BEA WebLogic Adapter for PeopleSoft User's Guide
Version 5.5.011
For WebLogic Server 9.1

DN3501560.0306

March 13, 2006

Preface

This document explains how to use the BEA WebLogic Adapter for PeopleSoft, which is used to develop client-server interfaces between PeopleSoft and other applications. It describes how to use the BEA WebLogic Adapter for PeopleSoft with Application Explorer to develop online connections to PeopleSoft applications.

How This Manual Is Organized

The following table lists the numbers and titles of the chapters and appendixes for this manual with a brief description of the contents of each chapter and appendix.

Chapter/Appendix		**Contents**
1	Introducing the BEA WebLogic Adapter for PeopleSoft	Provides an overview of the BEA WebLogic Adapter for PeopleSoft and summarizes how to use it to integrate PeopleSoft systems with other applications.
2	Configuring the BEA WebLogic Adapter for PeopleSoft	Describes how to configure the BEA WebLogic Adapter for PeopleSoft.
3	Generating Component Interface APIs	Describes how to generate Component Interface APIs for use with the BEA WebLogic Adapter for PeopleSoft.
4	Configuring the PeopleSoft Message Router	Describes how to configure the TCP/IP Target Connector (in PeopleSoft release 8.4) and the TCP/IP Handler (in PeopleSoft release 8.1).
5	Creating XML Schemas or Web Services for PeopleSoft	Describes how to create XML schemas and Web services for PeopleSoft business objects using Servlet Application Explorer.
6	Listening for PeopleSoft Events	Describes how to use Servlet Application Explorer to connect to PeopleSoft and listen for events.
7	Management and Monitoring	Describes the management and monitoring tools provided by iBSE and JCA.
8	Troubleshooting and Error Messages	Explains the limitations and workarounds when connecting to PeopleSoft.

Chapter/Appendix		Contents
A	Using Component Interfaces	Describes how to create, secure, and test a Component Interface for use with the BEA WebLogic Adapter for PeopleSoft.
B	Using PeopleSoft 8 Integration Broker	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.
C	Best Practices and Performance Considerations	Provides best practices and performance considerations for the BEA WebLogic Adapter for PeopleSoft when deployed to the BEA WebLogic Server.

Documentation Conventions

The following table lists and describes the conventions that apply throughout this manual.

Convention	Description
`THIS TYPEFACE` or `this typeface`	Denotes syntax that you must enter exactly as shown.
`this typeface`	Represents a placeholder (or variable) in syntax for a value that you or the system must supply.
`underscore`	Indicates a default setting.
this typeface	Represents a placeholder (or variable), a cross-reference, or an important term.
this typeface	Highlights a file name or command.
Key + Key	Indicates keys that you must press simultaneously.
`{ }`	Indicates two or three choices; type one of them, not the braces.
`|`	Separates mutually exclusive choices in syntax. Type one of them, not the symbol.
`...`	Indicates that you can enter a parameter multiple times. Type only the parameter, not the ellipsis points (…).

Convention	Description
. . .	Indicates that there are (or could be) intervening or additional commands.

Contact Us!

Your feedback on the BEA WebLogic Adapter for PeopleSoft documentation is important to us.

Send us e-mail at docsupport@bea.com if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the BEA WebLogic Adapter for PeopleSoft documentation.

In your e-mail message, please indicate that you are using the documentation for BEA WebLogic Adapter for PeopleSoft and the version of the documentation.

If you have any questions about this version of BEA WebLogic Adapter for PeopleSoft, or if you have problems using the BEA WebLogic Adapter for PeopleSoft, contact BEA Customer Support through BEA WebSUPPORT at www.bea.com. You can also contact Customer Support by using the contact information provided on the Customer Support Card which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number
- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

Help Us to Serve You Better

To help our consultants answer your questions effectively, please be prepared to provide specifications and sample files and to answer questions about errors and problems.

The following tables list the specifications our consultants require.

Platform	
Operating System	
OS Version	
Product List	
Adapters	
Adapter Deployment	For example, JCA, or Integration Business Services Engine.
Container Version	

The following table lists components. Specify the version in the column provided.

Component	**Version**
Adapter	
EIS (DBMS/APP)	
HOTFIX / Service Pack	

The following table lists the types of Application Explorer. Specify the version (and platform, if different than listed previously) in the columns provided.

Application Explorer Type	**Version**	**Platform**
Swing		
Servlet		
ASP		

In the following table, specify the JVM version and vendor in the columns provided.

Version	**Vendor**

The following table lists additional questions to help us serve you better.

Request/Question	**Error/Problem Details or Information**
Provide usage scenarios or summarize the application that produces the problem.	
Did this happen previously?	
Can you reproduce this problem consistently?	
Any **change in the application environment**: software configuration, EIS/ database configuration, application, and so forth?	
Under what circumstance does the problem *not* occur?	
Describe the **steps** to reproduce the problem.	
Describe the **problem**.	
Specify the **error** message(s).	

The following table lists error/problem files that might be applicable.

XML schema
XML instances
Other input documents (transformation)
Error screen shots
Error output files
Trace and log files
Log transaction

Contents

CHAPTER 1

Introducing the BEA WebLogic Adapter for PeopleSoft

Topics:

- Key Features of the BEA WebLogic Adapter for PeopleSoft
- How the BEA WebLogic Adapter for PeopleSoft Works
- Deployment Information for the BEA WebLogic Adapter for PeopleSoft

This section provides an overview of the BEA WebLogic Adapter for PeopleSoft and describes how to integrate other systems with PeopleSoft systems.

Key Features of the BEA WebLogic Adapter for PeopleSoft

The BEA WebLogic Adapter for PeopleSoft provides a means to exchange real-time business data between PeopleSoft systems and other application, database, or external business partner systems. The adapter enables external applications for inbound and outbound processing with PeopleSoft.

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. Services also are referred to as component methods.
- **Events:** Applications use this capability to access PeopleSoft data only when a PeopleSoft business event occurs. Events also are referred to as messages.

When you access a PeopleSoft component from another application, you use:

- **Component Interfaces.** If a Component Interface does not exist, create, secure, and test one. For more information, see Appendix A, *Using Component Interfaces* or your PeopleSoft documentation.

  If the Component Interface exists, but you modified it, secure and test it. For more information, see Appendix A, *Using Component Interfaces* or 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. For more information, see Chapter 3, *Generating Component Interface APIs*.
- **Schemas and Web services.** Create schemas or Web services for the component methods. For more information, see Chapter 5, *Creating XML Schemas or Web Services for PeopleSoft*.

To receive a message from PeopleSoft, you use:

- **The Integration environment.** Configure and test your PeopleSoft Integration Broker (release 8.4) or Application Messaging environment (release 8.1). For information on properly configuring the environment, see Appendix B, *Using PeopleSoft 8 Integration Broker* or 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). For more information, see Chapter 4, *Configuring the PeopleSoft Message Router*.

How the BEA WebLogic Adapter for PeopleSoft Works

The adapter uses your application server and XML messages to enable non-PeopleSoft applications to communicate and exchange transactions using one of the following two facilities:

- PeopleSoft Component Interface
- PeopleSoft Integration Broker (in release 8.4) or Application Messaging Manager (in release 8.1).

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.

The adapter is bidirectional, enabling it to:

- Detect an event by receiving an XML document from PeopleSoft using Integration Broker or Application Messaging.
- Pass an XML request document to execute an instance of the PeopleSoft Component Interface and its method.

PeopleSoft Enterprise Application Integration Architecture

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 BEA WebLogic Adapter for PeopleSoft uses the PeopleSoft framework and leverages various integration access methods to provide the most flexibility and functionality.

Integration access methods supported by the BEA WebLogic Adapter for PeopleSoft include:

- PeopleSoft Java™ API using Component Interfaces.
- PeopleSoft XML using Integration Broker or 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 (for example. 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 facilitates the integration of PeopleSoft XML with PeopleSoft. The BEA WebLogic Adapter for PeopleSoft provides a handler that must be configured within the PeopleSoft application gateway using TCP/IP transport services.

Deployment Information for the BEA WebLogic Adapter for PeopleSoft

The BEA WebLogic Adapter for PeopleSoft works in conjunction with one of the following components:

- Integration Business Services Engine (iBSE)
- Enterprise Connector for J2EE™ Connector Architecture (JCA)

When the adapter is hosted in a third party application server environment, Application Explorer, used to configure PeopleSoft connections, create Web services, and configure event capabilities, can be configured to work in a Web services environment in conjunction with iBSE or JCA. When working in a JCA environment, the connector uses the Common Client Interface (CCI) to provide fast integration services using adapters instead of using Web services.

Deployment Information Roadmap

The following table lists the location of deployment and user information for components of the BEA WebLogic Adapter for PeopleSoft.

Deployed Component	**For more information, see**
Application Explorer	Chapters 5, 6 and 7 of this guide
Integration Business Services Engine (iBSE)	*BEA WebLogic ERP Adapter Installation and Configuration*
Enterprise Connector for J2EE Connector Architecture (JCA)	*BEA WebLogic ERP Adapter Installation and Configuration*

Application Explorer

Application Explorer uses an explorer metaphor to browse the PeopleSoft system for metadata. The explorer enables you to create XML schemas and Web services for the associated object. In addition, you can create ports and channels to listen for events in PeopleSoft. External applications that access PeopleSoft through the BEA WebLogic Adapter for PeopleSoft use either XML schemas or Web services to pass data between the external application and the adapter.

Integration Business Services Engine

The Integration Business Services Engine (iBSE) exposes—as Web services—enterprise assets that are accessible from adapters regardless of the programming language or the particular operating system.

iBSE simplifies the creation and execution of Web services when running:

- Custom and legacy applications
- Database queries and stored procedures
- Packaged applications
- Terminal emulation and screen-based systems
- Transactional systems

Web services is a distributed programming architecture that solves Enterprise Application Integration (EAI) hurdles that other programming models cannot. It enables programs to communicate with one another using a text-based, platform- and language-independent message format called XML.

Coupled with a platform and language independent messaging protocol called SOAP (Simple Object Access Protocol), XML enables application development and integration by assembling previously built components from multiple Web services.

Enterprise Connector for J2EE Connector Architecture (JCA)

The Enterprise Connector for J2EE Connector Architecture (JCA) enables developers of JCA-compliant applications to deploy adapters as JCA resources. The connector is supported on J2EE-compliant application servers such as your application server.

The Connector for JCA is distributed as a standard Resource Adapter Archive (RAR) for deployment to the application server. Thus, the connector can be used in systems that are non-compliant, although services such as pooled connections are not available.

CHAPTER 2

Configuring the BEA WebLogic Adapter for PeopleSoft

Topics:

- Specifying the Version of PeopleSoft
- Installing the Adapter Component Interfaces
- Installing the TCP/IP Message Router for the BEA WebLogic Adapter for PeopleSoft
- Copying PeopleSoft Files Into the Lib Directory

This section describes how to configure the BEA WebLogic Adapter for PeopleSoft. You must:

- Select your version of PeopleSoft.
- Install the adapter Component Interfaces.
- Install the adapter TCP/IP message router.
- Copy the psjoa.jar file (and, for PeopleSoft release 8.1, the pstools.properties file) into the iWay55\lib directory.

Specifying the Version of PeopleSoft

The BEA WebLogic Adapter for PeopleSoft supports multiple versions of PeopleSoft. However, one version can be incompatible with another. The adapter must recognize the version you use. After installation, files for both versions of PeopleSoft appear in the iWay55\lib directory. The default location for this directory on Windows is:

```
C:\Program Files\iWay55\lib
```

On non-Windows systems, you use the corresponding location.

Procedure: How to Specify the Version of PeopleSoft

To specify the correct version of PeopleSoft:

1. To ensure the adapter functions properly, remove the file that does not correspond to your version.

   For PeopleSoft **8.4x** releases, remove the following file:

   ```
   iwpsci81.jar
   ```

   For PeopleSoft **8.1x** releases, remove the following file:

   ```
   iwpsci84.jar
   ```

2. After you change the contents of the lib directory, restart all iWay components.

Installing the Adapter Component Interfaces

The BEA WebLogic Adapter for PeopleSoft includes two custom Component Interfaces. Application Explorer uses these Component Interfaces to create schemas for events and services.

To configure Component Interfaces for use by the BEA WebLogic Adapter for PeopleSoft:

1. Import and build the Component Interfaces.
2. Configure Component Interface security.
3. Test the Component Interfaces.

Importing and Building the Component Interfaces

The Component Interfaces supplied with the BEA WebLogic Adapter for PeopleSoft are delivered through a PeopleSoft project.

- For Release 8.4, the project is the IWY_CI_84 project, packaged in iwpsci84.zip.
- For Release 8.1, the project is the IWY_CI_81 project, packaged in iwpsci81.zip.

These files are installed with iWay 5.5.

On Windows, their default location is:

`C:\Program Files\iWay55\etc\misc\peoplesoft`

On non-Windows systems, use the corresponding location. If this location does not exist, contact your distributor for copies of the relevant files.

Procedure: How to Import and Build the Component Interfaces

To import the IWY_CI_81 or IWY_CI_84 project to PeopleSoft 8:

1. Unzip *iwpsci81.zip* or *iwpsci84.zip* to a directory of your choice.

   The unzip process creates its own subdirectory. For example, if you extract the file to c:\temp, it creates c:\temp\IWY_CI_81 or c:\temp\IWY_CI_84.

2. Launch the PeopleSoft 8 Application Designer in two-tier mode.
3. To open the Copy From File Select Project dialog box:

   In PeopleSoft **8.4**, select *Copy Project* from the Tools menu and then, select *From File*.

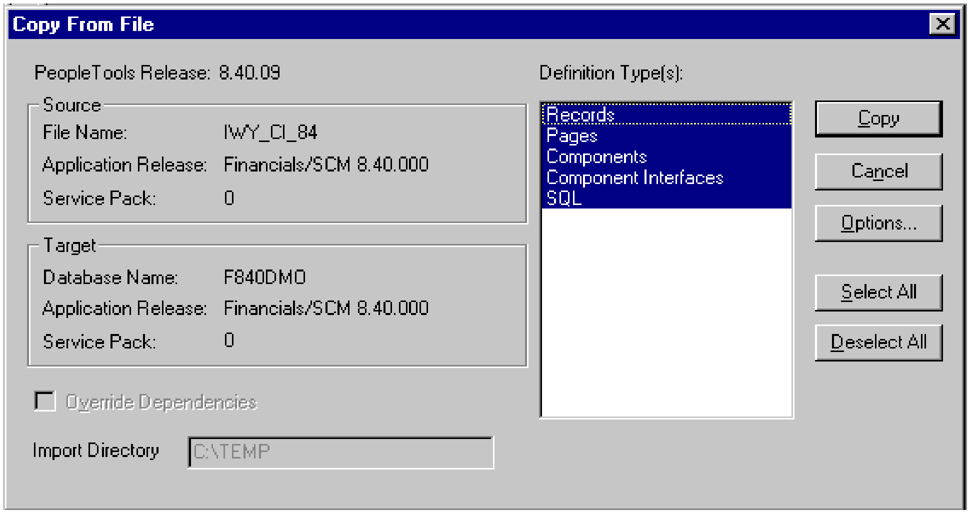
   In PeopleSoft **8.1**, select *Copy Project from File* from the File menu.

   The Copy Project From File dialog box opens.

4. Navigate to the original directory to which you unzipped the file.
5. To open the Copy From File dialog box, click *Open* (in 8.4) or *Copy* (in 8.1).

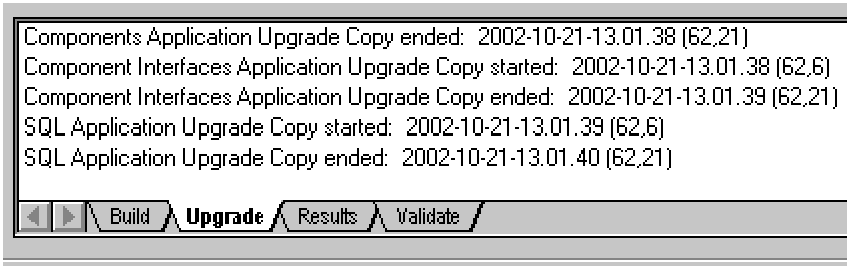
   Note: Although the following images illustrate PeopleSoft release 8.4, the corresponding instructions are accurate for releases 8.1 and 8.4.

The following image shows the PeopleSoft Application Designer Copy From File dialog box. It includes PeopleTools release and target information on the left, a Definition Type(s) pane, and five buttons, Copy, Cancel, Options, Select All, and Deselect All.



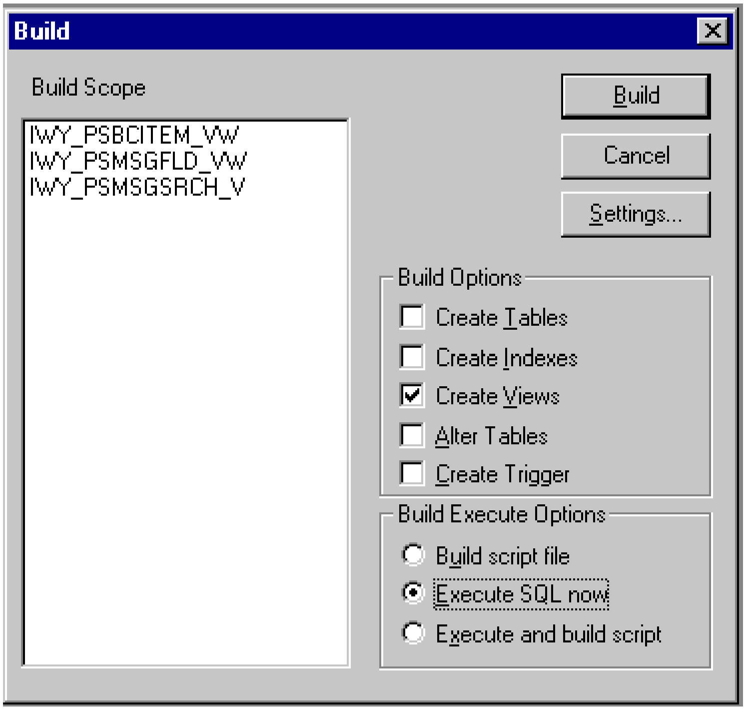
6. Highlight all the objects that appear under Definition Types and click *Copy*.

   The following image shows a message generated by Application Designer that indicates successful completion of the copying.



7. To build the views in the project, from the Build menu, select *Project*.

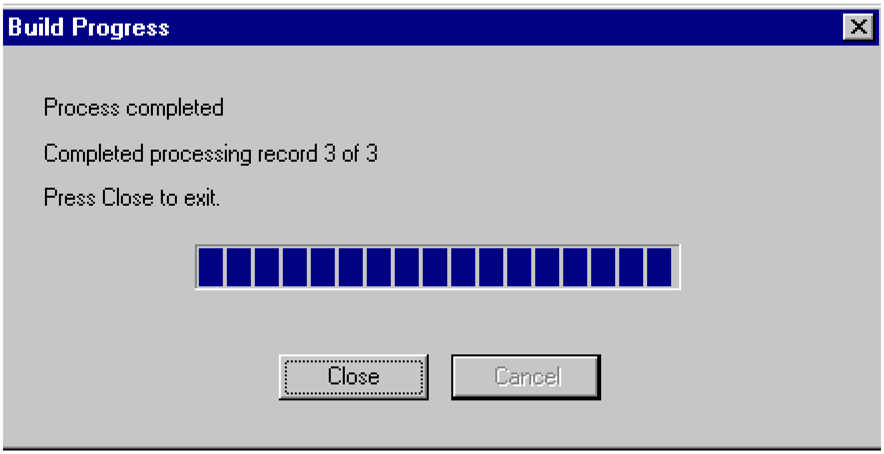
The following image shows the Build dialog box. It contains a Build Scope pane, a Build Options pane, and a Build Execute Options pane. It also contains Build, Cancel, and Settings buttons.



- **a.** From Build Options, select the *Create Views* check box.
- **b.** In the Build Execute Options pane, select the customary option for your site. (For example, in the previous image, the Execute SQL now option is selected.)

8. Click *Build*.

The following image shows the Application Designer Build Progress status pane. It contains a summary of the process completed and a progress bar. Only the Close button is active.



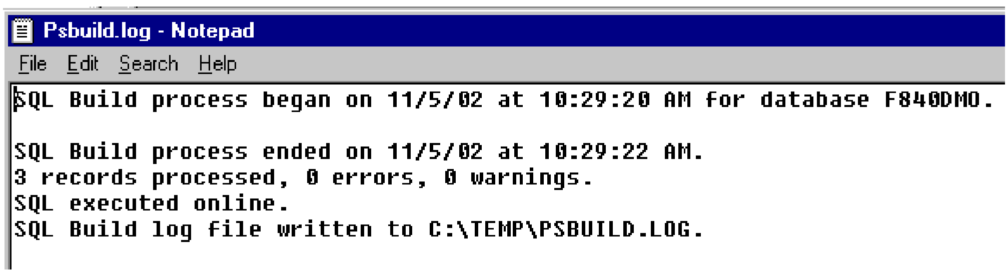
Note: There are no errors or warnings.

a. To ensure that the records were created correctly, use your native SQL Tool to view the records from the generated view.

b. If the records were not correctly generated, click *Close*.

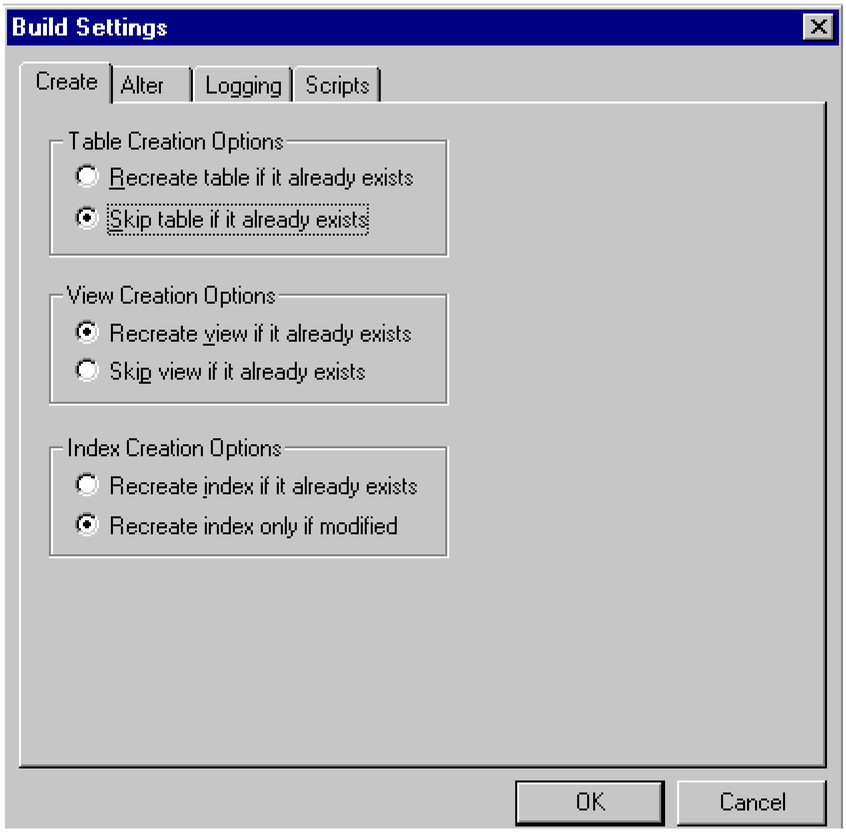
9. Double-click the SQL Build log statement.

The following image shows the PsBuild log file, which contains information about the SQL build process.



10. If you encounter problems, check the Build settings options by choosing *Build* and then, *Settings*.

The following image shows the Build Settings dialog box. It includes four tabs: Create (active), Alter, Logging, and Scripts. The Create tab includes three panes: Table Creation Options, View Creation Options, and Index Creation Options. Each pane includes two options. One is to recreate the table, view, or index if it already exists. The other is to skip the table or view if it already exists or to recreate the index only if modified. The dialog box also includes OK and Cancel buttons.



Depending on the application server database for PeopleSoft 8, a database may require the Tablespace name. For more information regarding this function, consult your PeopleSoft 8 database administrator.

You have finished importing and building the Component Interfaces. To configure security for Component Interfaces, see *Configuring Component Interface Security* on page 2-8.

Configuring Component Interface Security

Application Explorer requires the custom Component Interfaces that you imported and built in the previous procedure, *Import and Build the Component Interfaces* on page 2-3, so you must ensure that all Application Explorer users have access to these Component Interfaces. As with all PeopleSoft objects, security is assigned at the Permission List level. Review your site security requirements to determine the users who will work with Application Explorer and then, set Component Interface security for each distinct Permission List belonging to those users.

Note: These Component Interfaces are required for creating schemas and Integration Business Services, and they are required at run time for using the Find method. They have only Get and Find access and cannot be used to update your PeopleSoft database; this minimizes possible security exposures.

In PeopleSoft release 8.1, you may set security in 2-, 3-, or 4-tier mode; in release 8.4, you may set security in 4-tier mode only.

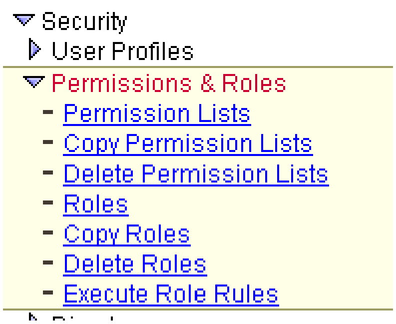
The following procedure describes how to configure security for all supported releases of PeopleSoft in all supported modes. The images in the procedure reflect PeopleSoft release 8.4 in 4-tier mode.

Procedure: How to Configure Component Interface Security

To configure security for each adapter Component Interface:

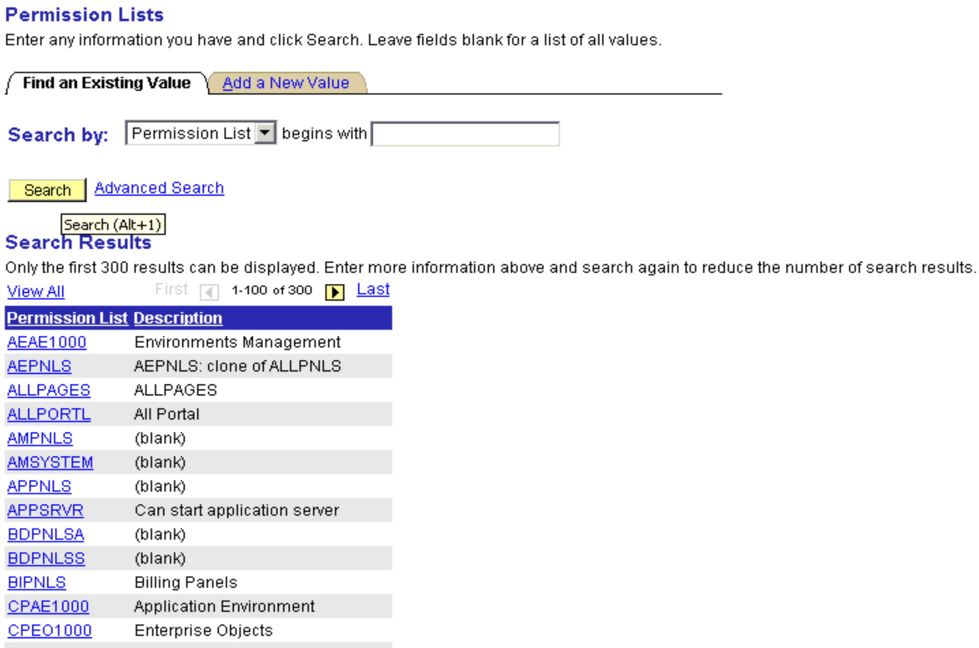
1. From the PeopleSoft user interface menu, choose *PeopleTools*, *Security*, *User Profiles*, *Permissions & Roles*, and then *Permission Lists*.

   The following image shows the expanded Security menu, which displays a list of options under Permissions & Roles.

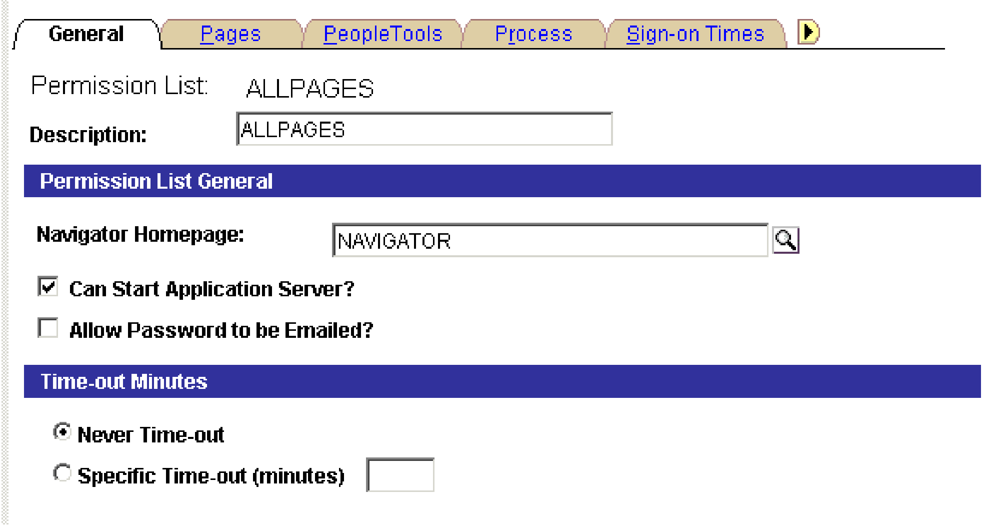
2. Click *Permission Lists*.

The following image shows the Permission Lists pane. It contains two tabs, Find an Existing Value (active) and Add a New Value, a Search by drop-down list, an input field called “begins with,” a Search button, an Advanced Search hyperlink, and a Search Results section.



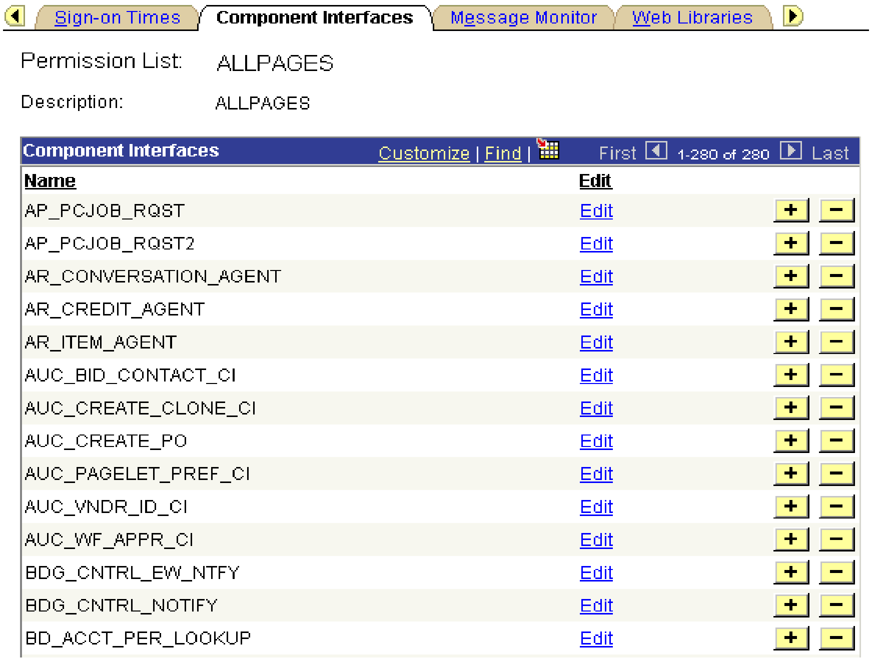
3. Click *Search* and select the relevant Permission List.

The following image shows that the Permission List appears on the General tab which is active. The other tabs are Pages, PeopleTools, Process, and Sign-on Times.



4. To display the Component Interfaces tab, click the arrow to the right of the Sign-on Times tab.

The following image shows the Component Interfaces tab with a list of interfaces. In addition to the Sign-on Times tab, the Message Monitor and Web Libraries tabs appear but are inactive.



a. To add a new row to the Component Interfaces list, select the plus sign (+).

b. Type or select the *IWY_CI_ATTRIBUTES* Component Interface.

5. Click *Edit*.

The Component Interface Permissions pane for the IWY_CI_ATTRIBUTES component interface opens. It includes a column for Methods and drop-down lists for the Method Access including ones corresponding to Get and Find.

a. To set the Get and Find methods to Full Access, click *Full Access (All)*.

b. Click *OK*.

6. Repeat the process for the IWY_CI_MESSAGES Component Interface.

7. Scroll to the bottom of the Component Interfaces pane and click *Save*.

You have finished configuring security for the Component Interfaces delivered with the BEA WebLogic Adapter for PeopleSoft. To test these Component Interfaces, see *Testing the Component Interfaces* on page 2-12.

Testing the Component Interfaces

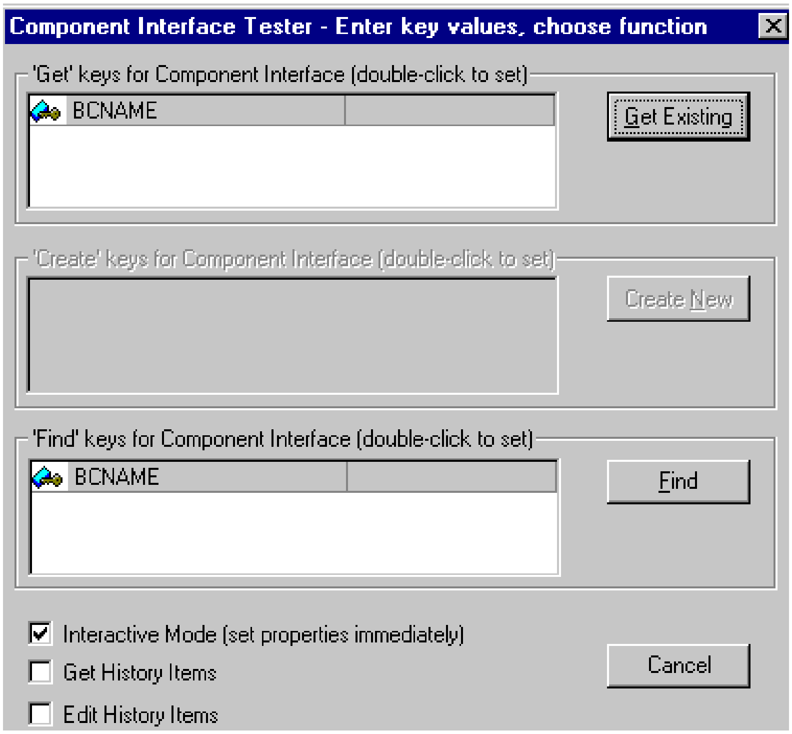
You must test each of the PeopleSoft Component Interfaces before using them with the adapter.

Procedure: How to Test the Component Interfaces

To test the Component Interfaces:

1. In PeopleSoft Application Designer, open the *IWY_CI_ATTRIBUTES* Component Interface.
2. Choose *Tools* and then, *Test Component Interface.*

   The following image shows the Component Interface Tester dialog box. It contains three panes: 'Get' keys, 'Create' keys (inactive), and 'Find' keys all for Component Interface. It includes three check boxes: Interactive Mode, Get History (selected), and Edit History Items. It also includes four buttons: Get Existing (active), Create New (inactive), Find button, and Cancel.

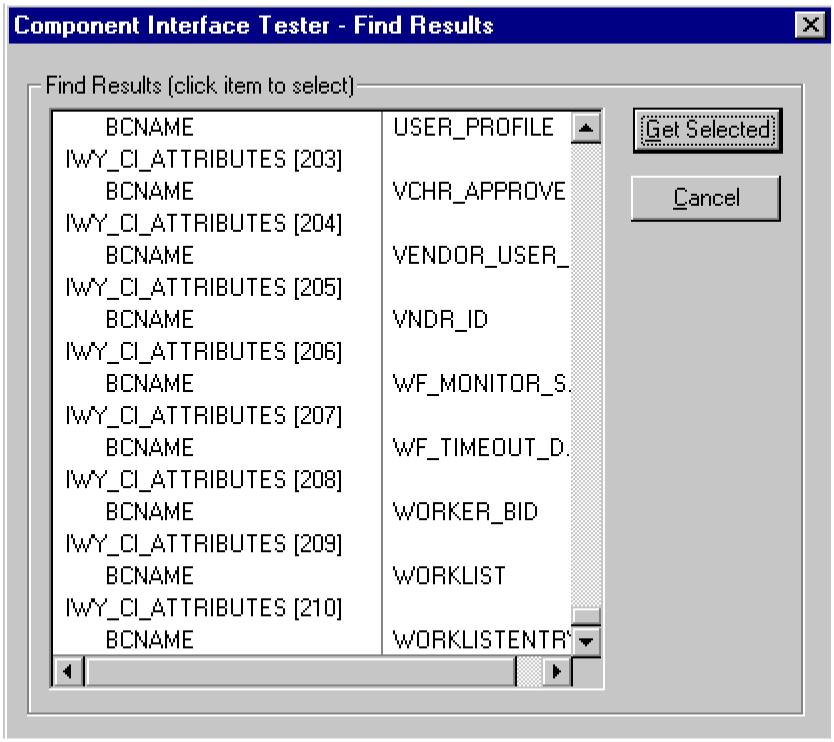


   Note: The Create New button is inactive because the Add method is not applicable to this Component Interface.

3. Click the *Find* button.

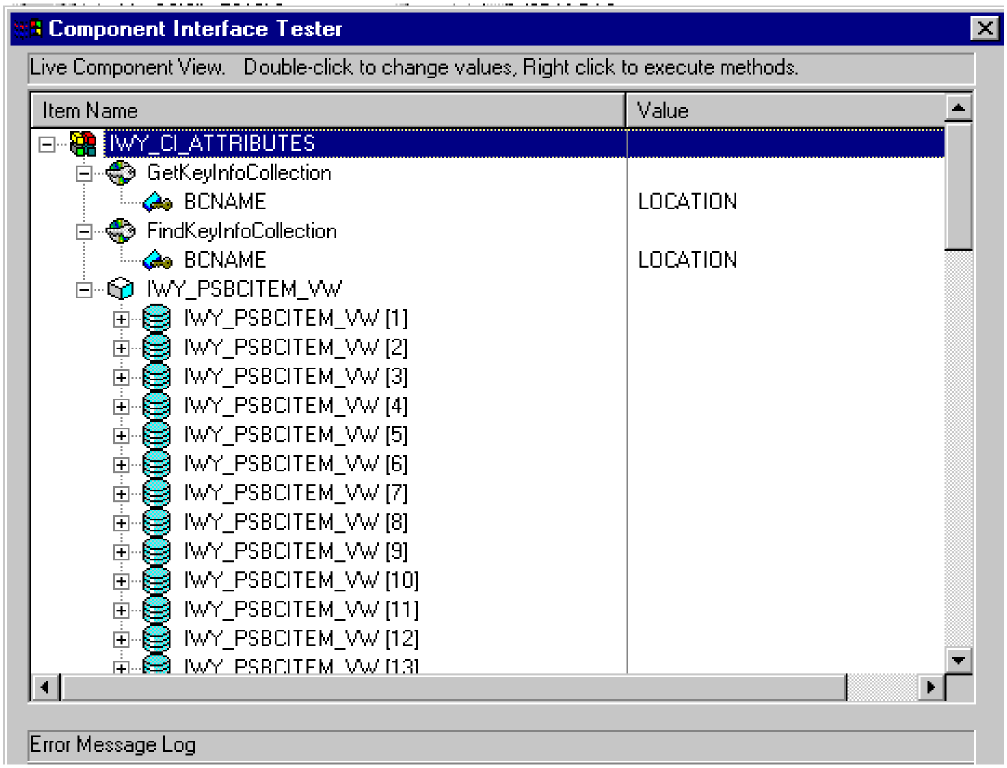
   Entries for the underlying component appear. A message may appear stating that display is limited to a certain number of entries. This is not a significant limitation.

   The following image shows the Component Interface Tester - Find Results dialog box. It contains a Find Results field and Get Selected and Cancel buttons.



4. Highlight one of the lines with its corresponding key in the Find Results window and click the *Get Selected* button.

The relevant data for the selected key appears as shown in the following image of the Component Interface Tester. The Tester includes an Item Name pane and a Value pane. The image shows the values for the keys, GetKeyInfoCollection and FindKeyInfoCollection, and the list of IWY_PSBCITEM_VW databases. It also contains an Error Message Log pane.

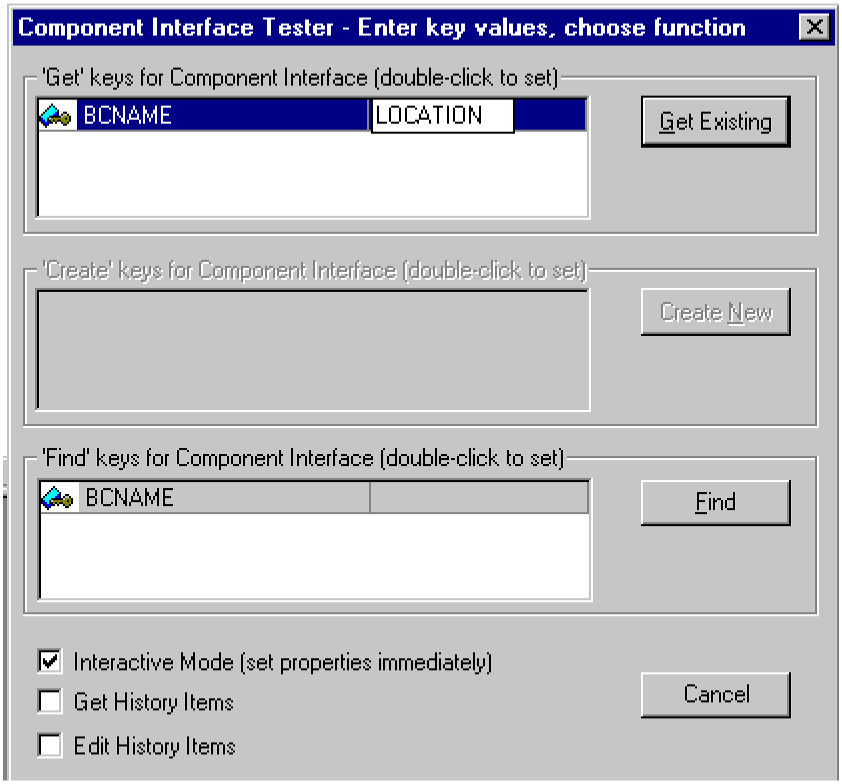


When this window appears, the Component Interface was successfully tested for the Find method.

5. In the Component Tester Interface dialog box, click the *Get* button.

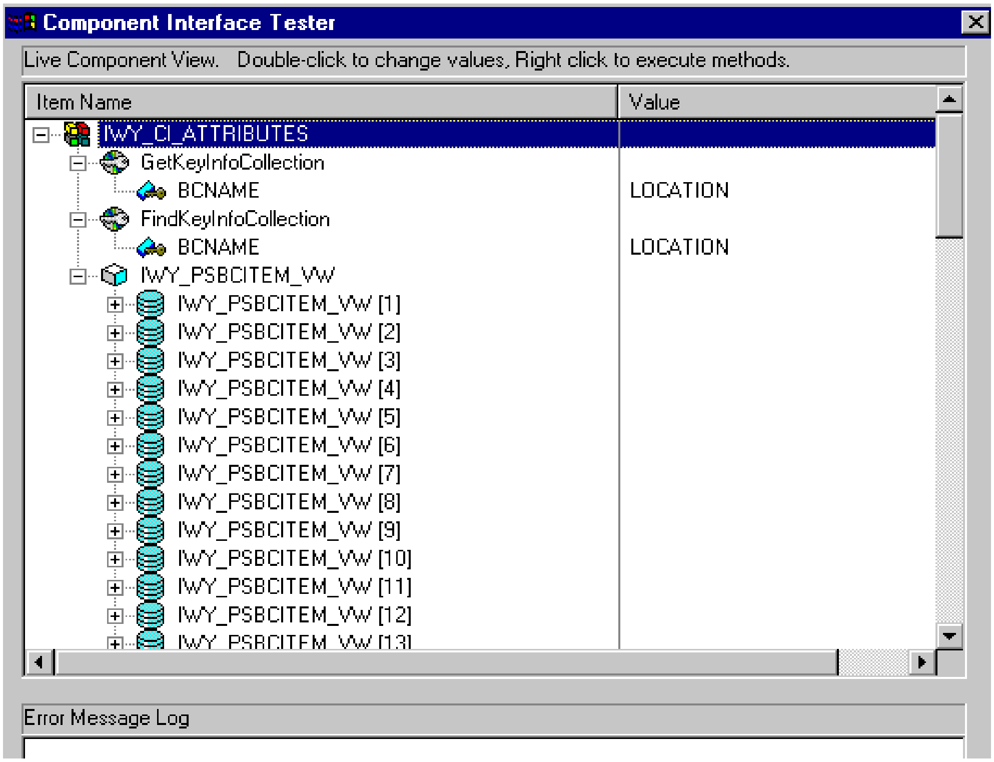
   For the Get method, an existing key must be entered.

