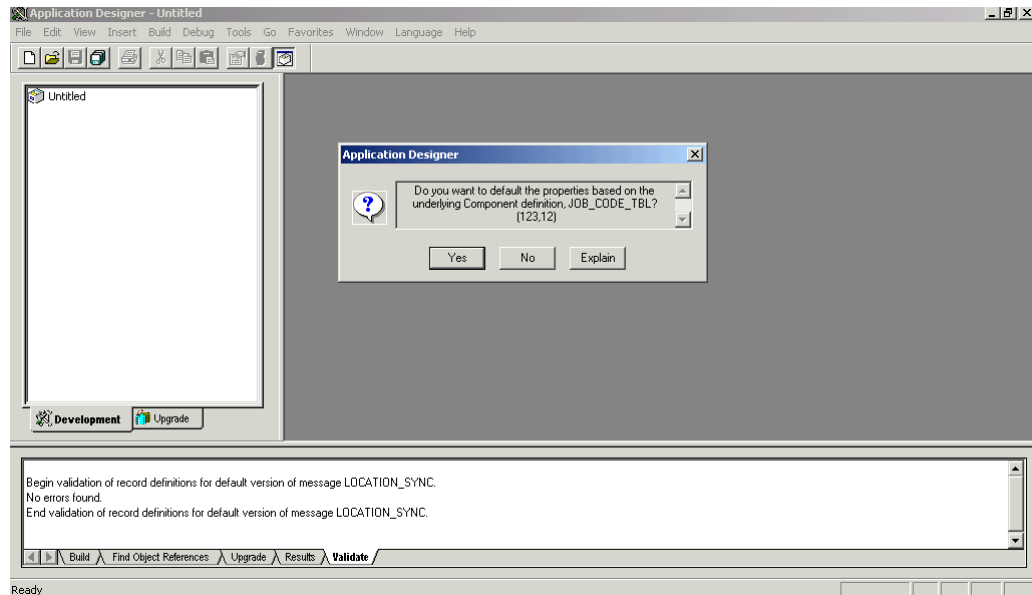
- a. Select **Component Interface**.
- b. Click **OK**.

The Select Source Component for Component Interface dialog is displayed.



3. Highlight the component to use as a basis for the component interface and click **Select**.

The Application Designer dialog is displayed.



Note: If the component interface is large, expose the component properties manually.

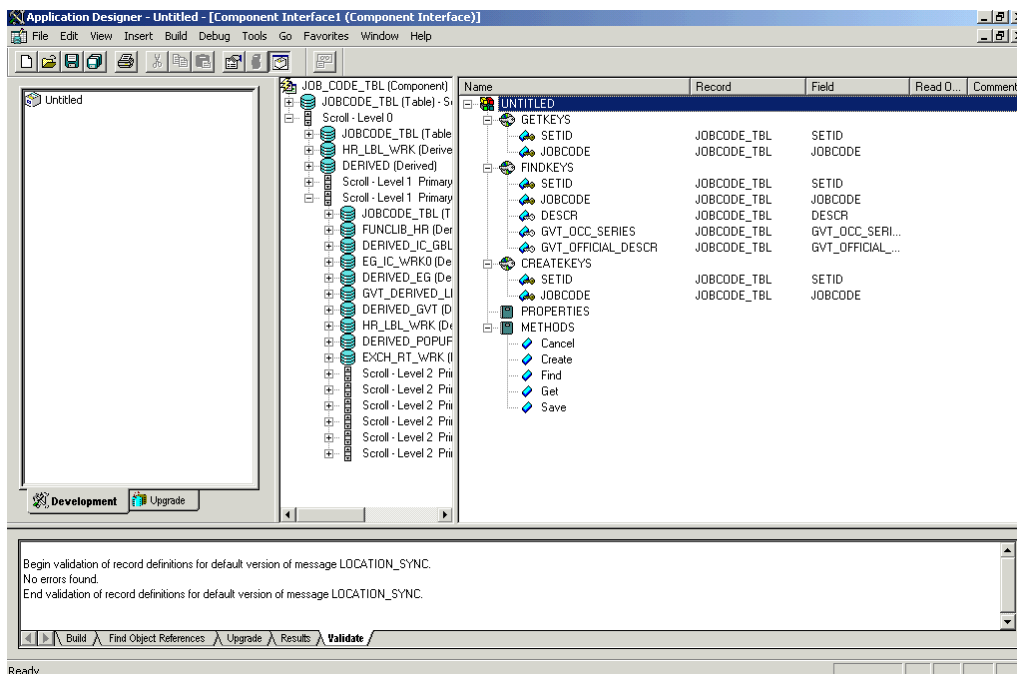
4. To create the component interface without displaying properties and to expose component properties manually, click **No**.

Perform the following steps:

- a. Drag the relevant fields from the left pane to the right pane.
- b. To select various functions to perform, right-click either the right or left pane, depending on which pane is active.

For a complete list of functions, see the PeopleBooks documentation.

5. To create the component interface and display the properties of the underlying component interface, click **Yes**.



Standard Methods

The standard methods for the component interface are:

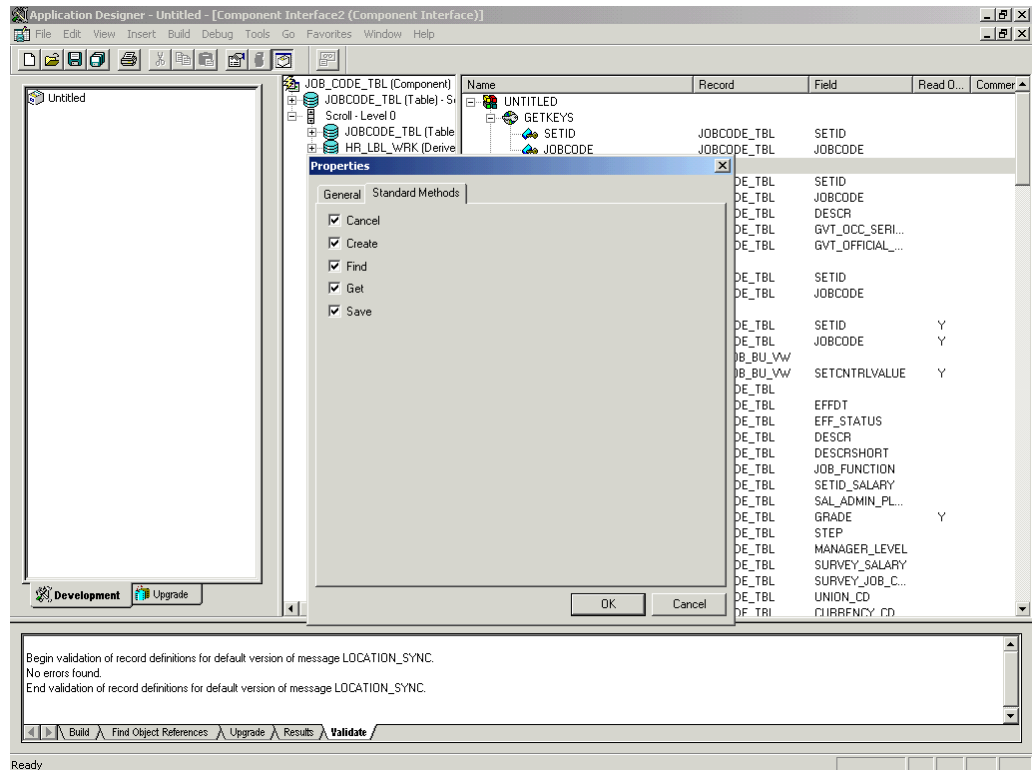
- Create
- Find
- Get
- Save

Only those methods in the underlying component are available. For example, if the underlying component does not contain Add capabilities, Create is not available.

C.1.1.2 Viewing or Changing Available Methods

To view or change available methods:

1. Display the Component Interface Properties dialog.



2. Click the **Standard Methods** tab.
3. Select the desired methods.

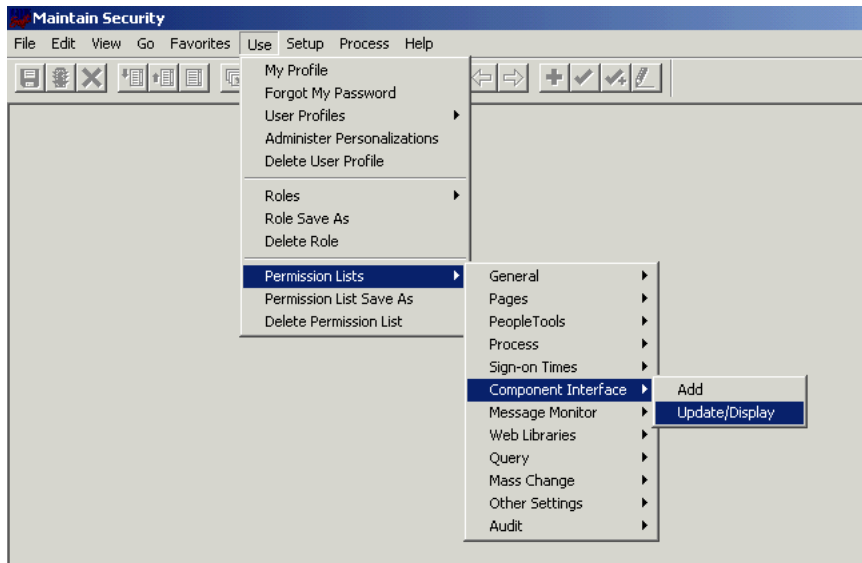
C.1.2 Securing a Component Interface

You must set up security for the component interface before you can begin testing.

C.1.2.1 Configuring Component Interface Security for PeopleSoft Version 8.1x

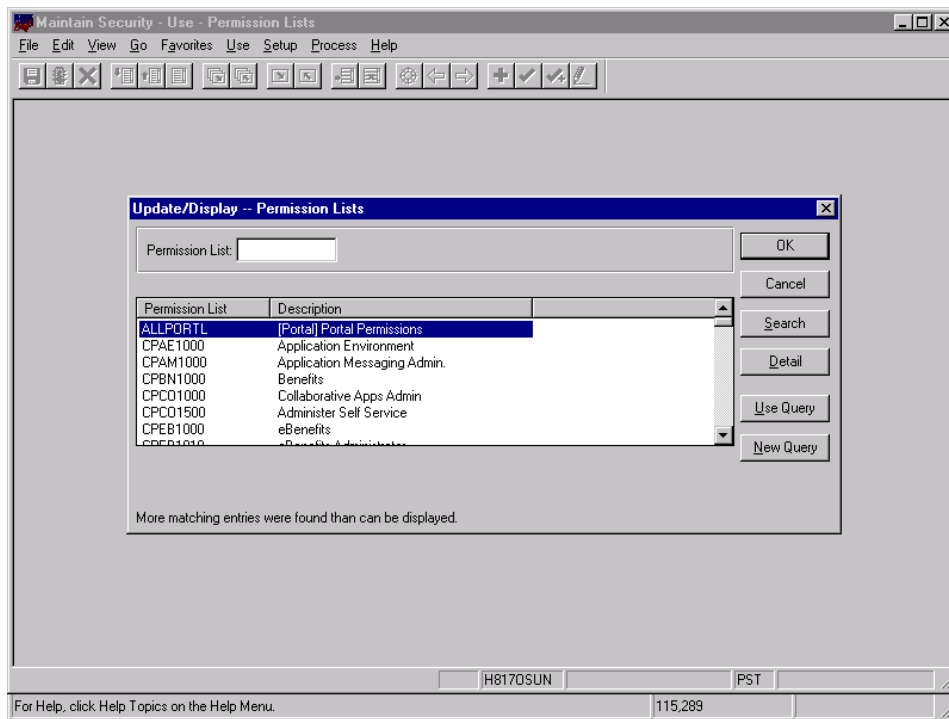
The following procedure describes how to configure component interface security for PeopleSoft Version 8.1 in 2- and 3-tier mode.

To configure component interface security:



1. From the **Use** menu, select **Permission Lists**, **Component Interface**, and then click **Update/Display**.

The Permission Lists dialog is displayed.

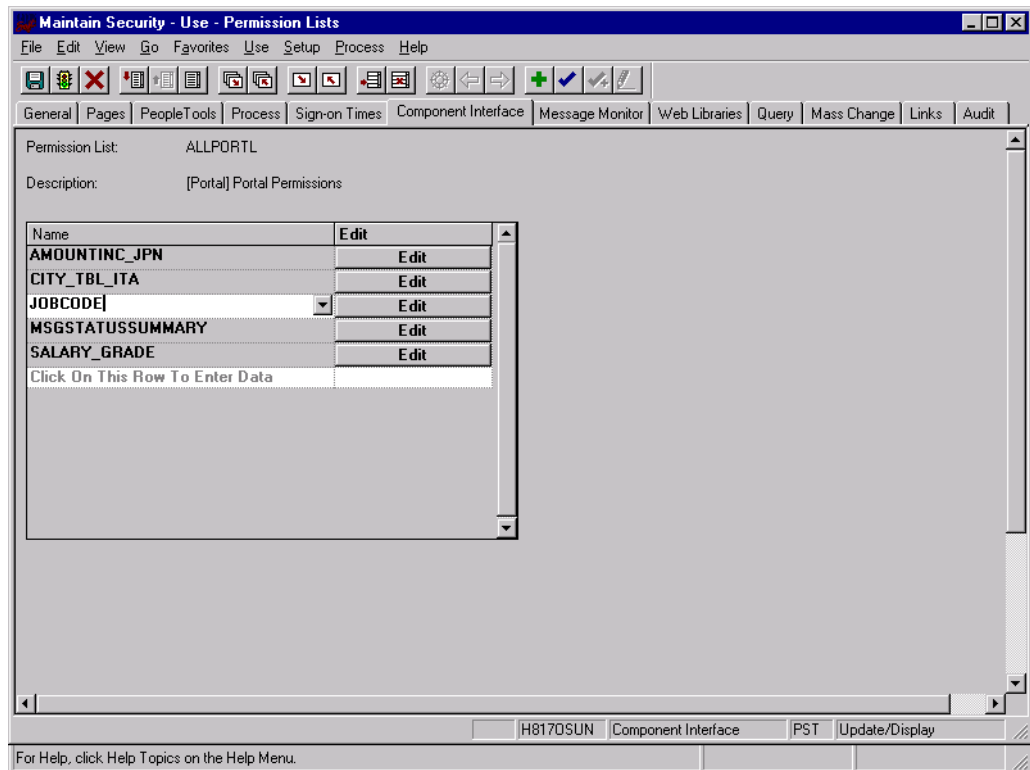


Before Security can be set, you must identify the permission lists.

2. Select the relevant permission list and click **OK**.

For more information on permission lists, see the PeopleBooks documentation.

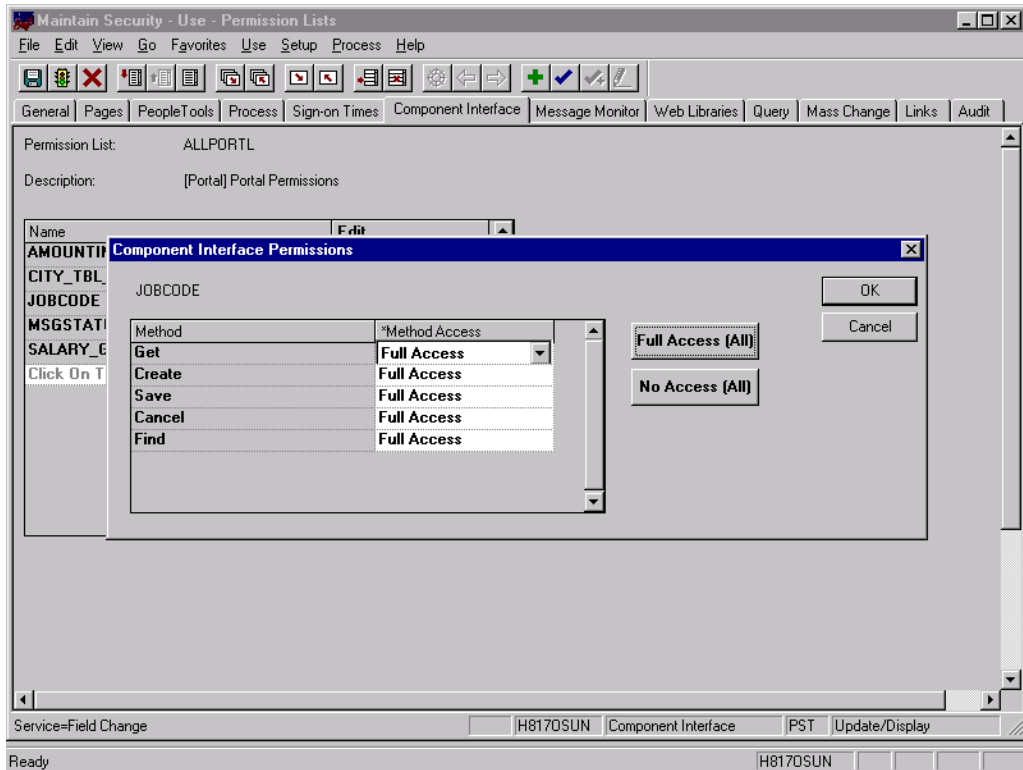
The following pane is displayed.



3. Insert the new component interface that you created.
4. Click **Edit**.

When you select the component interface, all available methods appear, including user-defined methods. You can specify whether this particular Permission List must have full or partial access.

In the following example, the ALLPORTL Permission List has full access to all methods.

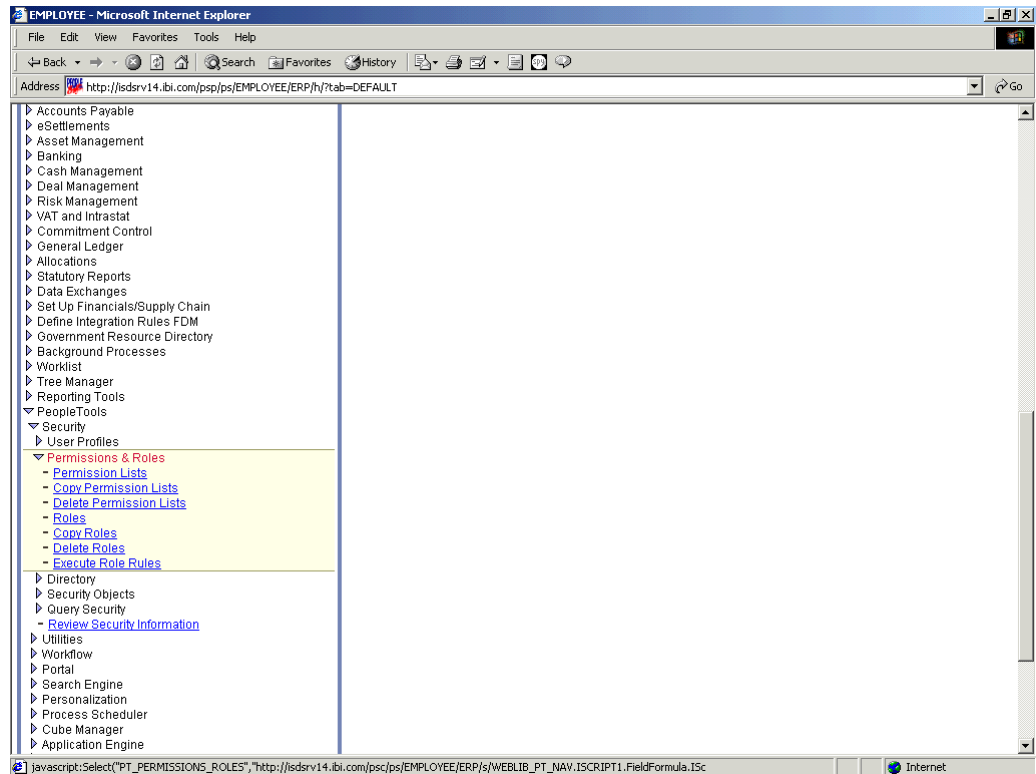


5. Select the desired level of access.
6. Click OK.

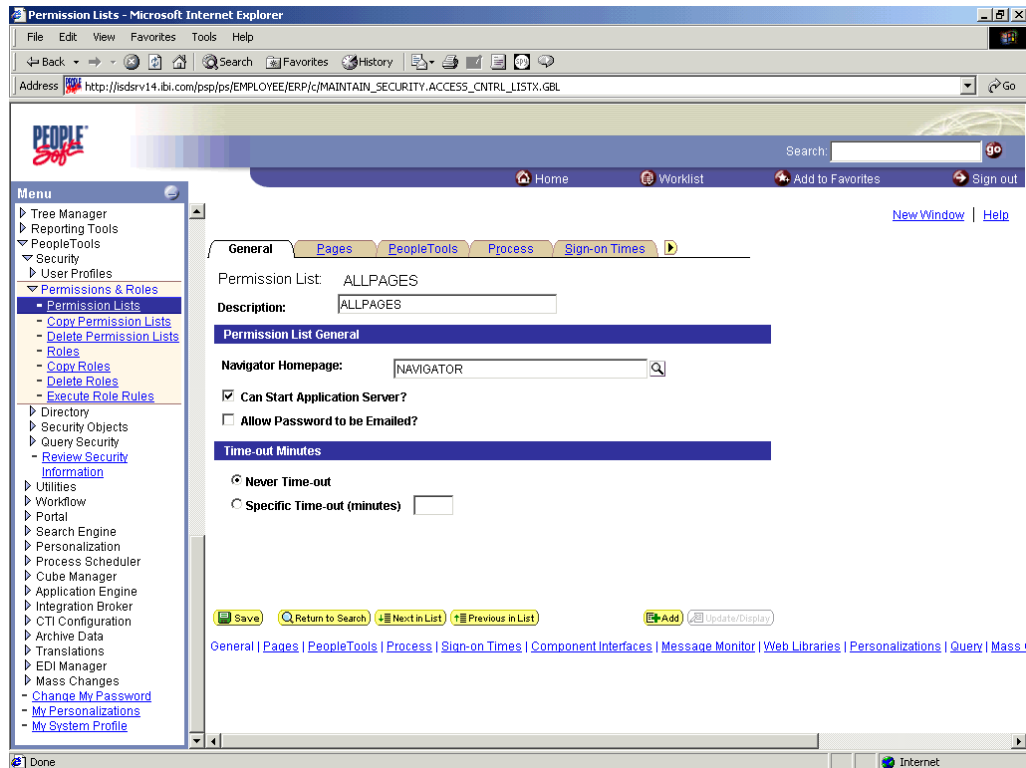
C.1.2.2 Configuring Component Interface Security for PeopleSoft Version 8.4 or Higher

The following procedure describes how to configure component interface security for PeopleSoft Version 8.4 or higher.

