BEA WebLogic Adapter for PeopleSoft® 8

User Guide

Release 7.0
Document Date: October 2002

Copyright

Restricted Rights Legend

Trademarks or Service Marks

BEA WebLogic Adapter for PeopleSoft 8 User Guide

Part Number	Date
N/A	October 2002

Table of Contents

5. Using PeopleSoft 8 Application Messaging

6. Creating Event Schema for Application Messages

About This Document

This document explains how to use the BEA WebLogic Adapter for PeopleSoft 8, which is used to develop client-server interfaces between PeopleSoft 8 and other applications. It describes how to use the BEA WebLogic Adapter for PeopleSoft 8 with WebLogic Integration and the BEA Application Explorer to develop online connections to PeopleSoft 8 applications.

This document is organized as follows:

- Chapter 1, "Introducing the BEA WebLogic Adapter for PeopleSoft 8," provides an overview of the PeopleSoft architecture, component interface, application messaging manager, and describes how to use the BEA Application Explorer with the BEA WebLogic Adapter for PeopleSoft 8.
- Chapter 2, "Using the Component Interface," illustrates how to use the component interface.
- Chapter 3, "Creating BEA Schemas for PeopleSoft Component Interfaces," describes how to create BEA schemas for PeopleSoft Component Interfaces.
- Chapter 4, "Creating and Deploying Application Views," describes how to create application views that provide the business-level interface to the service and events supported by the BEA WebLogic Adapter for PeopleSoft 8.
- Chapter 5, "Using PeopleSoft 8 Application Messaging," discusses how to use and create PeopleSoft message channels that enable you to pass PeopleSoft XML to the WebLogic environment.
- Chapter 6, "Creating Event Schema for Application Messages," illustrates how to create BEA schemas for the PeopleSoft 8 event adapter.

What You Need to Know

This document is written for system integrators who develop client interfaces between PeopleSoft 8 and other applications. It describes how to use the BEA WebLogic Adapter for PeopleSoft 8 and how to develop application environments with specific focus on message integration. It is assumed that readers have the following skill set:

- PeopleSoft 8 general product knowledge, including Application Designer and PeopleTools.
- PeopleSoft Application Messaging.
- PeopleSoft Component Interface.
- Business application knowledge in a specific application area.
- Knowledge of the PeopleSoft 8 processes and data model in the required application area.
- General knowledge of client-server concepts.

Related Information

The following documents provide additional information for the associated software components:

- *BEA WebLogic Adapter for PeopleSoft 8 Installation and Configuration Guide*
- *BEA WebLogic Adapter for PeopleSoft 8 Release Notes*
- *BEA Application Explorer Installation and Configuration Guide*
- BEA WebLogic Server installation and user documentation, which is available at the following URL:

  `http://edocs.bea.com/more_wls.htm`

- BEA WebLogic Integration installation and user documentation, which is available at the following URL:

  `http://edocs.bea.com/more_wli.htm`

- PeopleSoft 8 documentation online or on CD-ROM. For example, you can obtain information on the following topics:
  - PeopleSoft Component Interface
  - PeopleSoft Internet Architecture Administration
  - PeopleSoft Application Messaging

Contact Us!

Your feedback on the BEA WebLogic Adapter for PeopleSoft 8 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 8 documentation.

In your e-mail message, please indicate that you are using the documentation for the BEA WebLogic Adapter for PeopleSoft 8 release.

If you have any questions about this version of the BEA WebLogic Adapter for PeopleSoft 8 User Guide, or if you have problems using BEA WebLogic Adapter for PeopleSoft 8, 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

Documentation Conventions

The following documentation conventions are used throughout this document.

Convention	Item
boldface text	Indicates terms defined in the glossary.
Ctrl+Tab	Indicates that you must press two or more keys simultaneously.
italics	Indicates emphasis or book titles.

`monospace text`	Indicates code samples, commands and their options, data structures and their members, data types, directories, and file names and their extensions. Monospace text also indicates text that you must enter from the keyboard.
	Examples:
	`#include <iostream.h> void main ( ) the pointer psz`
	`chmod u+w *`
	`\tux\data\ap`
	`.doc`
	`tux.doc`
	`BITMAP`
	`float`

`monospace boldface text`	Identifies significant words in code. *Example*: `void` **`commit`** `( )`
`monospace italic text`	Identifies variables in code. *Example*: `String` *`expr`*
UPPERCASE TEXT	Indicates device names, environment variables, and logical operators. *Examples*: LPT1 SIGNON OR

Convention	Item
{ }	Indicates a set of choices in a syntax line. The braces themselves should never be typed.
[ ]	Indicates optional items in a syntax line. The brackets themselves should never be typed. *Example*: `buildobjclient [-v] [-o name ] [-f file-list]... [-l file-list]...`
|	Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.
. . .	Indicates one of the following in a command line: ■ That an argument can be repeated several times in a command line ■ That the statement omits additional optional arguments ■ That you can enter additional parameters, values, or other information The ellipsis itself should never be typed. *Example*: `buildobjclient [-v] [-o name ] [-f file-list]... [-l file-list]...`
. . .	Indicates the omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.

CHAPTER

1 Introducing the BEA WebLogic Adapter for PeopleSoft 8

This section provides an overview of how the BEA WebLogic Adapter for PeopleSoft 8 can be used to integrate PeopleSoft systems with other applications. It includes the following topics:

- PeopleSoft EAI Architecture
- PeopleSoft Component Interface
- PeopleSoft Application Messaging Manager
- Using the BEA Application Explorer With the BEA WebLogic Adapter for PeopleSoft 8

The BEA WebLogic Adapter for PeopleSoft 8 provides a means to exchange real-time business data between PeopleSoft systems and other applications, databases, or external business partner systems. The adapter allows for inbound and outbound processing with PeopleSoft.

The BEA WebLogic Adapter for PeopleSoft 8 allows non-PeopleSoft applications to communicate and exchange transactions with PeopleSoft by using WebLogic Integration and XML messages. Applications that require access to PeopleSoft data when a PeopleSoft business event occurs use WebLogic Integration application views, events, and business process workflows to receive messages from PeopleSoft through the adapter. Applications that must interact with PeopleSoft to cause a new PeopleSoft business event use WebLogic Integration application views, services, and business process workflows to send request messages to PeopleSoft through the adapter. If an event in PeopleSoft is triggered, then the PeopleSoft XML event document is sent to the adapter to be processed by WebLogic Integration.

The adapter uses WebLogic Integration and XML messages to allow non-PeopleSoft applications to communicate and exchange transactions with :

- PeopleSoft Component Interface facility.
- PeopleSoft Application Messaging Manager facility.

For component interfaces, the adapter connects to the PeopleSoft Application Server by accessing all APIs for the Component Interfaces that correspond to its supported business objects. Every Component Interface contains the business component's data and business logic, thus alleviating the requirement for the adapter to duplicate the processes defined within the business component.

The adapter is bi-directional, meaning it can detect an event by receiving an XML document from PeopleSoft through Application Messaging. It can also execute a Component Interface by passing an XML request document to execute an instance of the PeopleSoft component interface and its method.

PeopleSoft EAI Architecture

PeopleSoft provides for integration with other applications and systems through its Component Interface framework and its Application Manager facility. The BEA WebLogic Adapter for PeopleSoft 8 makes use of 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 8 include:

- PeopleSoft Java API using Component Interface.
- 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 the PeopleSoft system, such as 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 a number of generic Component Interfaces with each of their applications. These are called Enterprise Integration Points (EIPs). Customers can also develop their own custom Component Interfaces, or they can modify EIPs as required.

PeopleSoft Application Messaging Manager

When using PeopleSoft XML to integrate with PeopleSoft, the interface is facilitated by PeopleSoft Application Messaging. The BEA WebLogic Adapter for PeopleSoft 8 uses a delivered handler that must be configured within the PeopleSoft application gateway using TCP/IP transport services.

Using the BEA Application Explorer With the BEA WebLogic Adapter for PeopleSoft 8

The BEA Application Explorer uses an explorer metaphor for browsing the PeopleSoft system for Component Interfaces and Application Messages. The function of the BEA Application Explorer is to create service schemas for the associated component interface and to create event schemas from applicable XML message definitions.

When running a PeopleSoft service using the component interface java API, you create schemas for the service against a particular component interface.

The steps required to create service and event schemas are described in Chapter 3, "Creating BEA Schemas for PeopleSoft Component Interfaces."

CHAPTER

2 Using the Component Interface

This section describes how to create component interfaces for use with the BEA WebLogic Adapter for PeopleSoft 8. It includes the following topics:

Component Interface Creation

Before the BEA WebLogic Adapter for PeopleSoft 8 can use a component interface, you must create the interface within PeopleSoft and then compile it outside of the PeopleSoft environment.

The class files created by this external compilation are made available to the WebLogic environment in the `CLASSPATH` variable in the `startWebLogic.cmd` file.

Component interfaces are created in the PeopleSoft Application Designer. For more information about using the PeopleSoft Application Designer, see your PeopleBooks documentation.

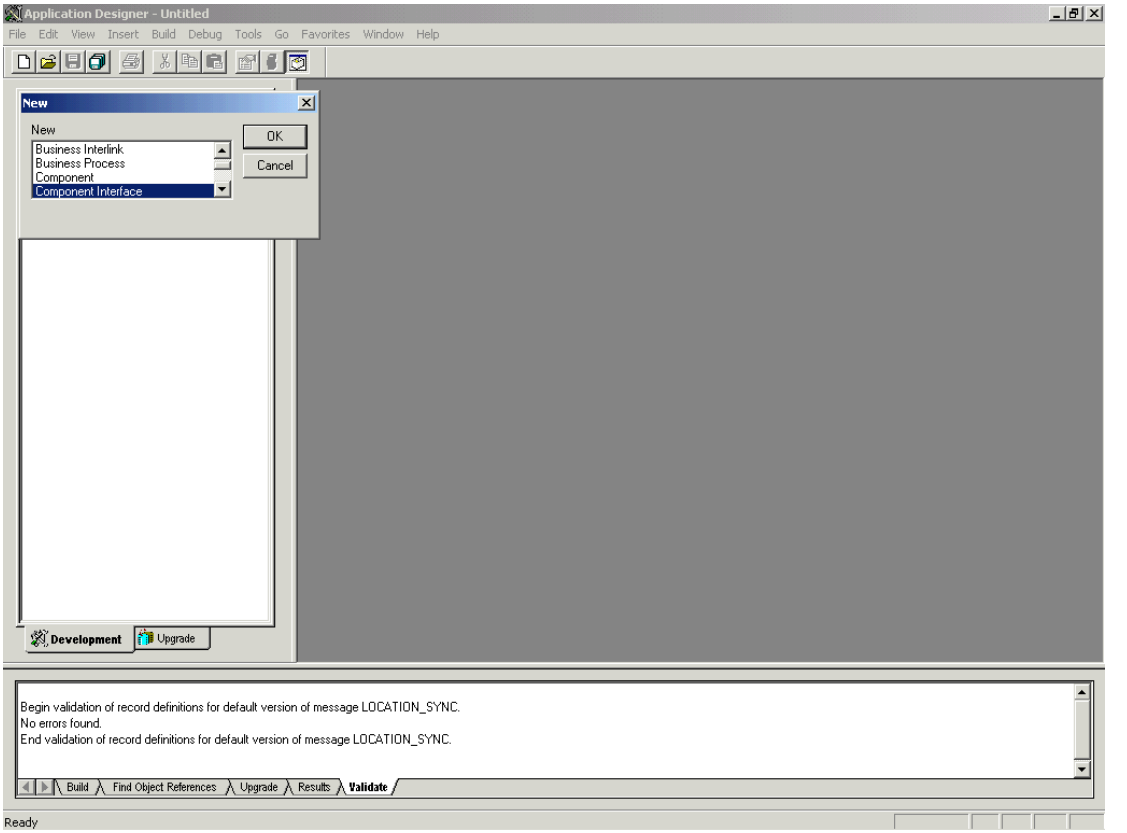
Creating a New Component Interface

To create a component interface:

1. Launch the PeopleSoft Application Designer.
2. Choose File→New.

The New dialog box opens.

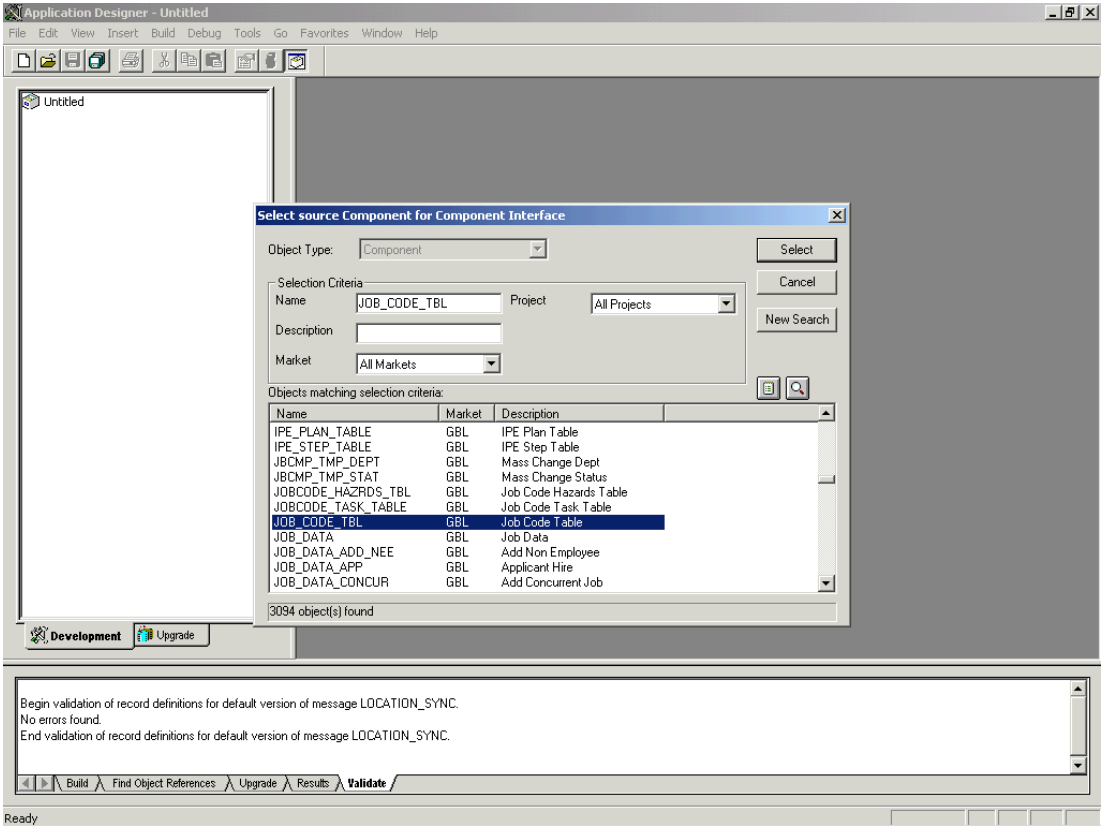
Figure 2-1 New Dialog Box



3. Select Component Interface and click OK.

The Select Source Component for Component Interface window opens.

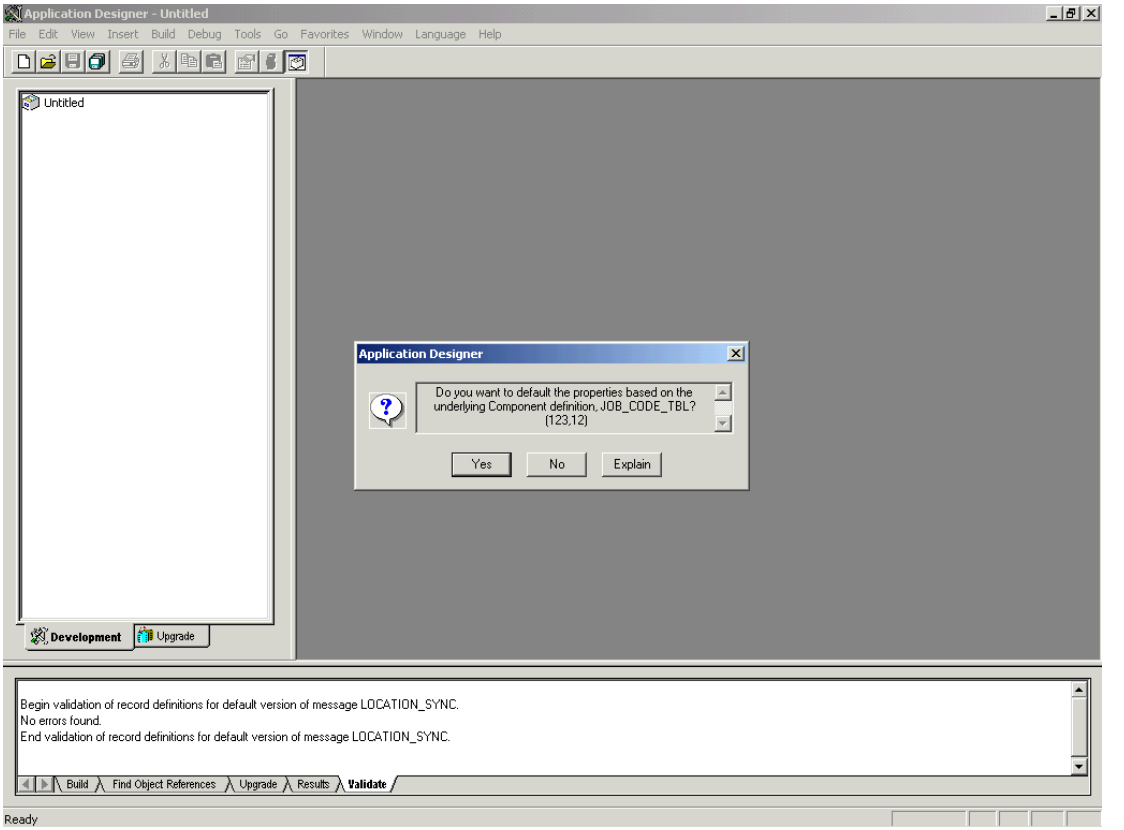
Figure 2-2 Select Source Component Dialog Box



4. Select the component to use as a basis for the component interface.
5. Click Select.

The Application Designer dialog box opens.

Figure 2-3 Application Designer Dialog Box



6. Choose one of the following options:

   - Click No to create the component interface without displaying properties.

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

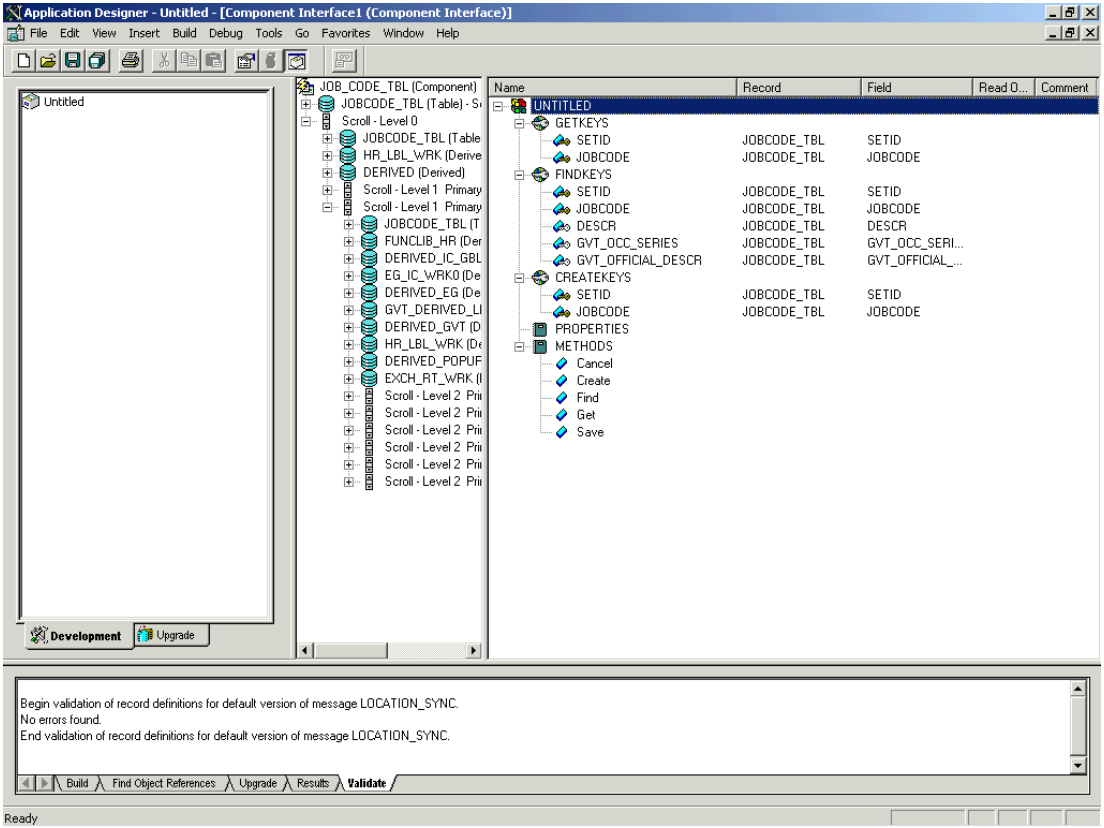
     To expose component properties manually, click No and drag the relevant fields from the left pane to the right pane.

     You can right-click both panes to select various functions to perform depending on which pane is active.

For a complete list of functions, see the PeopleBooks documentation.

- Click Yes to start creating the component interface with the properties of the underlying component interface displayed.

Figure 2-4 Application Designer - Component Interface



Methods

The standard methods created for the component interface are:

- Create
- Find
- Get
- Save

Only those methods that are in the underlying component are available. For example, if the underlying component does not contain Add capabilities, the Create method is not available. You can add methods after the component interface has been saved. These are called user-defined methods.

How to Create User-Defined Methods

1. Right-click anywhere in the component interface view and select View.
2. Select View Peoplecode from the shortcut menu

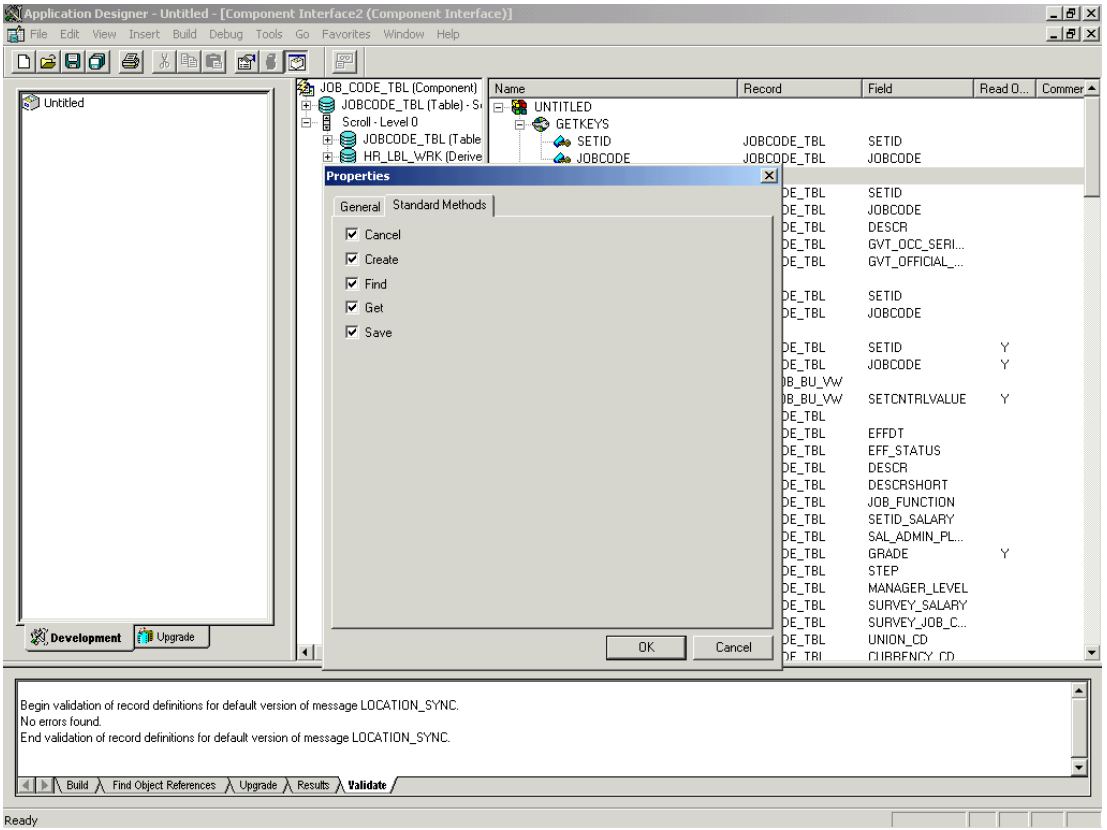
   or

   Select an object in the component interface. Then select View, Application Designer, and View Peoplecode.

How to View or Change Available Methods

1. Open the Component Interface Properties dialog box.

Figure 2-5 Properties Dialog Box - Standard Methods Tab



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

Properties

You may add properties from the records in the component view. You can delete any 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.

You can right-click both panes to select various functions to perform depending on which pane is active. For a complete list of functions, see the PeopleBooks documentation.

Component Interface Security

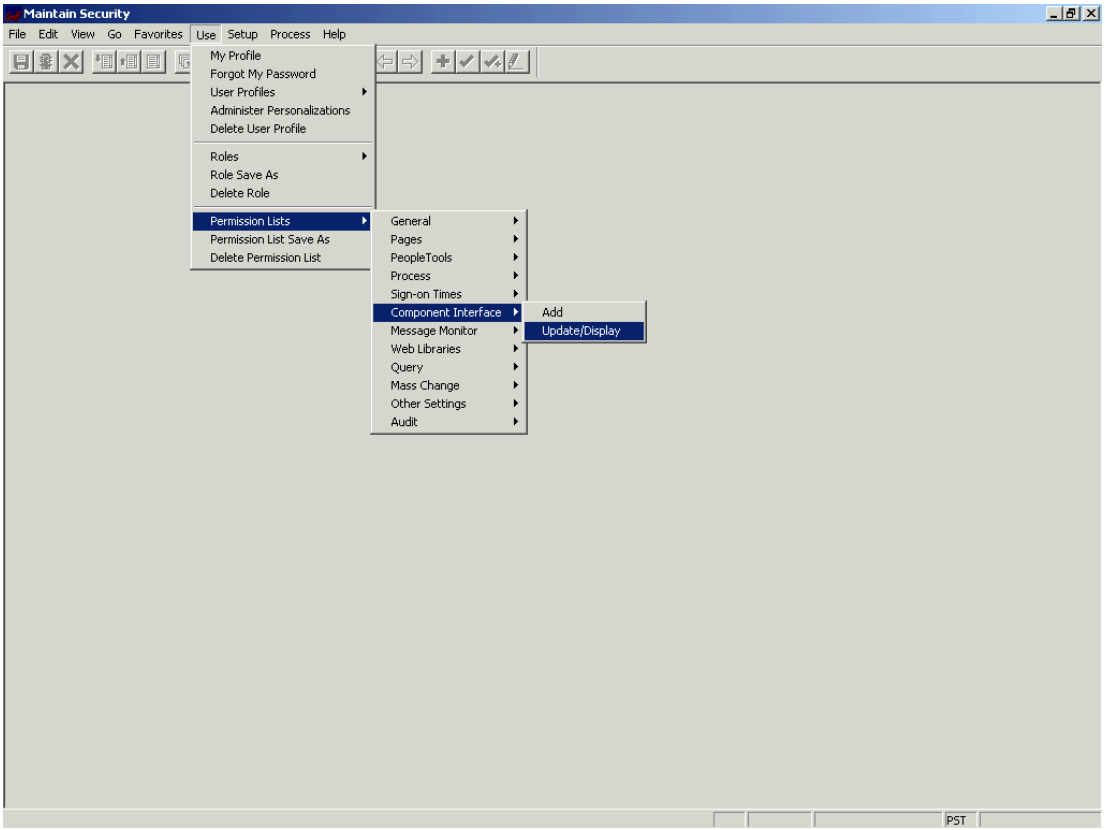
You must set up security for the component interface before you can begin testing.

Configuring Component Interface Security for PeopleSoft Version 8.1x

The following procedure describes how to configure component interface security for PeopleSoft Version 8.1x.

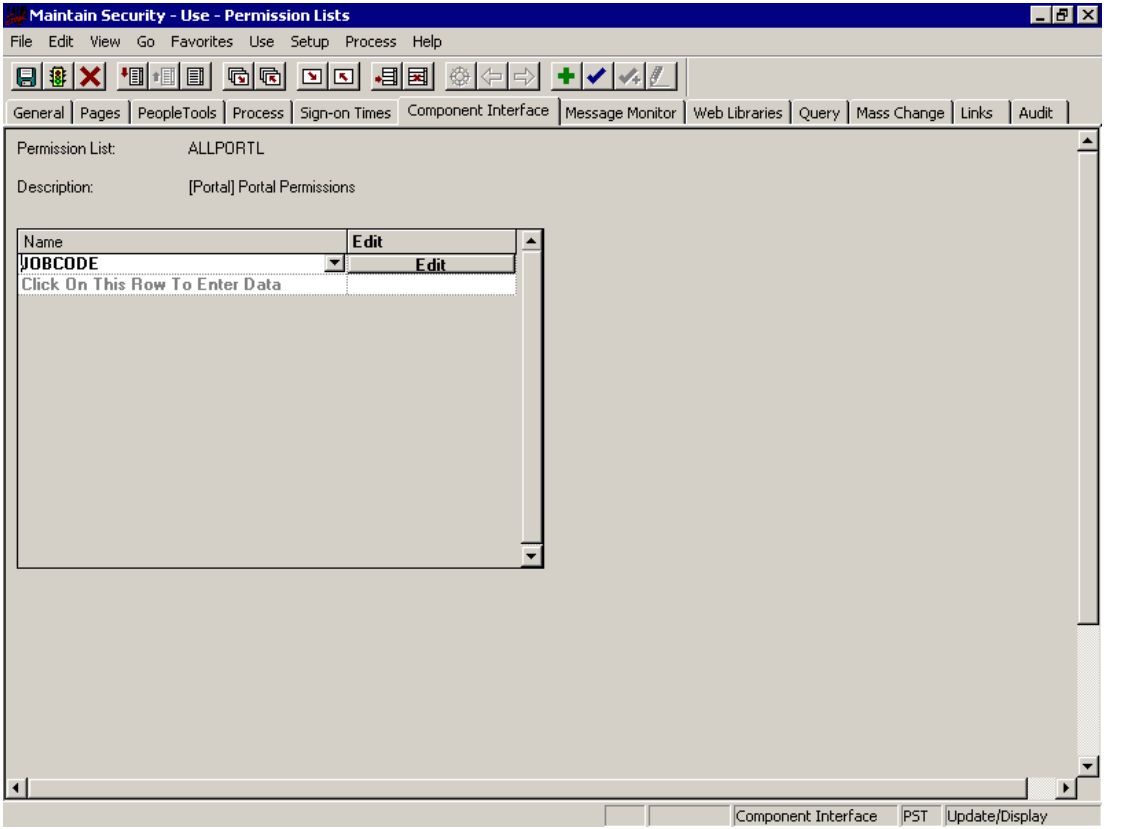
1. Choose Setup→Permission Lists→Component Interface→Update/Display.
2. Choose the relevant Permission list.

Figure 2-6 Component Interface Security



Before Security can be set, the Permission List(s) to be used must already be identified. For more information on Permission Lists, see the PeopleBooks documentation.

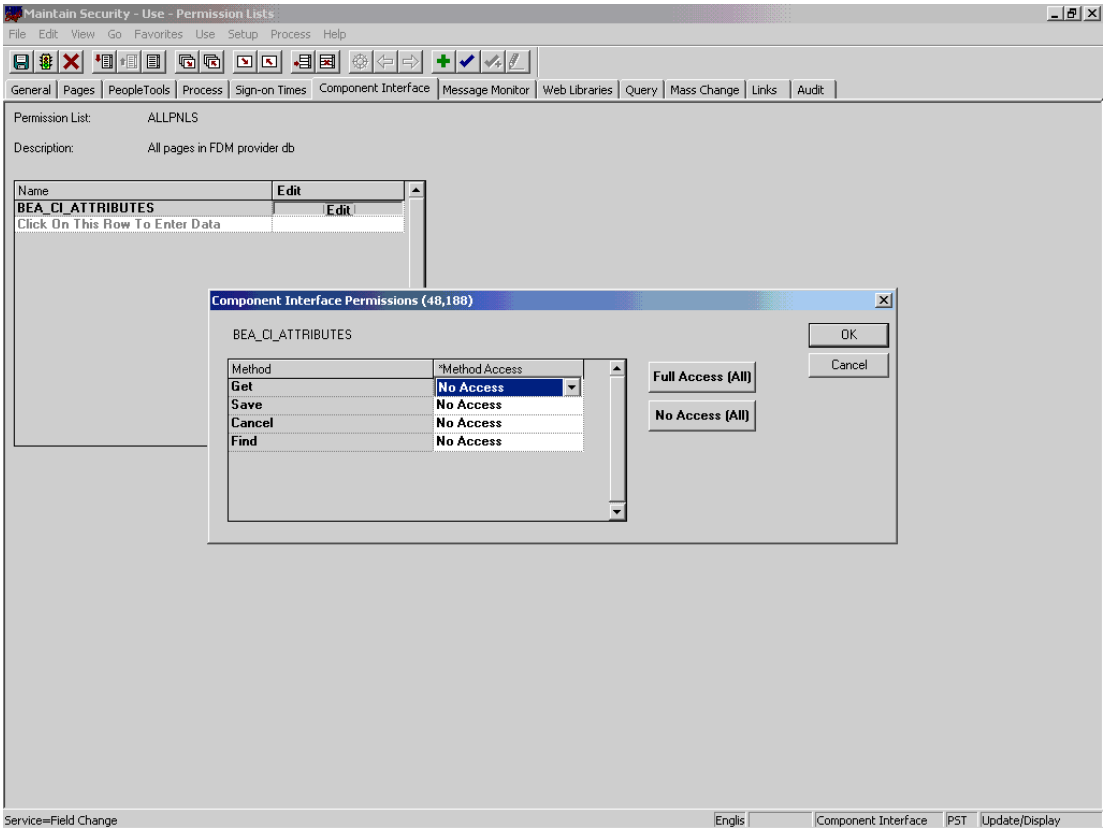
Figure 2-7 Maintain Security - User - Permissions Lists Window



1. Insert the new Component Interface that you created.
2. Click Edit.
3. Select the desired access for this Permission List.
4. Click OK.

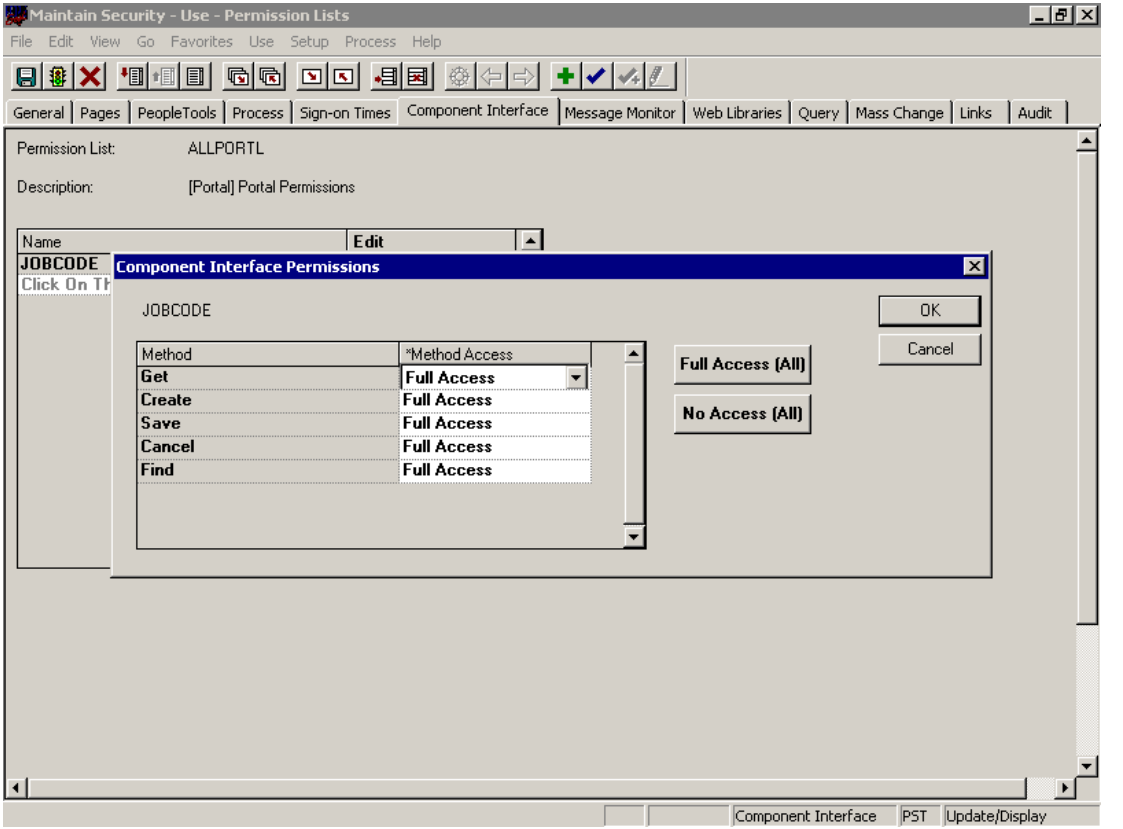
When the relevant Permission List is chosen, the Component Interface displays all available methods, including user-defined methods. This enables you to specify whether this particular Permission List should have Full or Partial Access.

Figure 2-8 Component Interface Permissions



In the following example, the ALLPORTL Permission List is given Full Access to all methods.

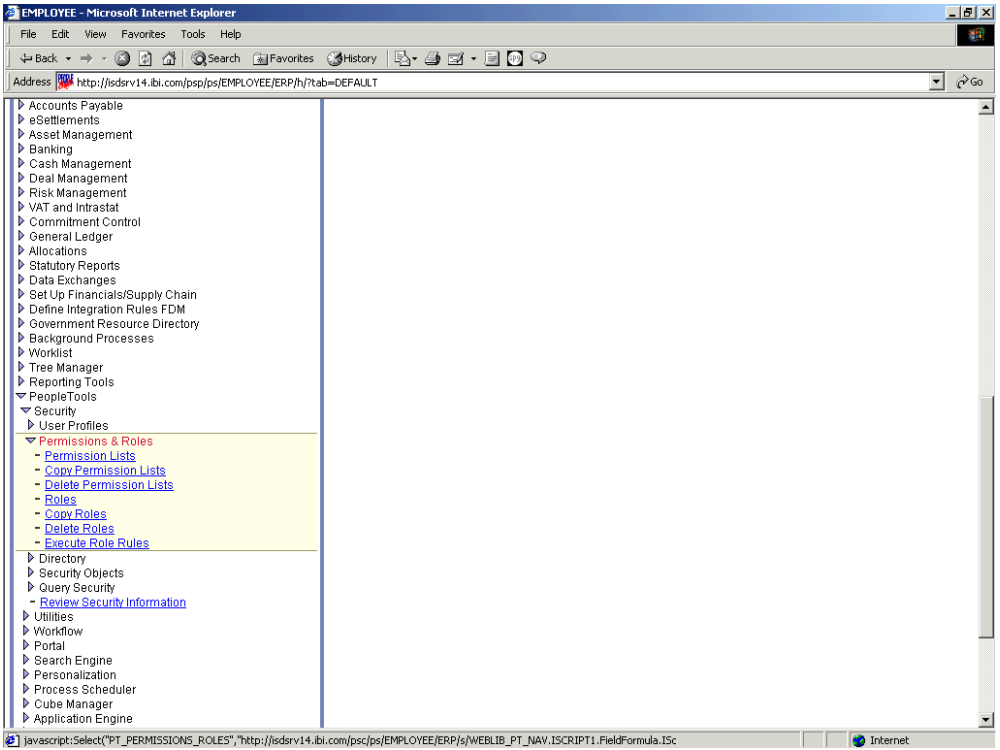
Figure 2-9 Permission Assignment Box



Configuring Component Interface Security for PeopleSoft Version 8.4 or Higher

The following sample procedure describes how to configure component interface security for PeopleSoft Version 8.4 or higher.

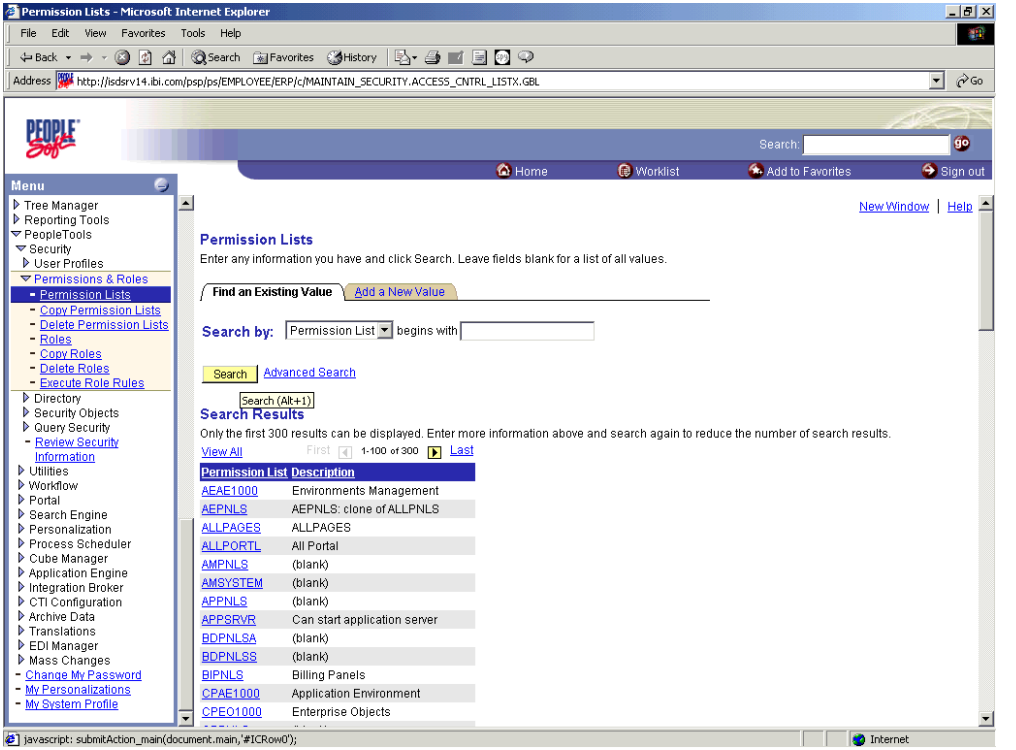
Figure 2-10 Permission Lists Window



1. Click PeopleTools→Security→User Profiles→Permissions & Roles→ Permission Lists.
2. Click the Search button.

The Permissions List search window opens.

