Oracle® Application Server Adapter

for PeopleSoft User's Guide

10*g* Release 2 (10.1.2)

Part No. B14060-01

November 2004

Oracle Application Server Adapter for PeopleSoft User's Guide, 10*g* Release 2 (10.1.2)

Part No. B14060-01

Copyright © 2004, Oracle. All rights reserved.

Primary Author: Suzanne McGrath

Contributing Author: Meera Srinivasan

Contributor: Arvind Jain

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

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

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

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

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

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

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

Contents

C Using Component Interfaces

D Using PeopleSoft 8 Integration Broker

Glossary

Index

Send Us Your Comments

Oracle Application Server Adapter for PeopleSoft User's Guide, 10*g* Release 2 (10.1.2)

Part No. B14060-01

Oracle welcomes your comments and suggestions on the quality and usefulness of this publication. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most about this manual?

If you find any errors or have any other suggestions for improvement, please indicate the title and part number of the documentation and the chapter, section, and page number (if available). You can send comments to us in the following ways:

- Electronic mail: appserverdocs_us@oracle.com
- FAX: (650) 506-7375. Attn: Oracle Application Server Documentation Manager
- Postal service:

  Oracle Corporation
  Oracle Application Server Documentation
  500 Oracle Parkway, Mailstop 1op6
  Redwood Shores, CA 94065
  USA

If you would like a reply, please give your name, address, telephone number, and electronic mail address (optional).

If you have problems with the software, please contact your local Oracle Support Services.

Preface

This guide explains how to use Oracle Application Server and the Oracle Application Server Adapter for PeopleSoft to access PeopleSoft Component Interfaces. In this guide you will learn how to define a delivery channel for PeopleSoft and add an interaction to generate native events, which are XML instances defined by XSD (XML payload defined by an XML Schema Definition instance). In this guide you will also find a chapter describing the datatype mapping between PeopleSoft and XSD.

This preface contains these topics:

- Documentation Accessibility
- Intended Audience
- Organization
- Related Documentation
- Conventions

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. 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 additional information, visit the Oracle Accessibility Program Web site at

`http://www.oracle.com/accessibility/`

Accessibility of Code Examples in Documentation JAWS, a Windows screen reader, 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, JAWS 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.

Intended Audience

Oracle Application Server Adapter for PeopleSoft User's Guide is intended for anyone who performs the following tasks:

- Creates delivery channels and interactions with a PeopleSoft system
- Maintains applications

To use this document, you need some knowledge of PeopleSoft Component Interfaces.

Organization

This document contains:

Chapter 1, "Introduction to Oracle Application Server Adapter for PeopleSoft"

This chapter describes the Oracle Application Server Adapter for PeopleSoft and the hardware and software requirements.

Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer"

This chapter provides instructions for starting Application Explorer, for creating projects, establishing a connection to PeopleSoft, and creating schemas and Web services. It also explains how to configure the event adapter.

Chapter 3, "Deployment and Integration"

This chapter describes Oracle Containers for J2EE (OC4J) deployment and Oracle Application Server Integration InterConnect integration.

Chapter 4, "Examples"

This chapter contains examples.

Chapter 5, "Troubleshooting and Error Messages"

This chapter describes how to troubleshoot and interpret error messages.

Chapter 6, "Advanced Topics"

This chapter includes advanced topics for expert users.

Appendix A, "Generating Component Interface APIs"

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

Appendix B, "Configuring the PeopleSoft Message Router"

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

Appendix C, "Using Component Interfaces"

This appendix describes how to create new component interfaces and how to modify existing component interfaces.

Appendix D, "Using PeopleSoft 8 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.

Related Documentation

For more information, see these Oracle resources:

- *Oracle Application Server Adapter Concepts*
- *Oracle Application Server Adapters Installation Guide*
- *Oracle Application Server Administrator's Guide*
- *Oracle Application Server Concepts*
- *Oracle Application Server Containers for J2EE User's Guide*
- *Oracle Application Server Integration InterConnect User's Guide*

Printed documentation is available for sale in the Oracle Store at

```
http://oraclestore.oracle.com
```

To download free release notes, installation documentation, white papers, or other collateral, please visit the Oracle Technology Network (OTN). You must register online before using OTN; registration is free and can be done at

```
http://www.oracle.com/technology/membership
```

If you already have a username and password for OTN, then you can go directly to the documentation section of the OTN Web site at

```
http://www.oracle.com/technology/documentation
```

Conventions

This section describes the conventions used in the text and code examples of this documentation set. It describes:

- Conventions in Text
- Conventions in Code Examples
- Conventions for Windows Operating Systems

Conventions in Text

We use various conventions in text to help you more quickly identify special terms. The following table describes those conventions and provides examples of their use.

Convention	Meaning	Example
Bold	Bold typeface indicates terms that are defined in the text or terms that appear in a glossary, or both.	When you specify this clause, you create an **index-organized table**.
Italics	Italic typeface indicates book titles or emphasis.	*Oracle Database Concepts* Ensure that the recovery catalog and target database do *not* reside on the same disk.

Convention	Meaning	Example
`UPPERCASE monospace (fixed-width) font`	Uppercase monospace typeface indicates elements supplied by the system. Such elements include parameters, privileges, datatypes, RMAN keywords, SQL keywords, SQL*Plus or utility commands, packages and interactions, as well as system-supplied column names, database objects and structures, usernames, and roles.	You can specify this clause only for a `NUMBER` column. You can back up the database by using the `BACKUP` command. Query the `TABLE_NAME` column in the `USER_TABLES` data dictionary view. Use the `DBMS_STATS.GENERATE_STATS` procedure.
`lowercase monospace (fixed-width) font`	Lowercase monospace typeface indicates executables, filenames, directory names, and sample user-supplied elements. Such elements include computer and database names, net service names, and connect identifiers, as well as user-supplied database objects and structures, column names, packages and classes, usernames and roles, program units, and parameter values. **Note:** Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.	Enter `sqlplus` to open SQL*Plus. The password is specified in the `orapwd` file. Back up the datafiles and control files in the `/disk1/oracle/dbs` directory. The `department_id`, `department_name`, and `location_id` columns are in the `hr.departments` table. Set the `QUERY_REWRITE_ENABLED` initialization parameter to `True`. Connect as `oe` user. The `JRepUtil` class implements these interactions.
`lowercase italic monospace (fixed-width) font`	Lowercase italic monospace font represents placeholders or variables.	You can specify the *`parallel_clause`*. Run `U`*`old_release`*`.SQL` where *`old_release`* refers to the release you installed prior to upgrading.

Conventions in Code Examples

Code examples illustrate SQL, PL/SQL, SQL*Plus, or other command-line statements. They are displayed in a monospace (fixed-width) font and separated from normal text as shown in this example:

```
SELECT username FROM dba_users WHERE username = 'MIGRATE';
```

The following table describes typographic conventions used in code examples and provides examples of their use.

Convention	Meaning	Example
[ ]	Brackets enclose one or more optional items. Do not enter the brackets.	`DECIMAL (`*`digits`*` [ , `*`precision`*` ])`
{ }	Braces enclose two or more items, one of which is required. Do not enter the braces.	`{ENABLE | DISABLE}`

|	A vertical bar represents a choice of two or more options within brackets or braces. Enter one of the options. Do not enter the vertical bar.	`{ENABLE | DISABLE}`
		`[COMPRESS | NOCOMPRESS]`

. . .	Horizontal ellipsis points indicate either:
	■ That we have omitted parts of the code that are not directly related to the example
	■ That you can repeat a portion of the code
		`CREATE TABLE ... AS `*`subquery`*`;`
		`SELECT `*`col1`*`, `*`col2`*`, ... , `*`coln`*` FROM employees;`

Convention	Meaning	Example

.
.
.
	Vertical ellipsis points indicate that we have omitted several lines of code not directly related to the example.	`SQL> SELECT NAME FROM V$DATAFILE;`
		`NAME`
		`------------------------------------`
		`/fsl/dbs/tbs_01.dbf`
		`/fs1/dbs/tbs_02.dbf`
		`.`
		`.`
		`.`
		`/fsl/dbs/tbs_09.dbf`
		`9 rows selected.`

Other notation	You must enter symbols other than brackets, braces, vertical bars, and ellipsis points as shown.	`acctbal NUMBER(11,2);`
		`acct    CONSTANT NUMBER(4) := 3;`

Italics	Italicized text indicates placeholders or variables for which you must supply particular values.	CONNECT SYSTEM/*system_password* DB_NAME = *database_name*

UPPERCASE	Uppercase typeface indicates elements supplied by the system. We show these terms in uppercase in order to distinguish them from terms you define. Unless terms appear in brackets, enter them in the order and with the spelling shown. However, because these terms are not case sensitive, you can enter them in lowercase.	`SELECT last_name, employee_id FROM employees;`
		`SELECT * FROM USER_TABLES;`
		`DROP TABLE hr.employees;`

lowercase	Lowercase typeface indicates programmatic elements that you supply. For example, lowercase indicates names of tables, columns, or files.
	Note: Some programmatic elements use a mixture of UPPERCASE and lowercase. Enter these elements as shown.
		`SELECT last_name, employee_id FROM employees;`
		`sqlplus hr/hr`
		`CREATE USER mjones IDENTIFIED BY ty3MU9;`

Conventions for Windows Operating Systems

The following table describes conventions for Windows operating systems and provides examples of their use.

Convention	Meaning	Example
Choose Start >	How to start a program.	To start the Database Configuration Assistant, choose Start > Programs > Oracle - *HOME_NAME* > Configuration and Migration Tools > Database Configuration Assistant.
File and directory names	File and directory names are not case sensitive. The following special characters are not allowed: left angle bracket (<), right angle bracket (>), colon (:), double quotation marks ("), slash (/), pipe (|), and dash (-). The special character backslash (\) is treated as an element separator, even when it appears in quotes. If the file name begins with \\, then Windows assumes it uses the Universal Naming Convention.	`c:\winnt"\"system32 is the same as C:\WINNT\SYSTEM32`

Convention	Meaning	Example
`C:\>`	Represents the Windows command prompt of the current hard disk drive. The escape character in a command prompt is the caret (^). Your prompt reflects the subdirectory in which you are working. Referred to as the *command prompt* in this manual.	`C:\oracle\oradata>`

Special characters	The backslash (\) special character is sometimes required as an escape character for the double quotation mark (") special character at the Windows command prompt. Parentheses and the single quotation mark (’) do not require an escape character. Refer to your Windows operating system documentation for more information on escape and special characters.	`C:\>exp scott/tiger TABLES=emp QUERY=\"WHERE job=’SALESMAN’ and sal<1600\"`
		`C:\>imp SYSTEM/`*`password`*` FROMUSER=scott TABLES=(emp, dept)`

`HOME_NAME`	Represents the Oracle home name. The home name can be up to 16 alphanumeric characters. The only special character allowed in the home name is the underscore.	`C:\> net start Oracle`*`HOME_NAME`*`TNSListener`
`ORACLE_HOME` and *`ORACLE_BASE`*	In releases prior to Oracle8i release 8.1.3, when you installed Oracle components, all subdirectories were located under a top level *`ORACLE_HOME`* directory. For Windows, the default location was `C:\orant`. This release complies with Optimal Flexible Architecture (OFA) guidelines. All subdirectories are not under a top level *`ORACLE_HOME`* directory. There is a top level directory called *`ORACLE_BASE`* that by default is `C:\oracle`. If you install the latest Oracle release on a computer with no other Oracle software installed, then the default setting for the first Oracle home directory is `C:\oracle\ora`*`nn`*, where *`nn`* is the latest release number. The Oracle home directory is located directly under *`ORACLE_BASE`*. All directory path examples in this guide follow OFA conventions. Refer to *Oracle Database Platform Guide for Windows* for additional information about OFA compliances and for information about installing Oracle products in non-OFA compliant directories.	Go to the *`ORACLE_BASE`*`\`*`ORACLE_HOME`*`\rdbms\admin` directory.

1

Introduction to Oracle Application Server Adapter for PeopleSoft

Oracle Application Server connects to a PeopleSoft system through the Oracle Application Server Adapter for PeopleSoft (OracleAS Adapter for PeopleSoft). The OracleAS Adapter for PeopleSoft provides connectivity and executes interactions on a PeopleSoft system. This chapter discusses the following topics:

- Adapter Features
- PeopleSoft Concepts
- Integration with PeopleSoft

Adapter Features

The OracleAS 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. The OracleAS Adapter for PeopleSoft can be deployed as a JCA 1.0 resource adapter. This deployment is referred to as the OracleAS Adapter JCA. It can also be deployed as a Web services servlet and as such is referred to as the Oracle Application Server Adapter Business Services Engine (BSE).

The adapter uses XML messages to enable non-PeopleSoft applications to communicate and exchange transactions with PeopleSoft using services and events.

- Services: Applications use this capability to initiate a PeopleSoft business event.
- Events: Applications use this capability to access PeopleSoft data only when a PeopleSoft business event occurs.

To support event functionality, two features are implemented:

- Port

  A **port** associates a particular business object exposed by an adapter with a particular disposition. A disposition defines the protocol and location of the event data. The port defines the end point of the event consumption.

  The port is the Oracle adapter component that pushes the event received from the EIS to the adapter client. The only port supported in this release is Remote Method Invocation (RMI). It is used for integration with Oracle Application Server Integration InterConnect (OracleAS Integration InterConnect).

- Channel

A **channel** represents configured connections to particular instances of back-end or other types of systems. A channel binds one or more event ports to a particular **listener** managed by an adapter.

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. This enables the user to choose the optimal channel component based on deployment requirements.

The OracleAS Adapter for PeopleSoft provides:

- Synchronous and asynchronous, bidirectional message interactions for PeopleSoft component interfaces and PeopleSoft Application Messaging Manager.
- Oracle Application Server Adapter Application Explorer (Application Explorer), a GUI tool that explores PeopleSoft metadata and builds XML schemas or Web services.
- XML schemas for the JCA 1.0 resource adapter.
- Web services for the OracleAS Adapter 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*

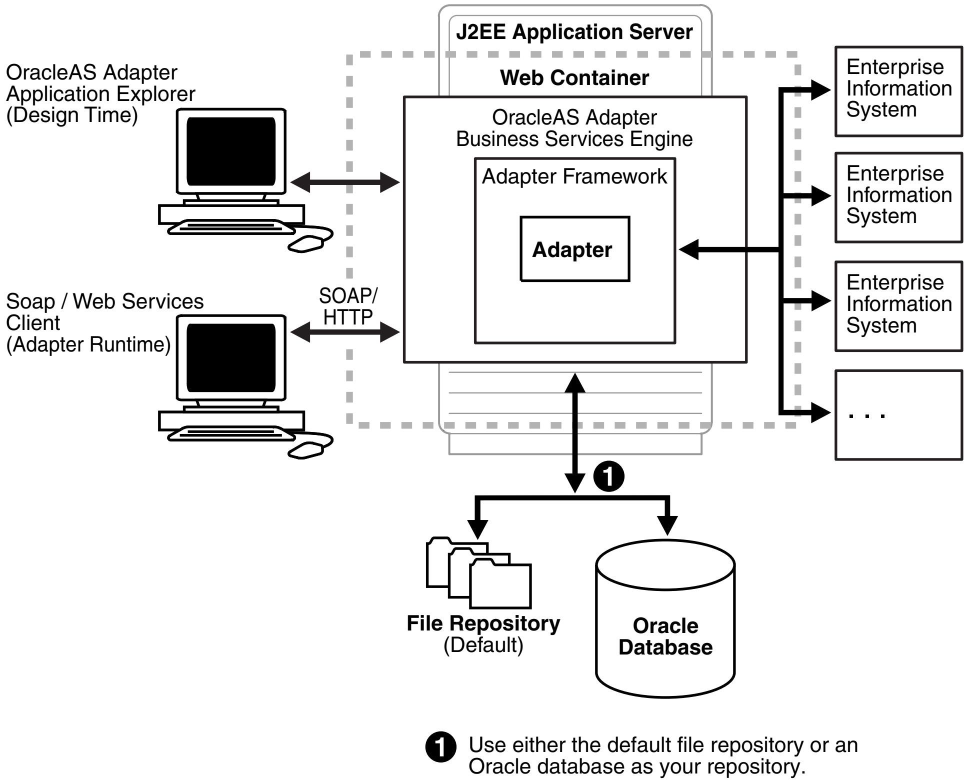
Oracle Application Server Adapter Business Services Engine Architecture

Figure 1–1 shows the generic architecture for the Oracle Web service adapter for packaged applications. The adapter works in conjunction with the Oracle Application Server Adapter Business Services Engine (BSE), as deployed to a Web container in a J2EE application server.

Oracle Application Server Adapter Application Explorer (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 consuming requests from clients, interacting with the EIS, and sending responses from the EIS back to clients.

Figure 1–1 Oracle Application Server Adapter Business Services Engine Architecture

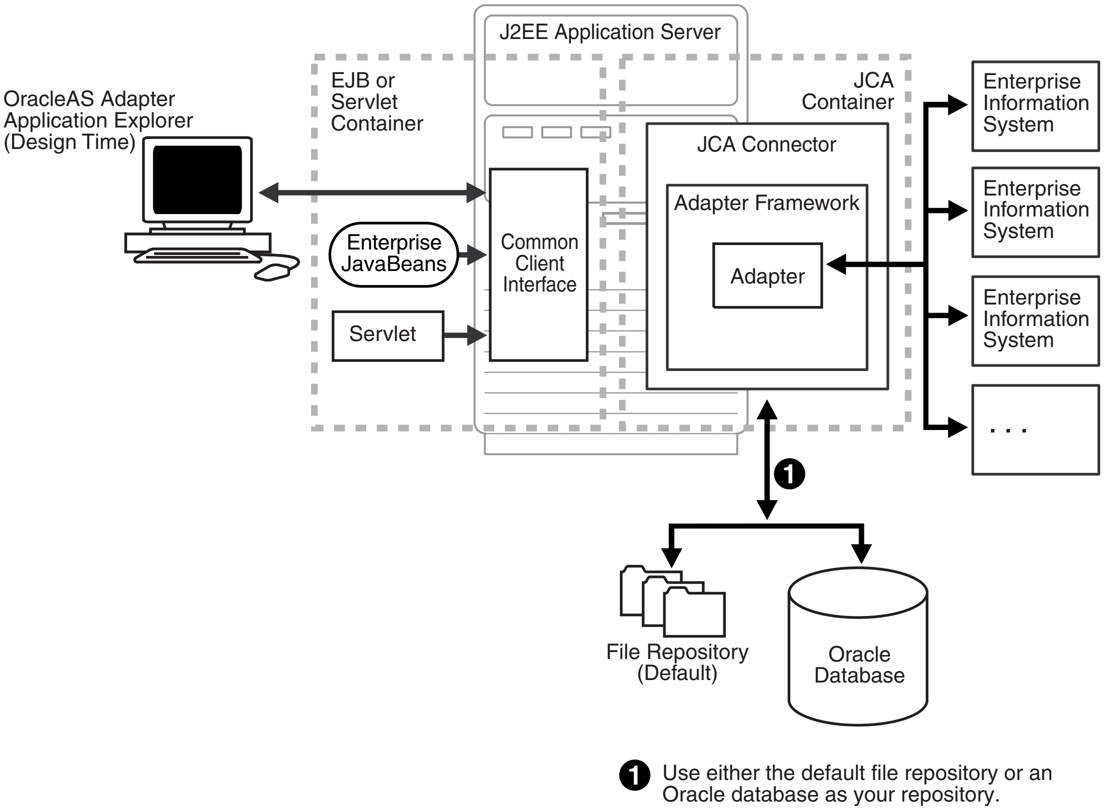

J2EE Application Server
Web Container
OracleAS Adapter Application Explorer (Design Time)
OracleAS Adapter Business Services Engine
Adapter Framework
Adapter
Enterprise Information System
Enterprise Information System
Enterprise Information System
. . .
Soap / Web Services Client (Adapter Runtime)
SOAP/ HTTP
1
File Repository (Default)
Oracle Database
1 Use either the default file repository or an Oracle database as your repository.

Oracle Application Server Adapter Generic JCA Architecture

Figure 1–2 shows the generic architecture for the OracleAS Adapter JCA for packaged applications. The OracleAS Adapter JCA is deployed to a standard JCA Container and serves as host container to the adapters. The connector is configured with a repository.

Application Explorer, a design-tool that works in conjunction 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 Application Server Adapter Generic JCA Architecture



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

PeopleSoft Concepts

PeopleSoft provides for 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. The OracleAS 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 the OracleAS Adapter for PeopleSoft include:

- PeopleSoft Java API using component interfaces
- PeopleSoft XML using Application Messaging

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 also can develop their own custom component interfaces, or they can modify EIP as required.

PeopleSoft Application Messaging Manager

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

Integration with PeopleSoft

The OracleAS Adapter for PeopleSoft enables you to:

- Access a PeopleSoft component using a component interface. Component methods are also referred to as services.
- Receive messages from a PeopleSoft application. Messages are also 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, "Adapter Configuration Using OracleAS Adapter Application Explorer" 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 Chapter 1, "Introduction to Oracle Application Server Adapter for PeopleSoft" for more information.

See Also:

- *Oracle Application Server Adapter Concepts*
- *Oracle Application Server Adapters Installation Guide*

2

Adapter Configuration Using OracleAS Adapter Application Explorer

OracleAS 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 Settings for BSE or JCA
- Creating a Repository Project
- Establishing a Connection (Target) for PeopleSoft
- Viewing Application System Objects
- Creating a Web Service or a Business Service
- Configuring an Event Adapter

Starting Application Explorer

The server must be started where Application Explorer is deployed.

To start Application Explorer:

1. Ensure the server is started where Application Explorer is deployed.
2. On Windows, invoke `iaexplorer.exe`, found under *`OracleAS_home`*\adapters\application\tools or on UNIX, invoke the iwae script, `iwae.sh`, found under *`OracleAS_home`*/adapters/application/tools.

OracleAS_home
Is the directory where the Oracle Application Server is installed.

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

Configuring Settings for BSE or JCA

Before a repository project can be created, you must configure BSE. You need not configure the OracleAS Adapter JCA because the `ra.xml` file is configured automatically during installation.

Configuring the OracleAS Adapter Business Services Engine

After BSE is deployed to Oracle Application Server, you can configure it through the BSE configuration page. This configuration is required only when pointing to BSE using a database repository.

To configure BSE:

1. Open the following page in your browser:

   ```
   http://hostname:port/ibse
   ```

hostname
Is the hostname of the Oracle Application Server.

port
Is the HTTP port for the Oracle Application Server.

For example,

```
http://localhost:7777/ibse
```

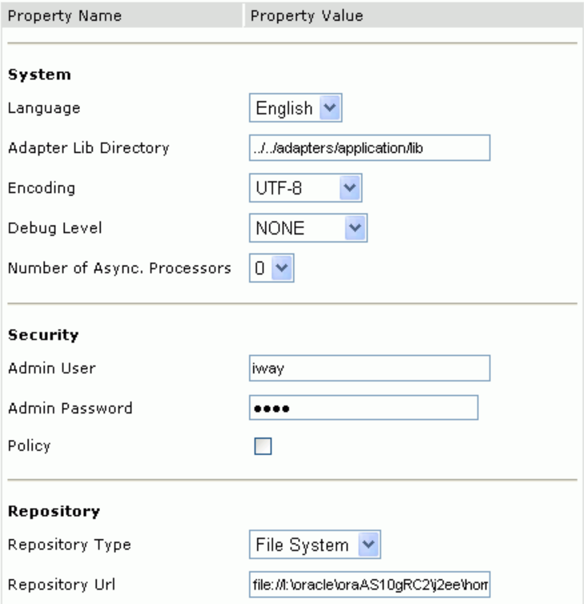
> **Note:** The first time you access this page, it may take time to load.

2. When prompted, log on.

   When first installed, the user ID and the password are:

   - User name: iway
   - Password: iway

     The BSE configuration page opens.



Property Name	Property Value
System	
Language	English
Adapter Lib Directory	../../adapters/application/lib
Encoding	UTF-8
Debug Level	NONE
Number of Async. Processors	0
Security	
Admin User	iway
Admin Password	●●●●
Policy	□
Repository	
Repository Type	File System
Repository Url	file:///l:\oracle\oraAS10gRC2\j2ee\hom

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

   ```
   OracleAS_home\adapters\application\lib
   ```

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

4. For security purposes, type a new password in the **Admin Password** field.

   > **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. See "Configuring an Oracle Repository" on page 2-5 for information on switching to a database repository.

5. Click **Save.**

Configuring BSE System Settings

To configure BSE system settings:

1. Open the **BSE configuration** page by entering the following URL:

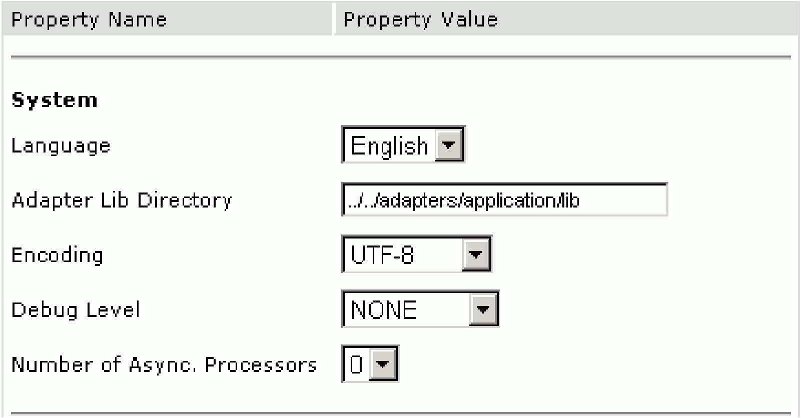
   ```
   http://hostname:port/ibse/IBSEConfig
   ```

hostname
Is the machine where BSE is installed.

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 opens.



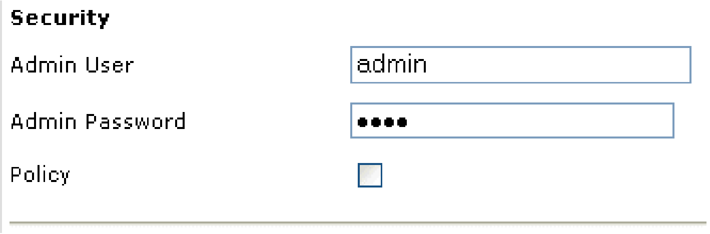
Property Name	Property Value
System	
Language	English
Adapter Lib Directory	../../adapters/application/lib
Encoding	UTF-8
Debug Level	NONE
Number of Async. Processors	0

2. Configure the system settings.

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

Parameter	Description
Language	Specify your required language.
Adapter Lib Directory	Type the full path to the directory where the adapter jar files reside.
Encoding	Specify the default encoding from one of the following options: UTF-8 EBCDIC-CP-US ISO-88859-1 Shift JIS UNICODE
Debug Level	Specify the debug level from one of the following options: None Fatal Error Warning Info Debug
Number of Async. Processors	Select the number of asynchronous processors.

3. Configure the security settings.



Security

Admin User: admin

Admin Password: ●●●●

Policy: □

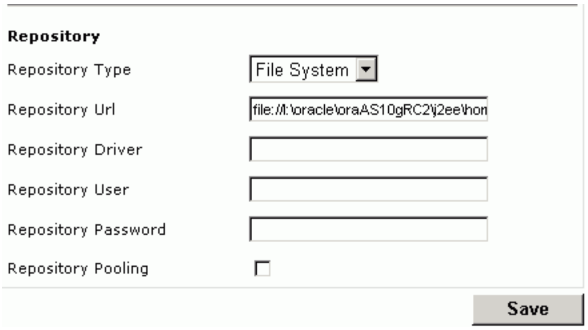
The following table lists the parameters with descriptions of the information to provide.

Parameter	Description
Admin User	Provide a BSE administrator ID.
Admin Password	Type the password associated with the BSE administrator ID.
Policy	Select the check box to enable policy security.

4. 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 2-5 and "Configuring an Oracle Repository" on page 2-5 for more information.



Repository

Repository Type	File System
Repository Url	file://I:\oracle\oraAS10gRC2\j2ee\hon
Repository Driver	
Repository User	
Repository Password	
Repository Pooling	□

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: Oracle File
Repository URL	Type the URL to use when opening a connection to the database.
Repository Driver	Provide the driver class to use when opening a connection to the database (optional).
Repository User	Type the user ID to use when opening a connection to the database.
Repository Password	Type the password associated with the user ID.
Repository Pooling	Select the check box to enable pooling.

5. 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 machine. 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.

The default location for the repository on Windows is:

```
OracleAS_home\j2ee\OC4J_CONTAINER\applications\ws-app-adapter
\ibse\ibserepo.xml
```

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.

Configuring an Oracle Repository

To configure an Oracle repository:

1. Contact your database administrator to obtain an Oracle user ID and password to create the BSE repository.

   This user ID should have rights to create and modify tables as well as the ability to create and execute stored procedures.

2. Open a command prompt and navigate to the setup directory. Its default location on Windows is:

```
OracleAS_home\adapters\application\etc
```

For other platforms, see the corresponding location.

This directory contains SQL to create the repository tables in the following file:

```
iwse.ora
```

> **Note:** If Oracle is not on the same machine as the Oracle Application Server, copy the iwse.ora file to the Oracle machine. Then, from a command prompt on the Oracle machine, navigate to the directory containing the iwse.ora file.

3. Type the following command:

```
sqlplus userid/password @database @ iwse.ora
```

Creating a Repository Project

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

At 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 runtime.

A default JCA repository is created for the default ManagedConnectionFactory. The name of this project is `jca_sample`.

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

Creating a Web Service Project for the Web Service Adapter

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

Defining a New Configuration for BSE

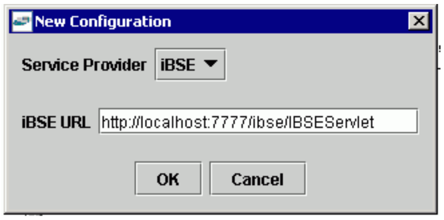
To create a new repository project for BSE:

1. Right-click **Configurations** and select **New.**

   The New Configuration dialog box opens.

2. Enter a name for the new configuration (for example, SampleConfig) and click **OK.**

   The following opens.



Service Provider: iBSE

iBSE URL: http://localhost:7777/ibse/IBSEServlet

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

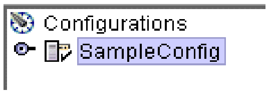
   ```
   http://hostname:port/ibse/IBSEServlet
   ```

hostname
Is the machine where your application server resides.

port
Is the port number where the application server is listening.

5. Click **OK.**

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



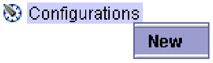
Creating a Repository Project for the OracleAS Adapter JCA

To create a repository project for the OracleAS Adapter JCA using Application Explorer, you must first define a new configuration.

Defining a New Configuration for JCA

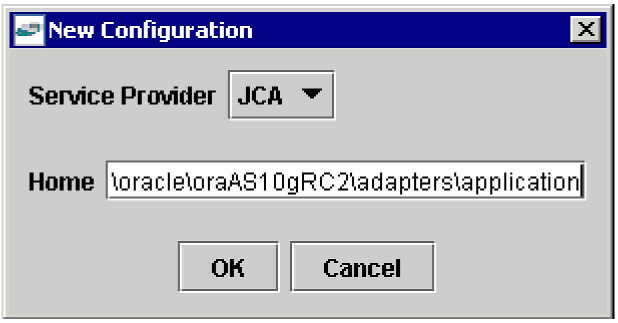
To define a new configuration for JCA:

1. Right-click **Configurations** and select **New.**



   The New Configuration dialog box opens.

2. Enter a name for the new configuration (for example, SampleConfig) and click **OK.**



Service Provider: JCA

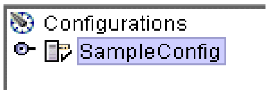
Home: \oracle\oraAS10gRC2\adapters\application

3. From the **Service Provider** list, select **JCA.**
4. In the **Home** field, enter a path to your JCA configuration directory where the repository, schemas, and other information is stored, for example:

   ```
   OracleAS_home\adapters\application
   ```

5. Click **OK.**

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



The OracleAS Adapter JCA repository project file is stored in `*OracleAS_home*\adapters\application\config\*configuration_name*`

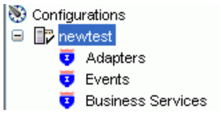
`configuration_name`
Is the name of the configuration you created; for example, SampleConfig.

Connecting to a New Configuration

To connect to a new configuration:

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

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



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 to configure listeners that listen for events in PeopleSoft.

Use the **Business Services** folder 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.

Establishing a Connection (Target) for PeopleSoft

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

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.

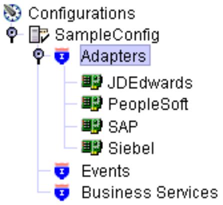
When you launch Application Explorer, the left pane displays (as nodes) the application systems supported by Application Explorer, based on the adapters that are installed.

Creating a New Target

To connect to PeopleSoft for the first time, you must create a new 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 box opens.

   a. In the **Name** field, type a descriptive name, for example, PsoftTarget.

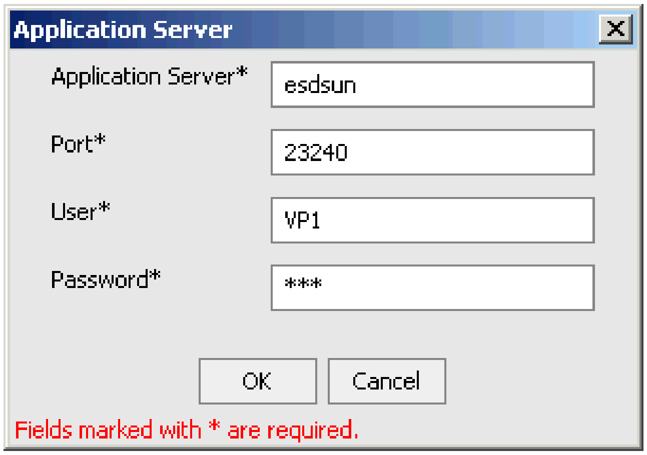
   b. In the **Description** field, type a description for the target (optional).

   c. From the **Target Type** list, select a target type.

      The default value is Application Server.

3. Click **OK.**

   The Application Server dialog box opens where you must specify connection information for PeopleSoft and the application server that is hosting PeopleSoft.



Application Server

Application Server* esdsun

Port* 23240

User* VP1

Password* ***

Fields marked with * are required.

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

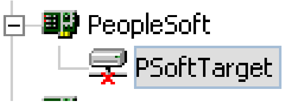
   b. In the **Port** field, type the port number where the PeopleSoft application is listening.

   c. In the **User** field, type a valid user ID for the PeopleSoft application.

   d. In the **Password** field, type a valid password for the PeopleSoft application.

4. Click **OK.**

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



You are ready to connect to your PeopleSoft target.

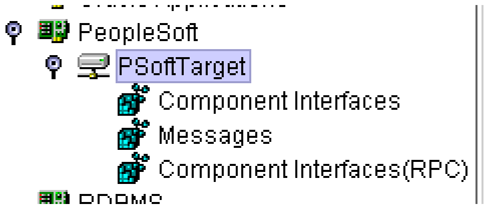
Connecting to a Defined PeopleSoft Target

To connect to a 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 box opens, populated with values you entered for the connection parameters.
4. Verify your connection parameters. If required, provide the password.
5. Right-click the target name and select **Connect.**

   The x icon disappears, indicating that the node is connected.
6. Expand the **PsoftTarget** node to reveal the list of PeopleSoft business objects.



Managing a Target

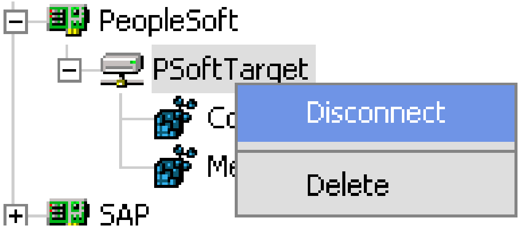
Although you can maintain multiple open connections to different transaction processing systems, it is recommended that you disconnect from connections not in use. After you disconnect, you can modify an existing target.

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.**

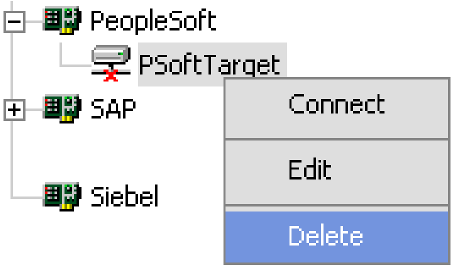
   Disconnecting from PeopleSoft drops the connection with PeopleSoft, but the node remains.

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

Deleting a Connection to PeopleSoft

To delete 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 **Delete.**

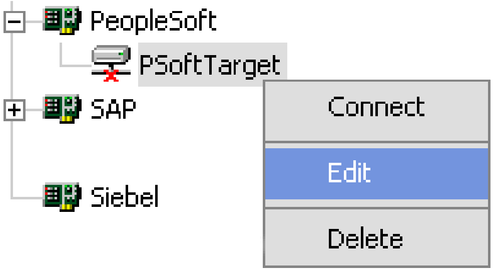
   The node disappears from the list of available connections.

Modifying a Target

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

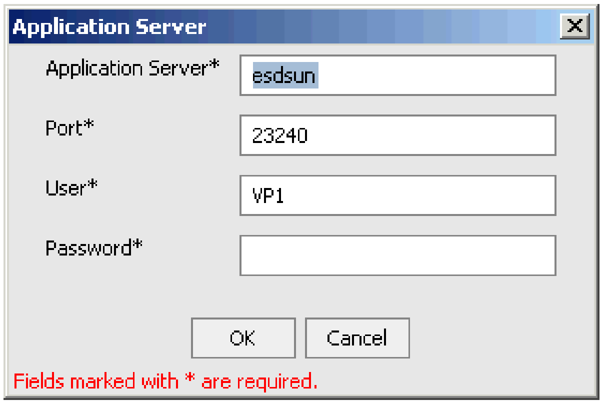
To edit a target:

1. Ensure that the target you want to edit is disconnected.



2. In the left pane, right-click the target and select **Edit.**

   The Application Server dialog box opens.



Application Server

Application Server* esdsun

Port* 23240

User* VP1

Password*

Fields marked with * are required.

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

Viewing Application System Objects

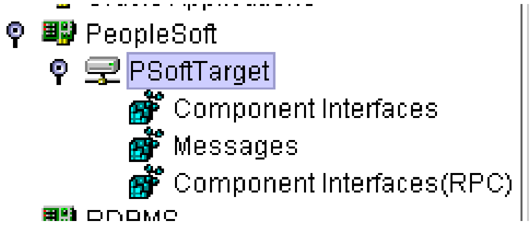
After you are connected to PeopleSoft, Application Explorer enables you to explore and browse business object metadata. For example, Application Explorer enables you to view PeopleSoft Component Interface and Message metadata stored in the PeopleSoft business object repository.

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 to expose the available system objects.



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.

Creating XML Schemas

After you browse the PeopleSoft business object repository, you can generate XML request and response schemas for the object you wish 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 OraclesAS Adapter JCA

To create XML request and response schemas for a PeopleSoft Component Interface against an OraclesAS Adapter JCA 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 a Web Service or a Business Service

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 (JCA) implementation of adapters, Web services are not available. When the adapters are deployed to use the OracleAS Adapter for JCA, the Common Client Interface provides integration services using the adapters.

To create a Web service:

1. Expand the **PeopleSoft** node and then, expand 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 Business Service.**

   The Create Web Service dialog box opens.

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

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

   b. Specify a service name if you are creating a new service. This name identifies the Web service in the list of services under the **Business Services** node.

   c. Type a description for the service (optional).

   d. Select one of the available licenses.
4. Click **Next.**

   The License and Method dialog box opens.

   a. In the **License** 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, type a descriptive name for the method.

   c. In the **Description** field, type a brief description of the method.
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 Business 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. If you are not on the Business Services node of Application Explorer, click the node to access Web services.
2. If it is not expanded, expand the list of Web services under **Business Services.**
3. Expand the **Services** node.
4. Select the name of the business service you want to test.

   The business service name appears as a link in the right pane.
5. In the right pane, click the named business services link.

   The test option appears in the right pane.

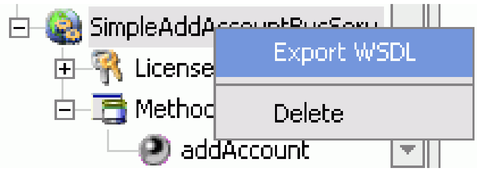
   If you are testing a Web service that requires XML input, an input field appears.
6. Enter the appropriate input.
7. Click **Invoke.**

   Application Explorer displays the results.

Generating Web Services Description Language from a Web Service

Generating Web Services Description Language (WSDL) from a Web service enables you to make the Web service available to other services within a host server.

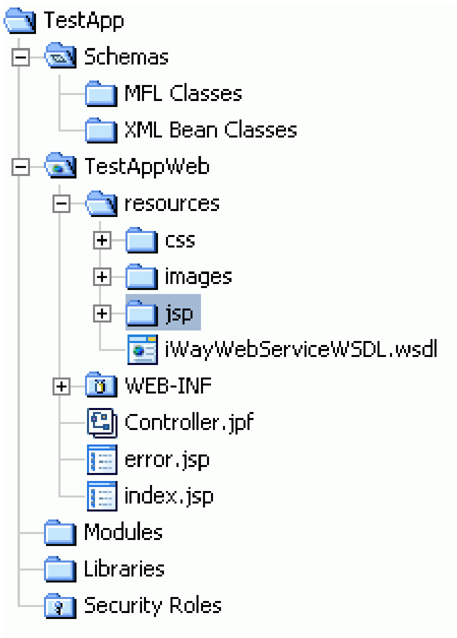
To generate WSDL from a Web service:



1. After you create a Web service, right-click the Web service name and select **Export WSDL.**

   The Save dialog box opens.
2. Choose a location for the file and add a **.wsdl** file extension.
3. Click **Save.**

   The WSDL file appears under the resources folder of your Web application:



Configuring an Event Adapter

Events are generated as a result of 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.

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

- Port

A 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. See "Creating an Event Port" on page 2-15 for more information.

- Channel

A 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 the adapter. See "Creating a Channel" on page 2-17 for more information.

> **Note:** Oracle Containers for J2EE (OC4J) currently conforms to JCA 1.0, which does not call for event capabilities. When conforming to JCA 1.0, only service interactions are supported.

Creating an Event Port

Application Explorer enables you to create event ports from the Adapters node or from the Events node.

Creating an Event Port from the Adapters Node

You can bypass the Events node and create an event port directly from the Adapters node.

1. Select the PeopleSoft Integration Object for which you want to create an event port.
2. Right-click the component and select **Add Port.**

   The Add Port dialog box opens.

   a. Type a name for the event port and provide a brief description.

   b. From the list, select the required disposition, for example, File.

   c. Type the disposition url.
3. Click **OK.**

   See "Creating an Event Port from the Events Node" on page 2-16 for information on configuring port dispositions.

Creating an Event Port from the Events Node

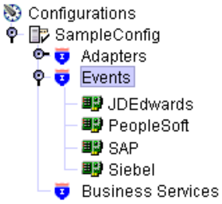
The following procedures describe how to create an event port from the Events node for various dispositions using Application Explorer. You can switch between a BSE and a JCA deployment by choosing one or the other from the menu in the upper right of Application Explorer.

See "Creating an Event Port from the Events Node" on page 2-16 for information on creating an event port directly from the Adapters node.

Creating an Event Port for RMI

To create a specific event port for RMI:

1. Click the **Events** node.



2. Expand the **PeopleSoft** node.
3. Right-click the **Ports** node and select **Add Port.**

   The Add Port dialog box opens.

   a. Type a name for the event port and provide a brief description.

   b. From the **Protocol** list, select **RMI.**

   c. In the **URL** field, specify a destination file to which the event data is written using the following format:

   ```
   rmi://host:port;RemoteObject=[APPNAME];errorTo=[pre-defined port name or
   another disposition url]
   ```

   d. From the **Disposition** protocol list, select **RMI.**

   The following table defines the parameters for the disposition.

Parameter	Description
host	The host name or IP address from which the RMI server accepts RMI requests. If you omit this attribute, the RMI server will accept RMI requests from any host.
port	The port number on which the RMI server listens for RMI requests.
RemoteObject	A home or Enterprise JavaBeans (EJB) object.
errorTo	Predefined port name or another disposition URL to which error logs are sent.

4. Click **OK.**

   The port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created.

   You are ready to associate the event port with a channel. See "Creating a Channel" on page 2-17 for more information.

The port appears under the ports node in the left pane. In the right pane, a table appears that summarizes the information associated with the event port you created.

Editing an Event Port

To edit an event port:

1. In the left pane, select the event port you want to edit.
2. Right-click the port and select **Edit.**

   The Edit Port pane opens.

3. Make the required changes and click **OK.**

Deleting an Event Port

To delete an event port:

1. In the left pane, select the event port you want to delete.
2. Right-click the port and select **Delete.**

   A confirmation dialog box opens.

3. To delete the event port you selected, click **OK.**

   The event port disappears from the list in the left pane.

Creating a Channel

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

Creating a Channel Using Specific Protocols

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

- TCP
- HTTP
- File

The following procedures explain how to create these channels.

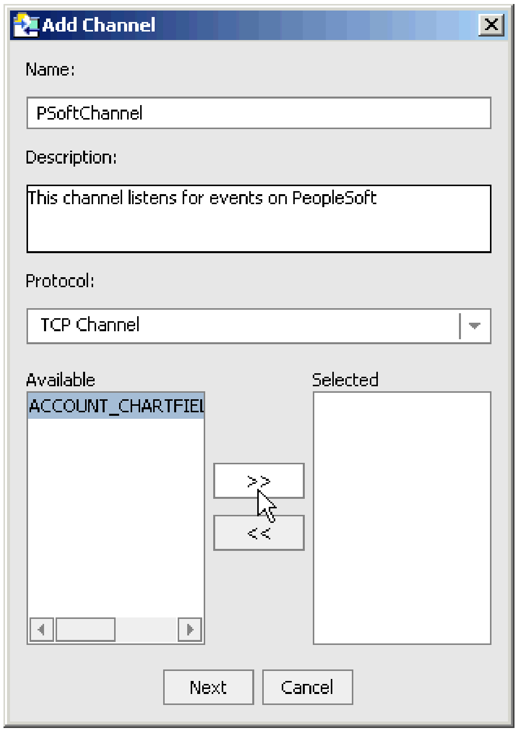
> **Note:** OC4J currently conforms to JCA 1.0, which does not call for event capabilities. When conforming to JCA 1.0, only service interactions are supported.

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 box opens.



Name: PSoftChannel

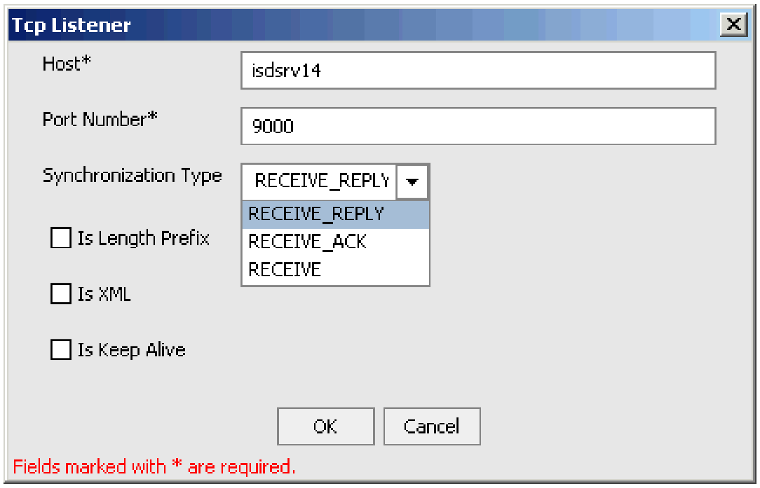
Description: This channel listens for events on PeopleSoft

Protocol: TCP Channel

ACCOUNT_CHARTFIEL

   a. Type a name for the channel, for example, NewChannel.
   b. Type a brief description.
   c. From the **Protocol** list, select **TCP Channel.**
   d. Click the double right **(>>)** arrow button to transfer the ports to the list of selected ports.
4. Click **Next.**
5. Enter the information that is specific to your PeopleSoft system and the channel you are creating.
6. Click **Next.**

   The TCP Listener dialog box opens.



Tcp Listener

Host* isdsrv14

Port Number* 9000

Synchronization Type RECEIVE_REPLY

□ Is Length Prefix

□ Is XML

□ Is Keep Alive

Fields marked with * are required.

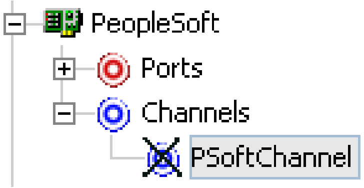
a. In the **Host** field, specify the host for your TCP listener (required).

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

c. From the **Synchronization Type** list, select the synchronization type for your TCP listener.

7. Click **OK.**

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



   When you select the event port, the channel information appears in the right pane.

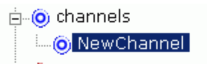
8. Click **OK.**

   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.

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

   The channel becomes active.



   The X that was over the icon in the left pane disappears.

10. To stop the channel, right-click the connected channel node and select **Stop.**

    The channel becomes inactive and an X appears over the icon.

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 box opens.

   a. Type a name for the channel, for example, NewChannel.

   b. Type a brief description.

   c. From the list, select **HTTP Listener.**

   d. Select an event port from the list of available ports.

      To transfer the port to the list of available ports, click the double right **(>>)** arrow. To associate all the event ports, control-click to select each port or click one port and press **Control+A.** Then, click the double right **(>>)** arrow button.

4. Click **Next.**

5. When the dialog box opens, type the system information as specified in the following table:

Parameter	Description
Port	Port on which to listen for PeopleSoft event data.
Server port	Port on which the host database is listening.
Synchronization Type	Choose from three synchronization options: REQUEST REQUEST_RESPONSE REQUEST_ACK

6. Click **OK.**

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

   The channel also 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.

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

   The channel you created becomes active.

   The X that was over the icon in the left pane disappears.