To configure interface security:

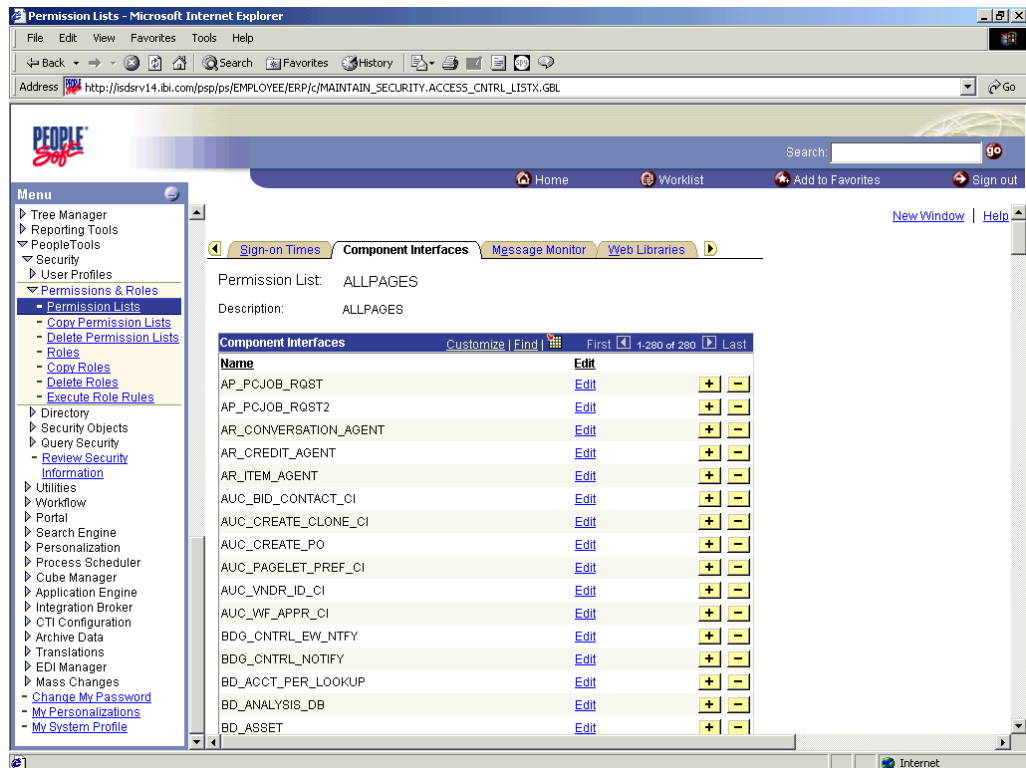


1. Expand **PeopleTools**, **Security**, **User Profiles**, and **Permissions & Roles** and then click **Permission Lists**.
2. Click **Search**.
The Permission Lists Search pane is displayed.
3. Select the relevant permission list.
The following pane is displayed.



4. Click the right arrow next to the **Sign-on Times** tab.

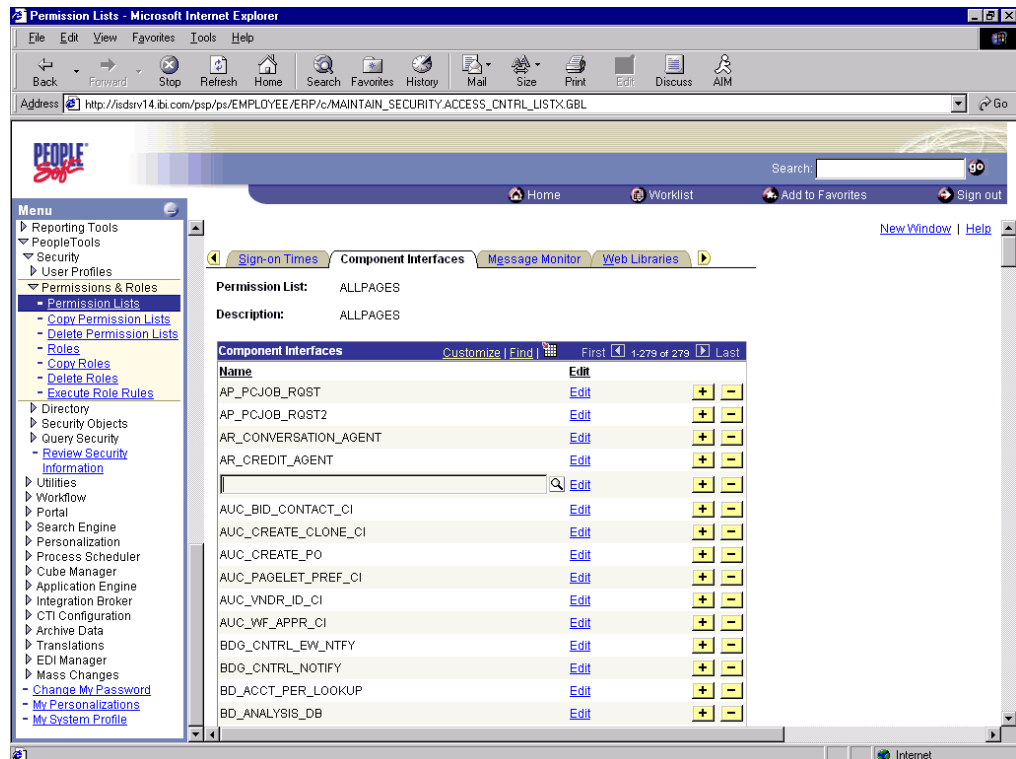
The **Component Interfaces** tab appears.



5. Click the **Component Interfaces** tab.

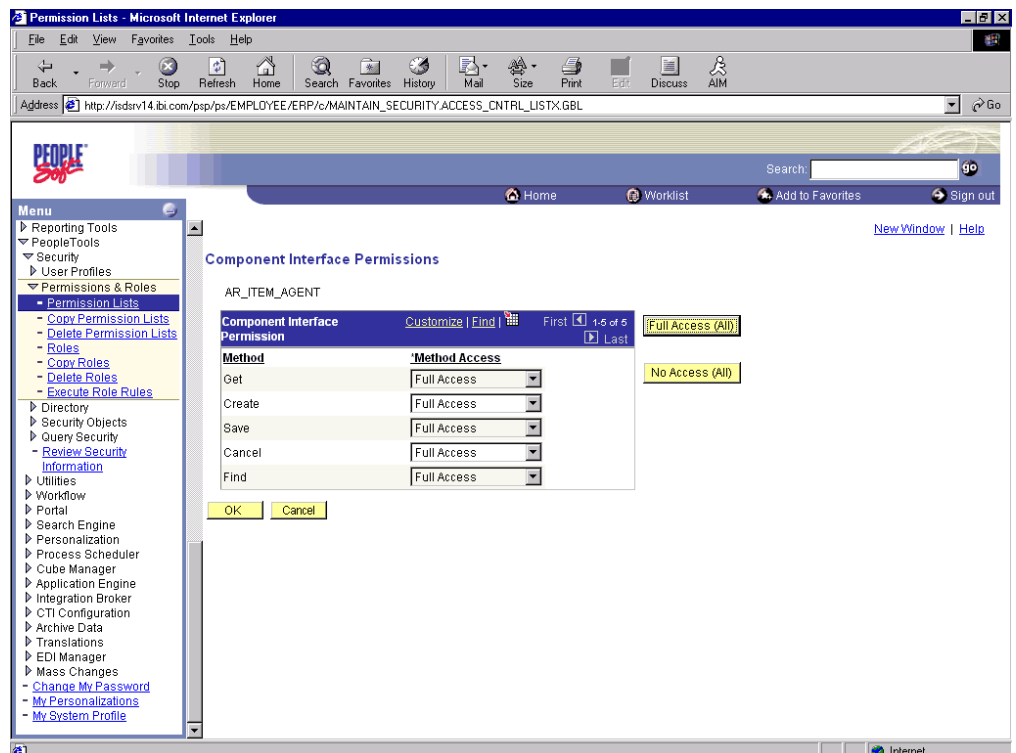
6. Click + to add a new row to the Component Interfaces list.

A field appears where you can enter the component interface name.



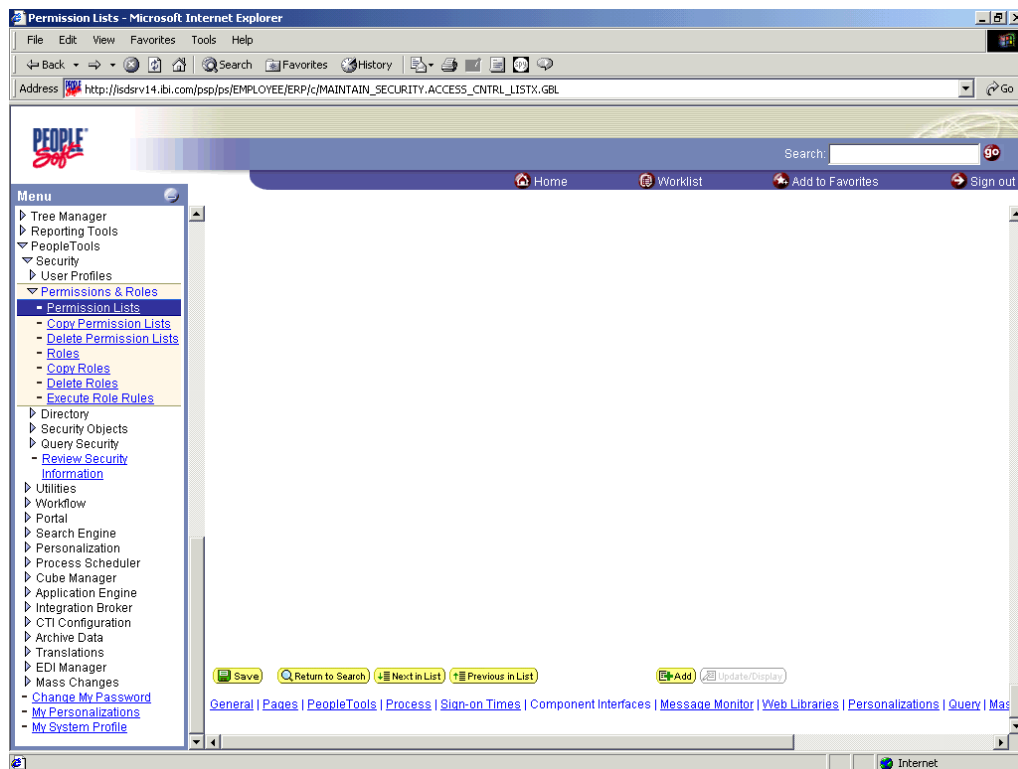
7. Enter the component interface name and click Edit.

This example uses the component interface AR_ITEM_AGENT.



8. From the lists, select the desired access level for each method.
9. Click **OK**.

The following pane is displayed.



10. Scroll down in the right pane and click **Save**.

C.1.3 Testing a Component Interface

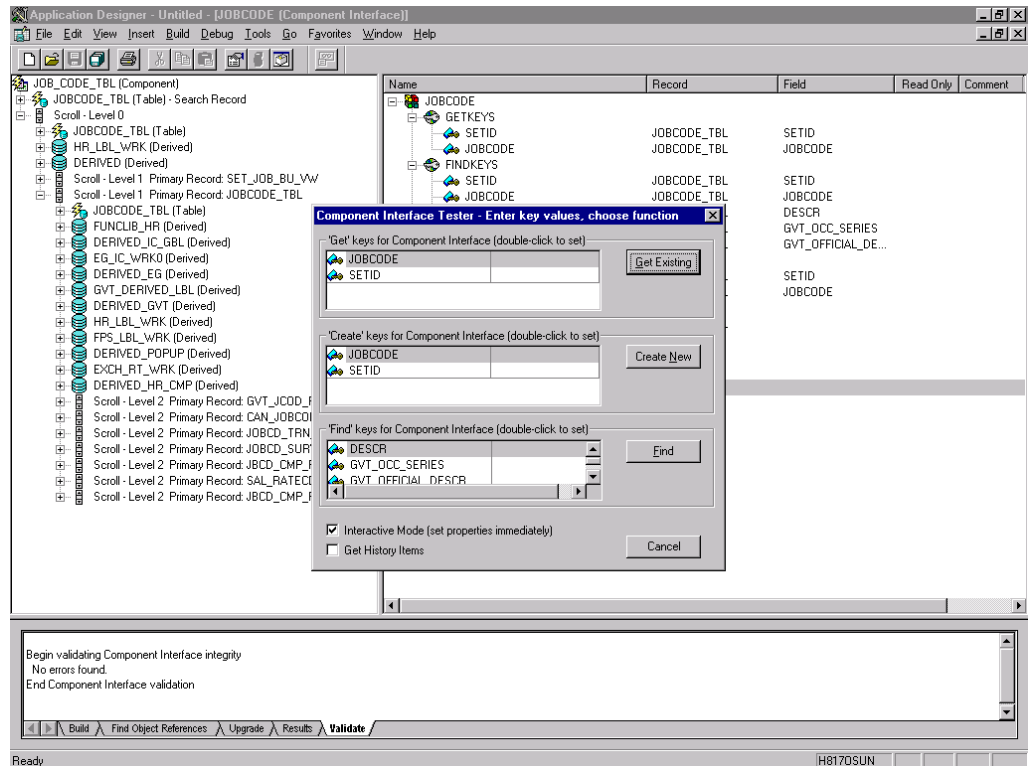
Oracle Application Adapter for PeopleSoft uses PeopleSoft metadata and component interfaces; therefore, it can accommodate new or modified component interfaces. The adapter makes no assumptions about component interfaces except that they are logical and valid. Each component interface must be tested before being used as a source for the adapter.

If changes are made to the underlying application by the user or by a PeopleSoft upgrade and the changes invalidate a component interface, the user must repair the invalid component interface before the adapter uses it.

Testing a Component Interface

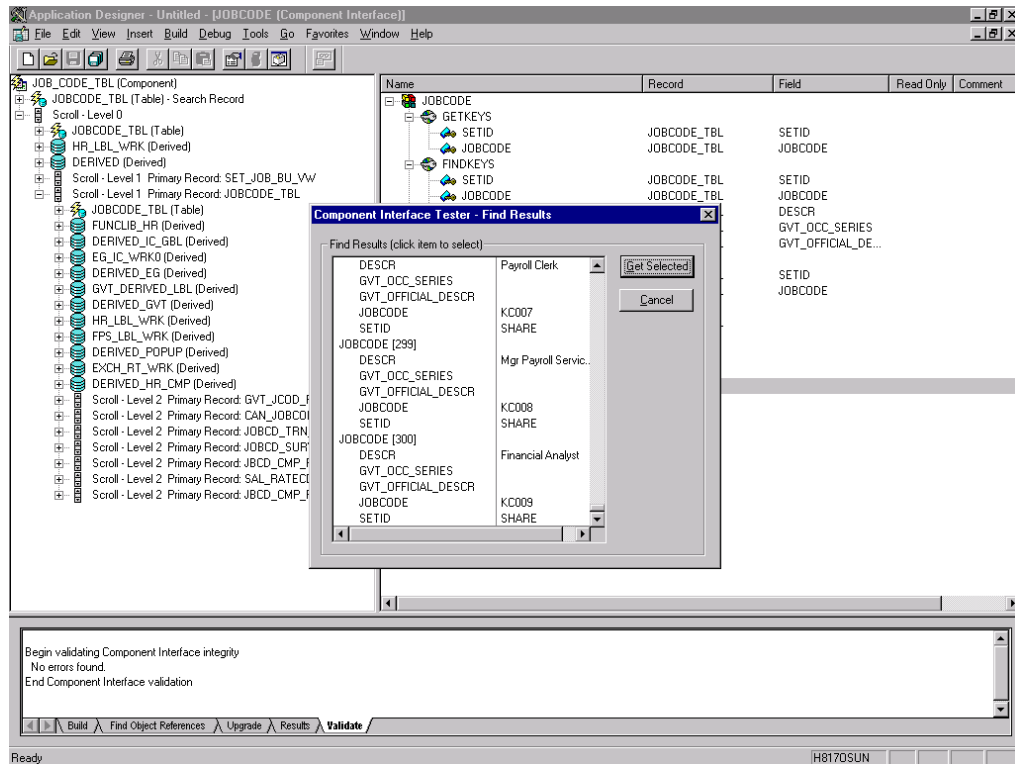
To test a component interface:

1. In Application Designer, select **Test Component Interface** from the **Tools** menu.
The Component Interface Tester dialog is displayed.

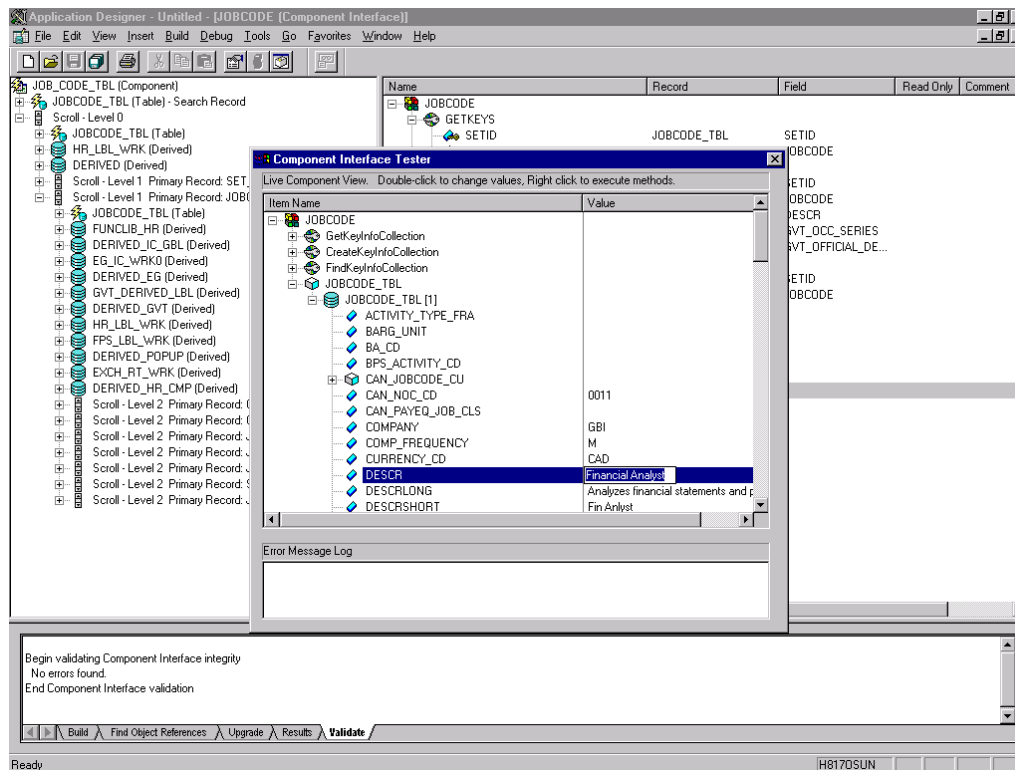


2. If required, click the Component Interface Tester dialog to bring it to the foreground.
3. To test the component interface, use one the following methods.
 - To test the component interface using the Find method, click **Find**.

