# BEA WebLogic Adapter for PeopleSoft User's Guide

DN3501560.0406

April 20, 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** | Using Web Services Policy-Based Security | Describes how to configure Integration Business Services policy-based security. |
| **8** | Management and Monitoring | Describes the management and monitoring tools provided by iBSE and JCA. |

| Chapter/Appendix | | Contents |
|---|---|---|
| **9** | Troubleshooting | Explains the limitations and workarounds when connecting to PeopleSoft. |
| **A** | Using Application Explorer to Create XML Schemas and Business Services | Describes how to use Java Swing Application Explorer running in BEA WebLogic Workshop to create XML schemas and Integration Business Services for PeopleSoft Component Interfaces (CI) and Messages. |
| **B** | Using Application Explorer for Event Handling | Describes how to use Java Swing Application Explorer running in BEA WebLogic Workshop to create events for PeopleSoft. |
| **C** | Using Component Interfaces | Describes how to create, secure, and test a Component Interface for use with the BEA WebLogic Adapter for PeopleSoft. |
| **D** | 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. |

## Documentation Conventions

The following table lists the conventions that apply in this manual and a description of each.

| 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. |

| Convention | Description |
|---|---|
| { } | 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 (…). |
| .<br>.<br>. | 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, 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 |

| |
|---|
| XML schema |
| 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 PeopleSoft systems with other applications.

# 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 work with:

- **Component Interfaces.** If a Component Interface does not exist, create, secure, and test one. For more information, see Appendix C, *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 C, *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 work with:

- **The Integration environment.** Configure and test your PeopleSoft Integration Broker (release 8.4) or Application Messaging environment (release 8.1). To ensure that the environment is properly configured, see Appendix D, *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 BEA WebLogic 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 Application Messaging Manager

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 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 greatest amount of flexibility and functionality.

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

- PeopleSoft Java™ API using Component Interfaces.
- PeopleSoft XML using Application Messaging.

## PeopleSoft Component Interface

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

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

PeopleSoft delivers generic Component Interfaces with each of its applications. These are called Enterprise Integration Points (EIP). Customers also can develop their own custom Component Interfaces, or they can modify EIP as required.

### PeopleSoft Application Messaging Manager

PeopleSoft Application Messaging 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 Servlet Application Explorer with either of the following components:

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

Both iBSE and the Connector for JCA are deployed to your application environment with Application Explorer and the adapters.

### Deployment Information Roadmap

The following table lists the location of deployment information for the iWay Adapter for PeopleSoft 8. A description of Application Explorer, the Integration Business Services Engine (iBSE), and the Enterprise Connector for J2EE Connector Architecture (JCA) follow the table.

| Deployed Component | For more information, see |
|---|---|
| Application Explorer | Chapters 6 and 7 of this guide<br>Appendix A and B of this guide when using Application Explorer inside WebLogic Workshop<br>*Application Explorer User's 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*<br>*Connector for JCA for BEA WebLogic User's Guide* |

## Application Explorer

Application Explorer uses an explorer metaphor to browse the PeopleSoft system for metadata and create Web services and events.

## Integration Business Services Engine

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

iBSE 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 but 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

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 BEA WebLogic 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:**



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

- Specify which version of PeopleSoft you are using.
- 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 PeopleSoft Version

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
```

Use the corresponding location on non-Windows systems.

To ensure the adapter functions properly, remove the file that does not correspond to your release:

- For PeopleSoft 8.4, remove the following file:

  ```
  iwpsci81.jar
  ```

- For PeopleSoft 8.1, remove the following file:

  ```
  iwpsci84.jar
  ```

After changing 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 PeopleSoft Release 8.4, it is the IWY_CI_84 project, packaged in iwpsci84.zip.
- For PeopleSoft Release 8.1, it 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
```

Use the corresponding location on non-Windows systems. 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.

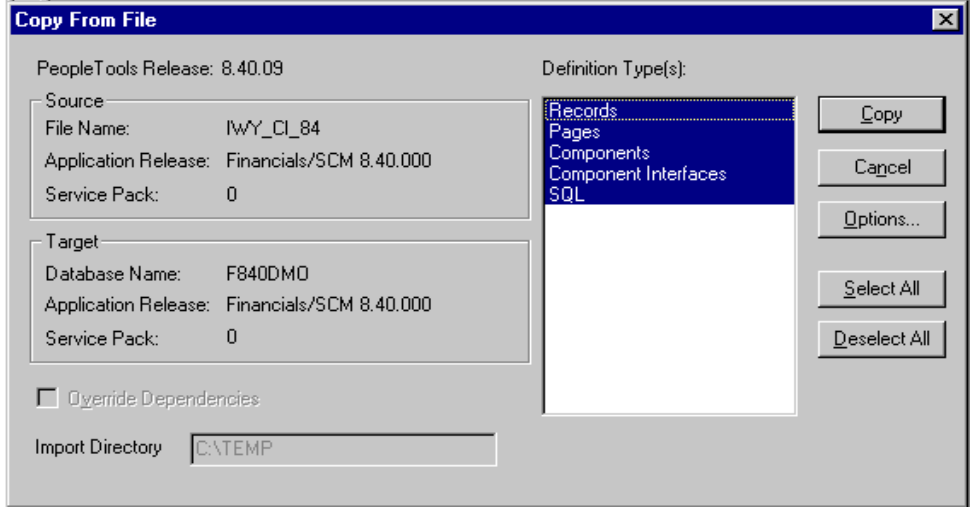
   The unzip process creates its own subdirectory. For example, if you extract the file to c:\temp, it creates c:\temp\IWY_CI_81 or c:\temp\IWY_CI_84.

2. Launch the PeopleSoft 8 Application Designer in two-tier mode.
3. Open the Copy From File Select Project dialog box as follows:
   - In PeopleSoft 8.4, choose *Copy Project* from the Tools menu, and then choose *From File*.
   - In PeopleSoft 8.1, choose *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. Click *Open* (in release 8.4) or *Copy* (in release 8.1) to open the Copy From File dialog box.

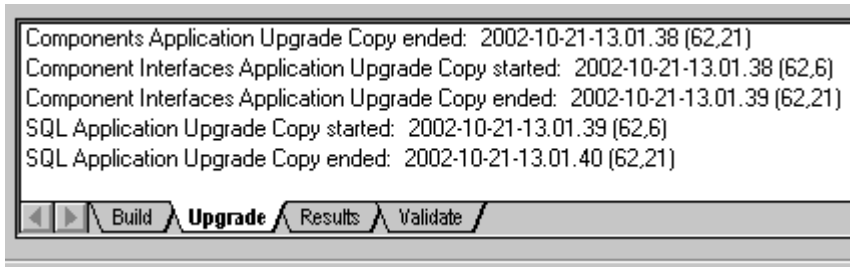
   **Note:** Although the following images illustrate PeopleSoft release 8.4, the corresponding instructions are accurate for releases 8.1 and 8.4.

   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.



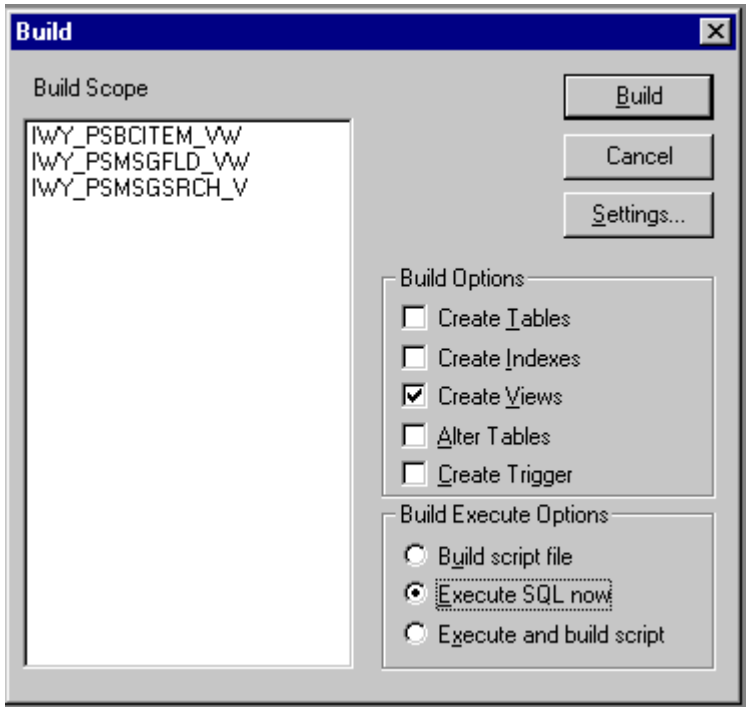

6. Highlight all objects listed under Definition Types and click *Copy*.

The following image shows a message generated by Application Designer, which 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. The steps to build the options follow.

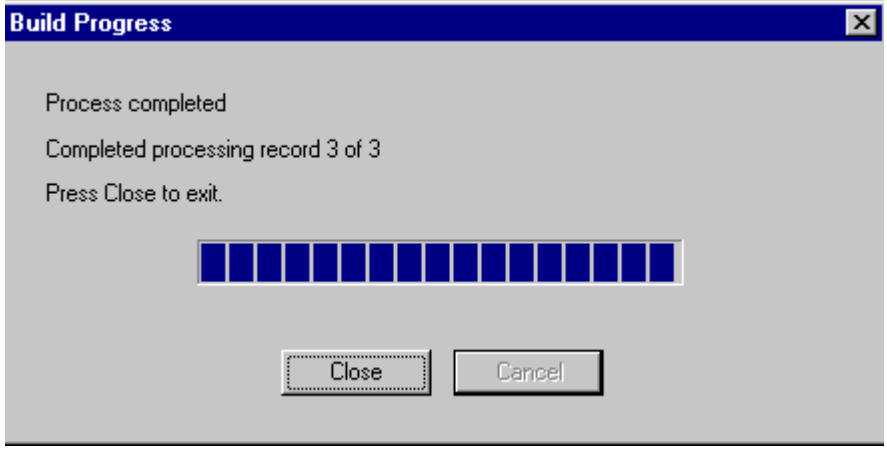



**a.** From Build Options, select *Create Views*.

**b.** Select your site's customary option in Build Execute Options. (In the previous figure, Execute SQL is selected.)

**8.** Click *Build*.

The following image shows the Application Designer Build Progress status window. It contains a summary of the process completed, a status bar, and Close and Cancel buttons.



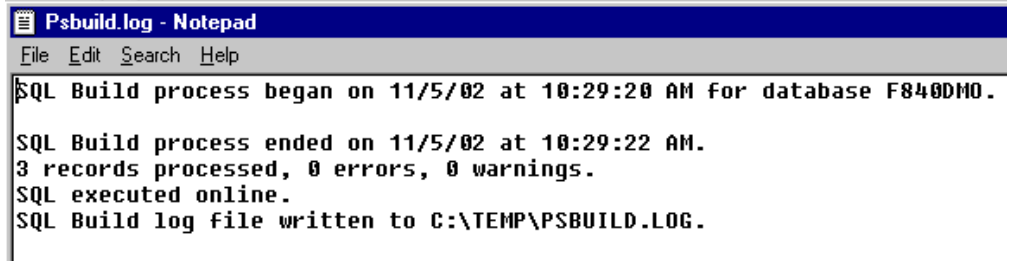

**Note:** There are zero errors and zero warnings.

**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 view was not correctly generated, click *Close*.

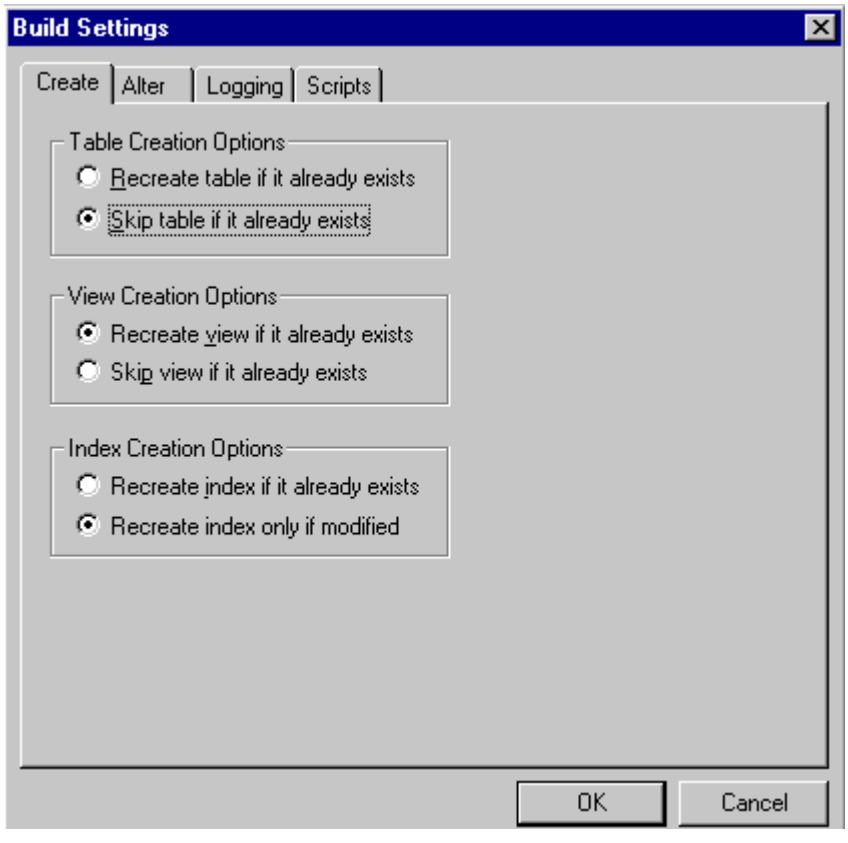
**9.** Double-click the SQL Build log statement.

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 (selected), Alter, Logging, and Scripts, and three options: Table Creation Options, View Creation Options, and Index Creation Options. It also includes OK and Cancel buttons.




Depending on the application server database for PeopleSoft 8, some databases 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 2-7.

## Configuring Component Interface Security

Application Explorer requires the custom Component Interfaces that you imported and built in the previous step, so you need to 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 which users are going to 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 used 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 any possible security exposure.

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

The following steps describe how to configure security for all supported releases of PeopleSoft in all supported modes. The figures shown in the steps reflect PeopleSoft release 8.4 in 4-tier mode.

### Procedure: How to Configure Component Interface Security

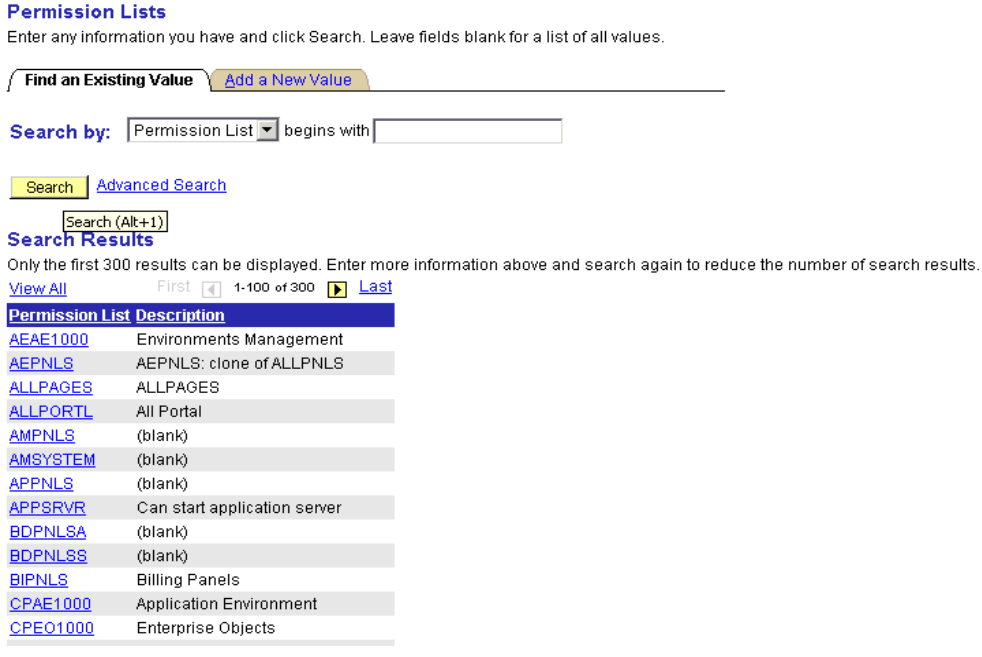
To configure security for each BEA WebLogic Adapter for PeopleSoft Component Interface:

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

   The following image shows the expanded Security menu displaying a list of choices under Permissions & Roles, with Permission Lists being the first choice.

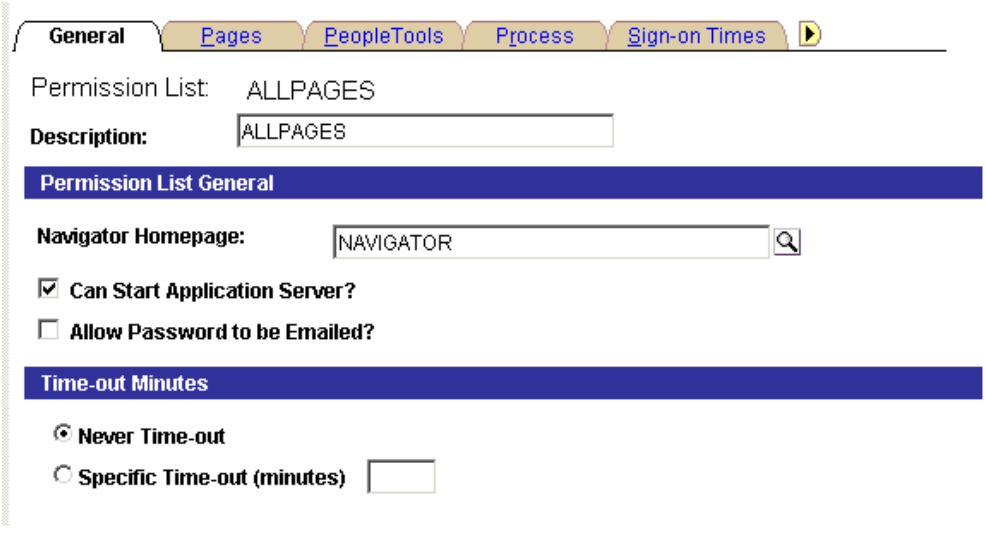

2. Click *Search* and select the relevant Permission List.

The following image shows the Permission Lists pane. It contains the following: Find an Existing Value tab (selected), Add a New Value tab, Search by list, begins with input field, Search button, Advanced Search hyperlink, and a Search Results section.



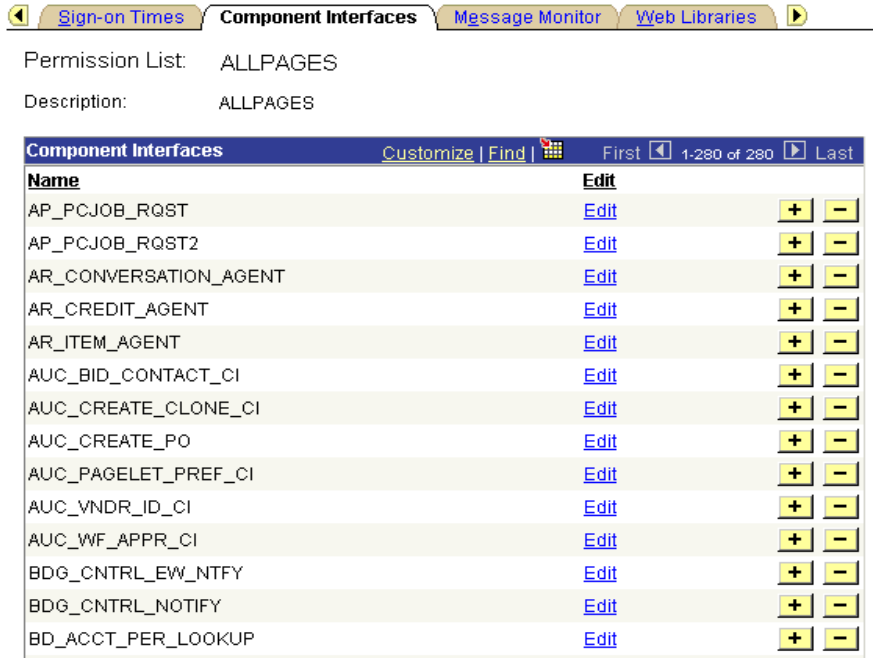

**3.** To display the Component Interfaces tab, click the arrow to the right of the Sign-on Times tab.

The following image shows the Permission List opening on the General tab, the first of five tabs. 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.




5. To add a new row to the Component Interfaces list, select the plus sign (+).
6. Enter or select the *IWY_CI_ATTRIBUTES* Component Interface and click *Edit*.
   a. To set the Get and Find methods to Full Access, click *Full Access (All)*.
   b. Click *OK*.
7. Repeat the process for the IWY_CI_MESSAGES Component Interface.
8. Scroll to the bottom of the Component Interfaces pane and click *Save*.

You have finished configuring security for the Component Interfaces delivered with BEA WebLogic Adapter for PeopleSoft. To test these Component Interfaces, see *Testing the Component Interfaces* on page 2-11.

## Testing the Component Interfaces

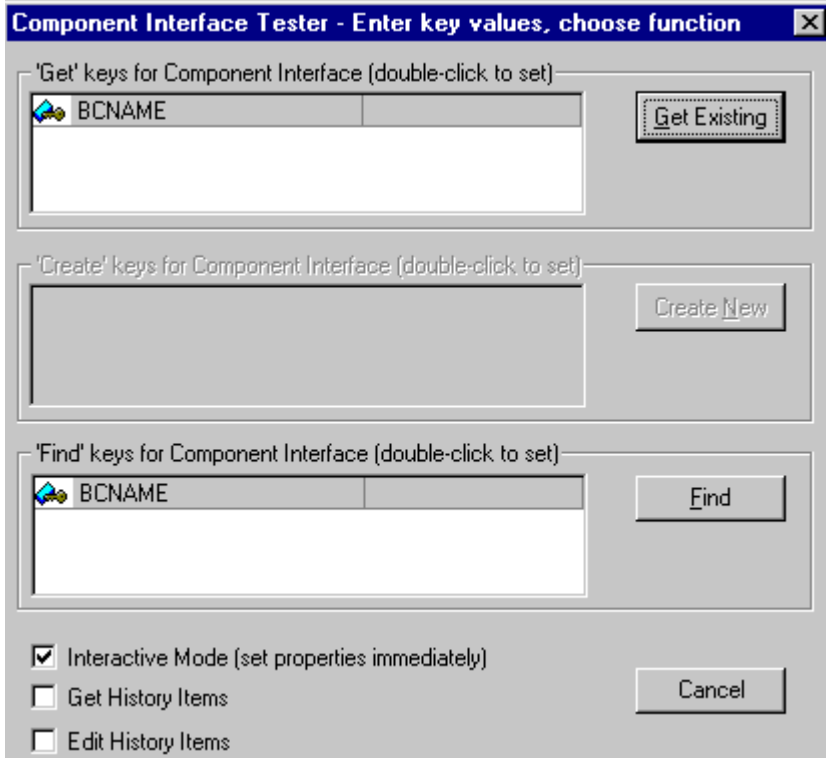
You must test each of the BEA WebLogic Adapter for PeopleSoft Component Interfaces before using them.

### 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 for Component Interface, 'Create' keys for Component Interface (unavailable), and 'Find' keys for Component Interface. It also includes an Interactive Mode check box, a Get History check box (selected), an Edit History Items check bo, Get Existing button (selected), Create New button (unavailable), Find button, and a Cancel button.

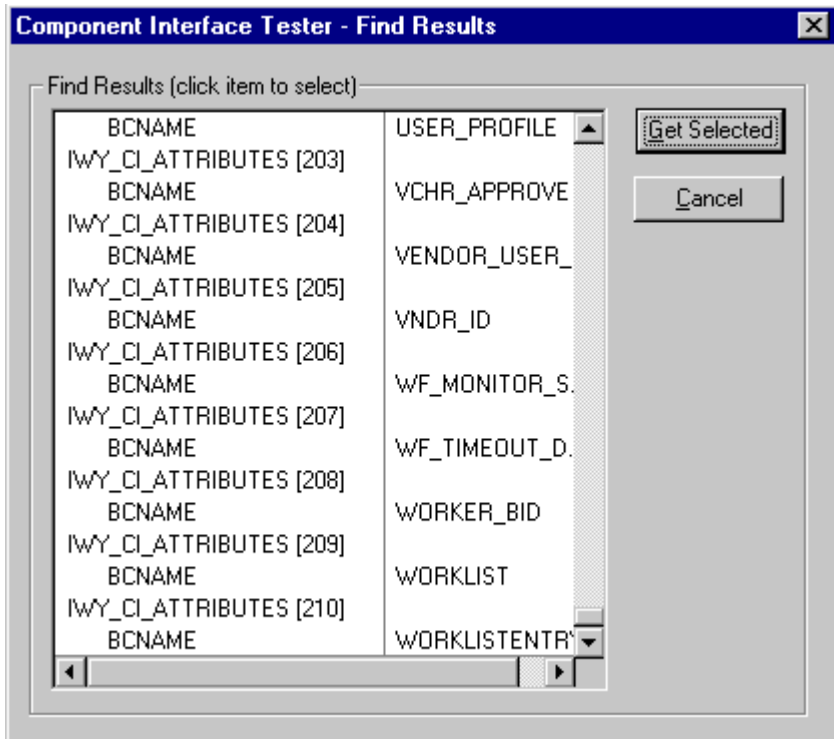


**Note:** The Create New button is inactive because the Add method is not applicable to this Component Interface.

3. Click the *Find* button.

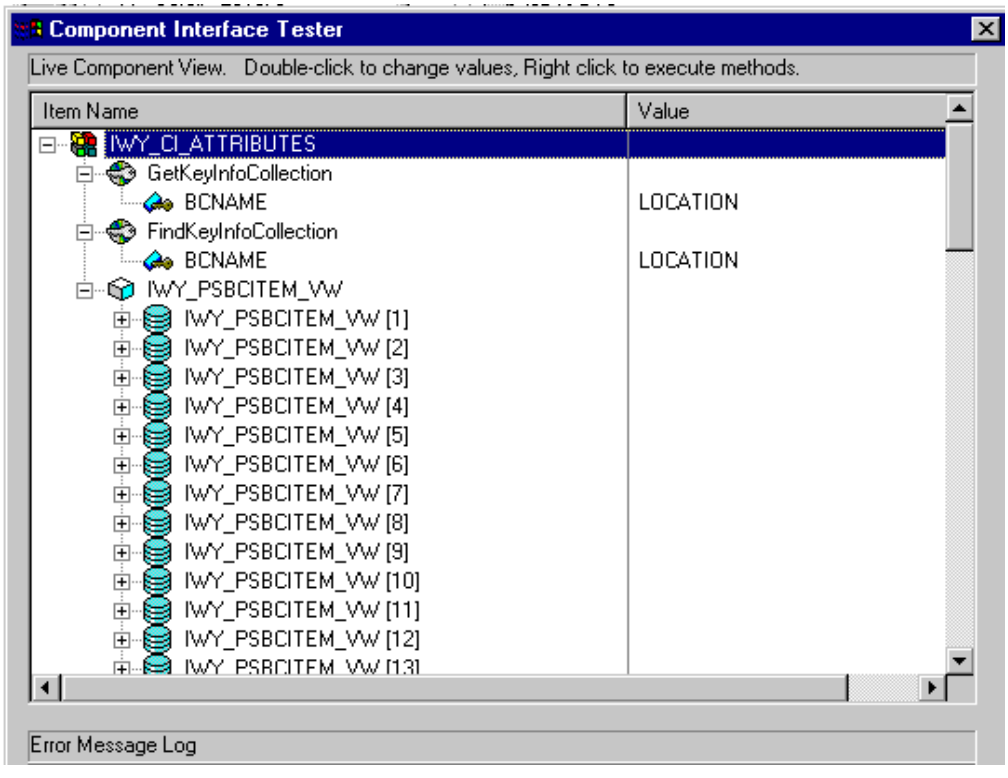
   Entries for the underlying component appear. A message may appear stating that display is limited to a certain number of entries; this is not a problem.

   The following image shows the Component Interface Tester - Find Results dialog box. It has 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. It shows the values for the keys, GetKeyInfoCollection and FindKeyInfoCollection, and the list of IWY_PSBCITEM_VW databases.

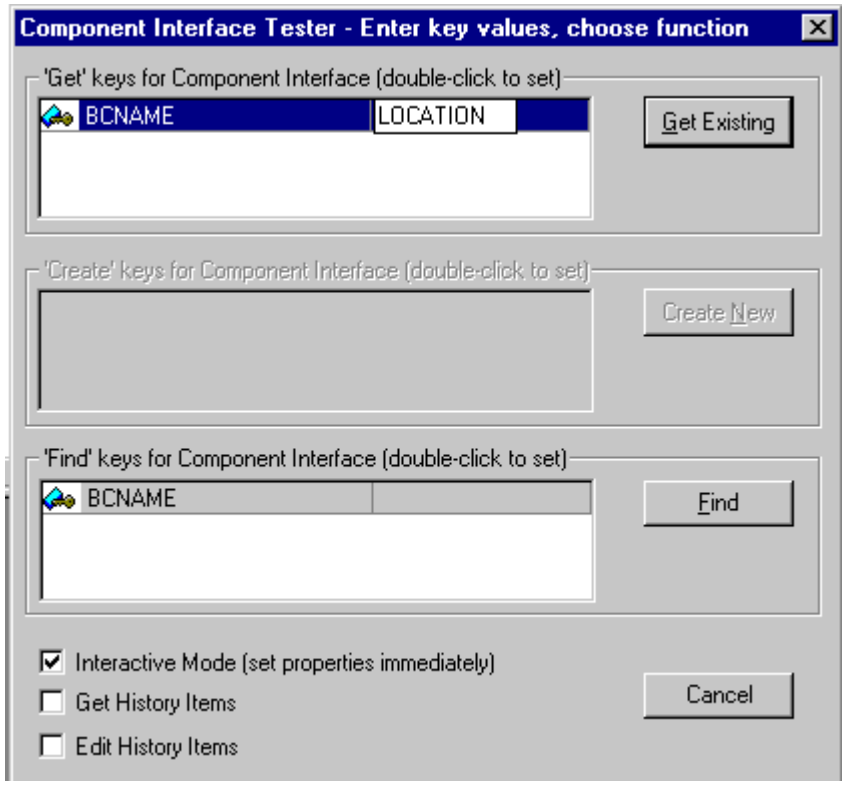



When this window appears, the Component Interface was successfully tested for the Find method.

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

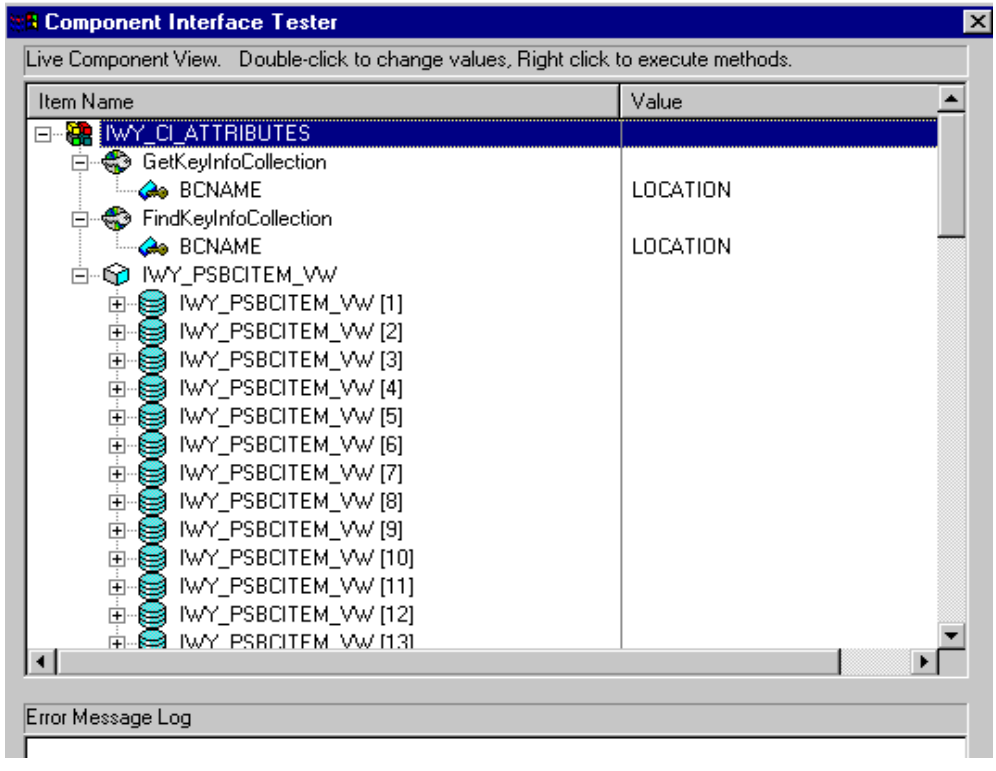
For the Get method, an existing key must be entered.

The following image shows the Component Interface Tester dialog box. It contains three panes: 'Get' keys for Component Interface, 'Create' keys for Component Interface (unavailable), and 'Find' keys for Component Interface. It also includes an Interactive Mode check box, a Get History check box (selected), an Edit History Items check bo, Get Existing button (selected), Create New button (unavailable), Find button, and a Cancel button.




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 an Error Message Log pane.




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

6. Repeat this process 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 type of TCP/IP message router required for your PeopleSoft release:

- For Release 8.4, install the TCP/IP target connector. (*Installing the TCP/IP Target Connector for PeopleSoft Release 8.4* on page 2-16)
- For Release 8.1, install the TCP/IP handler. (*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
```

Use the corresponding location on non-Windows systems.

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. Use any extraction utility appropriate for your platform.
2. Port *TCPIPTARGET84.class* to the platform where the PeopleSoft gateway Web server is located.
3. Place *TCPIPTARGET84.class* in the PeopleSoft server target connector directory. This may vary by Web or application server.

   BEA WebLogic is usually:

   ```
   c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\Web-inf\classes\com\peoplesoft\pt\integrationgateway\targetconnector
   ```

## 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
```

Use the corresponding location on non-Windows systems.

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

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

1. Port iwpsevent81.jar to the platform on which the PeopleSoft gateway Web server is located.
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 may vary by release and by Web server, but usually is:

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

3. Issue the jar command to extract the class files required by PeopleSoft.

   This is a sample command:

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

   It displays the following output 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 this, the file(s) below must be in the iWay55\lib directory. The default location for this directory on Windows is:

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

Use the corresponding location on non-Windows systems.

Ensure the following is 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, be sure to copy the new release's psjoa.jar file into the iWay55\lib directory and restart all components.

- pstools.properties (for PeopleSoft 8.1)

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

  ```
  PS_HOME\web\jmac
  ```

- psoftcrmci.jar file. This 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.

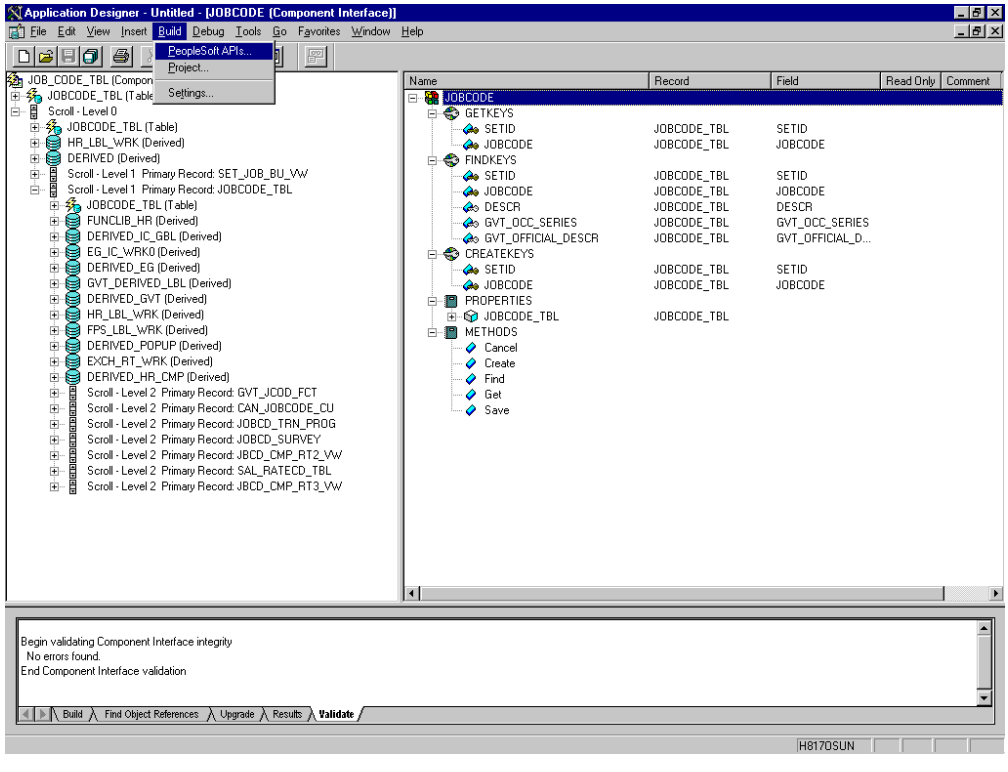
Before using your Component Interface, you must apply security to it and test it. For information about these tasks, as well as how to create a Component Interface, see Appendix C, *Using Component Interfaces*.

## Procedure: How to Create a PeopleSoft API Java Program

To create a PeopleSoft API Java program:

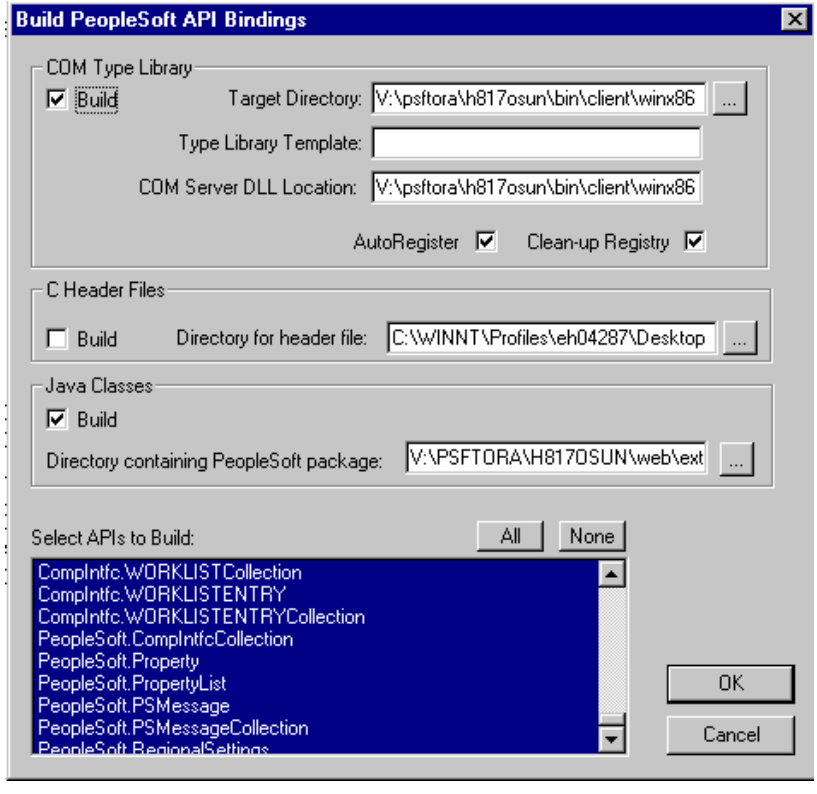
**1.** Open the *PeopleSoft Application Designer*.

The following image shows the PeopleSoft Application Designer Component Interface. PeopleSoft APIs is selected from the Build menu.



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

   The following image shows the Build PeopleSoft API Bindings dialog box where you can choose to create options for the COM Type Library and Java Classes Build.




3. Because you are creating Java files, ensure you deselect *COM Type Library Build*.
4. Ensure that Java Classes Build is selected and then, select a directory on your local machine where the Java files are to be placed, for example, c:\psoft8_components.
5. 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.

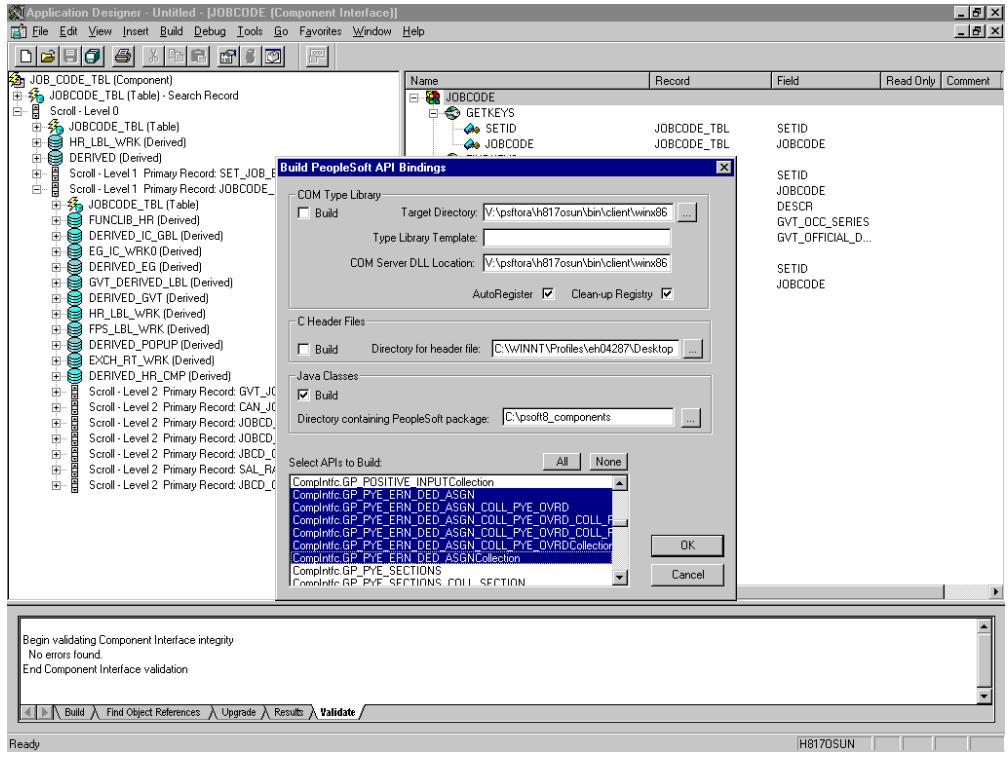
## Procedure: How to Build All of the API Files

To build all files:

1. In the Build PeopleSoft API Bindings dialog box, select the default, *All* (potentially a large number).
2. 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 illustrates the GP_PYE_ERN_DED_ASGN Component Interface from the HR 8.1 application. This image shows the Build PeopleSoft API Bindings dialog box. It contains a COM Type Library Build check box (selected), Target Directory field, Type Library Template field, COM Server DLL Location, C Header Files Build check box, Directory for Header file field, Java Classes Build check box, Directory containing PeopleSoft package field, Select APIs to Build pane, and All, None, OK, and Cancel buttons.




You are now ready to compile the Java files. For more information, see *Compiling the PeopleSoft API Java Programs* on page 3-6.

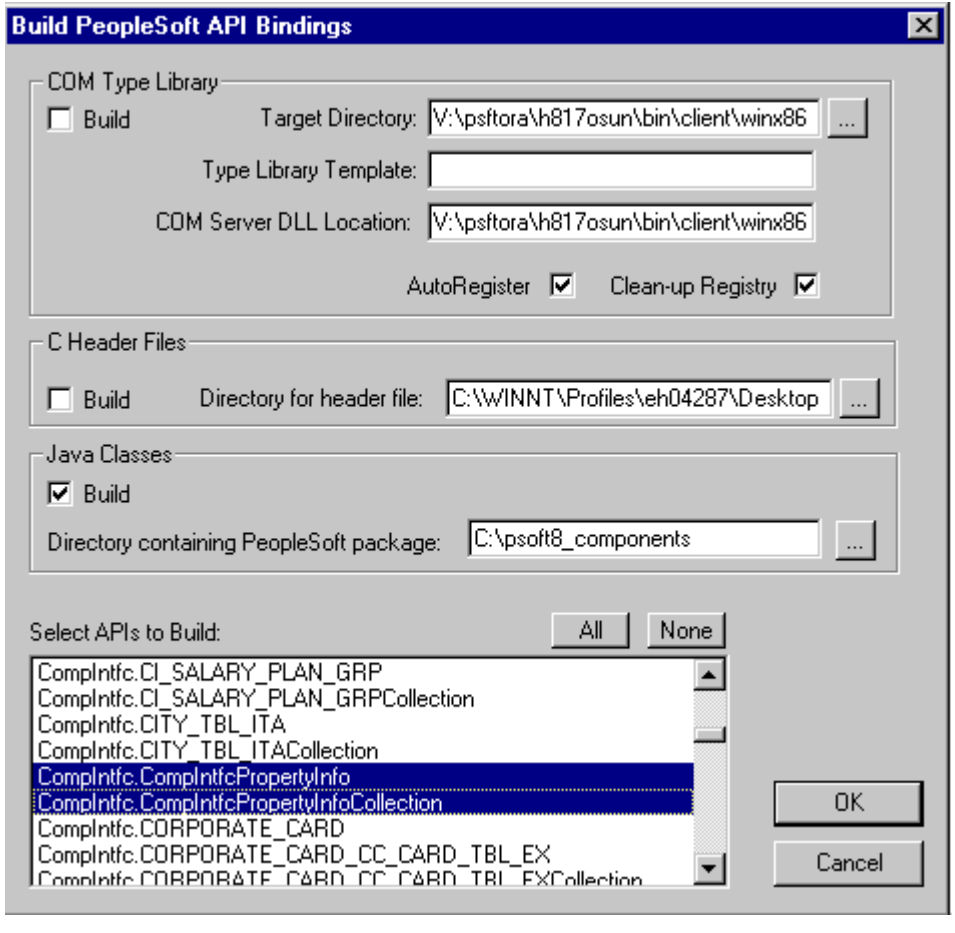
## Procedure: How to Build APIs for a Specific Component Interface

To build APIs for a specific Component Interface or interfaces:

1. In the Build PeopleSoft API Bindings dialog box, click *None*.

   This clears the selected APIs, so you can select the appropriate APIs for your Component Interface. These APIs begin with the name of your Component Interface. There may be fewer than five, or more than 50 APIs, for a particular Component Interface.

   The following image shows the Build PeopleSoft API Bindings dialog box. It contains a COM Type Library Build check box (selected), Target Directory field, Type Library Template field, COM Server DLL Location, C Header Files Build check box, Directory for Header file field, Java Classes Build check box, Directory containing PeopleSoft package field, Select APIs to Build pane, and All, None, OK, and Cancel buttons.



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

2. Select these items in the same step as the Component Interface build or select them separately.
3. Click *OK*.

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

You are now ready to compile the Java files. For more information, see *Compiling the PeopleSoft API Java Programs* on page 3-6.

## Compiling the PeopleSoft API Java Programs

PeopleSoft places the Java programs to compile in the directory called

*psoft8_components*\PeopleSoft\Generated\CompIntfc

where:

*psoft8_components*

Is the directory specified during the build process.

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

The process for compiling the PeopleSoft API Java Programs depends on whether you are compiling on the 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

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. Ensure that the psjoa.jar file is in the Java class path before you compile the programs.
2. Compile the Java programs and ensure that you include the \PeopleSoft\Generated\CompIntfc path.

   **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.
3. Compress the class files into a JAR file.

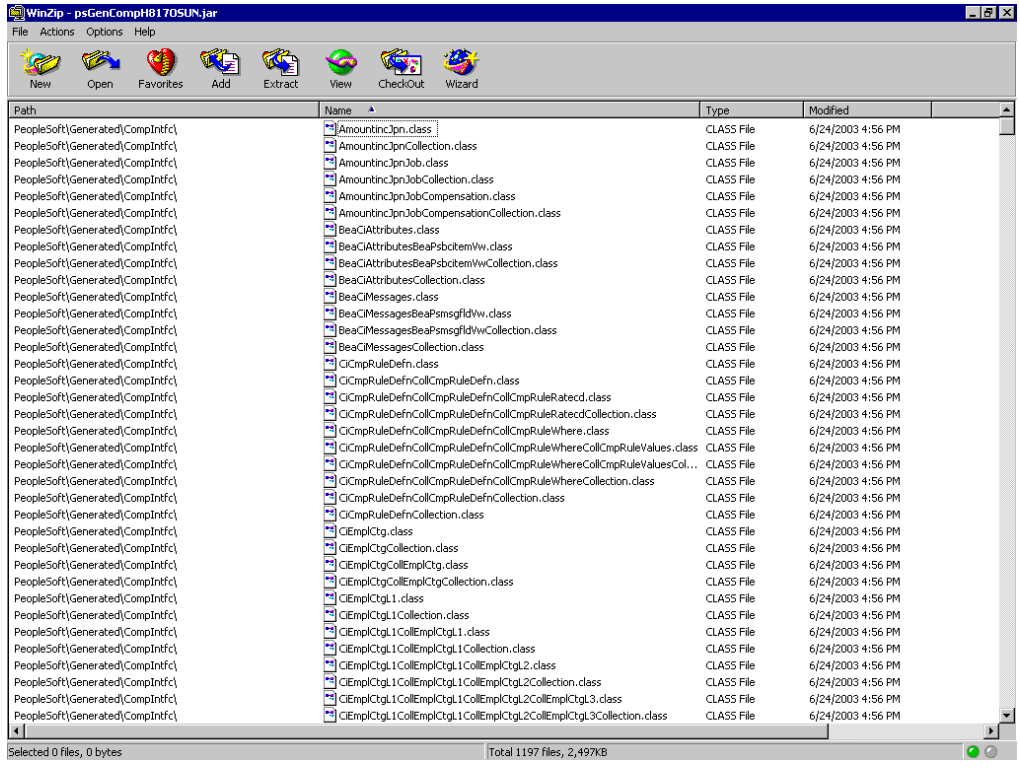
   The following Windows BAT file, if run from the psoft8_components directory, creates a correct JAR file:

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

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

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

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




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

**5.** Place the JAR file in the 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.

**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.

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.

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

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

The following configuration topics assume you are familiar with PeopleSoft Integration Broker (in release 8.4) or Application Messaging (in release 8.1). If not, see Appendix D, *Using PeopleSoft 8 Integration Broker*, for basic information about configuring and testing. For a complete description *before* you work with the BEA WebLogic Adapter for PeopleSoft, see your PeopleSoft documentation.

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

# Configuring the TCP/IP or HTTP Target Connector for PeopleSoft 8.4

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

To configure the PeopleSoft 8.4 TCP/IP or HTTP Target Connector to send messages to your BEA WebLogic 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 4-2 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.