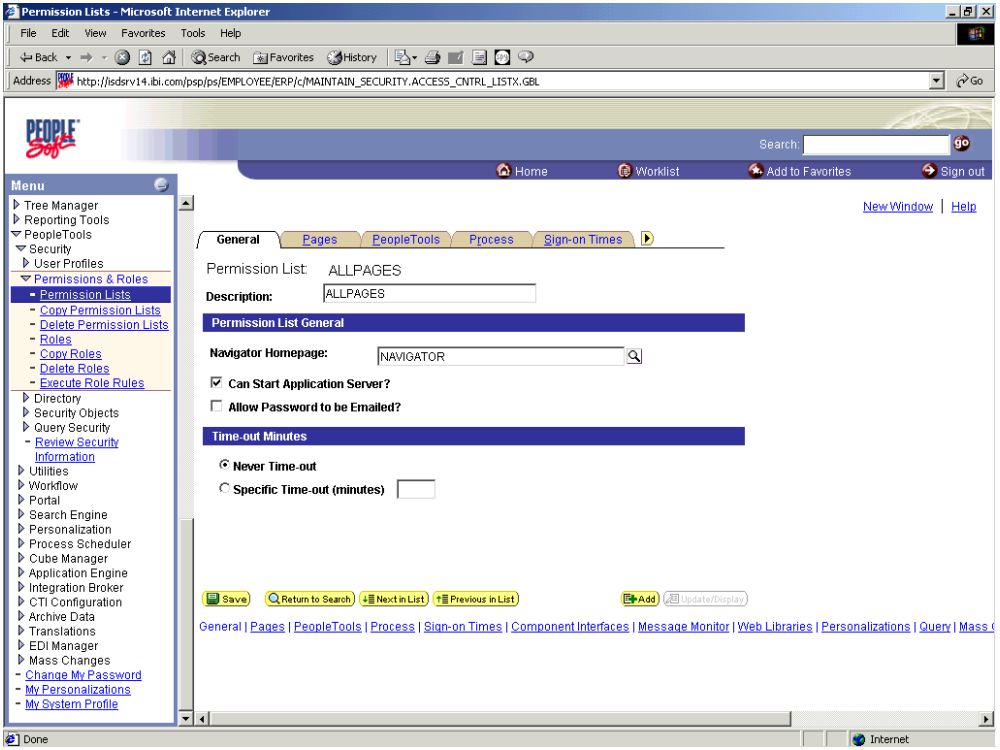
Figure 2-11 Permission Lists Search Window



3. Select the relevant Permission List.

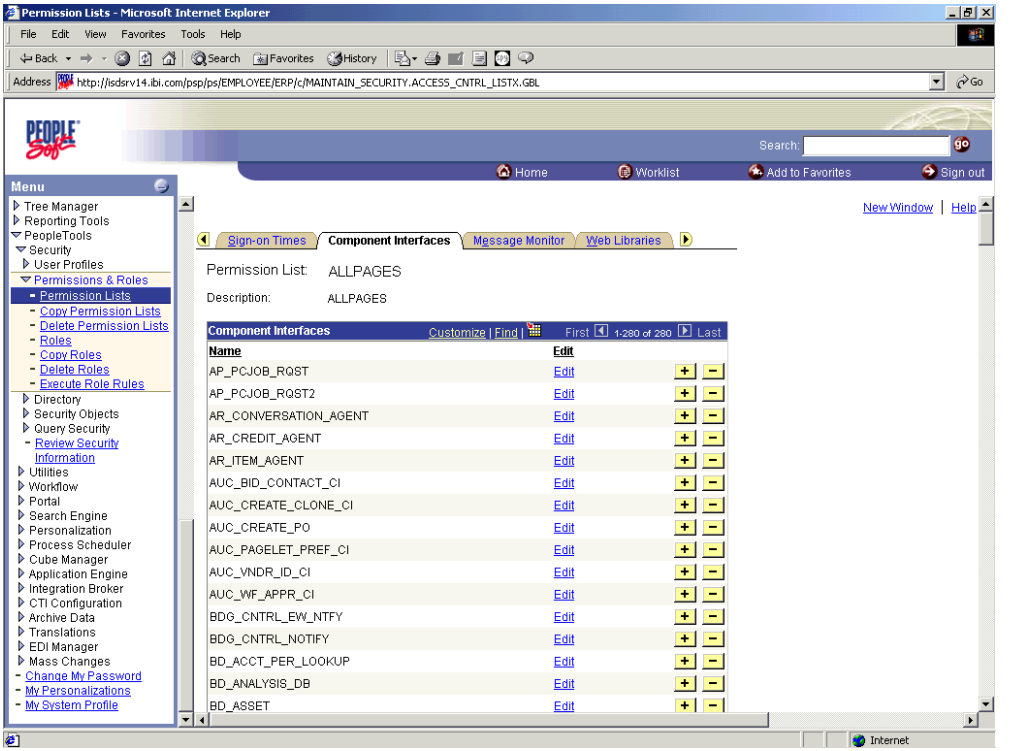
The Permissions List window with several tabs opens.

Figure 2-12 Finding the Component Interfaces tab



4. Click the right arrow next to the Sign-on Times tab to display the Component Interfaces tab.

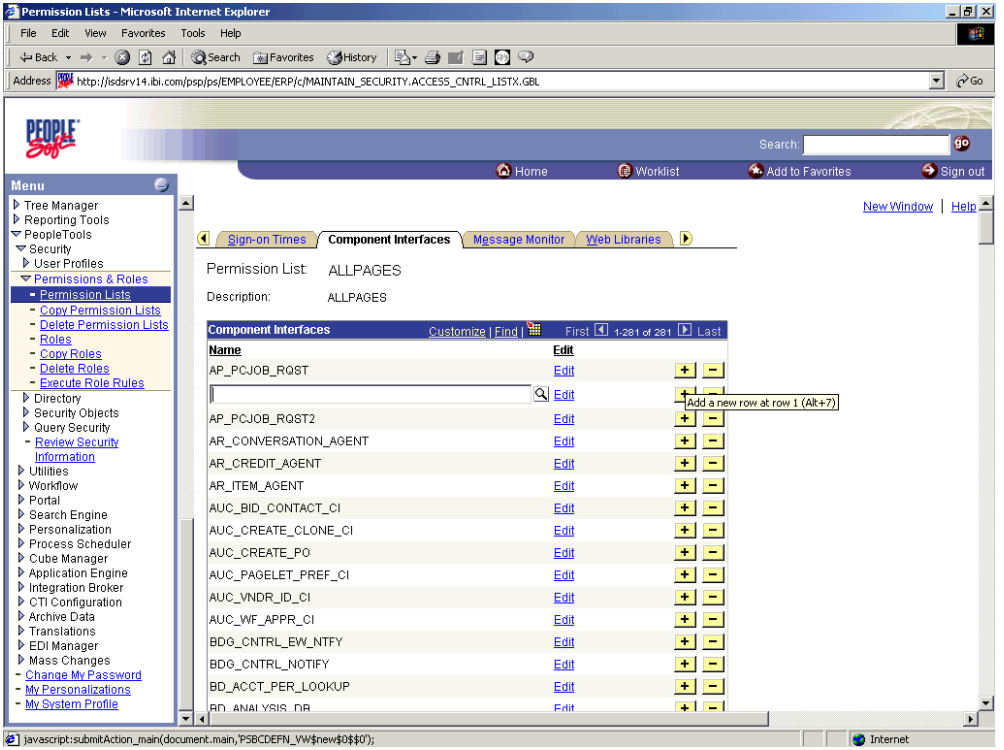
Figure 2-13 Permissions List Window - Component Interfaces Tab



5. Click the Component Interfaces tab.

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

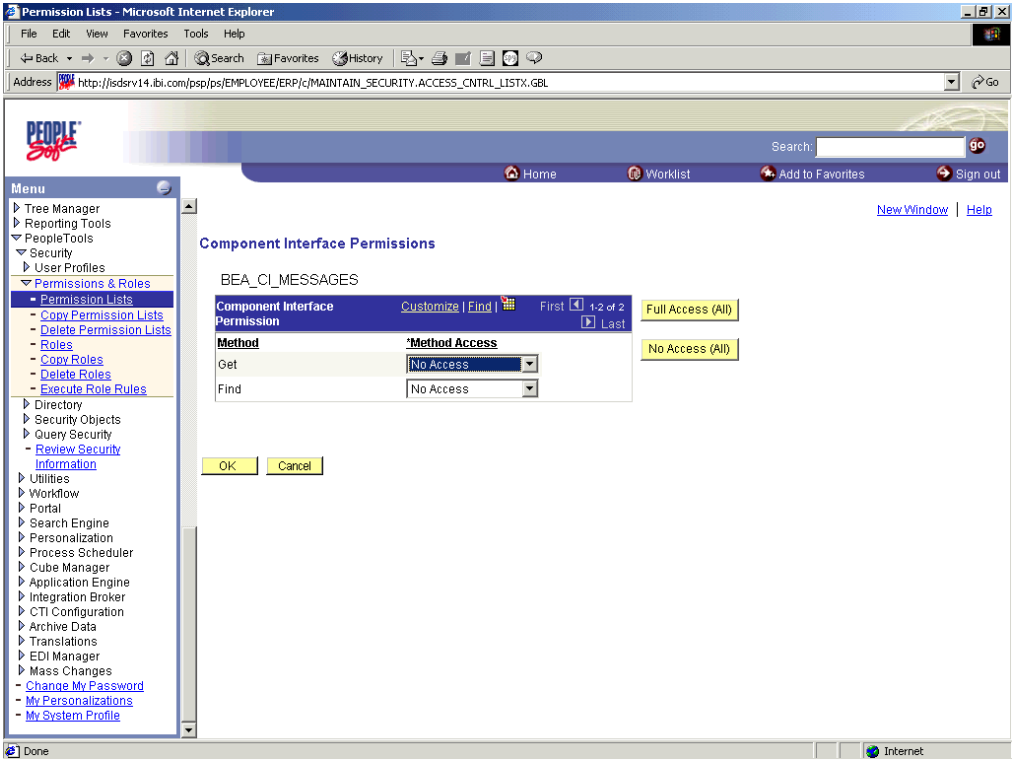
Figure 2-14 Component Interfaces List



7. Enter the component interface name and click Edit.

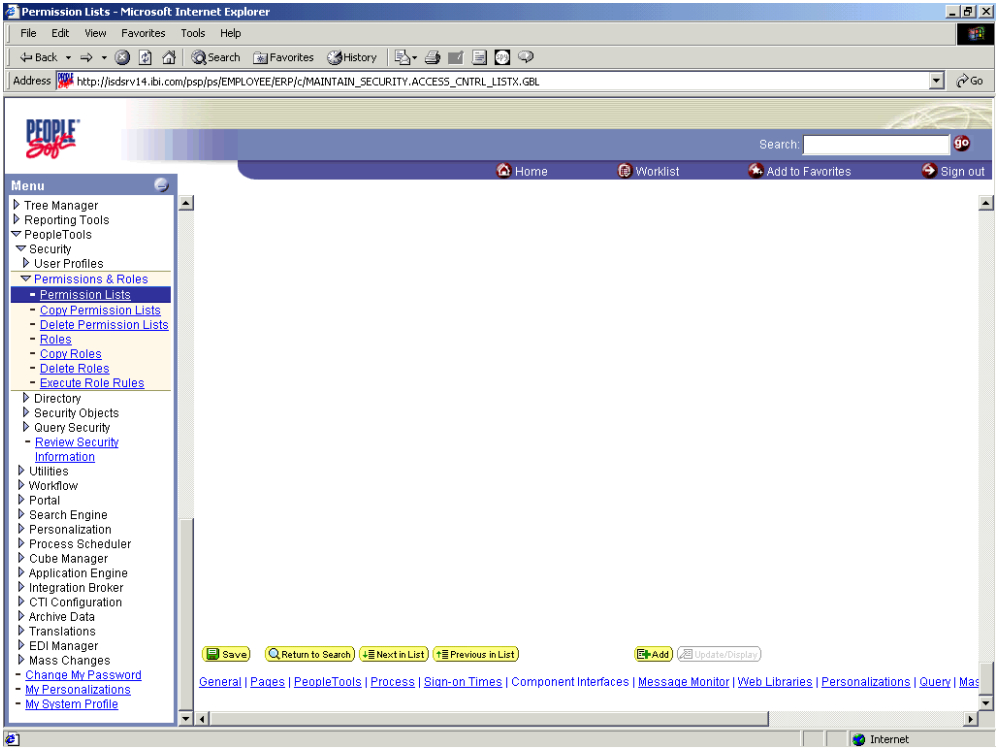
In this example, a component interface used by the adapter opens.

Figure 2-15 Component Interface Permissions Window



8. Select Full Access from the drop-down lists as the Method Access for the Get and Find methods.
9. Click OK.

Figure 2-16 Component Interfaces Window



10. After you have configured security for your component interface, scroll down to the bottom of the Component Interface Permissions window and click Save.

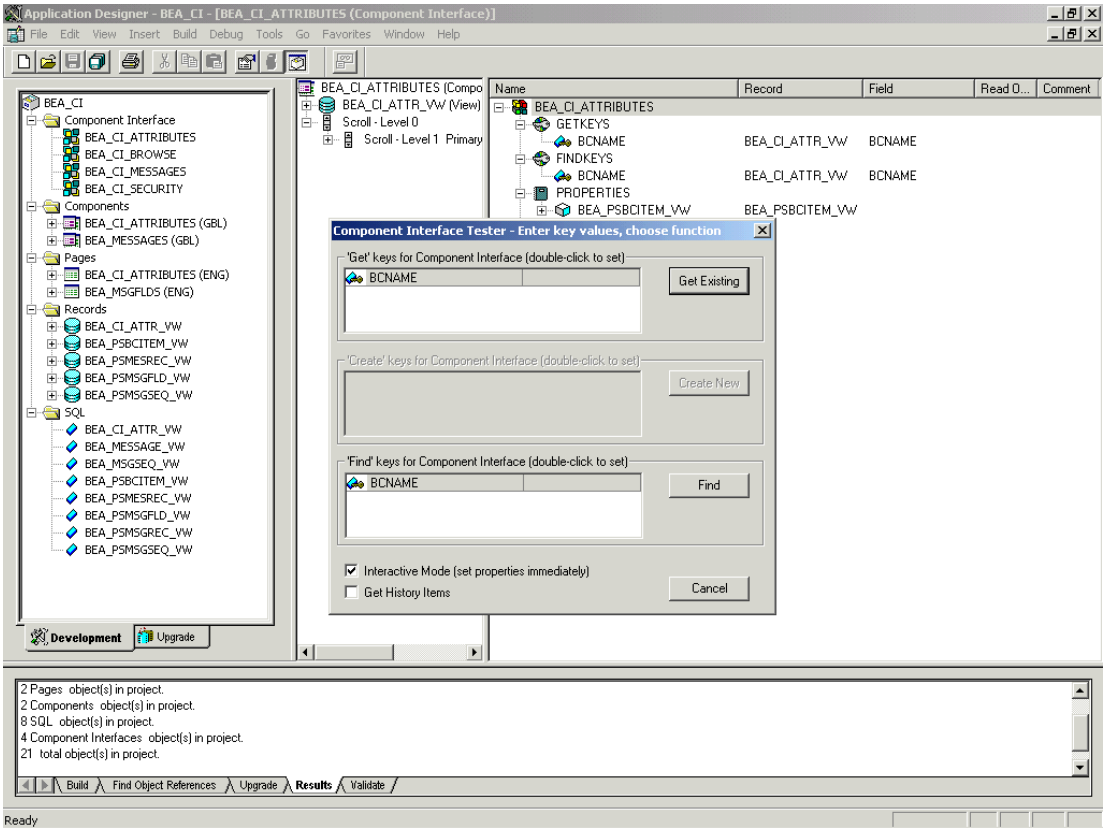
Testing a Component Interface

The BEA WebLogic Adapter for PeopleSoft 8 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 should be tested before being used as a source for the adapter. If changes are made to the underlying data by the user or by means of a PeopleSoft upgrade, and these changes invalidate a component interface, the user must repair the invalid component interface before the adapter uses it.

1. In Application Designer, choose Tools→Test Component Interface. The Component Interface Tester dialog box opens but is minimized.

2. Click the Component Interface Tester dialog box to bring it to the foreground.

Figure 2-17 Component Interface Tester

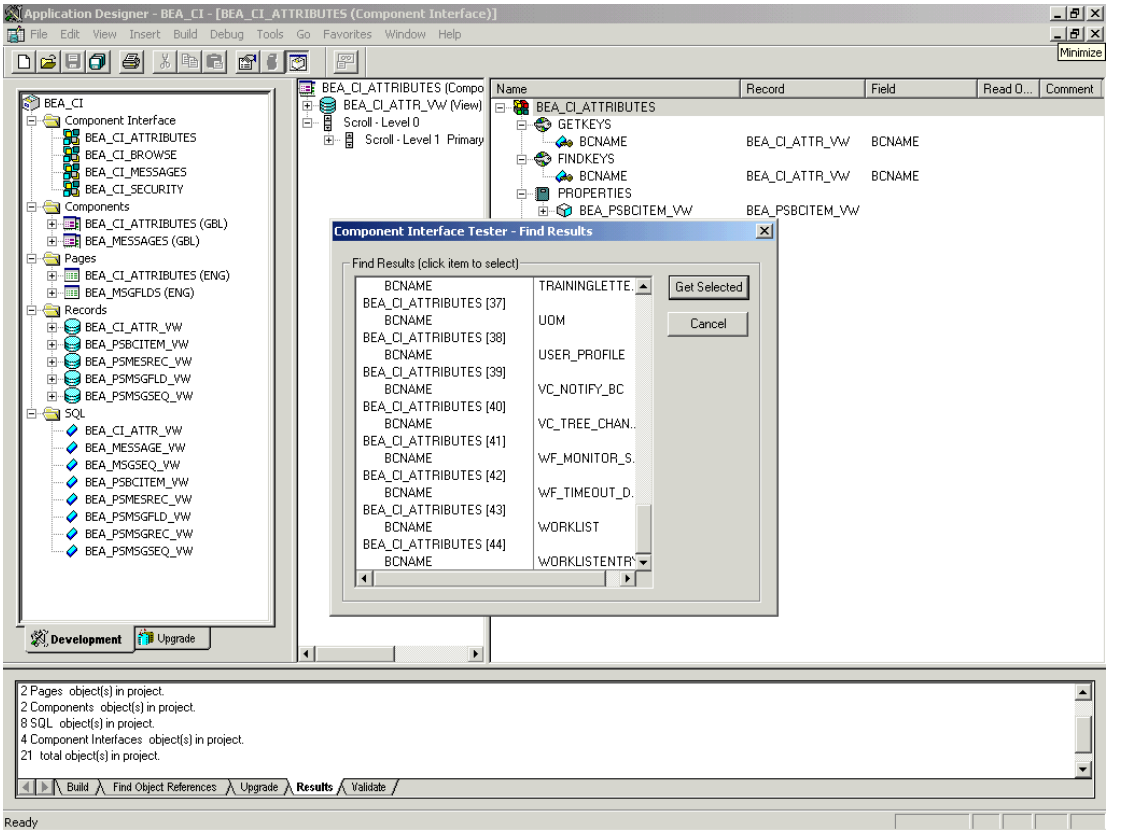


In this example, the Create New option is grayed out because the Add method is not applicable to this component.

Find Option

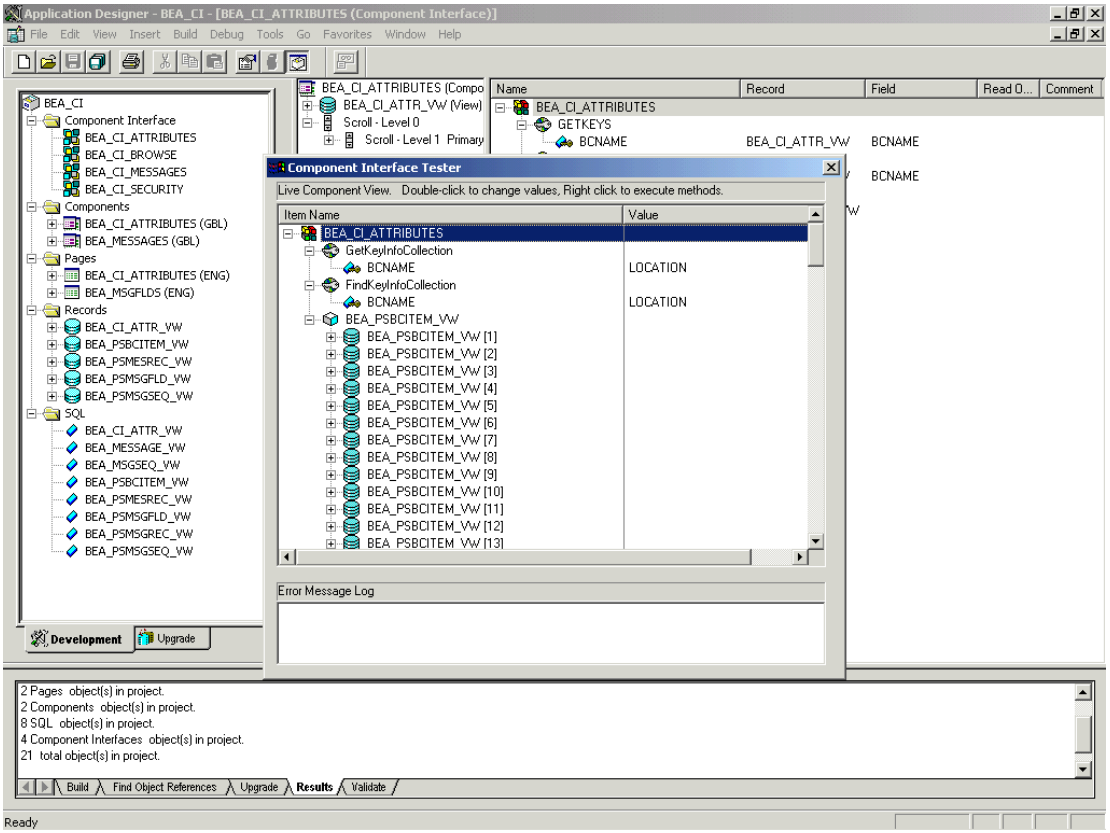
The Find option displays all possible entries for the underlying component.

Figure 2-18 Component Interface Tester - Find Results Box



When a field in the left pane is highlighted and you click the Get Selected button, the relevant data for that particular key appears.

Figure 2-19 Display of Data for Selected Key

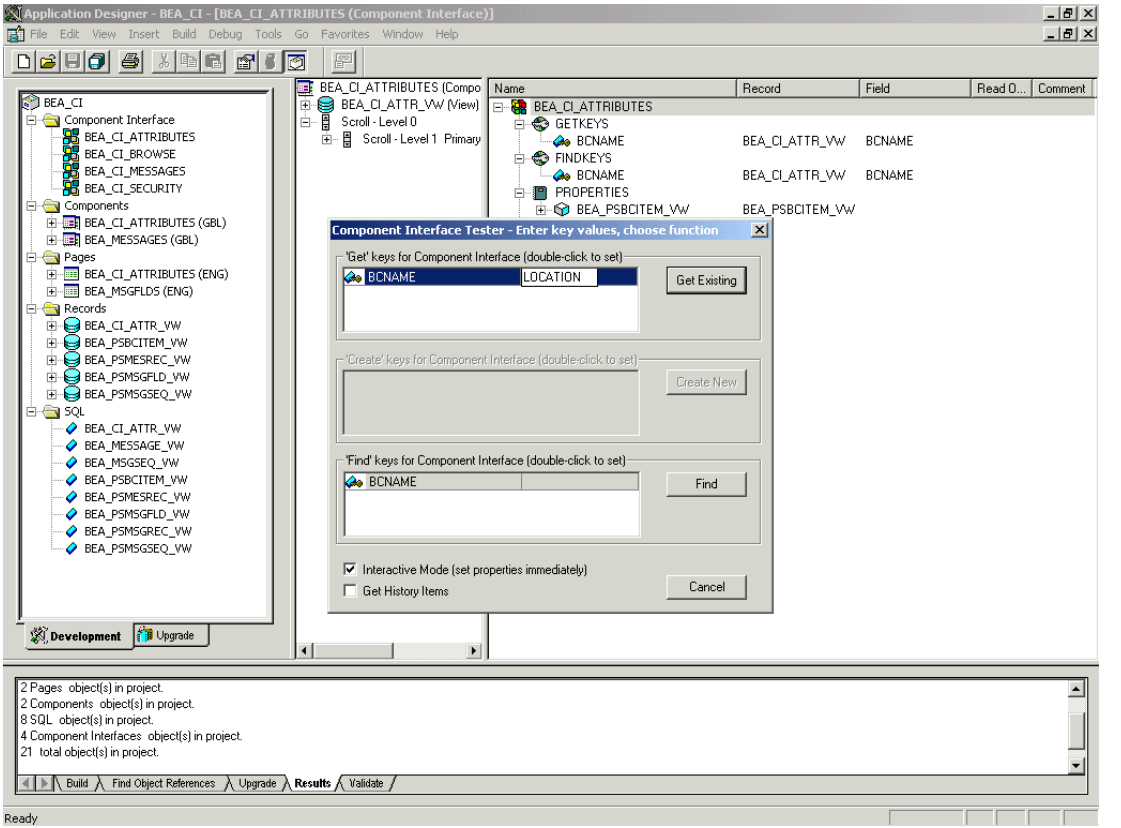


The values in the field could be changed had Read-Only access not been specified in the Permission List in use.

Get Option

To use the Get option, enter an existing Key by clicking the Get Existing button.

Figure 2-20 Key Selection to Expose Properties



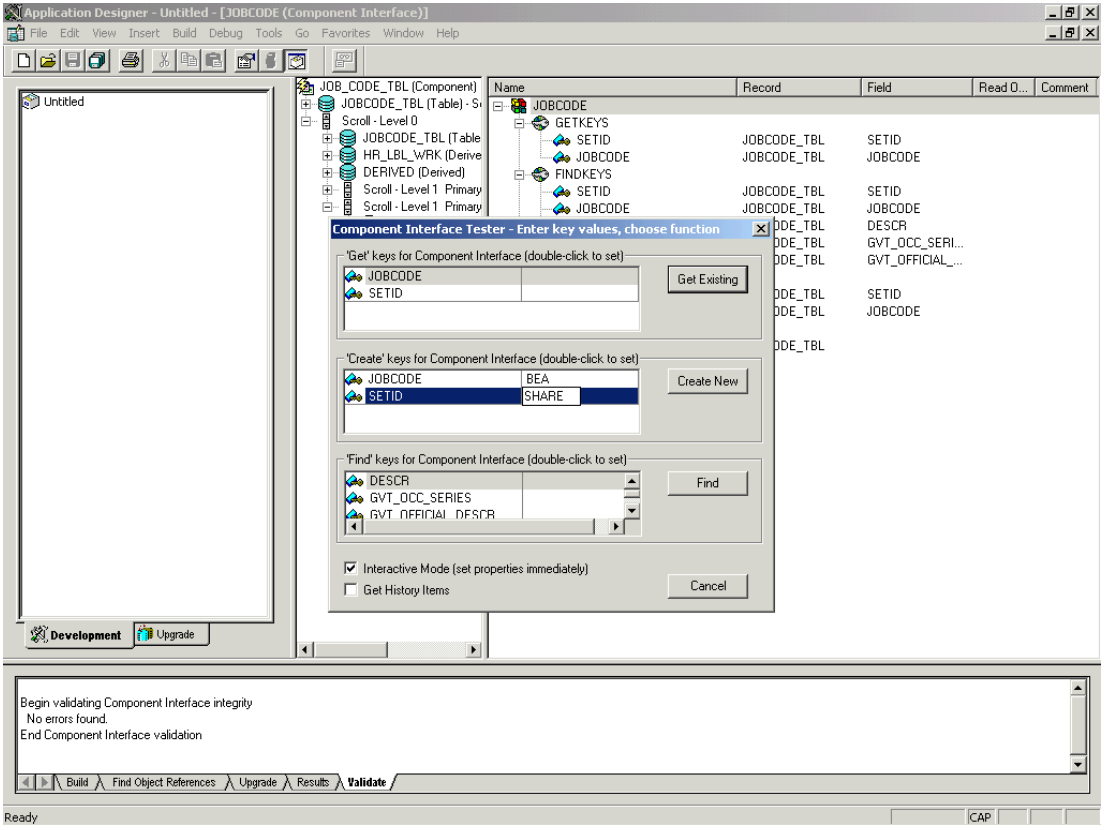
This returns the exposed properties for the key entered.

You can change values if Update access has been specified.

Create Option

The following Component Interface has all Methods enabled.

Figure 2-21 Component Interface Tester



The relevant ‘Create’ keys are entered.This enables the creating, updating, and selecting of the properties of the underlying component.

When valid values are entered in ‘Create’ keys, the JOBCODE Data Display window opens after the Table name is expanded with default data in place.

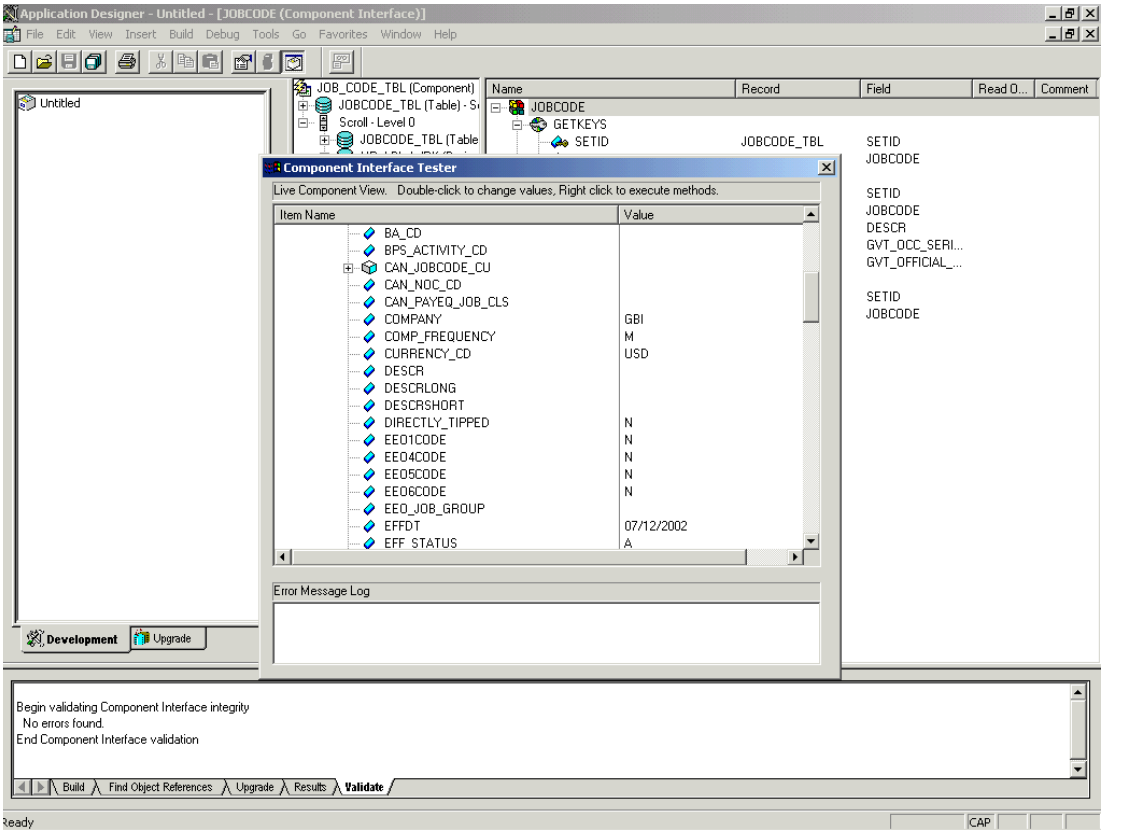
Any fields may be changed at this point. Any changes are validated against the underlying Business Logic for the component.

When you finish making changes, you can right-click the top item in the pane.

The Save option is displayed and used.

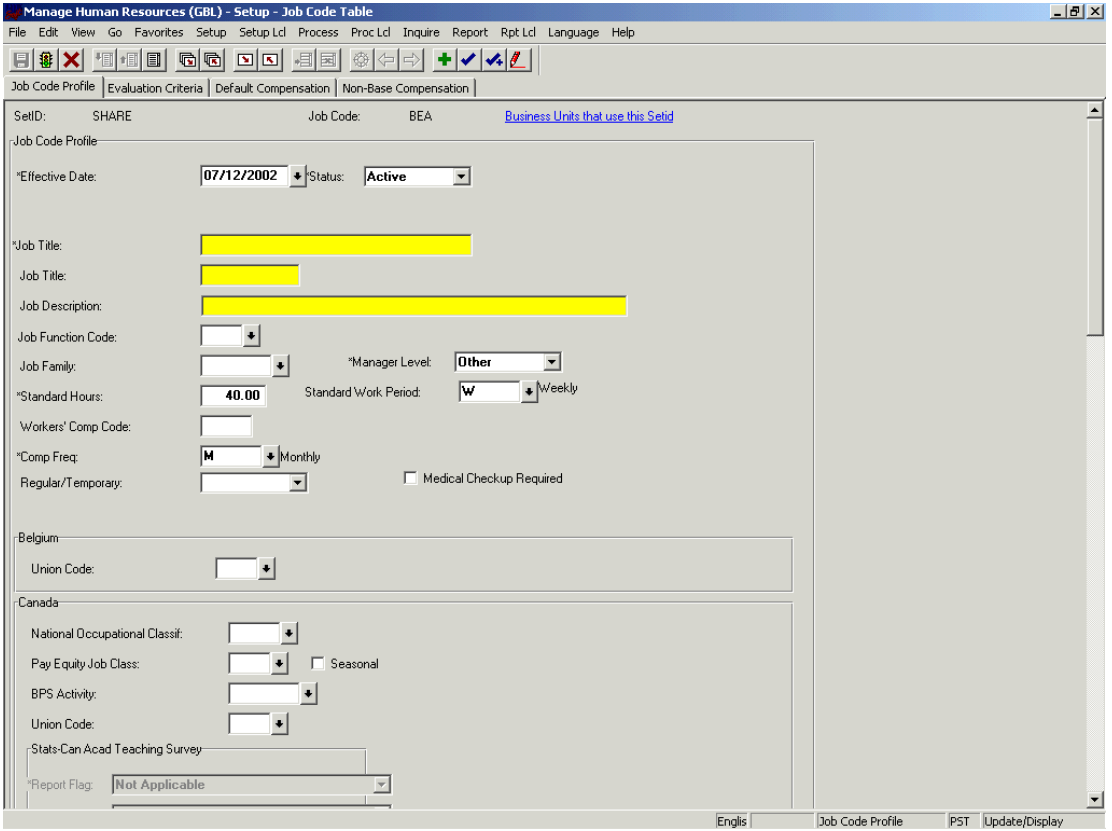
The keys used to create the record can be used with the Get Method for viewing data.

Figure 2-22 JOBCODE Data Display



The data added can be viewed in the PeopleSoft Component as shown in the following screen. Notice the Effective Date as one of the default values.

Figure 2-23 Job Code Profile



Generating Component Interface APIs

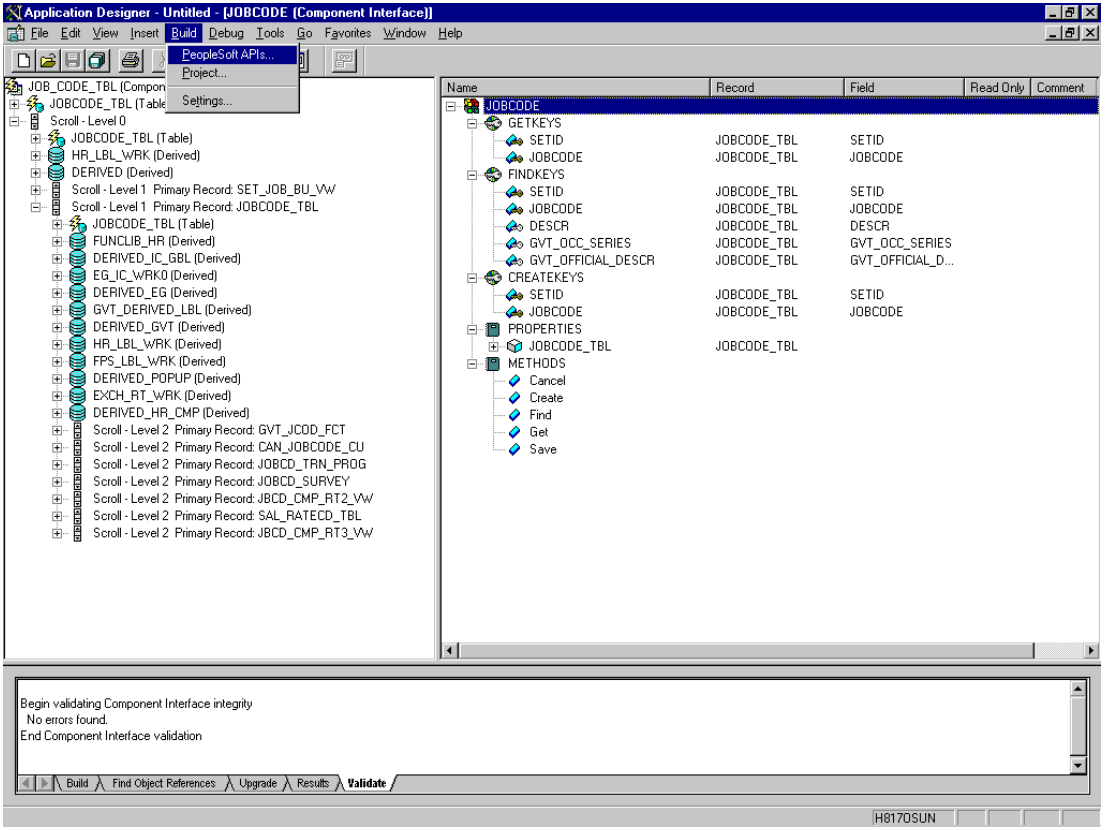
Regardless of whether you are using a PeopleSoft supplied Enterprise Integration Point (EIP) or a custom-developed Component Interface, a PeopleSoft API must be created to enable communications with the PeopleSoft application. This API is simply a collection of Java class files which reside on the client machine and intermediate between the client application layer and PeopleSoft.

Building the PeopleSoft API Java Programs

To build the PeopleSoft API Java programs:

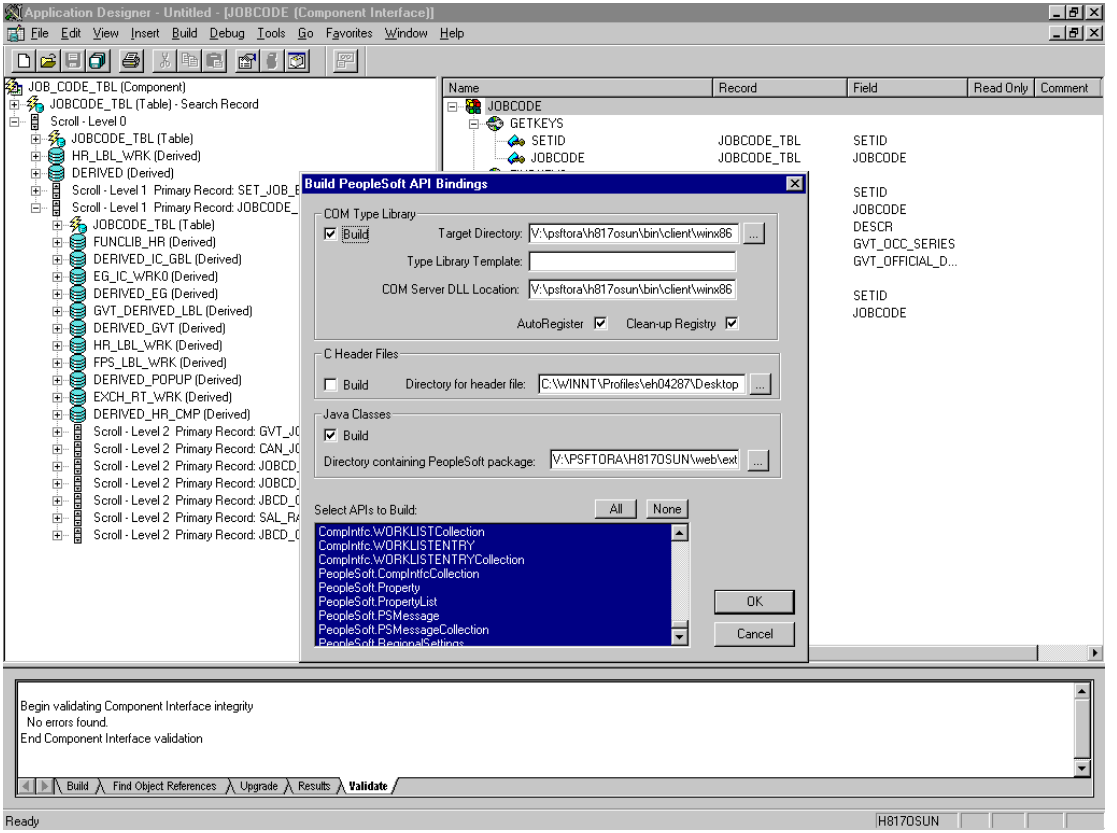
1. From the PeopleSoft Application Designer, open any Component Interface.
2. Click the right pane, then select Build→PeopleSoft APIs.

Figure 2-24 Choosing Build PeopleSoft APIs



You are prompted for the types of bindings to create.

Figure 2-25 Build PeopleSoft API Bindings Dialog Box



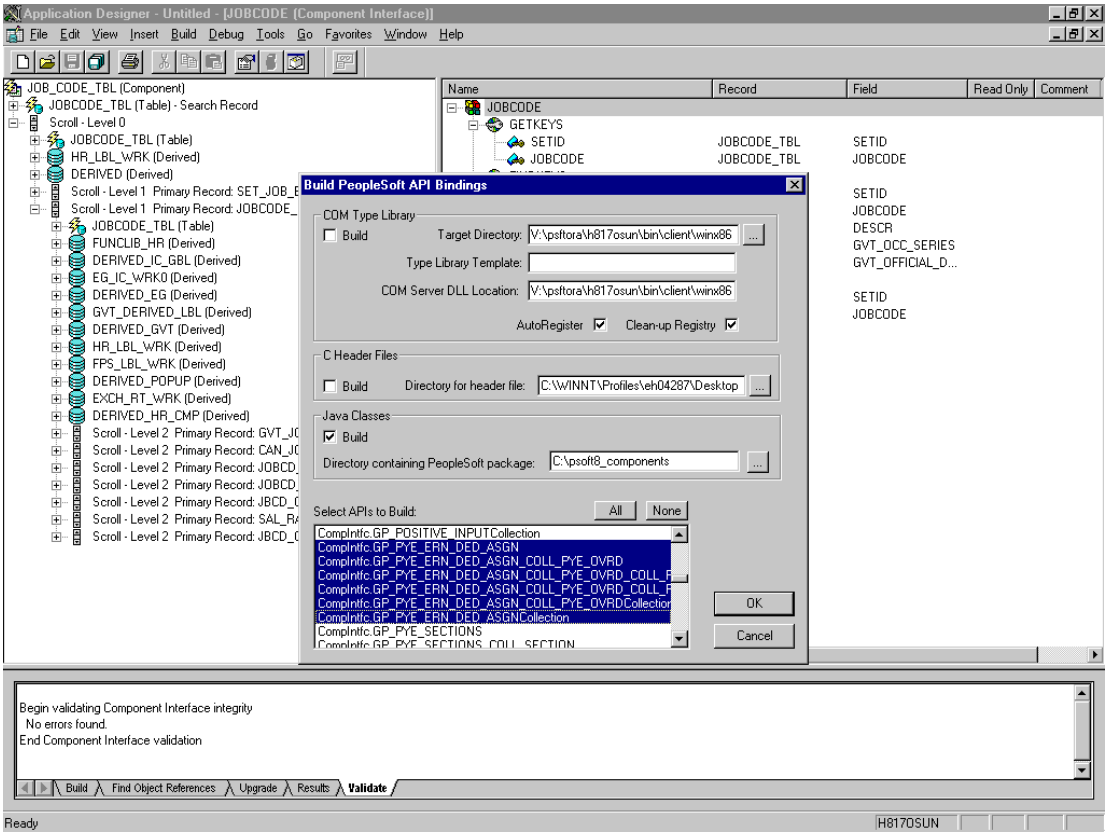
3. Since you are creating Java files, make sure to clear the selected COM Type Library Build.
4. Select a directory on your local machine where the Java files will be placed.

   The figures in this topic use `c:\psoft_components`.
5. Select the APIs to build.

   You may choose the default of ALL APIs (which can get quite large) or you may select individual APIs.

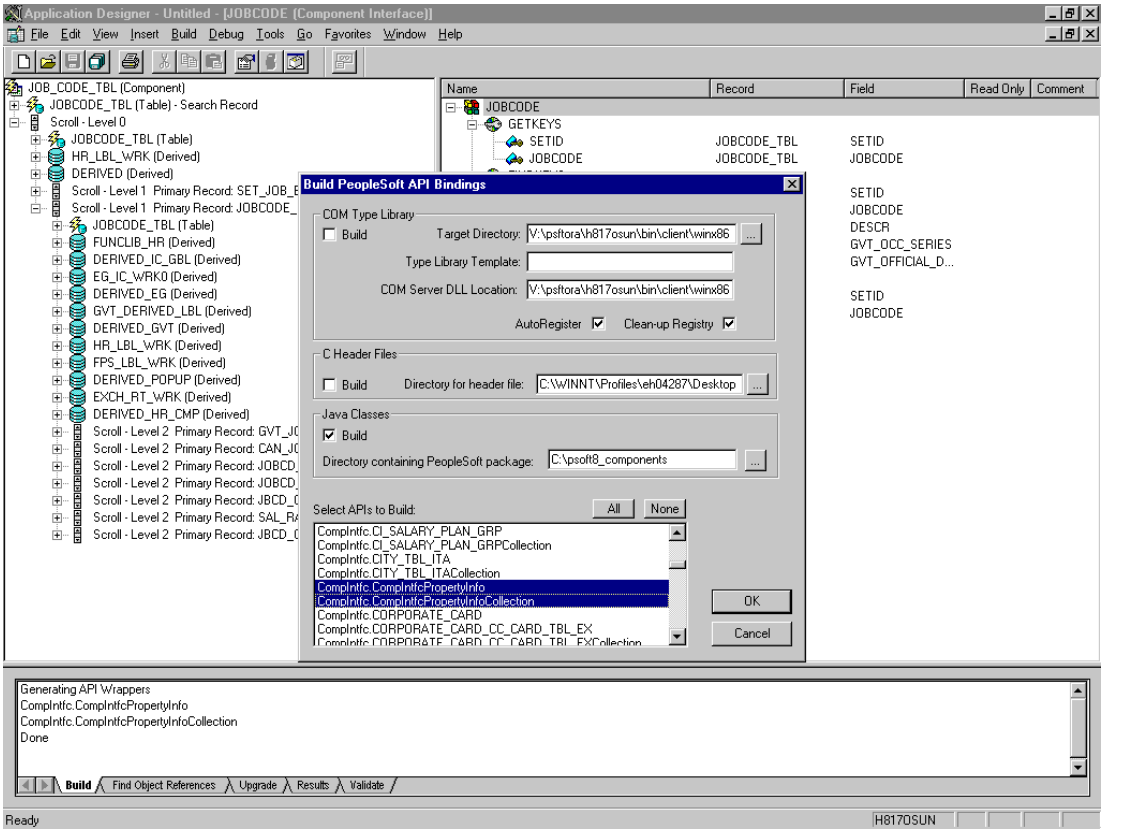
6. If you decide to:
   - Build all files, click OK. PeopleSoft will generate the files. This will take a few minutes. Once the process is complete, there will be a message in the output window. You are now ready to compile the Java files, as described in "Compiling the PeopleSoft API Java Programs" on page 34.
   - Create APIs for a specific Component Interface or Component Interfaces, continue with the following step.
7. Click None. This clears the selected APIs.
8. Select the APIs appropriate for your Component Interface. These will all begin with the name of your Component Interface. There may be fewer than five or more than 50 APIs to build for a particular Component Interface. The following figure illustrates the `GP_PYE_ERN_DED_ASGN` Component Interface from the HR 8.1 application.