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

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 box opens.

a. Type a name for the channel, for example, NewChannel.

b. Type a brief description.

c. From the list, select **File Listener.**

d. Select an event port from the list of available ports.

To transfer the port to the list of available ports, click the double right **(>>)** arrow. To associate all the event ports, control-click to select each port or click one port and press **Control+A.** Then, click the double right **(>>)** arrow button.

4. Click **Next.**

5. When the dialog box opens, type 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 as a result of the operation.

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

Parameter	Description
Synchronization Type	Target file system location for the PeopleSoft XML file.
Response/Ack Directory	Choose from three options: REQUEST REQUEST_RESPONSE REQUEST_ACK

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 opens, providing the channel description, channel status, and available ports. All the information is associated with the channel you created.

The channel also 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.

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

The channel you created becomes active.

The X that was over the icon in the left pane disappears.

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

Editing a Channel

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 opens.
3. Make the required changes to the channel configuration and click **Finish.**

Deleting a Channel

To delete a channel:

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

   A confirmation dialog box opens.
3. To delete the channel you selected, click **OK.**

   The channel disappears from the list in the left pane.

3

Deployment and Integration

This chapter describes Oracle Containers for J2EE (OC4J) deployment and integration with OracleAS Integration InterConnect.

This chapter discusses the following topics:

- OC4J Integration
- OracleAS Adapter BSE Integration with OracleAS Integration InterConnect

See Also:

- *Oracle Application Server Integration InterConnect User's Guide*
- *Oracle Application Server Containers for J2EE User's Guide*

OC4J Integration

The following topic shows the basic commands for using CCI with packaged application adapters.

See Also:

- "OC4J Containers" in *Oracle Application Server Adapter Concepts*
- "Deployment and Integration through J2CA" in *Oracle Application Server Adapter Concepts*

Application Development Using the CCI API

The following example shows the code structure for using CCI with packaged application adapters. The code sample is shown in five steps.

Step 1. Obtain the Connection Factory

The connection factory is obtained by JNDI lookup.

```
InitialContext context = new InitialContext();
ConnectionFactory cf = (ConnectionFactory)context.lookup(iwayJndi)
```

Step 2. Obtaining a Connection for the Adapter

IWAFConnectionSpec is an implementation of ConnectionSpec used for creating a design time or runtime service adapter connection. The ConnectionSpec has seven parameters. Connection Pooling is fully supported and established based on these parameters, except log level.

Parameter Name	Description
adapterName	Name of the packaged application adapter.
config -	Adapter configuration name. NOT REQUIRED FOR IWAEAdapter.
language	Default is en.
country	Default is us.
userName	User name. If provided, it overwrites configuration.
password	Password. If provided, it overwrites configuration.
logLevel	It overwrites the level set by the ManagedConnectionFactory property.

> **Note:** Currently the OracleAS Adapter JCA supports only basic security mapping. The DEBUG log level provides detailed information on the mapping behavior. It functions as follows:
>
> - If the userName and password are not set, and no security is provided by the application server, the OracleAS Adapter JCA will still let it pass and rely on the adapter configuration security information.
> - If userName and password are set, these values will overwrite the adapter configuration. The OracleAS Adapter JCA compares this information with the security information provided by the application server and log in case the values do not match. However, it still allows the information through.

The iWAFConnectionSpec can be made to invoke an interaction with PeopleSoft by specifying the adapter name and configuration parameters in the ConnectionSpec. For example,

```
iWAFConnectionSpec cs = new IWAFConnectionSpec();
  cs.setAdapterName(ADAPTER);
  cs.setConfig(TARGET);
  cs.setLogLevel(LOG_LEVEL);  // Adapter layer log level
  Connection c = cf.getConnection(cs);// where cf is the connection factory
```

In this snippet, `ADAPTER` and `TARGET` refer to the adapter being invoked, in this case PeopleSoft, and the name of a target defined in Application Explorer. For more information, see "Complete Code Sample" on page 3-3.

Step 3. Create interaction with interactionSpec for runtime

```
Interaction i = c.createInteraction();
  IWAFInteractionSpec is = new IWAFInteractionSpec();
  is.setFunctionName(IWAFInteractionSpec.PROCESS);
```

Two functions can be set: PROCESS and IWAE. PROCESS are used at runtime. IWAE is used when you are using the IAEAdapter at design time.

Step 4. Create Input Record and Execute Interaction

In this case, to complete the EIS invocation, a PeopleSoft message is referenced. The schema is provided by Application Explorer.

A standard JCA indexed record is used in this example:

```
// Use JCA IndexRecord, named "input" for runtime processing.
IndexedRecord rIn = cf.getRecordFactory().createIndexedRecord("input");
rIn.add(msg_run);
  IndexedRecord rOut = (IndexedRecord)i.execute(is, rIn);
System.out.println((String)rOut.get(0));
```

A special record is supported in this example:

```
//IWAFRecord rIn = new IWAFRecord("input");
//rIn.setRootXML(msg_run);
//IWAFRecord response = executeRunInteraction(c, rIn);
  //IWAFRecord rOut = (IWAFRecord)i.execute(is, rIn);
//System.out.println(rOut.getRootXML());
```

msg_run

Is an instance XML document generated from the schema created by Application Explorer. For example, the following is a sample PeopleSoft request XML document.

```
<?xml version="1.0" encoding="UTF-8" ?>
<PS8>
  <component perform="browse">LOCATION</component>
  <key name="Setid">SHARE</key>
  <key name="Location">ALBERTA</key>
</PS8>
```

Complete Code Sample

The following is a sample of the complete code:

```
import javax.resource.cci.*;
import com.ibi.afjca.cci.*;
import com.ibi.afjca.spi.*;

/**
 * The purpose of this sample is to illustrate how to use the IWAF Universal
 * JCA connector.
 *
 * Author: Marcelo Borges
 * Date: August, 2004
 */
public class IWAFJCASimple {

private static String  HOME    = "c:/iway/xfoc/components/iwafcont/dist";
private static String  CONFIG    = "base";
private static String  LOG_LEVEL = "FATAL";

private static String ADAPTER = "PeopleSoft";
private static String TARGET  = "PeopleSoft_connection";

// Input Message
private static String msg_run = "<PeopleSoft/>";

  public static void main(String[] args) throws Exception {

// 1. Getting the Connection factory through JNDI lookup
// ----------------------------------------------------------
```

```
  InitialContext context = new InitialContext();
  ConnectionFactory cf = (ConnectionFactory)context.lookup(iwayJndi)
  // 2. Getting a connection for a particular adapter target, in this case
PeopleSoft
// ------------------------------------------------------------
  IWAFConnectionSpec cs = new IWAFConnectionSpec();
  cs.setAdapterName(ADAPTER);
  cs.setConfig(TARGET);
  cs.setLogLevel(LOG_LEVEL);  // Adapter layer log level
  Connection c = cf.getConnection(cs);// where cf is the connection factory

  // 3. Create interaction with interactionSpec for RUNTIME
// ------------------------------------------------------------
  Interaction i = c.createInteraction();
  IWAFInteractionSpec is = new IWAFInteractionSpec();
  is.setFunctionName("PROCESS");

  // 4. Create input Record and execute interaction
// ------------------------------------------------------------

  // 4.1 Using JCA standard Indexed Record
// Use JCA IndexRecord, named "input" for runtime processing.
IndexedRecord rIn = cf.getRecordFactory().createIndexedRecord("input");
rIn.add(msg_run);
  IndexedRecord rOut = (IndexedRecord)i.execute(is, rIn);
System.out.println((String)rOut.get(0));

  // 4.2 Our own Record is supported here
//IWAFRecord rIn = new IWAFRecord("input");
//rIn.setRootXML(msg_run);
//IWAFRecord response = executeRunInteraction(c, rIn);
  //IWAFRecord rOut = (IWAFRecord)i.execute(is, rIn);
//System.out.println(rOut.getRootXML());

  } // main()

}
```

Creating a Managed Connection Factory

The `OC4J-ra.xml` descriptor provides OC4J-specific deployment information for resource adapters. For example, the default jca_sample configuration in Application Explorer is represented in the `OC4J-ra.xml` file as follows:

```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD
Oracle Connector 9.04//EN"
"http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

The parameters are defined in the following table:

Parameter Name	Description
IWayHome	The base installation directory for the OracleAS packaged application adapter.
IWayConfig	The adapter configuration name as defined in Application Explorer. For example, the OracleAS Adapter for PeopleSoft has a preconfigured jca_sample configuration in the 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 BSE repository. See "Configuring BSE System Settings" on page 2-3 for more information.
IWayRepoUser	User name to use when connecting to the database. This is necessary only when using an Oracle database as the BSE repository. See "Configuring BSE System Settings" on page 2-3 for more information.
IWayRepoPassword	Password. If provided, it overwrites configuration. This is necessary only when using an Oracle database as the BSE repository. See "Configuring BSE System Settings" on page 2-3 for more information.
loglevel	It overwrites the level set by the ManagedConnectionFactory property.

Creating Multiple Managed Connection Factories

To establish Multiple Managed Connection Factories, you must edit the OC4J-ra.xml file to add the required information. The file can contain more than one <connector-factory> element. By adding more <connector-factory> elements, you can create Multiple Managed connection factories. For example, the default jca_sample configuration in Application Explorer is represented in the OC4J-ra.xml file as follows:

```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD
Oracle Connector 9.04//EN"
"http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

To create Multiple Managed Connection Factories, you must add new <connector-factory> nodes in the file. For example:

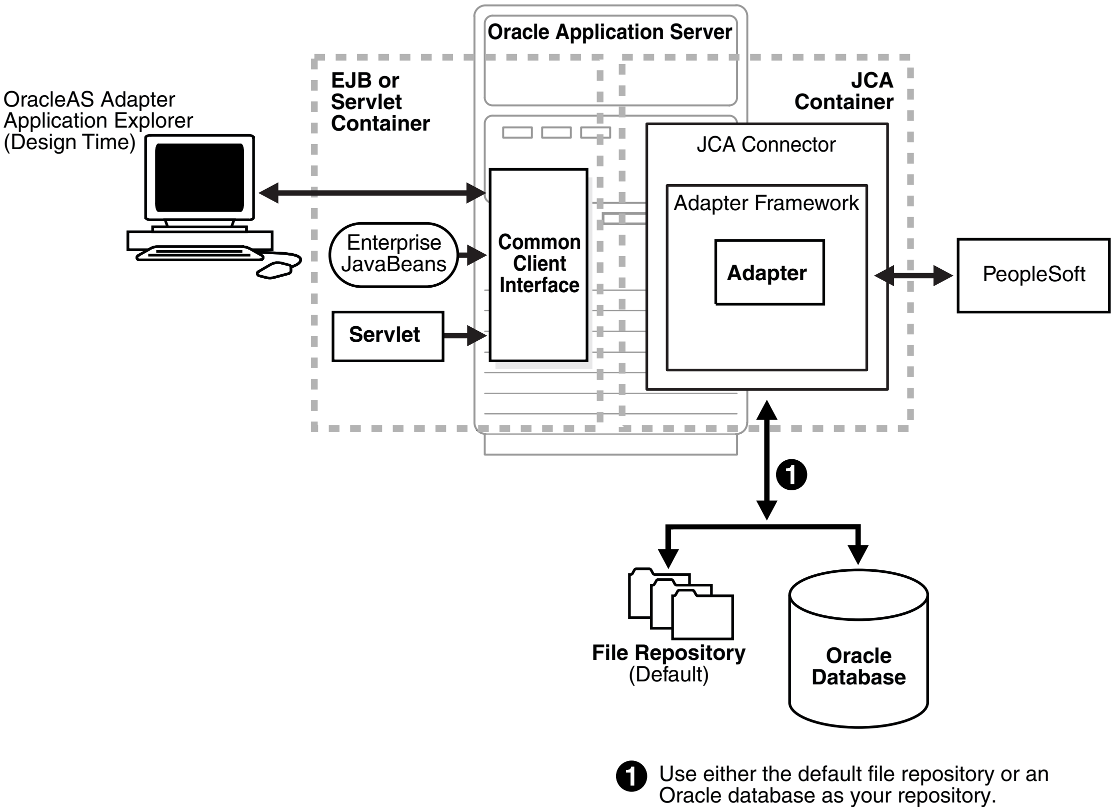
```
<?xml version="1.0"?>
<!DOCTYPE oc4j-connector-factories PUBLIC "-//Oracle//DTD
Oracle Connector 9.04//EN"
"http://xmlns.oracle.com/ias/dtds/oc4j-connector-factories-9_04.dtd">
```

```
<oc4j-connector-factories>
   <connector-factory location="eis/OracleJCAAdapter/DefaultConnection1"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
<connector-factory location="eis/OracleJCAAdapter/DefaultConnection2"
connector-name="IWAFJCA10">
   <config-property name="IWayHome" value="../../adapters/application"/>
   <config-property name="IWayConfig" value="jca_sample2"/>
   <config-property name="IWayRepoURL" value=""/>
   <config-property name="IWayRepoUser" value=""/>
   <config-property name="IWayRepoPassword" value=""/>
   <config-property name="logLevel" value="debug"/>
   </connector-factory>
</oc4j-connector-factories>
```

Oracle Application Server Adapter JCA Architecture

Figure 3–1 shows deployment of the Connector to the Oracle Application Server. In a runtime service scenario, an EJB, Servlet, or Java program client makes CCI calls to JCA 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 Adapter JCA Architecture



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

OracleAS Adapter BSE Integration with OracleAS Integration InterConnect

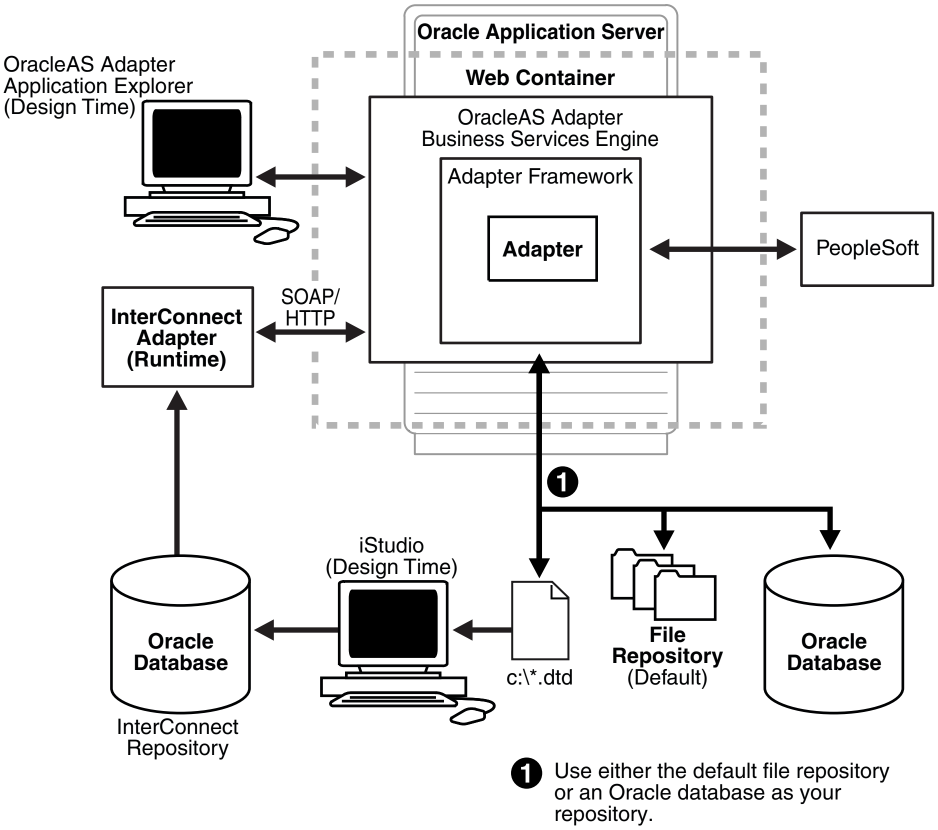
See Also: "Deployment and Integration through OracleAS Web Services" in *Oracle Application Server Adapter Concepts*

BSE Architecture as Deployed to Oracle Application Server

Figure 3–2 shows adapter framework deployment with BSE to OracleAS Integration InterConnect. In a runtime service scenario, the OracleAS Integration InterConnect EIS Adapter Plugin (EIS Adapter Plugin) receives DTD-compliant XML from the **agent** component of the EIS Adapter Plugin. The EIS Adapter Plugin strips runtime information from the XML, wraps the XML in a SOAP envelope, and sends the result to BSE, including the runtime information in the SOAP request. BSE receives the request, removes the envelope, retrieves Web service method metadata, including adapter and connection information from the repository, and makes the adapter request.

BSE receives the adapter response, wraps the response XML in a SOAP envelope, and returns it to the EIS Adapter Plugin. The EIS Adapter Plugin then strips the SOAP envelope, strips the namespace prefix, if present, and passes the DTD-compliant XML to the agent component of the EIS Adapter Plugin.

Figure 3–2 BSE Architecture as Deployed to Oracle Application Server


Oracle Application Server
Web Container
OracleAS Adapter Application Explorer (Design Time)
OracleAS Adapter Business Services Engine
Adapter Framework
Adapter
PeopleSoft
InterConnect Adapter (Runtime)
SOAP/ HTTP
1
iStudio (Design Time)
Oracle Database
InterConnect Repository
c:\*.dtd
File Repository (Default)
Oracle Database
1 Use either the default file repository or an Oracle database as your repository.

Upon installation of the Oracle Web Services Adapter, an `adapter.ini` file is created. The file consists of all the initialization parameters that the adapter reads at startup. Some of the parameters in this file are configurable.

See Also: *Oracle Application Server Adapters Installation Guide*

4 Examples

This chapter contains the following examples:

- Creating Events
- PeopleSoft Event Integration
- PeopleSoft Service Integration

The event and service scenarios shown depend on the following prerequisites and configuration steps:

Prerequisites

- OracleAS Adapter for PeopleSoft installed on the Oracle Application Server.
- OracleAS Database adapter deployed and configured.
- OracleAS Integration InterConnect Adapter Plugin for EIS installed and running.

> **See Also:** *Oracle Application Server Adapter Installation Guide*

Configuration Steps

The examples present all the configuration steps necessary for demonstrating service and event integration with PeopleSoft. The following cross references are given to identify where more information can be obtained.

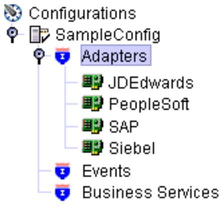
1. Configure the OracleAS Adapter for PeopleSoft for services and events. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for more information.
2. Configure OracleAS Integration InterConnect iStudio for service and event interactions. For more information, see the following service and event steps.

Creating Events

The following example describes how to create events for LOCATION_SYNC.VERSION_1 and how to create a channel and a port.

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 box opens.

   a. In the **Name** field, type a descriptive name, for example, PsoftTarget.

   b. In the **Description** field, type a description for the target (optional).

   c. From the **Target Type** list, select a target type.

      The default value is Application Server.

3. Click **OK.**

   The Application Server dialog box opens where you must specify connection information for PeopleSoft and the application server that is hosting PeopleSoft.

4. Verify your connection parameters. If required, provide the password.

5. Right-click the target name and select **Connect.**

   The x icon disappears, indicating that the node is connected.

1. Open Application Explorer.

2. Expand the **Adapters** node.

   a. Expand the **PeopleSoft** node.

   b. Click the target name, for example, PSoftTarget, under the **PeopleSoft** node.

   The Connection dialog box opens, populated with values you entered.

3. Verify your connection parameters. If required, provide the password.

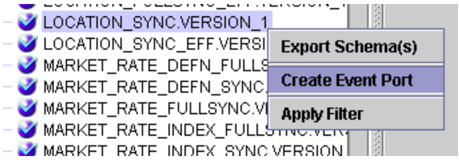
4. Right-click the target name and select **Connect.**

   The x icon disappears, indicating that the node is connected.

5. From the expanded Adapter list, expand the **PeopleSoft** node, expand **Messages**, and then select **LOCATION_SYNC.VERSION_1.**

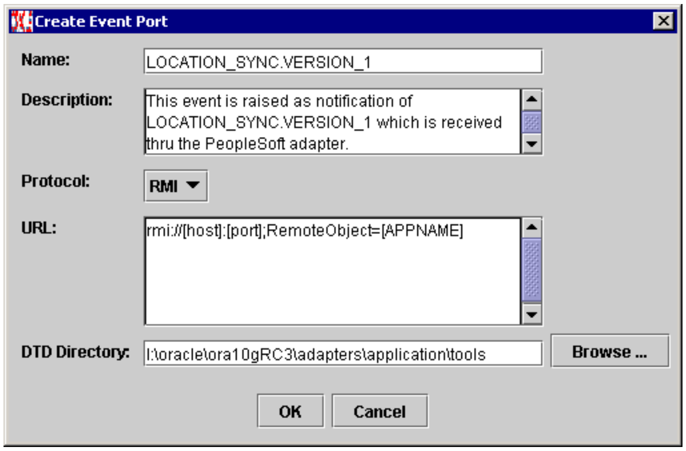
Creating Events for LOCATION.SYNC.VERSION1

To create events for LOCATION_SYNC.VERSION_1:



1. Right-click the **LOCATION_SYNC.VERSION_1** node and select **Create Event Port.**

   The Create Event Port dialog box opens.



Name: LOCATION_SYNC.VERSION_1

Description: This event is raised as notification of LOCATION_SYNC.VERSION_1 which is received thru the PeopleSoft adapter.

Protocol: RMI

URL: rmi://[host]:[port];RemoteObject=[APPNAME]

DTD Directory: I:\oracle\ora10gRC3\adapters\application\tools

a. Type a name for the event port and provide a brief description.

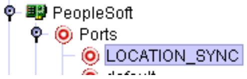
b. From the list, select the required disposition, for example, RMI.

c. Type the disposition URL.

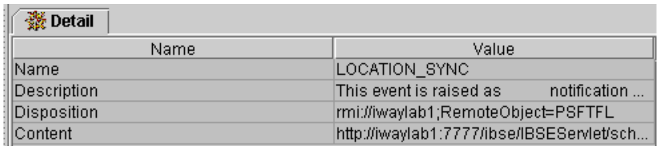
d. Type (or browse to) the path containing the DTD directory.

2. Click **OK.**

The port appears under the ports node in the left pane.



In the right pane, a table appears that summarizes the information associated with the event port you created.



Name	Value
Name	LOCATION_SYNC
Description	This event is raised as notification ...
Disposition	rmi://iwaylab1;RemoteObject=PSFTFL
Content	http://iwaylab1:7777/ibse/IBSEServlet/sch...

You can now associate the event port with a channel.

Creating a Channel

To create a channel:

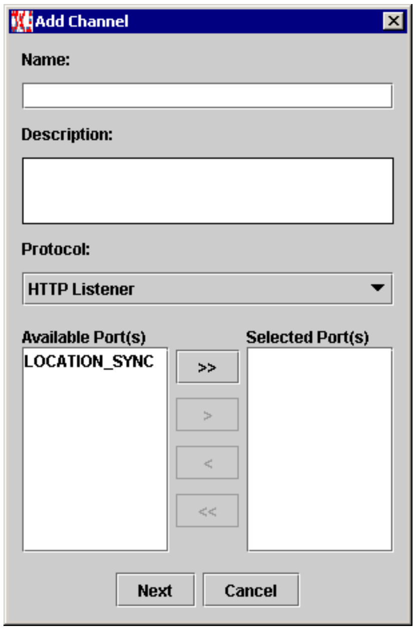
1. In the left pane, click the **Events** node.

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 box opens.



a. Type a name for the channel, for example, `TEST_CHANNEL`.

b. Type a brief description.

c. From the **Protocol** list, select **HTTP Listener.**

d. Select an event port from the list of available ports. To select more than one, hold down the **Ctrl** key and click the ports.

e. To transfer the ports to the list of selected ports, click the **double right (>>)** arrow button.

4. Click **Next.**

5. When the dialog box opens, type the system information as specified in the following table:

Parameter	Description
Port	Port on which to listen for PeopleSoft event data.
Server port	Port on which the host database is listening.
Synchronization Type	Choose from three synchronization options: REQUEST REQUEST_RESPONSE REQUEST_ACK

6. 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.

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

   The channel you created becomes active.

   The X that was over the icon in the left pane disappears.

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

PeopleSoft Event Integration

This topic illustrates PeopleSoft event integration. The procedures describe design time and runtime.

OracleAS Integration InterConnect Design Time

The following procedures describe how to start the repository, create a common view, and then define invoked and implemented procedures. Then, it describes how to export PL/SQL code from iStudio.

Starting the Repository

To start the repository, double-click the `start.bat` file located in the following directory:

```
OracleAS_home\ora92InterCon\oai\9.0.4\repository\start.bat
```

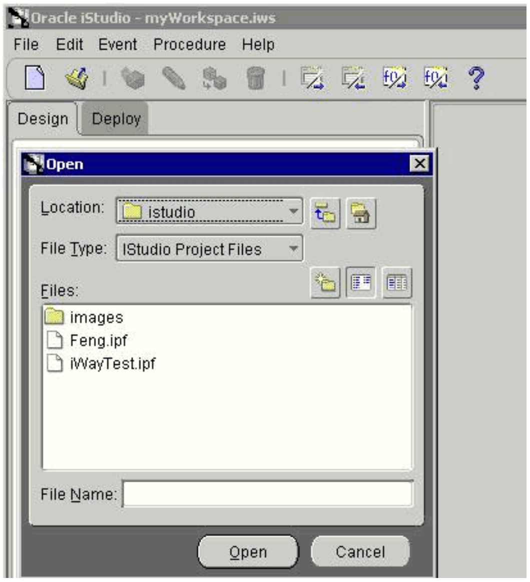
Creating a Common View

To create a Common View:

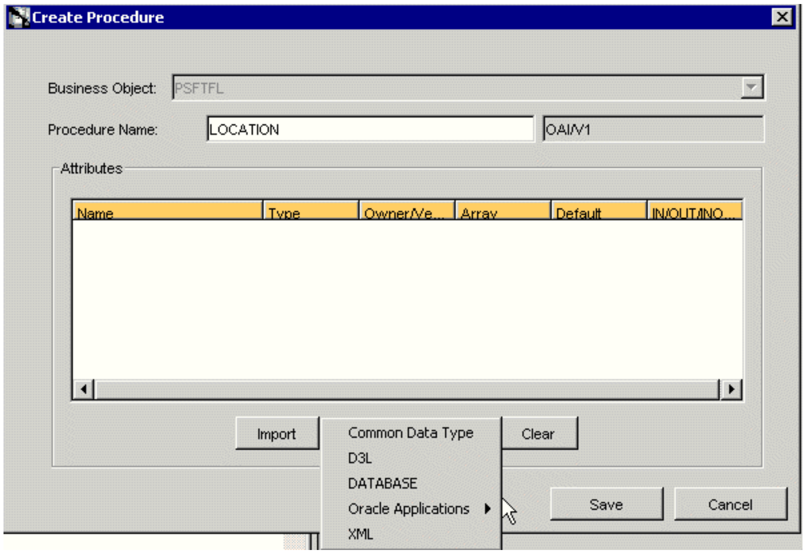
1. Start Oracle iStudio by double-clicking the `start.bat` file located in the following directory:

   ```
   OracleAS_home\ora92iStudio\oai\9.0.4\istudio\iStudio.bat
   ```

   iStudio opens.



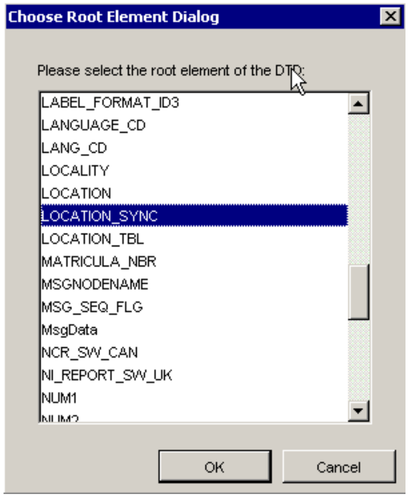
2. Open a project.
3. Open **Common Views** and **Business Objects.**
4. Create a Business Object called `PSFTFL`.



5. Create a new procedure under **PSFTFL** and type `LOCATION` as the procedure name.

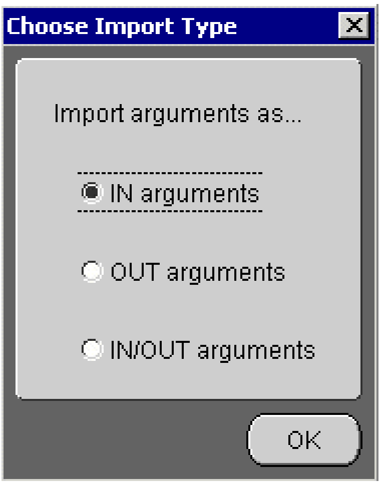
6. Open the DTD generated from Application Explorer and load it.

   The Choose Root Element dialog box opens.



7. Select the root element, `PS8`, for this example.

8. Click **OK.**



Choose Import Type

Import arguments as...

● IN arguments

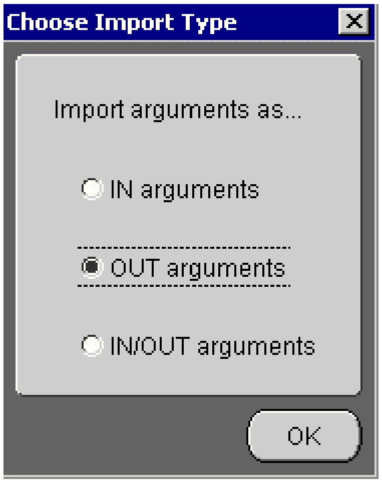
○ OUT arguments

○ IN/OUT arguments

9. Select **IN arguments** as the import type for the request DTD and click **OK.**

10. Import the response DTD, select the root element, and click **OK.**

    The Choose Import Type dialog box opens.



Choose Import Type

Import arguments as...

○ IN arguments

● OUT arguments

○ IN/OUT arguments

11. Select **OUT arguments** as the import type for the response DTD and click **OK.**

12. To save the new procedure, click **Save.**

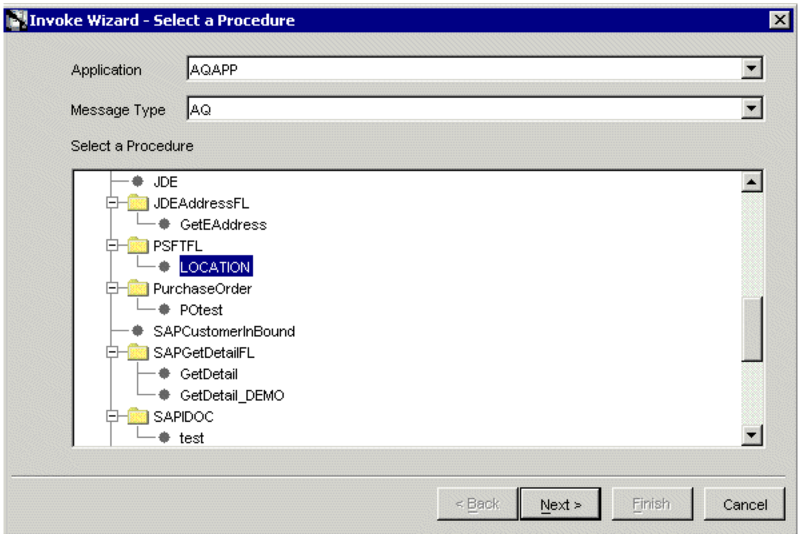
Creating an Invoked Procedure

To create an invoked procedure:

1. Create a new application called **AQAPP.**

2. Right-click **Invoked Procedures** and select **New.**

   The Invoke Wizard - Select a Procedure window opens.

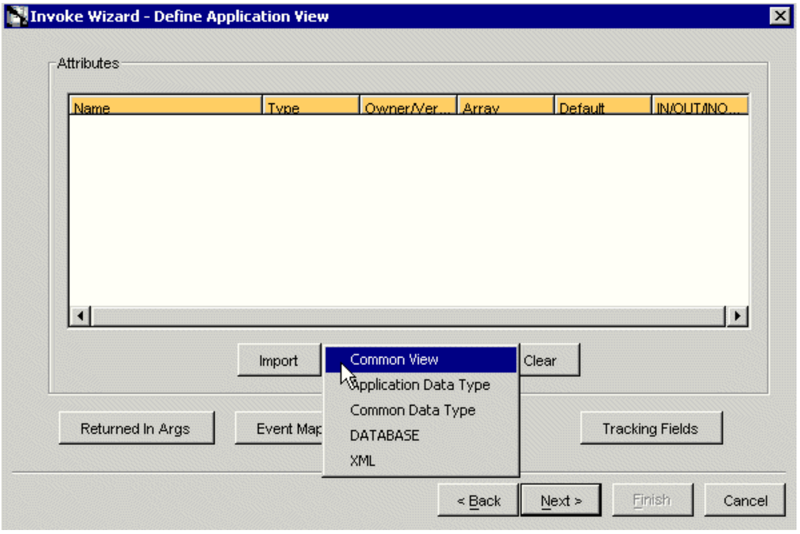


a. From the **Message Type** list, select **AQ.**

b. Expand the `PSFTFL` business object as the event and select **LOCATION.**

3. Click **Next.**

   The Invoke Wizard - Define Application View window opens.

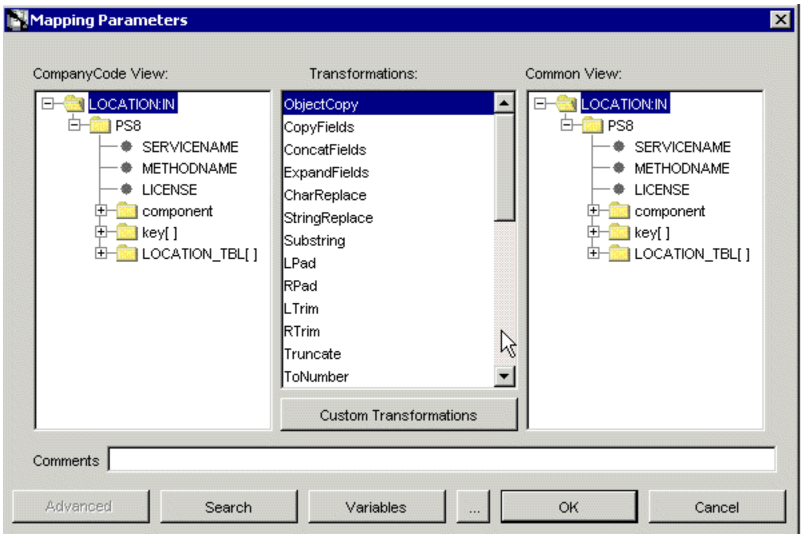


a. Click **Import.**

b. Select **Common View.**

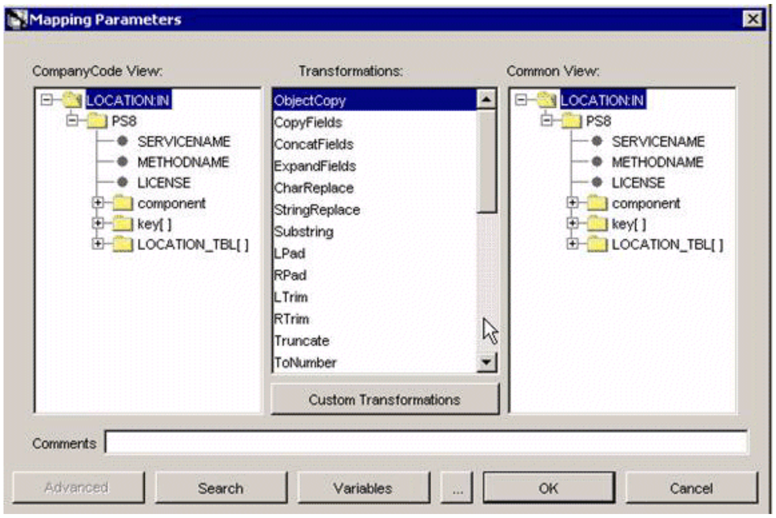
4. Click **Next.**

5. Click **New** to create a mapping between the Common View and the Application View for the IN parameters.



In this example, the Application View and the Common View have the same structure. All the attributes can be mapped by using ObjectCopy Transformation.

6. Click **Apply** and then **OK.**

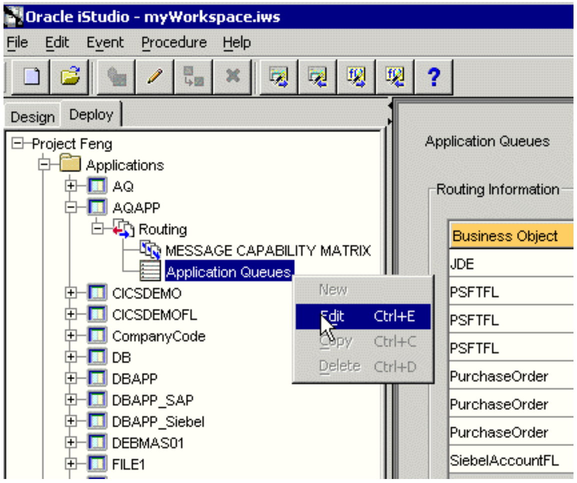


7. Click **Finish.**

You have completed defining the application definition for the invoked procedure.

Defining Queues for AQ

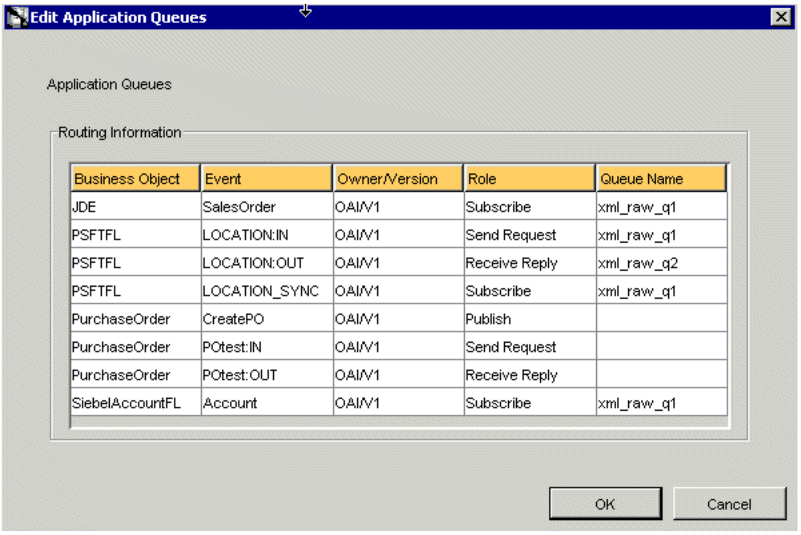
You must create and specify a queue for Advance Queuing (AQ):



Business Object
JDE
PSFTFL
PSFTFL
PSFTFL
PurchaseOrder
PurchaseOrder
PurchaseOrder
SiebelAccountFL

1. To specify the queue in iStudio, select the **Deploy** tab, **Project, Application, AQAPP, Routing,** and **Application Queues.**

2. Right-click **Application Queues** and select **Edit.**



Business Object	Event	Owner/Version	Role	Queue Name
JDE	SalesOrder	OAI/V1	Subscribe	xml_raw_q1
PSFTFL	LOCATION:IN	OAI/V1	Send Request	xml_raw_q1
PSFTFL	LOCATION:OUT	OAI/V1	Receive Reply	xml_raw_q2
PSFTFL	LOCATION_SYNC	OAI/V1	Subscribe	xml_raw_q1
PurchaseOrder	CreatePO	OAI/V1	Publish	
PurchaseOrder	POtest:IN	OAI/V1	Send Request	
PurchaseOrder	POtest:OUT	OAI/V1	Receive Reply	
SiebelAccountFL	Account	OAI/V1	Subscribe	xml_raw_q1

3. In the **Queue Name** column, type the queue name, in this example, **xml_raw_ql** for `PSFTFL LOCATION:IN` event, and **xml_raw_q2** for `PSFTFL LOCATION:OUT` event.

4. To finish, click **OK.**

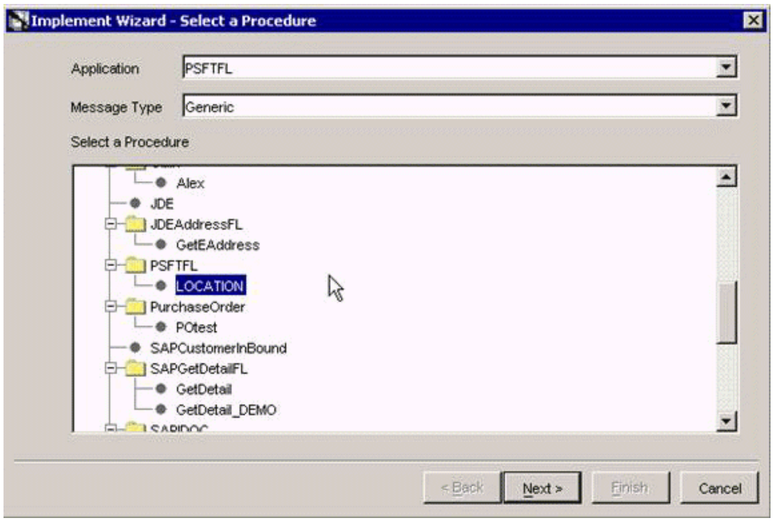
5. To create xml_raw_ql and xml_raw_q2 queues, execute the following SQL under the appropriate schema, in this example, the AQAPP schema.

```
EXECUTE dbms_aqadm.create_queue_table (queue_table => 'RawMsgs_qtab', queue_
payload_type => 'RAW', multiple_consumers => FALSE);
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q1', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q1');
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q2', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q2');
```

Implementing the Procedure

To implement the procedure:

1. Create a new application called `PSFTFL`, expand the application, right-click **Implemented Procedure**, and select **New.**



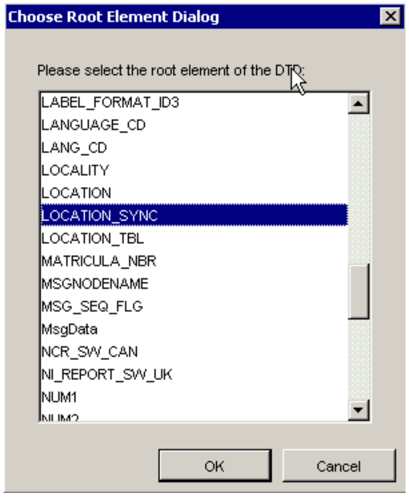
Implement Wizard - Select a Procedure

Application: PSFTFL

Message Type: Generic

2. From the **Message Type** list, select **Generic.**
3. Select **LOCATION** under **PSFTFL** as the procedure.
4. Click **Next.**
5. Open the DTD generated from Application Explorer and load it.

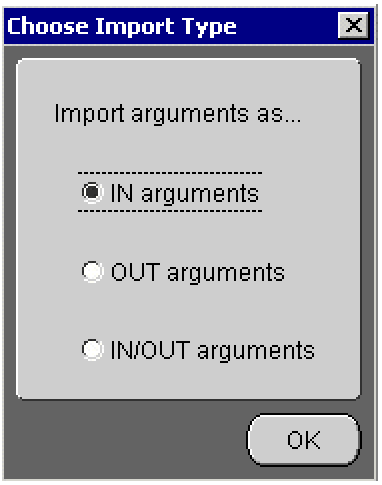
   The Choose Root Element dialog box opens.



Choose Root Element Dialog

Please select the root element of the DTD:

6. Select the root element, `PS8`, for this example.
7. Click **OK.**



8. Select **IN arguments** as the import type for the request DTD and click **OK.**
9. In the **Name** field, type the root element of the request DTD if it is not populated automatically after importing the request DTD.

Editing the adapter.ini File

To edit the adapter.ini file:

1. Open the `adapter.ini` file.
2. Add the following two lines to `adapter.ini` for the adapter:

```
// Bridge class
bridge_class=com.iwaysoftware.iwbridge.IWBridge
// IBSE URL
ibse_url=http://lee.ibi.com:7777/ibse/IBSEServlet/XDSOAPRouter
```

lee.ibi.com
Is the URL of the server.

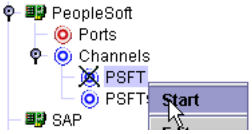
7777
Is the port number.

Runtime

1. Start Oracle Application Server or make sure it is running. Restart OC4J if necessary by executing the following command:

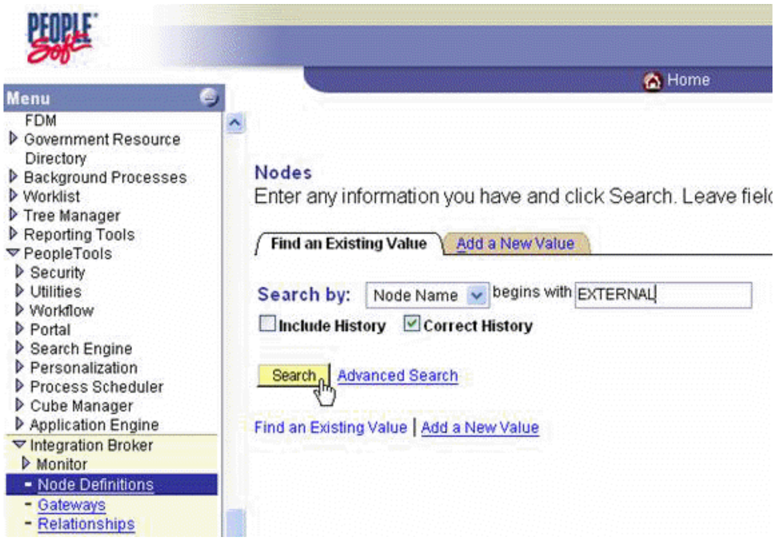
```
\OracleAS_HOME\opmn\bin\opmnctl  stopproc process-type=home
\OracleAS_HOME\opmn\bin\opmnctl  startproc process-type=home
```

2. Start the channel in Application Explorer:



3. Start both publishing adapter and subscribing adapter by executing the following commands:

```
I:\oracle\ora10gRC3\integration\interconnect\adapters\AQAPP\start.bat
I:\oracle\ora10gRC3\integration\interconnect\adapters\PSFTFL\start.bat
```



Nodes

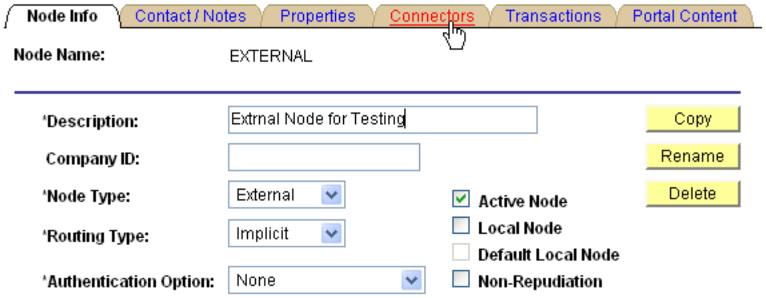
Enter any information you have and click Search. Leave fiel

Search by: Node Name begins with EXTERNAL

☐ Include History ☑ Correct History

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

5. Type the node name you will be using, for example, `EXTERNAL`; then click **Search.**



Node Name: EXTERNAL

*Description: Extrnal Node for Testing

Company ID:

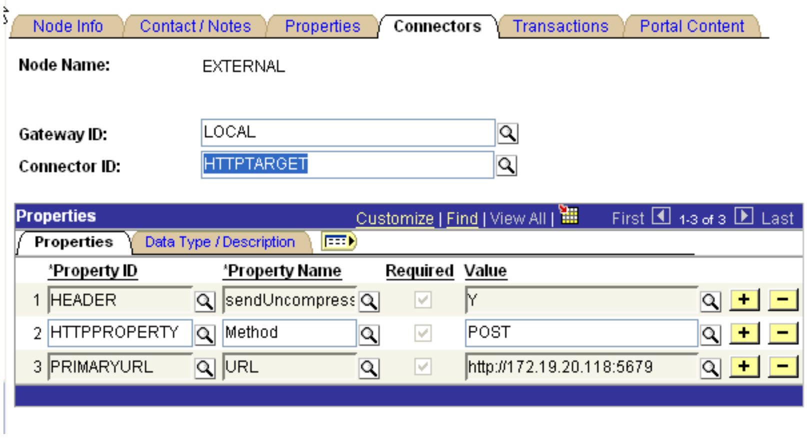
*Node Type: External

*Routing Type: Implicit

*Authentication Option: None

☑ Active Node
☐ Local Node
☐ Default Local Node
☐ Non-Repudiation

6. Click the **Connectors** tab.



Node Name: EXTERNAL

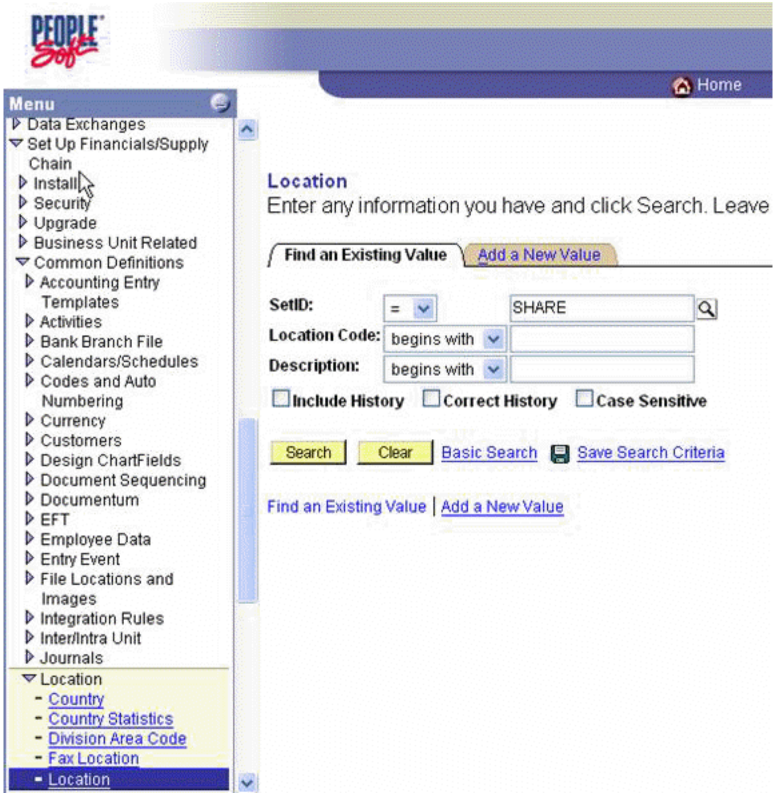
Gateway ID: LOCAL

Connector ID: HTTPTARGET

	*Property ID	*Property Name	Required	Value
1	HEADER	sendUncompress	☑	Y
2	HTTPPROPERTY	Method	☑	POST
3	PRIMARYURL	URL	☑	http://172.19.20.118:5679

7. Enter **HTTPTARGET** in the **Connector ID** field; then enter the **iBSE listener URL** and its port in the **PRIMARYURL** field.