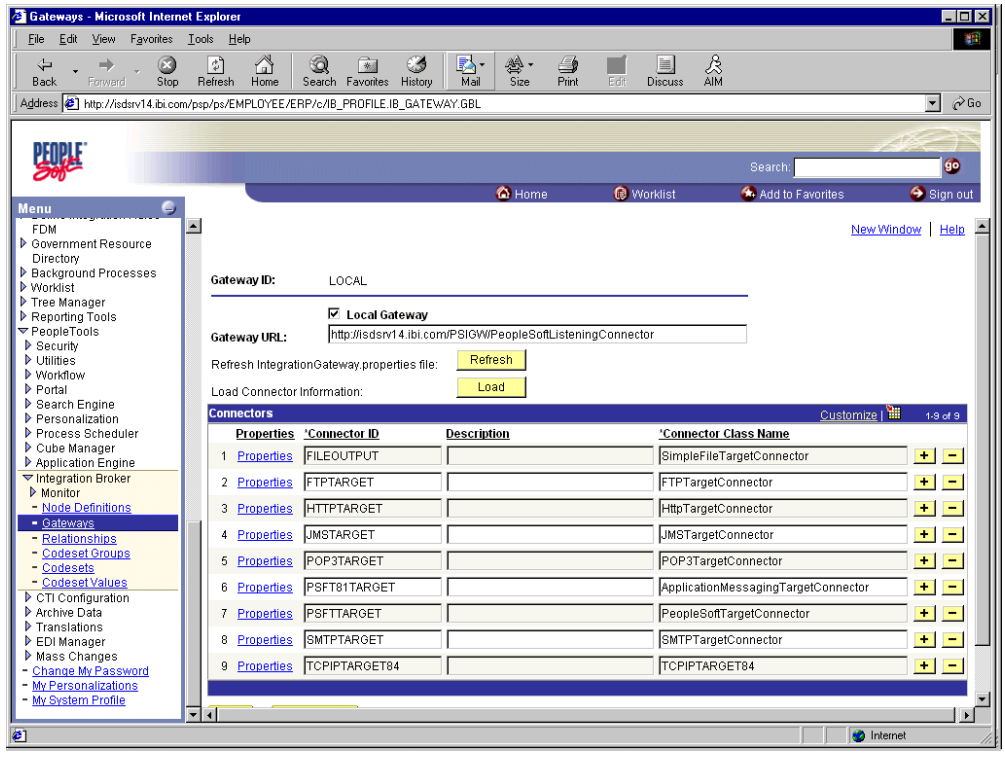
   **Note:** Starting with release 8.4, the Integration Broker is delivered with an HTTP Outbound Connector. This connector can be used in place of the iWAY TCP/IP84 Connector for sending messages to your BEA WebLogic Server.

## 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. Gateways is selected in the left pane. The right pane contains the following components: Gateway URL field, Connectors list, Refresh button, and Load button.

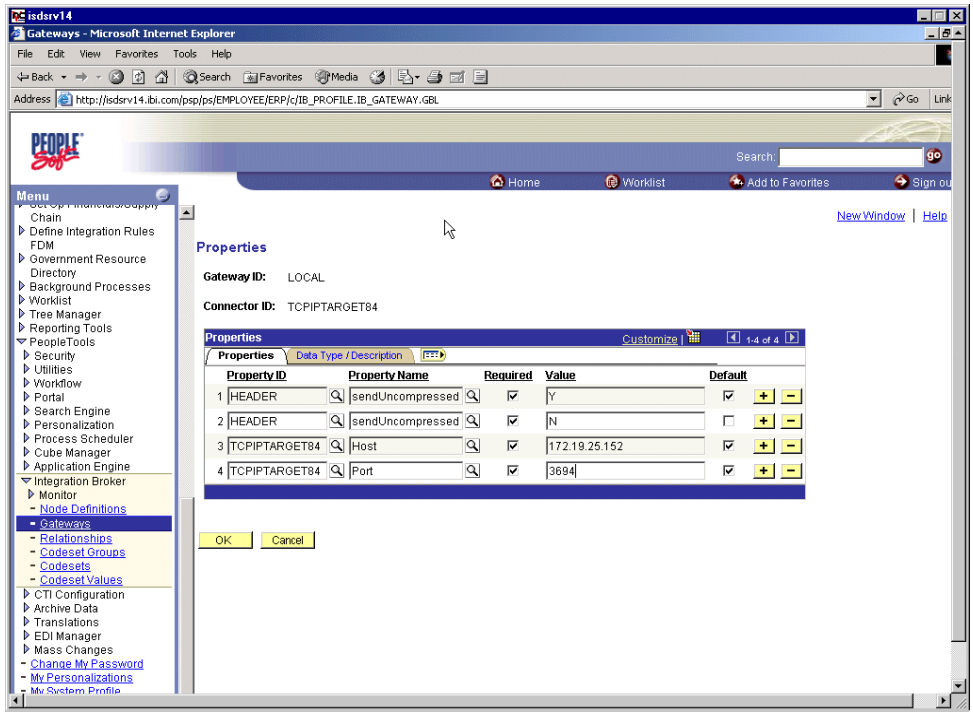



**4.** If you do not see the TCPIPTARGET84 Connector ID, click *Load* and scroll to locate *TCPIPTARGET84* in the list.

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

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

The following image shows the Properties pane for TCPIPTARGET84. Gateways is selected in the left pane. The right pane contains the following components: Gateway URL field, Connectors list, Refresh button, and Load button.




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

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

**5.** Click *OK*.

The Gateway window opens.

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

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

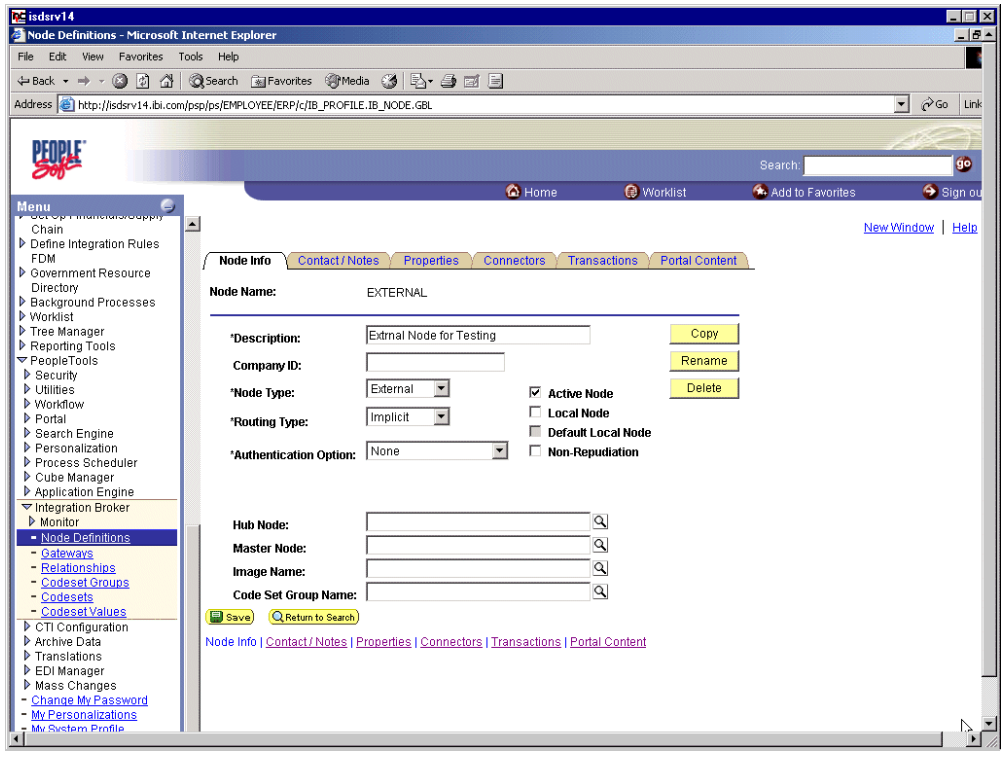
## Procedure: How to Configure the Node for the TCP/IP84 Connector

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

   The following image shows the Node Info tab selected in PeopleSoft 8 Integration Broker.




   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 BEA WebLogic Server.

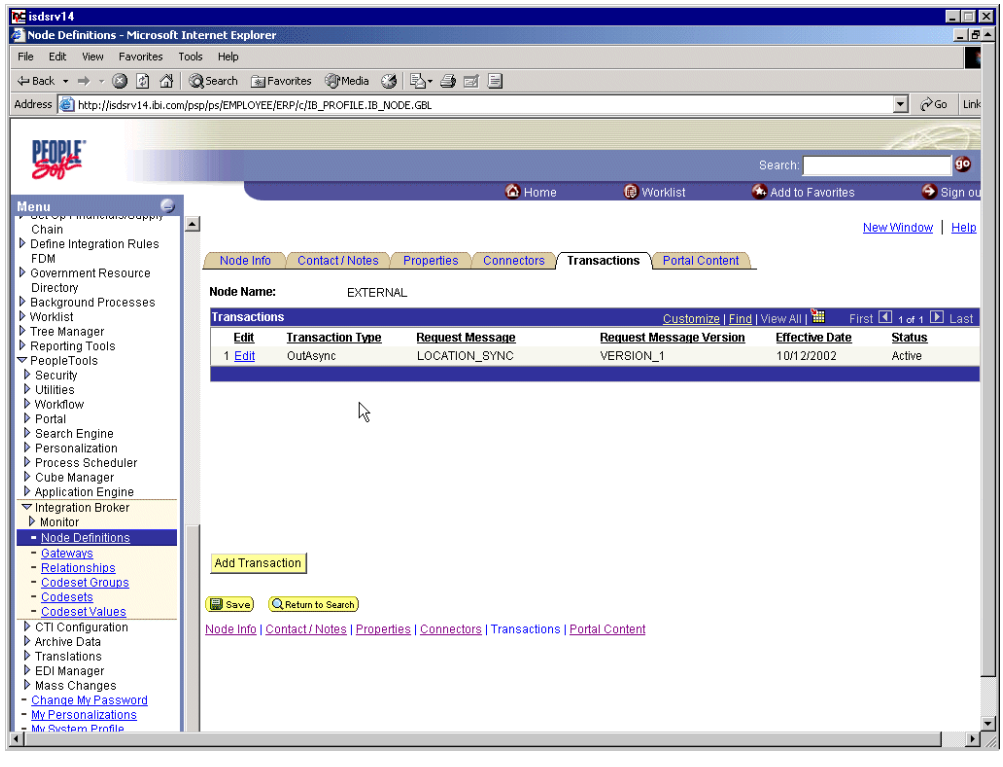
You can accept or override the default values for individual nodes.

**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 Add Transaction button, Save, and Return to Search buttons.



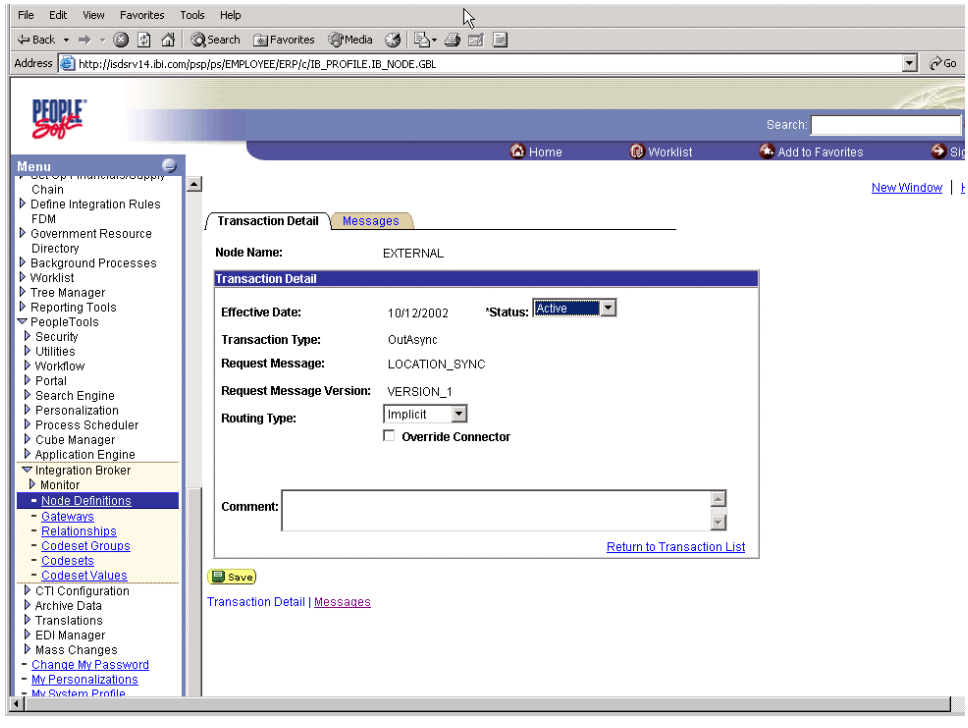

**a.** If there are no transactions, click *Add Transaction* to add the message with which you are working.

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

**b.** To view transaction details for the LOCATION_SYNC message, click *Edit*.

The Transaction Detail tab becomes available.

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




You can add the message with which you are working.

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

**d.** Click *Save*.

**5.** 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*.

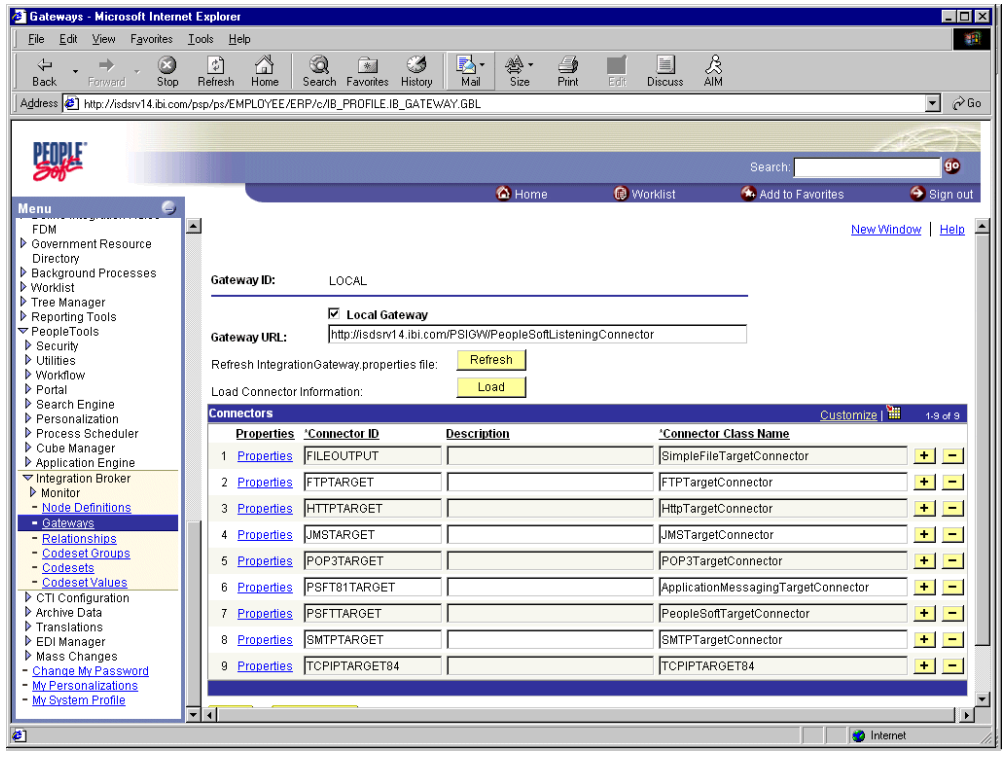
You are ready to send XML messages to your PeopleSoft XML Listener.

## 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, and contains the following: Local Gateway check box, Gateway URL field, Return button, Load button, Properties hyperlink, Connector ID field, Description field, and the Connector Class Name field.



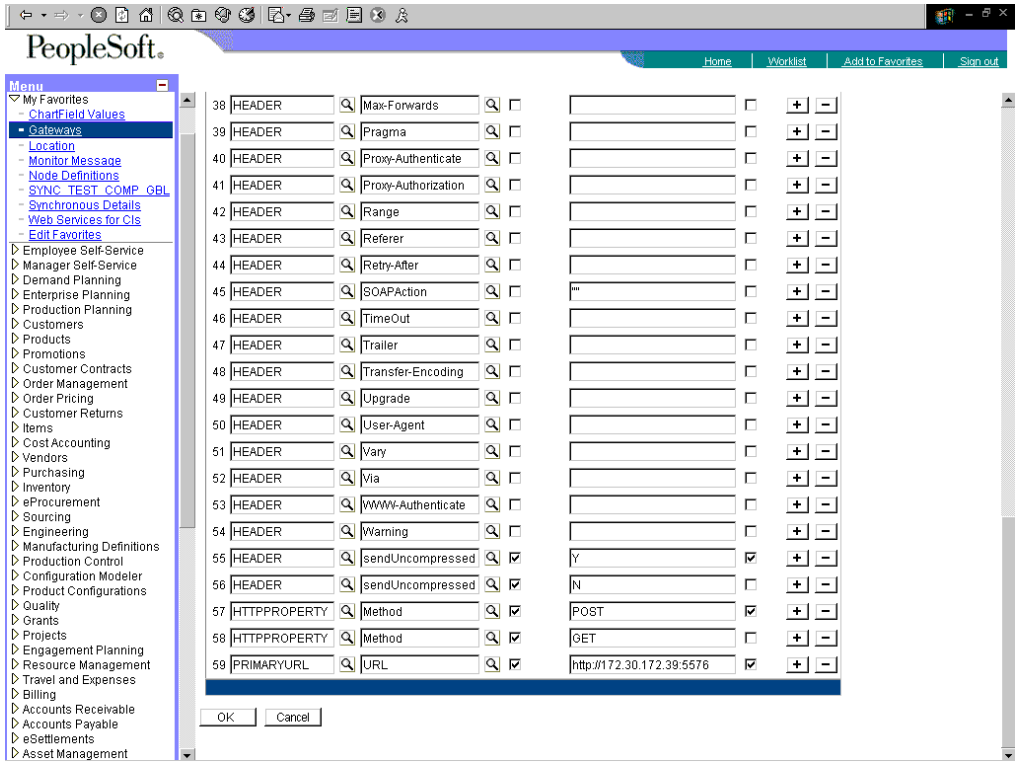

   a. If you do not see the HTTPTARGET Connector ID, click *Load*.

      If it still does not appear, your Gateway was not installed properly.

   b. Check with your PeopleSoft system administrator.

4. Click the *Properties* URL for HTTPTARGET.

The Properties pane for HTTPTARGET opens, as shown in the following image. Gateways is selected from 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*.

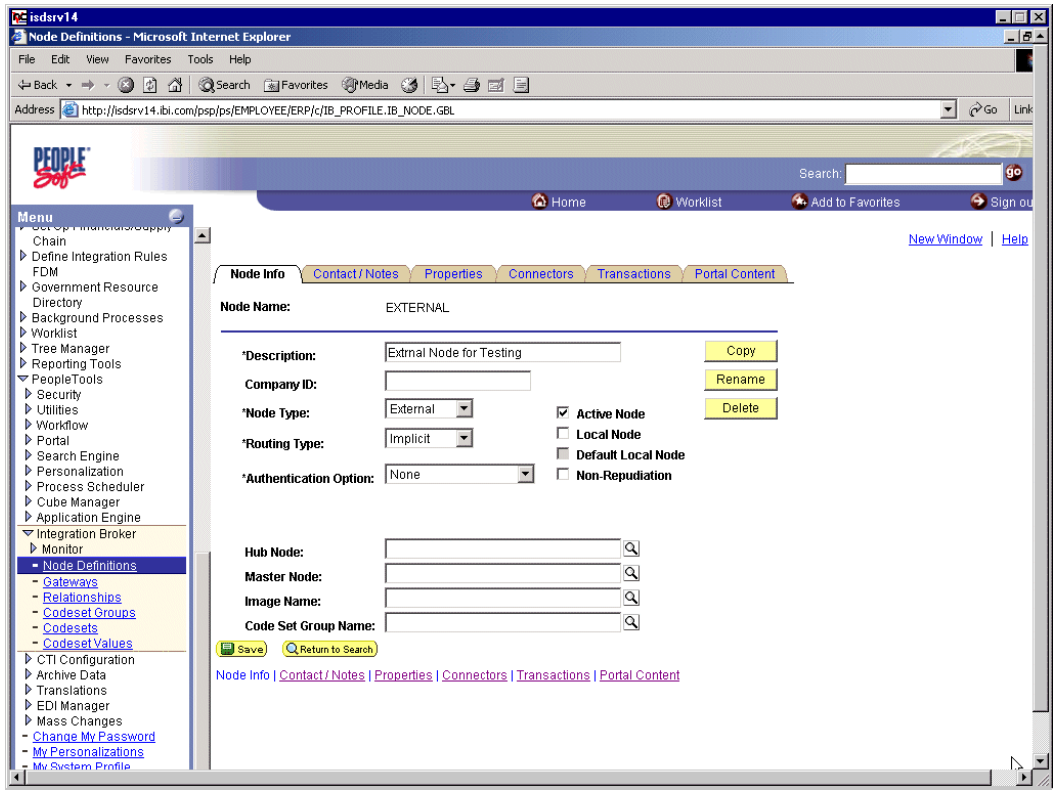
   The Gateway window opens.

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

The following image shows the Connector configuration pane for External node. It includes the following: Description field, Company ID field, Node Type list, Routing Type list, Authentication Options list, Active Node check box, Local Node check box, Default Local Node check box, Non Repudiation check box, Hub Node field, Master Node field, Image Name field, and Code Set Group Name field. It also contains Copy, Rename, and Delete buttons.




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.

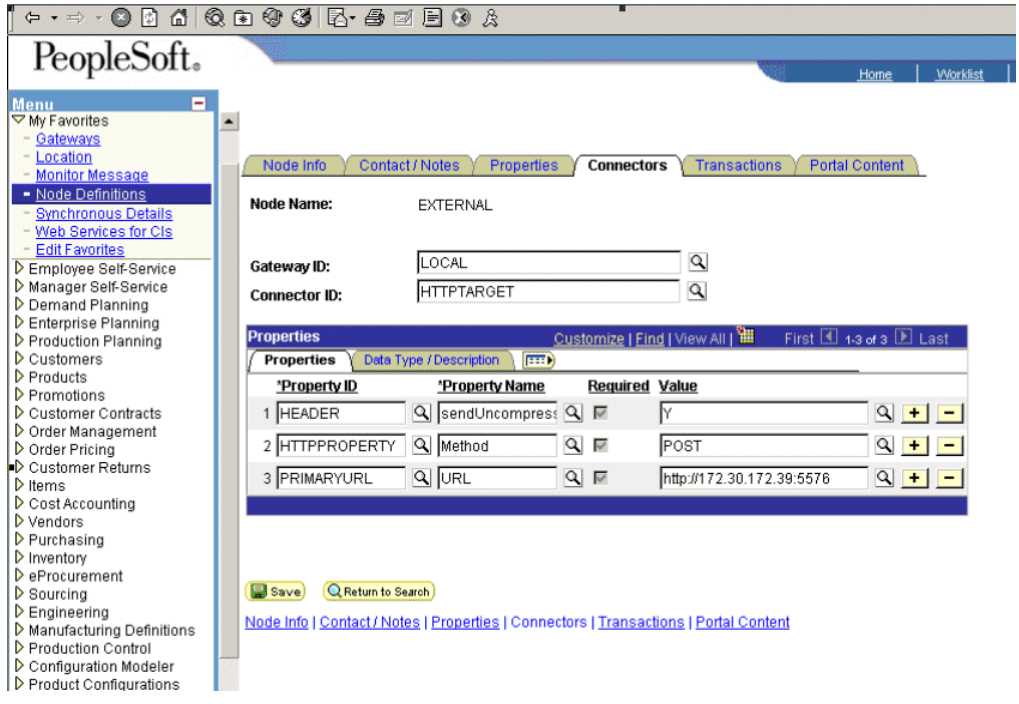
   This procedure uses a node called EXTERNAL. For more information about creating and using nodes, see Appendix D, *Using PeopleSoft 8 Integration Broker* or your PeopleSoft documentation.

   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 contains the following: Gateway ID field, Connector ID field, Property ID field, Property Name field, Required check box, Value field, and Save and Return to Search.




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

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

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

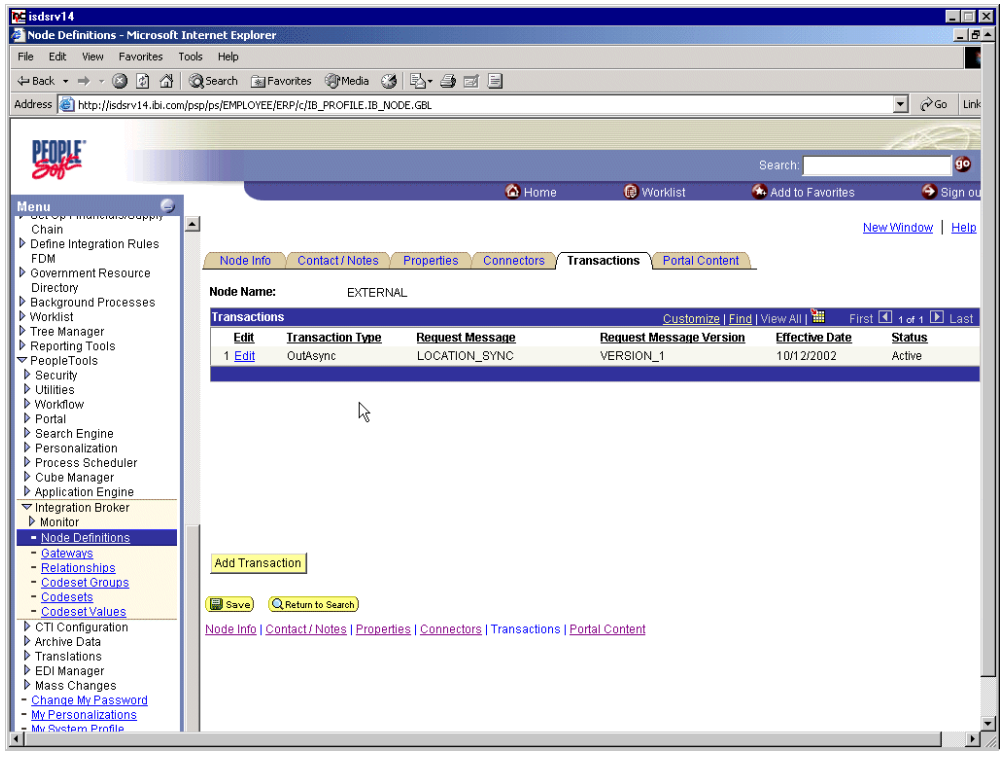
**Note:** For complex business situations you can configure multiple nodes and multiple listeners.

**c.** Click *Save*.

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

**4.** Select the *Transactions* tab.

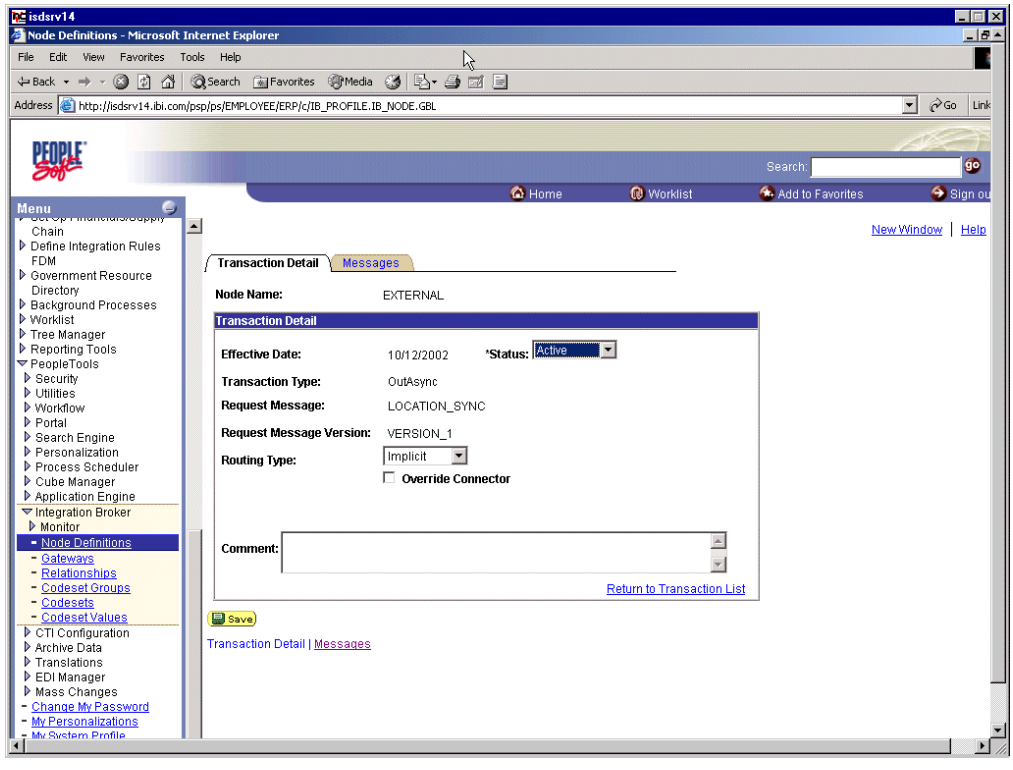
The following image shows the Transactions tab selected in PeopleSoft 8 Integration Broker. It contains Transaction information, and Add Transaction, Save, and Return to Search buttons.




**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 the following information: Effective Date, Status list, Transaction Type information, Request Message information, Request Message Version information, Routing Type list, and Comment pane.




You can add the message with which you are working.

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

**b.** Click *Save*.

**6.** Return to the *Transaction List*.

**a.** If there are other transactions, edit them.

**b.** Set the status to *Inactive*.

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

**7.** Click *Save* on the Transaction List.

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

# Configuring the TCP/IP Handler for PeopleSoft 8.1

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

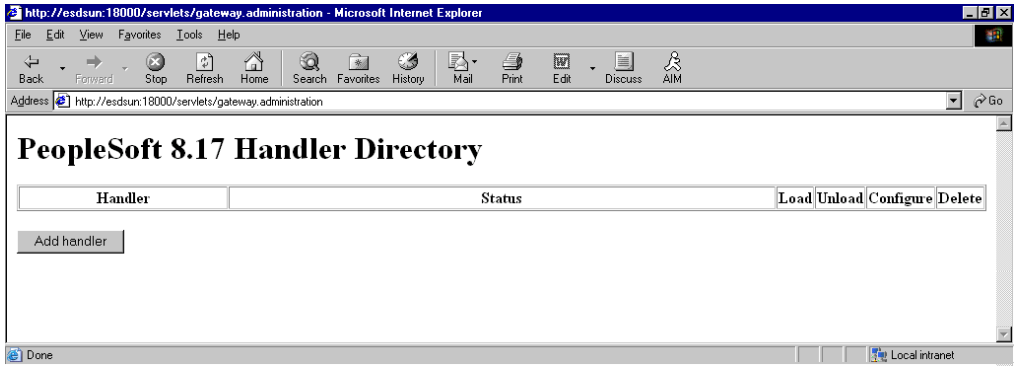
## Procedure: How to Configure the TCP/IP Handler for PeopleSoft 8.1

To configure the TCP/IP Handler for PeopleSoft 8.1to send messages to your BEA WebLogic Server:

1. In a Web browser, launch the *PeopleSoft 8.1 Gateway Configuration* servlet interface.
2. If the Simple File Handler is currently loaded, unload and delete it before proceeding.

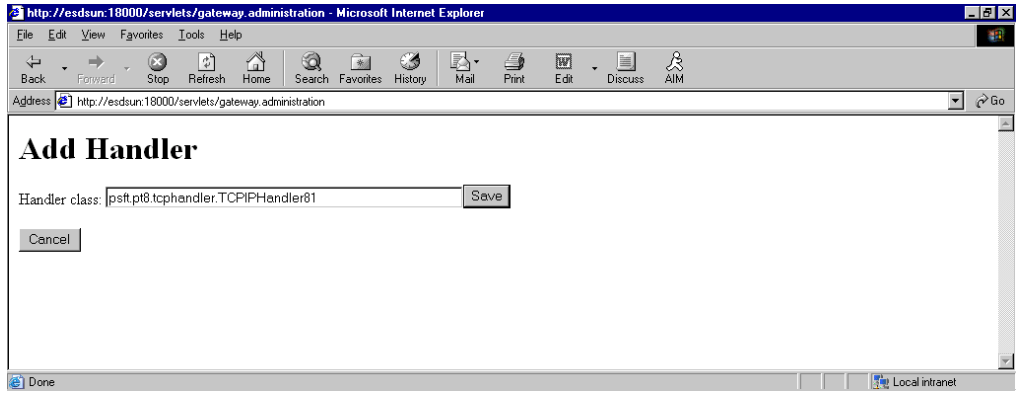
   You must see an empty Handler directory.

   The following image shows an empty PeopleSoft Handler directory with the Add handler button.



3. Click *Add handler*.

   The following image shows a loaded Add Handler directory. It contains information about the handler path and status, and Unload, Configure, and Add Handler.



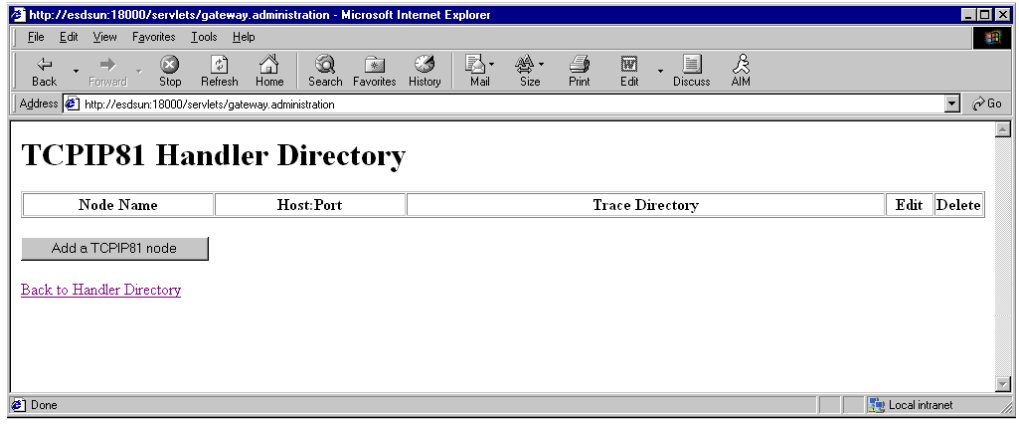

   a. Type the full path of TCPIPHandler81 (the following is case-sensitive): *psft.pt8.tcphandler.TCPIPHandler81*

   b. Click *Save*.

4. 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.




6. Click *Add a TCPIP81 node*.

The screens illustrating this procedure show a node named EXTERNAL. For more information about creating and using nodes, see Appendix D, *Using PeopleSoft 8 Integration Broker* or your PeopleSoft documentation.

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

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

The system does not validate your entries.

8. Click *Save*.
9. For your changes to take effect, click *Back to Handler Directory* to return to the PeopleSoft 8.1 Handler Directory window.
10. Click *Unload and re-Load TCPIPHandler81*.

You are now ready to send messages from PeopleSoft to your BEA WebLogic 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.

Test your configuration to ensure that:

- BEA WebLogic Server is up and running.
- The server name and/or port number for PeopleSoft and BEA WebLogic 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:**

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

   b. Select *Monitor Message*.

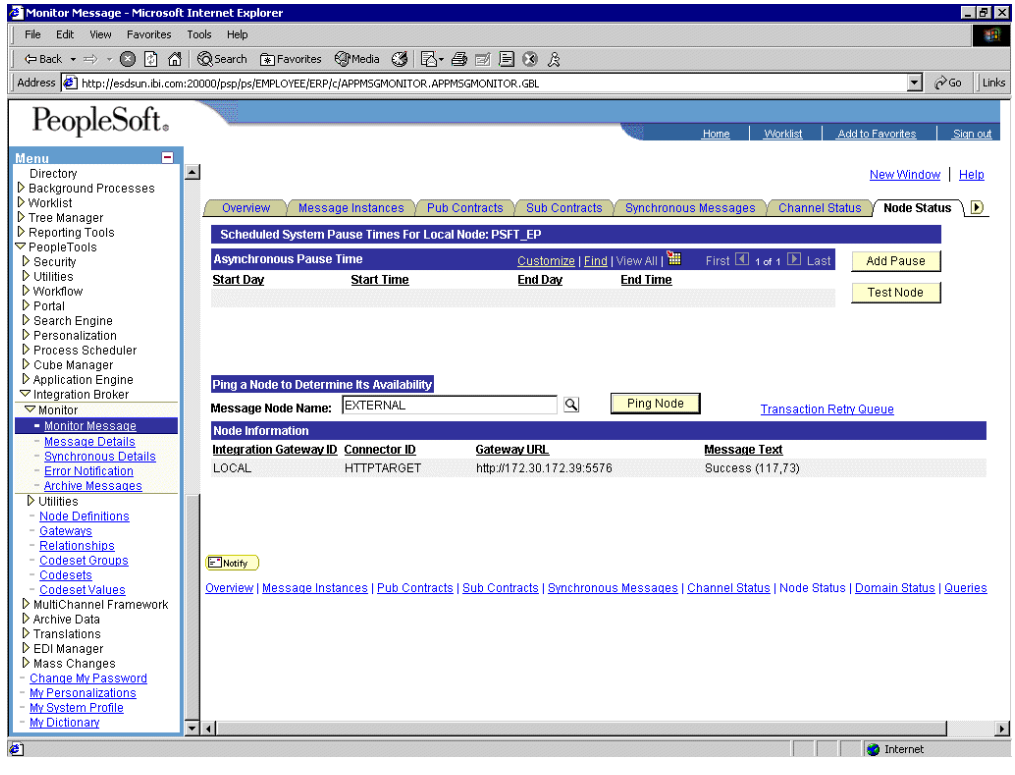
   **For PeopleSoft 8.1:**

   a. In the menu pane, expand *Home*, *PeopleTools*, *Application Message Monitor*, and *Use*.

   b. Select *Application Message Monitor*.

3. Click the *Node Status* tab.

   The following image shows the Node Status tab and contains information about scheduled system pause times for the local node, pinging a node to determine its availability, and 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 BEA WebLogic 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.

Application Explorer is a Web application running within a servlet container that is accessible through a Web browser. It is packaged as an archive located in the following directory:

```
drive:\iWay55\etc\setup\iwae.war
```

You must deploy the iwae.war file through a servlet container or J2EE application server. In addition, PeopleSoft must be installed, configured, and available for client access. Application Explorer need not reside on the same system as the application system being accessed, but network access is required.

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 WebLogic Platform 8.1*, *User Projects*, *your domain for iWay*, and then, click *Start Server*.

## 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 domain you are using for iWay.

## 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`*

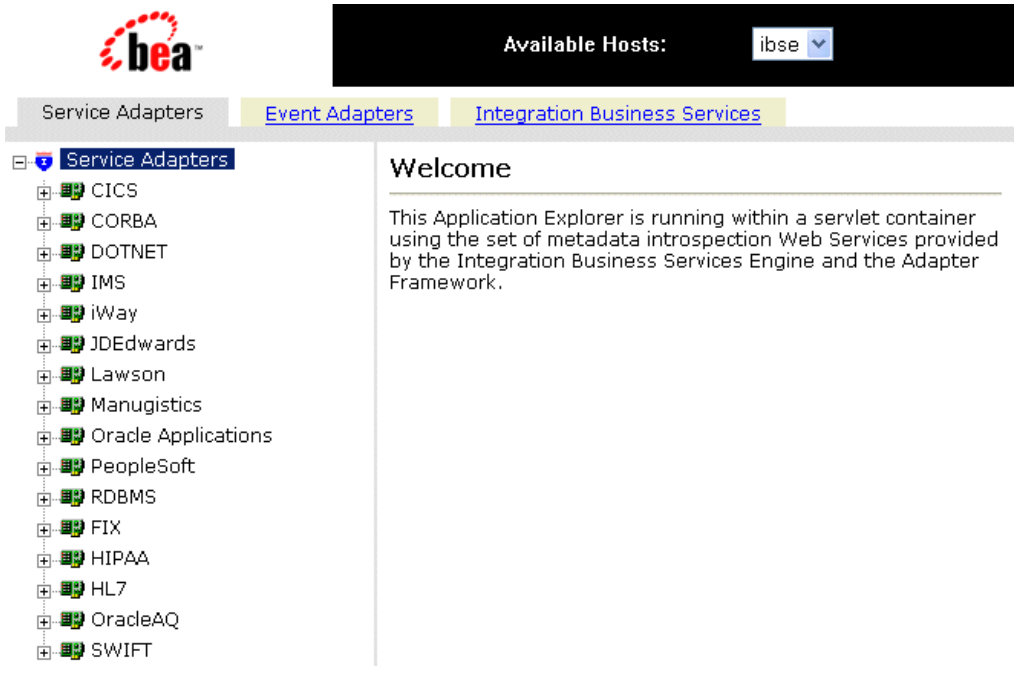
   Is the name of the machine where your application server is running.

   *`port`*

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

   The Application Explorer opens.

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



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 this 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 5-7.

## Procedure: How to Create a New Target

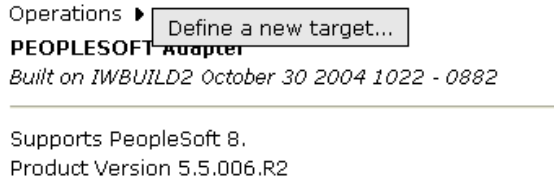
To create a new target using Application Explorer:

1. Click *Service Adapters*.

   The following image shows the Application Explorer with the PeopleSoft node selected in the left pane and the Operations menu available in the right pane.

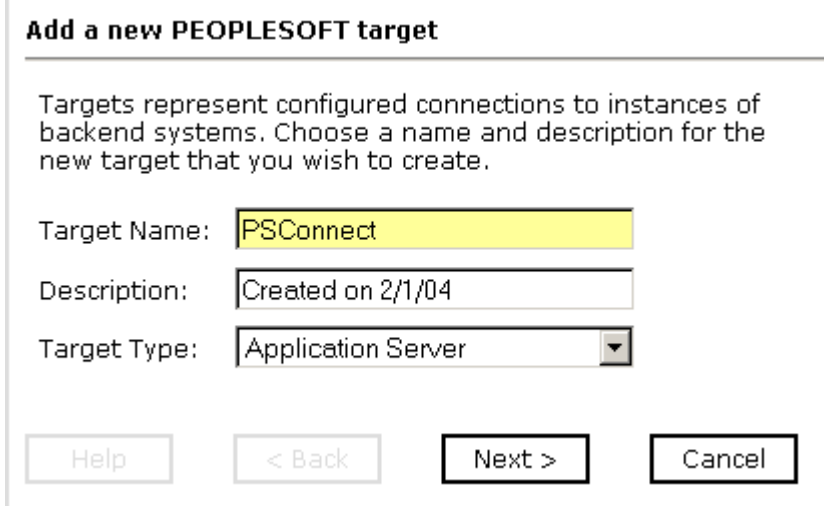

2. Click the *PeopleSoft* node.
3. Move the pointer over *Operations*.

The following image shows the Define a new target menu option that appears in the right pane, as well as title and product version information for the adapter.




**4.** Select *Define a new target*.

The following image shows the Add a new PeopleSoft target pane that opens on the right, with fields for Target Name and Description, and a Target Type drop-down list and buttons to choose whether to proceed to the next pane or to cancel the action.




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

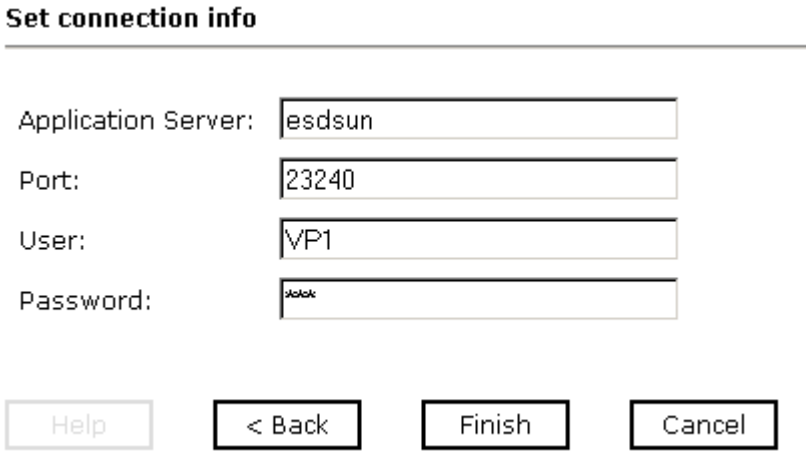
**b.** In the Description field, type a brief description for the connection.

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

The default value is Application Server.

5. Click *Next*.

   The following image shows the ports node selected in the left pane and the Set connection info pane on the right, with fields to enter the Application Server name, Port number, the User ID, and an associated password. The right pane includes buttons to click to go back to the previous screen, to finish, or to cancel the action.




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

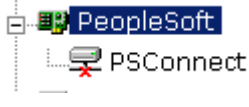
   b. In the Port field, type the port number where the PeopleSoft application is listening.

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

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

6. Click *Finish*.

   The following image shows the PeopleSoft target, PSConnect, that appears below the PeopleSoft node in the left pane.



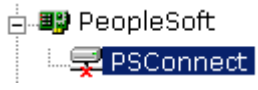

   You are ready to connect to your PeopleSoft target.

## Procedure: How to Connect to a Target

To connect to a target using Application Explorer:

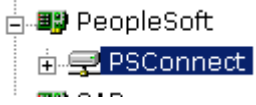
1. Expand the *PeopleSoft* node and select the target you defined, for example, PSConnect.

   The following image shows the defined but disconnected PSConnect target below the expanded PeopleSoft ports node in the left pane.




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*.

   In the left pane, the following image shows the PSConnect node changed to reflect that a connection was made.




4. Expand the *PSoftConnect* node.

   The following PeopleSoft business objects appear:

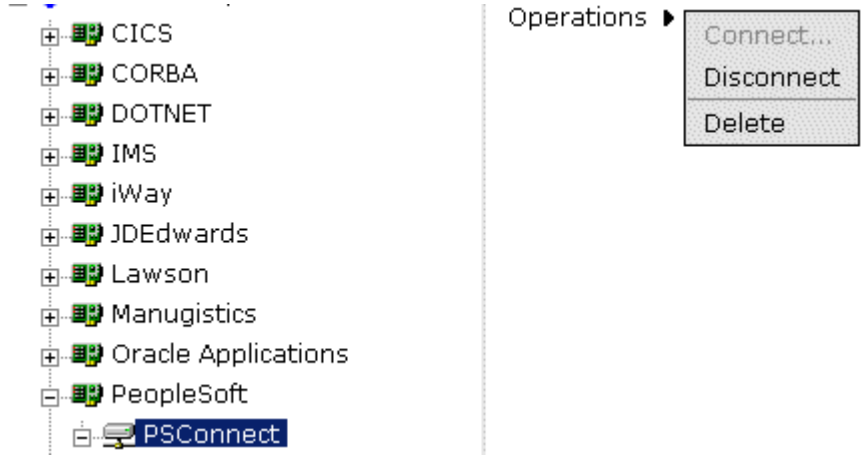
   - Component Interfaces
   - Messages
   - Component Interfaces (RPC)

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

The following image shows the connected PSConnect target below the expanded PeopleSoft ports node in the left pane and the Application Explorer with the Operations pop-up menu displayed in the right pane.




Disconnecting from the application system drops the connection, but the node remains.

The PSConnect node in the left pane 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 5-9.

Although you can maintain multiple open connections to different application systems, it is recommended to close connections when you are not using them. For information on disconnecting from a target, see *How to Disconnect From a Target* on page 5-7.

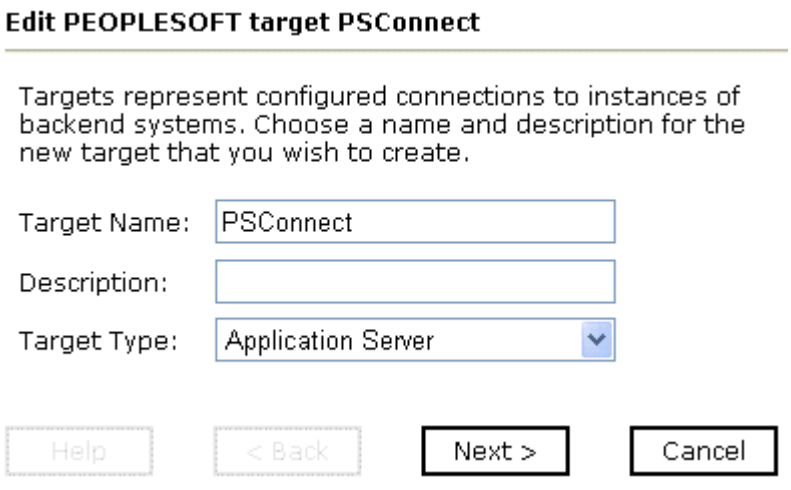
In addition to closing a target, you can delete a target that is no longer required. You can delete it whether or not it is closed. If open, the target automatically closes before it is deleted. For more information, see *How to Delete a Target* on page 5-9.

## 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 following image shows the Edit PeopleSoft pane, with fields for Target Name and Description, and a Target Type drop-down list. The right pane includes buttons to click to choose whether to proceed to the next pane or to cancel the action.



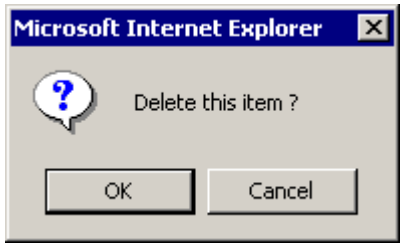

3. Modify the connection information.
4. To display additional information, click *Next*.
5. After you complete your edits in the next pane, click *Finish*.

## 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*.

The 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

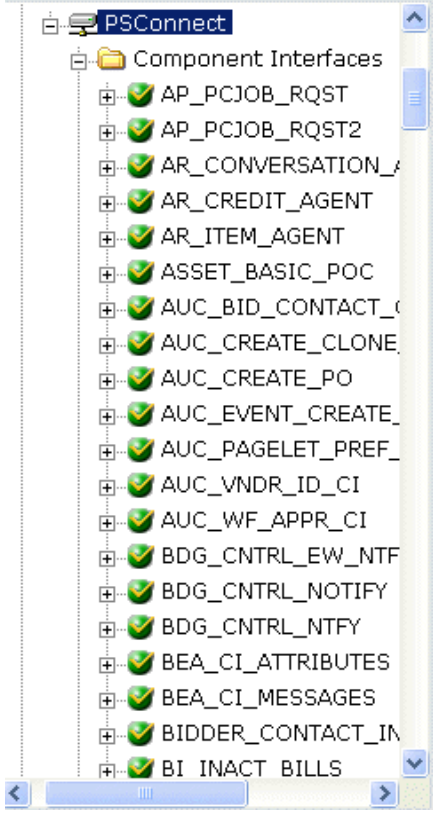
After you connect to PeopleSoft, Servlet Application Explorer enables you to explore and browse business object metadata. For example, Application Explorer enables you to view PeopleSoft Component Interface and Message metadata stored in the PeopleSoft business object repository.