Figure 2-26 Selecting APIs for a Component Interface



9. In addition to the APIs for your chosen Component Interface, you must also generate the API files for the generic Component Interface properties. You may select these items in the same step as the Component Interface build or they may be done separately.

Figure 2-27 Selecting Generic API Files for Component Interfaces

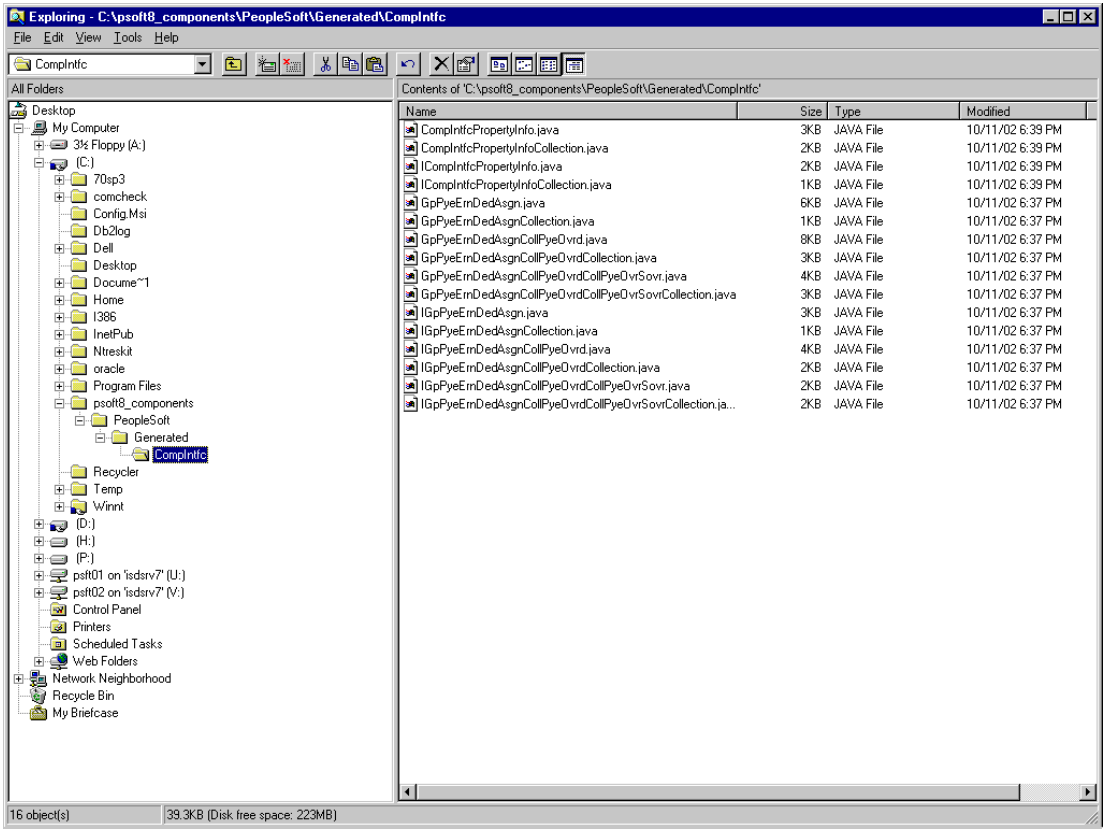


10. Click OK. PeopleSoft will generate the files. This will take a few minutes. Once the process is complete, there will be a message in the output window. You are now ready to compile the Java files, as described in "Compiling the PeopleSoft API Java Programs" on page 34.

Compiling the PeopleSoft API Java Programs

PeopleSoft places the Java programs to be compiled in a directory structure *psoft8_components*\PeopleSoft\generated\CompIntfc; here, *psoft8_components* is the directory specified during the build process. If you chose to generate all APIs, note that a second directory, *psoft8_components*\PeopleSoft\generated\PeopleSoft, is also created; you do not need to access it.

Figure 2-28 Browsing PeopleSoft\generated\CompIntfc



Notice that there are two Java programs for every API file that you had selected when you built the Java programs, as described in "Building the PeopleSoft API Java Programs" on page 29.

To compile the PeopleSoft API Java programs:

1. Before you can compile the Java programs, you need the PeopleSoft Java Object Adapter, `psjoa.jar,` which you can find on your PeopleSoft Application Server under the `PS_HOME\Web\psjoa` directory. Copy `psjoa.jar` from the PeopleSoft Application Server to your local machine, and ensure that it is in the Java class path when you compile.

2. Compile the Java programs.

   When you compile, you must remember to respect the `\PeopleSoft\generated\CompIntfc` pathing.

   The following sample compile code would be placed in the `psoft8_components` directory; the code assumes that `psjoa.jar` is also 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
   ```

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

3. Regardless of where you place the class file, the CLASSPATH variable in the `startWebLogic.cmd` file must be set to point to the directory directly above `\PeopleSoft\generated\CompIntfc`. For example, if you were to keep the class files in the `psoft8_components` directory, then `psoft8_components` would be in the class path.

   Alternatively, you may compress the class files into a `jar` file and place the `jar` file in your class path.

The following code, if placed in the `psoft8_components` directory, will create a 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
```

3 Creating BEA Schemas for PeopleSoft Component Interfaces

This section provides the information you need to create schemas for PeopleSoft component interfaces. It includes the following topics:

Services

With the BEA WebLogic Adapter for PeopleSoft 8, you can access and integrate PeopleSoft Business Objects using Component Interfaces and the Java API. The execution of a component begins with the receipt of a service request document and, in most cases, the result is an XML response document indicating a result with an associated status code.

Before you can invoke an adapter service, you must create BEA request and response schemas for the service. Use the BEA Application Explorer to generate these schemas directly against a PeopleSoft Component Interface.

The following topic illustrates how to create BEA schemas for the service adapter. For more information on the BEA Application Explorer, also see the *BEA Application Explorer Installation & Configuration Guide*.

1. Open the BEA Application Explorer:

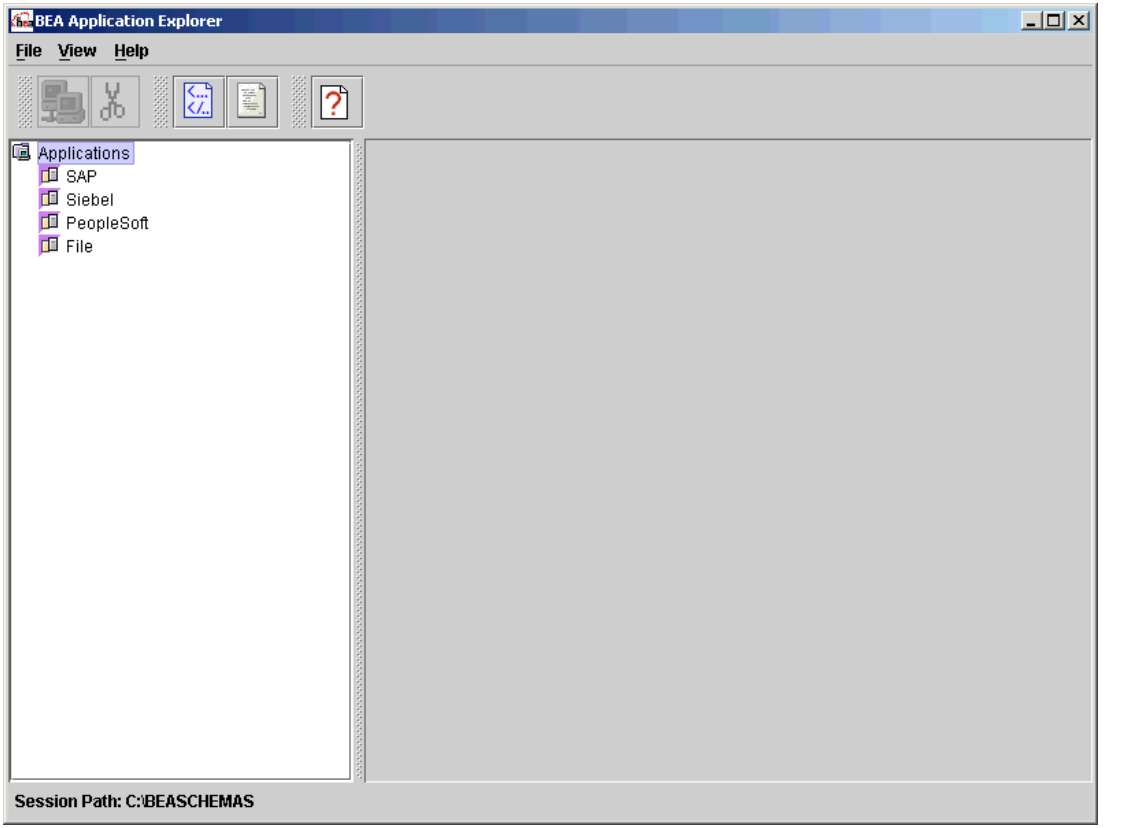
Figure 3-1 BEA Application Explorer

Establishing the Working Directory

You should establish the directory associated with your WebLogic Integration server for use to import event and service XML schemas into the application view repository.

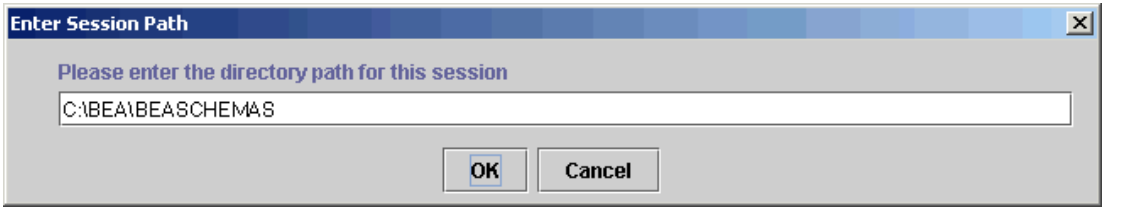
1. On the Tools menu, click File and select Session.

Figure 3-2 BEA Application Explorer Session



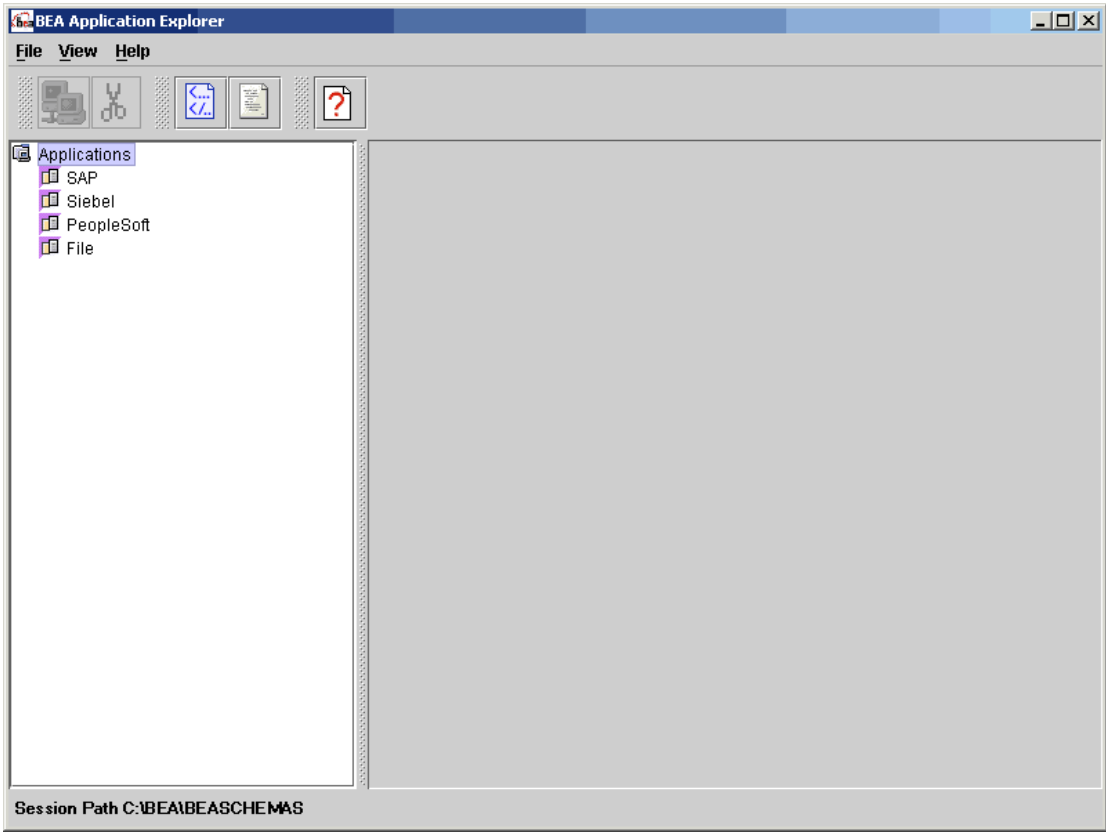
2. Enter a folder name. In the following example, `C:\BEA\BEASCHEMAS` serves as the BEA Application Explorer working directory. This is the location of the generated schemas.

Figure 3-3 Session Path Dialog Box



3. Click OK.

Figure 3-4 Explorer Session with Path Established

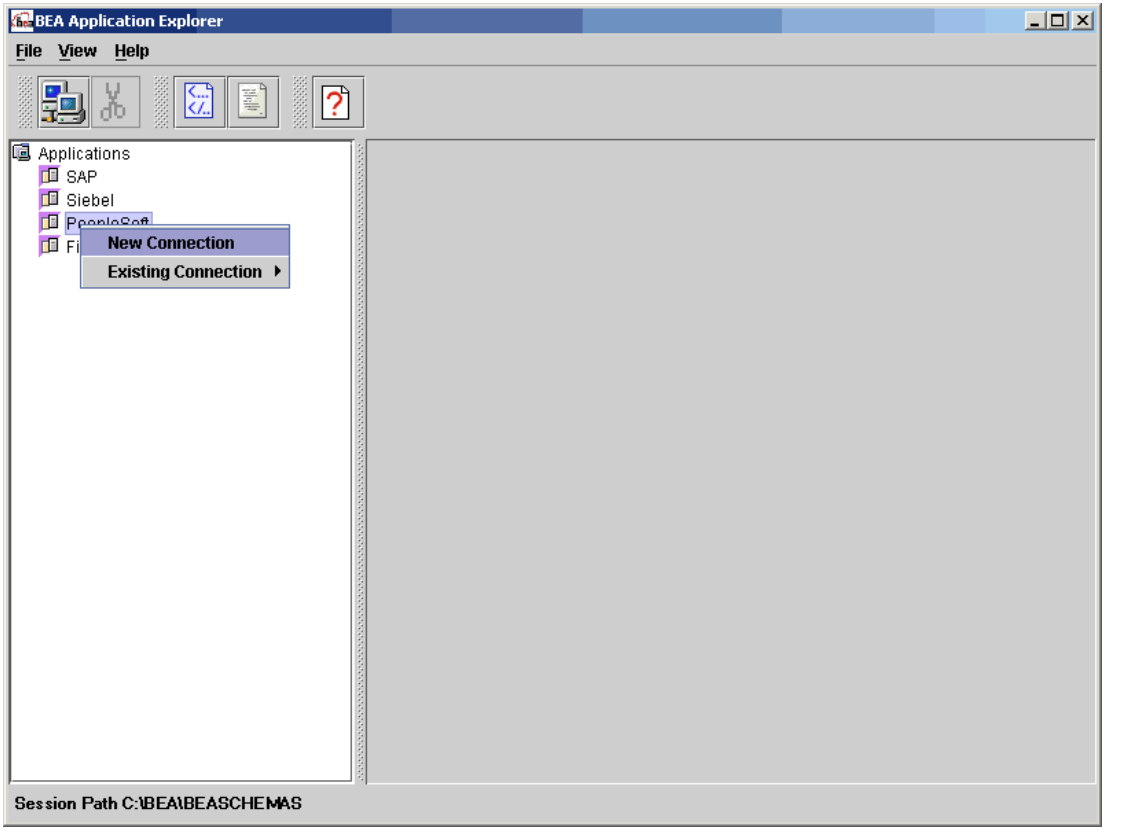


Notice the Session Path at the bottom of the Explorer window.

Establishing a Connection to PeopleSoft

1. Click PeopleSoft to enter a new connection.

Figure 3-5 Connection to PeopleSoft

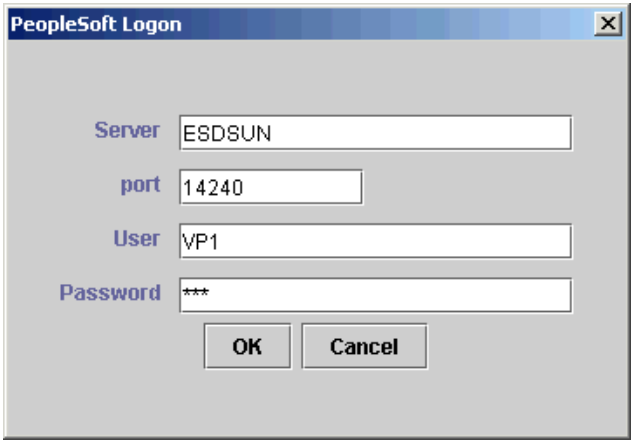


2. Enter a name for the PeopleSoft connection. For example: PeopleSoftConnection.

Figure 3-6 PeopleSoft Connection Name

3. Click OK.

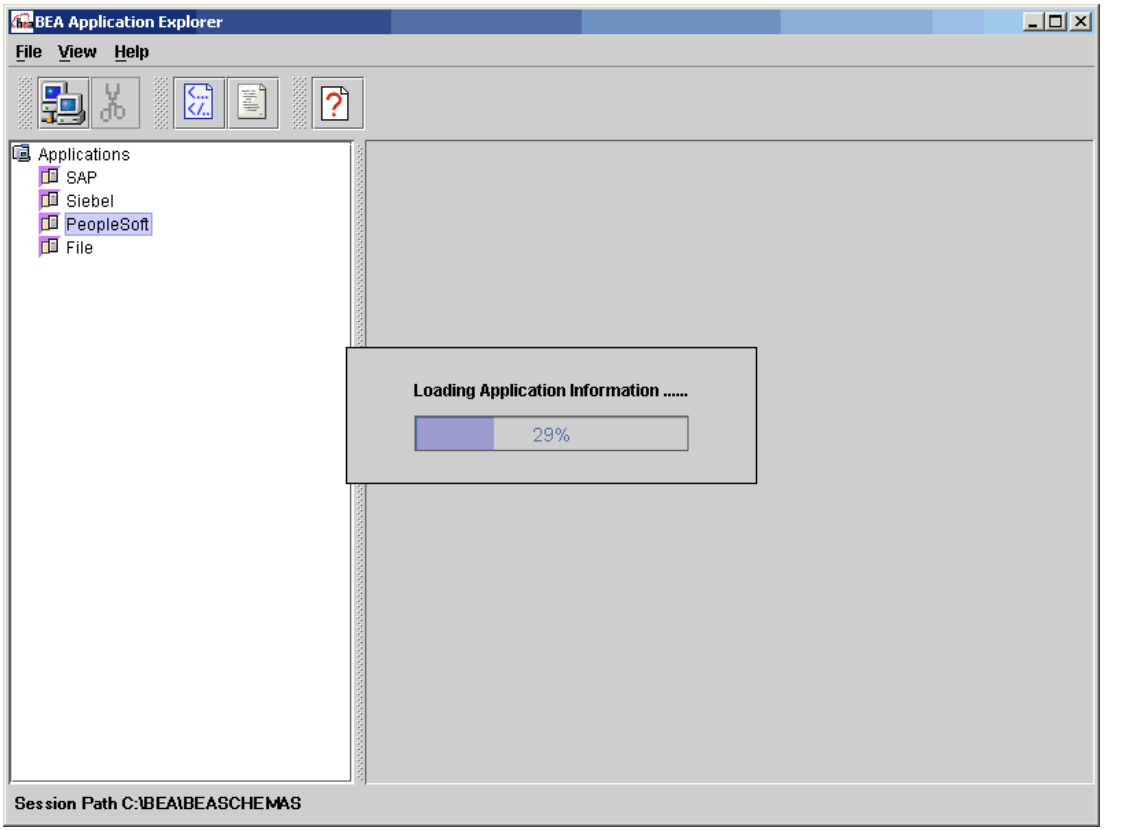
Figure 3-7 PeopleSoft Logon



4. Enter the parameters required for PeopleSoft Client applications to connect to PeopleSoft:
   - PeopleSoft Application Server
   - PeopleSoft Application Server Port
   - User
   - Password

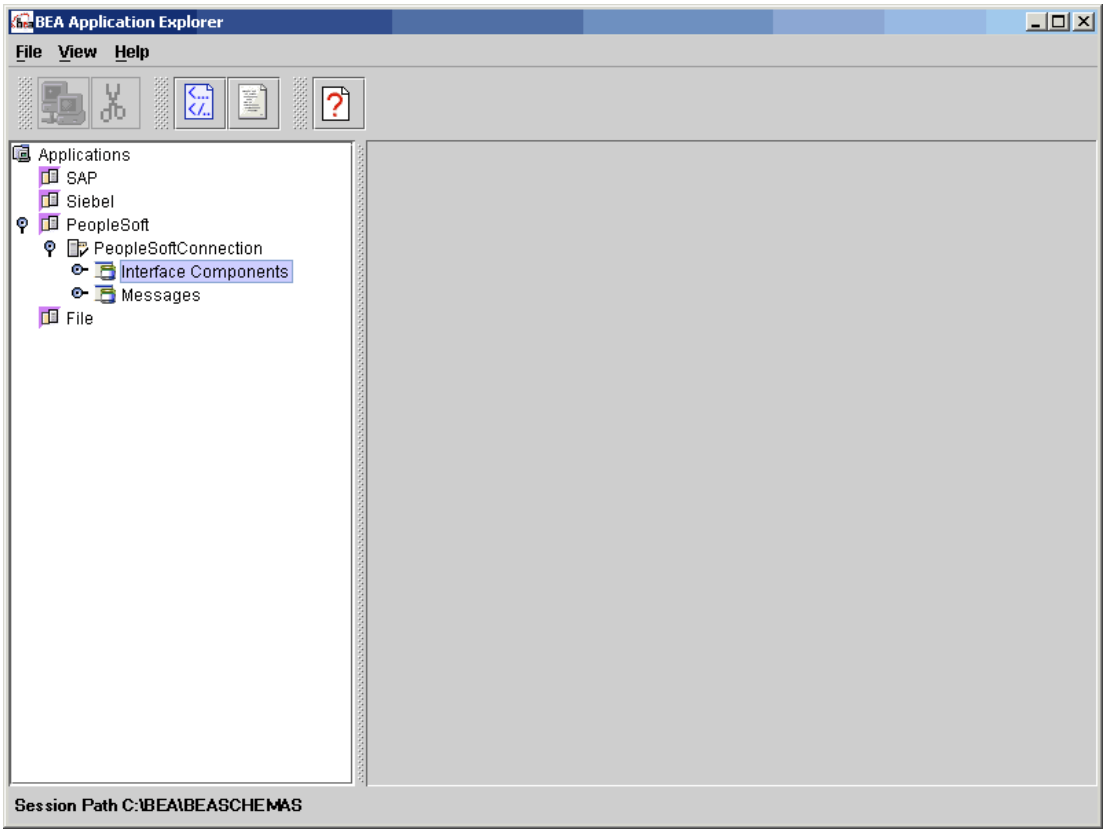
   For more information, see your PeopleSoft documentation or consult your PeopleSoft System Administrator.
5. Click OK.

Figure 3-8 Application Load Process



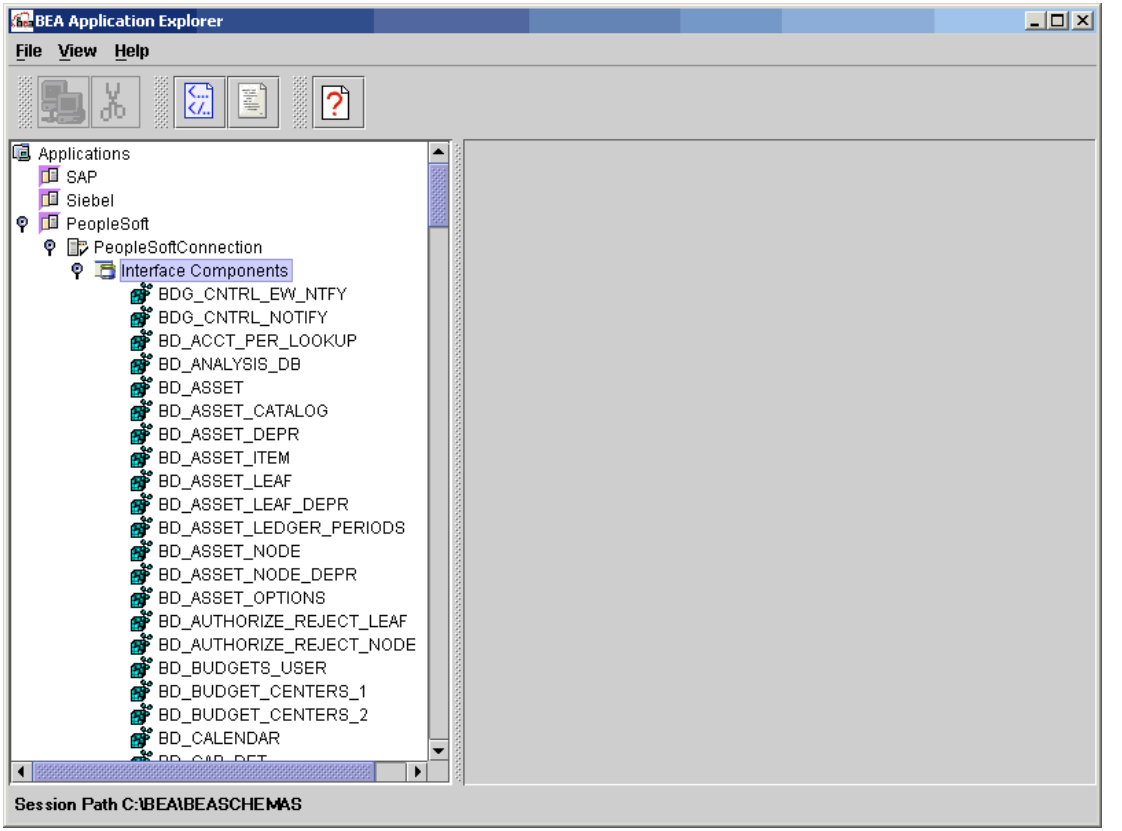
The process of loading the internal cached file may take several minutes. This speeds the process for subsequent displays and schema creations for other Component Interfaces. Once connected, the BEA Application Explorer displays a list of Component Interfaces.

Figure 3-9 Business Objects



6. Expand Business Objects to browse available Component Interfaces.

Figure 3-10 Available Business Objects



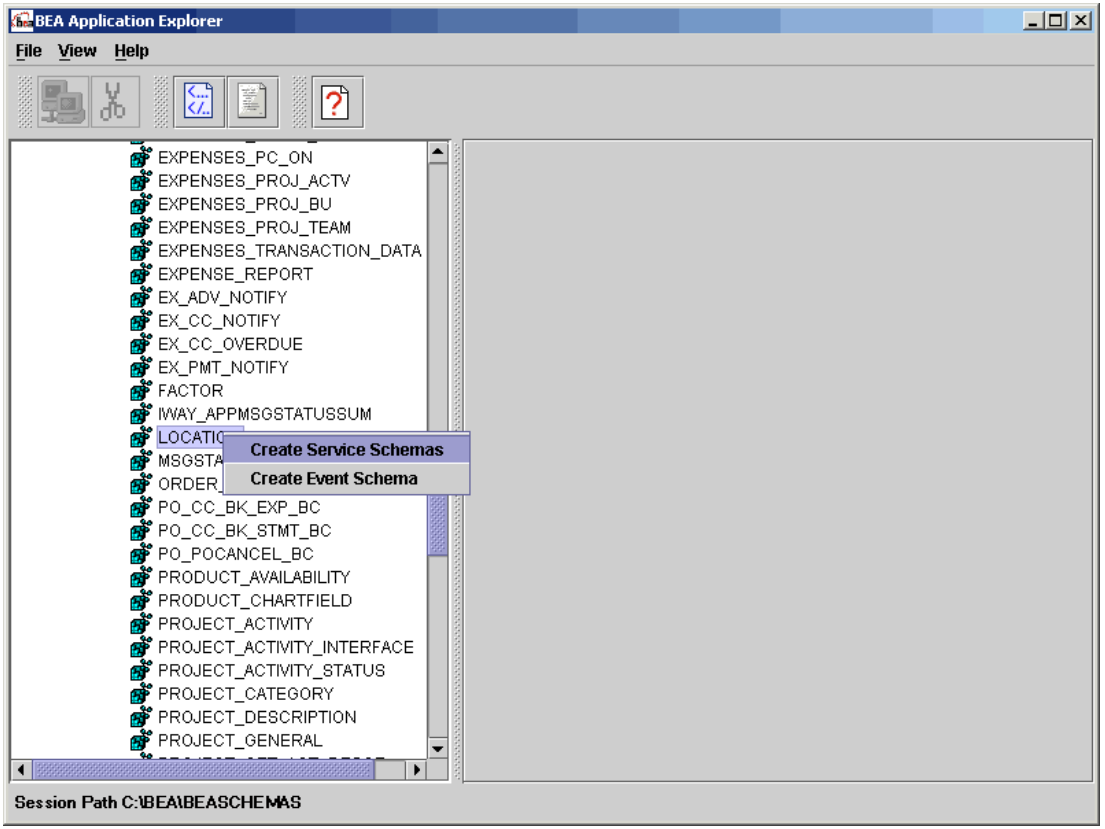
7. Select a component and right-click to generate service request and response schemas. The BEA Application Explorer generates:
   - Service XML request schema
   - Service XML response schema

Component Interface LOCATION

To generate service request and response schemas for the Component Interface LOCATION:

1. Right-click LOCATION.

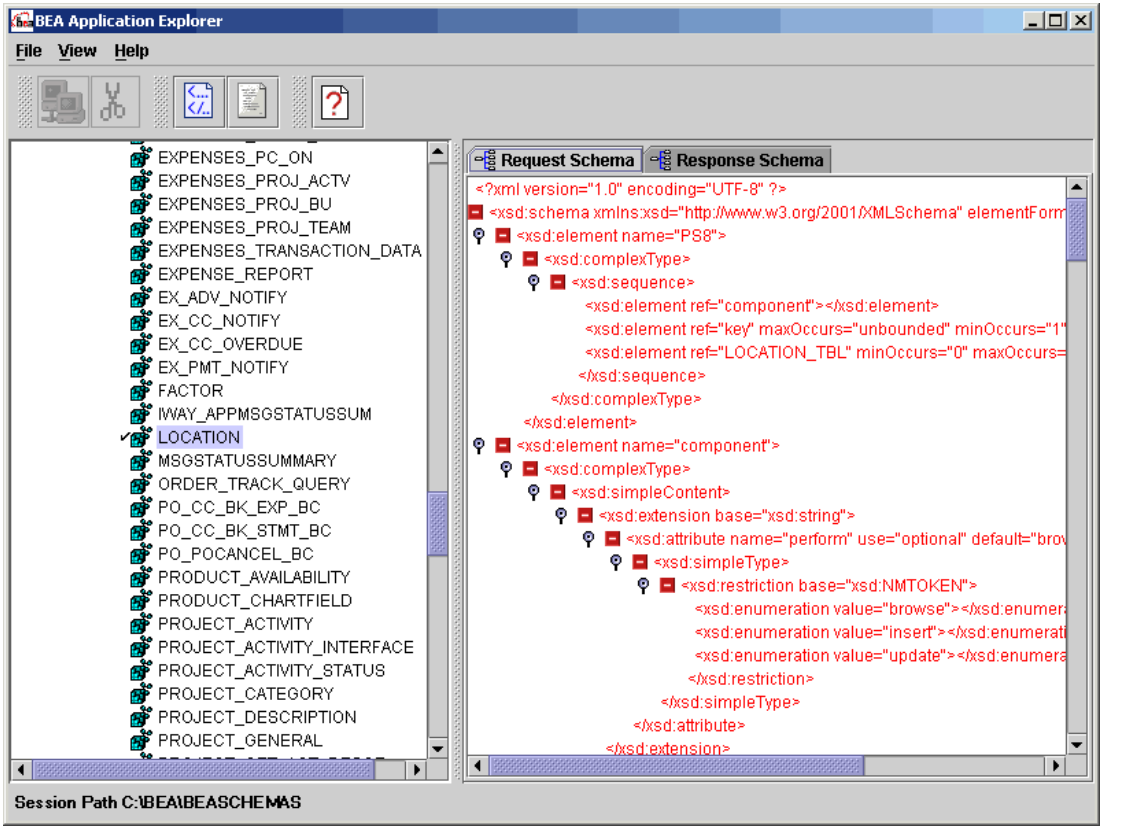
Figure 3-11 Schema Creation



2. Select Create Service Schemas. The BEA Application Explorer accesses the PeopleSoft repository to build XSD schemas, which are then published to the WebLogic Integration repository.

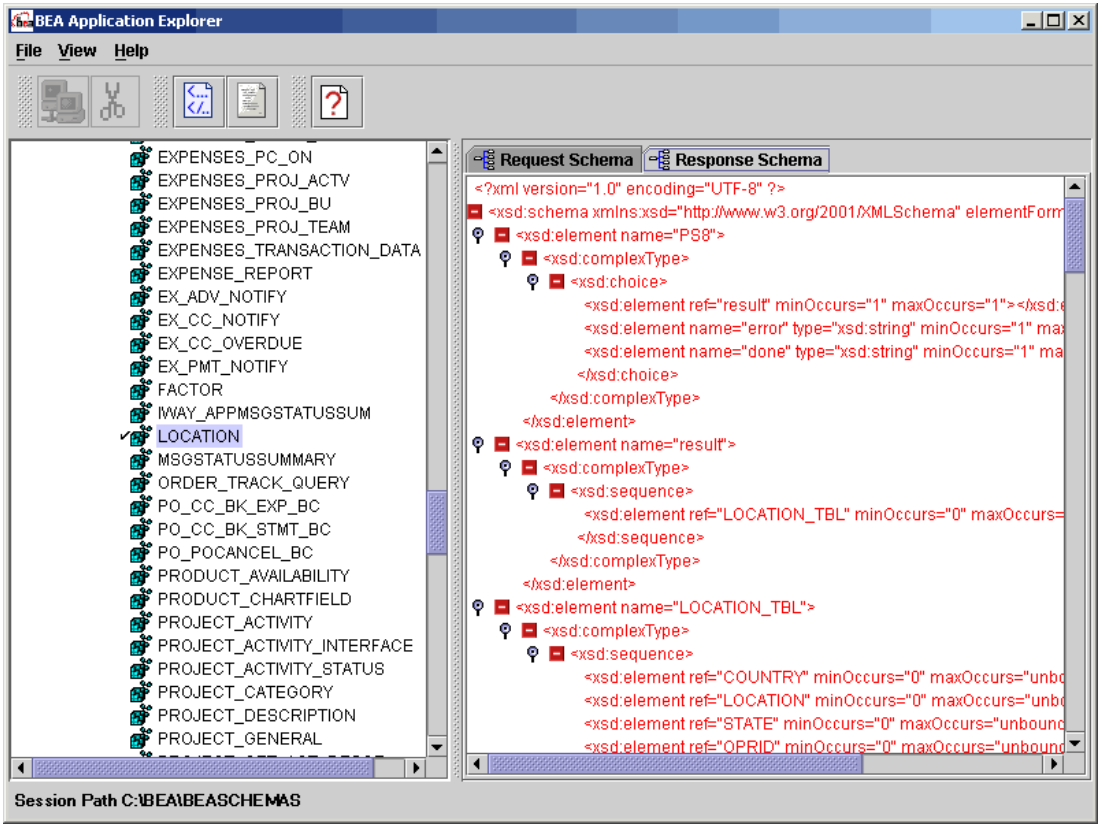
You can view the request schema:

Figure 3-12 Request Schema



You can also view the response schema:

Figure 3-13 Response Schema



A directory structure is created automatically within the working directory, `C:\BEA\BEASCHEMAS`. The BEA Application Explorer creates a folder called PeopleSoft with subfolders for each configured PeopleSoft connection. These subfolders contain the schemas created for each connection.

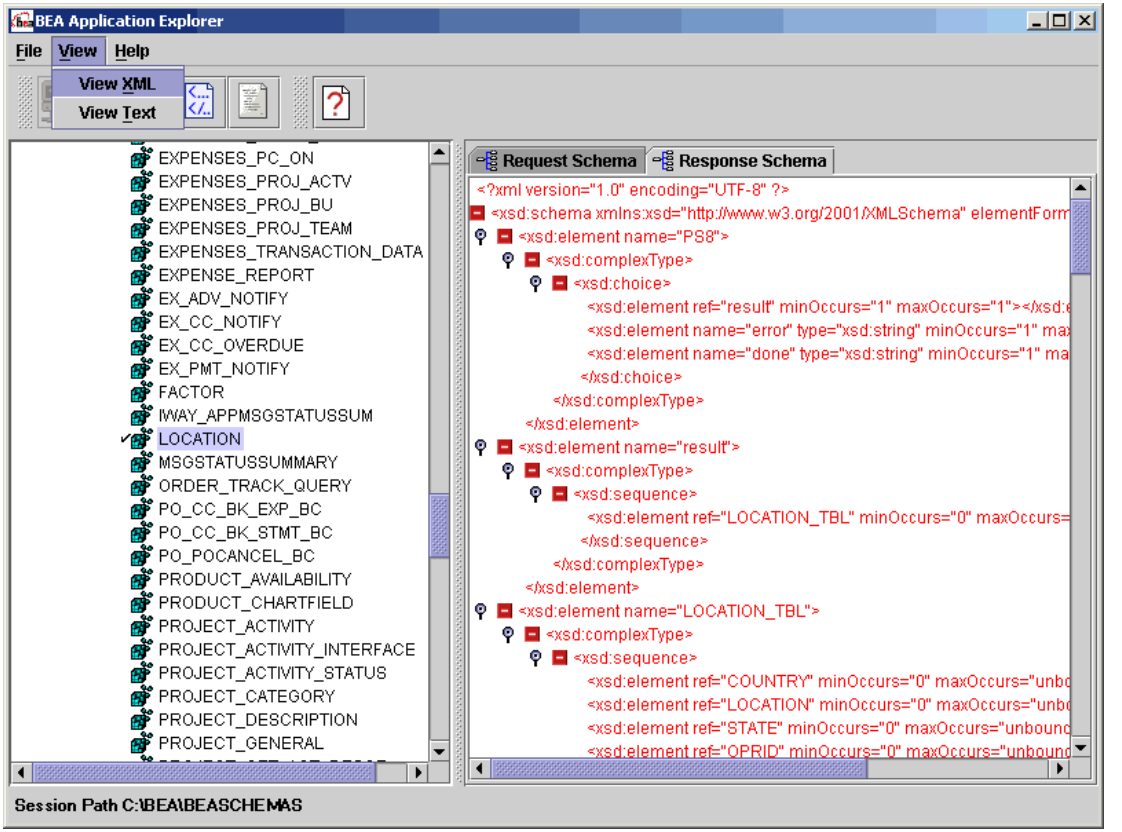
In this case, the schemas are located in the folder called PeopleSoftConnection. This is the connection name you established when you connected to PeopleSoft using the BEA Application Explorer.

The following members have been added:

- `manifest.xml`
- `service_LOCATION.xsd`
- `service_LOCATION_response.xsd`

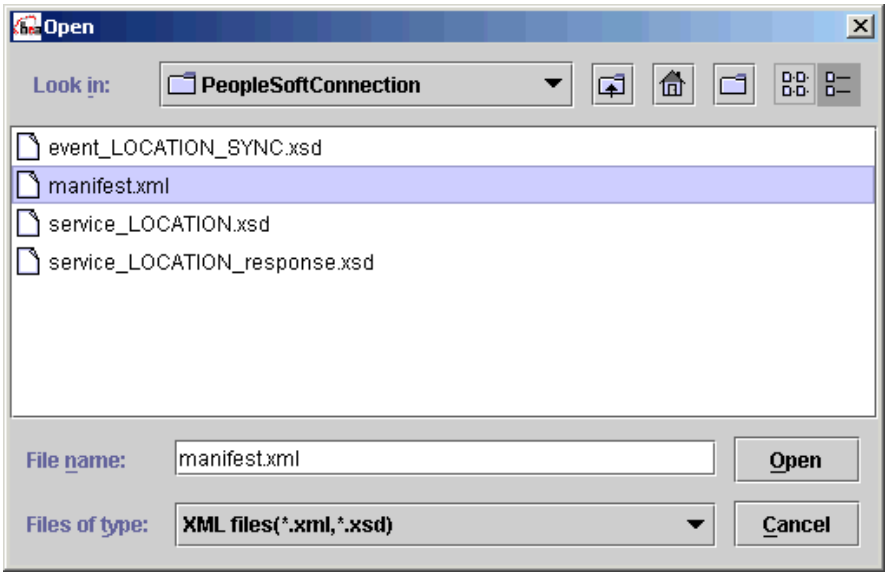
You can also view the created schemas and manifest files using the BEA Application Explorer. Point to the BEA Application Explorer working directory (View XML) to select an XML file to view.

Figure 3-14 XML View



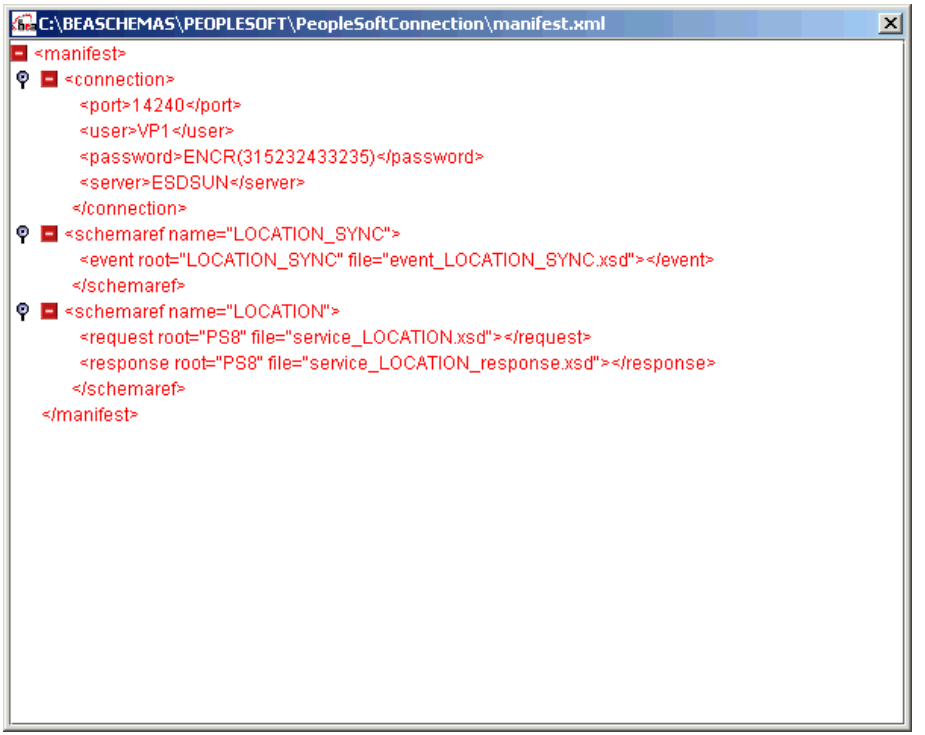
3. Select the `manifest.xml` file.