The Component Interface Tester - Find Results dialog displays all of the possible entries for the underlying component. If there are more than 300 entries, a message appears.



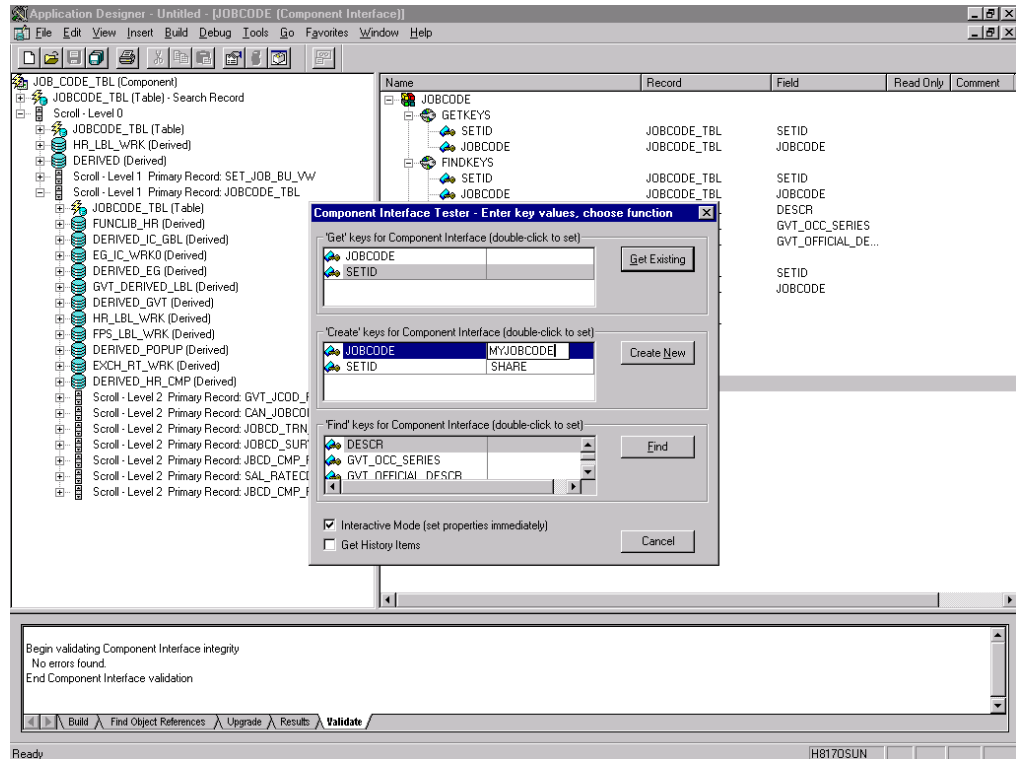
- In the left pane of the Find Results dialog, select a field.
 - To display the relevant data for that particular field, click **Get Selected**.
- The following dialog is displayed.



- If the security settings permit, you can change the values in the individual fields.
4. To test the component interface using the Get method, perform the following steps:
 - a. Enter the existing key(s).
 - b. Click **Get Existing**.

This returns the exposed properties for the key that you entered. You can change values if Update access was specified.

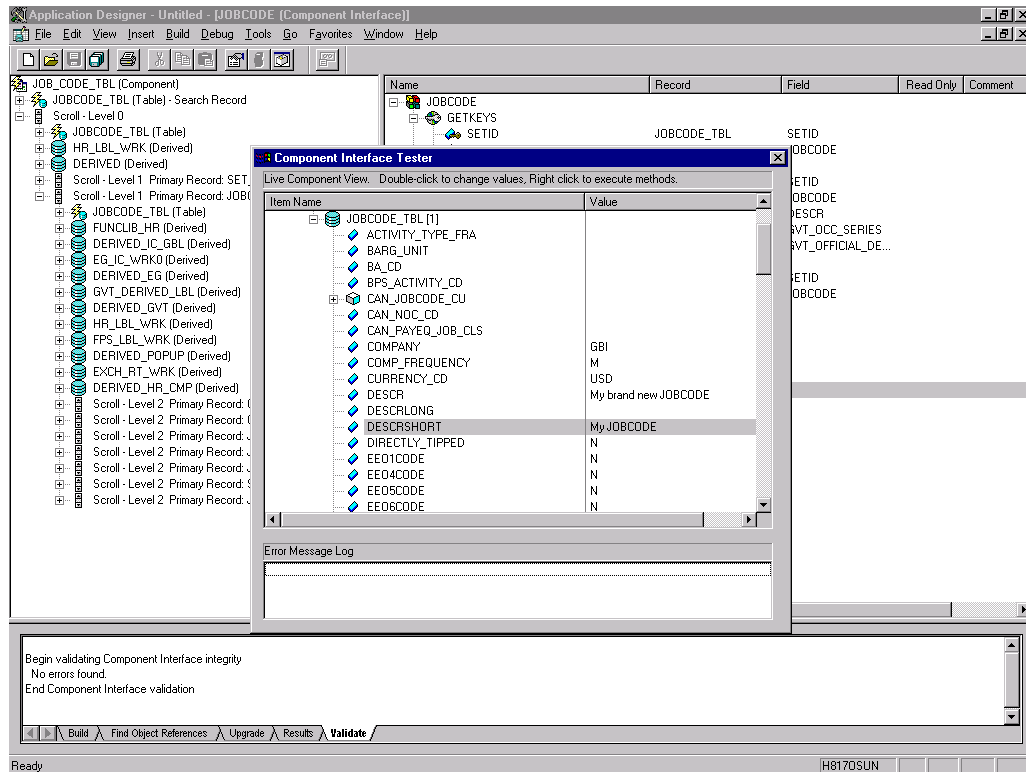
Alternatively, you can test using the Create method.



Perform the following steps:

- a. Enter all required key values.
- b. Click **Create New**.

When you enter valid values in Create keys, a pane showing the JOBCODE data is displayed after the Table name is expanded with default data in place.



You can change fields at this point. Changes are validated against the component's underlying business logic.

- c. After you finish making changes, right-click the top item in the pane.
5. To save your changes, click **Save**.

The keys used to create the record can be used with the Get method for viewing data. The data that was added can be viewed in the PeopleSoft Component as shown in the following example. The Effective Date is one of the default values.

You have finished testing the component interface. Before using the component interface, you must generate its API. See [Appendix A, "Generating Component Interface APIs"](#) for more information.

Using PeopleSoft Integration Broker

This appendix describes how to configure and test PeopleSoft Integration Broker (release 8.4) and PeopleSoft Application Messaging (release 8.1) using a PeopleSoft-supplied File Output interface. In PeopleSoft release 8.1, the messaging architecture is called Application Messaging and includes Application Messaging Gateway. In release 8.4, the messaging architecture is called Integration Broker, which includes Integration Gateway. When discussing release-generic issues, this section uses release 8.4 terminology. When discussing release-specific issues, it uses release-specific terminology.

Note: This section is not a substitute for PeopleSoft documentation. For more complete and up-to-date information on PeopleSoft Messaging and Integration Broker, see the *PeopleSoft Online Library* for your PeopleSoft system.

D.1 PeopleSoft Integration Broker

PeopleSoft Integration Broker provides a mechanism for communicating with the outside world using XML files. Communication can take place between different PeopleSoft applications or between PeopleSoft and third-party systems.

To subscribe to data, third-party applications can accept and process XML messages posted by PeopleSoft using the available PeopleSoft connectors or by adding a custom built connector to the Integration Gateway. This topic primarily covers publishing outbound asynchronous messages from a PeopleSoft system to a third-party application using the delivered File Output connector. For information on outbound synchronous messages, see "[Using Outbound Synchronous Messages](#)" on page D-23.

To send a message, you must properly configure various internal structures and processes. The following descriptions are generally release-generic. Detailed differences between releases 8.1 and 8.4 are discussed in other topics.

- Message

A Message is a container for the data that goes into the XML. It contains basic structural information, such as records and fields. The Message must be in an Active status to send the XML file

- Message Channel

The Message Channel is a mechanism for structuring records into logical groupings. Each Message can belong to only one Message Channel. The Message Channel must be in an Active (Run) status for the Message to be delivered.

In release 8.1, the Message Channel also provides preliminary routing instructions; you can specify which Message Nodes handle the message. Each Message Channel can route messages to multiple Message Nodes

- Message Node

Message node functionality changed from 8.1 to 8.4:

In release 8.1, the primary function of the Message Node is to specify which Gateway receives the messages.

In release 8.4, much of the "intelligence" that was built into the Message Channel moved to the Message Node. This provides additional flexibility over release 8.1. You can specify which messages the Message Node can handle. In addition, the Gateway Connector is bound to the Message Node. Each Message Node can route messages to only one Connector.

- Integration Gateway

The Integration Gateway is a program that runs on the PeopleSoft Web Server. It is the physical hub between PeopleSoft and the third-party system.

- Target Connector/Handler

Connectors are Java programs that run under the control of the Integration Gateway and control the final output destination of the XML file. PeopleSoft release 8.4 comes with several connectors including HTTP, FTP, SMTP, JMS, POP3, and a `Simple File` connector that places the file in a directory on the Web Server. This section discusses the Simple File connector.

- PeopleCode

PeopleCode is the programming tool provided with PeopleTools that enables you to create complex application functionality. A message can only be initiated using specific PeopleCode instructions. This code is typically triggered by an application event, such as creating a new database entry through an online panel or through a batch job.

Most of the examples in this section use the `LOCATION_SYNC` message, which is a PeopleSoft Enterprise Integration Point (EIP) and is supplied with most PeopleSoft applications. If `LOCATION_SYNC` is not part of your package, you may use any supplied message.

D.1.1 Configuring Integration Broker in PeopleSoft 8.4

You can configure PeopleSoft 8.4 to send an asynchronous outbound message to the File Output connector.

To configure application messaging in PeopleSoft 8.4:

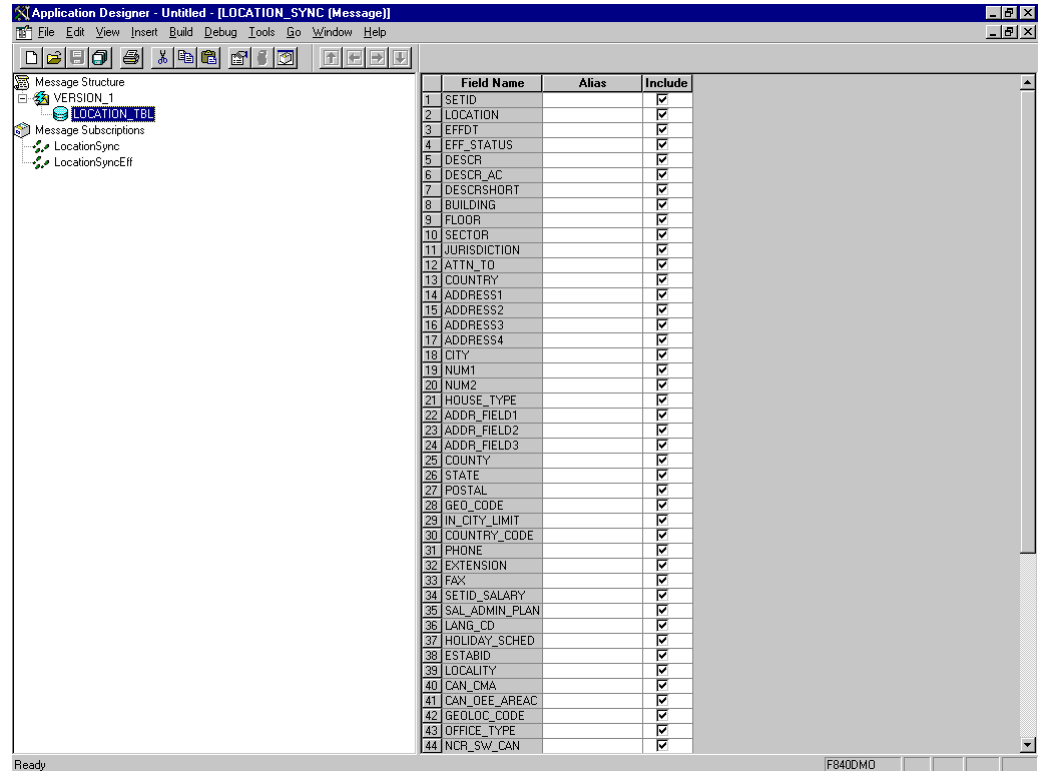
1. Ensure that the message is active and is routed to the proper Message Channel.
2. Configure the Message Channel.
3. Configure the `IntegrationGateway.properties` file to communicate with your PeopleSoft 8.4 application.
4. Configure the Integration Gateway and File Output connector.
5. Create and configure a new Gateway node.

These tasks are described in detail in the following procedures.

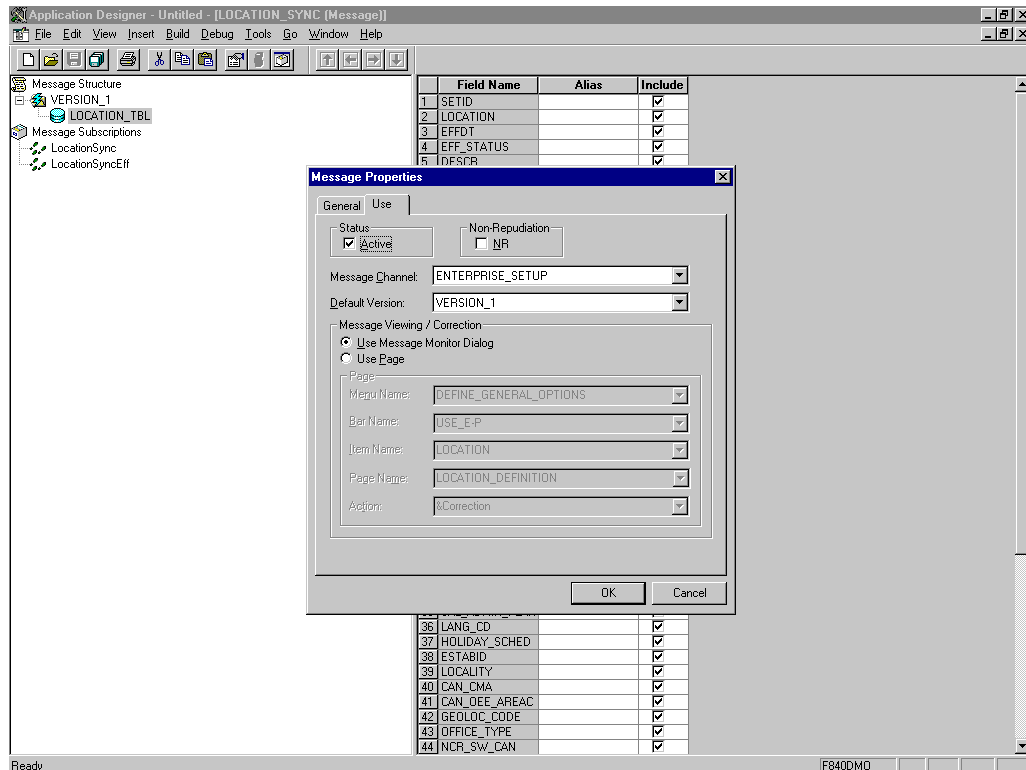
Ensuring the Message Is Active and Is Routed Correctly

To ensure that the message is active and is routed to the proper Message Channel:

1. Open Application Designer.
2. On the **File** menu, point to **Open**, click **Message**, and then open the **LOCATION_SYNC** message.



3. To view the fields that are included in the Message, highlight **LOCATION_TBL**.
4. Right-click **LOCATION_TBL** and select **Properties**.
The Message Properties dialog is displayed.



5. Select the **Use** tab.
6. Ensure the **Active** check box is selected.

The message is routed to the Message Channel, ENTERPRISE_SETUP, and the default message version is VERSION_1 (messages can have multiple versions).

7. Click **OK** and then save the message.

You have finished ensuring that the message is active and is routed correctly.

Configuring the IntegrationGateway.properties File

To configure the IntegrationGateway.properties file:

1. Open the IntegrationGateway.properties file using the editor of your choice.
2. Find the section of the file that specifies the JOLT connect string setting for the default application server. This is usually near line 75, and looks similar to the following:

```
## JOLT connect string setting for optional Default Application Server. Do NOT
specify a NODENAME.
#
# Example:
#ig.isc.serverURL=//MYSERVER:9000
#ig.isc.userid=MYUSERID
#ig.isc.password=MYPASSWORD
#ig.isc.toolsRel=8.40
```

3. Uncomment (or copy and uncomment) the four lines that specify the connection.
4. Enter the appropriate information.

In the following example, the tools release is 8.40.09:

```
ig.isc.serverURL=//isdsv14:9000
ig.isc.userid=VP1
ig.isc.password=VP1
ig.isc.toolsRel=8.40.09
```

The PeopleSoft tools release must be precise to the last decimal.

Note: With release 8.42, the password must be stored in an encrypted format. PeopleSoft provides a script called `PSCipher.bat` (`PSCipher.sh` on UNIX) to accomplish encryption. Typically, this script is located in the path of the `IntegrationGateway.properties` file. Follow the instructions supplied by PeopleSoft to run this script.

You have finished configuring the `IntegrationGateway.properties` file.

Configuring the Integration Gateway and the File Output Connector

To configure the Integration Gateway and the File Output Connector:

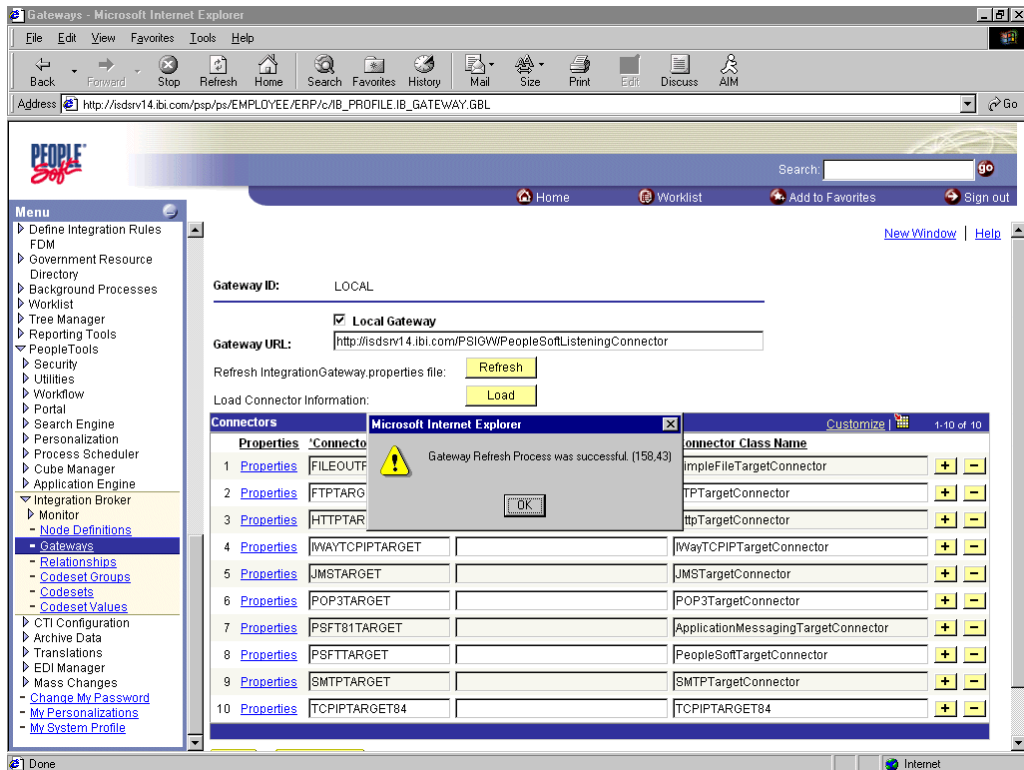
1. In a Web browser, open your PeopleSoft 8.4 application in 4-tier mode.
2. In the Menu pane, expand **PeopleTools, Integration Broker**, and then click **Gateways**.
3. Open the LOCAL Gateway ID and enter the following Gateway URL:

```
host name/PSIGW/PeopleSoftListeningConnector
```

Where `host name` is the name of your PeopleSoft Web Server.

4. Click **Refresh**.

A message appears stating the outcome of the refresh process.