## 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 home 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
```

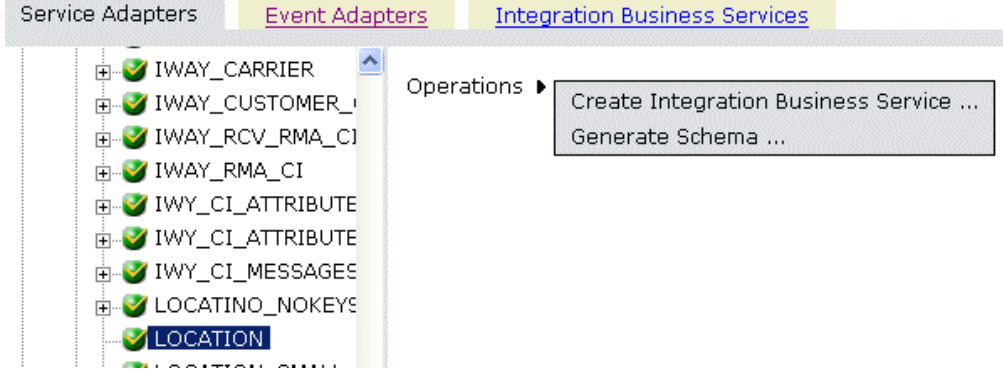
where:

*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

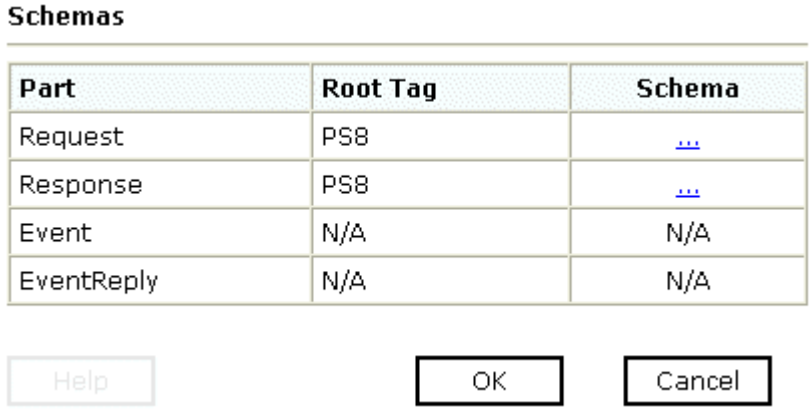
The following image shows a list of PeopleSoft Component Interfaces in the left pane. In the right pane, the Operations menu appears expanded and shows the Create Integration Business Service and Generate Schema menu options.



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 following image shows the Schemas pane on the right with a table that defines the root tag for each schema and provides associated hyperlinks.



**Schemas**

| Part | Root Tag | Schema |
|---|---|---|
| Request | PS8 | ... |
| Response | PS8 | ... |
| Event | N/A | N/A |
| EventReply | N/A | N/A |

Help OK Cancel

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

The following image shows the XML schema.


**4.** Click the *Back* button on your Web browser to return to the previous window.

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.

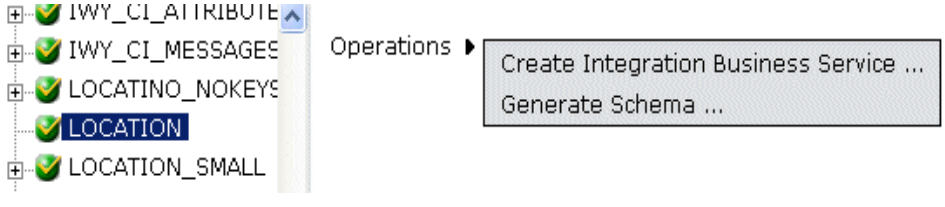
You can make Web services available to other services within a host server by generating WSDL (Web Services Description Language) from the Web service.

## 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.

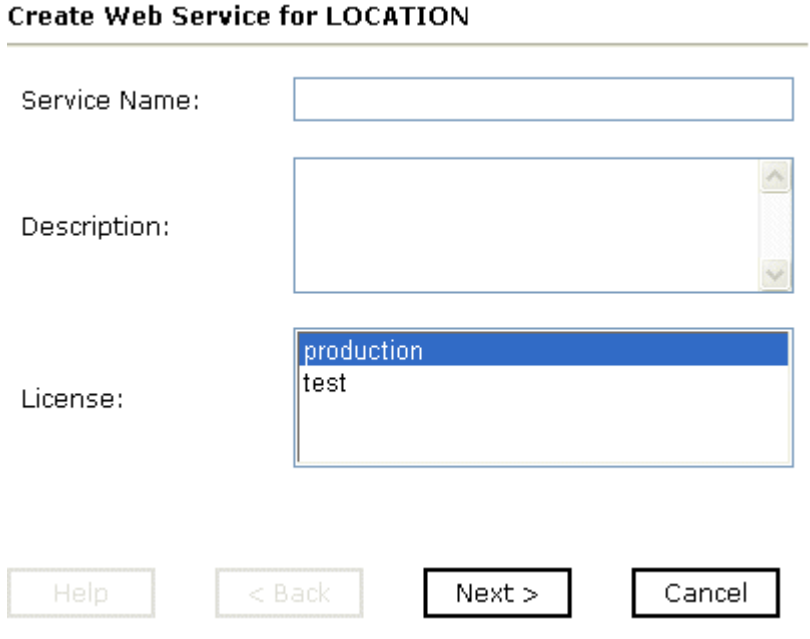


3. In the right pane, move the pointer over *Operations* and select *Create Integration Business Services*.

   The Create Web Service pane opens. The following image shows the Create Web Service for LOCATION pane in which the next step described the action you can take.


4. Select the *Create a new service* or *Use an existing service* option button and click *Next*.

   The following pane opens. The following image shows the next pane containing three text fields for entry as described in the next steps.



**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.

**c.** From the License list, select *production* or *test*.

**5.** Click *Next*.

The following image shows the next pane containing two text fields for entry as described in the next steps.


**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.

**6.** 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.

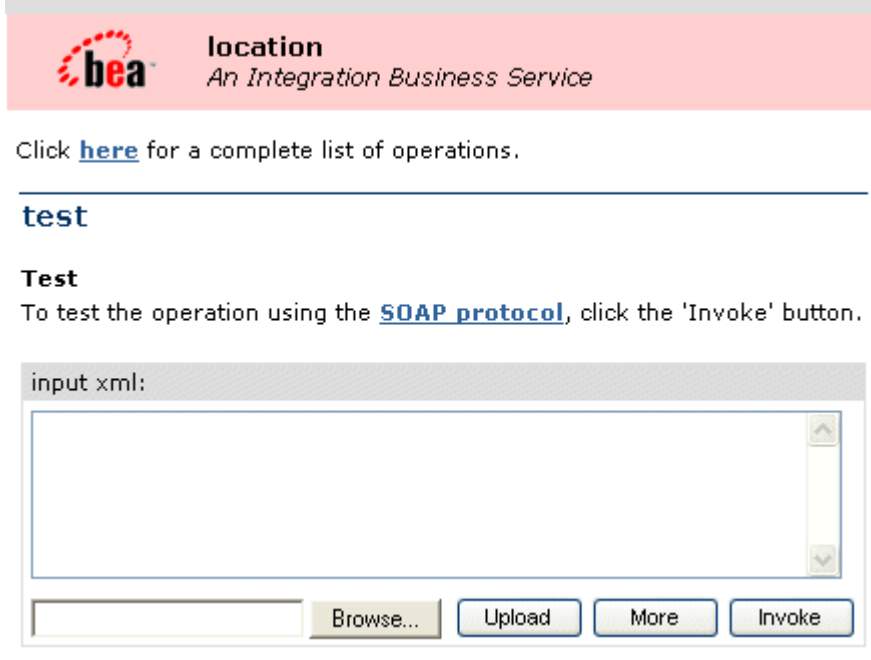
**2.** If it is not expanded, expand the list of Web services under Integration Business Services.

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.

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




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*.

**7.** Click *Invoke*.

Application Explorer displays the results in the right pane.

The following image shows a sample XML returned by iBSE.


## 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.

- **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 6-14.

# 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 the 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.
- **JMS.** The JMS queue disposition allows 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 allows 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 allows 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 6-13.

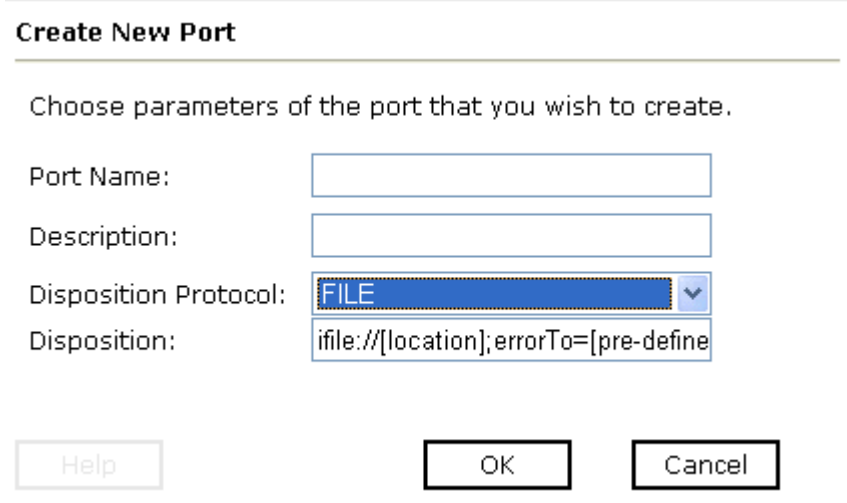
## Procedure: How to Create an Event Port for the File Disposition

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. The following image shows the Create New Port pane 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.

**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 describes the parameters for the disposition.

| Parameter | Description |
|---|---|
| location | The destination and file name of the document where event data is written, for example,<br>`ifile://D:\in\x.txt;errorTo=ifile://D:\error.` |
| errorTo | Predefined port name or another disposition URL where error logs are sent. |

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 6-14.

## 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 Point Application Explorer to 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. Follow the steps in *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.
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 6-14.

## Procedure: How to Point Application Explorer to a JCA Deployment

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

1. Follow the steps in *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.
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 6-14.

## 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.

   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 defines 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. A predefined port name or another full URL. Optional. |
| errorTo | Location where error documents are sent. A 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 6-14.

## 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.

   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 defines 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. A 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 6-14.

## 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.

   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 defines 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:<br>`http://localhost:7001/ibse/IBSEServlet/test/webservice.ibs?wsdl`<br>where:<br>*`webservice`*<br>Is the name of the Web service you created using Application Explorer.<br>This value can be found by navigating to the Integration Business Services tab and opening the *Service Description* hyperlink in a new window. The WSDL URL appears in the Address field.<br>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 | The method that will be called by the SOAP disposition. For example:<br>*webservice.method@test@@*<br>where:<br>*webservice*<br>Is the name of the Web service you created using Application Explorer.<br>*method*<br>Is the method being used.<br>*test*<br>Is the license that is being used by the Web service.<br>This value can be found by navigating to the Integration Business Services tab and opening the *Service Description* hyperlink in a new window. Perform a search for *soapAction*.<br>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 | The location to which responses are posted, which can be a predefined port name or another URL. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |
| errorTo | The location to which error logs are sent. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

**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 6-14.

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

To create a port for a JMS queuedisposition 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.

   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 defines 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:<br>`javax.jms.QueueConnectionFactory` |
| jndi_url | The URL to use to contact the JNDI provider. The syntax of this URL depends on which JNDI provider is being used. This value corresponds to the standard JNDI property:<br>`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:<br>`weblogic.jndi.WLInitialContextFactory.` |
| user | A user ID associated with this queue. |
| password | The password for this user ID. |
| errorTo | The location where error logs are sent. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

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 6-14.

## 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.

   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=[channnel
   name];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 defines 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 an 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. A 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 6-14.

## 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*.

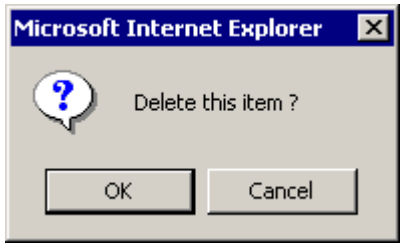
   The Edit Port pane opens on the right.
3. Make the required changes to the event port configuration fields.
4. Click *OK*.

## 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*.

The 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.

# 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 6-14. For information on editing a channel, see *How to Edit a Channel* on page 6-23. 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 6-17.
- **File.** For more information, see *How to Create a File Channel* on page 6-18.
- **MQSeries.** For more information, see *How to Create an MQSeries Channel* on page 6-20.

## Procedure: How to Create a Channel

To create a channel using Application Explorer:

1. Click the *Event Adapters* tab.

   The Event Adapters pane opens.

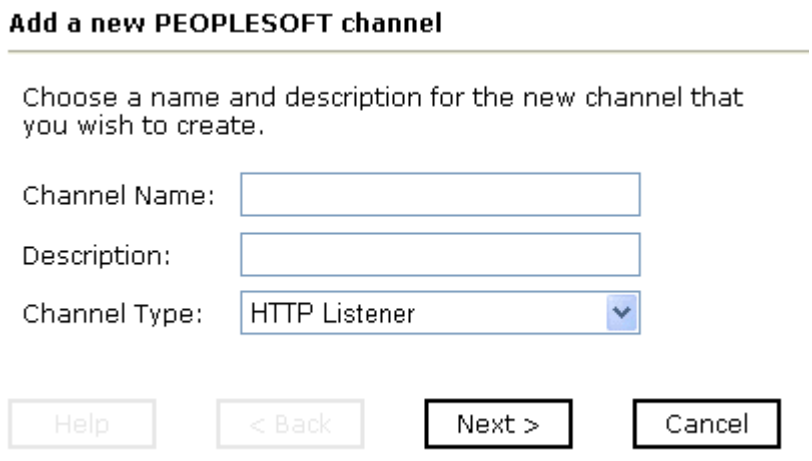
   a. In the left pane, expand the *PeopleSoft* node.

      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. The following image shows the Add a new PEOPLESOFT channel pane where you add a new channel by following the steps described next.



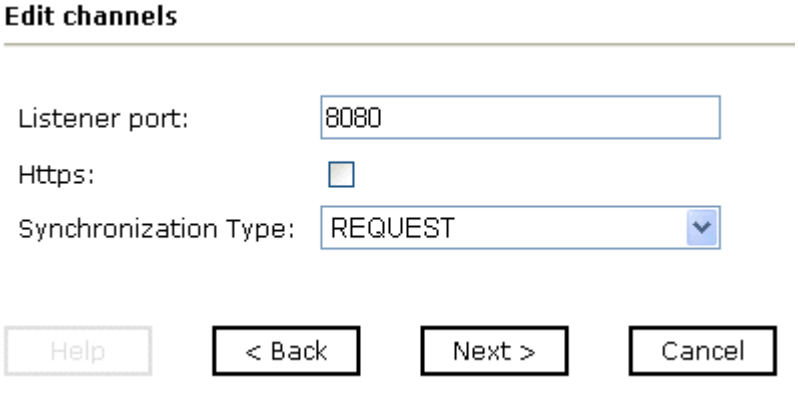

**a.** In the Channel Name field, type a name, for example, TEST_CHANNEL.

**b.** In the Description field, type a brief description.

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

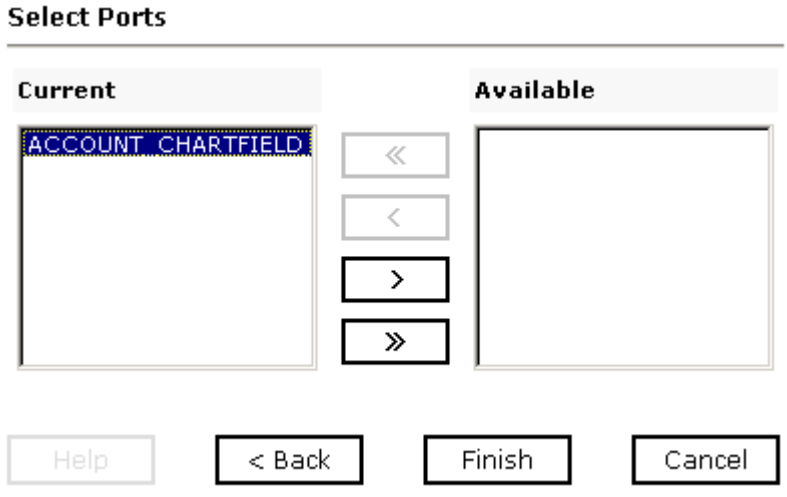
**3.** Click *Next*.

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



4. Enter the information that is specific to your PeopleSoft system and the channel you are creating.
5. Click *Next*.

   The Select Ports pane opens. The following image shows the Select Ports pane with two list boxes where you select the port by clicking the arrow buttons to move from the Current list box to the Available list box.



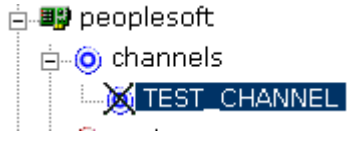

   a. Select an event port from the list of current ports.
   b. To transfer the port to the list of available ports, click the single right arrow button. To associate all event ports, click the double right arrow button.

6. 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.

   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.

   The following image shows the newly created channel with an X over its icon.



You must start the channel to activate your event configuration.

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 an HTTP Channel

To create an HTTP channel using Application Explorer:

1. Click the *Event Adapters* tab.

   The Event Adapters pane opens.

   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.

   c. From the drop-down list, select *HTTP Listener*.

3. Click *Next*.

   The Edit Channels pane opens on the right.

4. Provide the values based on the description in the following table:

| Parameter | Description |
|---|---|
| www root | Root directory where PeopleSoft event data is posted. |
| Listener port | Port on which to listen for PeopleSoft event data. |
| Synchronization Type | Choose from three synchronization options:<br>• REQUEST<br>• REQUEST_RESPONSE<br>• 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*.

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

**8.** To stop the channel, move the pointer over *Operations* and select *Stop the channel*.

## Procedure: How to Create a File Channel

To create a File channel using Application Explorer:

**1.** Click the *Event Adapters* tab.

The Event Adapters pane opens.

**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.

**c.** From the drop-down list, select *File Listener*.

**3.** Click *Next*.

The Edit Channels pane opens with three tabs in the right pane.

**a.** In the Request tab, enter values for the parameters as described in the following table.

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

**b.** In the Response tab, enter values for the parameters as described in the following table.

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

**c.** In the Advanced tab, enter values for the parameters as described 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. The default is 3 seconds. |
| Processing Mode | • Sequential indicates single processing of requests.<br>• 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*.

The channel becomes active. The X over the icon disappears.

**7.** To stop the channel, move the pointer over *Operations* and select *Stop the channel*.

## Procedure: How to Create an MQSeries Channel

To create an MQSeries channel using Application Explorer:

**1.** Click the *Event Adapters* tab.

The Event Adapters pane opens.

**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.

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

**3.** Click *Next*.

The Edit Channels window opens with three tabs in the right pane.

**a.** In the Request tab, enter values for the parameters as described in the following table.

| Parameter | Description |
|---|---|
| Queue manager name | Name of the queue manager to which the server must connect. |

| Parameter | Description |
|---|---|
| MQ server host for MQClient operation | The host on which the MQ Server is located (MQ Client only). |
| MQ server port for MQClient operation | The number to connect to an MQ Server queue manager (MQ client only). |
| MQ server channel for MQClient operation | The number to connect to an MQ Server queue manager (MQ client only). |
| Document type XML | Accept the default selection. |
| Request queue name | Queue where the message is routed and where request documents are received. The name of the queue is case-sensitive and conforms to the following format:<br>`Host\queue type$\qName`<br>where:<br>`Host`<br>Is the machine name where the MQSeries queuing system is running.<br>`queue type`<br>Is the type of queue. Private queues are queues that are not published in Active Directory and appear only on the local computer where they reside. Private queues are accessible only by Message Queuing applications that recognize the full path name or format name of the queue. For private queues, enter *Private$*.<br>`qName`<br>Is the name of the queue where messages are placed, for example,<br>`iwaykxc1\Private$\peoplesoft` |

**b.** In the Response tab, enter values for the parameters as described in the following table.

| Parameter | Description |
|---|---|
| Synchronization Type | Choose from three synchronization options:<br>• REQUEST<br>• REQUEST_RESPONSE<br>• REQUEST_ACK |

**c.** In the Advanced tab, enter values for the parameters as described in the following table.

| Parameter | Description |
|---|---|
| Error Directory | Directory to which documents with errors are written. |
| Message wait interval (msec): | Interval (in milliseconds) when to check for new input. The default is 3 seconds. Optional. |
| Mode of operation | • Sequential indicates single processing of requests.<br>• Threaded indicates processing of multiple requests simultaneously. |
| Thread limit | If you selected threaded processing, indicate the maximum number of requests that can be processed simultaneously. |

**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*.

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

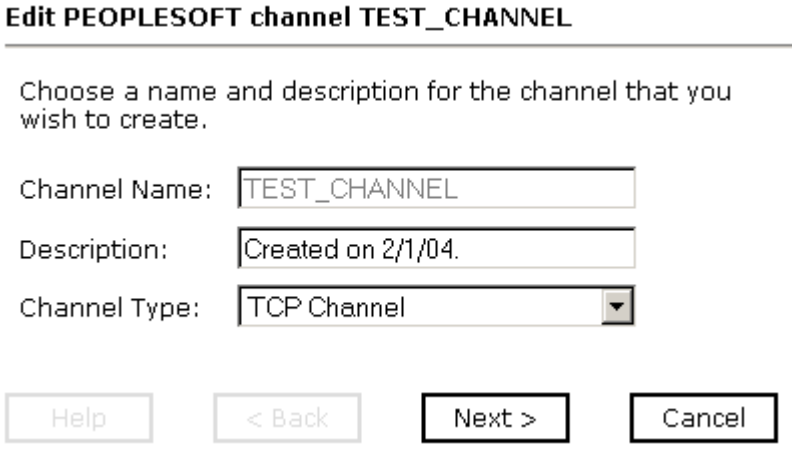
7. To stop the channel, move the pointer over *Operations* and select *Stop the channel*.

## 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. The following image shows the Edit PEOPLESOFT channel TEST_CHANNEL pane which contains current information for Channel Name, Description, and Channel Type.



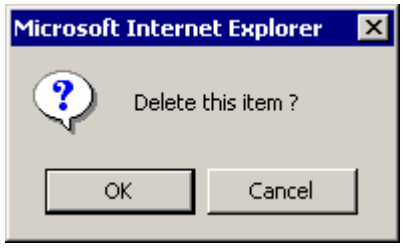

3. After you make the required changes to the channel configuration fields, click *Finish*.

## 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*.

The 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

# Using Web Services Policy-Based Security

**Topics:**

- Integration Business Services Policy-Based Security
- Configuring Integration Business Services Policy-Based Security

Servlet Application Explorer provides a security feature called Integration Business Services policy-based security. The following topics describe how this feature works and how to configure it.

**Note:** For the iWay 5.5 RG2 Release, it is recommended that policy-based security not be enabled.

# Integration Business Services Policy-Based Security

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

Servlet Application Explorer controls the use of business services that use adapters with a feature called policy-based security. This feature enables an administrator to apply *policies* to Integration Business Services (iBS) to deny or permit their execution.

A *policy* is a set of privileges associated with the execution of a business service that can be applied to an existing or new iBS. When you assign specific rights or privileges inside a policy, you need not recreate privileges for every iBS that has security issues in common with other Integration Business Services. Instead, you can use one policy for many Integration Business Services.

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

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

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

When a policy is not applied, the default value for an iBS is to “grant all.” For example, anyone can execute the iBS until the Resource Execution policy is associated to the iBS. At that time, only users granted execution permission, or those who do not belong to a group that was denied execution permissions, have access to the iBS.

# Configuring Integration Business Services Policy-Based Security

Before you create instances of policies, you must have a minimum of one user or one group to associate to an instance. You can create users and groups using Servlet Application Explorer. For more information, see *How to Create a User to Associate With a Policy* on page 7-3 or *How to Create a Group to Associate With a Policy* on page 7-5.

An execution policy governs who can execute the business service to which the policy is applied. For more information, see *How to Create an Execution Policy* on page 7-8.

You configure the IP and Domain Restriction policy type slightly differently from other policy types. The IP and Domain Restriction policy type controls connection access to iBSE and therefore, need not be applied to an individual business service. You need not create a policy, however, you must enable the Security Policy option in Servlet Application Explorer. For more information, see *How to Configure IP and Domain Restrictions* on page 7-11.

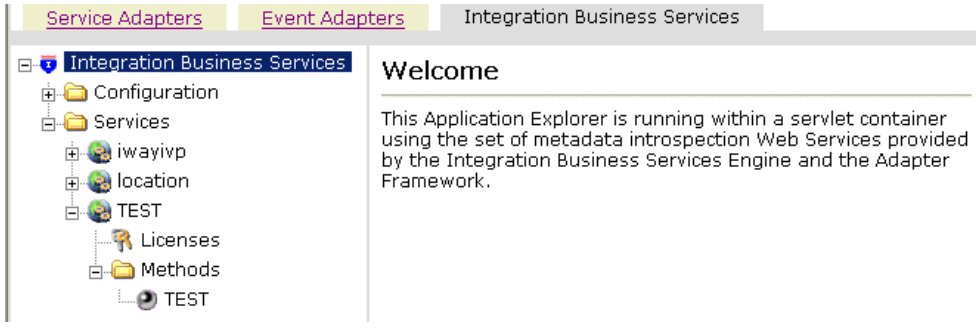
**Note:** For the iWay 5.5 RG2 Release, it is recommended that policy-based security not be enabled.

## Procedure: How to Create a User to Associate With a Policy

To create a user to associate with a policy:

1. Open *Servlet Application Explorer.*

   The following image shows the window that opens and includes three tabs corresponding to Service Adapters, Event Adapters, and Integration Business Services. The Integration Business Services tab is active and displays a Welcome screen on the right and the Integration Business Services node expanded on the left.




   a. Click the *Integration Business Services* tab.

   b. Expand the *Configuration* node.

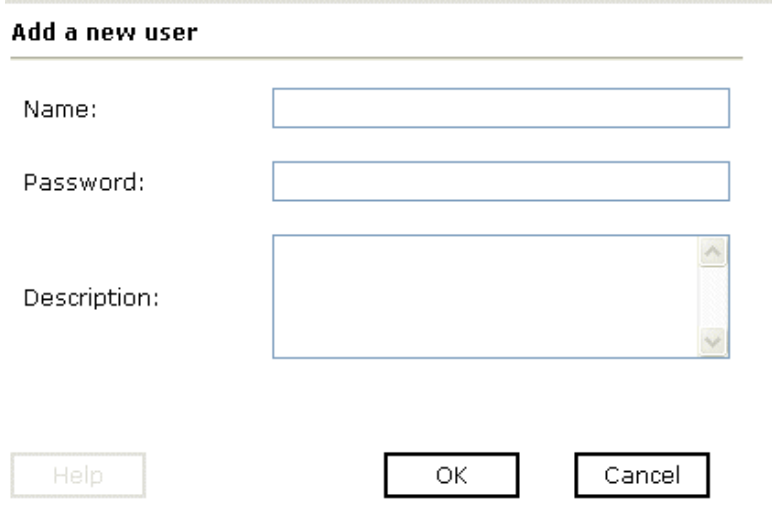
   c. Expand the *Security* node.

   d. Expand the *Users and Groups* node.

**e.** Select *Users*.

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

The following image shows the Add a new user pane that opens and includes fields where you enter a user name, a password, and a description of the user. The pane includes a Help button, an OK button to instruct the system to accept inputs, and a Cancel button to escape from the pane.



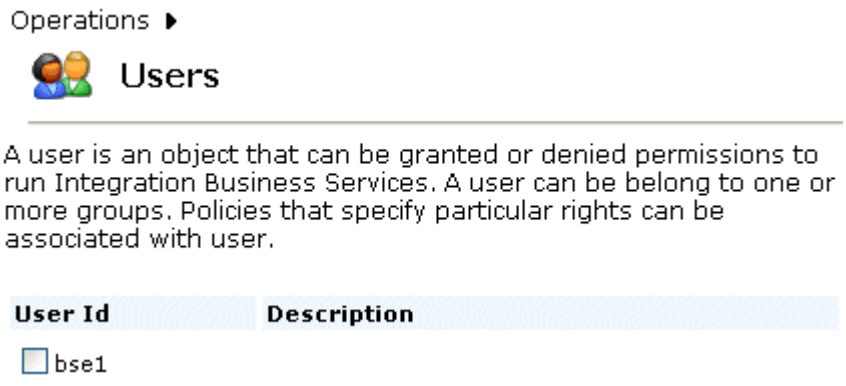

**a.** In the Name field, type a user ID.

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

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

**3.** Click *OK*.

The following image opens and shows a new user added to the configuration. It includes a definition of a user and a user ID and description.

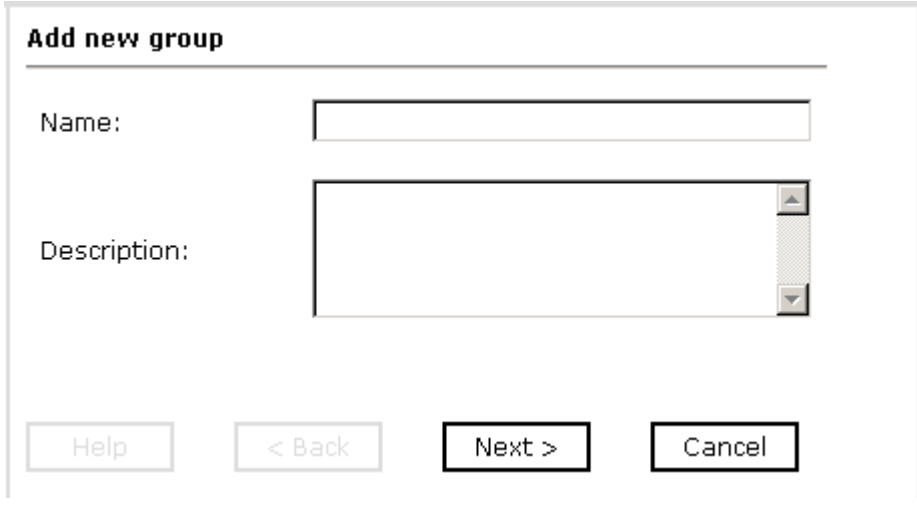



## Procedure: How to Create a Group to Associate With a Policy

To create a group to associate with a policy:

1. Open *Servlet Application Explorer*.
   a. Click the *Integration Business Services* tab.
   b. Expand the *Configuration* node.
   c. Expand the *Security* node.
   d. Expand the *Users and Groups* node.
   e. Select *Groups*.
2. In the right pane, move the pointer over *Operations* and click *Add*.

The following image shows the Add new group pane that opens with fields where you enter a name and a description for the group. To continue after typing inputs, click the Next button. The pane also includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.

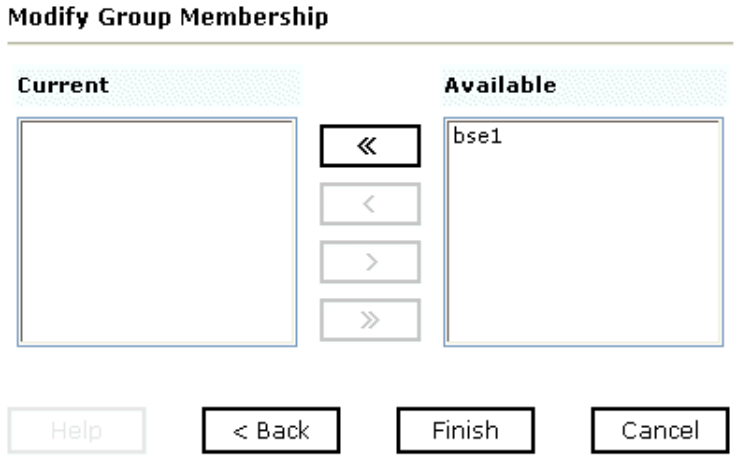



**a.** In the Name field, type a a name for the group.

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

**3.** Click *Next*.

The following image shows the Modify Group Membership pane where you can move users to or from a group using the arrow keys to move them between the Current and Available lists and then clicking the Finish button. The pane includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.

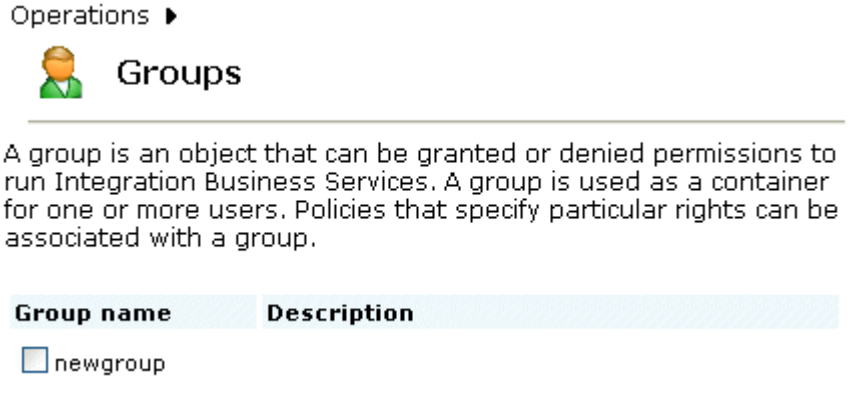



You can either highlight a single user in the list of available users and add it to the current list by clicking the left arrow, or you can click the double left arrow to add all users in the list of available users to the group.

4. After you select a minimum of one user, click *Finish*.

   The new group is added.

   The following image shows a pane with a new group added to the configuration. It includes a definition of a group and the group name and description.



## Procedure: How to Create an Execution Policy

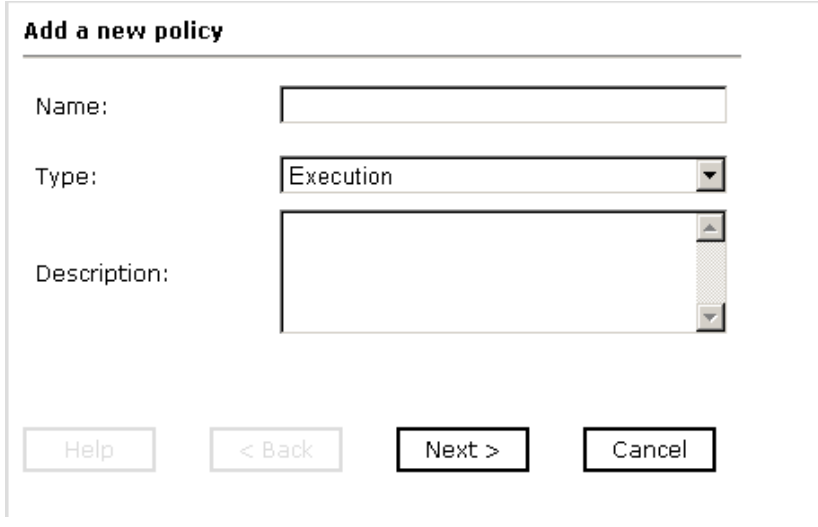
To create an execution policy:

1. Open *Servlet Application Explorer*.
   a. Click the *Integration Business Services* tab.
   b. Expand the *Configuration* node.
   c. Select *Policies*.

   The following image shows the Policies pane on the right where you apply a policy. The Operations menu displays three options, Build/Rebuild, Add, and Refresh.


2. Move the pointer over *Operations* and click *Add*.

The following image shows the Add a new policy pane that opens with fields for entering the name, type, and description of the policy. To continue, click the Next button. The pane includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.



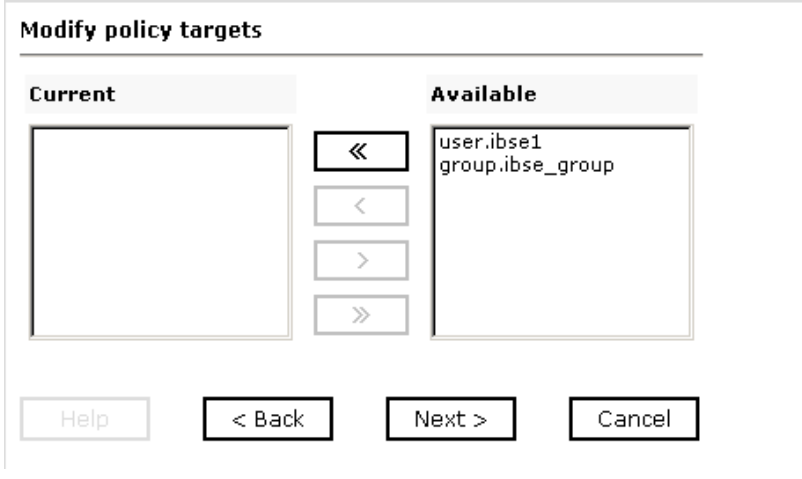

**a.** In the Name field, type a a name for the policy.

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

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

**3.** Click *Next*.

The following image shows the Modify policy targets pane that opens and includes a list of current and available targets and arrow buttons to move targets from one list to the other. The pane also includes a Help button, a Back button to return to the previous screen, a Next button to continue to the next screen, and a Cancel button to escape from the pane.

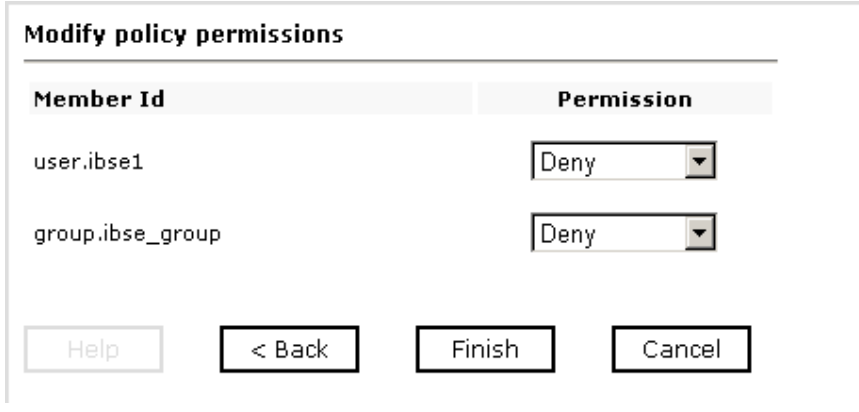



4. Select a minimum of one user or group from the Available pane.

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

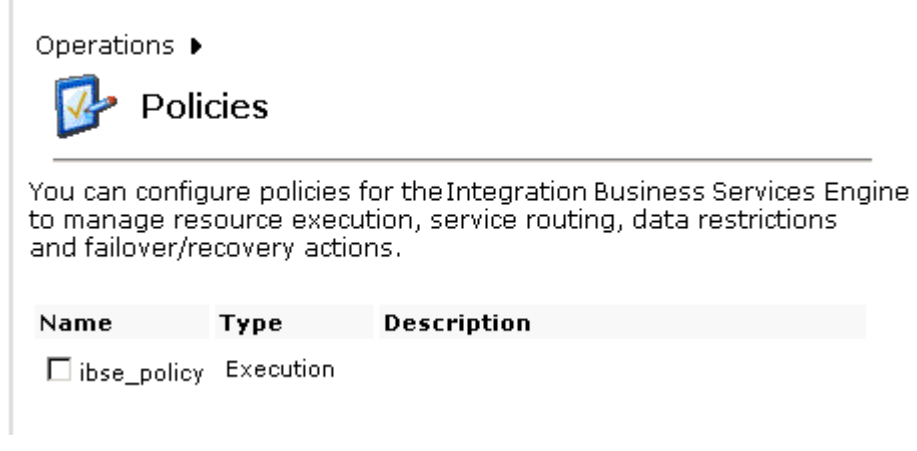
5. Click *Next*.

The following image shows the Modify policy permissions pane that opens and includes drop-down lists where you can select to grant or deny permission to members and then click a button to finish. The pane also includes a Help button, a Back button to return to the previous screen, and a Cancel button to escape from the pane.




6. To assign whether users or groups may execute the iBSE, select *Grant* to permit execution or *Deny* to restrict execution from a Permission drop-down list.
7. Click *Finish*.

   The following image shows the pane that summarizes your configuration. It includes a definition of policies and the name, type, and description of the policies.




## Procedure: How to Configure IP and Domain Restrictions

To configure IP and domain restrictions:

1. Open *Servlet Application Explorer.*

**a.** Select the *Integration Business Services* tab.

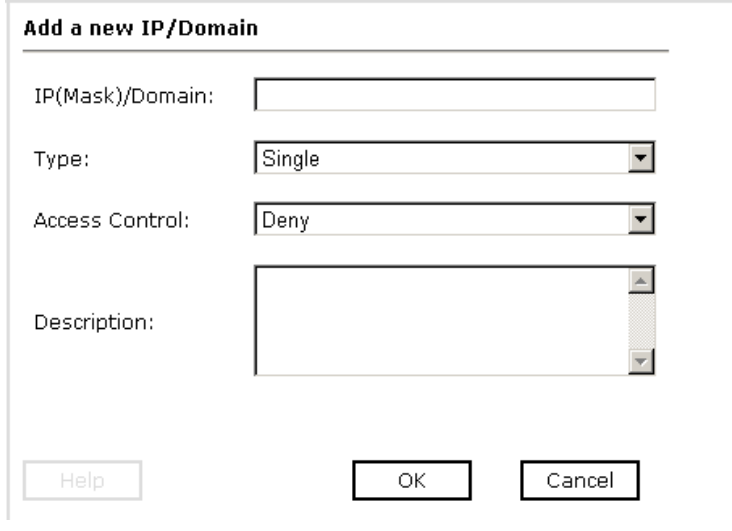
**b.** Expand the *Configuration* node.

**c.** Expand the *Security* node.

**d.** Select *IP and Domain*.

**2.** In the right pane, move the pointer over *Operations* and click *Add*.

The following image shows the Add a new IP/Domain pane that opens where you enter information for the IP/Domain in four fields. You must select a type of restriction from a drop-down list before you can enter information in the IP(Mask)/Domain field. The pane also includes a Help button, an OK button to instruct the system to accept inputs, and a Cancel button to escape from the pane.




**a.** From the Type drop-down list, select the type of restriction.

**b.** In the IP(Mask)/Domain field, type the IP or domain name using the following guidelines.

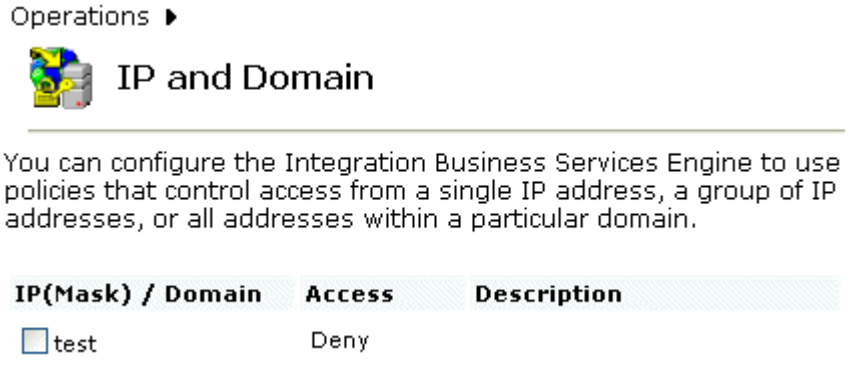
If you select Single (Computer) from the Type drop-down list, you must provide the IP address for that computer. If you only know the DNS name for the computer, click *DNS Lookup* to obtain the IP Address based on the DNS name.

If you select Group (of Computers), you must provide the IP address and subnet mask for the computer group.

If you select Domain, you must provide the domain name, for example, yahoo.com.

3. From the Access Control drop-down list, select *Grant* to permit access or *Deny* to restrict access for the IP addresses and domain names you are adding.
4. Click *OK*.

   The following image shows the pane that opens and summarizes your configuration including the domain name, whether access is granted or denied, and a description (optional).



CHAPTER 8

# Troubleshooting

**Topics:**

- Troubleshooting
- Integration Business Services Engine Error Messages

This section explains the limitations and workarounds when connecting to PeopleSoft.

The adapter-specific errors listed in this appendix can arise whether using the adapter with a JCA or with an iBSE configuration.

# Troubleshooting

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

- Application Explorer
- PeopleSoft
- JCA
- iBSE

**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.

## Reference: Application Explorer

The following table describes the errors and its corresponding solutions for Application Explorer.

| Error | Solution |
|---|---|
| Cannot connect to the BEA WebLogic Adapter for PeopleSoft from Application Explorer. | Ensure that:<br>• PeopleSoft is running.<br>• The PeopleSoft user ID and password is correct.<br>• The port number is correct.<br>• The custom Component Interface is properly installed. |
| The following error message appears:<br>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. |

| **Error** | **Solution** |
|---|---|
| PeopleSoft does not appear in the Application Explorer Adapter node list. | Ensure that the PeopleSoft JAR files, iwpsci84.jar and psjoa.jar, are added to the lib directory. |
| The following error message appears:<br>Jolt Session Pool cannot provide a connection to the appserver. This appears to be because there is no available application server domain. [Fri Aug 27 13:06:27 EDT 2004] bea.jolt.ServiceException: Invalid Session | The host name or port number for PeopleSoft is incorrect. |
| Properties are not displayed for a Component Interface. | You are using the wrong iwpsci8x.jar file. |
| Cannot generate schemas. | If the error message "Index: -1, Size:0" appears, or if you can log on to Application Explorer but you cannot see any Component Interfaces or Messages, then you may have both the iwpsci81.jar and iwpsci84.jar files in your lib directory. Stop your server, remove the unrequired jar file, and restart your server. |
| Pstools.properties file has not been initialized. | This file is required for PeopleSoft 8.1 release, if customer is using PeopleSoft 8.1, then they should add this file. This error message may appear, even if you are not using PeopleSoft 8.1, in this case ignore this error message. |

## Reference: PeopleSoft

The following table describes the errors and its corresponding solutions for PeopleSoft.

| Error | Solution |
|---|---|
| Services are not working properly when using the PeopleSoft Component Interface testing tool in three-tier mode. | To test properly using the Component Interface testing tool:<br>Open Application Designer.<br>Select the Component Interface.<br>Use the test tool.<br>If service works in test tool, then review the XML and check for redundant fields in XML. |
| The following error message appears:<br>Jolt Session Pool cannot provide a connection to the appserver. This appears to be because there is no available application server domain. [Fri Aug 27 13:06:27 EDT 2004] bea.jolt.ServiceException: Invalid Session | The host name or port number for PeopleSoft is incorrect. |
| Component Interfaces and Messages do not appear in the adapter tree. | The project is not installed properly on the PeopleSoft system. |

## Reference: JCA

The following table describes the errors and its corresponding solutions for JCA.

| Error | Solution |
|---|---|
| In Application Explorer, the following error message appears when you attempt to connect to a JCA configuration:<br>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. |

# Integration Business Services Engine Error Messages

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

## General Error Handling in iBSE

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 Integration Business Service. If an error occurs, how it is exposed depends on the adapter and the API or interfaces that the adapter uses. A few scenarios cause the SOAP gateway to generate a SOAP fault. In general, anytime the SOAP agent inside 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 this example, iBSE did not receive an element in the SOAP request message that is mandatory for the WSDL for this business 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. Since adapters use the target system interfaces and APIs, whether or not an exception is raised depends on how the target systems interface or API treats the error condition. If a SOAP request message is passed to an adapter by the SOAP agent in iBSE, and that request is invalid based on the WSDL for that service, the adapter may raise an exception yielding a SOAP fault.

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

## BEA WebLogic Adapter for PeopleSoft Invalid SOAP Request

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

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

## Empty Result From PeopleSoft Request

When the BEA WebLogic Adapter for PeopleSoft executes a Component Interface as a business service using input parameters passed in the SOAP request that do not match records in PeopleSoft, the following SOAP response is generated.

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

## Failure to Connect to PeopleSoft

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

```
<?xml version="1.0" enco
ding="ISO-8859-1" ?>
<SOAP-ENV:Envelope
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
<SOAP-ENV:Body>
  <SOAP-ENV:Fault>
   <faultcode>SOAP-ENV:Server</faultcode>
  <faultstring>java.lang.Exception: Error Logon to PeopleSoft
   System<faultstring>
 </SOAP-ENV:Fault>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

## Invalid SOAP Request

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

## Empty Result From an Adapter Request

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

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

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

CHAPTER 9

# 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 Servlet Application Explorer, you can use managing and monitoring tools provided by the Integration Business Services Engine (iBSE) and the iWay Connector for JCA to measure the performance of 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 to manage and monitor services and events currently in use and to display 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://localhost:port/ibse/IBSEConfig`

   where:

   *`localhost`*

   Is the machine where the application server is running.

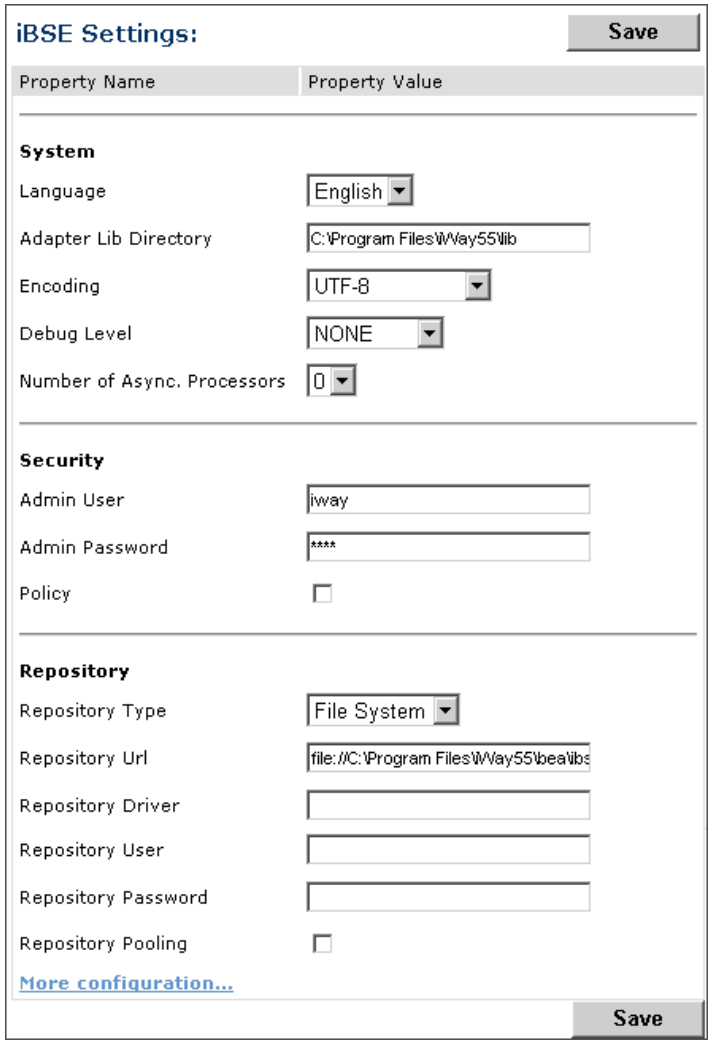
   *`port`*

   Is the HTTP port for the application server.

   The following image shows the iBSE Settings window that opens. It lists property names and includes fields where you can enter values for each property. To configure system settings, the System pane contains 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 adapters 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 click 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://localhost:port/ibse/IBSEStatus`

where:

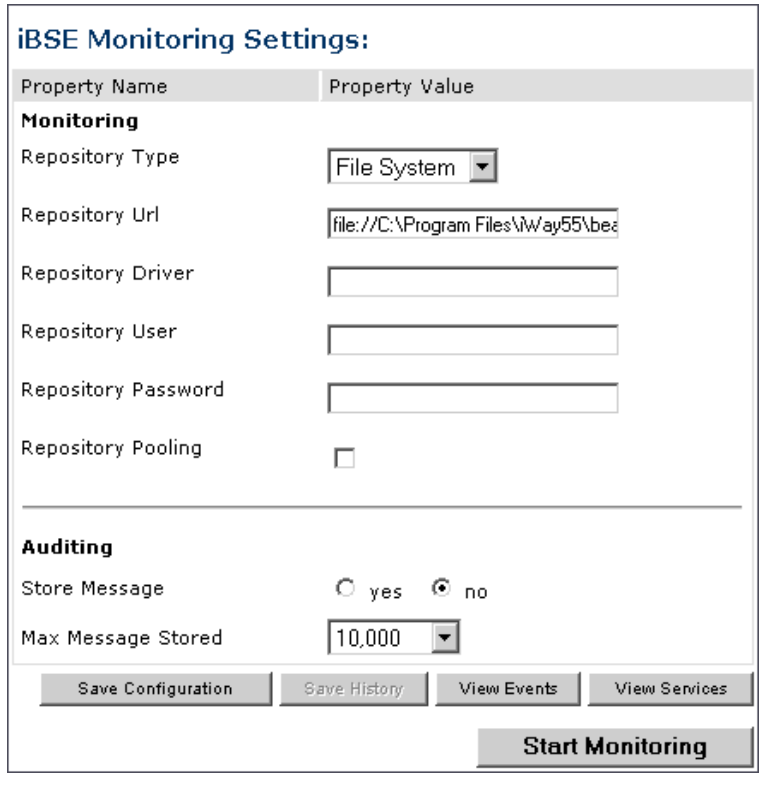
*localhost*

Is the machine where the application server is running.

*port*

Is the HTTP port for the application server.