The following image shows the Component Interface Tester dialog box. It contains three panes: 'Get' keys, 'Create' keys (inactive), and 'Find' keys, all for Component Interface. It also includes three check boxes: Interactive Mode, Get History (selected), and Edit History Items. It includes four buttons: Get Existing (active), Create New button (inactive), Find button, and Cancel.



The exposed properties for the key that is entered are returned.

The following image shows the Component Interface Tester dialog box. It includes an Item Name pane and a Value pane as well as an Error Message Log pane.



If the previous window opens, the Component Interface has been successfully tested for the Get method.

6. Repeat the previous steps for the IWY_CI_MESSAGES Component Interface.

You have finished testing the Component Interfaces.

Installing the TCP/IP Message Router for the BEA WebLogic Adapter for PeopleSoft

To enable PeopleSoft to send an XML event document to iWay components using TCP/IP, you must install the TCP/IP message router required for your PeopleSoft release.

- For Release 8.4, install the TCP/IP target connector.

  For more information, see *Installing the TCP/IP Target Connector for PeopleSoft Release 8.4* on page 2-17.

- For Release 8.1, install the TCP/IP handler.

  For more information, see *Installing the TCP/IP Handler for PeopleSoft Release 8.1* on page 2-17.

Note: If you are not using PeopleSoft messages for event handling, you may skip this topic.

Installing the TCP/IP Target Connector for PeopleSoft Release 8.4

The TCP/IP target connector for PeopleSoft release 8.4 is installed with iWay 5.5. The default location on Windows is:

```
C:\Program Files\iWay55\etc\misc\peoplesoft\iwpsevent84.jar
```

On non-Windows systems, use the corresponding location.

If this location does not exist, contact your distributor for copies of the relevant files.

Procedure: How to Install the TCP/IP Target Connector for PeopleSoft Release 8.4

To install the TCP/IP target connector for PeopleSoft Release 8.4:

1. Extract *TCPIPTARGET84.class* from iwpsevent84.jar. using an extraction utility appropriate for your platform.
2. Port *TCPIPTARGET84.class* to the platform where the PeopleSoft gateway Web server resides.
3. Place *TCPIPTARGET84.class* in the PeopleSoft server target connector directory.

   This directory may vary according to your Web or application server.

Installing the TCP/IP Handler for PeopleSoft Release 8.1

The TCP/IP target connector for PeopleSoft release 8.1 is installed with iWay 5.5. The default location on Windows is:

```
C:\Program Files\iWay55\etc\misc\peoplesoft\iwpsevent81.jar
```

On non-Windows systems, use the corresponding location.

If this location does not exist, contact your distributor for copies of the relevant files.

Procedure: How to Install the TCP/IP Handler for PeopleSoft 8.1

To install the TCP/IP Handler for PeopleSoft release 8.1:

1. Port *iwpsevent81.jar* to the platform where the PeopleSoft gateway Web server resides.
2. Place *iwpsevent81.jar* in the servletclasses directory under the PeopleSoft Web server.
3. Extract the embedded class files.

Example: Installing the TCP/IP Handler on a UNIX System

To install the TCP/IP handler for PeopleSoft release 8.1 on a UNIX system:

1. Log on to the UNIX system with the proper PeopleSoft ID and permissions.
2. Navigate to the PeopleSoft Web servlets directory.

   This directory may vary by release and by Web server, but usually is:

   ```
   $PS_HOME/webserv/servletclasses
   ```

3. To extract the class files required by PeopleSoft, issue the JAR command, for example:

   ```
   jar -xvf /tmp/iwpsevent81.jar
   ```

The following output appears on a Sun/Solaris system:

```
$ jar -xvf /tmp/iwpsevent81.jar
created: META-INF/
extracted: META-INF/MANIFEST.MF
extracted: psft/pt8/tcphandler/TCPIPHandler81$Entry.class
extracted:
psft/pt8/tcphandler/TCPIPHandler81$HandlerEntry.class
extracted:
psft/pt8/tcphandler/TCPIPHandler81$PublicationHandler.class
extracted: psft/pt8/tcphandler/TCPIPHandler81.class
$
```

Note: The files are placed in a new directory, tcphandler, under psft/pt8.

Copying PeopleSoft Files Into the Lib Directory

Application Explorer creates XSD schemas and Integration Business Services from PeopleSoft Component Interfaces and creates XSD schemas from PeopleSoft messages. To do so, the following file(s) must be in the iWay55\lib directory. The default location for this directory on Windows is:

```
C:\Program Files\iWay55\lib
```

On non-Windows systems, use the corresponding location.

You must ensure that the following are in the lib directory:

- PeopleSoft Java Object Adapter (psjoa.jar)

  This file provides a low level interface between client applications and PeopleSoft. This file is provided with PeopleSoft and can be found in the following directory:

  `PS_HOME\web\PSJOA`

  where:

  PS_HOME
  Is the PeopleSoft home directory.

  The psjoa.jar file is different for every version of PeopleSoft. When you upgrade your Peopletools release, ensure you copy the psjoa.jar file for the new release into the iWay55\lib directory and restart all components.

- pstools.properties (for PeopleSoft 8.1.x)

  PeopleSoft release 8.1x requires an additional file, pstools.properties found in the following directory:

  `PS_HOME\web\jmac`

- psoftcrmci.jar file

  This file is a set of Java classes that were generated from PeopleSoft Component Interfaces. For more information, see Chapter 3, *Generating Component Interface APIs*.

CHAPTER 3

Generating Component Interface APIs

Topics:

- Building the PeopleSoft API Java Programs
- Compiling the PeopleSoft API Java Programs

This section describes how to build and compile Component Interface APIs to use with the BEA WebLogic 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.

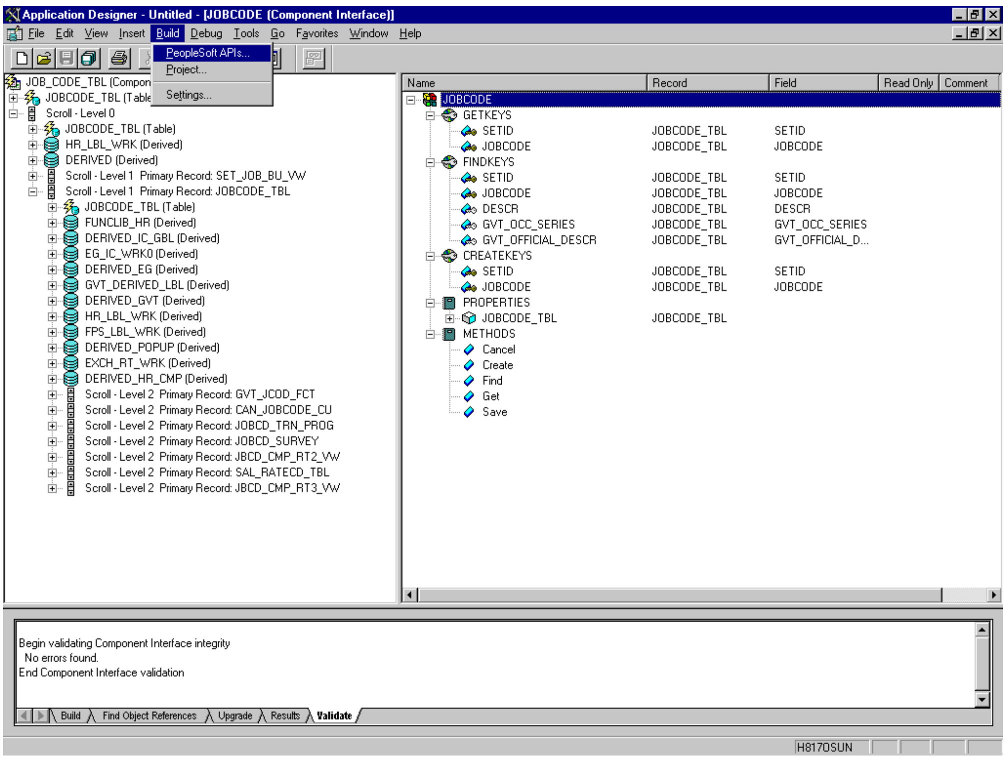
Before using your Component Interface, you must apply security to it and test it. For information about these tasks, as well as how to create a Component Interface, see Appendix A, *Using Component Interfaces*.

Procedure: How to Create a PeopleSoft API Java Program

To create a PeopleSoft API Java program:

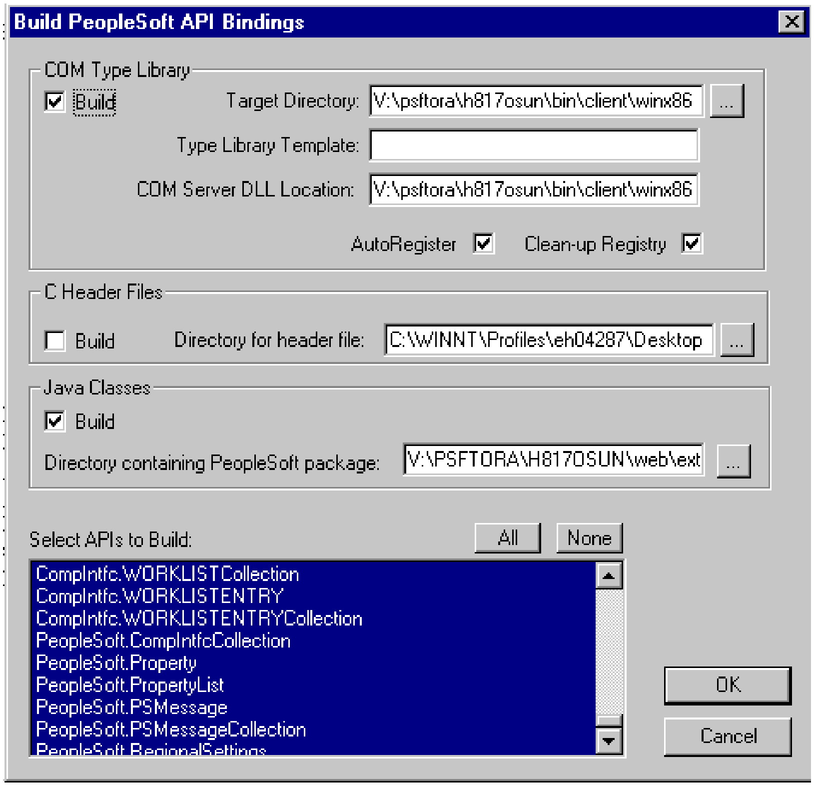
1. Open the PeopleSoft Application Designer.

   The following image shows the PeopleSoft Application Designer Component Interface. The upper part is divided into two panes, and lower part displays a pane for viewing validation. The Build menu is open and the PeopleSoft APIs option is selected.



2. To open a Component Interface, click the right pane and from the Build menu, select *PeopleSoft APIs*.

The Build PeopleSoft API Bindings dialog box opens where you can select options for the COM Type Library and Java Classes in their respective panes as shown in the following image. Currently, the Build check box is selected in the COM Type Library pane which includes the following fields: Target Directory, Type Library Template, and COM Server DLL Location. The AutoRegister and Clean-up Registery check boxes are selected. The dialog box also contains the C Header Files pane that includes a Build check box (clear) and the Directory for Header file field.



a. Because you are creating Java files, clear the *Build* check box for COM Type Library.

b. If it is not already selected, select the *Build* check box in the Java Classes pane and then, select a directory on your local machine where the Java files are to be placed, for example, c:\psoft8_components.

3. To build all files, follow the steps in *How to Build All of the API Files on page 3-4*. To build APIs for specific Component Interfaces, follow the steps in *How to Build APIs for a Specific Component Interface* on page 3-5.

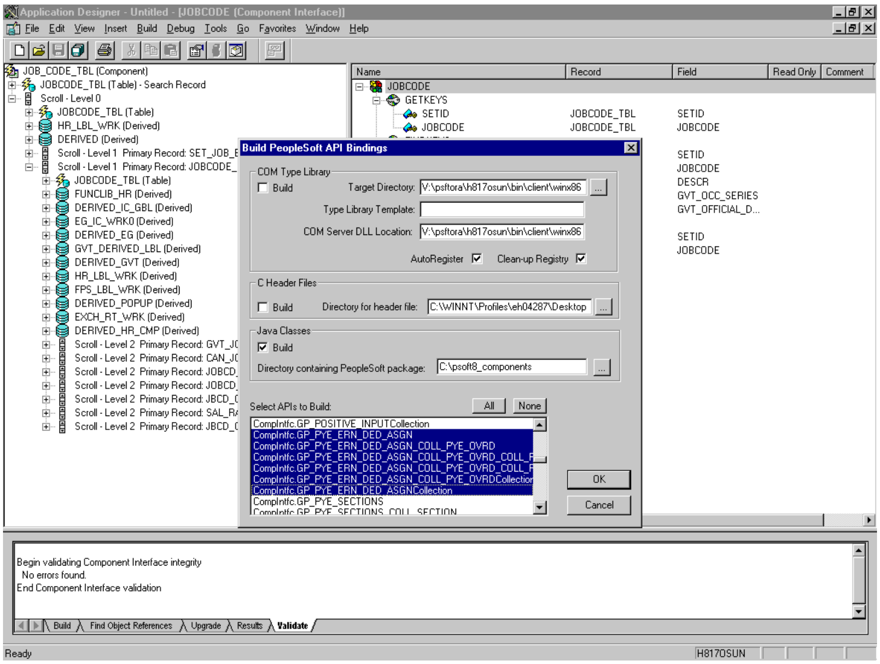
Procedure: How to Build All of the API Files

To build all files:

1. Complete the steps in *How to Create a PeopleSoft API Java Program* on page 3-2. In the Build PeopleSoft API Bindings dialog box:
   a. Select the Build check box for Java Classes.
   b. Specify a directory on your local machine where the Java files are to be placed, for example, c:\psoft8_components.
2. In the same dialog box, select the default, *All* (potentially a large number).
3. Click *OK*.

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

   The following image shows the GP_PYE_ERN_DED_ASGN Component Interface from the HR 8.1 application. It shows the Build PeopleSoft API Bindings dialog box, with the Build check box in the Java Classes pane selected, and the location of the Java files. The list of APIs appears, and the All, None, OK, and Cancel buttons are active.



You are ready to compile the Java files. For more information, see *Compiling the PeopleSoft API Java Programs* on page 3-7.

Procedure: How to Build APIs for a Specific Component Interface

To build APIs for a specific Component Interface or interfaces:

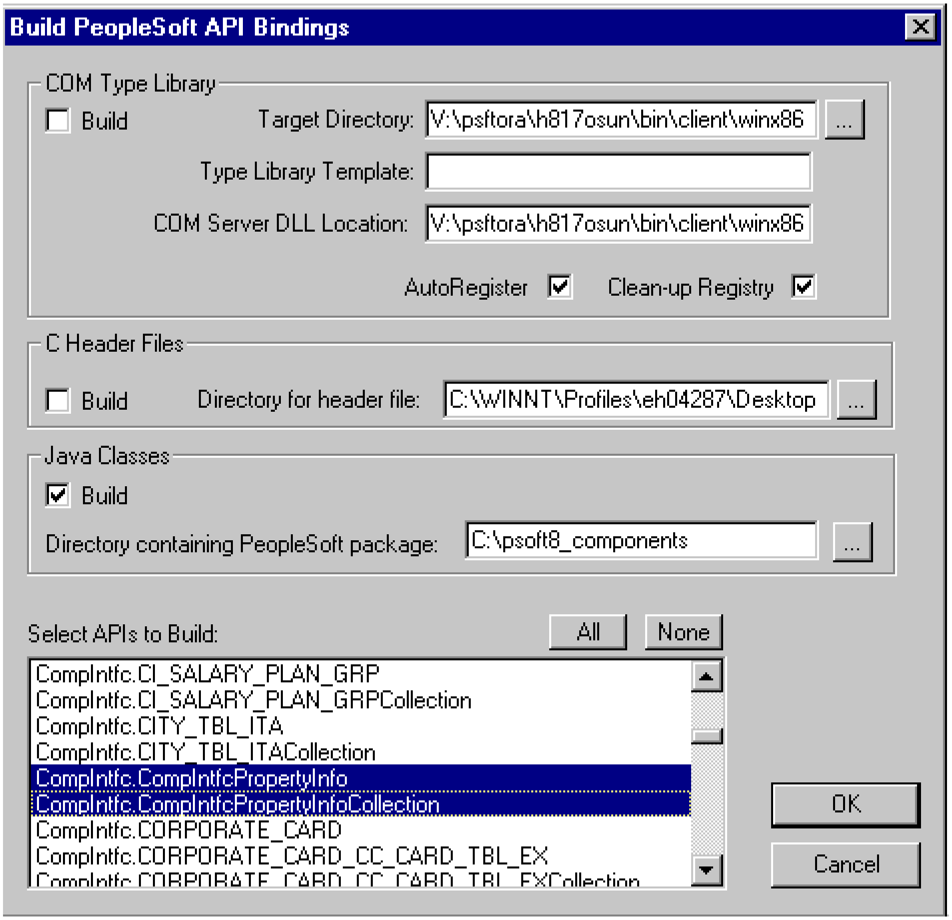
1. Complete the steps in *How to Create a PeopleSoft API Java Program* on page 3-2. In the Build PeopleSoft API Bindings dialog box:
   - **a.** Select the Build check box for Java Classes.
   - **b.** Specify a directory on your local machine where the Java files are to be placed, for example, c:\psoft8_components.
2. In the In the same dialog box, select, click *None*.

   Selecting None clears the selected APIs, so you can select the appropriate ones for your Component Interface. There may be fewer than five, or more than 50 APIs, for a particular Component Interface. The APIs begin with the name of your Component Interface.
   - **a.** Select the APIs for your Component Interface.
   - **b.** 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
3. Select these properties in the same step as the Component Interface build process or select them separately.

The following image shows the Build PeopleSoft API Bindings dialog box, with the Build check box in the Java Classes pane selected, and the location of the Java files. The list of APIs and the generic properties appear, and the All, None, OK, and Cancel buttons are active.



4. 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 ready to compile the Java files. For more information, see *Compiling the PeopleSoft API Java Programs* on page 3-7.

Compiling the PeopleSoft API Java Programs

PeopleSoft places the Java programs to compile in the directory called

psoft8_components\PeopleSoft\Generated\CompIntfc

where:

psoft8_components

Is the directory specified during the build process.

If you chose to generate all APIs, the systems creates a second directory, that you are not required to access, called

psoft8_components\PeopleSoft\Generated\PeopleSoft

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.

- To compile the PeopleSoft API Java programs 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.

  For more information, see *Building the PeopleSoft API Java Programs* on page 3-2.
- To compile the PeopleSoft API Java programs on **a machine other than the one where you installed Application Explorer**, see *How to Compile the PeopleSoft API Java Programs on Another Machine* on page 3-7.

Note: There are two Java programs for every API file that you selected when you built the Java programs. For more information, see *Building the PeopleSoft API Java Programs* on page 3-2.

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. For more information, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual.

Procedure: How to Compile the PeopleSoft API Java Programs on Another Machine

Note: If you are running on UNIX, do the compile and JAR steps on Windows NT 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 NT to UNIX, the file format must be set to binary.

To compile the PeopleSoft API Java programs 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.
2. Ensure that the psjoa.jar file is in the Java class path before you compile the programs.

3. Compile the Java programs and ensure that you include the following path:

   ```
   \PeopleSoft\Generated\CompIntfc
   ```

   Note: The path is case-sensitive.

   The following Windows NT 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
   ```

   where:

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

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

4. Compress the class files into a JAR file.

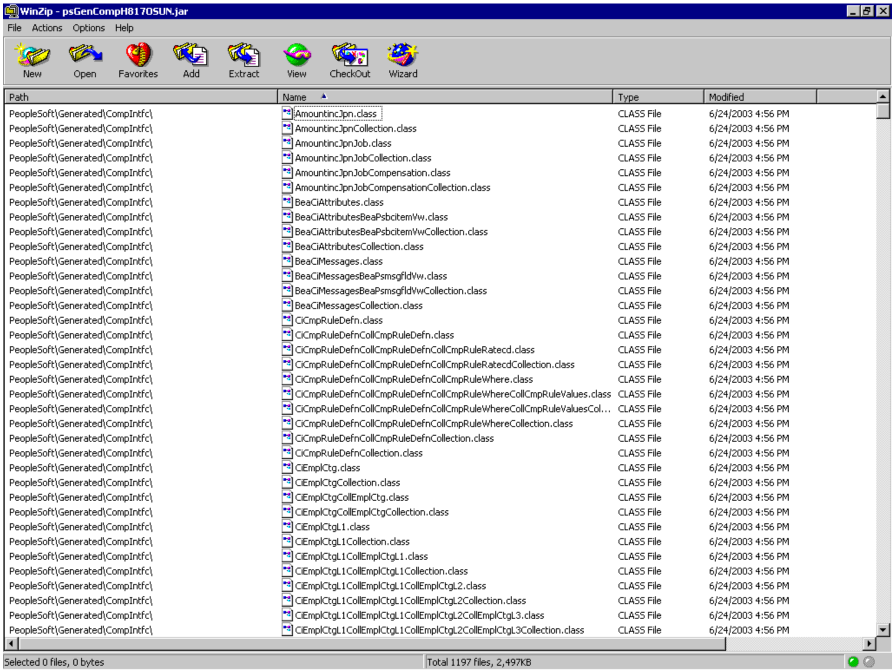
   The following Windows BAT file, if run from the psoft8_components directory, creates a correct JAR file:

   ```
   @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.

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

The following image shows the PeopleSoft JAR files opened in the WinZip application.



6. If the JAR file does not use the following case-sensitive path, you must go back and correct it:

   `PeopleSoft\Generated\CompIntfc\`

7. Place the JAR file in the common lib directory.

   This enables the BEA WebLogic Adapter for PeopleSoft to communicate with the PeopleSoft Component Interface.

   For the current Windows NT version of the product, the default location is

   `iway55\lib`

   For UNIX, the location is

   `iway55/lib`

   where:

   `iway55`

   Is the full path to your iWay installation.

   For more information, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual.

CHAPTER 4

Configuring the PeopleSoft Message Router

Topics:

- Configuring the TCP/IP or HTTP Target Connector for PeopleSoft 8.4
- Configuring the TCP/IP Handler for PeopleSoft 8.1
- Testing Your PeopleSoft Configuration

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

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 TCP/IP message routing software, provided with the BEA WebLogic Adapter for PeopleSoft, passes XML documents from PeopleSoft Integration Gateway to your application server.

The HTTP Outbound Connector, provided by PeopleSoft, may be used in place of the iWAY TCP/IP84 Connector in release 8.4.

This configuration topic assumes you are familiar with PeopleSoft Integration Broker (in release 8.4). If not, see Appendix B, *Using PeopleSoft 8 Integration Broker*, for basic information about configuring and testing. For a complete description *before* you work with the BEA WebLogic Adapter for PeopleSoft, see your PeopleSoft documentation.

The procedures in this topic assume that your Integration Broker environment is configured and tested. For more information, see Appendix B, *Using PeopleSoft 8 Integration Broker*.

To configure the PeopleSoft 8.4 TCP/IP or HTTP Target Connector to send messages to your application server:

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

   For more information, see *How to Configure the Gateway for the TCP/IP Target Connector* on page 4-3 or *How to Configure the Gateway for the HTTP Target Connector* on page 4-8.

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

2. Configure the node.

   For more information see *How to Configure the Node for the TCP/IP84 Connector* on page 4-5 or *How to Configure the Node to Use the HTTP Connector* on page 4-10.

   Note: Starting with release 8.4, the Integration Broker is delivered with an HTTP Outbound Connector. This connector can be used in place of the iWAY TCP/IP84 Connector for sending messages to your application server.

Procedure: How to Configure the Gateway for the TCP/IP Target Connector

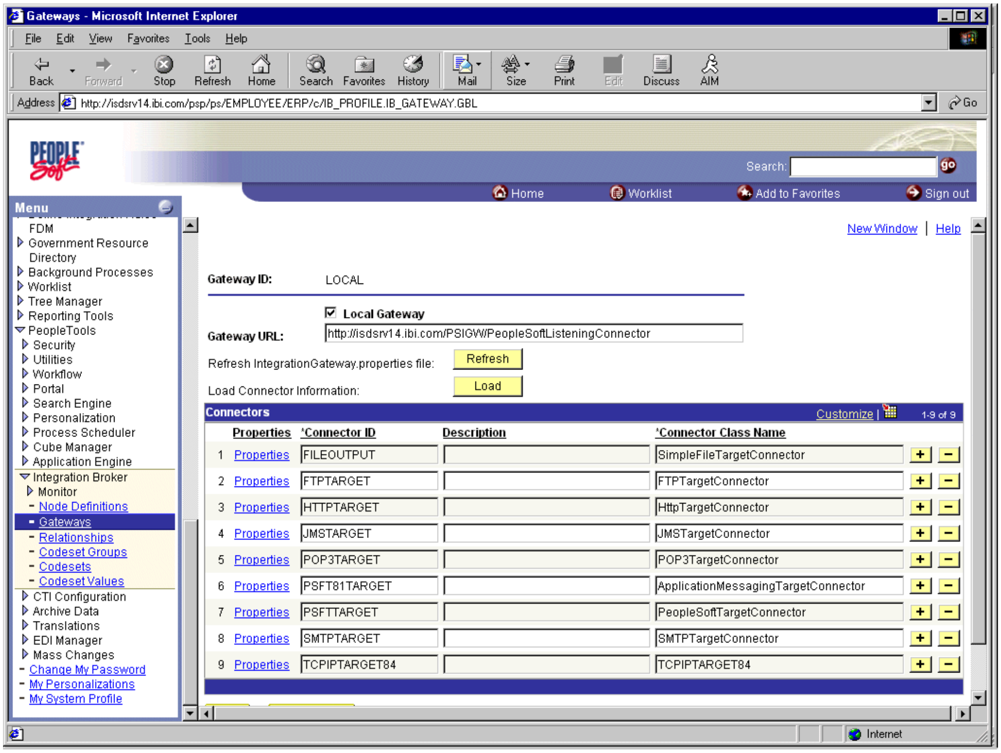
To configure the gateway for the TCP/IP Target Connector:

1. In a Web browser, open your PeopleSoft release 8.4 application.

2. In the Menu pane, expand *PeopleTools*, *Integration Broker,* and then click *Gateways*.

3. Open the *LOCAL Gateway ID.*

The following image shows the PeopleSoft Gateway ID pane on the right. The right pane includes the following: the Local Gateway check box, the Gateway URL field, fields with information for Connector ID, Description, and Connector Class Name, Properties hyperlinks, and the Refresh and Load buttons.

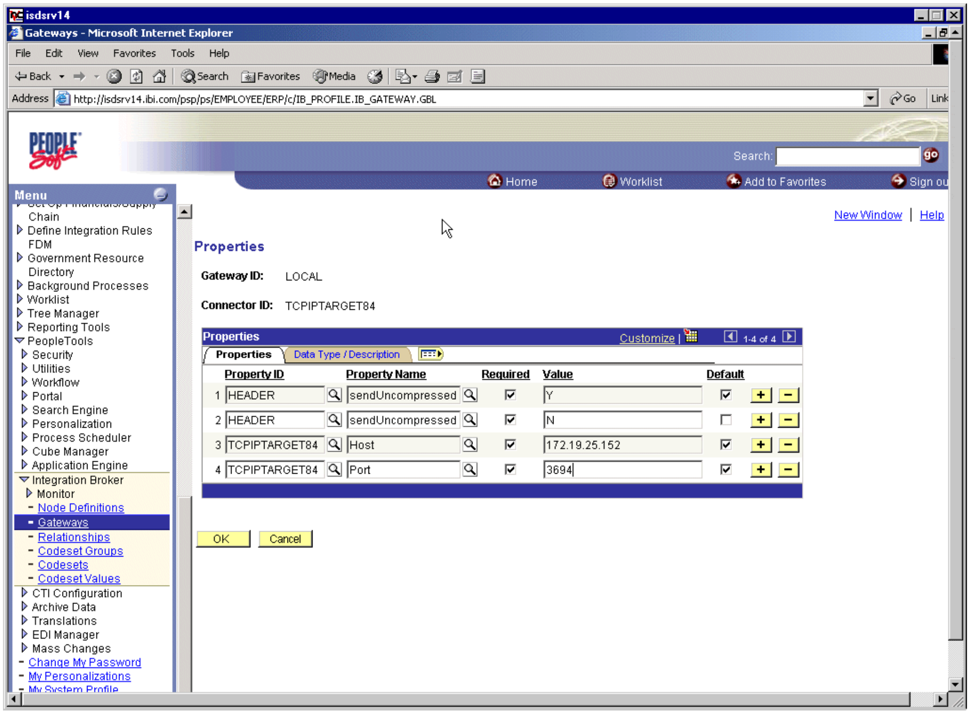


a. If you do not see the TCPIPTARGET84 Connector ID, click *Load* and scroll to locate *TCPIPTARGET84* in the list.

If TCPIPTARGET84 still does not appear, the connector class file was not installed in the Integration Gateway. For information about installing the TCPIPTAGER84 connector, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual.

b. Click the *Properties* hyperlink for TCPIPTARGET84.

The following image shows the Properties pane for TCPIPTARGET84. Gateways is selected in the Menu pane. The right pane contains the following components: the Gateway ID, Connector ID, and OK and Cancel buttons.



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

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

4. Click *OK*.

The Gateway window opens.

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

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

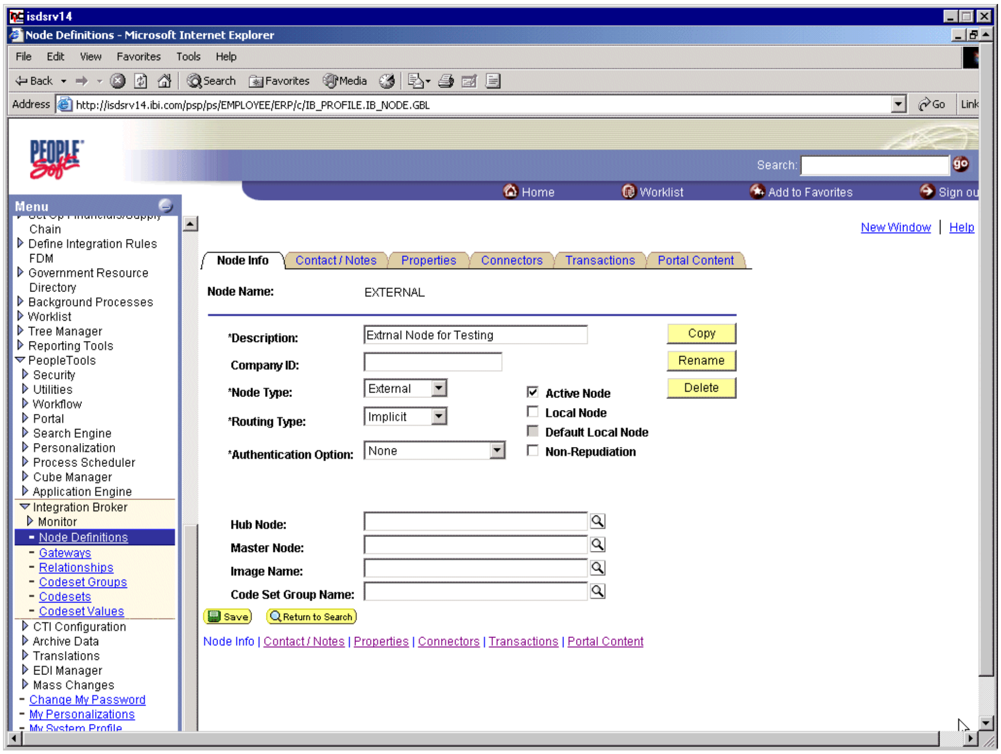
Procedure: How to Configure the Node for the TCP/IP84 Connector

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 B, *Using PeopleSoft 8 Integration Broker* or your PeopleSoft documentation.

   The following image shows the Node Info tab for External node on the right. It includes the following: Description, Company ID, Hub Node, Master Node, Image Name, and Code Set Group Name fields; Node Type, Routing Type, and Authentication Options lists; Active Node, Local Node, Default Local Node, and Non-Repudiation check boxes; and Copy, Rename, and Delete buttons.

   a. From the Node Type drop-down list, select *External*.

   b. From the Routing Type drop-down list, select *Implicit*.

3. Select the *Connectors* tab.

a. Select *TCPIPTARGET84* as the Connector ID.

   Default values appear for the host and the port.

b. Type values for the host and the port for the machine and port that route XML to your application server.

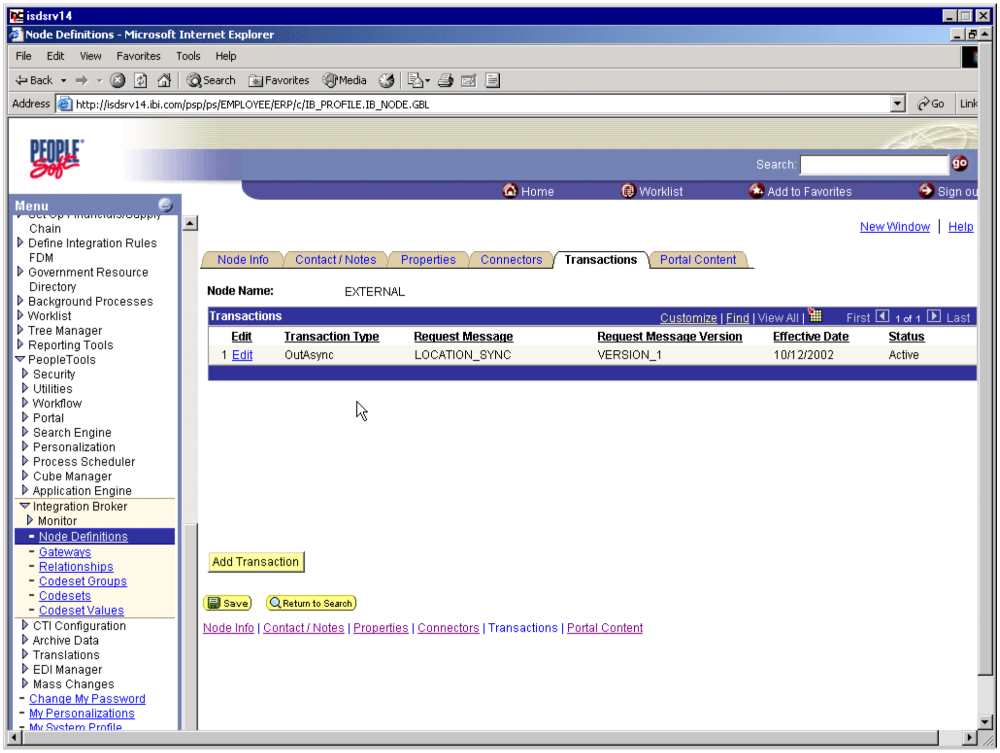
   You can accept or override the default values for individual nodes.

c. Click *Save*.

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

4. Select the *Transactions* tab.

   The following image shows the TCP/IP84 Connector Transaction tab for the External node type. It contains transaction details and the Add Transaction, Save, and Return to Search buttons.

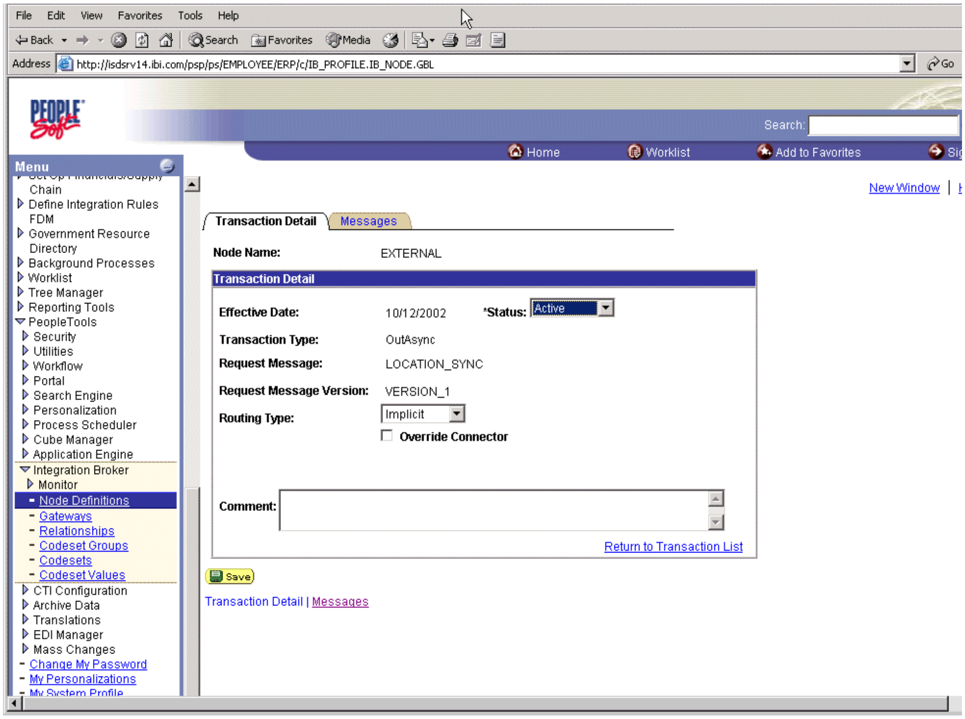


a. If there are no transactions, click *Add Transaction* to add the message with which you are working.

   In this procedure, the node is already configured with the LOCATION_SYNC message.

b. To make the Transaction Detail tab available and view transaction details for the LOCATION_SYNC message, click *Edit*.

The following image shows the TCP/IP84 Connector Transaction Detail tab, which contains information for the following: Node Name, Effective Date, Transaction Type, Request Message, Request Message Version, and Routing Type. The Comment field is currently empty.



You can add the message with which you are working.

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

5. Click *Save* and return to the *Transactions* tab.

a. To edit additional transactions, click the *Edit* hyperlink to navigate to the Transaction Detail tab.

b. In the Transaction Detail tab, from the Status drop-down list, select *Inactive*.

Inactive status is for initial testing only. After you test your configuration, you can change the status to Active and have as many nodes and transactions as required to satisfy your business requirements.

6. Click *Save*.

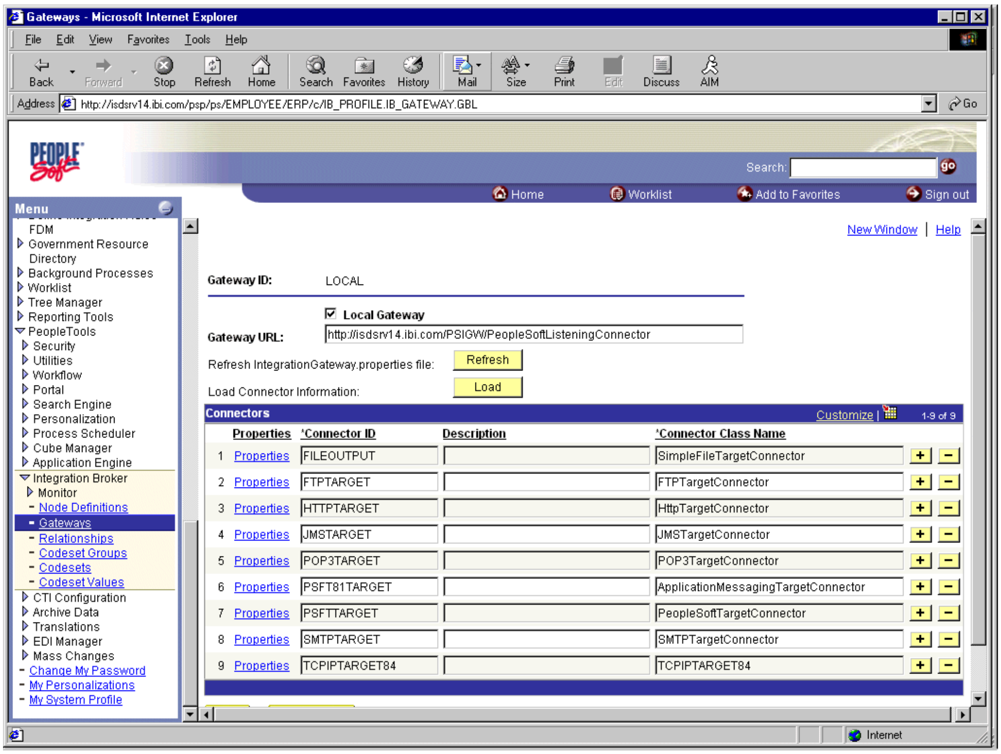
You are ready to send XML messages to your PeopleSoft XML Listener.

Procedure: How to Configure the Gateway for the HTTP Target Connector

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*, *Integration Broker*, and then click *Gateways*.
3. Open the *LOCAL Gateway ID*.

   The following image shows the HTTP Target Connector Gateway ID pane on the right and includes the following: the Local Gateway check box, the Gateway URL field, fields with information for Connector ID, Description, and Connector Class Name, Properties hyperlinks, and the Refresh and Load buttons.

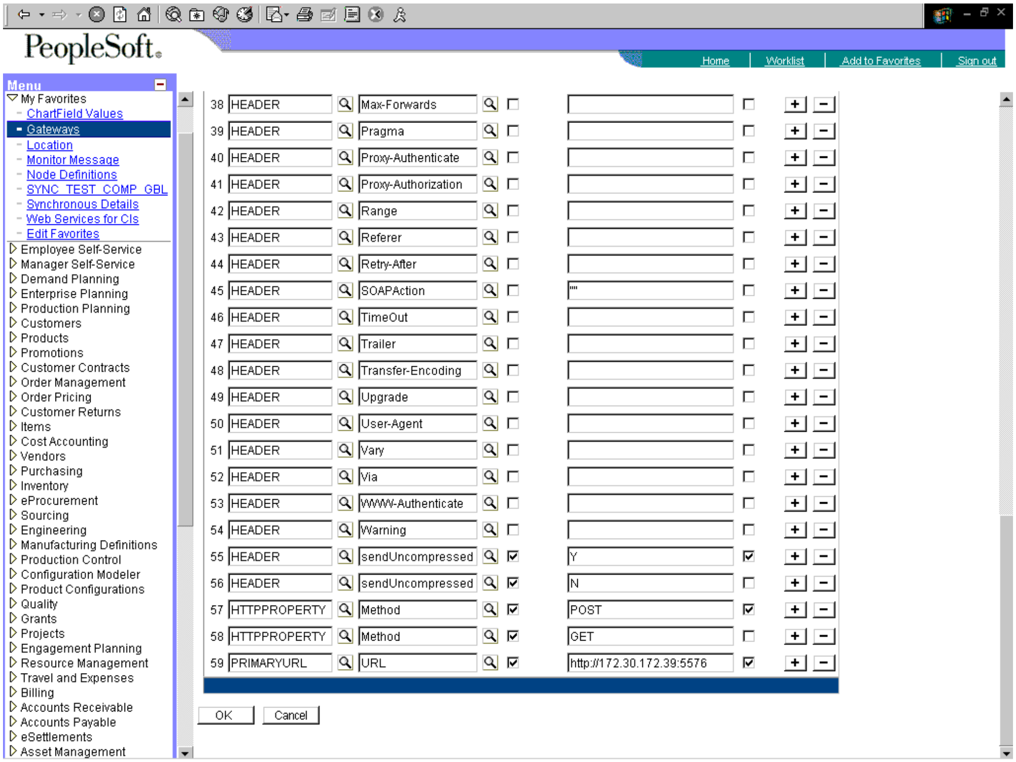
   

   a. If you do not see the HTTPTARGET Connector ID, click *Load*.

   b. If it still does not appear, consult with your PeopleSoft system administrator as your Gateway was not installed properly.

4. Click the *Properties* hyperlink for HTTPTARGET.

The Properties pane for HTTPTARGET opens, as shown in the following image. Gateways is selected in the Menu pane on the left. On the right, information about Headers appears.



5. 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.

6. Click *OK*.

   The Gateway window opens.

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

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

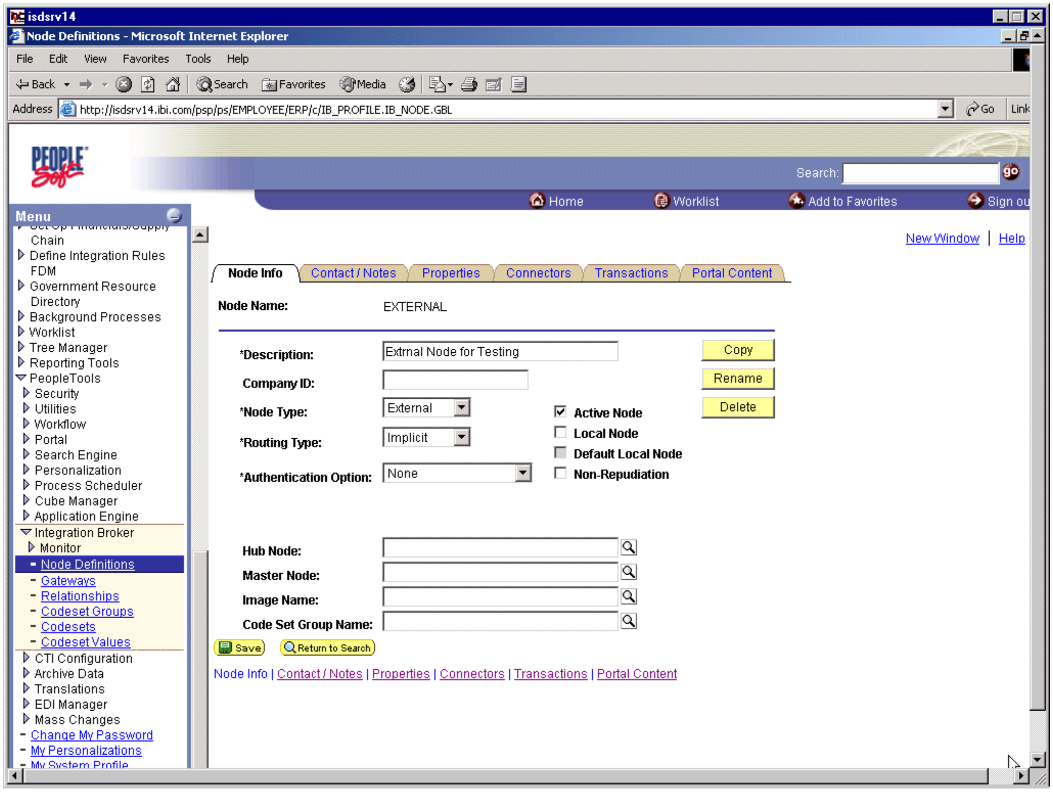
Procedure: How to Configure the Node to Use the HTTP Connector

To configure the node to use the HTTP Connector:

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

   This procedure uses a node called EXTERNAL. For more information about creating and using nodes, see Appendix B, *Using PeopleSoft 8 Integration Broker* or your PeopleSoft documentation.