5. Click **OK** and scroll down to click **Save**.

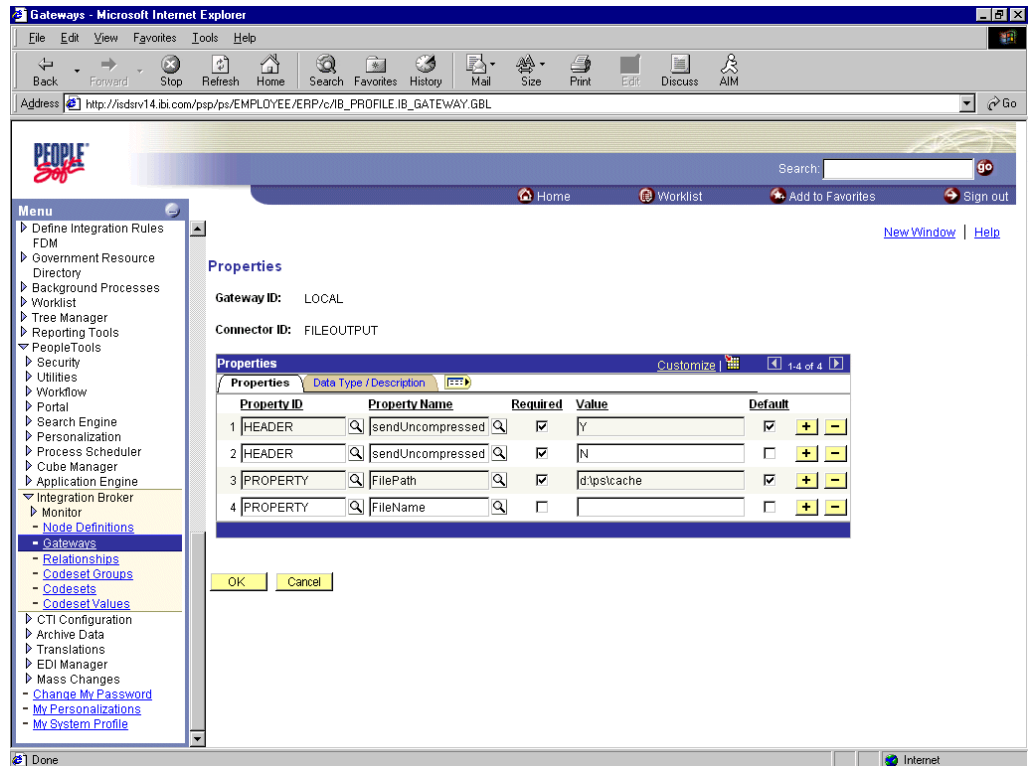
You must click Save before continuing.

6. Click the **Properties** link for the FILEOUTPUT Connector ID.

The Properties window for the FILEOUTPUT Connector is displayed.

7. Accept or overwrite the default values.

In the following figure, the FilePath PROPERTY from the c:\temp default was changed to d:\ps\cache.



8. To return to the Gateway window, click **OK**.

9. Scroll down and click **Save**.

You have finished configuring the Integration Gateway and the File Output Connector.

Creating and Configuring a New Gateway Node

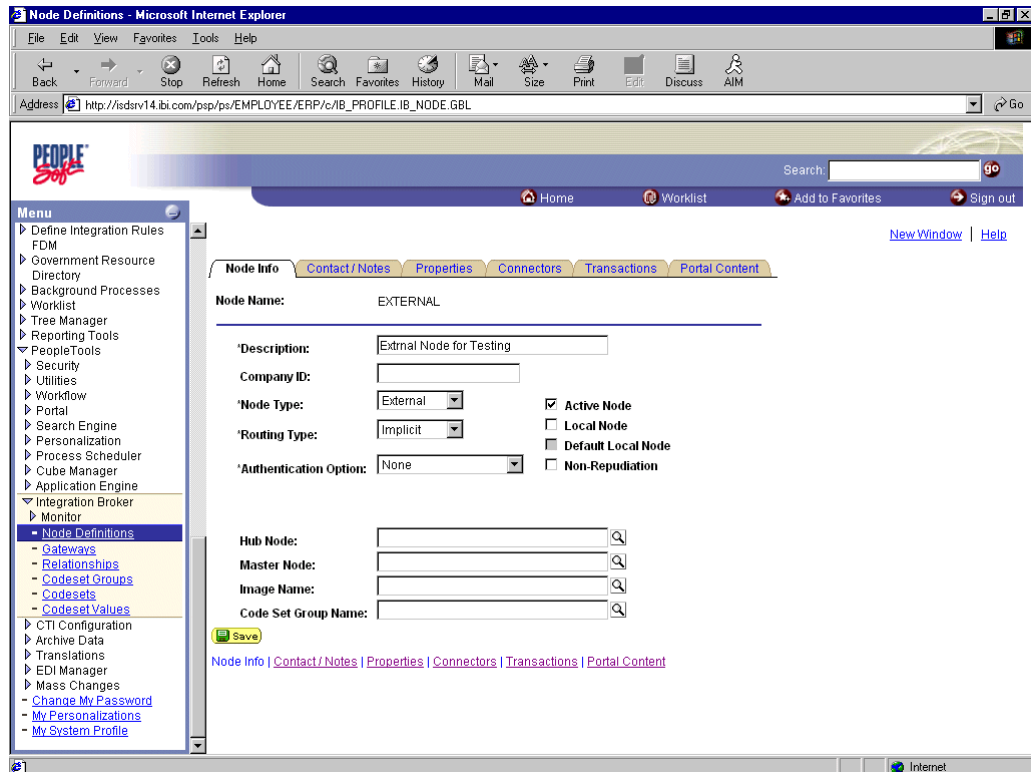
To create and configure a new Gateway Node:

1. In the Menu pane, expand **PeopleTools**, **Integration Broker**, and then, click **Node Definitions**.
2. Select the **Add a New Value** tab.
3. In the **Node Name** field, enter a node name.

It is recommended that you name your first (trial) message node **EXTERNAL**. After successfully configuring and sending messages using this node, you can create additional message nodes with names appropriate for your application.

4. Click **Add**.

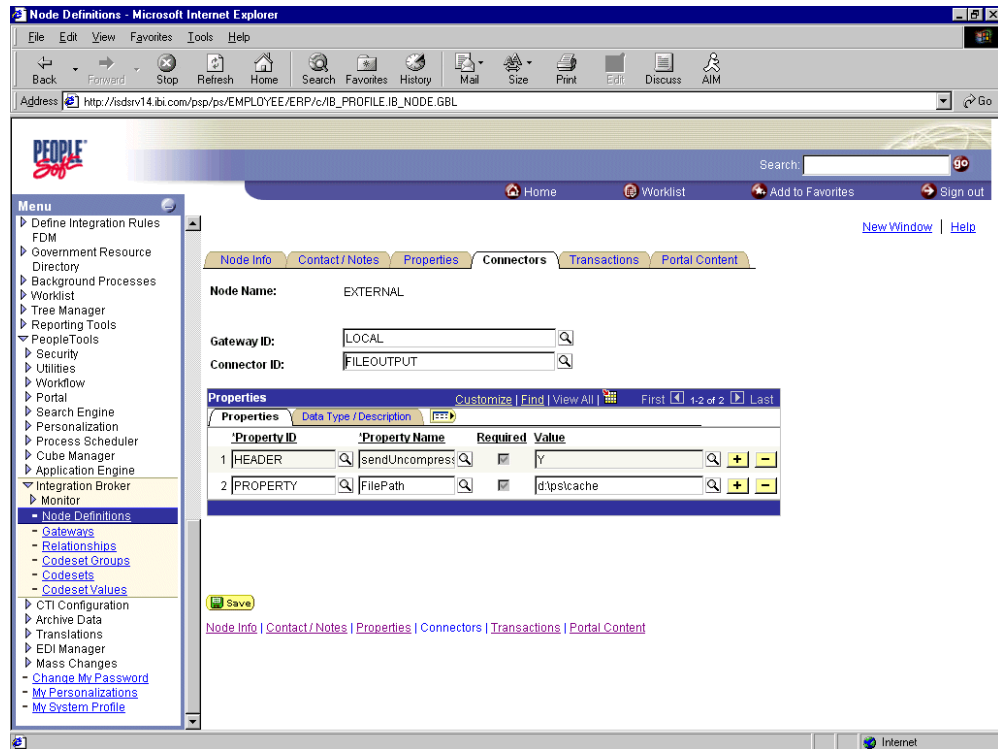
The Node Info tab becomes available.



Perform the following steps:

- a. In the **Description** field, enter an appropriate description.
 - b. From the **Node Type** list, select **EXTERNAL**.
 - c. From the **Routing Type** list, select **Implicit**.
5. Select the **Connectors** tab.

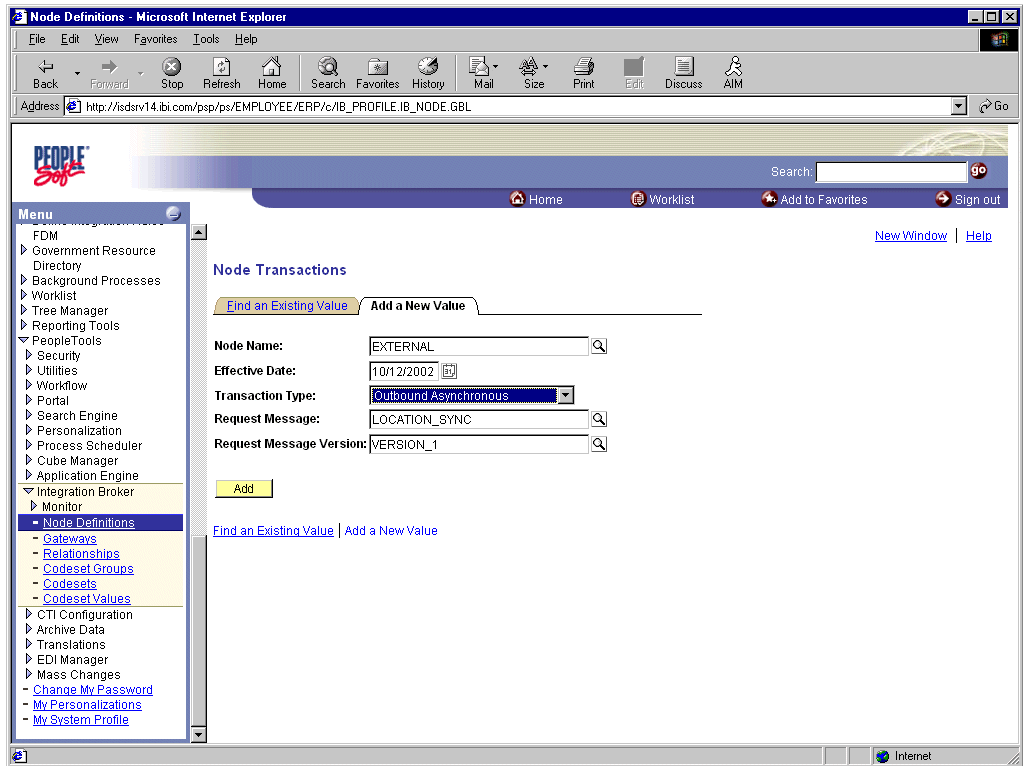
The **Connectors** tab becomes available.



Perform the following steps:

- a. Specify **LOCAL** for the **Gateway ID**.
 - b. Specify **FILEOUTPUT** for the **Connector ID**.
 - c. Accept or overwrite the default Gateway property values.
6. Click **Save**.
 7. To specify the transactions to route messages to your node, select the **Transactions** tab and click **Add Transaction**.

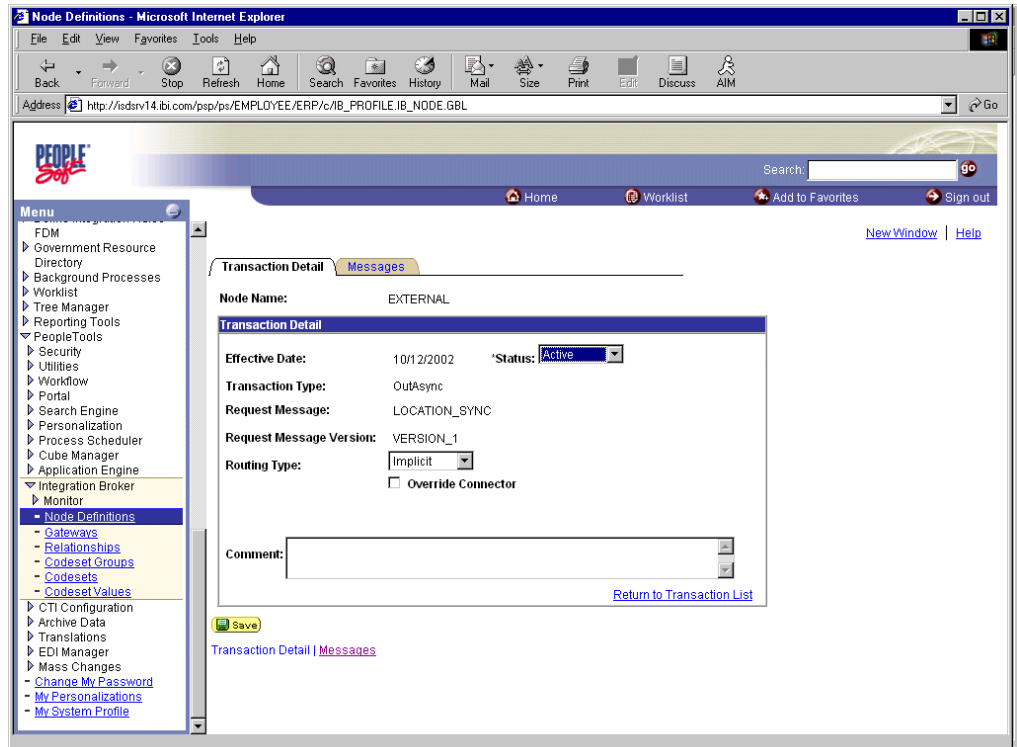
The Node Transactions pane is displayed.



Perform the following steps:

- a. From the **Transaction Type** list, select **Outbound Asynchronous**.
 - b. In the **Request Message** field, specify **LOCATION_SYNC**.
 - c. In the **Request Message Version** field, specify **VERSION_1**.
- In the **Request Message Version** field, specify **VERSION_1**.
8. Click **Add**.

The Transaction Detail pane is displayed.



Perform the following steps:

- a. Verify that the **Routing Type** is **Implicit**.
- b. Click **Save**, then click the **Return to Transaction List** link.
- c. To ensure that your data entry is not lost, click **Save** again.

You have finished creating and configuring the new Gateway Node.

See "[Viewing the PeopleCode for a Message](#)" on page D-25 for more information.

D.2 Configuring Application Messaging in PeopleSoft Release 8.1

You can configure PeopleSoft 8.1 to send an asynchronous outbound message to the Simple File Handler.

To configure application messaging in PeopleSoft 8.1:

1. Create and configure a new Message Node.
2. Ensure the message is active and is routed to the proper Message Channel.
3. Configure the Message Channel.
4. Configure the Simple File Handler in the Gateway.

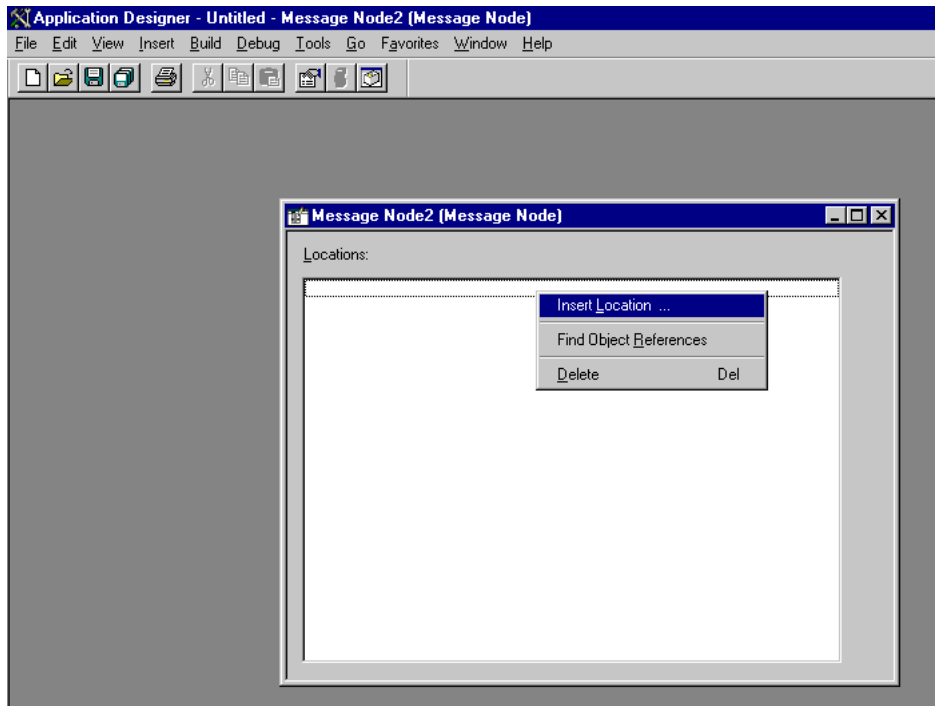
These tasks are described in detail in the following procedures.

Creating and Configuring a New Message Node

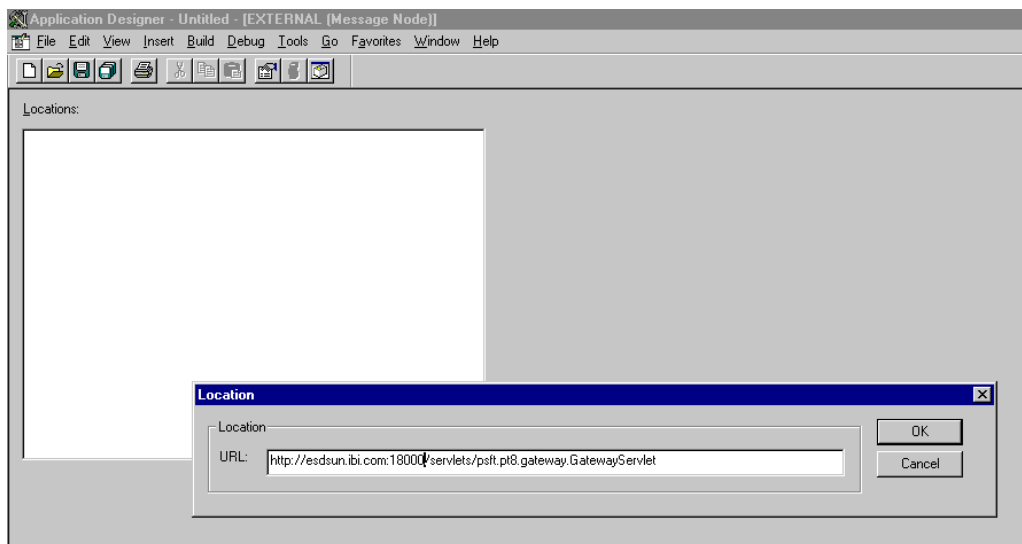
To create and configure a new message node:

1. Select **New** from the File menu and click **Message Node**.

A Message Node window is displayed.