8. To save the configuration, click **Save.**



Location

Enter any information you have and click Search. Leave

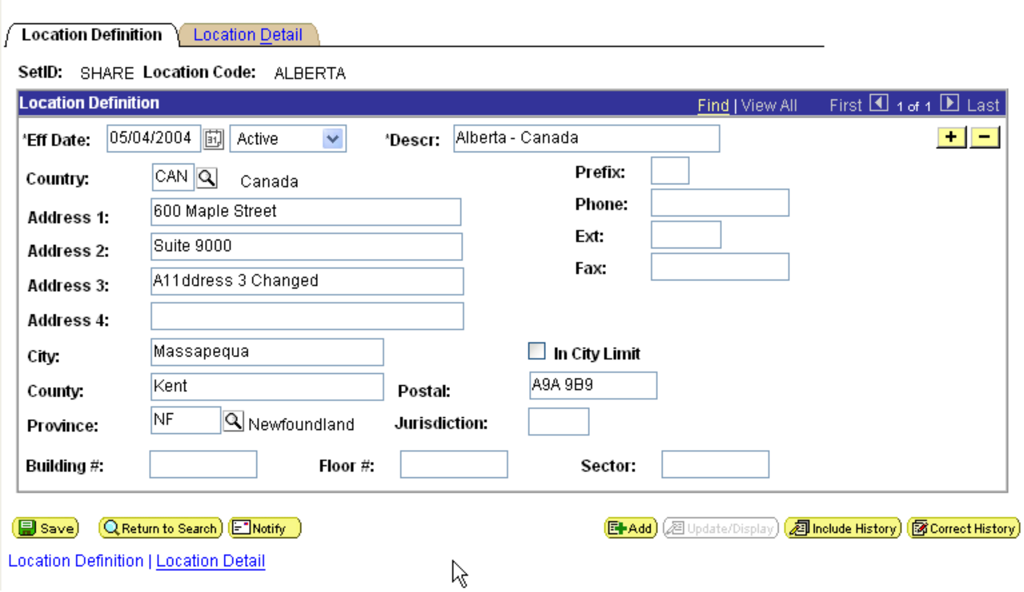
SetID: = SHARE

Location Code: begins with

Description: begins with

□ Include History □ Correct History □ Case Sensitive

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



SetID: SHARE Location Code: ALBERTA

Location Definition

*Eff Date: 05/04/2004 Active *Descr: Alberta - Canada

Country: CAN Canada

Address 1: 600 Maple Street

Address 2: Suite 9000

Address 3: A11ddress 3 Changed

Address 4:

City: Massapequa

County: Kent

Province: NF Newfoundland

Building #: Floor #: Sector:

Prefix:

Phone:

Ext:

Fax:

□ In City Limit

Postal: A9A 9B9

Jurisdiction:

10. 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 BSE listener.

Verifying Results

The following code is an example of the PSFTFL log file:

```
Tue Nov 09 20:45:18 GMT-05:00 2004: The message was sent to topic(s) {oai_hub_
queue=[AQAPP]}. Processing Time = 10,203 ms.
<?xml version = '1.0' encoding = 'UTF-8'?>
<!DOCTYPE MSG>
<MSG>
  <H>
     <BO>PSFTFL</BO>
     <EN>LOCATION_SYNC</EN>
     <EV>OAI/V1</EV>
     <MV>OAI/V1</MV>
     <T>0</T>
     <SN>PSFTFL</SN>
     <SA>PSFTFL</SA>
     <SAID>41</SAID>
     <CI>PSFTFL1100051108265</CI>
  </H>
  <B>
    <AO N = "LOCATION_SYNC_CO">
      <AO N = "LOCATION_SYNC">
        <AO N = "FieldTypes">
          <AO N = "LOCATION_TBL">
            <A N = "class">R</A>
            <AO N = "SETID">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "LOCATION">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "EFFDT">
              <A N = "type">DATE</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "EFF_STATUS">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "DESCR">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "DESCR_AC">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "DESCRSHORT">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "BUILDING">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
            </AO>
            <AO N = "FLOOR">
              <A N = "type">CHAR</A>
              <A N = "PCDATA"></A>
```

```
</AO>
<AO N = "SECTOR">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "JURISDICTION">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ATTN_TO">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "COUNTRY">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS1">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS2">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS3">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDRESS4">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "CITY">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NUM1">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NUM2">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "HOUSE_TYPE">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD1">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD2">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD3">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
```

```
           </AO>
           <AO N = "COUNTY">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "STATE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "POSTAL">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "GEO_CODE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "IN_CITY_LIMIT">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "COUNTRY_CODE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "PHONE">
Transform: CopyFields
            <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "EXTENSION">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "FAX">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "SETID_SALARY">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "SAL_ADMIN_PLAN">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "LANG_CD">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "HOLIDAY_SCHEDULE">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "ESTABID">
             <A N = "type">CHAR</A>
             <A N = "PCDATA"></A>
           </AO>
           <AO N = "LOCALITY">
             <A N = "type">CHAR</A>
```

```
  <A N = "PCDATA"></A>
</AO>
<AO N = "CAN_CMA">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "CAN_OEE_AREACD">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "GEOLOC_CODE">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "OFFICE_TYPE">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NCR_SW_CAN">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "TBS_OFFICE_CD_CAN">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "WRKS_CNCL_ID_LCL">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "SPK_COMM_ID_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_AREA_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "INDUST_INSP_ID_GER">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "NI_REPORT_SW_UK">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_GEOLOC_CD">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_DESIG_AGENT">
  <A N = "type">CHAR</A>
  <A N = "PCDATA"></A>
</AO>
<AO N = "SOC_SEC_WRK_CTR">
  <A N = "type">CHAR</A>
```

```
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "MATRICULA_NBR">
      <A N = "type">NUMBER</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "LABEL_FORMAT_ID2">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "LABEL_FORMAT_ID3">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "USG_LBL_FORMAT_ID">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "COMMENTS_2000">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "REG_REGION">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
  </AO>
  <AO N = "PSCAMA">
    <A N = "class">R</A>
    <AO N = "LANGUAGE_CD">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "AUDIT_ACTN">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "BASE_LANGUAGE_CD">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "MSG_SEQ_FLG">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "PROCESS_INSTANCE">
      <A N = "type">NUMBER</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "PUBLISH_RULE_ID">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "MSGNODENAME">
      <A N = "type">CHAR</A>
      <A N = "PCDATA"></A>
    </AO>
  </AO>
</AO>
<AO N = "MsgData">
```

```
<AO N = "Transaction">
  <AO N = "LOCATION_TBL">
    <A N = "class">R</A>
    <AO N = "SETID">
      <A N = "PCDATA">SHARE</A>
    </AO>
    <AO N = "LOCATION">
      <A N = "PCDATA">ACT001x</A>
    </AO>
    <AO N = "EFFDT">
      <A N = "PCDATA">2002-07-26</A>
    </AO>
    <AO N = "EFF_STATUS">
      <A N = "PCDATA">A</A>
    </AO>
    <AO N = "DESCR">
      <A N = "PCDATA">Testing PSoft Runtime - 2</A>
    </AO>
    <AO N = "DESCR_AC">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "DESCRSHORT">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "BUILDING">
      <A N = "PCDATA">2</A>
    </AO>
    <AO N = "FLOOR">
      <A N = "PCDATA">29</A>
    </AO>
    <AO N = "SECTOR">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "JURISDICTION">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "ATTN_TO">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "COUNTRY">
      <A N = "PCDATA">USA</A>
    </AO>
    <AO N = "ADDRESS1">
      <A N = "PCDATA">2 Pennn Plazaa</A>
    </AO>
    <AO N = "ADDRESS2">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "ADDRESS3">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "ADDRESS4">
      <A N = "PCDATA"></A>
    </AO>
    <AO N = "CITY">
      <A N = "PCDATA">New York</A>
    </AO>
    <AO N = "NUM1">
      <A N = "PCDATA"></A>
    </AO>
```

```
<AO N = "NUM2">
  <A N = "PCDATA"></A>
</AO>
<AO N = "HOUSE_TYPE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD1">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD2">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ADDR_FIELD3">
  <A N = "PCDATA"></A>
</AO>
<AO N = "COUNTY">
  <A N = "PCDATA"></A>
</AO>
<AO N = "STATE">
  <A N = "PCDATA">NY</A>
</AO>
<AO N = "POSTAL">
  <A N = "PCDATA">10121</A>
</AO>
<AO N = "GEO_CODE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "IN_CITY_LIMIT">
  <A N = "PCDATA"></A>
</AO>
<AO N = "COUNTRY_CODE">
  <A N = "PCDATA">212</A>
</AO>
<AO N = "PHONE">
  <A N = "PCDATA">736-4433</A>
</AO>
<AO N = "EXTENSION">
  <A N = "PCDATA">3694</A>
</AO>
<AO N = "FAX">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SETID_SALARY">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SAL_ADMIN_PLAN">
  <A N = "PCDATA"></A>
</AO>
<AO N = "LANG_CD">
  <A N = "PCDATA"></A>
</AO>
<AO N = "HOLIDAY_SCHEDULE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "ESTABID">
  <A N = "PCDATA"></A>
</AO>
<AO N = "LOCALITY">
  <A N = "PCDATA"></A>
</AO>
```

```
<AO N = "CAN_CMA">
  <A N = "PCDATA"></A>
</AO>
<AO N = "CAN_OEE_AREACD">
  <A N = "PCDATA"></A>
</AO>
<AO N = "GEOLOC_CODE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "OFFICE_TYPE">
  <A N = "PCDATA"></A>
</AO>
<AO N = "NCR_SW_CAN">
  <A N = "PCDATA"></A>
</AO>
<AO N = "TBS_OFFICE_CD_CAN">
  <A N = "PCDATA"></A>
</AO>
<AO N = "WRKS_CNCL_ID_LCL">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SPK_COMM_ID_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_AREA_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "TARIFF_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "INDUST_INSP_ID_GER">
  <A N = "PCDATA"></A>
</AO>
<AO N = "NI_REPORT_SW_UK">
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_GEOLOC_CD">
  <A N = "PCDATA"></A>
</AO>
<AO N = "GVT_DESIG_AGENT">
  <A N = "PCDATA"></A>
</AO>
<AO N = "SOC_SEC_WRK_CTR">
  <A N = "PCDATA"></A>
</AO>
<AO N = "MATRICULA_NBR">
  <A N = "PCDATA">0</A>
</AO>
<AO N = "LABEL_FORMAT_ID2">
  <A N = "PCDATA"></A>
</AO>
<AO N = "LABEL_FORMAT_ID3">
  <A N = "PCDATA"></A>
</AO>
<AO N = "USG_LBL_FORMAT_ID">
  <A N = "PCDATA"></A>
</AO>
<AO N = "COMMENTS_2000">
  <A N = "PCDATA"></A>
</AO>
```

```
              <AO N = "REG_REGION">
                <A N = "PCDATA"></A>
              </AO>
            </AO>
          </AO>
        </AO>
      </AO>
    </AO>
  </B>
</MSG>
Tue Nov 09 20:45:18 GMT-05:00 2004: Outbound Transform Engine: beginning to
transform message.
Tue Nov 09 20:45:18 GMT-05:00 2004: Outbound Transform Engine: performing
transform.
```

Subscribing Adapter (AQAPP) Console Log

```
Tue Nov 09 20:45:11 GMT-05:00 2004: AQ Adapter: received the message from the
Agent and will now write it to AQ.
Tue Nov 09 20:45:12 GMT-05:00 2004: AQ Adapter: successfully converted the OAI
message to XML
<?xml version = '1.0' encoding = 'UTF-8' standalone = 'yes'?>
<LOCATION_SYNC>
   <FieldTypes>
      <LOCATION_TBL class="R">
         <SETID type="CHAR"></SETID>
         <LOCATION type="CHAR"></LOCATION>
         <EFFDT type="DATE"></EFFDT>
         <EFF_STATUS type="CHAR"></EFF_STATUS>
         <DESCR type="CHAR"></DESCR>
         <DESCR_AC type="CHAR"></DESCR_AC>
         <DESCRSHORT type="CHAR"></DESCRSHORT>
         <BUILDING type="CHAR"></BUILDING>
         <FLOOR type="CHAR"></FLOOR>
         <SECTOR type="CHAR"></SECTOR>
         <JURISDICTION type="CHAR"></JURISDICTION>
         <ATTN_TO type="CHAR"></ATTN_TO>
         <COUNTRY type="CHAR"></COUNTRY>
         <ADDRESS1 type="CHAR"></ADDRESS1>
         <ADDRESS2 type="CHAR"></ADDRESS2>
         <ADDRESS3 type="CHAR"></ADDRESS3>
         <ADDRESS4 type="CHAR"></ADDRESS4>
         <CITY type="CHAR"></CITY>
         <NUM1 type="CHAR"></NUM1>
         <NUM2 type="CHAR"></NUM2>
         <HOUSE_TYPE type="CHAR"></HOUSE_TYPE>
         <ADDR_FIELD1 type="CHAR"></ADDR_FIELD1>
         <ADDR_FIELD2 type="CHAR"></ADDR_FIELD2>
         <ADDR_FIELD3 type="CHAR"></ADDR_FIELD3>
         <COUNTY type="CHAR"></COUNTY>
         <STATE type="CHAR"></STATE>
         <POSTAL type="CHAR"></POSTAL>
         <GEO_CODE type="CHAR"></GEO_CODE>
         <IN_CITY_LIMIT type="CHAR"></IN_CITY_LIMIT>
         <COUNTRY_CODE type="CHAR"></COUNTRY_CODE>
         <PHONE type="CHAR"></PHONE>
         <EXTENSION type="CHAR"></EXTENSION>
         <FAX type="CHAR"></FAX>
         <SETID_SALARY type="CHAR"></SETID_SALARY>
         <SAL_ADMIN_PLAN type="CHAR"></SAL_ADMIN_PLAN>
         <LANG_CD type="CHAR"></LANG_CD>
      <HOLIDAY_SCHEDULE type="CHAR"></HOLIDAY_SCHEDULE>
      <ESTABID type="CHAR"></ESTABID>
      <LOCALITY type="CHAR"></LOCALITY>
      <CAN_CMA type="CHAR"></CAN_CMA>
      <CAN_OEE_AREACD type="CHAR"></CAN_OEE_AREACD>
      <GEOLOC_CODE type="CHAR"></GEOLOC_CODE>
      <OFFICE_TYPE type="CHAR"></OFFICE_TYPE>
      <NCR_SW_CAN type="CHAR"></NCR_SW_CAN>
      <TBS_OFFICE_CD_CAN type="CHAR"></TBS_OFFICE_CD_CAN>
      <WRKS_CNCL_ID_LCL type="CHAR"></WRKS_CNCL_ID_LCL>
      <SPK_COMM_ID_GER type="CHAR"></SPK_COMM_ID_GER>
      <TARIFF_AREA_GER type="CHAR"></TARIFF_AREA_GER>
      <TARIFF_GER type="CHAR"></TARIFF_GER>
      <INDUST_INSP_ID_GER type="CHAR"></INDUST_INSP_ID_GER>
      <NI_REPORT_SW_UK type="CHAR"></NI_REPORT_SW_UK>
      <GVT_GEOLOC_CD type="CHAR"></GVT_GEOLOC_CD>
      <GVT_DESIG_AGENT type="CHAR"></GVT_DESIG_AGENT>
      <SOC_SEC_WRK_CTR type="CHAR"></SOC_SEC_WRK_CTR>
      <MATRICULA_NBR type="NUMBER"></MATRICULA_NBR>
      <LABEL_FORMAT_ID2 type="CHAR"></LABEL_FORMAT_ID2>
      <LABEL_FORMAT_ID3 type="CHAR"></LABEL_FORMAT_ID3>
      <USG_LBL_FORMAT_ID type="CHAR"></USG_LBL_FORMAT_ID>
      <COMMENTS_2000 type="CHAR"></COMMENTS_2000>
      <REG_REGION type="CHAR"></REG_REGION>
   </LOCATION_TBL>
   <PSCAMA class="R">
      <LANGUAGE_CD type="CHAR"></LANGUAGE_CD>
      <AUDIT_ACTN type="CHAR"></AUDIT_ACTN>
      <BASE_LANGUAGE_CD type="CHAR"></BASE_LANGUAGE_CD>
      <MSG_SEQ_FLG type="CHAR"></MSG_SEQ_FLG>
      <PROCESS_INSTANCE type="NUMBER"></PROCESS_INSTANCE>
      <PUBLISH_RULE_ID type="CHAR"></PUBLISH_RULE_ID>
      <MSGNODENAME type="CHAR"></MSGNODENAME>
   </PSCAMA>
</FieldTypes>
<MsgData>
   <Transaction>
      <LOCATION_TBL class="R">
         <SETID>SHARE</SETID>
         <LOCATION>ACT001x</LOCATION>
         <EFFDT>2002-07-26</EFFDT>
         <EFF_STATUS>A</EFF_STATUS>
         <DESCR>Testing PSoft Runtime - 2</DESCR>
         <DESCR_AC></DESCR_AC>
         <DESCRSHORT></DESCRSHORT>
         <BUILDING>2</BUILDING>
         <FLOOR>29</FLOOR>
         <SECTOR></SECTOR>
         <JURISDICTION></JURISDICTION>
         <ATTN_TO></ATTN_TO>
         <COUNTRY>USA</COUNTRY>
         <ADDRESS1>2 Pennn Plazaa</ADDRESS1>
         <ADDRESS2></ADDRESS2>
         <ADDRESS3></ADDRESS3>
         <ADDRESS4></ADDRESS4>
         <CITY>New York</CITY>
         <NUM1></NUM1>
         <NUM2></NUM2>
         <HOUSE_TYPE></HOUSE_TYPE>
         <ADDR_FIELD1></ADDR_FIELD1>
```

```
            <ADDR_FIELD2></ADDR_FIELD2>
            <ADDR_FIELD3></ADDR_FIELD3>
            <COUNTY></COUNTY>
            <STATE>NY</STATE>
            <POSTAL>10121</POSTAL>
            <GEO_CODE></GEO_CODE>
            <IN_CITY_LIMIT></IN_CITY_LIMIT>
            <COUNTRY_CODE>212</COUNTRY_CODE>
            <PHONE>736-4433</PHONE>
            <EXTENSION>3694</EXTENSION>
            <FAX></FAX>
            <SETID_SALARY></SETID_SALARY>
            <SAL_ADMIN_PLAN></SAL_ADMIN_PLAN>
            <LANG_CD></LANG_CD>
            <HOLIDAY_SCHEDULE></HOLIDAY_SCHEDULE>
            <ESTABID></ESTABID>
            <LOCALITY></LOCALITY>
            <CAN_CMA></CAN_CMA>
            <CAN_OEE_AREACD></CAN_OEE_AREACD>
            <GEOLOC_CODE></GEOLOC_CODE>
            <OFFICE_TYPE></OFFICE_TYPE>
            <NCR_SW_CAN></NCR_SW_CAN>
            <TBS_OFFICE_CD_CAN></TBS_OFFICE_CD_CAN>
            <WRKS_CNCL_ID_LCL></WRKS_CNCL_ID_LCL>
            <SPK_COMM_ID_GER></SPK_COMM_ID_GER>
            <TARIFF_AREA_GER></TARIFF_AREA_GER>
            <TARIFF_GER></TARIFF_GER>
            <INDUST_INSP_ID_GER></INDUST_INSP_ID_GER>
            <NI_REPORT_SW_UK></NI_REPORT_SW_UK>
            <GVT_GEOLOC_CD></GVT_GEOLOC_CD>
            <GVT_DESIG_AGENT></GVT_DESIG_AGENT>
            <SOC_SEC_WRK_CTR></SOC_SEC_WRK_CTR>
            <MATRICULA_NBR>0</MATRICULA_NBR>
            <LABEL_FORMAT_ID2></LABEL_FORMAT_ID2>
            <LABEL_FORMAT_ID3></LABEL_FORMAT_ID3>
            <USG_LBL_FORMAT_ID></USG_LBL_FORMAT_ID>
            <COMMENTS_2000></COMMENTS_2000>
            <REG_REGION></REG_REGION>
         </LOCATION_TBL>
      </Transaction>
   </MsgData>
</LOCATION_SYNC>
```

PeopleSoft Service Integration

This topic illustrates PeopleSoft service integration. The procedures describe design time and runtime.

OracleAS Integration InterConnect Design Time

The following procedures describe how to start the repository and create a common view and then, define invoked and implemented procedures. Then, it describes how to export PL/SQL code from iStudio.

Starting the Repository

To start the repository, double-click the `start.bat` file located in the following directory:

```
OracleAS_home\ora92InterCon\oai\9.0.4\repository\start.bat
```

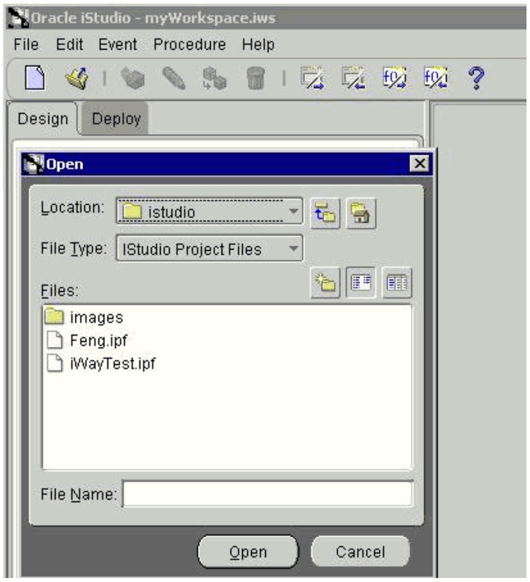
Creating a Common View

To create a Common View:

1. Start Oracle iStudio by double-clicking the `start.bat` file located in the following directory:

   ```
   OracleAS_home\ora92iStudio\oai\9.0.4\istudio\iStudio.bat
   ```

   iStudio opens.



Location: istudio

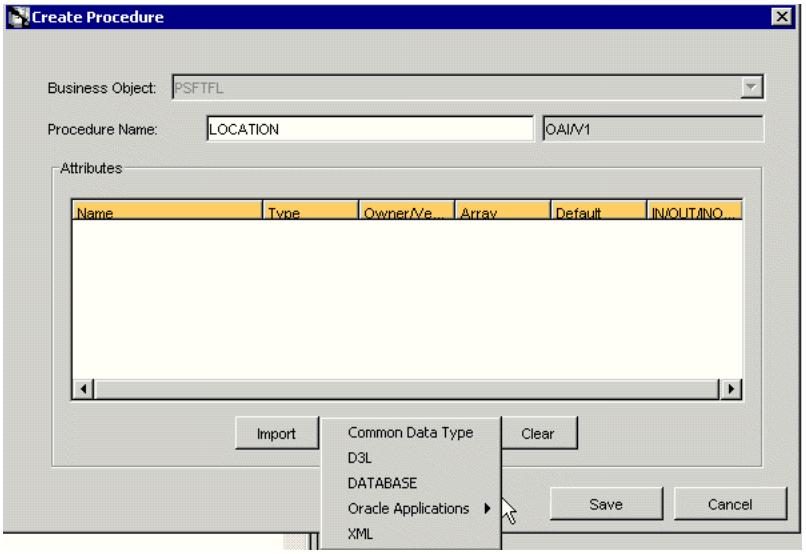
File Type: IStudio Project Files

Files:

images
Feng.ipf
iWayTest.ipf

File Name:

2. Open a project.
3. Open **Common Views** and **Business Objects.**
4. Create a Business Object called **PSFTFL.**



Business Object: PSFTFL

Procedure Name: LOCATION OAI/V1

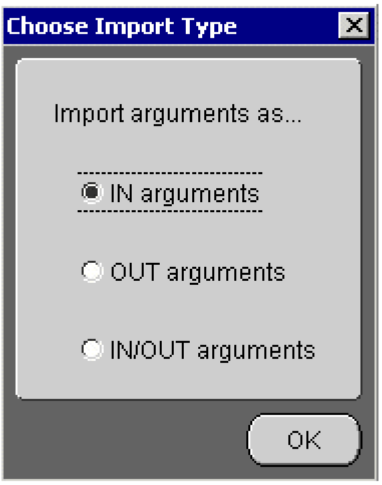
Attributes

Name	Type	Owner/Ve...	Array	Default	IN/OUT/INO...

5. Create a new procedure under `PSFTFL` and type `LOCATION` as the procedure name.
6. Open the DTD generated from Application Explorer and load it.

The Choose Root Element dialog box opens.

7. Select the root element, `PS8`, for this example.
8. Click **OK.**



Choose Import Type

Import arguments as...

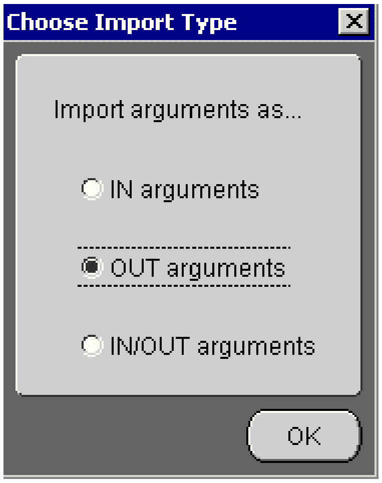
● IN arguments

○ OUT arguments

○ IN/OUT arguments

9. Select **IN arguments** as the import type for the request DTD and click **OK.**
10. Import the response DTD, select the root element, and click **OK.**

    The Choose Import Type dialog box opens.



Choose Import Type

Import arguments as...

○ IN arguments

● OUT arguments

○ IN/OUT arguments

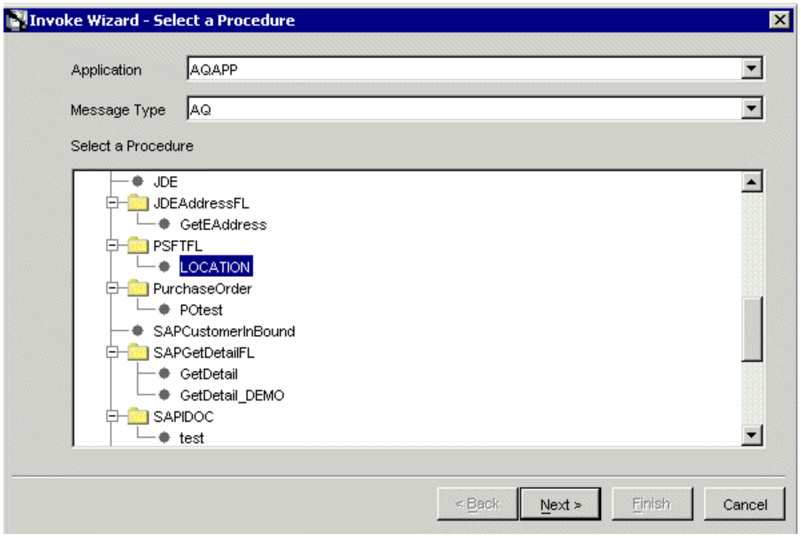
11. Select **OUT arguments** as the import type for the response DTD and click **OK.**
12. To save the new procedure, click **Save.**

Creating an Invoked Procedure

To create an invoked procedure:

1. Create a new application called **AQAPP.**
2. Right-click **Invoked Procedures** and select **New.**

   The Invoke Wizard - Select a Procedure window opens.


Invoke Wizard - Select a Procedure

Application: AQAPP

Message Type: AQ

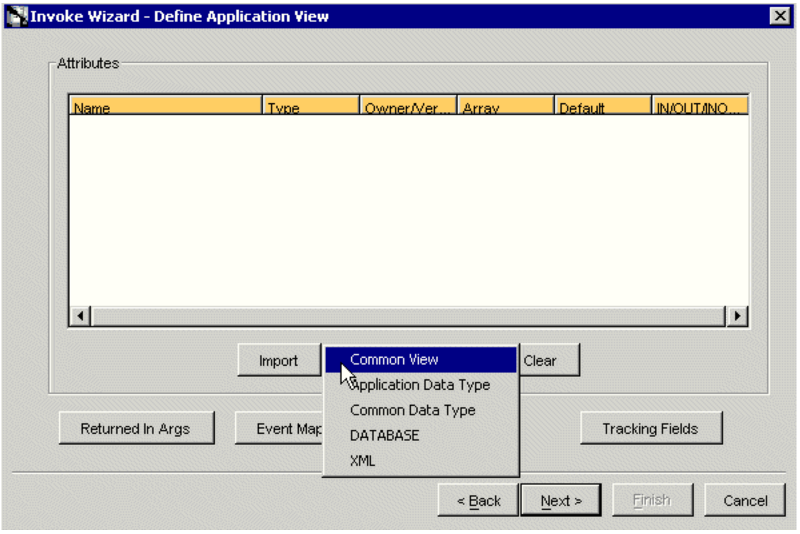
Select a Procedure

a. From the **Message Type** list, select **AQ.**

b. Expand the `PSFTFL` business object as the event and select **LOCATION.**

3. Click **Next.**

The Invoke Wizard - Define Application View window opens.


Invoke Wizard - Define Application View

Attributes

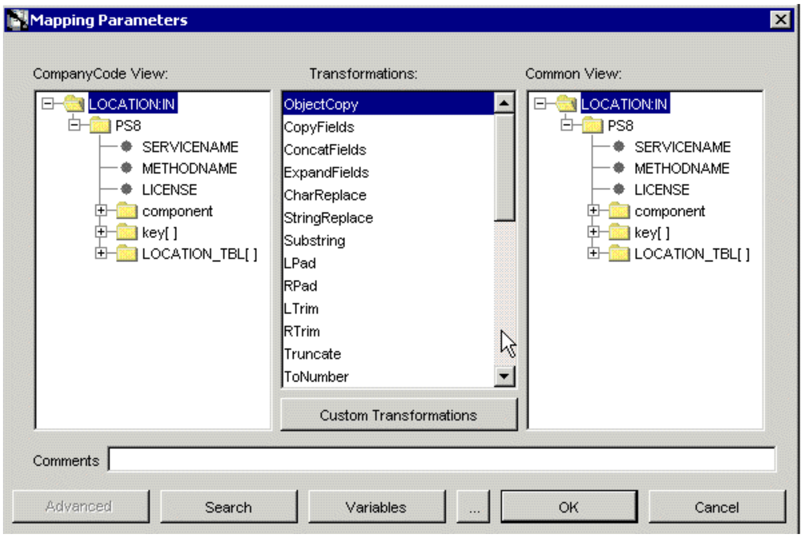
Name	Type	Owner/Ver...	Array	Default	IN/OUT/INO...

a. Click **Import.**

b. Select **Common View.**

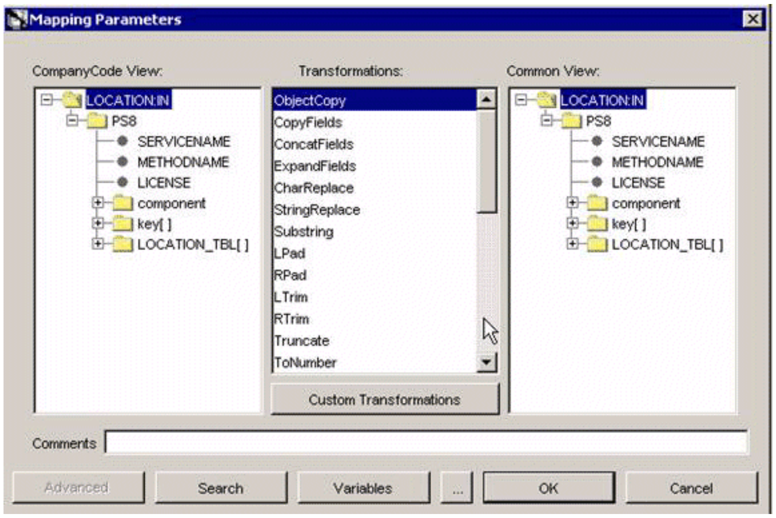
4. Click **Next.**

5. Click **New** to create a mapping between the Common View and the Application View for the IN parameters.



In this example, the Application View and the Common View have the same structure. All the attributes can be mapped by using ObjectCopy Transformation.

6. Click **Apply** and then, **OK.**

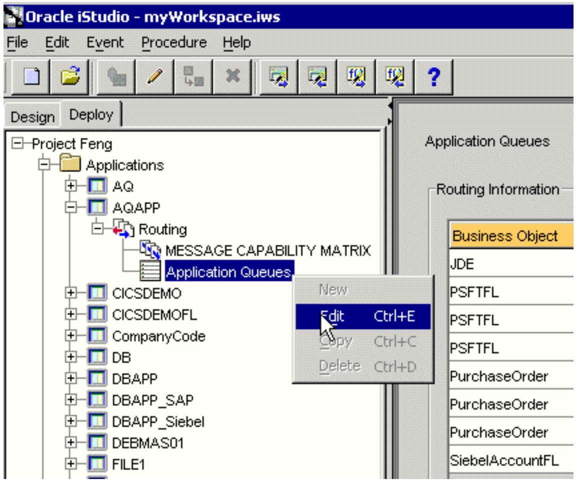


7. Click **Finish.**

   You have completed defining the application definition for the invoked procedure.

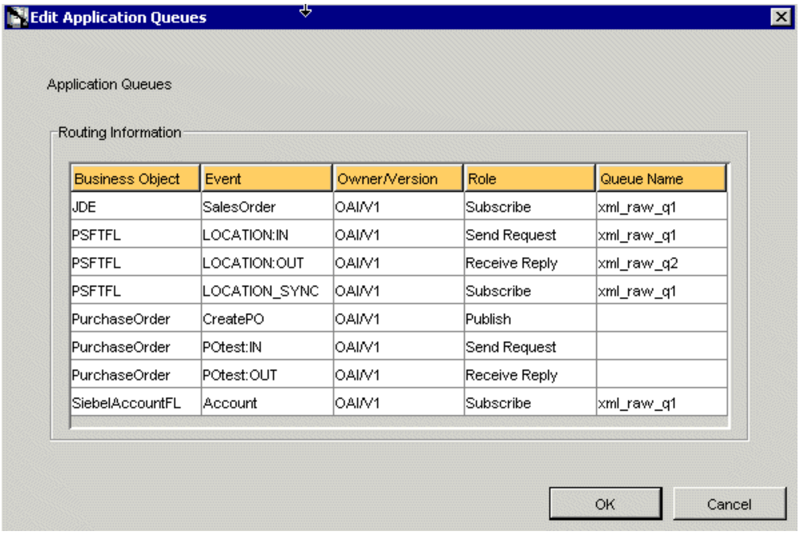
Defining Queues for AQ

You must create and specify a queue for Advance Queuing (AQ):



Business Object
JDE
PSFTFL
PSFTFL
PSFTFL
PurchaseOrder
PurchaseOrder
PurchaseOrder
SiebelAccountFL

1. To specify the queue in iStudio, select the **Deploy** tab, **Project, Application, AQAPP, Routing,** and **Application Queues.**
2. Right-click **Application Queues** and select **Edit.**



Business Object	Event	Owner/Version	Role	Queue Name
JDE	SalesOrder	OAI/V1	Subscribe	xml_raw_q1
PSFTFL	LOCATION:IN	OAI/V1	Send Request	xml_raw_q1
PSFTFL	LOCATION:OUT	OAI/V1	Receive Reply	xml_raw_q2
PSFTFL	LOCATION_SYNC	OAI/V1	Subscribe	xml_raw_q1
PurchaseOrder	CreatePO	OAI/V1	Publish	
PurchaseOrder	POtest:IN	OAI/V1	Send Request	
PurchaseOrder	POtest:OUT	OAI/V1	Receive Reply	
SiebelAccountFL	Account	OAI/V1	Subscribe	xml_raw_q1

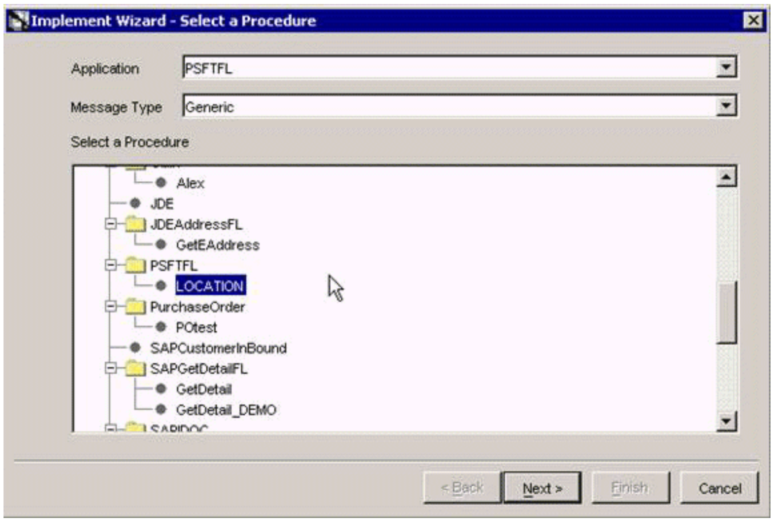
3. In the **Queue Name** column, type the queue name, in this example, **xml_raw_ql** for `PSFTFL LOCATION:IN` event, and **xml_raw_q2** for `PSFTFL LOCATION:OUT` event.
4. To finish, click **OK.**
5. To create xml_raw_ql and xml_raw_q2 queues, execute the following SQL under the appropriate schema, in this example, the AQAPP schema.

```
EXECUTE dbms_aqadm.create_queue_table (queue_table => 'RawMsgs_qtab', queue_
payload_type => 'RAW', multiple_consumers => FALSE);
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q1', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q1');
EXECUTE dbms_aqadm.create_queue (queue_name => 'xml_raw_q2', queue_table =>
'RawMsgs_qtab');
EXECUTE dbms_aqadm.start_queue (queue_name => 'xml_raw_q2');
```

Implementing the Procedure

To implement the procedure:

1. Create a new application called `PSFTFL`, expand the application, right-click **Implemented Procedure**, and select **New.**



Implement Wizard - Select a Procedure

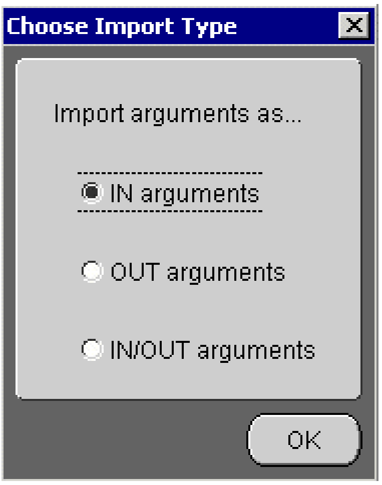
Application: PSFTFL

Message Type: Generic

Select a Procedure

2. From the **Message Type** list, select **Generic.**
3. Select **LOCATION** under **PSFTFL** as the procedure.
4. Click **Next.**
5. Open the DTD generated from Application Explorer and load it.

   The Choose Root Element dialog box opens.
6. Select the root element, `PS8`, for this example.
7. Click **OK.**



Choose Import Type

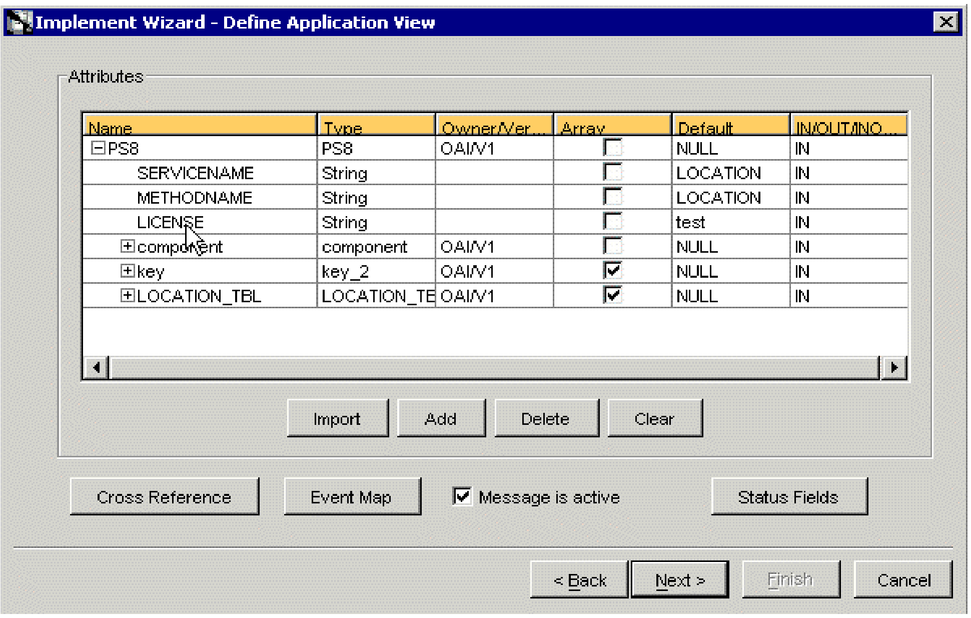
Import arguments as...

● IN arguments

○ OUT arguments

○ IN/OUT arguments

8. Select **IN arguments** as the import type for the request DTD and click **OK.**
9. In the **Name** field, type the root element of the request DTD if it is not populated automatically after importing the request DTD.



Implement Wizard - Define Application View

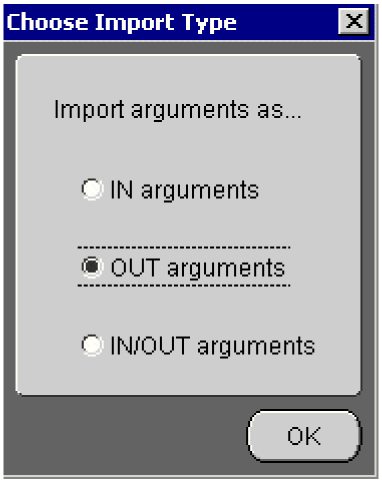
Attributes

Name	Type	Owner/Ver...	Array	Default	IN/OUT/INO...
⊟PS8	PS8	OAI/V1	□	NULL	IN
SERVICENAME	String		□	LOCATION	IN
METHODNAME	String		□	LOCATION	IN
LICENSE	String		□	test	IN
⊞component	component	OAI/V1	□	NULL	IN
⊞key	key_2	OAI/V1	☑	NULL	IN
⊞LOCATION_TBL	LOCATION_TE	OAI/V1	☑	NULL	IN

☑ Message is active

10. Import the response DTD, select `PS8` as the root element, and then click **OK.**

11. Select OUT arguments as the Import Type for the response.



Choose Import Type

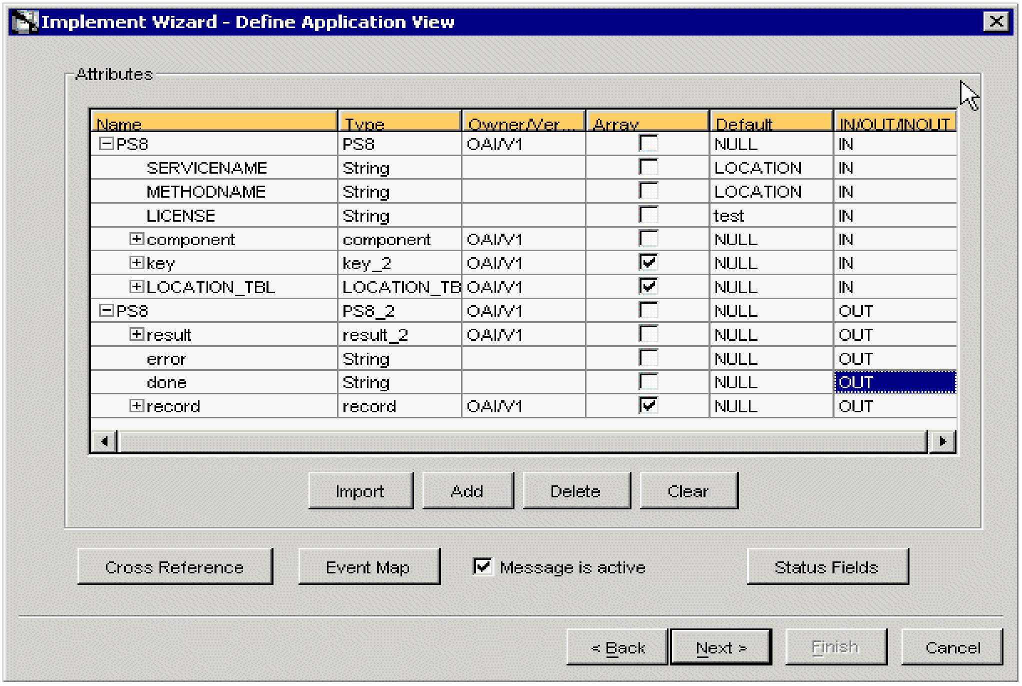
Import arguments as...

○ IN arguments

● OUT arguments

○ IN/OUT arguments

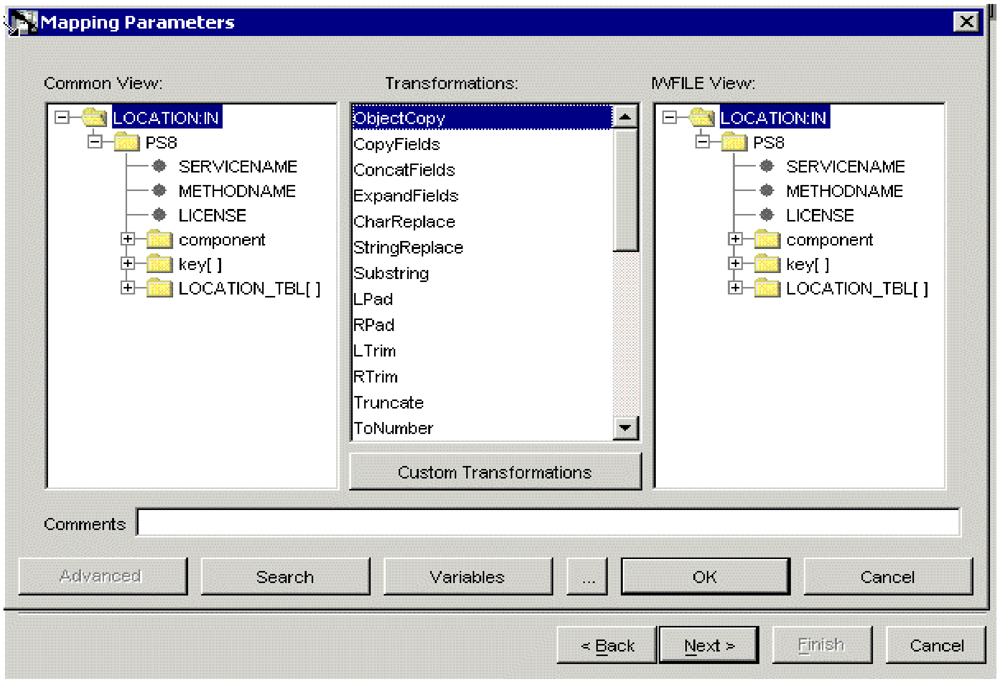
You have now imported both request and response DTDs into iStudio.



Name	Type	Owner/Ver...	Array	Default	IN/OUT/INOUT
⊟PS8	PS8	OAI/V1	☐	NULL	IN
SERVICENAME	String		☐	LOCATION	IN
METHODNAME	String		☐	LOCATION	IN
LICENSE	String		☐	test	IN
⊞component	component	OAI/V1	☐	NULL	IN
⊞key	key_2	OAI/V1	☑	NULL	IN
⊞LOCATION_TBL	LOCATION_TB	OAI/V1	☑	NULL	IN
⊟PS8	PS8_2	OAI/V1	☐	NULL	OUT
⊞result	result_2	OAI/V1	☐	NULL	OUT
error	String		☐	NULL	OUT
done	String		☐	NULL	OUT
⊞record	record	OAI/V1	☑	NULL	OUT

☑ Message is active

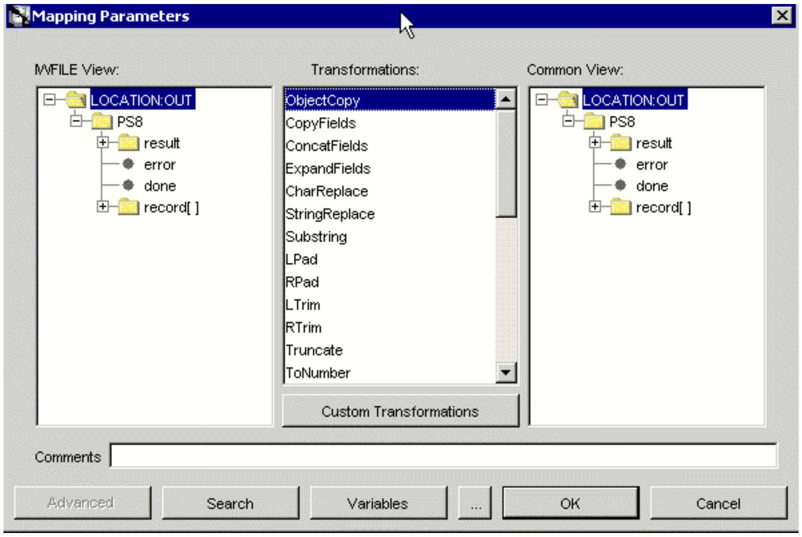
12. To define a mapping between the Application View and the Common View, click **Next** and then **New.**



Because the structures are the same, use ObjectCopy transformation for the mapping.