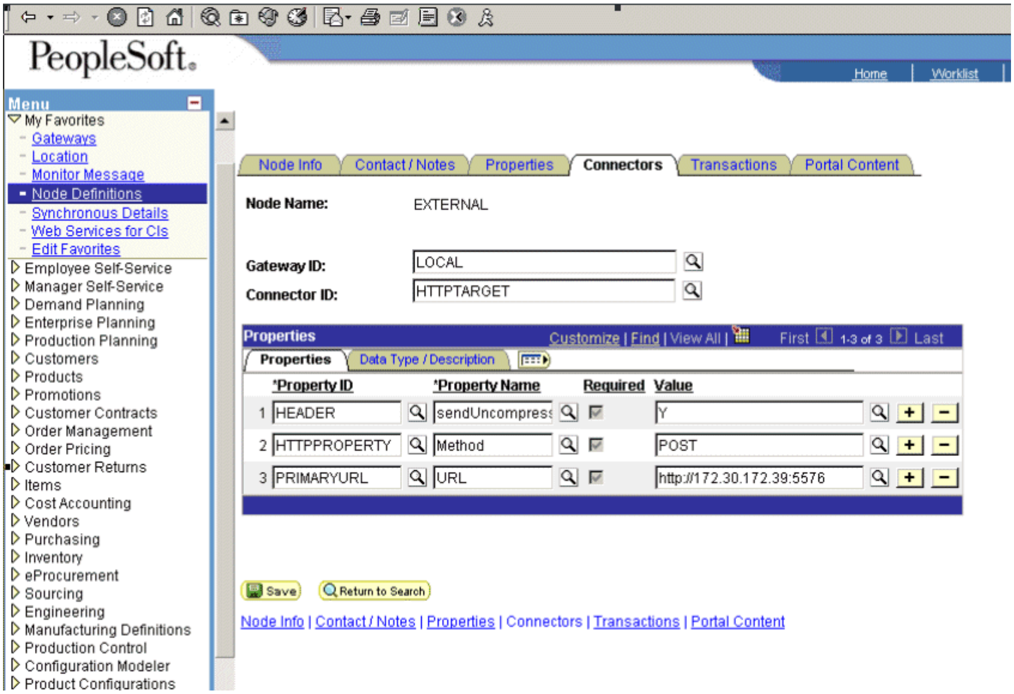
   The following image shows the Node Info tab for External node on the right. It includes the following: Description, Company ID, Hub Node, Master Node, Image Name, and Code Set Group Name fields; Node Type, Routing Type, and Authentication Options lists; Active Node, Local Node, Default Local Node, and Non-Repudiation check boxes; and Copy, Rename, and Delete buttons.



   a. From the Node Type drop-down list, select *External*.
   b. From the Routing Type drop-down list, select *Implicit*.

3. Select the *Connectors* tab.

The following image shows the PeopleSoft Integration Broker Connectors tab for the External node. It includes the following: Gateway ID, Connector ID, Property ID, Property Name, and Value fields and Save and Return to Search buttons.



a. Change the Connector ID to *HTTPTARGET*.

b. Type a value for each property based on the information in the following table.

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

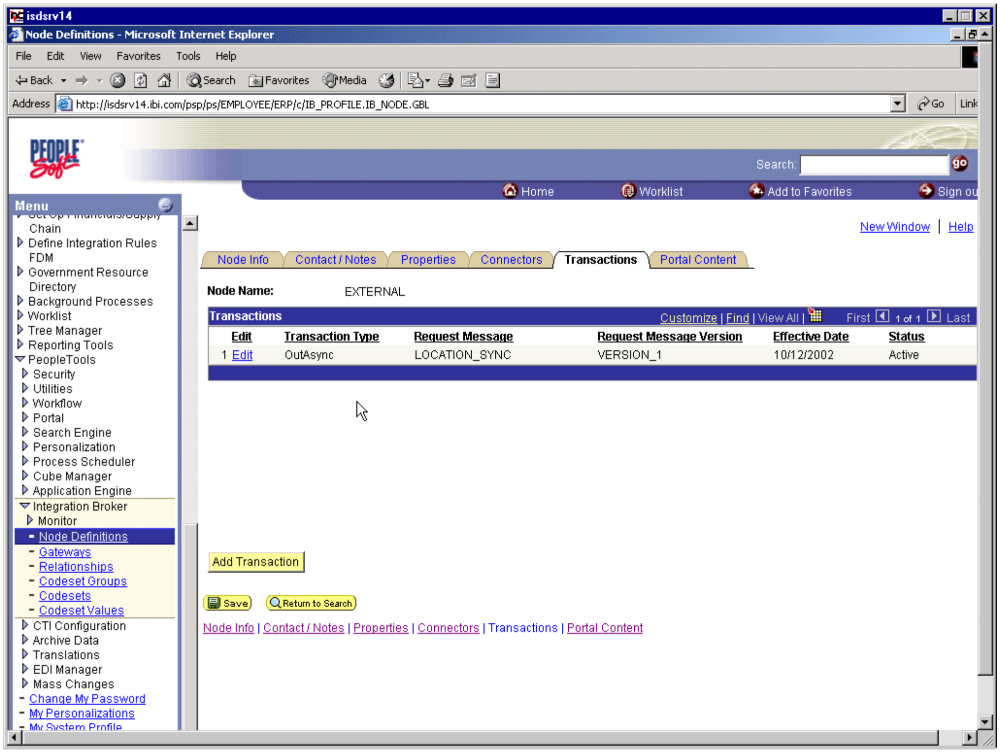
Note: For complex business situations you can configure multiple nodes and multiple listeners.

c. Click *Save*.

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

4. Select the *Transactions* tab.

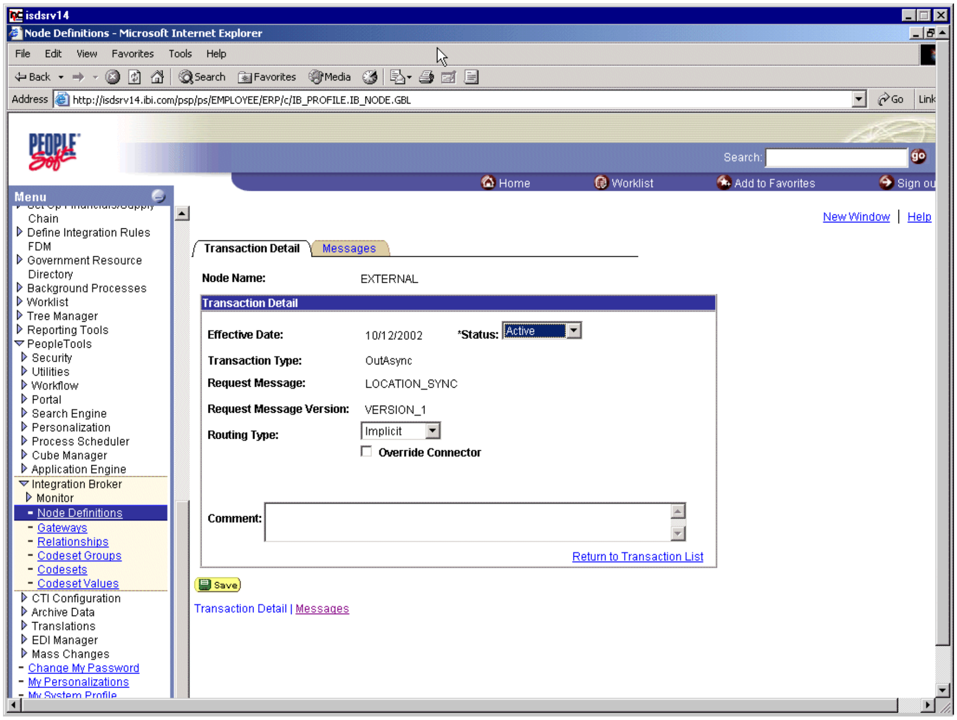
The following image shows the Transactions tab selected. It contains the node name, Transaction information, and the Add Transaction, Save, and Return to Search buttons.



5. If there are no transactions, click *Add Transaction*.

In this procedure, the node is already configured with the LOCATION_SYNC message.

The following image shows the PeopleSoft Integration Broker Transaction Detail tab for the External node. It contains information for the following: Effective Date, Status, Transaction Type, Request Message, Request Message Version, and Routing Type. Currently, the Comment field is empty.



You can add the message with which you are working.

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

b. Click *Save*.

6. Click *Return to 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 can change the status to Active and have as many nodes and transactions as required to satisfy your business requirements.

7. Click *Save*.

You are ready to send XML messages to your PeopleSoft XML Listener.

Configuring the TCP/IP Handler for PeopleSoft 8.1

This configuration topic assumes you are familiar with Application Messaging (in release 8.1) and assumes that your Application Messaging environment is properly configured and tested. If not, see Appendix B, *Using PeopleSoft 8 Integration Broker*, for basic information about configuring and testing. For a complete description before you work with the BEA WebLogic Adapter for PeopleSoft, see your PeopleSoft documentation.

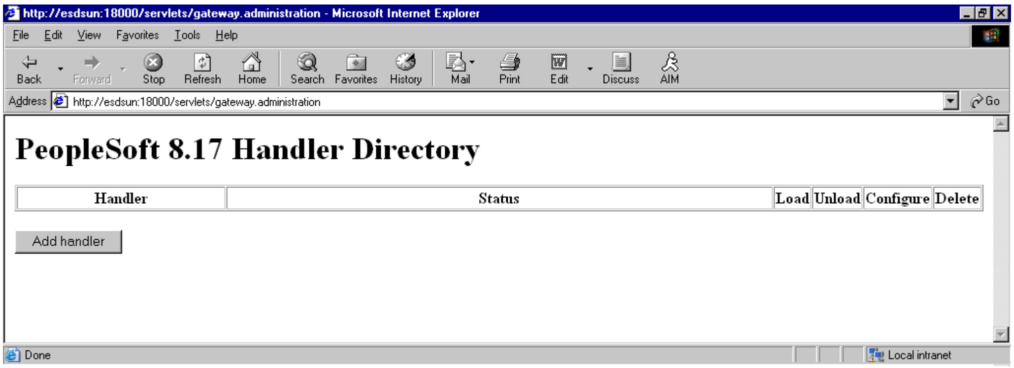
Procedure: How to Configure the TCP/IP Handler for PeopleSoft 8.1

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

   The following image shows an empty PeopleSoft Handler directory with areas to view status, Load, Unload, Configure, or Delete. The Add handler button appears on the left.

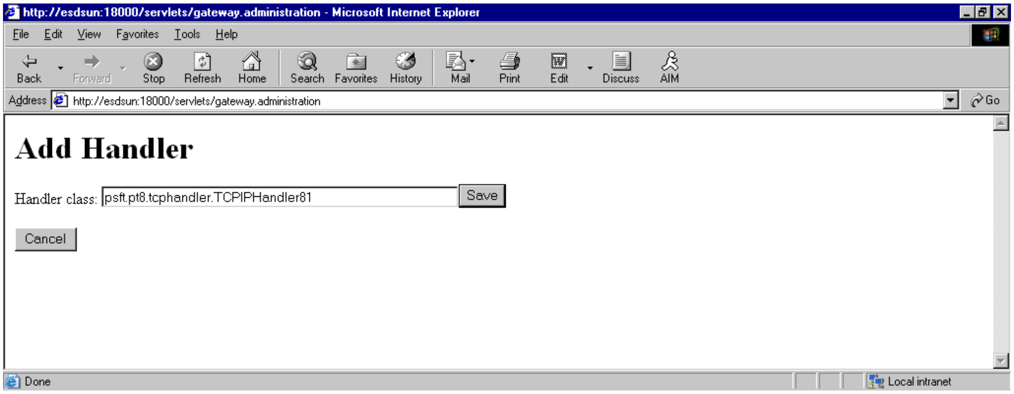
   a. Click *Add handler.*

   b. In the Handler class field, type the full path of TCPIPHandler81 (case-sensitive):

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

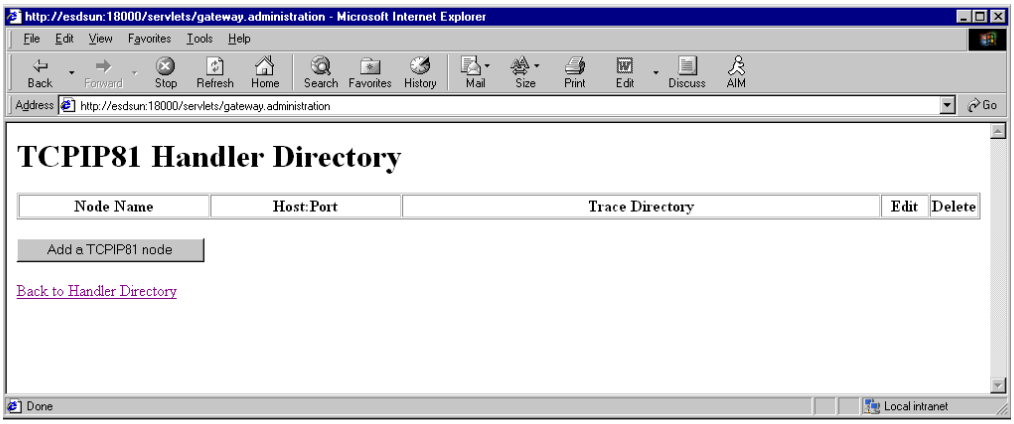
3. Click *Save.*

The following image shows a loaded Add Handler directory.



4. When you return to the PeopleSoft Handler directory, click *Load*.
5. Click *Configure*.

   The following image shows the Add TCPIP81 Handler Directory window. It contains an Add a TCPIP81 node button and a Back to Handler Directory hyperlink.



   a. Click *Add a TCPIP81 node*.

      This procedure uses a node named EXTERNAL. For more information about creating and using nodes, see Appendix B, *Using PeopleSoft 8 Integration Broker* or your PeopleSoft documentation.

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

Field	Value Example	Description
Node Name	EXTERNAL	Name of the TCP/IP node.
Host Name	172.19.25.152	Machine on which your PeopleSoft XML listener is listening for incoming messages.
Port	3694	Port on which your PeopleSoft XML listener is listening for incoming messages.
Trace Directory	/tmp	Directory where a trace file is created when errors occur in message delivery.

The system does not validate your entries.

6. Click *Save*.

7. For your changes to take effect, click *Back to Handler Directory* to return to the PeopleSoft 8.1 Handler Directory window.

8. Click *Unload and re-Load TCPIPHandler81*.

You are now ready to send messages from PeopleSoft to your application server.

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.

You test your configuration to ensure that:

- Your application server is up and running.
- The server name and/or port number for PeopleSoft and your application server match.
- The default page for HTTP exists.

Procedure: How to Test a PeopleSoft Configuration

To test a PeopleSoft configuration:

1. In a Web browser, open your PeopleSoft application.

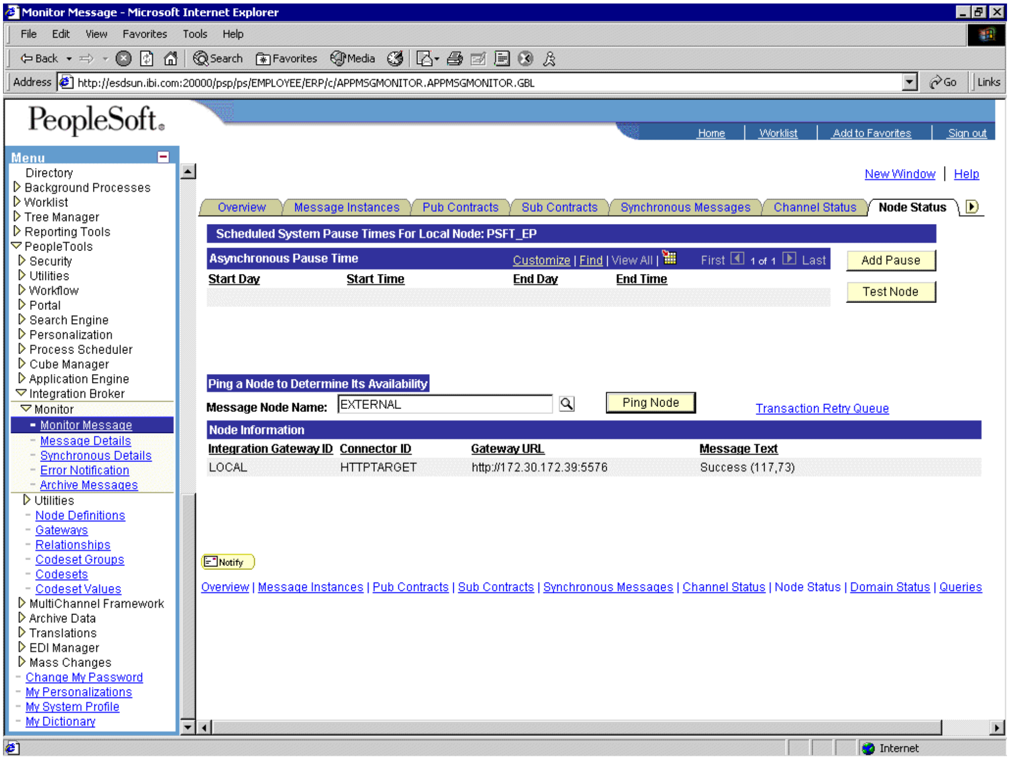
2. Navigate to the message monitoring menu.

For PeopleSoft 8.4: In the Menu pane, expand *PeopleTools, Integration Broker, Monitor,* and then select *Monitor Message.*

For PeopleSoft 8.1: In the Menu pane, expand *Home*, *PeopleTools*, *Application Message Monitor*, *Use,* and then select *Application Message Monitor.*

3. Click the *Node Status* tab.

The following image shows the Node Status tab which contains areas where you can schedule system pause times for the local node, ping a node to determine its availability, and view node information.



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

b. Click *Ping Node.*

If you properly configured both PeopleSoft and your application server, you receive a Success message.

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

CHAPTER 5

Creating XML Schemas or Web Services for PeopleSoft

Topics:

- Overview
- Starting Servlet Application Explorer
- Establishing a Target for PeopleSoft
- Modifying a Target
- Viewing Application System Objects
- Creating an XML Schema
- Generating a Web Service for PeopleSoft

This section describes how to create XML schemas and generate Web services (business services) for PeopleSoft business objects using Application Explorer.

Overview

The BEA WebLogic Adapter for PeopleSoft 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.

For more information on installing and configuring Application Explorer, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual.

Starting Servlet Application Explorer

Before you can use Servlet Application Explorer, you must start your application server.

Procedure: How to Start BEA WebLogic Server on Windows

To start BEA WebLogic Server on Windows:

1. Click the *Start* menu.
2. Select *Programs*, *BEA Products*, and then click *WebLogic Server 9.0*.

Procedure: How to Start BEA WebLogic Server on UNIX

To start BEA WebLogic Server on UNIX or from a command line:

Enter the following at the prompt:

```
BEA_HOME/user_projects/domains/DOMAIN_NAME/startWebLogic.cmd
```

where:

`BEA_HOME`
: Is the directory where BEA WebLogic is installed.

`DOMAIN_NAME`
: Is the name of your domain.

Procedure: How to Open Servlet Application Explorer

To open Application Explorer:

1. Ensure that your application server is running.
2. Enter the following URL in your browser:

   ```
   http://hostname:port/iwae/index.html
   ```

where:

`hostname`

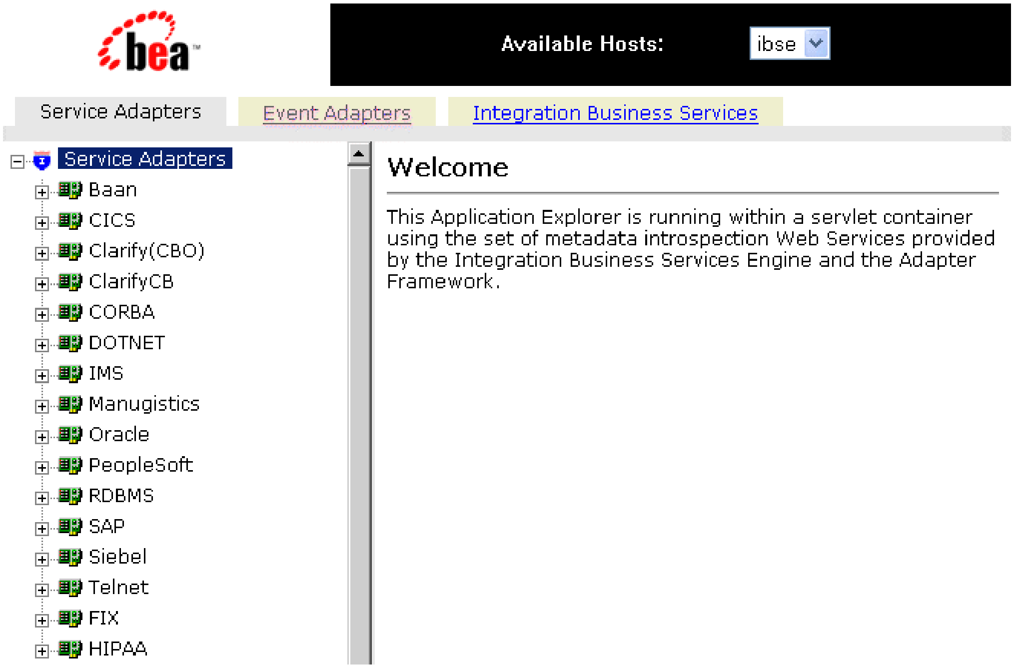
Is the name of the machine where your application server is running.

`port`

Is the port for the domain you are using for BEA. The port for the default domain is 7001.

Application Explorer opens.

The Service Adapters tab is active, and a list of the supported adapters appears on the left as shown in the following image. In the upper right, the Available Hosts drop-down list displays the Connector for JCA or a Servlet iBSE instance you can access. A Welcome message appears on the right.



For more information on adding instances, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual.

You are ready to create new targets for PeopleSoft.

Establishing a Target for PeopleSoft

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 automatically is saved after you create it. For information on creating a target, see *How to Create a New Target* on page 5-4.

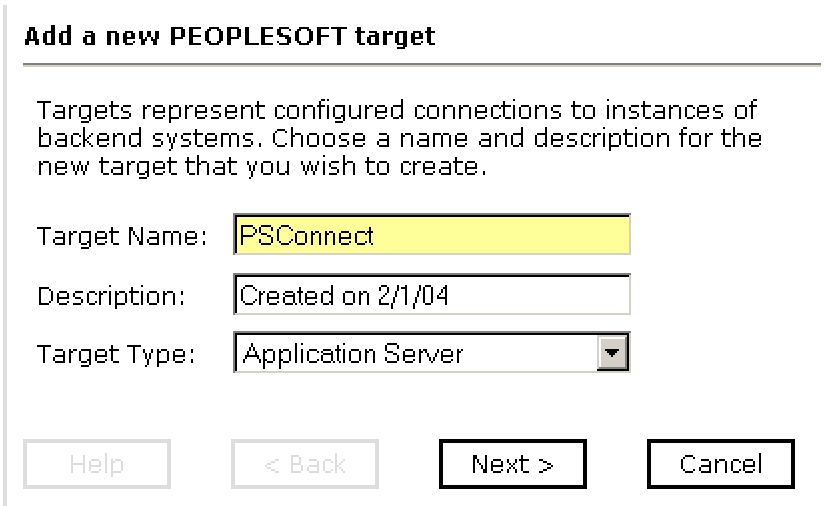
You must establish a connection to the system every time you start Application Explorer or after you disconnect from the system. When you open Application Explorer, a list of supported application systems appears in the left pane. The list is based on the adapters that you installed and for which you have licenses. For information on connecting to a target, see *How to Connect to a Target* on page 5-6.

Procedure: How to Create a New Target

To create a new target using Application Explorer:

1. Click *Service Adapters*.
2. Click the *PeopleSoft* node.
3. Move the pointer over *Operations* and select *Define a new target*.

   The Add a new PeopleSoft target pane opens on the right as shown in the following image.



   a. In the Target Name field, type a descriptive name for the target, for example, PSConnect.

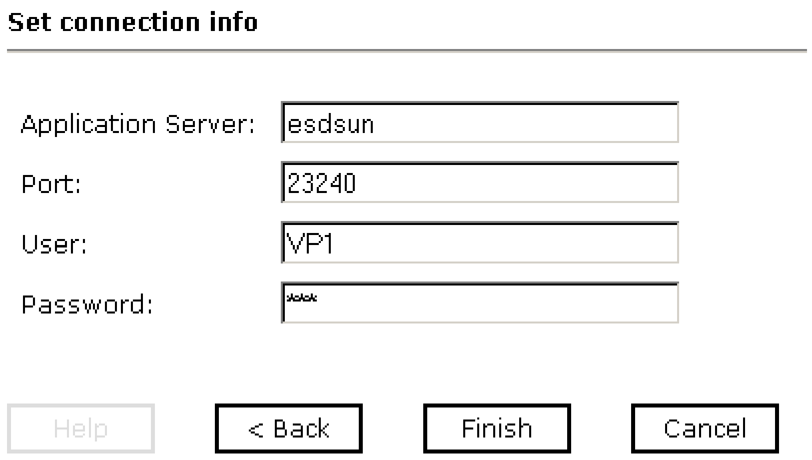
   b. In the Description field, type a brief description for the connection (optional).

c. From the Target Type drop-down list, select the type of target to which you are connecting.

   The default value is Application Server.

4. Click *Next*.

   The Set connection info pane opens on the right as shown in the following image.



   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 listens.

   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.

5. Click *Finish*.

   The PeopleSoft target, PSConnect, appears below the PeopleSoft node in the left pane as shown in the following image.

You are ready to connect to your PeopleSoft target.

Procedure: How to Connect to a Target

To connect to a target using Application Explorer:

1. Expand the *PeopleSoft* node and select the target you defined, for example, PSConnect, as shown in the following image.

2. In the right pane, move the pointer over *Operations* and select *Connect*.
3. In the Password field, enter a valid password and click *OK*.

   The following image shows the change in the PSConnect node that reflects that a connection was made.

4. Expand the *PSoftConnect* node.

   The following PeopleSoft business objects appear:

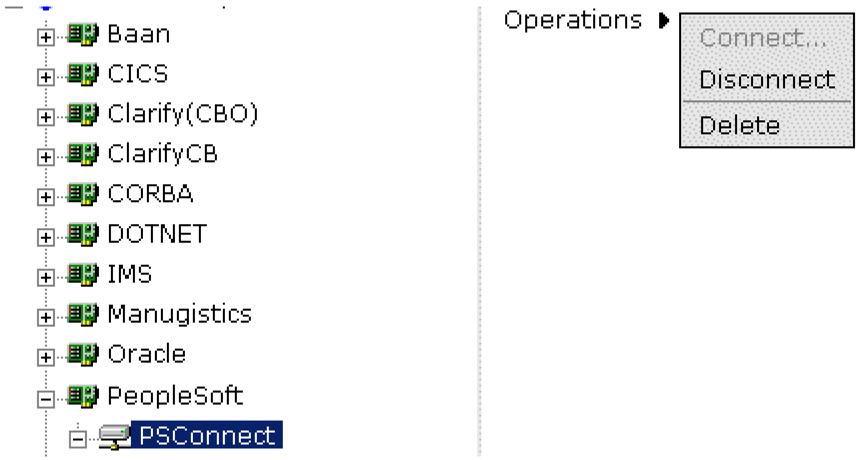
   - Component Interfaces
   - Messages
   - Component Interfaces (RPC)

Procedure: How to Disconnect From a Target

To disconnect from a target using Application Explorer:

1. In the left pane, click the target to which you are connected, for example, PSConnect.

The following image shows the PSConnect target below the expanded PeopleSoft ports node in the left pane and the Operations menu in the right pane.



2. Move the pointer over *Operations* and select *Disconnect*.

   Disconnecting from the application system drops the connection, but the node remains. The PSConnect node changes to reflect that a connection was closed.

Modifying a Target

After you create a target for PeopleSoft using Servlet Application Explorer, you can edit any of the information that you provided previously. For more information, see *How to Edit a Target* on page 5-7.

Although you can maintain multiple open connections to different application systems, it is recommended that you close connections when you are not using them. For information on disconnecting from a target, see *How to Disconnect From a Target* on page 5-6.

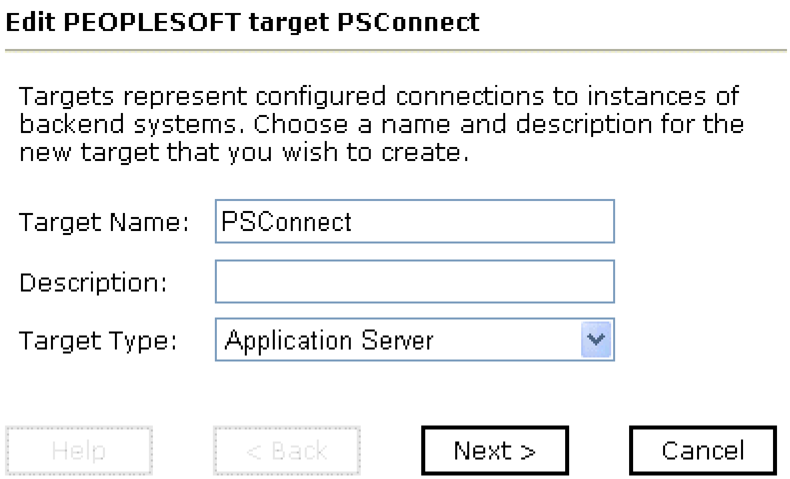
In addition to closing a connection, you can delete a target that is no longer required. You can delete it whether or not it is closed. If open, the target automatically closes before it is deleted. For more information, see *How to Delete a Target* on page 5-8.

Procedure: How to Edit a Target

To edit a target using Application Explorer:

1. In the left pane, click the target, for example, PSConnect.
2. In the right pane, move the pointer over *Operations* and select *Edit*.

The Edit PeopleSoft target pane opens on the right as shown in the following image. It shows the fields for Target Name and Description, the Target Type drop-down list, and buttons to click to choose whether to proceed to the next pane or to cancel the action.



3. Modify the connection information.
4. To display additional information, click *Next*.
5. After you complete your edits in the next pane, click *Finish*.

Procedure: How to Delete a Target

To delete a target using Application Explorer:

1. In the left pane, click the target, for example, PSConnect.
2. In the right pane, move the pointer over *Operations* and select *Delete*.

   A confirmation dialog box opens, as shown in the following image.

3. To delete the target you selected, click *OK*.

   The PSConnect node disappears from the left pane.

Viewing Application System Objects

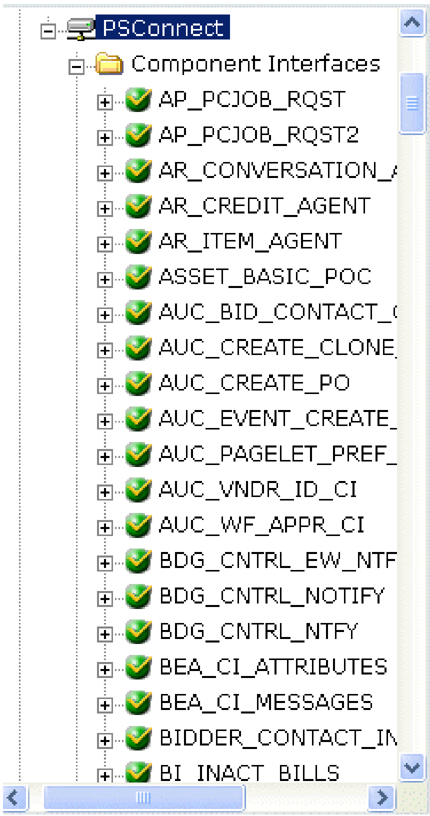
After you connect to PeopleSoft, Servlet Application Explorer enables you to explore and browse business object metadata. For example, Application Explorer enables you to view PeopleSoft Component Interface and Message metadata stored in the PeopleSoft business object repository.

Procedure: How to View Application System Objects

To view application system objects:

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

   The following image shows the list of PeopleSoft Component Interfaces that appears in the left pane.



You can now generate schemas. For more information, see *Creating an XML Schema*.

Creating an XML Schema

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. The exact location of the schemas differs depending on whether you deploy Application Explorer with an iBSE or a JCA configuration.

When the adapter is used with an **iBSE configuration**, Application Explorer stores the schemas in a subdirectory of the BEA installation directory, for example,

`C:\Program Files\iway55\bea\ibse\wsdl\schemas\service\peoplesoft\Psoft`

where:

`Psoft`

Is the name of the connection to the PeopleSoft system as defined in Application Explorer.

Under this directory, Application Explorer creates subdirectories containing schemas.

When the adapter is used with a **JCA configuration**, Application Explorer stores the schemas under a \schemas subdirectory of the BEA home directory, for example,

`C:\Program Files\iWay55\bea\config\base\schemas\peoplesoft\PsoftServer`

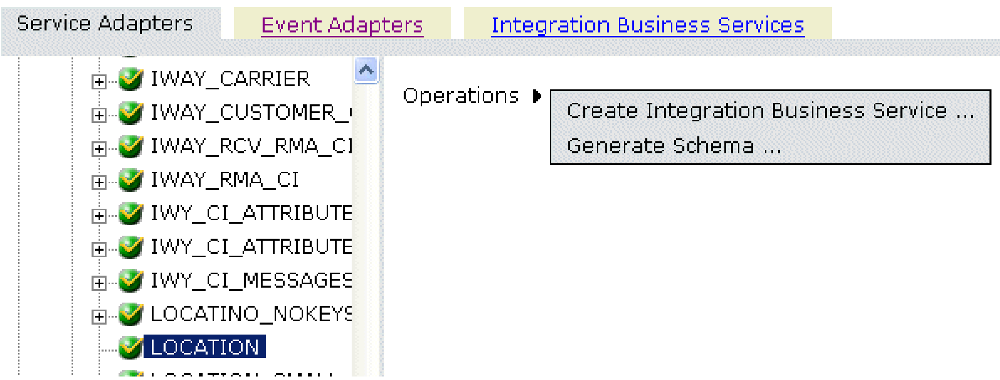
where:

`PsoftServer`

Is the name of the connection to the PeopleSoft system as defined in Application Explorer. Application Explorer stores the schemas in this directory.

Procedure: How to Create an XML Schema

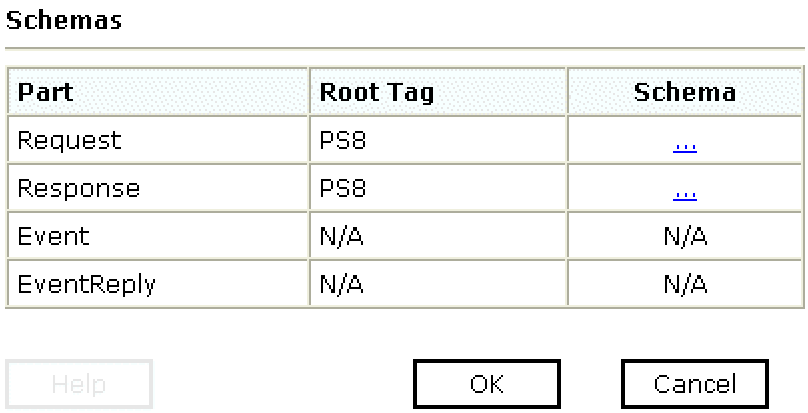
The following image shows a list of PeopleSoft Component Interfaces in the left pane. In the right pane, the Operations menu appears expanded and shows the Create Integration Business Service and Generate Schema menu options.



To create XML request and response schemas for a PeopleSoft Component Interface using Application Explorer:

1. Select the Component Interface you require.
2. In the right pane, move the pointer over *Operations* and select *Generate Schema*.

   The Schemas pane opens on the right with a table that defines the root tag for each schema and provides associated hyperlinks as shown in the following image.



3. Click the hyperlink associated with the type of schema you want to view.

The XML schema appears on the right as shown in the following image.

4. To return to the previous window, click the *Back* button on your Web browser.

   After you create schemas, you can create Web services.

   After you create schemas, you can also create events. For more information, see Chapter 6, *Listening for PeopleSoft Events*.

Generating a Web Service for PeopleSoft

You can generate Web services for PeopleSoft. To generate Web services, you must deploy the adapter in a Web services environment using Integration Business Services Engine (iBSE). iBSE exposes functionality as Web services and serves as a gateway to heterogeneous back-end applications and databases.

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. For the caller or sender, a Web service can be considered a "black box" that may require input and delivers a result. Business services can be integrated within an enterprise as well as across enterprises on any communication technology stack, whether asynchronous or synchronous, in any format.

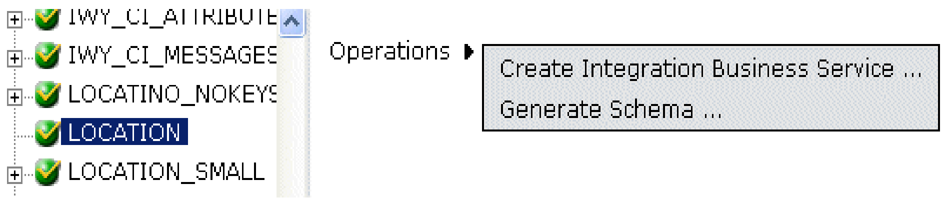
You can make Web services available to other services within a host server by generating WSDL (Web Services Description Language) from the Web service.

Procedure: How to Create a Web Service

To create a Web service for PeopleSoft:

1. If you have not already done so, connect to a PeopleSoft target as described in *Establishing a Target for PeopleSoft* on page 5-4.
2. Expand the PeopleSoft node and select the interface for which you want to create a Web service.

   The following image shows a list of PeopleSoft Component Interfaces in the left pane. In the right pane, the Operations menu appears expanded and shows the Create Integration Business Service and Generate Schema menu options.

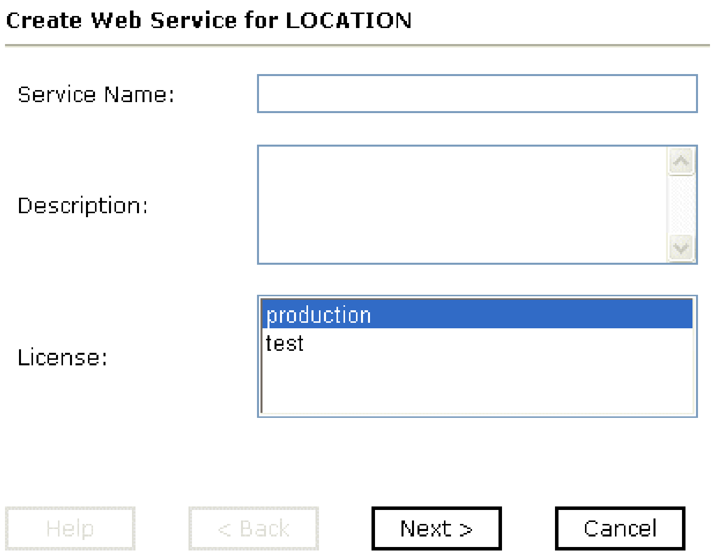
   

3. In the right pane, move the pointer over *Operations* and select *Create Integration Business Services*.

   The Create Web Service pane opens on the right as shown in the following image.

   You can select to create a new service or you can select to use an existing service.

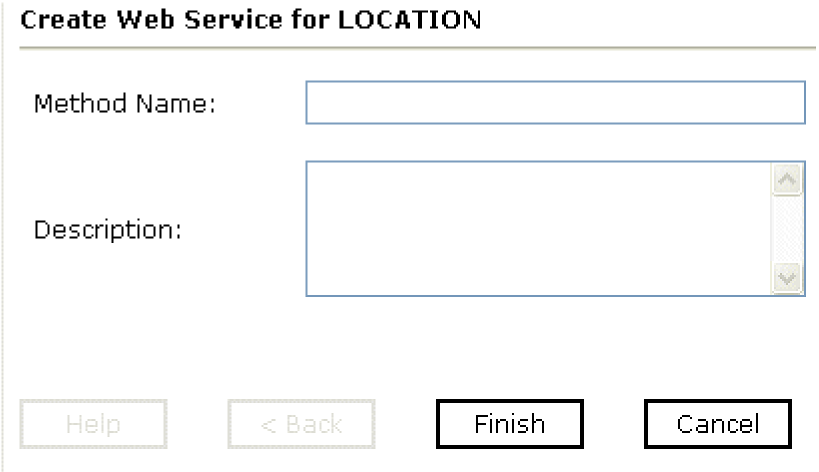
   If you select to create a new service, the following Create Web Service pane opens.



   a. In the Service Name field, type a descriptive name for the Web service.
   b. In the Description field, type a brief description of the Web service (optional).
   c. From the License list, select *production* or *test*.

4. Click *Next*.

   The following image shows the next pane that opens where you continue to input information.

   a. In the Method Name field, type a descriptive name for the method.
   b. In the Description field, type a brief description of the method (optional).

5. Click *Finish*.

   Application Explorer switches the view to the Integration Business Services tab, and the new Web service appears in the left pane.

Testing a Web Service

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

Procedure: How to Test a Web Service

To test a Web service:

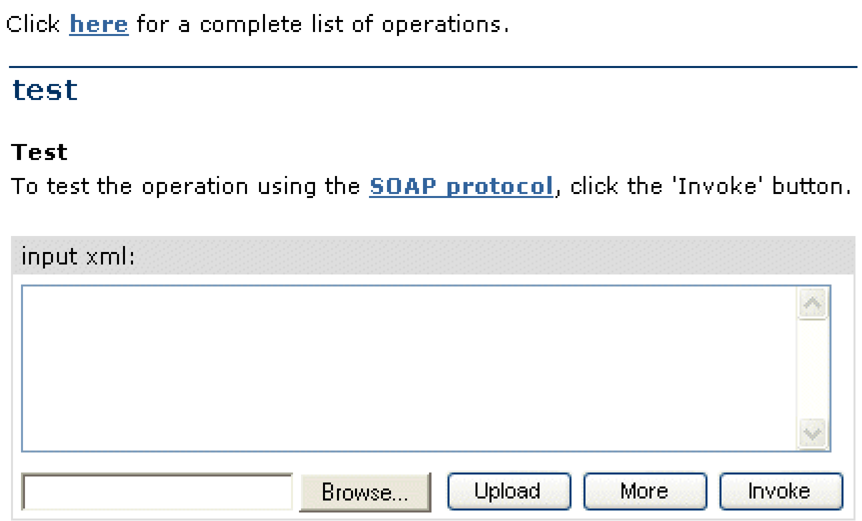
1. If you are not on the Integration Business Services tab of Application Explorer, click the tab to access Web services.
2. If it is not expanded, expand the list of Web services under Integration Business Services.
3. Expand the *Services* node.
4. Select the name of the Web service you want to test.

   The Web service name appears as a hyperlink in the right pane.

5. In the right pane, click the named Web services hyperlink.

   The test option appears in the right pane.

   If you are testing an Integration Business Service that requires XML input, an input xml field appears as shown in the following image.



6. In the input xml field, either type a sample XML document that queries the service or browse to the location of an XML instance and click *Open*.

   For a sample request, see *Retrieving a List of Locations* on page 5-17.

7. Click *Invoke*.

   Application Explorer displays the results in the right pane.

The following image shows a sample XML returned by iBSE.

```
  <?xml version="1.0" encoding="UTF-8" ?>
- <SOAP-ENV:Envelope
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns:SOAP-
    ENV="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-
    instance">
  - <SOAP-ENV:Body>
    - <GetLocationResponse
        xmlns="urn:iwaysoftware:ibse:jul2003:GetLocation:r
        cid="0714715118B19677314F2AFA356F0EEE">
      - <PS8>
        - <result>
          - <LOCATION_TBL row="1">
              <SETID>SHARE</SETID>

                <LOCATION>ALBERTA</LOCATION>
              <EFFDT>05/25/2006</EFFDT>

                <EFF_STATUS>A</EFF_STATUS>
              <DESCR>Alberta -
                Canada</DESCR>
              <DESCR_AC_0 />
              <DESCRSHORT />
              <BUILDING />
              <FLOOR />
```

Example: Retrieving a List of Locations

The following sample run-time input XML file retrieves a list of locations using the LOCATION Component Interface.