2. Right-click anywhere inside the white space and select **Insert Location**.
The Location URL box is displayed.

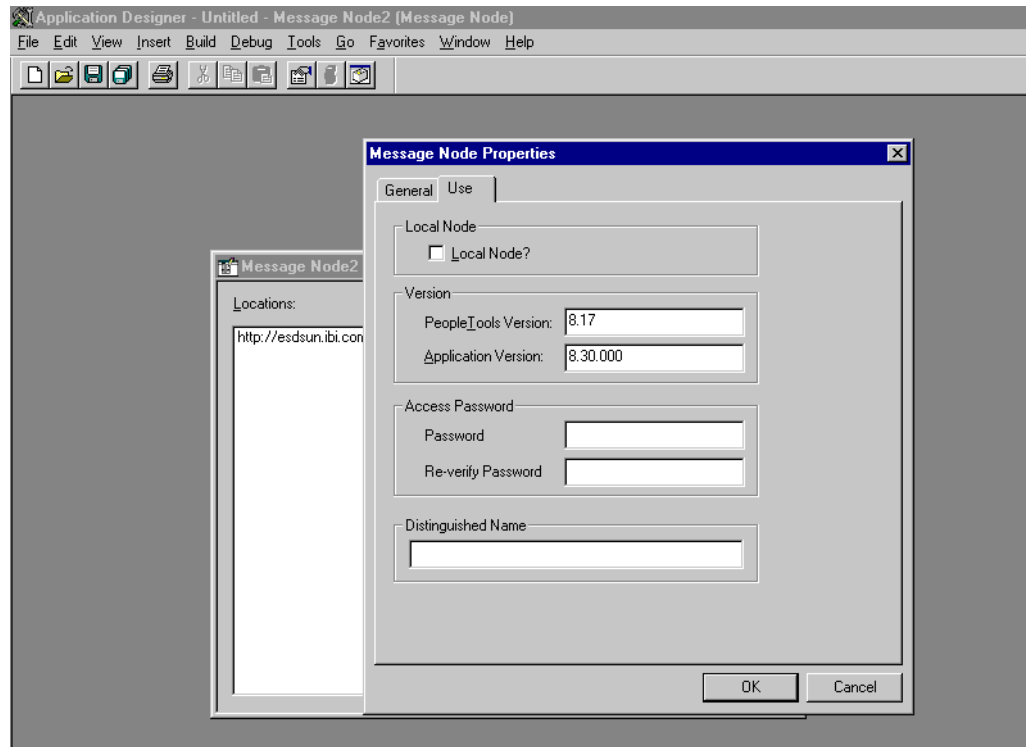


3. Enter the following URL for the PeopleSoft Application Gateway (handler directory):

host name:port/servlets/psft.pt8.gateway.GatewayServlet

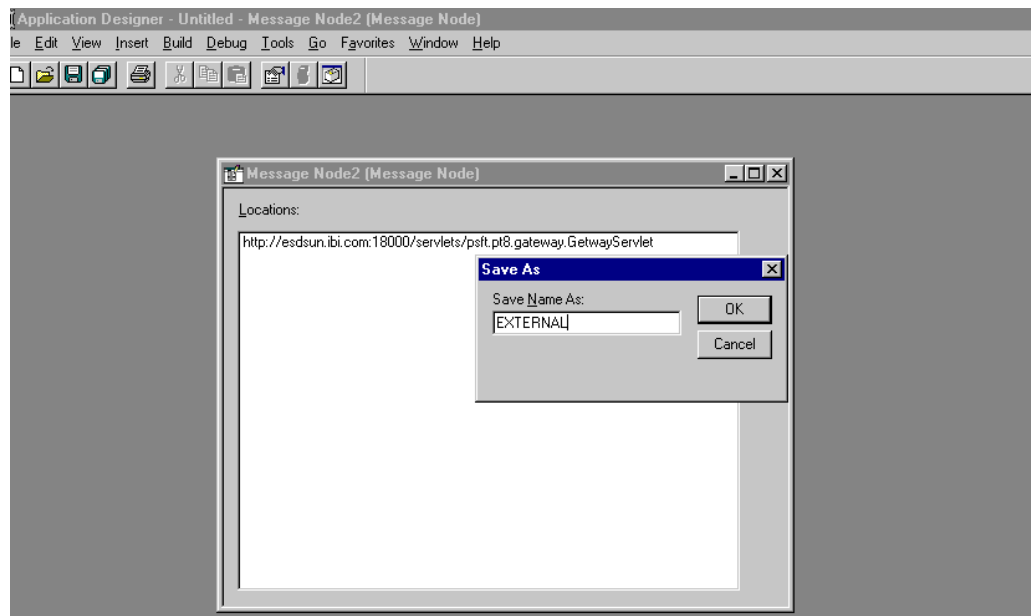
Where *host name* is the host name of your PeopleSoft Web server and *port* is the socket on which the server is listening. The characters you enter after *host name* are case-sensitive.

4. Click **OK**.
5. Display the Message Node Properties dialog.



Perform the following steps:

- a. Select the **Use** tab.
 - b. In the text boxes, enter the PeopleTools and Application Version numbers.
 - c. Click **OK**.
6. Display the Save As dialog.



7. To save the Message Node, click **OK**.

It is recommended that you name your first (trial) message node EXTERNAL. After successfully configuring and sending messages using this node, you can create additional message nodes with names appropriate for your application.

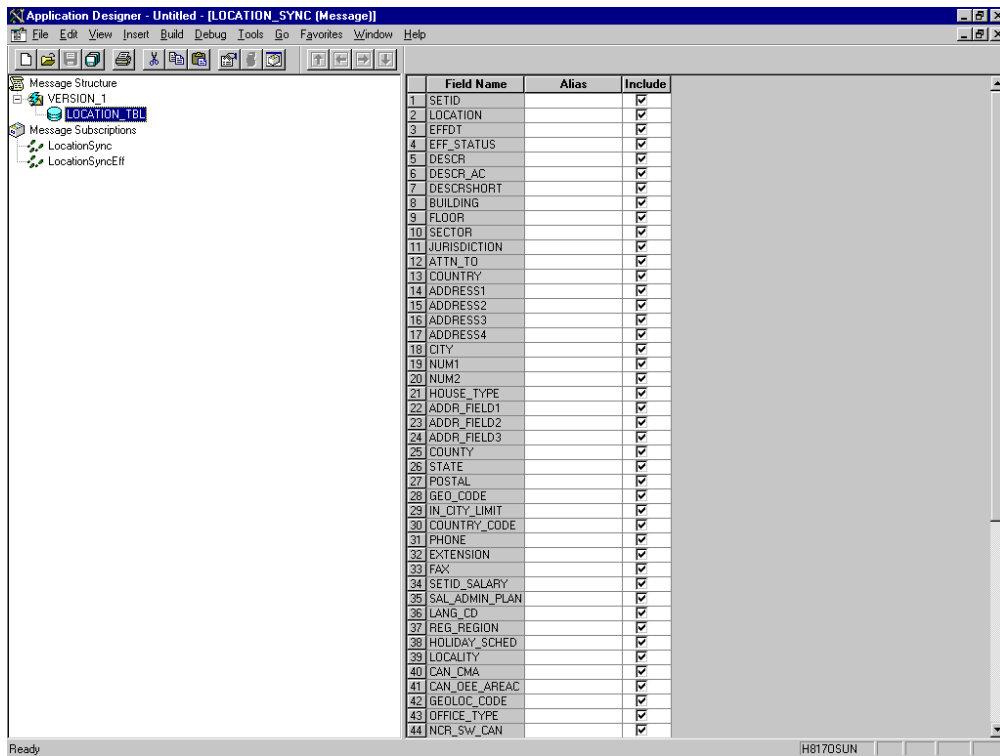
If you intend to migrate this message node to a different PeopleSoft environment (for example, from Test to QA), you can create a PeopleSoft project and insert the Message Node into the project.

You have finished creating and configuring the message node.

Ensuring the Message Is Active and Is Routed Correctly

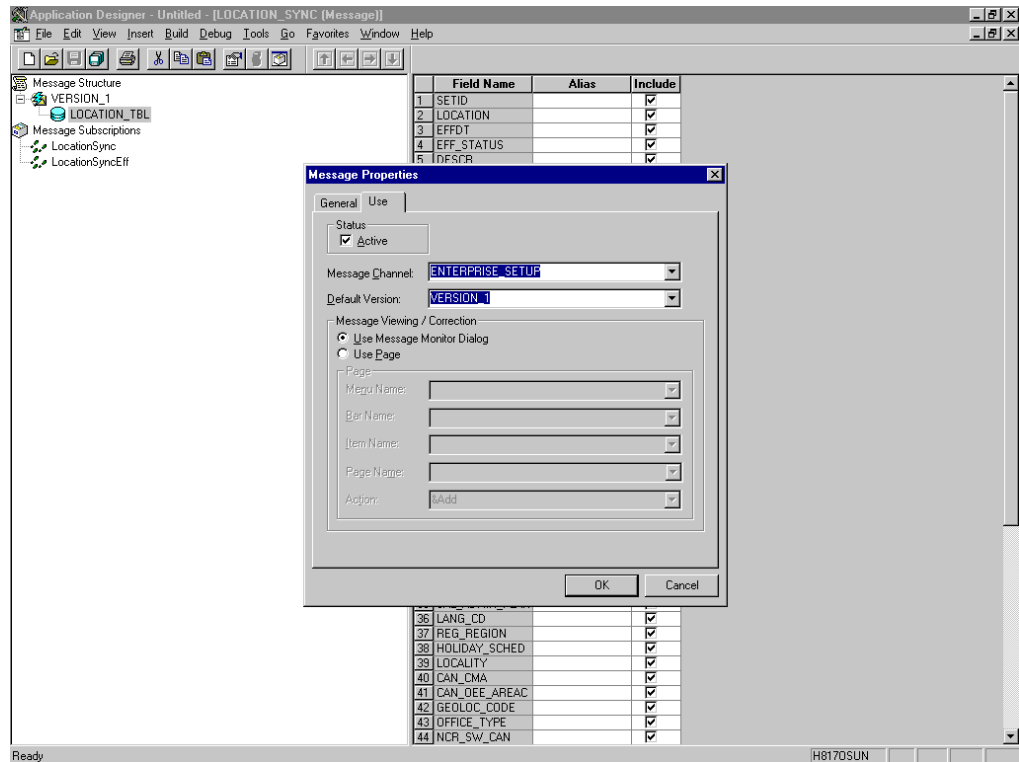
To ensure that the message is active and is routed to the proper message channel:

1. Open Application Designer.
2. On the **File** menu, point to **Open**, click **Message**, and open the **LOCATION_SYNC** message.



3. To view the fields that are included in the message, highlight **LOCATION_TBL**.
4. Right-click **LOCATION_TBL** and select **Properties**.

The Message Properties dialog is displayed.



Perform the following steps:

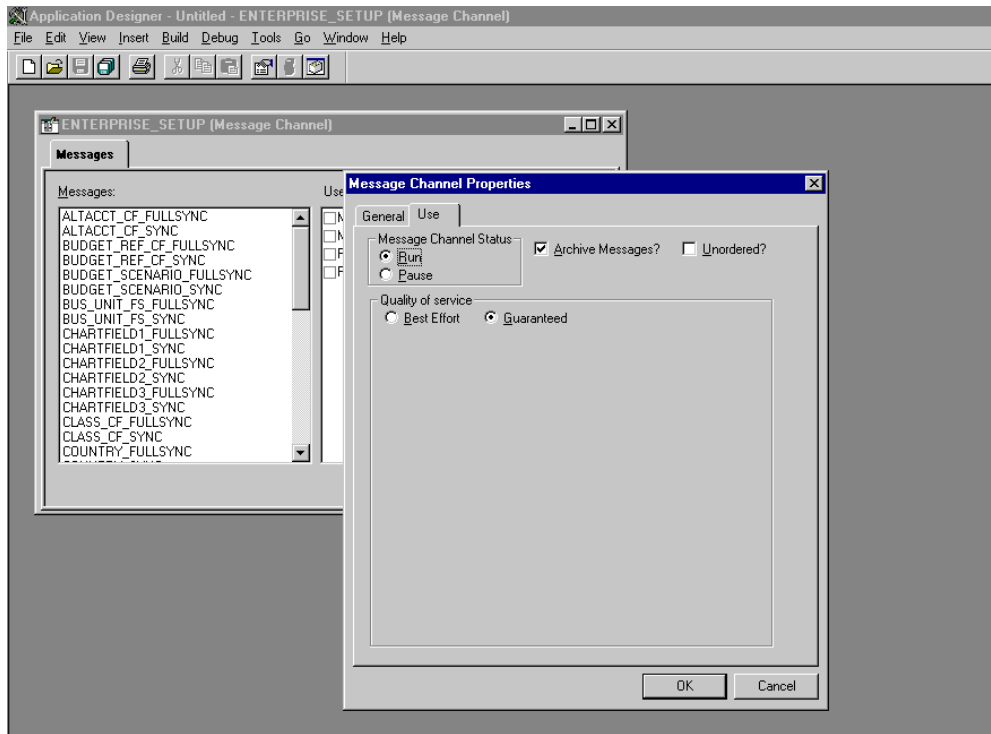
- a. Select the **Use** tab.
 - b. Ensure the **Status** check box is selected, which indicates that the message is active.
 - c. From the Message Channel list, select **ENTERPRISE_SETUP**.
 - d. From the Default Version list, select **VERSION_1** (messages can have multiple versions).
5. Click **OK**.
 6. Save the message.

Configuring the Message Channel

To configure the message channel:

1. From the **File** menu, choose **Open** and click **Message Channel**.
2. To open the **ENTERPRISE_SETUP** Message Channel, select **ENTERPRISE_SETUP**.
3. Right-click **ENTERPRISE_SETUP** and select **Properties**.

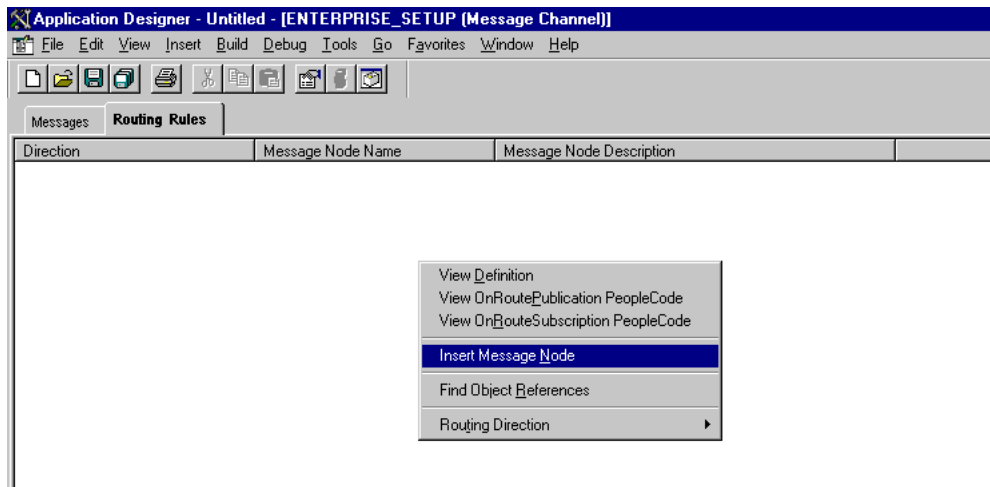
The Message Channel Properties dialog is displayed.



Perform the following steps:

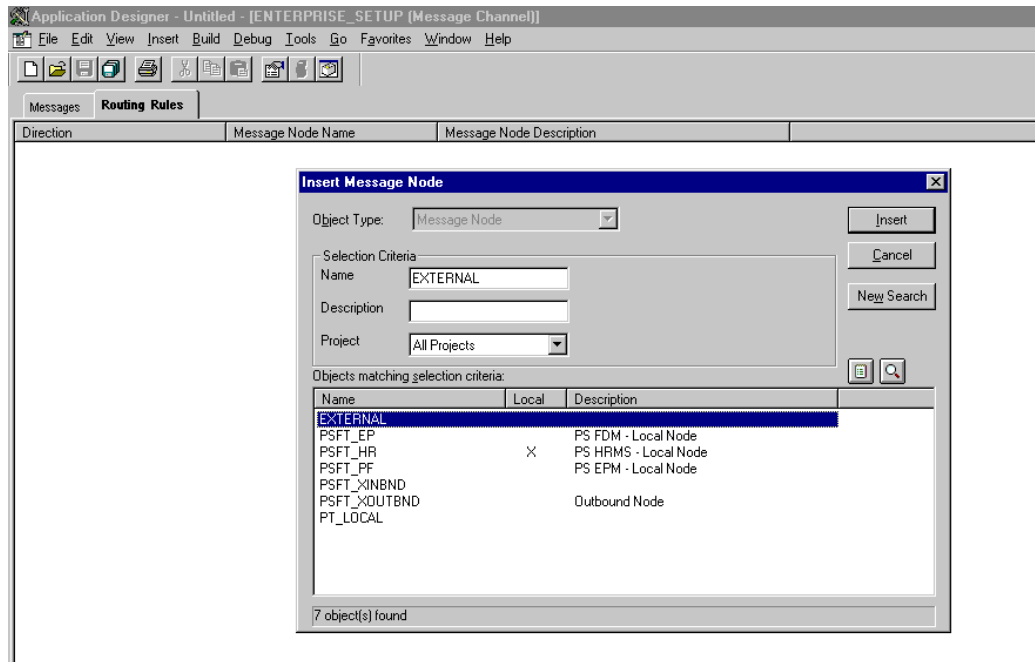
- a. Select the **Use** tab.
 - b. Ensure that the Message Channel status is set to **Run**.
 - c. Click **OK**.
4. From the left pane, select the **Routing Rules** tab.

The pane is blank.



5. Right-click the pane and select **Insert Message Node**.

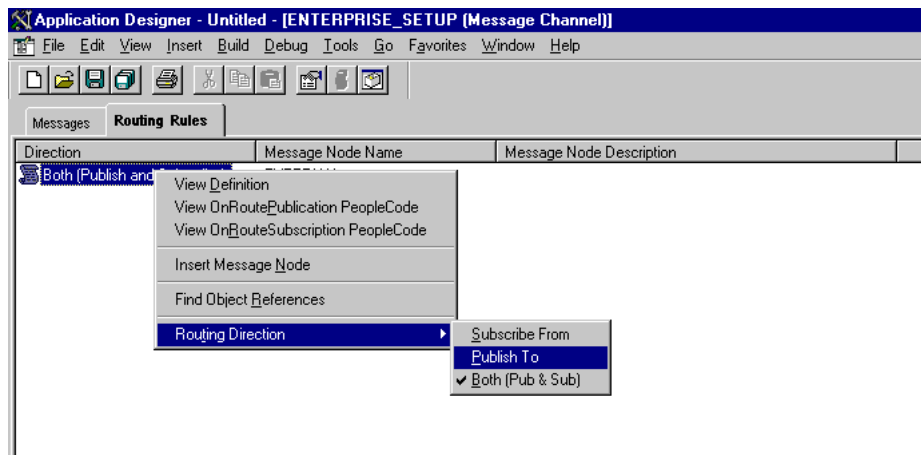
The Insert Message Node dialog is displayed.



Perform the following steps:

- a. Select the message node that you created in "Creating and Configuring a New Message Node" on page D-11, for example, EXTERNAL.
 - b. Click **Insert**.
6. Click **Cancel**.

Information appears on the Routing Rules tab.



Perform the following steps:

- a. Right-click the message node and point to **Routing Direction**.
 - b. From the **Routing Direction** menus, select **Publish To**.
7. Save the Message Channel, and if you require it, place it in your project.
- You have finished configuring the Message Channel.

Configuring the Simple File Handler in the Gateway

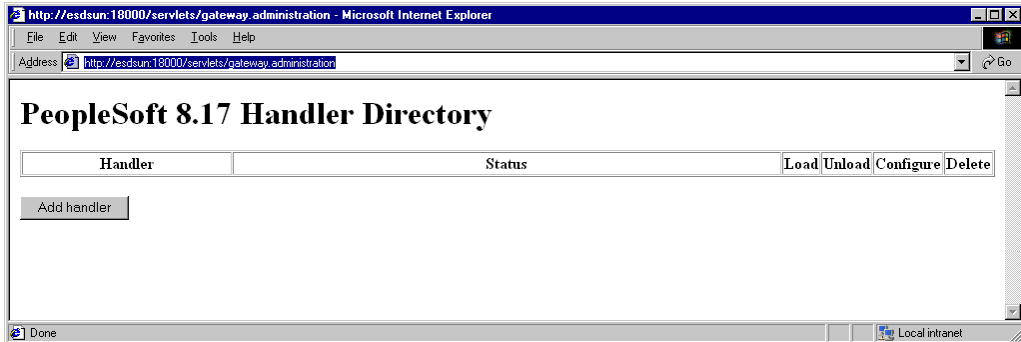
To configure the Simple File Handler in the Gateway:

1. In a Web browser, launch the PeopleSoft 8.1 configuration servlet interface (also known as the server gateway) by entering the following URL:

host name:port/servlets/gateway.administration

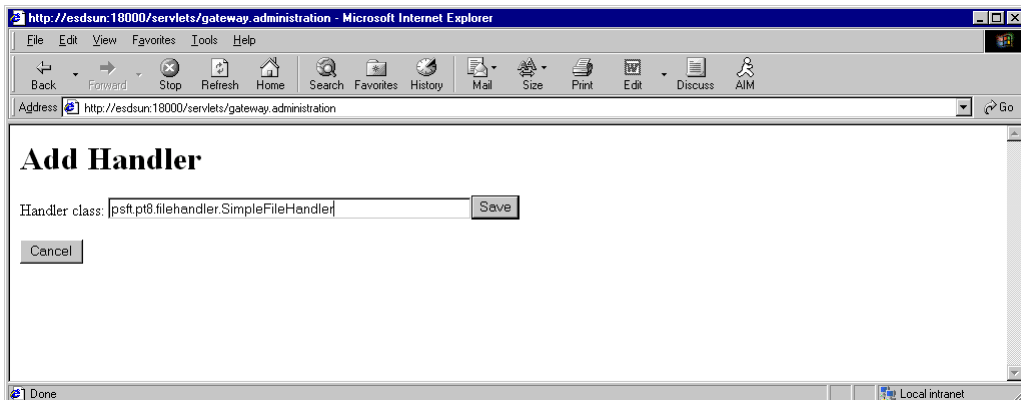
Where *host name* is the name of the application server that hosts PeopleSoft and *port* is the port number on which the application server is listening.