Figure 3-15 File Selection



For example, the `manifest.xml` file for the component interface LOCATION contains the connection and configuration information. This can be used to test access to PeopleSoft using the WebLogic Integration JSP console test pages.

Figure 3-16 manifest.xml



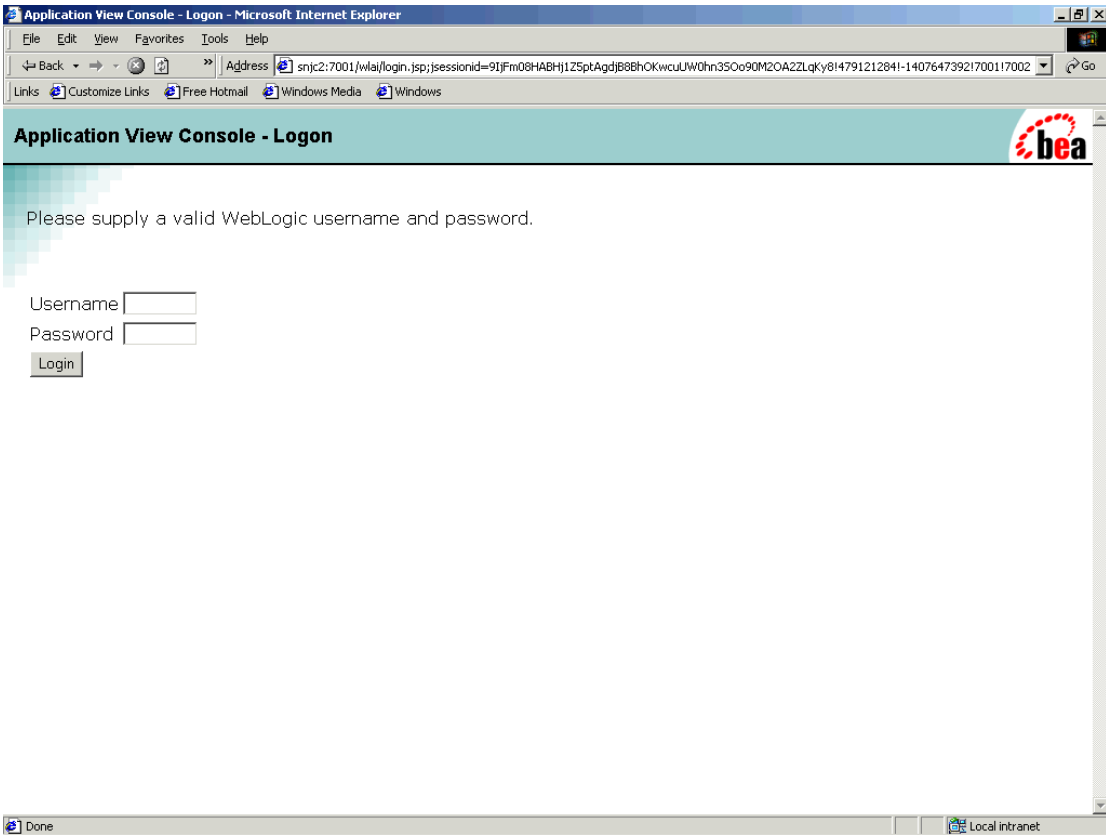
4 Creating and Deploying Application Views

This section describes how to create XML-based interfaces between WebLogic Servers and specific Enterprise Information System (EIS) applications. It includes the following topics:

Creating Application Views for PeopleSoft Component Interfaces

When you define an application view, you are creating an XML-based interface between WebLogic Server and a particular EIS application within your enterprise. After you create the application view, a business analyst can use it to create business processes that use the application. For any adapter, you can create any number of application views, each with any number of services and events.

Figure 4-1 Application View Console - Logon Window



To log on to the WebLogic Integration Application View Console:

1. Enter a User Name and Password.

Note: If the user name is not `system`, it must be included in the `adapter` group. For more information on adding the administrative server user name to the `adapter` group, see the BEA Application Explorer *for PeopleSoft Installation and Configuration Guide*.

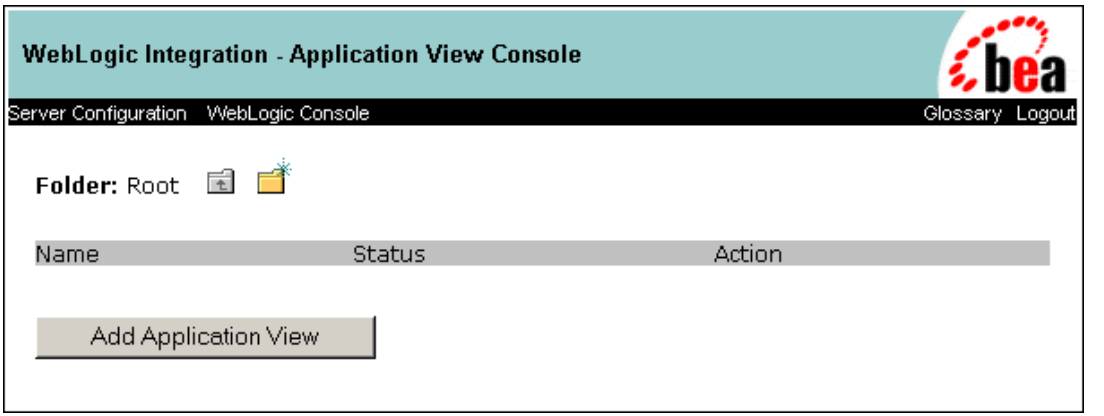
2. Click Login.

For more information, see "Logging On to the WebLogic Integration Application View Console" in "Defining an Application View" in *Using Application Integration*:

- For WebLogic Integration 7.0, see `http://edocs.bea.com/wli/docs70/aiuser/2usrdef.htm`
- For WebLogic Integration 2.1, see `http://edocs.bea.com/wlintegration/v2_1sp/aiuser/2usrdef.htm`

The Application View Console can be found at `http://`*`host:port`*`/wlai`. Here, *`host`* is the IP address or DNS name where WebLogic Integration Server is installed, and *`port`* is the socket on which the server is listening. The default port, if not changed at install time, is 7001.

Figure 4-2 Application View Console

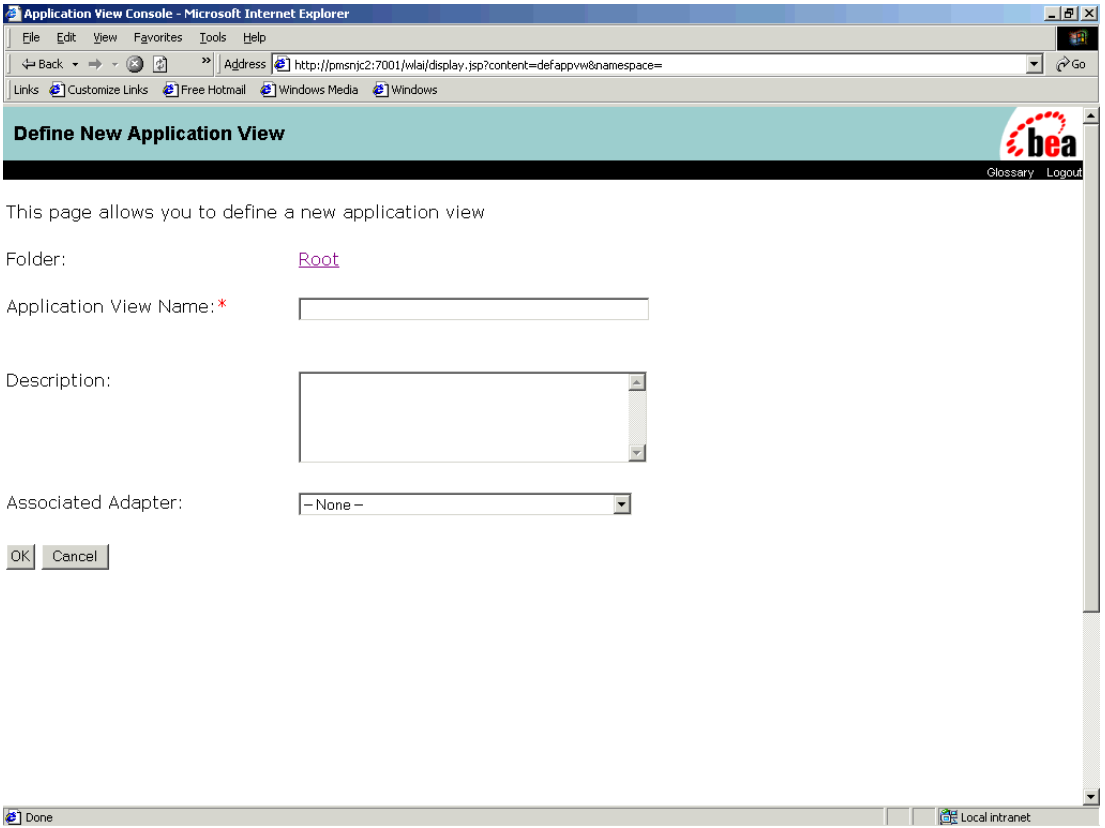


3. Click Add Application View to create a new application view for the appropriate adapter. An application view enables a set of business processes for this adapter's target EIS application. For more information, see "Defining an Application View" in *Using Application Integration*:

   - For WebLogic Integration 7.0, see `http://edocs.bea.com/wli/docs70/aiuser/2usrdef.htm`
   - For WebLogic Integration 2.1, see `http://edocs.bea.com/wlintegration/v2_1sp/aiuser/2usrdef.htm`

   The Define New Application View window opens.

Figure 4-3 Application View Console - Define New Application View Window



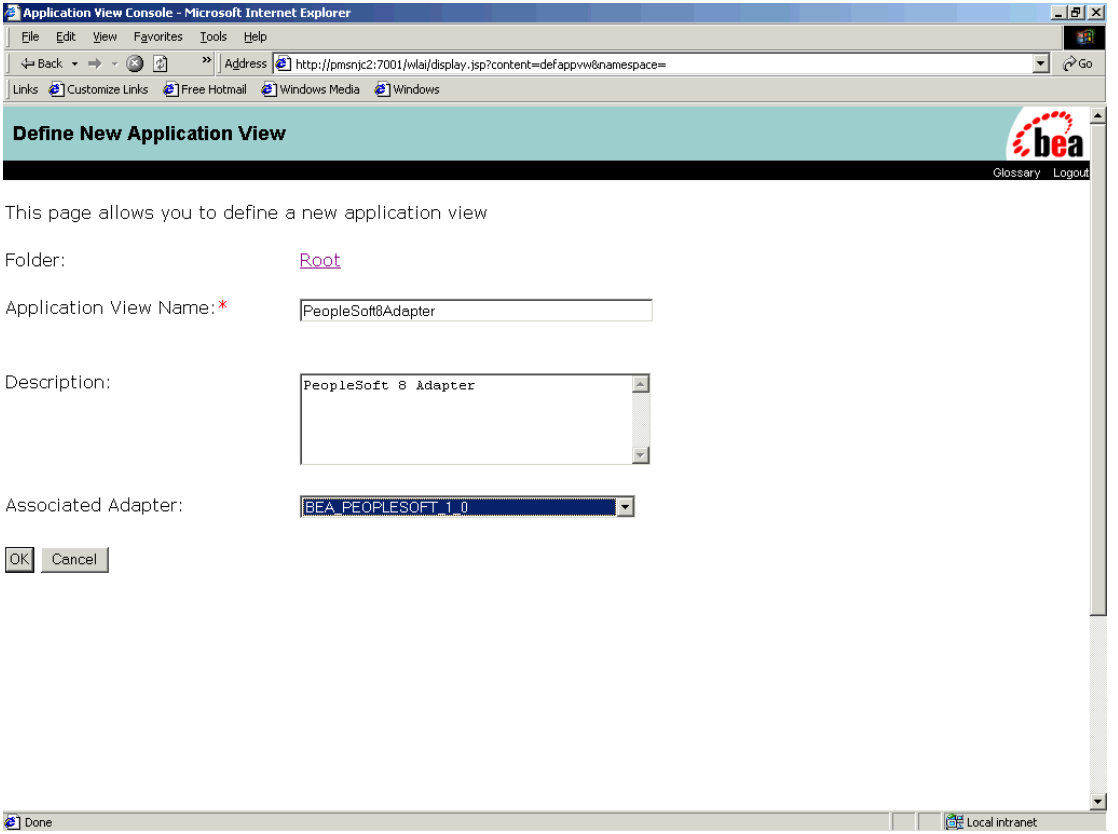
4. In the Define New Application View window, add the following information:

   a. In the Application View Name field, enter a name. The name should describe the set of functions performed by this application. Each application view name must be unique to its adapter. Valid characters include a-z, A-Z, 0-9, and _ (underscore).

   b. In the Description field, enter any relevant notes. Users view these notes when they utilize this application view with business process management workflows.

   c. From the Associated Adapter drop-down list, select the BEA_PEOPLESOFT_1_0 Adapter to use when creating this application view.

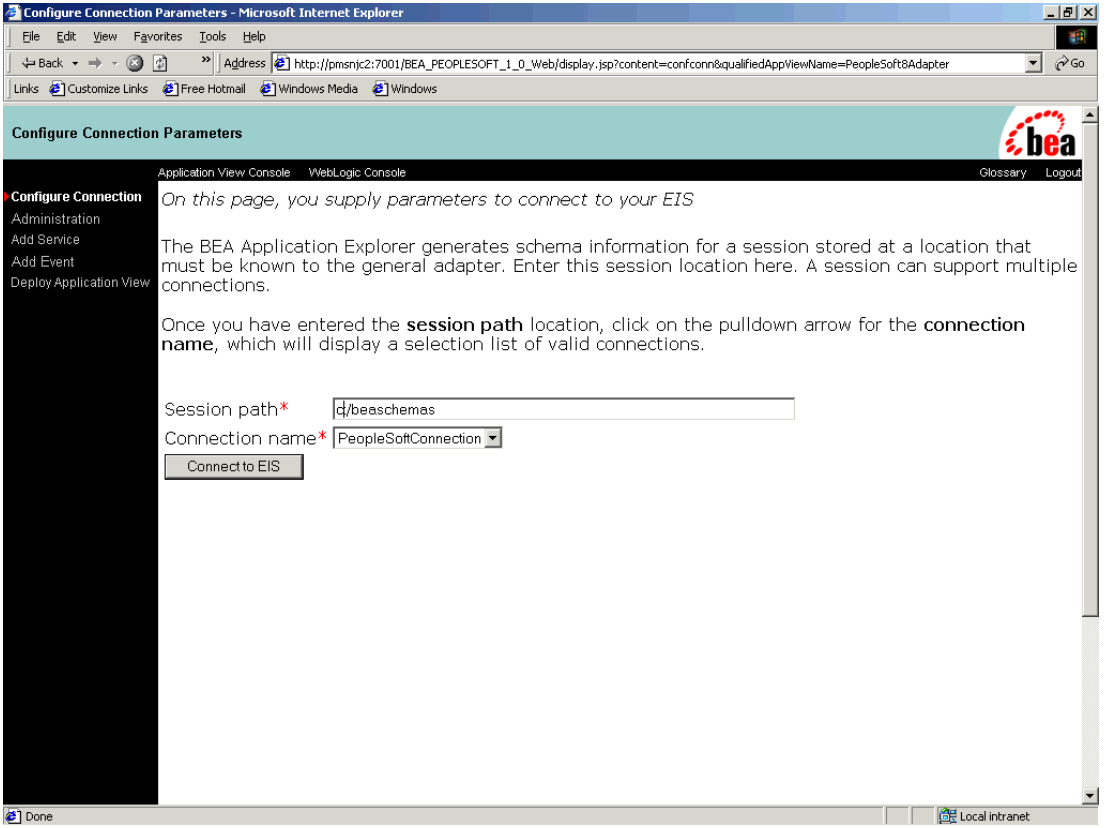
Figure 4-4 Define New Application View Window - With Information



5. Click OK.

The Configure Connection Parameters window opens.

Figure 4-5 Configure Connection Parameters



6. In the Configure Connection Parameters window, define the location of the schema definitions for the service request. This information is required for the application view to interact with the target EIS. You must enter this information only once per application view.
   - Session path is the location of the working directory established for the Application Explorer.
   - Connection name is the name of the connection used for creating schemas. The Application Explorer creates this folder for you.
7. Click Connect to EIS. The Application View Administration window opens.

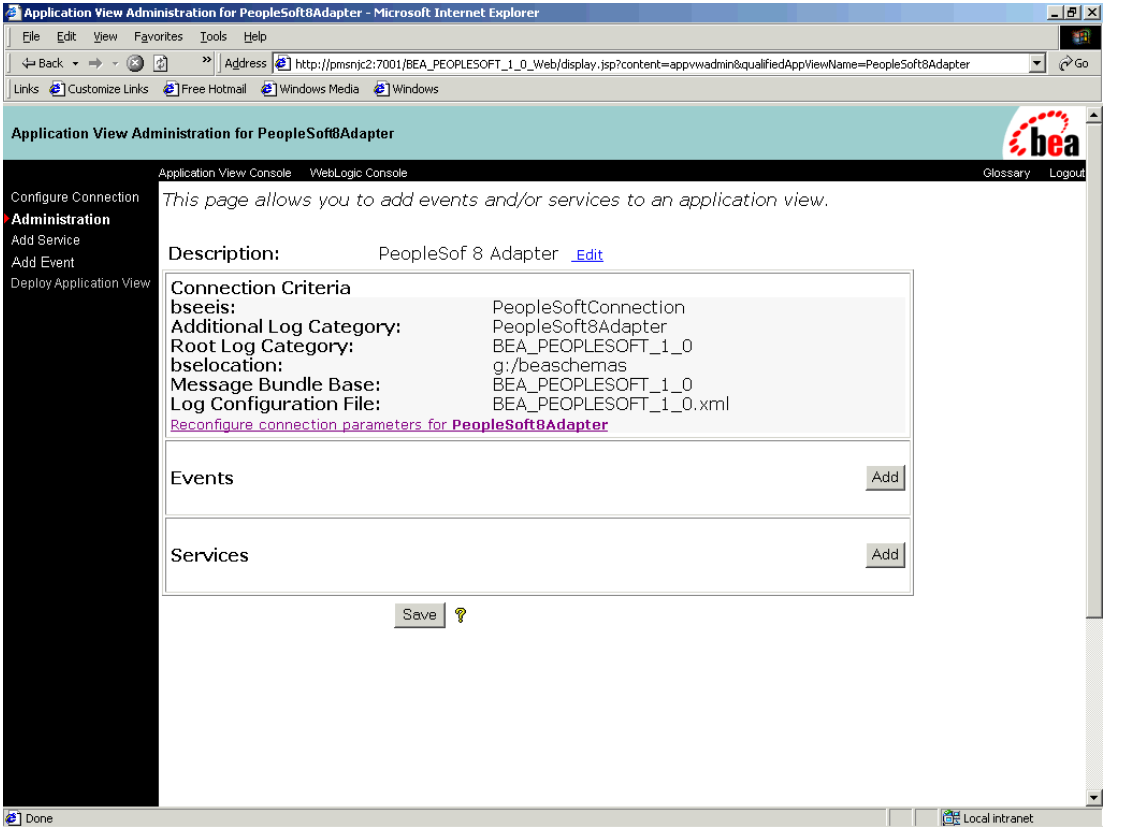
Adding a PeopleSoft Service to an Application View

The PeopleSoft service is the process by which a component is executed in the PeopleSoft system. If a business process management workflow was established with request and response variables, the appropriate variables are populated with PeopleSoft request and response XML documents.

After you create and configure an application view, add services that support the application's functions.

1. While the application view is open, click Administration. The Application View Administration window opens.

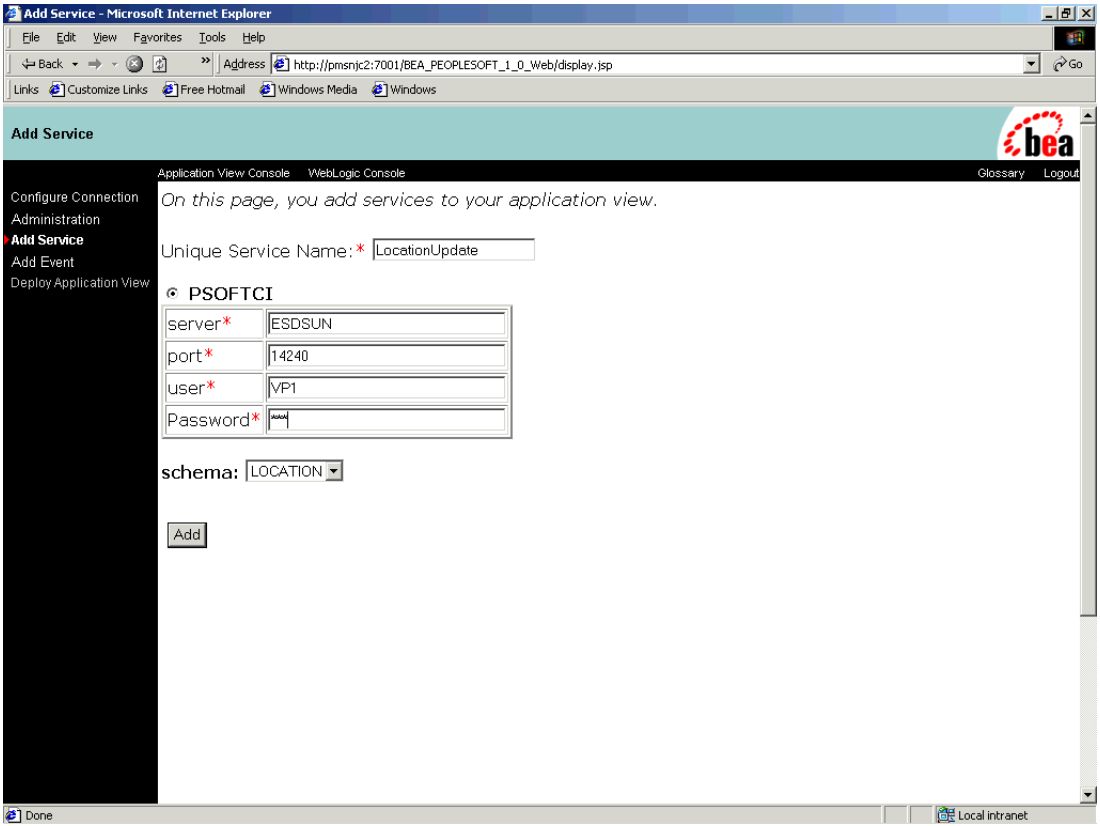
Figure 4-6 Application View Administration Window



2. Click Add Service.

The Add Service window opens.

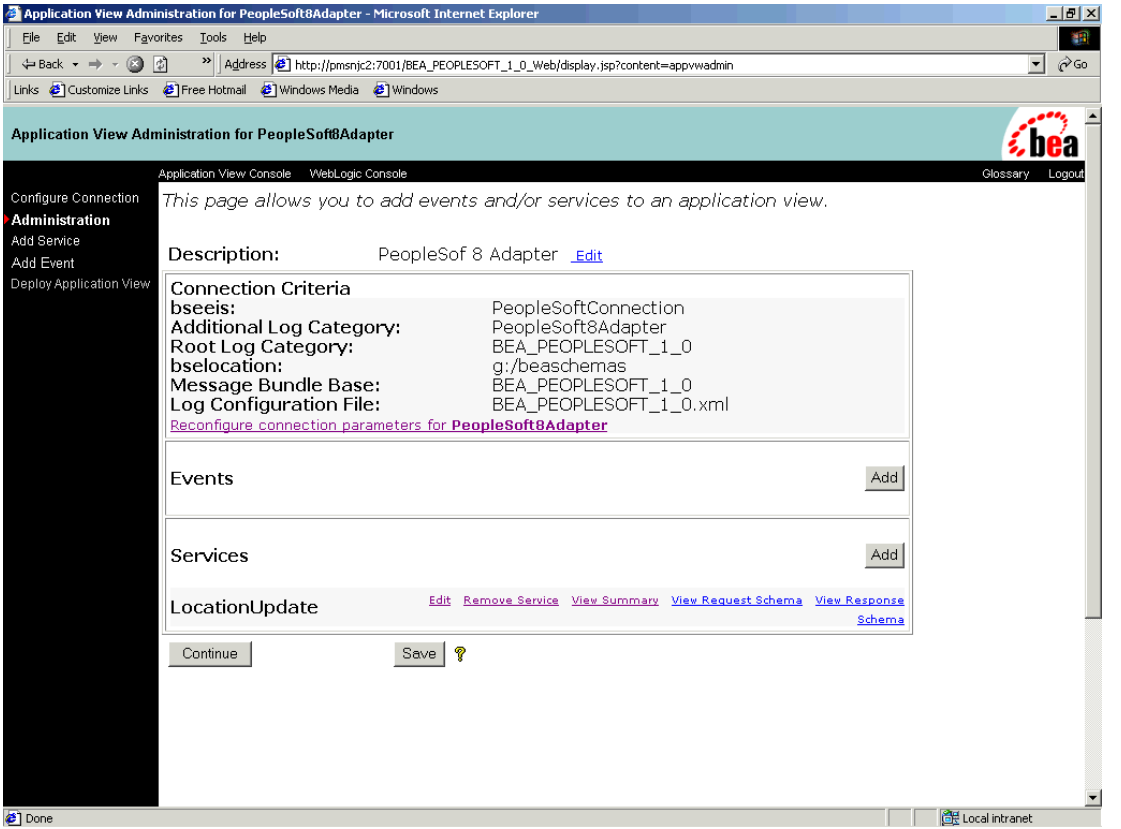
Figure 4-7 Add Service Window



3. In the Unique Service Name field, enter a name. The name should describe the function performed by this service. Each service name must be unique to its application view. Valid characters include a-z, A-Z, 0-9, and _ (underscore).
4. Provide the connection information to the PeopleSoft system. Consult your PeopleSoft administrator for the values to enter.
5. When you are finished, click Add.

The service LocationUpdate is added.

Figure 4-8 Application View Administration Window



6. Click Continue.

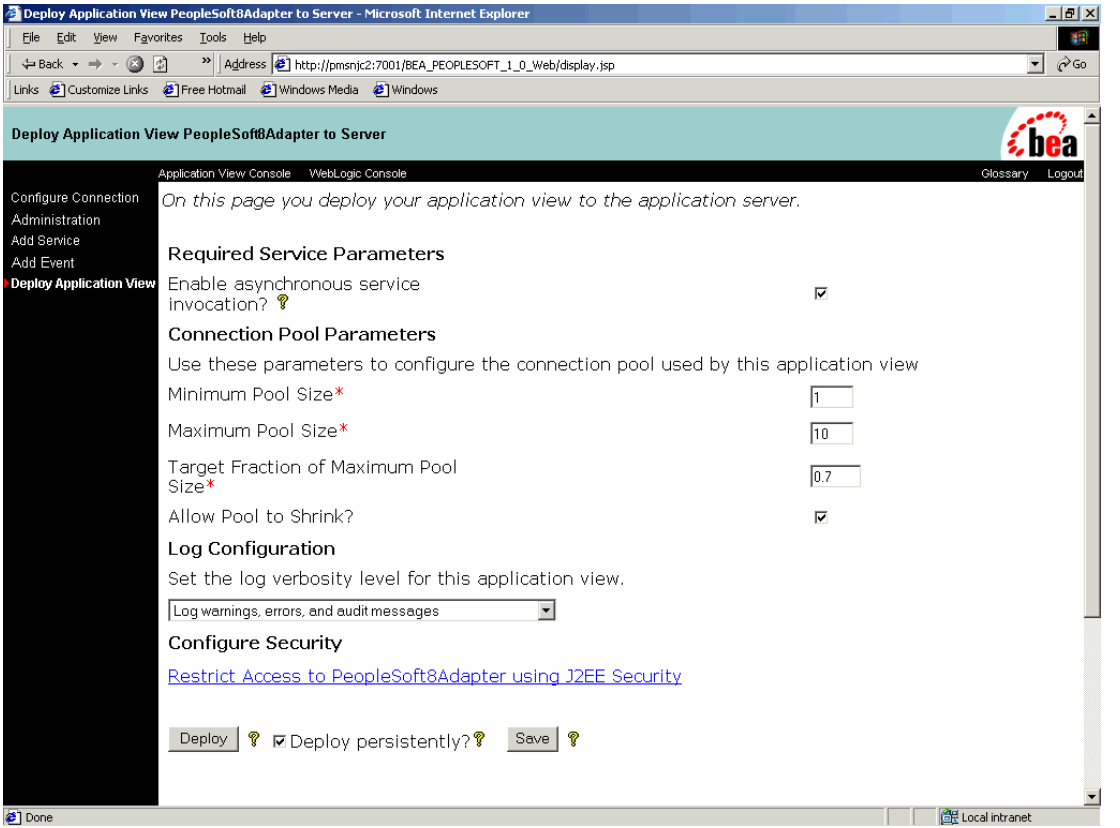
Deploying an Application View

You can deploy an application view when you have added at least one event or service to it. You must deploy an application view before you can test its services and events or use it in the WebLogic Server environment.

Application view deployment places relevant metadata about its services and events into a run-time metadata repository. Deployment makes the application view available to other WebLogic Server clients. This means business processes can interact with the application view, and you can test the application view's services and events.

Note: To enable business process management workflows or other authorized clients to asynchronously call the services (if any) of this application view, select Enable Asynchronous Service Invocation.

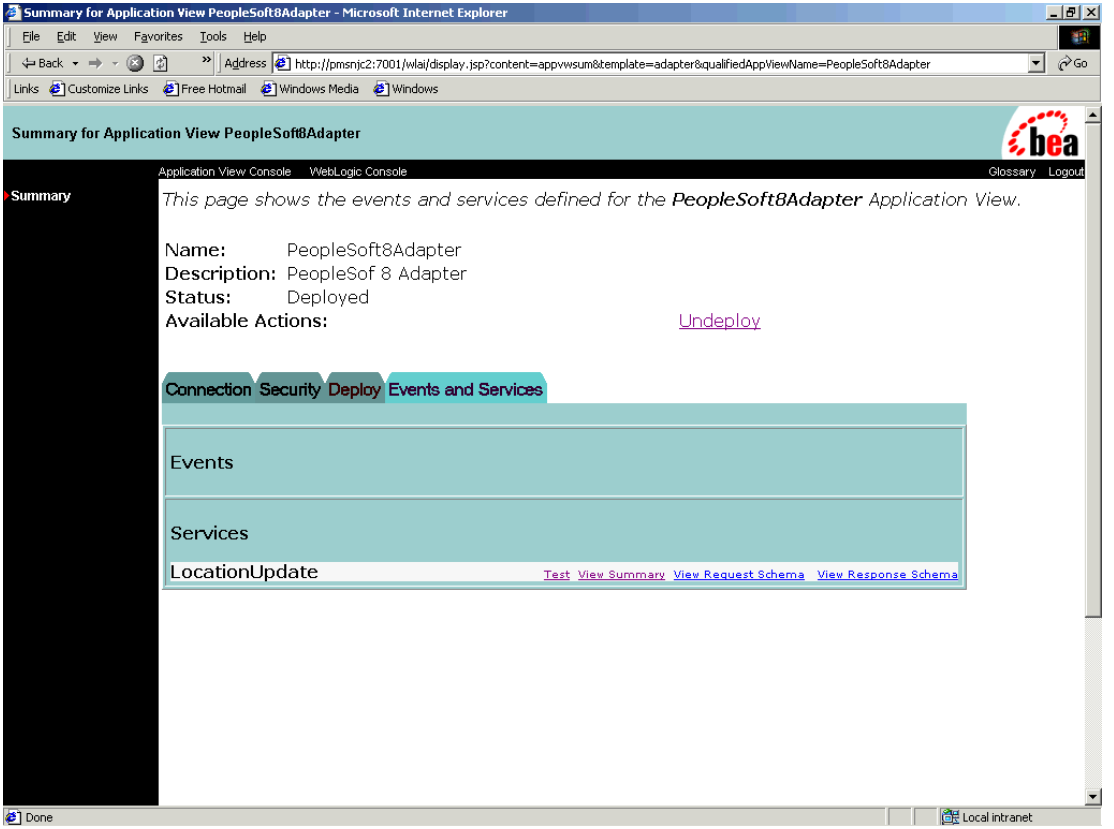
Figure 4-9 Deploy Application View Window



1. To deploy the application view, click Deploy. You may choose to click Save and deploy the application view at a later time.

The Summary for Application View window opens.

Figure 4-10 Summary for Application View Window



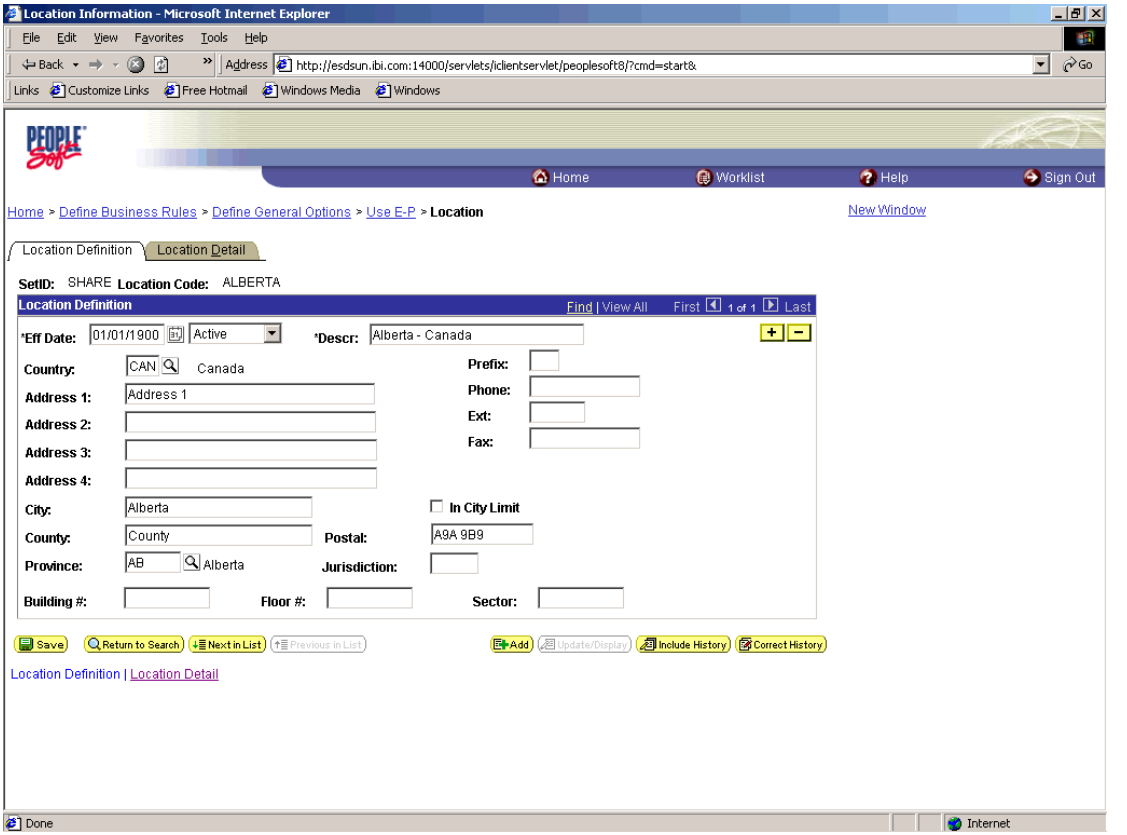
After you create and deploy an application view that contains services, test the application view services. Testing evaluates whether or not the application view service interacts properly with the target adapter.

2. To test application view services in the Current Services area, find the service and click Test.

3. Before executing the service to update the location table, verify the location in PeopleSoft to ensure that the update has taken place.

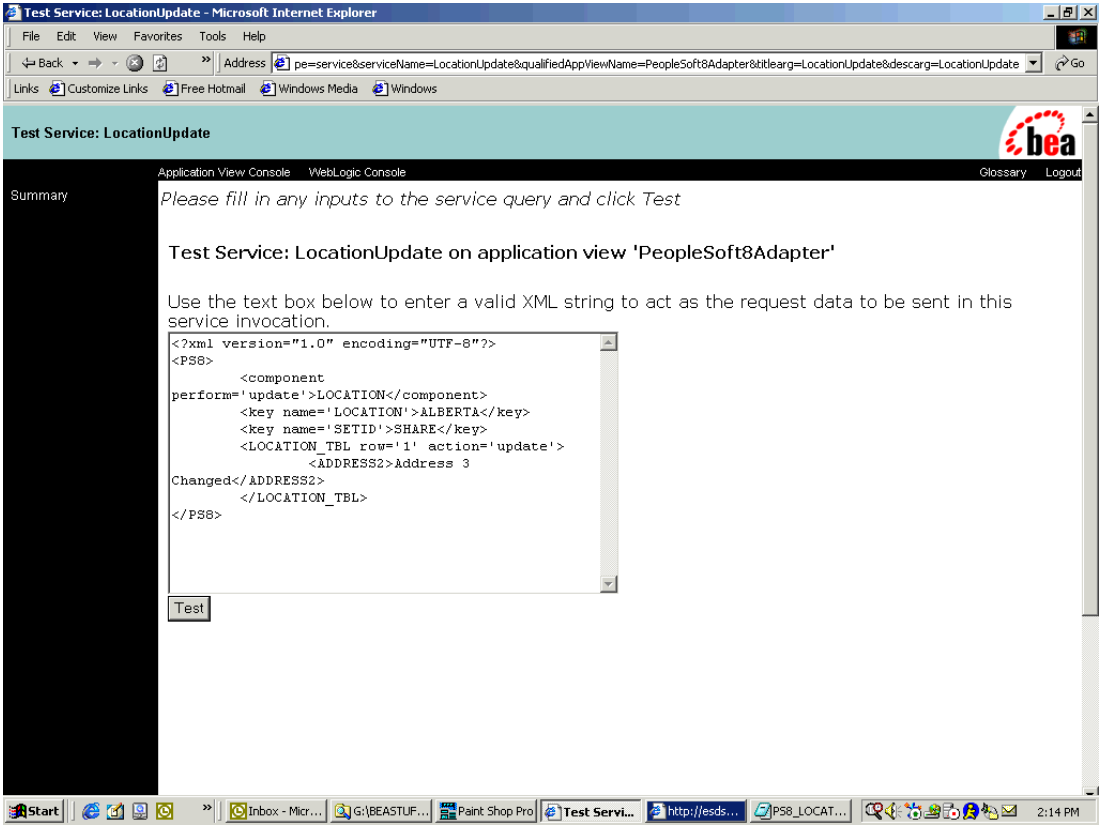
4. Using the PeopleSoft Web front end, check the values of the specific location. The location chosen for update is Alberta. Notice that Address 3 contains no information (blank).

Figure 4-11 Location Information Window



5. Enter the appropriate XML for the adapter.

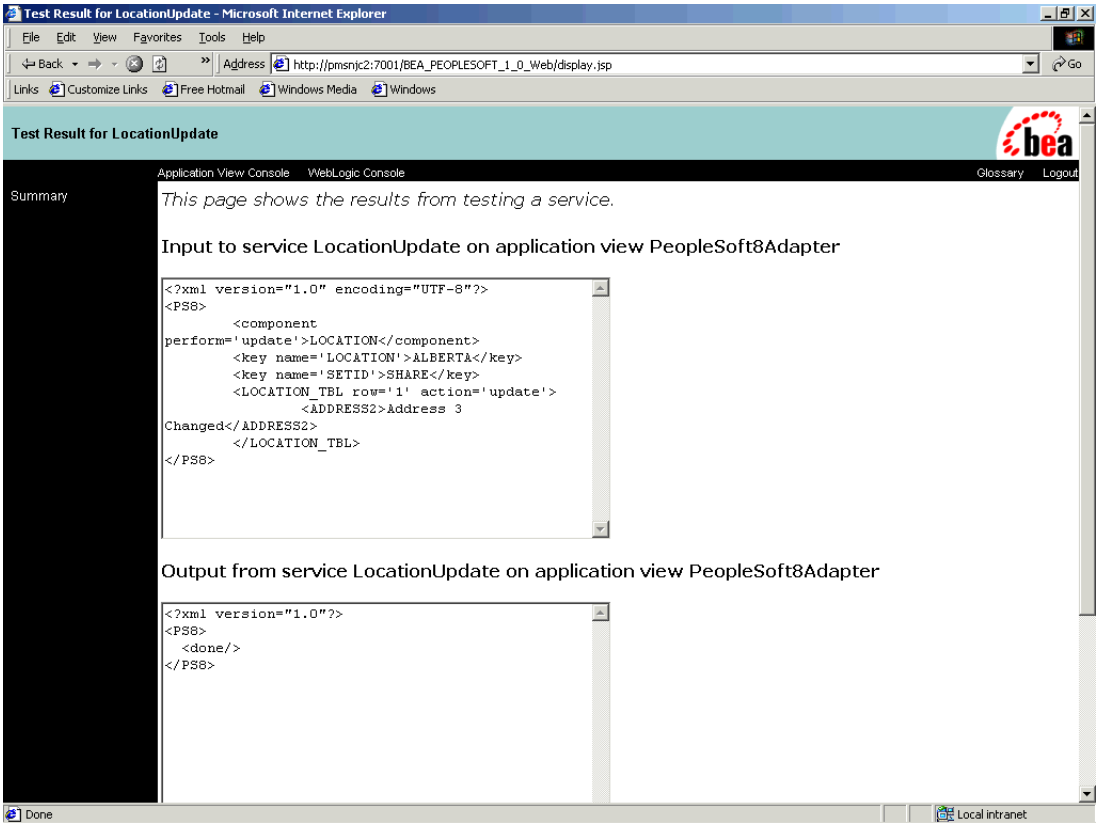
Figure 4-12 Test Service LocationUpdate Window



6. Click Test.

The Test Service window opens.

Figure 4-13 Test Results Summary Window

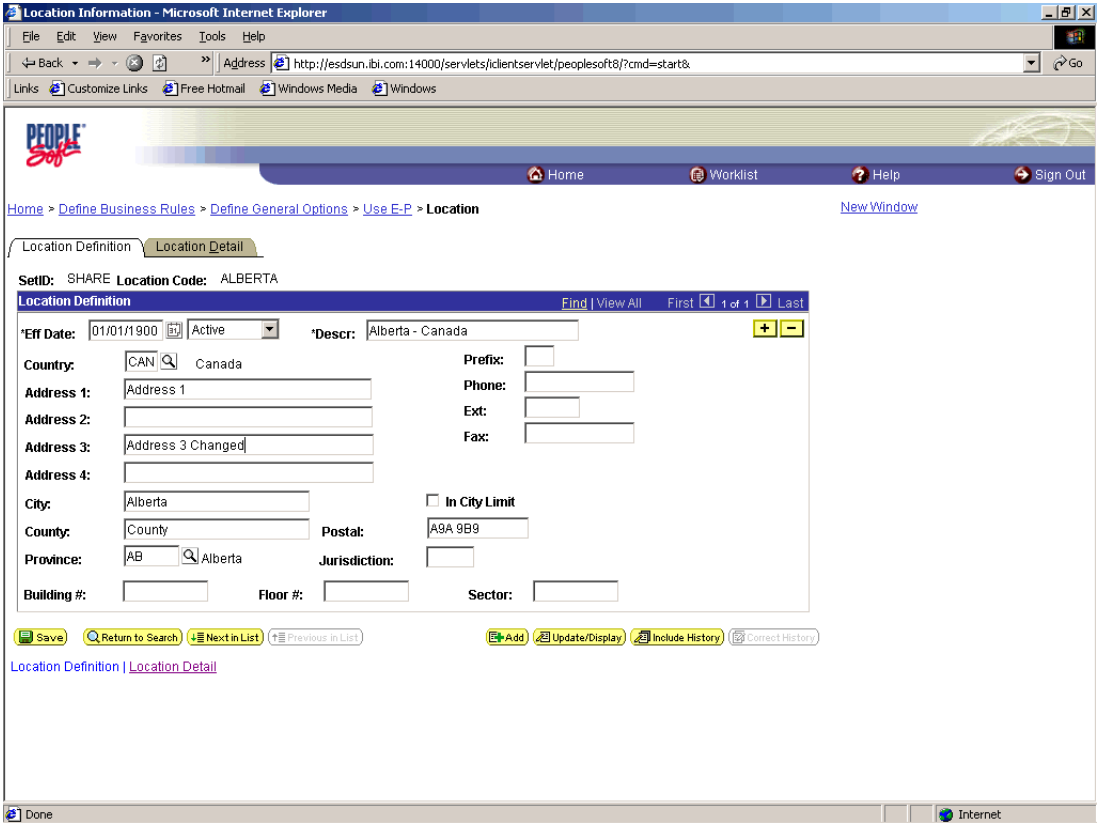


If the test fails, the Test Result window displays only a Timed Out message.