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

CHAPTER 6

Listening for PeopleSoft Events

Topics:

- Understanding Event Functionality
- Creating, Editing, or Deleting a Port
- Creating, Editing, or Deleting a Channel

This section describes how to use Servlet Application Explorer to connect to PeopleSoft and listen for events. Several port dispositions are available, and you can choose the technique that best suits your requirements.

Understanding Event Functionality

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 Servlet Application Explorer. To create an event, you must create a port and a channel.

The following is a description of how ports and channels work.

- 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. For more information, see *Creating, Editing, or Deleting a Port* on page 6-2.

- 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. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Creating, Editing, or Deleting a Port

You can create, edit, or delete an event port using Servlet Application Explorer.

You create a port for a PeopleSoft Message from the Service Adapters tab or from the Event Adapters tab. You can switch between an iBSE and a JCA implementation using the Available Hosts drop-down list in Application Explorer. The following dispositions are available when using Servlet Application Explorer in conjunction with an iBSE implementation.

- **File.** The File disposition uses a file URL to specify the destination file name or directory where the event document will be written. During run time, the destination file name may require indexing to avoid overwriting. For more information, see *How to Create an Event Port for the File Disposition* on page 6-3.
- **HTTP.** The HTTP disposition uses an HTTP URL to specify an HTTP end point to which the event document is posted. For more information, see *How to Create a Port for the HTTP Disposition* on page 6-5.
- **iBSE.** The iBSE disposition enables an event to launch a business service method. For more information, see *How to Create a Port for the iBSE Disposition* on page 6-7.

- **JMSQ.** The JMS queue disposition enables an event to be enqueued to a JMS queue. For more information, see *How to Create an Event Port for the JMS Queue Disposition* on page 6-11.
- **SOAP.** The SOAP disposition enables an event to launch a business service specified by a WSDL file. A SOAP action is optional; "" is the default value. For more information, see *How to Create a Port for the SOAP Disposition* on page 6-9.
- **MSMQ.** The Microsoft Message Queuing (MSMQ) disposition supports public and private queues. For more information, see *How to Create a Port for the MSMQ Disposition* on page 6-8.
- **MQSeries.** The MQSeries disposition enables an event to be enqueued to an MQSeries queue. Both queue manager and queue name may be specified. For more information, see *How to Create a Port for the MQSeries Disposition* on page 6-12.
- **MAIL.** The MAIL disposition option will be supported in a future release.

The following dispositions are available when using the Servlet Application Explorer in conjunction with a JCA implementation:

- **File.** The File disposition uses a file URL to specify the destination file name or directory where the event document will be written. During run time, the destination file name may require indexing to avoid overwriting. For more information, see *How to Create an Event Port for the File Disposition* on page 6-3.
- **HTTP.** The HTTP disposition uses an HTTP URL to specify an HTTP end point to which the event document is posted. For more information, see *How to Create a Port for the HTTP Disposition* on page 6-5.
- **JMS.** The JMS queue disposition enables an event to be enqueued to a JMS queue. For more information, see *How to Create an Event Port for the JMS Queue Disposition* on page 6-11.
- **MQSeries.** The MQSeries disposition enables an event to be enqueued to an MQSeries queue. Both queue manager and queue name may be specified. For more information, see *How to Create a Port for the MQSeries Disposition* on page 6-12.

For information on editing a port, see *How to Edit an Event Port* on page 6-13. For information on deleting a port, see *How to Delete an Event Port* on page 6-14.

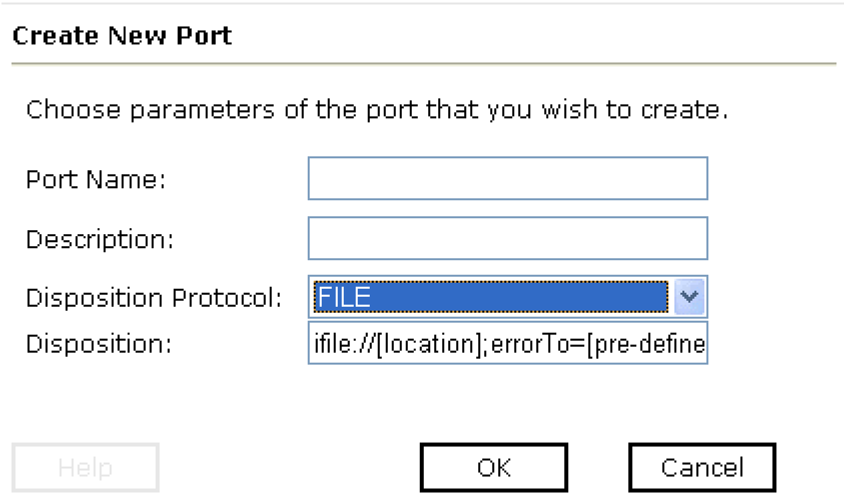
Procedure: How to Create an Event Port for the File Disposition

To create a specific event port for the File disposition using Application Explorer:

1. Click the *Event Adapters* tab.
2. In the left pane, expand the *PeopleSoft* node.
3. Select the *ports* node.

4. Move the pointer over *Operations* and select *Add a new port*.

The Create New Port pane opens on the right as shown in the following image where you choose parameters for the new port.



a. In the Port Name field, type a name for the event.

Note: Ensure that you specify a name that conforms to standards set by PeopleSoft. For example, when using PeopleSoft, periods are not allowed. You must remove all instances of this character.

b. In the Description field, type a brief description (optional).

c. From the Disposition Protocol drop-down list, select *FILE*.

d. In the Disposition field, type a File destination to which event data is written.

When pointing Application Explorer to an **iBSE** deployment, specify the destination file using the following format:

```
ifile://[location];
errorTo=[pre-defined port name or another disposition url]
```

When pointing Application Explorer to a **JCA** deployment, provide the full path to the directory.

The following table lists and describes the parameters for the disposition.

Parameter	Description
location	Destination and file name of the document where event data is written, for example, `ifile://D:\in\x.txt;errorTo=ifile://D:\error.`
errorTo	Location where error documents are sent. Predefined port name or another full URL. Optional.

5. Click *OK*.

The event 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. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Create a Port for the HTTP Disposition

To create a port for an HTTP disposition using Application Explorer:

1. Click the *Event Adapters* tab.

2. In the left pane, expand the *PeopleSoft* node.

3. Select the *ports* node.

4. Move the pointer over *Operations* and select *Add a new port*.

The Create New Port pane opens on the right.

To point the Application Explorer to an iBSE deployment, follow the steps in *How to Create a Port for the HTTP Disposition for an iBSE Deployment* on page 6-5. To point the Application Explorer to a JCA Deployment, see *How to Create a Port for the HTTP Disposition for a JCA Deployment* on page 6-6.

Procedure: How to Create a Port for the HTTP Disposition for an iBSE Deployment

To create a port for the HTTP Disposition and point Application Explorer to an iBSE deployment:

1. Perform the procedure, *How to Create a Port for the HTTP Disposition* on page 6-5.

2. In the Port Name field, type a name for the event.

3. In the Description field, type a brief description (optional).

4. From the Disposition Protocol drop-down list, select *HTTP*.

5. In the Disposition field, enter an HTTP destination.
6. To point Application Explorer to an iBSE deployment, use the following format

   ```
   ihttp://[myurl];responseTo=[pre-defined port name or another
   disposition url];
   ```

   where:

   `url`
   : Is the URL target for the post operation, for example,

     `http://myhost:1234/docroot`

   `responseTo`
   : Is the location where responses are posted, if desired.

7. 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. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Create a Port for the HTTP Disposition for a JCA Deployment

To create a port for the HTTP Disposition and point Application Explorer to a JCA deployment:

1. Perform the procedure, *How to Create a Port for the HTTP Disposition* on page 6-5.
2. In the Port Name field, type a name for the event.
3. In the Description field, type a brief description (optional).
4. From the Disposition Protocol drop-down list, select *HTTP*.
5. In the Disposition field, enter an HTTP destination.
6. To point Application Explorer to a JCA deployment, use the following format

   `http://`*`host:port/uri`*

   where:

   `host:port`
   : Is the combination of the name of the host on which the Web server resides and the port on which the server is listening for the post operation.

   `uri`
   : Is the universal resource identifier that completes the url specification.

7. 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. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Create a Port for the iBSE Disposition

To create a port for an iBSE disposition using Application Explorer:

1. Click the *Event Adapters* tab.
2. In the left pane, expand the *PeopleSoft* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create New Port pane opens on the right.

   a. In the Port Name field, type a name for the event.

   b. In the Description field, type a brief description (optional).

   c. From the Disposition Protocol drop-down list, select *iBSE*.

   d. In the Disposition field, enter an iBSE destination in the form of:

   ```
   ibse:svcName.mthName;
   responseTo=[pre-defined port name or another disposition url];
   errorTo=[pre-defined port name or another disposition url]
   ```

   The following table lists and describes the parameters for the disposition.

Parameter	Description
svcName	Name of the service created with iBSE.
mthName	Name of the method created for the business service.
responseTo	Location where responses to the business service are posted. Predefined port name or another full URL. Optional.
errorTo	Location where error documents are sent. Predefined port name or another full URL. Optional.

5. 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 port you created.

You are ready to associate the event port with a channel. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Create a Port for the MSMQ Disposition

To create a port for an MSMQ disposition using Application Explorer:

1. Click the *Event Adapters* tab.
2. In the left pane, expand the *PeopleSoft* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create New Port pane opens on the right.

   a. In the Port Name field, type a name for the event.

   b. In the Description field, type a brief description (optional).

   c. From the Disposition Protocol drop-down list, select *MSMQ*.

   d. In the Disposition field, enter a MSMQ destination in the form of:

   ```
   msmq:/host/private$/qName;
   errorTo=[pre-defined port name or another disposition url]
   ```

   The following table lists and describes the parameters for the disposition.

Parameter	Description
host	Machine name where the Microsoft Queuing system is running.
Queue Type	For private queues, enter *Private$*. Private queues are queues that are not published in Active Directory. They appear only on the local computer that contains them. Private queues are accessible only by Message Queuing applications that recognize the full path name or format name of the queue.
qName	Name of the private queue where messages are placed.
errorTo	Location where error documents are sent. Predefined port name or another full URL. Optional.

5. 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 port you created.

You are ready to associate the event port with a channel. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Create a Port for the SOAP Disposition

To create a port for a SOAP disposition:

1. Click the *Event Adapters* tab.
2. In the left pane, expand the *PeopleSoft* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create New Port pane opens on the right.

   a. In the Port Name field, type a name for the event.

   b. In the Description field, type a brief description (optional).

   c. From the Disposition Protocol drop-down list, select *SOAP*.

   d. In the Disposition field, enter an SOAP destination, using the following format:

   ```
   soap:[wsdl-url];soapaction=[myaction];
   method=[web service method];namespace=[namespace];
   responseTo=[pre-defined port name or another disposition URL];
   errorTo=[pre-defined port name or another disposition url]
   ```

   The following table lists and describes the parameters for the disposition.

Parameter	Description
wsdl-url	The URL to the WSDL file that is required to create the SOAP message, for example: `http://localhost:7001/ibse/IBSEServlet/test/webservice.ibs?wsdl` where: *webservice* Is the name of the Web service you created using Application Explorer. To find this value, navigate to the *Integration Business Services* tab and open the *Service Description* hyperlink in a new window. The WSDL URL appears in the Address field. You can also open the WSDL file in a third party XML editor (for example, XMLSPY) and view the SOAP request settings to find this value.

Parameter	Description

soapaction	Method called by the SOAP disposition, for example:
	`webservice.method@test@@`
	where:
	`webservice`
	Is the name of the Web service you created using Application Explorer.
	`method`
	Is the method being used.
	`test`
	Is the license that is being used by the Web service.
	To find this value, navigate to the *Integration Business Services* tab and open the *Service Description* hyperlink in a new window. Perform a search for *soapAction*.
	You can also open the WSDL file in a third party XML editor (for example, XMLSPY) and view the SOAP request settings to find this value.

method	The Web service method you are using. This value can be found in the WSDL file.
namespace	The XML namespace you are using. This value can be found in the WSDL file.
responseTo	Location where responses are posted. Predefined port name or another disposition URL. Optional. The URL must be complete, including the protocol.
errorTo	Location where error documents are sent. Predefined port name or another full URL. Optional.

5. Click *OK*.

The event 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. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Create an Event Port for the JMS Queue Disposition

To create a port for a JMS queue disposition using Application Explorer:

1. Click the *Event Adapters* tab.
2. In the left pane, expand the *PeopleSoft* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create New Port pane opens on the right.

   a. In the Port Name field, type a name for the event.

   b. In the Description field, type a brief description (optional).

   c. From the Disposition Protocol drop-down list, select *JMSQ*.

   d. In the Disposition field, enter a JMS destination.

   When pointing Application Explorer to an **iBSE** deployment, use the following format:

   ```
   jmsq:myQueueName@myQueueFac;jndiurl=[myurl];
   jndifactory=[myfactory];user=[user];password=[xxx];
   errorTo=[pre-defined port name or another disposition url]
   ```

   When pointing Application Explorer to a **JCA** deployment, use the following format:

   ```
   jms:jmsqueue@jmsfactory;jndiurl=;jndifactory=;
   ```

   The following table lists and describes the parameters for the disposition.

Parameter	Description
queue	Name of a queue to which events are emitted.
Connection Factory	A resource that contains information about the JMS Server. The WebLogic connection factory is: `javax.jms.QueueConnectionFactory`
jndi_url	The URL to use to contact the JNDI provider. The syntax of this URL depends on the JNDI provider that is used. This value corresponds to the standard JNDI property: `java.naming.provider.url`

Parameter	Description
jndi_factory	Is JNDI context.INITIAL_CONTEXT_FACTORY and is provided by the JNDI service provider. For WebLogic Server, the WebLogic factory is: `weblogic.jndi.WLInitialContextFactory`
user	User ID associated with this queue.
password	Password for the user ID.
errorTo	Location where error documents are sent. Predefined port name or another full URL. Optional.

5. 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 port you created.

You are ready to associate the event port with a channel. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Create a Port for the MQSeries Disposition

To create a port for an MQSeries disposition using Application Explorer:

1. Click the *Event Adapters* tab.
2. In the left pane, expand the *PeopleSoft* node.
3. Select the *ports* node.
4. Move the pointer over *Operations* and select *Add a new port*.

   The Create New Port pane opens on the right.

   a. In the Port Name field, type a name for the event.

   b. In the Description field, type a brief description (optional).

   c. From the Disposition Protocol drop-down list, select *MQSeries*.

   d. In the Disposition field, enter an MQSeries destination.

      When pointing Application Explorer to an **iBSE** deployment, use the following format:

```
mqseries:/qManager/qName;host=[hostname];port=[port];
channel=[channnelname];
errorTo=[pre-defined port name or another disposition url]
```

When pointing Application Explorer to a **JCA** deployment, use the following format:

```
mq:qmanager@respqueue;host=;port=;channel=
```

The following table lists and describes the parameters for the disposition.

Parameter	Description
qManager	Name of the queue manager to which the server must connect.
qName or respqueue	Name of the queue where messages are placed.
host	Host on which the MQ server is located (MQ Client only).
port	Number to connect to MQ server queue manager (MQ client only).
channel	Case-sensitive name of the channel that connects with the remote MQ server queue manager (MQ client only). The default channel name for MQSeries is SYSTEM.DEF.SVRCONN.
errorTo	Location where error documents are sent. Predefined port name or another full URL. Optional.

5. 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. For more information, see *Creating, Editing, or Deleting a Channel* on page 6-15.

Procedure: How to Edit an Event Port

To edit an event port:

1. Select the event port you want to edit.

2. In the right pane, move the pointer over *Operations*, and select *Edit*.

The Edit Port pane opens on the right.

3. Make the required changes to the event port configuration fields.

4. Click *OK*.

Procedure: How to Delete an Event Port

To delete an event port:

1. Select the event port you want to delete.
2. In the right pane, move the pointer over *Operations*, and select *Delete*.

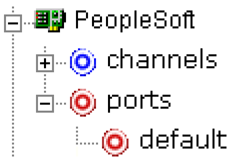
   A confirmation dialog box opens, as shown in the following image.

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

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

Using the Default Event Port

When using Application Explorer to connect to PeopleSoft and listen for events, a default event port is available at all times as shown in the following image.



The default event port can be used for testing purposes or when you do not want to route event data to a specific port you configured. The default port is enabled when you start a channel that does not have a specific event port assigned.

The default event data is actually a file disposition that writes to an out.xml file in the following output directory:

```
ifile://./eventOut/out.xml
```

If you have deployed your adapter in a BEA WebLogic environment, this directory is created in your user domain.

Creating, Editing, or Deleting a Channel

All defined event ports must be associated with a channel. You can create, edit, or delete a channel for your event adapter using Servlet Application Explorer. For information on creating a channel, see *How to Create a Channel* on page 6-15. For information on editing a channel, see *How to Edit a Channel* on page 6-22. For information on deleting a channel, see *How to Delete a Channel* on page 6-23.

You can also create a channel using one of the following protocols:

- **HTTP.** For more information, see *How to Create an HTTP Channel* on page 6-18.
- **File.** For more information, see *How to Create a File Channel* on page 6-19.
- **TCP.** For more information, see *How to Create a TCP Channel* on page 6-21.

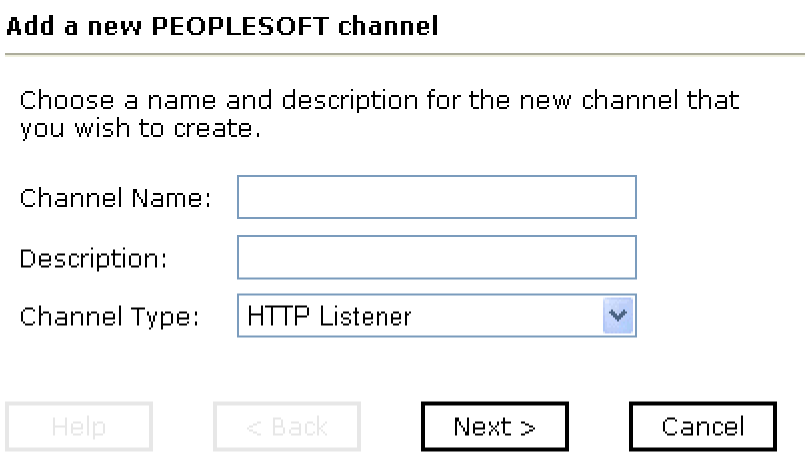
Procedure: How to Create a Channel

To create a channel using Application Explorer:

1. Click the *Event Adapters* tab.
   a. In the left pane, expand the *PeopleSoft* node.

      The ports and channels nodes appear.
   b. Click the *channels* node.
2. In the right pane, move the pointer over *Operations* and select *Add a new channel*.

   The Add a new PEOPLESOFT channel pane opens on the right as shown in the following image.



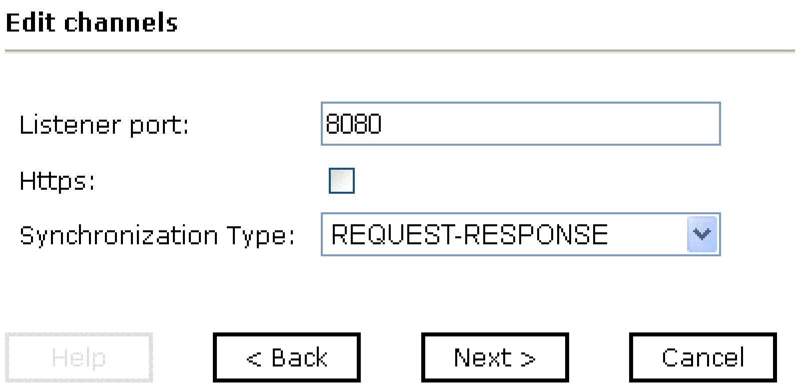
   a. In the Channel Name field, type a name, for example, TEST_CHANNEL.

b. In the Description field, type a brief description (optional).

c. From the Channel Type drop-down list, select a channel type, for example, HTTP Listener.

3. Click *Next*.

   The Edit channels pane opens on the right as shown in the following image where you enter the listener port number and choose the synchronization type from the drop-down list.

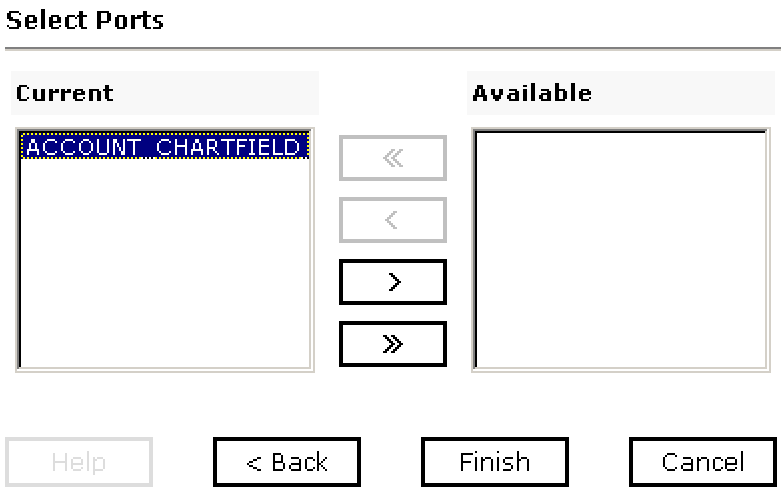


4. Enter information specific to your PeopleSoft system and the channel you are creating.
5. For a secure HTTP connection, select the Https check box.
6. From the Synchronization Type drop-down list, select *REQUEST-ACK* or *REQUEST-RESPONSE*.

   Note: The PeopleSoft channel will not work if the the synchronization type is set to REQUEST

7. Click *Next*.

The Select Ports pane opens as shown in the following image. Current ports appear on the left and available ports appear on the right. Arrow buttons are active to use to move the port from one list to the other. Back, Finish, and Cancel buttons are active.



a. Select an event port from the list of current ports.

b. To transfer the port to the list of available ports, click the single right arrow button. To associate all event ports, click the double right arrow button.

8. Click *Finish*.

The channel appears in the left pane under the channels port with an X over the icon to indicate that the channel is disconnected as shown in the following image.

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

You must start the channel to activate your event configuration.

9. In the right pane, move the pointer over *Operations* and select *Start the channel*.

When the channel is activated, the X over the icon in the left pane disappears.

To stop the channel at any time, you can move the pointer over Operations and select the option to stop the channel.

Procedure: How to Create an HTTP Channel

To create an HTTP channel using Application Explorer:

1. Click the *Event Adapters* tab.
   a. In the left pane, expand the *PeopleSoft* node.
      The ports and channels nodes appear.
   b. Click the *channels* node.
2. In the right pane, move the pointer over *Operations* and select *Add a new channel*.
   The Add a new PEOPLESOFT channel pane opens on the right.
   a. Type a name for the channel, for example, NewChannel.
   b. Type a brief description (optional).
   c. From the drop-down list, select *HTTP Listener*.
3. Click *Next*.
   The Edit Channels pane opens on the right.
4. Provide the required values based on the list of parameters and their descriptions in the following table.

Parameter	Description
Listener port	Port on which to listen for PeopleSoft event data.
HTTPS	For a secure HTTP connection, select this check box.
Synchronization Type	Choose from three synchronization options: • REQUEST • REQUEST_RESPONSE • REQUEST_ACK

5. Click *Next*.
   The Select Ports pane opens.
   a. Select an event port from the list of current ports.
   b. To transfer the port to the list of available ports, click the single right arrow button. To associate all the event ports, click the double right arrow button.

6. Click *Finish*.

   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.

   In the right pane, a summary provides the channel description, channel status, and available ports. All the information is associated with the channel you created.

7. In the right pane, move the pointer over *Operations* and select *Start the channel*.

   When the channel is activated, the X over the icon in the left pane disappears.

   To stop the channel at any time, you can move the pointer over Operations and select the option to stop the channel.

Procedure: How to Create a File Channel

To create a File channel using Application Explorer:

1. Click the *Event Adapters* tab.
   a. In the left pane, expand the *PeopleSoft* node.

      The ports and channels nodes appear.

   b. Click the *channels* node.
2. In the right pane, move the pointer over *Operations* and select *Add a new channel*.

   The Add a new PEOPLESOFT channel pane opens on the right.

   a. Type a name for the channel, for example, NewChannel.
   b. Type a brief description (optional).
   c. From the drop-down list, select *File Listener.*
3. To open the Edit Channels pane, click *Next*.
   a. In the Request tab, enter values based on the descriptions in the following table.

Parameter	Description
Polling Location	Target file system location for the PeopleSoft XML file.
File Mask	File name to use for the output file generated as a result of the operation.

b. In the Response tab, enter values for the parameters based on the descriptions in the following table.

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

c. In the Advanced tab, enter values for the parameters based on the descriptions in the following table.

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

4. Click *Next*.

The Select Ports pane opens.

a. Select an event port from the list of current ports.

b. To transfer the port to the list of available ports, click the single right arrow button. To associate all the event ports, click the double right arrow button.

5. Click *Finish*.

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.

In the right pane, a summary provides the channel description, channel status, and available ports. All the information is associated with the channel you created.

6. Move the pointer over *Operations* and select *Start the channel*.

   When the channel is activated, the X over the icon in the left pane disappears.

   To stop the channel at any time, you can move the pointer over Operations and select the option to stop the channel.

Procedure: How to Create a TCP Channel

To create an TCP channel using Application Explorer:

1. Click the *Event Adapters* tab.
   a. In the left pane, expand the *PeopleSoft* node.

      The ports and channels nodes appear.
   b. Click the *channels* node.
2. In the right pane, move the pointer over *Operations* and select *Add a new channel*.

   The Add a new PEOPLESOFT channel pane opens on the right.
   a. Type a name for the channel, for example, NewChannel.
   b. Type a brief description (optional).
   c. From the drop-down list, select *TCP Listener.*
3. Click *Next*.

   The Edit Channels window opens on the right.
4. Enter values for the parameters based on the descriptions in the following table:

Property Name	Property Description
Host	Host machine on which the adapter is installed.
Port Number	Socket number on which the host listens for incoming messages from a client application.

Property Name	**Property Description**
Synchronization Type	Choose from the following synchronization options: • RECEIVE_REPLY • RECEIVE_ACK • RECEIVE
Is Length Prefix	For PeopleSoft events that return data that is not in XML format. The TCP/IP event application must prefix the data with a 4-byte binary length field when writing data to the TCP/IP port.
Is XML	For PeopleSoft events that return data in XML format. No preparser is required.
Is Keep Alive	Maintains continuous communication between the event transaction and the channel.

5. Click *Next*.

The Select Ports pane opens.

a. Select an event port from the list of current ports.

b. To transfer the port to the list of available ports, click the single right arrow button. To associate all the event ports, click the double right arrow button.

6. Click *Finish*.

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.

In the right pane, a summary provides the channel description, channel status, and available ports. All the information is associated with the channel you created.

7. Move the pointer over *Operations* and select *Start the channel*.

When the channel is activated, the X over the icon in the left pane disappears.

To stop the channel at any time, you can move the pointer over Operations and select the option to stop the channel.

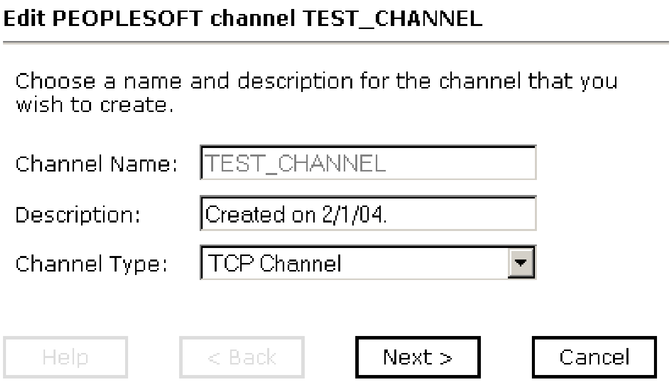
Procedure: How to Edit a Channel

To edit a channel:

1. In the left pane, select the channel you want to edit.

2. In the right pane, move the pointer over *Operations* and select *Edit*.

   The Edit channel pane opens on the right and includes current information for Channel Name, Description, and Channel Type as shown in the following image.



3. Make the required changes to the channel configuration fields.
4. Click *Next* to continue making changes on the next pane.
5. When you are finished making your changes, click *Finish*.

Procedure: How to Delete a Channel

To delete a channel:

1. In the left pane, select the channel you want to delete.
2. In the right pane, move the pointer over *Operations* and select *Delete*.

   A confirmation dialog box opens, as shown in the following image.

3. To delete the channel you selected, click *OK*.

   The channel disappears from the list in the left pane.

CHAPTER 7

Management and Monitoring

Topics:

- Managing and Monitoring Services and Events Using iBSE
- Managing and Monitoring Services and Events Using the JCA Test Tool
- Setting Engine Log Levels
- Configuring Connection Pool Sizes
- Migrating Repositories
- Exporting or Importing Targets
- Retrieving or Updating Web Service Method Connection Information
- Starting or Stopping a Channel Programmatically

After you create services and events using Application Explorer, you can use management and monitoring tools to measure the performance in your run-time environment. This section describes how to configure and use these features.

Managing and Monitoring Services and Events Using iBSE

Integration Business Services Engine (iBSE) provides a console that enables you to manage and monitor services and events currently in use. The console also displays resource usage and invocation statistics. These indicators can help you adjust your environment for optimum efficiency.

The following monitoring levels are available for services:

- System
- Service
- Method

The following monitoring levels are available for events:

- System
- Channel
- Port

Procedure: How to Configure Monitoring Settings

To configure monitoring settings:

1. Ensure that your BEA WebLogic Server is started.
2. To access the monitoring console, enter the following URL in your Web browser:

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

   where:

   `hostname`

   Is the machine where the application server is running.

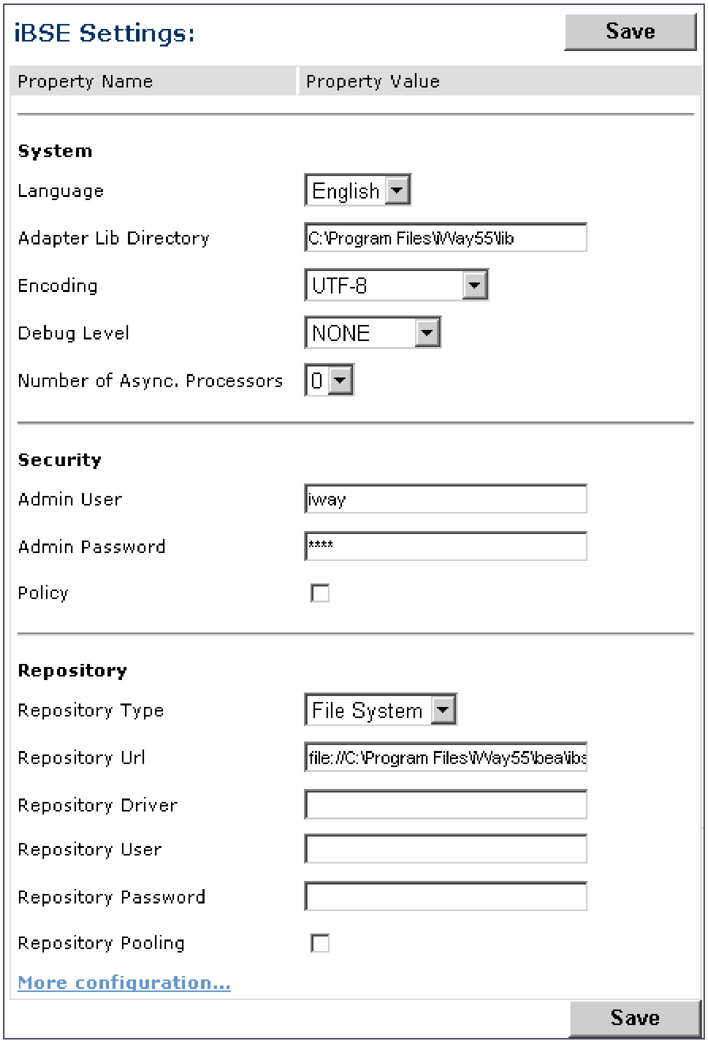
   `port`

   Is the HTTP port for the application server.

   The iBSE Settings window opens as shown in the following image. It consists of three panes. To configure system settings, the System pane includes drop-down lists for selecting language, encoding, the debug level, and the number of asynchronous processors. It also contains a field where you can enter a path to the adapter lib directory.

   To configure security settings, the Security pane contains fields for typing the Admin User name and the associated password and a check box for specifying policy.

To configure repository settings, the Repository pane contains a drop-down list for selecting the repository type; fields to type information for the repository URL, driver, user, and password; and a check box where you can enable repository pooling. In the upper and lower right of the window is a Save button. In the lower left of the window is an option to access more configuration settings.



3. Click *More configuration*.

Tip: To access the monitoring console directly, enter the following URL in your Web browser:

`http://`*`hostname`*`:`*`port`*`/ibse/IBSEStatus`

where:

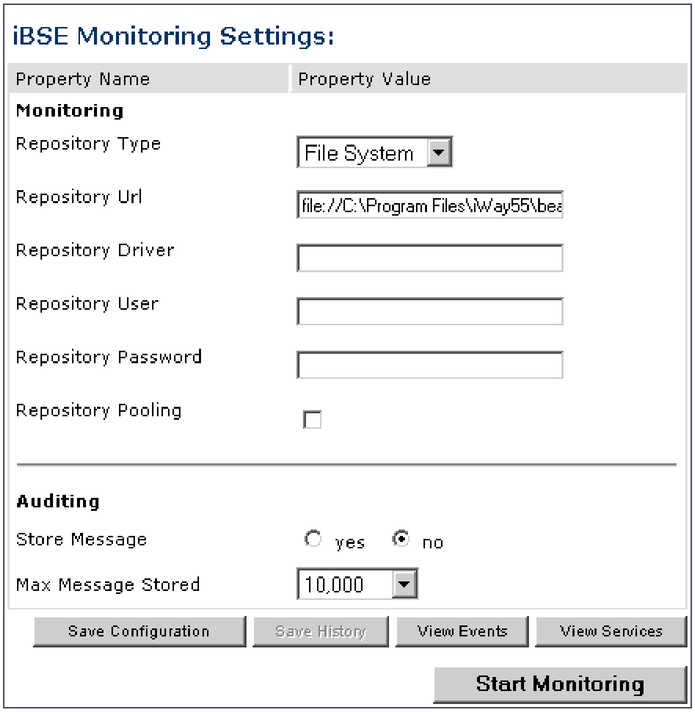
`hostname`

Is the machine where the application server is running.

`port`

Is the HTTP port for the application server.

The iBSE Monitoring Settings window which is divided into two panes opens as shown in the following image. At the bottom of the window is a row of command buttons that enable you to save your configuration, view events, or view services. The Save History button is inactive.



a. In the Monitoring pane, from the Repository Type drop-down list, select the type of repository you are using.

b. To connect to the database in the Repository Url field, type a JDBC URL.

- **c.** To connect to the database in the Repository Driver field, type a JDBC Class.
- **d.** To access the monitoring repository database, type a user ID and password.
- **e.** To enable pooling, select the *Repository Pooling* check box.
- **f.** In the Auditing pane, click *yes* if you want to store messages.

  This option is disabled by default.

  Note: You must start and then stop monitoring to enable this option.
- **g.** From the drop-down list, select the maximum number of messages to store.

  By default, 10,000 is selected.

  Note: Depending on your environment and the number of messages that are exchanged, storing a large number of messages may affect system performance. If you require more information about your system resources, consult your system administrator.
- **h.** Click *Save Configuration*.

4. Click *Start Monitoring*.

iBSE begins to monitor all services and events currently in use. If you selected the option to store messages, iBSE stores messages.

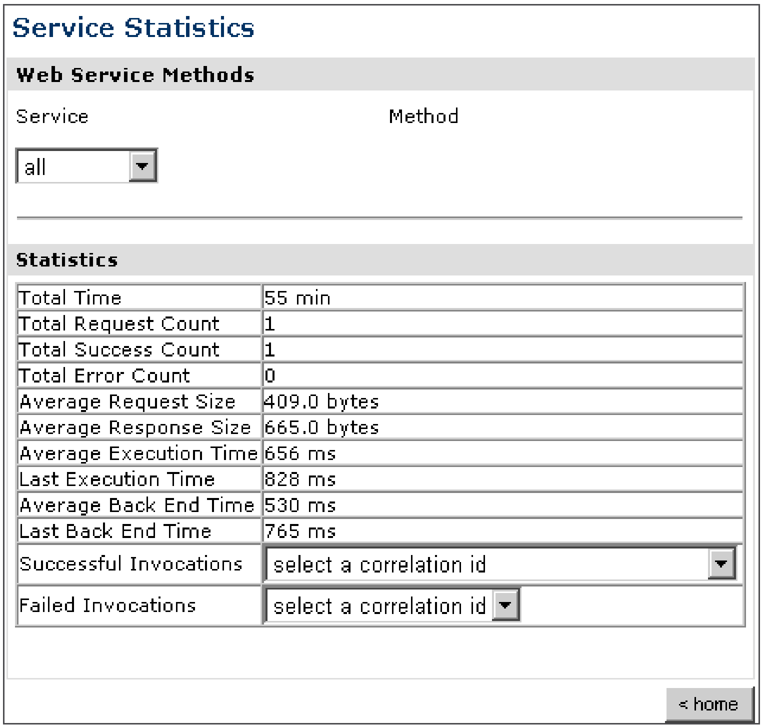
5. To stop monitoring, click *Stop Monitoring*.

Procedure: How to Monitor Services

To monitor services:

1. Ensure that your BEA WebLogic Server is started.
2. From the iBSE Monitoring Settings window, click *Start Monitoring*.
3. Click *View Services*.

   The System Level Summary (Service Statistics) window opens, as shown in the following image. The Web Service Methods pane contains a drop-down list where you select a service. On the right, space is reserved for a drop-down list of methods that will appear. The Statistics pane contains a table with a summary of service statistics and two drop-down lists where you can select a successful or failed invocation to view more information about that service. At the bottom of the window is a home button to click to return to the iBSE Monitoring Settings window.



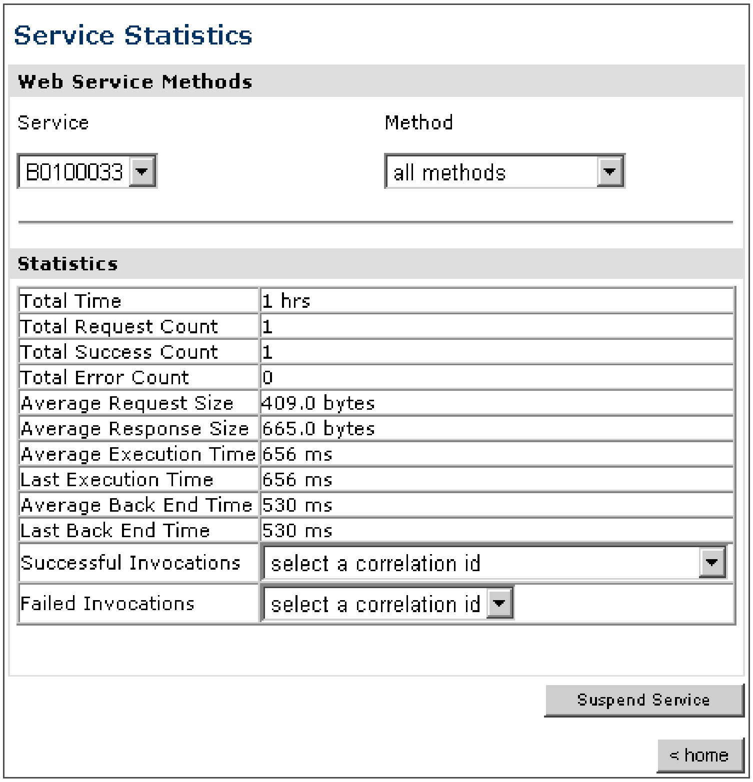
The system level summary provides services statistics at a system level.

The following table lists and describes each service statistic.

Statistic	Description
Total Time	Total amount of time iBSE monitors services. The time starts after you click Start Monitoring in the iBSE Monitoring Settings window.
Total Request Count	Total number of services requests made during the monitoring session.
Total Success Count	Total number of successful service executions.
Total Error Count	Total number of errors encountered.
Average Request Size	Average size of an available service request.
Average Response Size	Average size of an available service response.
Average Execution Time	Average execution time for a service.
Last Execution Time	Last execution time for a service.
Average Back End Time	Average back-end time for a service.
Last Back End Time	Last back-end time for a service.
Successful Invocations	Successful services arranged by correlation ID. To retrieve more information for a service, select the service from the drop-down list.
Failed Invocations	Failed services arranged by correlation ID. To retrieve more information for a service, select the service from the drop-down list.

4. Select a service from the drop-down list.

   The System Level Summary (Service Statistics) window opens as shown in the following image. The Web Service Methods pane contains a drop-down list on the left where you select a service and a drop-down list on the right where you select a service method. The Statistics pane contains a table with a summary of service statistics and two drop-down lists. To view more information about that service, you can select it from the Successful Invocations or Failed Invocations drop-down list. To suspend or resume a service, you can click a button in the lower right. To return to the iBSE Monitoring Settings window, you click the home button.

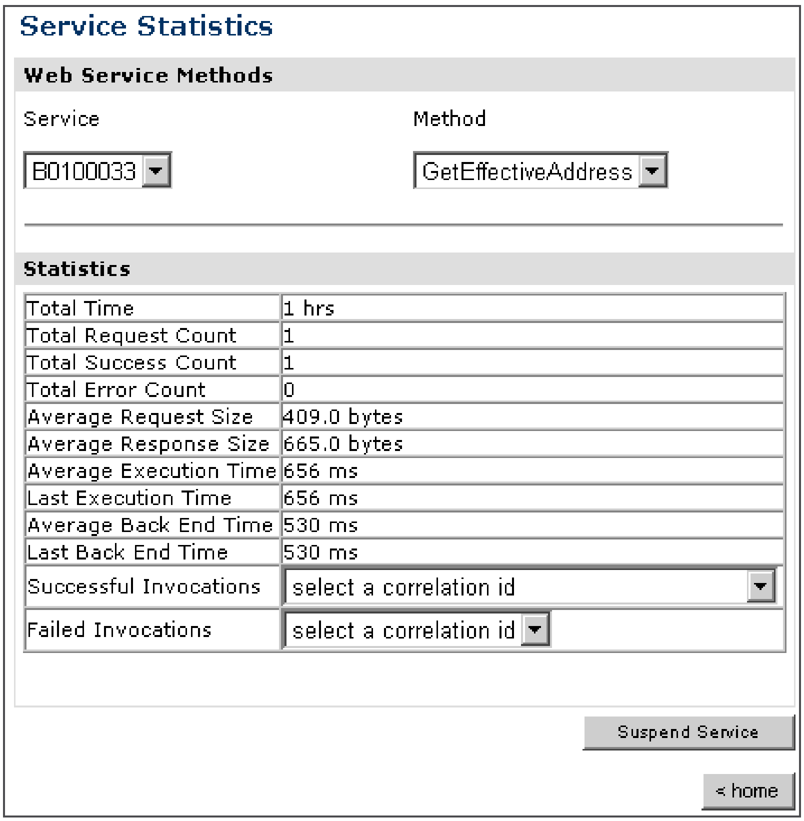


   a. To stop a service at any time, click *Suspend Service*.

   b. To restart the service, click *Resume Service*.

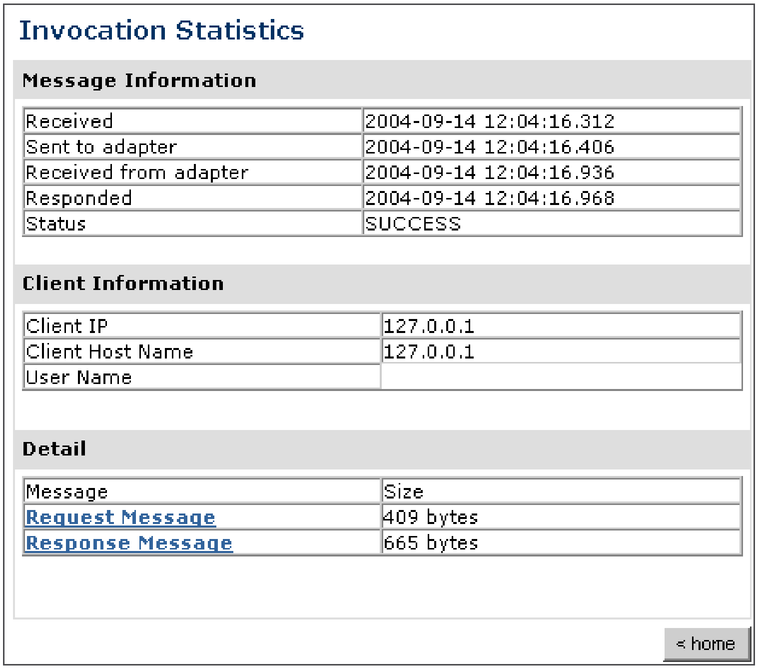
5. Select a method for the service from the Method drop-down list.