13. Click **Apply** and then **OK.**

14. Click **Next** to define the mapping for OUT arguments, using the ObjectCopy transformation.



15. To complete the definition of the implemented procedure, click **Next** and then **Finish.**

Editing the adapter.ini File

To edit the adapter.ini file:

1. Open the `adapter.ini` file.
2. Add the following two lines to `adapter.ini` for the adapter:

```
// Bridge class
bridge_class=com.iwaysoftware.iwbridge.IWBridge

// IBSE URL
ibse_url=http://lee.ibi.com:7777/ibse/IBSEServlet/XDSOAPRouter
```

lee.ibi.com
Is the URL of the server.

7777
Is the port number.

OracleAS Integration InterConnect Runtime

The following topic describes how to verify service integration using the OracleAS Adapter for PeopleSoft.

Verifying Service Integration

To verify service integration:

1. Start the Oracle Application Server or ensure that the server is running.
2. Restart OC4J, if required, by executing the following command:

```
\OracleAS_home\opmn\bin\opmnctl  stopproc process-type=home
\OracleAS_home\opmn\bin\opmnctl  startproc process-type=home
```

3. Check the status of OC4J by executing the following command:

```
\OracleAS_home\opmn\bin\opmnctl  status
```

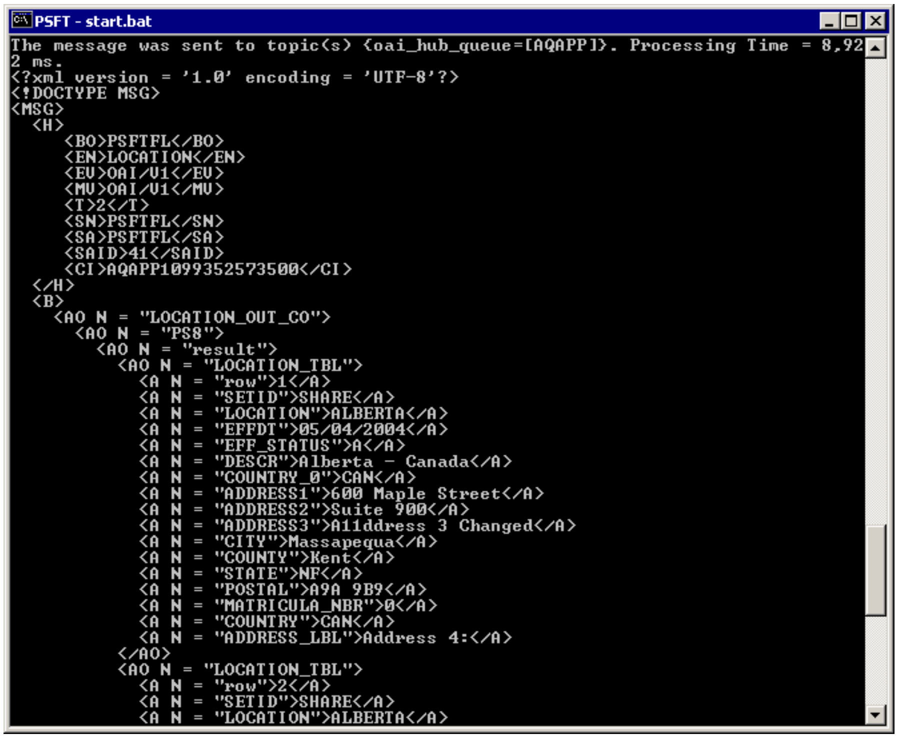
4. Invoke and implement the adapter by executing the following commands:

```
\InterConnect_HOME\oai\9.0.4\adapters\PSFTFL\start.bat
\InterConnect_HOME\oai\9.0.4\adapters\AQAPP\start.bat
```

5. Log on to SQL*Plus with `AQAPP`, in this example, and execute the following command:

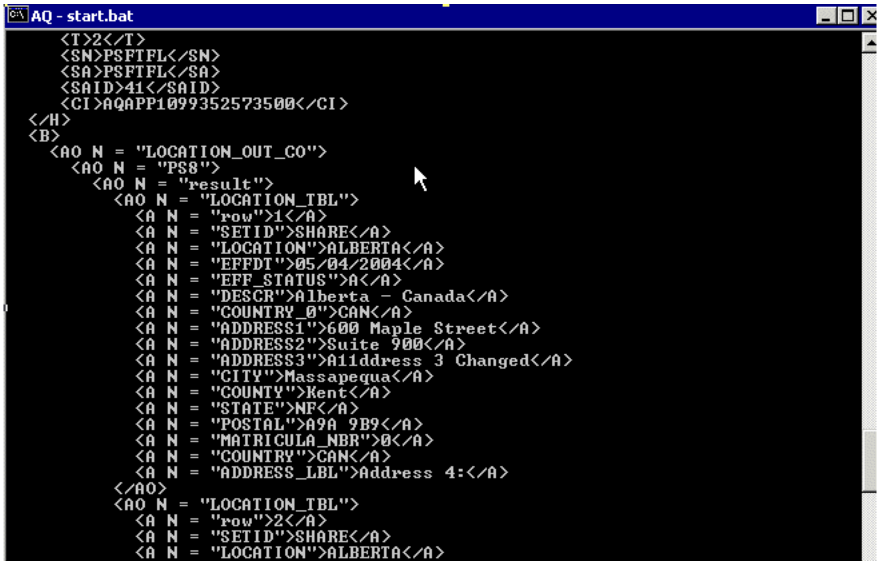
```
DECLARE
      enqueue_options     dbms_aq.enqueue_options_t;
      message_properties  dbms_aq.message_properties_t;
      msgid               RAW(16);
      payload             RAW(5000);
BEGIN
      payload := utl_raw.cast_to_raw('<?xml version="1.0" encoding="UTF-8"?>
<PS8 SERVICENAME="LOCATION" METHODNAME="LOCATION" LICENSE="test">
<component perform="browse">LOCATION</component>
<key name="SETID">SHARE</key>
<key name="LOCATION">ALBERTA</key>
</PS8>');
      dbms_aq.enqueue(queue_name         => 'xml_raw_q1',
                      enqueue_options    => enqueue_options,
                      message_properties => message_properties,
                      payload            => payload,
                      msgid              => msgid);
      COMMIT;
END;
/
```

The following image shows the `PSFTFL` example. It receives a reply from PeopleSoft and returns the reply to the hub.



```
The message was sent to topic(s) {oai_hub_queue=[AQAPP]}. Processing Time = 8,92
2 ms.
<?xml version = '1.0' encoding = 'UTF-8'?>
<!DOCTYPE MSG>
<MSG>
  <H>
     <BO>PSFTFL</BO>
     <EN>LOCATION</EN>
     <EV>OAI/V1</EV>
     <MV>OAI/V1</MV>
     <T>2</T>
     <SN>PSFTFL</SN>
     <SA>PSFTFL</SA>
     <SAID>41</SAID>
     <CI>AQAPP1099352573500</CI>
  </H>
  <B>
    <AO N = "LOCATION_OUT_CO">
      <AO N = "PS8">
        <AO N = "result">
          <AO N = "LOCATION_TBL">
            <A N = "row">1</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
            <A N = "EFFDT">05/04/2004</A>
            <A N = "EFF_STATUS">A</A>
            <A N = "DESCR">Alberta - Canada</A>
            <A N = "COUNTRY_0">CAN</A>
            <A N = "ADDRESS1">600 Maple Street</A>
            <A N = "ADDRESS2">Suite 900</A>
            <A N = "ADDRESS3">A11ddress 3 Changed</A>
            <A N = "CITY">Massapequa</A>
            <A N = "COUNTY">Kent</A>
            <A N = "STATE">NF</A>
            <A N = "POSTAL">A9A 9B9</A>
            <A N = "MATRICULA_NBR">0</A>
            <A N = "COUNTRY">CAN</A>
            <A N = "ADDRESS_LBL">Address 4:</A>
          </AO>
          <AO N = "LOCATION_TBL">
            <A N = "row">2</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
```

The following image shows the AQ example. It receives a reply from the hub and writes the data to the database table.



```
     <T>2</T>
     <SN>PSFTFL</SN>
     <SA>PSFTFL</SA>
     <SAID>41</SAID>
     <CI>AQAPP1099352573500</CI>
  </H>
  <B>
    <AO N = "LOCATION_OUT_CO">
      <AO N = "PS8">
        <AO N = "result">
          <AO N = "LOCATION_TBL">
            <A N = "row">1</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
            <A N = "EFFDT">05/04/2004</A>
            <A N = "EFF_STATUS">A</A>
            <A N = "DESCR">Alberta - Canada</A>
            <A N = "COUNTRY_0">CAN</A>
            <A N = "ADDRESS1">600 Maple Street</A>
            <A N = "ADDRESS2">Suite 900</A>
            <A N = "ADDRESS3">A11ddress 3 Changed</A>
            <A N = "CITY">Massapequa</A>
            <A N = "COUNTY">Kent</A>
            <A N = "STATE">NF</A>
            <A N = "POSTAL">A9A 9B9</A>
            <A N = "MATRICULA_NBR">0</A>
            <A N = "COUNTRY">CAN</A>
            <A N = "ADDRESS_LBL">Address 4:</A>
          </AO>
          <AO N = "LOCATION_TBL">
            <A N = "row">2</A>
            <A N = "SETID">SHARE</A>
            <A N = "LOCATION">ALBERTA</A>
```

5 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 OracleAS Adapter JCA or with a BSE configuration.

Troubleshooting

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

- Application Explorer
- PeopleSoft
- OracleAS Adapter JCA
- BSE

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

- The OracleAS Adapter JCA trace information can be found under the `*OracleAS_home*\opmn\logs` directory.
- BSE trace information can be found under the `*OracleAS_home*\j2ee\home\applications\ws-app-adapter\ibse\ibselogs` directory.
- The log file for Application Explorer can be found under the `*OracleAS_home*\adapters\application\tools` directory.

Application Explorer

To use Application Explorer on Windows for debugging or testing purposes, invoke the ae batch script, `ae.bat`, found under *OracleAS_home*`\adapters\application\tools` or on UNIX invoke the ae script, `ae.sh`, found under *OracleAS_home*`/adapters/application/tools`.

Error	Solution
Cannot connect to the OracleAS Adapter for PeopleSoft from Application Explorer.	Ensure that: ■ PeopleSoft is running. ■ The PeopleSoft user ID and password are correct. ■ The port number is correct. ■ The custom component interface is properly installed.
The following error message appears: java.lang.IllegalStateException: java.lang.Exception: Error Logon to PeopleSoft System	You have provided invalid connection information for PeopleSoft or the wrong `psjoa.jar` is in the lib directory.
PeopleSoft does not appear in the Application Explorer Adapter node list.	Ensure that the PeopleSoft JAR files, `iwpsci84.jar` and `psjoa.jar`, are added to the lib directory.
The following error message appears: 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	The host name or port number for PeopleSoft is incorrect.
Properties are not displayed for a component interface.	You are using the wrong `iwpsci8x.jar` 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 `iwpsci81.jar` and `iwpsci84.jar` files in your lib directory. Stop your server, remove the unrequired jar file, and restart the server.

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: 1. Open Application Designer. 2. Select the Component Interface. 3. Use the test tool. If service works in test tool, then review the XML and check for redundant fields in XML.
The following error message appears: 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	The host name or port number for PeopleSoft is incorrect.
Component Interfaces and Messages do not appear in the adapter tree.	The project is not installed properly on the PeopleSoft system.

OracleAS Adapter JCA

Error	Solution
In Application Explorer, the following error message appears when you attempt to connect to an OracleAS Adapter JCA 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, *OracleAS_home*\adapters\application

BSE Error Messages

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

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 execution time, various conditions can cause errors in BSE when Web services that use adapters are running. 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. 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.

Adapter-Specific Error Handling

When an adapter raises an exception during execution, 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 or not 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 Server Adapter for PeopleSoft Invalid SOAP Request

When the PeopleSoft agent receives a SOAP request message that does not conform to the WSDL for the Web service being executed, 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

When the OracleAS 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, the following SOAP response is generated.

```
<SOAP-ENV:Enve
lope 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

When the OracleAS Adapter for PeopleSoft cannot connect to PeopleSoft, the following SOAP response is generated.

```
<?xml version="1.0" enco
ding="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

When the Oracle Application Server Adapter receives a SOAP request message that does not conform to the WSDL for the Web services being executed, 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 Application Server Adapter Request

When the adapter executes a SOAP request using input parameters passed that do not match records in the target system, 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>
```

6 Advanced Topics

This chapter includes the following topics:

- Using Web Services Policy-Based Security
- Migrating Repositories

Using Web Services Policy-Based Security

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 Policy-Based Security

Web services provide a layer of abstraction between the back-end business logic they invoke, and the user or application running the Web service. This enables easy application integration but raises the issue of controlling the use and execution of critical and sensitive business logic that is run as a Web service.

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 runtime 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 runtime, 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.

Configuring Web Services Policy-Based Security

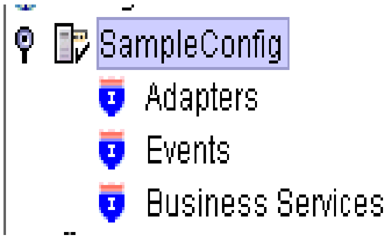
The following procedures describe how to configure Web services policy-based security.

Creating and Associating a User with a Policy

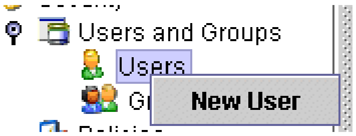
Before you create instances of policies, you must have a minimum of one user or one group to associate to an instance. You can create users and groups using Application Explorer.

1. Open Application Explorer.
2. Right-click the configuration to which you want to connect, for example, `SampleConfig.` See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.
3. Select **Connect.**

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

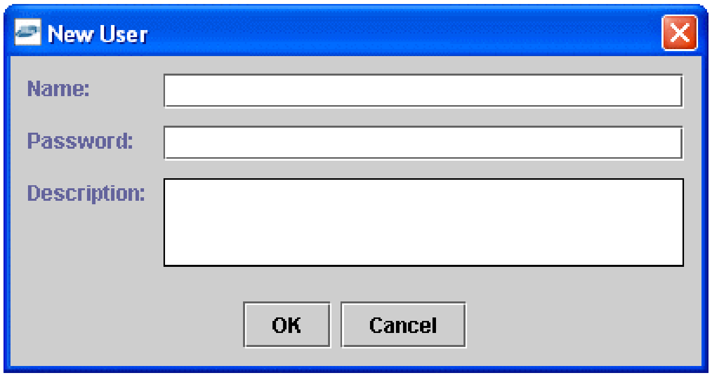


   a. Expand the **Business Services** node by clicking the plus (+) symbol.
   b. Expand the **Configuration** node by clicking the plus (+) symbol.
   c. Expand the **Security** node by clicking the plus (+) symbol.
   d. Expand the **Users and Groups** node by clicking the plus (+) symbol.



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

   The New User dialog box opens.

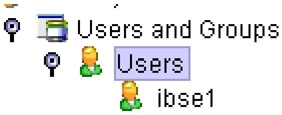


a. In the **Name** field, type a user ID.

b. In the **Password** field, type the password associated with the user ID.

c. In the **Description** field, type a description of the user (optional).

5. Click **OK.**



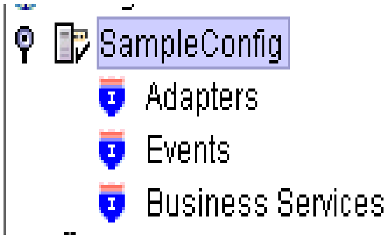
The new user is added under the Users node.

Creating a Group to Use With a Policy

To create a group to use with a policy:

1. Open Application Explorer.
2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.
3. Select **Connect.**

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

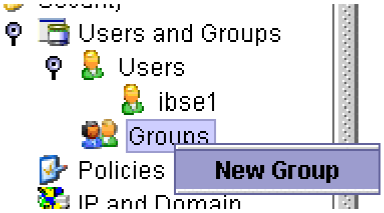


a. Expand the **Business Services** node by clicking the plus (+) symbol.

b. Expand the **Configuration** node by clicking the plus (+) symbol.

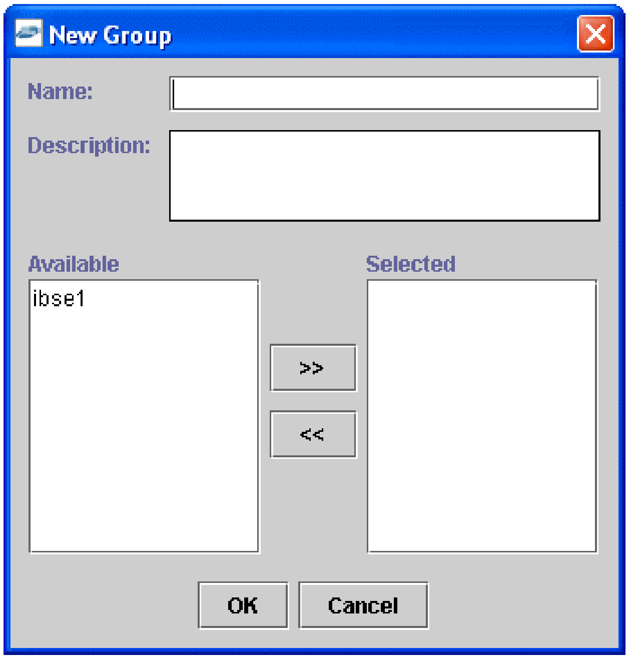
c. Expand the **Security** node by clicking the plus (+) symbol.

d. Expand the **Users and Groups** node by clicking the plus (+) symbol.



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

   The New Group dialog box opens.



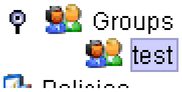
   a. In the **Name** field, type a a name for the group.

   b. In the **Description** field, type 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.

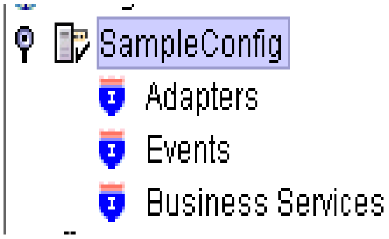
To create an execution policy:

1. Open Application Explorer.

2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.

3. Select **Connect.**

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

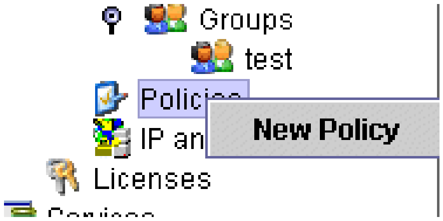


   a. Expand the **Business Services** node by clicking on the plus (+) symbol.

   b. Expand the **Configuration** node by clicking on the plus (+) symbol.

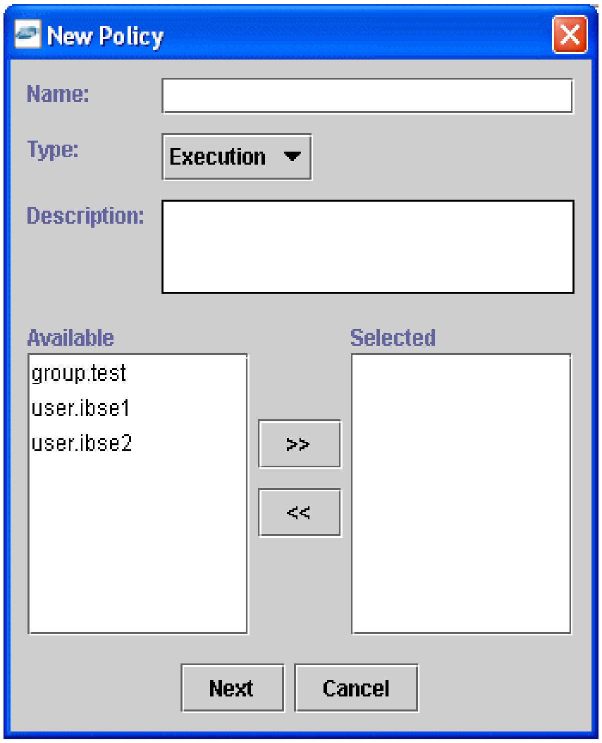
   c. Expand the **Security** node by clicking on the plus (+) symbol.

   d. Expand the **Policies** node by clicking on the plus (+) symbol.



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

   The New policy dialog box opens.



Type: Execution

   a. In the **Name** field, type a a name for the policy.

b. From the Type list, select **Execution.**

c. In the **Description** field, type a description for the policy (optional).

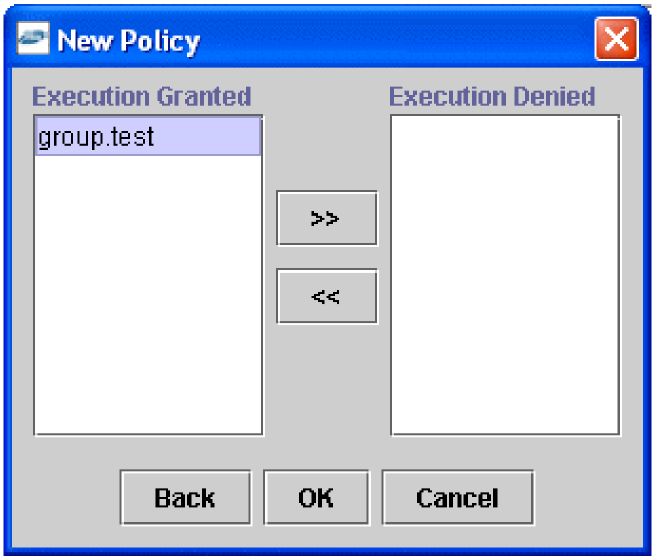
d. From the available list of users in the left pane, select one or more users and add them to the **Selected** list by clicking the double right-facing arrow.

> **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 box opens.

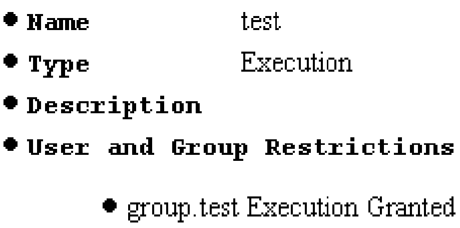


7. To grant permission to a user or group to execute a business service, select the user or group and move them into the **Execution Granted** list by selecting the double left-facing arrow.

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

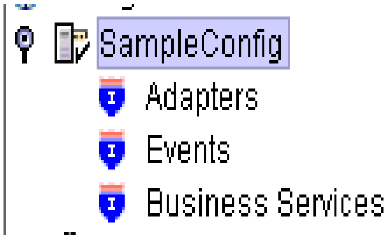
You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to BSE and therefore need not be applied to individual Web services. You need not create a policy; however, you must enable the Security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

1. Open Application Explorer.

2. Right-click the configuration to which you want to connect, for example, SampleConfig. See Chapter 2, "Adapter Configuration Using OracleAS Adapter Application Explorer" for information on creating a new configuration.

3. Select **Connect.**

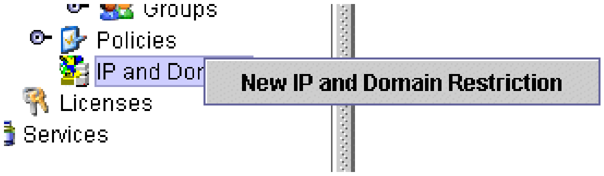
   Nodes appear for Adapters, Events, and Business Services (also known as Web services).



   a. Expand the **Business Services** node by clicking the plus (+) symbol.

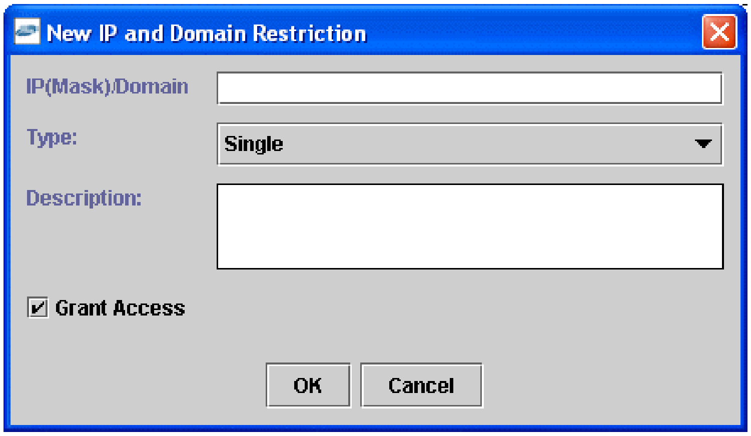
   b. Expand the **Configuration** node by clicking the plus (+) symbol.

   c. Expand the **Security** node by clicking the plus (+) symbol.



4. Right-click **IP and Domain** and select **New IP and Domain Restriction.**

   The New IP and Domain Restriction dialog box opens.



IP(Mask)/Domain

Type: Single

Description:

☑ Grant Access

   a. In the **IP(Mask)/Domain** field, type 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.

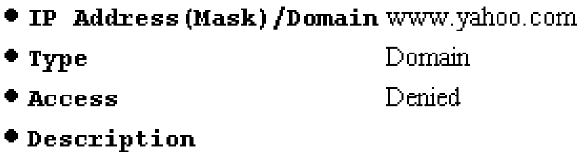
c. In the **Description** field, type a description (optional).

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**	

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 runtime. For management purposes, you can migrate BSE and JCA 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.

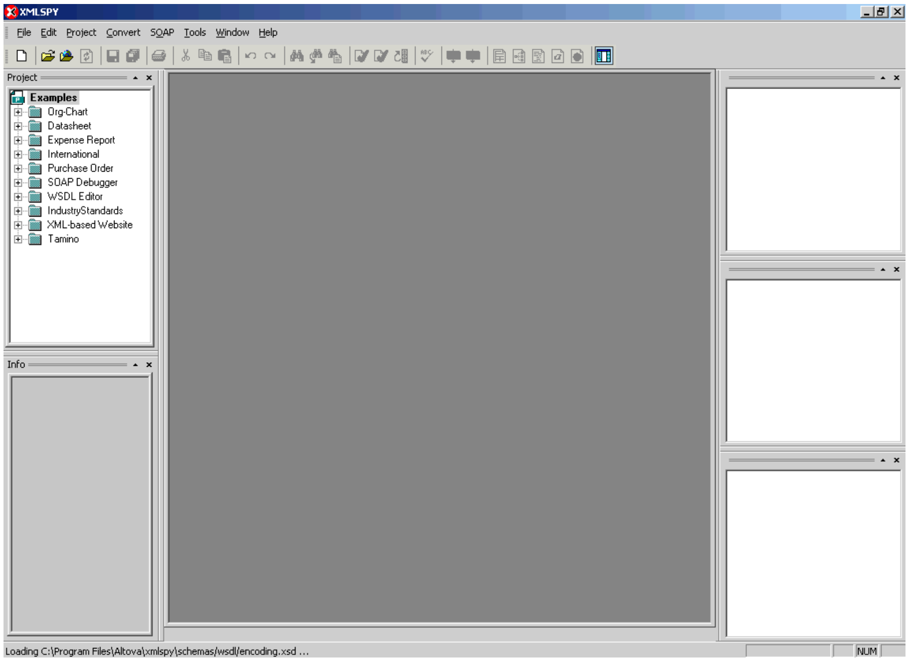
Migrating a BSE Repository

To migrate a BSE repository:

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

```
http://localhost:7777/ibse/IBSEServlet/admin/iwcontrol.ibs
```

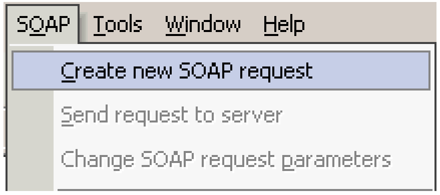
2. Open a third party XML editor, for example, XMLSPY.



Loading C:\Program Files\Altova\xmlspy\schemas\wsdl\encoding.xsd ...

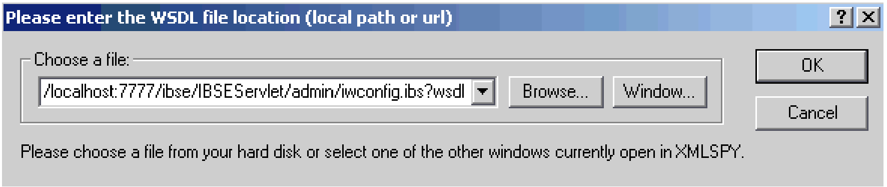
3. In the menu bar, click **SOAP.**

   A list of options appears.



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

   The WSDL file location dialog box opens.



Choose a file: /localhost:7777/ibse/IBSEServlet/admin/iwconfig.ibs?wsdl

Please choose a file from your hard disk or select one of the other windows currently open in XMLSPY.

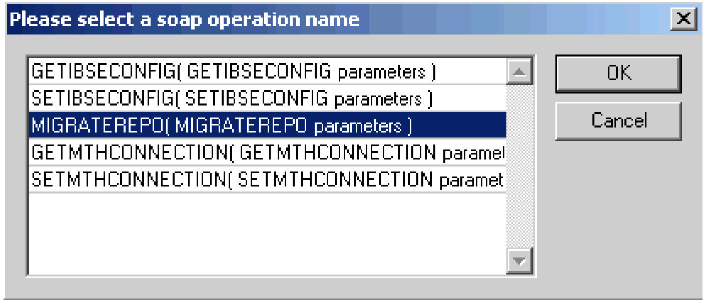
   a. In the **Choose a file** field, paste the BSE control service URL.

   b. Append **?wsdl** to the URL, for example:

```
http://localhost:7777/ibse/IBSEServlet/admin/iwcontrol.ibs?wsdl
```

5. Click **OK.**

   The soap operation name dialog box opens and lists the available control methods.

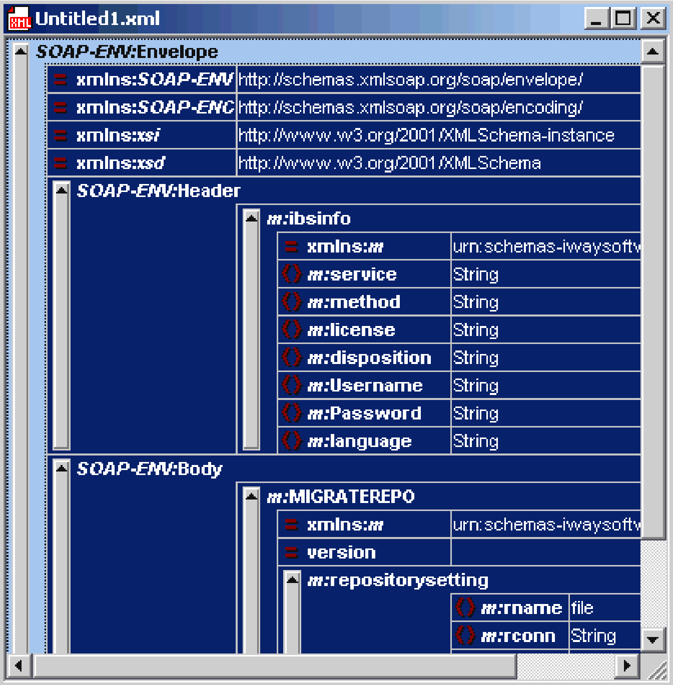


Please select a soap operation name

GETIBSECONFIG(GETIBSECONFIG parameters)
SETIBSECONFIG(SETIBSECONFIG parameters)
MIGRATEREPO(MIGRATEREPO parameters)
GETMTHCONNECTION(GETMTHCONNECTION paramet
SETMTHCONNECTION(SETMTHCONNECTION paramet

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

   The following window opens, which shows the structure of the SOAP envelope.



Untitled1.xml

SOAP-ENV:Envelope

xmlns:SOAP-ENV	http://schemas.xmlsoap.org/soap/envelope/
xmlns:SOAP-ENC	http://schemas.xmlsoap.org/soap/encoding/
xmlns:xsi	http://www.w3.org/2001/XMLSchema-instance
xmlns:xsd	http://www.w3.org/2001/XMLSchema

SOAP-ENV:Header

m:ibsinfo

xmlns:m	urn:schemas-iwaysoftw
m:service	String
m:method	String
m:license	String
m:disposition	String
m:Username	String
m:Password	String
m:language	String

SOAP-ENV:Body

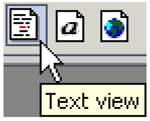
m:MIGRATEREPO

xmlns:m	urn:schemas-iwaysoftw
version	

m:repositorysetting

m:rname	file
m:rconn	String

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>
```

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.

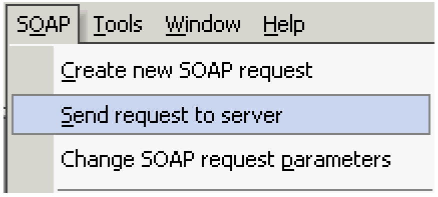
    - If you want to migrate a single Web service from the current BSE repository, enter the Web service name in the `<m:servicename>` tag, for example:

      ```
      <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. In the menu bar, click **SOAP** and select **Send request to server.**

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

Migrating a JCA Repository

To migrate a JCA repository:

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

   ```
   OracleAS_home\adapters\application
   ```

2. Locate and copy the `repository.xml` file.

3. Place this file in a new JCA configuration directory to migrate the existing repository.

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

A

Generating Component Interface APIs

This section describes how to generate component interface APIs to use with the OracleAS Adapter for PeopleSoft.

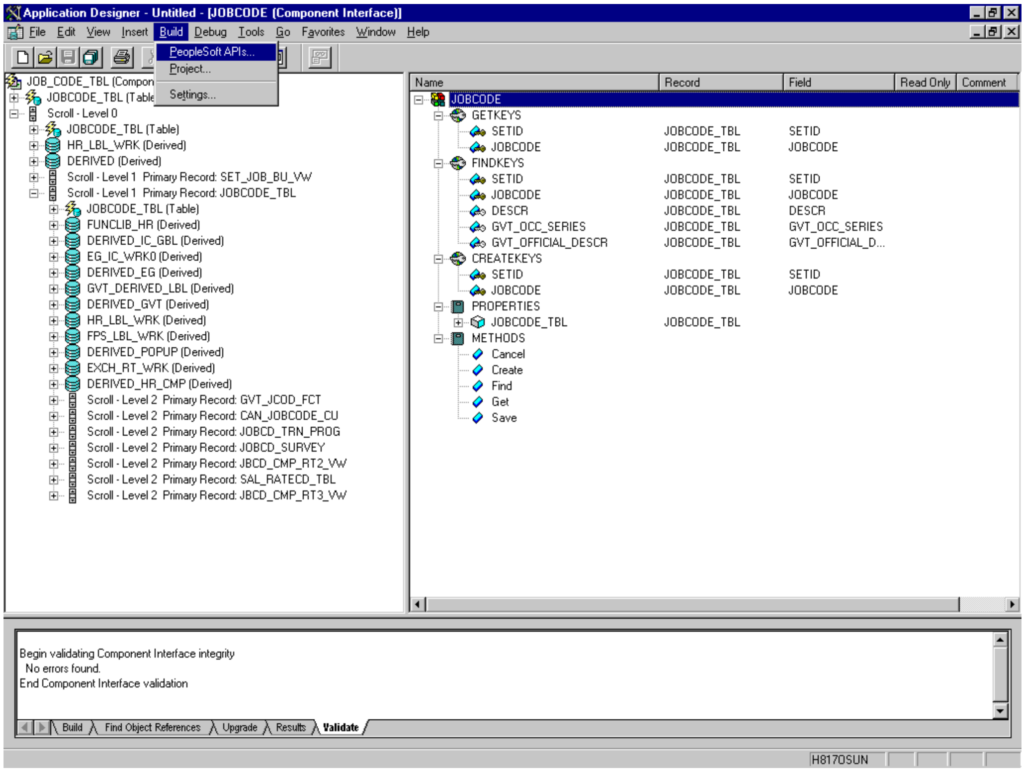
Building the PeopleSoft API Java Programs

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 machine 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.



```
JOB_CODE_TBL (Compon
  JOBCODE_TBL (Table
  Scroll - Level 0
    JOBCODE_TBL (Table)
    HR_LBL_WRK (Derived)
    DERIVED (Derived)
    Scroll - Level 1  Primary Record: SET_JOB_BU_VW
    Scroll - Level 1  Primary Record: JOBCODE_TBL
      JOBCODE_TBL (Table)
      FUNCLIB_HR (Derived)
      DERIVED_IC_GBL (Derived)
      EG_IC_WRK0 (Derived)
      DERIVED_EG (Derived)
      GVT_DERIVED_LBL (Derived)
      DERIVED_GVT (Derived)
      HR_LBL_WRK (Derived)
      FPS_LBL_WRK (Derived)
      DERIVED_POPUP (Derived)
      EXCH_RT_WRK (Derived)
      DERIVED_HR_CMP (Derived)
      Scroll - Level 2  Primary Record: GVT_JCOD_FCT
      Scroll - Level 2  Primary Record: CAN_JOBCODE_CU
      Scroll - Level 2  Primary Record: JOBCD_TRN_PROG
      Scroll - Level 2  Primary Record: JOBCD_SURVEY
      Scroll - Level 2  Primary Record: JBCD_CMP_RT2_VW
      Scroll - Level 2  Primary Record: SAL_RATECD_TBL
      Scroll - Level 2  Primary Record: JBCD_CMP_RT3_VW
```

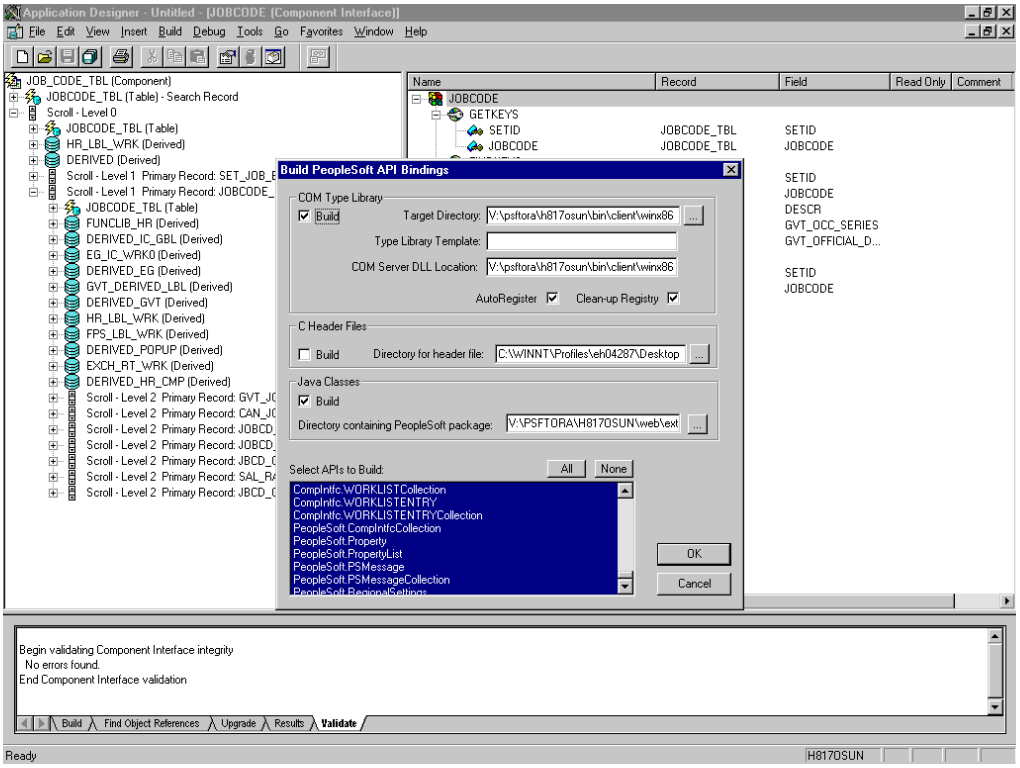
Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
DESCR	JOBCODE_TBL	DESCR		
GVT_OCC_SERIES	JOBCODE_TBL	GVT_OCC_SERIES		
GVT_OFFICIAL_DESCR	JOBCODE_TBL	GVT_OFFICIAL_D...		
CREATEKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
PROPERTIES				
JOBCODE_TBL	JOBCODE_TBL			
METHODS				
Cancel				
Create				
Find				
Get				
Save				

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

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

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

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


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_E
Scroll - Level 1 Primary Record: JOBCODE_
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JC
Scroll - Level 2 Primary Record: CAN_JC
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JBCD_C
Scroll - Level 2 Primary Record: SAL_RA
Scroll - Level 2 Primary Record: JBCD_C
Name
Record
Field
Read Only
Comment
JOBCODE
GETKEYS
SETID
JOBCODE
JOBCODE_TBL
JOBCODE_TBL
SETID
JOBCODE
SETID
JOBCODE
DESCR
GVT_OCC_SERIES
GVT_OFFICIAL_D...
SETID
JOBCODE
Build
Find Object References
Upgrade
Results
Validate
Ready
H817OSUN

Build PeopleSoft API Bindings

COM Type Library

☑ Build Target Directory: V:\psftora\h817osun\bin\client\winx86

Type Library Template:

COM Server DLL Location: V:\psftora\h817osun\bin\client\winx86

AutoRegister ☑ Clean-up Registry ☑

C Header Files

☐ Build Directory for header file: C:\WINNT\Profiles\eh04287\Desktop

Java Classes

☑ Build

Directory containing PeopleSoft package: V:\PSFTORA\H817OSUN\web\ext

Select APIs to Build: All None

CompIntfc.WORKLISTCollection
CompIntfc.WORKLISTENTRY
CompIntfc.WORKLISTENTRYCollection
PeopleSoft.CompIntfcCollection
PeopleSoft.Property
PeopleSoft.PropertyList
PeopleSoft.PSMessage
PeopleSoft.PSMessageCollection
PeopleSoft.RegionalSettings

OK Cancel

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

4. Because you are creating Java files, ensure you deselect **COM Type Library Build.**

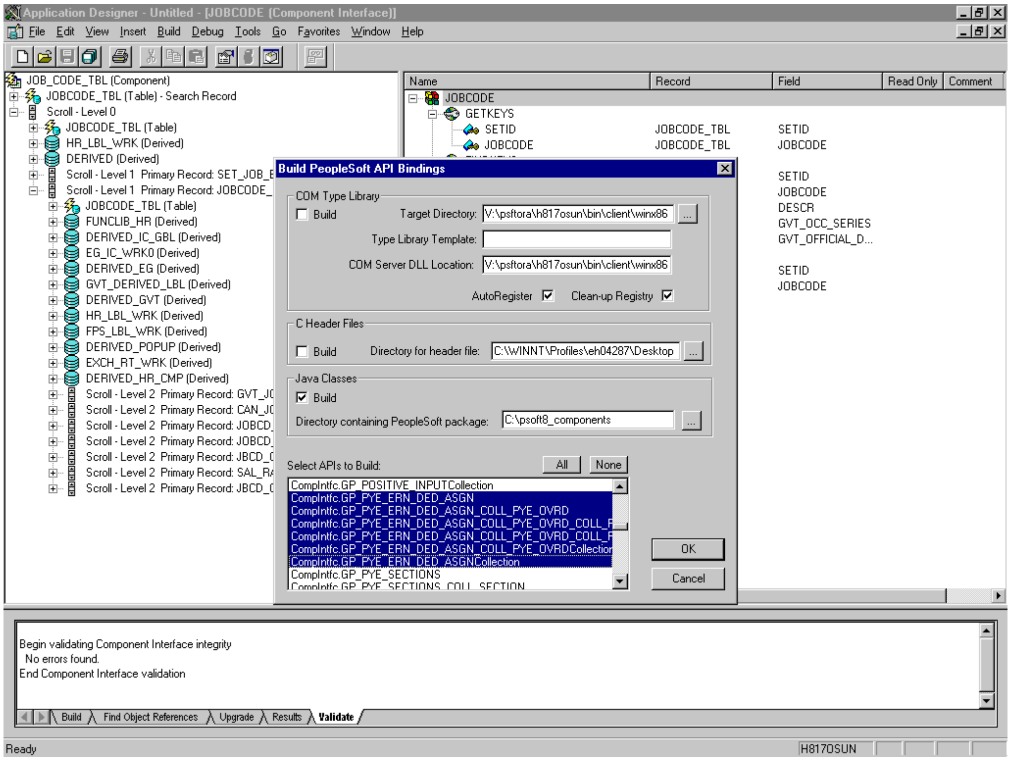
5. Ensure that **Java Classes Build** is selected, and then select a directory on your local machine where the Java files are to be placed, for example, `c:\psoft8_components`.

   - 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.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Name
Record
Field
Read Only
Comment
OK
Cancel
All
None
Build
Find Object References
Upgrade
Results
Validate
Ready
H817OSUN

JOB_CODE_TBL (Component)
- JOBCODE_TBL (Table) - Search Record
- Scroll - Level 0
  - JOBCODE_TBL (Table)
  - HR_LBL_WRK (Derived)
  - DERIVED (Derived)
  - Scroll - Level 1 Primary Record: SET_JOB_E
  - Scroll - Level 1 Primary Record: JOBCODE_
    - JOBCODE_TBL (Table)
    - FUNCLIB_HR (Derived)
    - DERIVED_IC_GBL (Derived)
    - EG_IC_WRK0 (Derived)
    - DERIVED_EG (Derived)
    - GVT_DERIVED_LBL (Derived)
    - DERIVED_GVT (Derived)
    - HR_LBL_WRK (Derived)
    - FPS_LBL_WRK (Derived)
    - DERIVED_POPUP (Derived)
    - EXCH_RT_WRK (Derived)
    - DERIVED_HR_CMP (Derived)
    - Scroll - Level 2 Primary Record: GVT_JC
    - Scroll - Level 2 Primary Record: CAN_JC
    - Scroll - Level 2 Primary Record: JOBCD_
    - Scroll - Level 2 Primary Record: JOBCD_
    - Scroll - Level 2 Primary Record: JBCD_C
    - Scroll - Level 2 Primary Record: SAL_RA
    - Scroll - Level 2 Primary Record: JBCD_C

Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
		SETID		
		JOBCODE		
		DESCR		
		GVT_OCC_SERIES		
		GVT_OFFICIAL_D...		
		SETID		
		JOBCODE		

Build PeopleSoft API Bindings

COM Type Library
☐ Build Target Directory: V:\psftora\h817osun\bin\client\winx86
Type Library Template:
COM Server DLL Location: V:\psftora\h817osun\bin\client\winx86
AutoRegister ☑ Clean-up Registry ☑

C Header Files
☐ Build Directory for header file: C:\WINNT\Profiles\eh04287\Desktop

Java Classes
☑ Build
Directory containing PeopleSoft package: C:\psoft8_components

Select APIs to Build:

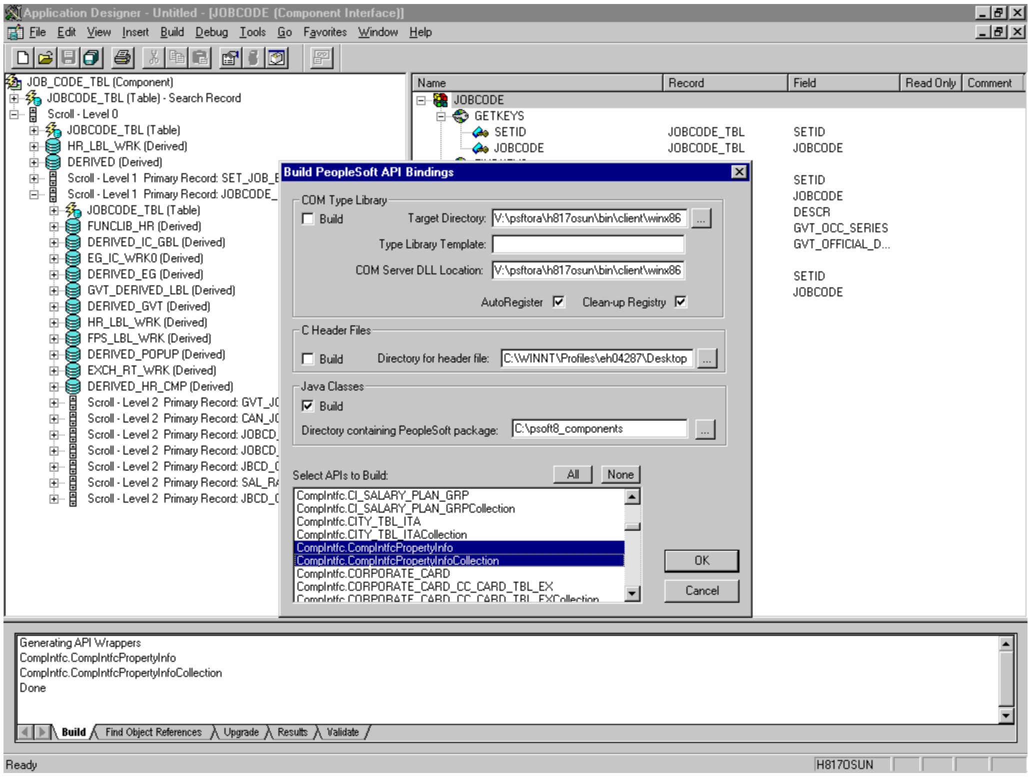
CompIntfc.GP_POSITIVE_INPUTCollection
CompIntfc.GP_PYE_ERN_DED_ASGN
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRD
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRD_COLL_F
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRD_COLL_F
CompIntfc.GP_PYE_ERN_DED_ASGN_COLL_PYE_OVRDCollection
CompIntfc.GP_PYE_ERN_DED_ASGNCollection
CompIntfc.GP_PYE_SECTIONS
CompIntfc.GP_PYE_SECTIONS_COLL_SECTION

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

- 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 dialog box displays the following APIs, including generic component interface properties.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_E
Scroll - Level 1 Primary Record: JOBCODE_
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JC
Scroll - Level 2 Primary Record: CAN_JC
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JOBCD_
Scroll - Level 2 Primary Record: JBCD_C
Scroll - Level 2 Primary Record: SAL_RA
Scroll - Level 2 Primary Record: JBCD_C
Name Record Field Read Only Comment
JOBCODE
GETKEYS
SETID JOBCODE_TBL SETID
JOBCODE JOBCODE_TBL JOBCODE
SETID
JOBCODE
DESCR
GVT_OCC_SERIES
GVT_OFFICIAL_D...
SETID
JOBCODE
Build Find Object References Upgrade Results Validate
Ready
H817OSUN

Build PeopleSoft API Bindings

COM Type Library

□ Build Target Directory: V:\psftora\h817osun\bin\client\winx86

Type Library Template:

COM Server DLL Location: V:\psftora\h817osun\bin\client\winx86

AutoRegister ☑ Clean-up Registry ☑

C Header Files

□ Build Directory for header file: C:\WINNT\Profiles\eh04287\Desktop

Java Classes

☑ Build

Directory containing PeopleSoft package: C:\psoft8_components

Select APIs to Build: All None

```
CompIntfc.CI_SALARY_PLAN_GRP
CompIntfc.CI_SALARY_PLAN_GRPCollection
CompIntfc.CITY_TBL_ITA
CompIntfc.CITY_TBL_ITACollection
CompIntfc.CompIntfcPropertyInfo
CompIntfc.CompIntfcPropertyInfoCollection
CompIntfc.CORPORATE_CARD
CompIntfc.CORPORATE_CARD_CC_CARD_TBL_EX
CompIntfc.CORPORATE_CARD_CC_CARD_TBL_EXCollection
```

OK Cancel

```
Generating API Wrappers
CompIntfc.CompIntfcPropertyInfo
CompIntfc.CompIntfcPropertyInfoCollection
Done
```

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.