7. Verify in PeopleSoft that the update has taken place.

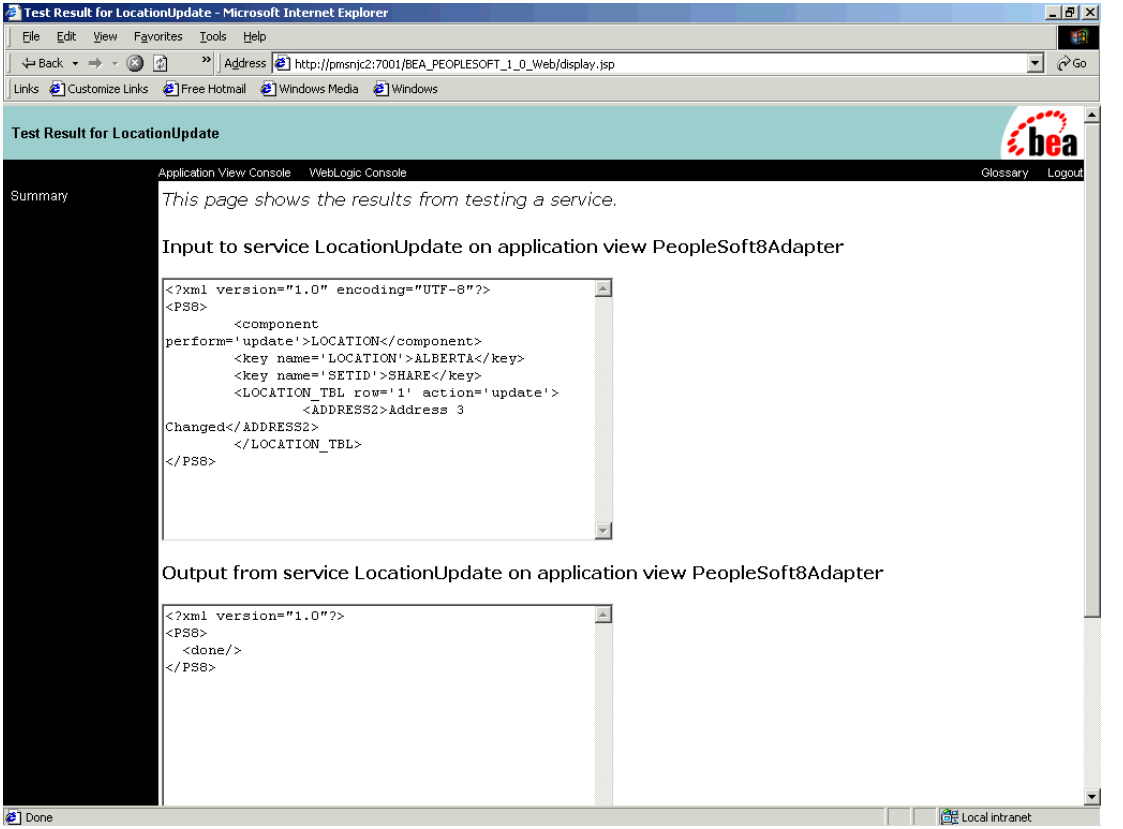
   The updated verification information appears. Notice the change of address.

Figure 4-14 Update Verification Window



The following test shows the execution of a component that returns multiple row sets.

Figure 4-15 Test Results Window

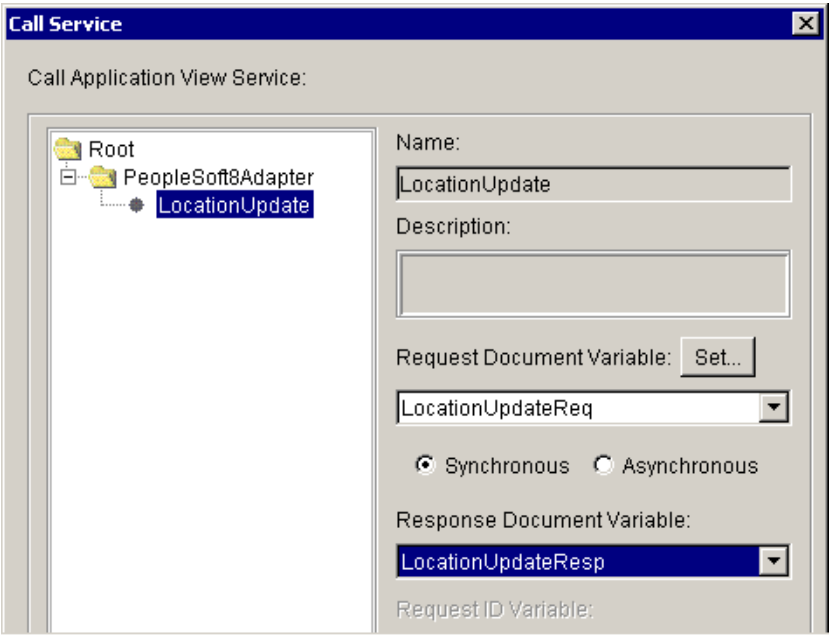


The Resource Adapter has now been successfully deployed and tested.

You can now write custom code to exploit the adapter or create a business process management workflow. For more information, see "Using Application Views in the Studio" in *Using Application Integration*:

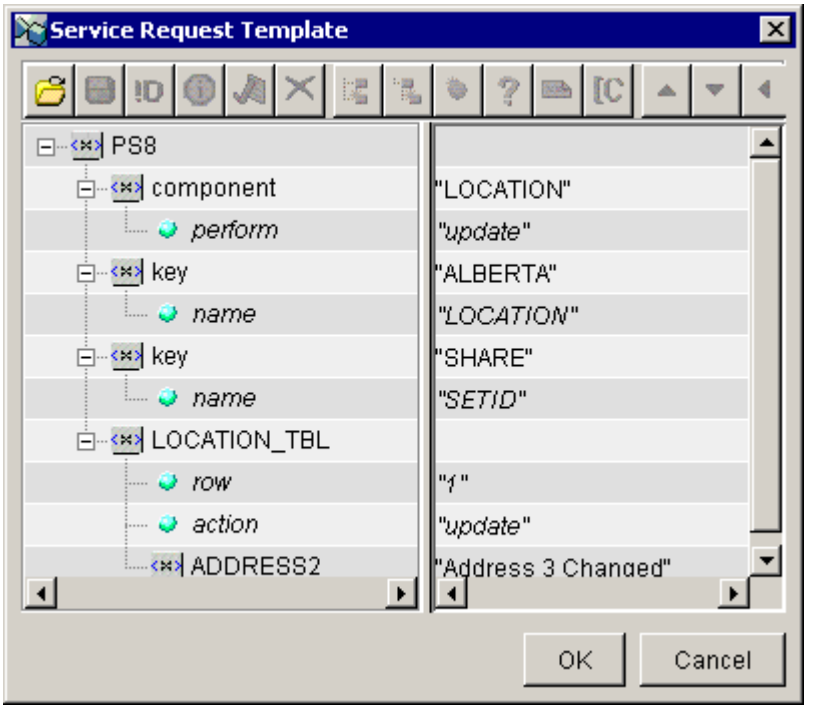
- For WebLogic Integration 7.0, see
  `http://edocs.bea.com/wli/docs70/aiuser/3usruse.htm`
- For WebLogic Integration 2.1, see
  `http://edocs.bea.com/wlintegration/v2_1sp/aiuser/3usruse.htm`

Figure 4-16 WebLogic Integration Studio Window



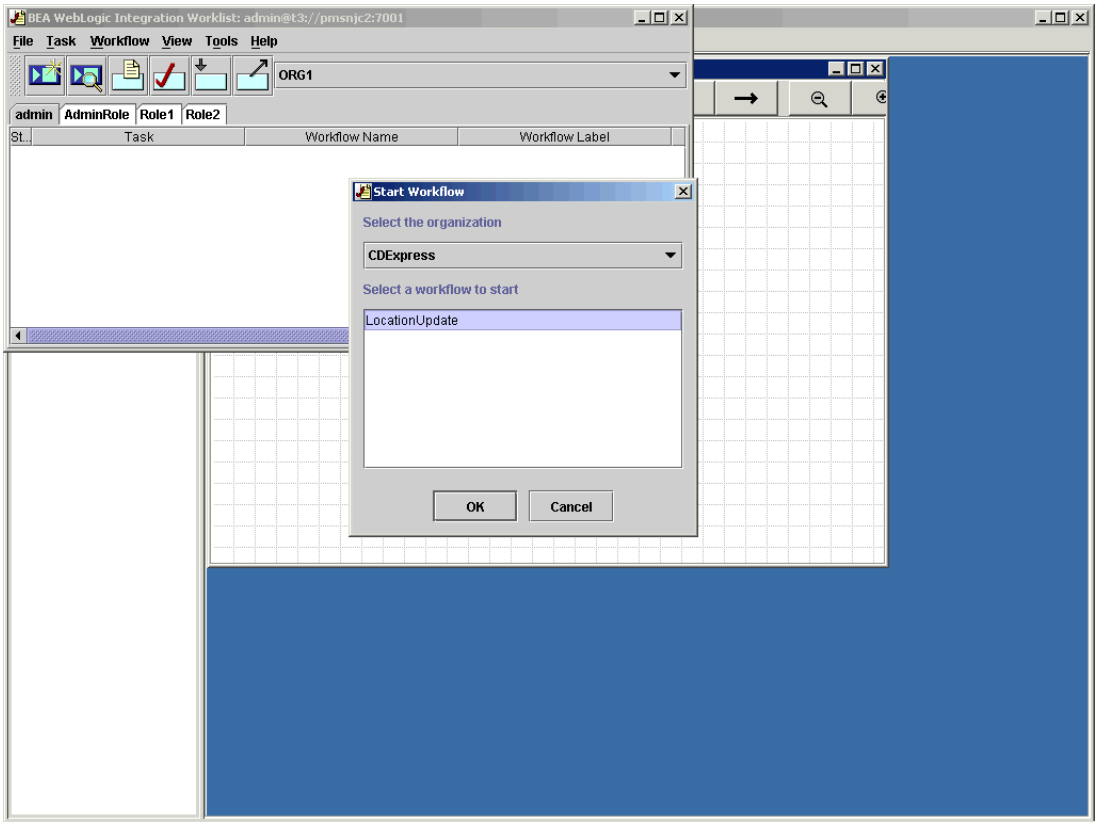
The PeopleSoft request appears as follows:

Figure 4-17 Service Request Template Dialog Box



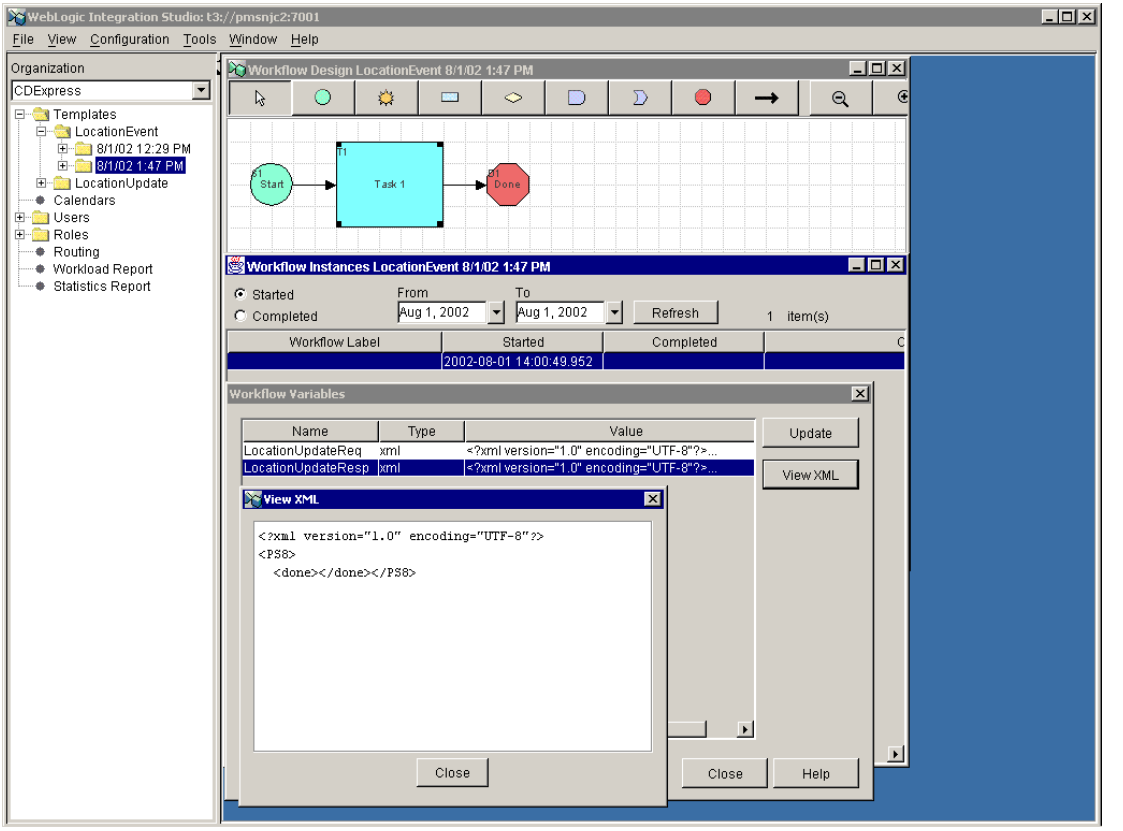
Start the workflow:

Figure 4-18 Start Workflow Dialog Box



The PeopleSoft response appears as follows:

Figure 4-19 WebLogic Integration Studio Response

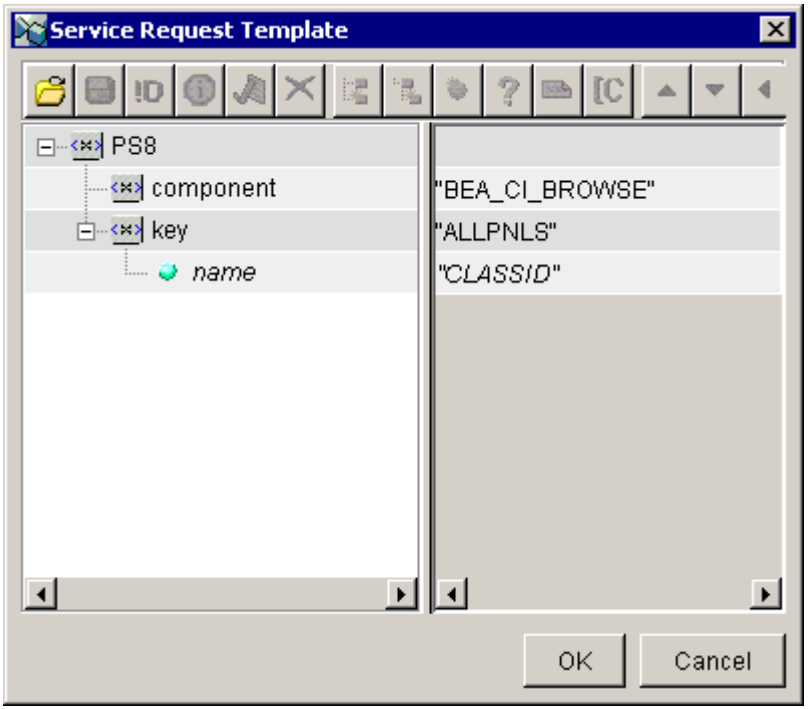


Component Interface Metadata

The example below shows the execution of a supplied component called BEA_CI_BROWSE, which displays the list of Component Interfaces available for a particular security class. This component is used by the BEA Application Explorer to create the schemas that are automatically published to the WebLogic Application Integration repository.

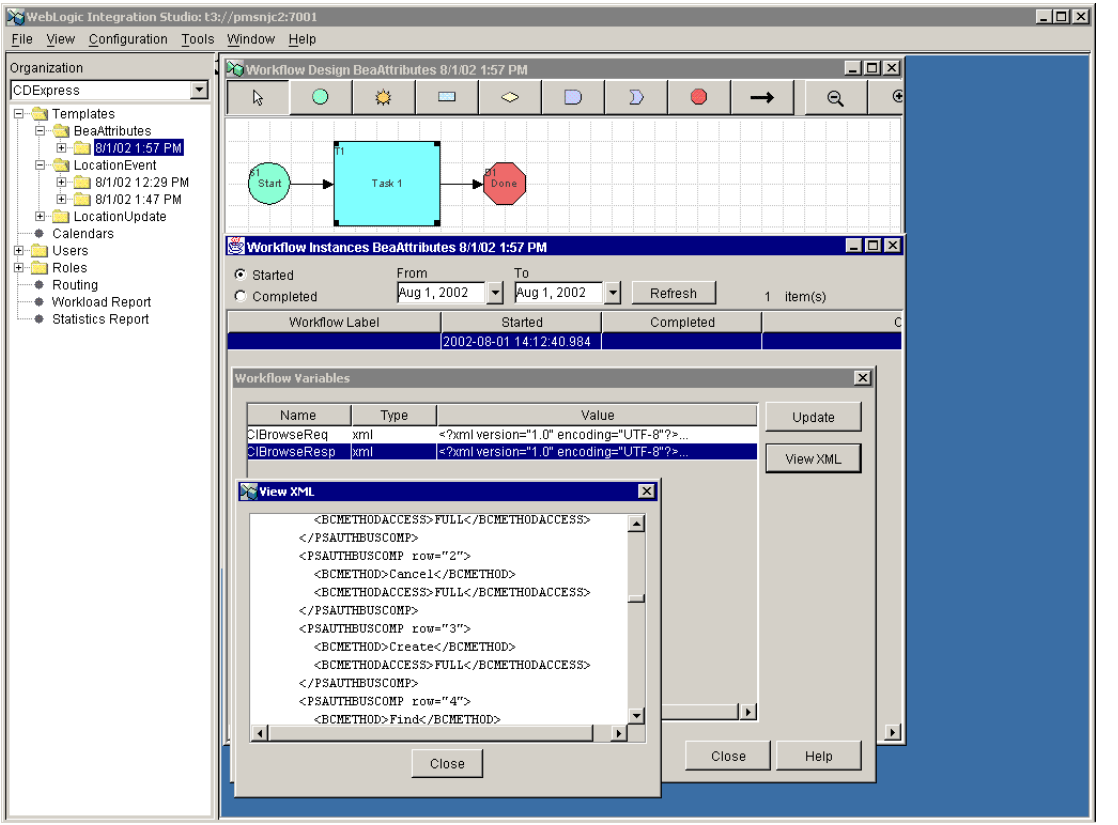
The following workflow response contains the list of all available components. Note that the actual execution of this component and the functional use of its output are used by BEA Application Explorer.

Figure 4-20 Service Request Template Dialog Box



The response contains information on all available components.

Figure 4-21 Service Request Component Response



5 Using PeopleSoft 8 Application Messaging

This section discusses how to use and create PeopleSoft message channels that enable you to pass PeopleSoft XML to the WebLogic Server. It includes the following topics:

When using PeopleSoft XML to integrate with PeopleSoft, the interface is facilitated by PeopleSoft Application Messaging. The BEA WebLogic Adapter for PeopleSoft 8 uses a delivered handler that is configured within the PeopleSoft application gateway using TCP/IP transport services.

The following sections discuss the use and creation of PeopleSoft message channels that enable the passing of PeopleSoft XML to the WebLogic Server. To configure WebLogic Server for the handler, see the *BEA WebLogic Adapter for PeopleSoft 8 Installation and Configuration Guide*.

PeopleSoft Application Messaging Overview

The BEA WebLogic Adapter for PeopleSoft 8 supports the integration of PeopleSoft applications and third party systems by publishing business events as XML-formatted messages.

To subscribe to data, third party applications can accept and process XML messages posted by PeopleSoft by adding to the already available PeopleSoft handlers, a custom Java subscription handler (or plug-in) to the PeopleSoft Application Messaging Gateway Servlet.

PeopleSoft Handlers

PeopleSoft handlers integrate PeopleSoft XML with the outside world. PeopleSoft provides a set of common interfaces that write PeopleSoft XML as a result of a business event, such as an addition to a table or a modification to some piece of data. Two common PeopleSoft handlers are the SimpleFileHandler and the MQSeriesHandler. Depending on the handler specified in the PeopleSoft Message Channel definition, PeopleSoft posts an XML document to either a file or to an MQSeries Queue.

WebLogic Integration can handle events associated with files and MQSeries queues. A standard format adapter for files or MQSeries can be used as the event adapter.

TCP/IP communications from PeopleSoft to WebLogic Integration involves the installation and configuration of the BEA PeopleSoft TCP/IP handler. The BEA WebLogic Adapter for PeopleSoft 8 supports this type of event adapter.

The BEA TCP/IP Handler

The BEA WebLogic Adapter for PeopleSoft 8 enables the posting of PeopleSoft XML directly to the WebLogic Integration environment by using a special handler. This handler is a plug-in class file that is installed within the PeopleSoft Gateway Web application server. This handler is the BEA PeopleSoft TCP/IP handler. The WebLogic Integration can receive PeopleSoft event XML from the Application Messaging Gateway servlet through TCP.

The next topic discusses how to create an application messaging process which is mandatory for event processing in PeopleSoft.

The installation and configuration of the handler software is described in the *BEA WebLogic Adapter for PeopleSoft 8 Installation and Configuration Guide*.

Configuring PeopleSoft for Application Messaging

The PeopleSoft XML output is triggered within PeopleSoft as a result of a business event. The PeopleSoft environment must be configured correctly for application messaging. This internal PeopleSoft configuration should be attempted only by a person with knowledge of PeopleSoft. The installation of the software and the BEA TCP/IP handler is described in the *BEA WebLogic Adapter for PeopleSoft 8 Installation and Configuration Guide*.

The WebLogic Integration environment can receive a PeopleSoft XML document as an event using standard PeopleSoft handlers such as file or MQSeries, or it can use the specialized BEA TCP/IP handler.

In all cases, you must:

- Establish a Message Node to publish to the appropriate handler.
- Select or create a Message Channel with routing rules pointing to the handler.
- Select or create a Message Definition that defines the business event (such as Add or Update transaction).

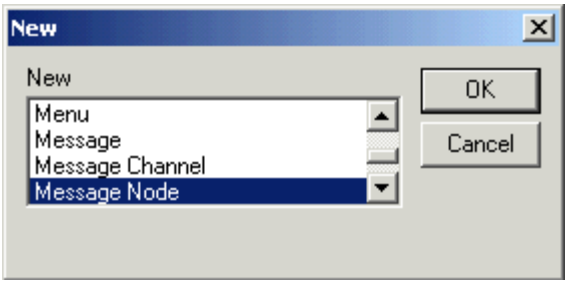
For information about installing the BEA TCP/IP Handler and how to configure the handler on the PeopleSoft Gateway Server, see the *BEA WebLogic Adapter for PeopleSoft 8 Installation and Configuration Guide.*

How to Set up a Message Node to Publish to the BEA TCP/IP Handler for PeopleSoft 8.1

This section illustrates an example of how to publish a message from the LOCATION_TBL.

1. Create a project (recommended).
2. To set up a message node, choose File→New.
3. When the New dialog box appears, select MessageNode.

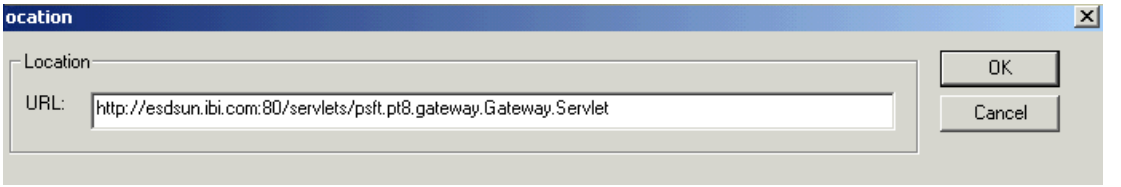
Figure 5-1 New Dialog Box



4. Click OK.

   The Location dialog box appears.

Figure 5-2 Location Dialog Box

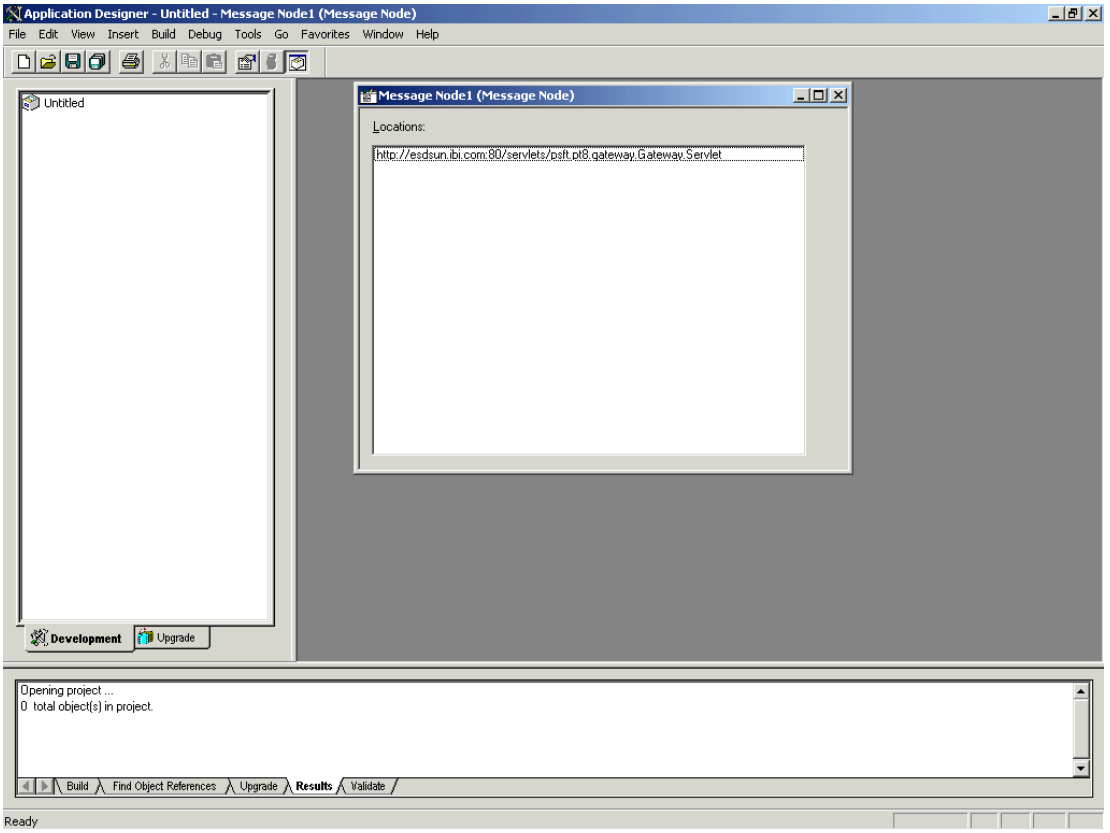


   a. Enter the URL of the PeopleSoft Application Gateway (handler directory).

   b. Click OK.

   The Message Node Window opens.

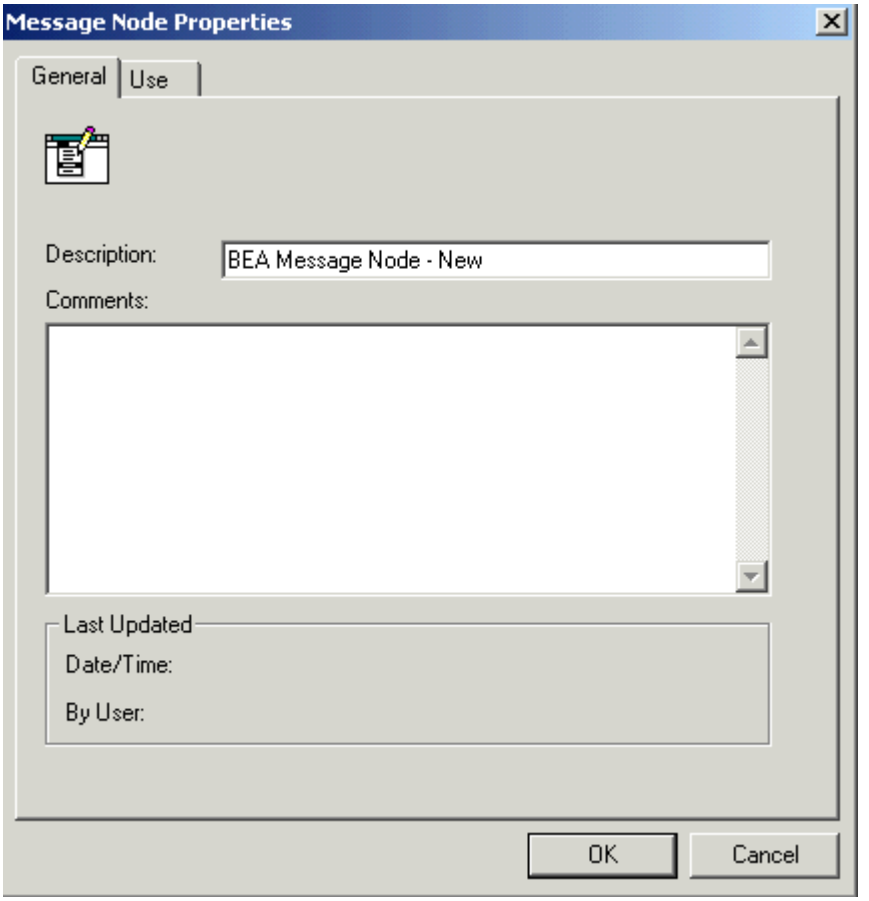
Figure 5-3 Application Designer - Message Node Window



5. Choose File→Object Properties.

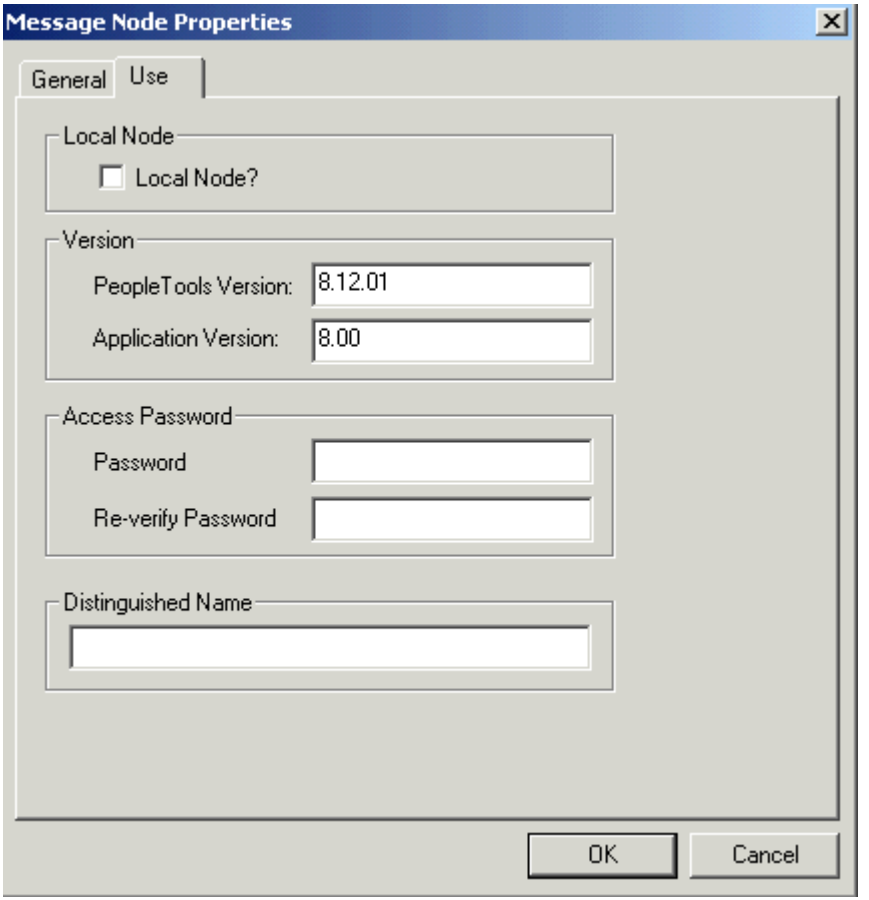
The Message Node Properties dialog box opens.

Figure 5-4 Message Node Properties Dialog Box



6. Click the Use tab.

Figure 5-5 Message Node Properties Dialog Box - Use Tab



a. Enter the PeopleTools Version, for example, 8.12.01.

b. Click OK.

7. Save the message node and provide a name, such as BEA_NODE.
8. Insert the message node into your project (recommended).
9. Establish a Message Channel by selecting an existing message channel provided by PeopleSoft or creating a new one. See "Select or Create a Message Channel" on page 5-9.

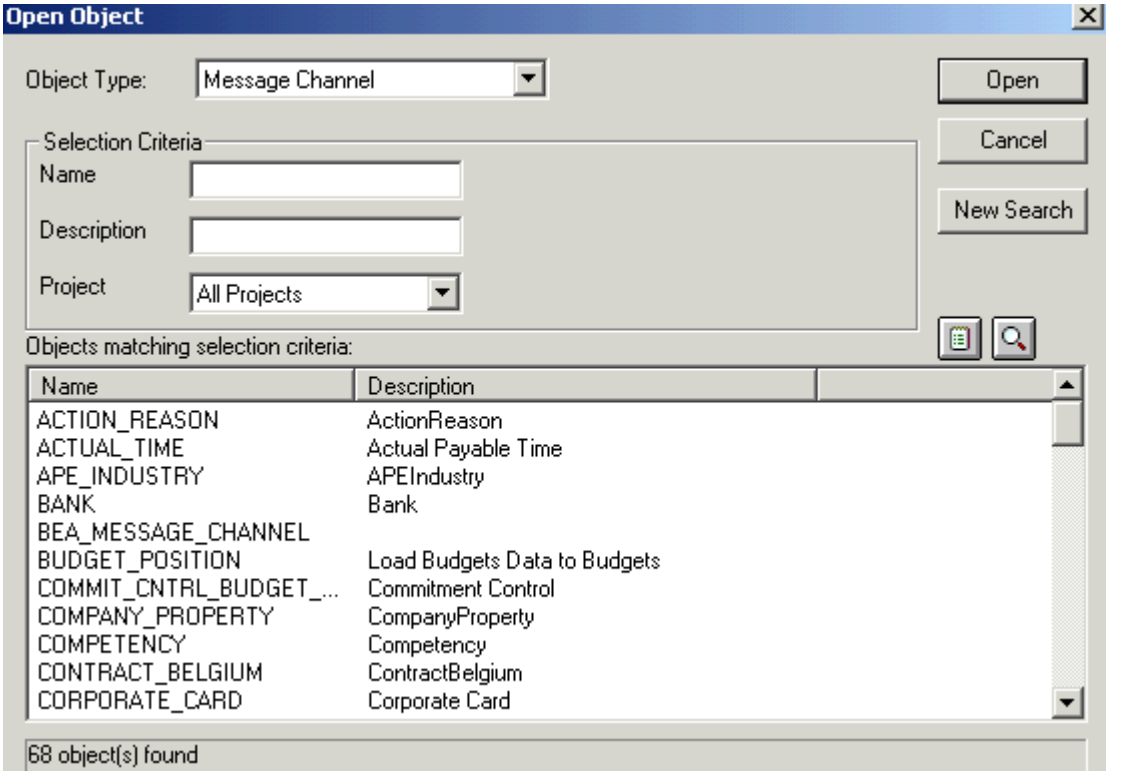
Select or Create a Message Channel

For this example, select the ENTERPRISE_SETUP message channel. To select a message channel:

1. Choose File→Open.

   The Open Object dialog box appears.

Figure 5-6 Open Object Dialog Box



   a. Select Message Channel in the Object Type drop-down list.

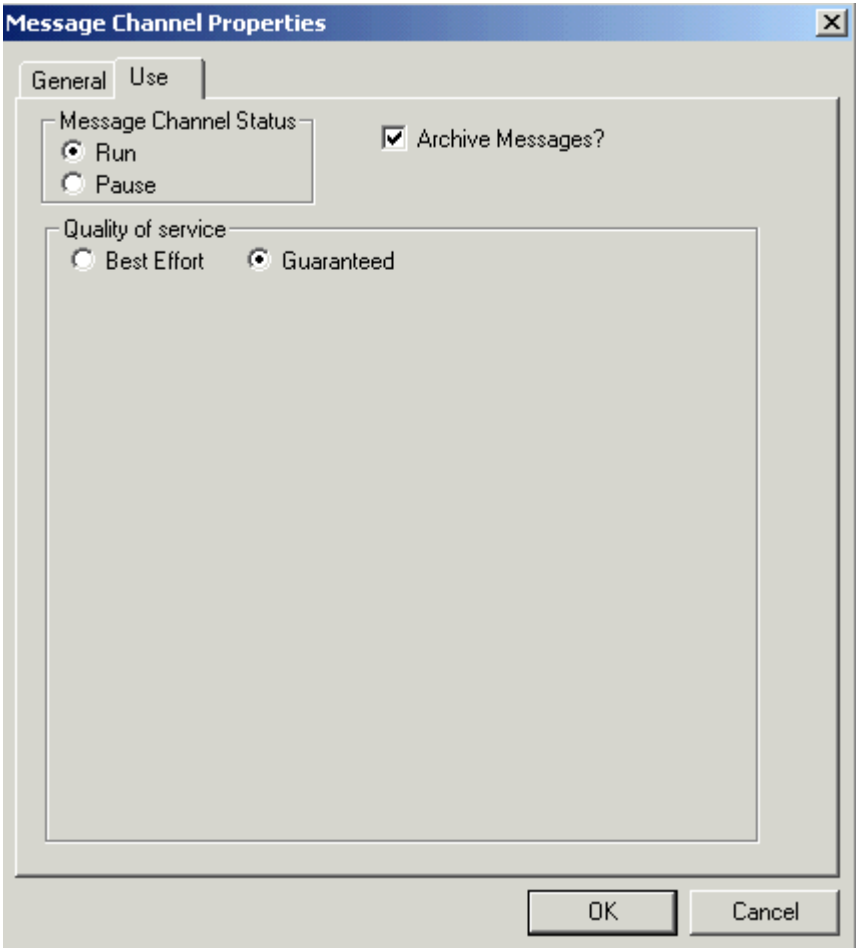
   b. Select ENTERPRISE_SETUP from the list of objects in the lower pane.

   c. Click Open.

The message channel, ENTERPRISE_SETUP, opens with the Messages tab active (not illustrated).

2. Open the Message Channel Properties dialog box.

Figure 5-7 Message Channel Properties Dialog Box



Note: Message archiving and quality of service are site dependent.

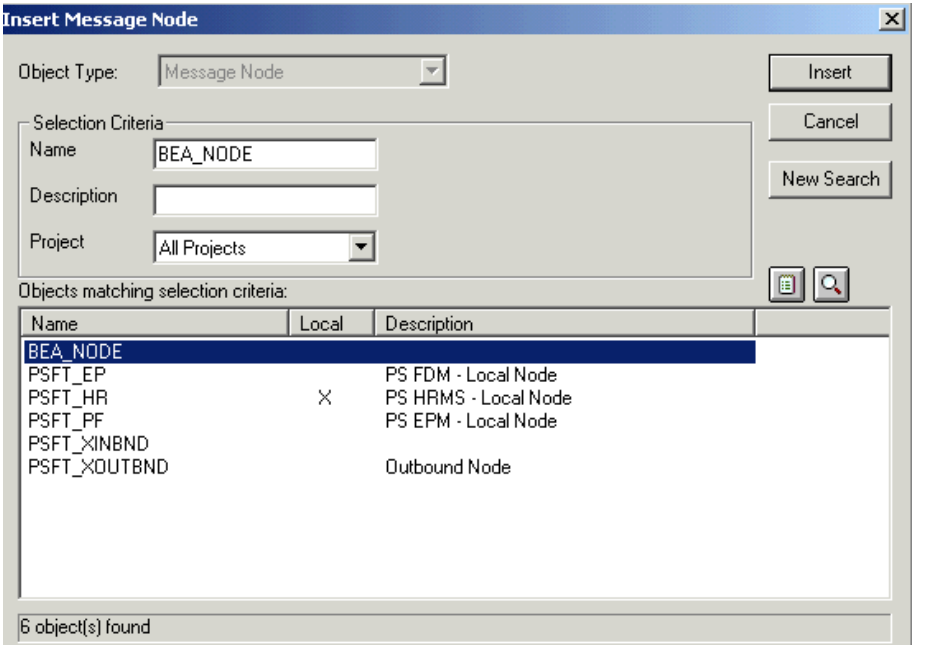
a. Click the Use tab.

b. Ensure that the Message Channel Status is set to Run and click OK.

3. Select the Routing Rules tab (not illustrated).
4. Select Insert Message Node.

   The Insert Message Node dialog box appears.

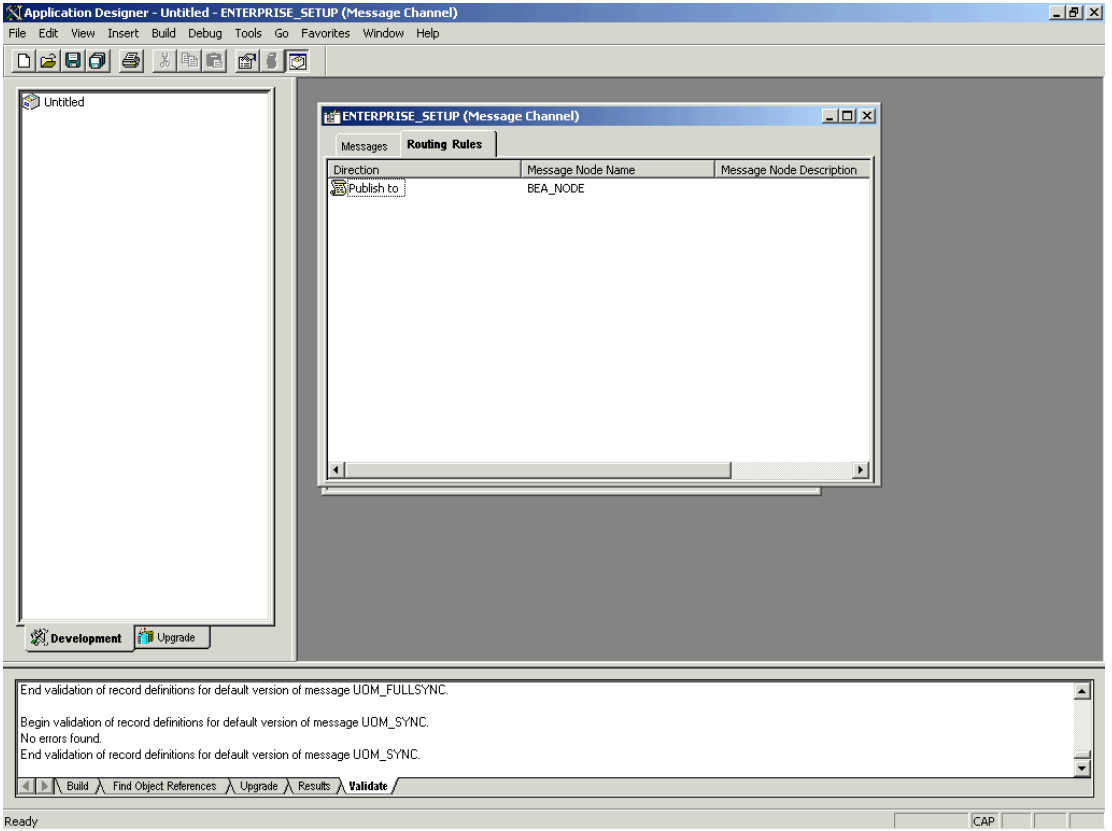
Figure 5-8 Insert Message Node Dialog Box



   a. Select your previously created message node, BEA_NODE.
   b. Click Insert.

The ENTERPRISE_SETUP window opens.