   The Method Level Summary (Service Statistics) window opens as shown in the following image. The Web Service Methods pane contains a drop-down list on the left where you select a service and a drop-down list on the right where you select a service method. The Statistics pane contains a table with a summary of service statistics and two drop-down lists. To view more information about that service, you can select it from the Successful Invocations or Failed Invocations drop-down list. To suspend or resume a service, you can click a button in the lower right. To return to the iBSE Monitoring Settings window, you click the home button.



6. For additional information about a successful service and its method, select a service based on its correlation ID from the Successful Invocation drop-down list.

The Invocation Level Statistics window opens as shown in the following image. The Message Information pane contains a table of information about the message. The Client Information pane contains a table of information about the client. The Detail pane contains a table that shows the size of the request and response messages.



7. To view the XML request document in your Web browser, click *Request Message*.

You can also view the XML response document for the service.

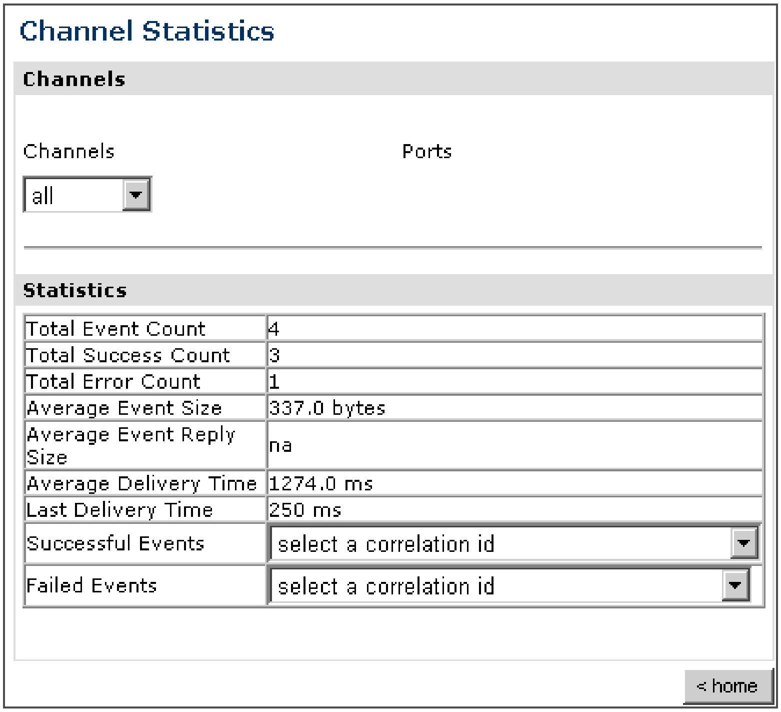
8. To return to the iBSE Monitoring Settings window, click *home*.

Procedure: How to Monitor Events

To monitor events:

1. Ensure that your BEA WebLogic Server is started.
2. In the iBSE Monitoring Settings window, click *Start Monitoring*.
3. Click *View Events*.

   The System Level Summary (Channel Statistics) window opens as shown in the following image. The Channels pane contains a drop-down list on the left where you select a channel. On the right, space is reserved for a drop-down list of ports that will appear. The Statistics pane contains a table with a summary of event statistics and two drop-down lists where you can select a successful or failed event to view more information about that event. In the lower right of the window is a home button.



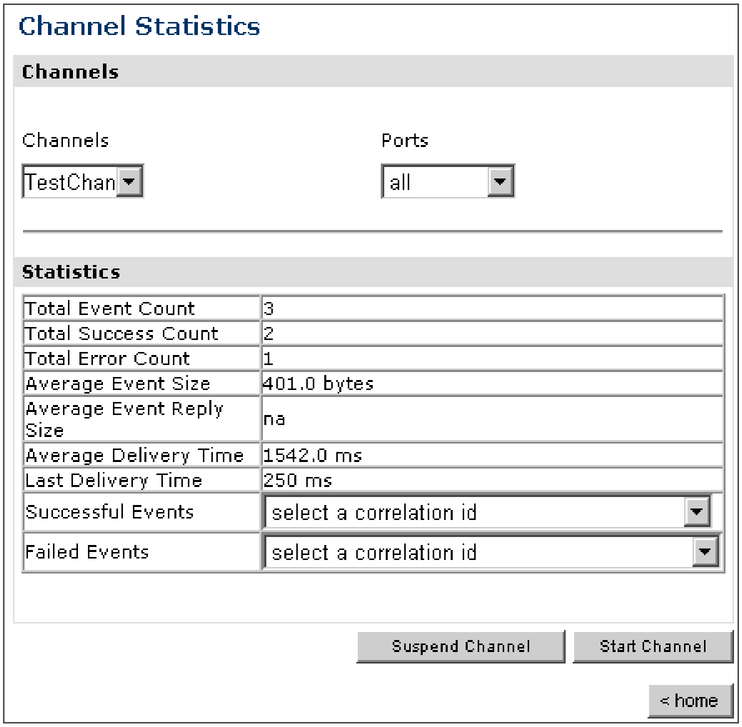
The system level summary provides event statistics at a system level.

The following table lists and describes each event statistic.

Statistic	Description
Total Event Count	Total number of events.
Total Success Count	Total number of successful event executions.
Total Error Count	Total number of errors encountered.
Average Event Size	Average size of an available event request.
Average Event Reply Size	Average size of an available event response.
Average Delivery Time	Average delivery time for an event.
Last Delivery Time	Last delivery time for an event.
Successful Events	Successful events arranged by correlation ID. To retrieve more information for an event, select the event from the drop-down list.
Failed Events	Failed events arranged by correlation ID. To retrieve more information for an event, select the event from the drop-down list.

4. Select a channel from the drop-down list.

The Channel Level Event Summary (Channel Statistics) window opens as shown in the following image. The Channels pane contains a drop-down list on the left where you select a channel and a drop-down list on the right where you select a port. The Statistics pane contains a table with a summary of event statistics and two drop-down lists where you can select a successful or failed event to view more information about that event. In the lower right of the window is a button to click to suspend or resume a channel and a home button to click to return to the iBSE Monitoring Settings window.

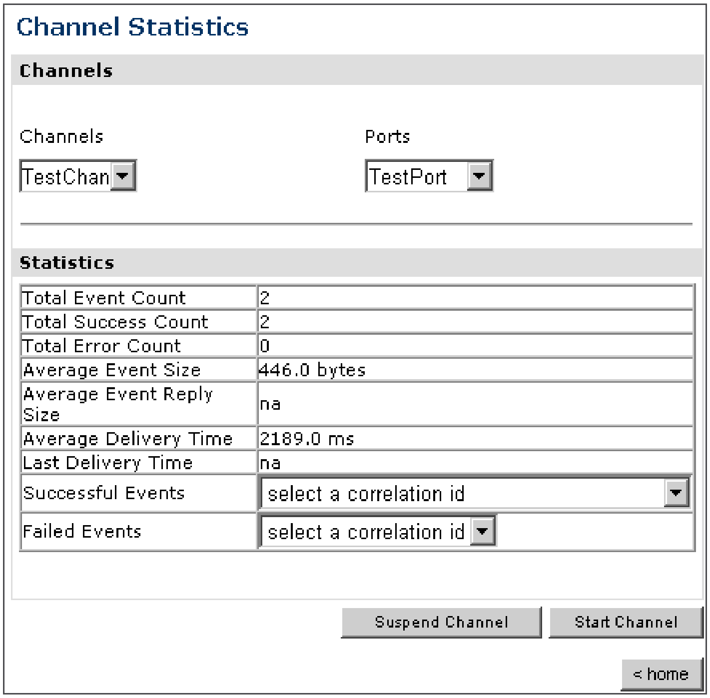


a. To stop a channel at any time, click *Suspend Channel*.

b. To start the channel, click *Start Channel*.

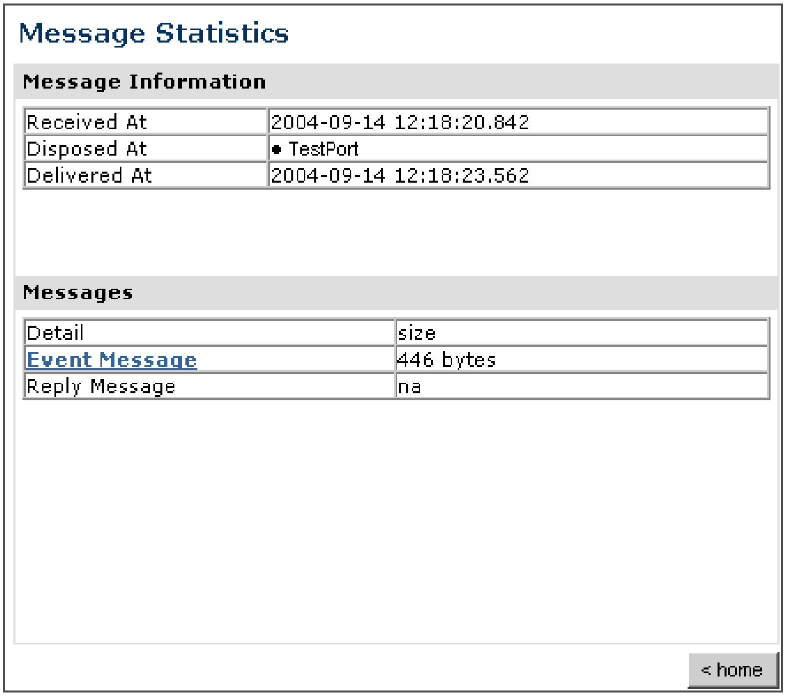
5. From the Ports drop-down list, select a port for the channel.

The Port Level Event Summary (Channel Statistics) window opens as shown in the following image. The Channels pane contains a drop-down list on the left where you select a channel and a drop-down list on the right where you select a port. The Statistics pane contains a table with a summary of event statistics and two drop-down lists where you can select a successful or failed event to view more information about that event. In the lower right of the window is a button to click to suspend or resume a channel and a home button to click to return to the iBSE Monitoring Settings window.



6. For more information about a successful event and its port, select an event based on its correlation ID from the Successful Events drop-down list.

   The Event Level Statistics (Message Statistics) window opens as shown in the following image. The Message Information pane contains a table of information pertaining to the event message. The Messages pane contains a table that shows the size of the event and reply messages.



   a. To view the XML event document in your Web browser, click *Event Message*.

   b. To return to the iBSE Monitoring Settings window, click *home*.

Managing and Monitoring Services and Events Using the JCA Test Tool

The JCA Test Tool, which is also known as the JCA Installation Verification Program (IVP), provides a console to manage and monitor services and events currently in use. The console also displays resource usage and invocation statistics. These indicators can help you adjust your environment for optimum efficiency.

Procedure: How to Manage and Monitor Services Using the JCA Test Tool

To manage and monitor services using the JCA Test Tool:

1. Open a Web browser to:

   `http://hostname:port/iwjcaivp`

   where:

   hostname

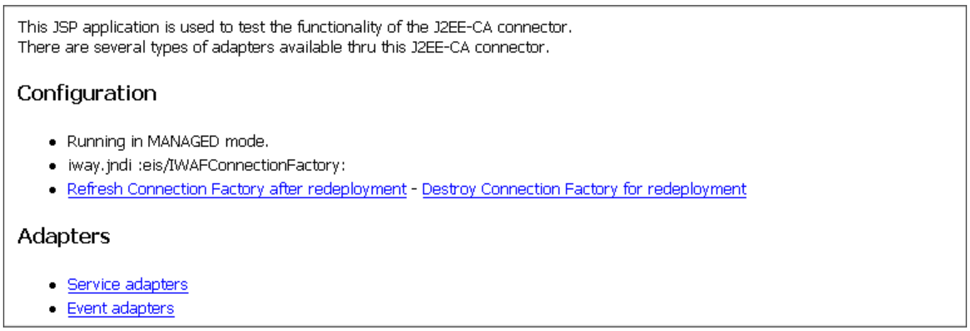
   Is the name of the machine where your application server is running.

   port

   Is the port for the domain you are using. The port for the default domain is 7001, for example:

   `http://localhost:7001/iwjcaivp`

   The following image shows the JCA Test Tool pane that opens. The pane contains a description of the function of the tool and configuration information, including options to change your connection settings. It also provides options for viewing service or event adapters.



   The JCA Test Tool runs in managed mode by default.

2. Perform the following steps to monitor the latest service adapter configuration.

   Note: You must perform these steps for every new adapter target that is created using a JCA implementation of Application Explorer. In addition, you also must perform these steps for every new JCA configuration that is created using Application Explorer.

   a. Click *Destroy Connection Factory for redeployment*.

   b. Redeploy the JCA connector.

   c. In the JCA Test Tool, click *Refresh Connection Factory after redeployment*.

3. Click *Service adapters*.

4. Select a service adapter to monitor.

   a. Click the desired target for your service adapter.

   b. In the Request area, enter a user name and password.

   c. In the Input Doc area, enter a request document created from the request schema for your service.

5. Click *Send*.

   The following image shows the updated statistics that appear for your service if the request is successful. The statistics include the total number of requests, successes, and errors and the average and last execution time in milliseconds.

TotalRequestCount	: 1
TotalSuccessCount	: 1
TotalErrorCount	: 0
AverageExcecutionTime	: 857 msec.
LastExcecutionTime	: 857 msec.

Procedure: How to Manage and Monitor Events Using the JCA Test Tool

To manage and monitor events using the JCA Test Tool:

1. Open a Web browser to:

   `http://`*`hostname`*`:`*`port`*`/iwjcaivp`

   where:

   `hostname`

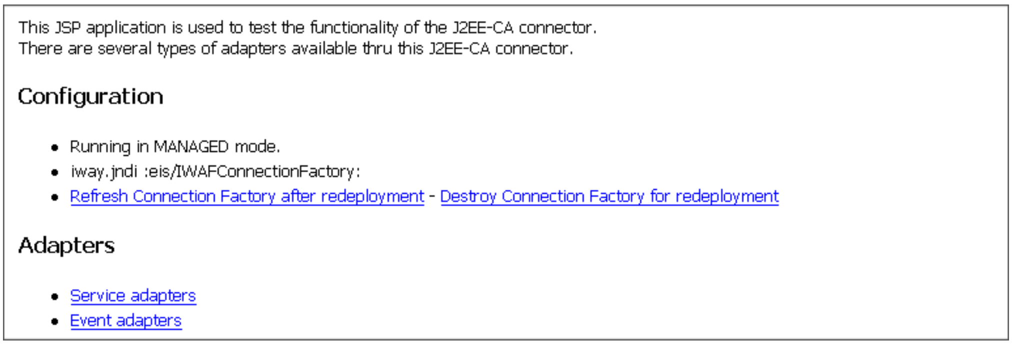
   Is the name of the machine where your application server is running.

   `port`

   Is the port for the domain you are using. The port for the default domain is 7001, for example:

   `http://localhost:7001/iwjcaivp`

   The JCA Test Tool pane opens as shown in the following image. The pane contains a description of the function of the tool and configuration information, including options to change your connection settings. It also provides options for viewing service or event adapters.



   The JCA Test Tool runs in managed mode by default.

2. To monitor the latest event adapter configuration, perform the following steps.

   Note: You must perform these steps for every new adapter target that is created using a JCA implementation of Application Explorer. In addition, you must also perform these steps for every new JCA configuration that is created using Application Explorer.

   a. Click *Destroy Connection Factory for redeployment*.

   b. Redeploy the JCA connector.

   c. In the JCA Test Tool, click *Refresh Connection Factory after redeployment*.

3. Click *Event adapters*.

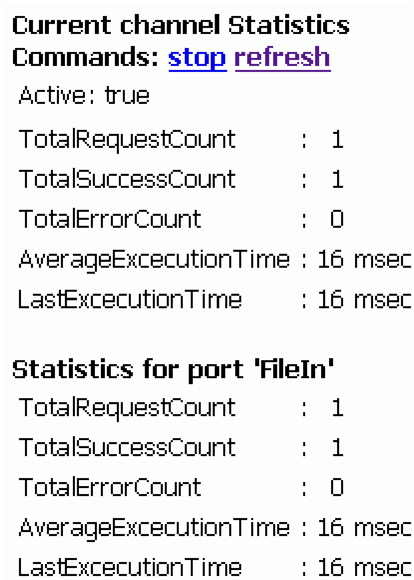
   The Event Adapters pane opens.

4. Select the event adapter to monitor.

5. Click the desired channel for your event adapter.

6. Click *start*.

   The following image shows the updated statistics for your channel and the port. The statistics include the total number of requests, successes, and errors and the average and last execution time in milliseconds. The upper right of the pane contains options to start or refresh the channel.



Setting Engine Log Levels

The following procedures describe how to set engine log levels. For more information, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual.

Procedure: How to Enable Tracing for Servlet iBSE

To enable tracing for Servlet iBSE:

1. Open the Servlet iBSE configuration window at:

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

   where:

   hostname
   : Is the name of the machine where your application server is running.

   port
   : Is the port for the domain you are using. The port for the default domain is 7001, for example:

   `http://localhost:7001/ibse/IBSEConfig`

2. In the System pane, from the Debug drop-down list, select the level of tracing.
3. Click *Save*.

The default location for the trace information on Windows is:

`C:\Program FIles\bea\ibse\ibselogs`

Procedure: How to Enable Tracing for JCA

To enable tracing for JCA:

1. Extract the ra.xml file from the iwafjca.rar file.
2. Open the extracted ra.xml file in a text editor.
3. Locate and change the following setting:

   LogLevel. This setting can be set to DEBUG, INFO, or ERROR.

   ```
   <context-param>
   <config-property>
       <config-property-name>LogLevel</config-property-name>
       <config-property-type>java.lang.String</config-property-type>
       <config-property-value></config-property-value>
   </config-property>
   ```

   For example:

   ```
   <config-property-value>DEBUG</config-property-value>
   ```

   Leave the remainder of the file unchanged.
4. Save the file and exit the editor.
5. Add the file back to the iwafjca.rar file.
6. Redeploy the connector.

A directory in the configuration directory contains the logs. For example, on Windows the default location is the following:

```
C:\Program Files\iway55\config\base\log
```

The log files are named jca_*.log.

You should also review the logs generated by your application server.

Configuring Connection Pool Sizes

The following topic describes how to configure connection pool sizes for the Connector for JCA.

Procedure: How to Configure Connection Pool Sizes

To configure connection pool sizes:

1. Extract the ra.xml file from the iwafjca.rar file.
2. Open the extracted ra.xml file in a text editor.
3. Locate and change the following setting:

   pool-params. The JCA Resource Connector has an initial capacity value of 0 by default and cannot be changed. The maximum capacity value is 10 by default and can be changed to a higher value.

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE weblogic-connection-factory-dd (View Source for full
doctype...)>
- <weblogic-connection-factory-dd>
  <connection-factory-name>IWAFJCA</connection-factory-name>
  <jndi-name>eis/IWAFConnectionFactory</jndi-name>
  - <pool-params>
    <initial-capacity>0</initial-capacity>
    <max-capacity>10</max-capacity>
    <capacity-increment>1</capacity-increment>
    <shrinking-enabled>false</shrinking-enabled>
    <shrink-period-minutes>200</shrink-period-minutes>
  </pool-params>
  <security-principal-map />
</weblogic-connection-factory-dd>
```

4. Save the file and exit the editor.
5. Return the ra.xml file to the iwafjca.rar file.
6. Redeploy the connector.

Migrating Repositories

During design time, a repository is used to store metadata that is created when using Application Explorer to:

- Configure adapter connections.
- Browse EIS objects.
- Configure services.
- Configure listeners to listen for EIS events.

For more information on configuring repositories, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual.

The information in the repository also is referenced at run time. For management purposes, you can migrate iBSE and JCA repositories to new destinations without affecting your existing configuration. For example, you can migrate a repository from a development environment to a production environment. The BEA WebLogic Server must be restarted to detect new repository changes.

For more information on migrating repositories, see the *iWay 5.5.011 for BEA WebLogic 9.1 Migration Guide*.

Exporting or Importing Targets

After you migrate your repository, you can export or import targets with their connection information and persistent data between repositories.

Procedure: How to Export a Target

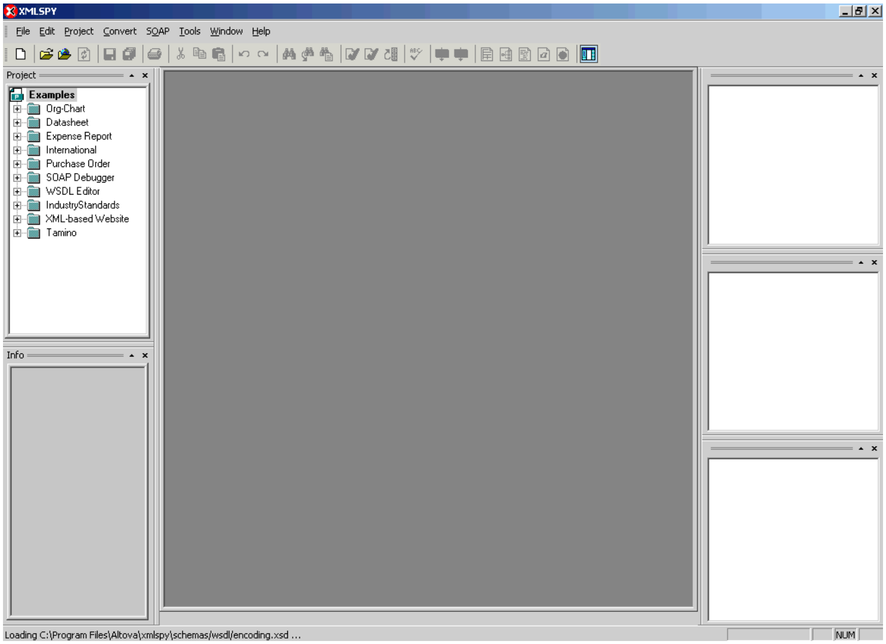
To export a target:

1. Copy the iBSE administrative services for Application Explorer URL, for example:

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

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

The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE administrative services for Application Explorer URL.

5. Click *OK*.

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

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

   A window opens that shows the structure of the SOAP envelope.

7. Locate the *Text view* icon in the tool bar.

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:EXPORTTARGET
xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:af">
<m:target>String</m:target>
<m:name>String</m:name>
</m:EXPORTTARGET>
```

a. For the <m:target> tag, replace the String placeholder with the EIS target system name as it appears in Application Explorer and verify whether this value is case sensitive.

b. For the <m:name> tag, replace the String placeholder with the name of the target you want to export.

10. From the SOAP menu, select *Send request to server.*

A response is returned that contains the <m: exporttime> and <m: contents> elements. You must use these elements when importing your target.

Procedure: How to Import a Target

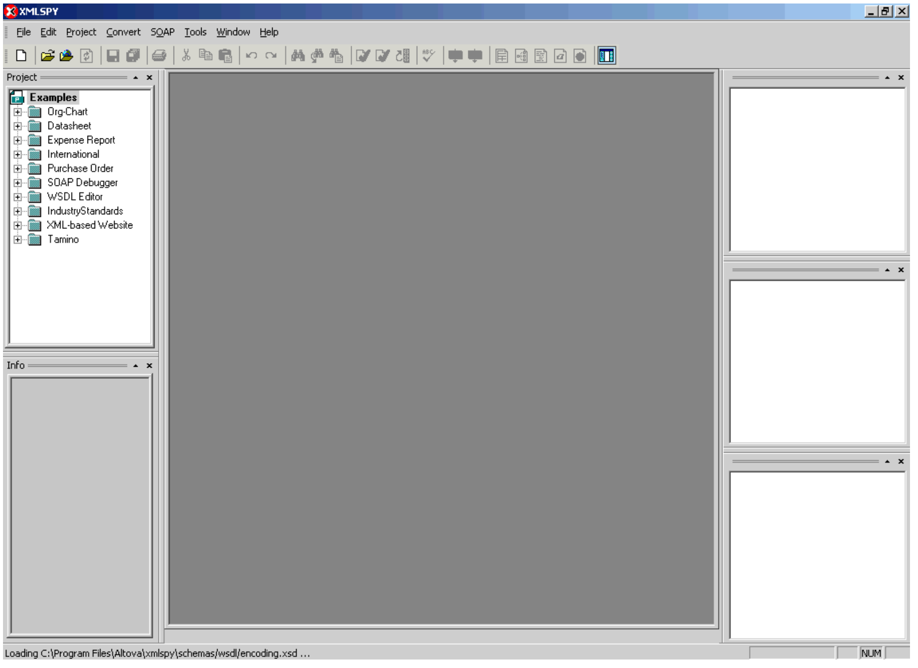
To import a target:

1. Copy the iBSE administrative services for Application Explorer URL, for example:

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

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

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE administrative services for Application Explorer URL and click *OK*.

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

5. Select the *IMPORTTARGET(IMPORTTARGET parameters)* control method and click *OK*.

   A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the tool bar.
7. 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.
8. Locate the following section:

```
<m:IMPORTTARGET
xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:af">
<m:targetinstance>
<m:target>String</m:target>
<m:name>String</m:name>
<m:description>String</m:description>
<m:repositoryid>String</m:repositoryid>
<m:exporttime>2001-12-17T09:30:47-05:00</m:exporttime>
<m:contents>R0lGODlhcgGSALMAAAQCAEMmCZtuMFQxDS8b</m:contents>
</m:targetinstance>
</m:IMPORTTARGET>
```

   a. For the <m:target> tag, replace the String placeholder with the EIS target system name.
   b. For the <m:name> tag, replace the String placeholder with the new name of the target you want to import.
   c. For the <m:description> tag, replace the String placeholder with a description of the target.
   d. For the <m:repositoryid> tag, copy and paste the contents of the <m:repositoryid> tag that was returned when you exported your target.
   e. For the <m: exporttime> tag, copy and paste the contents of the <m: exporttime> tag that was returned when you exported your target.
   f. For the <m: contents> tag, copy and paste the contents of the <m: contents> tag that was returned when you exported your target.
9. From the SOAP menu, select *Send request to server.*

Retrieving or Updating Web Service Method Connection Information

After you migrate your repository, you can retrieve or update connection information for your Web service methods.

Procedure: How to Retrieve Web Service Method Connection Information

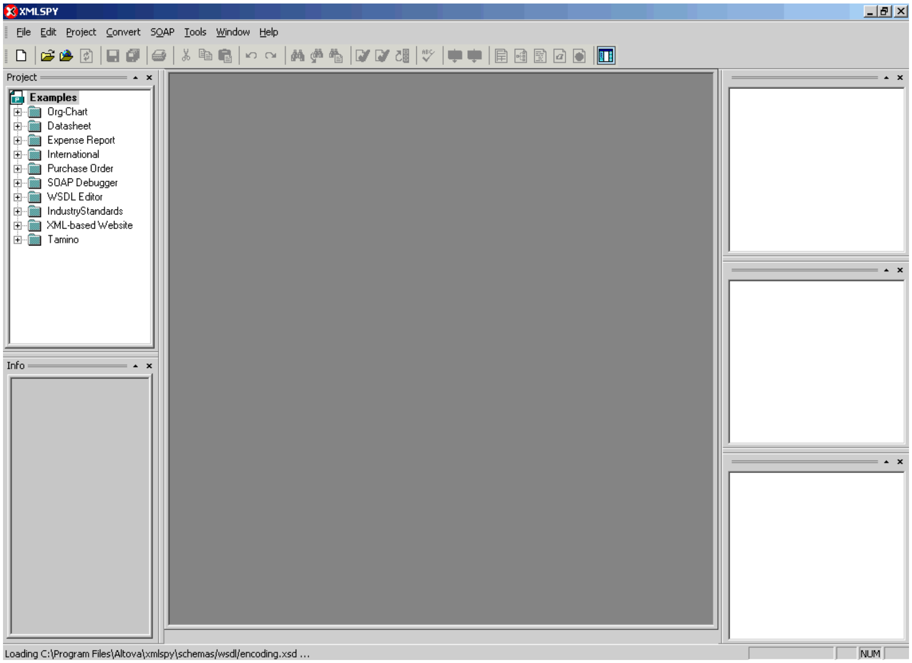
To retrieve Web service method connection information:

1. Copy the iBSE configuration service URL, for example:

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

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

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE configuration service URL, and click *OK*.

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

5. Select the *GETMTHCONNECTION(GETMTHCONNECTION parameters)* control method and click *OK*.

   A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the tool bar.

7. 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.

8. Locate the following section:

```
<m:GETMTHCONNECTION
xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config">
<m:servicename>String</m:servicename>
<m:methodname>String</m:methodname>
</m:GETMTHCONNECTION>
```

   a. For the <m:servicename> tag, replace the String placeholder with the name of the Web service.

   b. For the <m:methodname> tag, replace the String placeholder with name of the Web service method.

9. From the SOAP menu, select *Send request to server.*

   A response is returned that contains the <m: descriptor> element. You must use this element when updating your Web service method.

Procedure: How to Update Web Service Method Connection Information

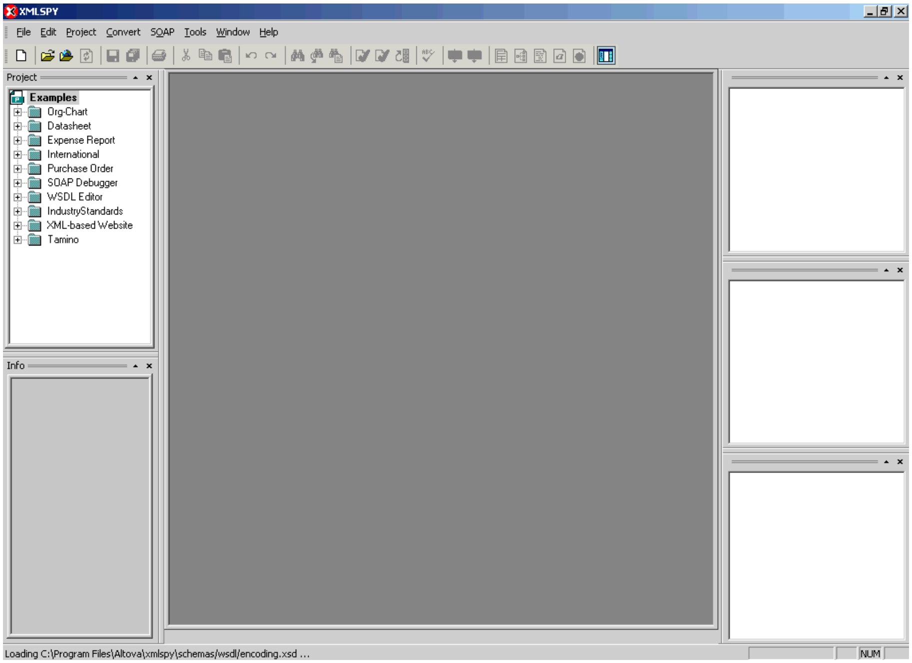
To update Web service method connection information:

1. Copy the iBSE configuration service URL, for example:

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

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

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.



3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE configuration service URL, and click *OK*.

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

5. Select the *SETMTHCONNECTION(SETMTHCONNECTION parameters)* control method and click *OK*.

   A window opens that shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the tool bar.
7. 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.
8. Locate the following section:

```
<m:SETMTHCONNECTION
xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config">
<m:servicename>String</m:servicename>
<m:methodname>String</m:methodname>
<m:descriptor format="" channel="">
     <m:option title="">
       <m:group title="">
          <m:param/>
       </m:group>
     </m:option>
</m:descriptor>
</m:SETMTHCONNECTION>
```

   a. For the <m:servicename> tag, replace the String placeholder with the name of the Web service.
   b. For the <m:methodname> tag, replace the String placeholder with the name of the Web service method.
   c. For the <m: descriptor> tag, copy and paste the contents of the <m: descriptor> tag that was returned when you retrieved Web Service method connection information.
9. Modify the contents of the <m: descriptor> tag to change the existing Web Service method connection information.
10. From the SOAP menu, select *Send request to server.*

Starting or Stopping a Channel Programmatically

The following topic describes how to start or stop a channel programmatically.

Procedure: How to Start a Channel Programmatically

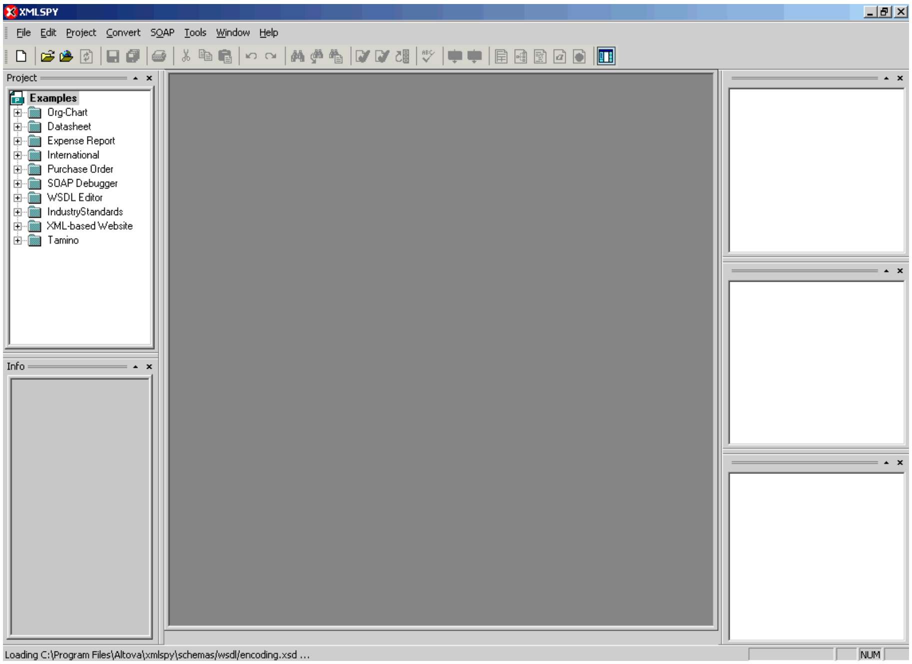
To start a channel programmatically:

1. Copy the iBSE control event URL, for example:

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

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

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.

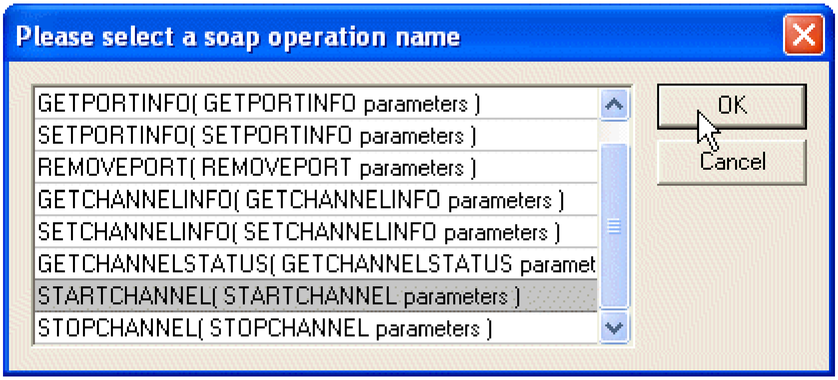


3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE control event URL, and click *OK*.

   The following image shows the soap operation name dialog box that opens with a list of available control methods.



5. Select the *STARTCHANNEL(STARTCHANNEL parameters)* control method and click *OK*.

   A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the tool bar.

7. 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.

8. Locate the following section:

```
<SOAP-ENV:Body>
  <m:STARTCHANNEL
     xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:event">
           <m:channel>String</m:channel>
  </m:STARTCHANNEL>
</SOAP-ENV:Body>
```

9. For the <m:channel> tag, replace the String placeholder with the name of the channel you want to start.

10. From the SOAP menu, select *Send request to server.*

Procedure: How to Stop a Channel Programmatically

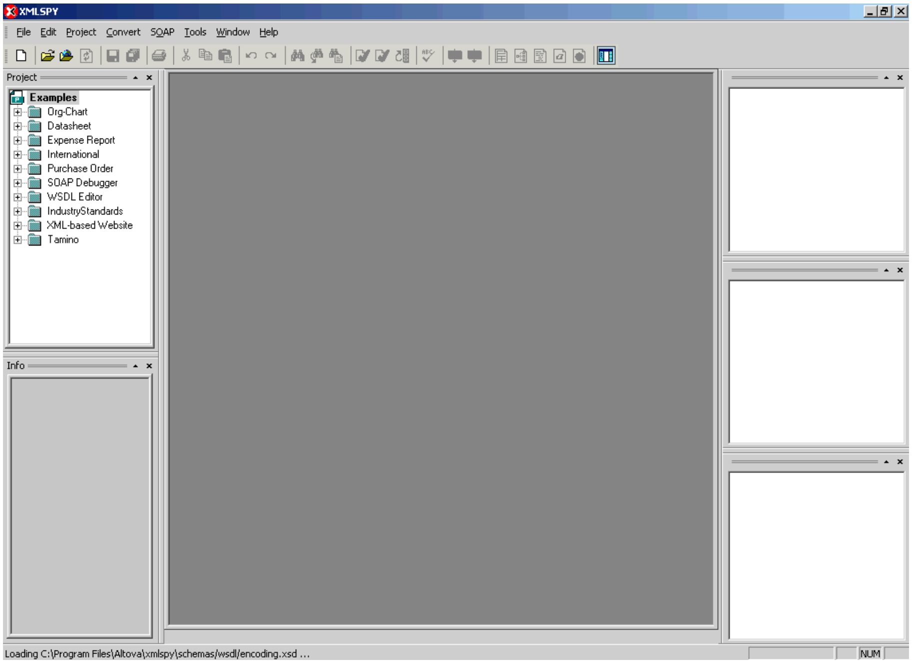
To stop a channel programmatically:

1. Copy the iBSE control event URL, for example:

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

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

   The following image shows the XMLSPY window. The upper left has a Project pane that contains a tree of sample files, and the lower left has a blank Info pane. The middle pane is blank. The right side is divided into three blank panes.

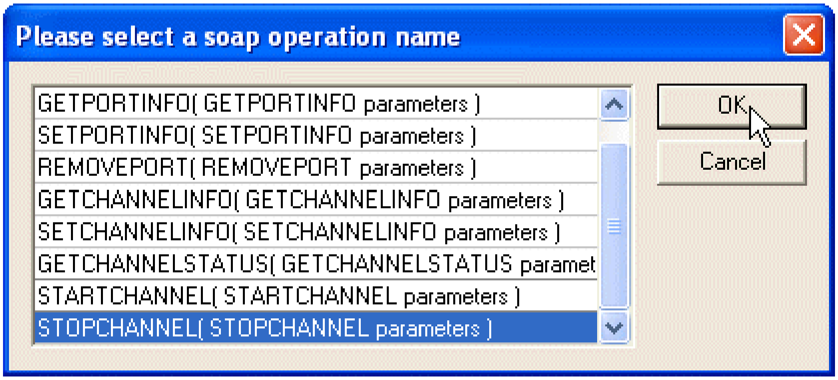


3. From the SOAP menu, select *Create new SOAP request*.

   The WSDL file location dialog box opens.

4. In the Choose a file field, paste the iBSE control event URL, and click *OK*.

   The following image shows the soap operation name dialog box that opens with a list of available control methods.



5. Select the *STOPCHANNEL(STOPCHANNEL parameters)* control method and click *OK*.

   A window opens, which shows the structure of the SOAP envelope.

6. Locate the *Text view* icon in the tool bar.

7. 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.

8. Locate the following section:

```
<SOAP-ENV:Body>
  <m:STOPCHANNEL
     xmlns:m="urn:schemas-iwaysoftware-com:dec2002:iwse:event">
           <m:channel>String</m:channel>
  </m:STOPCHANNEL>
</SOAP-ENV:Body>
```

9. For the <m:channel> tag, replace the String placeholder with the name of the channel you want to stop.

10. From the SOAP menu, select *Send request to Server.*

CHAPTER 8

Troubleshooting and Error Messages

Topics:

- Error Messages in Application Explorer
- Error Messages in PeopleSoft
- Error Messages in JCA
- Error Messages in iBSE

This section provides troubleshooting information for the following four categories:

- Application Explorer
- PeopleSoft
- JCA
- iBSE

The following topics explain limitations and workarounds when connecting to PeopleSoft. The adapter-specific errors described in this section can arise whether using the adapter with a JCA or with an iBSE configuration.

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

- The JCA trace information can be found under the C:\Program Files\iWay55\config\base\log directory.
- iBSE trace information can be found under the C:\Program Files\iWay55\bea\ibse\ibselogs directory.
- The log file for Application Explorer can be found under the C:\Program File\iWay55\tools\iwae\bin directory.

Error Messages in Application Explorer

The following table lists errors and solutions when using Application Explorer with the BEA WebLogic Adapter for PeopleSoft.

Error	Solution
Cannot connect to the BEA WebLogic Adapter for PeopleSoft from Application Explorer. The following error message appears: `Problem activating adapter`	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. The psjoa.jar file version is specific to the PeopleTools release.
PeopleSoft does not appear in the Application Explorer Adapter node list.	Ensure that the PeopleSoft JAR files, iwpsci84.jar (or iwpsci81.jar) and psjoa.jar, are added to the lib directory.
Logon failure error at run time	If the password for connecting to your PeopleSoft system is not specified when creating a target or with the Edit option in Application Explorer, you cannot connect to PeopleSoft. The connection password is not saved in repository.xml. Update the password using the Edit option in Application Explorer and then restart the application server.

Error	**Solution**
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 do not appear 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 view any Component Interfaces or Messages, then both the iwpsci81.jar and iwpsci84.jar files might be in your lib directory. Stop your server, remove the JAR file that is not required, and restart your server.
Pstools.properties file was not initialized.	The Pstools.properties file is required for the PeopleSoft 8.1 release. If you are using PeopleSoft 8.1, then you must add this file. This error message may appear, even if you are not using PeopleSoft 8.1. In this case, ignore the error message.

Error Messages in PeopleSoft

The following table lists errors and solutions when using PeopleSoft.

Error	Solution
Services are not working properly when using the PeopleSoft Component Interface test tool in three-tier mode.	To test properly using the Component Interface test tool: **1.** Open Application Designer. **2.** Select the Component Interface. **3.** Use the test tool. If the service works in test tool, then review the XML and check for redundant fields in XML.
Component Interfaces and Messages do not appear in the adapter tree.	The project is not installed properly on the PeopleSoft system.
Return error code -1 is received from PeopleSoft at run time, for example: <LOCATIONProcessResponse xmlns="http://xmlns.oracle.com/LOCATION"> <error xmlns="">-1</error> </LOCATIONProcessResponse>	You are either using the incorrect version of psjoa.jar, or you have both the iwpsci81.jar and iwpsci84.jar files in your lib directory. In the second case, you must delete the unused JAR file and then restart the server. The psjoa.jar file version is specific to the PeopleTools release.
Pstools.properties file was not initialized.	This file is required for PeopleSoft 8.1. If you are using PeopleSoft 8.1, you should add this file. If you are not using PeopleSoft 8.1 and this error message still appears, ignore the message.
Cannot find Component Interface {CI name}	The reason may be either of the following: • The Java API for the selected component interface is not found in the API JAR file. Check the Java API for the class file for the CI. If not found, add the class file for the CI. • The component interface name is mentioned incorrectly in the request document.

Error	Solution
`Not Authorized (90,6) Failed to execute PSSession request`	The component interface does not have the access required to perform the operation. Change the permission settings in the PeopleSoft, Security, Permission list for the component interface.
`Must also provide values for keys{keyname}`	The reason may be one of the following: • The request XML document does not have the element for the mandatory key. Include the keyname and the value in the request document. • The Key field name is mentioned incorrectly in the request document. • The Perform operation is mentioned incorrectly in the request XML document.

Error Messages in JCA

The following table describes an error and its corresponding solution for JCA.

Error	Solution
In Application Explorer, the following error message appears when you attempt to connect to a 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, `C:\Program Files\iWay55`

Error Messages in iBSE

This topic discusses the types of errors that can occur when processing Web services through Integration Business Services Engine (iBSE).

General Error Handling in iBSE

Integration Business Services Engine (iBSE) 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 iBSE when Integration Business 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 iBSE passes a SOAP request message to the adapter required for the Web service. If an error occurs, the way 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, when the SOAP agent inside iBSE 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 iBSE 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 the previous example, iBSE did not receive an element in the SOAP request message that is required for the WSDL for this Web service.

Adapter-Specific Error Handling

When an adapter raises an exception during execution, the SOAP agent in iBSE 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.

Because adapters use the target system interfaces and APIs, whether or not an exception is raised depends on how the target system interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in iBSE and that request is invalid based on the WSDL for that service, the adapter may raise an exception yielding a SOAP fault.

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

Example: BEA WebLogic Adapter for PeopleSoft Invalid SOAP Request

When the BEA WebLogic Adapter for PeopleSoft 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>
```

Example: Empty Result From SOAP Request

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

Example: Failure to Connect to PeopleSoft

When the BEA WebLogic Adapter for PeopleSoft cannot connect to PeopleSoft when executing a Web service, 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>
```

Example: Invalid SOAP Request

When the adapter receives a SOAP request message that does not conform to the WSDL for the Web service 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>
```

Example: Empty Result From an Adapter Request

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

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.