Compiling the PeopleSoft API Java Programs

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

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 machine where you installed Application Explorer or on another machine.

> **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.

To compile the PeopleSoft API Java programs:

- If you are compiling on the same machine 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 machine other than the one where you installed Application Explorer:

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 BAT 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
```

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

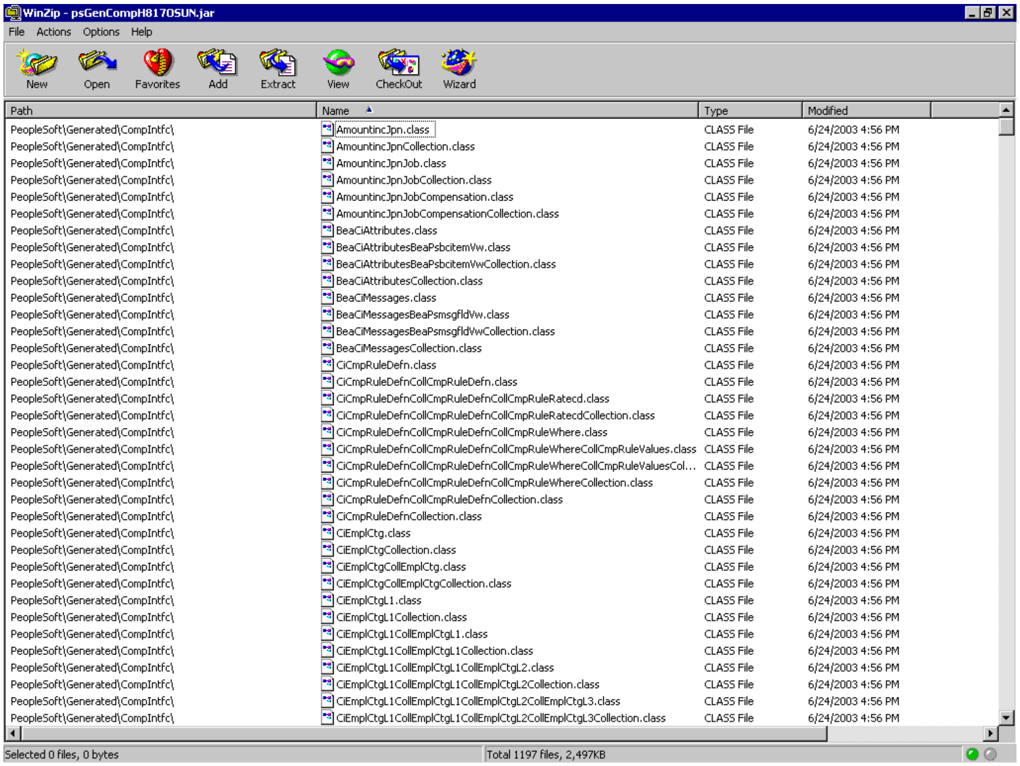
This code places the class files in the same directory with the Java files, but you can choose a different location depending on your site requirements.

3. Compress the class files into a JAR file.

   The following Windows BAT 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.



Path	Name	Type	Modified
PeopleSoft\Generated\CompIntfc\	AmountincJpn.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJob.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJobCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJobCompensation.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	AmountincJpnJobCompensationCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributes.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributesBeaPsbcitemVw.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributesBeaPsbcitemVwCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiAttributesCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessages.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessagesBeaPsmsgfldVw.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessagesBeaPsmsgfldVwCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	BeaCiMessagesCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefn.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefn.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleRatecd.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleRatecdCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhere.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhereCollCmpRuleValues.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhereCollCmpRuleValuesCol...	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollCmpRuleWhereCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollCmpRuleDefnCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiCmpRuleDefnCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtg.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgCollEmplCtg.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgCollEmplCtgCollection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1Collection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1Collection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2Collection.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2CollEmplCtgL3.class	CLASS File	6/24/2003 4:56 PM
PeopleSoft\Generated\CompIntfc\	CiEmplCtgL1CollEmplCtgL1CollEmplCtgL2CollEmplCtgL3Collection.class	CLASS File	6/24/2003 4:56 PM

Selected 0 files, 0 bytes

Total 1197 files, 2,497KB

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

5. Place the JAR file in the adapters common lib directory. This enables the OracleAS Adapter for PeopleSoft to communicate with the PeopleSoft component interface.

   ```
   OracleAS_home\adapters\application\lib
   ```

Note: If you are running on UNIX, do the compile and JAR steps on Windows and then move the file to your UNIX machine. 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.

B

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 8 Integration Broker" for more information. For a complete description *before* you work with the OracleAS 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.

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 8 Integration Broker" for more information.

1. Configure the gateway for the TCP/IP Target Connector or HTTP Target Connector.

   > **Note:** This step is optional when configuring the HTTP Connector.

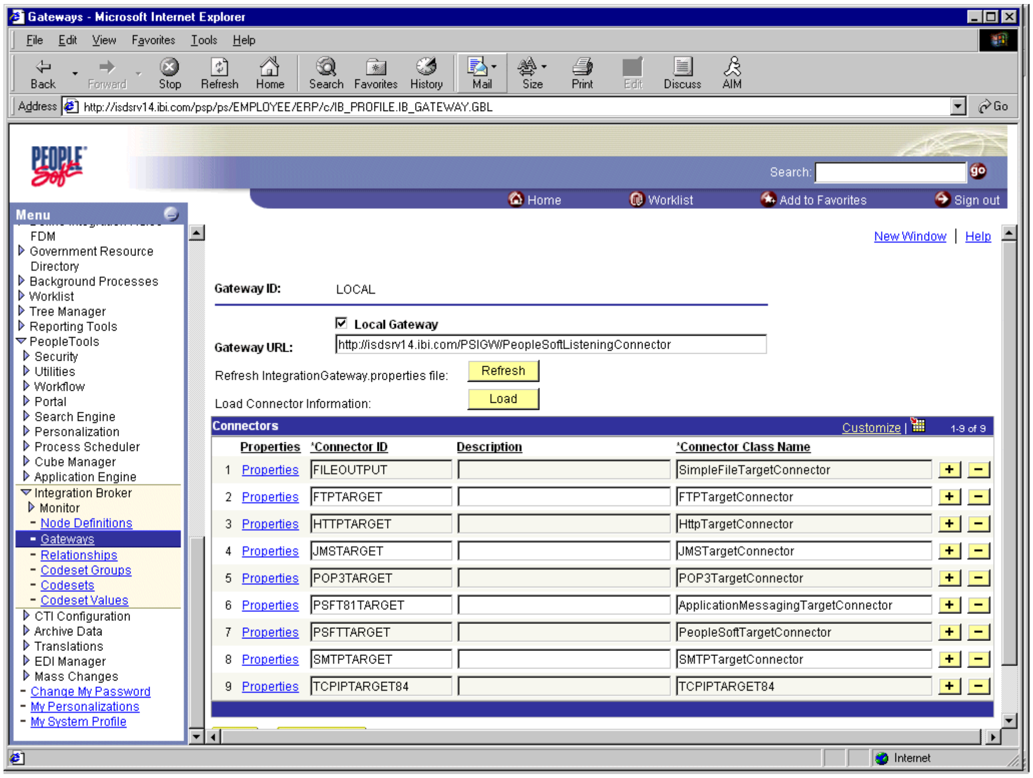
2. Configure the node.

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 opens.



Gateway ID: LOCAL

☑ Local Gateway

Gateway URL: http://isdsrv14.ibi.com/PSIGW/PeopleSoftListeningConnector

Refresh IntegrationGateway.properties file: Refresh

Load Connector Information: Load

Connectors — Customize | 1-9 of 9

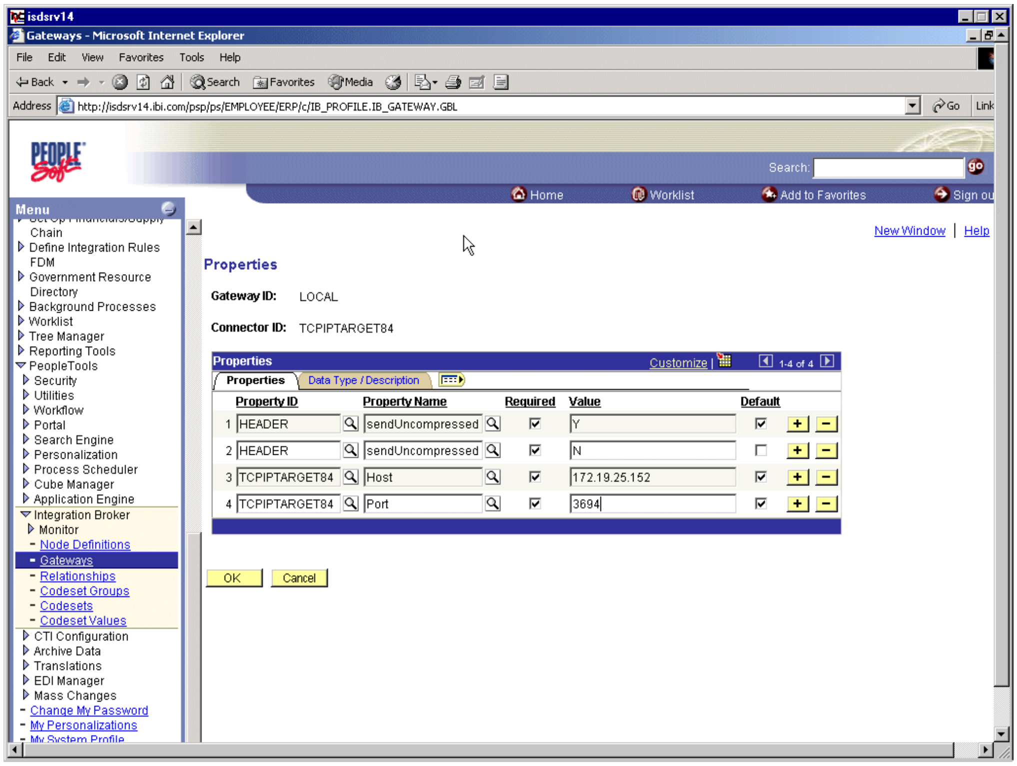
	Properties	*Connector ID	Description	*Connector Class Name		
1	Properties	FILEOUTPUT		SimpleFileTargetConnector	+	−
2	Properties	FTPTARGET		FTPTargetConnector	+	−
3	Properties	HTTPTARGET		HttpTargetConnector	+	−
4	Properties	JMSTARGET		JMSTargetConnector	+	−
5	Properties	POP3TARGET		POP3TargetConnector	+	−
6	Properties	PSFT81TARGET		ApplicationMessagingTargetConnector	+	−
7	Properties	PSFTTARGET		PeopleSoftTargetConnector	+	−
8	Properties	SMTPTARGET		SMTPTargetConnector	+	−
9	Properties	TCPIPTARGET84		TCPIPTARGET84	+	−

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.

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

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



Properties

Gateway ID: LOCAL

Connector ID: TCPIPTARGET84

Properties — Properties | Data Type / Description

	Property ID	Property Name	Required	Value	Default
1	HEADER	sendUncompressed	☑	Y	☑
2	HEADER	sendUncompressed	☑	N	☐
3	TCPIPTARGET84	Host	☑	172.19.25.152	☑
4	TCPIPTARGET84	Port	☑	3694	☑

b. Type the values for the host and the port for the machine on which your PeopleSoft XML listener is listening for incoming messages.

5. Click **OK.**

   The Gateway window opens.

6. Scroll to the bottom of the window and click **Save.**

You have finished configuring the gateway for the TCP/IP Target Connector.

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 8 Integration Broker" or your PeopleSoft documentation.

   a. Select **External** from the **Node Type** list.

   b. Select **Implicit** from the **Routing Type** list.

3. Select the **Connectors** tab.

   a. Select **TCPIPTARGET84** as the Connector ID.

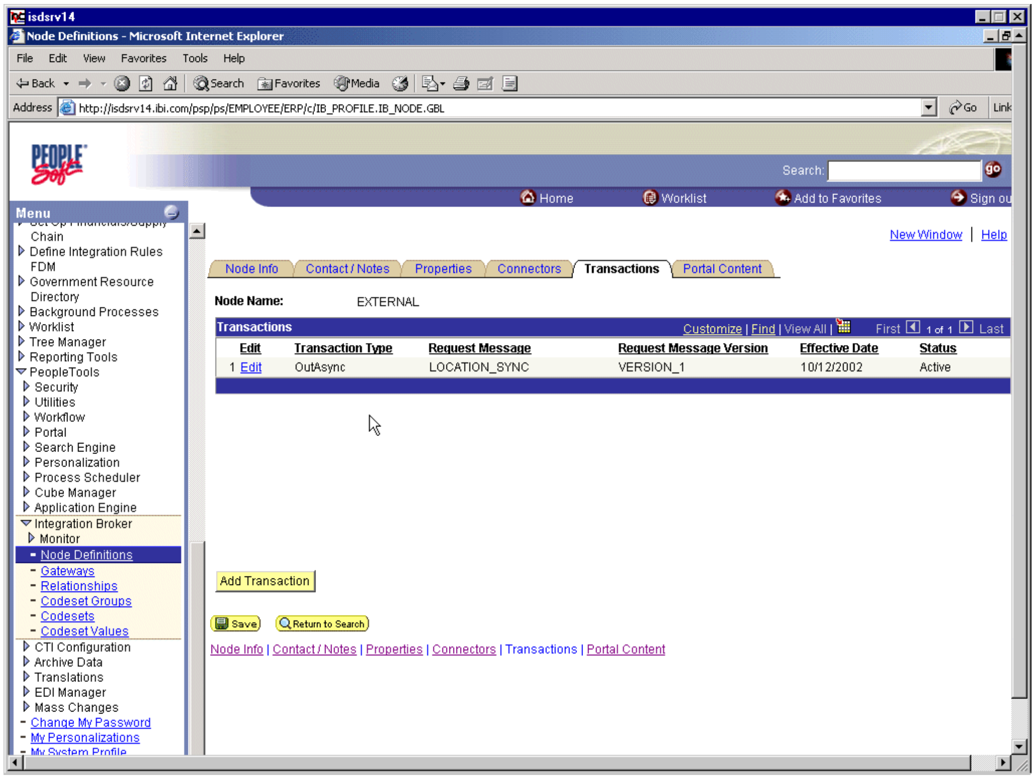
   Default values appear for the host and the port.

b. Type the values for the host and the port for the machine and port that route XML to Oracle Application 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.



Node Name: EXTERNAL

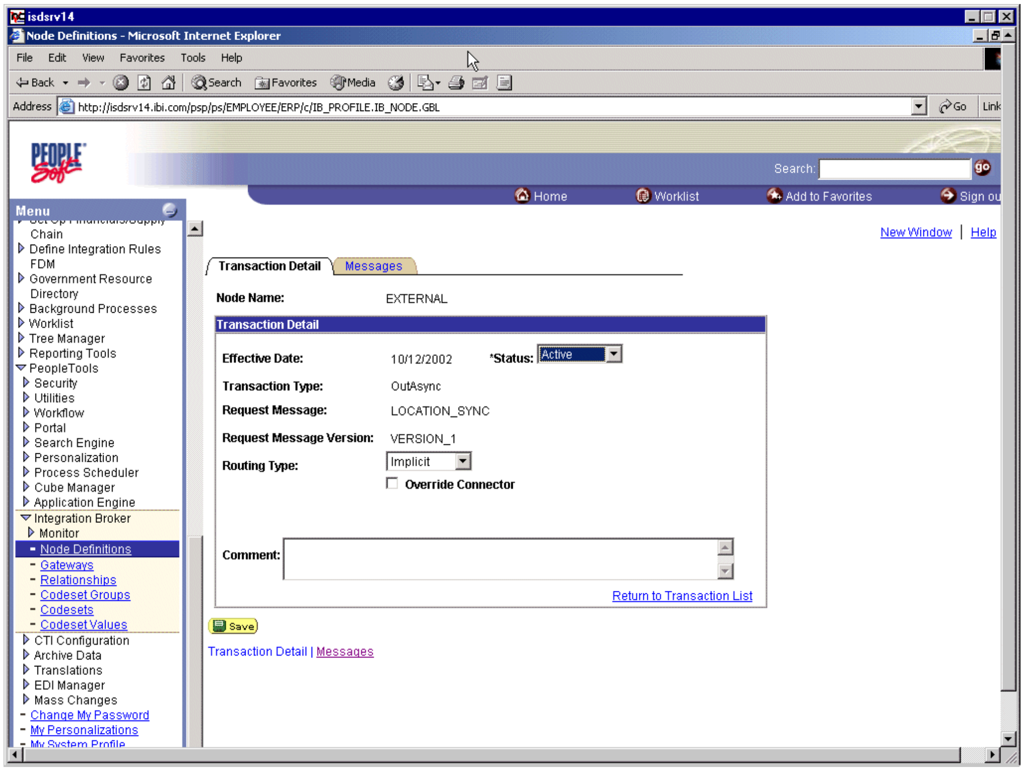
Transactions

	Edit	Transaction Type	Request Message	Request Message Version	Effective Date	Status
1	Edit	OutAsync	LOCATION_SYNC	VERSION_1	10/12/2002	Active

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



Node Name: EXTERNAL

Transaction Detail

Effective Date: 10/12/2002 *Status: Active

Transaction Type: OutAsync

Request Message: LOCATION_SYNC

Request Message Version: VERSION_1

Routing Type: Implicit

□ Override Connector

Comment:

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.

   a. To edit additional transactions, click the **Edit** link to navigate 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 are ready to send XML messages to your PeopleSoft XML listener.

Configuring the HTTP Target Connector

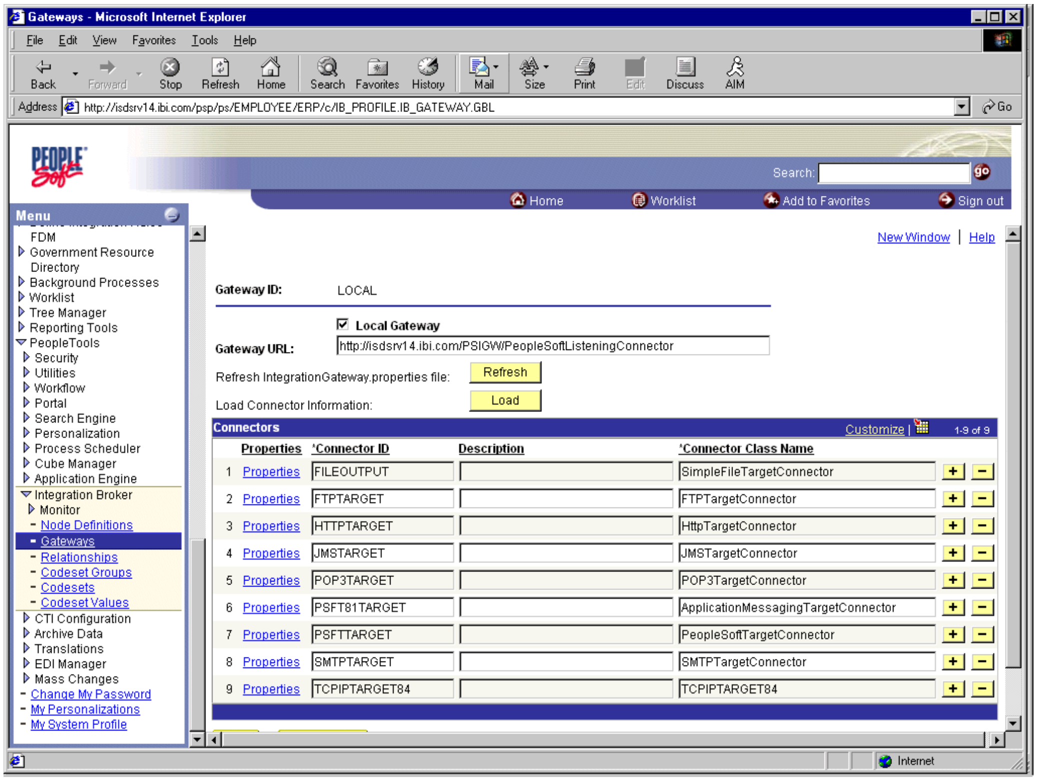
The HTTP Target Connector is supplied with your PeopleSoft application, and no special configuration steps are required. If you choose, you may configure default connection values on the Gateway. You can override these values when you configure the node. To configure the gateway for the HTTP Target Connector:

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 opens.



Gateway ID: LOCAL

☑ Local Gateway

Gateway URL: http://isdsrv14.ibi.com/PSIGW/PeopleSoftListeningConnector

Refresh IntegrationGateway.properties file: Refresh

Load Connector Information: Load

Connectors Customize | 1-9 of 9

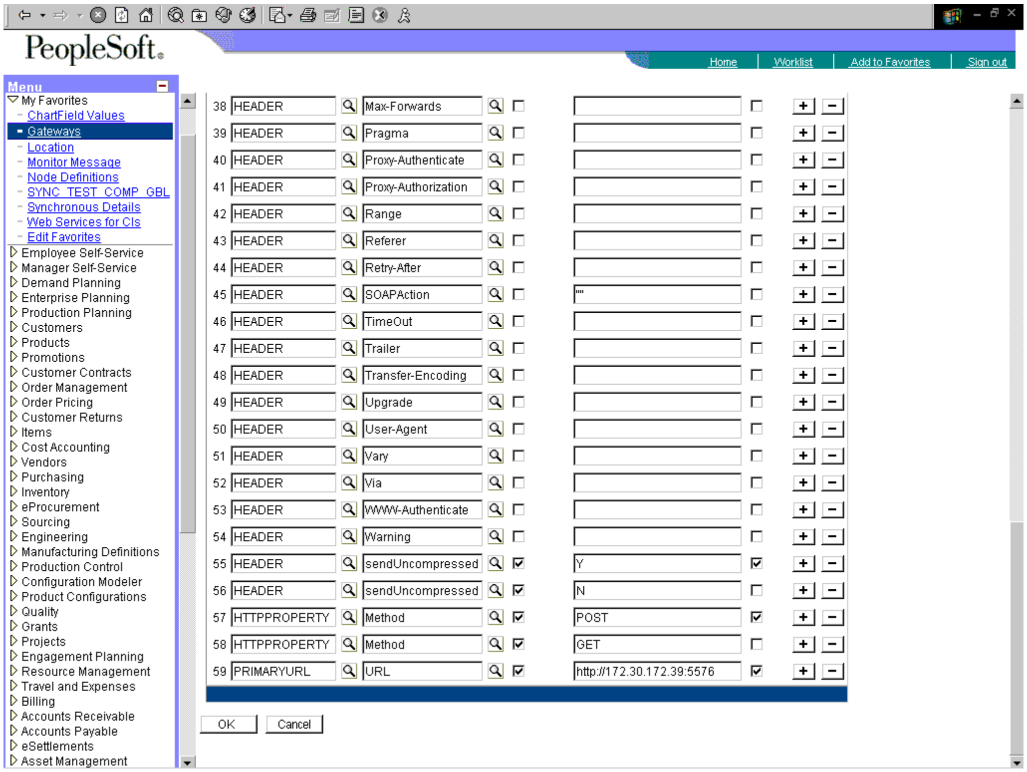
	Properties	*Connector ID	Description	*Connector Class Name		
1	Properties	FILEOUTPUT		SimpleFileTargetConnector	+	−
2	Properties	FTPTARGET		FTPTargetConnector	+	−
3	Properties	HTTPTARGET		HttpTargetConnector	+	−
4	Properties	JMSTARGET		JMSTargetConnector	+	−
5	Properties	POP3TARGET		POP3TargetConnector	+	−
6	Properties	PSFT81TARGET		ApplicationMessagingTargetConnector	+	−
7	Properties	PSFTTARGET		PeopleSoftTargetConnector	+	−
8	Properties	SMTPTARGET		SMTPTargetConnector	+	−
9	Properties	TCPIPTARGET84		TCPIPTARGET84	+	−

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 opens, with default values.



38	HEADER	Max-Forwards	□		□	+	−
39	HEADER	Pragma	□		□	+	−
40	HEADER	Proxy-Authenticate	□		□	+	−
41	HEADER	Proxy-Authorization	□		□	+	−
42	HEADER	Range	□		□	+	−
43	HEADER	Referer	□		□	+	−
44	HEADER	Retry-After	□		□	+	−
45	HEADER	SOAPAction	□	""	□	+	−
46	HEADER	TimeOut	□		□	+	−
47	HEADER	Trailer	□		□	+	−
48	HEADER	Transfer-Encoding	□		□	+	−
49	HEADER	Upgrade	□		□	+	−
50	HEADER	User-Agent	□		□	+	−
51	HEADER	Vary	□		□	+	−
52	HEADER	Via	□		□	+	−
53	HEADER	WWW-Authenticate	□		□	+	−
54	HEADER	Warning	□		□	+	−
55	HEADER	sendUncompressed	☑	Y	☑	+	−
56	HEADER	sendUncompressed	☑	N	□	+	−
57	HTTPPROPERTY	Method	☑	POST	☑	+	−
58	HTTPPROPERTY	Method	☑	GET	□	+	−
59	PRIMARYURL	URL	☑	http://172.30.172.39:5576	☑	+	−

6. Scroll to the bottom and type a value for the **PRIMARYURL.**

   This is the default HTTP address (machine 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 opens.

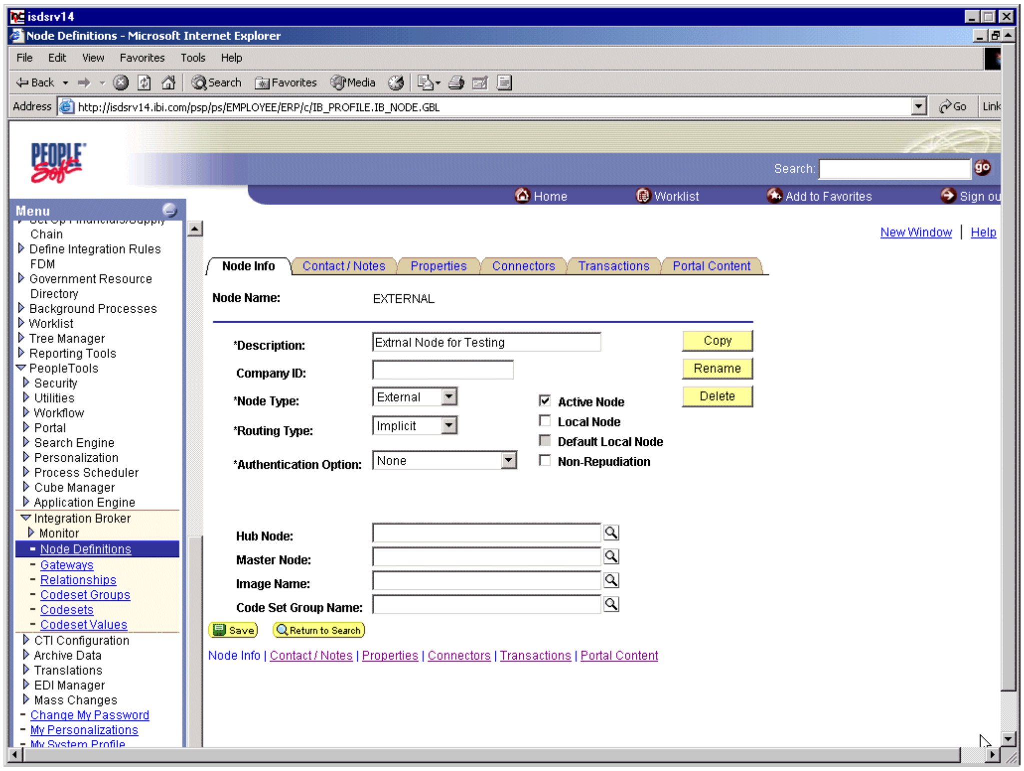
8. Scroll to the bottom of the window and click **Save.**

You have finished configuring the gateway for the HTTP Target Connector.

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 Application Server.

To configure the node to use the HTTP Connector:



Node Name: EXTERNAL

*Description: Extrnal Node for Testing

Company ID:

*Node Type: External

*Routing Type: Implicit

*Authentication Option: None

☑ Active Node

□ Local Node

□ Default Local Node

□ Non-Repudiation

Hub Node:

Master Node:

Image Name:

Code Set Group Name:

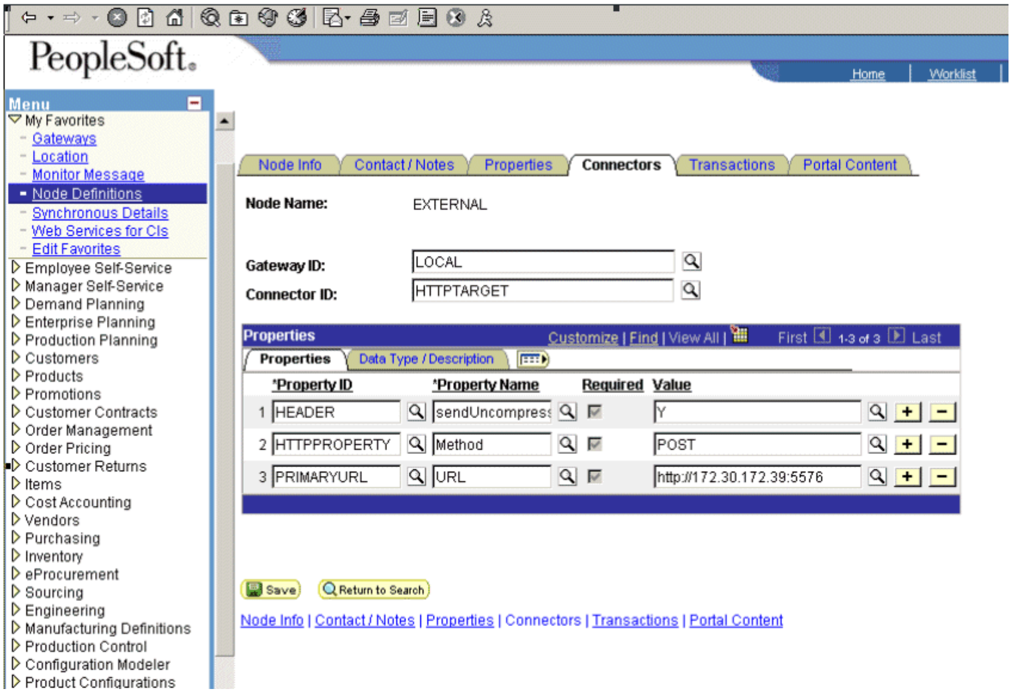
1. In the Menu pane, expand **PeopleTools**, **Integration Broker**, and then click **Node Definitions.**

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 8 Integration Broker" or your PeopleSoft documentation.

   a. From the **Node Type** list, select **External.**

   b. From the **Routing Type** list, select **Implicit.**



Node Name: EXTERNAL

Gateway ID: LOCAL

Connector ID: HTTPTARGET

Properties

	*Property ID	*Property Name	Required	Value
1	HEADER	sendUncompress	☑	Y
2	HTTPPROPERTY	Method	☑	POST
3	PRIMARYURL	URL	☑	http://172.30.172.39:5576

3. Select the **Connectors** tab.

   a. Change the Connector ID to **HTTPTARGET.**

   b. Type a value for each property based on the following:

Property ID	Property Name	Value
HEADER	sendUncompressed	Y
HTTPPROPERTY	Method	POST
PRIMARYURL	URL	URL and the port of the HTTP listener

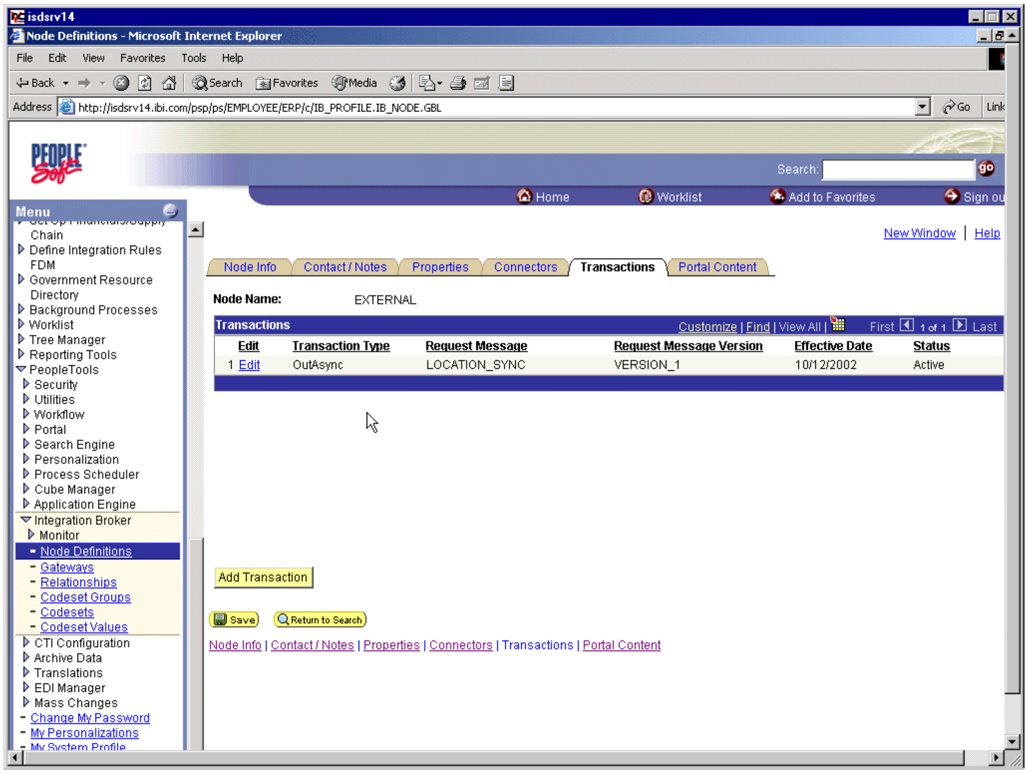
Note: For complex business situations you can configure multiple nodes and multiple listeners.

4. Click **Save.**

5. If you are warned that you are changing the Connector, click **OK.**

6. Select the **Transactions** tab.

   The following pane opens.



Node Name: EXTERNAL

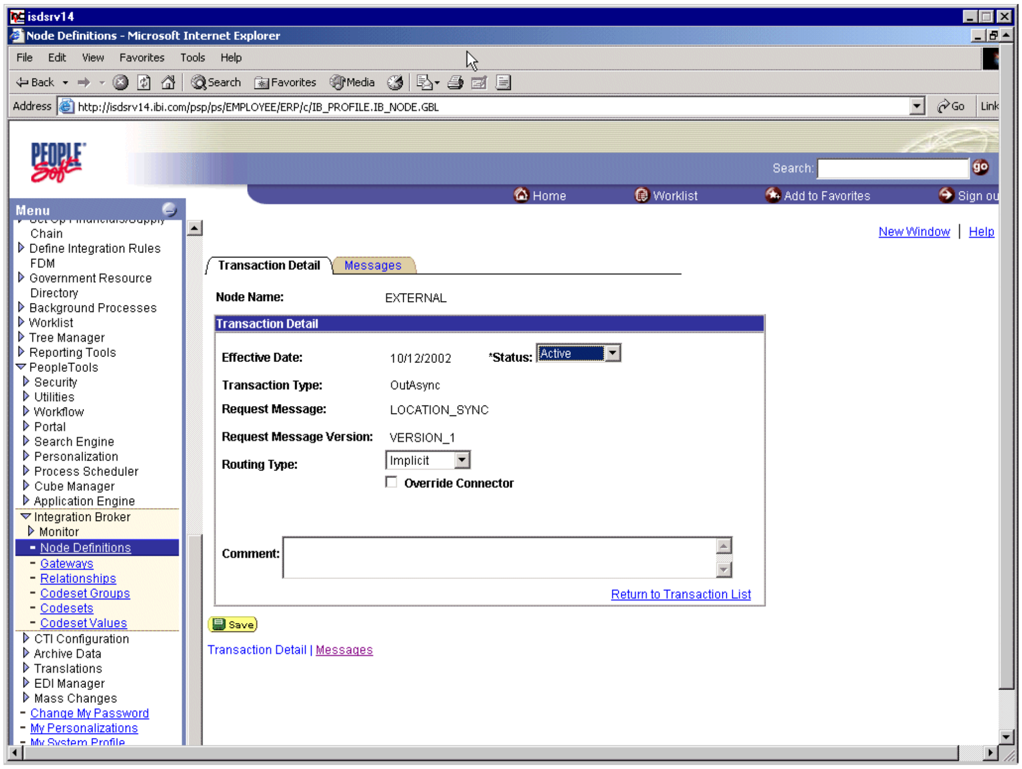
Transactions

	Edit	Transaction Type	Request Message	Request Message Version	Effective Date	Status
1	Edit	OutAsync	LOCATION_SYNC	VERSION_1	10/12/2002	Active

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.



Node Name: EXTERNAL

Transaction Detail

Effective Date: 10/12/2002 *Status: Active

Transaction Type: OutAsync

Request Message: LOCATION_SYNC

Request Message Version: VERSION_1

Routing Type: Implicit

□ Override Connector

Comment:

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.

   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 are ready to send XML messages to your PeopleSoft XML listener.

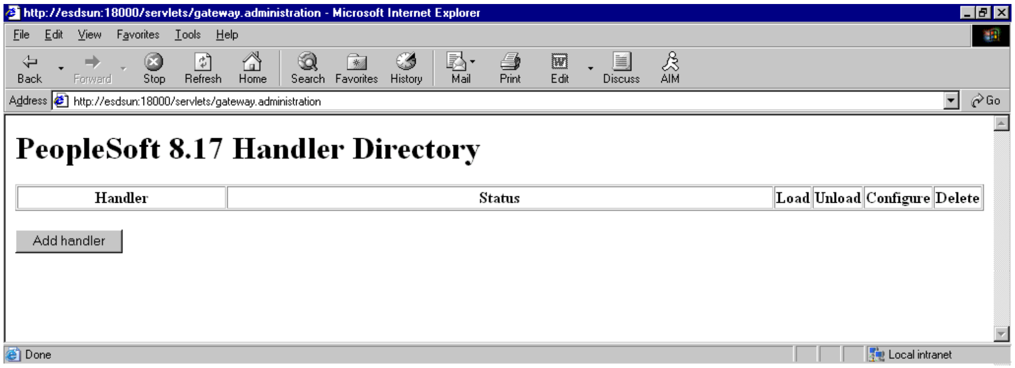
Configuring the TCP/IP Handler for PeopleSoft 8.1

The following procedure assumes that your Application Messaging environment is properly configured and tested. See Appendix D, "Using PeopleSoft 8 Integration Broker" for more information.

To configure the TCP/IP Handler for PeopleSoft 8.1 to send messages to Oracle Application 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.

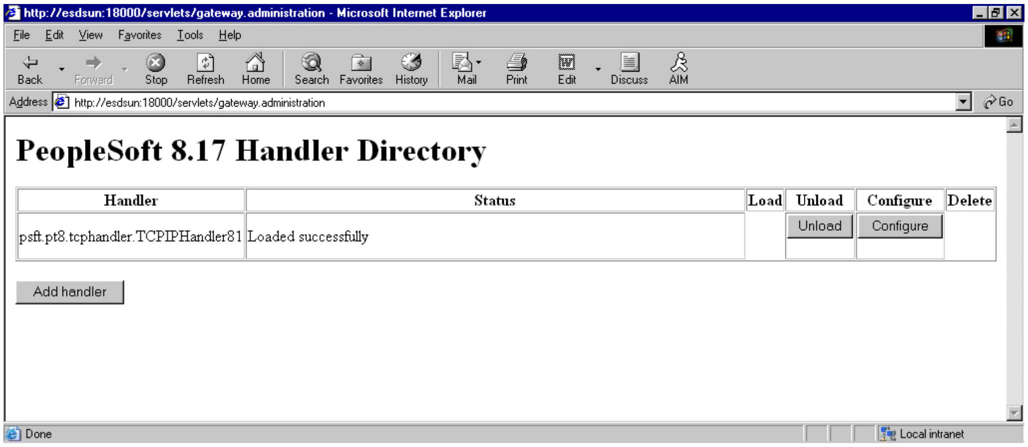


Address http://esdsun:18000/servlets/gateway.administration

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete

3. Click **Add handler.**


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete
psft.pt8.tcphandler.TCPIPHandler81	Loaded successfully		Unload	Configure	

Add handler

a. Type the full path of TCPIPHandler81 (case-sensitive):

```
psft.pt8.tcphandler.TCPIPHandler81
```

b. Click **Save.**

4. Click **Load.**

   The PeopleSoft Handler Directory window opens.

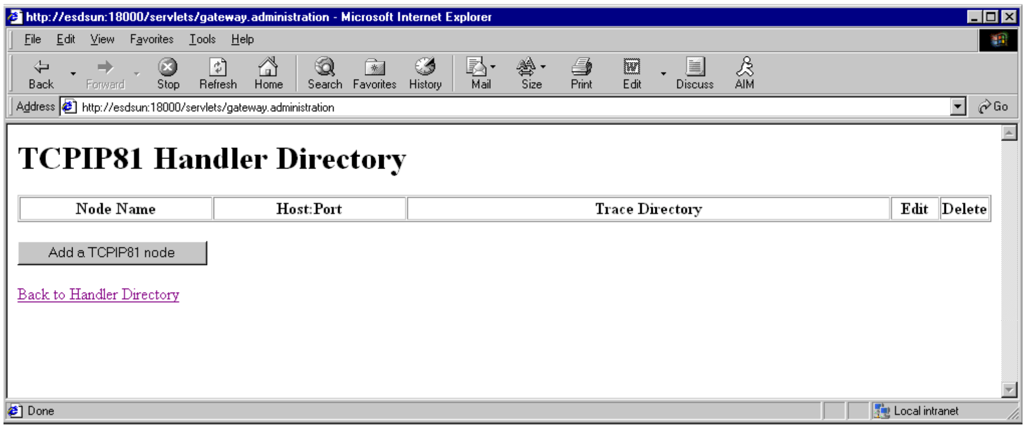
5. Click **Configure.**

   The TCPIP81 Handler Directory window opens.

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 8 Integration Broker" or your PeopleSoft documentation.

   The Add TCPIP81 Handler window opens.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

TCPIP81 Handler Directory

Node Name	Host:Port	Trace Directory	Edit	Delete

Add a TCPIP81 node

Back to Handler Directory

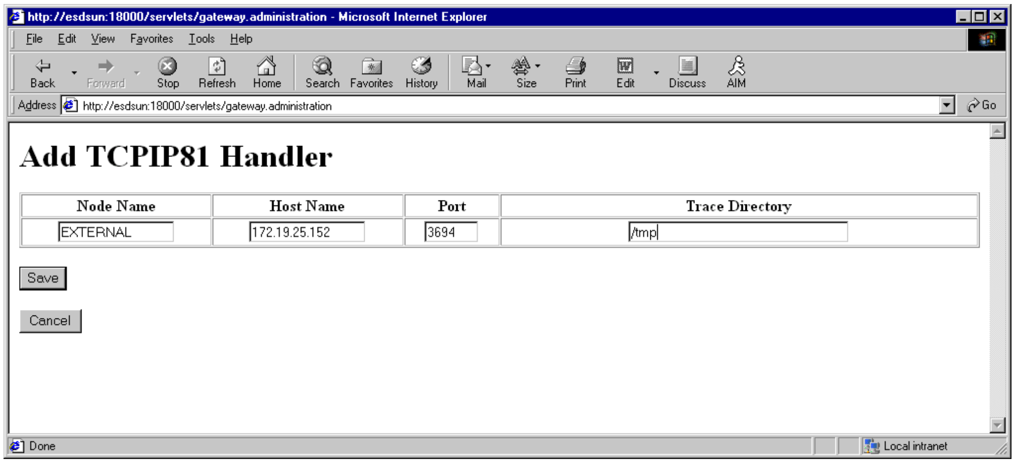
7. Enter the requested values based on the information in the following table.

Field	Value Example	Description
Node Name	EXTERNAL	The name of the TCP/IP node.
Host Name	172.19.25.152	The machine 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 TCPIP81 Handler Directory window opens.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Save
Cancel
Done
Local intranet

Add TCPIP81 Handler

Node Name	Host Name	Port	Trace Directory
EXTERNAL	172.19.25.152	3694	/tmp

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 are now ready to send messages from PeopleSoft to your OracleAS Adapter for PeopleSoft.

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 Application Server is up and running.

- The server name and port number for PeopleSoft and Oracle Application 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:

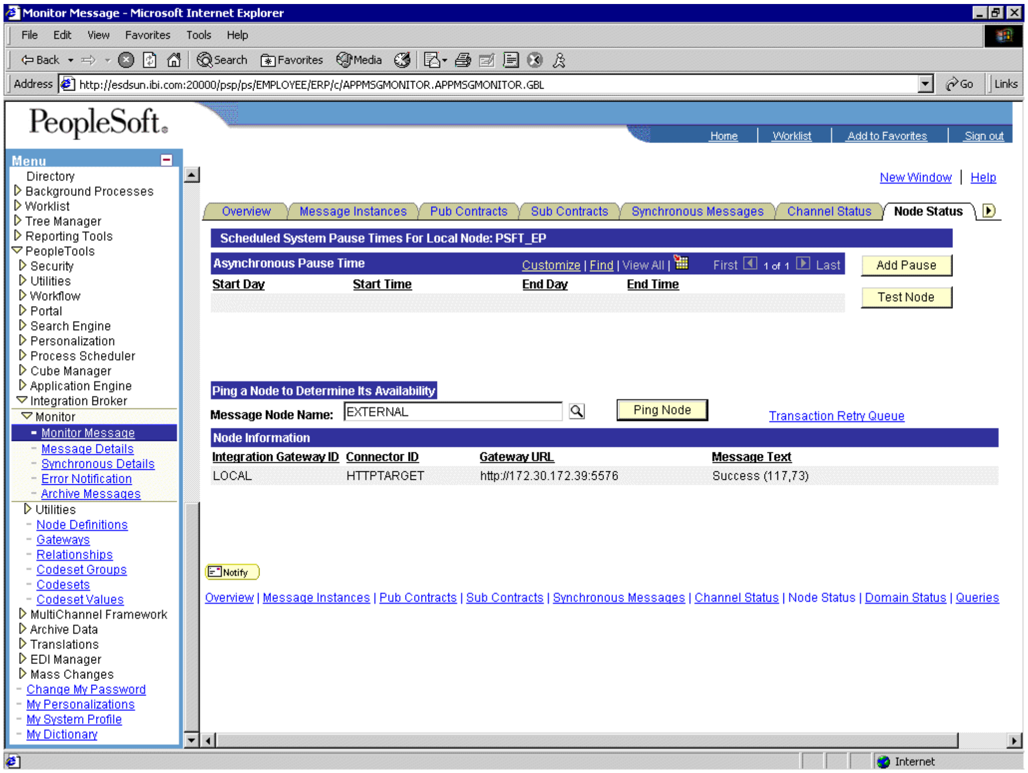
   a. In the menu pane, expand **PeopleTools, Integration Broker,** and **Monitor.**

   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.



Scheduled System Pause Times For Local Node: PSFT_EP

Asynchronous Pause Time

Start Day	Start Time	End Day	End Time

Ping a Node to Determine Its Availability

Message Node Name: EXTERNAL

Node Information

Integration Gateway ID	Connector ID	Gateway URL	Message Text
LOCAL	HTTPTARGET	http://172.30.172.39:5576	Success (117,73)

a. From the Message Node Name list, select your node.

b. Click **Ping Node.**

If you properly configured both PeopleSoft and Oracle Application Server, you receive a Success message.

An error indicates a configuration problem. For more information, see the Integration Broker error log.

C

Using Component Interfaces

This appendix describes how to create new component interfaces—and how to modify existing component interfaces—for use with the OracleAS 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.

Creating a Component Interface

You create component interfaces using the PeopleSoft Application Designer. For more information about Application Designer, see your PeopleSoft documentation.

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 type 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.

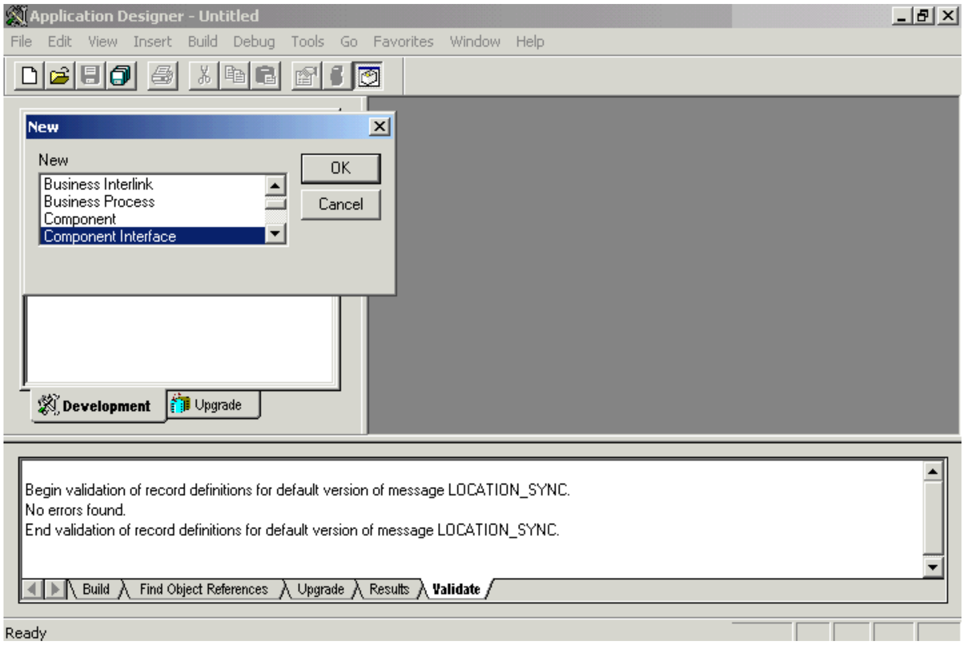
Properties may have various icons adjacent to them. For example, EMPLID has an icon indicating that it is a key field from the underlying record. NAME has an icon indicating that it is an alternate key field from the underlying record. For a complete list of property icons, see the PeopleBooks documentation.

Creating a New Component Interface

To create a component interface:

1. Open the PeopleSoft Application Designer.
2. Select **New** from the **File** menu.

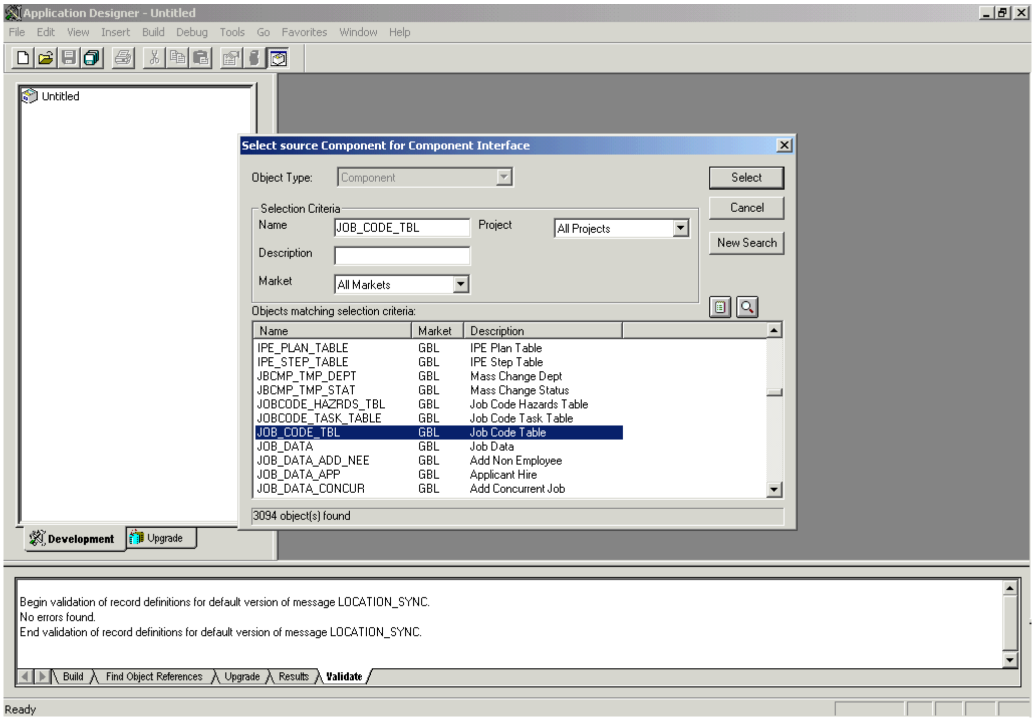
   The New dialog box opens.