Figure 5-9 Application Designer - ENTERPRISE_SETUP Window



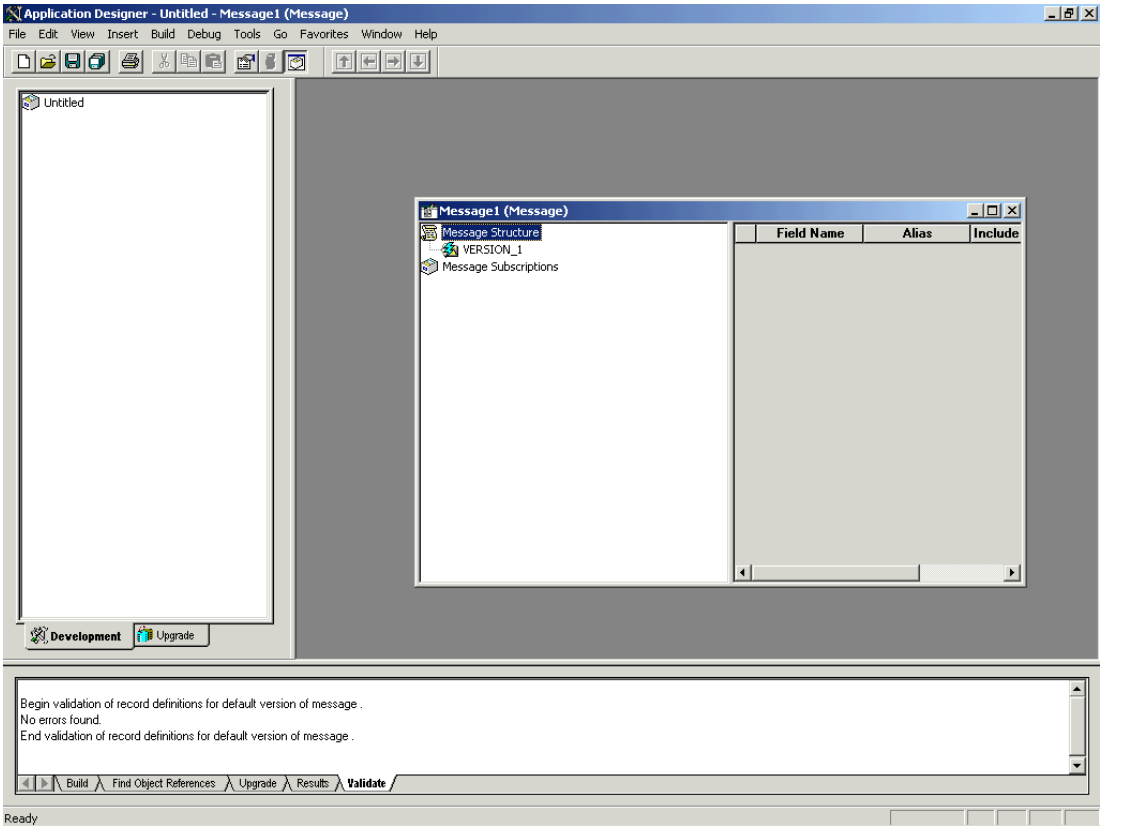
5. Right-click BEA_NODE.
6. Select Routing Direction→Publish To.
7. Save the message channel.
8. Insert the message channel into the project (recommended).

How to Create a Message

To create a new message, perform the following steps in Application Designer:

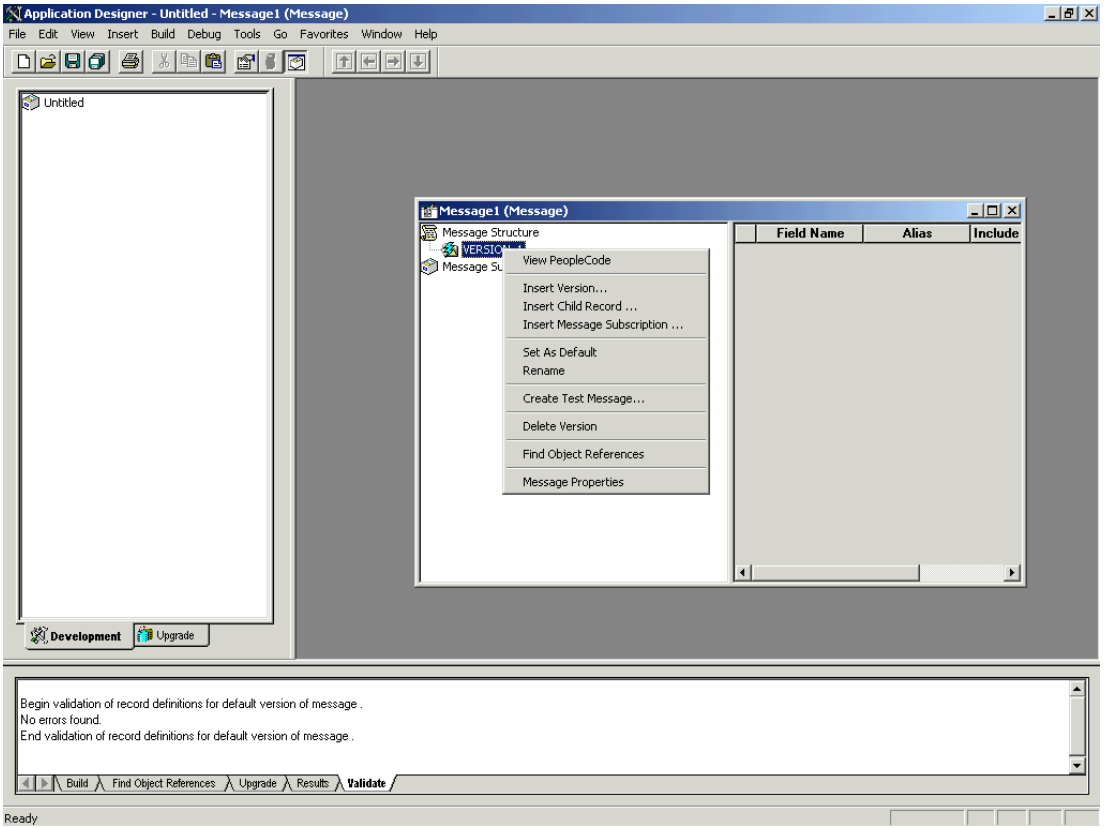
1. Choose File→New→Message.

Figure 5-10 Application Designer - Message Window



2. In the Message window, right-click the version number.

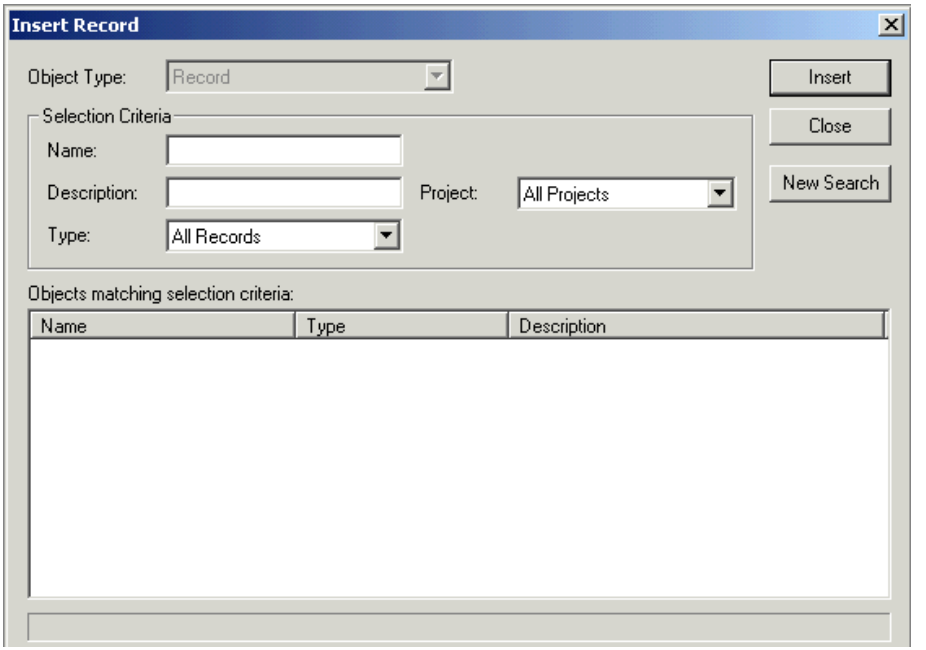
Figure 5-11 Application Designer - Message Window - Shortcut Menu



3. Select Insert Child Record from the shortcut menu.

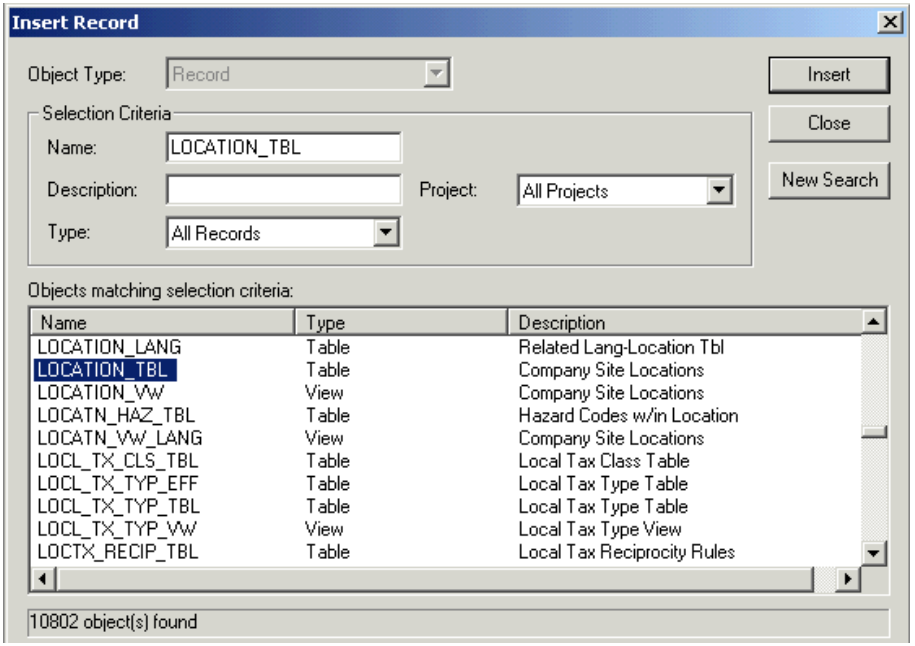
   The Insert Record dialog box opens.

Figure 5-12 Insert Record Dialog Box



4. Click Insert to obtain a list of records or type the name of the record required.

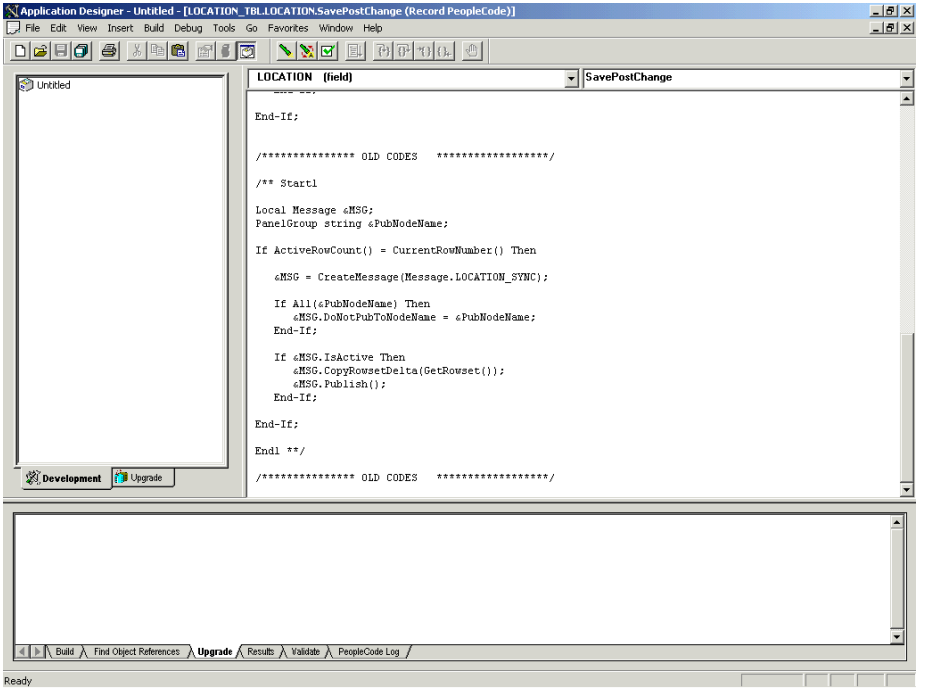
Figure 5-13 Insert Record Dialog Box - List of Records



Use the Location Table listed here as the basis for the message.

The SavePostChange PeopleCode of the LOCATION_TBL record displays the following code in the right pane. This is what triggers the Message being published. You must add similar code to any tables that must have data published online.

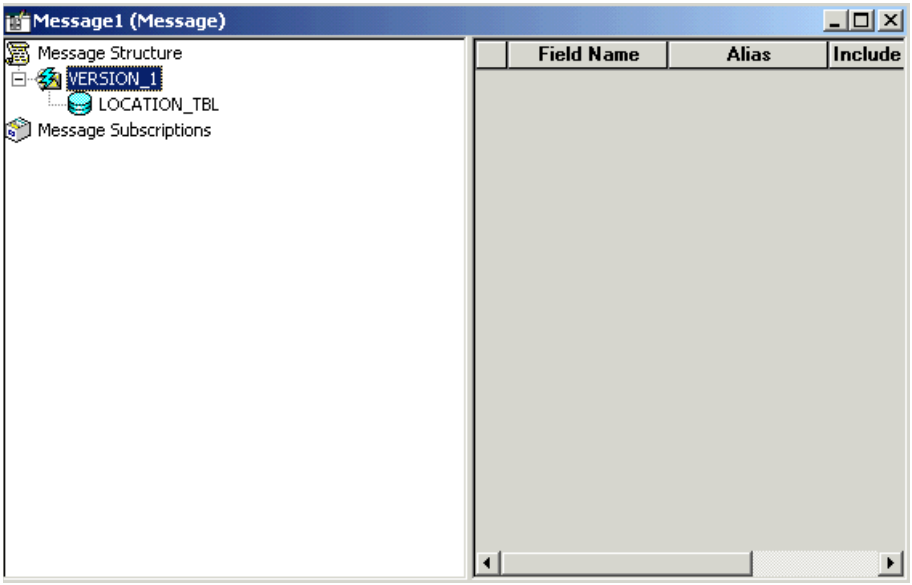
Figure 5-14 Application Designer - PeopleCode Window



5. Click Insert in the Insert Record dialog box.

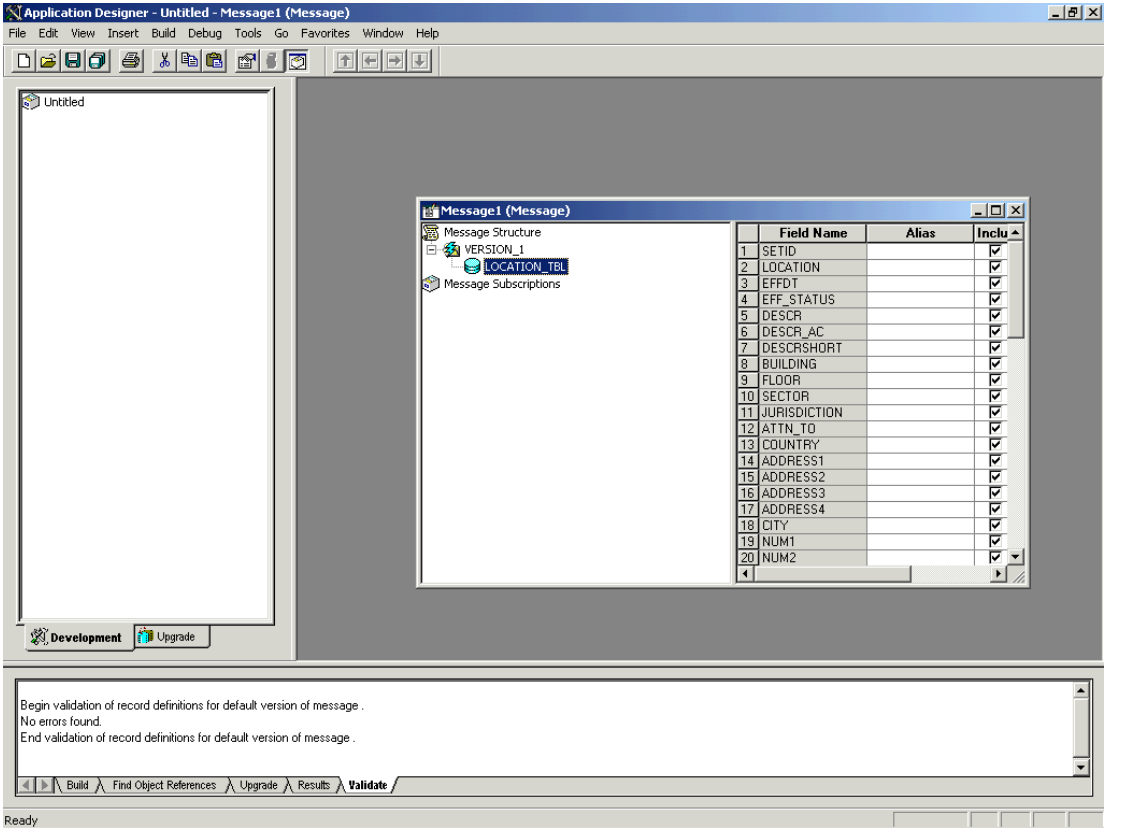
   The Message window opens.

Figure 5-15 Message Window



6. Double-click the table name to display the table fields.

Figure 5-16 Application Designer - Table Fields

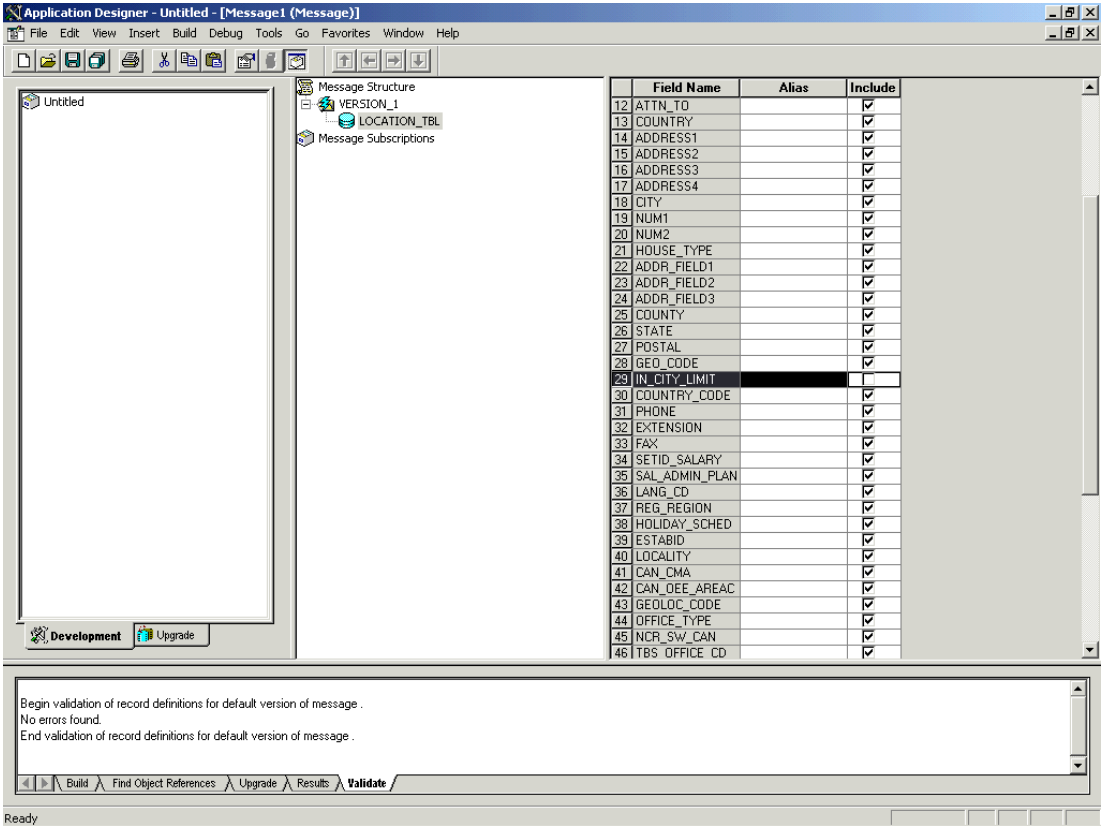


You may not require every field for the message.

7. Click the appropriate check box in the Include column to de-select the check box and exclude any fields not required.

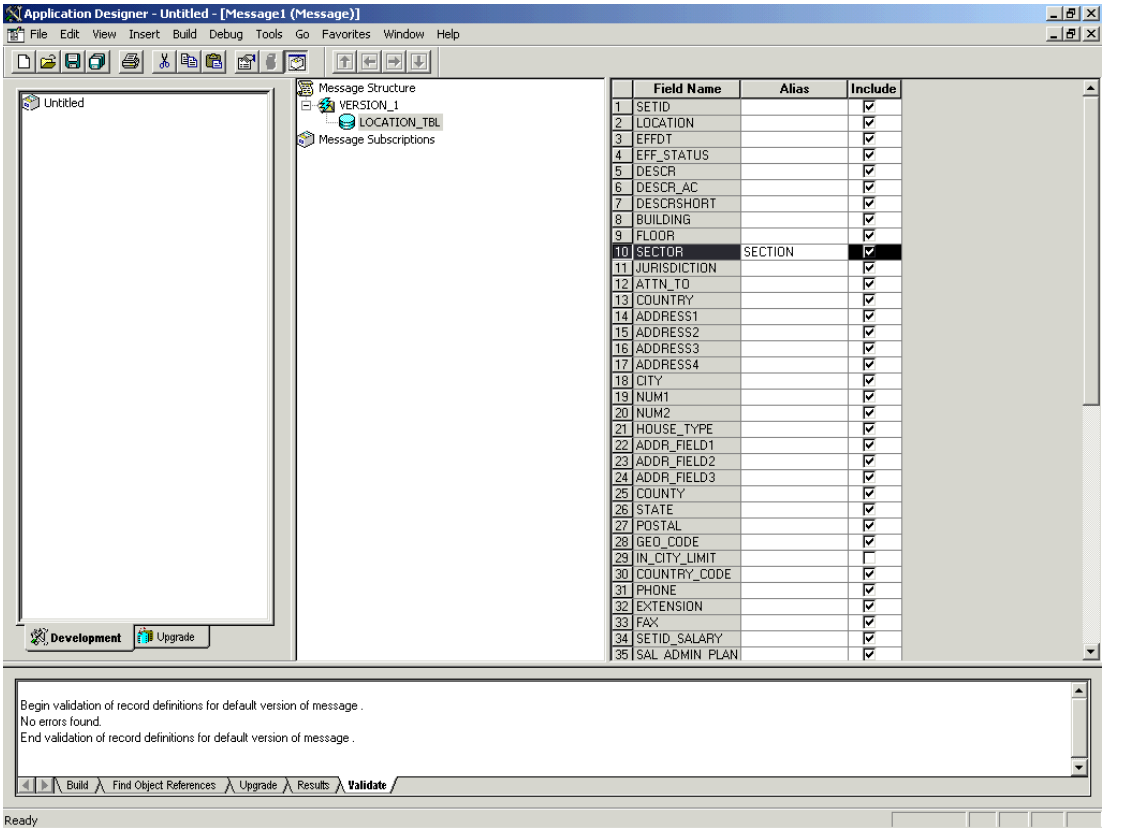
   In the example below, Floor, House Type, and In City Limit fields have been excluded from the message.

Figure 5-17 Application Designer - Excluding Table Fields



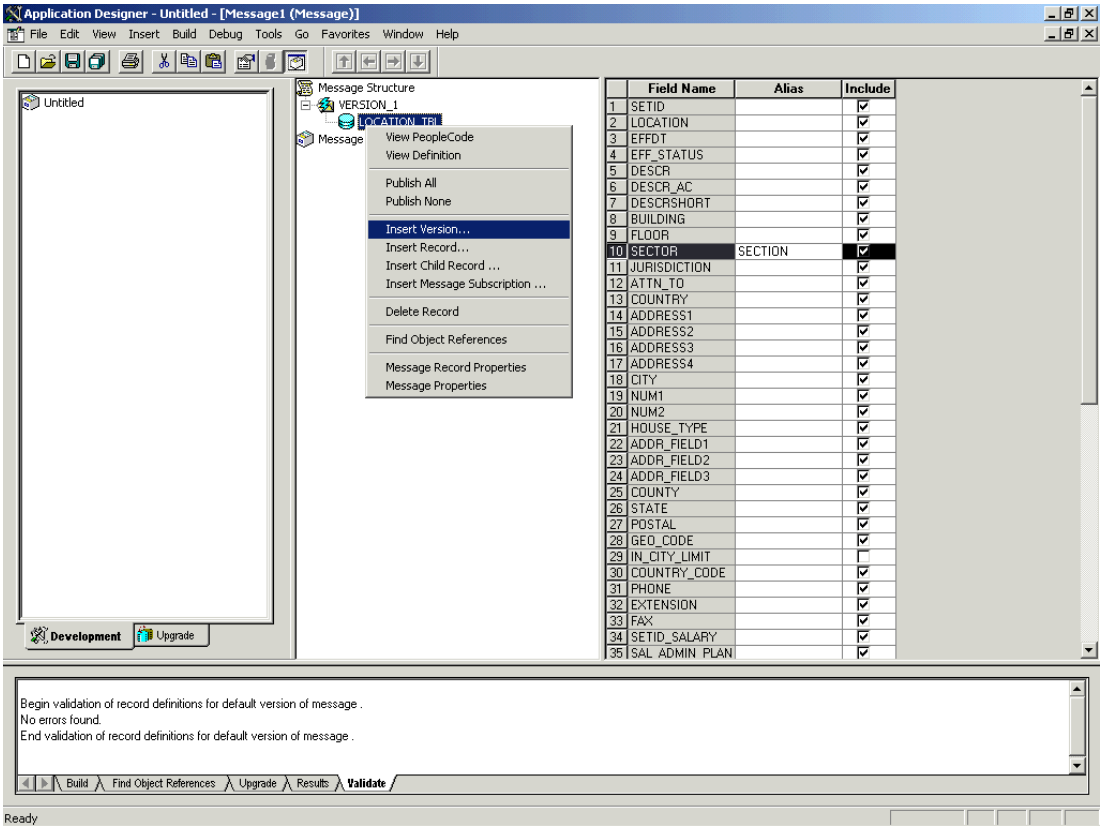
8. To change a field's output name, enter the new name in the Alias field. For example, in the following window, the Sector field has been renamed to Section.

Figure 5-18 Application Designer - Renaming Table Fields



9. To create a new version of the message, right-click the version and select Insert Version from the shortcut menu as shown in the following figure. This is exactly the same procedure as previously illustrated.

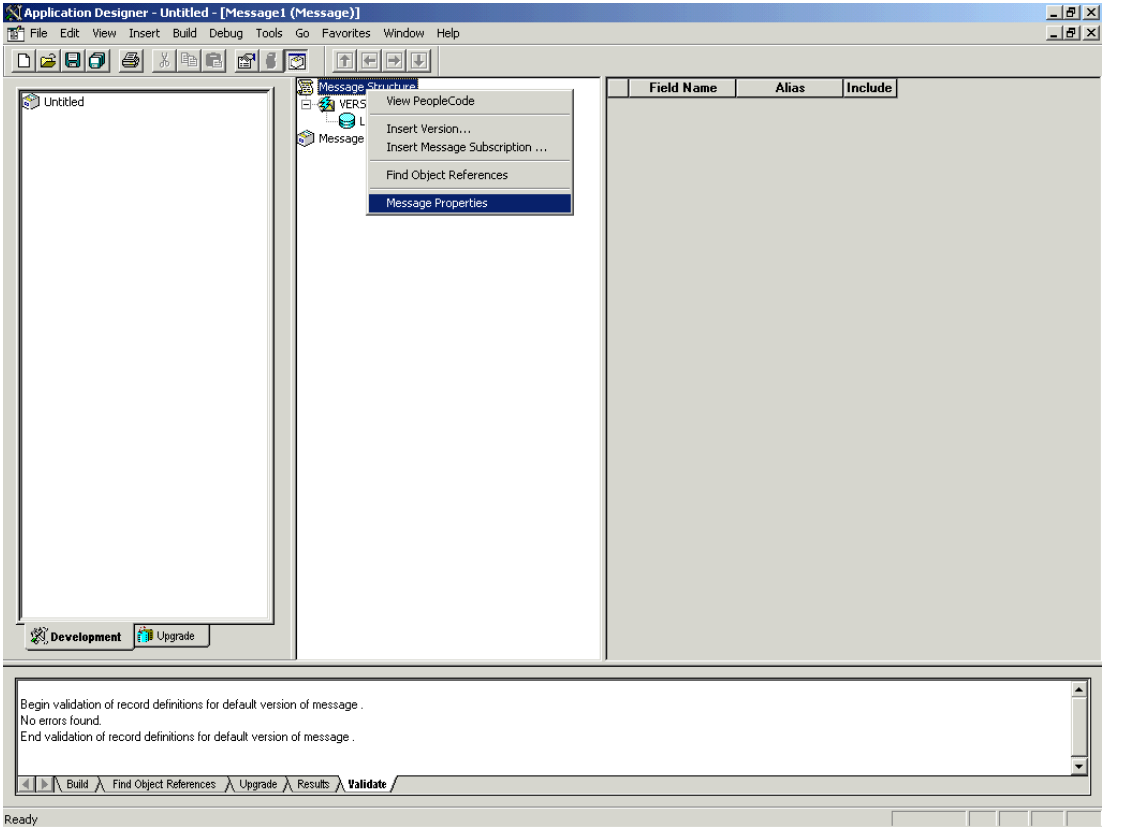
Figure 5-19 Application Designer - Creating New Version of the Message



After you are satisfied that your message is set up correctly, check that it is going to the right channel. You must set the Message Channel.

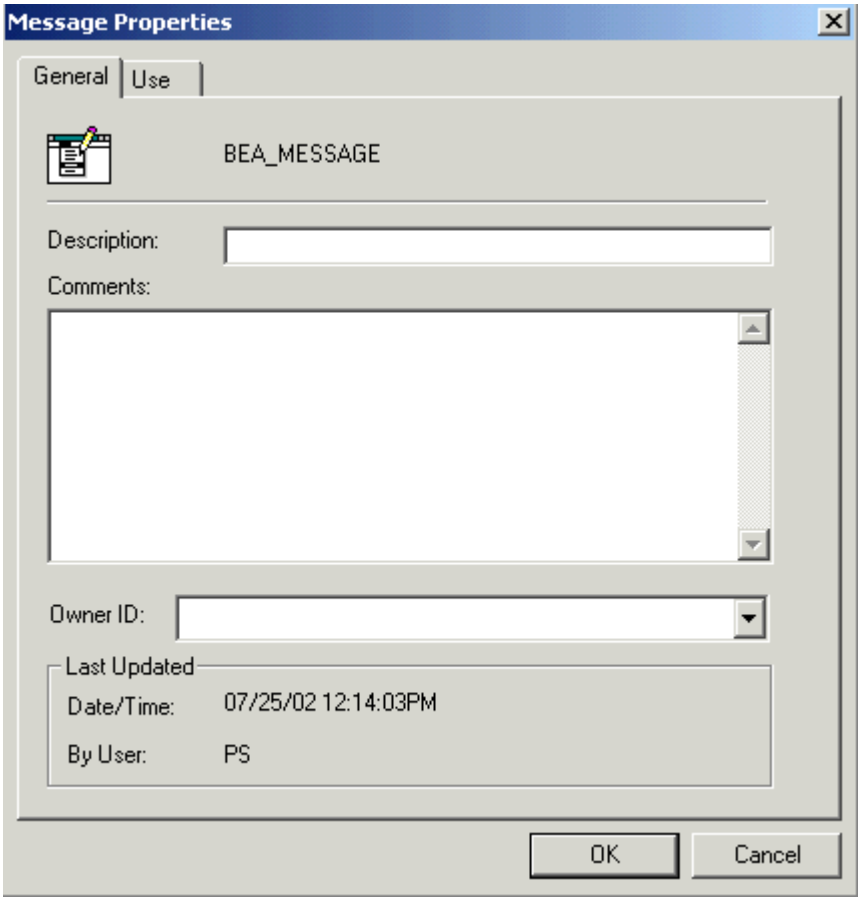
1. Open the Message Properties dialog box. For example, you can right-click the middle pane and select Message Properties from the shortcut menu.

Figure 5-20 Application Designer - Shortcut Menu



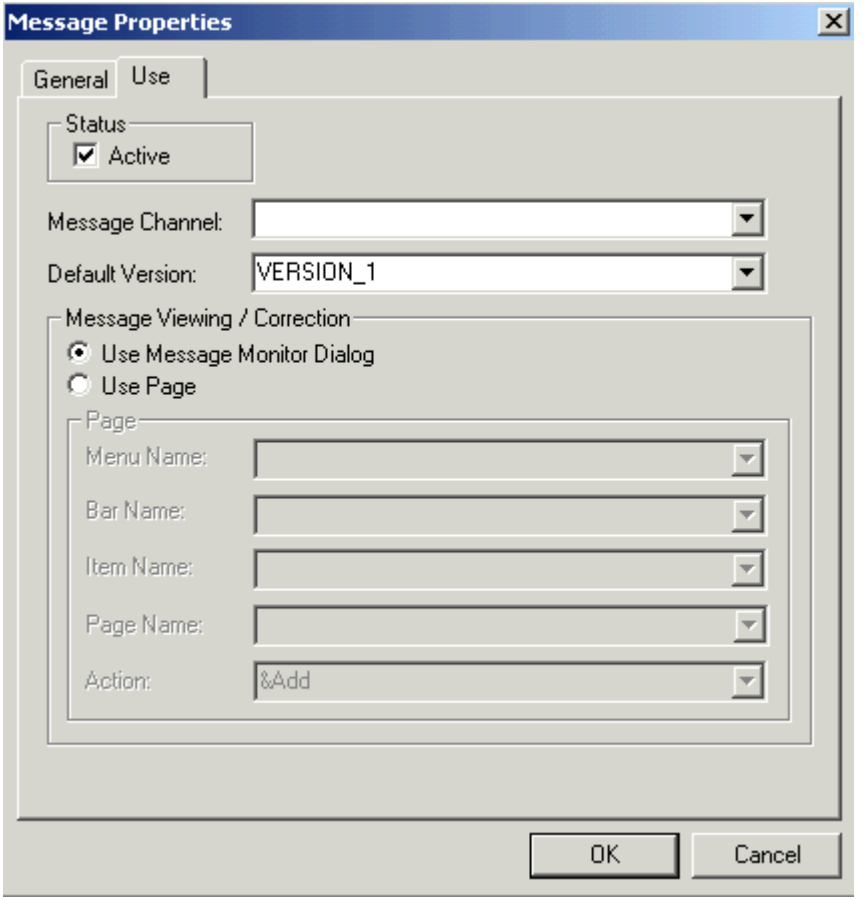
The Message Properties dialog box opens.

Figure 5-21 Message Properties



2. Enter a description and comments.
3. Then, click the Use tab.

Figure 5-22 Message Properties - Use Tab



a. The Status check box beside Active must be checked.

b. Click the Message Channel drop-down list box or enter the relevant Message Channel required.

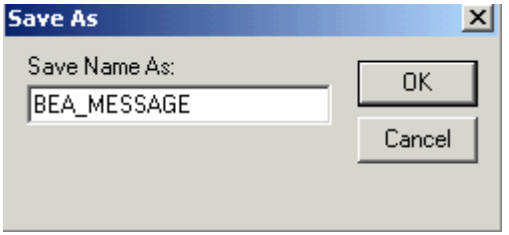
   The Use Message Monitor Dialog option button should be selected. Otherwise, the message does not display in the Application Message Monitor.

c. Enter the message details.

d. Click OK.

4. Choose File→Save As.
5. When the Save As dialogue box appears, enter the new file name and click OK.

Figure 5-23 Save As Dialog Box



You can now select the new message channel or a previously created one.

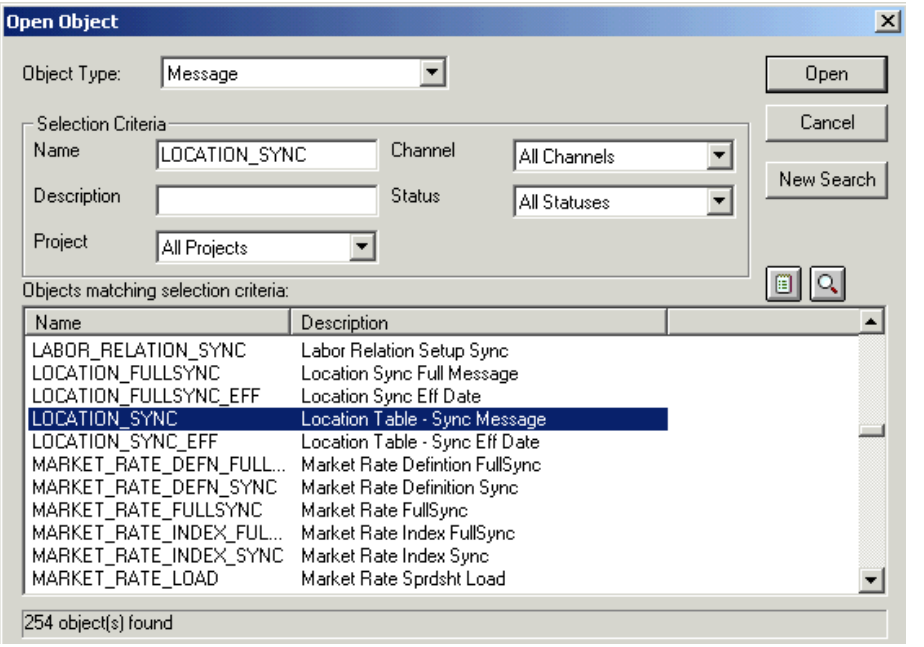
How to Configure an Existing Message

Select an existing message provided by PeopleSoft or create a new one. The creation of a new message is explained in "How to Create a Message" on page 13. For more information, ask your PeopleSoft administrator.

1. Choose File→Open→Message.

   The Open Object dialog box appears.

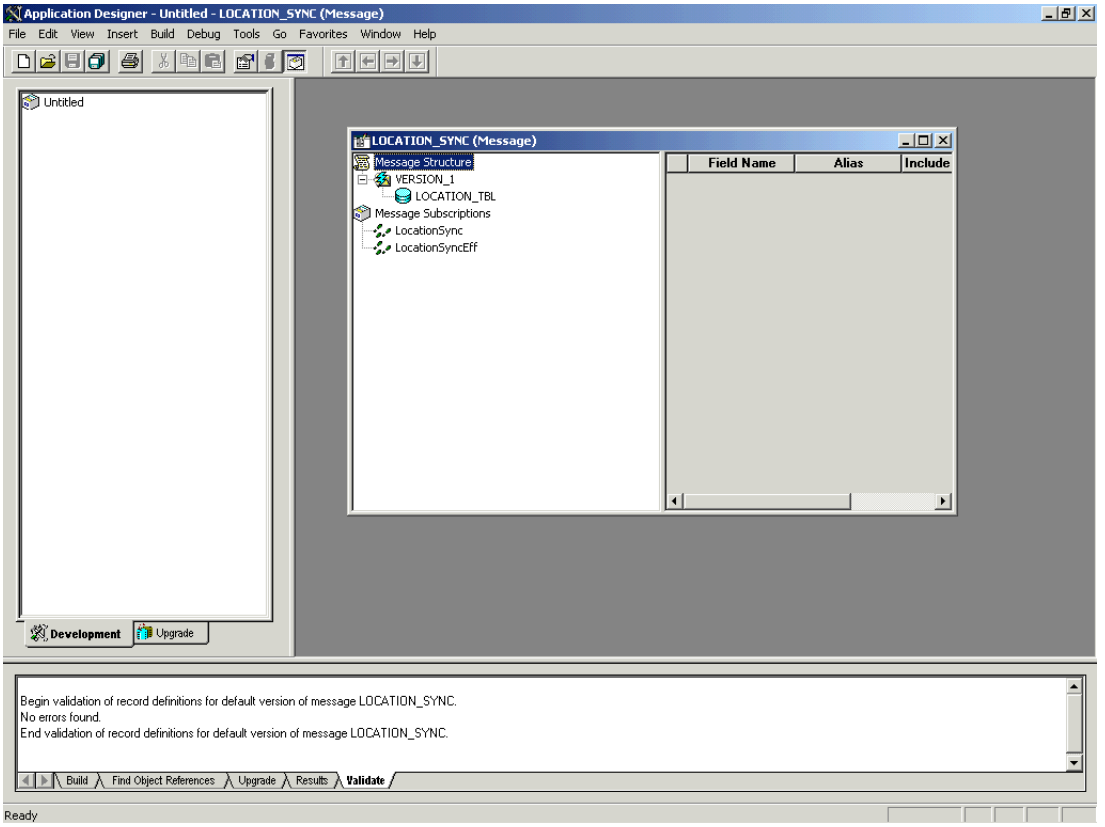
Figure 5-24 Open Object Dialog Box



2. Double-click the LOCATION_SYNC message.

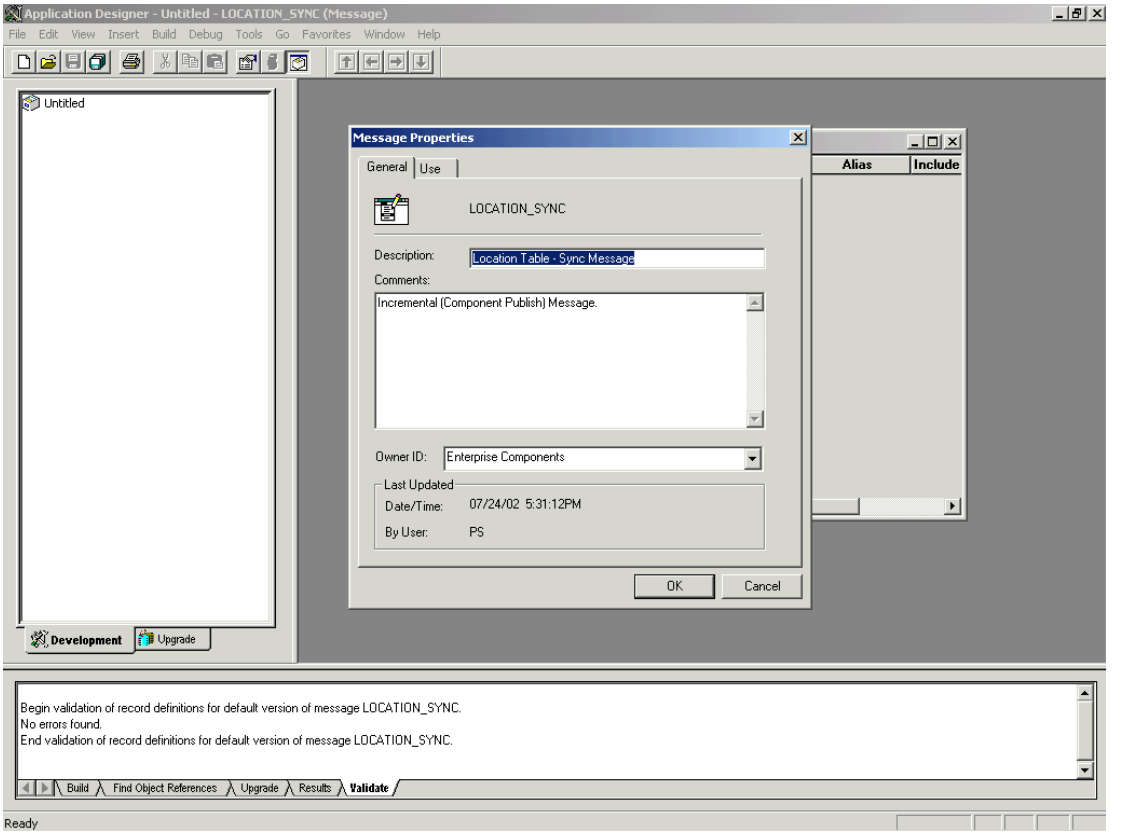
The LOCATION_SYNC window opens.