```
<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>
```

APPENDIX A

Using Component Interfaces

Topics:

- Creating a Component Interface
- Viewing or Modifying Available Methods
- Securing a Component Interface
- Testing a Component Interface

This section describes how to create new Component Interfaces—and how to modify existing Component Interfaces—for use with the BEA WebLogic Adapter for PeopleSoft. You also can use Component Interfaces supplied by PeopleSoft with your application.

Before using a Component Interface you must apply security to it and test it. After securing and testing a Component Interface, you must generate its API, as described in Chapter 3, *Generating Component Interface APIs*.

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.

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 a property 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.

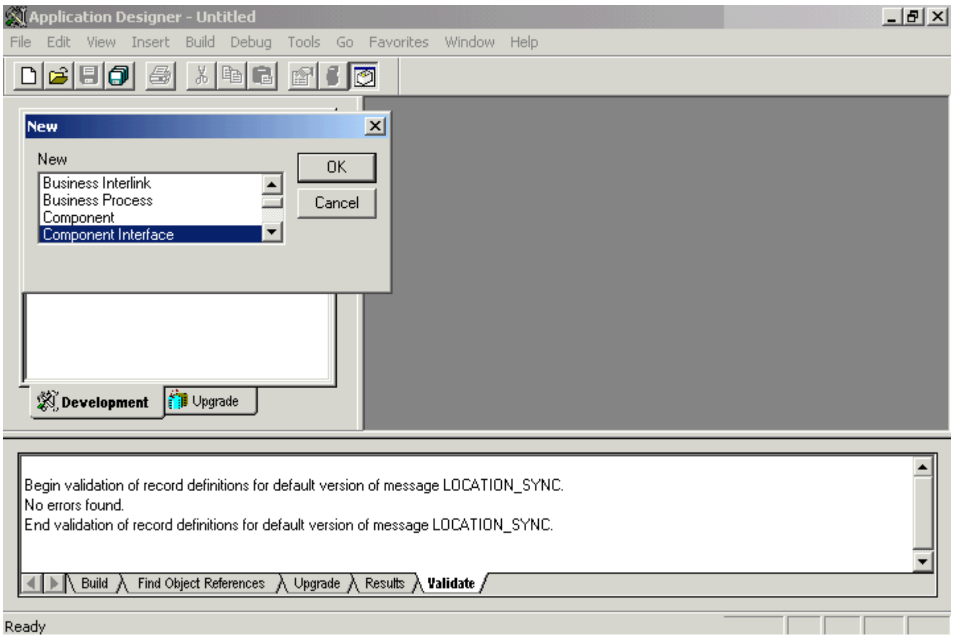
A property may an icon adjacent to it. For example, EMPLID has an icon indicating that it is a key field from the underlying record. NAME has an icon indicating that it is an alternate key field from the underlying record. For a complete list of property icons, see the PeopleBooks documentation.

Procedure: How to Create a New Component Interface

To create a Component Interface:

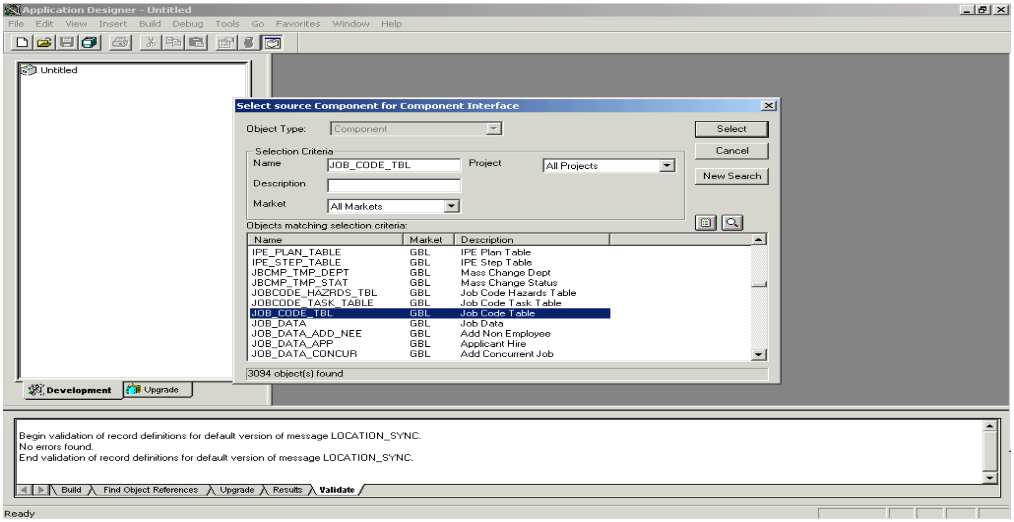
1. Open the PeopleSoft Application Designer.
2. From the File menu, select *New*.

   The New dialog box opens in the PeopleSoft Application Designer, with a list of objects, as shown in the following image.



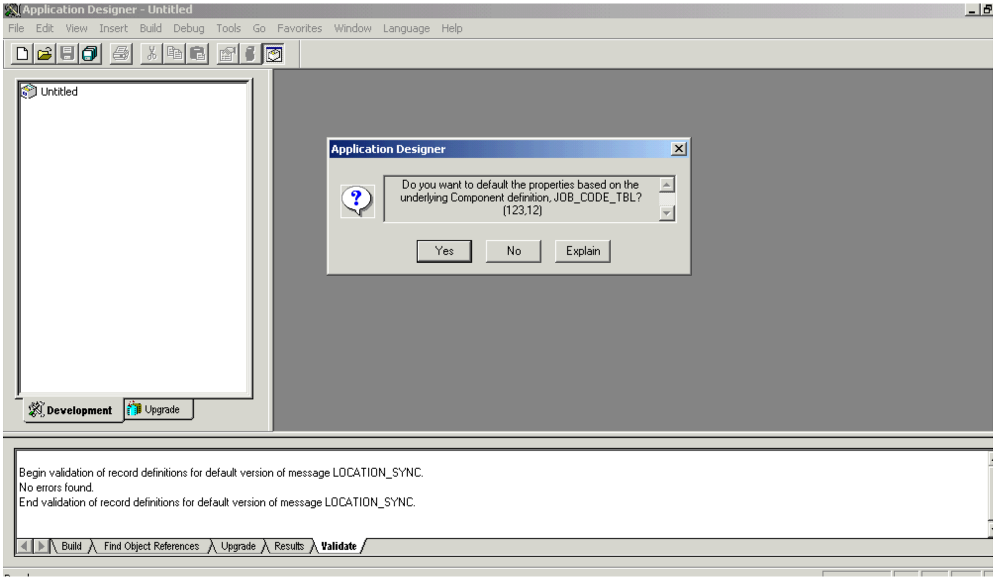
3. Select *Component Interface*.
4. Click *OK*.

   The Select Source Component for Component Interface dialog box opens as shown in the following image. It contains the following: Object Type drop-down list (unavailable), the Selection Criteria pane (which includes a Project drop-down list and Name, Description, and Market fields), and an Objects marking selection criteria pane. It also contains Select, Cancel, and New Search buttons.



5. Highlight the component to use as a basis for the Component Interface and click *Select*.

   The Application Designer dialog box, which asks a question and contains Yes, No, and Explain buttons, opens as shown in the following image.



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

6. Choose one of the following options:

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

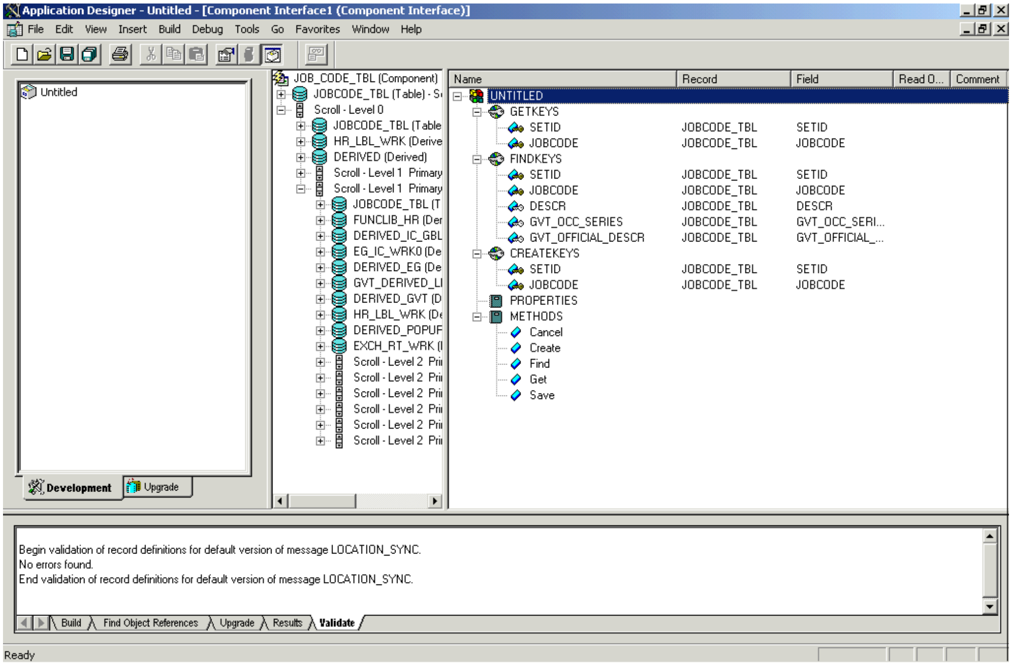
   a. Drag the relevant fields from the left pane to the right pane.

   b. To select the 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.

   To create the Component Interface and to **display the properties of the underlying Component Interface,** click *Yes*.

The following image shows the Application Designer component interface. The underlying properties appear in the right pane in a tree structure with information in columns for Name, Record, Field, Read, and Comments.



Viewing or Modifying Available Methods

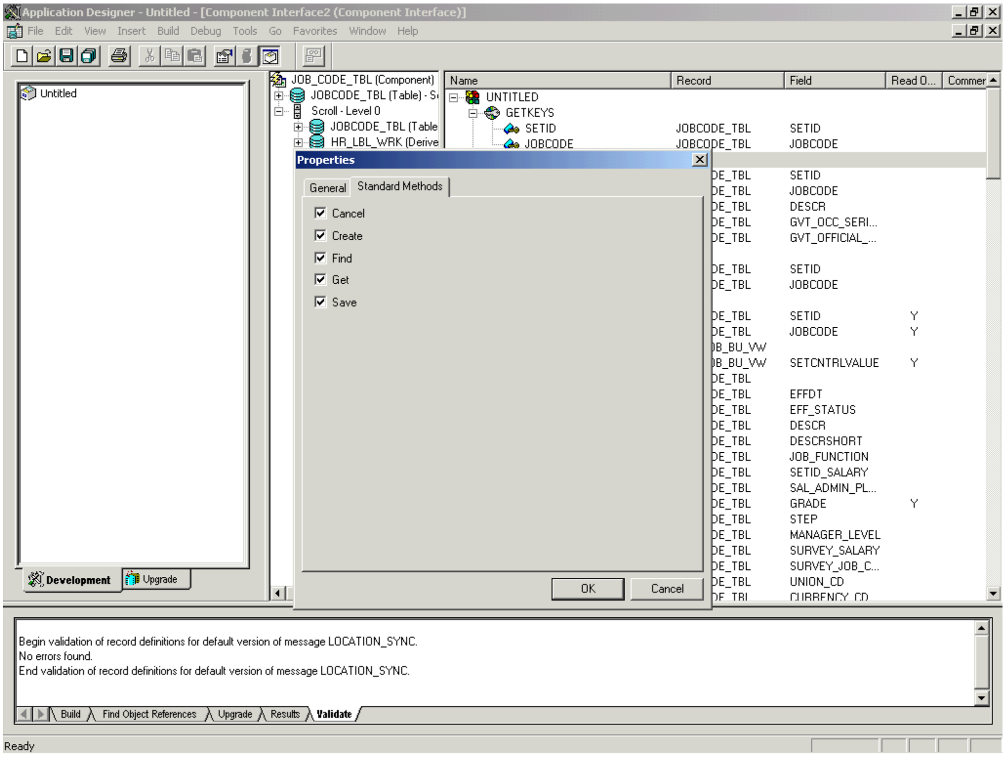
The standard methods for the Component Interface are:

- Create
- Find
- Get
- Save

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

Procedure: How to View or Change Available Methods

The following image shows the Properties dialog box with the Standard Methods tab active. The check boxes for standard methods and for Cancel are selected.



To view or change available methods:

1. Open the Component Interface Properties dialog box.
2. Click the *Standard Methods* tab.
3. Select the desired methods.
4. Click *OK*.

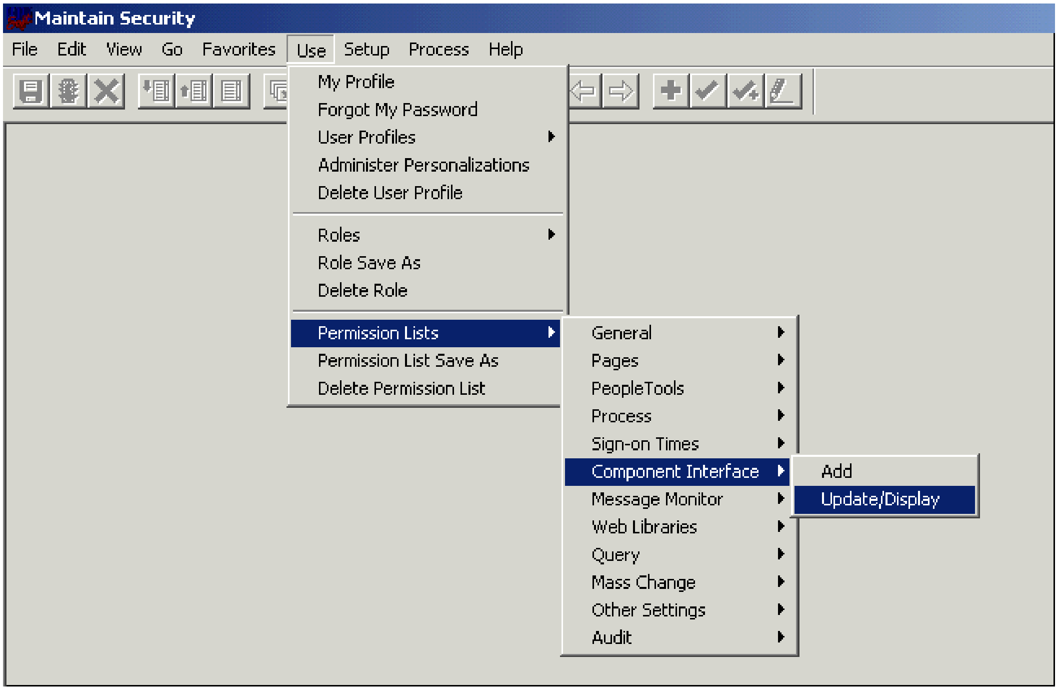
Securing a Component Interface

You must configure security for the Component Interface before you can begin to test.

For information on configuring security for PeopleSoft Version 8.1x in two- and three-tier mode, see *How to Configure Component Interface Security for PeopleSoft Version 8.1x* on page A-7. For information on configuring security for PeopleSoft Version 8.4, see *How to Configure Interface Security for PeopleSoft Version 8.4* on page A-11.

Procedure: How to Configure Component Interface Security for PeopleSoft Version 8.1x

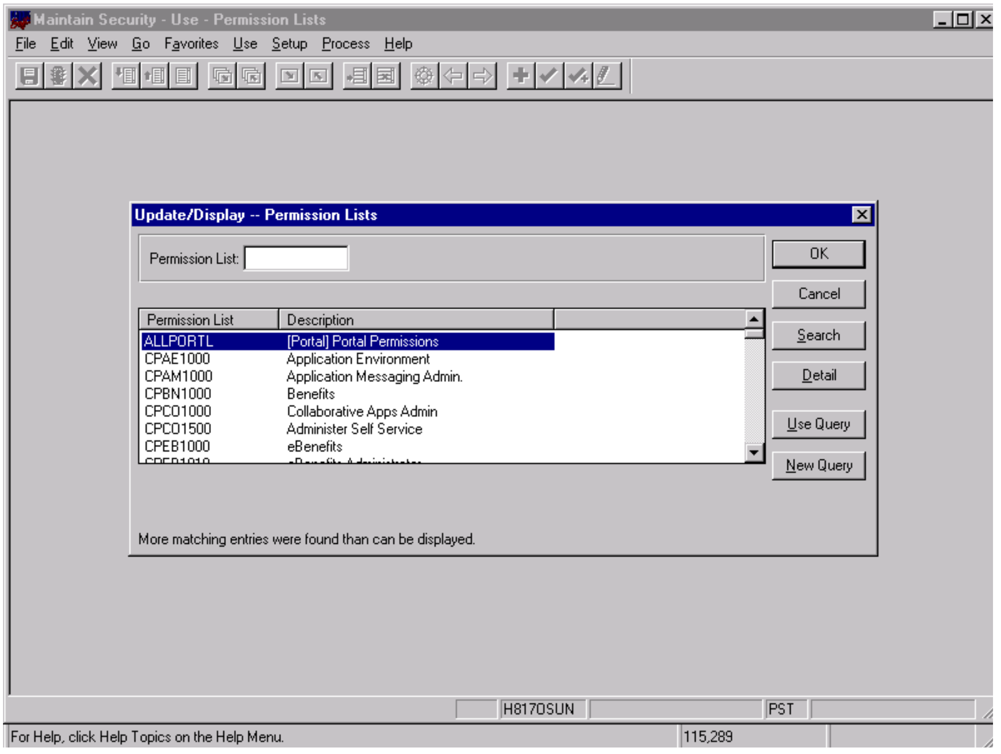
The following image shows the Maintain Security window with the Use menu expanded.



To configure Component Interface security:

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

   The Update/Display Permission Lists dialog box opens as shown in the following image. It contains a Permission List field, Permission List and Description information, and OK, Cancel, Search, Detail, Use Query, and New Query buttons.

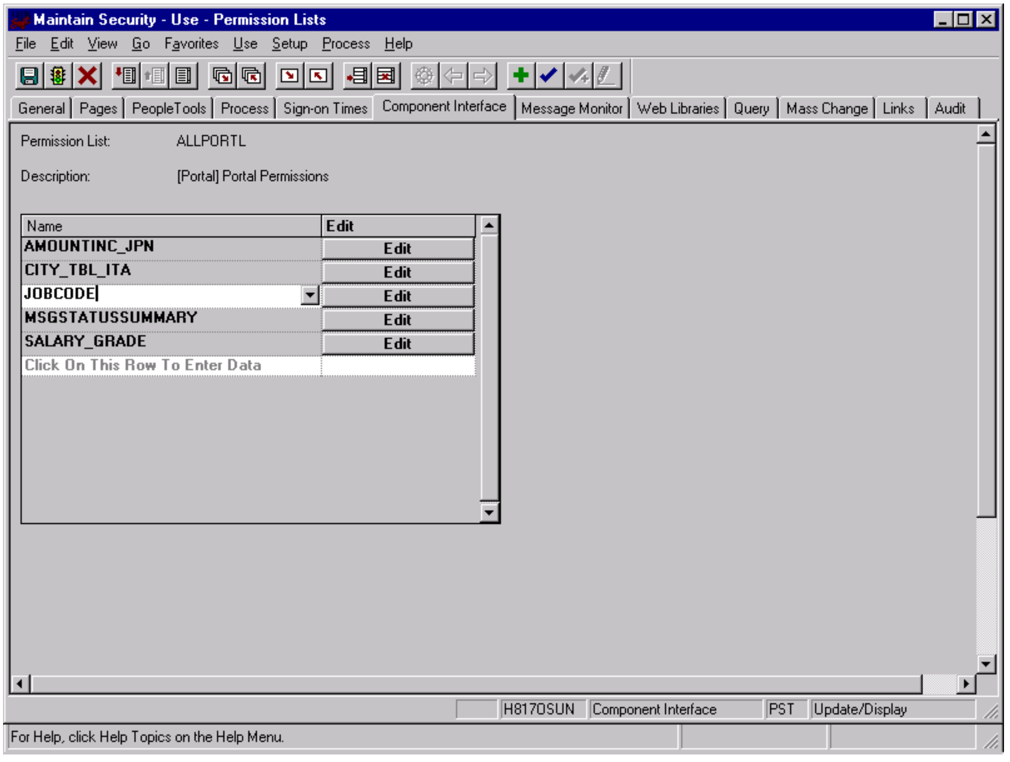


   Before Security can be set, you must identify the permission list(s).

2. Select the relevant permission list and click *OK*.

For information on permission lists, see the PeopleBooks documentation.

The following image shows the Permission Lists pane and contains information about the ALLPORTL permission list in the Name column. The Component Interface tab is active.

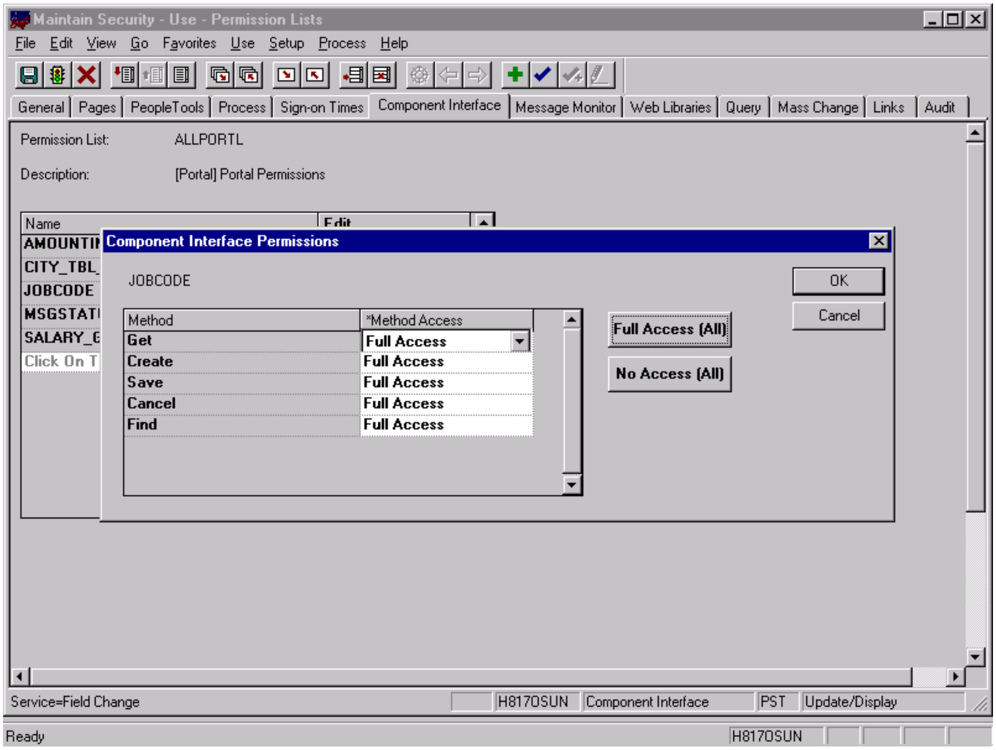


3. To insert the new Component Interface you created, click *Click On This Row To Enter Data*.

4. When the Edit button appears, click *Edit*.

When you select the Component Interface, all available methods appear, including user-defined methods. You can then specify whether this particular Permission List must have Full or Partial Access.

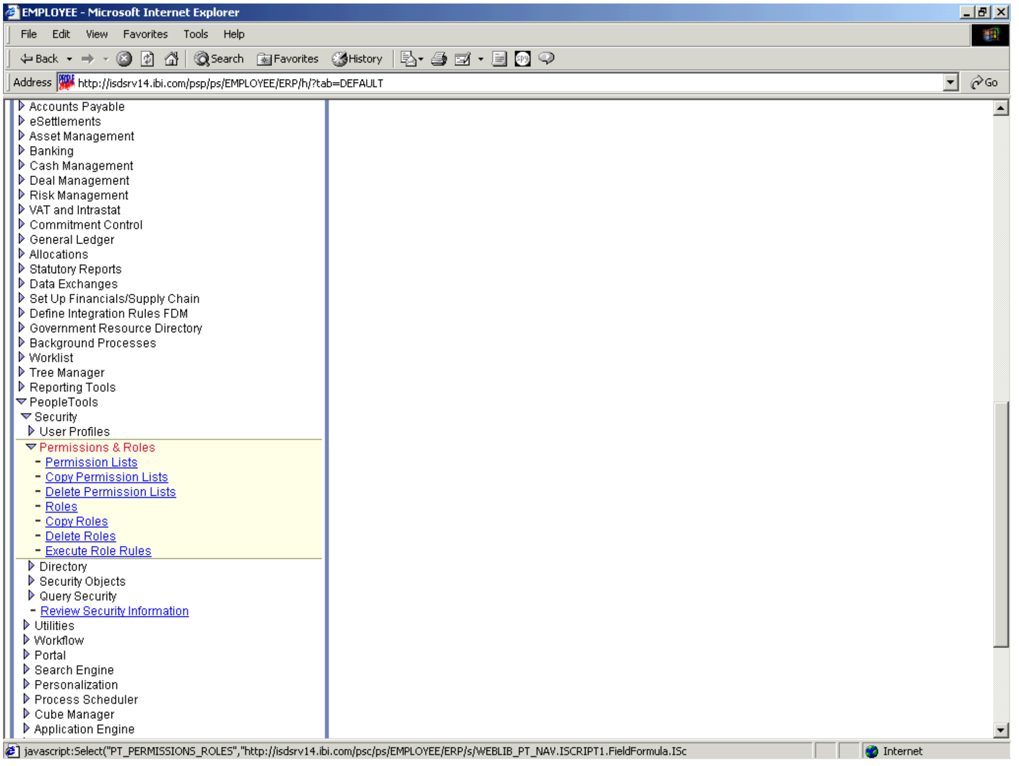
The following image shows the ALLPORTL Permission List with full access to all methods (Get, Create, Save, Cancel, and Find). It also contains OK, Cancel, Full Access All, and No Access All buttons.



5. To select the desired level of access, click in the Method Access column opposite the method and select from the drop-down list.
6. Click *OK*.

Procedure: How to Configure Interface Security for PeopleSoft Version 8.4

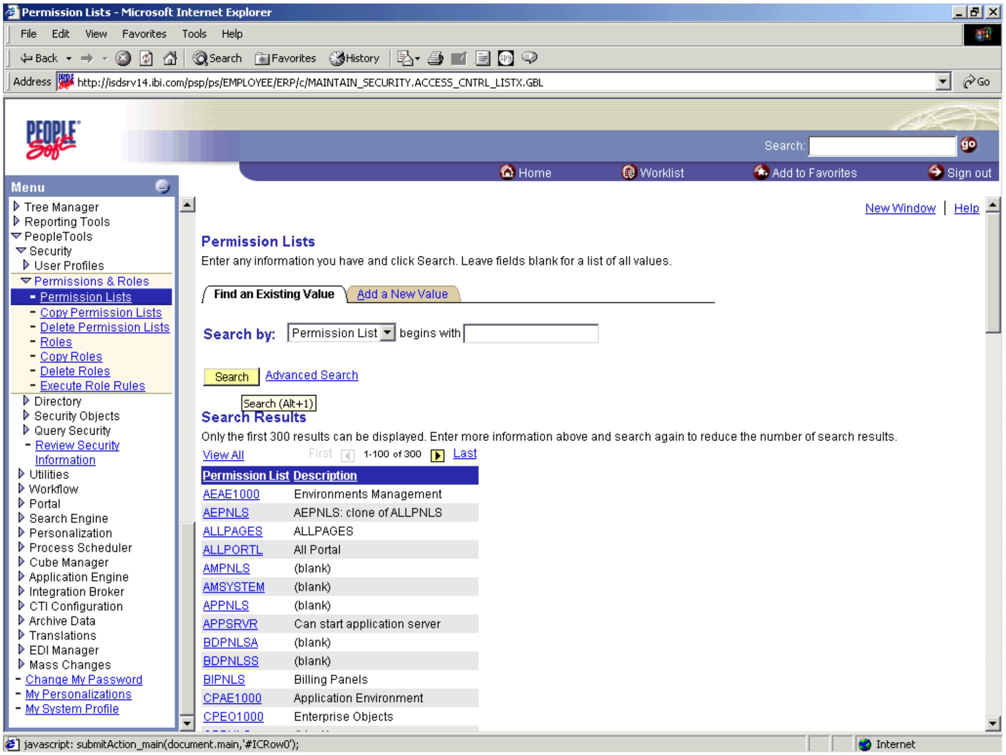
The following image shows the configure component interface security window. Permissions and Roles is expanded in the left pane.



To configure interface security:

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

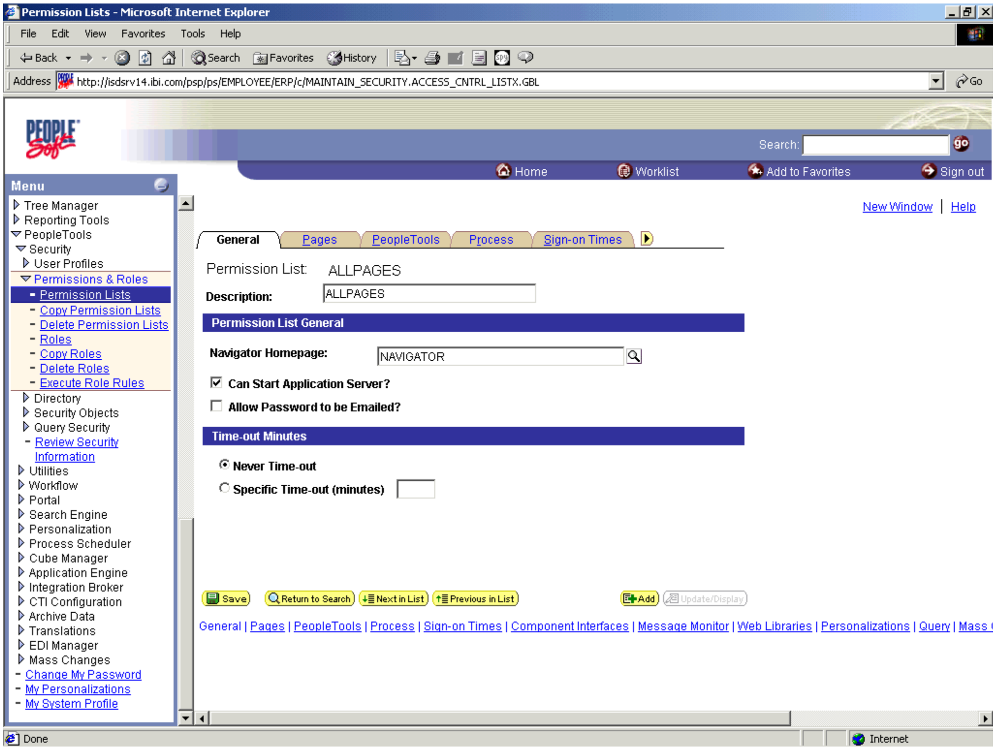
   The following image shows the Permission Lists pane on the right with the Find an Existing Value tab active. You can view an existing value in the Permission List column with its associated description and search for results.



2. Click *Search*.

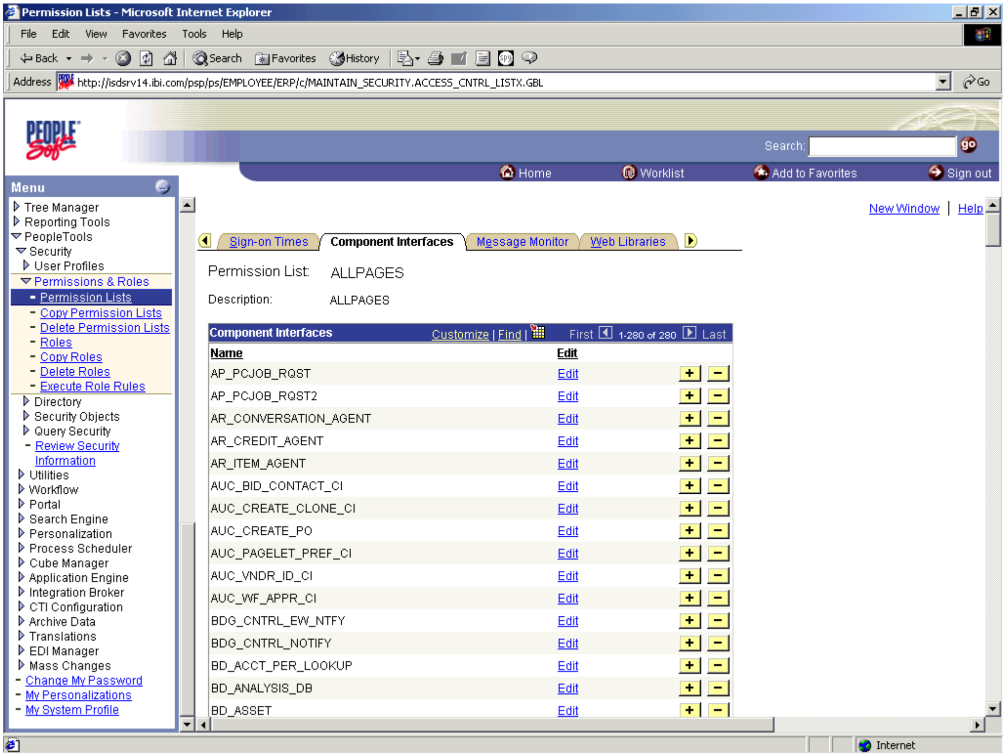
3. Select the relevant permission list.

   The following image shows the General tab and contains information for the ALLPAGES permission list. It contains Description and Navigator Homepage fields; check boxes for selecting Can Start Application Server? and Allow Password to be Emailed?; and options to choose between Never Time-out and Specific Time-out.



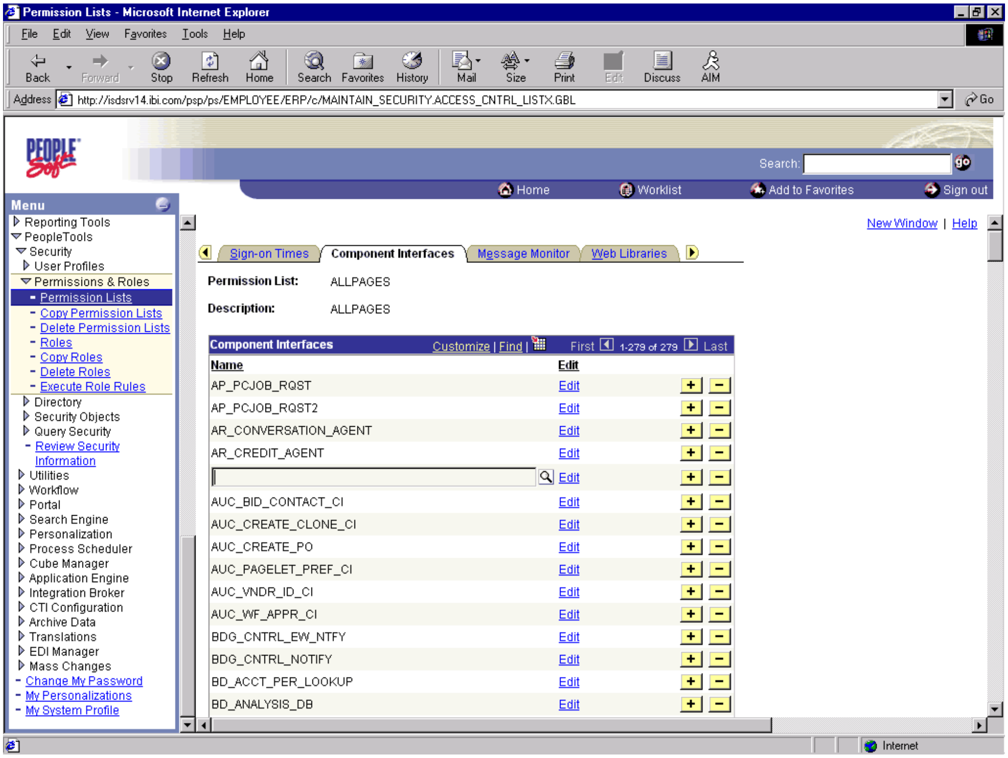
4. To view the Component Interfaces tab, click the right arrow next to the Sign-on Times tab and click the *Component Interfaces* tab.

The following image shows the Component Interfaces tab and lists component interfaces for the ALLPAGES permission list.



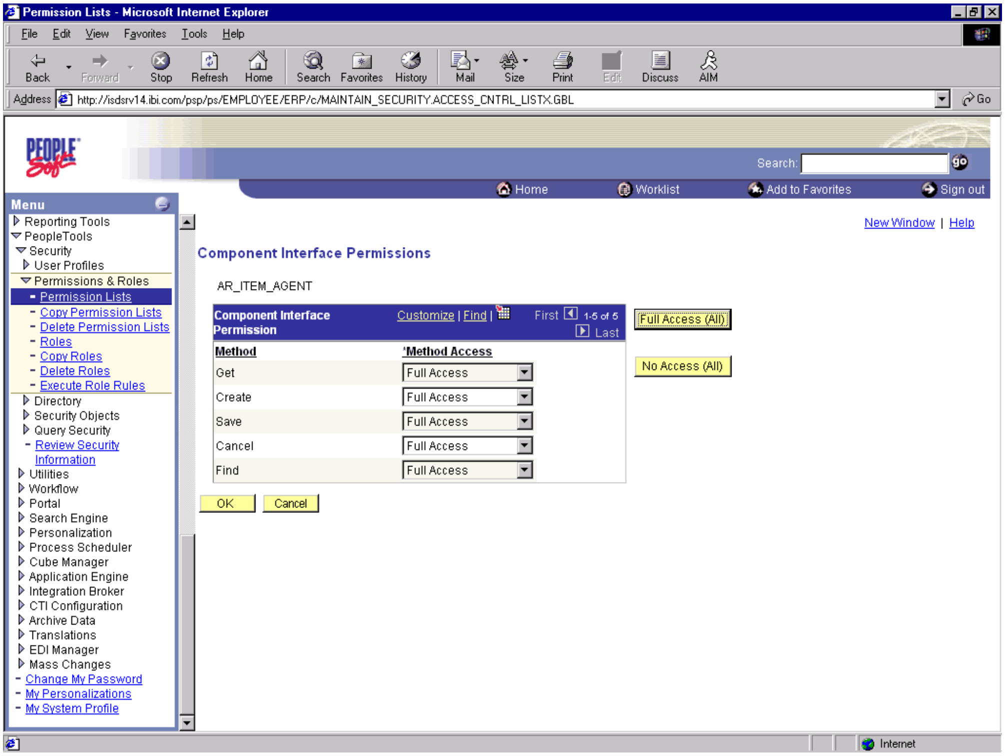
5. To add a new row to the Component Interfaces list, click the *plus* button.

The following image shows the Component Interfaces tab and lists component interfaces for the ALLPAGES permission list. It includes a component interface name field where you clicked the plus sign to add a row.



6. Type the Component Interface name and click *Edit*.

   The following image shows the Component Interface Permissions pane for the component interface AR_ITEM_AGENT. It includes a column for the Method; drop-down lists for the Method Access corresponding to: Get, Create, Save, Cancel, and Find; and the buttons for Full Access (active) or No Access.



   a. From the drop-down lists, select the desired access level for each method.

   b. Click *OK*.

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

Testing a Component Interface

The BEA WebLogic 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. Therefore, each Component Interface must be tested before it can be used as a source for the adapter.

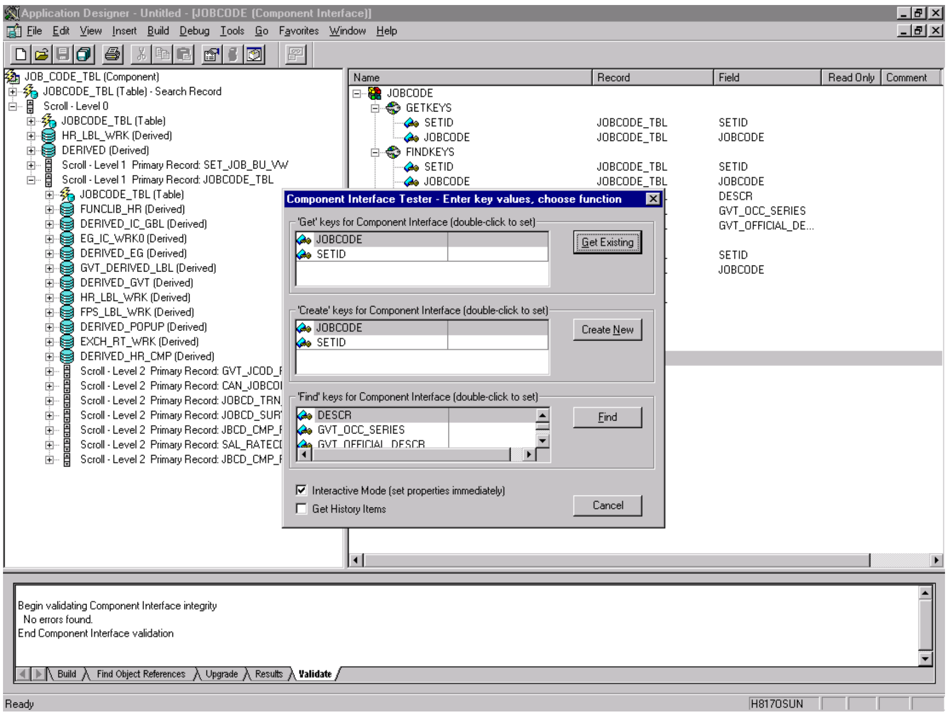
If changes are made to the underlying application by the user or by a PeopleSoft upgrade, and the changes invalidate a Component Interface, the user must repair the invalid Component Interface before the adapter uses it.

Procedure: How to Test a Component Interface

To test a Component Interface:

1. In Application Designer, from the Tools menu, select *Test Component Interface*.

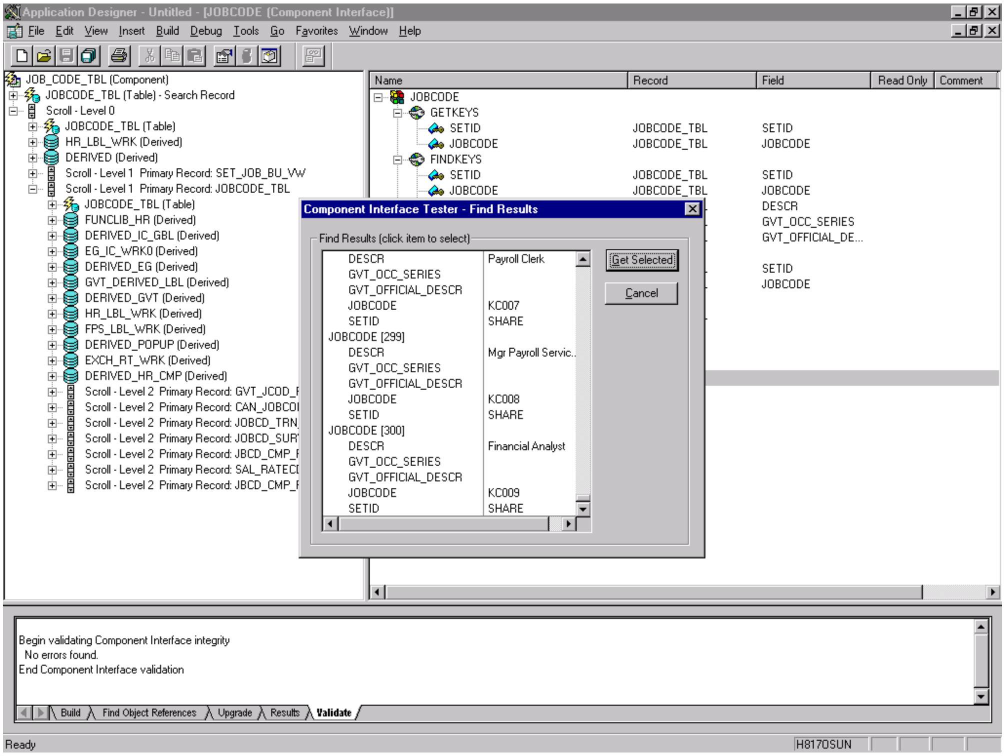
   The Component Interface Tester dialog box opens as shown in the following image. It contains the following panes: Get keys for Component Interface, Create keys for Component Interface, and Find keys for Component Interface. It also contains Get Existing, Create New, Find, and Cancel buttons and check boxes for Interactive Mode and Get History Items.



2. To test the Component Interface, use one the following methods:

To test the Component Interface using the **Find method,** click *Find*.

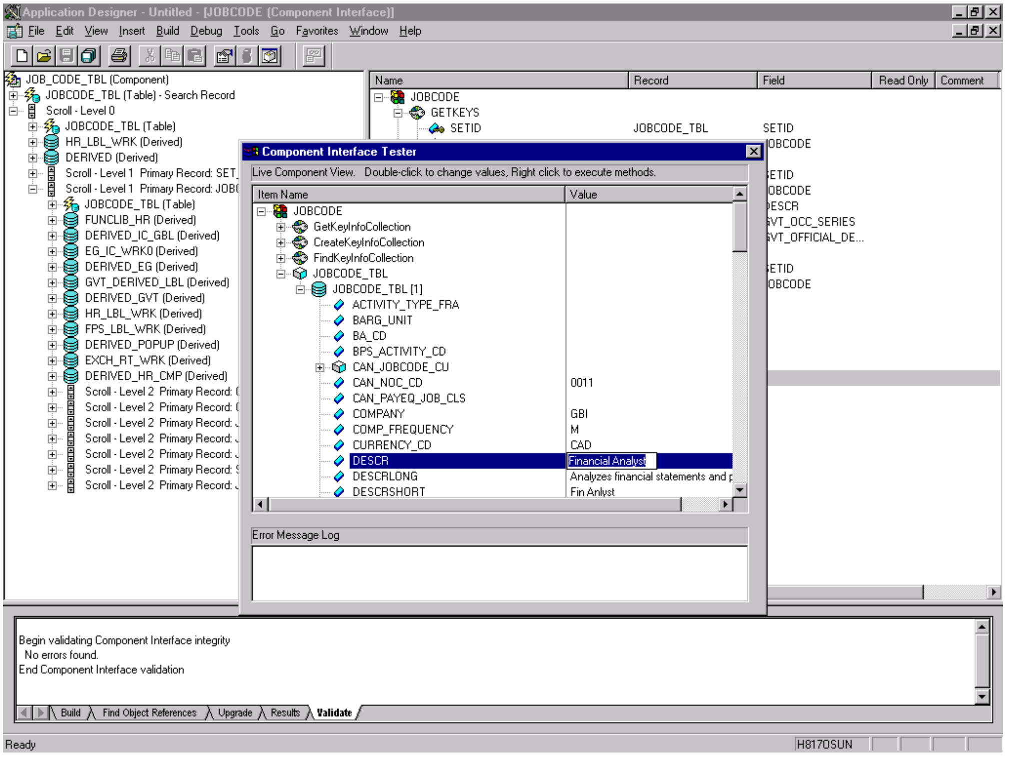
The Component Interface Tester - Find Results dialog box opens and displays all of the possible entries for the underlying component. If there are more than 300 entries, a message appears as shown in the following image.



a. In the left pane of the Find Results dialog box, select a field.

b. To display the relevant data for that particular field, click *Get Selected*.

The following image shows the Component Interface Tester that opens.



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

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.

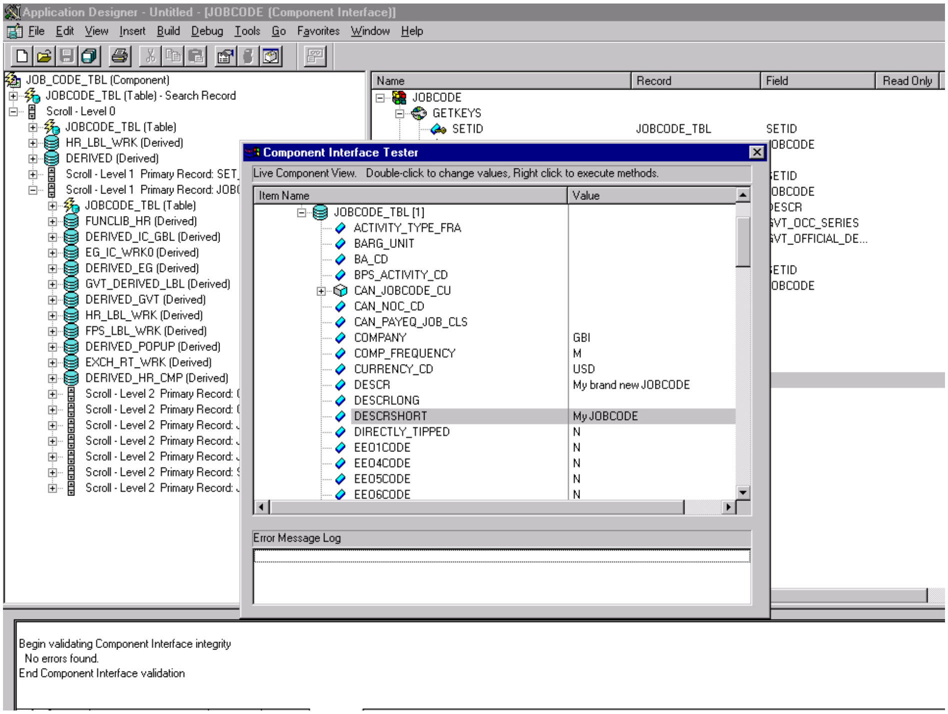
Alternatively, you can test using the Create method.

To test the Component Interface using the **Create method:**

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 after the Table name is expanded with default data in place appears as shown in the following image.



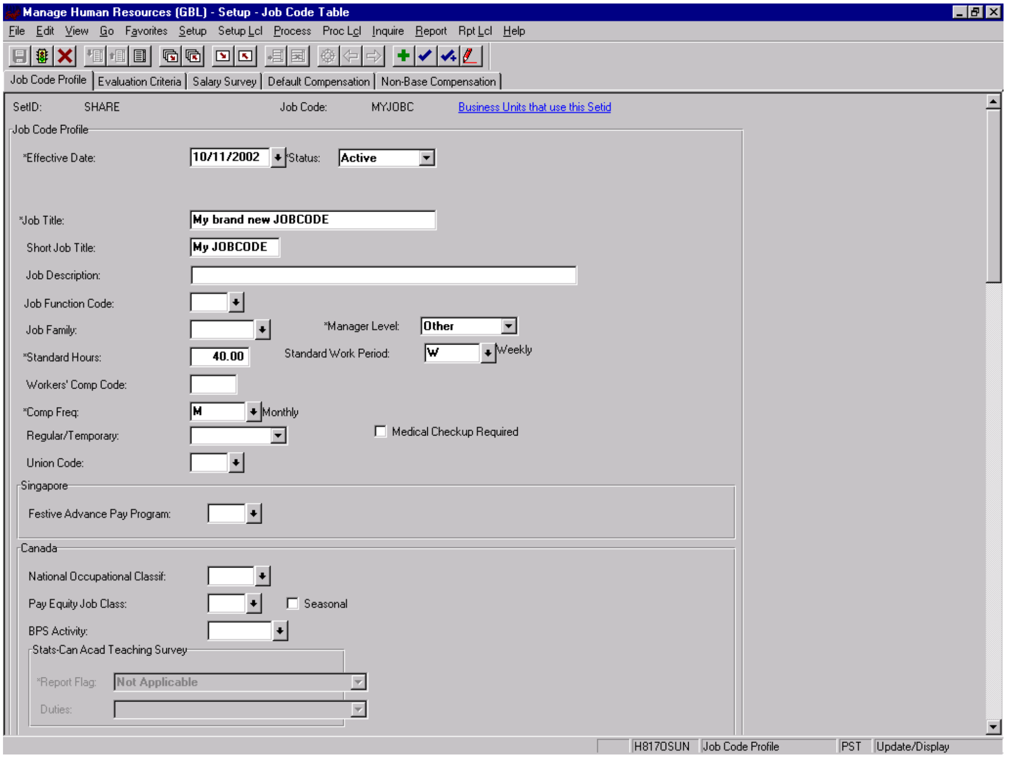
At this point, you can change fields. Changes are validated against the underlying business logic of the component.

c. After you finish making changes, right-click the top item in the pane.

3. 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.

You can view the data that was added in the PeopleSoft Component as shown in the following image. The Job Code Profile pane displays information in fields and drop-down lists relating to job title, description, and so forth.



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. For more information, see Chapter 3, *Generating Component Interface APIs*.

APPENDIX B

Using PeopleSoft 8 Integration Broker

Topics:

- PeopleSoft Integration Broker
- Configuring Integration Broker in PeopleSoft 8.4
- Configuring Application Messaging in PeopleSoft Release 8.1
- Viewing the PeopleCode for a Message
- Testing the Integration Broker
- Using Outbound Synchronous Messages

This section 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 and 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 intended as a helpful supplement; it 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 B-26.

To send a message, you must properly configure various internal structures and processes. The following descriptions are generally release-generic. Details of differences between releases 8.1 and 8.4 are discussed in other topics.

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 can use any supplied message.

- **Message.** A Message is a container for the data that goes into the XML. It contains basic structural information, such as records and fields. To send the XML file, the Message must be in an Active status.
- **Message Channel.** The Message Channel is a mechanism for structuring records into logical groupings. Each Message can belong to only one Message Channel. For the Message to be delivered, the Message Channel must be in an Active (Run) status.

  In release 8.1, the Message Channel also provides preliminary routing instructions; you can specify the Message Nodes that 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 the Gateway that receives the messages.

  Much of the “intelligence” built into the Message Channel in release 8.1 moved to the Message Node in release 8.4 which provides additional flexibility. 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 be initiated only by using specific PeopleCode instructions. This code is usually triggered by an application event, such as creating a new database entry through an online panel or through a batch job.

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 IntegrationGateway.properties file to communicate with your PeopleSoft 8.4 application.
3. Configure the Integration Gateway and File Output connector.
4. Create and configure a new Gateway node.

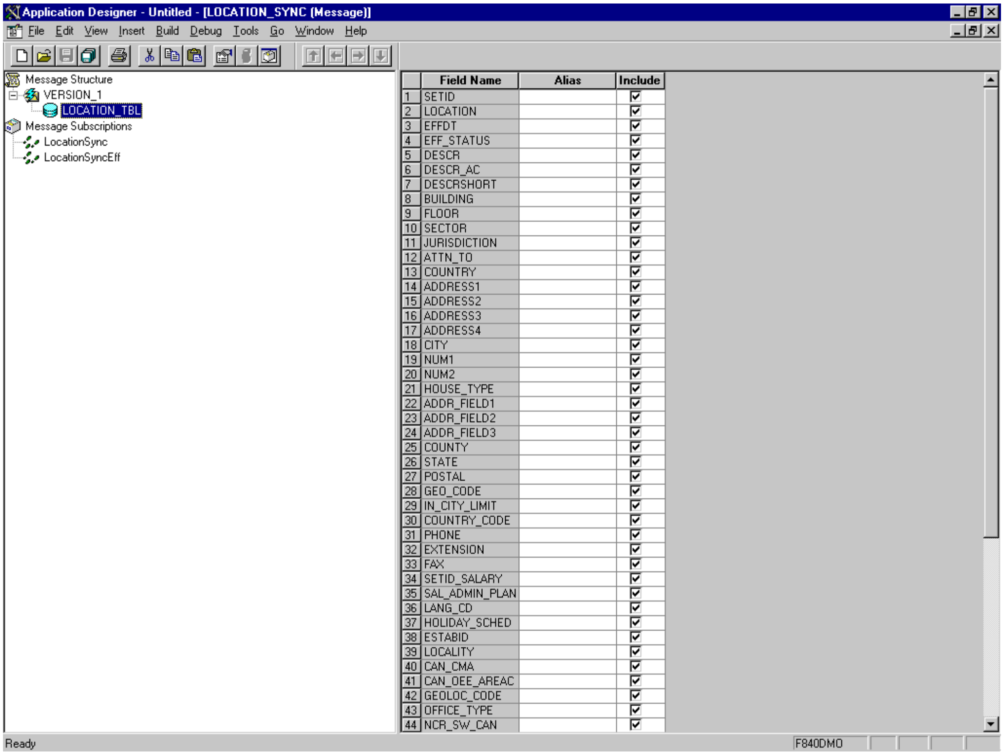
These tasks are described in detail in the following procedures.

Procedure: How to Ensure 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.

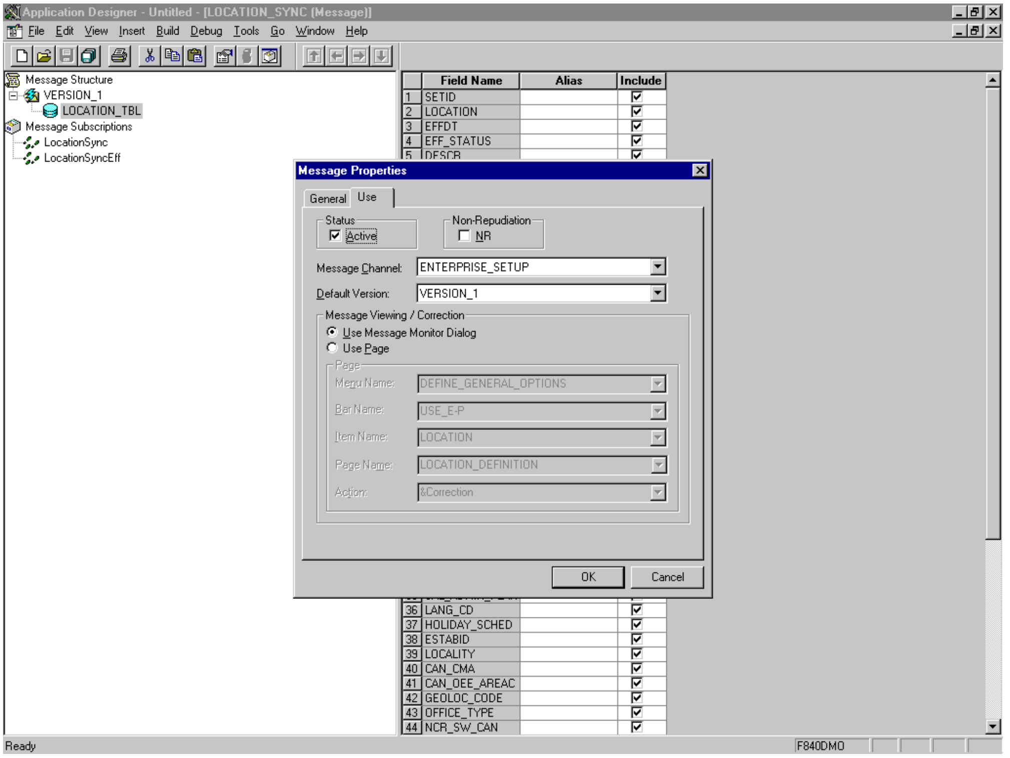
   The Application Designer opens as shown in the following image. The message field names appear in a column with corresponding columns for the field alias and a check box indicating whether to include the field.



3. To view the fields that are included in the Message, select *LOCATION_TBL*.

4. To view the Properties dialog box, right-click *LOCATION_TBL* and select *Properties*.

The Message Properties dialog box opens as shown in the following image and includes Non-Repudiation and Status check boxes, Message Channel and Default Version lists, and Use Message Monitor Dialog and Use Page options.



a. Select the *Use* tab.

b. Ensure the *Status* check box is selected, indicating that the message is active.

The message is routed to the Message Channel, ENTERPRISE_SETUP, and the default message version is VERSION_1 (messages can have multiple versions).

5. Click *OK*.

6. Save the message.

You have ensured that the message is active and routed correctly.

Procedure: How to Configure the IntegrationGateway.properties File

To configure the IntegrationGateway.properties file:

1. Using the editor of your choice, open the *IntegrationGateway.properties* file.
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. Usually, this script is located in the path of the IntegrationGateway.properties file. To run this script, follow the instructions provided by PeopleSoft.

You have finished configuring the IntegrationGateway.properties file.

Procedure: How to Configure 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 type the following Gateway URL:

   `machine-name/PSIGW/PeopleSoftListeningConnector`

   where:

   `machine-name`

   Is the URL of your PeopleSoft Web Server.

   a. Click *Refresh*.

      A message appears stating the outcome of the refresh process.

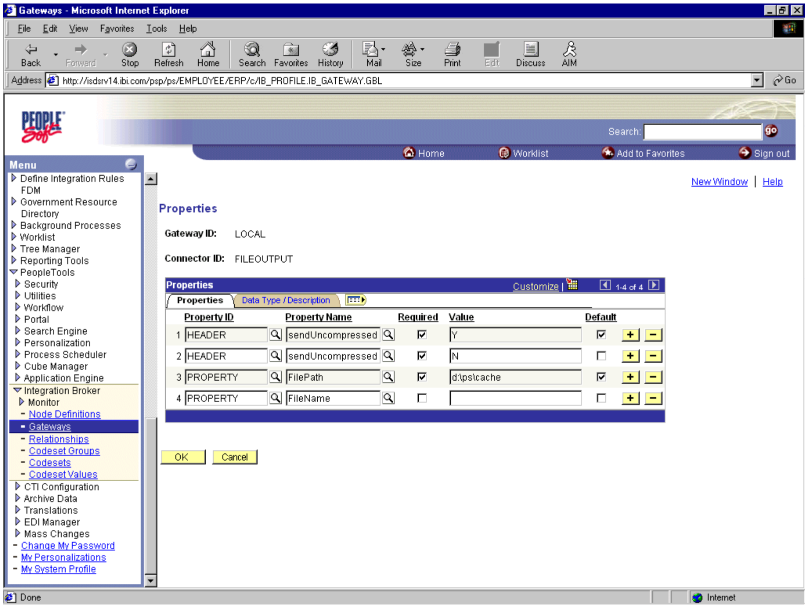
   b. Click *OK*.

4. Scroll down and click *Save*.

   You must click Save before continuing.

5. Click the *Properties* hyperlink for the FILEOUTPUT Connector ID.

   The Properties pane for the FILEOUTPUT Connector opens, as shown in the following image. Information for the Gateway ID and the Connector ID appears. The Properties tab is active and includes fields for the Property ID, Name, and Value. Check boxes indicating whether required and whether the value is the default also appear on the tab.



   a. Accept or overwrite the default values.

      In the previous image, the FilePath PROPERTY from the c:\temp default was changed to d:\ps\cache.

   b. To return to the Gateway window, click *OK*.

6. Scroll down and click *Save*.

You have finished configuring the Integration Gateway and the File Output Connector.

Procedure: How to Create and Configure a New Gateway Node

To create and configure a new Gateway Node:

1. In the Menu pane, expand *PeopleTools*, *Integration Broker*, and click *Node Definitions*.
   a. Select the *Add a New Value* tab.
   b. 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.
2. Click *Add*.

   The Node Info tab becomes available.
   a. In the Description field, type an appropriate description.
   b. From the Node Type drop-down list, select *EXTERNAL*.
   c. From the Routing Type drop-down list, select *Implicit*.
3. Click the *Connectors* tab to make it available.
   a. For the Gateway ID, specify *LOCAL*.
   b. For the Connector ID, specify *FILEOUTPUT*.
   c. Accept or overwrite the default Gateway property values.
4. Click *Save*.
5. To specify the transactions to route messages to your node, select the *Transactions* tab.
6. Click *Add Transaction*.
   a. From the Transaction Type drop-down list, select *Outbound Asynchronous*.
   b. In the Request Message field, specify *LOCATION_SYNC*.
   c. In the Request Message Version field, specify *VERSION_1*.
7. Click *Add*.

   The Transaction Detail pane opens.
   a. In the Routing Type drop-down list, verify that the value is *Implicit*.
   b. Click *Save*.
   c. Click the *Return to Transaction List* hyperlink.
   d. To ensure that your data entry is not lost, click *Save* again.

      You have finished creating and configuring the new Gateway Node.
8. Continue with the instructions in *Viewing the PeopleCode for a Message* on page B-22.

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.

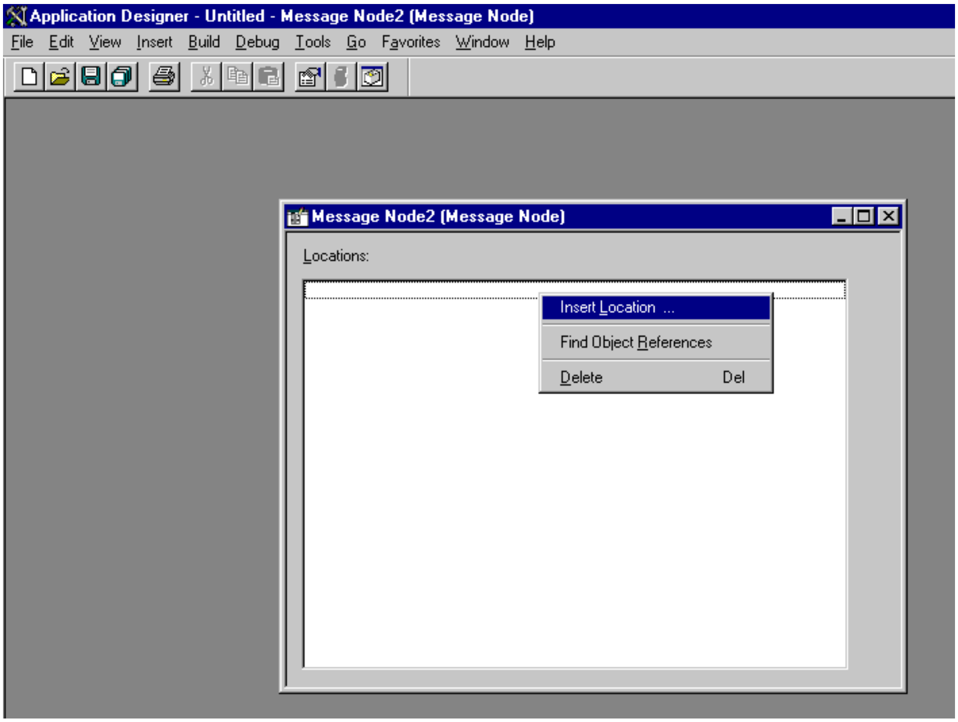
These tasks are described in detail in the following procedures.

Procedure: How to Create and Configure a New Message Node

To create and configure a new Message Node:

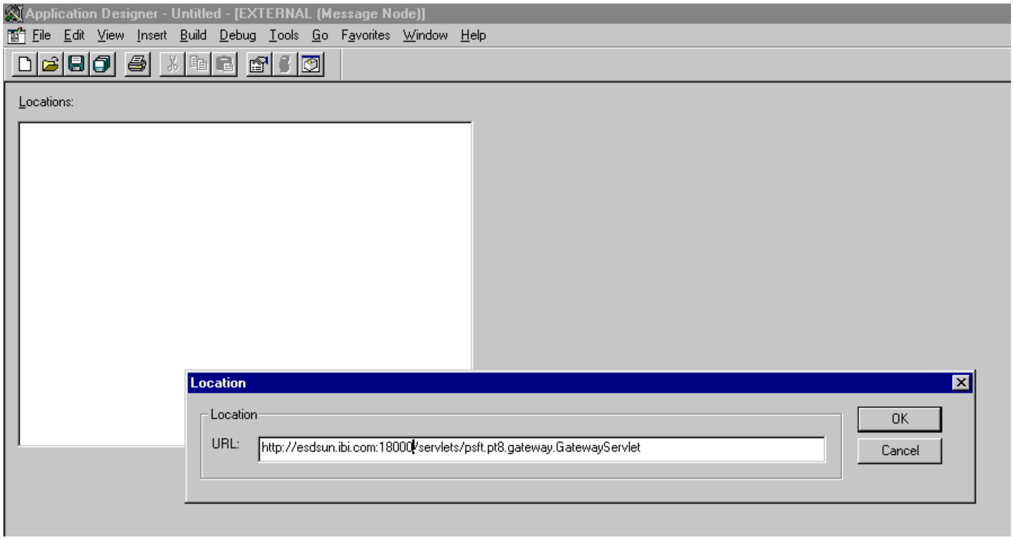
1. From the File menu, select *New* and click *Message Node*.

   The Message Node window opens and contains three options: Insert Location, Find Object References, and Delete as shown in the following image.



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

The Location URL box opens where you type the URL for the location as shown in the following image.



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

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

where:

`machine-name`

Is the URL of your PeopleSoft Web server.

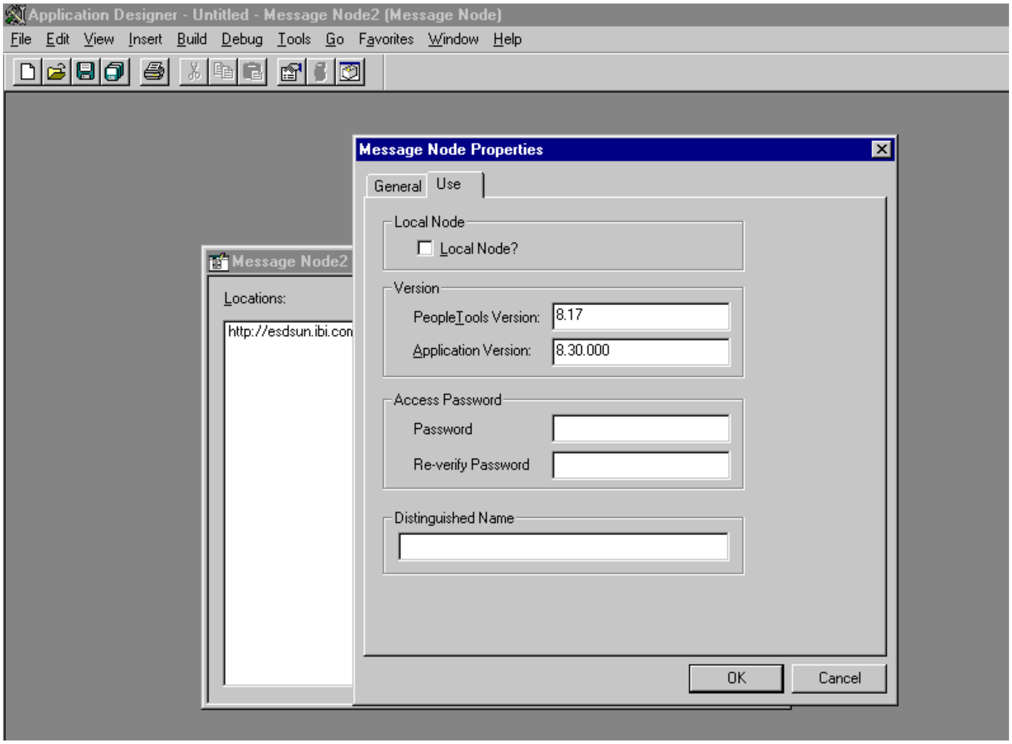
`port`

Is the socket on which the server is listening.

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

4. Click *OK*.

The Message Node Properties dialog box opens. It contains a Local Node check box and PeopleTools Version, Application Version, Password, Re-verify Password, and Distinguished Name fields.

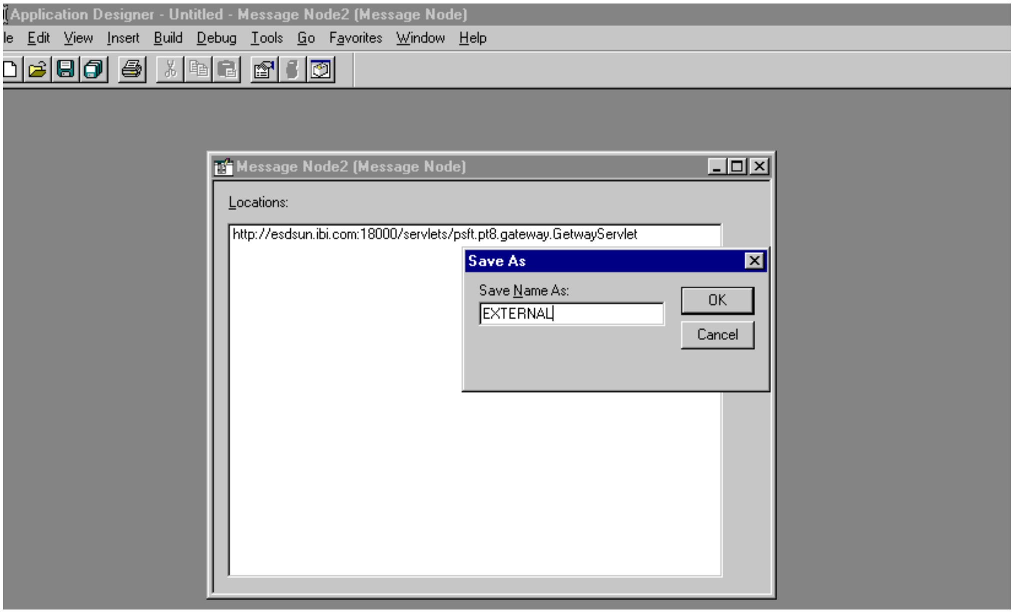


a. Select the *Use* tab.

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

5. Click *OK*.

The following image shows the Save As dialog box with a Save Name As field.



6. 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.

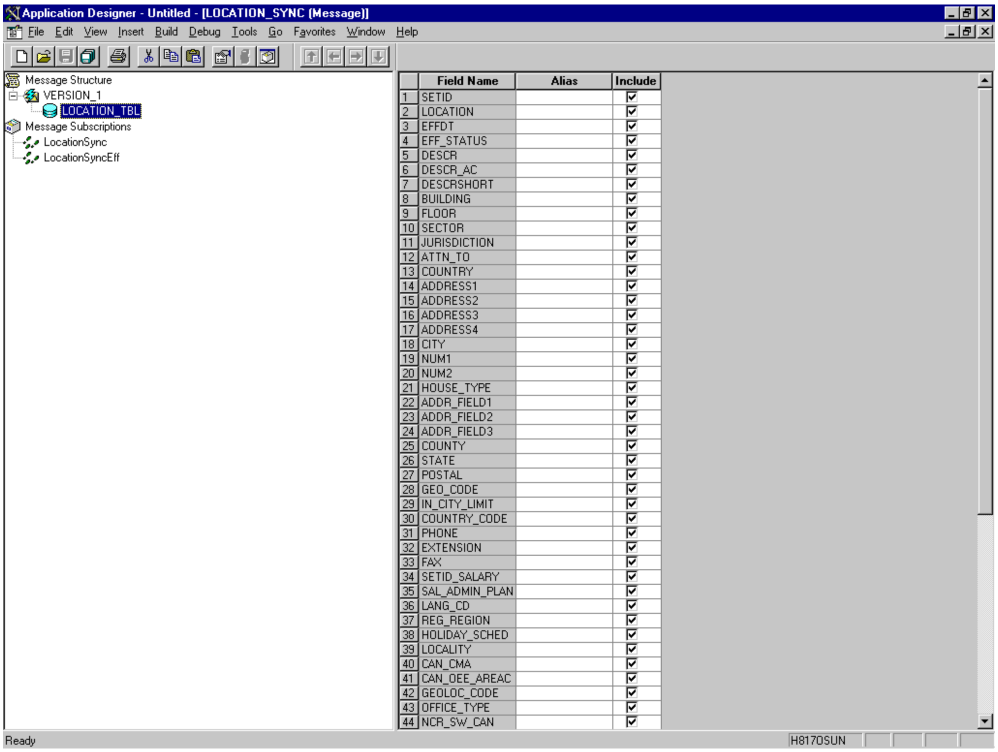
You have finished creating and configuring the message node.

Procedure: How to Ensure That the Message Is Active and Is Routed Correctly

To ensure that the message is active and is routed to the proper message channel:

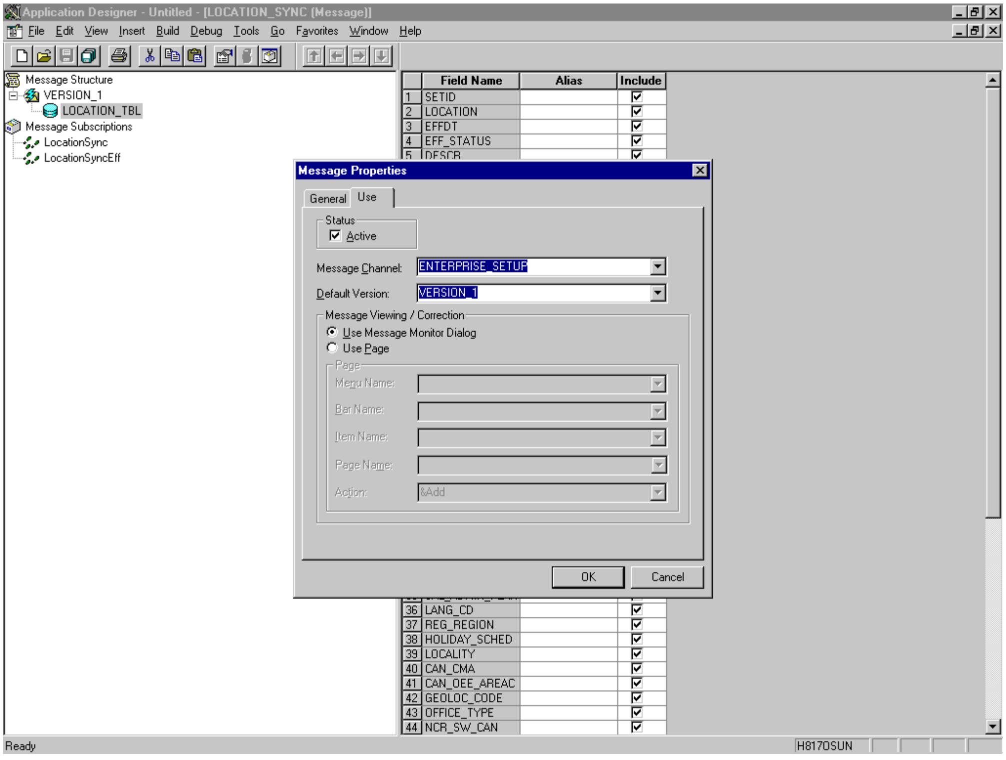
1. Open Application Designer.
2. On the File menu, point to *Open*, click *Message*, and open the *LOCATION_SYNC* messageTo view the fields that are included in the message, highlight *LOCATION_TBL*.

   The following image shows the LOCATION_TBL field, selected in the left pane. The message field names appear in a column with corresponding columns for the field alias and a check box indicating whether to include the field.



3. Right-click *LOCATION_TBL* and select *Properties*.

The Message Properties dialog box opens as shown in the following image and includes Non-Repudiation and Status check boxes, Message Channel and Default Version lists, and Use Message Monitor Dialog and Use Page options.



a. Select the *Use* tab.

b. Ensure the *Status* check box is selected, indicating that the message is active.

c. From the Message Channel drop-down list, select *ENTERPRISE_SETUP*.

d. From the Default Version drop-down list, select *VERSION_1* (messages can have multiple versions).

4. Click *OK*.

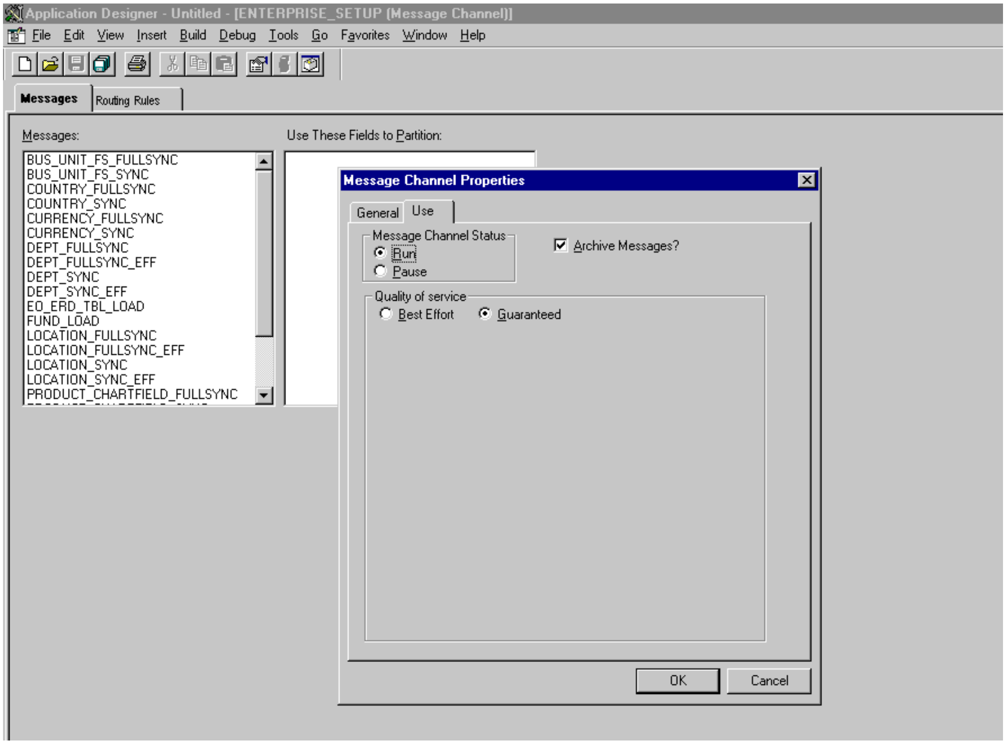
5. Save the message.

Procedure: How to Configure the Message Channel

To configure the Message Channel:

1. Open the *ENTERPRISE_SETUP* Message Channel.
2. Right-click *ENTERPRISE_SETUP* and select *Properties*.

   The Message Channel properties dialog box opens as shown in the following image. It contains Message Channel Status and Quality of Service options and an Archive Messages? check box.



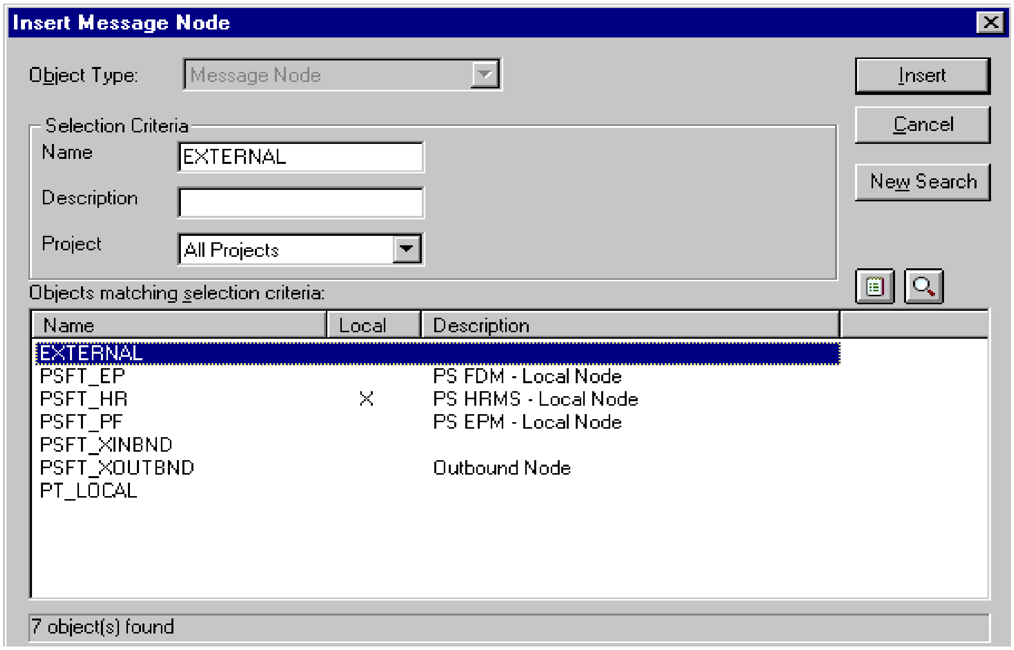
   a. Select the *Use* tab.
   b. Ensure that Message Channel status is set to *Run*.
   c. Click *OK*.

   You are returned to Application Designer.

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

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

The Insert Message Node dialog box opens, as shown in the following image with a list of objects that match the selection criteria.

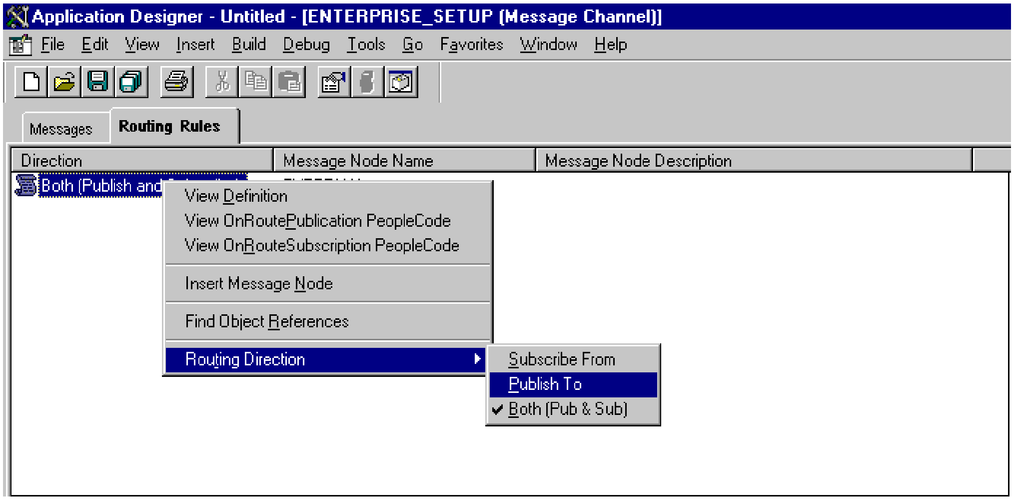


a. Select the message node, for example, EXTERNAL, that you created in *How to Create and Configure a New Message Node* on page B-10.

b. Click *Insert*.

5. Click *Cancel*.

You are returned to the routing Rules tab where you can select from expanding menus as shown in the following image.



a. Right-click the message node and point to *Routing Direction*.

b. From the Routing Direction menu, select *Publish To*.

6. Save the Message Channel, and if you require it, place it in your Project.

You have finished configuring the Message Channel.

Procedure: How to Configure 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 typing the following URL:

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

where:

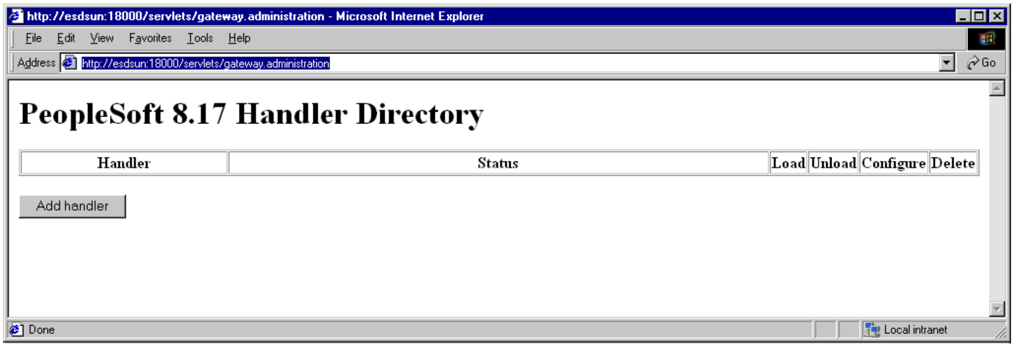
`machine-name`

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

`port`

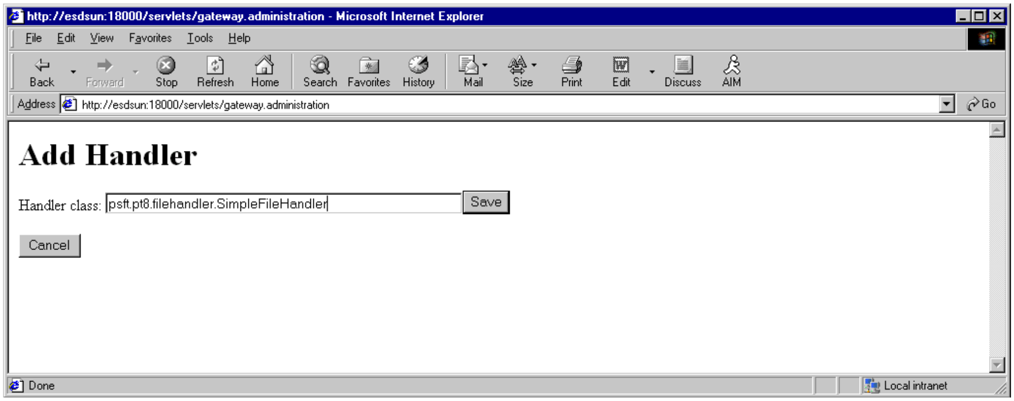
Is the port number on which the application server is listening.

The Handler Directory window opens, as shown in the following image.



2. Click *Add handler.*

The Add Handler window opens, as shown in the following image.



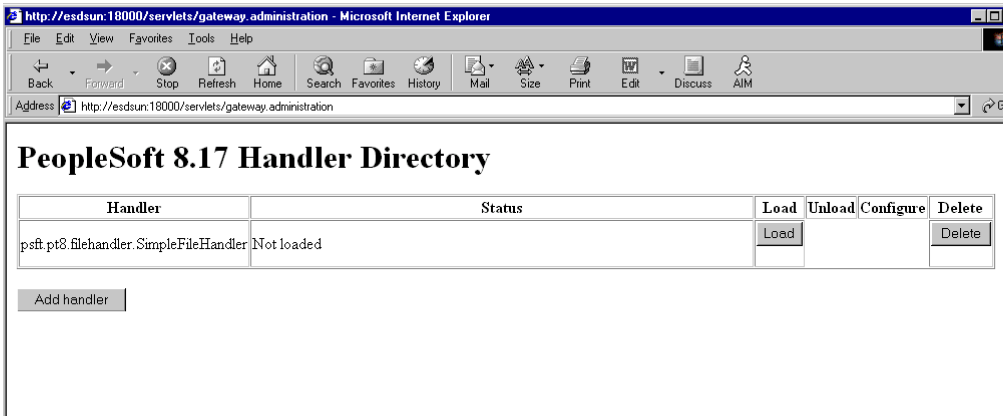
3. Type the following full name of the Simple File Handler class:

```
psft.pt8.filehandler.SimpleFileHandler
```

Note: The name is case-sensitive.

4. Click *Save*.

The Handler Directory window reopens, as shown in the following image.

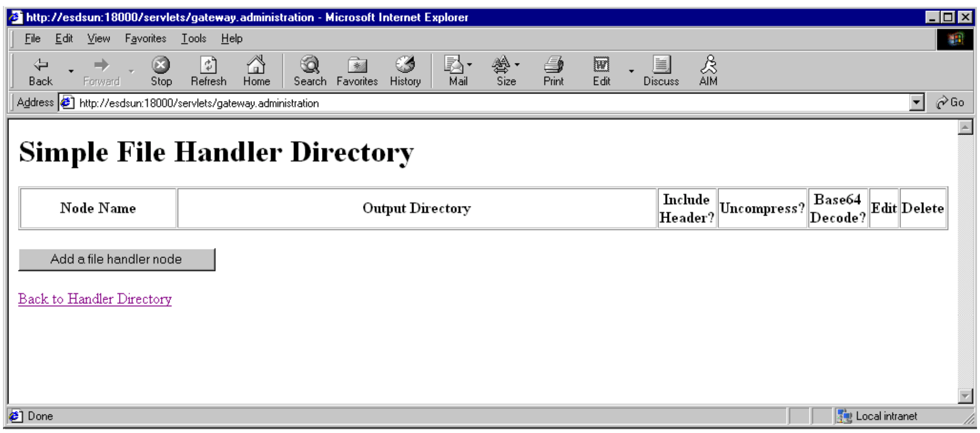


5. To load the handler, click *Load*.

After the handler loads, "Loaded successfully" appears in the Status column.

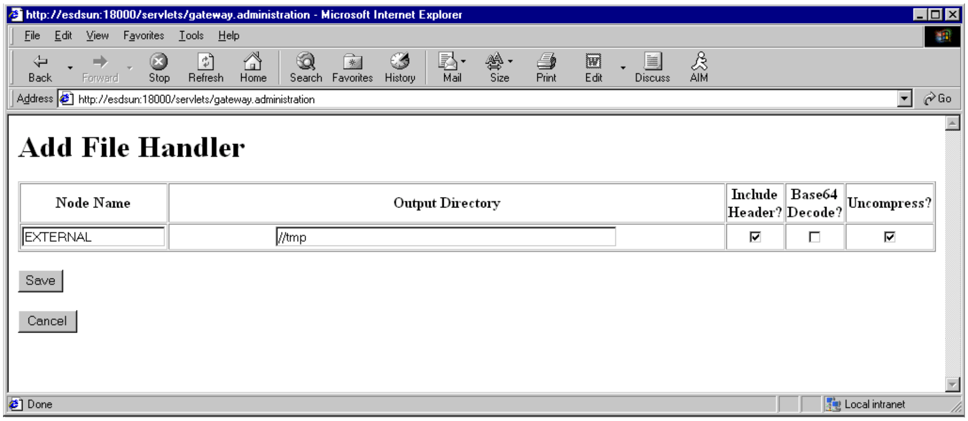
6. Click *Configure*.

The Simple File Handler Directory window opens, as shown in the following image.



7. Click *Add a file handler node*.

   The Add File Handler window opens, as shown in the following image.



   a. In the Node Name field, type the name of the Message Node, for example, EXTERNAL, that you created in *How to Create and Configure a New Message Node* on page B-10.

   b. Select an output directory appropriate for your server environment.

      The example illustrated in the previous image runs under UNIX. The default directory under Windows NT is c:\\temp\\file. (The double slashes may not be required for your environment.)

   c. Select the desired output file properties.

8. Click *Save*.

You have finished configuring the Simple File Handler.

Viewing the PeopleCode for a Message

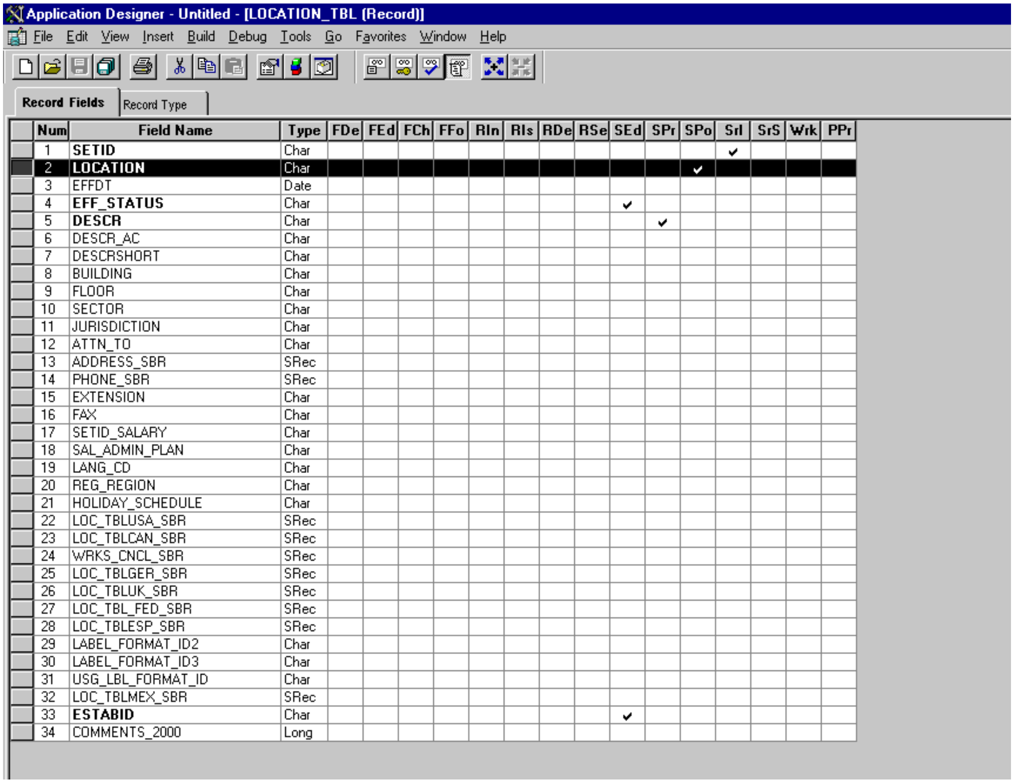
Messages are initiated by the PeopleCode that is attached to a record. Usually, this record is one of the records associated with the message itself.

Procedure: How to View the PeopleCode for a Message

To view the PeopleCode for a message:

1. Open Application Designer.

   The following image shows an open Application Designer which contains a list of field names with their corresponding types and other information.



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															

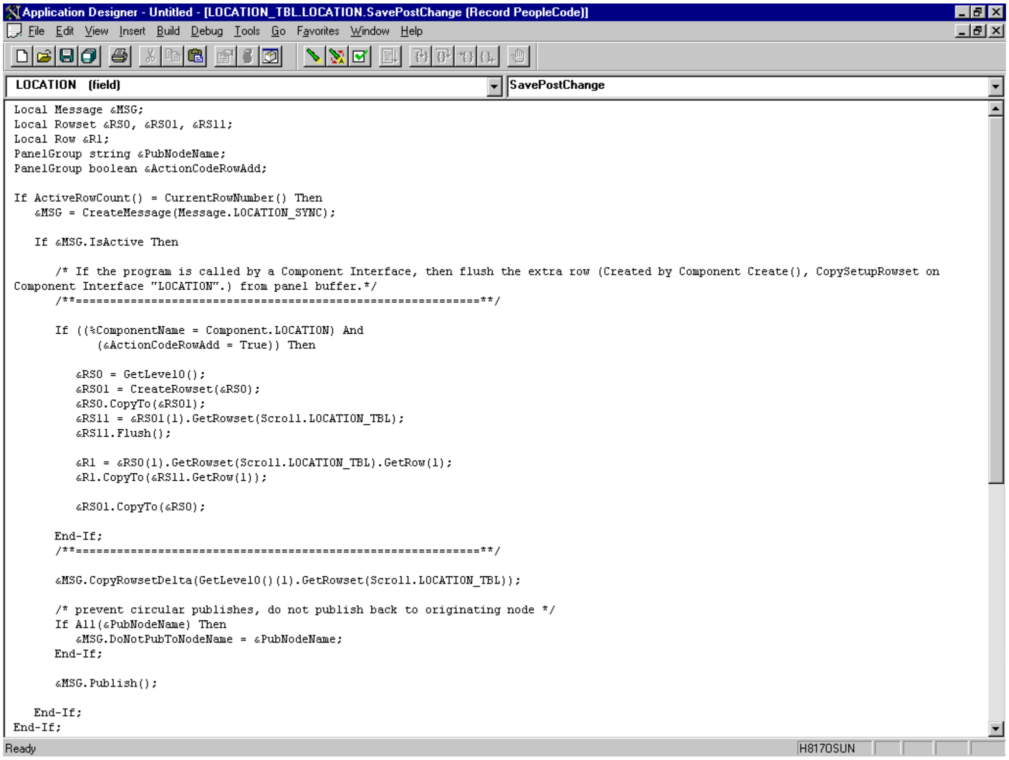
2. Select the *Record Fields* tab.

   a. Select the *LOCATION_TBL* record.

   b. Select the PeopleCode display option.

3. Select the *Save Post Change (SPo)* box for the LOCATION field.

The PeopleCode that initiates a LOCATION_SYNC message appears, as shown in the following image.



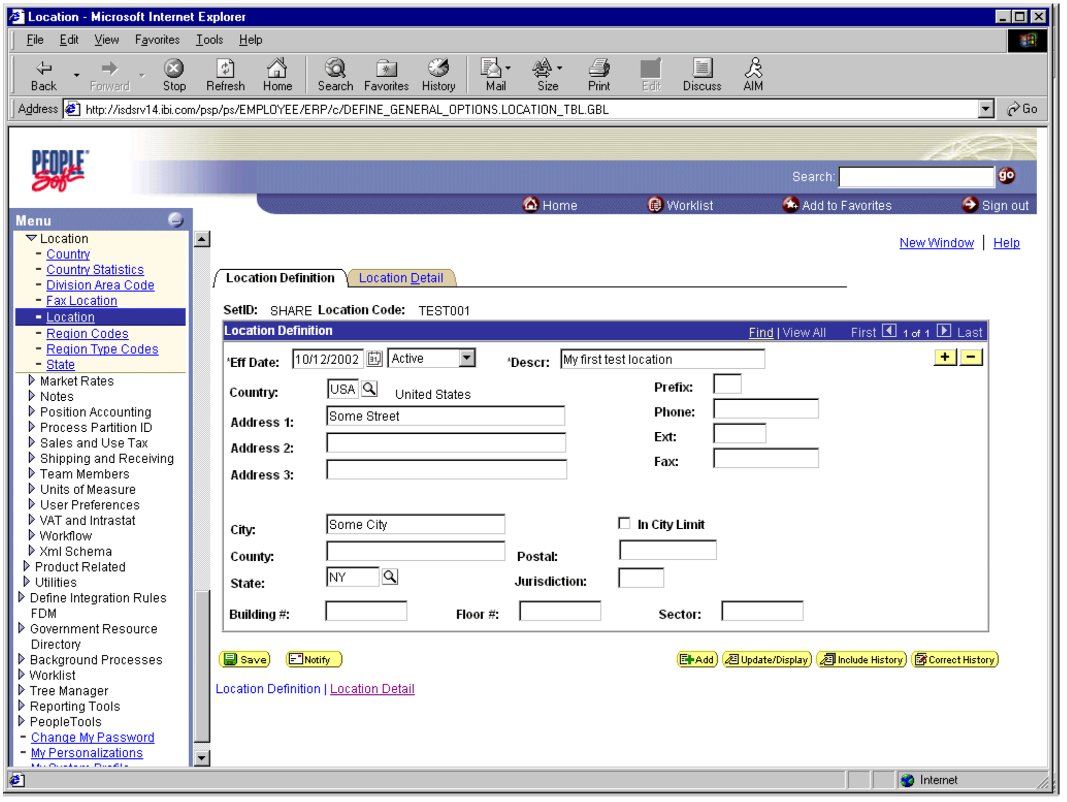
For more information about PeopleCode, consult your PeopleSoft Online Library.

You have viewed the PeopleCode for a message. You can now test Integration Broker (in PeopleSoft 8.4) or Application Messaging (in PeopleSoft 8.1).

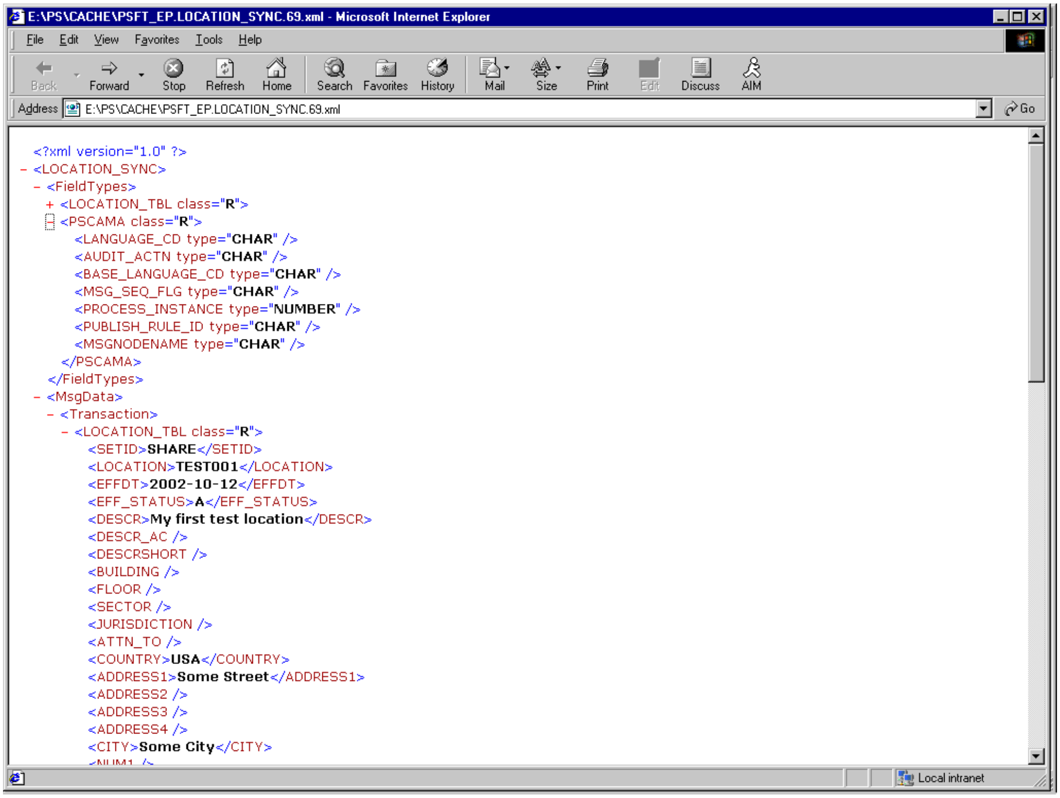
Testing the Integration Broker

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 image shows a Financials 8.4 application with the Location Definition tab selected. A new location with a SetID of SHARE and a Location Code of TEST001 was added.



The following image shows a portion of the XML output from a test message.



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 you PeopleSoft Online Library. If you still cannot 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.

You can use an existing node, or you can create a new node to configure outbound synchronous messages. For information on creating and configuring a node, see *How to Create and Configure a New Gateway Node* on page B-9. In either case, you must configure your outbound synchronous transaction.

The BEA WebLogic Adapter for PeopleSoft can work with PeopleSoft outbound synchronous messages. Outbound synchronous messages involve additional configuration steps, both within PeopleSoft and in your application server.

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. For more information on outbound asynchronous messages, see *Configuring Integration Broker in PeopleSoft 8.4* on page B-3.

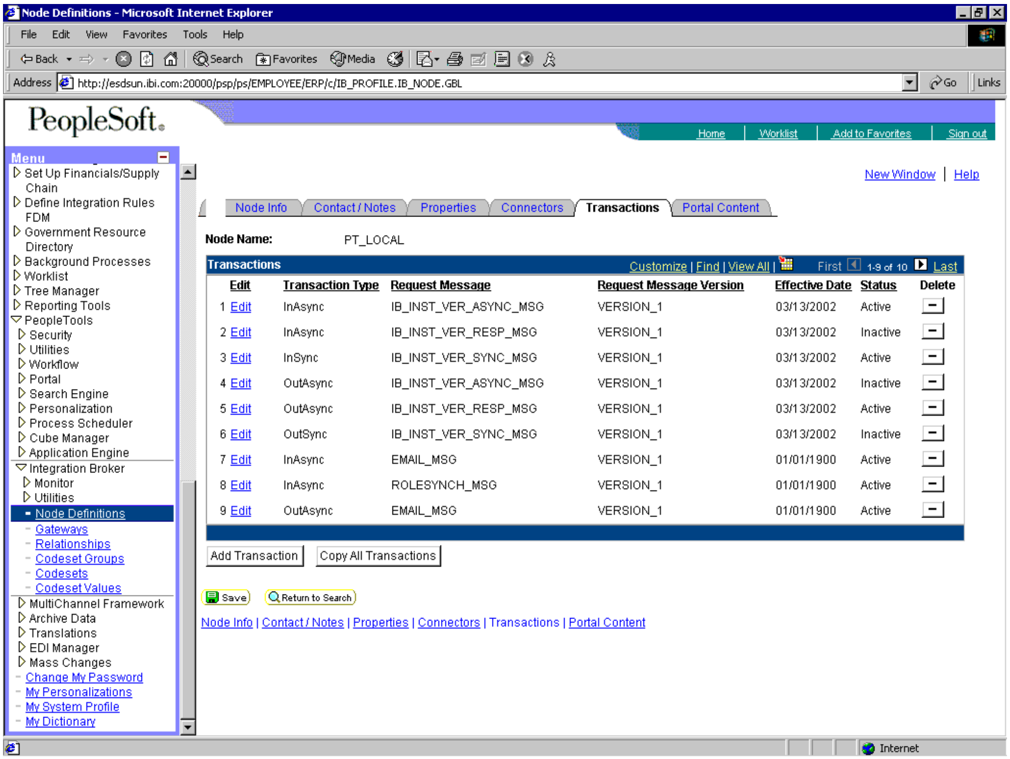
Ensure that both outbound and inbound messages are created and active. PeopleSoft provides template examples called IB_INST_VER_SYNC_MSG and IB_INST_VER_RESP_MSG. For information on examining these messages, see *How to Ensure the Message Is Active and Is Routed Correctly* on page B-4.

Example: Configuring an Outbound Synchronous Message

The following example uses a node and transaction delivered by PeopleSoft. However, this example is for illustrative purposes only and does not 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.

1. Navigate to the *Node Definitions* page and open the *PT_LOCAL* node.
2. Click the *Transactions* tab.

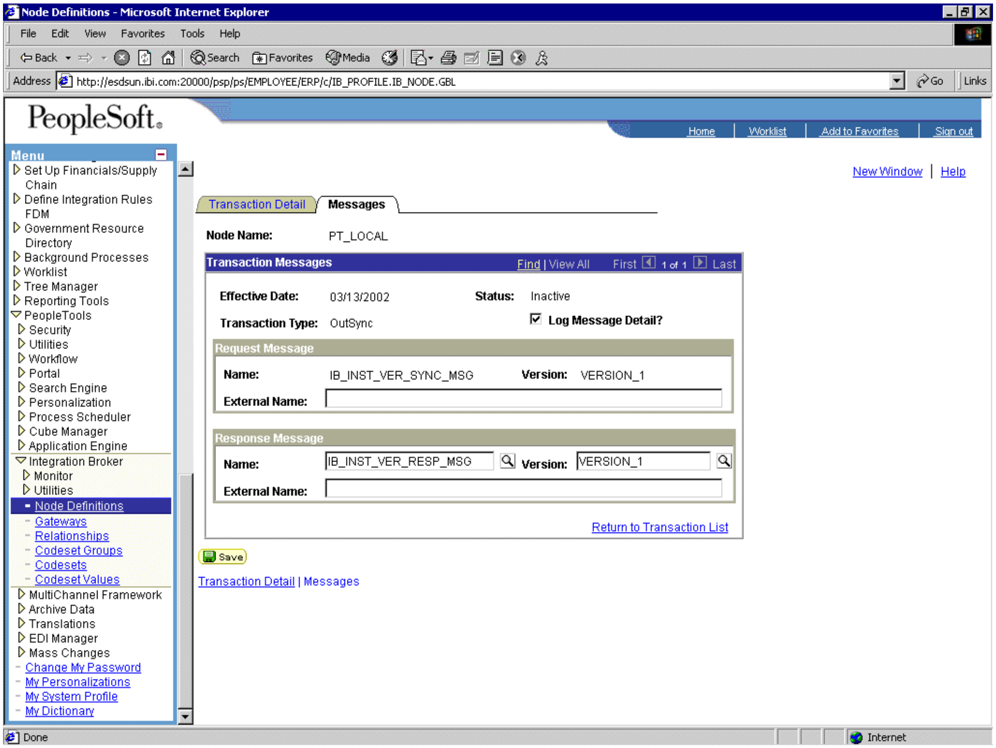
   The Transactions pane opens where you can edit transaction types and request messages as shown in the following image.



One outbound synchronous message, IB_INST_VER_SYNC_MSG, appears in the Transaction Type list.

3. In the IB_INST_VER_SYNC_MSG row, click the *Edit* hyperlink.

The Transaction Detail and Messages tabs become available, as shown in the following image.



4. Click the *Messages* tab.

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 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.

Example: Viewing the PeopleCode for a Financials Synchronous Outbound Message

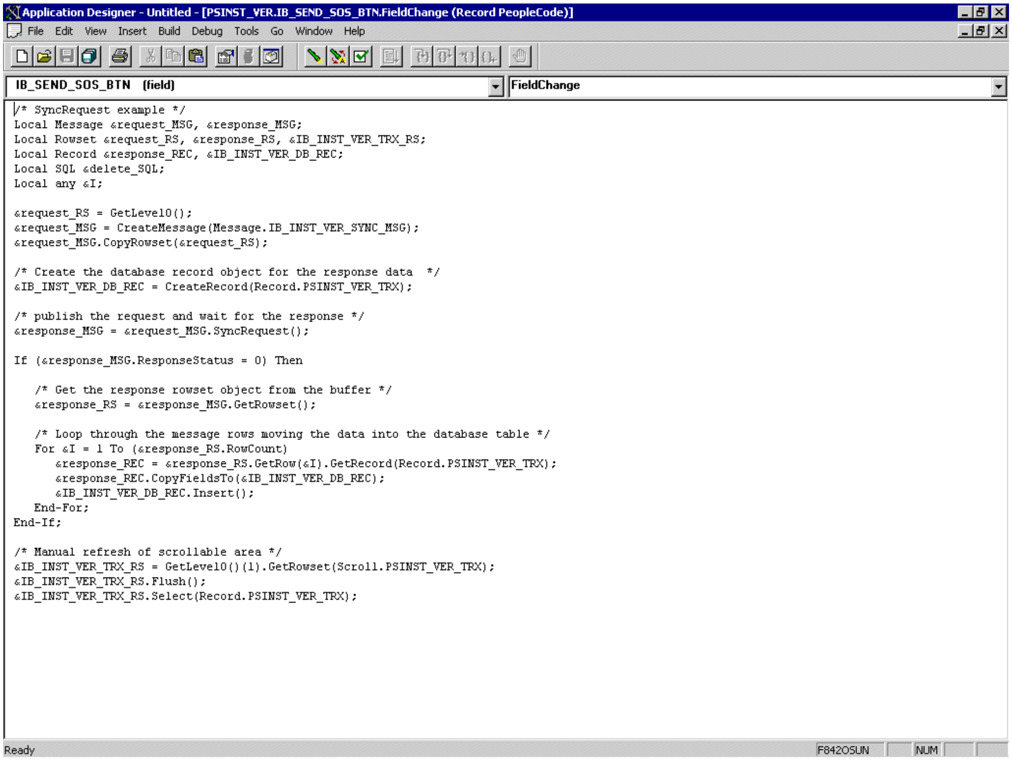
The sample PeopleCode in the following example is for a synchronous outbound message. It differs from asynchronous outbound in that it must handle a response message.

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 PeopleCode:

1. From Application Explorer, 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 code appears as shown in the following image.