The following image shows the iBSE Monitoring Settings window that opens. It lists property names and includes a corresponding field where you can enter values for each property. The Monitoring 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. The Auditing pane contains an option button to click to specify whether to store a message and a drop-down list where you can select the maximum messages to store. At the bottom of the window is a row of buttons that you can click to save your configuration, view events, or view services. The Save History button is inactive. After you enter properties and choose whether to save or view, you can click the Start Monitoring button.



**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, click the *Repository Pooling* check box.

**f.** In the Auditing pane, select *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.** Select the maximum number of messages you want 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 need 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

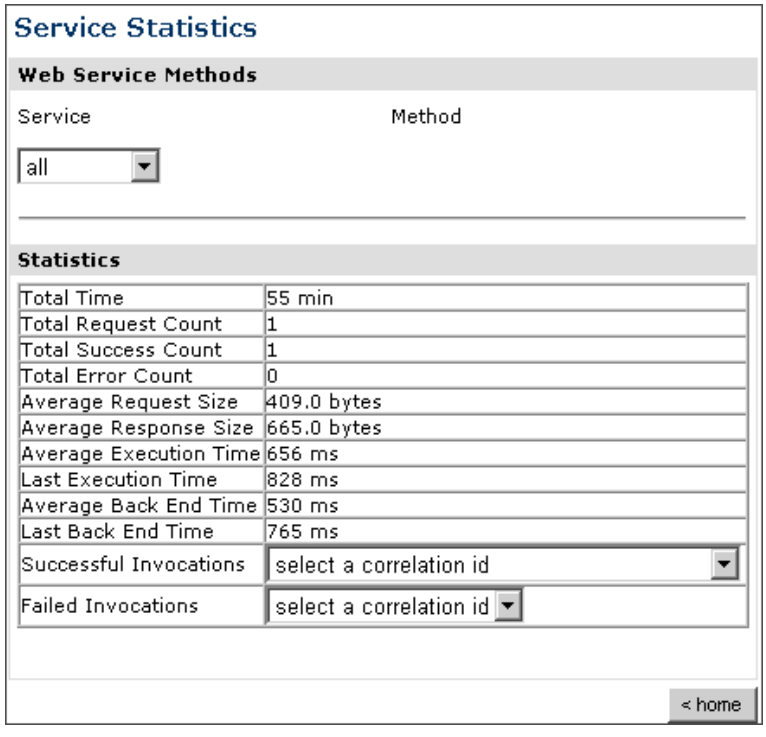
To monitor services:

**1.** Ensure that your BEA WebLogic Server is started.

**2.** From the iBSE Monitoring Settings window, click *Start Monitoring*.

**3.** Click *View Services*.

The following image shows the System Level Summary (Service Statistics) window that opens. 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.

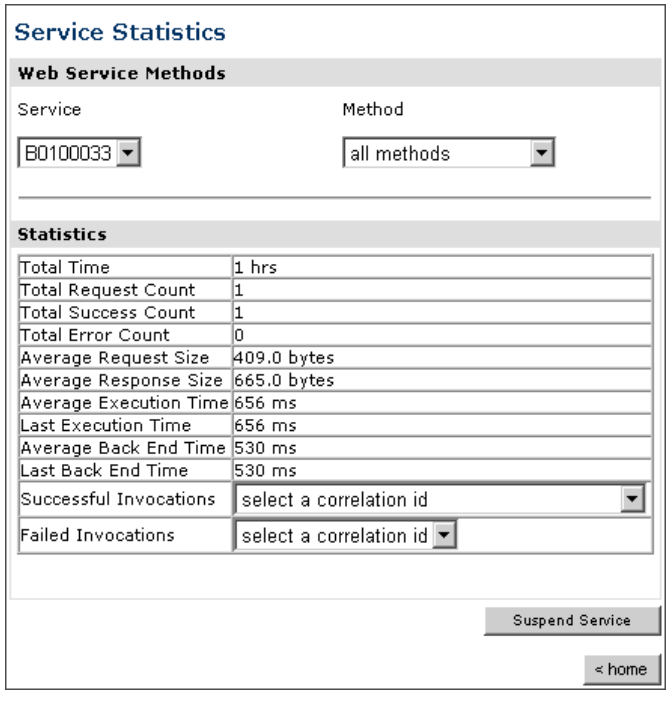
The following table consists of two columns, one that lists the name of each statistic and the other that describes the corresponding service statistic.

| 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 that were made during the monitoring session. |

| Statistic | Description |
|---|---|
| Total Success Count | Total number of successful service executions. |
| Total Error Count | Total number of errors that were encountered. |
| Average Request Size | Average size of an available service request. |
| Average Response Size | Average size of an available service response size. |
| 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 | A list of successful services arranged by correlation ID. To retrieve more information for a service, you can select the service from the drop-down list. |
| Failed Invocations | A list of failed services arranged by correlation ID. To retrieve more information for a service, you can select the service from the drop-down list. |

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

The following image shows the System Level Summary (Service Statistics) window that opens. 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 (also located in the lower right).

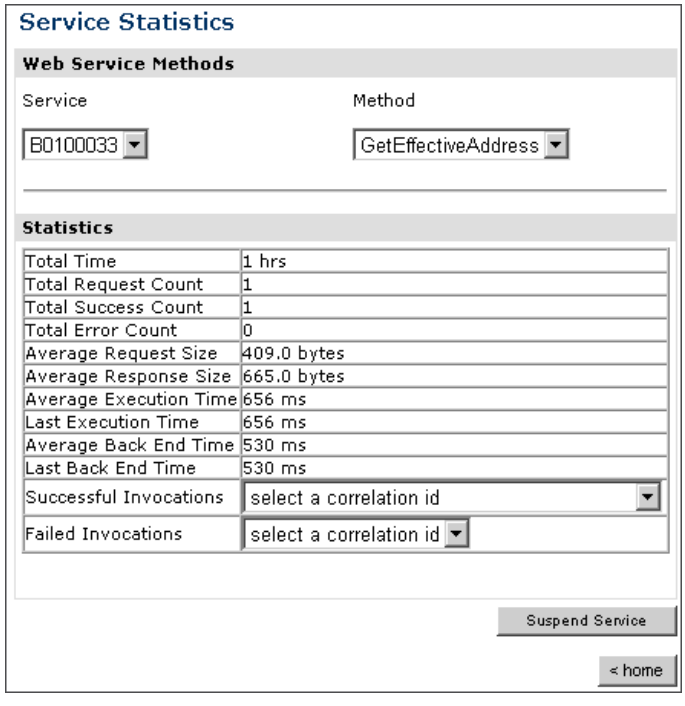



**a.** To stop a service at any time, click *Suspend Service*.

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

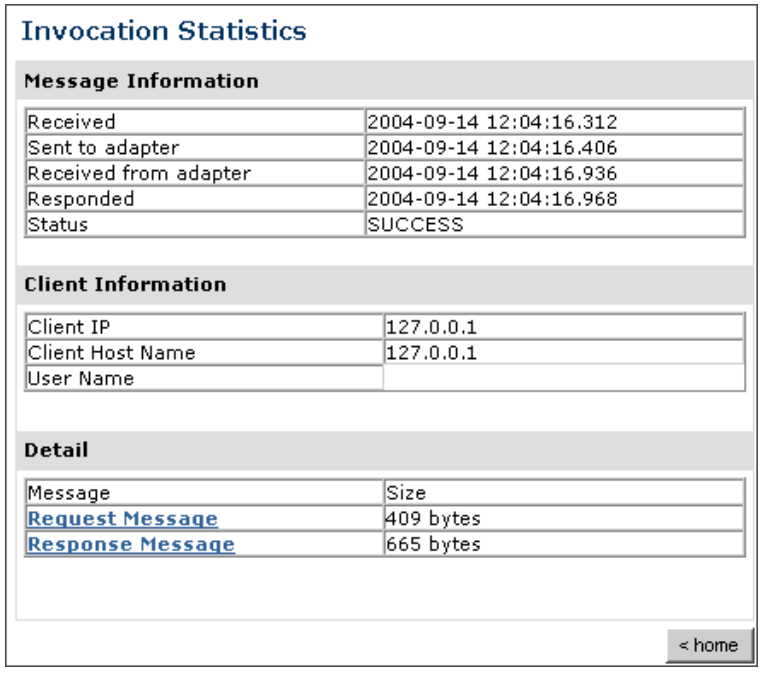
**5.** Select a method for the service from the Method drop-down list.

The following image shows the Method Level Summary (Service Statistics) window that opens. 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 (also located in the lower right).




**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 following image shows the Invocation Level Statistics window that opens. 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, with options to click to view the respective XML documents. In the lower right of the window is a home button to click to return to the iBSE Monitoring Settings window.




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.

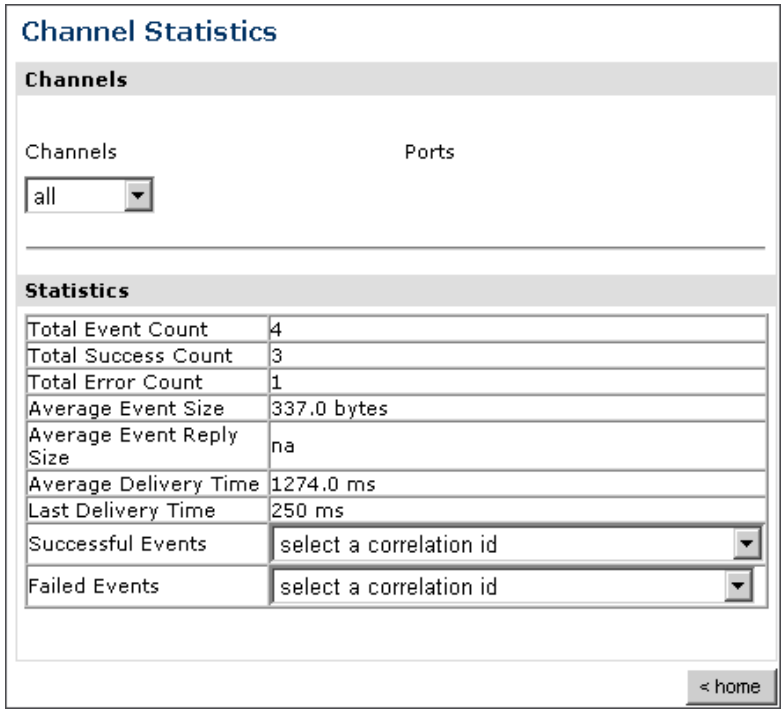
8. To return to the iBSE Monitoring Settings window, click *home*.

## 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 following image shows the System Level Summary (Channel Statistics) window that opens. 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 to click to return to the iBSE Monitoring Settings window.




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

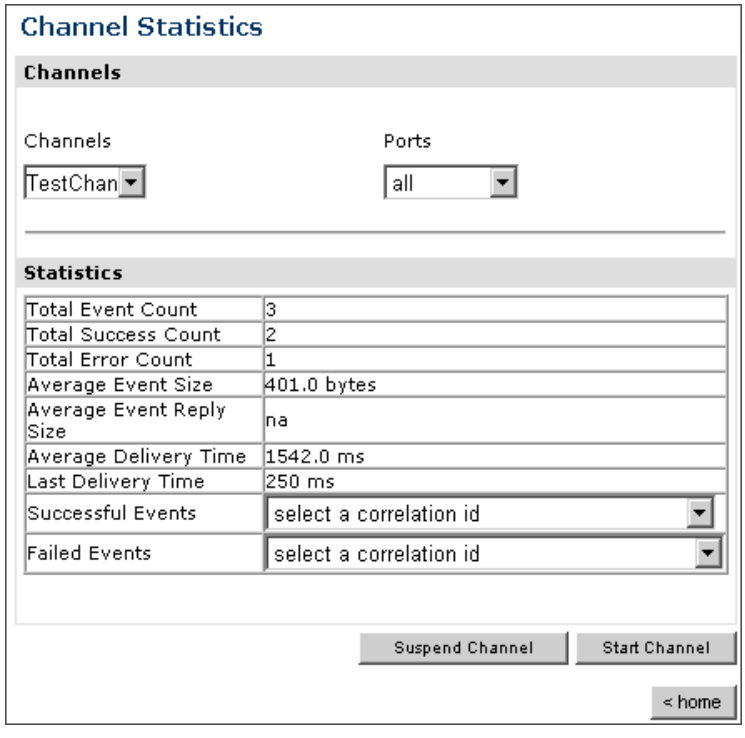
The following table consists of two columns, one that lists the name of each statistic and the other that describes the corresponding event statistic.

| 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 that were encountered. |
| Average Event Size | Average size of an available event request. |

| Statistic | Description |
|---|---|
| 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 | List of successful events arranged by correlation ID. To retrieve more information for an event, select the event from the drop-down list. |
| Failed Events | List of 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 following image shows the Channel Level Event Summary (Channel Statistics) window that opens. 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.

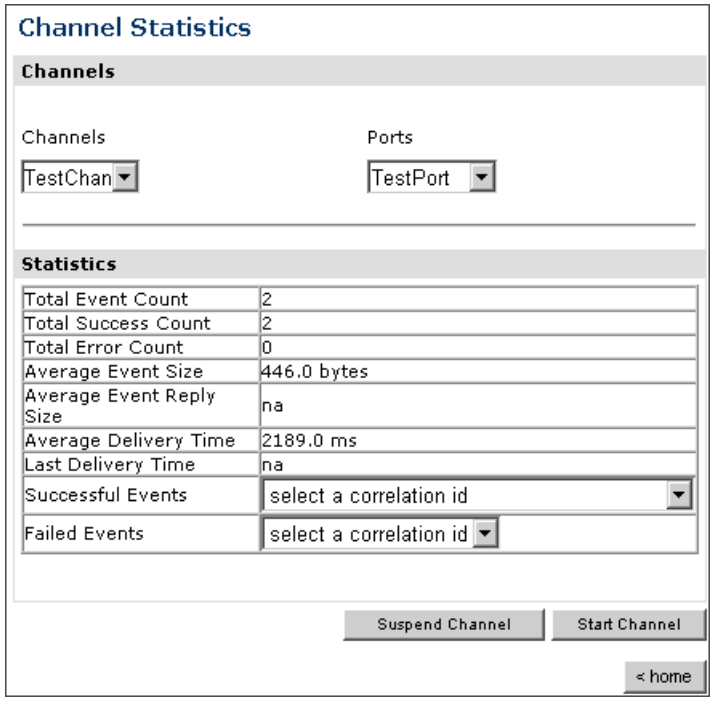



**a.** To stop a channel at any time, click *Suspend Channel*.

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

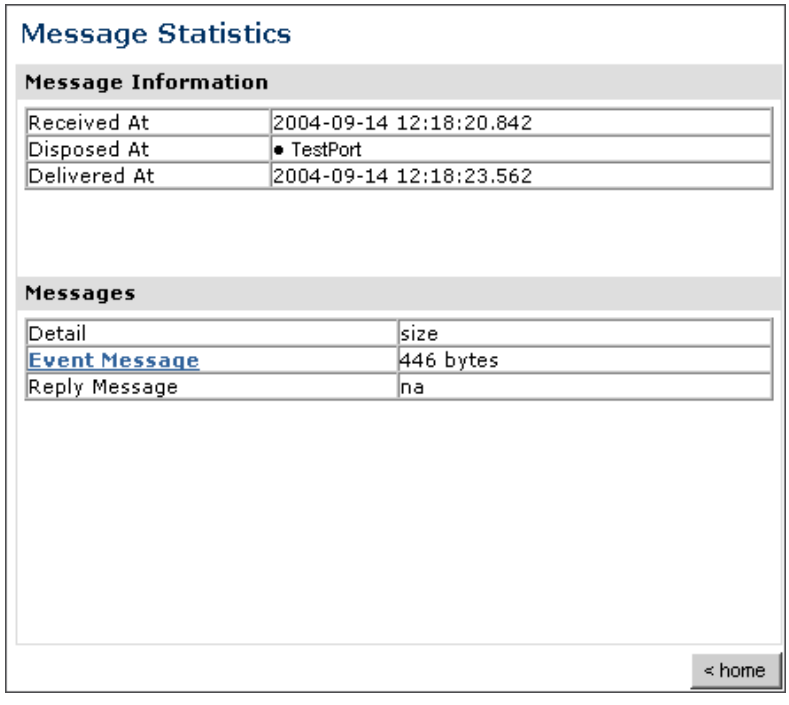
**5.** From the Ports drop-down list, select a port for the channel.

The following image shows the Port Level Event Summary (Channel Statistics) window that opens. 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 following image shows the Event Level Statistics (Message Statistics) window that opens. 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, with an option to view an XML document of the event message. In the lower right of the window is a home button to click to return to the iBSE Monitoring Settings window.




**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 and to display 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://localhost:port/iwjcaivp`

   where:

   *localhost*

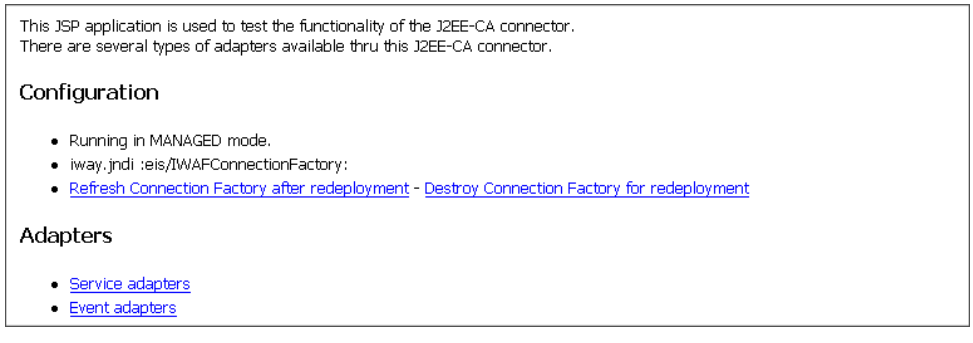
   Is the name of the machine where your application server is running.

   *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 page that opens. The page 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.

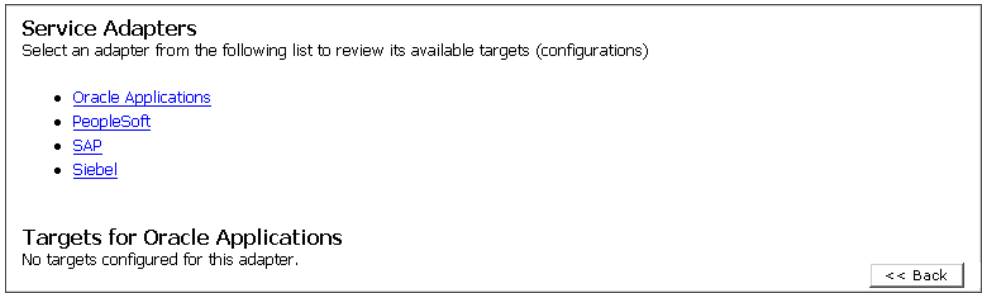
**a.** Click *Destroy Connection Factory for redeployment*.

**b.** Redeploy the JCA connector module using the BEA WebLogic Server console.

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

**3.** Click *Service adapters*.

The following image shows the Service Adapters page that opens. The page provides a live list of available service adapters and a list of targets configured for a specific adapter. In the lower right is a Back button to click to return to the previous page.




**4.** Select a service adapter to monitor.

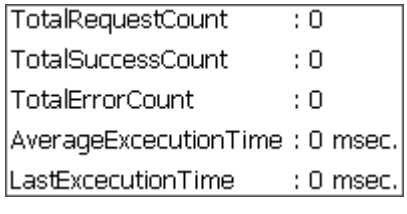
**a.** Click the desired target for your service adapter.

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

**c.** In the Input Doc area, enter a request document that was 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.



## 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://localhost:port/iwjcaivp`

   where:

   *`localhost`*

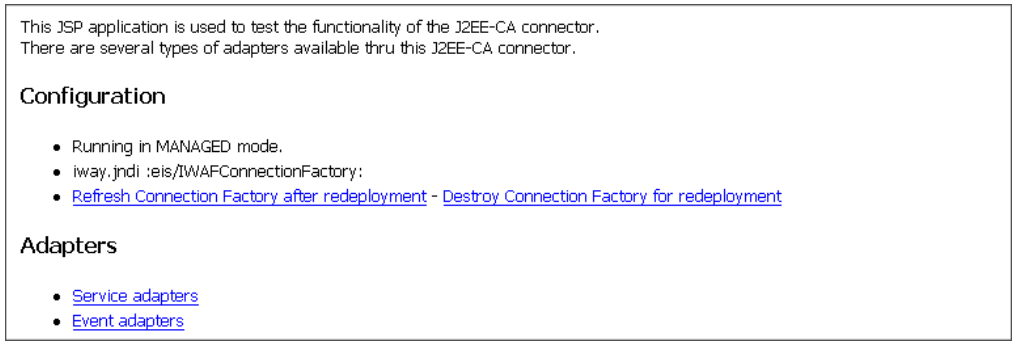
   Is the name of the machine where your application server is running.

   *`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 page that opens. The page 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 event 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 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 module using the BEA WebLogic Server console.

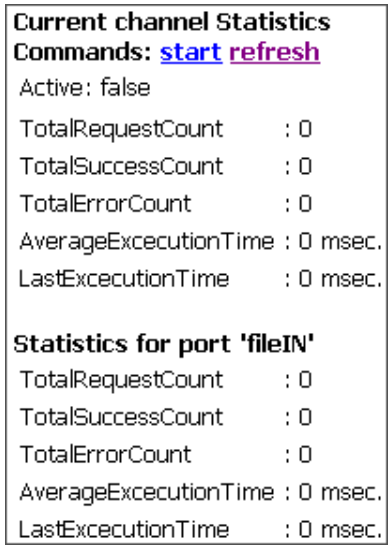
   **c.** In the JCA Test Tool, click *Refresh Connection Factory after redeployment*.

3. Click *Event adapters*.

The Event Adapters page 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. There are options to click in the upper right of the page to start or refresh the channel.




## Setting Engine Log Levels

The following section describes how to set engine log levels for Servlet iBSE and JCA. For more information, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

## Procedure: How to Enable Tracing for Servlet iBSE

To enable tracing for Servlet iBSE:

1. Open the Servlet iBSE configuration page at:

   `http://localhost:port/ibse/IBSEConfig`

   where:

   *localhost*

   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. Open the extracted ra.xml file in a text editor.
2. 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>
   ```

   A directory in the configuration directory contains the logs.

   a. Review the logs generated by your application server.

   b. Leave the remainder of the previous file unchanged.

3. Save the file and exit the editor.

4. Redeploy the connector.

# Configuring Connection Pool Sizes

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

## Procedure: How to Configure Connection Pool Sizes

To configure connection pool sizes:

1. Open the extracted ra.xml file in a text editor.
2. 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>
```

3. Save the file and exit the editor.
4. Redeploy the connector.

# Migrating Repositories

During design time, a repository is used to store metadata created when using Application Explorer to configure adapter connections, browse EIS objects, configure services, and configure listeners to listen for EIS events. For more information on configuring repositories, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

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 may want to migrate a repository from a development environment to a production environment. The BEA WebLogic Server must be restarted to detect new repository changes.

## File Repositories

If you want to migrate a File repository to another destination, copy the ibserepo.xml file from the following path:

```
drive:\Program Files\iWay55\bea\ibse\ibserepo.xml
```

where:

*drive*

Is the location of your iWay 5.5 installation.

You can place the ibserepo.xml file in a new location that is a root directory of the iBSE Web application, for example:

```
drive:\ProductionConfig\bea\ibse\ibserepo.xml
```

## iBSE Repositories

The following topic describes how to migrate an iBSE repository that is configured for Oracle. You can follow the same procedure if you want to migrate an iBSE repository that is configured for Microsoft SQL Server 2000, Sybase, or DB2. However, when you are configuring a new environment, you must execute the script that creates the repository tables for your database. In addition, verify that all required files and drivers for your database are in the class path. For more information on configuring repositories, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

**Note:** The following procedure allows you to migrate only Web services. If migrating event handling information is one of your requirements, you must migrate at the database level. For more information, see *Migrating Event Handling Configurations* on page 9-26.

### Procedure: How to Migrate an iBSE Repository Configured for Oracle

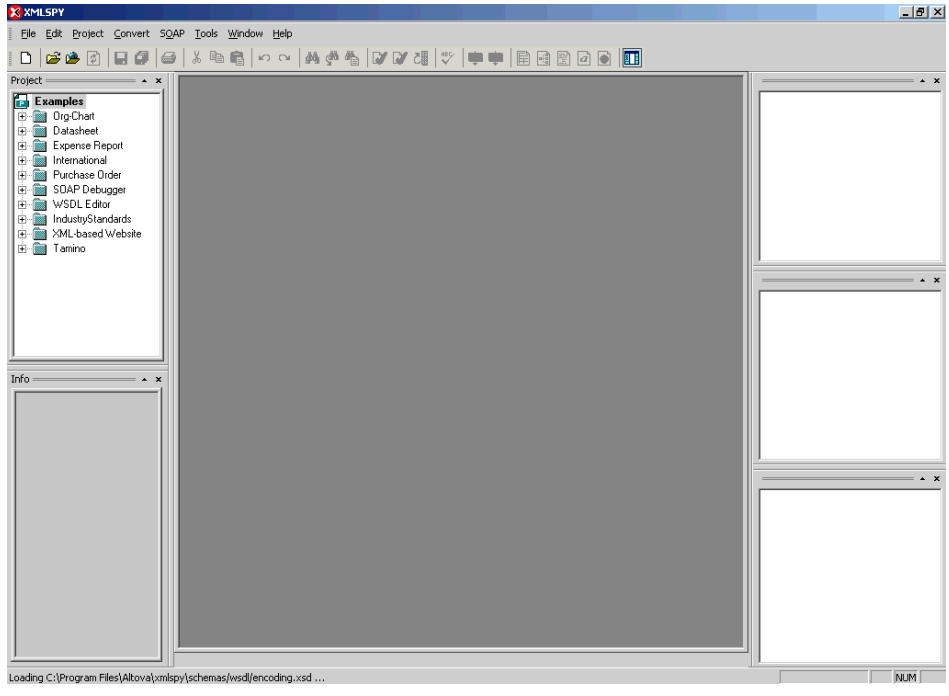
To migrate an iBSE repository that is configured for Oracle:

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

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

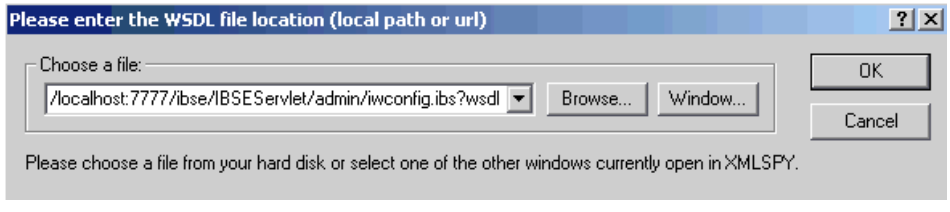
2. Open a third party XML editor, for example, XMLSPY.

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*.

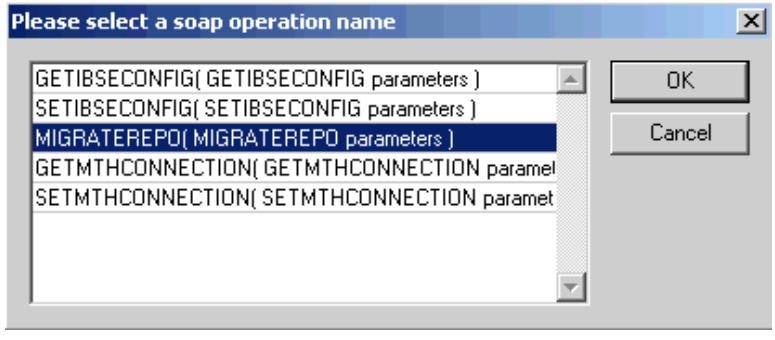
The following image shows the WSDL file location dialog box that opens, where you enter a local path or URL. The dialog includes Browse, Window, OK, and Cancel buttons.




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

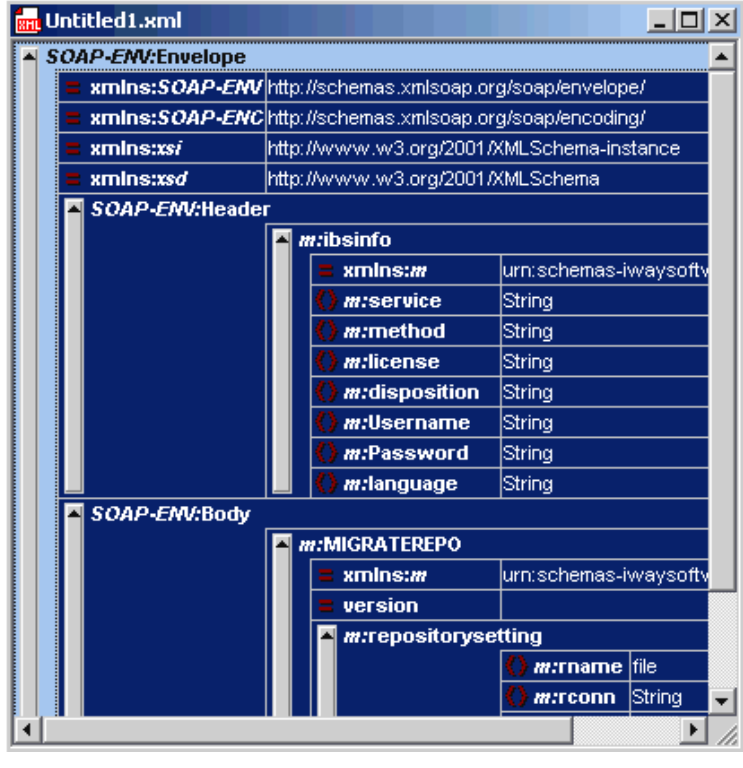
**5.** Click *OK*.

The following image shows the soap operation name dialog box that opens with a list of available control methods. You can select from the list and click OK or to escape from the dialog box, you can click Cancel.




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

The following image shows a portion of the window that opens with the structure of the SOAP envelope. It includes information about location and schemas.



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

   In the following image, the pointer points to the Text view icon.

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

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

9. Locate the following section:

```
<m:MIGRATEREPO
xmlns:m="urn:schemas-iwaysoftware-com:jul2003:ibse:config" version="">
<m:repositorysetting>
<m:rname>oracle</m:rname>
<m:rconn>String</m:rconn>
<m:rdriver>String</m:rdriver>
<m:ruser>String</m:ruser>
<m:rpwd>String</m:rpwd>
</m:repositorysetting>
<m:servicename>String</m:servicename>
</m:MIGRATEREPO>
```

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

      For example, the Oracle repository URL has the following format:

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

   b. For the <m:rdriver> tag, replace the String placeholder with the location of your Oracle driver.

      **Note:** This is an optional tag. If you do not specify a value, the default Oracle JDBC driver is used.

   c. For the <m:ruser> tag, replace the String placeholder with a valid user name to access the Oracle repository.

   d. For the <m:rpwd> tag, replace the String placeholder with a valid password to access the Oracle repository.

10. Perform one of the following migration options.

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

```
<m:servicename>Service1</m:servicename>
```

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

```
<m:servicename>Service1</m:servicename>
<m:servicename>Service2</m:servicename>
```

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

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

    Your iBSE repository and the Web services you specified migrate to the new Oracle repository URL that you specified.

## JCA Repositories

The following procedure describes how to migrate a JCA repository. For more information on configuring JCA repositories, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

### Procedure: How to Migrate a JCA Repository

To migrate a JCA repository:

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

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

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

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

   Your JCA repository migrates to the new JCA configuration directory.

## Migrating Event Handling Configurations

This topic describes how to migrate your iBSE repositories at a database level for Microsoft SQL Server 2000, Oracle, Sybase, or DB2. You can use this information to migrate event handling information, for example, port or channel configurations.

### Procedure How to Migrate a Microsoft SQL Server 2000 Repository

To migrate a Microsoft SQL Server 2000 repository:

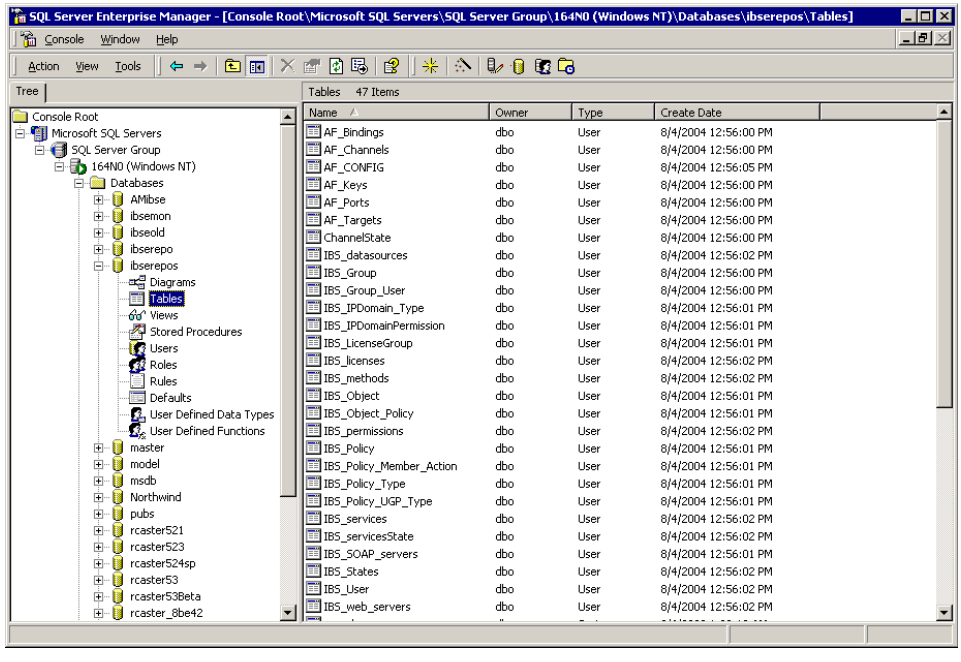
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

`C:\Program Files\iWay55\etc\setup`

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

`iwse.sql`

You can use iwse.sql to create the database tables that are used by iBSE. For example, the following image shows the tree in the left pane and tables in the right pane. The tables are listed by name in one column with corresponding columns for information about owner, type, and the date the table was created.




For more information on configuring the Microsoft SQL Server 2000 repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

**2.** To migrate the tables that were created by the iwse.sql script for iBSE, use your Microsoft SQL Server 2000 database tool set. For more information, consult your database administrator.

## Procedure How to Migrate an Oracle Repository

To migrate an Oracle repository:

1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

   ```
   C:\Program Files\iWay55\etc\setup
   ```

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

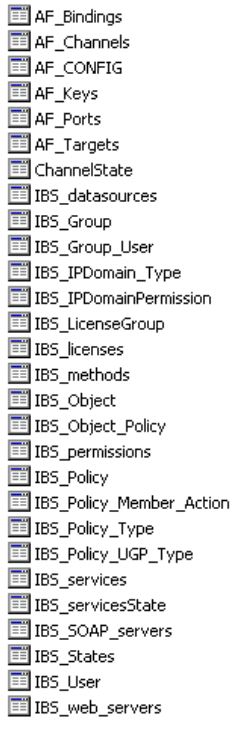
   For Oracle 8:

   ```
   iwse.ora
   ```

   For Oracle 9:

   ```
   iwse.ora9
   ```

2. To create the Oracle database tables that are used by iBSE, use the SQL script as shown in the example in the following image that shows a list of tables.




   For more information on configuring the Oracle repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

3. To migrate the tables that were created by the SQL script for iBSE, use your Oracle database tool set. For more information, consult your database administrator.

## Procedure How to Migrate a Sybase Repository

To migrate a Sybase repository:

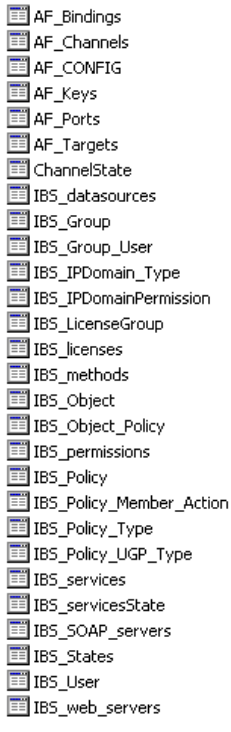
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

   ```
   C:\Program Files\iWay55\etc\setup
   ```

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

   ```
   sybase-iwse.sql
   ```

2. To create the Sybase database tables that are used by iBSE, use the SQL script as shown in the example in the following image that shows a list of tables.

   For more information on configuring the Sybase repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

3. To migrate the tables that were created by the SQL script for iBSE, use your Sybase database tool set. For more information, consult your database administrator.

## Procedure How to Migrate a DB2 Repository

To migrate a DB2 repository:

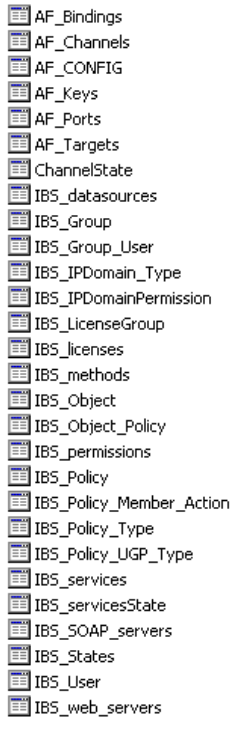
1. Open a command prompt and navigate to the iWay setup directory. The default location on Windows is:

   ```
   C:\Program Files\iWay55\etc\setup
   ```

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

   ```
   db2-iwse.sql
   ```

2. To create the DB2 database tables that are used by iBSE, use the SQL script as shown in the example in the following image that shows a list of tables.




   For more information on configuring the DB2 repository, see the *iWay Installation and Configuration for BEA WebLogic* documentation.

   You can migrate the tables that were created by the SQL script for iBSE using your DB2 database toolset. For more information, consult your database administrator.

# 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

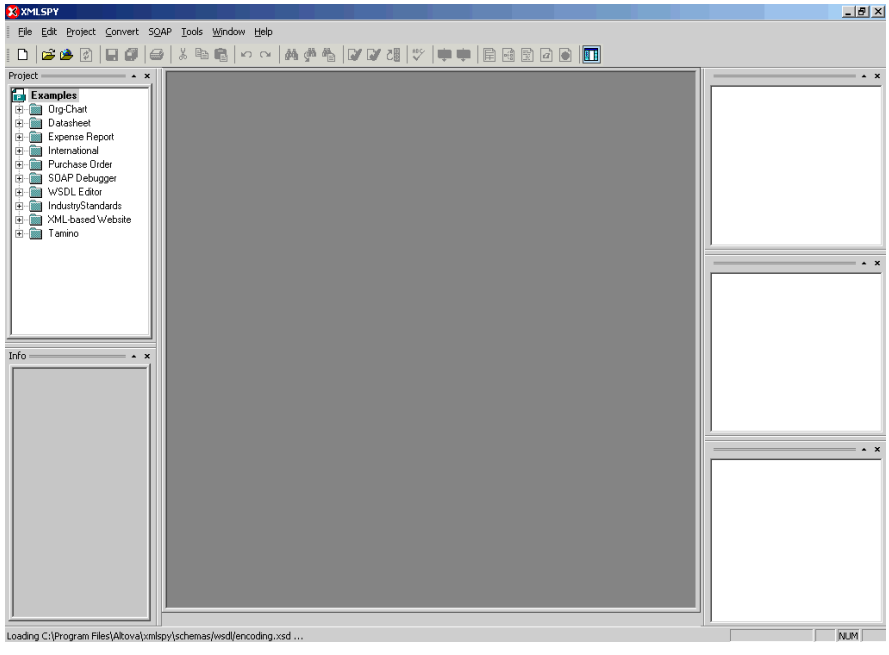
To export a target:

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.

   In the following image, the pointer points to the Text view icon.


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

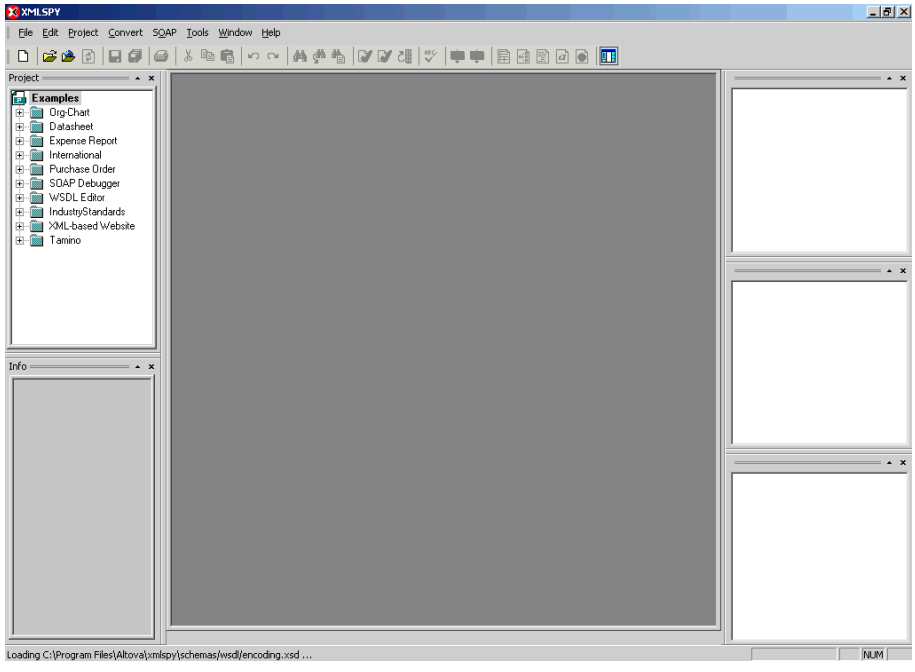
To import a target:

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 toolbar.

In the following image, the pointer points to the Text view icon.


**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

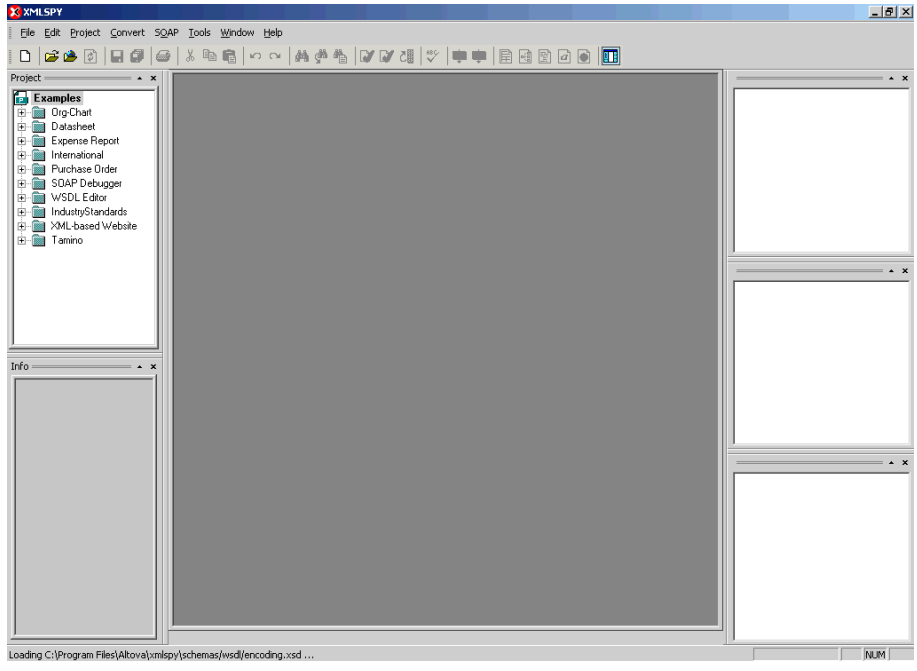
To retrieve Web service method connection information:

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 toolbar.

   In the following image, the pointer points to the Text view icon.


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

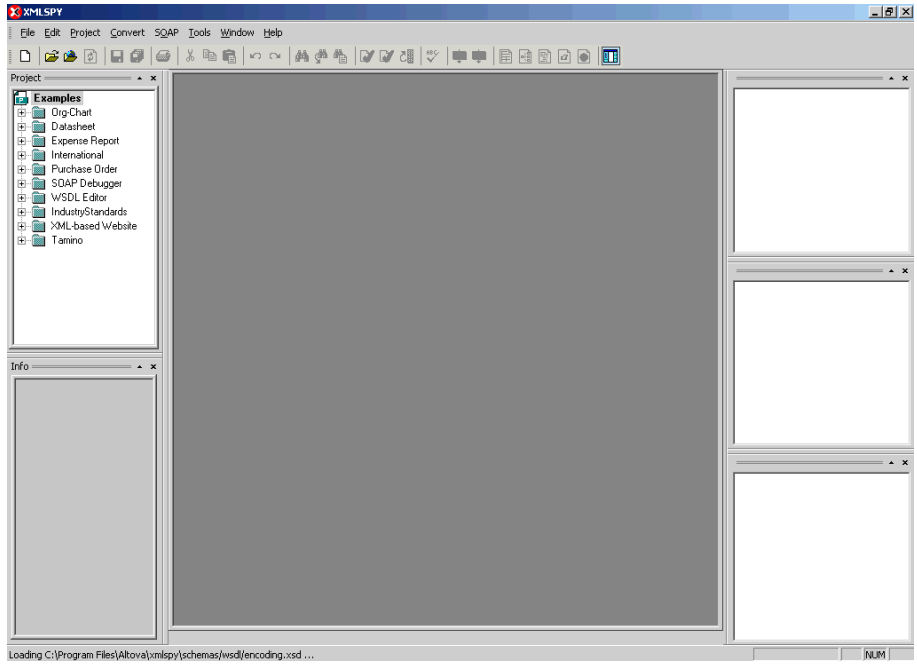
To update Web service method connection information:

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 toolbar.

   In the following image, the pointer points to the Text view icon.


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

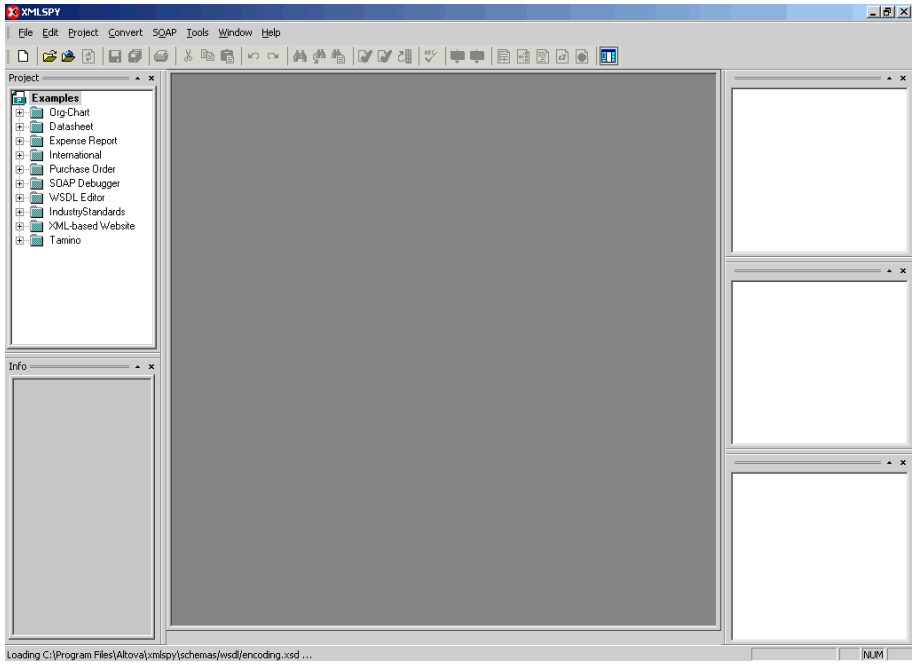
To start a channel programmatically:

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.

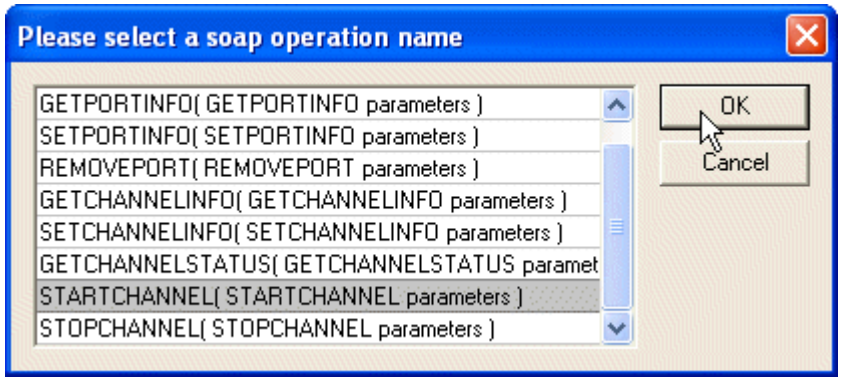



3. From the SOAP menu, select *Create new SOAP request*.

   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. You can select one and click OK or to escape from the dialog box, you can click Cancel.




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 toolbar.

   In the following image, the pointer points to the Text view icon.


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

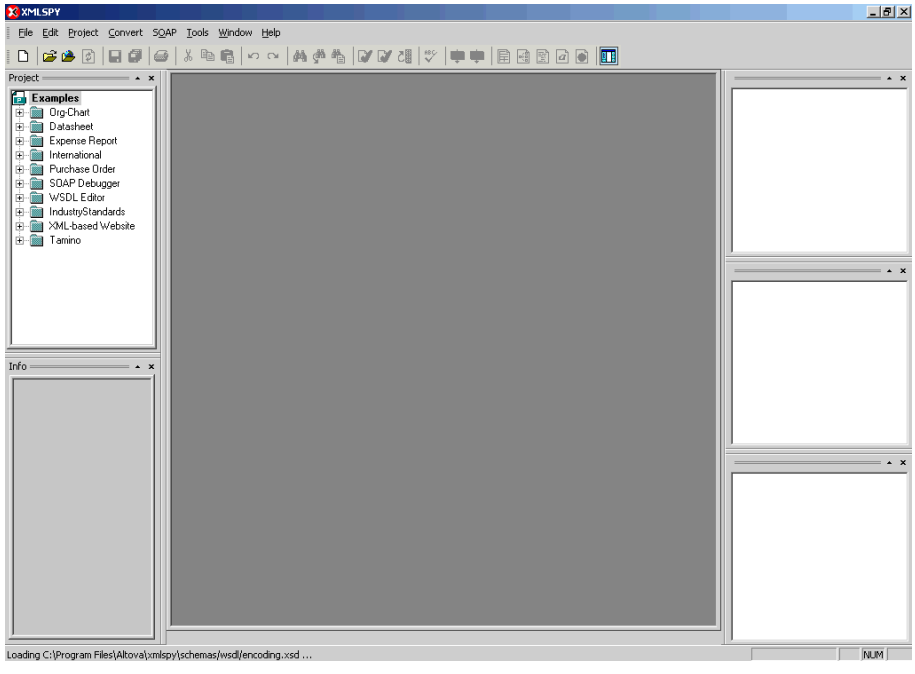
To stop a channel programmatically:

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.

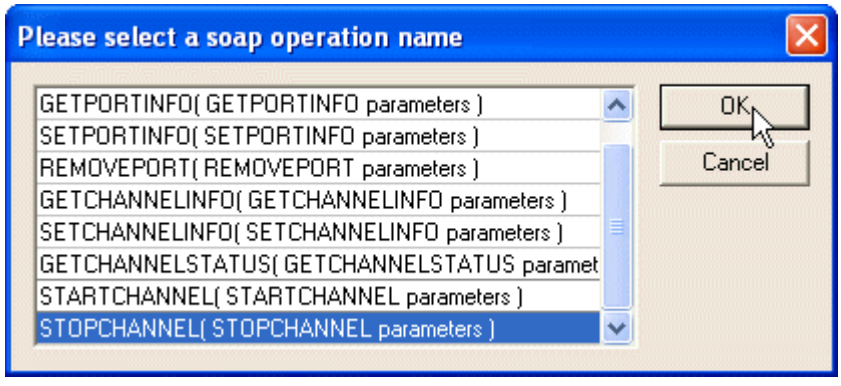



3. From the SOAP menu, select *Create new SOAP request*.

   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. You can select one and click OK or to escape from the dialog box, you can click Cancel.