```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

   a. Select **Component Interface.**
   b. Click **OK.**

      The Select Source Component for Component Interface dialog box opens.



Select source Component for Component Interface

Object Type: Component

Selection Criteria

Name	JOB_CODE_TBL	Project	All Projects
Description			
Market	All Markets		

Objects matching selection criteria:

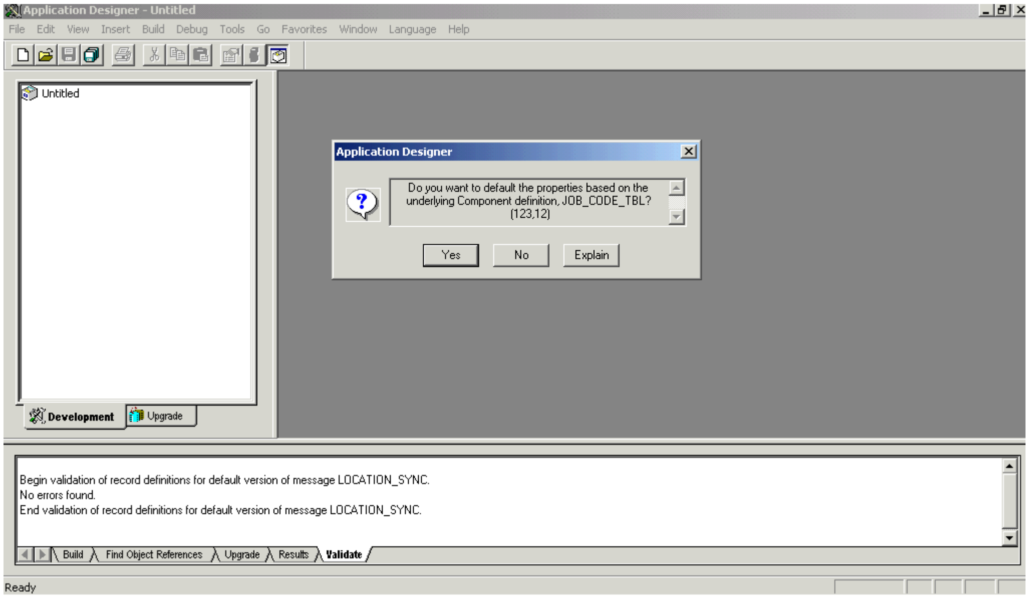
Name	Market	Description
IPE_PLAN_TABLE	GBL	IPE Plan Table
IPE_STEP_TABLE	GBL	IPE Step Table
JBCMP_TMP_DEPT	GBL	Mass Change Dept
JBCMP_TMP_STAT	GBL	Mass Change Status
JOBCODE_HAZRDS_TBL	GBL	Job Code Hazards Table
JOBCODE_TASK_TABLE	GBL	Job Code Task Table
JOB_CODE_TBL	GBL	Job Code Table
JOB_DATA	GBL	Job Data
JOB_DATA_ADD_NEE	GBL	Add Non Employee
JOB_DATA_APP	GBL	Applicant Hire
JOB_DATA_CONCUR	GBL	Add Concurrent Job

3094 object(s) found

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

3. Highlight the component to use as a basis for the component interface and click **Select.**

   The Application Designer dialog box opens.



Do you want to default the properties based on the underlying Component definition, JOB_CODE_TBL? (123,12)

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

Note: If the component interface is large, expose the component properties manually.

---

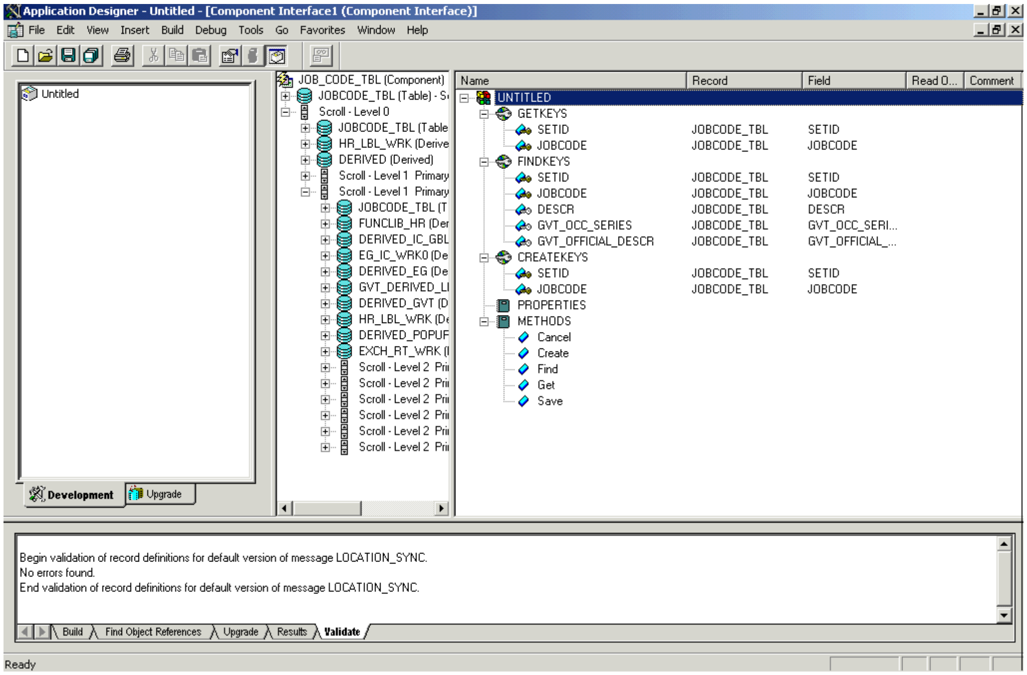
4. To create the component interface without displaying properties and to expose component properties manually, click **No.**

   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.**



Name	Record	Field	Read O...	Comment
UNTITLED				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
DESCR	JOBCODE_TBL	DESCR		
GVT_OCC_SERIES	JOBCODE_TBL	GVT_OCC_SERI...		
GVT_OFFICIAL_DESCR	JOBCODE_TBL	GVT_OFFICIAL_...		
CREATEKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
PROPERTIES				
METHODS				
Cancel				
Create				
Find				
Get				
Save				

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

Standard Methods

The standard methods for the component interface are:

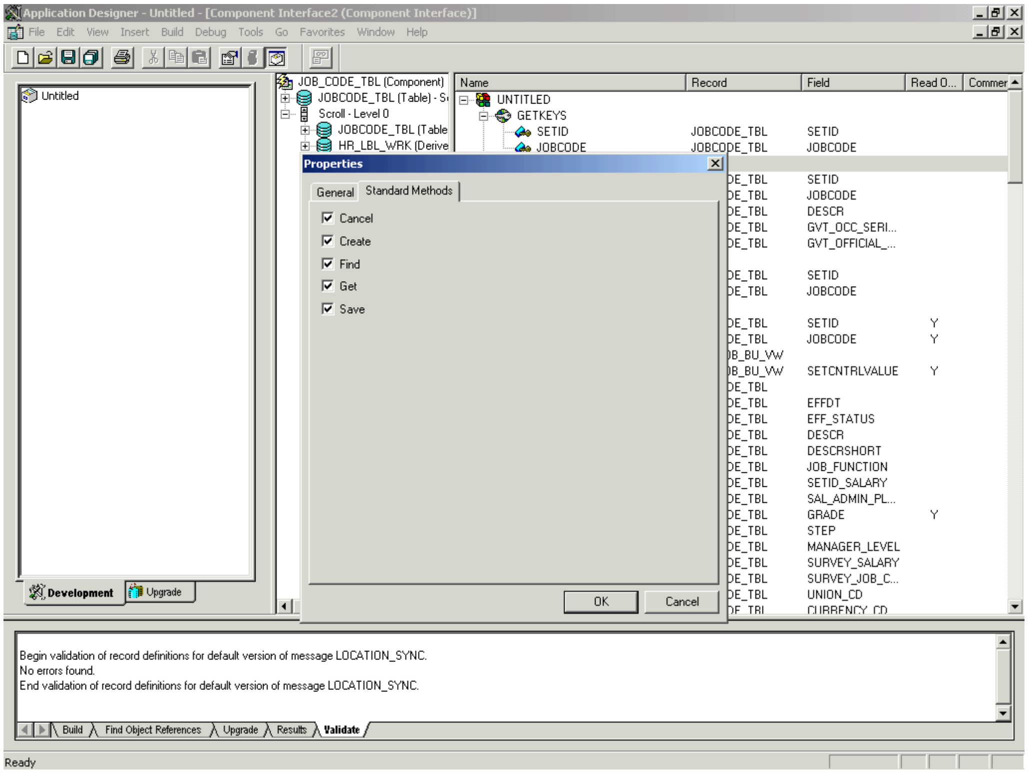
- Create
- Find
- Get
- Save

Only those methods in the underlying component are available. For example, if the underlying component does not contain Add capabilities, Create is not available.

Viewing or Changing Available Methods

To view or change available methods:

1. Open the Component Interface Properties dialog box.



☑ Cancel
☑ Create
☑ Find
☑ Get
☑ Save

Name	Record	Field	Read O...	Commer
UNTITLED				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
	DE_TBL	SETID		
	DE_TBL	JOBCODE		
	DE_TBL	DESCR		
	DE_TBL	GVT_OCC_SERI...		
	DE_TBL	GVT_OFFICIAL_...		
	DE_TBL	SETID		
	DE_TBL	JOBCODE		
	DE_TBL	SETID	Y	
	DE_TBL	JOBCODE	Y	
	)B_BU_VW			
	)B_BU_VW	SETCNTRLVALUE	Y	
	DE_TBL			
	DE_TBL	EFFDT		
	DE_TBL	EFF_STATUS		
	DE_TBL	DESCR		
	DE_TBL	DESCRSHORT		
	DE_TBL	JOB_FUNCTION		
	DE_TBL	SETID_SALARY		
	DE_TBL	SAL_ADMIN_PL...		
	DE_TBL	GRADE	Y	
	DE_TBL	STEP		
	DE_TBL	MANAGER_LEVEL		
	DE_TBL	SURVEY_SALARY		
	DE_TBL	SURVEY_JOB_C...		
	DE_TBL	UNION_CD		
	DE_TBL	CURRENCY_CD		

```
Begin validation of record definitions for default version of message LOCATION_SYNC.
No errors found.
End validation of record definitions for default version of message LOCATION_SYNC.
```

2. Click the **Standard Methods** tab.
3. Select the desired methods.

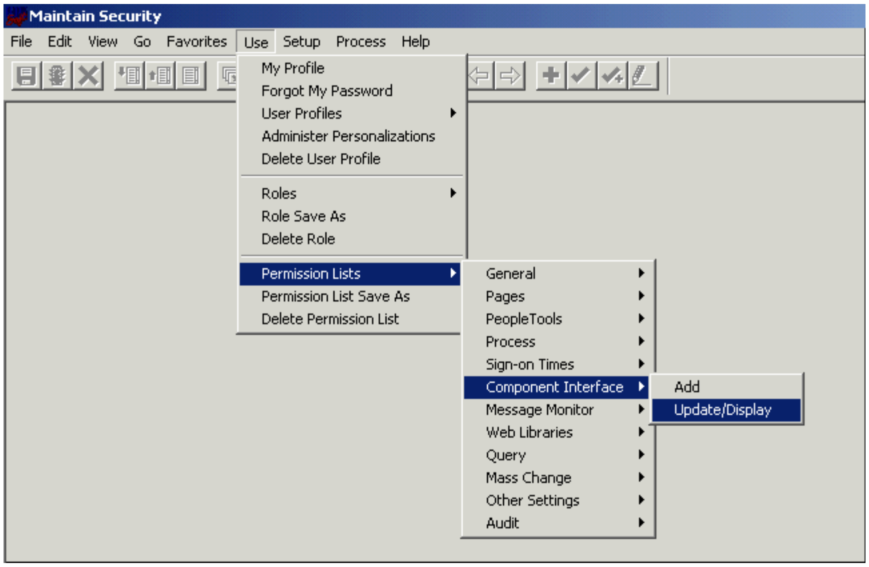
Securing a Component Interface

You must set up security for the component interface before you can begin testing.

Configuring Component Interface Security for PeopleSoft Version 8.1x

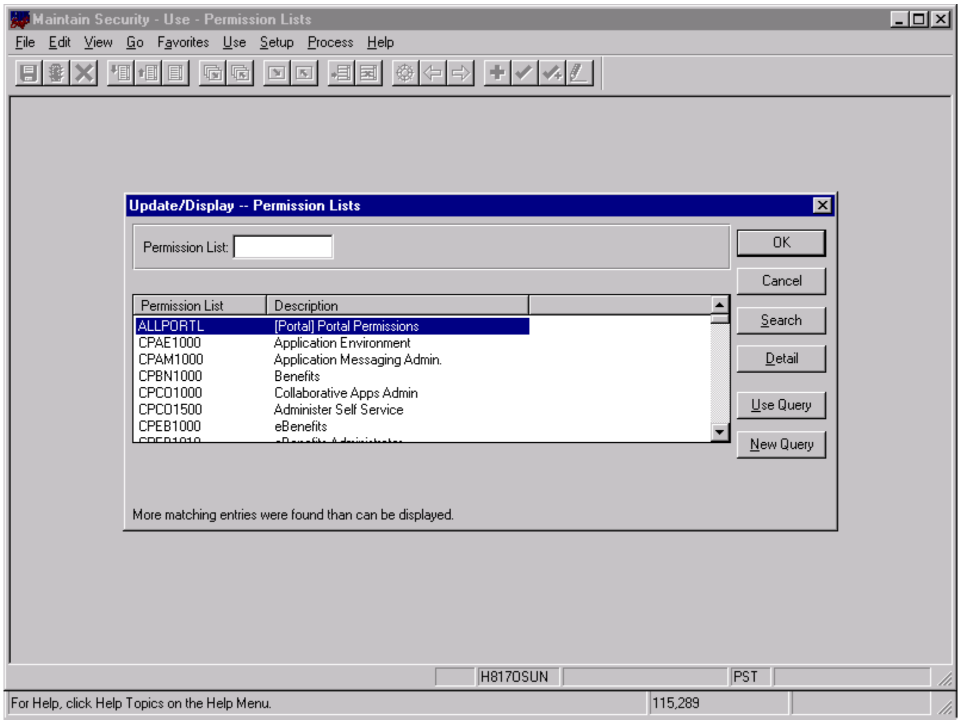
The following procedure describes how to configure component interface security for PeopleSoft Version 8.1 in 2- and 3-tier mode.

To configure component interface security:



1. From the **Use** menu, select **Permission Lists, Component Interface,** and then click **Update/Display.**

   The Permission Lists dialog box opens.


Maintain Security - Use - Permission Lists
File Edit View Go Favorites Use Setup Process Help
Update/Display -- Permission Lists
OK
Cancel
Search
Detail
Use Query
New Query
H817OSUN
PST
115,289

Permission List:

Permission List	Description
ALLPORTL	[Portal] Portal Permissions
CPAE1000	Application Environment
CPAM1000	Application Messaging Admin.
CPBN1000	Benefits
CPCO1000	Collaborative Apps Admin
CPCO1500	Administer Self Service
CPEB1000	eBenefits

More matching entries were found than can be displayed.

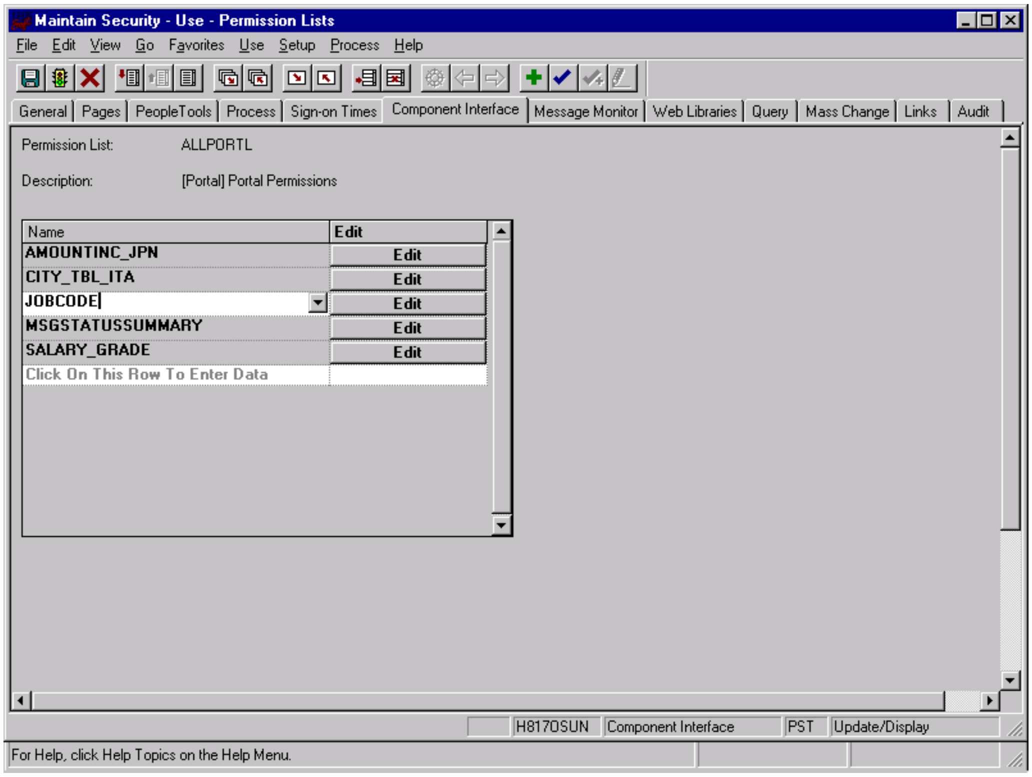
For Help, click Help Topics on the Help Menu.

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 opens.



Permission List: ALLPORTL

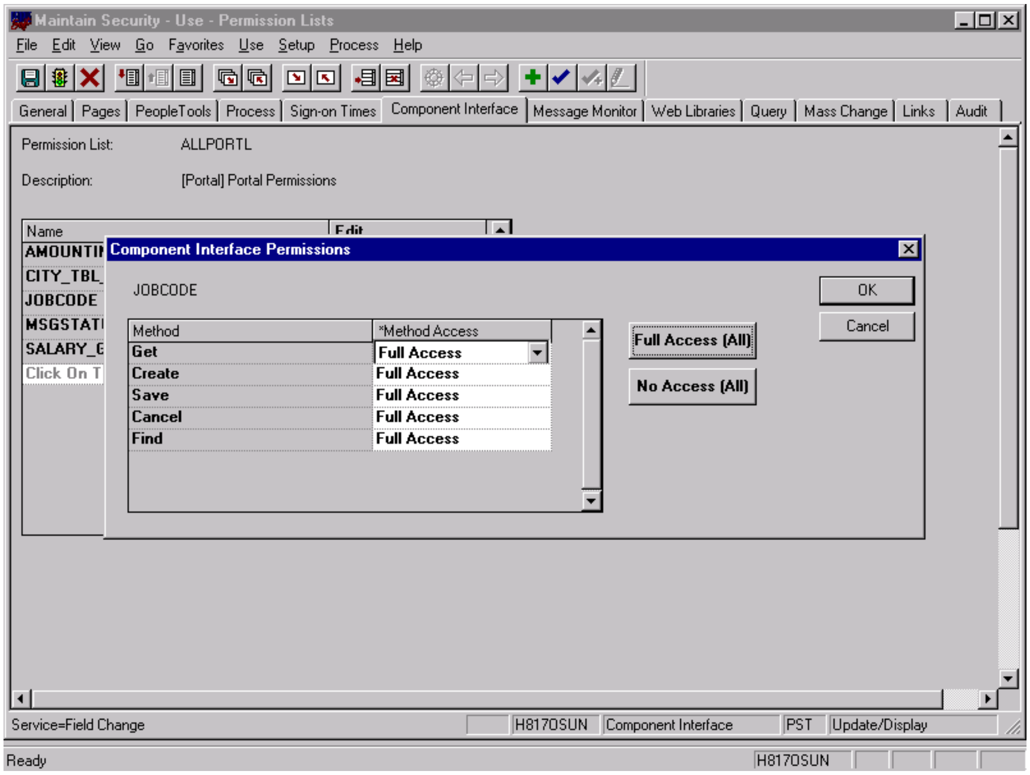
Description: [Portal] Portal Permissions

Name	Edit
AMOUNTINC_JPN	Edit
CITY_TBL_ITA	Edit
JOBCODE	Edit
MSGSTATUSSUMMARY	Edit
SALARY_GRADE	Edit
Click On This Row To Enter Data	

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. This enables you to 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.


Maintain Security - Use - Permission Lists
File Edit View Go Favorites Use Setup Process Help
General Pages PeopleTools Process Sign-on Times Component Interface Message Monitor Web Libraries Query Mass Change Links Audit
Component Interface Permissions
OK
Cancel
Full Access (All)
No Access (All)
Ready

Permission List: ALLPORTL

Description: [Portal] Portal Permissions

Name	Edit
AMOUNTI	
CITY_TBL_	
JOBCODE	
MSGSTAT	
SALARY_	
Click On T	

JOBCODE

Method	*Method Access
Get	Full Access
Create	Full Access
Save	Full Access
Cancel	Full Access
Find	Full Access

Service=Field Change H8170SUN Component Interface PST Update/Display

Ready H8170SUN

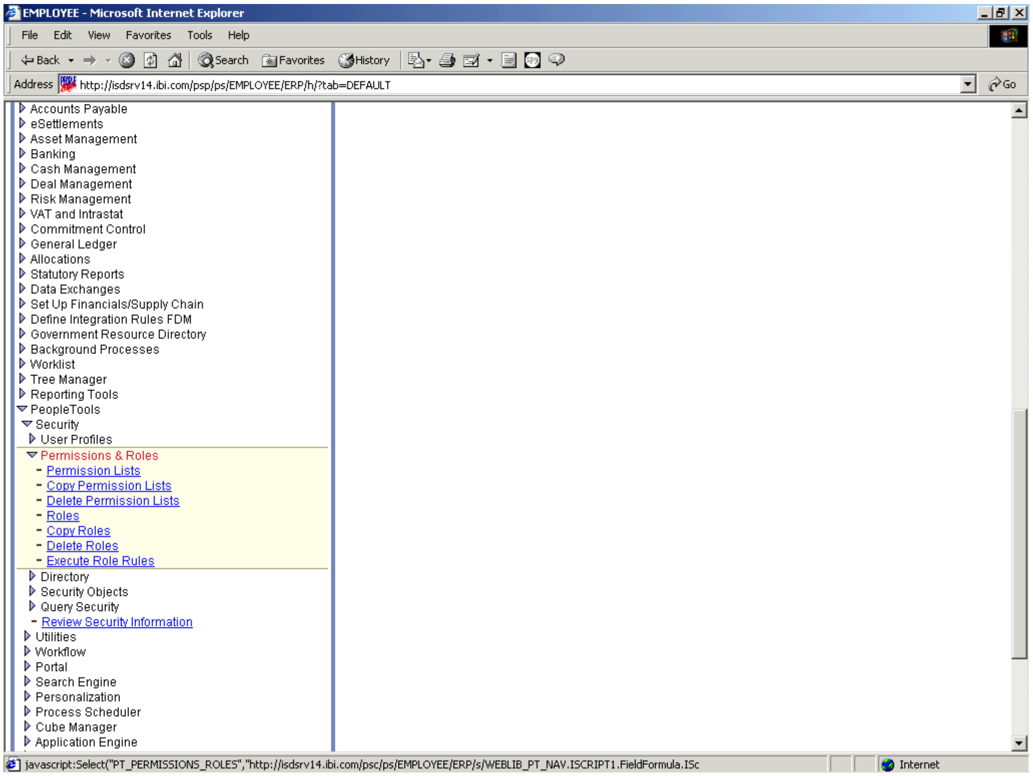
5. Select the desired level of access.

6. Click **OK.**

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:



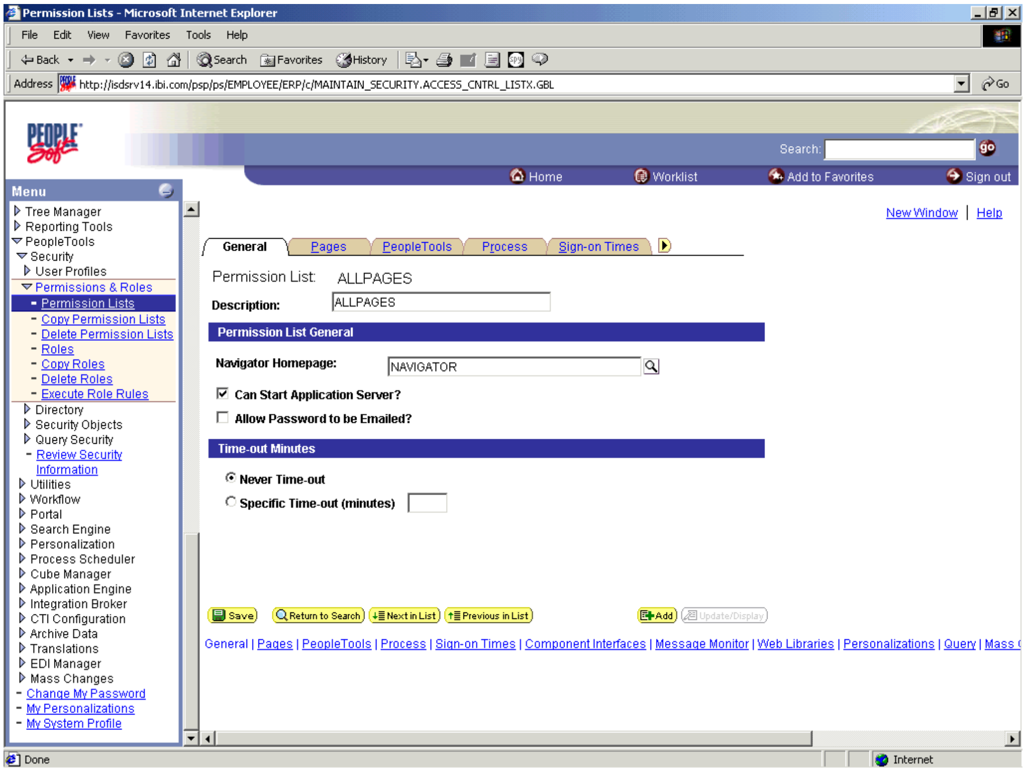
1. Expand **PeopleTools, Security, User Profiles,** and **Permissions & Roles** and then click **Permission Lists.**

2. Click **Search.**

   The Permission Lists Search pane opens.

3. Select the relevant permission list.

   The following pane opens.


Permission Lists - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address http://isdsrv14.ibi.com/psp/ps/EMPLOYEE/ERP/c/MAINTAIN_SECURITY.ACCESS_CNTRL_LISTX.GBL
Go
PEOPLE Soft
Search: go
Home Worklist Add to Favorites Sign out
Menu
Tree Manager
Reporting Tools
PeopleTools
Security
User Profiles
Permissions & Roles
Permission Lists
Copy Permission Lists
Delete Permission Lists
Roles
Copy Roles
Delete Roles
Execute Role Rules
Directory
Security Objects
Query Security
Review Security Information
Utilities
Workflow
Portal
Search Engine
Personalization
Process Scheduler
Cube Manager
Application Engine
Integration Broker
CTI Configuration
Archive Data
Translations
EDI Manager
Mass Changes
Change My Password
My Personalizations
My System Profile
New Window | Help
General Pages PeopleTools Process Sign-on Times
Save Return to Search Next in List Previous in List Add Update/Display
General | Pages | PeopleTools | Process | Sign-on Times | Component Interfaces | Message Monitor | Web Libraries | Personalizations | Query | Mass
Done
Internet

Permission List: ALLPAGES

Description: ALLPAGES

Permission List General

Navigator Homepage: NAVIGATOR

☑ Can Start Application Server?

☐ Allow Password to be Emailed?

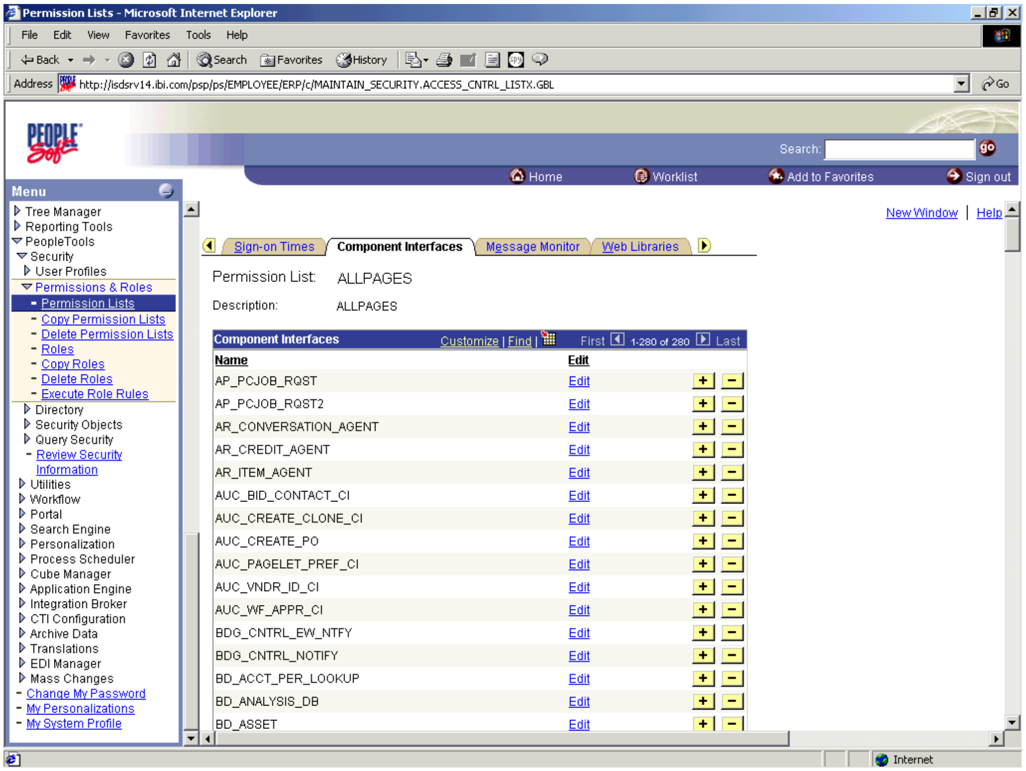
Time-out Minutes

● Never Time-out

○ Specific Time-out (minutes)

4. Click the right arrow next to the **Sign-on Times** tab.

   The **Component Interfaces** tab appears.



Permission List: ALLPAGES

Description: ALLPAGES

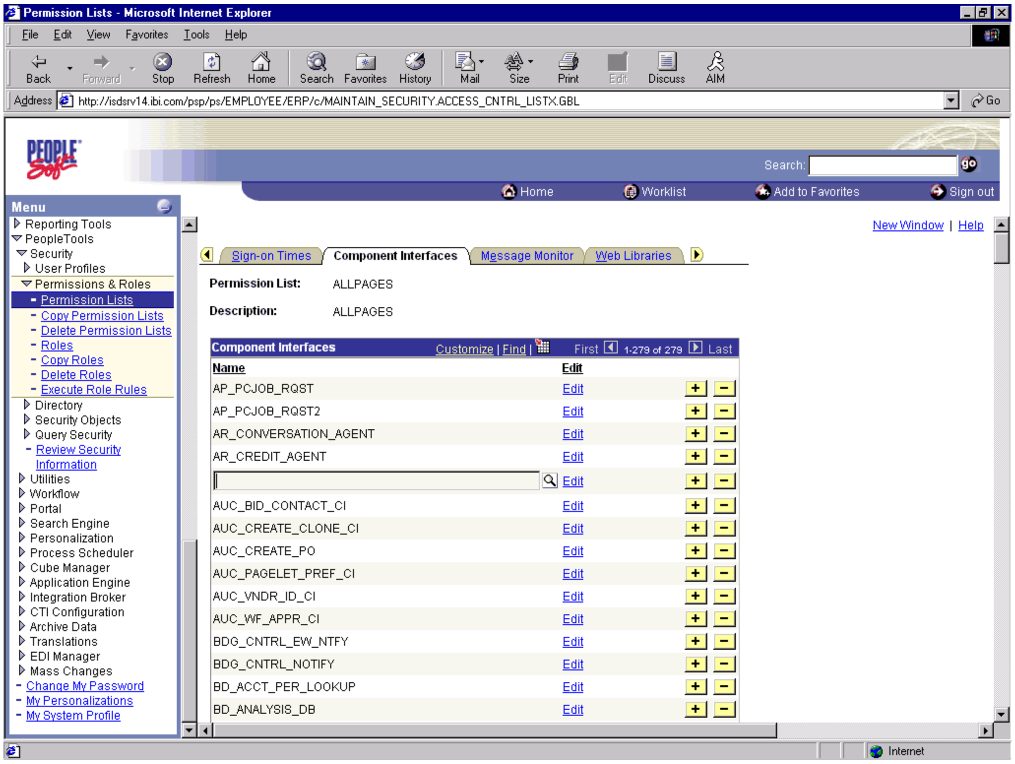
Component Interfaces — Customize | Find | First 1-280 of 280 Last

Name	Edit
AP_PCJOB_RQST	Edit
AP_PCJOB_RQST2	Edit
AR_CONVERSATION_AGENT	Edit
AR_CREDIT_AGENT	Edit
AR_ITEM_AGENT	Edit
AUC_BID_CONTACT_CI	Edit
AUC_CREATE_CLONE_CI	Edit
AUC_CREATE_PO	Edit
AUC_PAGELET_PREF_CI	Edit
AUC_VNDR_ID_CI	Edit
AUC_WF_APPR_CI	Edit
BDG_CNTRL_EW_NTFY	Edit
BDG_CNTRL_NOTIFY	Edit
BD_ACCT_PER_LOOKUP	Edit
BD_ANALYSIS_DB	Edit
BD_ASSET	Edit

5. Click the **Component Interfaces** tab.

6. Click the **+** button to add a new row to the Component Interfaces list.

   A field appears where you can type the component interface name.



Permission List: ALLPAGES

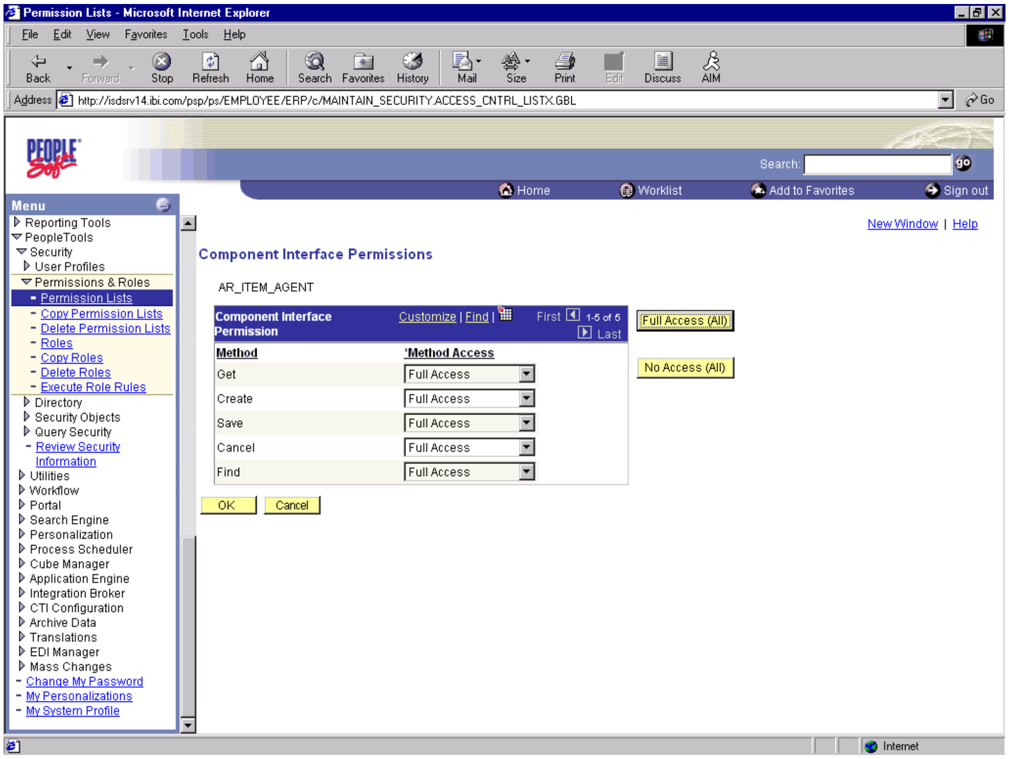
Description: ALLPAGES

Component Interfaces — Customize | Find | First 1-279 of 279 Last

Name	Edit		
AP_PCJOB_RQST	Edit	+	−
AP_PCJOB_RQST2	Edit	+	−
AR_CONVERSATION_AGENT	Edit	+	−
AR_CREDIT_AGENT	Edit	+	−
	Edit	+	−
AUC_BID_CONTACT_CI	Edit	+	−
AUC_CREATE_CLONE_CI	Edit	+	−
AUC_CREATE_PO	Edit	+	−
AUC_PAGELET_PREF_CI	Edit	+	−
AUC_VNDR_ID_CI	Edit	+	−
AUC_WF_APPR_CI	Edit	+	−
BDG_CNTRL_EW_NTFY	Edit	+	−
BDG_CNTRL_NOTIFY	Edit	+	−
BD_ACCT_PER_LOOKUP	Edit	+	−
BD_ANALYSIS_DB	Edit	+	−

7. Type the component interface name and click **Edit.**

   This example uses the component interface AR_ITEM_AGENT.



Component Interface Permissions

AR_ITEM_AGENT

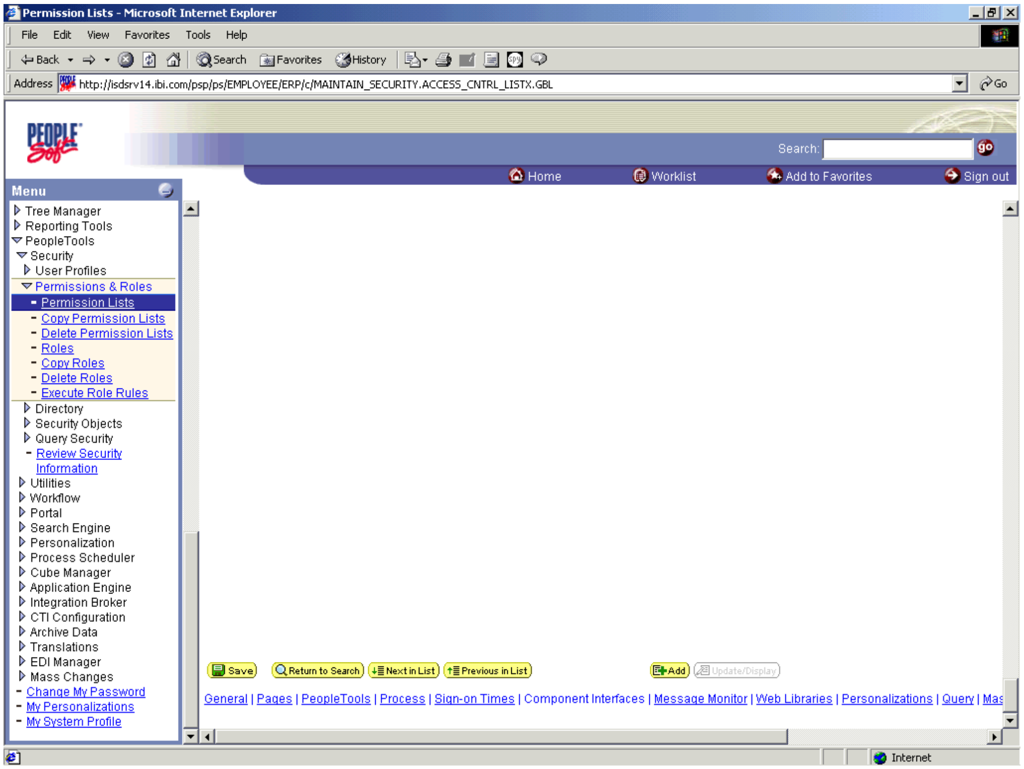
Component Interface Permission — Customize | Find | First 1-5 of 5 Last

Method	*Method Access
Get	Full Access
Create	Full Access
Save	Full Access
Cancel	Full Access
Find	Full Access

8. From the lists, select the desired access level for each method.

9. Click **OK.**

   The following pane opens.



10. Scroll down in the right pane and click **Save.**

Testing a Component Interface

The OracleAS 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.

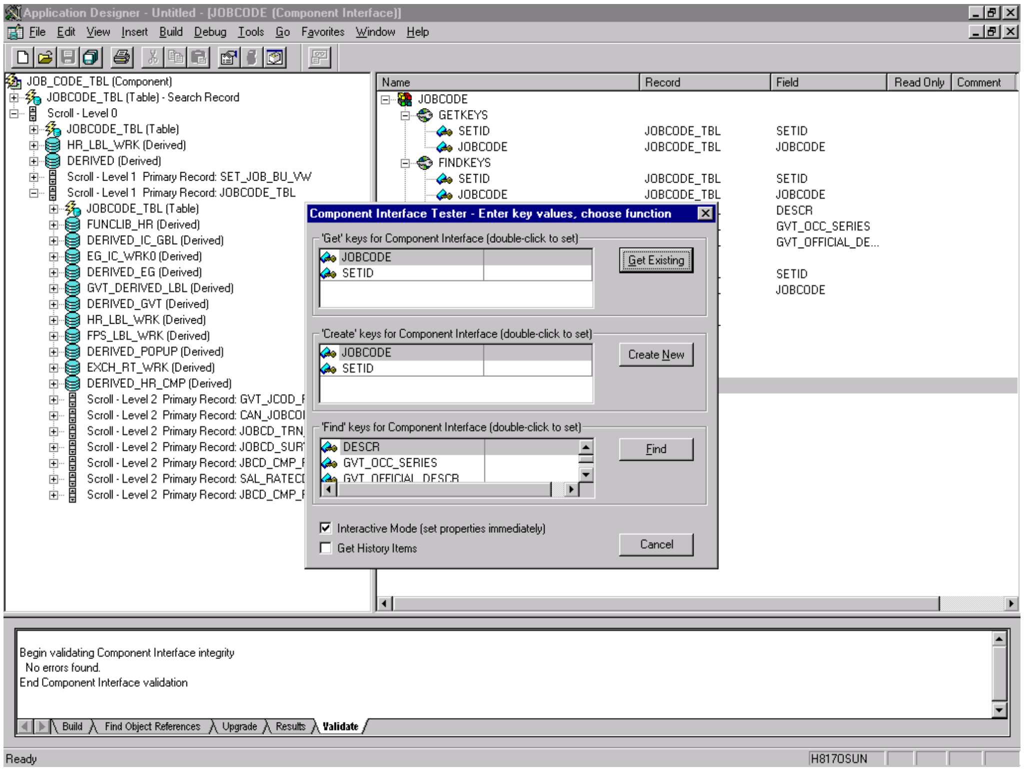
If changes are made to the underlying application by the user or by a PeopleSoft upgrade and the changes invalidate a component interface, the user must repair the invalid component interface before the adapter uses it.