The Handler Directory window is displayed.



2. Click **Add Handler**.

The Add Handler window is displayed.

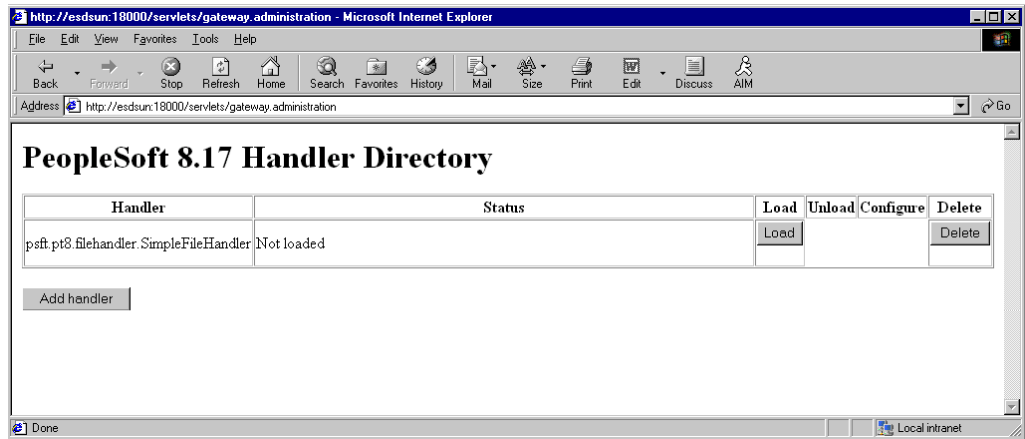


3. Enter the full name of the Simple File Handler class, `psft.pt8.filehandler.SimpleFileHandler`.

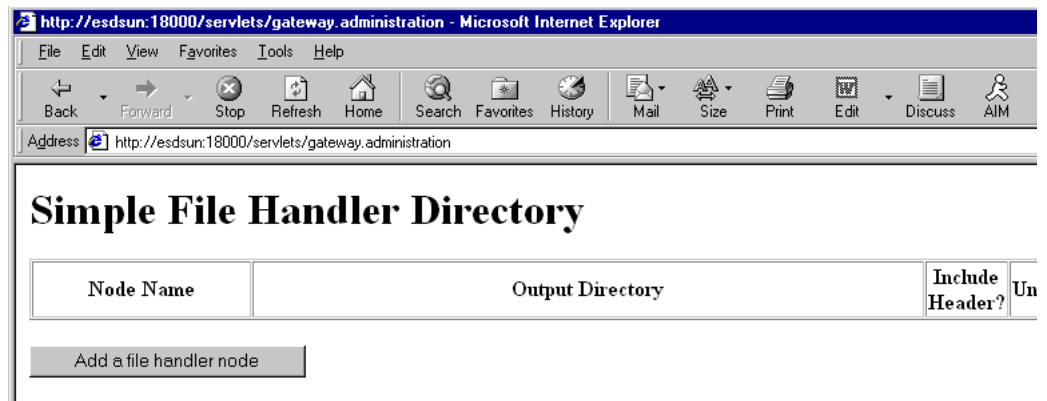
Note: The name is case-sensitive.

4. Click **Save**.

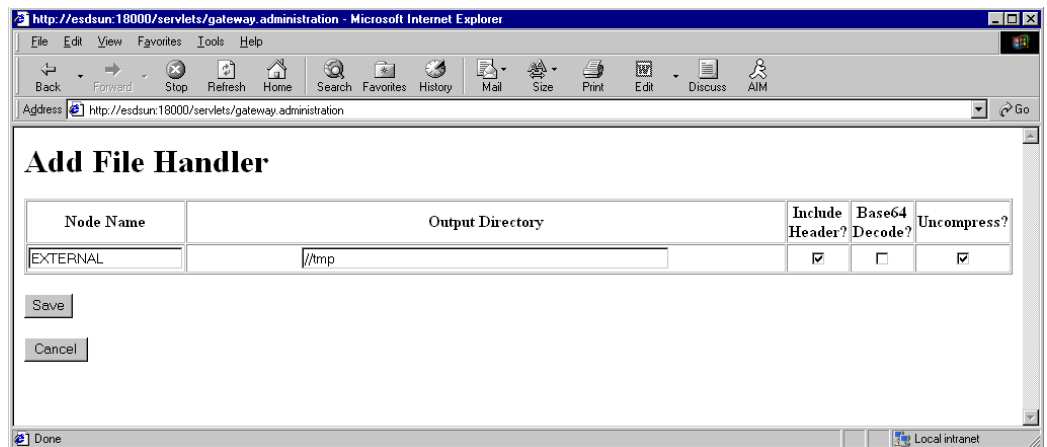
The Handler Directory window is displayed.



5. To load the handler, click **Load**.
After the handler loads, "Loaded successfully" appears in the Status column.
6. Click **Configure**.
The Simple File Handler Directory window is displayed.



7. Click **Add a file handler node**.
The Add File Handler window is displayed.



8. In the Node Name field, enter the name of the message node that you created in "Creating and Configuring a New Message Node" on page D-11, for example, EXTERNAL.

D.3 Viewing the PeopleCode for a Message

Messages are initiated by the PeopleCode that is attached to a record. Usually, this record is one of the records associated with the message itself.

1. Open Application Designer.

Num	Field Name	Type	FDe	FEd	FCh	FFo	RIn	RIs	RDc	RSe	SEd	SPr	SPo	Srl	SrS	Wk	PPr
1	SETID	Char															
2	LOCATION	Char											<input checked="" type="checkbox"/>				
3	EFFDT	Date															
4	EFF_STATUS	Char									<input checked="" type="checkbox"/>						
5	DESCR	Char										<input checked="" type="checkbox"/>					
6	DESCR_AC	Char															
7	DESCRSHORT	Char															
8	BUILDING	Char															
9	FLOOR	Char															
10	SECTOR	Char															
11	JURISDICTION	Char															
12	ATTN_TO	Char															
13	ADDRESS_SBR	SRec															
14	PHONE_SBR	SRec															
15	EXTENSION	Char															
16	FAX	Char															
17	SETID_SALARY	Char															
18	SAL_ADMIN_PLAN	Char															
19	LANG_CD	Char															
20	REG_REGION	Char															
21	HOLIDAY_SCHEDULE	Char															
22	LOC_TBLUSA_SBR	SRec															
23	LOC_TBLCAN_SBR	SRec															
24	WRKS_CNCL_SBR	SRec															
25	LOC_TBLGER_SBR	SRec															
26	LOC_TBLUK_SBR	SRec															
27	LOC_TBL_FED_SBR	SRec															
28	LOC_TBLESP_SBR	SRec															
29	LABEL_FORMAT_ID2	Char															
30	LABEL_FORMAT_ID3	Char															
31	USG_LBL_FORMAT_ID	Char															
32	LOC_TBLMEX_SBR	SRec															
33	ESTABID	Char									<input checked="" type="checkbox"/>						
34	COMMENTS_2000	Long															

2. On the **Record Fields** tab, select the **LOCATION_TBL** record.
3. Select the PeopleCode display option.
4. Select the **Save Post Change (SPo)** box for the **LOCATION** field.

The following window displays the PeopleCode that initiates a LOCATION_SYNC message.

```

Application Designer - Untitled - [LOCATION_TBL.LOCATION.SavePostChange (Record PeopleCode)]
File Edit View Insert Build Debug Tools Go Favorites Window Help

LOCATION (field) SavePostChange

Local Message <MSG;
Local Rowset <RS0, <RS01, <RS11;
Local Row <R1;
PanelGroup string <PubNodeName;
PanelGroup boolean <ActionCodeRowAdd;

If ActiveRowCount() = CurrentRowNumber() Then
  <MSG = CreateMessage(Message.LOCATION_SYNC);

  If <MSG.IsActive Then

    /* If the program is called by a Component Interface, then flush the extra row (Created by Component Create(), CopySetupRowset on
    Component Interface "LOCATION") from panel buffer.*/
    /**=====*/

    If ((%ComponentName = Component.LOCATION) And
        (<ActionCodeRowAdd = True)) Then

      <RS0 = GetLevel0();
      <RS01 = CreateRowset(<RS0);
      <RS0.CopyTo(<RS01);
      <RS11 = <RS01(1).GetRowset(Scroll.LOCATION_TBL);
      <RS11.Flush();

      <R1 = <RS0(1).GetRowset(Scroll.LOCATION_TBL).GetRow(1);
      <R1.CopyTo(<RS11.GetRow(1));

      <RS01.CopyTo(<RS0);

    End-If;
    /**=====*/

    <MSG.CopyRowsetDelta(GetLevel0() (1).GetRowset(Scroll.LOCATION_TBL));

    /* prevent circular publishes, do not publish back to originating node */
    If All(<PubNodeName) Then
      <MSG.DoNotPubToNodeName = <PubNodeName;
    End-If;

    <MSG.Publish();

  End-If;
End-If;
Ready H8170SUN

```

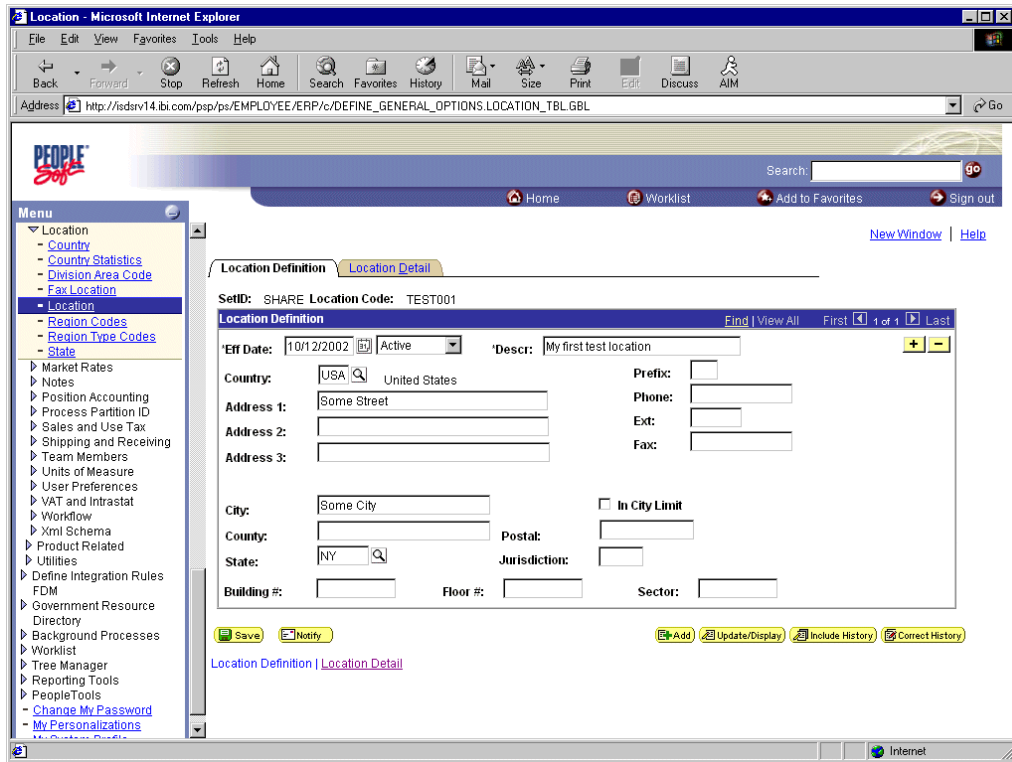
For more information about PeopleCode, consult your PeopleSoft Online Library.

You have finished viewing the PeopleCode for a message. You can now test Integration Broker (in PeopleSoft 8.4) or Application Messaging (in PeopleSoft 8.1).

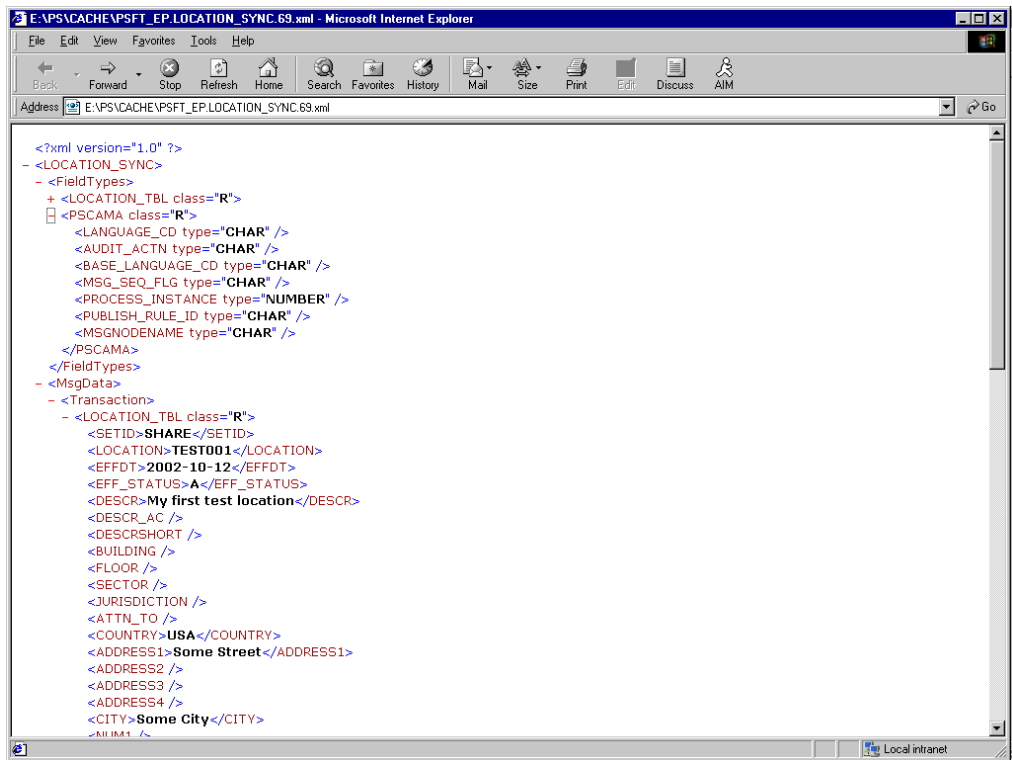
D.4 Testing the Integration Broker

To test the Integration Broker by generating a message, you can navigate to the Location Transaction window and add, update, or delete a location entry in your application. Depending on your application, the way you navigate varies.

The following example illustrates a Financials 8.4 application where a new location with a SetID of SHARE and a Location Code of TEST001 was added.



The following figure shows a portion of the XML output.



Note: The name of the file is PSFT_EP.LOCATION_SYNC.69.xml, which is the concatenation of PSFT_EP (the local Publishing Node), the name of the message, and the number of the Publication ID.

If you cannot send a message successfully, PeopleSoft provides a set of tools for monitoring the progress of your messages. In release 8.1, you use a tool called the Application Messaging Monitor. In release 8.4, you use the Monitor Menu in the Integration Broker.

For a complete description on how to isolate and resolve problems with your messaging environment, consult your PeopleSoft Online Library. If you are still unable to send your XML file, the PeopleSoft Customer Connection can help solve your problem.

D.5 Using Outbound Synchronous Messages

Starting with PeopleTools 8.4, you can send outbound synchronous messages. From a high-level point of view, the primary difference between outbound synchronous and asynchronous is that with outbound asynchronous, the transaction is completed whether the message is actually sent or received.

For synchronous outbound messages:

- The transaction must wait for a response from the external system before continuing.
- The transaction must process the response message.
- The external system must ensure that the response message is correctly formatted.

Oracle Application Adapter for PeopleSoft can work with PeopleSoft outbound synchronous messages. Outbound synchronous messages involve additional configuration steps, both within PeopleSoft and in Oracle WebLogic Server. This topic briefly describes the configuration requirements within PeopleSoft.

Note: The instructions in this topic build upon the instructions for outbound asynchronous messages. It is strongly recommended that you familiarize yourself with outbound asynchronous messaging before attempting outbound synchronous. See ["Configuring Integration Broker in PeopleSoft 8.4"](#) on page D-2 for more information on outbound asynchronous messages.

Ensure that both outbound and inbound messages are created and active. PeopleSoft provides template examples called IB_INST_VER_SYNC_MSG and IB_INST_VER_RESP_MSG. F. See ["Ensuring the Message Is Active and Is Routed Correctly"](#) on page D-3 for information on examining these messages.

D.5.1 Configuring Outbound Synchronous Messages

You can use an existing node, or you can create a new node to configure outbound synchronous messages. See ["Creating and Configuring a New Gateway Node"](#) on page D-7 for information on creating and configuring a node. In either case, you must set up your outbound synchronous transaction.

The following example uses a node and transaction delivered by PeopleSoft. However, this example is for illustrative purposes only and does not actually work as delivered

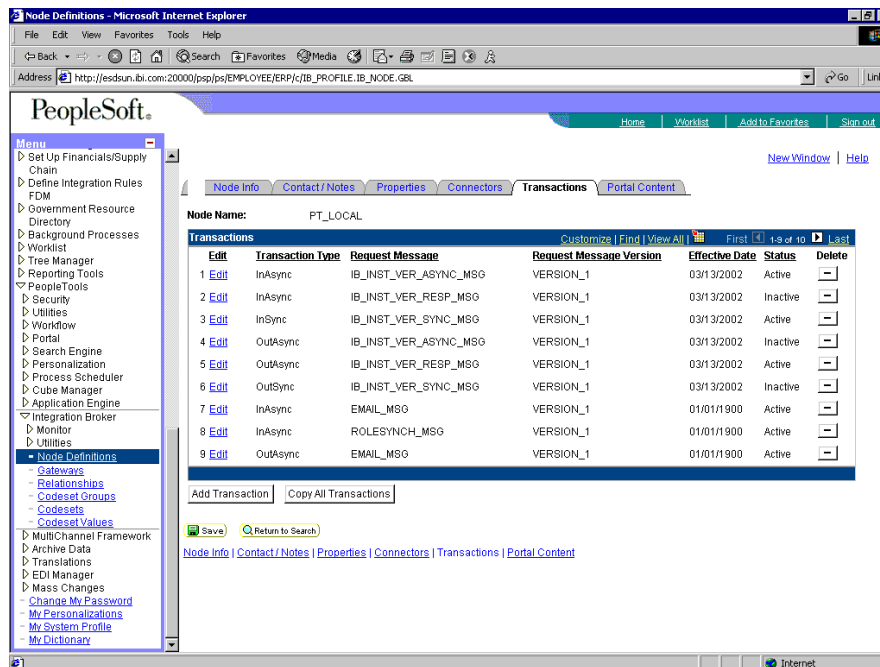
without additional steps. As of Financials release 8.42, there are no preconfigured outbound synchronous transactions that you can use for testing purposes.

Configuring an Outbound Synchronous Message

To configure an outbound synchronous message:

1. Navigate to the **Node Definitions** page and open the **PT_LOCAL** node.
2. Click the **Transactions** tab.

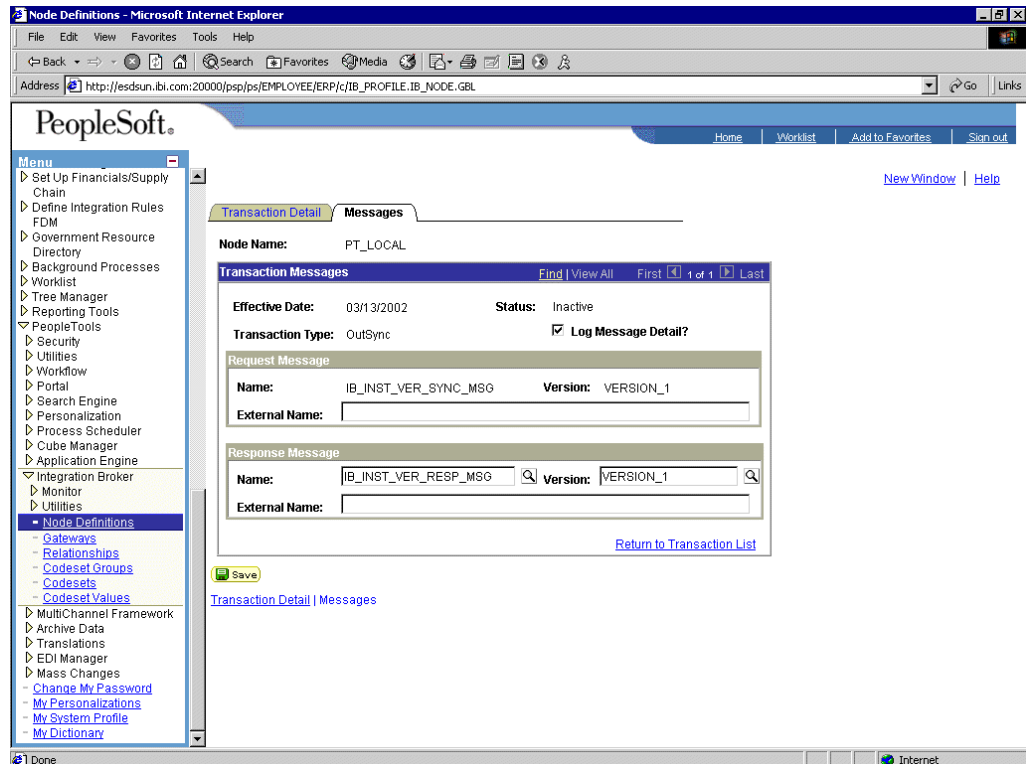
The Transactions pane is displayed.