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 toolbar.

   In the following image, the pointer points to the Text view icon.


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.*

APPENDIX A

# Using Application Explorer to Create XML Schemas and Business Services

**Topics:**

- Starting Application Explorer in BEA WebLogic Workshop
- Creating a New Configuration for iBSE or JCA
- Connecting to PeopleSoft
- Viewing a Business Object and Creating an XML Schema
- Creating a Business Service
- Adding a Control for a Resource in BEA WebLogic Workshop
- Adding an Extensible CCI Control to a BEA WebLogic Workshop Application

This section describes how to use Java Swing Application Explorer running in BEA WebLogic Workshop to create XML schemas and business services for PeopleSoft Component Interfaces (CI) and Messages.

# Starting Application Explorer in BEA WebLogic Workshop

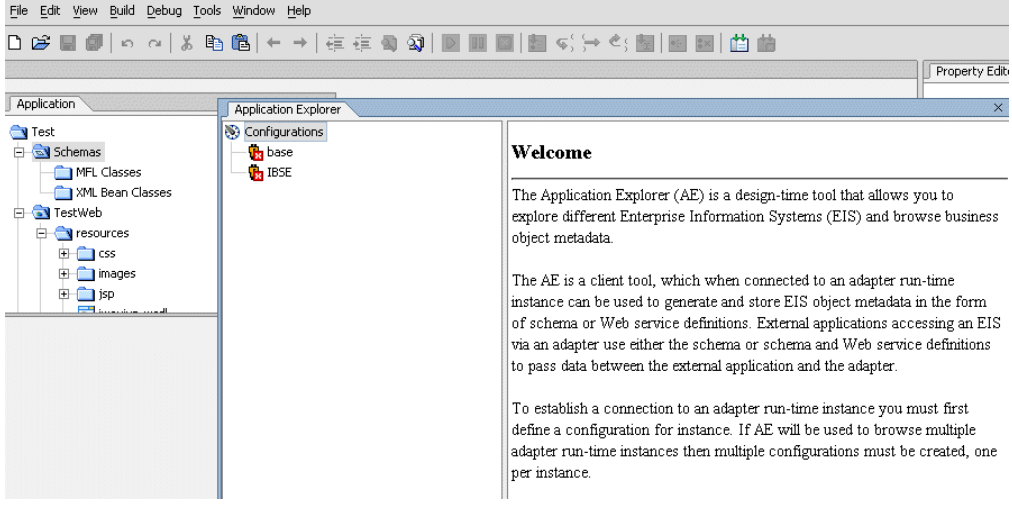
The server must be started where Application Explorer is running. Before you can use Application Explorer, you must start BEA WebLogic server.

***Procedure*** **How to Start Application Explorer in BEA WebLogic Workshop**

To start Application Explorer running in BEA WebLogic Workshop:

1. Before starting Application Explorer, ensure that BEA WebLogic Server is running.
2. Start BEA WebLogic Workshop.
3. From the BEA WebLogic Workshop View menu, select *Windows* and then, *Application Explorer*.

   Application Explorer opens in BEA WebLogic Workshop.



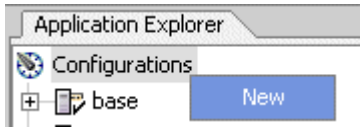

   You can resize and drag-and-drop the Application Explorer window within BEA WebLogic Workshop. For example, you can drag it to the upper part of BEA WebLogic Workshop.

# Creating a New Configuration for iBSE or JCA

Before you can start using Application Explorer, you must define a new configuration for iBSE or JCA.

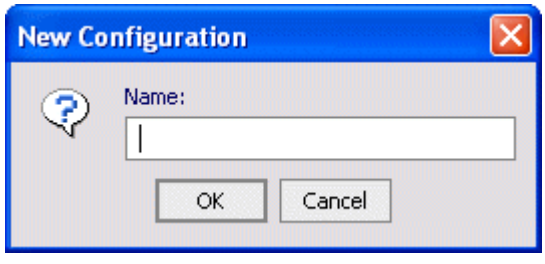
## *Procedure* How to Create a New Configuration for iBSE or JCA

To create a new configuration:




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

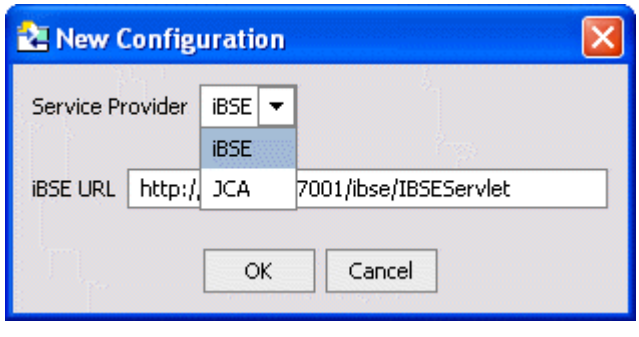
   The New Configuration dialog box opens.




2. Type the name of the new configuration and click *OK*.

   **Note:** If you are creating a new JCA configuration, type *base* in the name field. You must use this value if you are pointing to the default iWay configuration.

   The following dialog box opens.



**3.** From the Service Provider drop-down list, select *iBSE* or *JCA*.

- If you select iBSE, type the URL for iBSE, for example,

  `http://localhost:7001/ibse/IBSEServlet`

  where:

  *`localhost`* is where your application server is running.

- If you select JCA, enter the full path to the directory where iWay 5.5 is installed, for example,

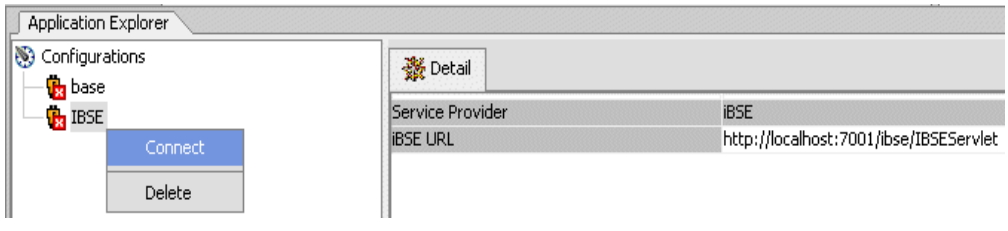
  `C:\Program Files\iWay55`

  where:

  *`iWay55`* is the full path to your iWay installation.

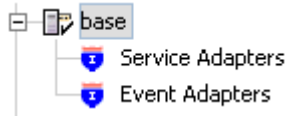
A node representing the new configuration appears under the Configurations node. The right pane provides details of the configuration you created.

After you add your configuration, you must connect to it.




**4.** Right-click the configuration to which you want to connect, and select *Connect*.

When you connect to JCA, the Service Adapters and Event Adapters nodes appear.




When you connect to iBSE, the Service Adapters, Event Adapters, and Integration Business Services nodes appear.

5. To display the service and event adapters that are installed, expand each node.

   The Service Adapters list includes a PeopleSoft node that enables you to connect to PeopleSoft metadata and create XML request and response schemas to use to listen for events or create business services. For more information, see *Creating a Business Service* on page A-14.

   The Event Adapters list includes a PeopleSoft node that enables you to create ports and channels for PeopleSoft event handling. For more information, see *Adding a Control for a Resource in BEA WebLogic Workshop* on page A-19.

# Connecting to PeopleSoft

To browse PeopleSoft Component Interfaces (CI) and Messages, you must create a PeopleSoft target and connect to it. The target serves as your connection point. You must establish a connection to PeopleSoft every time you start Application Explorer or after you disconnect from PeopleSoft.

The left pane displays the application systems supported by Application Explorer. These are based on the adapters you installed and are licensed to use.

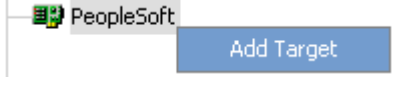
## Creating and Connecting to a Target

To connect to PeopleSoft for the first time, you must create a new target. The target is automatically saved after it is created.

### *Procedure* How to Create a New Target

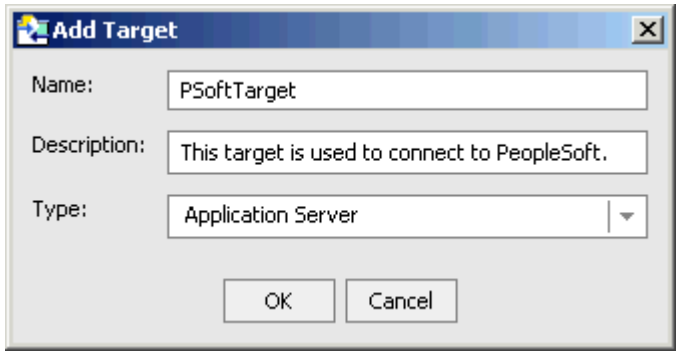
To create a target:

1. In the left pane, expand *Service Adapters* and click the *PeopleSoft* node.
2. To view the options, right-click the *PeopleSoft* node.



**3.** Select *Add Target*.

The Add target dialog box opens.




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

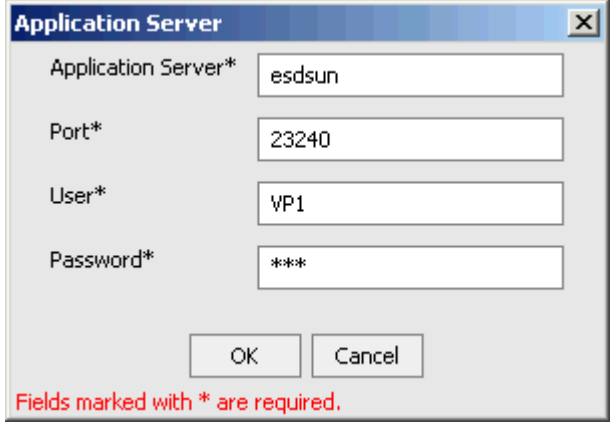
**b.** In the Description field, type a brief description of the target.

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

Application Server is the default value.

**4.** Click *OK*.

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

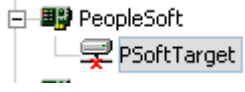


5. Specify the following information:

| Field | Description |
|---|---|
| Application Server | Host name or IP address for the machine that is hosting the PeopleSoft application. |
| Port | Port number on which the application server is listening. |
| User | Valid user name to access PeopleSoft. |
| Password | Valid password associated with the user name. |

6. Click *OK*.

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

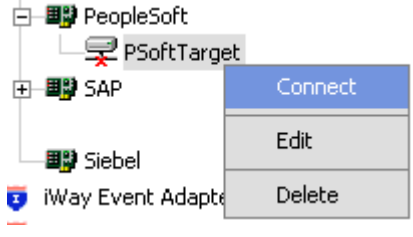



   You can now connect to the target you defined.

## *Procedure* How to Connect to a Target

To connect to a PeopleSoft target:

1. In the left pane, expand the *PeopleSoft* node and select the target to which you want to connect, for example, PSoft Target.
2. In the right pane, enter a password for that target.




3. In the left pane, right-click the target and select *Connect*.

   The PSoftTarget node in the left pane changes to reflect that a connection was made.

4. Expand the target node to reveal the list of PeopleSoft business objects.


## Managing a Target

Although you can maintain multiple open connections to different application systems, it is recommended that you close connections when they are not in use. After you disconnect, you can modify an existing target.

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

### *Procedure* How to Disconnect From a Target

To disconnect from a target:

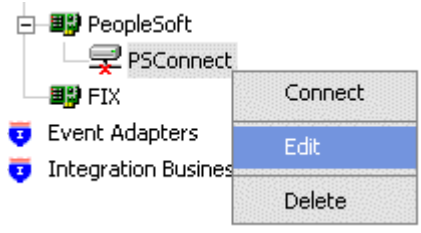

1. Right-click the PeopleSoft target from which you want to disconnect.
2. Select *Disconnect*.

   Disconnecting from the application system drops the connection, but the node remains. The PSoftTarget node in the left pane changes to reflect that you disconnected from the target.

## *Procedure* How to Edit a Target

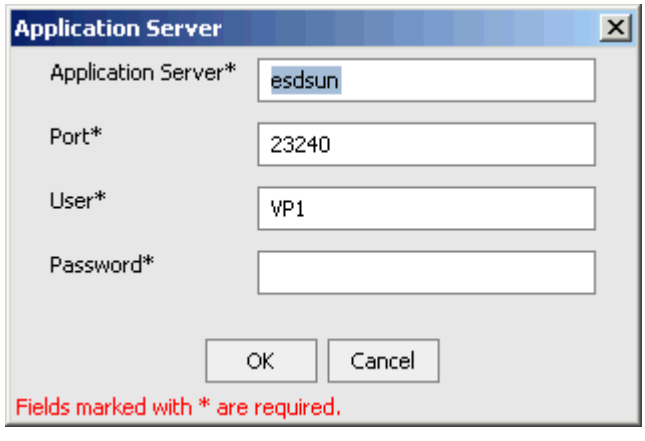
To edit a target:

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




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

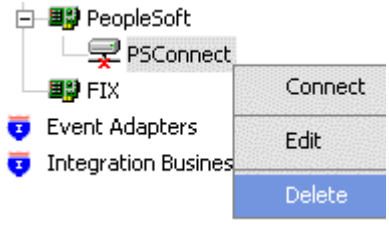
   The Application Server dialog box opens.




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

### *Procedure* How to Delete a Target

To delete a target:




1. In the left pane, right-click the target.
2. Select *Delete*.

   The PSoftTarget node disappears from the left pane.

## Viewing a Business Object and Creating an XML Schema

After you create a new configuration and connect to PeopleSoft, Application Explorer enables you to explore and browse business object metadata. For example, Application Explorer enables you to view Component Interfaces, Component Interfaces(RPC), and Message metadata stored in PeopleSoft.

For Component Interfaces(RPC), the BEA WebLogic Adapter for PeopleSoft enables Delete, Insert, Query, Update, and Find.

### *Procedure* How to View Metadata in PeopleSoft

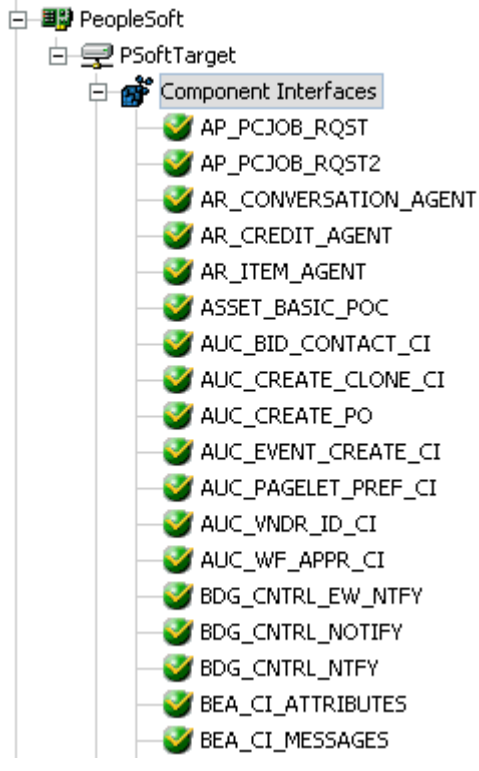
To view metadata in PeopleSoft:

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

   The available system objects appear.

**2.** To expand the PeopleSoft repository node you want to explore, click the icon to the left of the repository name, for example, Component Interfaces.

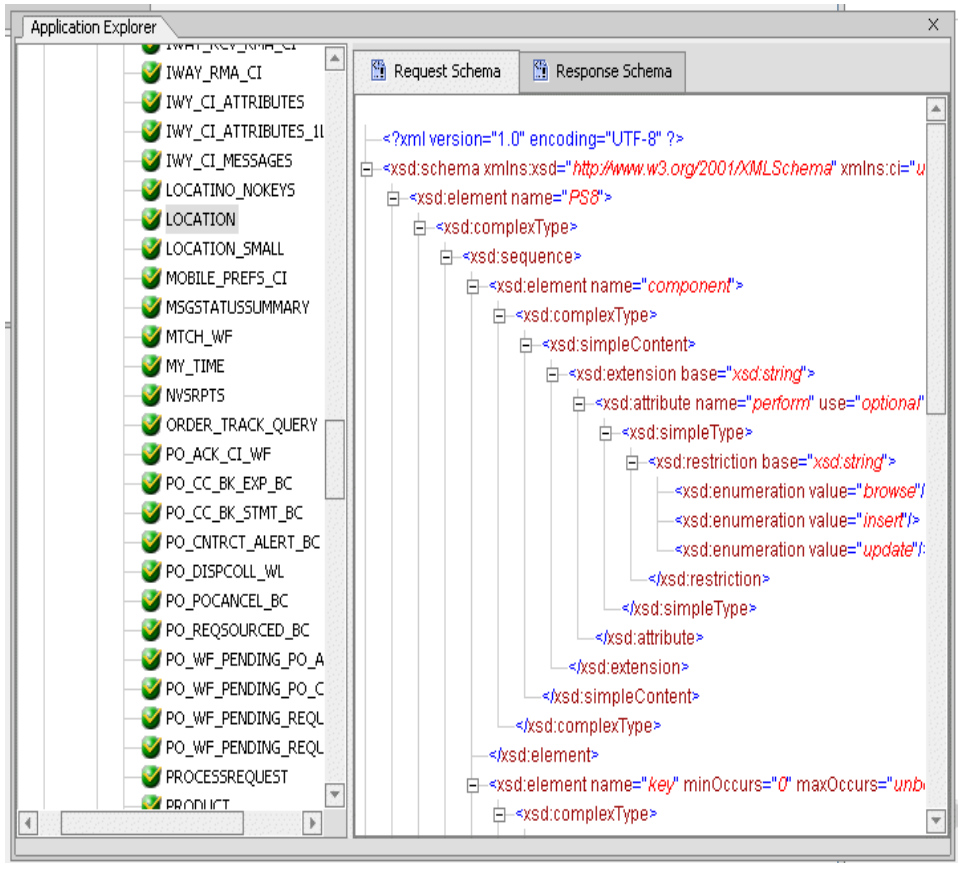
A list of business object groups appears.



**3.** Click the *LOCATION* Component Interface.

The following XML schemas appear for the LOCATION Component Interface:

- Request
- Response



4. To view the appropriate schema in the right pane, click the *Request Schema* or the *Response Schema* tab.

   The schema you select appears in the pane.


## *Reference* Schema Location

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

```
drive:\Program Files\iWay55\bea\ibse\wsdl\schemas\service\PeopleSoft
\PSoftConnect\S9AFF52F
```

where:

*PSoftConnect*

Is the name of the PeopleSoft target.

*S9AFF52F*

Is a randomly generated folder name indicating where the schemas are stored.

# Creating a Business Service

You can create a business service for objects you want to use with your adapter. To generate a business service, you must deploy the BEA WebLogic Adapter for PeopleSoft using the Integration Business Services Engine (iBSE). iBSE exposes functionality as Integration Business Services and serves as a gateway to heterogeneous back-end applications and databases.

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

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

Because Application Explorer runs within BEA WebLogic Workshop, you can easily incorporate business services into BEA WebLogic Workflows. To enable BEA WebLogic Workshop to use business services, you export the WSDL to a directory accessible to BEA WebLogic Workshop.

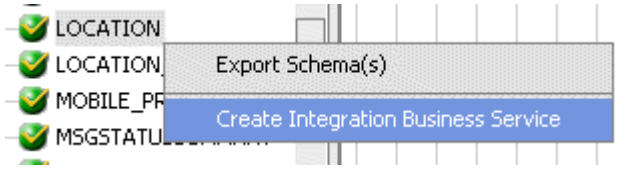
**Note:** In a J2EE Connector Architecture (JCA) implementation of adapters, business services are not available. When the adapters are deployed to use the Connector for JCA, the Common Client Interface provides integration services using the adapters. For more information, see the *BEA WebLogic ERP Adapter Installation and Configuration* manual and the *Connector for JCA for BEA WebLogic Server User's Guide*.

The following procedures use the PeopleSoft Component Interface called LOCATION as an example, which returns a list of vendor locations from PeopleSoft.

## *Procedure* How to Create a Business Service

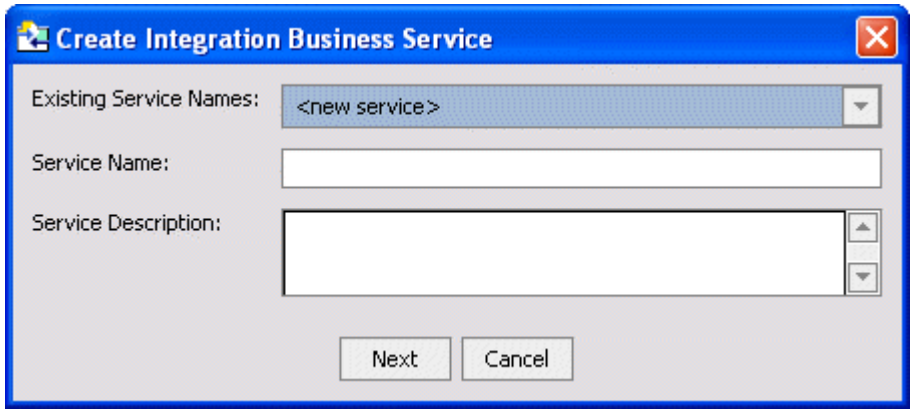
To create a business service:

1. In the left pane of Application Explorer, select *LOCATION* from the list of Component Interfaces.



2. Right-click *LOCATION* and select *Create Integration Business Service*.

   The Create Integration Business Service dialog box opens.



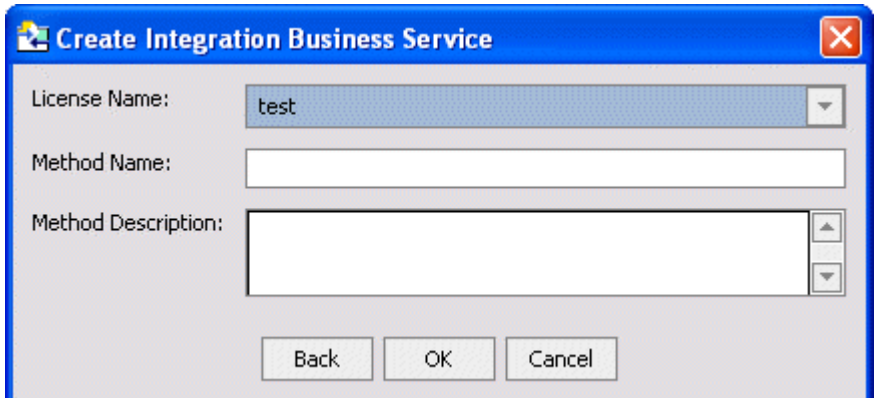

   a. From the Existing Service Names drop-down list, select whether you want to create a new service name or use an existing service name.

   b. In the Service Name field, type a name for the business service, for example, PSoftService.

   c. In the Service Description field, type a brief description of the business service.

3. Click *Next*.

   The Create Integration Business Service dialog box displays additional fields.



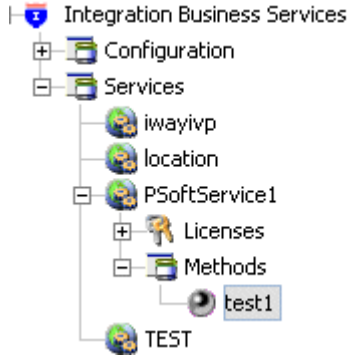

   a. From the License Name drop-down list, select a license.

   b. In the Method Name field, type a name for the method.

   c. In the Method Description field, type a brief description for the method.

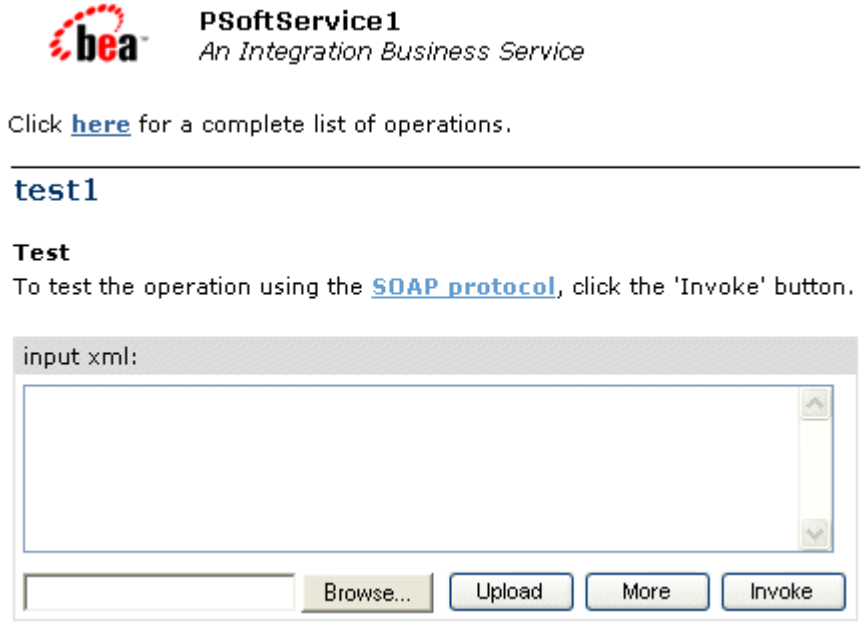
**4.** Click *OK*.

The business service and method appear below the Integration Business Services node.




In the left pane, all the available business services that were created appear. The PSoftService node is expanded, and the LOCATION method is automatically selected.

On the right, the test pane opens.




**5.** To invoke the service, enter a sample XML document in the input xml field.

For sample input XML, see *Retrieving a List of Locations*.

6. Click *Invoke*.

   The result appears in the right pane.

### *Example* Retrieving a List of Locations

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

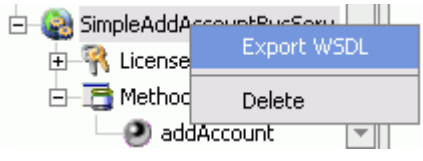
```
<?xml version="1.0" encoding="UTF-8"? >
<PS8>
  <component perform="browse">LOCATION</component>
  <key name="Setid">SHARE</key>
  <key name="Location">ALBERTA</key>
</PS8>
```

## Exporting WSDL for Use in BEA WebLogic Workshop Workflows

Because Application Explorer runs within BEA WebLogic Workshop, you can easily incorporate business services into BEA WebLogic Workflows. To enable BEA WebLogic Workshop to use business services, you simply export the WSDL to a directory accessible to BEA WebLogic Workshop.

### *Procedure* How to Export WSDL for Use in BEA WebLogic Workshop Workflows

To export WSDL to a directory accessible to BEA WebLogic Workshop:

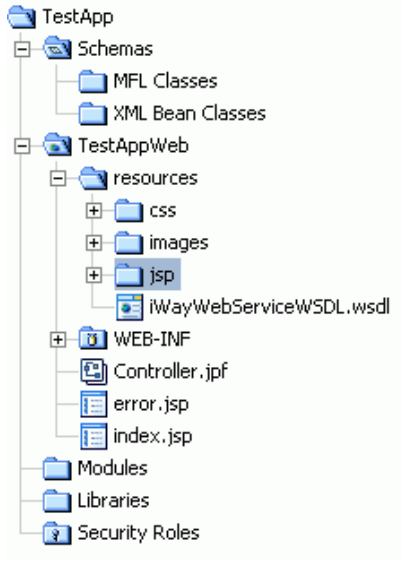



1. After you create a business service, right-click the business services name and select *Export WSDL*.

   The Save dialog box appears.

2. Save the WSDL to a directory accessible to BEA WebLogic Workshop, for example, the \resources directory in your BEA WebLogic Workshop Web application directory structure.

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




## Identity Propagation

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

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

**Note:** You can remove the following tags from the SOAP header, since they are not required:

```
<m:disposition>String</m:disposition>
```

```
<m:language>String</m:language>
```

# Adding a Control for a Resource in BEA WebLogic Workshop

Java controls provide a convenient way to incorporate access to resources. You can add controls in BEA WebLogic Workshop to use business services created by the Java Swing version of Application Explorer, or you can add controls that enable you to take advantage of the JCA resources of Application Explorer.

## Adding a Web Service Control to a BEA WebLogic Workshop Application

After you create a business service using Application Explorer and export the WSDL file, you can create a control for the business service.

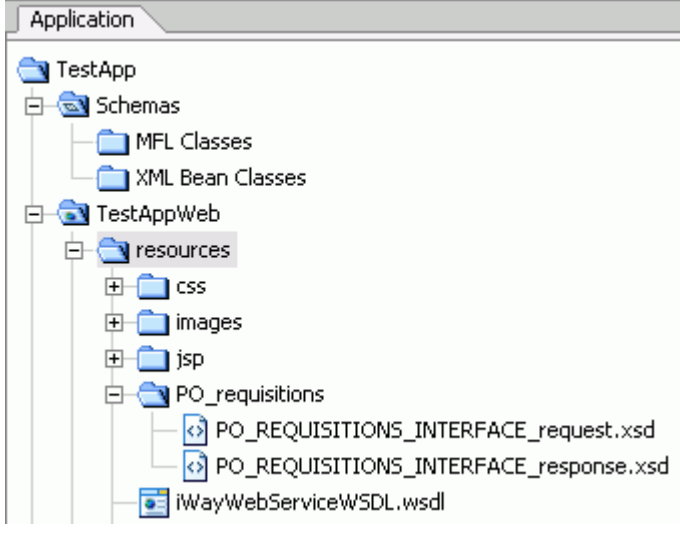
For more information on exporting a WSDL file, see *How to Export WSDL for Use in BEA WebLogic Workshop Workflows* on page A-17.

### *Procedure* How to Add a Business Service Control

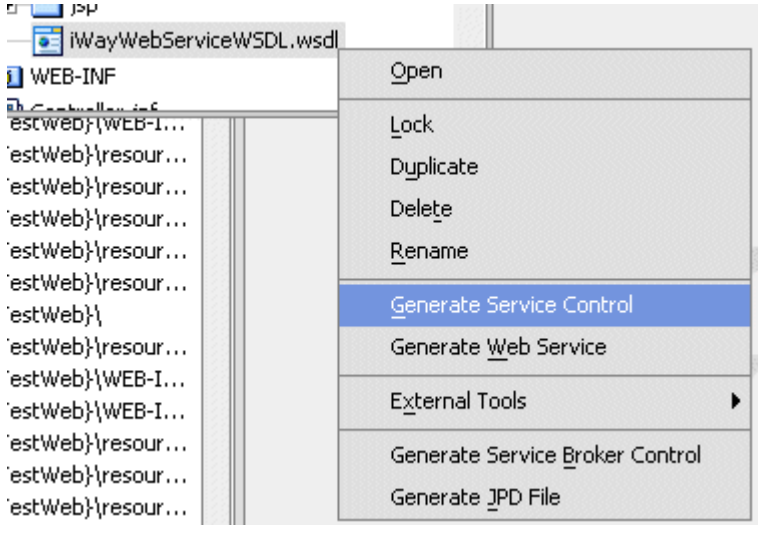
To add a business service control:

1. After exporting the WSDL file from Application Explorer, locate the file in the Application tab of your BEA WebLogic Workshop application.

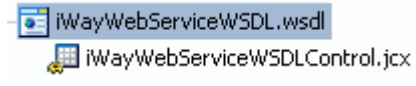
   For example, a WSDL file saved to the \resources directory in your BEA WebLogic Workshop Web application directory structure appears as follows.



2. Right-click the *WSDL* file and select *Generate Service Control*.




The control for the WSDL appears below the WSDL file in the resources tree.




# Adding an Extensible CCI Control to a BEA WebLogic Workshop Application

An iWay control enables access to resources provided by Application Explorer when it is used in conjunction with a JCA deployment. You must add an iWay control before using it in a BEA WebLogic Workshop application workflow.

The following topic describes the enhanced CCI control, which is extensible and provides JCX with typed inputs and outputs for JCA in BEA WebLogic Workshop.

## Overview

The extensible iWay CCI control provides:

- **Method and tag validation**. BEA WebLogic Workshop provides warnings regarding invalid methods and tags.
- **Improved error handling**.

You can define new methods that rely on the generic *service* and *authService* methods. For example, you can define a JCX with a new method without writing casting code or explicit transformations such as the following:

```
public iwaysoftwareAdapterPeoplesoftCiResponse.PS8Document
callpSoft(iwaysoftwareAdapterPeoplesoftCi.PS8Document aRequest) throws
Exception;
```

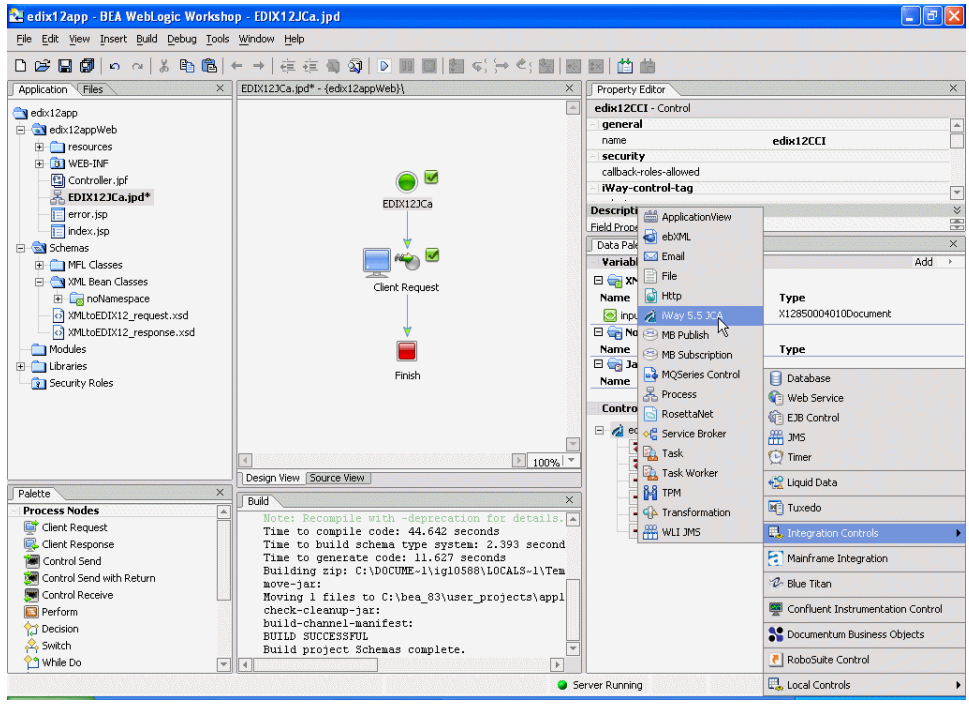
In addition, the extensible CCI control now generates a JCX file to which you can add your own methods. For more information, see *Defining a Control Using the Extensible CCI Control* on page A-21.

You can also use dynamic class casting to specify schema-based input or output XmlObjects to be casted into a pure XmlObject as a service method, which is expected by the CCI control. For more information, see *Using Dynamic Class Casting* on page A-27.

## *Example* Defining a Control Using the Extensible CCI Control

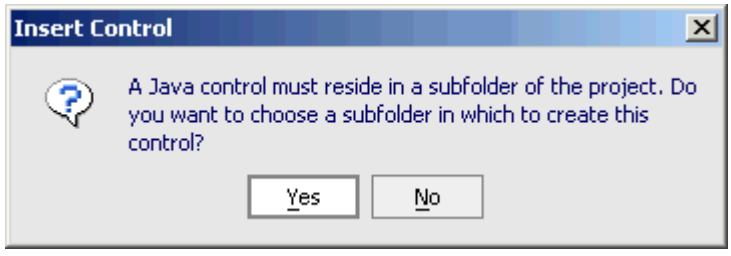
The following sample JCX demonstrates how to define a control for PeopleSoft using the extensible CCI control in BEA WebLogic Workshop.

1. Start BEA WebLogic Workshop and create a new project.



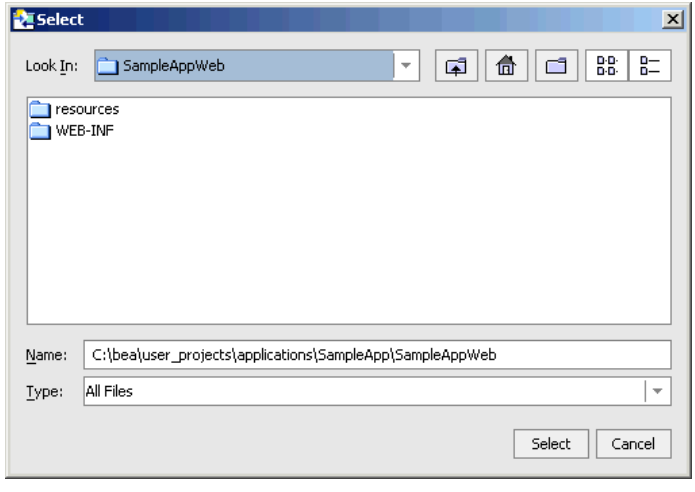

2. Click *Add* from the Controls section in the Data Palette tab, select *Integration Controls*, and click *iWay 5.5 JCA*.

   The Insert Control message box opens.



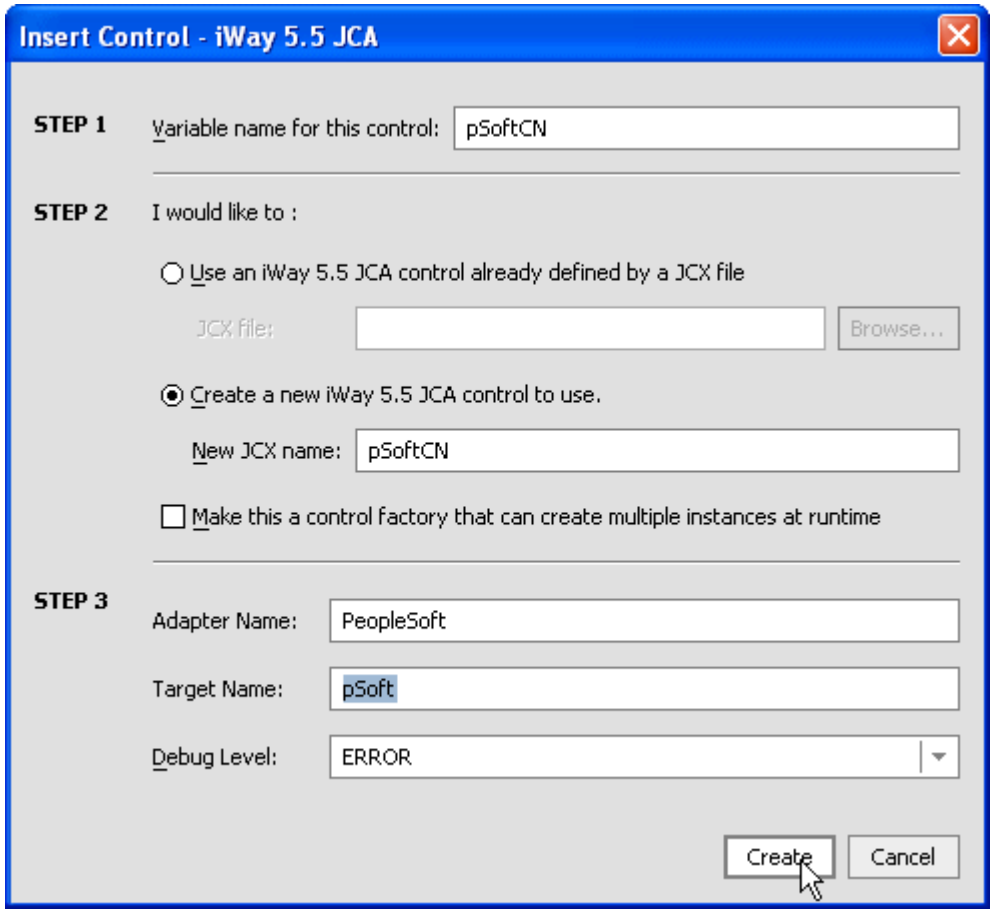

**3.** Click *Yes*.

The Select dialog box opens.




**4.** Choose a subfolder for the CCI control and click *Select*.

The Insert Control - iWay 5.5 JCA dialog box opens.



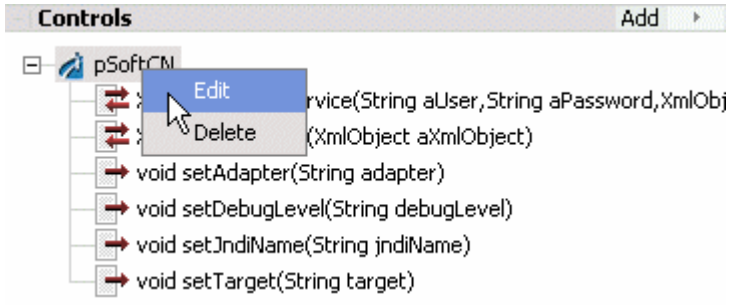

**a.** Provide a variable name for the control.

**b.** Click *Create a new iWay 5.5 JCA control to use* and provide a new JCX name.

**c.** Enter the adapter name, target name, and select a debug level from the drop-down list.

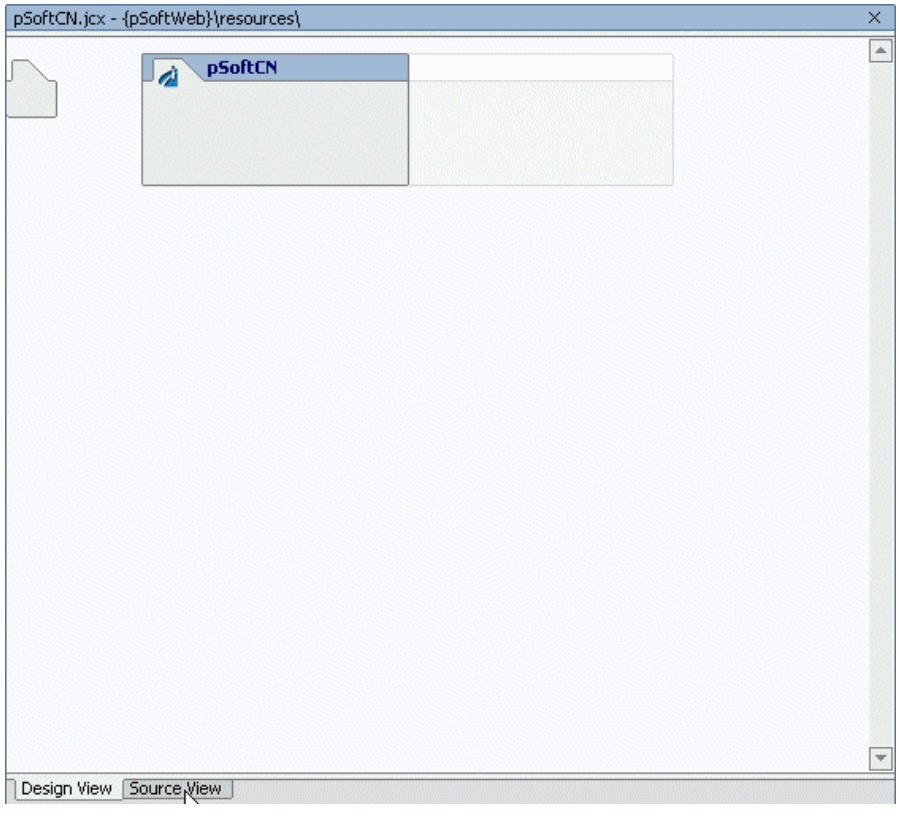
**5.** Click *Create*.

A new JCX file is created.



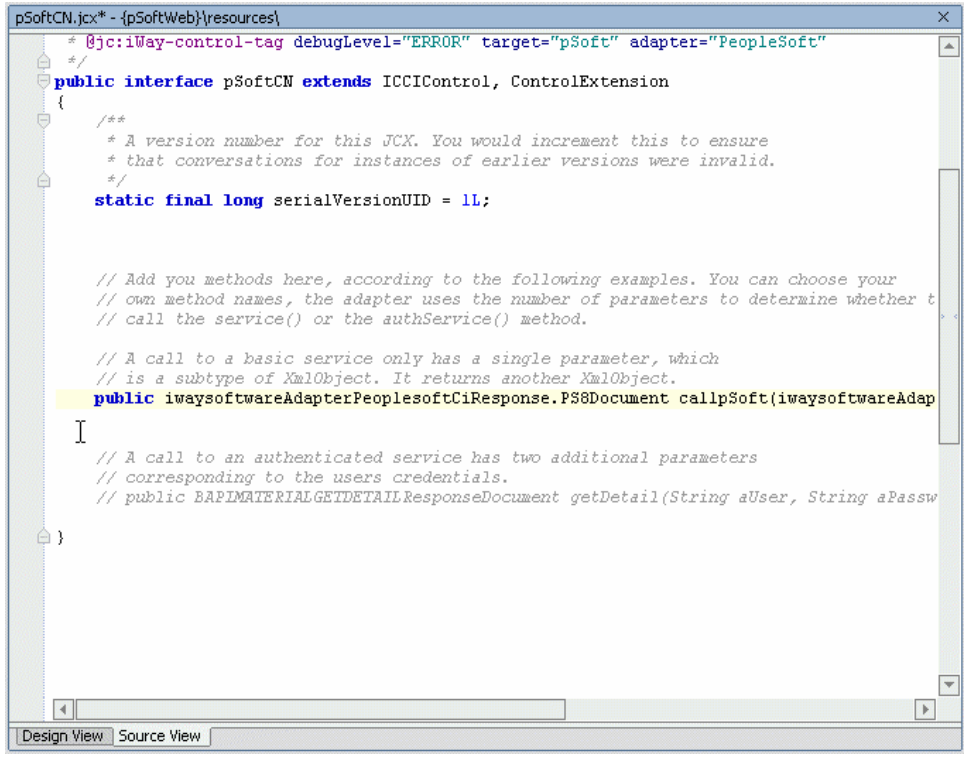

**6.** Right-click the control, for example, edix12CCI, and select *Edit*.

The Design View for the control opens.




**7.** Click the *Source View* tab.

The Source View for the control opens.




Perform the following steps:

**a.** Uncomment the public class definition.

**b.** Change the existing response data type to match your response data type according to the PeopleSoft schema, for example, iwaysoftwareAdapterPeoplesoftCiResponse.PS8Document.

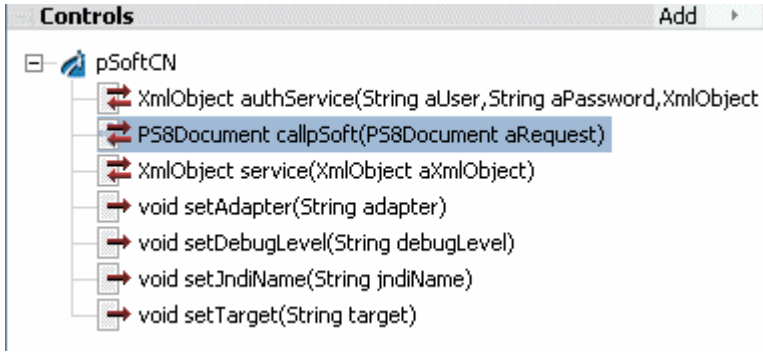
**c.** Change the existing method name to match your method, for example, callpSoft.

**d.** Change the existing incoming data type to match your request data type according to the PeopleSoft schema, for example, iwaysoftwareAdapterPeoplesoftCi.PS8Document.

The following syntax provides a sample PeopleSoft method call, which calls the adapter service:

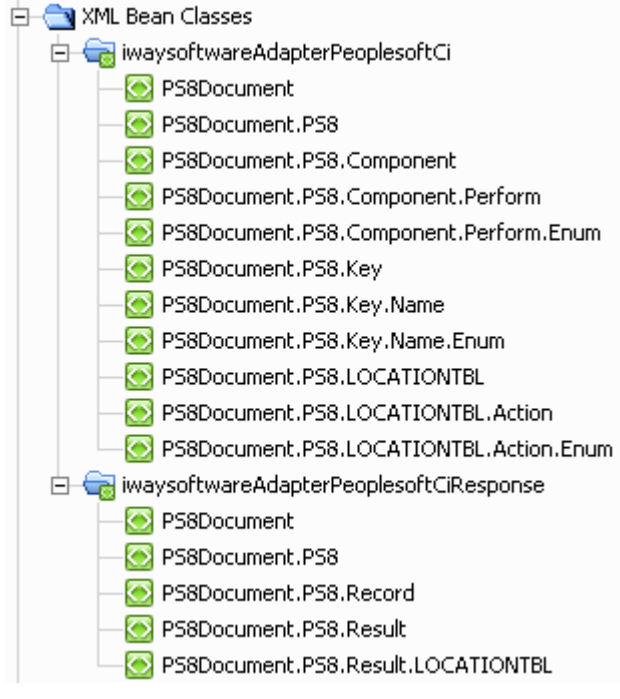
```
public iwaysoftwareAdapterPeoplesoftCiResponse.PS8Document
callpSoft(iwaysoftwareAdapterPeoplesoftCi.PS8Document aRequest) throws
Exception;
```

The following control is now available in BEA WebLogic Workshop and can be added to a workflow:




**Note:** You can view available data types under the *XML Bean Classes* folder in the *Application* tab, which are added once you import your XML request or response schemas from Application Explorer.

These data types are case sensitive and must be entered exactly as shown.



## Using the Extensible CCI Control

The extensible CCI control functions much like a database control since it generates JCX files to which you can add your own methods.

Your own methods can use the correct input and output types rather than the generic XmlObject types that the JCA control uses. Since the control is just a proxy that uses a reflection to call the relevant method, it handles the casting for you. You are no longer required to write custom code that does the cast or transformations that are cast between an XmlObject.