Figure 5-25 Application Designer - LOCATION_SYNC Window



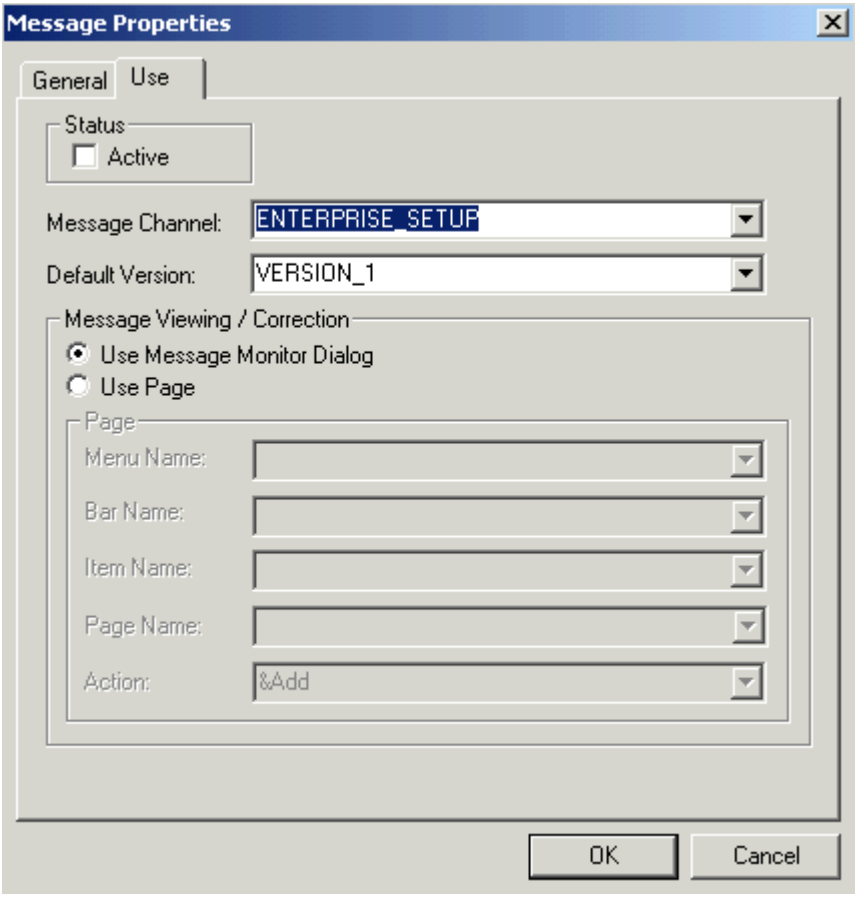
3. Display the Message Properties dialog box.

Figure 5-26 Message Properties Dialog Box



4. Click the Use tab in the Message Properties dialog box.

Figure 5-27 Message Properties - Use Tab



   a. Select the Active check box.

   b. Verify that the Message Channel drop-down list displays the message channel that you are using (for example, ENTERPRISE_SETUP).

   c. Click OK.

5. Save the message.

6. Insert the message into your project (recommended).

You are now ready to test the PeopleSoft event.

Test the Message Channel you have just created by using the PeopleSoft SimpleFileHandler. Testing outside of the WebLogic Server facilitates problem identification. As illustrated in the previous example, if you modify a location within PeopleSoft, an XML document representing the event is sent to a file which indicates that the message channel has been correctly configured.

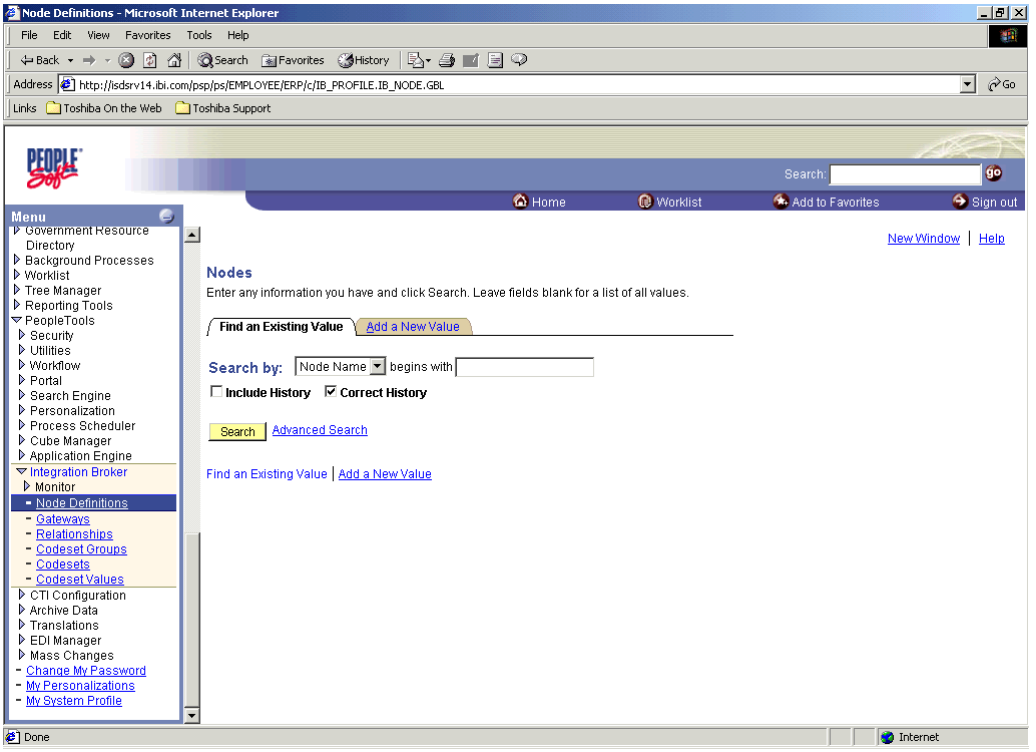
You can proceed with the WebLogic Integration.

Creating a New Node in PeopleSoft Version 8.4 or Higher

1. Click PeopleTools→Integration Broker→Node Definitions.

   The Node Definitions window opens.

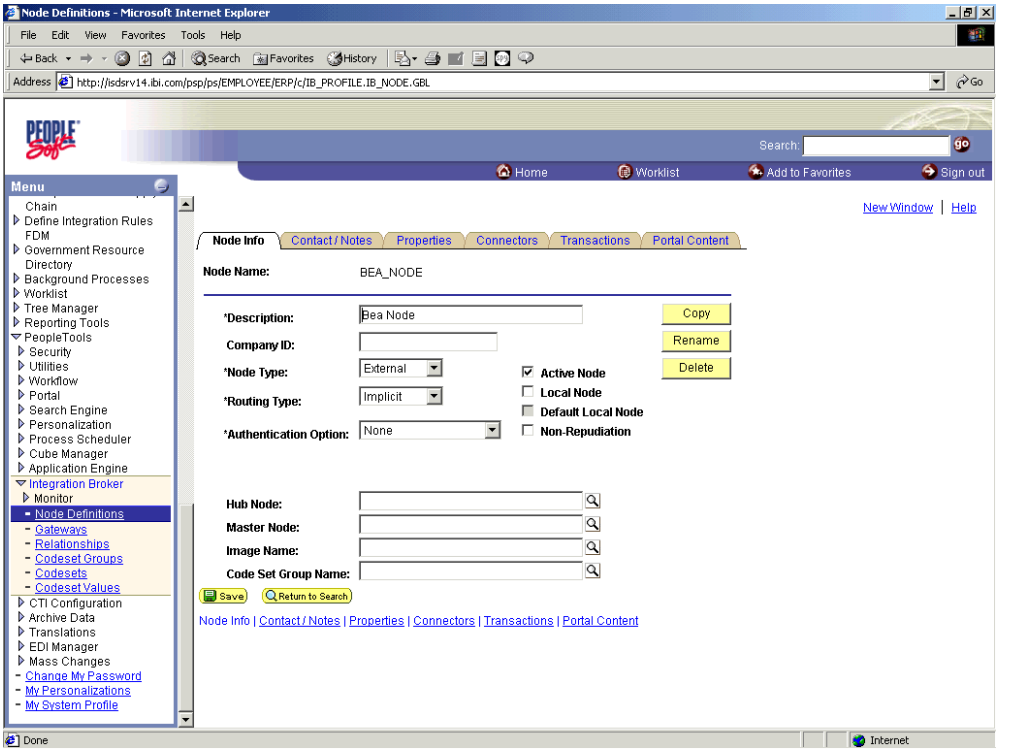
Figure 5-28 Nodes Definition Window



2. Click the Add a New Value tab.

   The Node Info tab appears.

Figure 5-29 Node Definitions Window - Node Info Tab



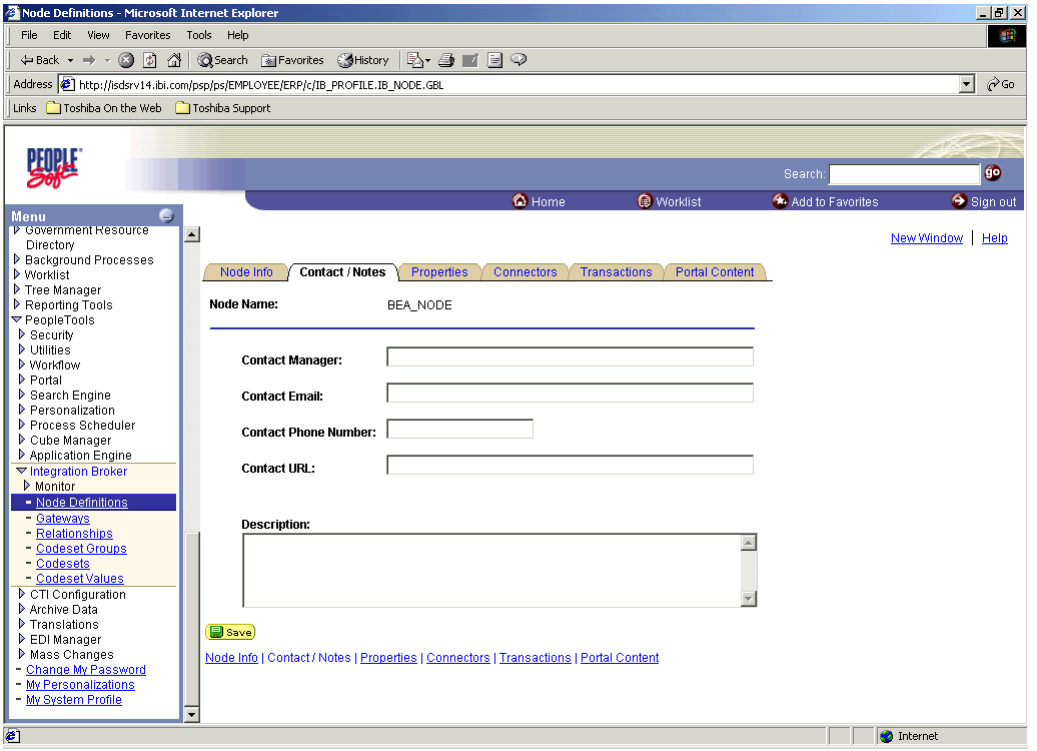
   a. Enter a description.

   b. Ensure that Node Type is set to External and that Routing Type is set to Implicit.

3. Click the Contact / Notes tab.

   This step is optional.

Figure 5-30 Node Definitions Window - Contact / Notes Tab

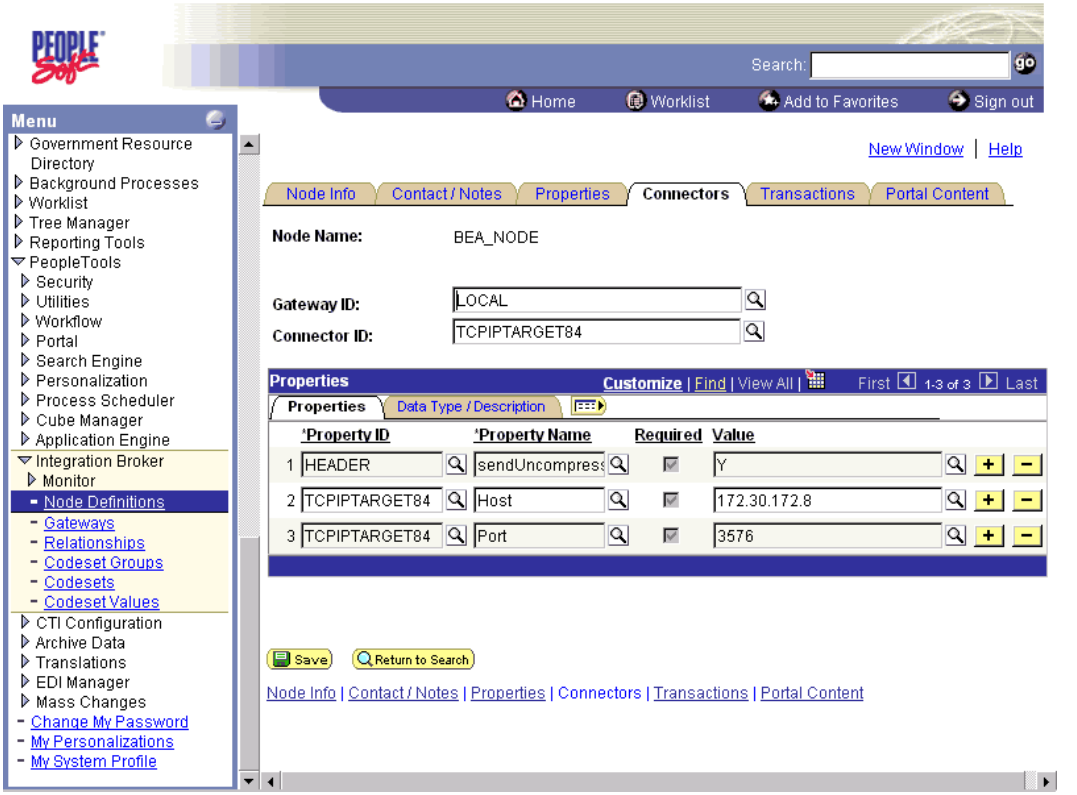


You can enter contact information if you are creating a remote node.

Properties are not required for the current node so you need not click the Properties tab.

4. Click the Connectors tab.

Figure 5-31 Node Definitions Window - Connectors Tab



The list of connectors that display are registered with the selected gateway. This field specifies the target connector appropriate to the communication method preferred by the current node.

a. Select the local Gateway ID.

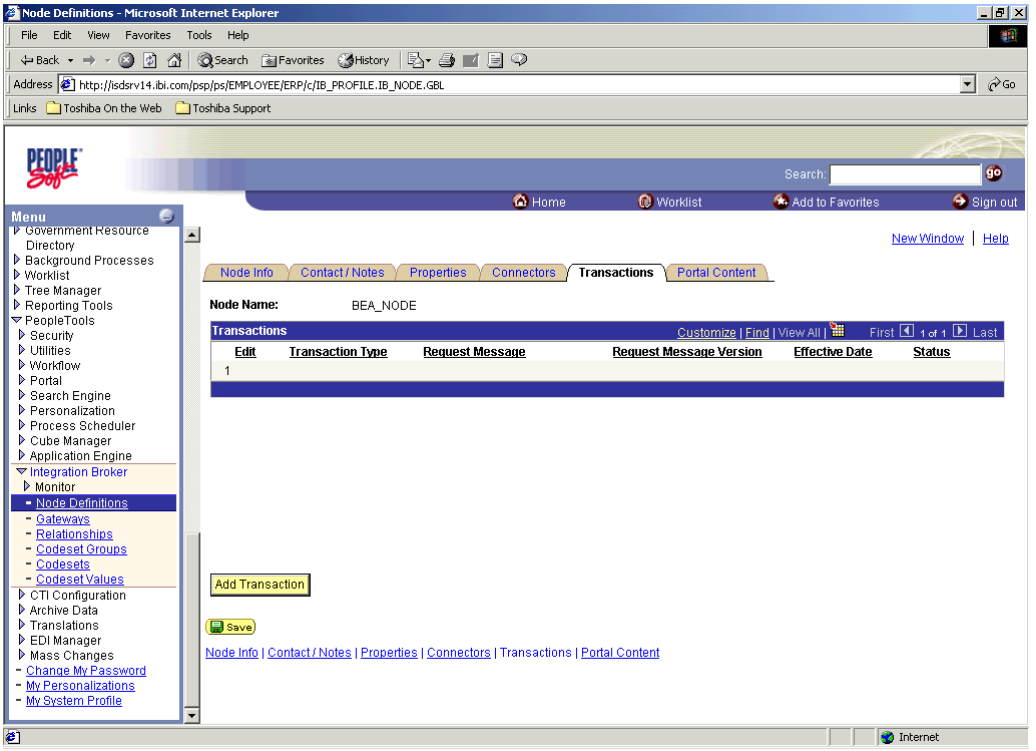
b. Select TCPIPTARGET84 as the Connector ID.

c. Enter the host name or IP address of the WebLogic Integration Server in the Property ID field of the host property name.

d. Enter the TCP port that was entered when the application view was added.

e. Click the Transactions tab.

Figure 5-32 Node Definitions Window - Transactions Tab

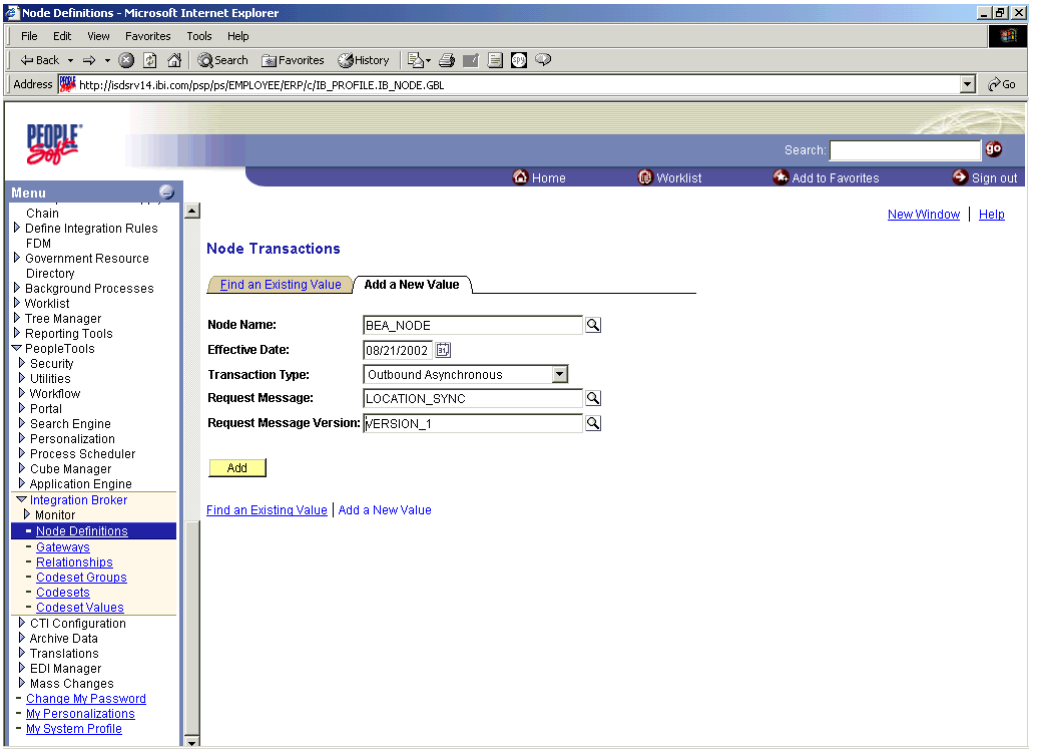


5. Click Save.

6. Click Add Transaction.

   Node Transactions appears.

Figure 5-33 Node Definitions Window - Node Transactions



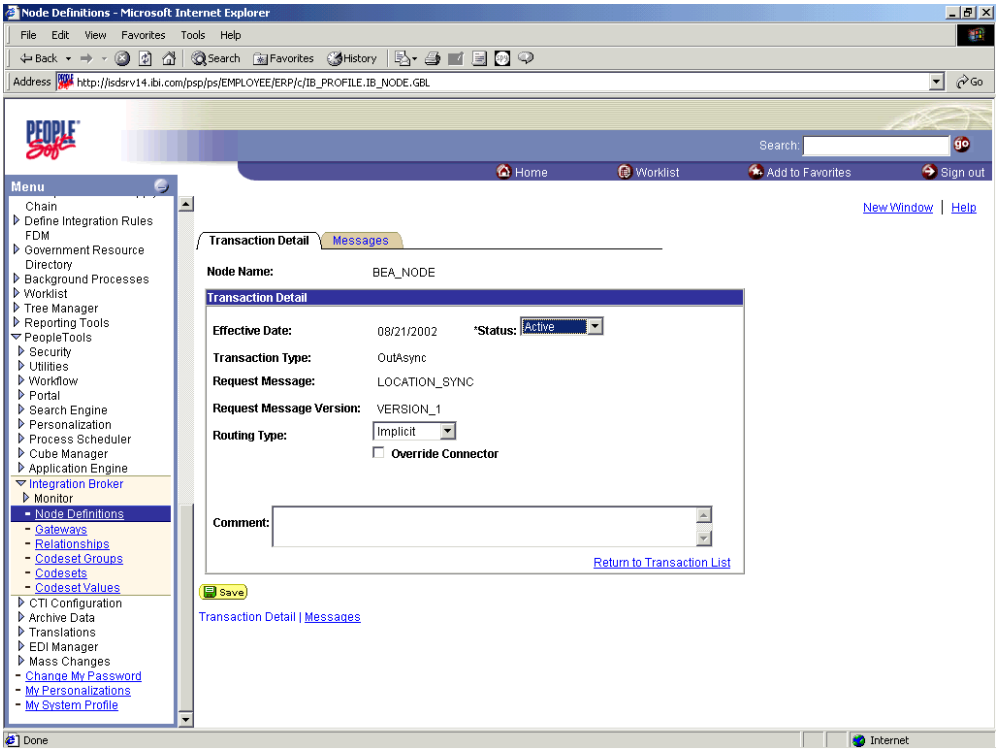
   a. Select a transaction type from the drop-down list. Currently, only the Outbound Asynchronous transaction type is supported.

   b. Enter a value for the Request Message field and a value for the Request Message Version field.

c. Click Add.

The Transaction Detail tab appears.

Figure 5-34 Node Definitions Window - Transaction Detail Tab



7. Ensure that Routing Type is set to Implicit.
8. Repeat the add transaction process for every message you are using.
9. Click Save.
10. Return to the Node Info tab and click Save.

6 Creating Event Schema for Application Messages

This section describes how to create schemas for the PeopleSoft 8 event adapter. It includes the following topics:

- Establishing the Working Directory
- Establishing a Connection to PeopleSoft
- Creating Event Schemas
- Creating Application Views for PeopleSoft XML
- Adding a PeopleSoft TCP/IP Event to an Application View
- Sample Event Using a Business Process Workflow

Before you invoke event processing in the WebLogic environment using a PeopleSoft 8 event adapter, you must create an event schema for the XML event. Use the BEA Application Explorer to generate the event schema directly against a PeopleSoft message object.

The BEA Application Explorer creates the schema definitions for the XML event schema.

This section illustrates how to create schemas for the PeopleSoft 8 event adapter. For additional information on using the BEA Application Explorer, see the *BEA Application Explorer Installation and Configuration Guide*.

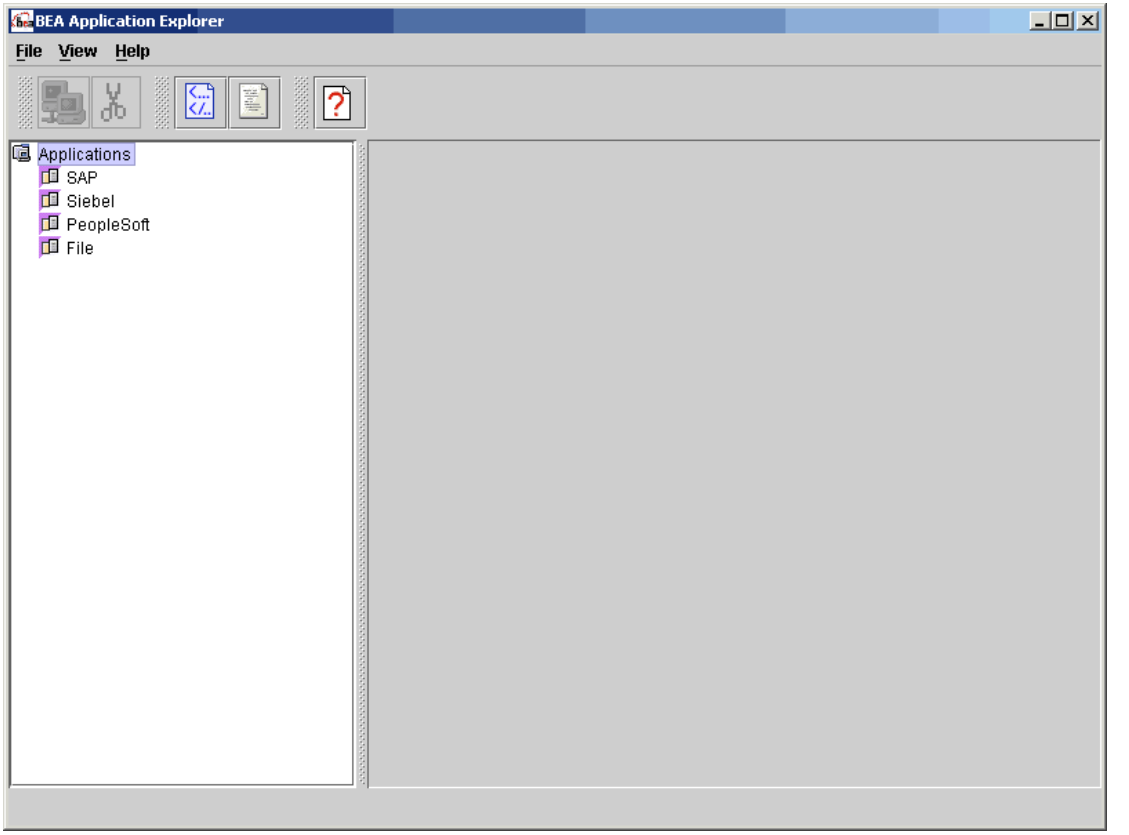
Launch the BEA Application Explorer:

Figure 6-1 BEA Application Explorer Initialization Window

Establishing the Working Directory

Establish the directory associated with your WebLogic Server to import event and service XML schemas into the application view repository.

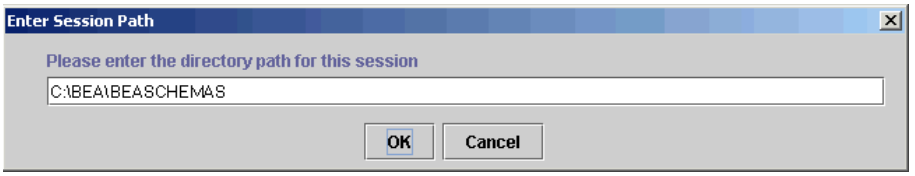
Figure 6-2 BEA Application Explorer



1. Choose File→Session.

   The Enter Session Path dialog box appears.

Figure 6-3 Enter Session Path Dialog Box



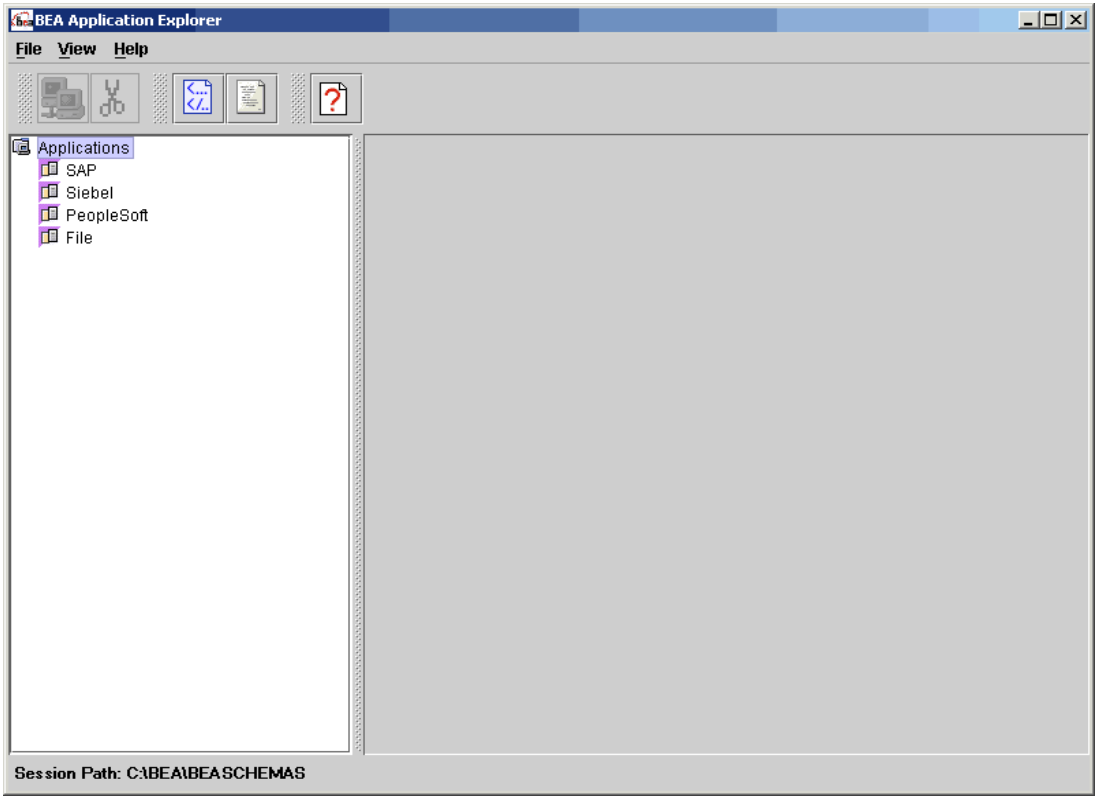
2. Enter a folder name, for example, `C:\BEA\BEASCHEMAS`.

   In this example `C:\BEA\BEASCHEMAS` serves as the BEA Application Explorer's working directory. This is the location in which schemas are placed when they are generated by the BEA Application Explorer.

3. Click OK.

   Note the Session Path at the bottom of the BEA Application Explorer window.

Figure 6-4 BEA Application Explorer

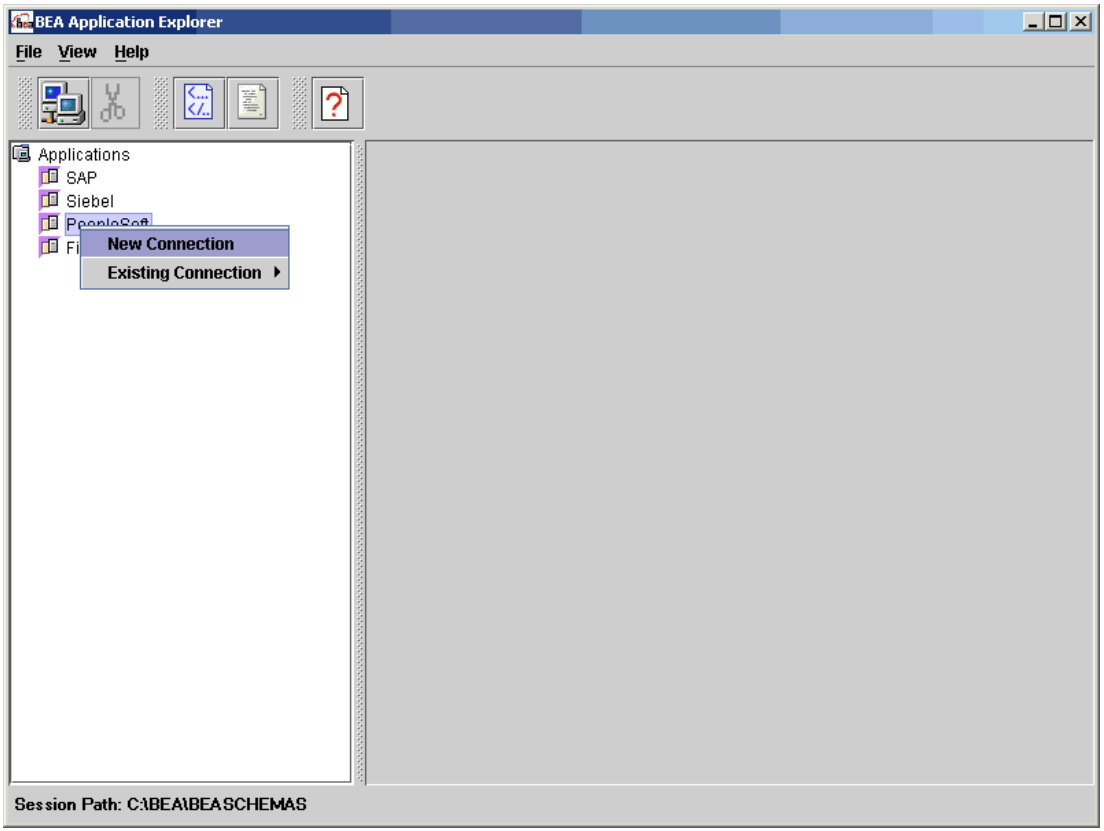


Establishing a Connection to PeopleSoft

To establish a connection to PeopleSoft:

1. Right-click PeopleSoft in the left pane and select New Connection.

Figure 6-5 BEA Application Explorer - New Connection



The Input dialog box appears.

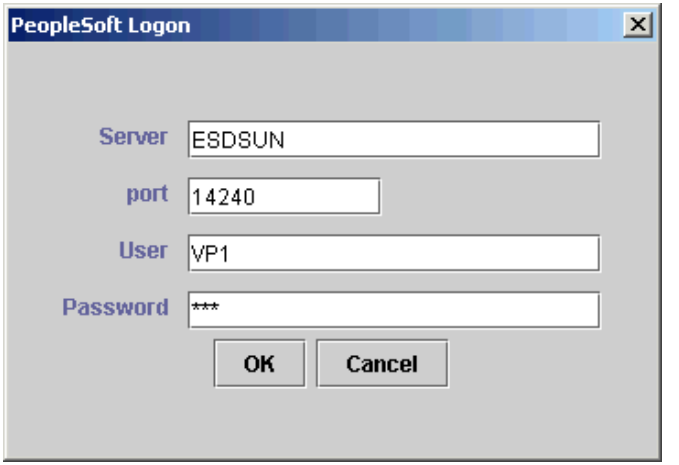
Figure 6-6 Input Dialog Box

a. Enter a name for the PeopleSoft connection, for example, PeopleSoftConnection.

b. Click OK.

The PeopleSoft Logon dialog box appears.

Figure 6-7 PeopleSoft Logon Dialog Box



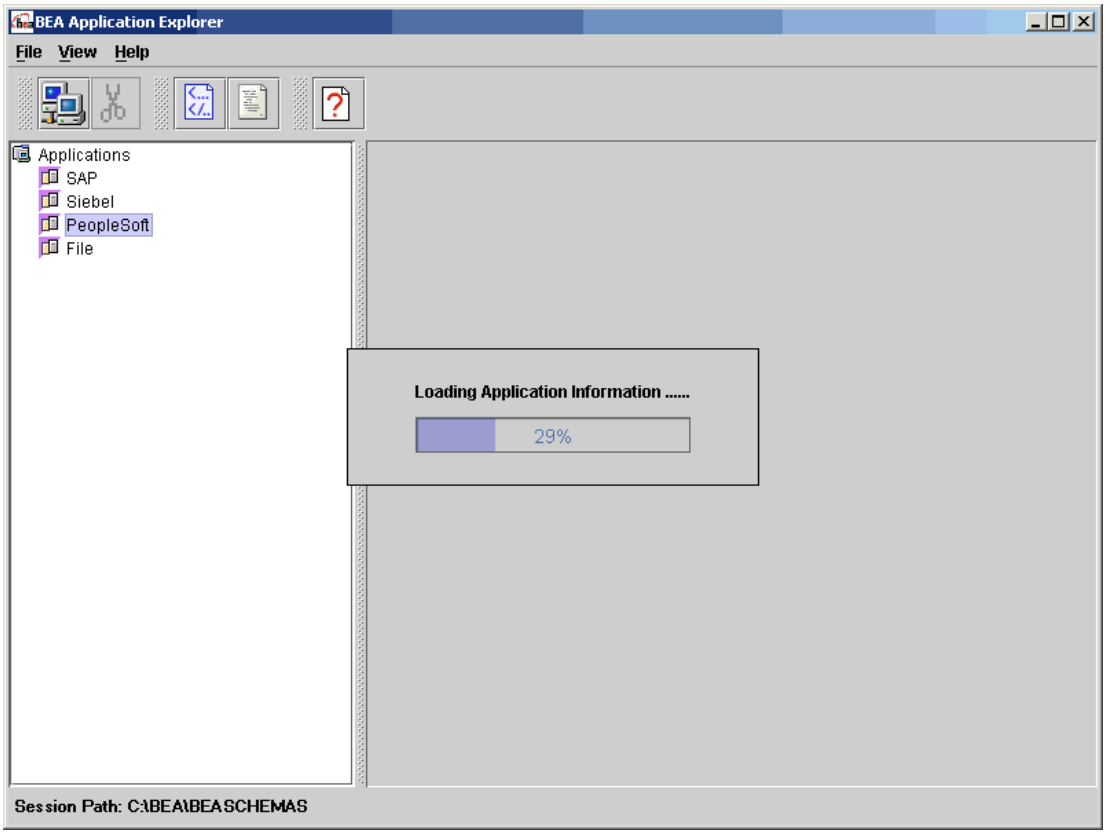
2. Enter the parameters to establish the connection to the PeopleSoft system.

   The configuration parameters you supply are those used by PeopleSoft client applications for connecting to the PeopleSoft system. The port number is the number of the Jolt Listener. For more information, see your PeopleSoft documentation or ask your PeopleSoft system administrator.

3. Click OK.

   The Loading Application Information status message appears.

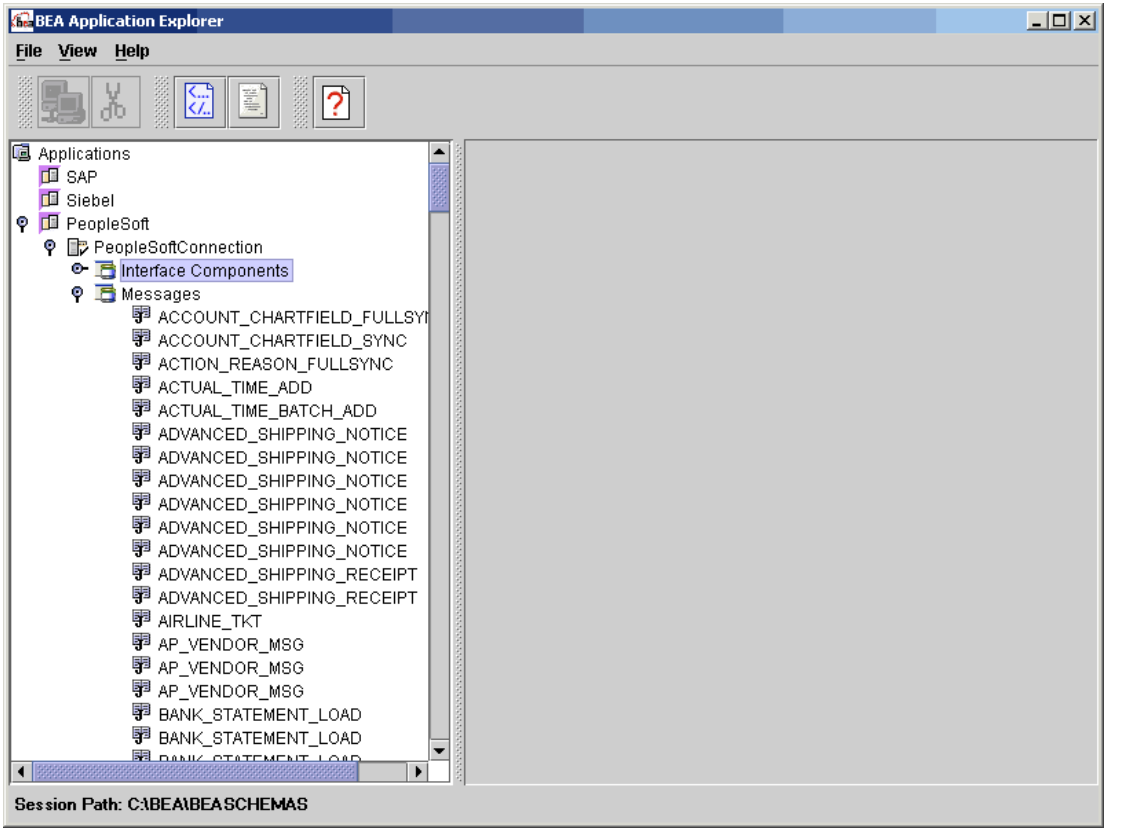
Figure 6-8 BEA Application Explorer - Loading Application Information



The process of loading the internal cached file may take several minutes. This speeds up the process for subsequent displays and schema creation for other component interfaces and messages.

After the application information loads, the Application Explorer connects to the PeopleSoft system to display a list of available messages.

Figure 6-9 BEA Application Explorer - Messages

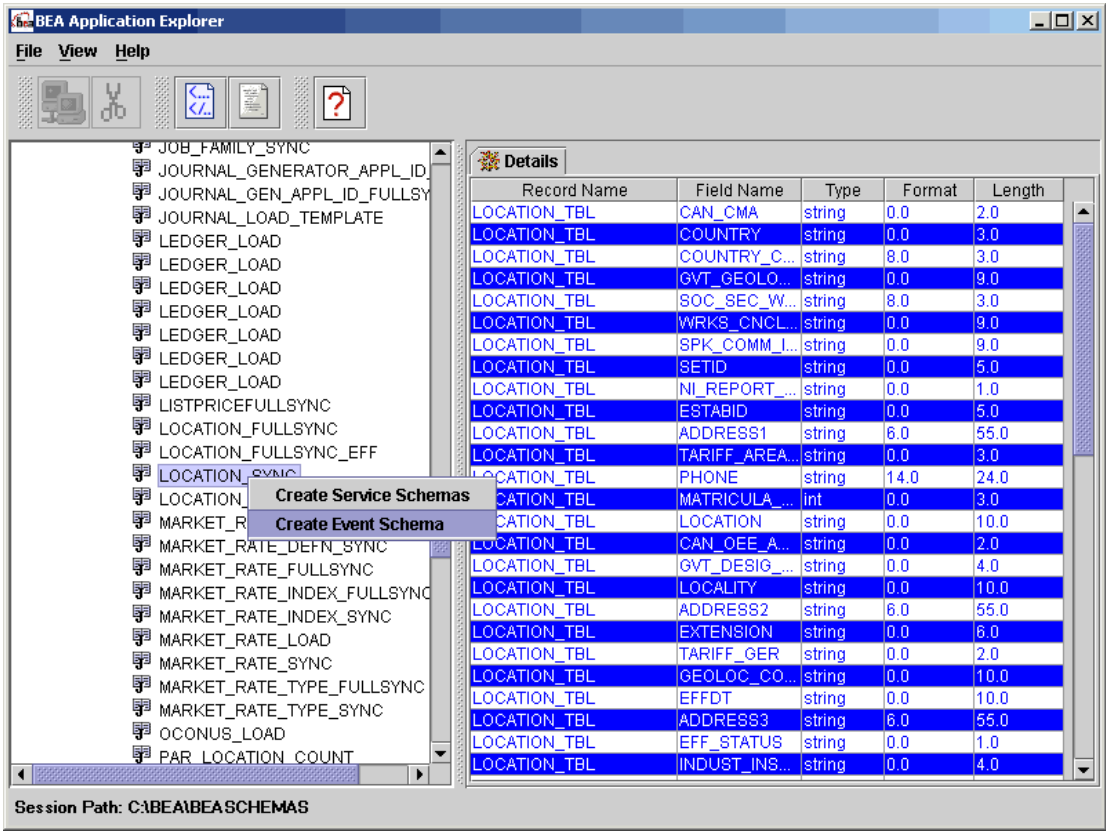


4. Browse all of the available messages in the PeopleSoft system by expanding Messages.

Creating Event Schemas

After you establish the working directory and the connection to PeopleSoft (see "Establishing the Working Directory" on page 3 and "Establishing a Connection to PeopleSoft" on page 6), you are ready to create event schemas.