```
/* 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);
```

APPENDIX C

Best Practices and Performance Considerations

Topics:

This topic provides best practices and performance considerations for the BEA WebLogic Adapter for PeopleSoft when deployed to the BEA WebLogic Server.

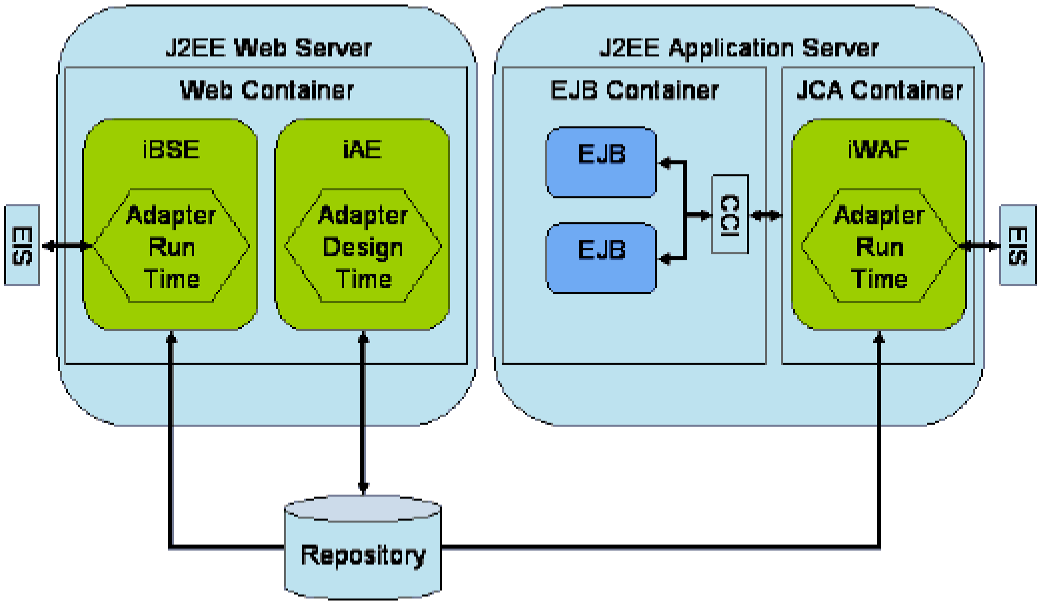
Overview of the BEA WebLogic Adapter for PeopleSoft

The BEA WebLogic Adapter for PeopleSoft provides a means to exchange real-time business data between PeopleSoft systems and other application, database, or external business partner systems. The adapter enables external applications for inbound and outbound processing with PeopleSoft.

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

Adapter Framework

The following diagram illustrates the BEA WebLogic Adapter for PeopleSoft framework.



The Integration Business Services Engine (iBSE) and Application Explorer (iAE) are deployed to the J2EE Web container. The Connector for JCA is deployed to the JCA container in the J2EE server. The repository is written at design time and read at run time. The container (iAE, iWAF, or iBSE) is responsible for the connection to the repository as shown.

Integrating With PeopleSoft

The adapter uses your application server and XML messages to enable non-PeopleSoft applications to communicate and exchange transactions using one of the following two facilities:

- PeopleSoft Component Interface

- PeopleSoft Integration Broker (in release 8.4) or Application Messaging Manager (in release 8.1).

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.

The adapter is bidirectional, enabling it to:

- Detect an event by receiving an XML document from PeopleSoft using Integration Broker or Application Messaging.

Pass an XML request document to execute an instance of the PeopleSoft Component Interface and its method.

Supported Tools and Technologies

The BEA WebLogic Adapter for PeopleSoft works in conjunction with the following components:

- Application Explorer
- Integration Business Services Engine (iBSE)
- Enterprise Connector for J2EE™ Connector Architecture (JCA)

Application Explorer

Application Explorer uses an explorer metaphor to browse the PeopleSoft system for metadata. The explorer enables you to create XML schemas and Web services for the associated object. In addition, you can create ports and channels to listen for events in your PeopleSoft application. External applications that access PeopleSoft through the BEA WebLogic Adapter for PeopleSoft use either XML schemas or Web services to pass data between the external application and the adapter.

Integration Business Services Engine

The Integration Business Services Engine (iBSE) exposes—as Web services—enterprise assets that are accessible from adapters regardless of the programming language or the particular operating system.

iBSE simplifies the creation and execution of Web services when running:

- Custom and legacy applications
- Database queries and stored procedures
- Packaged applications

- Terminal emulation and screen-based systems
- Transactional systems

Web services is a distributed programming architecture that solves Enterprise Application Integration (EAI) hurdles that other programming models cannot. It enables programs to communicate with one another using a text-based, platform- and language-independent message format called XML.

Coupled with a platform and language independent messaging protocol called SOAP (Simple Object Access Protocol), XML enables application development and integration by assembling previously built components from multiple Web services.

Enterprise Connector for J2EE Connector Architecture (JCA)

The Enterprise Connector for J2EE Connector Architecture (JCA) enables developers of JCA-compliant applications to deploy adapters as JCA resources. The connector is supported on J2EE-compliant application servers, such as your application server.

The Connector for JCA is distributed as a standard Resource Adapter Archive (RAR) for deployment to the application server. Thus, the connector can be used in systems that are non-compliant, although services such as pooled connections are not available.

Understanding Web Services and Java Connector Architecture Functionality

The following section describes how the BEA WebLogic Adapter for PeopleSoft can incorporate Web services and Java Connector Architecture technology.

Web Services

Web services allow PeopleSoft application calls to be made across the Internet or an intranet, using specialized versions of the XML language that allow a developer to specify the parameters, connections methods, and remote calls and store them for reference in a repository. At runtime, a person, an interface, or another function, can read this repository and automatically invoke the service. Web services currently do not have industry standards for transactional behavior. Web services are useful when your function calls must be made across firewall boundaries. Using Web services, you can use functions provided by external providers, as long as you know the function interface.

Java Connector Architecture

Java Connector Architecture (JCA) provides a reusable component model to build and deploy multi-tier applications that are platform and vendor-independent. JCA acts as a type of envelope or “container” that will allow the adapter to run inside the application server and connect to PeopleSoft and immediately return the results. JCA is useful when your PeopleSoft application system resides within a local intranet or is accesed directly. JCA implements JAVA Connection and Transaction models. JCA requires a resource adapter to be physically deployed on the host application server to access the remote EIS system.

Using combinations of JCA and Web services is possible. For example, a JCA application can be invoked via a Web service or a Web service may be implemented inside a JCA container. The standards and protocols are still evolving.

Understanding JCA Managed and Non-managed Modes

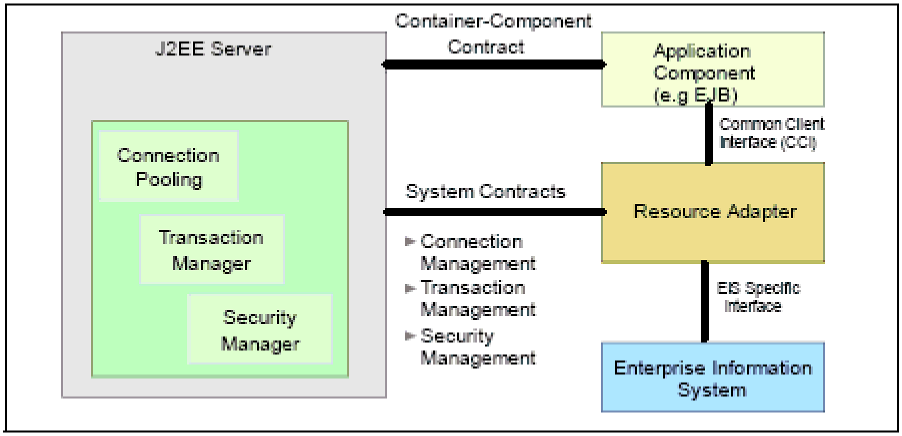
The J2EE Connector Architecture (JCA) provides a specification that standardizes access to Enterprise Information Systems (EIS). When a connection to an EIS is made, this connection can exist in a managed or non-managed mode, depending on your environment and deployment method.

The following section provides an overview of JCA managed and non-managed modes and their key differences. In addition, information relating to the deployment of the Connector for JCA in a managed or non-managed mode is provided.

JCA Managed Mode

By default, the Connector for JCA runs in managed mode when deployed to an application server, for example, BEA WebLogic. In this case, the BEA WebLogic server manages the connections, transactions, and security. As a result, the Connector for JCA provides features such as connection pooling. Connection pooling allows the BEA WebLogic server to pool connections to a back-end EIS, which results in efficient performance and increased scalability.

The following diagram illustrates the Connector for JCA (Resource Adapter) being used in managed mode. In this scenario, the application is running inside the J2EE application server.



In order to make connections to an EIS in managed mode, the Connector for JCA obtains an `IWAFConnectionFactory` from the BEA WebLogic server. In managed mode, the `IWAFConnectionFactory` is accessed from the JNDI and its properties will be configured by the BEA WebLogic server.

The following connection properties can be found in the weblogic-ra.xml file, which is packaged in the iwafjca.rar file.

```
<connection-instance>
        <jndi-name>eis/IWAFConnectionFactory</jndi-name>
         <connection-properties>
            <pool-params>
               <initial-capacity>0</initial-capacity>
               <max-capacity>10</max-capacity>
            </pool-params>
         </connection-properties>
</connection-instance>
```

The following connection properties can be found in the web.xml file, which is used by the JCA IVP test tool to run in managed mode.

```
<context-param>
    <description>JNDI name for the IWAF JCA Resource Adapter.
      If not provided, the application will create a new one based on
      iway.home, iway.config and iway.loglevel.
    </description>
    <param-name>iway.jndi</param-name>
    <param-value>eis/IWAFConnectionFactory</param-value>
</context-param>
```

The iway.jndi parameter is the connection factory name for Connector for JCA. The connection factory name under BEA WebLogic is eis/IWAFConnectionFactory. The JCA IVP test tool attempts to connect to the adapter via JNDI if JNDI is defined (managed mode). If JNDI is undefined, iway.home, iway.config, and iway.loglevel are used instead (non-managed-mode).

The default location of the web.xml file on Windows is:

```
C:\Program Files\iWay55\bea\iwfjcaivp\WEB-INF\web.xml
```

JCA Non-managed Mode

You can use the Connector for JCA in non-managed mode by embedding it in an application instead of deploying it to an application server. In this case, the application must manage connections, transactions, and security itself. To enable non-managed mode for the the Connector for JCA, you must specify the iWay55\lib connector library in the classpath. No RAR file or ra.xml descriptor is needed. In non-managed mode, the default connection manager is used.

Performance may be reduced when deploying the Connector for JCA in non-managed mode since connection pooling is not available.

Application Design

This section addresses best practice principles that can be used during application design stages when using the BEA WebLogic Adapter for PeopleSoft.

Web Services and JCA Usage Considerations

The following four factors explain the differences between deploying Web services and deploying the JCA option. Understanding the factors can help in selecting a deployment option.

1. Web services is the preferred deployment option because JCA:

- Can be deployed in a separate instance of BEA WebLogic server.
- Provides better distribution of load.
- Provides better isolation from any errors from third-party libraries.
- Provides better capability to isolate issues for debugging purposes.
- Conforms more closely to SOA model for building applications.

2. JCA provides slightly better performance

   JCA does provide slightly better performance than the Web services option; however, the difference decreases as the transaction rate increases.

3. The Web services and JCA options both provide identity propagation at run time.

The Web services option provides the capability to pass identity using the SOAP header. For the JCA option, user name and password can be passed using the connection spec of the CCI.

4. Transactions

   Because no adapters currently being resold by BEA support XA transactions, transactions are not a consideration when choosing between JCA and Web services.

Configuring a Clustered Environment

Events can be configured in a clustered BEA WebLogic environment. You can deploy iBSE or JCA to this environment.

A cluster consists of multiple server instances running simultaneously, yet appears to clients to be a single server instance. The server instances that contain a cluster can be run on one machine, but are usually run on multiple machines.

Service requests are processed through the HTTP router and routed to an available managed server.

Events are server-specific and are not processed through the HTTP router. You must configure each server separately.

A WebLogic cluster is a group of servers that, from the application point-of-view, operate as a single server. A cluster provides:

- **Scalability.** Additional servers can be added to the cluster dynamically to increase capacity.
- **High-availability.** Redundancy of multiple servers insulates applications from failures.

The following operations performed on one managed server will be replicated on all other managed servers:

- Create port and channel. Creates the channel and port under all available servers.
- Delete port and channel. Deletes the port and channel under all available servers.

The following operations must be performed on each server:

- Start channel. Starts the channel for the specific server.
- Stop channel. Stops the channel for the specific server.

For more information on configuring BEA WebLogic for deployment in a clustered environment, see the *Deploying WebLogic Integration Solutions* documentation.

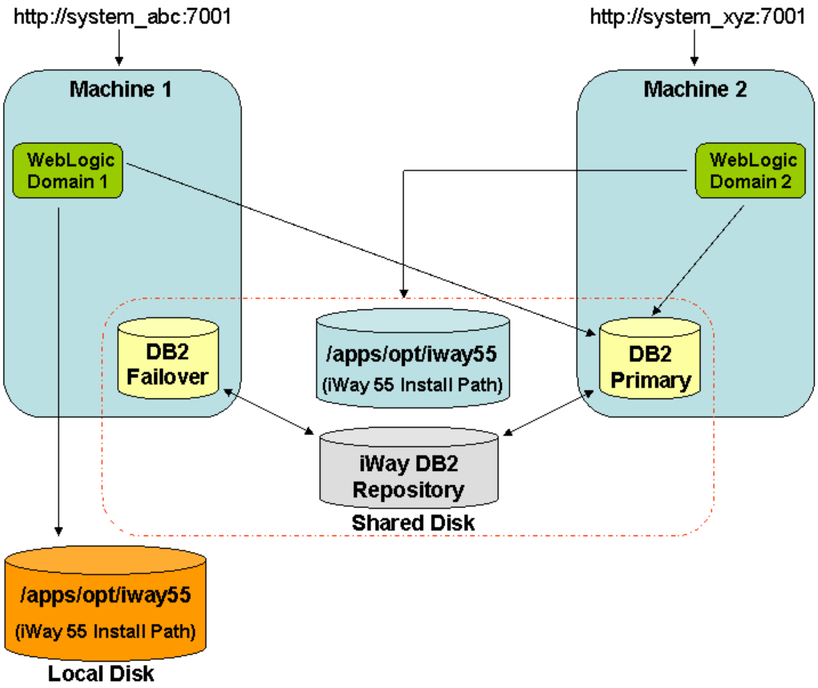
Load Balancing Support with WebLogic Server

The WebLogic Administration Console provides you with the tools to configure WebLogic for load balancing. Load balancing is one of the primary tools employed in making systems scalable. BEA WebLogic Adapter for PeopleSoft takes advantage of the sophisticated load balancing features of WebLogic clusters.

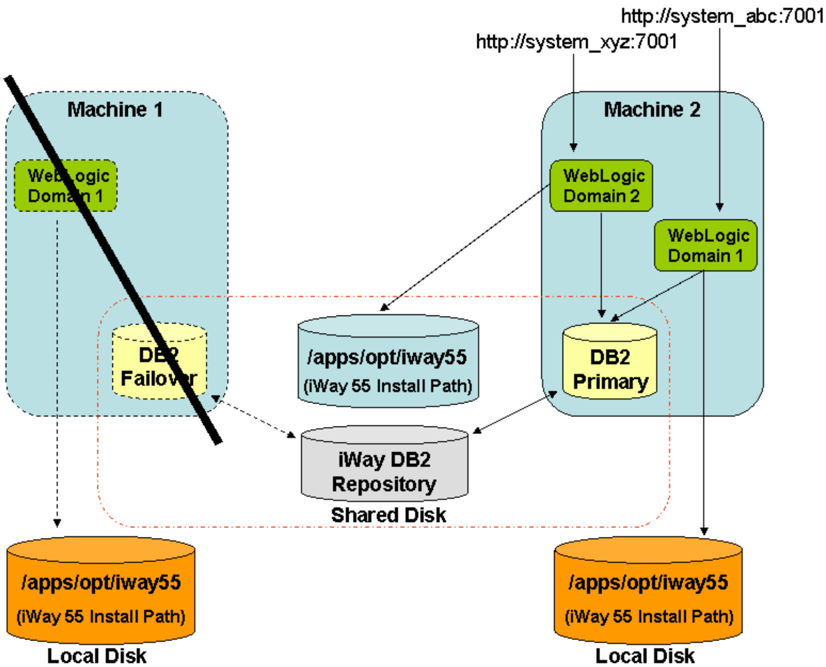
Failover Support with WebLogic Server

Failover is the ability of a system to detect a service failure and automatically retry the operation using another Gateway. This includes the ability to recover in the event of a server failure and the ability to find another instance of a service on an available server. Under the best of conditions the results of a service are idempotent, meaning that multiple invocations of the service, with the same input, produce the same output with no side effects. An example of an idempotent service would be a currency translation service. The service produces the same output each time it is run for a given input value and it has no side effects. Under these conditions it is safe to failover a request for any type of failure.

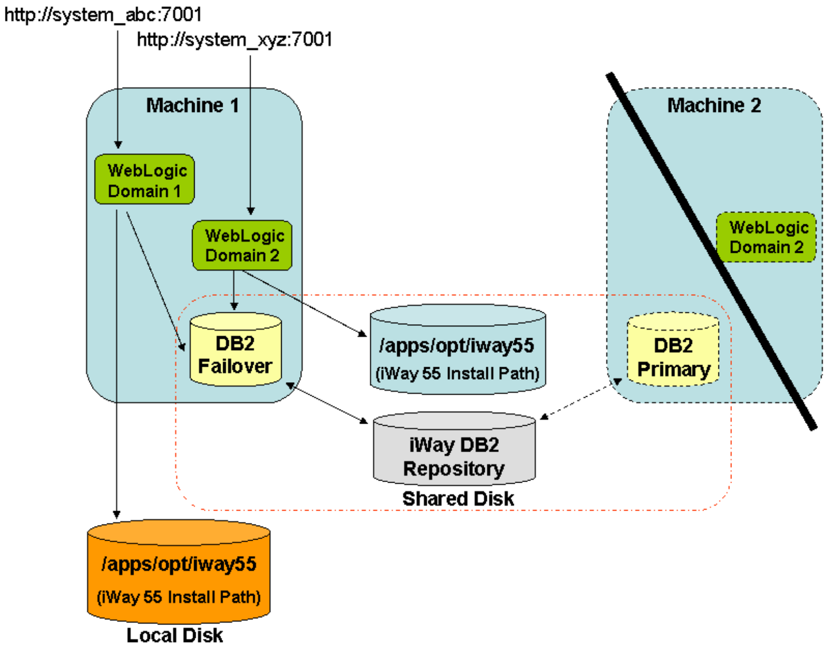
The following diagram illustrates a sample configuration where two WebLogic domains are configured on two separate machines. DB2 Failover and DB2 Primary represent a scenario where DB2 is configured for each WebLogic domain respectively. In the event of a failover condition, the DB2 Primary database instance would be switched over to the DB2 Failover instance.


http://system_abc:7001
http://system_xyz:7001
Machine 1
Machine 2
WebLogic Domain 1
WebLogic Domain 2
DB2 Failover
/apps/opt/iway55
(iWay 55 Install Path)
DB2 Primary
iWay DB2 Repository
Shared Disk
/apps/opt/iway55
(iWay 55 Install Path)
Local Disk

The following diagram illustrates a failover situation when the first machine becomes unavailable. As you can see, Machine 2 takes over in the clustered environment.



The following diagram illustrates a failover situation when the second machine becomes unavailable. In this case, Machine 1 takes over in the clustered environment.



Configuration Tuning

This section explores tuning considerations during design time and run time.

Repository Migration Using iBSE Administrative Services

iBSE Administrative Services allow a client application to export, import, and modify an existing iBSE repository, regardless of the underlying format. When suitably configured, the iBSE Administrative Services represent a best practice methodology for maintaining separate development, QA, and production environments.

For more information on using iBSE administrative services, see Chapter 4, *Management and Monitoring*.

JCA Troubleshooting

JCA provides the ability to set multiple log levels. The log level is set in the META-INF\ra.xml file.

To change defaults, you must:

1. Extract the META-INF\ra.xml file from the iwafjca.rar archive.

   a. Open a command prompt and navigate to the directory containing the connector, for example:

      ```
      iWay55\etc\setup
      ```

      where:

      `iWay55`
      Is the full path to your iWay 5.5 installation. The default is C:\Program Files\iWay55.

   b. Issue the following command:

      ```
      jar xvf iwafjca.rar META-INF/ra.xml
      ```

      The JAR command is located in the Java SDK bin directory which might not be in your search path. If you receive an error, execute the jar command using its full path. This path varies depending on which version of Java is installed, for example:

      ```
      C:\j2sdk1.4.1_03\bin\jar -xvf iwafjca.rar META-INF/ra.xml
      ```

      Note: Ensure you use the JAR command and not Winzip. Winzip does not properly extract Java related archives.

2. Open the extracted ra.xml file in a text editor.
3. Modify the contents of the <param-value> tags to change defaults.

   - **LogLevel.** Trace setting. This can be set to DEBUG, INFO, or ERROR.

     ```
     <context-param>
     <config-property>
         <config-property-name>LogLevel</config-property-name>
         <config-property-type>java.lang.String</config-property-type>
         <config-property-value></config-property-value>
     </config-property>
     ```

     For example:

     ```
     <config-property-value>DEBUG</config-property-value>
     ```

     Leave the remainder of this file unchanged.

4. Save the file and exit the editor.

5. Use the JAR command to return the ra.xml file to the META-INF directory within the archive. To do this:

   a. Ensure that you are in the following directory that contains the connector:

      `iWay55\etc\setup`

      where:

      iWay55

      Is the full path of your iWay 5.5 installation directory. The default is C:\Program Files\iWay55.

   b. Issue the following command:

      ```
      jar -uvf iwafjca.rar META-INF/ra.xml
      ```

6. Redeploy the connector.

   The trace information is written to the BEA WebLogic server log file.