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 box opens.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_BU_VW
Scroll - Level 1 Primary Record: JOBCODE_TBL
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JCOD_F
Scroll - Level 2 Primary Record: CAN_JOBCOI
Scroll - Level 2 Primary Record: JOBCD_TRN
Scroll - Level 2 Primary Record: JOBCD_SUR
Scroll - Level 2 Primary Record: JBCD_CMP_F
Scroll - Level 2 Primary Record: SAL_RATECI
Scroll - Level 2 Primary Record: JBCD_CMP_F
Component Interface Tester - Enter key values, choose function
'Get' keys for Component Interface (double-click to set)
JOBCODE
SETID
Get Existing
'Create' keys for Component Interface (double-click to set)
JOBCODE
SETID
Create New
'Find' keys for Component Interface (double-click to set)
DESCR
GVT_OCC_SERIES
GVT_OFFICIAL_DESCR
Find
☑ Interactive Mode (set properties immediately)
□ Get History Items
Cancel
Build
Find Object References
Upgrade
Results
Validate
Ready
H8170SUN

Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
		DESCR		
		GVT_OCC_SERIES		
		GVT_OFFICIAL_DE...		
		SETID		
		JOBCODE		

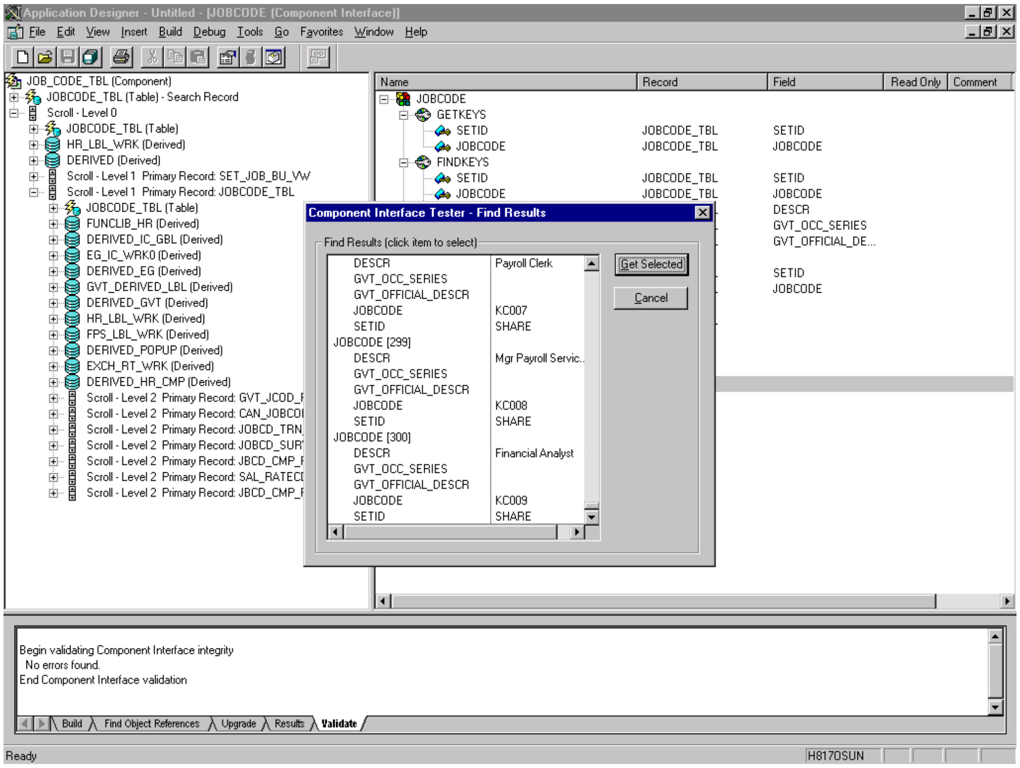
```
Begin validating Component Interface integrity
  No errors found.
End Component Interface validation
```

2. If required, click the Component Interface Tester dialog box 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 box opens, displaying all of the possible entries for the underlying component. If there are more than 300 entries, a message appears.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET_JOB_BU_VW
Scroll - Level 1 Primary Record: JOBCODE_TBL
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record: GVT_JCOD_I
Scroll - Level 2 Primary Record: CAN_JOBCOI
Scroll - Level 2 Primary Record: JOBCD_TRN
Scroll - Level 2 Primary Record: JOBCD_SUR
Scroll - Level 2 Primary Record: JBCD_CMP_I
Scroll - Level 2 Primary Record: SAL_RATECI
Scroll - Level 2 Primary Record: JBCD_CMP_I
Component Interface Tester - Find Results
Find Results (click item to select)
Get Selected
Cancel
Build
Find Object References
Upgrade
Results
Validate
Ready
H817OSUN

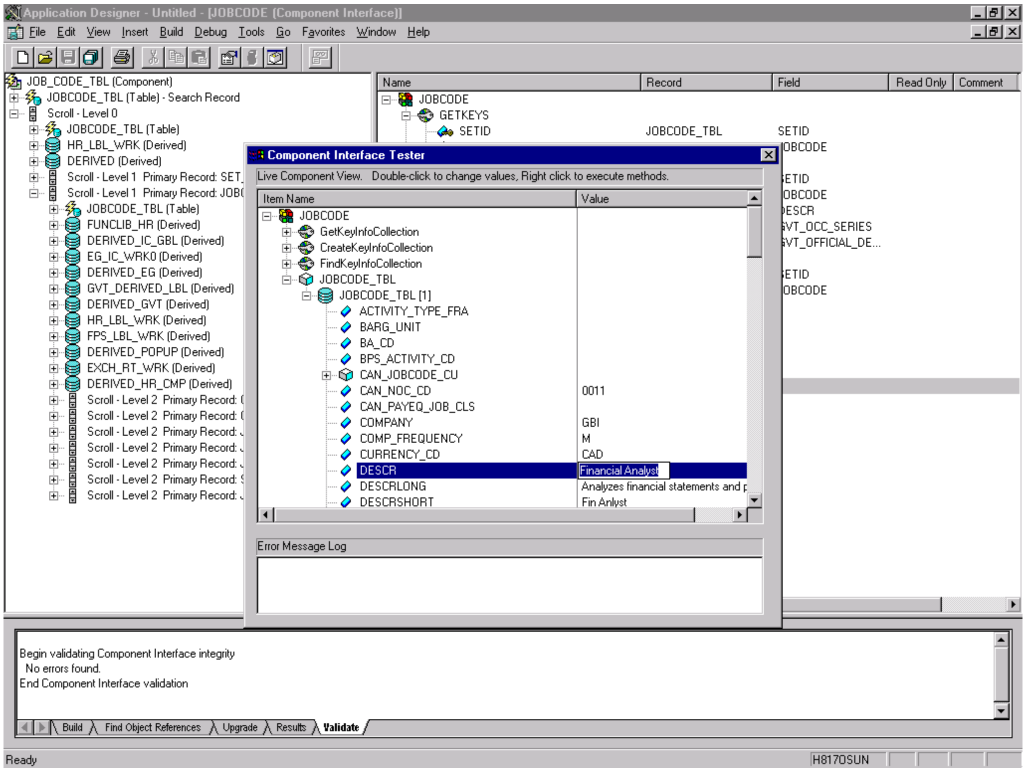
Name	Record	Field	Read Only	Comment
JOBCODE				
GETKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
FINDKEYS				
SETID	JOBCODE_TBL	SETID		
JOBCODE	JOBCODE_TBL	JOBCODE		
		DESCR		
		GVT_OCC_SERIES		
		GVT_OFFICIAL_DE...		
		SETID		
		JOBCODE		

DESCR	Payroll Clerk
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	
JOBCODE	KC007
SETID	SHARE
JOBCODE [299]	
DESCR	Mgr Payroll Servic..
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	
JOBCODE	KC008
SETID	SHARE
JOBCODE [300]	
DESCR	Financial Analyst
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	
JOBCODE	KC009
SETID	SHARE

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

- In the left pane of the Find Results dialog box, select a field.
- To display the relevant data for that particular field, click **Get Selected.**

  The following dialog box opens.



Item Name	Value
JOBCODE	
GetKeyInfoCollection	
CreateKeyInfoCollection	
FindKeyInfoCollection	
JOBCODE_TBL	
JOBCODE_TBL [1]	
ACTIVITY_TYPE_FRA	
BARG_UNIT	
BA_CD	
BPS_ACTIVITY_CD	
CAN_JOBCODE_CU	
CAN_NOC_CD	0011
CAN_PAYEQ_JOB_CLS	
COMPANY	GBI
COMP_FREQUENCY	M
CURRENCY_CD	CAD
DESCR	Financial Analyst
DESCRLONG	Analyzes financial statements and p
DESCRSHORT	Fin Anlyst

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

- If the security settings permit, you can change the values in the individual fields.

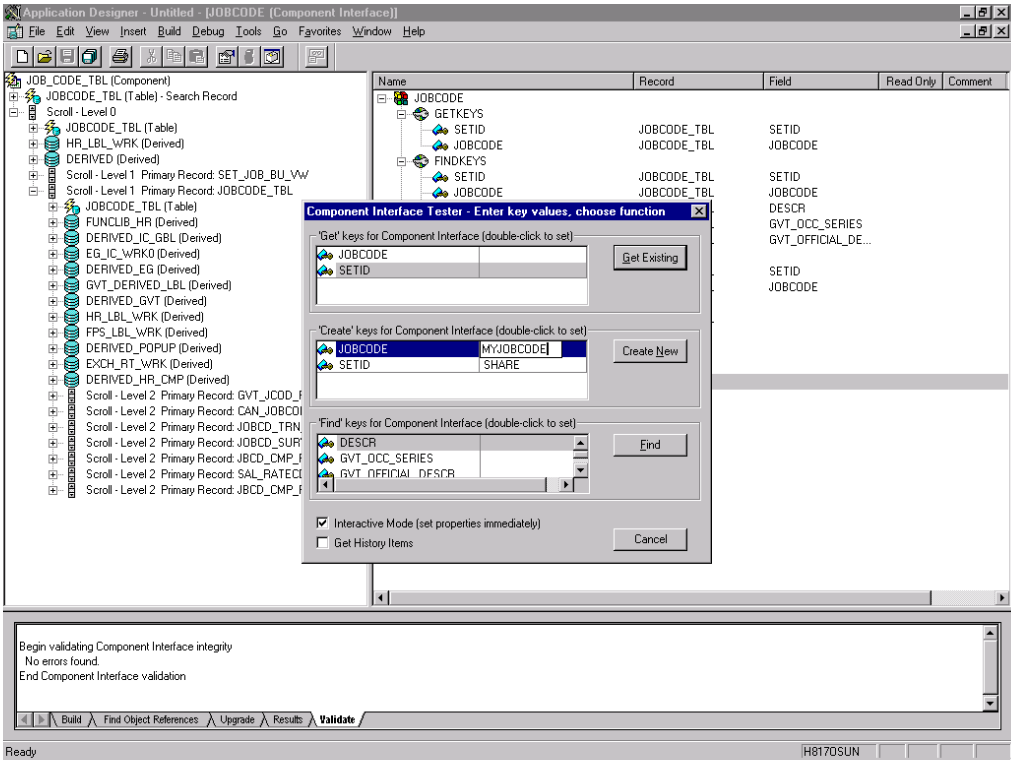
4. To test the component interface using the Get method:

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:


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Name
Record
Field
Read Only
Comment
Build
Find Object References
Upgrade
Results
Validate
Ready
H8170SUN

```
JOB_CODE_TBL (Component)
  JOBCODE_TBL (Table) - Search Record
  Scroll - Level 0
    JOBCODE_TBL (Table)
    HR_LBL_WRK (Derived)
    DERIVED (Derived)
    Scroll - Level 1  Primary Record: SET_JOB_BU_VW
    Scroll - Level 1  Primary Record: JOBCODE_TBL
      JOBCODE_TBL (Table)
      FUNCLIB_HR (Derived)
      DERIVED_IC_GBL (Derived)
      EG_IC_WRK0 (Derived)
      DERIVED_EG (Derived)
      GVT_DERIVED_LBL (Derived)
      DERIVED_GVT (Derived)
      HR_LBL_WRK (Derived)
      FPS_LBL_WRK (Derived)
      DERIVED_POPUP (Derived)
      EXCH_RT_WRK (Derived)
      DERIVED_HR_CMP (Derived)
      Scroll - Level 2  Primary Record: GVT_JCOD_F
      Scroll - Level 2  Primary Record: CAN_JOBCOI
      Scroll - Level 2  Primary Record: JOBCD_TRN
      Scroll - Level 2  Primary Record: JOBCD_SUR
      Scroll - Level 2  Primary Record: JBCD_CMP_F
      Scroll - Level 2  Primary Record: SAL_RATECI
      Scroll - Level 2  Primary Record: JBCD_CMP_F
```

```
Name                    Record          Field
JOBCODE
  GETKEYS
    SETID               JOBCODE_TBL     SETID
    JOBCODE             JOBCODE_TBL     JOBCODE
  FINDKEYS
    SETID               JOBCODE_TBL     SETID
    JOBCODE             JOBCODE_TBL     JOBCODE
                                        DESCR
                                        GVT_OCC_SERIES
                                        GVT_OFFICIAL_DE...

                                        SETID
                                        JOBCODE
```

Component Interface Tester - Enter key values, choose function

'Get' keys for Component Interface (double-click to set)

JOBCODE	
SETID	

Get Existing

'Create' keys for Component Interface (double-click to set)

JOBCODE	MYJOBCODE
SETID	SHARE

Create New

'Find' keys for Component Interface (double-click to set)

DESCR	
GVT_OCC_SERIES	
GVT_OFFICIAL_DESCR	

Find

☑ Interactive Mode (set properties immediately)

☐ Get History Items

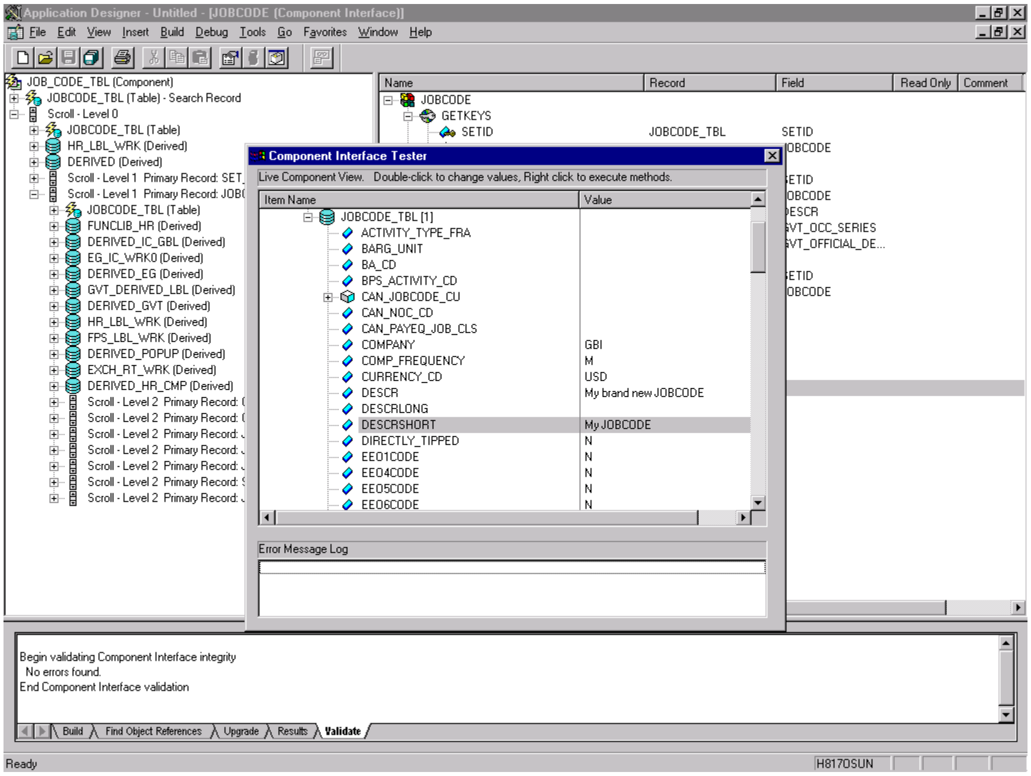
Cancel

```
Begin validating Component Interface integrity
 No errors found.
End Component Interface validation
```

a. Enter all required key values.

b. Click **Create New.**

When you enter valid values in Create keys, a pane that displays the JOBCODE data opens after the Table name is expanded with default data in place.


Application Designer - Untitled - [JOBCODE (Component Interface)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
JOB_CODE_TBL (Component)
JOBCODE_TBL (Table) - Search Record
Scroll - Level 0
JOBCODE_TBL (Table)
HR_LBL_WRK (Derived)
DERIVED (Derived)
Scroll - Level 1 Primary Record: SET
Scroll - Level 1 Primary Record: JOB
JOBCODE_TBL (Table)
FUNCLIB_HR (Derived)
DERIVED_IC_GBL (Derived)
EG_IC_WRK0 (Derived)
DERIVED_EG (Derived)
GVT_DERIVED_LBL (Derived)
DERIVED_GVT (Derived)
HR_LBL_WRK (Derived)
FPS_LBL_WRK (Derived)
DERIVED_POPUP (Derived)
EXCH_RT_WRK (Derived)
DERIVED_HR_CMP (Derived)
Scroll - Level 2 Primary Record:
Scroll - Level 2 Primary Record:
Scroll - Level 2 Primary Record:
Scroll - Level 2 Primary Record:
Scroll - Level 2 Primary Record:
Scroll - Level 2 Primary Record:
Scroll - Level 2 Primary Record:
Name
Record
Field
Read Only
Comment
JOBCODE
GETKEYS
SETID
JOBCODE_TBL
SETID
JOBCODE
SETID
JOBCODE
DESCR
GVT_OCC_SERIES
GVT_OFFICIAL_DE...
SETID
JOBCODE
Component Interface Tester
Live Component View. Double-click to change values, Right click to execute methods.
Error Message Log
Build
Find Object References
Upgrade
Results
Validate
Ready
H8170SUN

Item Name	Value
JOBCODE_TBL [1]	
ACTIVITY_TYPE_FRA	
BARG_UNIT	
BA_CD	
BPS_ACTIVITY_CD	
CAN_JOBCODE_CU	
CAN_NOC_CD	
CAN_PAYEQ_JOB_CLS	
COMPANY	GBI
COMP_FREQUENCY	M
CURRENCY_CD	USD
DESCR	My brand new JOBCODE
DESCRLONG	
DESCRSHORT	My JOBCODE
DIRECTLY_TIPPED	N
EEO1CODE	N
EEO4CODE	N
EEO5CODE	N
EEO6CODE	N

```
Begin validating Component Interface integrity
  No errors found.
End Component Interface validation
```

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 the **Save** icon.

   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.

D

Using PeopleSoft 8 Integration Broker

This appendix discusses 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.

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.

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:

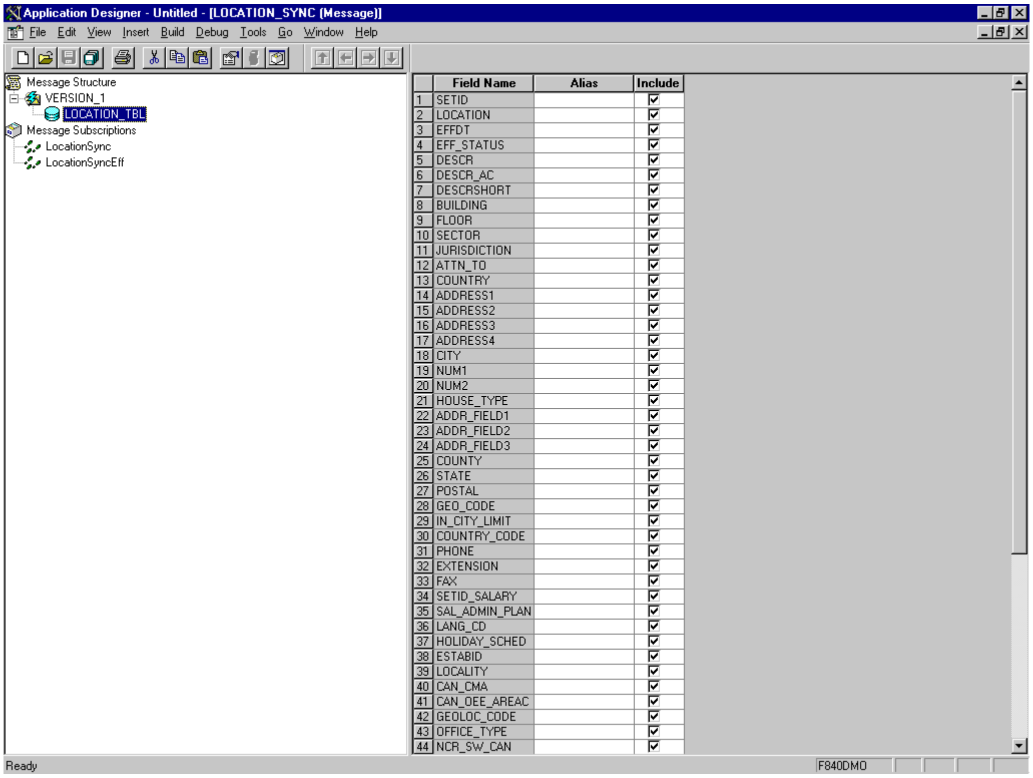
1. Ensure that the message is active and is routed to the proper Message Channel.
2. Configure the Message Channel.
3. Configure the IntegrationGateway.properties file to communicate with your PeopleSoft 8.4 application.
4. Configure the Integration Gateway and File Output connector.
5. Create and configure 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.

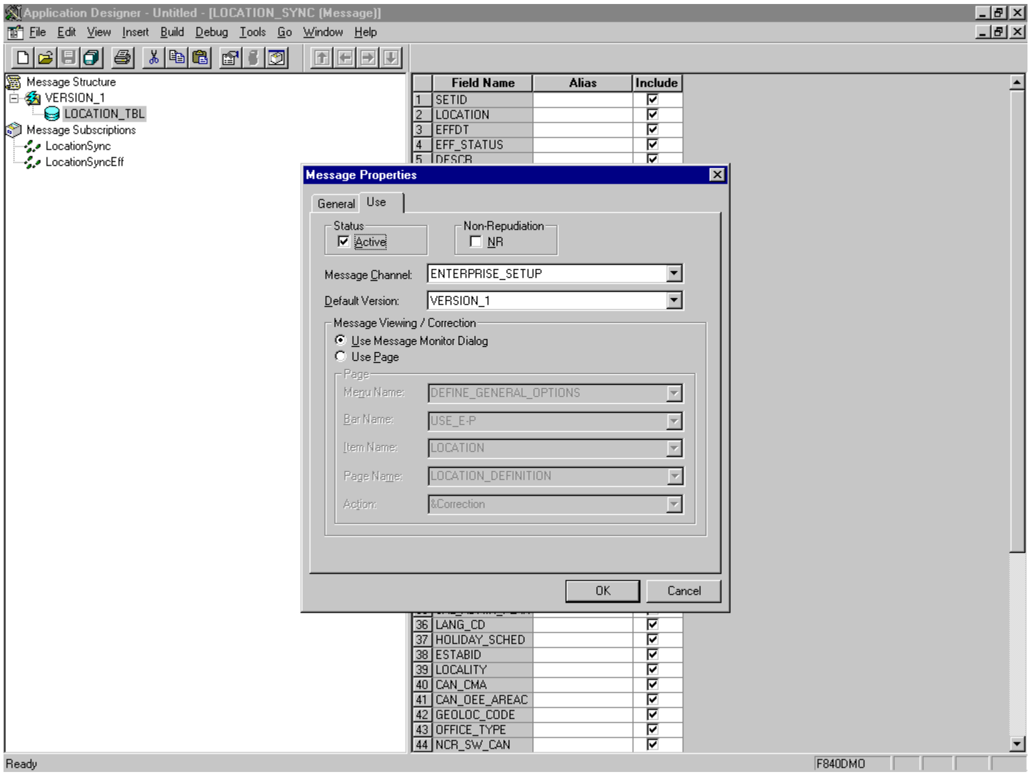


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Window Help
Message Structure
VERSION_1
LOCATION_TBL
Message Subscriptions
LocationSync
LocationSyncEff
Ready
F840DMO

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
6	DESCR_AC		☑
7	DESCRSHORT		☑
8	BUILDING		☑
9	FLOOR		☑
10	SECTOR		☑
11	JURISDICTION		☑
12	ATTN_TO		☑
13	COUNTRY		☑
14	ADDRESS1		☑
15	ADDRESS2		☑
16	ADDRESS3		☑
17	ADDRESS4		☑
18	CITY		☑
19	NUM1		☑
20	NUM2		☑
21	HOUSE_TYPE		☑
22	ADDR_FIELD1		☑
23	ADDR_FIELD2		☑
24	ADDR_FIELD3		☑
25	COUNTY		☑
26	STATE		☑
27	POSTAL		☑
28	GEO_CODE		☑
29	IN_CITY_LIMIT		☑
30	COUNTRY_CODE		☑
31	PHONE		☑
32	EXTENSION		☑
33	FAX		☑
34	SETID_SALARY		☑
35	SAL_ADMIN_PLAN		☑
36	LANG_CD		☑
37	HOLIDAY_SCHED		☑
38	ESTABID		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

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 box opens.


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Window Help
Ready
F840DMO

Message Structure
- VERSION_1
  - LOCATION_TBL

Message Subscriptions
- LocationSync
- LocationSyncEff

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
36	LANG_CD		☑
37	HOLIDAY_SCHED		☑
38	ESTABID		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

Message Properties

General | Use

Status: ☑ Active

Non-Repudiation: ☐ NR

Message Channel: ENTERPRISE_SETUP

Default Version: VERSION_1

Message Viewing / Correction
- ● Use Message Monitor Dialog
- ○ Use Page

Page
- Menu Name: DEFINE_GENERAL_OPTIONS
- Bar Name: USE_E-P
- Item Name: LOCATION
- Page Name: LOCATION_DEFINITION
- Action: &Correction

OK | Cancel

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=//isdsrv14:9000
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:

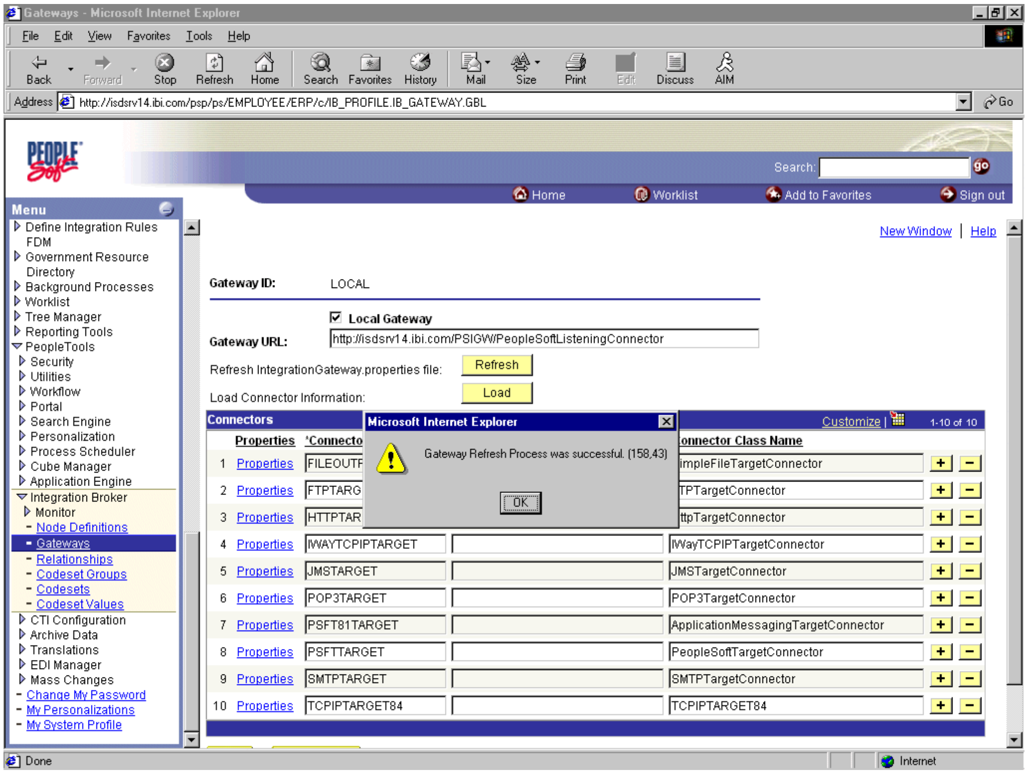
   ```
   machine-name/PSIGW/PeopleSoftListeningConnector
   ```

machine-name

Is the URL of your PeopleSoft Web Server.

4. Click **Refresh.**

   A message appears stating the outcome of the refresh process.



Address: http://isdsrv14.ibi.com/psp/ps/EMPLOYEE/ERP/c/IB_PROFILE.IB_GATEWAY.GBL

Gateway ID: LOCAL

☑ Local Gateway

Gateway URL: http://isdsrv14.ibi.com/PSIGW/PeopleSoftListeningConnector

Refresh IntegrationGateway.properties file: Refresh

Load Connector Information: Load

Connectors — 1-10 of 10

Microsoft Internet Explorer

Gateway Refresh Process was successful. (158,43)

	Properties	*Connecto		onnector Class Name
1	Properties	FILEOUTF		impleFileTargetConnector
2	Properties	FTPTARG		TPTargetConnector
3	Properties	HTTPTAR		ttpTargetConnector
4	Properties	IWAYTCPIPTARGET		IWayTCPIPTargetConnector
5	Properties	JMSTARGET		JMSTargetConnector
6	Properties	POP3TARGET		POP3TargetConnector
7	Properties	PSFT81TARGET		ApplicationMessagingTargetConnector
8	Properties	PSFTTARGET		PeopleSoftTargetConnector
9	Properties	SMTPTARGET		SMTPTargetConnector
10	Properties	TCPIPTARGET84		TCPIPTARGET84

5. Click **OK** and scroll down to click **Save.**

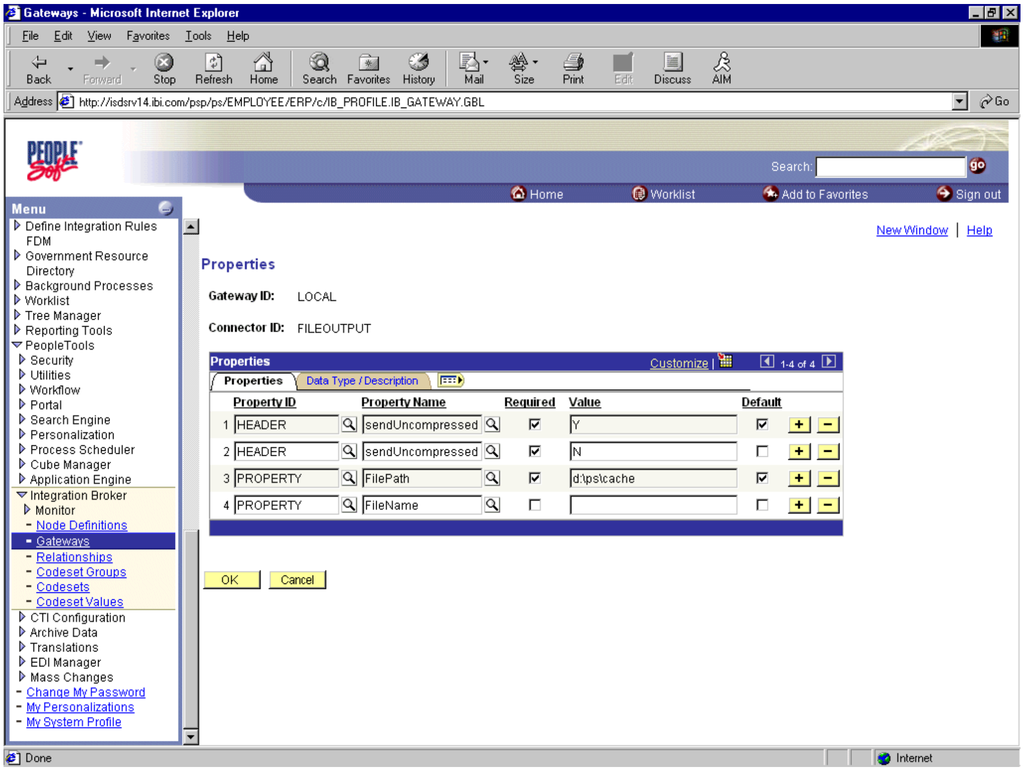
   You must click Save before continuing.

6. Click the **Properties** link for the FILEOUTPUT Connector ID.

   The Properties window for the FILEOUTPUT Connector opens.

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`.



Properties

Gateway ID: LOCAL

Connector ID: FILEOUTPUT

Properties

	Property ID	Property Name	Required	Value	Default
1	HEADER	sendUncompressed	☑	Y	☑
2	HEADER	sendUncompressed	☑	N	□
3	PROPERTY	FilePath	☑	d:\ps\cache	☑
4	PROPERTY	FileName	□		□

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.**

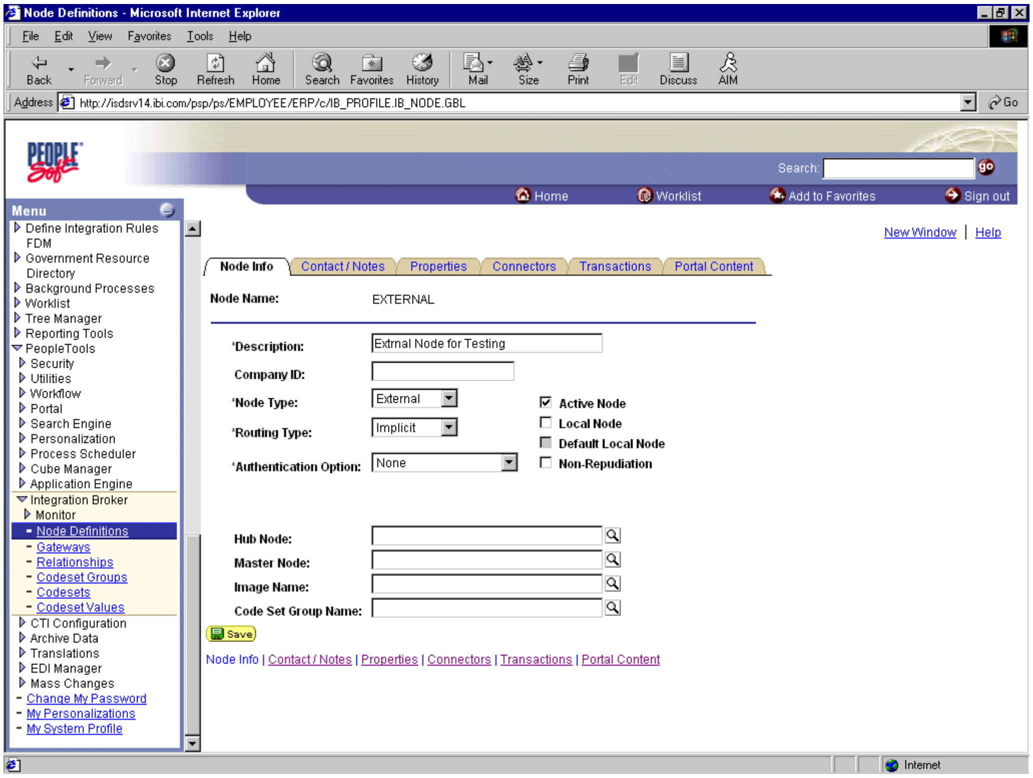
2. Select the **Add a New Value** tab.

3. In the **Node Name** field, type 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.


Node Definitions - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://isdsrv14.ibi.com/psp/ps/EMPLOYEE/ERP/c/IB_PROFILE.IB_NODE.GBL
Go
Search: go
Home Worklist Add to Favorites Sign out
Menu
Define Integration Rules FDM
Government Resource Directory
Background Processes
Worklist
Tree Manager
Reporting Tools
PeopleTools
Security
Utilities
Workflow
Portal
Search Engine
Personalization
Process Scheduler
Cube Manager
Application Engine
Integration Broker
Monitor
Node Definitions
Gateways
Relationships
Codeset Groups
Codesets
Codeset Values
CTI Configuration
Archive Data
Translations
EDI Manager
Mass Changes
Change My Password
My Personalizations
My System Profile
New Window | Help
Node Info | Contact / Notes | Properties | Connectors | Transactions | Portal Content
Save
Node Info | Contact / Notes | Properties | Connectors | Transactions | Portal Content
Internet

Node Name: EXTERNAL

*Description: Extrnal Node for Testing

Company ID:

*Node Type: External

*Routing Type: Implicit

*Authentication Option: None

☑ Active Node

□ Local Node

□ Default Local Node

□ Non-Repudiation

Hub Node:

Master Node:

Image Name:

Code Set Group Name:

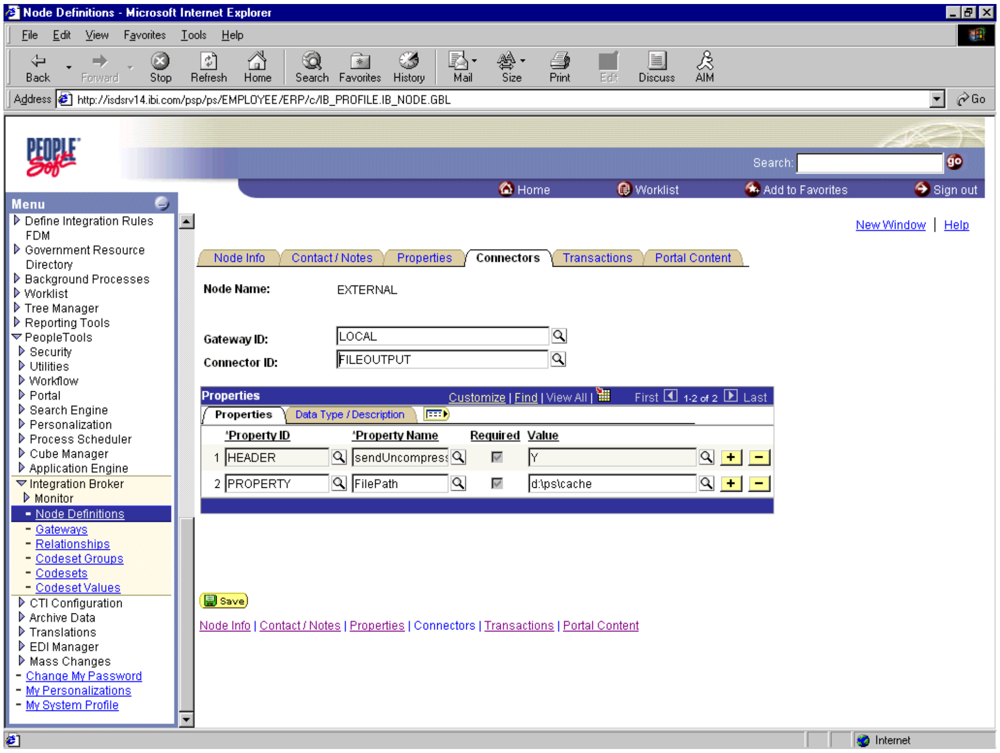
a. In the **Description** field, type an appropriate description.

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.



Node Name: EXTERNAL

Gateway ID: LOCAL

Connector ID: FILEOUTPUT

Properties

	*Property ID	*Property Name	Required	Value
1	HEADER	sendUncompress	☑	Y
2	PROPERTY	FilePath	☑	d:\ps\cache

a. Specify **LOCAL** for the **Gateway ID.**

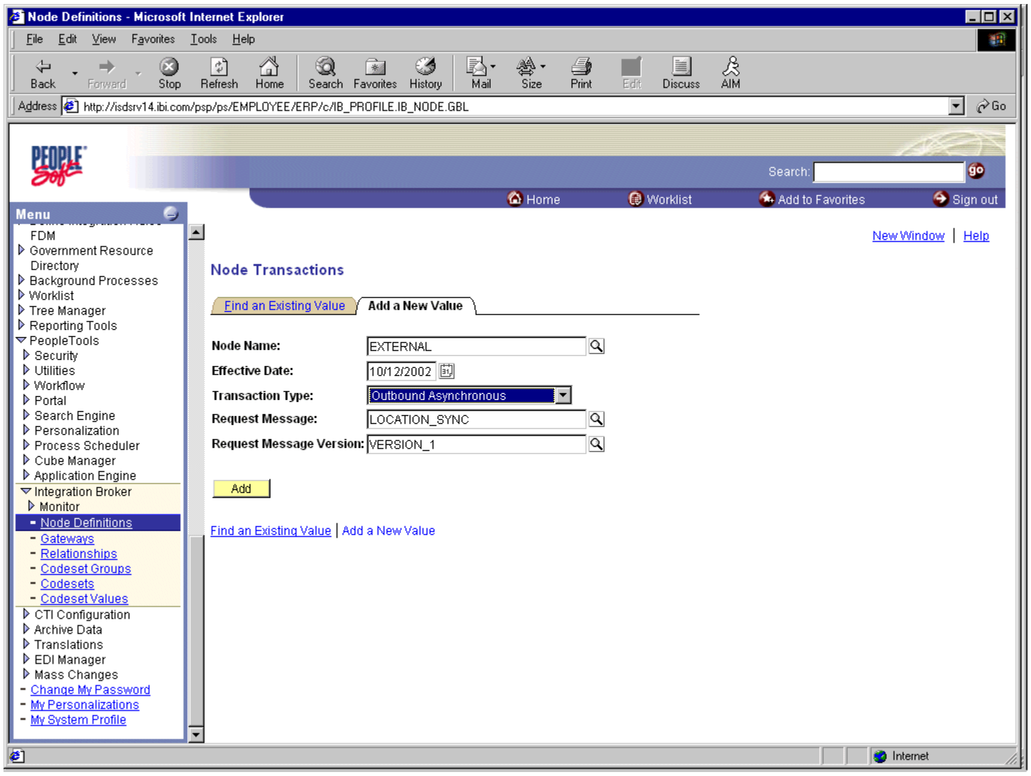
b. Specify **FILEOUTPUT** for the **Connector ID.**

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 opens.



Node Transactions

Node Name:	EXTERNAL
Effective Date:	10/12/2002
Transaction Type:	Outbound Asynchronous
Request Message:	LOCATION_SYNC
Request Message Version:	VERSION_1

a. From the **Transaction Type** list, select **Outbound Asynchronous.**

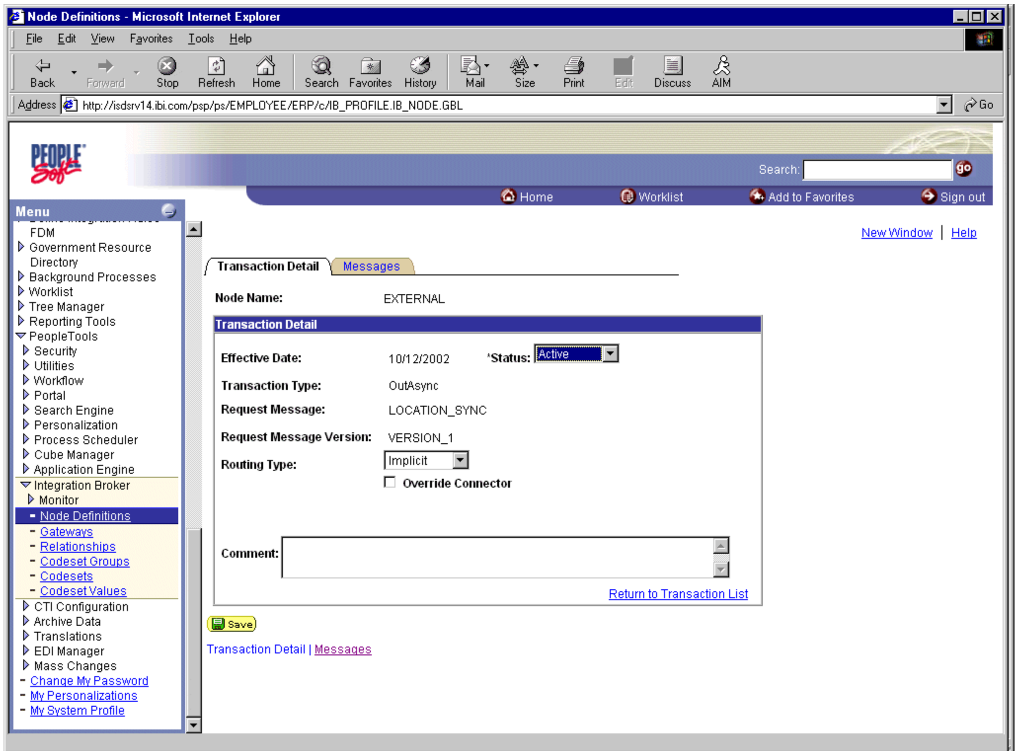
b. In the **Request Message** field, specify **LOCATION_SYNC.**

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 opens.



Node Name: EXTERNAL

Transaction Detail

Effective Date:	10/12/2002	*Status:	Active
Transaction Type:	OutAsync		
Request Message:	LOCATION_SYNC		
Request Message Version:	VERSION_1		
Routing Type:	Implicit		

□ Override Connector

Comment:

a. Verify that the **Routing Type** is **Implicit.**

b. Click **Save** and 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.

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.

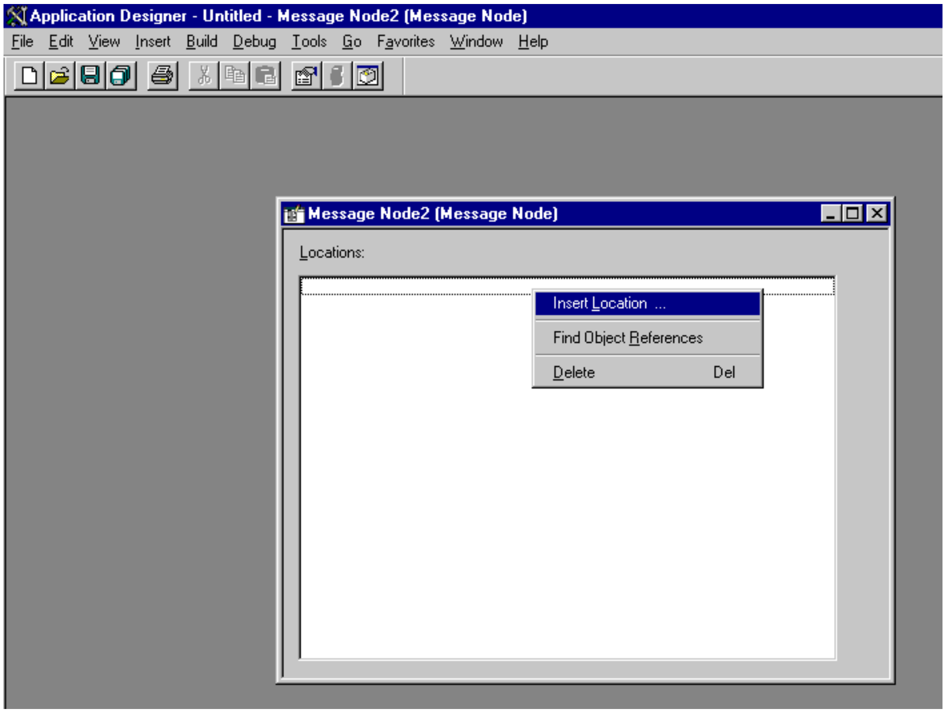
   These tasks are described in detail in the following procedures.

Creating and Configuring a New Message Node

To create and configure a new message node:

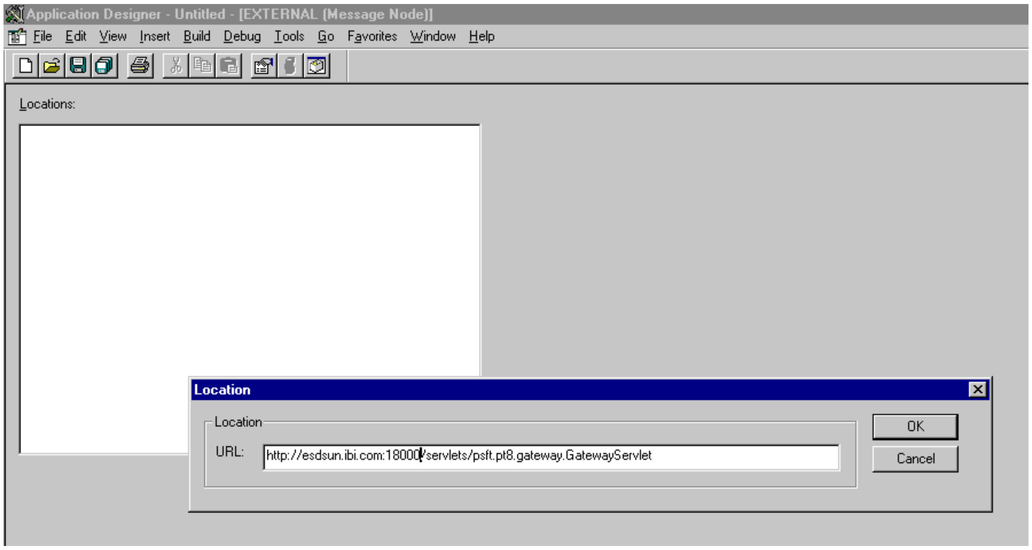
1. Select **New** from the File menu and click **Message Node.**

   A Message Node window opens.


Application Designer - Untitled - Message Node2 (Message Node)
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Node2 (Message Node)
Locations:
Insert Location ...
Find Object References
Delete Del

2. Right-click anywhere inside the white space and select **Insert Location.**

   The Location URL box opens.


Application Designer - Untitled - [EXTERNAL (Message Node)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Locations:
Location
Location
OK
Cancel

URL: http://esdsun.ibi.com:18000/servlets/psft.pt8.gateway.GatewayServlet

3. Type the following URL for the PeopleSoft Application Gateway (handler directory):

   ```
   machine-name:port/servlets/psft.pt8.gateway.GatewayServlet
   ```

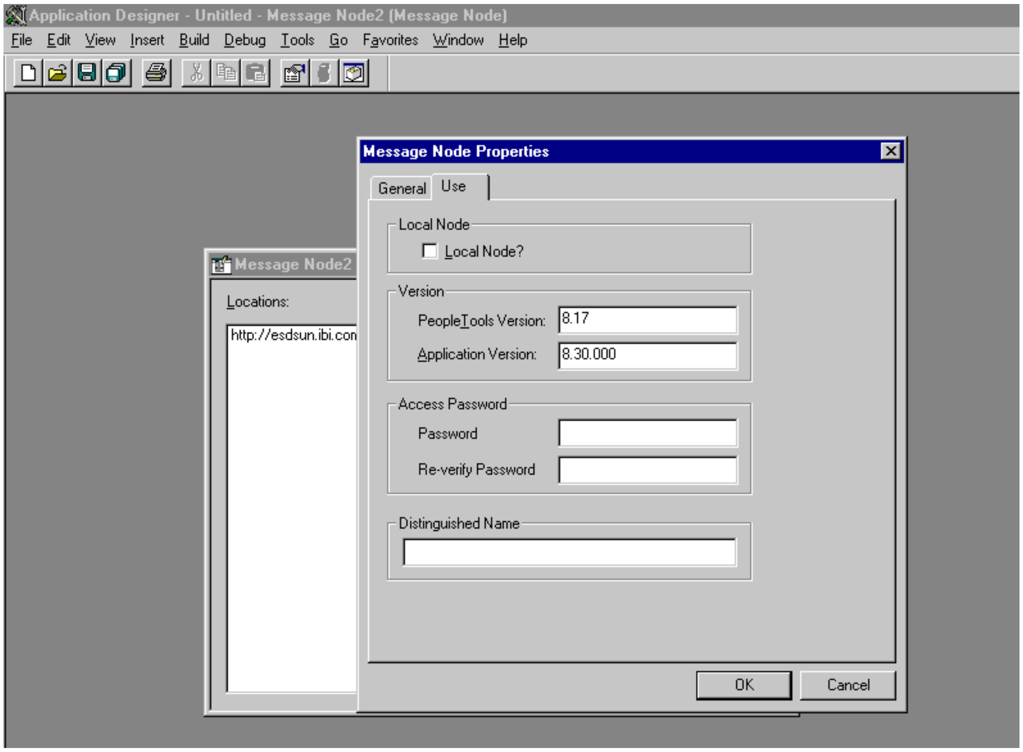
machine-name
Is the URL of your PeopleSoft Web server.

port
Is the socket on which the server is listening.

The characters you type after machine-name must be case-sensitive.

4. Click **OK.**

5. Invoke the Message Node Properties dialog box.


Application Designer - Untitled - Message Node2 (Message Node)
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Node2
Locations:
http://esdsun.ibi.con
Message Node Properties
General
Use
OK
Cancel

Local Node
☐ Local Node?