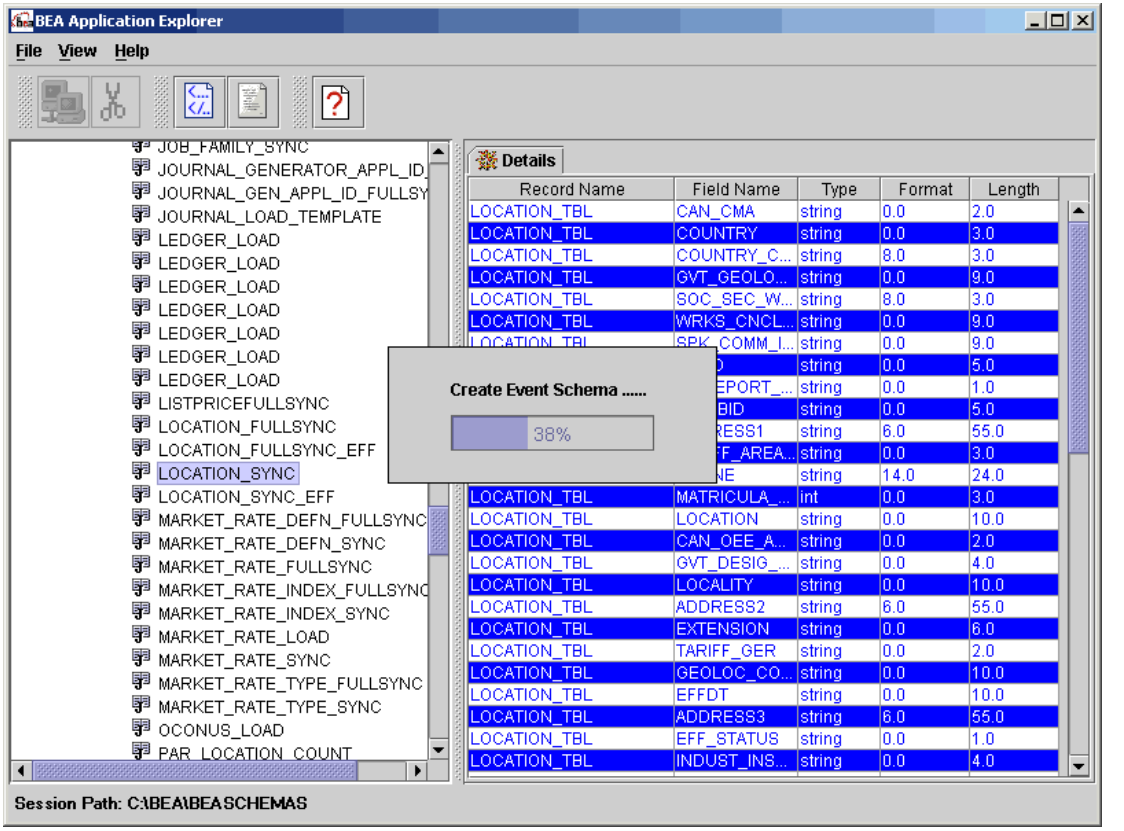
Figure 6-10 BEA Application Explorer - Creating an Event Schema



1. Right-click a PeopleSoft message.
2. Select Create Event Schema.

A Create Event Schema progress indicator appears, and the BEA Application Explorer generates the WebLogic Integration schema.

Figure 6-11 BEA Application Explorer - Creating an Event Schema



For example, after you right-click a message, for example, LOCATION_SYNC, and select Create Event Schema, the WebLogic Integration schema is generated and appears in the right pane.

A directory structure is created automatically within the working directory, `C:\BEA\BEASCHEMAS`.

The BEA Application Explorer creates a folder called peoplesoft. It also creates subfolders for each configured PeopleSoft connection to contain the schemas that are created. In this case, the schemas are located in the folder called PeopleSoftConnection. PeopleSoftConnection is the connection name you established when you connected to the PeopleSoft system using the BEA Application Explorer.

The following members have been added to folder, `C:\BEA\BEASCHEMAS\peoplesoft\PeopleSoftConnection`:

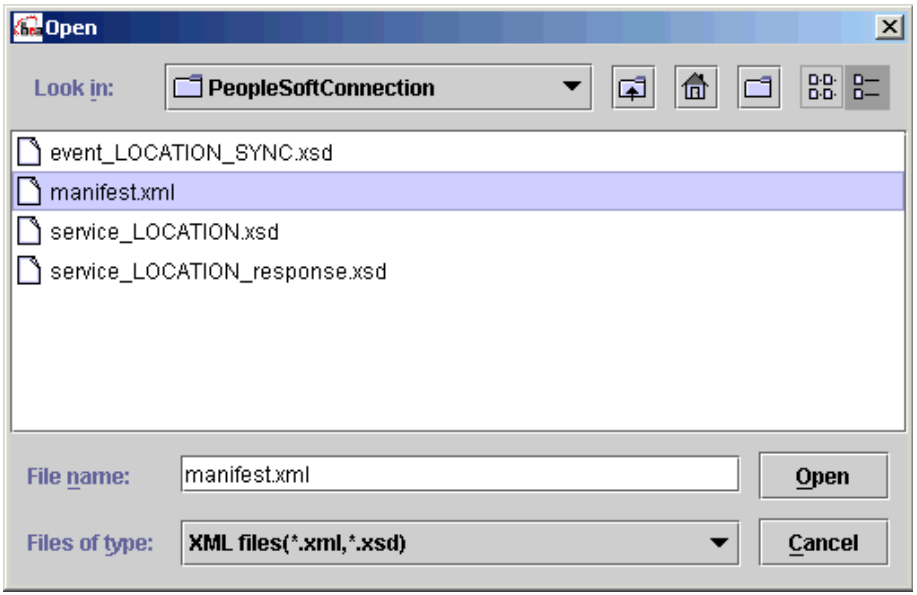
- `manifest.xml`
- `event_LOCATION_SYNC.xsd`

You also can view the created schemas using the BEA Application Explorer. This is a convenient way to browse the schemas that have been published for WebLogic Integration.

3. Select View→View XML.

   The Open dialog box appears.

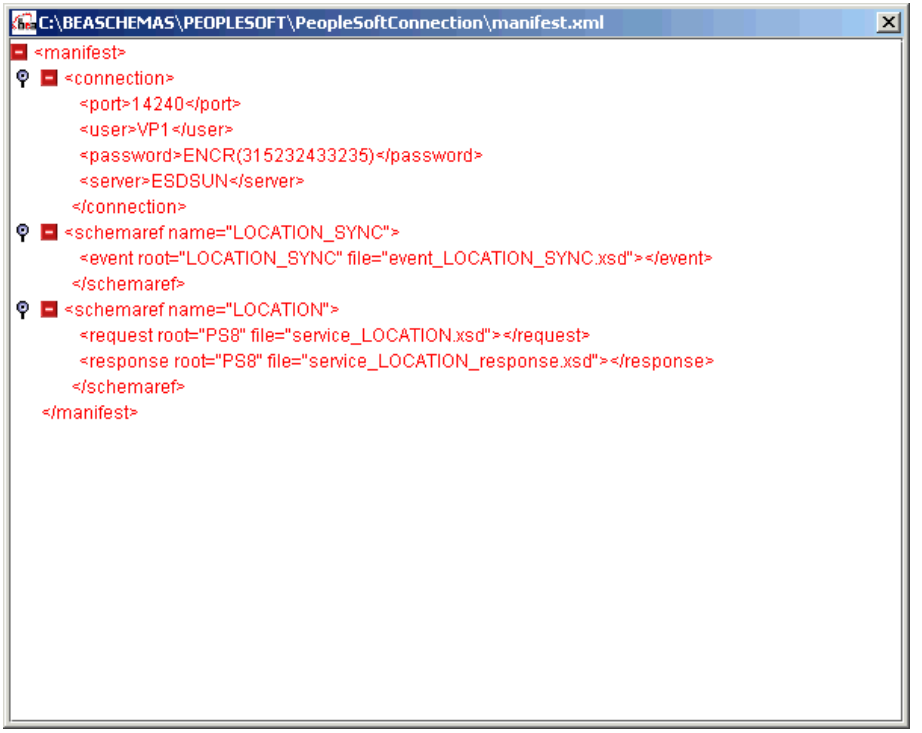
Figure 6-12 Open manifest.xml



4. Navigate to the current working directory under the session path to select the desired XML file to view any of the created schemas and `manifest.xml`.

For example, the `manifest.xml` file for the message LOCATION_SYNC contains connection and configuration information. You can use this to test event processing in a PeopleSoft system with business process management workflows.

Figure 6-13 manifest.xml File



Creating Application Views for PeopleSoft XML

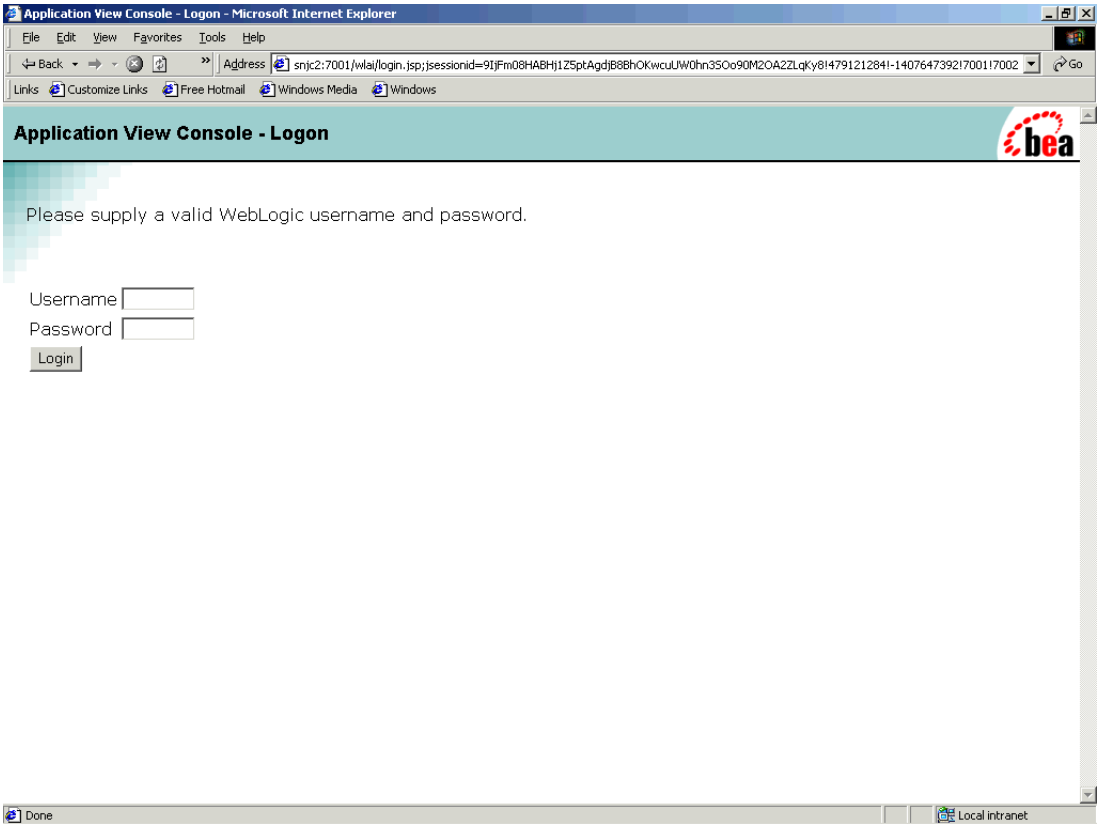
When you define an application view, you create an XML-based interface between WebLogic Server and a particular Enterprise Information System (EIS) application within your enterprise. After you create the application view, a business analyst can use it to create business processes that use the application. For any adapter, you can create any number of application views, each with any number of services and events.

1. Log on to the WebLogic Integration Application View Console. For more information, see "Logging On to the WebLogic Integration Application View Console" in "Defining an Application View" in *Using Application Integration*:

   - For WebLogic Integration 7.0, see `http://edocs.bea.com/wli/docs70/aiuser/2usrdef.htm`
   - For WebLogic Integration 2.1, see `http://edocs.bea.com/wlintegration/v2_1sp/aiuser/2usrdef.htm`

   The Application View Console can be found at `http://`*`host:port`*`/wlai`. Here, *`host`* is the IP address or domain name where the WebLogic Server is installed, and *`port`* is the socket on which the server is listening. The default port, if not changed at install time, is 7001.

Figure 6-14 Application View Console - Logon Window

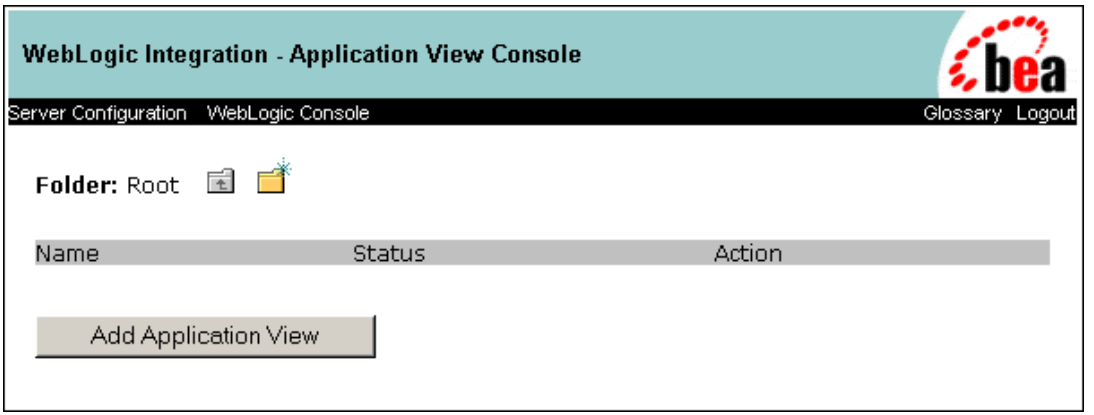


Note: If the user name is not `system`, it must be included in the `adapter` group. For more information on adding the administrative server user name to the `adapter` group, see the *BEA WebLogic Adapter for PeopleSoft Installation and Configuration Guide*.

2. Enter your password and click Login.

The WebLogic Integration - Application View Console window opens.

Figure 6-15 Application View Console Window



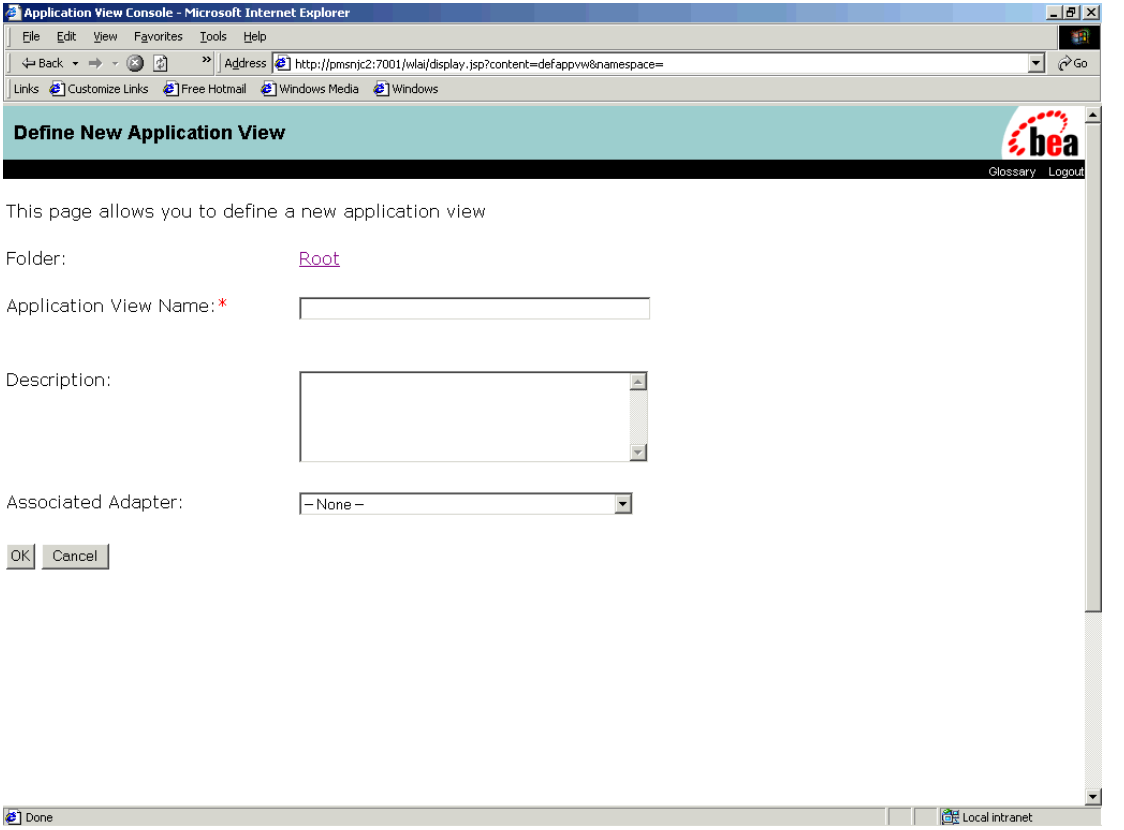
3. Click Add Application View to create a new application view for the adapter.

   An application view enables a set of business processes for this adapter's target EIS application. For more information, see "Defining an Application View" in *Using Application Integration*:

   - For WebLogic Integration 7.0, see
     `http://edocs.bea.com/wli/docs70/aiuser/2usrdef.htm`
   - For WebLogic Integration 2.1, see
     `http://edocs.bea.com/wlintegration/v2_1sp/aiuser/2usrdef.htm`

The Define New Application View window opens.

Figure 6-16 Define a New Application View Window



4. In the Define New Application View window, add the following information:

   a. In the Application View Name field, enter a name.

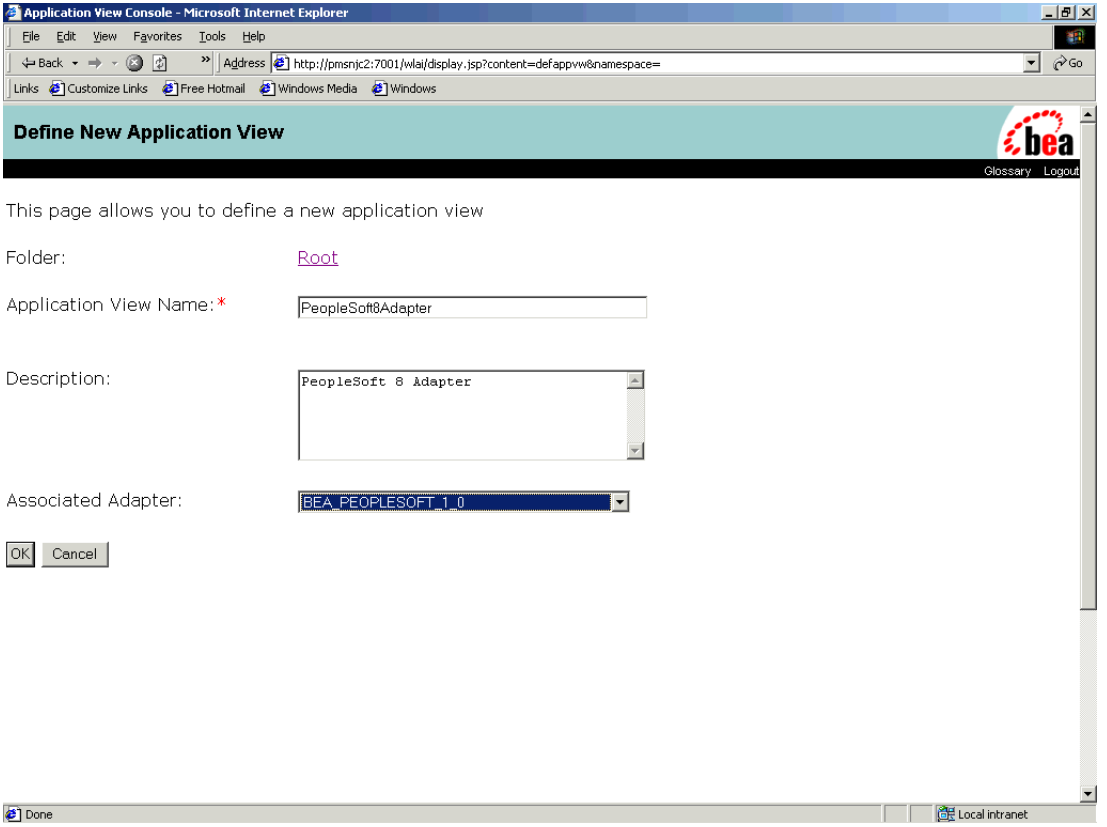
   This name should describe the set of functions performed by this application.

   Each application view name must be unique to its adapter. Valid characters include a-z, A-Z, 0-9, and _ (underscore).

   b. In the Description field, enter any relevant notes. These notes are viewed by users when they use this application view with business process management workflows.

c. From the Associated Adapter drop-down list, select the BEA_PEOPLESOFT_1_0 Adapter to create this application view.

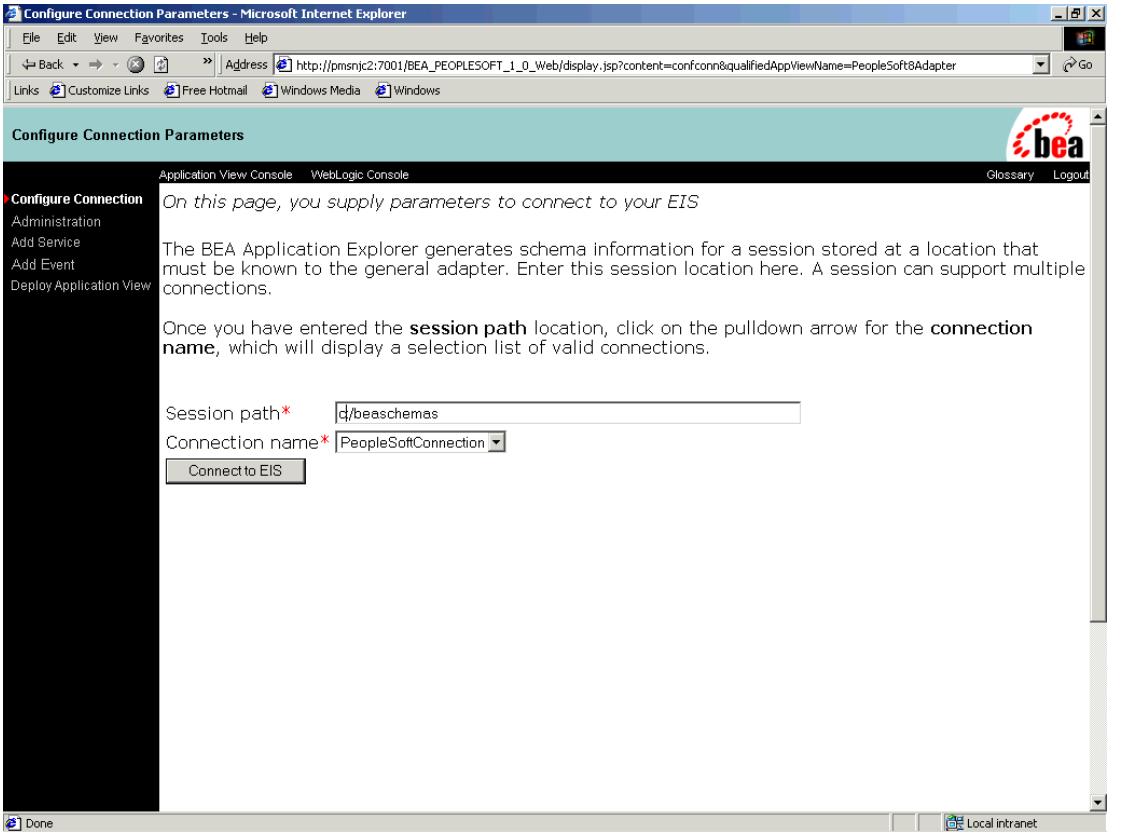
Figure 6-17 Define a New Application View - With Information Added



5. Click OK.

   The Configure Connection Parameters window opens.

Figure 6-18 Configure Connection Parameters Window



6. In the Configure Connection Parameters window, define the location of the schema definitions for the service or event request. This information is necessary for the application view to interact with the target EIS. Enter this information only once per application view.

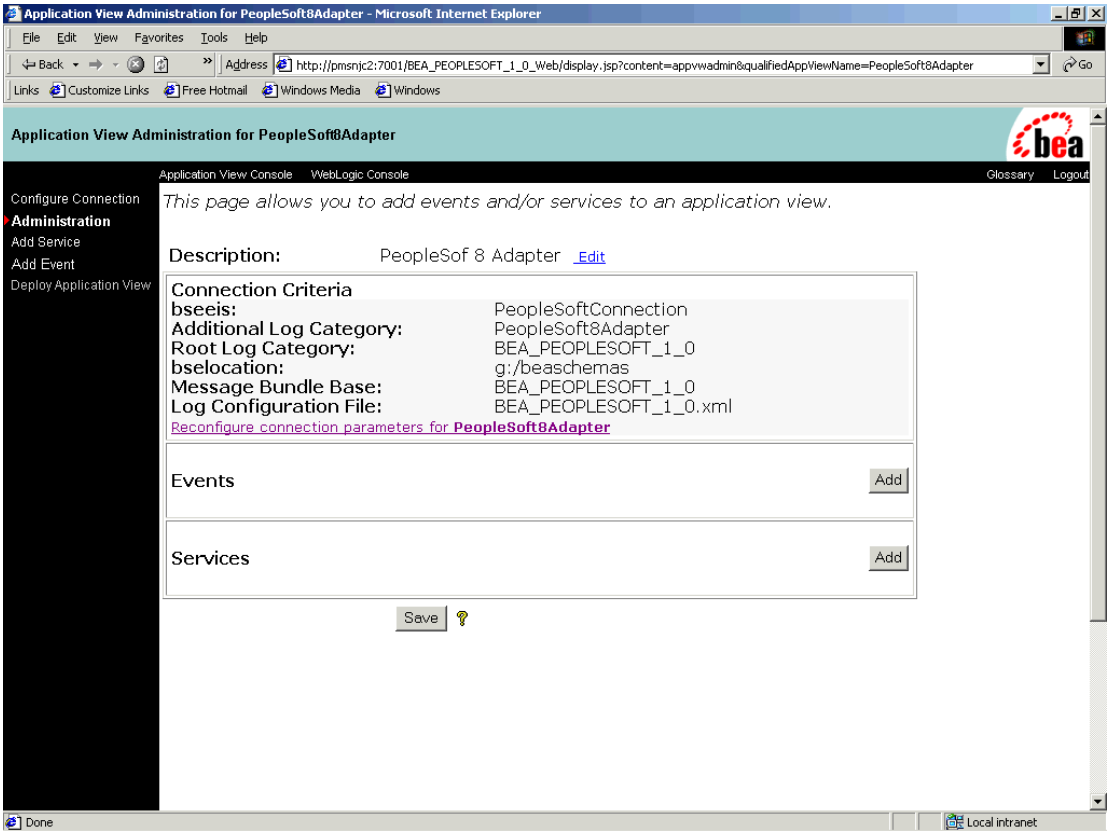
   - Session path is the location where BEA Application Explorer is installed and is two steps up from the connection directory (`C:\Program Files\BEA Systems\BEA Application Explorer\sessions\default`).

- Connection name is the name of the connection used for creating schemas, where the schema's `manifest.xml` file is located.

7. Click Connect to EIS.

   The Application View Administration window opens.

Figure 6-19 Application View Administration Window



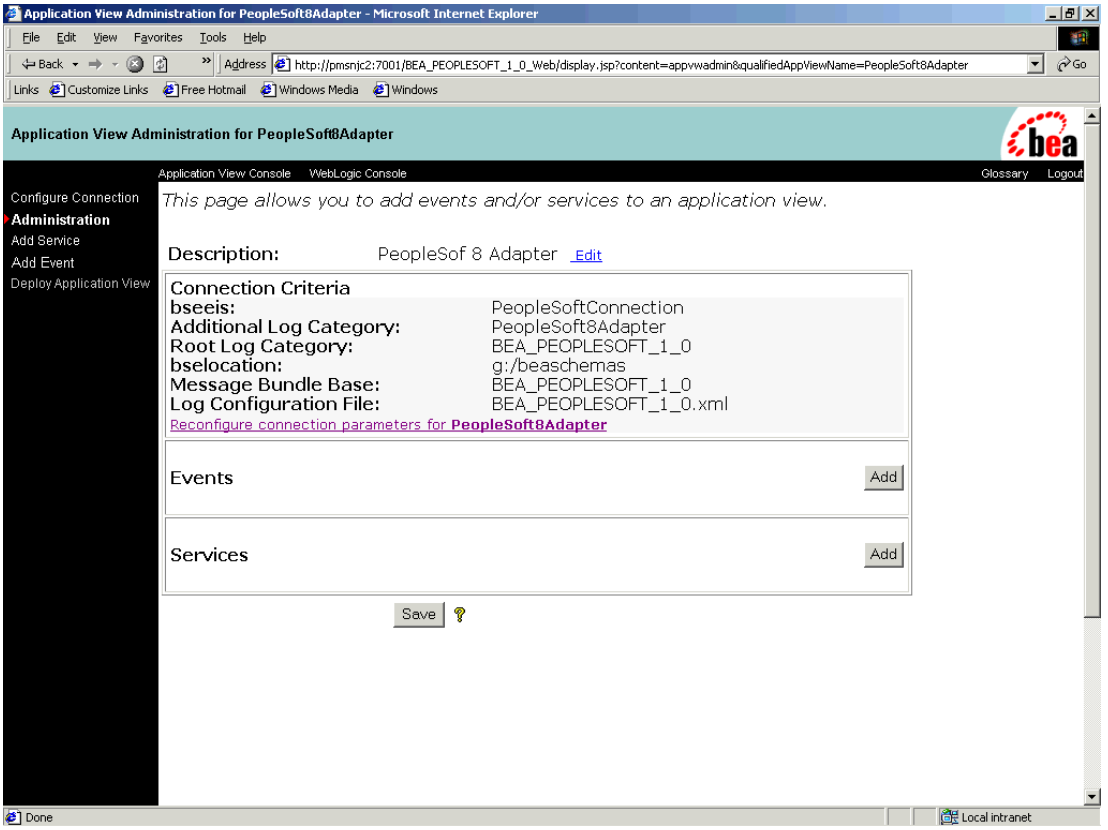
Adding a PeopleSoft TCP/IP Event to an Application View

The TCP/IP event is the process by which PeopleSoft, through Application Messaging, sends an XML file representing the PeopleSoft event to the WebLogic Integration. The PeopleSoft XML document is passed to an event variable that is set in a business process management workflow.

1. Click Add in the Events pane of the Application View Administration window.

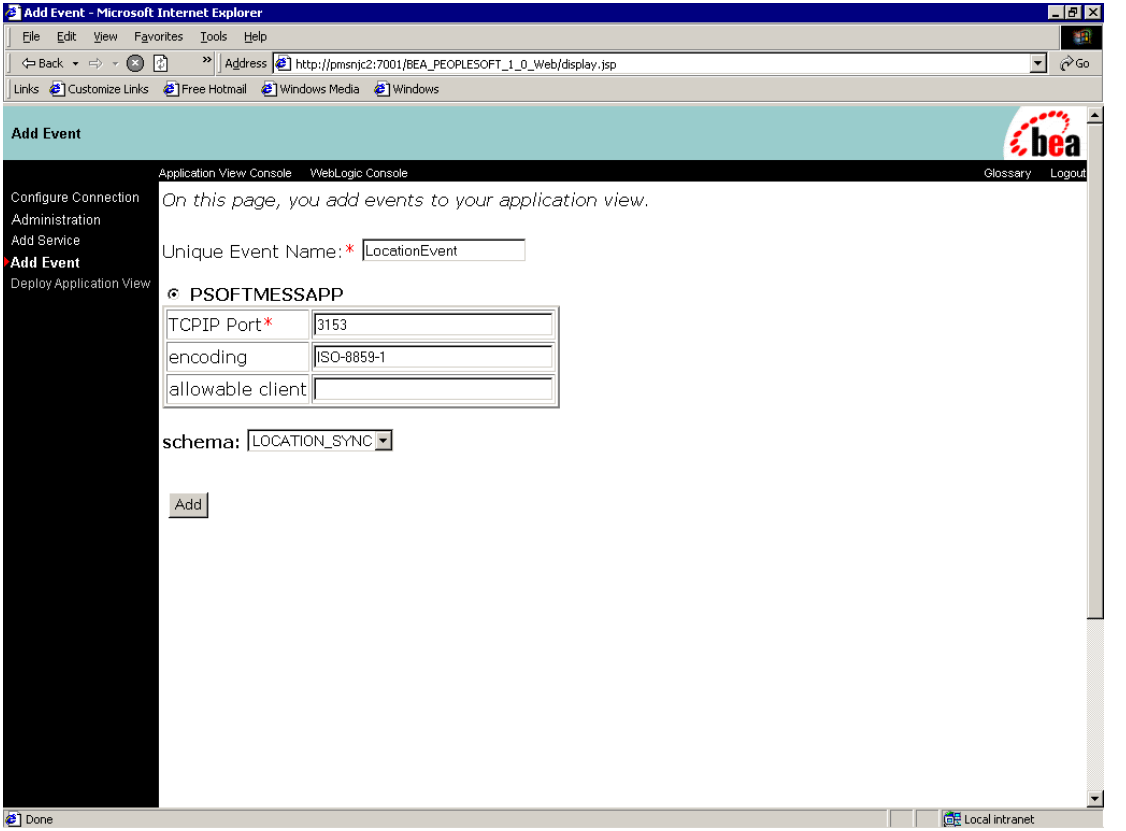
   For information about application views, see "Creating Application Views for PeopleSoft XML."

Figure 6-20 Application View Administration Console Window



The Add Event window opens.

Figure 6-21 Add Event Window

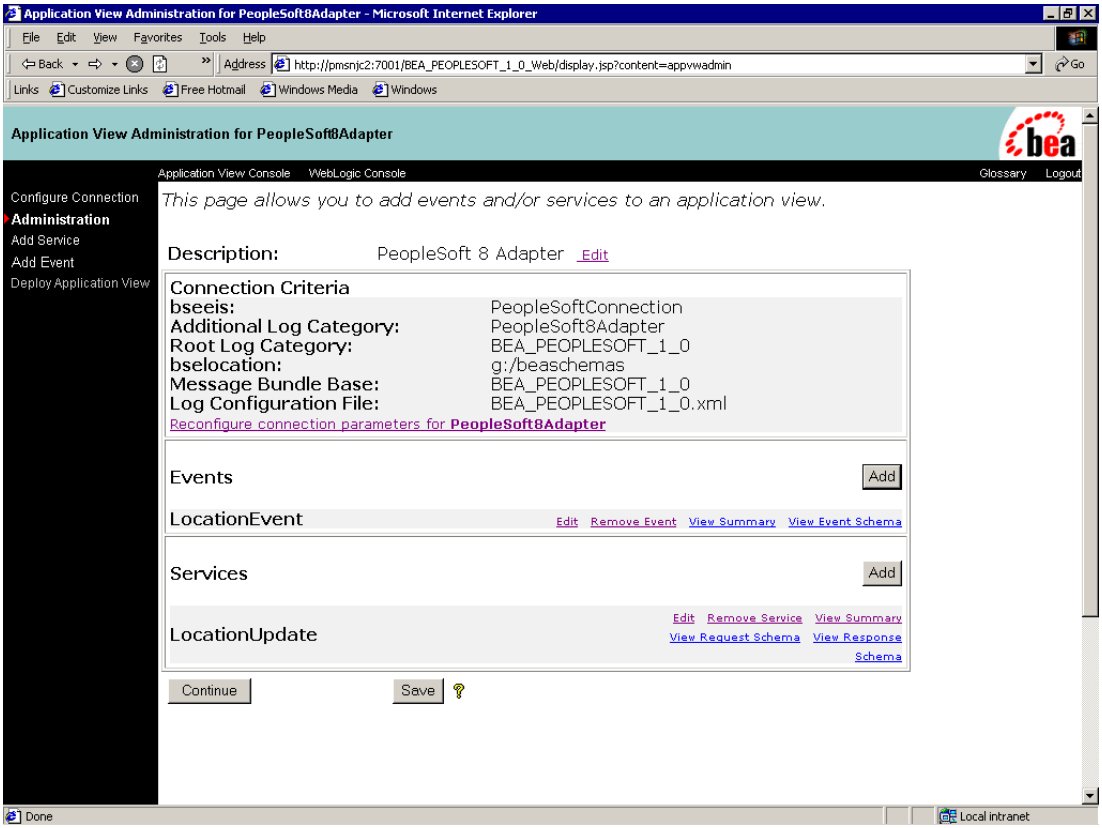


2. In the Unique Event Name field, enter an event name, for example, LocationEvent.
3. Click the PSOFTMESSAPP option button.
4. Enter the TCP/IP port that has been configured in the PeopleSoft application gateway. For more information, see the *BEA WebLogic Adapter for PeopleSoft 8 Installation and Configuration Guide*.

5. Click Add to add the event.

   Note that LocationEvent has been added to the Events pane.

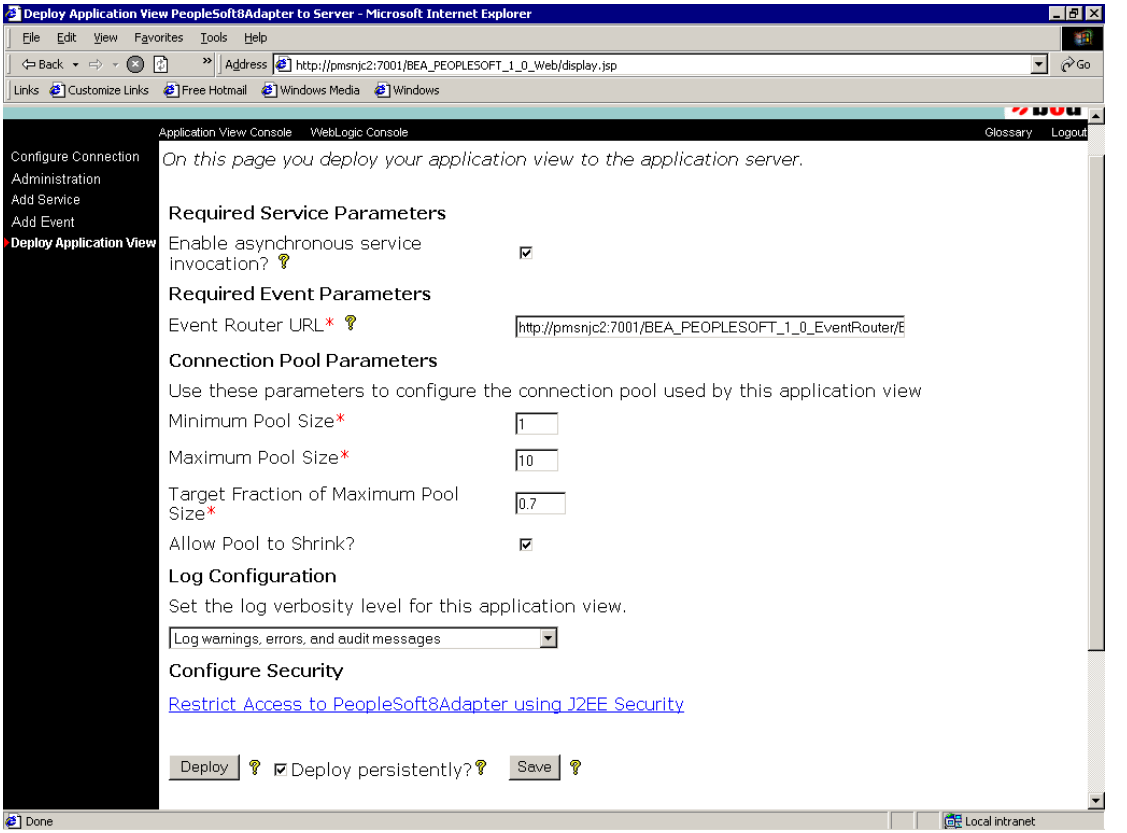
Figure 6-22 Application View Administration Window - LocationEvent Added



6. Click Continue to deploy the application view.

The Deploy Application View window opens.

Figure 6-23 Deploy Application View Window



7. Click Deploy. You also may choose to click Save and deploy the application view at a later time.

   The Summary for Application View window opens (not illustrated).

After the event is deployed, you can employ the event in business process workflows or write custom code. For more information, see "Using Application Views in the Studio" in *Using Application Integration*:

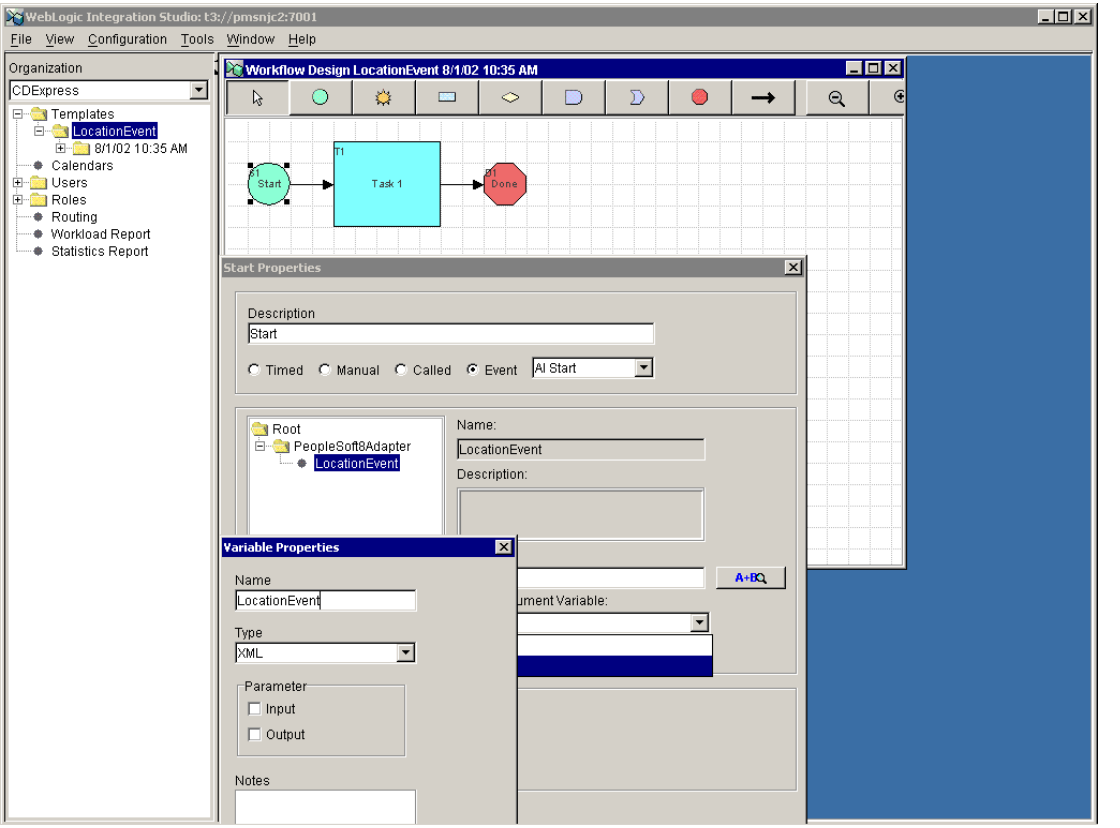
- For WebLogic Integration 7.0, see
  `http://edocs.bea.com/wli/docs70/aiuser/3usruse.htm`
- For WebLogic Integration 2.1, see
  `http://edocs.bea.com/wlintegration/v2_1sp/aiuser/3usruse.htm`

Sample Event Using a Business Process Workflow

The following screens illustrate the event using a business process workflow.

1. In the WebLogic Integration Studio, create the workflow for the event, LocationEvent.

Figure 6-24 Create Workflow for LocationEvent

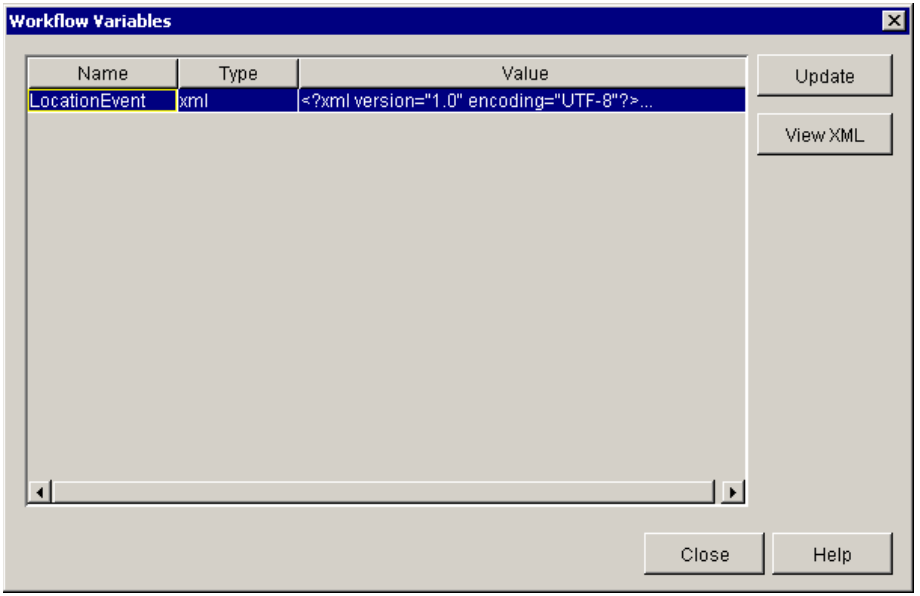


2. Establish a variable called LocationEvent, which contains the XML file that resulted from a modification to the Location table in PeopleSoft.

   There are several tasks that trigger an event in PeopleSoft, such as batch processing, doing data entry through application layering, or running a service that updates the underlying data structure. In any case, the modification triggers the messaging application process to post the PeopleSoft XML document to the WebLogic Integration variable.

3. Open the Workflow Variables dialog box.

Figure 6-25 Workflow Variables Dialog Box



4. Click View XML.

A closer look at the event variable displays the PeopleSoft XML document.

Figure 6-26 View XML Window