For example, instead of the generic XmlObject:

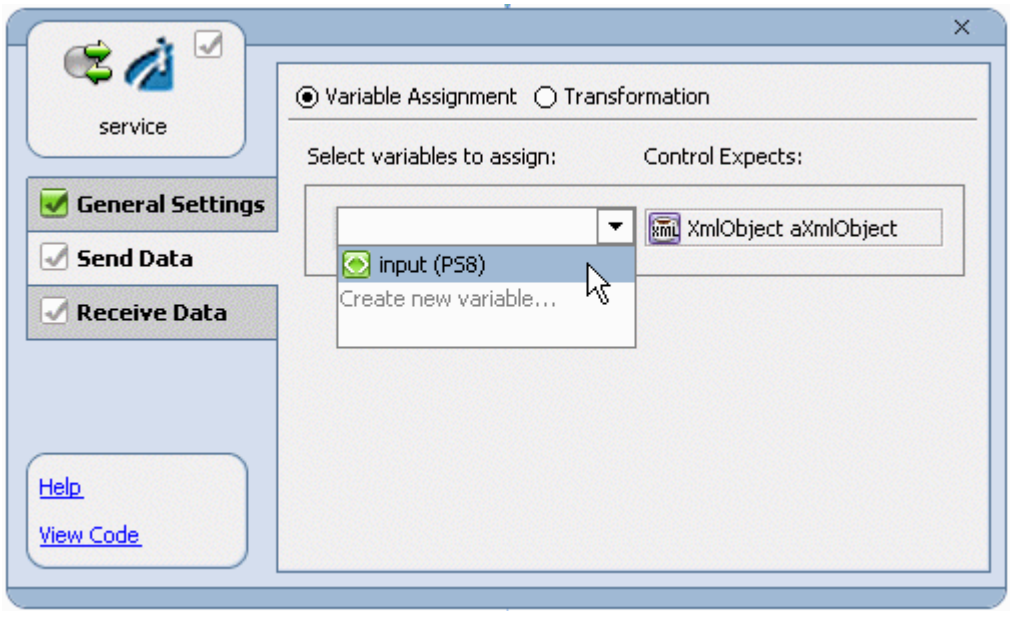
```
XmlObject service(XmlObject input) throws java.lang.Exception;
```

you call:

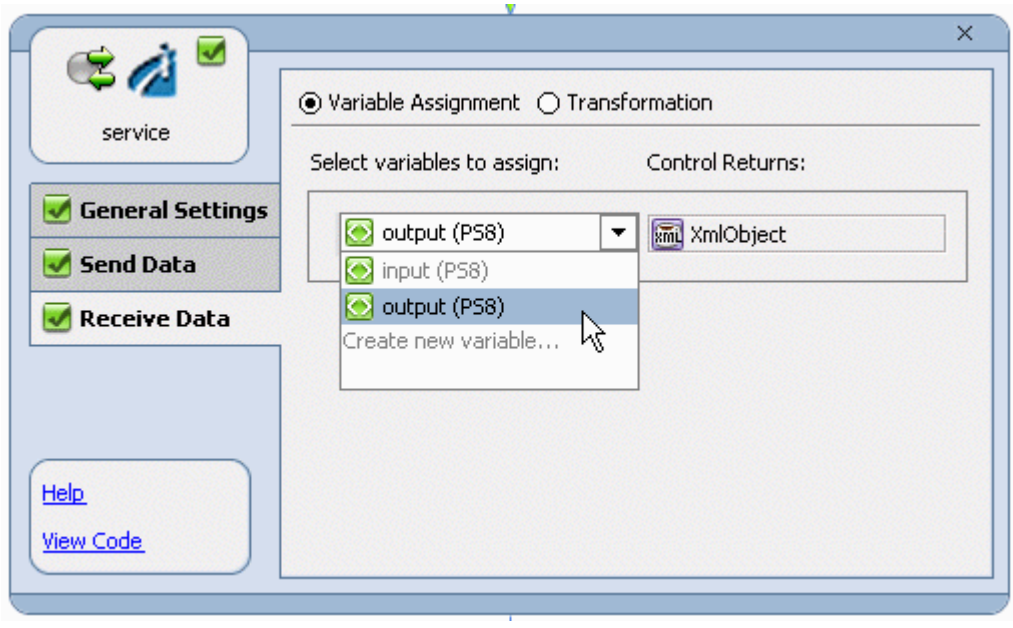
```
public iwaysoftwareAdapterPeoplesoftCiResponse.PS8Document
callpSoft(iwaysoftwareAdapterPeoplesoftCi.PS8Document aRequest) throws
Exception;
```

### *Example* Using Dynamic Class Casting

The following example uses dynamic class casting to specify a schema-based input XmlObject to be casted into a pure XmlObject as a service method, which is expected by the CCI control.



The following example uses dynamic class casting where the CCI control returns a pure XmlObject, which is casted dynamically into a schema-based output XmlObject.



APPENDIX B

# Using Application Explorer for Event Handling

**Topics:**

- Starting Application Explorer in BEA WebLogic Workshop
- Understanding Event Functionality
- Creating an Event Port
- Modifying an Event Port
- Creating a Channel
- Modifying a Channel
- Deploying Components in a Clustered BEA WebLogic Environment

This section describes how to use Java Swing Application Explorer running in BEA WebLogic Workshop to create events in PeopleSoft.

# Starting Application Explorer in BEA WebLogic Workshop

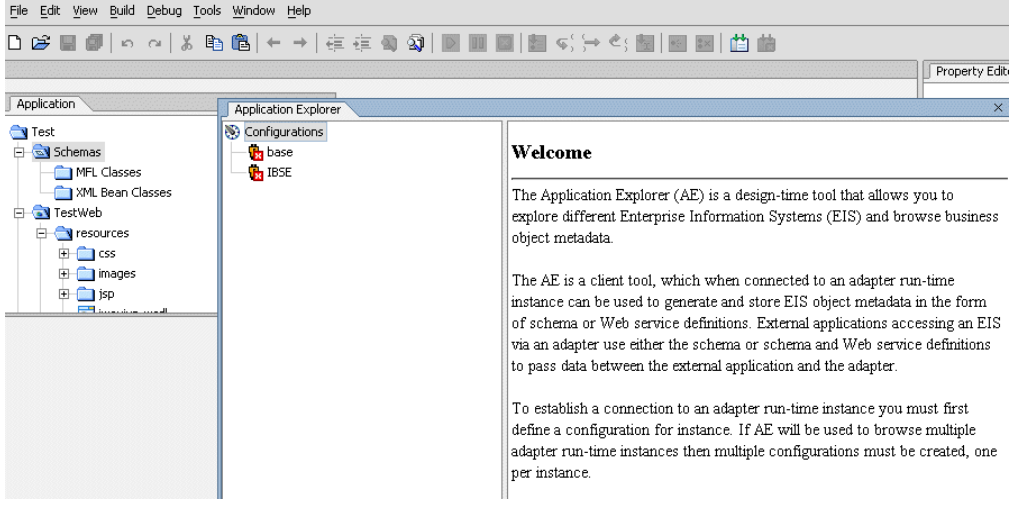
The server must be started where Application Explorer is running. Before you can use Application Explorer, you must start BEA WebLogic server.

## *Procedure* How to Start Application Explorer in BEA WebLogic Workshop

To start Application Explorer running in BEA WebLogic Workshop:

1. Before starting Application Explorer, ensure that BEA WebLogic Server is running.
2. Start BEA WebLogic Workshop.
3. From the BEA WebLogic Workshop View menu, select *Windows* and then, *Application Explorer*.

   Application Explorer opens in BEA WebLogic Workshop.




   You can resize and drag-and-drop the Application Explorer window within BEA WebLogic Workshop. For example, you can drag it to the upper part of BEA WebLogic Workshop.

## 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 must perform in response to activity, your application is a consumer of this event.

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

- Port

  A port associates a particular business object exposed by the adapter with a particular disposition. A disposition defines the protocol and location of the event data. The port defines the end point of the event consumption. For more information, see *Creating an Event Port on page B-3*.

- Channel

  A channel represents configured connections to particular instances of back-end systems. A channel binds one or more event ports to a particular listener managed by the adapter. For more information, see *Creating a Channel* on page B-19.

## Creating an Event Port

The following procedures describe how to create an event port using Application Explorer. The following port dispositions are available:

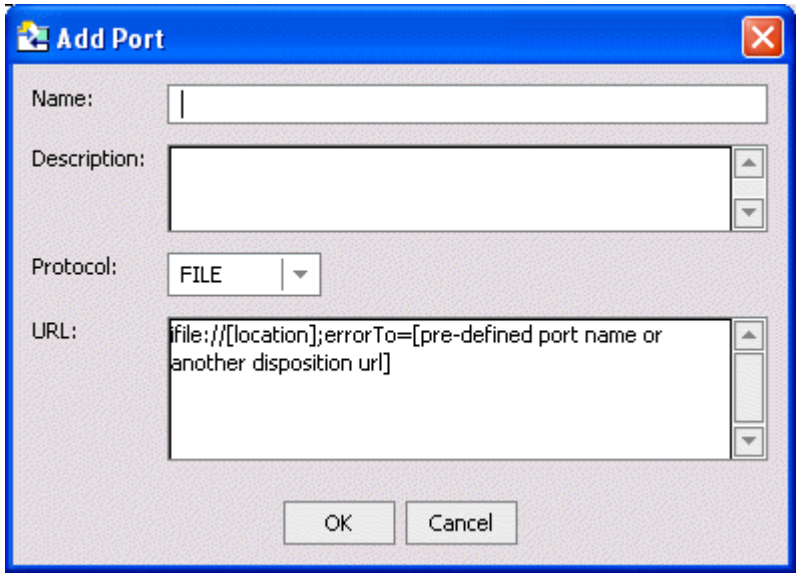
- FILE
- HTTP
- iBSE
- JMSQ
- MQ Series
- MSMQ
- SOAP

## *Procedure* How to Create an Event Port for File

To create an event port for File:

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

   The Add Port dialog box opens.




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

   b. In the Description field, type a brief description.

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

   d. In the URL field, type a destination file to which the 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 describes the URL parameters.

| Parameter | Description |
|---|---|
| location | The full directory path and file name to which the data is written. |
| errorTo | Location where error logs are sent. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

**4.** Click *OK*.

In the left pane, the event port appears below the Ports node.

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

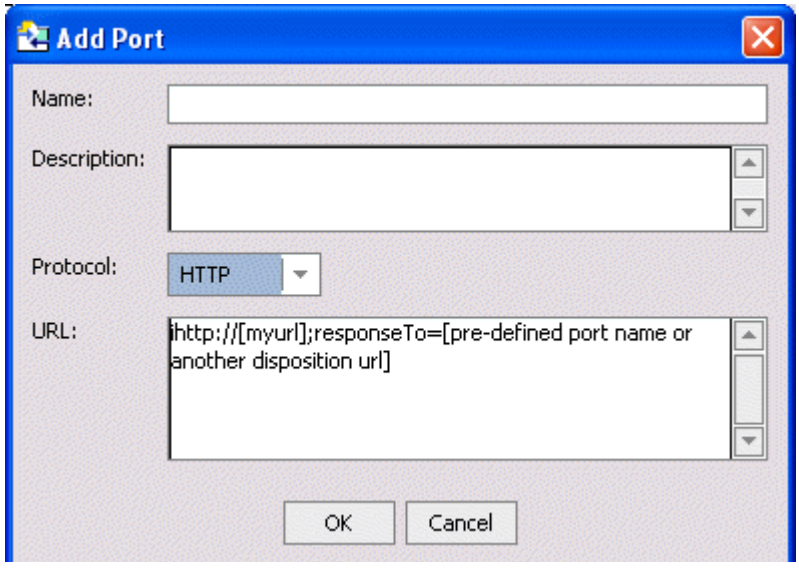
You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

## *Procedure* How to Create an Event Port for HTTP

The HTTP disposition uses an HTTP URL to specify an HTTP end point to which the event document is posted.

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

   The Add Port dialog box opens.




   a. In the Name field, type a name for the event port.
   b. In the Description field, type a brief description.
   c. From the Protocol drop-down list, select *HTTP*.
   d. In the URL field, enter an HTTP destination.

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

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

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

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

The following table describes the URL parameters.

| Parameter | Description |
|---|---|
| url | The URL target for the post operation. |
| respDest | Location where responses are posted. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |
| host | Name of the host on which the Web server resides. |
| port | Port number on which the Web server is listening. |

4. Click *OK*.

   The event port appears below the Ports node in the left pane.

5. To review the port settings, select the port name.

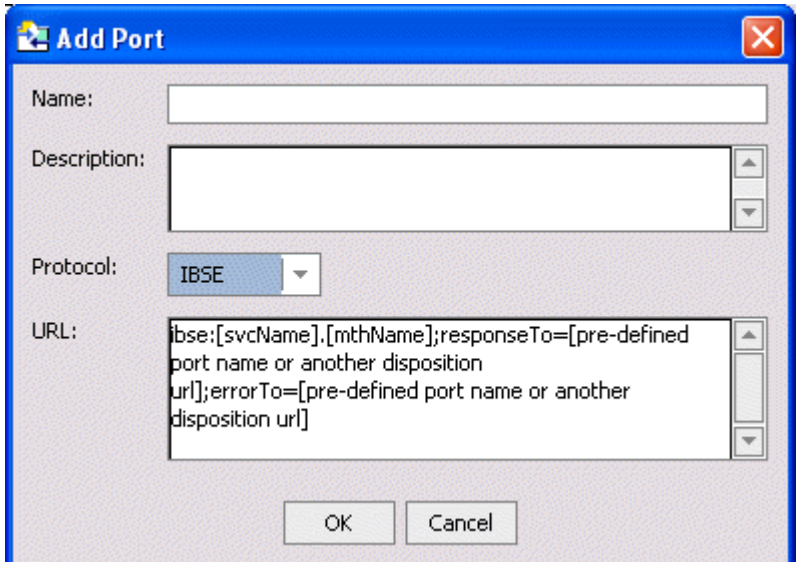
   In the right pane, a table appears that summarizes the information associated with the event port you created. You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

## *Procedure* How to Create an Event Port for iBSE

To create an event port for iBSE:

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

   The Add Port dialog box opens.




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

   b. In the Description field, type a brief description.

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

   d. In the URL field, enter an iBSE destination using the following format:

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

   The following table describes the disposition parameters.

| Parameter | Description |
|---|---|
| svcName | Name of the service created with iBSE. |
| methName | Name of the method created for the Integration Business Service. |

| Parameter | Description |
|---|---|
| respDest | Location where responses to the Integration Business Service are posted. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |
| errorDest | Location where error logs are sent. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

4. Click *OK*.

   The event port appears below the Ports node in the left pane.

5. To review the port settings, select the port name.

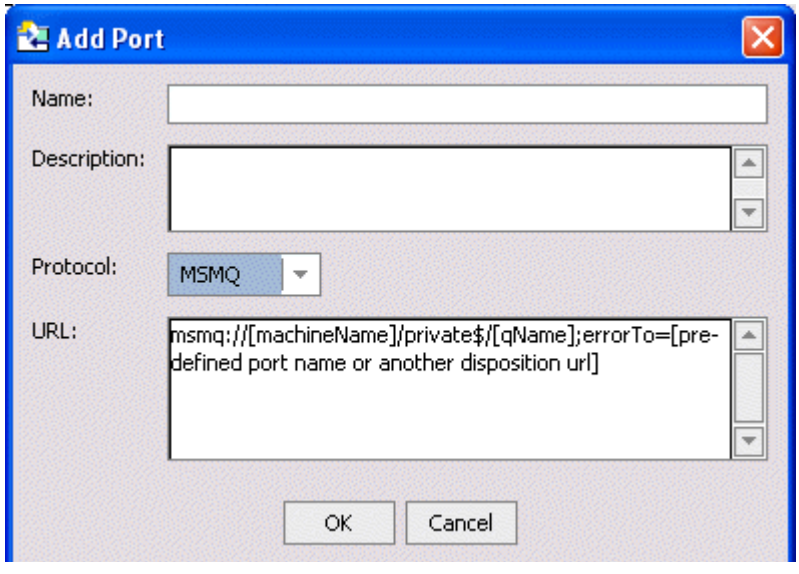
   In the right pane, a table appears that summarizes the information associated with the event port you created. You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

## *Procedure* How to Create an Event Port for MSMQ

To create an event port for a Microsoft Message Queuing (MSMQ) queue:

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

   The Add Port dialog box opens.




   a. In the Name field, type a name for the connection.

   b. In the Description field, type a description for the target name you just created.

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

   d. In the URL field, enter an MSMQ destination in the following format:

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

   The following table defines the disposition parameters.

| Parameter | Description |
|---|---|
| host | Name of the host on which the Microsoft Queuing system runs. |

| Parameter | Description |
|---|---|
| queueType | The type of queue. For private queues, enter *Private$*.<br><br>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. |
| queueName | Name of the queue where messages are placed. |
| errorDest | Location where error logs are sent. Optional.<br><br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

**4.** Click *OK*.

In the left pane, the event port appears below the Ports node.

**5.** To review the port settings, select the port name.

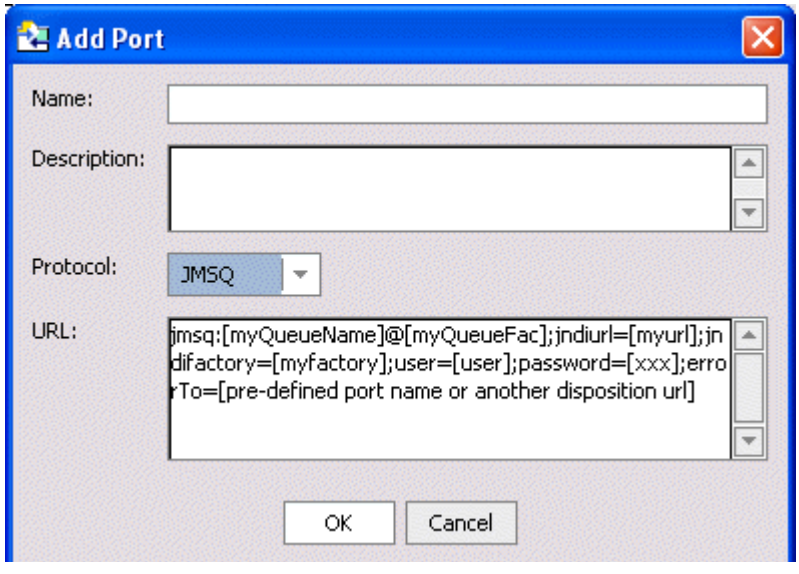
In the right pane, a table appears that summarizes the information associated with the port you created. You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

## *Procedure* How to Create a Port for JMS

To create a port for a JMS queue:

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

   The Add Port dialog box opens.




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

   b. In the Description field, type a brief description.

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

   d. In the URL 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 describes the URL parameters.

| Parameter | Description |
|---|---|
| queue | Name of a queue to which events are emitted. |
| conn_factory | The connection factory, a resource that contains information about the JMS Server. The WebLogic connection factory is:<br>`javax.jms.QueueConnectionFactory` |
| jndi_url | The URL of the application server. For BEA WebLogic Server, the URL is<br>`t3://host:port`<br>where:<br>*`host`*<br>Is the machine name where BEA WebLogic Server resides.<br>*`port`*<br>Is the port on which BEA WebLogic Server is listening. The default port if not changed at installation, is 7001. |
| jndi_factory | Is JNDI context.INITIAL_CONTEXT_FACTORY and is provided by the JNDI service provider. For BEA WebLogic Server, the WebLogic factory is weblogic.jndi.WLInitialContextFactory. |
| userID | User ID associated with this queue. |
| pass | Password associated with this user ID. |
| errorDest | Location where error logs are sent. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

4. Click *OK*.

   The event port appears below the Ports node in the left pane.

5. To review the port settings, select the port name.

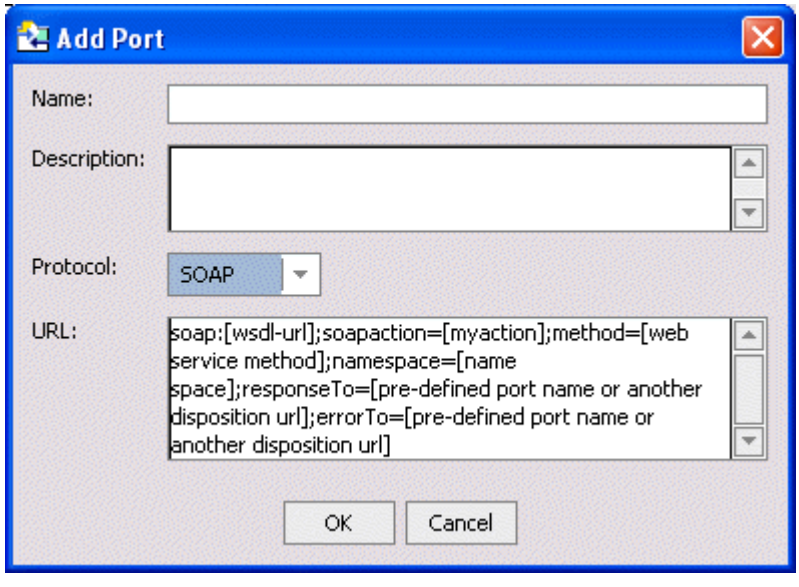
   In the right pane, a table appears that summarizes the information associated with the event port you created. You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

## *Procedure* How to Create a Port for the SOAP Disposition

To create a port for a SOAP disposition:

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

   The Add Port dialog box opens.




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

   b. In the Description field, type a brief description.

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

   d. In the URL 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 defines 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:<br>`http://localhost:7001/ibse/IBSEServlet/test/webservice.ibs?wsdl`<br>where:<br>*webservice*<br>Is the name of the Web service you created using Application Explorer.<br>This value can be found by navigating to the Integration Business Services tab and opening the *Service Description* hyperlink in a new window. The WSDL URL appears in the Address field.<br>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. |
| soapaction | The method that will be called by the SOAP disposition. This value can be found in the WSDL file. |
| 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 | The location to which responses are posted, which can be a predefined port name or another URL. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |
| errorTo | The location to which error logs are sent. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

4. 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.

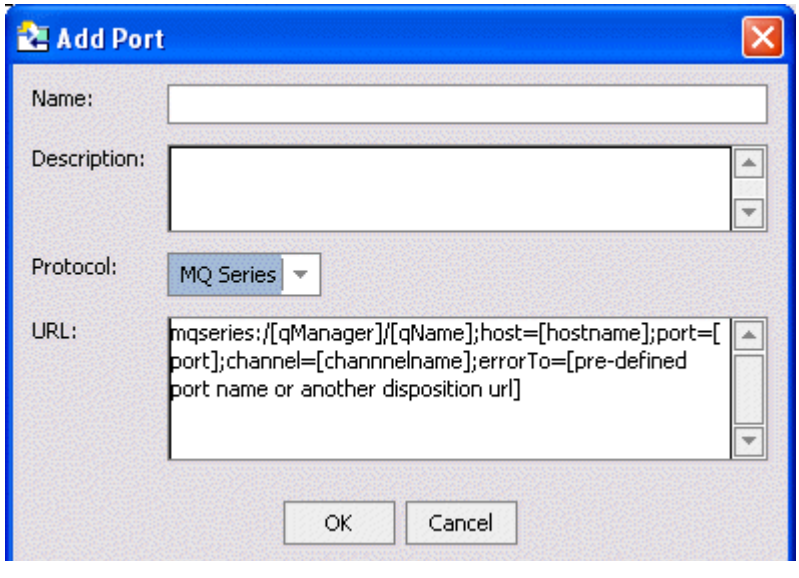
   In the right pane, a table appears that summarizes the information associated with the event port you created. You are ready to associate the event port with a channel. For more information, see *Creating a Channel* on page B-19.

## *Procedure* How to Create an Event Port for MQ Series

The MQ Series disposition allows an event to be enqueued to an MQ Series queue. Both queue manager and queue name may be specified. To create a port for an MQ Series queue:

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

   The Add Port dialog box opens.




   a. In the Name field, type a name for the event port.
   b. In the Description field, type a brief description.
   c. From the Protocol drop-down list, select *MQ Series*.
   d. In the URL field, enter an MQ Series destination.

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

```
mqseries:/qManager/qName;host=[hostname];port=[port];channel=[channnel
name];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 describes the URL parameters.

| Parameter | Description |
|---|---|
| qManager | Name of queue manager to which the server must connect. |
| qName | Name of the queue where messages are placed. |
| hostName | Name of the host on which MQ Series resides (MQ client only). |
| portNum | Port number for connecting to an MQ Server queue manager (MQ client only). |
| chanName | Case-sensitive name of the channel that connects with the remote MQ Server queue manager (MQ client only). The default MQ Series channel name is SYSTEM.DEF.SVRCONN. |
| errorDest | Location where error logs are sent. Optional.<br>A predefined port name or another disposition URL. The URL must be complete, including the protocol. |

4. Click *OK*.

   The event port appears below the Ports node in the left pane.

5. To review the port settings, select the port name.

   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 a Channel* on page B-19.

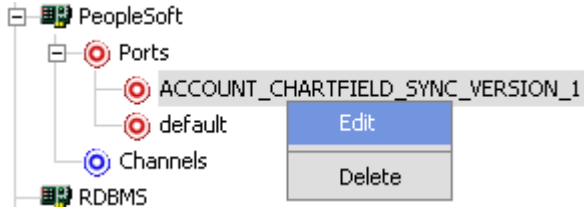
## Modifying an Event Port

The following procedures describe how to edit and delete an event port using Application Explorer. To review the port settings, select the port name. In the right pane, a table appears that summarizes the information associated with the event port you created.

## *Procedure* How to Edit an Event Port

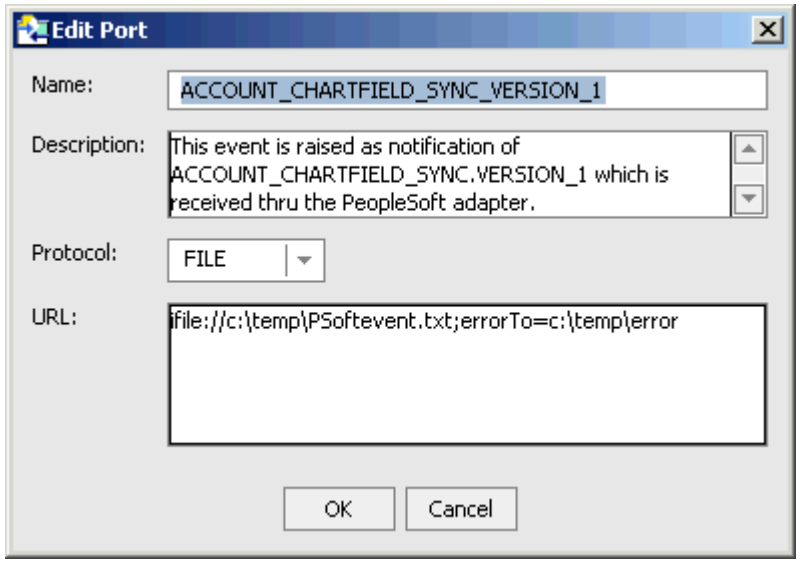
To edit an event port:

1. To view the available ports, click the *Ports* node in the left pane.




2. Right-click the port you want to edit, and select *Edit*.

   The Edit Port dialog box opens.

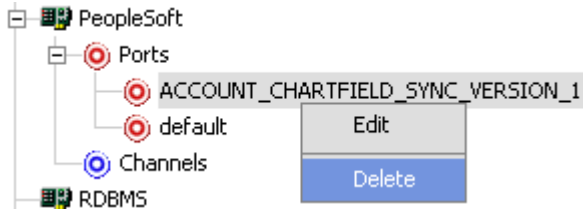



3. Make the required changes and click *OK*.

### *Procedure* How to Delete an Event Port

To delete an existing event port:

1. To view the available ports, click the *Ports* node in the left pane.




2. Right-click the port you want to remove, and select *Delete*.

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

## Creating a Channel

The following procedure describes how to create a channel for a PeopleSoft event. All defined event ports must be associated with a channel.

### *Procedure* How to Create a Channel

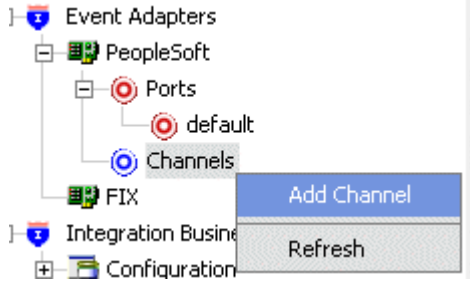
To create a channel:

1. In the left pane, below the configuration you created (for example, SampleConfig) expand the *Event Adapters* node.

   The list of adapters appears.

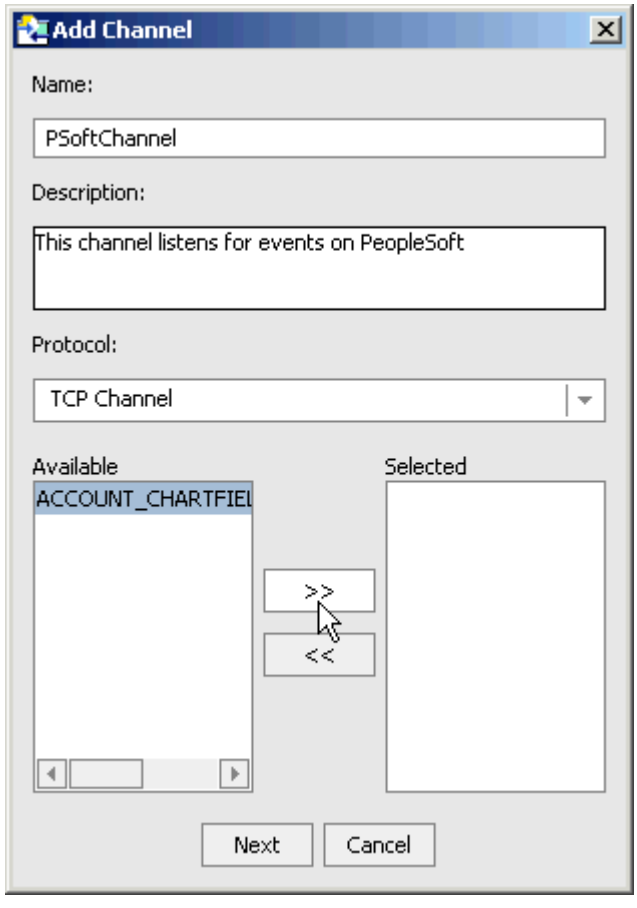
2. Click the adapter node, for example, PeopleSoft.

   The node expands and displays the Ports and Channels nodes.



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

The Add Channel dialog box opens.




**a.** In the Name field, type a name for the channel, for example, PSoftChannel.

**b.** In the Description field, type a brief description.

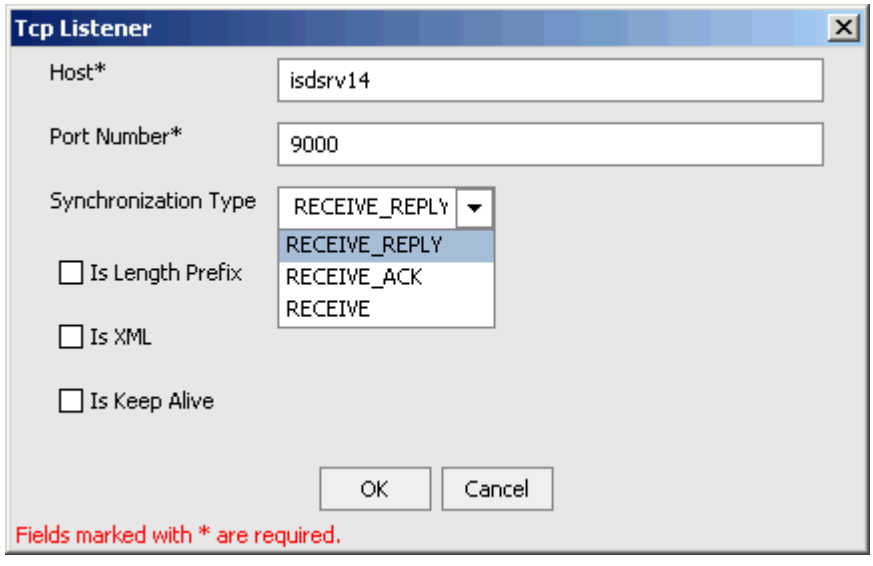
**c.** From the Protocol drop-down list, select *TCP Channel* or *HTTP Channel*.

PeopleSoft supports event handling using TCP and HTTP protocols.

**d.** To associate one or more available ports with this channel, select the port in the Available box and click the double right arrow button to move it to the Selected box.

**4.** Click *Next*.

The TCP Listener dialog box opens.




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

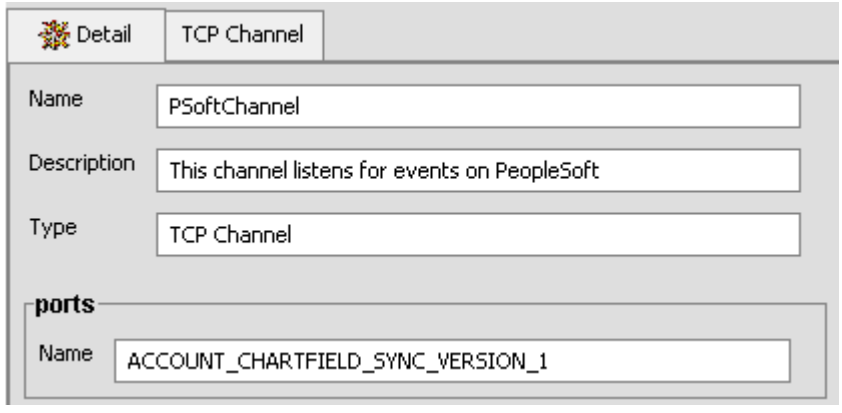
**b.** In the Port Number field, specify the port number for your TCP listener (required).

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

**5.** Click *OK*.

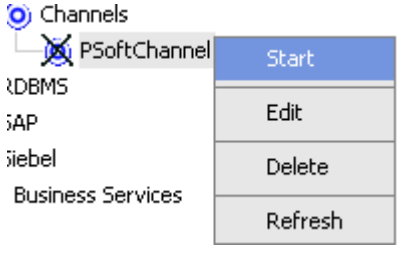
The channel appears below the Channels node in the left pane.

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




A Ports area appears on the Details tab that displays the name of the event port you assigned to this channel.

You are ready to start your channel to listen for events.




6. To activate your event configuration, right-click the channel node, for example, PSoftChannel.
   a. Select *Start*.
   b. To stop the channel at any time, right-click the channel and select *Stop*.

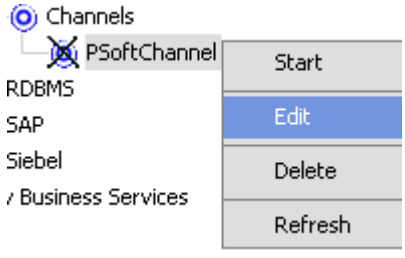
## Modifying a Channel

The following procedures describe how to edit and delete a channel using Application Explorer. To review the channel settings, you select the channel name. In the right pane, a table appears that summarizes the information associated with the channel you created.

## *Procedure* How to Edit a Channel

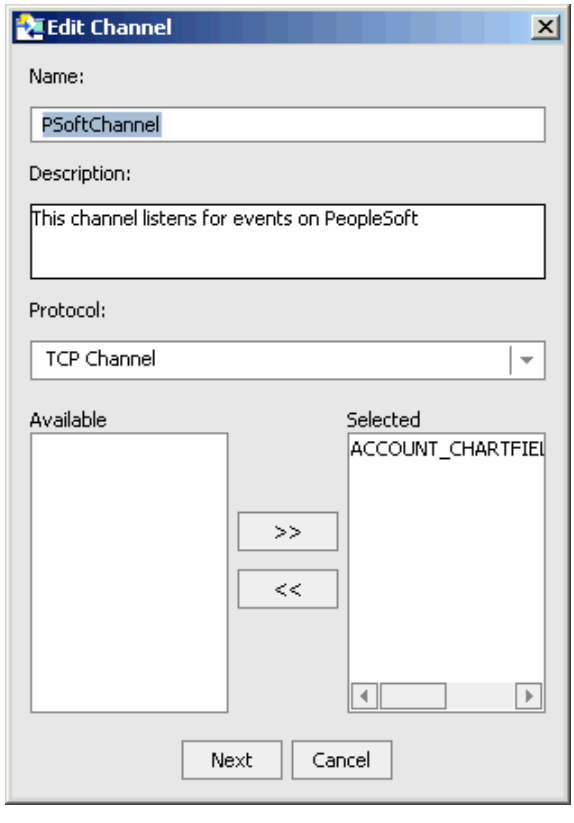
To edit a channel:

1. To view the available channels, click the *Channels* node in the left pane.




2. Right-click the channel you want to edit, for example, PSoftChannel, and select *Edit*.

   The Edit Channel dialog box opens.




3. Make the required changes to the channel configuration.

4. Click *Next*.

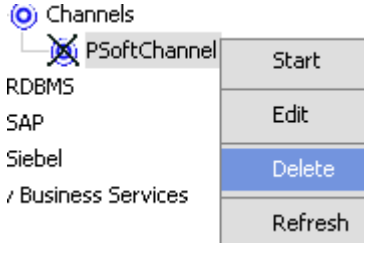
   The Message Server dialog box opens with three tabs, System, User, and Advanced. If changes are required, activate the appropriate tab and make the required changes.

5. Click *OK*.

### *Procedure* How to Delete a Channel

To delete an existing channel:

1. In the left pane, right-click the channel, for example, PSoftChannel.




2. Select *Delete*.

   The channel disappears from the Channels list.

## Deploying Components in a Clustered BEA WebLogic Environment

Events can be configured in a clustered BEA WebLogic environment. You can deploy iBSE or JCA to this environment. This topic uses iBSE as an example, but you can follow the same procedures when deploying JCA. The only difference is that you need to deploy the JCA connector .RAR file to the clustered 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.

Clustering provides the following benefits:

- Load balancing
- High availability

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.

## *Procedure* How to Deploy Components in a Clustered Environment

To deploy components in a clustered environment:

1. Using the BEA Configuration Wizard:
   a. Configure an administrative server to manage the managed servers.
   b. Add and configure as many managed servers as required.
   c. Add and configure an HTTP router. This does not have to be a part of WebLogic and can be an outside component.
   d. If you configure the HTTP router within WebLogic, start it by entering the following command:

      `StartManagedWebLogic HTTPROUTER http://localhost:7001`

      where:

      *HTTPROUTER* is the name of the server on which the HTTP router is running.

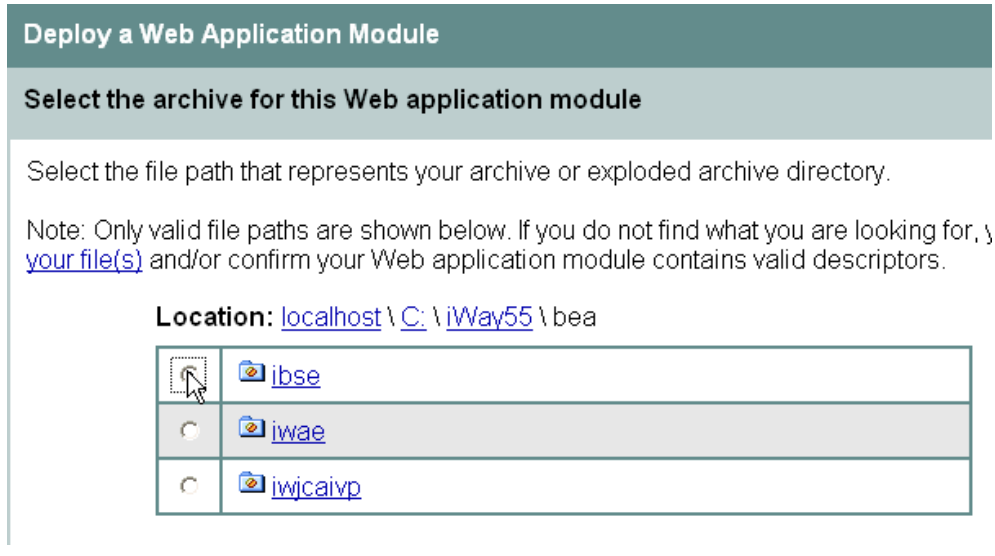
      `http://localhost:7001` is the location of the admin console.

   e. Add the managed servers to your cluster/clusters.

   For more information on configuring WebLogic Integration for deployment in a clustered environment, see *Deploying WebLogic Integration Solutions*.
2. Start the WebLogic Server and open WebLogic Server Console.
3. Deploy iBSE to the cluster by selecting *Web Application Modules* from the Domain Configurations section, and clicking *Deploy a new Web Application Module*.

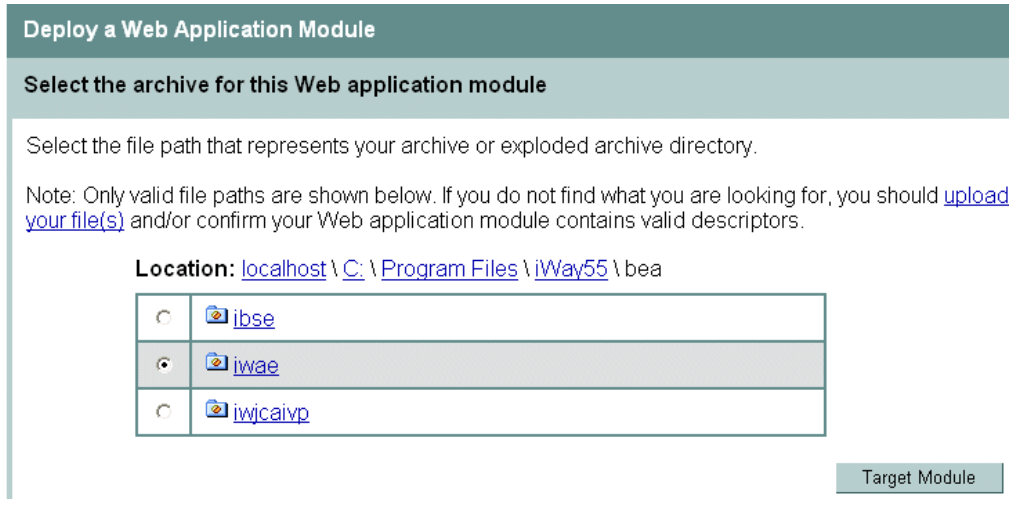
A page appears for you to specify where the Web application is located.

4. To deploy iBSE, select the option button next to the ibse directory and then click *Target Module*.

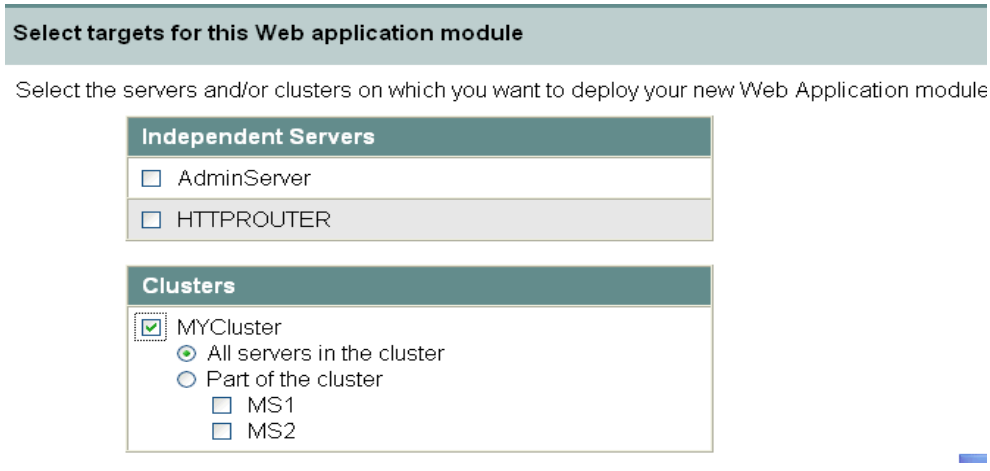



5. To deploy Servlet Application Explorer, select the option button next to the iwae directory and then click *Target Module*.

   If you are using Servlet Application Explorer, deploy it only on the admin server or one of the managed servers.

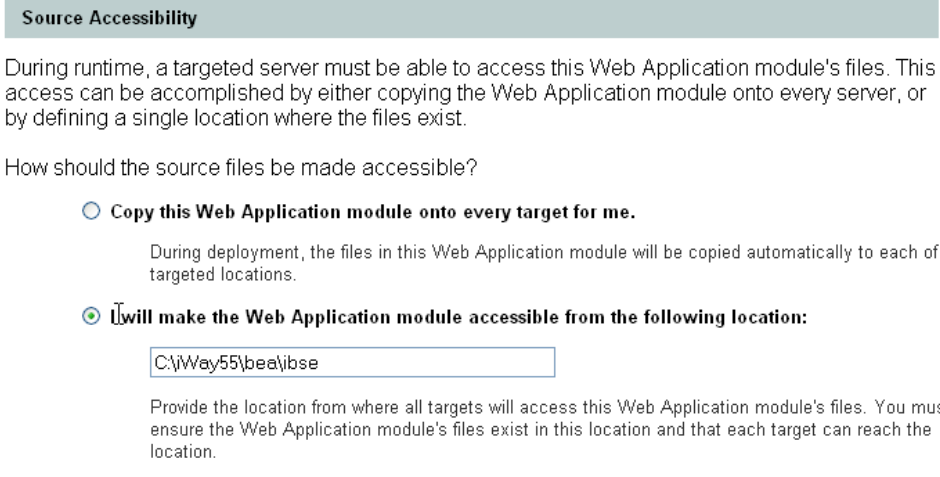


The following window opens.




**6.** Select the servers and/or clusters on which you want to deploy the application and click *Continue*.

The following window opens.




**7.** Select the *I will make the Web Application module accessible from the following location* option button and provide the location from which all targets will access iBSE.

It is recommended that you use a single instance of iBSE, rather than copying iBSE onto every target.

**Note:** iBSE must use a database repository (SQL or Oracle). Do not use a file repository. You can select this in the Repository Type drop-down list in the iBSE monitoring page. After configuring a database repository, you must restart all of the managed servers.

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

where:

*hostname*

Is where your application server is running. Use the IP address or machine name in the URL; do not use localhost.

*port*

Is the port specific to each server, since you deploy iBSE to an entire cluster. For example, 8001, 8002, or any other port that is specified for each managed node.

8. Click *Deploy*.

## *Procedure* How to Configure Ports and Channels in a Clustered Environment

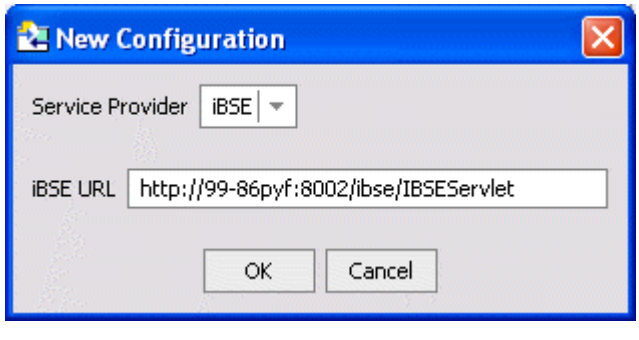
You can use Swing Application Explorer deployed in BEA WebLogic WorkShop or Servlet Application Explorer to configure ports and channels in a clustered environment.

**Note:** Before using Servlet Application Explorer in a clustered environment, you must edit the web.xml file and specify the correct URL to your iBSE deployment. The default location on Windows is:

```
C:\Program Files\iWay55\bea\iwae\WEB-INF\web.xml
```

For more information on configuring the web.xml file for Servlet Application Explorer, see the *BEA WebLogic ERP Adapter Installation and Configuration* documentation.

1. Open Swing Application Explorer in BEA WebLogic Workshop.
2. Create a new connection to the iBSE instance. For information on creating a new configuration, see *Understanding Event Functionality on page B-3*.

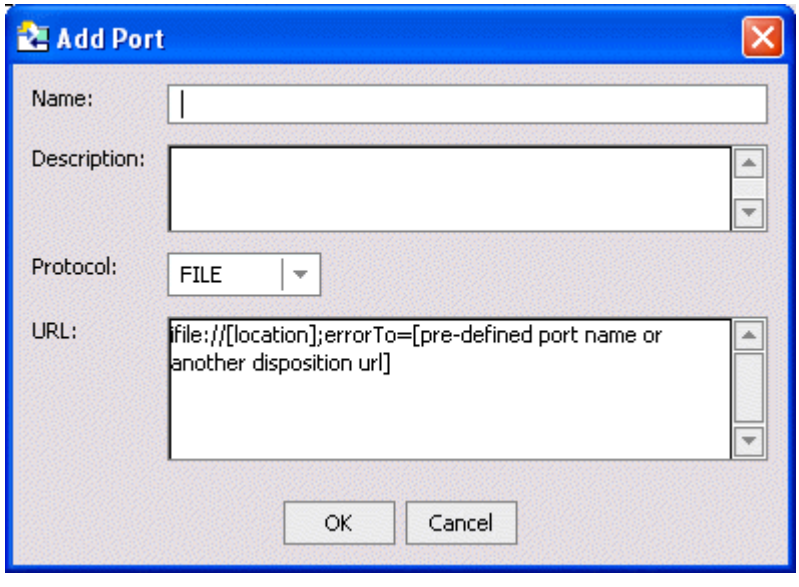



**Note:** Use the IP address or machine name in the URL; do not use localhost.

3. Connect to the new configuration and select the Events Adapters node in the left pane of Application Explorer.


4. Add a new port for the BEA WebLogic Adapter for PeopleSoft. For more information, see *Creating an Event Port on page B-3*.

   The Add Port dialog box opens.

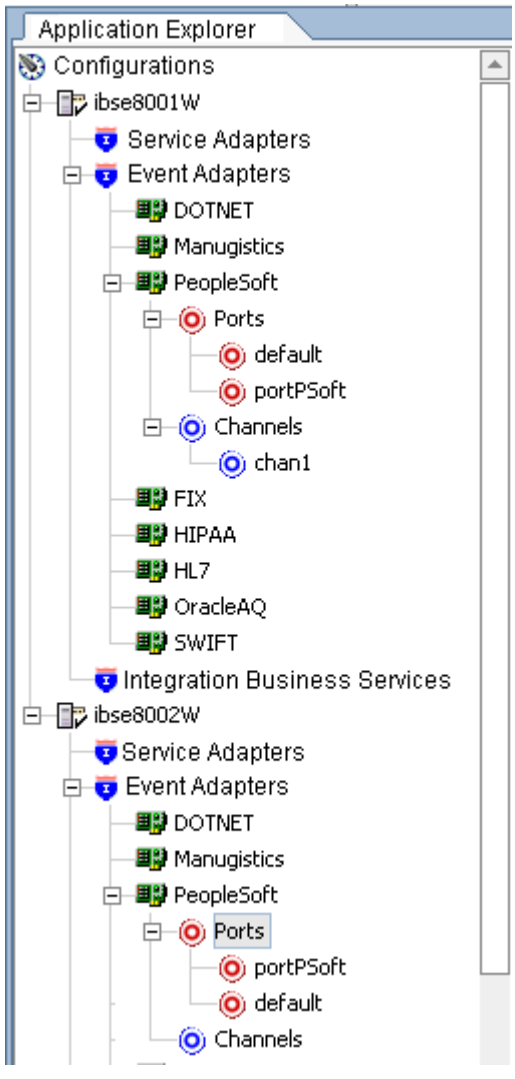



5. Create a channel and add the port you created. For more information, see *Creating a Channel* on page B-19.
6. Click *Next* and enter the application server parameters.
7. Start the channel.

**8.** Create a new configuration and connect to the second iBSE instance.

The connection to iBSE must be configured to each instance of the managed server.

The following image shows two configurations.




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.

APPENDIX C

# Using Component Interfaces

**Topics:**

- Creating a Component Interface
- Viewing or Changing 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 properties by clicking the property and then clicking again until you can type a new name. If you rename a property, it can be referenced in the Component Interface only by the new name, not by the underlying component name.

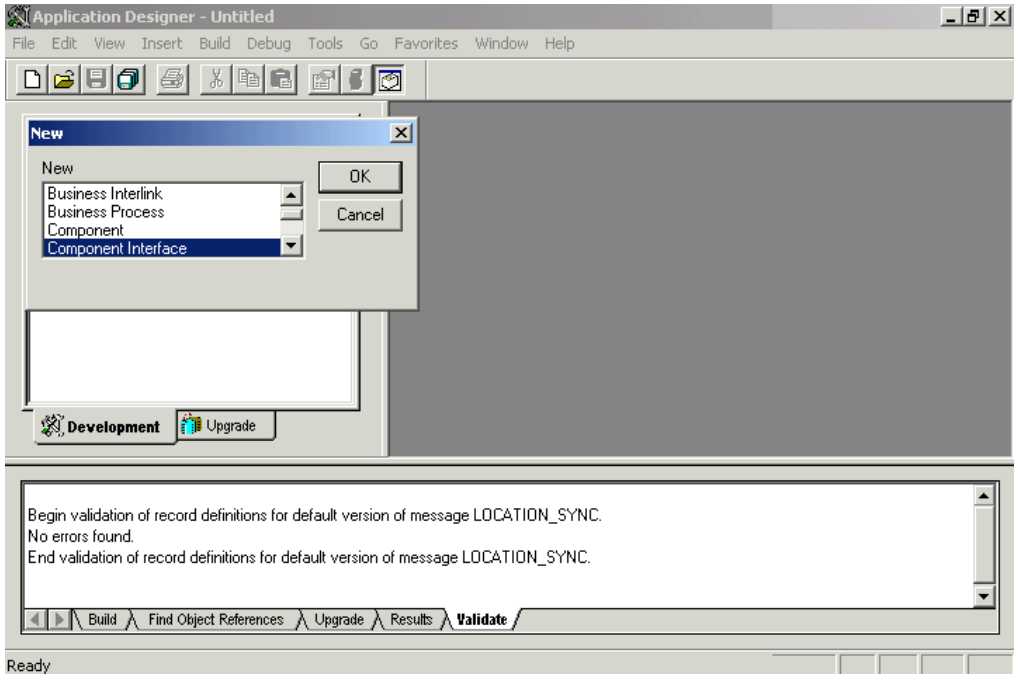
Properties may have various icons adjacent to them. For example, EMPLID has an icon indicating that it is a key field from the underlying record. NAME has an icon indicating that it is an alternate key field from the underlying record. For a complete list of property icons, see the PeopleBooks documentation.

## Procedure: How to Create a New Component Interface

To create a Component Interface:

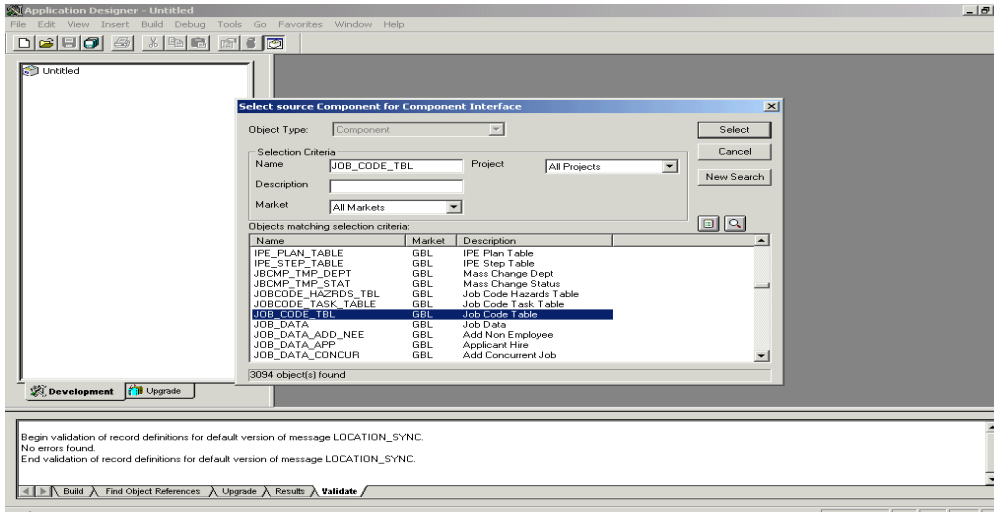
1. Open the PeopleSoft Application Designer.
2. From the File menu, select *New*.

The following image shows the PeopleSoft Application Designer New dialog box. Component Interface is selected. It also contains OK and Cancel buttons.



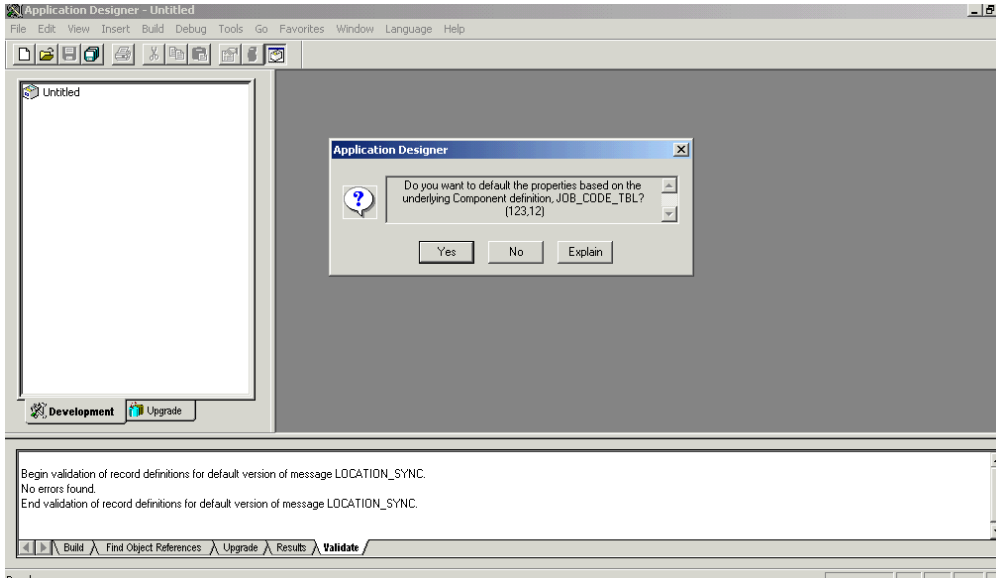

3. Select *Component Interface*.
4. Click *OK*.

The following image shows the Select Source Component for Component Interface dialog box. It contains the following: Object Type list (unavailable), Selection Criteria pane containing a Project list, 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 following image shows the Application Designer dialog box, containing Yes, No, and Explain buttons.




**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*.

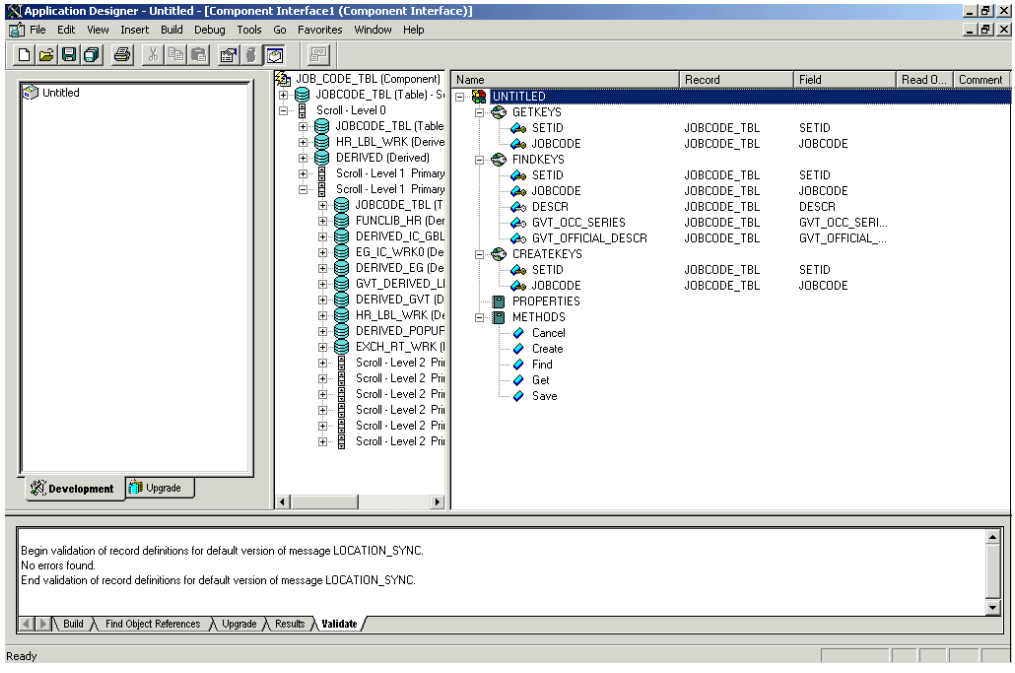
**a.** Drag the relevant fields from the left pane to the right pane.

**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 display the properties of the underlying Component Interface,** click *Yes*.

The following image shows the Application Designer component interface and underlying properties (Name, Record, Field, Read, and Comments).




# Viewing or Changing Available Methods

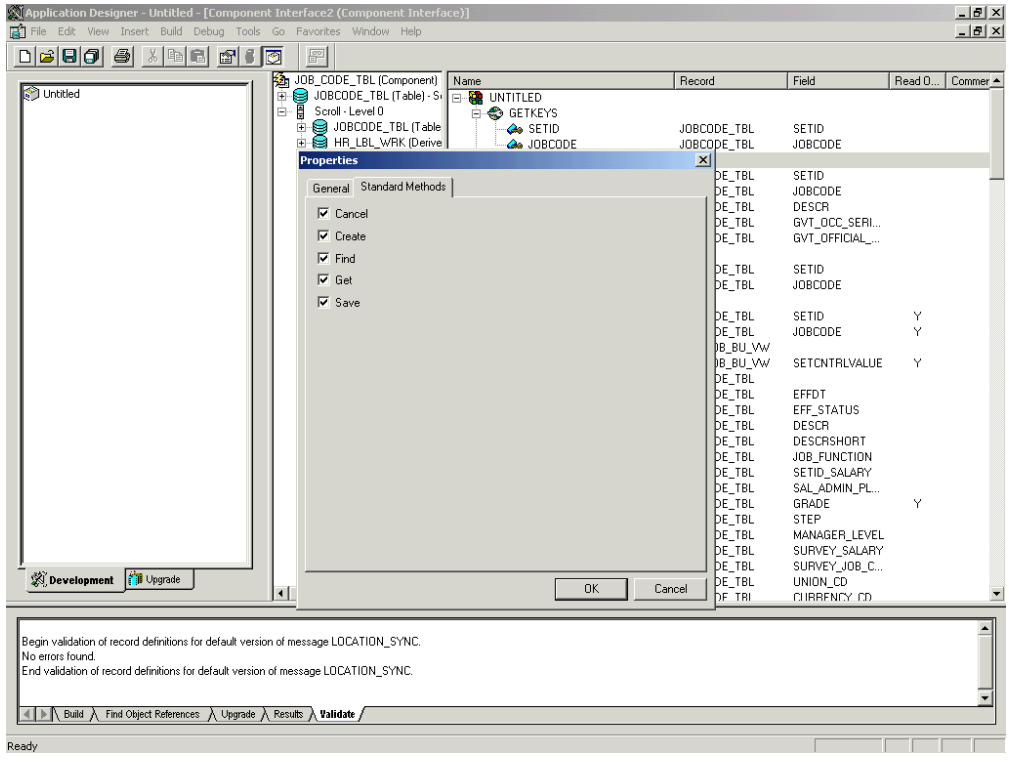
The standard methods for the Component Interface are:

- Create
- Find
- Get
- Save

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

### Procedure: How to View or Change Available Methods

The following image shows the open Component Interface Properties dialog box.




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.

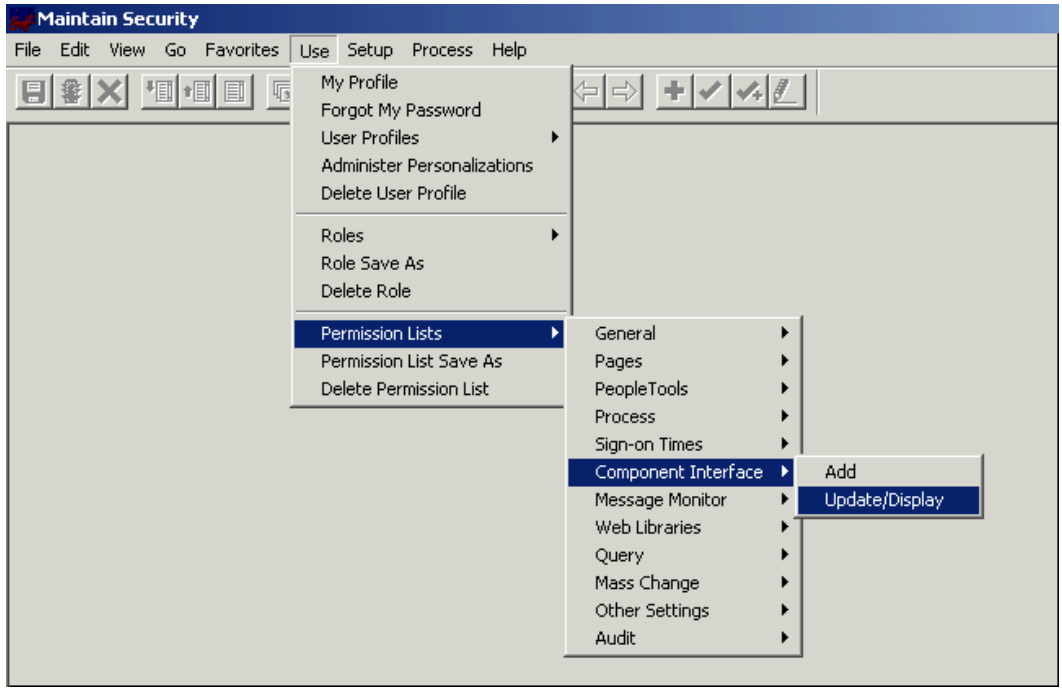
## Securing a Component Interface

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

For information on configuring security for PeopleSoft Version 8.1 in two- and three-tier mode, see *How to Configure Component Interface Security for PeopleSoft Version 8.1* on page C-8. For information on configuring security for PeopleSoft Version 8.4, see *How to Configure Interface Security for PeopleSoft Version 8.4* on page C-12.

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

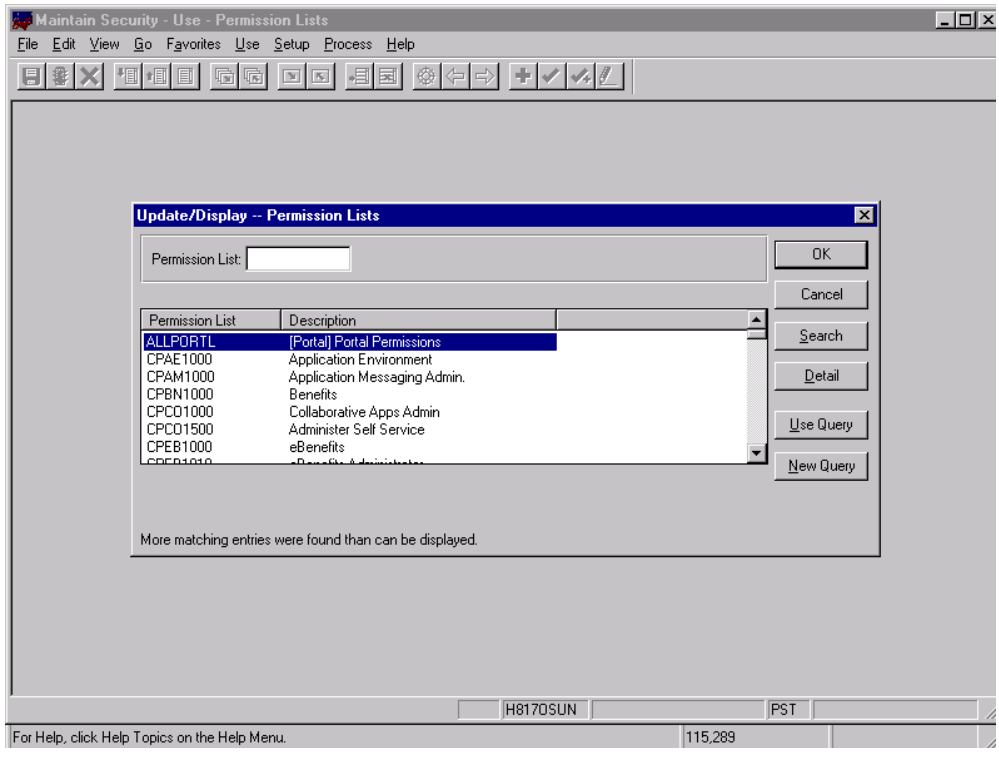
The following image shows the Component Interface Properties dialog box. It contains a General and Standard Methods tab (selected), and Cancel, Create, Find, Get, and Save check boxes.



To configure Component Interface security:

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

   The following image shows the Permission Lists dialog box. It contains a Permission List field, Permission List and Description information, and OK, Cancel, Search, Detail, Use Query, and New Query buttons.

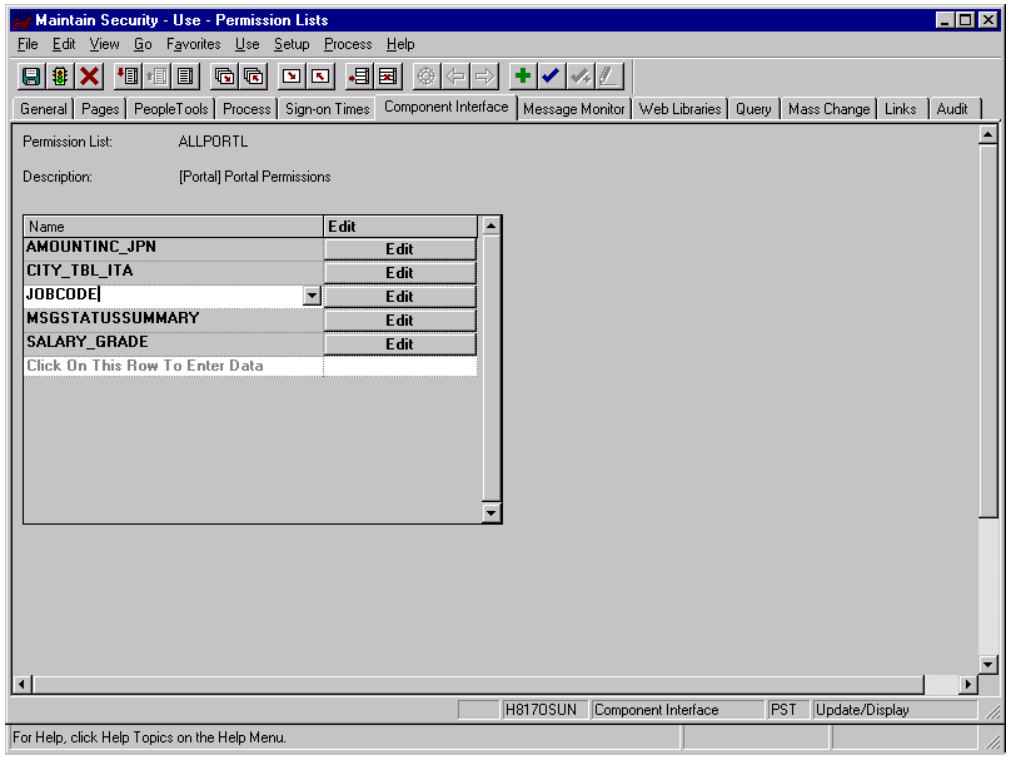



   Before Security can be set, you must identify the permission list(s).

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

   For information on permission lists, see the PeopleBooks documentation.

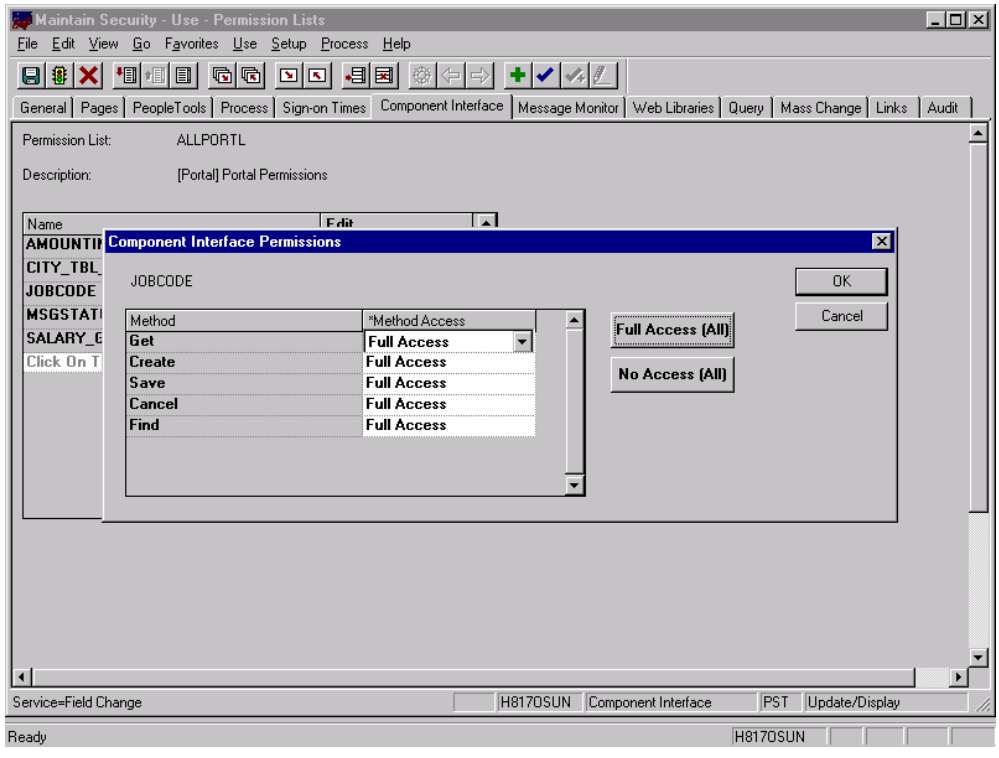
The following image shows the Permission Lists pane and contains information about the ALLPORTL permission list. The Component Interface tab is selected.




3. Insert the new Component Interface that you created.
4. Click *Edit*.

   When you select the Component Interface, all available methods appear, including user-defined methods. You can 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.



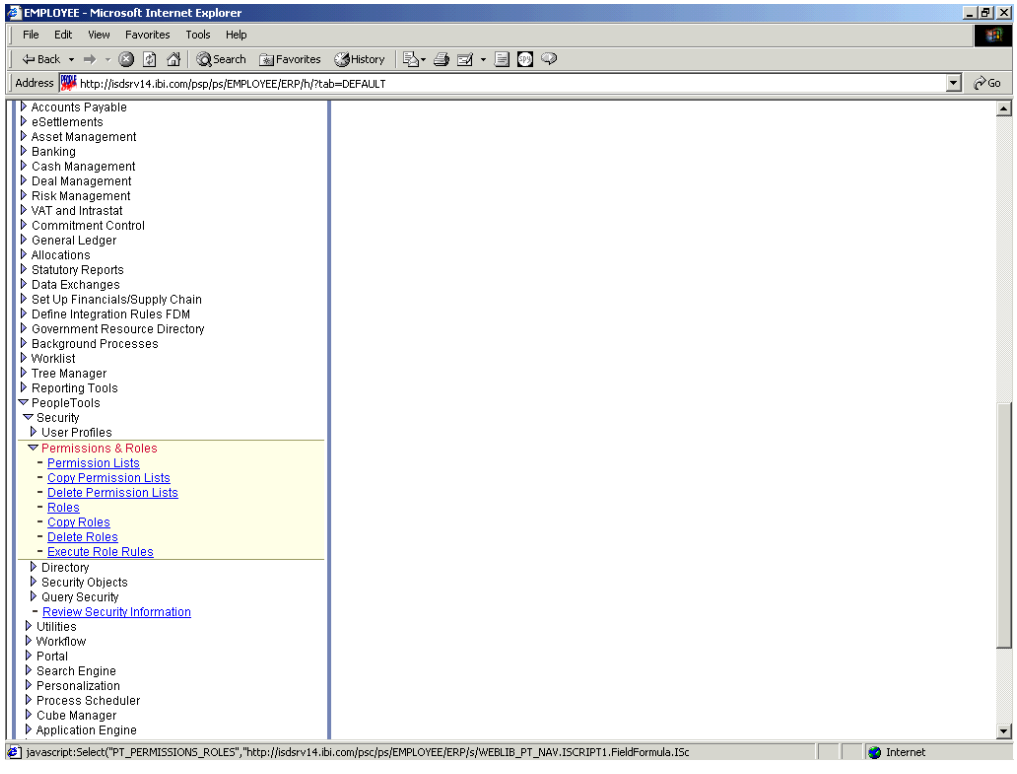

5. Select the desired level of access.
6. Click *OK*.

## Procedure: How to Configure Interface Security for PeopleSoft Version 8.4

To configure interface security:

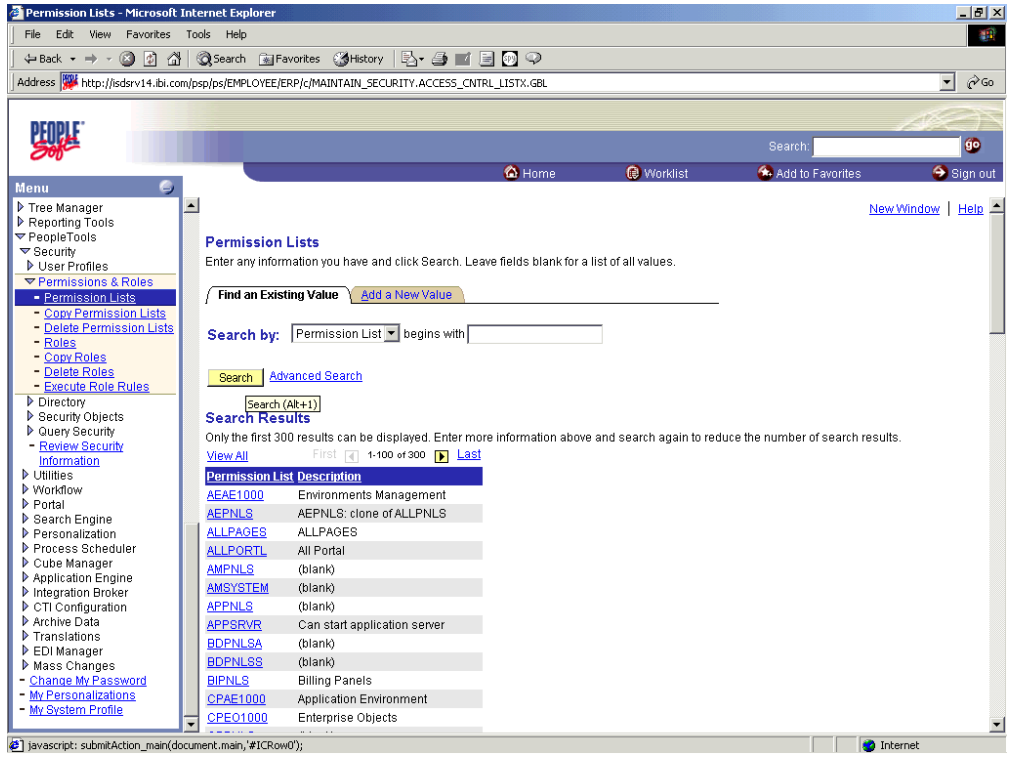
1. Expand *PeopleTools*, *Security*, *User Profiles*, and *Permissions & Roles* and then, click *Permission Lists*.

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



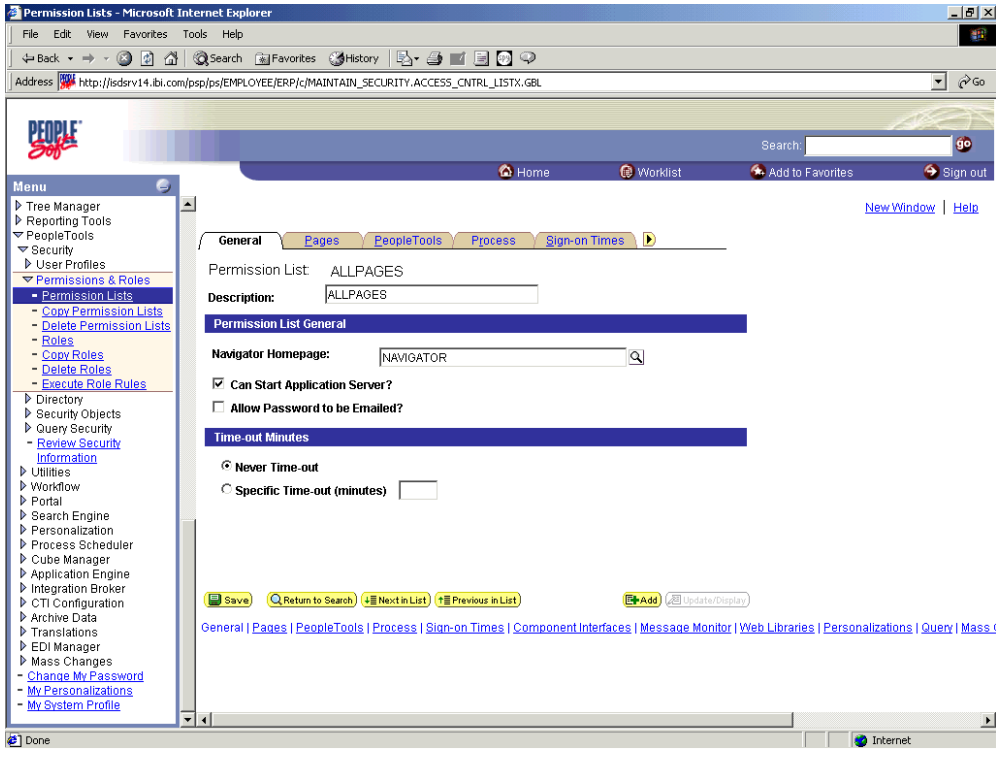
**2.** Click *Search*.

The following image shows the Permission list pane and contains information for the ALLPAGES permission list. It contains Description and Navigator Homepage fields, Can Start Application Server and Allow Password to be Emailed check boxes, and Never Time-out and Specific Time-out option buttons.



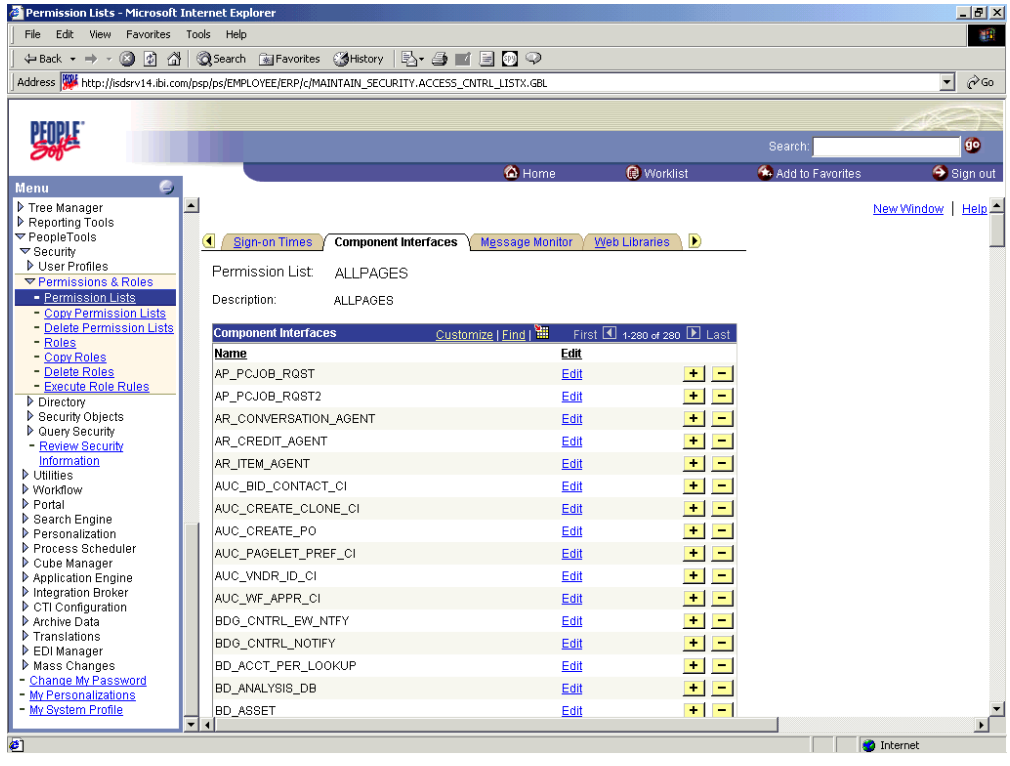
**3.** Select the relevant permission list.

The following image shows the Permission list pane and contains information for the ALLPAGES permission list. It contains Description and Navigator Homepage fields, Can Start Application Server and Allow Password to be Emailed check boxes, and Never Time-out and Specific Time-out option buttons.



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

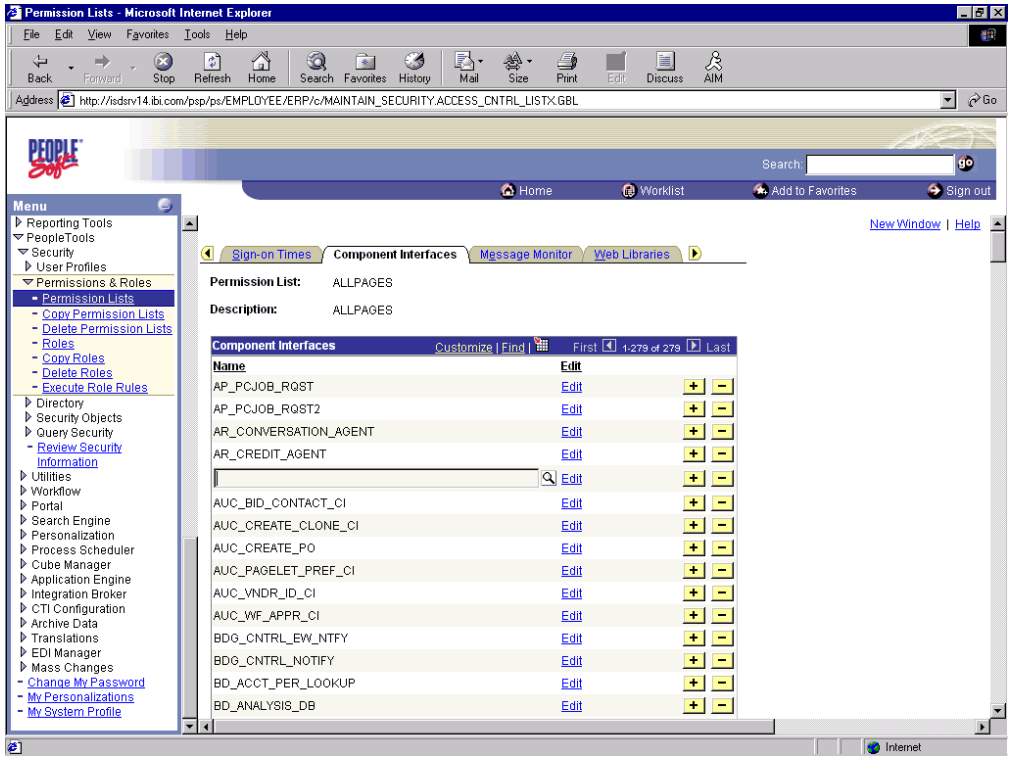
The following image shows the Component Interfaces tab and lists component interfaces for the ALLPAGES permission list.




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

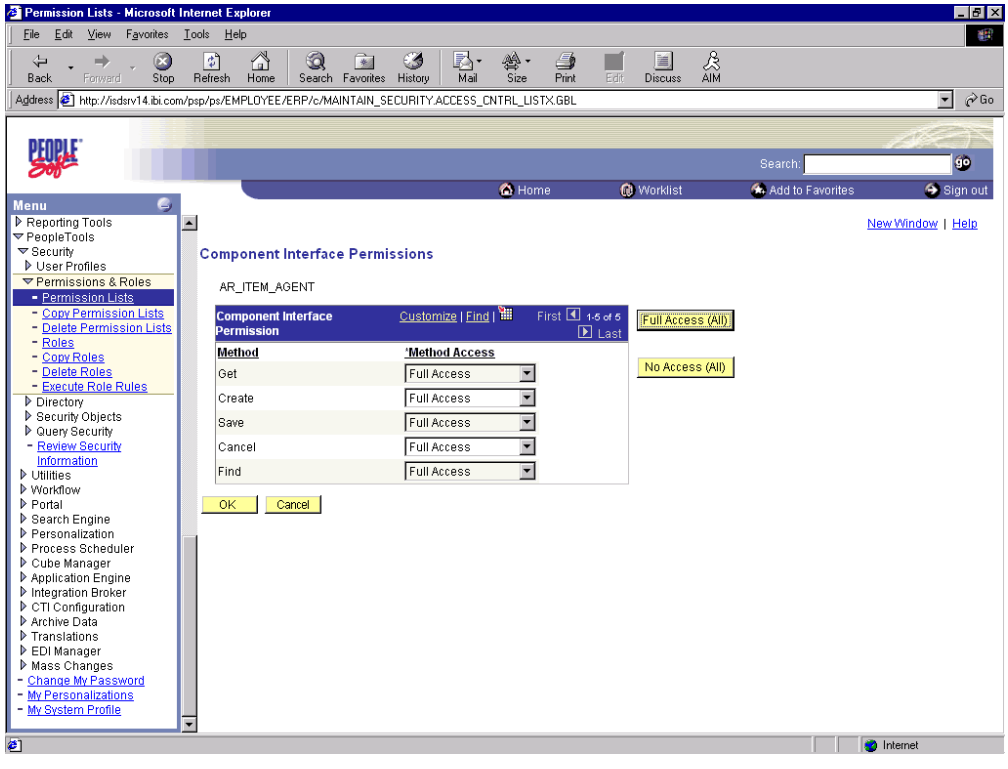
**6.** To add a new row to the Component Interfaces list, click the + button.

The following image shows the Component Interfaces tab and lists component interfaces for the ALLPAGES permission list. It includes a component interface name field.



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

   The following image shows the Component Interface Permissions window for the component interface AR_ITEM_AGENT. It includes the following Method Access lists: Get, Create, Save, Cancel, and Find.




8. From the drop-down lists, select the desired access level for each method.
9. Click *OK*.
10. 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.

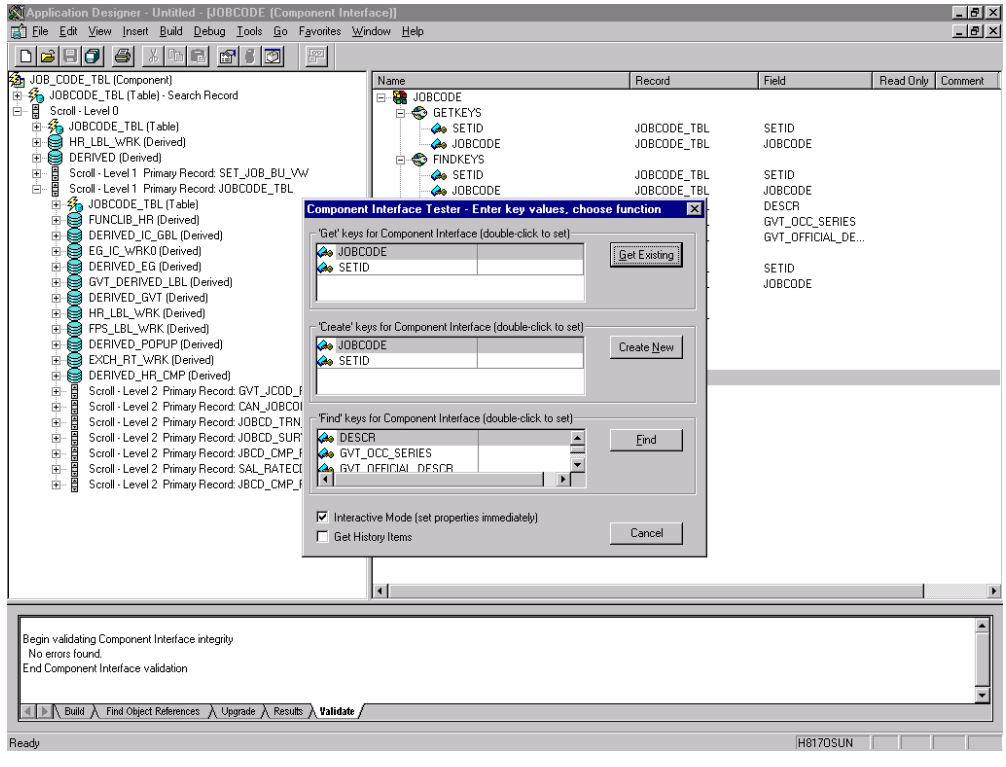
If changes are made to the underlying application by the user or by a PeopleSoft upgrade and the changes invalidate a Component Interface, the user must repair the invalid Component Interface before the adapter uses it.

## 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 following image shows the Component Interface Tester dialog box. 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 Interactive Mode and Get History Items check boxes.

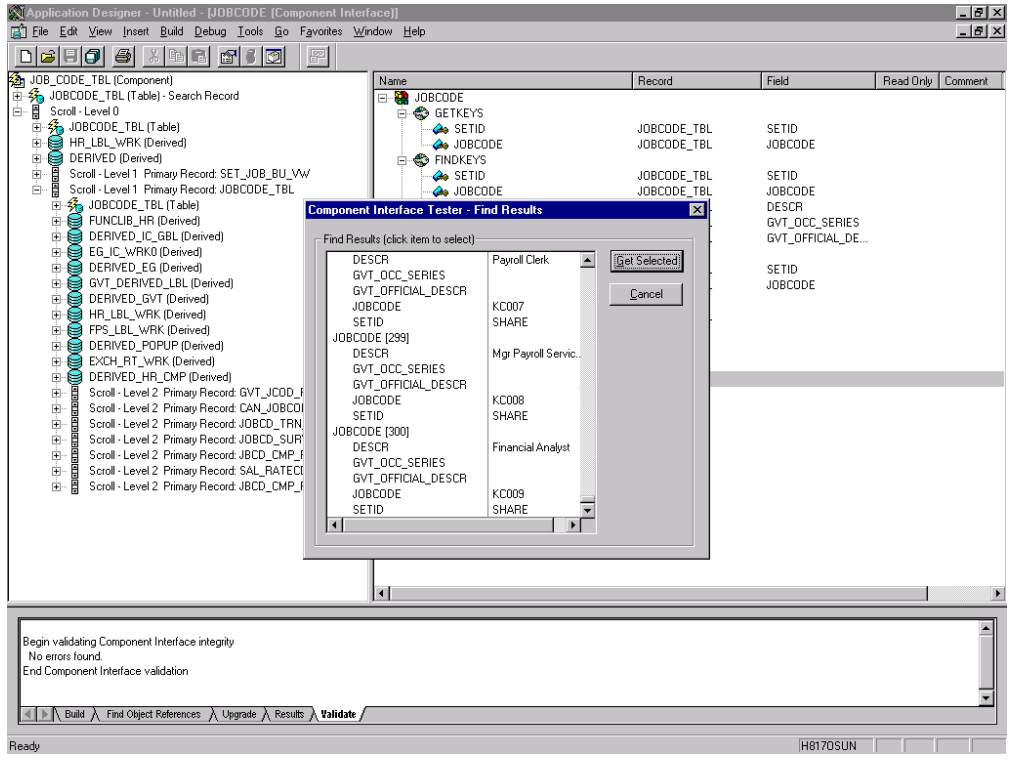



**2.** To test the Component Interface, use one the following methods:

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

The Component Interface Tester - Find Results dialog box opens, displaying all of the possible entries for the underlying component.

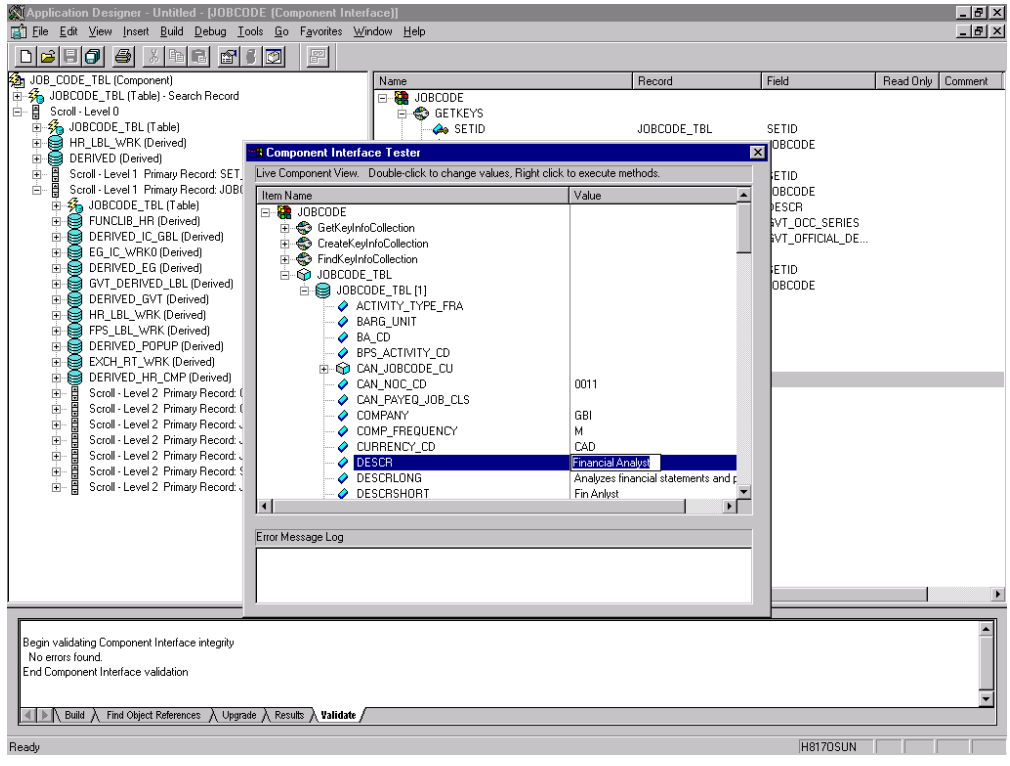
If there are more than 300 entries, a message appears as shown in the following image.




**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.

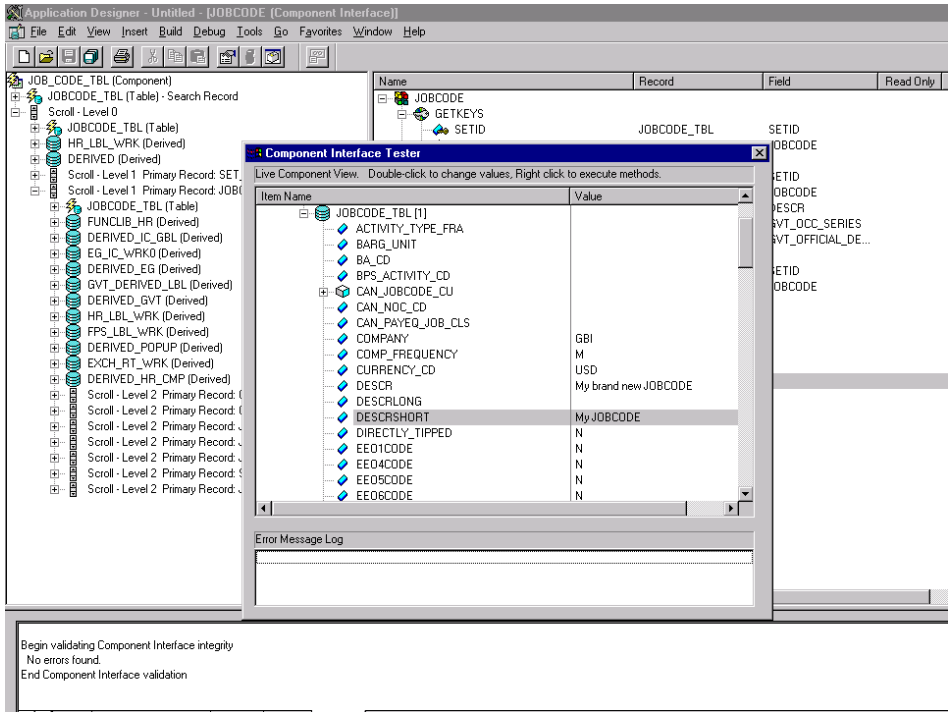
Alternatively, you can test using the Create method.

**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 as shown in the following image, a pane that displays the JOBCODE data opens after the Table name is expanded with default data in place. At this point, you can change fields.



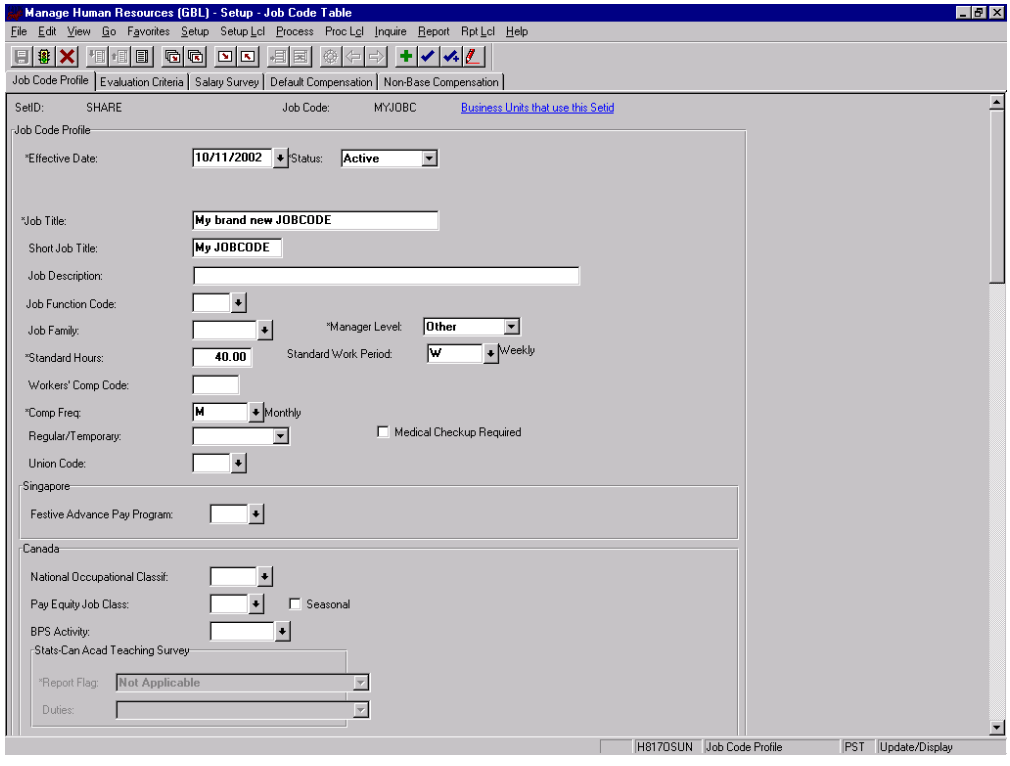

Changes are validated against the underlying business logic of the component.

**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 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 D

# 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 D-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.

- **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.

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.

## 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.

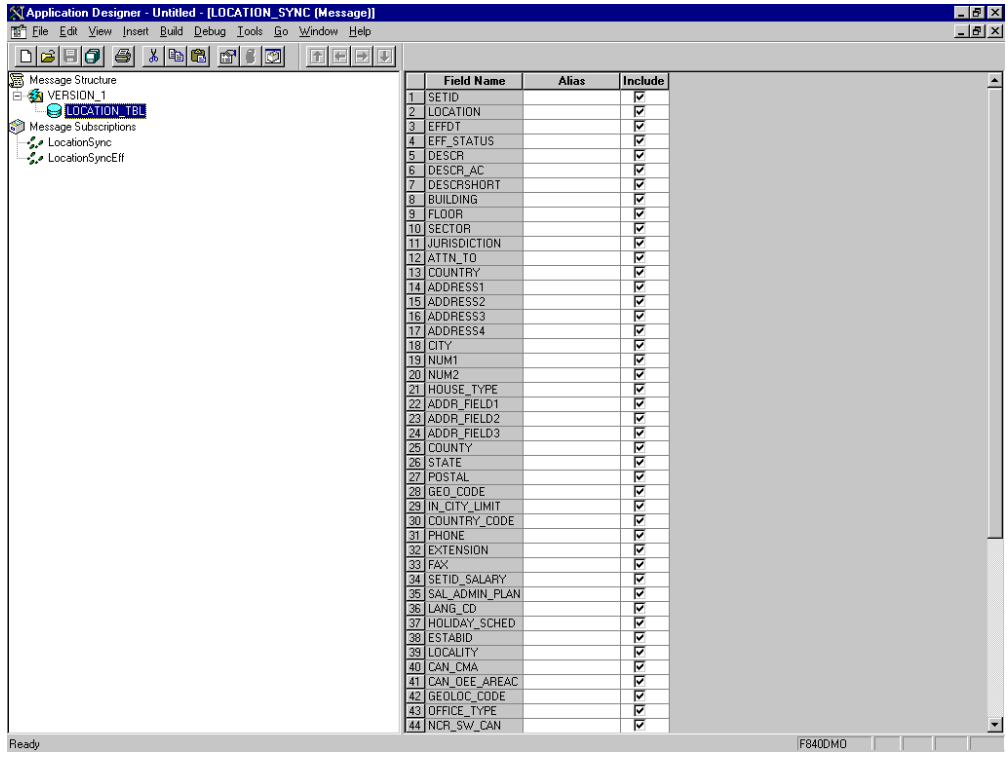
These tasks are described in detail in the following procedures.

### 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:

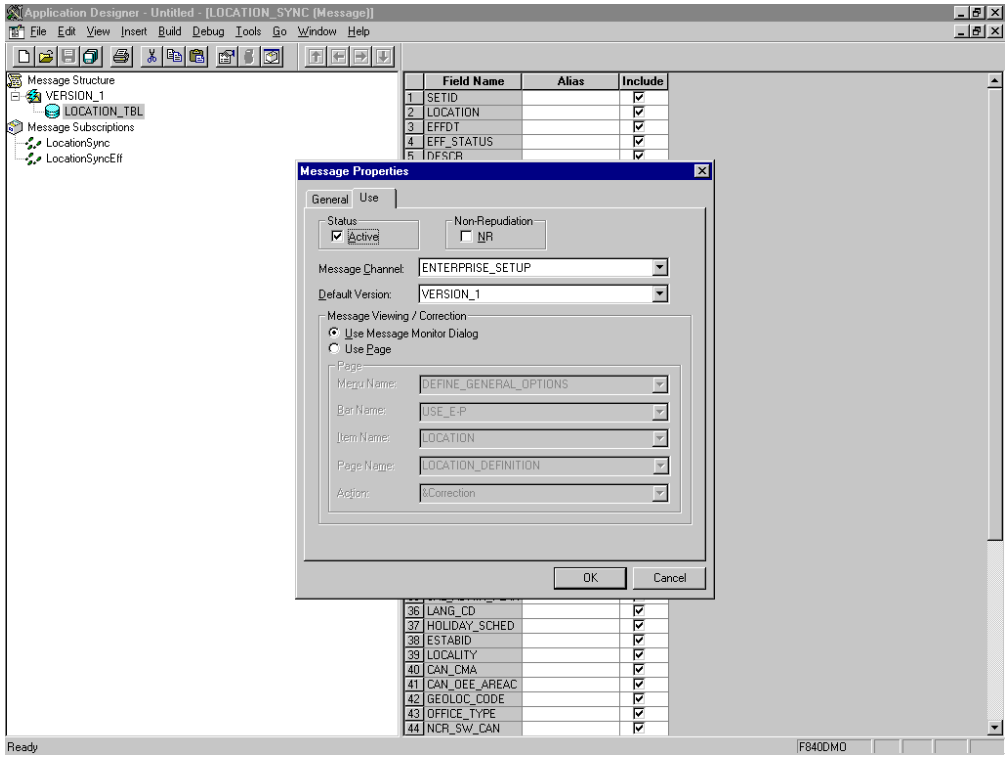
1. Open Application Designer.
2. On the File menu, point to *Open*, click *Message*, and open the *LOCATION_SYNC* message.

The following image shows the fields included in the message: Field Name, Alias, and Include.




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 following image shows the Message Properties dialog box, with the Use tab selected. It includes Non-Repudiation and Status check boxes (selected), Message Channel and Default Version lists, and Use Message Monitor Dialog and Use Page option buttons.




**a.** Select the *Use* tab.

**b.** Ensure the *Active* check box is selected.

The message is routed to the Message Channel, ENTERPRISE_SETUP, and the default message version is VERSION_1 (messages can have multiple versions).

**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

If your Web server is WebLogic, the IntegrationGateway.properties file resides in:

```
c:\bea\wlserver6.1\config\peoplesoft\applications\PSIGW\Web-inf\
```

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 supplied 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.

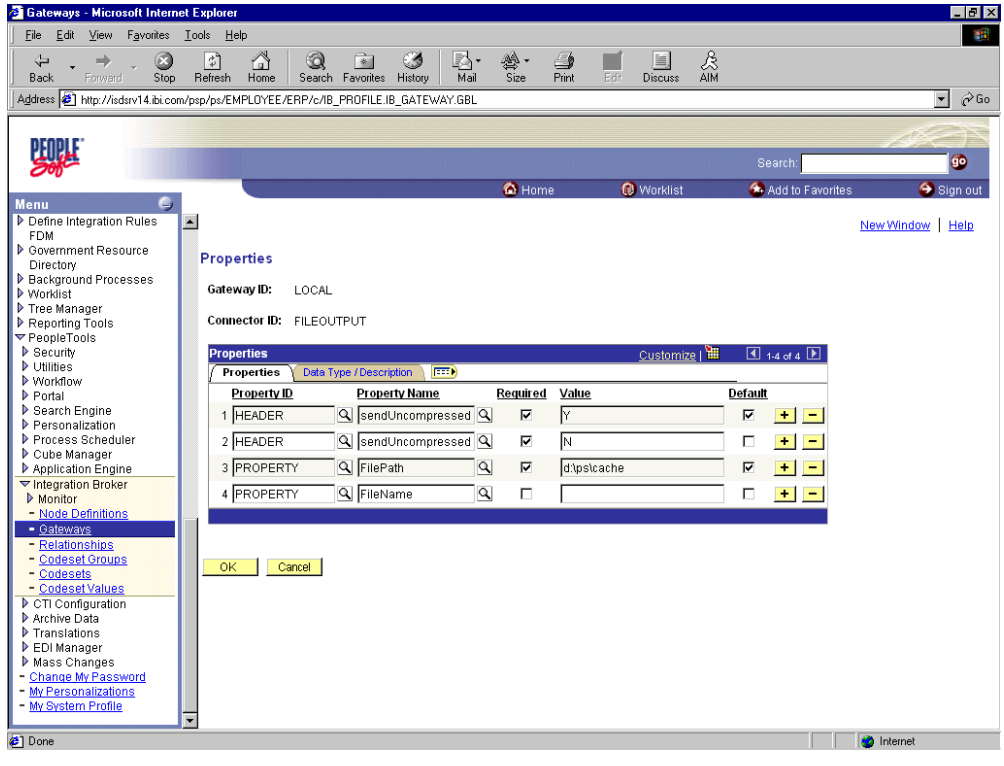
**b.** Click *OK*.

**4.** Scroll down and click *Save*.

You must click Save before continuing.

**5.** Click the *Properties* hyperlink for the FILEOUTPUT Connector ID.

The Properties window for the FILEOUTPUT Connector opens, as shown in the following image.




**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 then, 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 D-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.

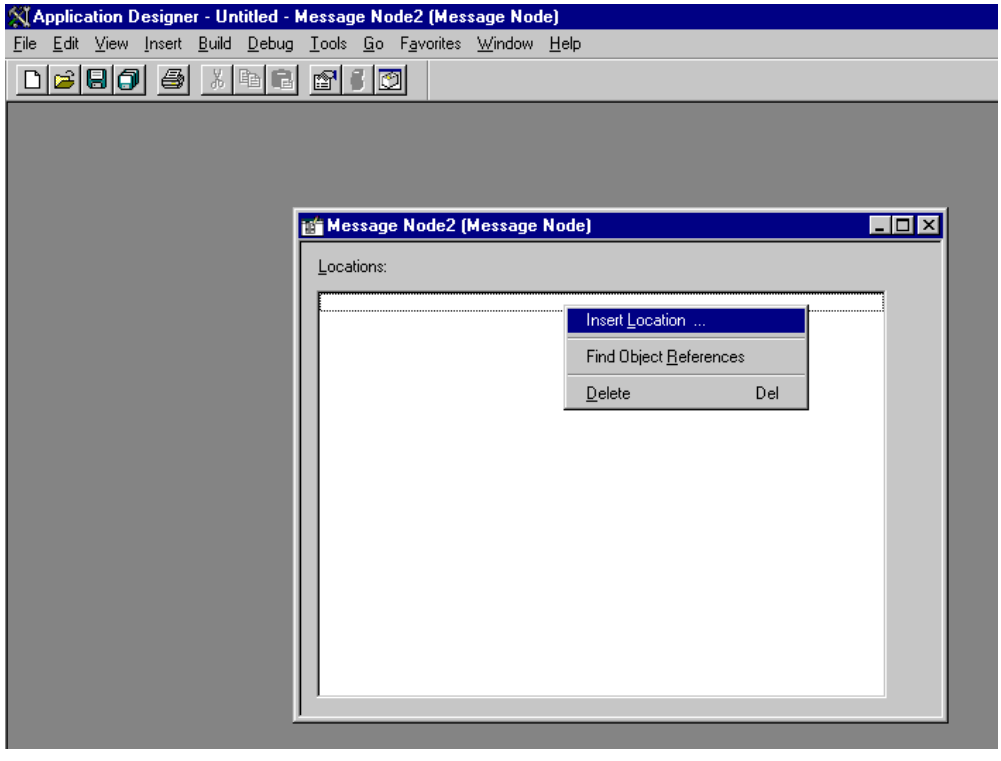
These tasks are described in detail in the following procedures.

## Procedure: How to Create and Configure a New Message Node

To create and configure a new Message Node:

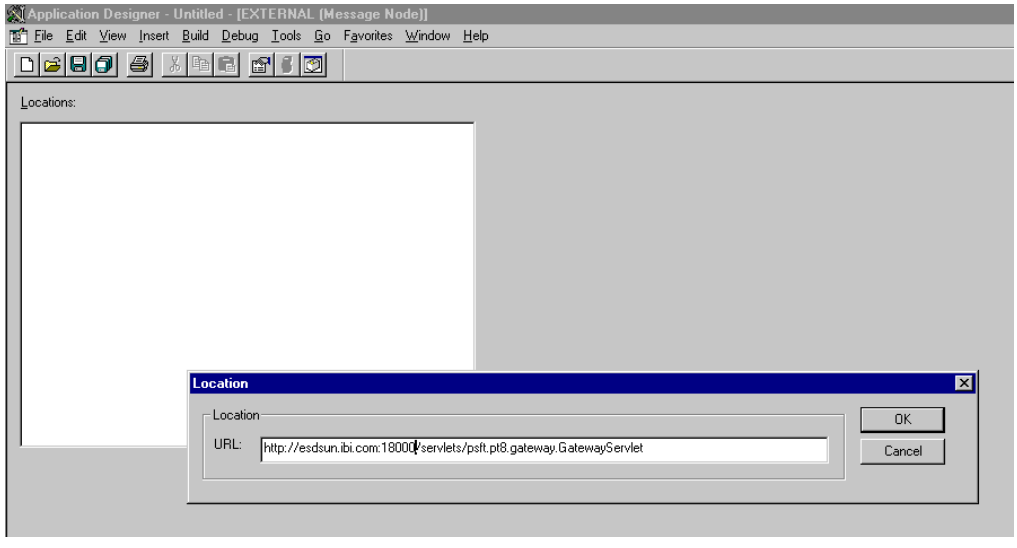
**1.** From the File menu, select *New* and click *Message Node*.

This image shows the Message Node window and contains three options: Insert Location, Find Object References, and Delete.




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

The following image shows the Location URL box. It also contains OK and Cancel buttons.




**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.

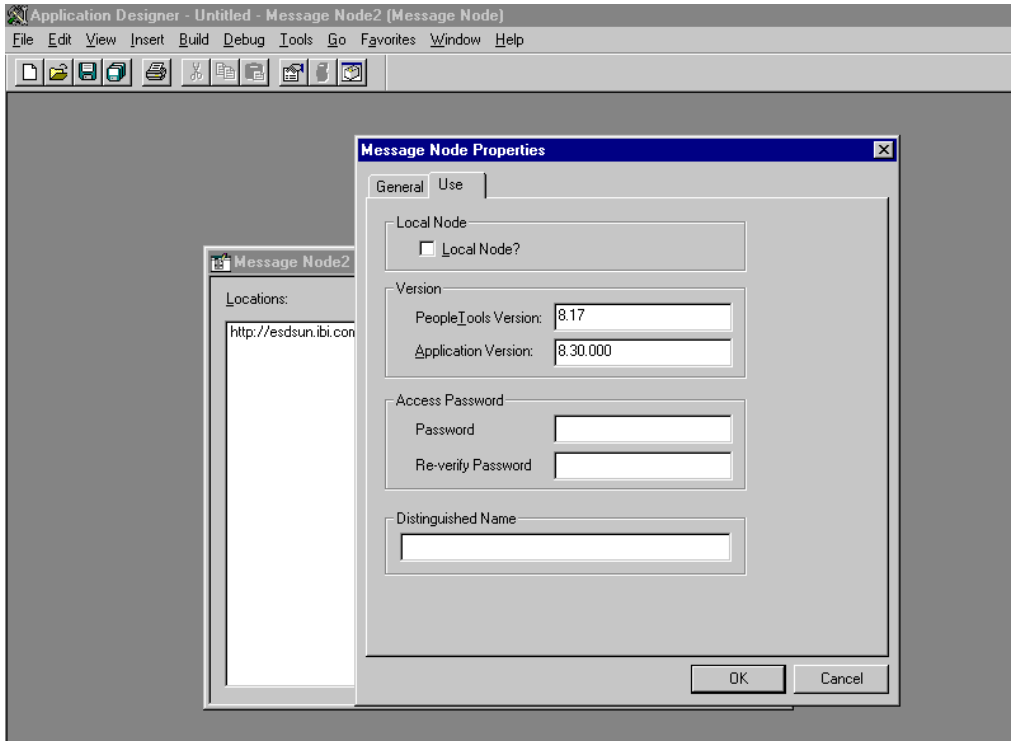
*port*

Is the socket on which the server is listening.

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

**4.** Click *OK*.

The following image shows the Message Node Properties dialog box. The Use tab is selected. It contains a Local Node check box, PeopleTools Version field, Application Version field, Password field, Re-verify Password field, and Distinguished Name field. It also contains OK and Cancel buttons.

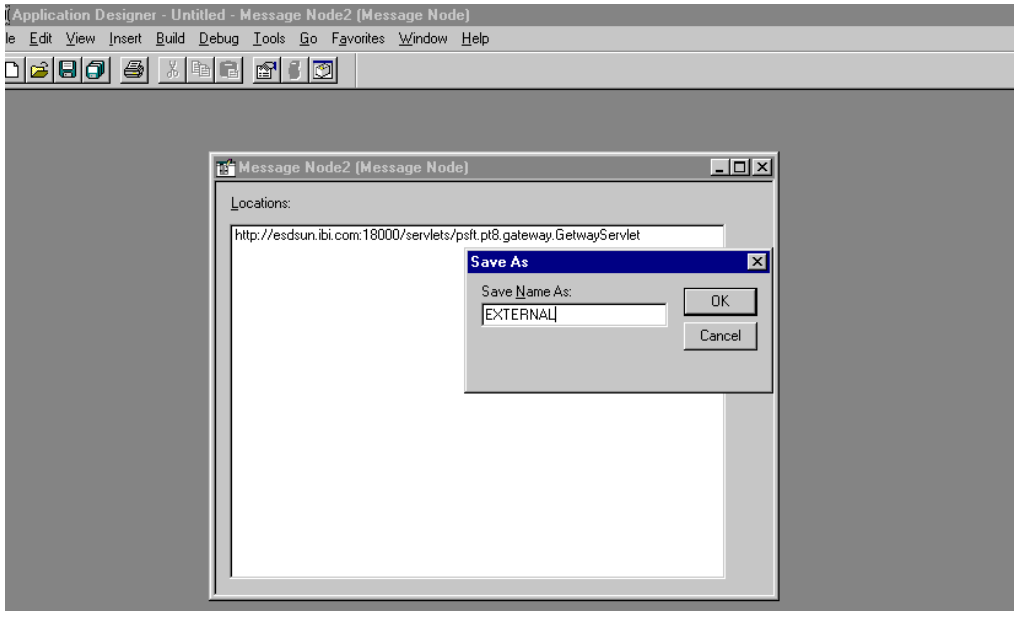



**a.** Select the *Use* tab.

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

5. Click *OK*.

   The following image shows the Save As dialog box. It contains a Save Name As field, and OK and Cancel buttons.




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.

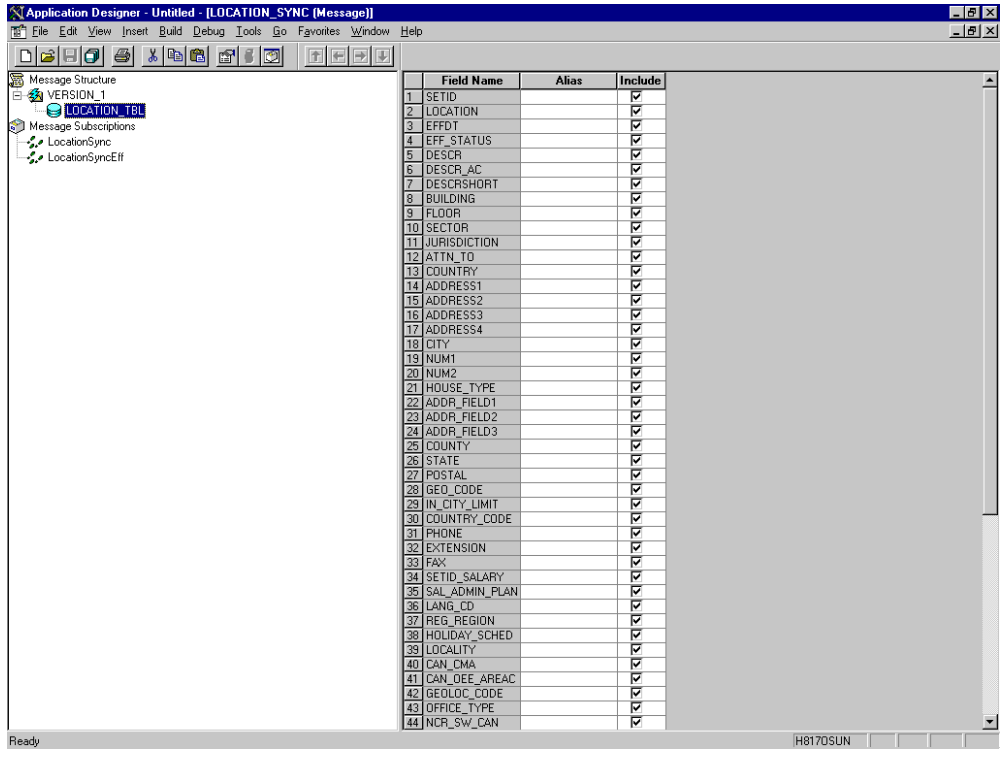
## Procedure: How to Ensure That the Message Is Active and Is Routed Correctly

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* message.To view the fields that are included in the message, highlight *LOCATION_TBL*.

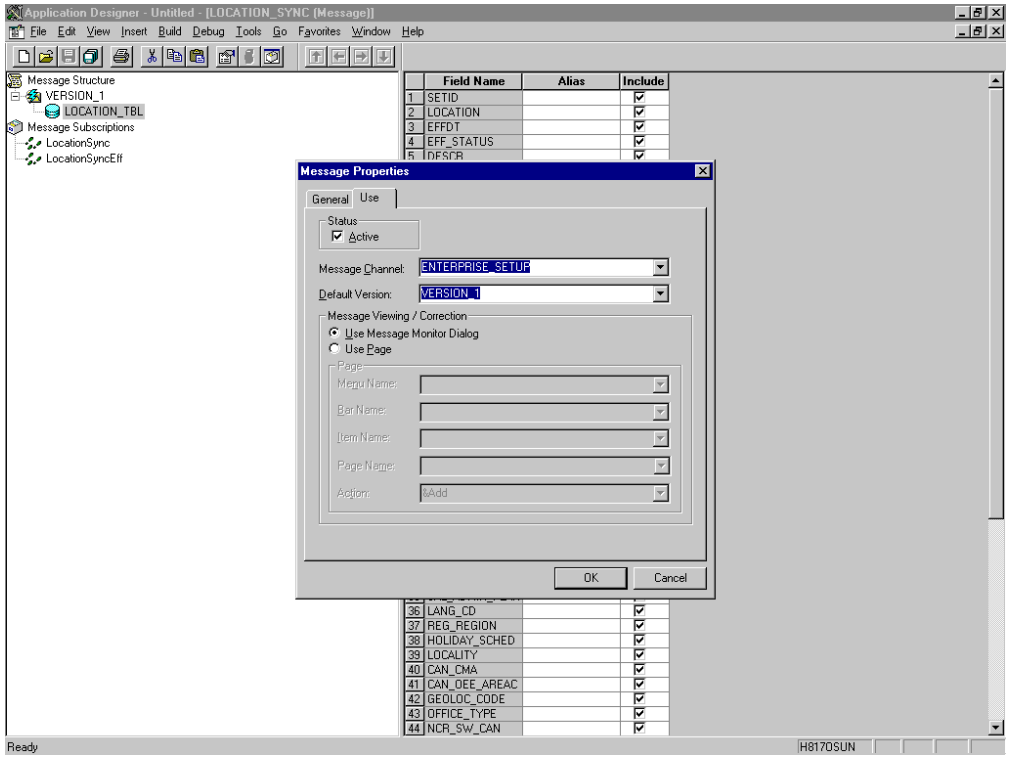
**3.** To view the fields that are included in the message, highlight *LOCATION_TBL*.

The following image shows the LOCATION_TBL field, selected in the left pane. It contains Field Name, Alias, and Include information.



**4.** Right-click *LOCATION_TBL* and select *Properties*.

The following image shows the Message Properties dialog box. It contains a Status check box, Use Message Monitor Dialog and Use Page option buttons, and Message Channel and Default Version lists. It also contains a page section (unavailable).




**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).

**5.** Click *OK*.

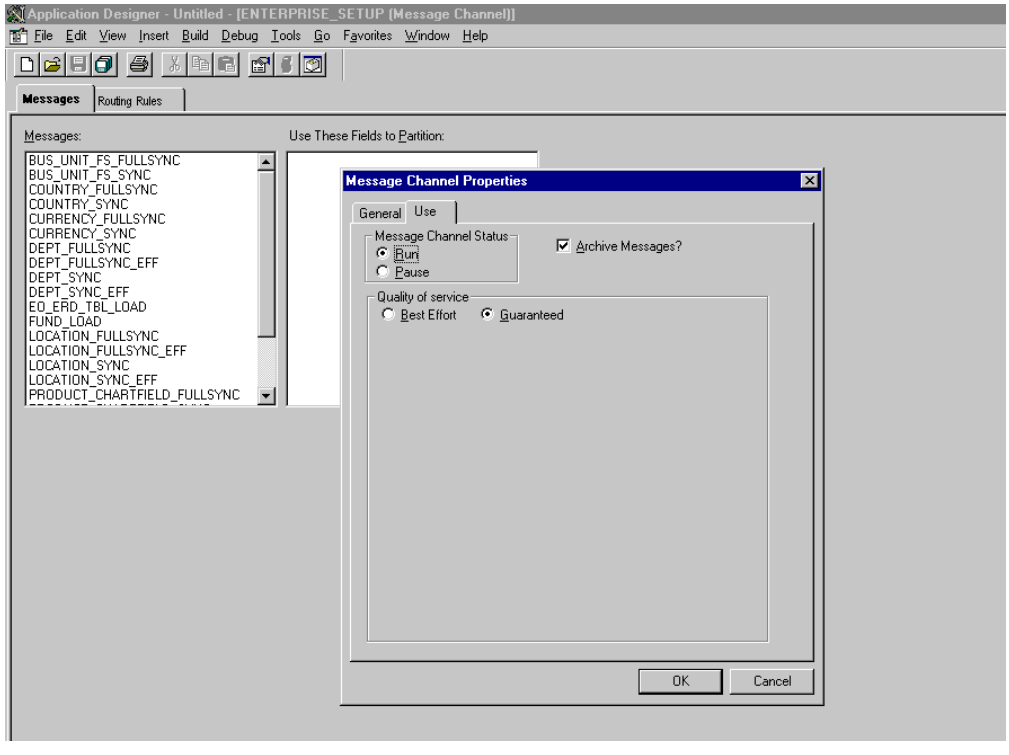
**6.** Save the message.

## 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 following image shows the Message Channel properties dialog box, with the Use tab selected. It contains Message Channel Status and Quality of Service option buttons, and Archive Messages and Unordered check boxes.



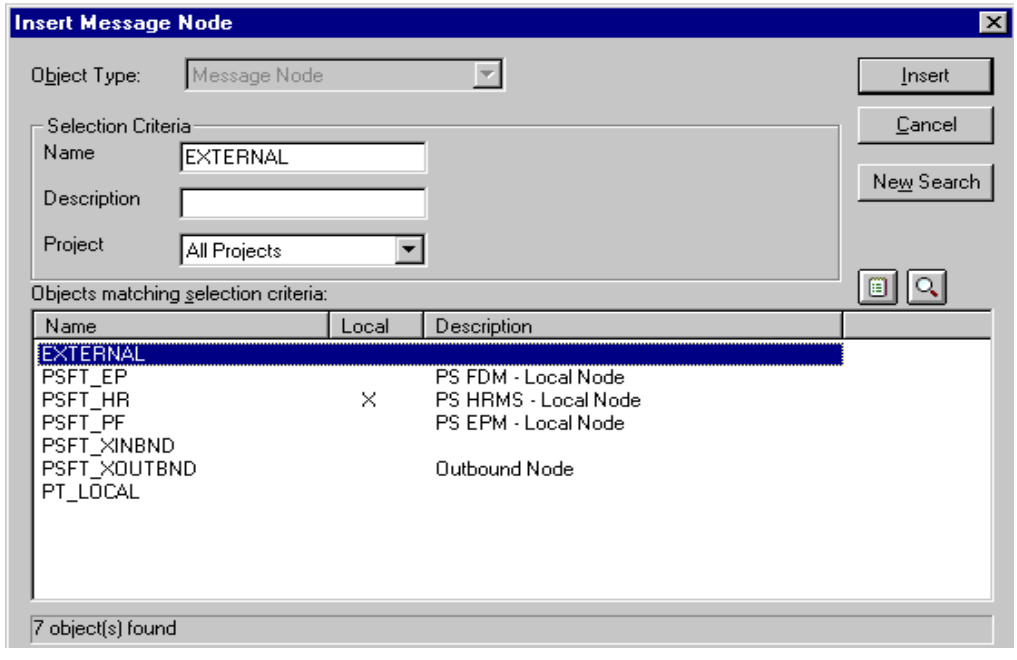

   a. Select the *Use* tab.

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

   c. Click *OK*.

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.




   a. Select the message node, for example, EXTERNAL, that you created in *How to Create and Configure a New Message Node* on page D-9.

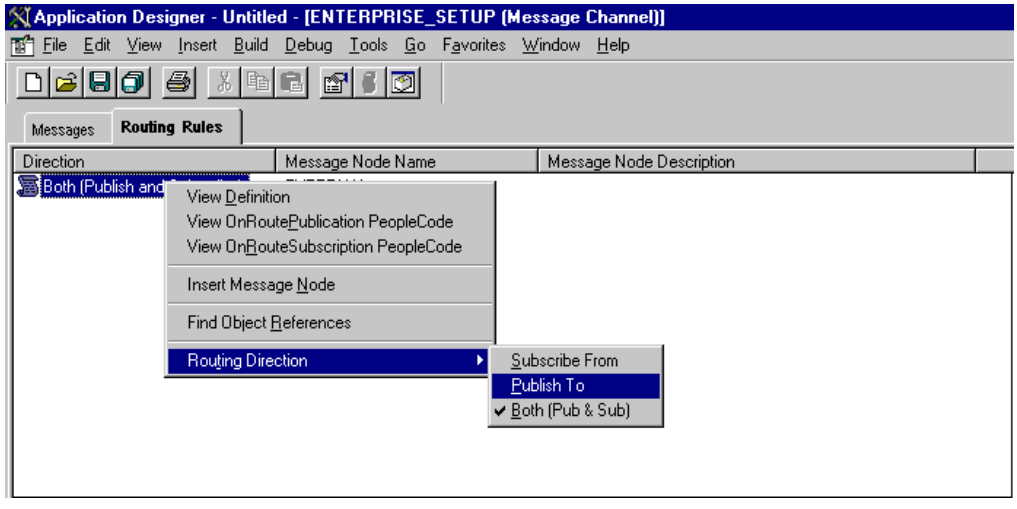
   b. Click *Insert*.

5. Click *Cancel*.

   a. Right-click the message node and point to *Routing Direction*.

   b. From the Routing Direction menu, select *Publish To*.

The following image illustrates the Routing Rules pane. The following options are available: View Definition, View OnRoute Publication PeopleCode, View OnRoute Subscription PeopleCode, Insert Message Node, Find Object References, and Routing Direction.




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:

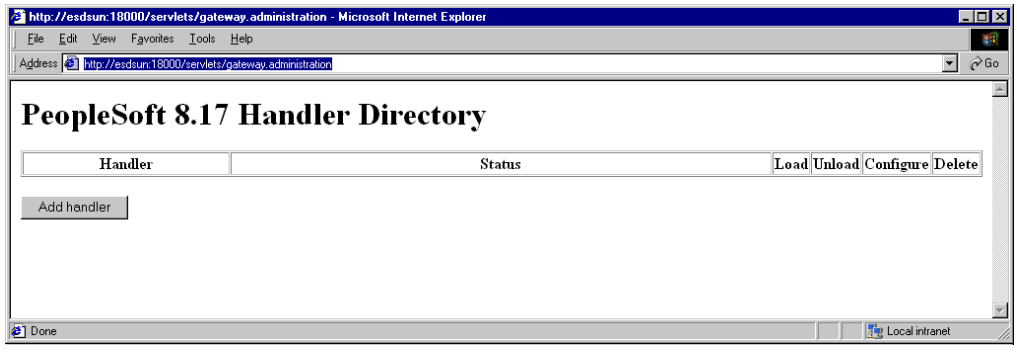
   *machine-name*

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

   *port*

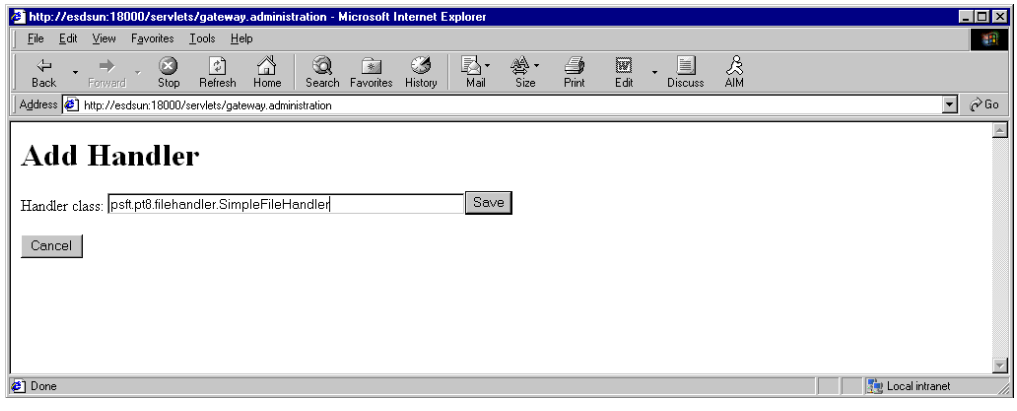
   Is the port number on which the application server is listening.

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.



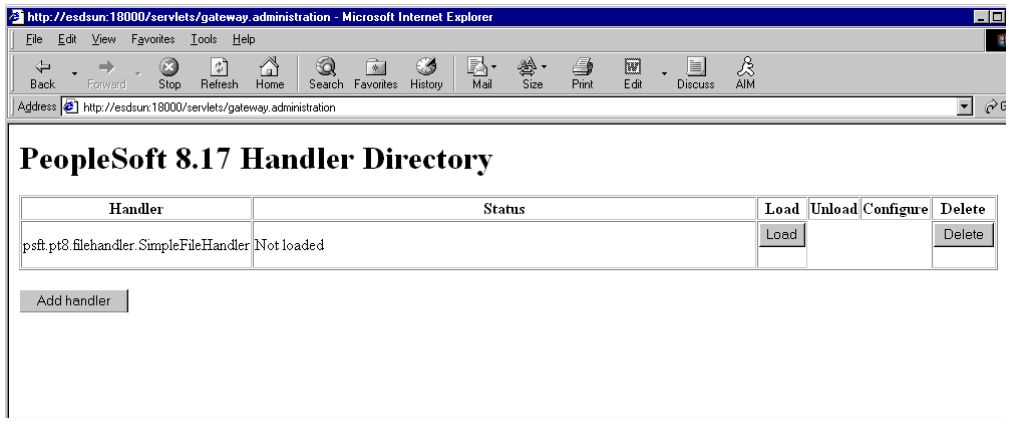

3. Type the following full name of the Simple File Handler class.

   ```
   psft.pt8.filehandler.SimpleFileHandler
   ```

   **Note:** The name is case-sensitive.

4. Click *Save*.

The Handler Directory window reopens, as shown in the following image.

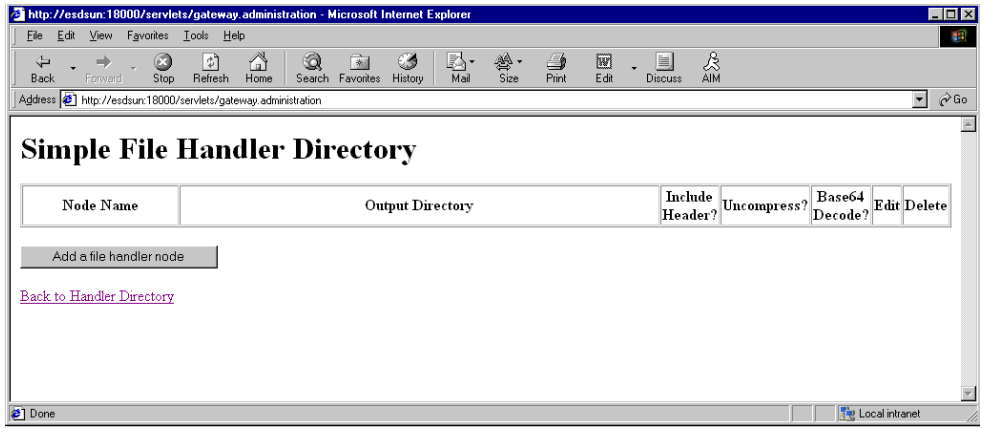



**5.** To load the handler, click *Load*.

After the handler loads, "Loaded successfully" appears in the Status column.

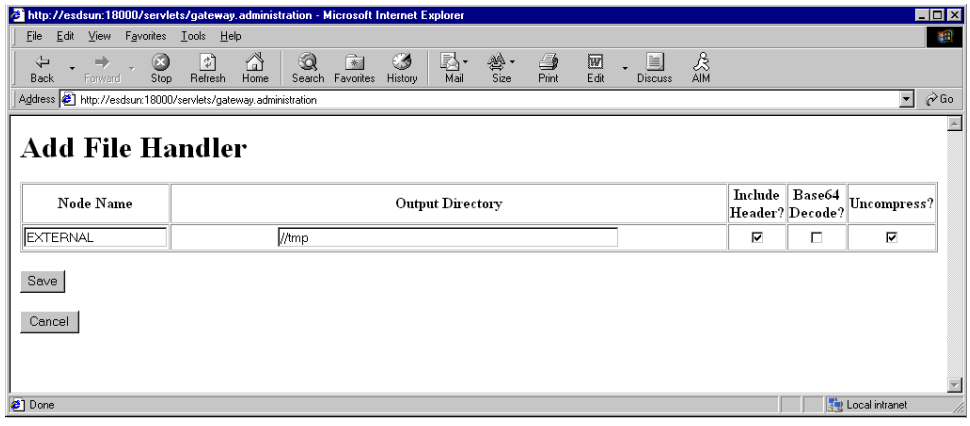
**6.** Click *Configure*.

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 D-9.

**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

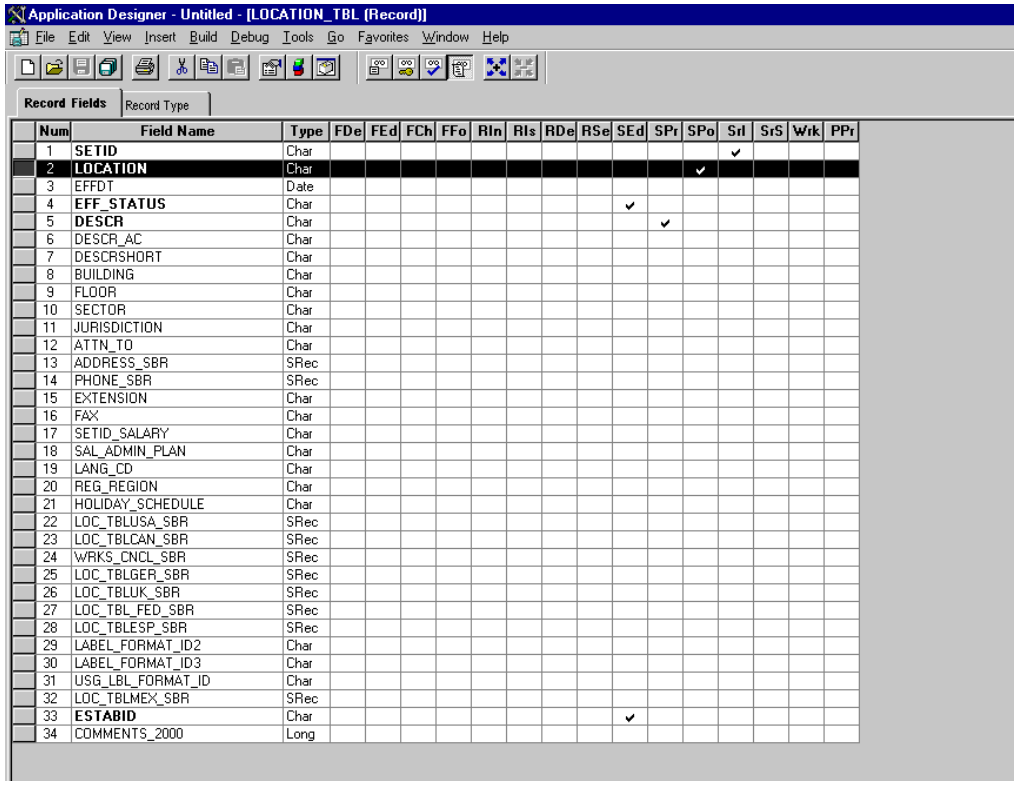
Messages are initiated by the PeopleCode that is attached to a record. Usually, this record is one of the records associated with the message itself.

## 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.




| 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 | | | | | | | | | | | | | | | |

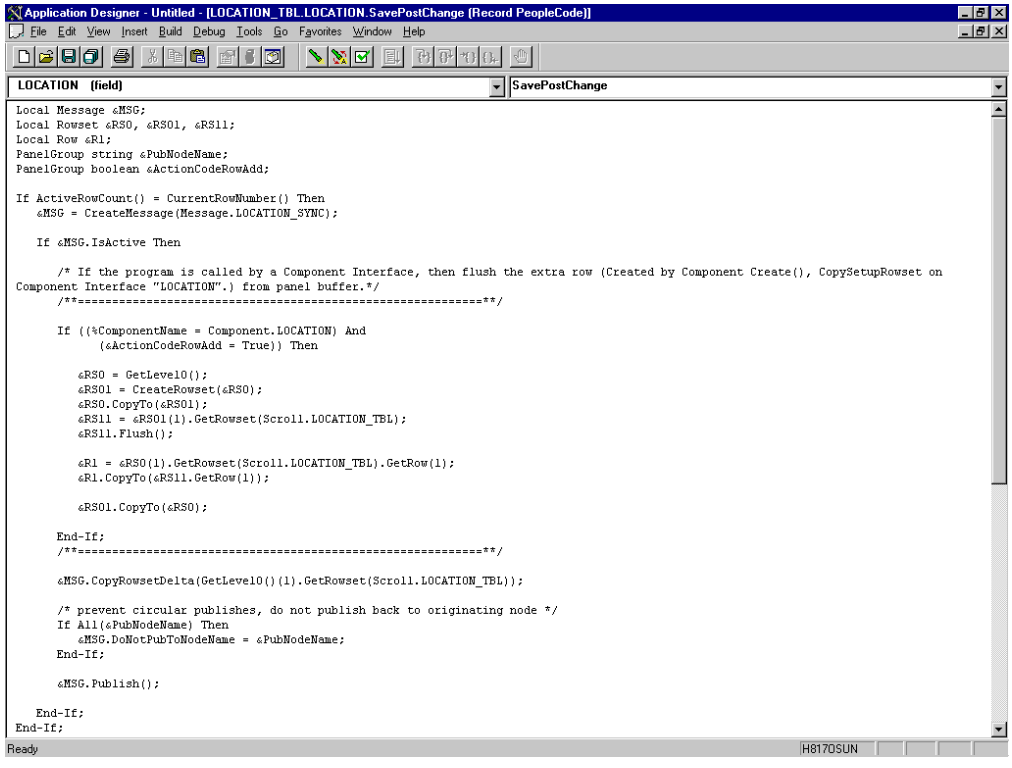
**2.** Select the *Record Fields* tab

**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.



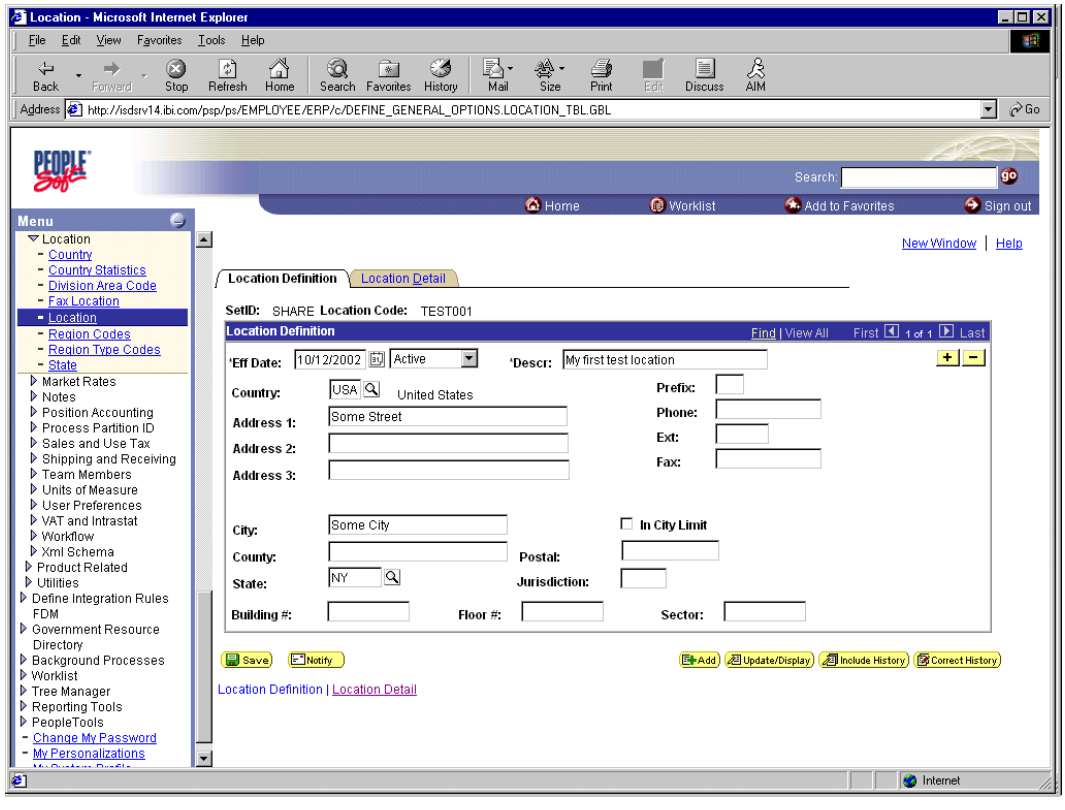

For more information about PeopleCode, consult your PeopleSoft Online Library.

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).

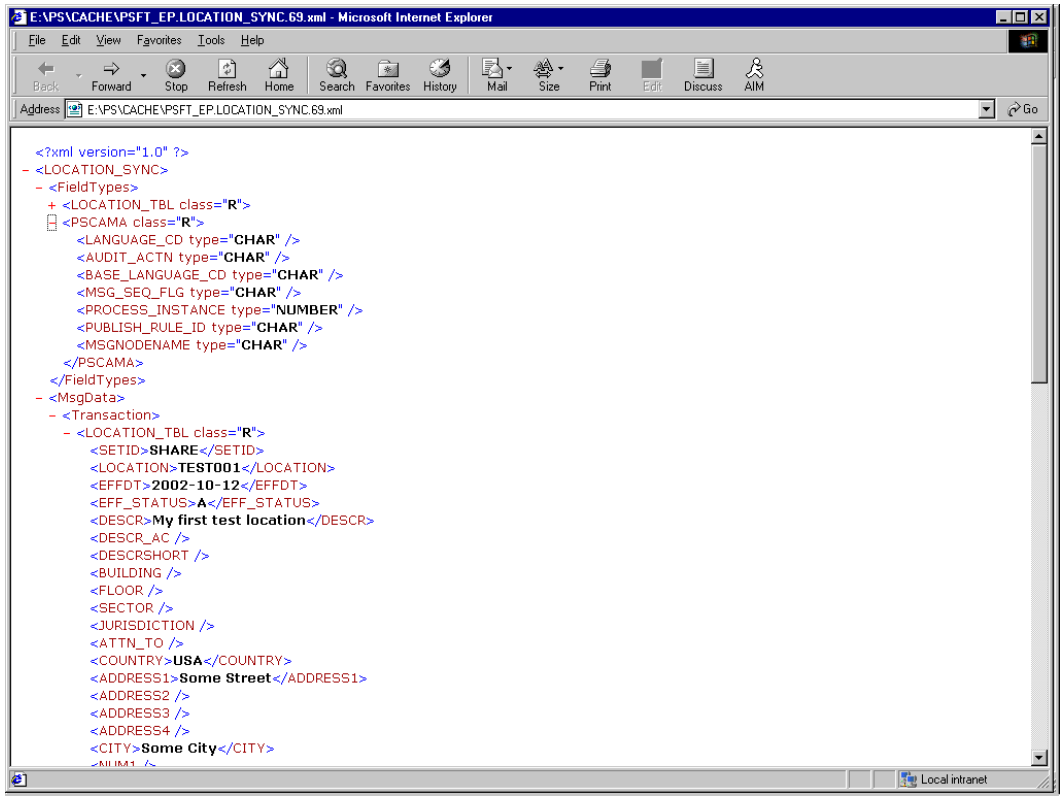
# Testing the Integration Broker

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 displays a Financials 8.4 application where 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.




**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 D-8. 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 BEA WebLogic 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 D-3.

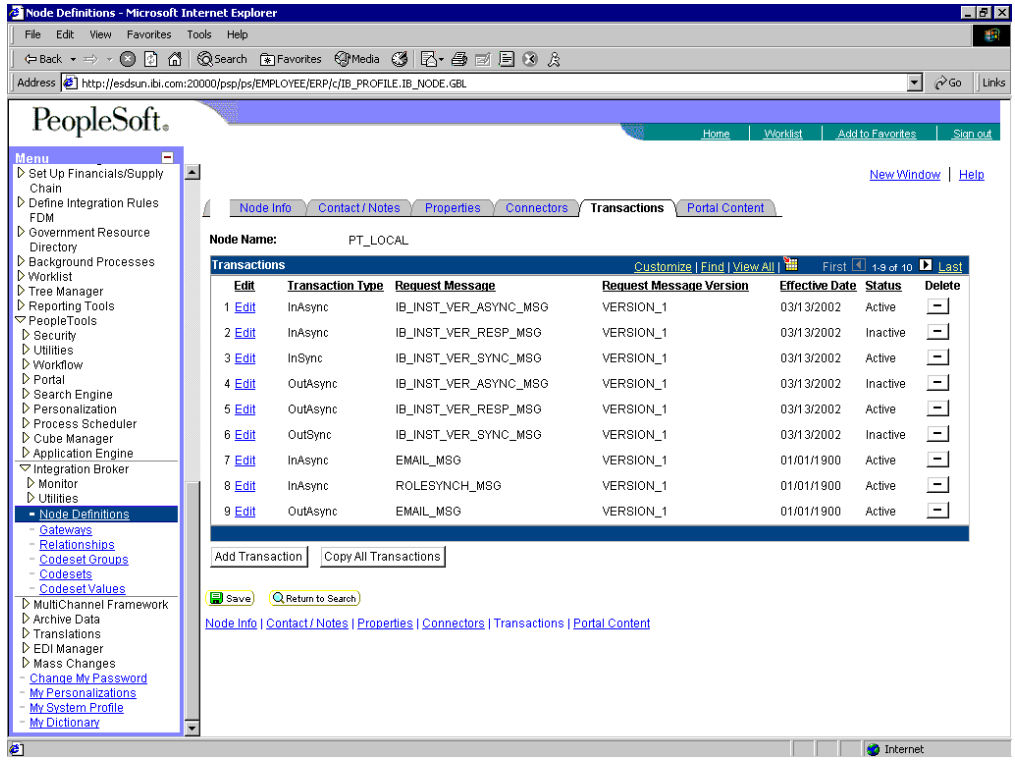
Ensure that both outbound and inbound messages are created and active. PeopleSoft provides template examples called IB_INST_VER_SYNC_MSG and IB_INST_VER_RESP_MSG. For information on examining these messages, see *How to Ensure the Message Is Active and Is Routed Correctly* on page D-3.

## 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.

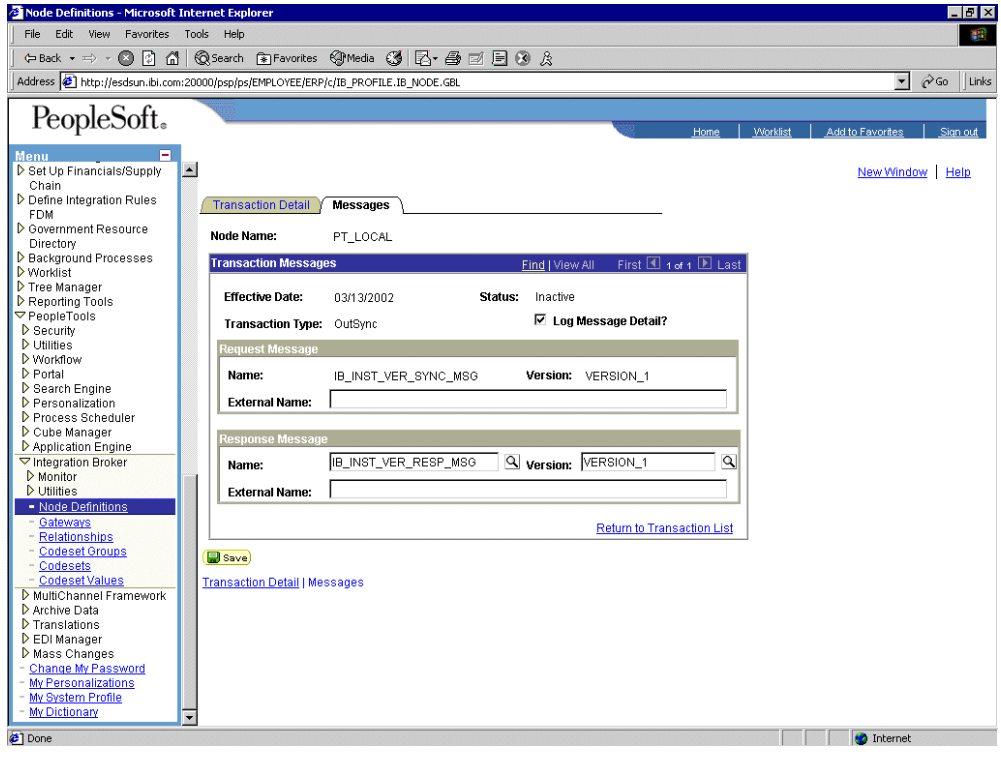
The following image shows the Transactions pane that opens.




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 Messages tab becomes 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 BEA WebLogic Server, you must transform the XML that is sent and returned from your final destination.

**Note:** You must use the PeopleSoft-supplied HTTP target connector when you are working with synchronous outbound messages. You cannot use the 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 image shows the code.