One outbound synchronous message, `IB_INST_VER_SYNC_MSG`, appears in the Transaction Type list.

3. Click the **Edit** link in the `IB_INST_VER_SYNC_MSG` row.

The following pane is displayed.



4. Click the **Messages** tab.

Both request and response messages appear. The target system must ensure that the response message follows the format of the request message. As the target system is your Oracle WebLogic Server, you must transform the XML that is sent and returned from your final destination.

Note: You must use the PeopleSoft-supplied HTTP target connector when you are working with synchronous outbound messages. You cannot use the TCP/IP84TARGET connector for outbound synchronous messages.

D.5.2 Viewing the PeopleCode for a Synchronous Message

The sample PeopleCode in the following example is for a synchronous outbound message. It differs from asynchronous outbound in that it must handle a response message.

Viewing the PeopleCode for a Financials Synchronous Outbound Message

The following sample code is supplied with the Financials application and is associated with the two messages IB_INST_VER_SYNC_MSG and IB_INST_VER_RESP_MSG.

To view the code, perform the following steps:

1. From Application Designer, open the **PSINST_VER** record.
2. Select the **PeopleCode** display option.
3. Select the **Field Change (FCh)** box for the **IB_SEND_SOS_BTN** field.

The following window is displayed.

```

Application Designer - Untitled - [PSINST_VER_IB_SEND_SOS_BTN.FieldChange (Record PeopleCode)]
File Edit View Insert Build Debug Tools Go Window Help
IB_SEND_SOS_BTN (field) FieldChange

/* SyncRequest example */
Local Message <request_MSG, <response_MSG;
Local Rowset <request_RS, <response_RS, <IB_INST_VER_TRX_RS;
Local Record <response_REC, <IB_INST_VER_DB_REC;
Local SQL <delete_SQL;
Local any <I;

<request_RS = GetLevel0();
<request_MSG = CreateMessage(Message.IB_INST_VER_SYNC_MSG);
<request_MSG.CopyRowset(<request_RS);

/* Create the database record object for the response data */
<IB_INST_VER_DB_REC = CreateRecord(Record.PSINST_VER_TRX);

/* publish the request and wait for the response */
<response_MSG = <request_MSG.SyncRequest();

If (<response_MSG.ResponseStatus = 0) Then

/* Get the response rowset object from the buffer */
<response_RS = <response_MSG.GetRowset();

/* Loop through the message rows moving the data into the database table */
For <I = 1 To (<response_RS.RowCount)
<response_REC = <response_RS.GetRow(<I).GetRecord(Record.PSINST_VER_TRX);
<response_REC.CopyFieldsTo(<IB_INST_VER_DB_REC);
<IB_INST_VER_DB_REC.Insert();
End-For;
End-If;

/* Manual refresh of scrollable area */
<IB_INST_VER_TRX_RS = GetLevel0()(1).GetRowset(Scroll.PSINST_VER_TRX);
<IB_INST_VER_TRX_RS.Flush();
<IB_INST_VER_TRX_RS.Select(Record.PSINST_VER_TRX);

```

D.6 Generating Events Using PeopleTools Tutorial

This section provides a tutorial that walks you through the PeopleSoft event generation process using PeopleTools version 8.48 and 8.49. The tutorial uses the PeopleTools version 8.48.02 and 8.90 Financial / SCM application. As a result, if you are using a different platform, then correlate the terms and commands specific to that operating system. For more information, refer to the appropriate user guide for each specific component.

D.6.1 Configuring PeopleSoft Services

This section describes how to configure PeopleSoft services.

Adding a Remote Node

To add a remote node:

1. Logon to PeopleSoft using the browser-based GUI (Pure Internet Architecture).
2. Select **PeopleTools, Integration Broker, Integration Setup**, followed by **Nodes**.

The Node Definitions tab is displayed.

Node Definitions | Connectors | Portal | W/S Security | Routings

Node Name: EXTERNAL Copy Node

Description: Message publish to third party Rename Node

Node Type: External Default Local Node
 Default Local Node
 Local Node
 Active Node
 Non-Repudiation
 Segment Aware Delete Node

Authentication Option: None

Default User ID: PS

3. Perform the following steps:
 - a. In the Node Name field, type a name for the new node definition, for example, **EXTERNAL**.
 - b. In the Description field, type a brief description for the new node definition.
 - c. From the Node Type drop-down list, select **External**.
 - d. From the Authentication Option drop-down list, ensure that the default option, **none**, is selected.
 - e. In the Default User ID field, type the user ID that is being used by the PeopleSoft system, for example, **PS**.
 - f. Select the **Active Node** check box.
4. Click the **Connectors** tab.
 The Connectors tab is displayed.

Node Definitions | **Connectors** | Portal | W/S Security | Routings

Node Name: EXTERNAL Ping Node

Details

Gateway ID: LOCAL

Connector ID: HTTPTARGET

Properties Customize | Find | First | 1-3 of 3 | Last

Property ID	Property Name	Required	Value
1 HEADER	sendUncompressed	<input checked="" type="checkbox"/>	Y
2 HTTPPROPERTY	Method	<input checked="" type="checkbox"/>	POST
3 PRIMARYURL	URL	<input checked="" type="checkbox"/>	http://bpelclient:1971

Password Encryption Utility

5. Perform the following steps:
 - a. In the Gateway ID field, type **LOCAL**.
 - b. In the Connector ID field, type **HTTPTARGET**.
 - c. For the PRIMARYURL value, enter the host and port number of the PeopleSoft adapter instance that is used to listen for events.
 - d. Leave the default values for the remaining properties.
6. Save your changes.

Selecting a Service

To select a service:

1. Select **PeopleTools, Integration Broker, Integration Setup**, followed by **Services**.
2. Search for the DEPT_SYNC service and select it.

The DEPT_SYNC Services pane is displayed.

Services

Service: DEPT_SYNC
 'Description: Dept Sync Incr. Message
 Comments: Incremental (Component Publish) Dept Sync Message
 Service Alias:
 Object Owner ID: Human Resources
 'Namespace: http://xmlns.oracle.com/Enterprise/Tools/services

[View WSDL](#) [Provide Web Service](#)

Service Operations

Service Operation:
 Operation Type: **ADD**

Existing Operations [Customize](#) | [Find](#) | [View All](#) | First 1-2 of 2 Last

Operation	Message Links	Operation.Default Version	Description	Active	Operation Type
ADD_DEPT.v1			Add Department	<input checked="" type="checkbox"/>	Asynch
DEPT_SYNC.VERSION_1			Dept Sync Incr. Message	<input checked="" type="checkbox"/>	Asynch

3. From the Existing Operations section on the lower-left, click the **DEPT_SYNC.VERSION_1** service operation.

The Service Operation - General pane is displayed.

Default Service Operation Version

'Version: VERSION_1 **Default** **Active**
 Version Description: Dept Sync Incr. Message
 Version Comments:
 Non-Repudiation Runtime Schema Validation

[Introspection](#)

Routing Status

Any-to-Local: Exists
 Local-to-Local: Exists

Routing Actions Upon Save

Regenerate Any-to-Local
 Regenerate Local-to-Local

4. Click the **Active** check box.
5. Click the **Routings** tab to add a new routing.

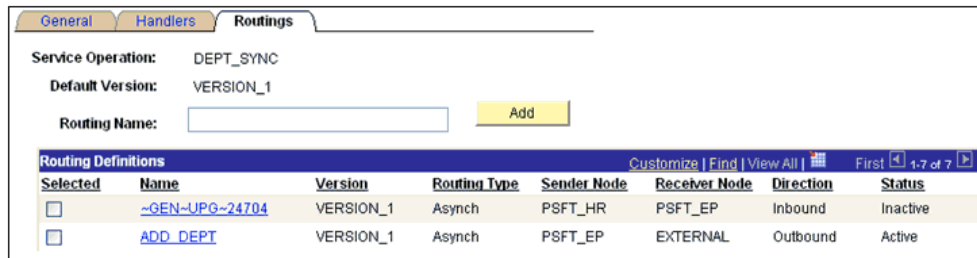
6. In the Routing Name field, type a name for the new routing, for example, **ADD_DEPT**.
7. Click **Add**.
8. Click the **Routing Definitions** tab to add a new routing definition.

9. Perform the following steps:
 - a. In the Sender Node field, type **PSFT_HR**, which is the default PeopleSoft node that publishes the message.
 - b. In the Receiver Node field, type **EXTERNAL**, which is the new node that has been created to subscribe the message published by PeopleSoft.
10. Click the **Connector Properties** tab.

Property ID	Property Name	Value
HEADER	sendUncompressed	Y
HTTPPROPERTY	Method	POST
PRIMARYURL	URL	http://bpelclient:1971

11. Perform the following steps:
 - a. In the Gateway ID field, type **LOCAL**.
 - b. In the Connector ID field, type **HTTPTARGET**.
 - c. For the PRIMARYURL value, enter the host and port number of the PeopleSoft adapter instance that is used to listen for events.
12. Click **Save** and then click **Return**.

You are returned to the Routing Definitions pane. Notice that the new routing definition (ADD_DEPT) is now added to the list.



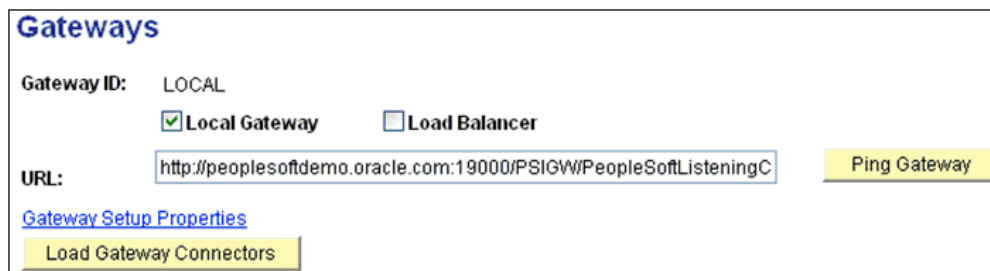
13. Click **Save** to save the details of the Service Operation.
14. Click the **Return to Service** link to return to the Services pane.
15. Click **Save** on the Services pane.

Activating the Gateway

To activate the gateway:

1. Select **PeopleTools, Integration Broker, Configuration**, followed by **Gateways**.
2. Search for the Integration Gateway ID, for example, **Local** and select it.

The Gateways pane is displayed.

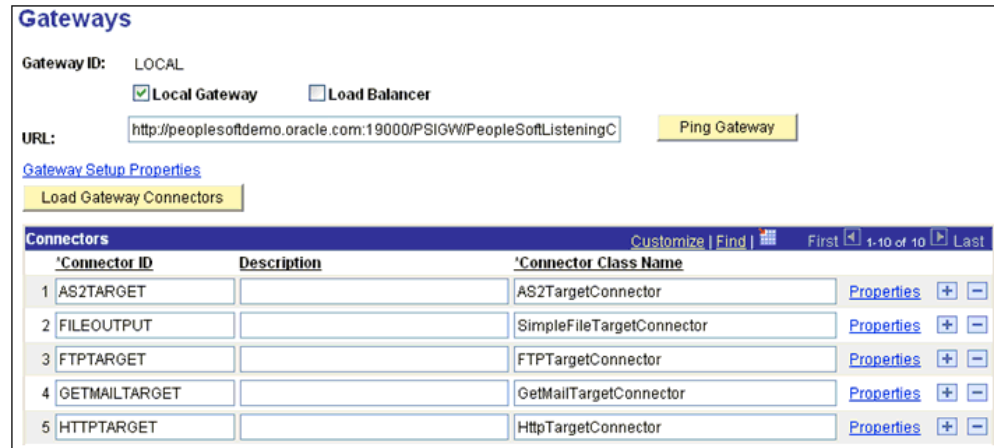


3. Click **Ping Gateway**.

A new browser window is displayed, which shows that the gateway is active.



4. Close the new browser window to return to the Gateways pane.



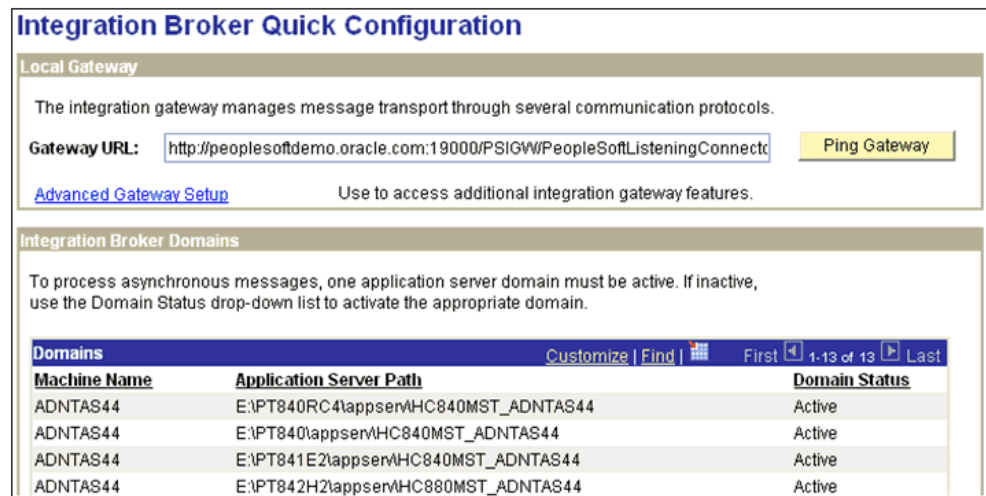
5. Ensure that **HTTPTARGET** is listed in the Connectors section. If it is not listed, click **Load Gateway Connectors** to refresh the available connectors.

Activating the Domain

To activate the domain:

1. Select **PeopleTools, Integration Broker, Configuration**, followed by **Quick Configuration**.

The Quick Configuration pane is displayed.



2. Ensure that the domain is active.

Activating the Service

To activate the service:

1. Select **PeopleTools, Integration Broker, Integration Setup**, followed by **Services**.

The Services pane is displayed.

2. Ensure that the service is active.

Activating the Service Operation (Queue and Message Node)

To activate the service operation:

1. Select **PeopleTools, Integration Broker, Configuration, Integration Setup,** followed by **Services.**

The Services pane is displayed.

Operation	Default Version	Description	Active	Operation Type
ADD_DEPT.v1		Add Department	<input checked="" type="checkbox"/>	Asynch
DEPT_SYNC.VERSION_1		Dept Sync Incr. Message	<input checked="" type="checkbox"/>	Asynch

2. From the Existing Operations section on the lower-left, click the **DEPT_SYNC.VERSION_1** service operation.

The Service Operation - General pane is displayed.

- In the message information section on the bottom of the page, note the value in the Queue Name field of the service operation.

Message Information

Type: Request

Message.Version: [View Message](#)

Queue Name: [View Queue](#) [Add New Queue](#)

- Click the **View Queue** link.
The Queue Definitions pane is displayed.

Queue Definitions

Queue Name: ENTERPRISE_SETUP **Archive** **Unordered**

Description: **Queue Status:**

Comments: **Object Owner ID:**

Operations Assigned to Queue

Service	Version
ALTACCT_CF_FULLSYNC	VERSION_1
ALTACCT_CF_SYNC	VERSION_1
BUDGET_REF_CF_FULLSYNC	VERSION_1
BUDGET_REF_CF_SYNC	VERSION_1
BUS_UNIT_FS_FULLSYNC	VERSION_1
BUS_UNIT_FS_SYNC	VERSION_1
CHARTFIELD1_FULLSYNC	VERSION_1
CHARTFIELD1_SYNC	VERSION_1
CHARTFIELD2_FULLSYNC	VERSION_1
CHARTFIELD2_SYNC	VERSION_1

Define Partitioning Fields

Include	Field	Alias Name
<input type="checkbox"/>	OPERATIONNAME	<input type="text"/>
<input type="checkbox"/>	PUBLISHER	<input type="text"/>
<input type="checkbox"/>	PUBPROC	<input type="text"/>

- Ensure that the Queue Status drop-down list has **Run** selected.
- Click **Return** to return to the Service operation pane.
- Click the **Routing** tab.

General **Handlers** **Routing**

Service Operation: DEPT_SYNC

Default Version: VERSION_1

Routing Name:

Routing Definitions

Selected	Name	Version	Routing Type	Sender Node	Receiver Node	Direction	Status
<input type="checkbox"/>	~GENERATED~94652240	VERSION_1	Asynch	~ANY~	PSFT_HR	Inbound	Active
<input type="checkbox"/>	~GEN~UPG~12357	VERSION_1	Asynch	PSFT_HR	PSFT_CR	Outbound	Inactive
<input type="checkbox"/>	ADD_DEPT	VERSION_1	Asynch	PSFT_HR	EXTERNAL	Outbound	Active
<input type="checkbox"/>	~GEN~UPG~20085	VERSION_1	Asynch	PSFT_HR	PSFT_EP	Outbound	Inactive
<input type="checkbox"/>	~GEN~UPG~25397	VERSION_1	Asynch	PSFT_HR	PSFT_HR	Local	Inactive
<input type="checkbox"/>	~GEN~UPG~13180	VERSION_1	Asynch	PSFT_HR	PSFT_LM	Outbound	Inactive

- Ensure that the routings are active.
- Click the **Return to Service** link on the bottom of the pane to return to the Services pane.
- Click **Return to Search** to go to the Services search pane.

In the above steps, if the status has been activated in any of the panes for any of the components, then make sure to save the status at the appropriate pane and the respective components.

11. Select **PeopleTools, Integration Broker, Configuration, Integration Setup, Services**, followed by **Nodes**.

The Node Definitions pane is displayed.

The screenshot shows the 'Node Definitions' configuration window. It includes the following fields and options:

- Node Name:** EXTERNAL
- Description:** Message publish to third party
- Node Type:** External (dropdown menu)
- Authentication Option:** None (dropdown menu)
- Default Local Node:**
- Local Node:**
- Active Node:**
- Non-Repudiation:**
- Segment Aware:**
- Action Buttons:** Copy Node, Rename Node, Delete Node

12. Ensure that the status of the External node is active.

This completes the configuration on the PeopleSoft side.

D.6.2 Triggering the Event in PeopleSoft

This section describes how to trigger the event in PeopleSoft.

Triggering the Event

To trigger the event:

1. Logon to PeopleSoft using the browser-based GUI (Pure Internet Architecture).
2. Navigate to **Main Menu, Set up HRMS, Foundation Tables**, followed by **Departments**.

The Departments pane is displayed.

The screenshot shows the 'Departments' search interface with the following details:

- Search Criteria:**
 - SETID:** = (dropdown) SHARE
 - Department:** begins with (dropdown) []
 - Description:** begins with (dropdown) []
 - Company:** begins with (dropdown) []
 - Location SetID:** begins with (dropdown) []
 - Location Code:** begins with (dropdown) []
 - Manager ID:** begins with (dropdown) []
 - Budget with Department:** begins with (dropdown) []
- Options:**
 - Include History
 - Correct History
 - Case Sensitive
- Action Buttons:** Search, Clear, Basic Search, Save Search Criteria

3. Click the **Add a New Value** tab.

The screenshot shows a web form titled "Departments". At the top, there are two tabs: "Find an Existing Value" and "Add a New Value". The "Add a New Value" tab is selected. Below the tabs, there are two input fields: "SetID:" with the value "SHARE" and a magnifying glass icon, and "Department:" with the value "BPELTEST01" and a magnifying glass icon. At the bottom left, there is a yellow "Add" button.

4. Enter the appropriate value in the SetID and Department fields
5. Click **Add**.

The Department Profile tab is displayed.

The screenshot shows the "Department Profile" form. At the top, there are tabs for "Department Profile" and "Comm. Acctg. and EG". The "Department Profile" tab is selected. The form displays "SetID: SHARE" and "Department: BPELTEST01". There is a link "Business Units that use this Setid". Below this, there are several fields: "Effective Date:" (08/25/2008), "Status:" (Active), "Description:" (BPEL Inbound Test), "Short Description:" (BPEL Inbou), "Location SetID:", "Location:", and "Company:". There is a section for "Manager Type" with radio buttons for "None", "EmpID" (with "Manager ID:" field), and "Position" (with "Manager Position:" field). Below that, there are "Budget Year End Date:" and "Budget Level:" (None). At the bottom, there is a section for "Payroll for North America" with a "Tax Location:" field.