Version
PeopleTools Version: 8.17
Application Version: 8.30.000

Access Password
Password:
Re-verify Password:

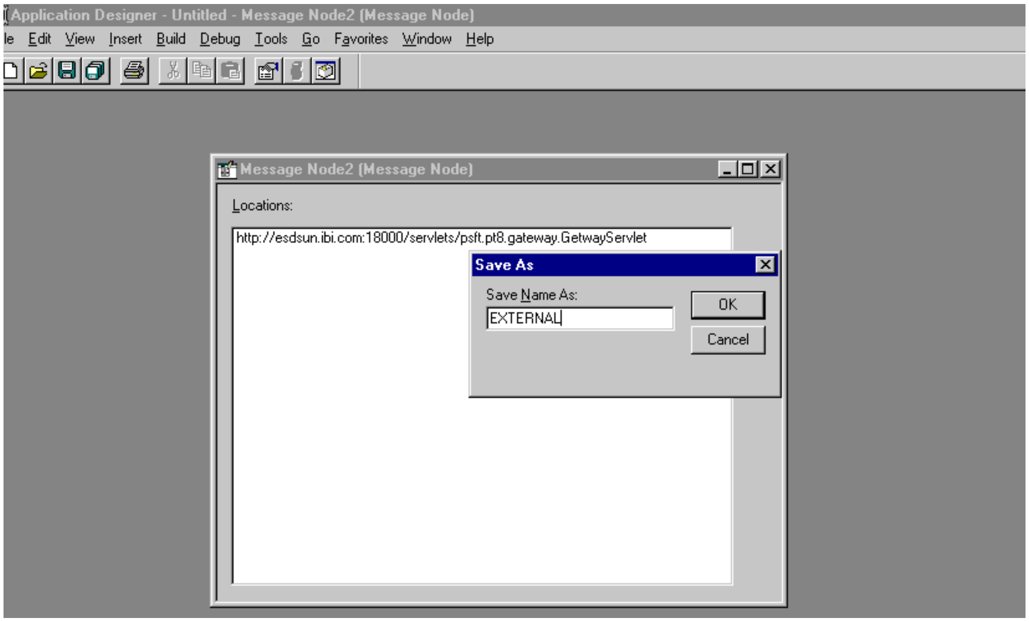
Distinguished Name:

   a. Select the **Use** tab.

   b. In the text boxes, type the PeopleTools and Application Version numbers.

   c. Click **OK.**

6. Invoke the Save As dialog box.


Application Designer - Untitled - Message Node2 (Message Node)
le Edit View Insert Build Debug Tools Go Favorites Window Help
Message Node2 (Message Node)
Save As
OK
Cancel

Locations: http://esdsun.ibi.com:18000/servlets/psft.pt8.gateway.GetwayServlet

Save Name As: EXTERNAL

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.

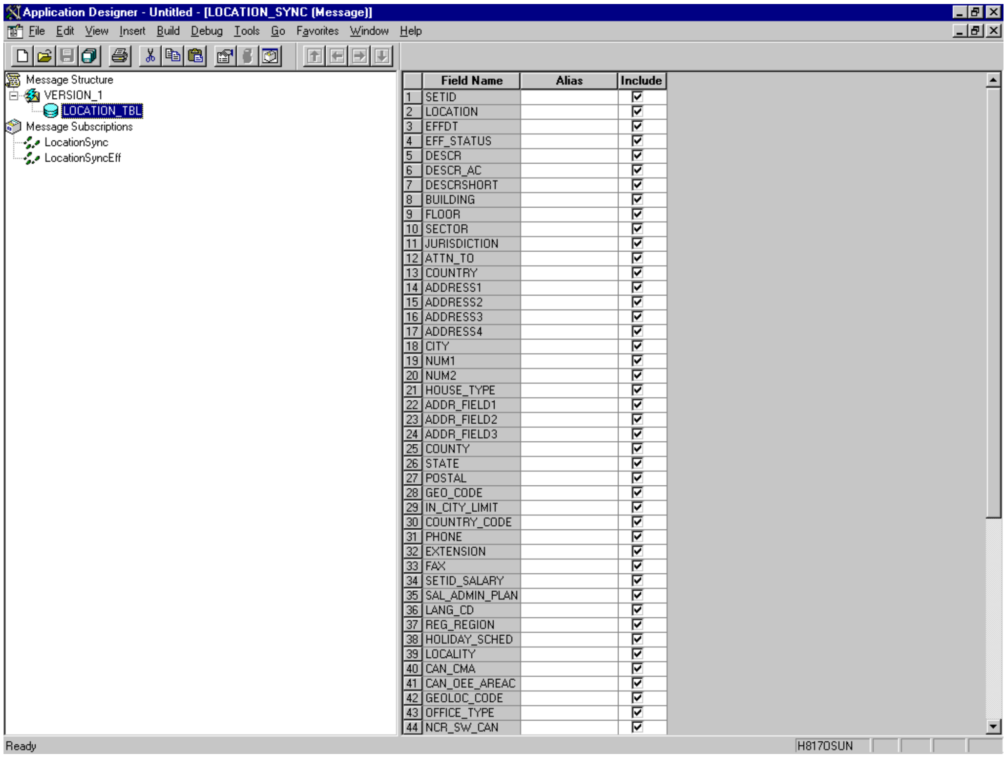
If you intend to migrate this message node to a different PeopleSoft environment (for example, from Test to QA), you can create a PeopleSoft project and insert the Message Node into the project.

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.

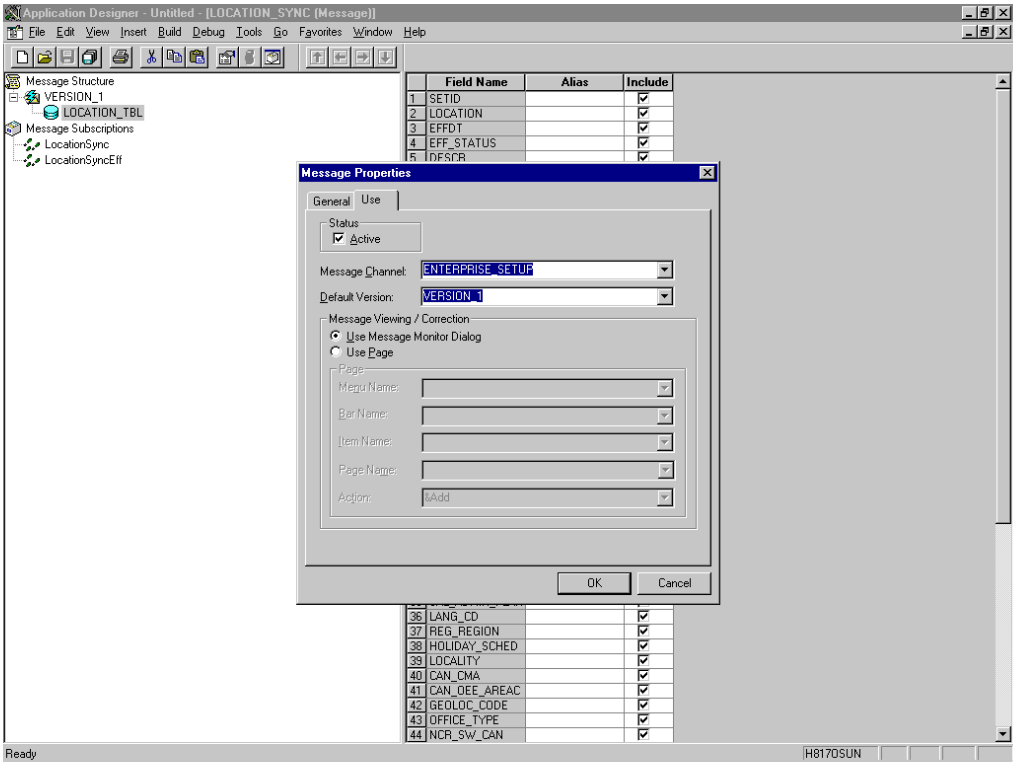


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Structure
VERSION_1
LOCATION_TBL
Message Subscriptions
LocationSync
LocationSyncEff
Ready
H817OSUN

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
6	DESCR_AC		☑
7	DESCRSHORT		☑
8	BUILDING		☑
9	FLOOR		☑
10	SECTOR		☑
11	JURISDICTION		☑
12	ATTN_TO		☑
13	COUNTRY		☑
14	ADDRESS1		☑
15	ADDRESS2		☑
16	ADDRESS3		☑
17	ADDRESS4		☑
18	CITY		☑
19	NUM1		☑
20	NUM2		☑
21	HOUSE_TYPE		☑
22	ADDR_FIELD1		☑
23	ADDR_FIELD2		☑
24	ADDR_FIELD3		☑
25	COUNTY		☑
26	STATE		☑
27	POSTAL		☑
28	GEO_CODE		☑
29	IN_CITY_LIMIT		☑
30	COUNTRY_CODE		☑
31	PHONE		☑
32	EXTENSION		☑
33	FAX		☑
34	SETID_SALARY		☑
35	SAL_ADMIN_PLAN		☑
36	LANG_CD		☑
37	REG_REGION		☑
38	HOLIDAY_SCHED		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

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 box opens.


Application Designer - Untitled - [LOCATION_SYNC (Message)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Message Structure
VERSION_1
LOCATION_TBL
Message Subscriptions
LocationSync
LocationSyncEff
Ready
H817OSUN

	Field Name	Alias	Include
1	SETID		☑
2	LOCATION		☑
3	EFFDT		☑
4	EFF_STATUS		☑
5	DESCR		☑
36	LANG_CD		☑
37	REG_REGION		☑
38	HOLIDAY_SCHED		☑
39	LOCALITY		☑
40	CAN_CMA		☑
41	CAN_OEE_AREAC		☑
42	GEOLOC_CODE		☑
43	OFFICE_TYPE		☑
44	NCR_SW_CAN		☑

Message Properties

General | Use

Status
☑ Active

Message Channel: ENTERPRISE_SETUP

Default Version: VERSION_1

Message Viewing / Correction
● Use Message Monitor Dialog
○ Use Page

Page
Menu Name:
Bar Name:
Item Name:
Page Name:
Action: &Add

OK | Cancel

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

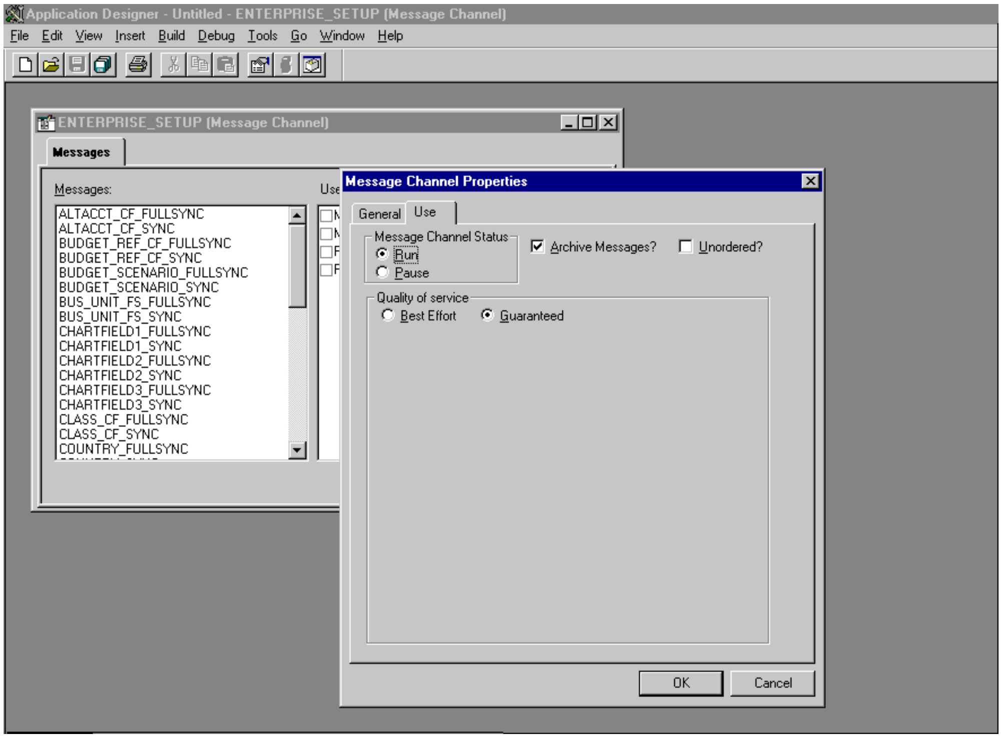
To configure the message channel:

1. From the **File** menu, point to **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 box opens.


Application Designer - Untitled - ENTERPRISE_SETUP (Message Channel)
File Edit View Insert Build Debug Tools Go Window Help
ENTERPRISE_SETUP (Message Channel)
Messages
Message Channel Properties
General
Use
OK
Cancel

Messages:

ALTACCT_CF_FULLSYNC
ALTACCT_CF_SYNC
BUDGET_REF_CF_FULLSYNC
BUDGET_REF_CF_SYNC
BUDGET_SCENARIO_FULLSYNC
BUDGET_SCENARIO_SYNC
BUS_UNIT_FS_FULLSYNC
BUS_UNIT_FS_SYNC
CHARTFIELD1_FULLSYNC
CHARTFIELD1_SYNC
CHARTFIELD2_FULLSYNC
CHARTFIELD2_SYNC
CHARTFIELD3_FULLSYNC
CHARTFIELD3_SYNC
CLASS_CF_FULLSYNC
CLASS_CF_SYNC
COUNTRY_FULLSYNC

Message Channel Status
● Run
○ Pause

☑ Archive Messages? □ Unordered?

Quality of service
○ Best Effort ● Guaranteed

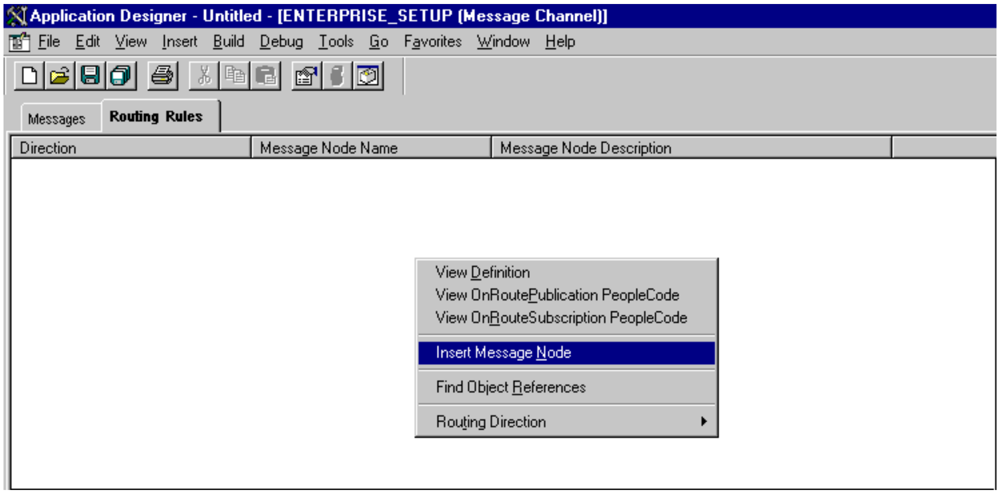
a. Select the **Use** tab.

b. Ensure that the Message Channel status is set to **Run.**

c. Click **OK.**

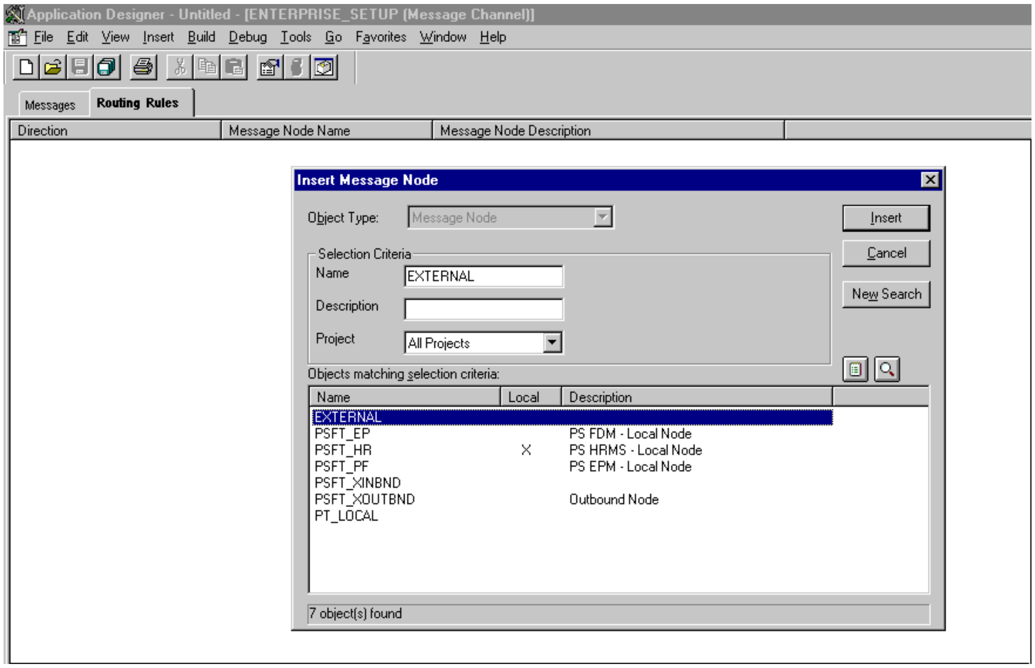
4. From the left pane, select the **Routing Rules** tab.

   The pane is blank.


Application Designer - Untitled - [ENTERPRISE_SETUP (Message Channel)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Messages
Routing Rules
Direction
Message Node Name
Message Node Description
View Definition
View OnRoutePublication PeopleCode
View OnRouteSubscription PeopleCode
Insert Message Node
Find Object References
Routing Direction

5. Right-click the pane and select **Insert Message Node.**

   The Insert Message Node dialog box opens.


Application Designer - Untitled - [ENTERPRISE_SETUP (Message Channel)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Messages
Routing Rules
Direction
Message Node Name
Message Node Description
Insert Message Node
Insert
Cancel
New Search

Object Type: Message Node

Selection Criteria

Name: EXTERNAL

Description:

Project: All Projects

Objects matching selection criteria:

Name	Local	Description
EXTERNAL		
PSFT_EP		PS FDM - Local Node
PSFT_HR	X	PS HRMS - Local Node
PSFT_PF		PS EPM - Local Node
PSFT_XINBND		
PSFT_XOUTBND		Outbound Node
PT_LOCAL		

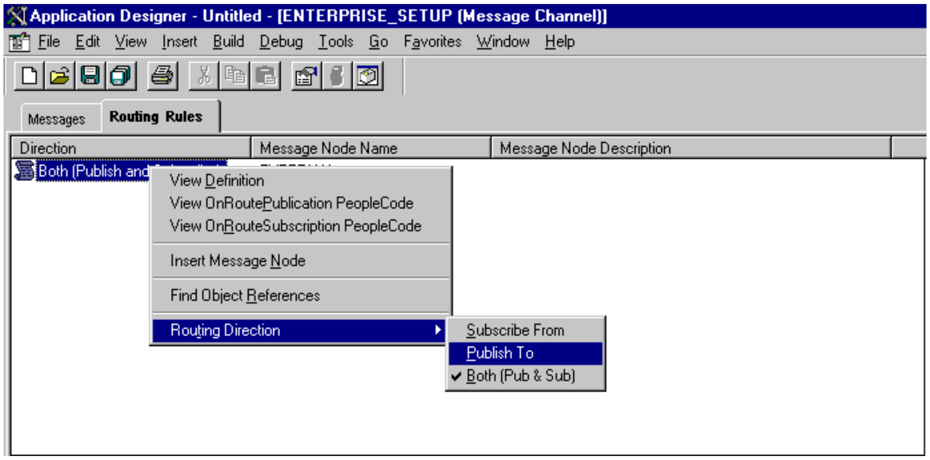
7 object(s) found

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.


Application Designer - Untitled - [ENTERPRISE_SETUP (Message Channel)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Messages
Routing Rules
Direction
Message Node Name
Message Node Description
Both (Publish and
View Definition
View OnRoutePublication PeopleCode
View OnRouteSubscription PeopleCode
Insert Message Node
Find Object References
Routing Direction
Subscribe From
Publish To
✓ Both (Pub & Sub)

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:

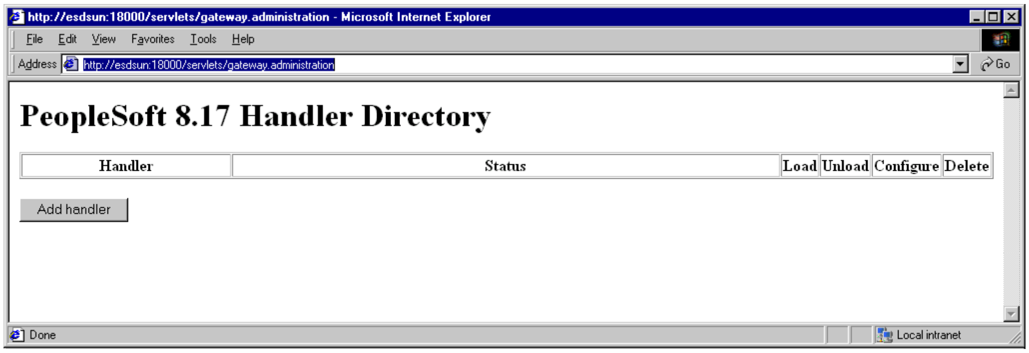
1. In a Web browser, launch the PeopleSoft 8.1 configuration servlet interface (also known as the server gateway) by entering the following URL:

```
machine-name:port/servlets/gateway.administration
```

machine-name
Is the name of the application server where PeopleSoft is hosted.

port
Is the port number on which the application server is listening.

The Handler Directory window opens.

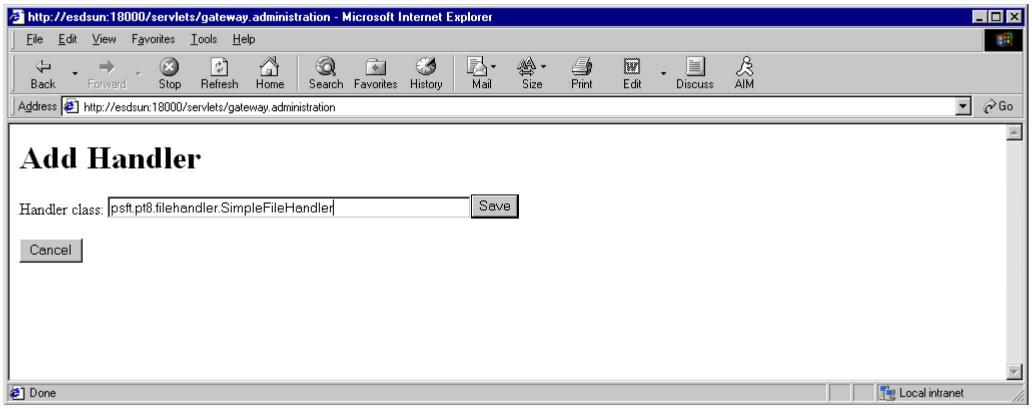

http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://esdsun:18000/servlets/gateway.administration
Go
Add handler
Done
Local intranet

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete

2. Click **Add Handler.**

   The Add Handler window opens.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Save
Cancel
Done
Local intranet

Add Handler

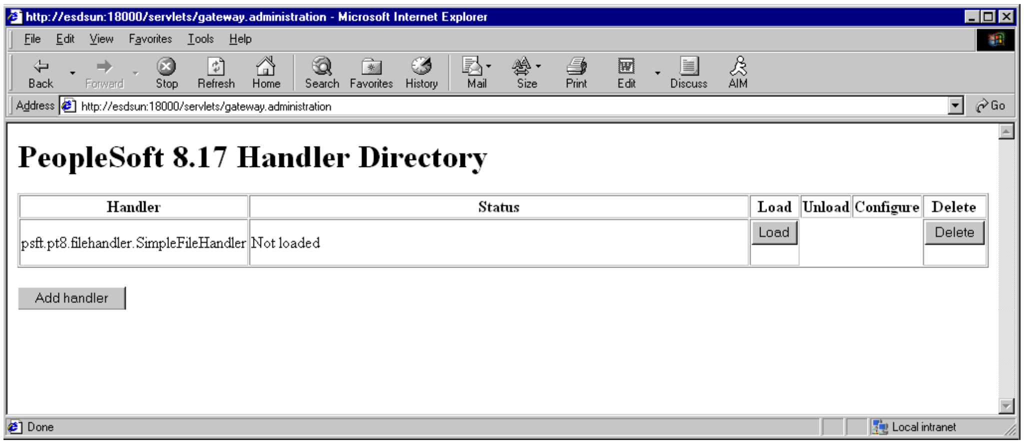
Handler class: psft.pt8.filehandler.SimpleFileHandler

3. Type 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 opens.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

PeopleSoft 8.17 Handler Directory

Handler	Status	Load	Unload	Configure	Delete
psft.pt8.filehandler.SimpleFileHandler	Not loaded	Load			Delete

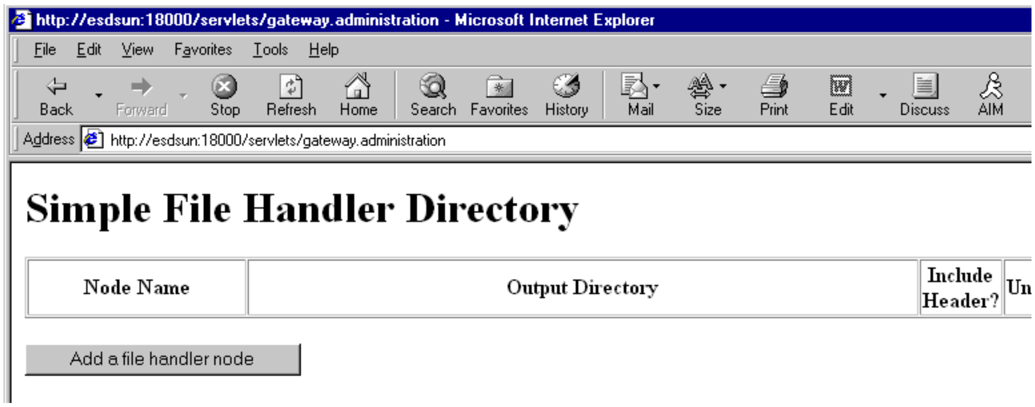
Add handler

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 opens.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration

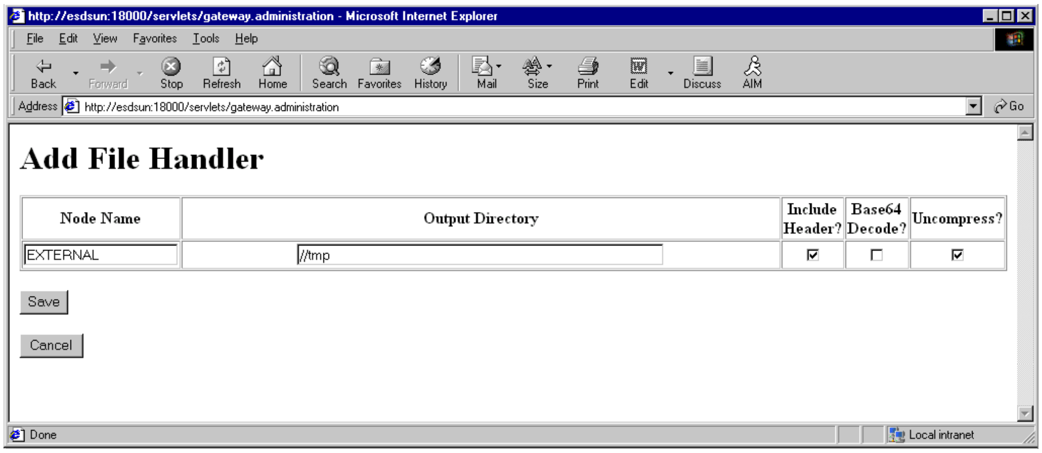
Simple File Handler Directory

Node Name	Output Directory	Include Header?	Un

Add a file handler node

7. Click **Add a file handler node.**

   The Add File Handler window opens.


http://esdsun:18000/servlets/gateway.administration - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address http://esdsun:18000/servlets/gateway.administration
Go
Done
Local intranet

Add File Handler

Node Name	Output Directory	Include Header?	Base64 Decode?	Uncompress?
EXTERNAL	//tmp	☑	□	☑

Save

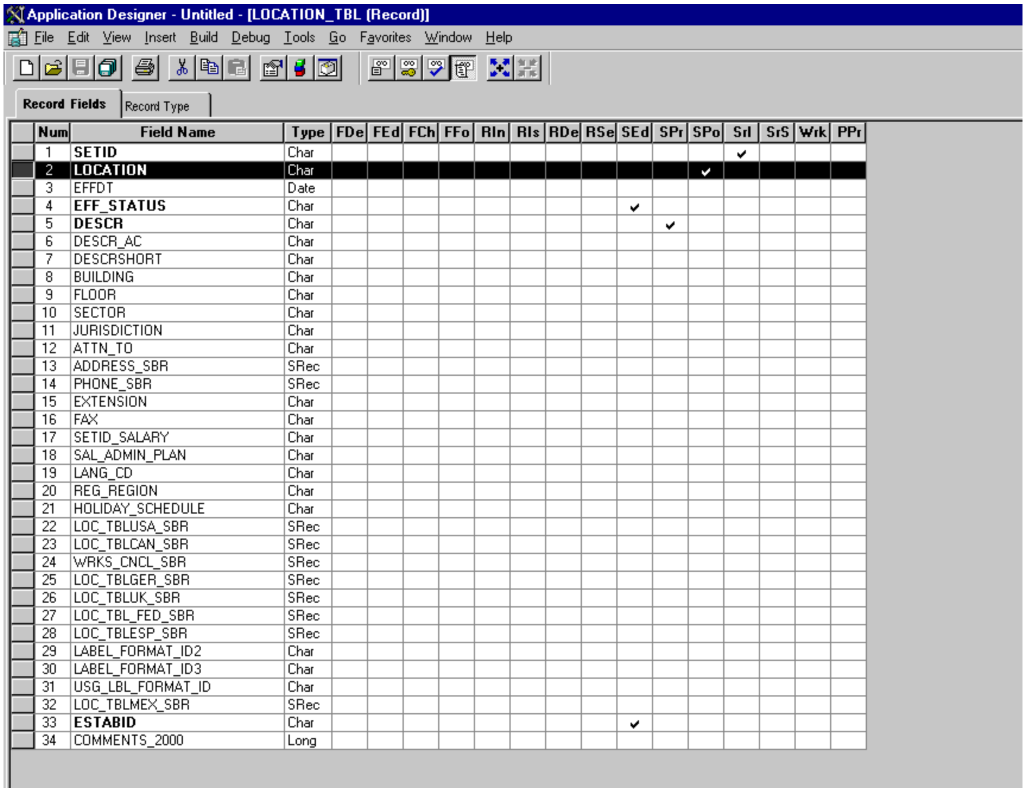
Cancel

8. In the Node Name field, type the name of the message node that you created in "Creating and Configuring a New Message Node" on page D-11, for example, EXTERNAL.

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.


Application Designer - Untitled - [LOCATION_TBL (Record)]
File Edit View Insert Build Debug Tools Go Favorites Window Help
Record Fields
Record Type

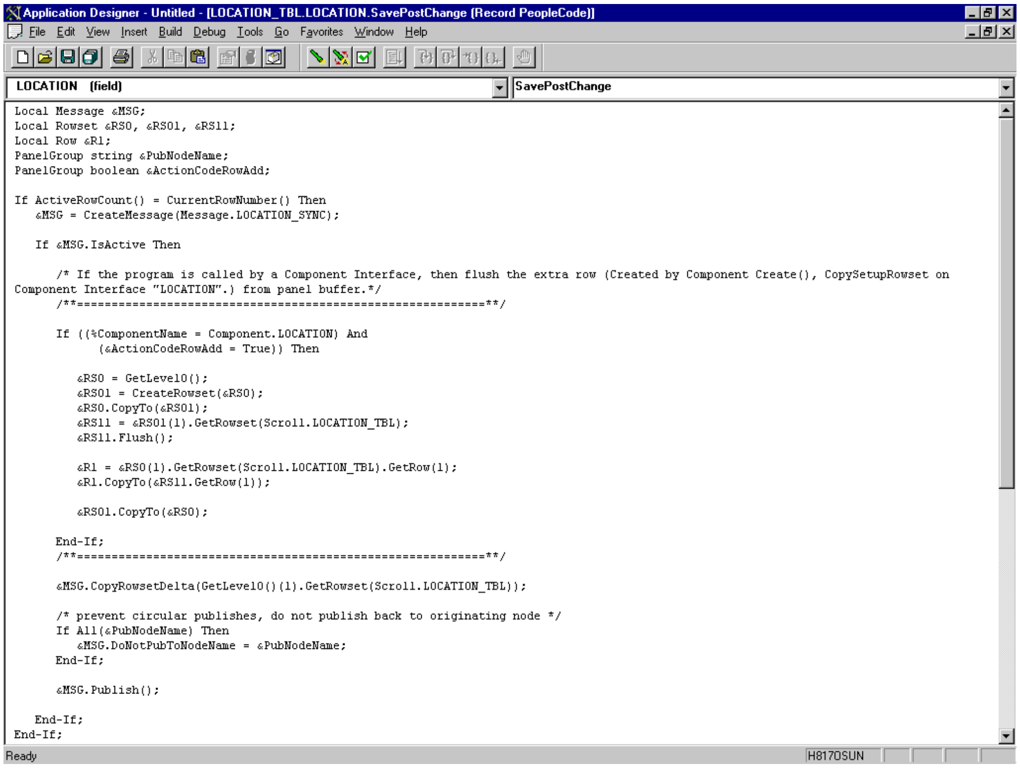
Num	Field Name	Type	FDe	FEd	FCh	FFo	RIn	RIs	RDe	RSe	SEd	SPr	SPo	SrI	SrS	Wrk	PPr
1	**SETID**	Char												✓			
2	**LOCATION**	Char											✓				
3	EFFDT	Date															
4	**EFF_STATUS**	Char									✓						
5	**DESCR**	Char										✓					
6	DESCR_AC	Char															
7	DESCRSHORT	Char															
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									✓						
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
Ready
H817OSUN

```
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;
```

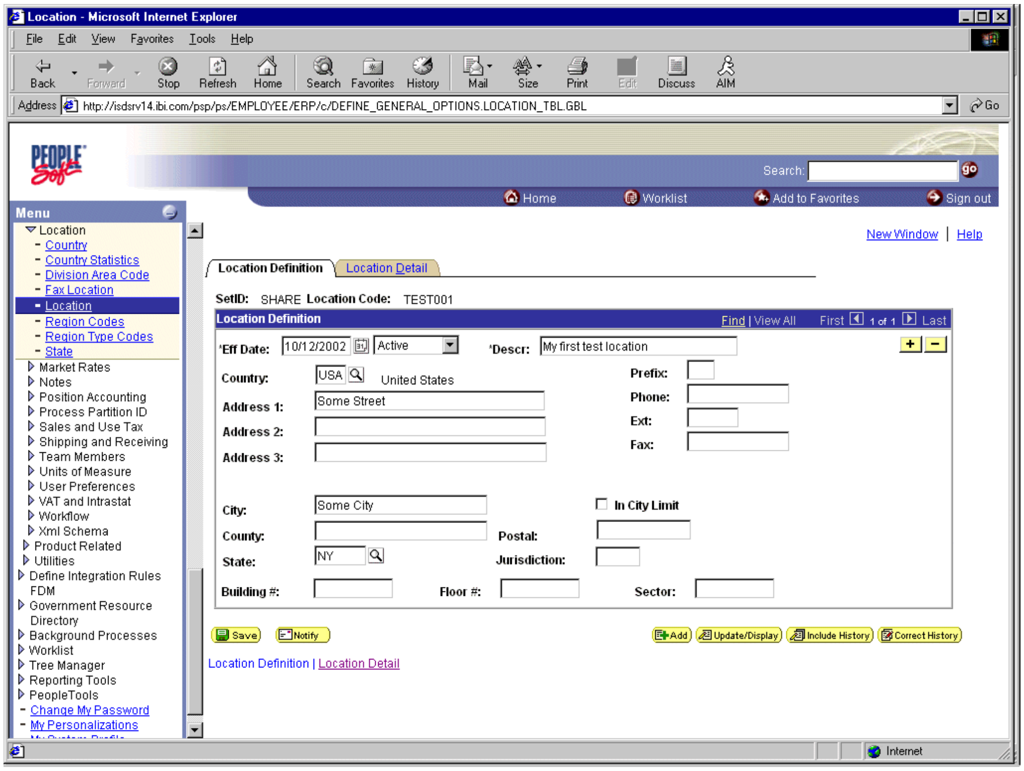
For more information about PeopleCode, consult your PeopleSoft Online Library.

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).

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.



SetID: SHARE Location Code: TEST001

Location Definition

*Eff Date: 10/12/2002 Active *Descr: My first test location

Country: USA United States

Address 1: Some Street

Address 2:

Address 3:

Prefix:

Phone:

Ext:

Fax:

City: Some City

☐ In City Limit

County:

Postal:

State: NY

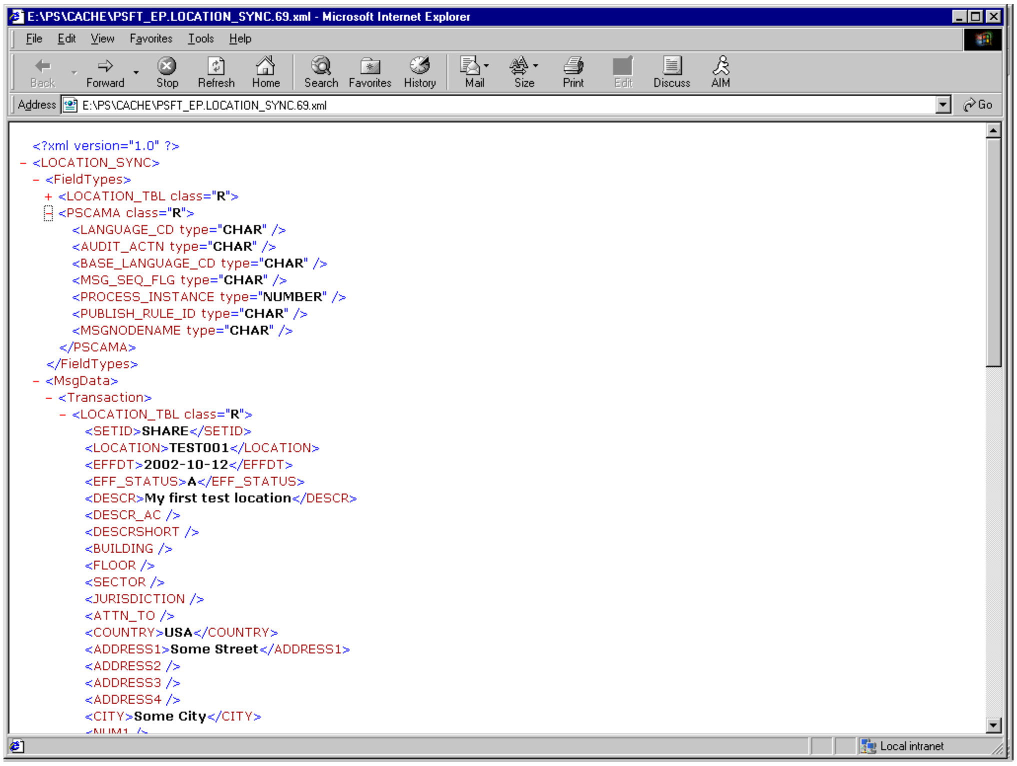
Jurisdiction:

Building #:

Floor #:

Sector:

The following figure shows a portion of the XML output.


E:\PS\CACHE\PSFT_EP.LOCATION_SYNC.69.xml - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Forward Stop Refresh Home Search Favorites History Mail Size Print Edit Discuss AIM
Address E:\PS\CACHE\PSFT_EP.LOCATION_SYNC.69.xml
Go
Local intranet

```
  <?xml version="1.0" ?>
- <LOCATION_SYNC>
  - <FieldTypes>
    + <LOCATION_TBL class="R">
    - <PSCAMA class="R">
        <LANGUAGE_CD type="CHAR" />
        <AUDIT_ACTN type="CHAR" />
        <BASE_LANGUAGE_CD type="CHAR" />
        <MSG_SEQ_FLG type="CHAR" />
        <PROCESS_INSTANCE type="NUMBER" />
        <PUBLISH_RULE_ID type="CHAR" />
        <MSGNODENAME type="CHAR" />
      </PSCAMA>
    </FieldTypes>
  - <MsgData>
    - <Transaction>
      - <LOCATION_TBL class="R">
          <SETID>SHARE</SETID>
          <LOCATION>TEST001</LOCATION>
          <EFFDT>2002-10-12</EFFDT>
          <EFF_STATUS>A</EFF_STATUS>
          <DESCR>My first test location</DESCR>
          <DESCR_AC />
          <DESCRSHORT />
          <BUILDING />
          <FLOOR />
          <SECTOR />
          <JURISDICTION />
          <ATTN_TO />
          <COUNTRY>USA</COUNTRY>
          <ADDRESS1>Some Street</ADDRESS1>
          <ADDRESS2 />
          <ADDRESS3 />
          <ADDRESS4 />
          <CITY>Some City</CITY>
          <NUM1 />
```

> **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.

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 or not 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.

The OracleAS Adapter for PeopleSoft can work with PeopleSoft outbound synchronous messages. Outbound synchronous messages involve additional configuration steps, both within PeopleSoft and in Oracle Application 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.

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.

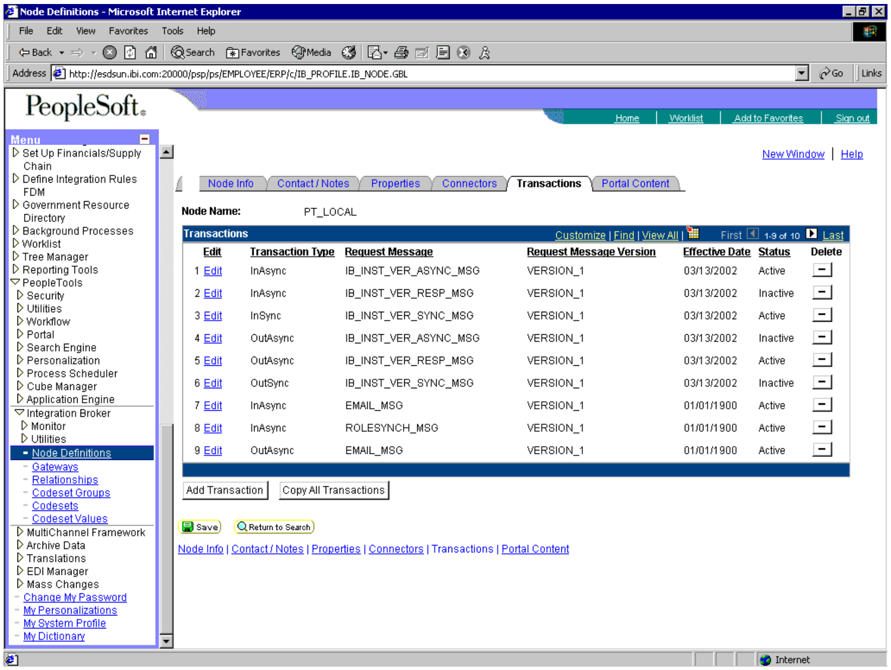
The following example uses a node and transaction delivered by PeopleSoft. However, this example is for illustrative purposes only and does not actually work as delivered without additional steps. As of Financials release 8.42, there are no preconfigured outbound synchronous transactions that you can 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 opens.



Node Name: PT_LOCAL

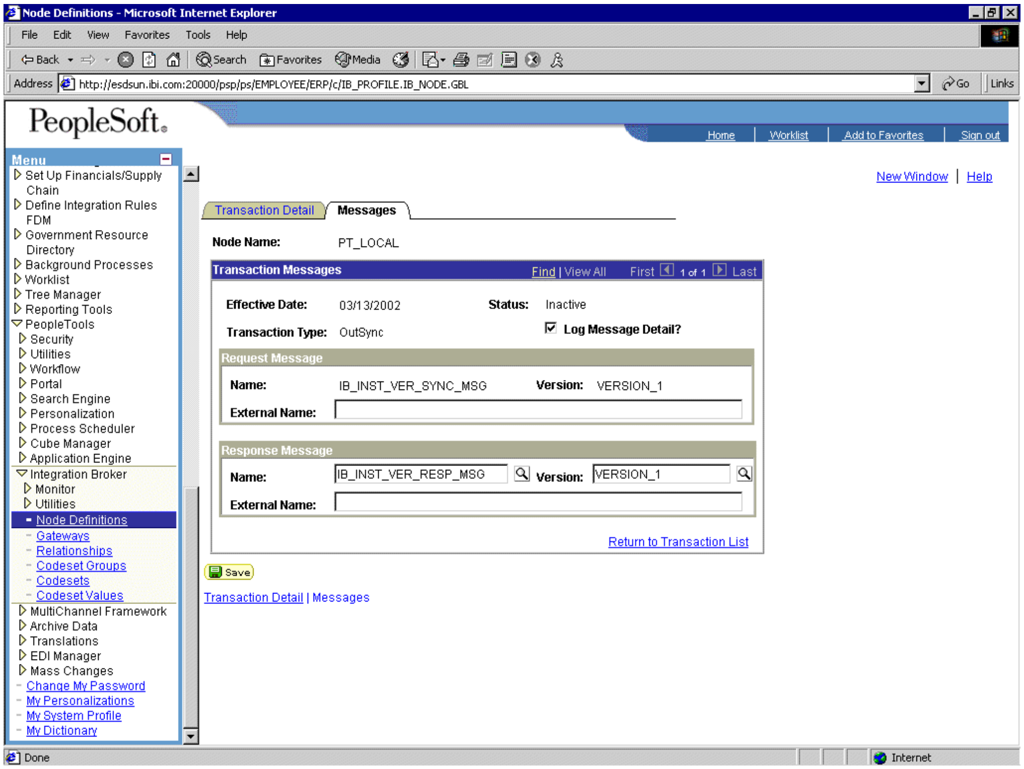
Transactions

	Edit	Transaction Type	Request Message	Request Message Version	Effective Date	Status	Delete
1	Edit	InAsync	IB_INST_VER_ASYNC_MSG	VERSION_1	03/13/2002	Active	−
2	Edit	InAsync	IB_INST_VER_RESP_MSG	VERSION_1	03/13/2002	Inactive	−
3	Edit	InSync	IB_INST_VER_SYNC_MSG	VERSION_1	03/13/2002	Active	−
4	Edit	OutAsync	IB_INST_VER_ASYNC_MSG	VERSION_1	03/13/2002	Inactive	−
5	Edit	OutAsync	IB_INST_VER_RESP_MSG	VERSION_1	03/13/2002	Active	−
6	Edit	OutSync	IB_INST_VER_SYNC_MSG	VERSION_1	03/13/2002	Inactive	−
7	Edit	InAsync	EMAIL_MSG	VERSION_1	01/01/1900	Active	−
8	Edit	InAsync	ROLESYNCH_MSG	VERSION_1	01/01/1900	Active	−
9	Edit	OutAsync	EMAIL_MSG	VERSION_1	01/01/1900	Active	−

   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 opens.



Node Name: PT_LOCAL

Transaction Messages

Effective Date: 03/13/2002 Status: Inactive

Transaction Type: OutSync ☑ Log Message Detail?

Request Message

Name: IB_INST_VER_SYNC_MSG Version: VERSION_1

External Name:

Response Message

Name: IB_INST_VER_RESP_MSG Version: VERSION_1

External Name:

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 Application 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 TCPIP84TARGET connector for outbound synchronous messages.

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.

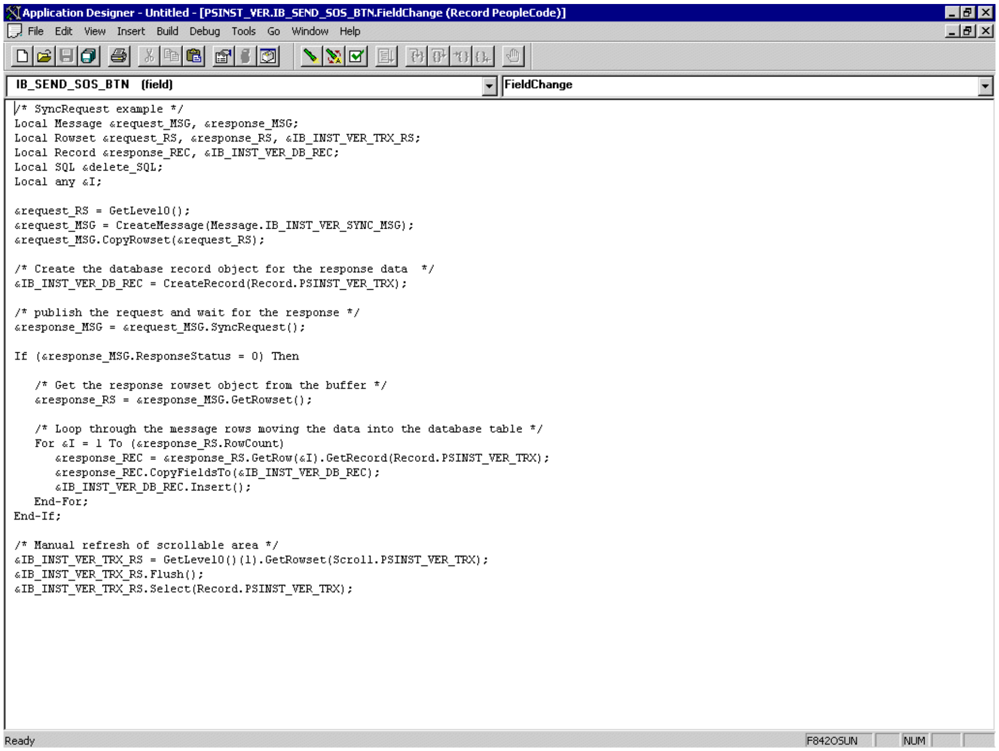
To view the code:

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 opens.



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
Ready
F842OSUN
NUM

```
/* 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);
```

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