6. Provide the necessary information according to your requirements to create a new Department record.
7. Click **Save** at the bottom of the pane when you are finished.

D.6.3 Verifying the Event Results

This section describes how to verify the event results.

Verifying the Results

To verify the results:

1. Logon to PeopleSoft using the browser-based GUI (Pure Internet Architecture).
2. Navigate to **Main Menu, PeopleTools, Integration Broker, Service Operation Monitor**, followed by **Asynchronous Services**.

The Monitor Overview tab is displayed.

Queue Name	Error	New	Started	Working	Done	Retry	Timeout	Edited	Canceled	Hold
ENTERPRISE_SETUP	0	0	0	0	5	0	0	0	0	0
PSRF_REPORTING_FOLDERS	0	0	0	0	1366	0	0	0	0	0
PSXP_MSG_CHNL	0	0	0	0	1	0	0	0	0	0
USER_PROFILE	0	0	0	0	1	0	0	0	0	0

Here you can view the results of the events at the queue level.

- Click the number link (5) queue (Enterprise_Setup) in the Result section.

The Operation Instances tab is displayed.

Select	Transaction ID	Queue Name	Publishing Node	Sub Queue	Status	Time Stamp
<input type="checkbox"/>	b727c644-72d0-11dd-9dd0-b53768245214	ENTERPRISE_SETUP	PSFT_HR	Sub Queue Link	Done	08/25/2008 2:07:48PM

Notice that a new record for the event (department) is available.

- Click the **Details** link on the lower-right.

A new browser window (Asynchronous Details) is displayed.

Subscriber Node	Segment	Status
EXTERNAL	1	Done

Notice that the status of the transaction is **DONE** for the Publishing and Subscriber nodes.

Glossary

adapter

Provides universal connectivity by enabling an electronic interface to be accommodated (without loss of function) to another electronic interface.

agent

Supports service protocols in listeners and documents.

business service

Also known as a Web service. A Web service is a self-contained, modularized function that can be published and accessed across a network using open standards. It is the implementation of an interface by a component and is an executable entity.

channel

Represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by an adapter.

listener

A component that accepts requests from client applications.

port

Associates a particular business object exposed by the adapter with a particular disposition. A disposition is a URL that defines the protocol and location of the event data. The port defines the end point of the event consumption.

Index

A

- access methods, 1-2
- access rights, 7-1
- adapter configuration
 - updating, 3-2
- adapter exceptions, 6-2 to 6-4
- Adapter Lib Directory parameter, 2-3 to 2-4
- adapter types
 - resource, 1-1, 3-1 to 3-2
- adapters, 1-1 to 1-2
 - configuring, 2-1 to 2-27, 3-2
 - deploying, 1-1, 1-3, 3-1 to 3-2, 4-1
 - integrating with BPEL Process Manager, 4-1
 - troubleshooting, 6-1 to 6-7
- Adapters node, 4-36
- Add Channel dialog box, 2-22
- Advanced tab, 2-26
- ae batch script, 6-2
- API Java programs
 - compiling, A-4 to A-6
 - creating, A-1 to A-4
- application adapters, 1-1 to 1-2
 - configuring, 2-1 to 2-27, 3-2
 - deploying, 1-1, 1-3, 3-1 to 3-2, 4-1
 - integrating with BPEL Process Manager, 4-1
 - troubleshooting, 6-1 to 6-7
- Application Explorer, 1-2 to 1-3, 2-11, 2-15, 4-6, 4-34
 - application systems and, 2-11
 - debugging and, 6-2
 - J2CA configuration and, 3-2
 - OracleAS Adapter J2CA and, 6-5
 - security and, 7-1, 7-6
 - testing and, 6-2
 - troubleshooting, 6-1 to 6-3
 - WSDL files and, 4-34
- Application Messaging, 1-2
- Application Messaging Manager, 1-3
- Application Server dialog box, 2-12
- Application Server parameter, 2-12
- application system objects, 2-15
- application systems
 - Application Explorer and, 2-11
 - supported, 2-11
- asynchronous support, 1-2
- Available list, 7-4 to 7-5

B

- BPEL Designer, 4-1
- BPEL Process Manager
 - integrating with adapters, 4-1
 - OracleAS Adapter for PeopleSoft and, 4-1
- browsing metadata, 2-15
- BSE (OracleAS Adapter Business Services Engine), 2-8
 - configuring, 2-4, 2-8
 - connection access to, 7-6
 - troubleshooting, 6-5
- BSE (OracleWLS Adapter Business Services Engine), 1-1, 1-3
- BSE configuration page, 2-3 to 2-4
- BSE control service URL, 7-8
- BSE deployment, 1-3
- BSE repositories
 - migrating, 7-8
- BSE settings window, 2-4
- BSE system settings, 2-4 to 2-6
- BSE URL field, 2-9
- business events, 1-1
- business functions, 2-17
- business objects, 2-11
 - browsing, 2-15
- business processes, 1-2
- business services
 - creating, 2-17 to 2-18
 - deploying, 7-1
 - testing, 2-18
- Business Services node, 7-2 to 7-4, 7-6

C

- CCI calls, 3-1
- channel configuration parameters, 2-24
 - Error Directory, 2-26
 - File Mask, 2-26
 - Host, 2-24
 - Is Keep Alive, 2-24
 - Is Length Prefix, 2-24
 - Is XML, 2-24
 - Poll interval, 2-26
 - Polling Location, 2-26
 - Port, 2-21, 4-35

- Port Number, 2-24
- Processing Mode, 2-26
- Response/Ack Directory, 2-26
- Server port, 2-21, 4-35
- Synchronization Type, 2-24 to 2-26
- Thread limit, 2-26
- channel types
 - File, 2-25 to 2-27
- channels, 2-19 to 2-27
 - creating, 2-19 to 2-27, 4-34 to 4-36
 - deleting, 2-27
 - editing, 2-27
 - ports and, 2-23
 - starting, 2-24, 4-36
 - stopping, 2-25, 4-36
 - testing and debugging, 4-36
- channels. *See also* listeners
- closing connections, 2-13
- component interface API, 1-3
- component interface properties, 6-3
- Component Interface testing tool
 - three-tier mode and, 6-3
- component interfaces, 1-2 to 1-3, 2-15
 - creating, C-1 to C-4
 - security and, C-5 to C-12
 - testing, C-12 to C-16
- component methods, 1-3
- configuration parameters, 3-4
 - IWayConfig, 3-4
 - IWayHome, 3-4
 - IWayRepoPassword, 3-4
 - IWayRepoURL, 3-4
 - IWayRepoUser, 3-4
 - Loglevel, 3-4
- Configuration under Business Services
 - node, 7-2 to 7-4, 7-6
- configurations
 - defining, 2-9 to 2-10
 - overwriting, 3-4
 - testing, B-14 to B-15
- Configurations node, 2-9
- configuring adapters, 3-2
- configuring BSE system settings, 2-4 to 2-6
- configuring File Output Connector, D-5 to D-7
- configuring Integration Broker, D-2
- configuring Integration Gateway, D-5 to D-7
- configuring IntegrationGateway.properties
 - file, D-4 to D-5
- configuring message channels, D-15 to D-17
- configuring message routers, B-1 to B-14
- configuring nodes
 - Gateway, D-7 to D-11
 - TCP/IP84 connector, B-3 to B-6
- configuring repositories, 2-6
- configuring target connectors, B-1 to B-14
- connecting to OracleAS Adapter J2CA, 6-5
- connecting to PeopleSoft, 2-11 to 2-13, 4-36, 6-2, 6-6
- Connection dialog box, 4-36
- connection parameters, 2-12 to 2-13, 4-35 to 4-36, 6-2
 - Application Server, 2-12

- Hostname, 2-9, 6-2 to 6-4
- Password, 2-12
- Port, 2-2 to 2-4, 2-12, 2-21, 4-35, 6-2 to 6-4
- User, 2-12
- connections
 - closing, 2-13
 - deleting, 2-14
 - establishing, 2-11 to 2-13, 4-36
- Connector
 - deploying to Oracle Application Server, 3-1
- connector factories, 3-2
- connector factory objects, 3-2
 - multiple, 3-4
- connectors
 - configuring, B-1 to B-14
- control methods, 7-9
- Create Business Service dialog box, 2-17
- Create Outbound JCA Service
 - (Request/Response), 4-7
- creating channels, 2-19 to 2-27, 4-34 to 4-36
- creating events, 4-36 to 4-37
- creating repository projects, 2-8 to 2-10
- creating schemas, 2-15 to 2-16, 6-3
- creating Web services, 2-17 to 2-18

D

- database connections
 - opening, 3-4
- databases
 - connecting to, 3-4
 - Oracle, 3-4
- Debug Level parameter, 2-5
- deleting channels, 2-27
- deleting connections, 2-14
- deploying adapters, 3-1 to 3-2, 4-1
- deployments
 - BSE, 1-3
- Description field, 2-17, 7-2 to 7-5, 7-7
- design time, 4-34, 7-8
 - configuring, 4-6, 4-34
- disconnecting from PeopleSoft targets, 2-13
- DNS Lookup option, 7-7
- DNS name, 7-7
- Domain Name System (DNS), 7-7
- domain names, 7-7
- Domain option, 7-7

E

- Eclipse. *See* JDeveloper
- Eclipse. *See* JDeveloper
- editing channels, 2-27
- editing targets, 2-14
- EIP (Enterprise Integration Points), 1-2
- EJB (Enterprise Java Beans), 3-1
- Encoding parameter, 2-5
- Enterprise Connector for J2EE Connector Architecture
 - (J2CA), 1-3
- Enterprise Integration Points (EIP), 1-2

Enterprise Java Beans (EJB), 3-1
Error Directory parameter, 2-26
error messages, 6-2 to 6-7
 target systems and, 6-6
event integration
 verifying, 4-51
event messages, 4-34
event ports
 creating, 4-36 to 4-37
events, 1-1, 1-3
 creating, 4-36 to 4-37
 triggering, 4-51
Execution Denied list, 7-6
Execution Granted list, 7-6
Existing Service Names list, 2-17
Export WSDL dialog box, 4-37

F

fault code elements, 6-5
fault string elements, 6-5
File channel, 2-25 to 2-27
File Mask parameter, 2-26
File Output Connector
 configuring, D-5 to D-7
file system repositories
 configuring, 2-6

G

Gateway node, D-7 to D-11
generating schemas, 2-15 to 2-16, 6-3
Grant Access check box, 7-7
Group (of Computers) option, 7-7
groups
 creating, 7-3
Groups node, 7-3 to 7-4

H

Host parameter, 2-24
Hostname parameter, 2-4, 2-9, 6-2 to 6-4
HTTP protocol, 2-23

I

inbound J2CA services, 4-34
installation directories, 3-4
instances of policy types, 7-1
Integration Broker, 1-2 to 1-3, D-1 to D-25
 configuring, D-2
Integration Gateway
 configuring, D-5 to D-7
IntegrationGateway.properties file
 configuring, D-4 to D-5
IP (Mask)/Domain field, 7-7
IP addresses, 7-7
IP and Domain Restriction policy type, 7-6
Is Keep Alive parameter, 2-24
Is Length Prefix parameter, 2-24
Is XML parameter, 2-24

IWayConfig parameter, 3-4
IWayHome parameter, 3-4
IWayRepoPassword parameter, 3-4
IWayRepoURL parameter, 3-4
IWayRepoUser parameter, 3-4
iwpsci84.jar file, 6-2, 6-3
iwpsci8x.jar file, 6-3

J

J2CA (Enterprise Connector for J2EE Connector
 Architecture), 1-3
 OracleAS Adapter and, 3-1
J2CA configuration
 Application Explorer and, 3-2
J2CA repositories
 migrating, 7-10
J2CA resource adapters, 1-1, 3-1
J2CA services, 4-34
JAR files, 6-2
 errors and, 6-3
Java program clients, 3-1
JDeveloper, 4-1

L

Language parameter, 2-5
License and Method dialog box, 2-17
License field, 2-17
licenses, 2-17
list of nodes, 6-2
listeners, 1-4, 2-11, B-10
log files, 6-1
log levels
 overwriting, 3-4
Loglevel parameter, 3-4

M

managed connector factories, 3-2
managed connector factory objects, 3-2
 multiple, 3-4
ManagedConnectionFactory parameter, 3-4
Mediator Inbound Process, 5-27
Mediator Outbound Process, 5-6
message channels
 configuring, D-15 to D-17
message routers
 configuring, B-1 to B-14
message types
 event, 4-34
messages, 1-1, 1-3
metadata, 2-15
 browsing, 2-15
 storing, 2-5, 7-8
Method Name field, 2-17
methods, 7-1
 changing, C-4
 viewing, C-4
migrating repositories, 7-8
migrating Web services, 7-10

MySAP Easy Access window, 4-51
MySAP Workbench, 4-51

N

Name field, 7-2 to 7-5
New Configuration dialog box, 2-9 to 2-10
New Group dialog box, 7-3
New Policy permissions dialog box, 7-5
New User dialog box, 7-2
Node list, 6-2
nodes
 Adapters, 4-36
 Business Services, 7-2 to 7-4
 Configuration under Business Services, 7-2 to 7-4, 7-6
 Configurations, 2-9
 connected, 4-36
 Gateway, D-7 to D-11
 Groups, 7-3 to 7-4
 PeopleSoft, 2-12, 4-36
 Policies, 7-4
 Security, 7-2 to 7-4, 7-6
 Service Adapters, 2-12
 TCP/IP84 connector, B-3 to B-6
 Users, 7-3
 Users and Groups, 7-2 to 7-3
Number of Async. Processors parameter, 2-5

O

OC4J-ra.xml file, 3-2 to 3-4
Oracle Application Server
 deployment of Connector to, 3-1
Oracle databases, 3-4
Oracle JDeveloper, 4-1
Oracle JDeveloper BPEL Designer. *See* BPEL Designer, JDeveloper, or Oracle JDeveloper
Oracle repositories
 migrating, 7-8
Oracle's Unified Method (OUM), viii
OracleAS Adapter
 installation directory and, 3-4
 J2CA and, 3-1
OracleAS Adapter Business Services Engine (BSE), 2-8
 configuring, 2-4, 2-8
 connection access to, 7-6
 troubleshooting, 6-5
OracleAS Adapter for PeopleSoft
 BPEL Process Manager and, 4-1
 deploying, 4-1
 troubleshooting, 6-1 to 6-7
OracleAS Adapter J2CA, 2-8 to 2-9
 Application Explorer and, 6-5
 connecting to, 6-5
OracleWLS Adapter Application Explorer. *See also* Application Explorer
OracleWLS Adapter Business Services Engine (BSE), 1-1, 1-3

OracleWLS Application Adapter for PeopleSoft
 configuring, 2-1 to 2-27
 deploying, 1-1

P

parameter types
 channel configuration, 2-24
 configuration, 3-4
 connection, 2-12, 4-35 to 4-36, 6-2
 repository, 2-5
 repository migration, 7-9
 system, 2-5
Password parameter, 2-3, 2-12, 7-2
passwords, 2-3, 3-4
PeopleSoft
 connecting to, 2-11 to 2-13, 6-2, 6-6
PeopleSoft API Java programs
 compiling, A-4 to A-6
 creating, A-1 to A-4
PeopleSoft business objects, 2-11
PeopleSoft Java API, 1-2
PeopleSoft node, 2-12, 4-36
PeopleSoft XML, 1-2
permissions, 7-1
 denying, 7-6
 granting, 7-6
policies, 7-1
 applying, 7-1
 creating, 7-4
Policies node, 7-4
policy types, 7-1
 IP and Domain Restriction, 7-6
policy-based security, 7-1 to 7-7
Poll interval parameter, 2-26
Polling Location parameter, 2-26
Port Number parameter, 2-9, 2-24
Port parameter, 2-2 to 2-4, 2-12, 2-21, 4-35, 6-2 to 6-4
ports
 channels and, 2-23
 creating, 4-36 to 4-37
privileges, 7-1
 setting, 7-1
Process Manager. *See* BPEL Process Manager
Processing Mode parameter, 2-26
properties, 3-4
Protocol list, 2-23
psjoa.jar file, 6-2, 6-3

R

repositories
 configuring, 2-6
 migrating, 7-8
Repository Driver parameter, 2-5
repository information
 storing, 2-6
repository migration parameters, 7-9
repository parameters
 Driver, 2-5

- Password, 2-5
- Pooling, 2-5
- Type, 2-5
- URL, 2-5
- User, 2-5
- Repository Password parameter, 2-5
- Repository Pooling parameter, 2-5
- repository projects
 - creating, 2-8 to 2-10
 - Web services and, 2-8
- Repository Type parameter, 2-5
- Repository URL parameter, 2-4, 2-5
- repository URLs, 7-10
- Repository User parameter, 2-5
- repository.xml file, 7-11
- request documents, 2-15
- Request Schema tab, 2-15, 2-16
- Request tab, 2-26
- resource adapters, 1-1, 3-1 to 3-2
- Resource Execution policy type, 7-1
- response documents, 2-15
- Response Schema tab, 2-15, 2-16
- Response tab, 2-26
- Response/Ack Directory parameter, 2-26
- runtime, 7-8

S

- schemas, 2-15
 - creating, 1-3, 2-15 to 2-16, 6-3
- security, 7-1 to 7-7
 - Application Explorer and, 7-1, 7-6
 - component interfaces and, C-5 to C-12
 - configuring, 7-2
- Security node, 7-2 to 7-4, 7-6
- security policies
 - applying, 7-1
 - creating, 7-4
- Security Policy option, 7-6
- Selected list, 7-4 to 7-5
- Server port parameter, 2-21, 4-35
- Service Adapters node, 2-12
- service names, 2-17
- Service Provider list, 2-9 to 2-10
- Service-Oriented Architecture (SOA), 4-1
- services, 1-1, 1-3
 - creating, 2-17 to 2-18
 - testing, 2-18
- servlets, 3-1
- Single (Computer) option, 7-7
- SOA (Service-Oriented Architecture), 4-1
- SOAP agents, 6-5
- SOAP envelopes, 7-9
- SOAP faults, 6-5
- soap operation name dialog box, 7-8
- SOAP requests, 6-5 to 6-7, 7-1, 7-5
 - creating, 7-8
 - errors and, 6-5 to 6-7
- SOAP responses, 6-5 to 6-7
- starting a channel, 2-24

- stopping a channel, 2-25
- storing metadata, 7-8
- Synchronization Type parameter, 2-24 to 2-26
- synchronous support, 1-2
- system objects, 2-15
- system parameters
 - Adapter Lib Directory, 2-5
 - Debug Level, 2-5
 - Encoding, 2-5
 - Language, 2-5
 - Number of Async. Processors, 2-5
- system settings
 - configuring, 2-4 to 2-6

T

- target connectors
 - configuring, B-1 to B-14
- target systems
 - errors and, 6-6
- targets
 - connecting to, 2-11 to 2-13, 4-36, 6-6
 - defining, 2-11 to 2-13
 - deleting, 2-14
 - disconnecting from, 2-13
 - editing, 2-14
- TCP Listener dialog box, 2-23
- TCP protocol, 2-23
- TCP/IP84 connector node, B-3 to B-6
- testing component interfaces, C-12 to C-16
- testing configurations, B-14 to B-15
- testing Web services, 2-18
- Thread limit parameter, 2-26
- trace information, 6-1
- transaction processing, 1-1
- transactions
 - storing, 2-5
- triggering events in MySAP ERP, 4-51
- troubleshooting, 6-1 to 6-7
 - Application Explorer, 6-1 to 6-3
 - BSE, 6-5
 - Web services, 6-5 to 6-7
- Type list, 7-5, 7-7

U

- updating adapter configuration, 3-2
- User ID parameter, 2-12
- User parameter, 2-3
- users
 - associating, 7-2
- Users and Groups node, 7-2 to 7-3
- Users node, 7-3

V

- viewing methods, C-4

W

- Web Service Definition Language

- (WSDL), 2-16 to 2-17
- Web service names, 2-17
- Web services, 1-1 to 1-3, 2-8
 - creating, 2-17 to 2-18
 - delivering, 2-5
 - deploying, 6-6, 7-1
 - integrating, 4-1
 - migrating, 7-10
 - repository projects and, 2-8
 - testing, 2-18
 - troubleshooting, 6-5 to 6-7
- Web services policy-based security, 7-1 to 7-7
- WSDL (Web Service Definition Language), 2-16 to 2-17
- WSDL documents, 4-1
- WSDL file location dialog box, 7-8
- WSDL files, 4-1
 - Application Explorer and, 4-34
 - creating, 4-34

X

- XML messages, 1-1
- XML schemas, 1-2, 2-1, 2-15
 - creating, 2-15 to 2-16