

WebSphere Business Integration adapters – Import and export

What this exercise is about	2
Lab requirements	2
What you should be able to do	2
Introduction	3
Description of the module.....	4
Business objects.....	4
Interfaces	5
Exercise instructions	5
Part 1: Initialize the workspace for this lab exercise	7
Part 2: Generate artifacts required for using the adapter with WebSphere Process Server	12
Instructions for PeopleSoft	12
Instructions for Siebel	19
Part 3: Configure the connector configuration files.....	22
Part 4: Assemble and test the PeopleSoftAdapter module.....	25
Part 5: Assemble and test the SiebelAdapter module	36
What you did in this exercise	42
Solution instructions	43
Task: Adding remote server to WebSphere Integration Developer test environment	44

What this exercise is about

The objective of this lab is to provide you with an understanding of how to use the WebSphere Integration Developer V6.0.2 tools to generate the code used to communicate with WebSphere Business Integration Adapters. You will also learn how to test the code using the integrated WebSphere Process Server V6.0 and the test framework.

Lab requirements

List of system and software required for the student to complete the lab.

- WebSphere Integration Developer V6.0.2 installed
- WebSphere Process Server V6.0 test environment installed
- Sample code in the directory C:\Labfiles602 (Windows®) or /tmp/LabFiles602 (Linux®)
- WebSphere Business Integration Toolset installed

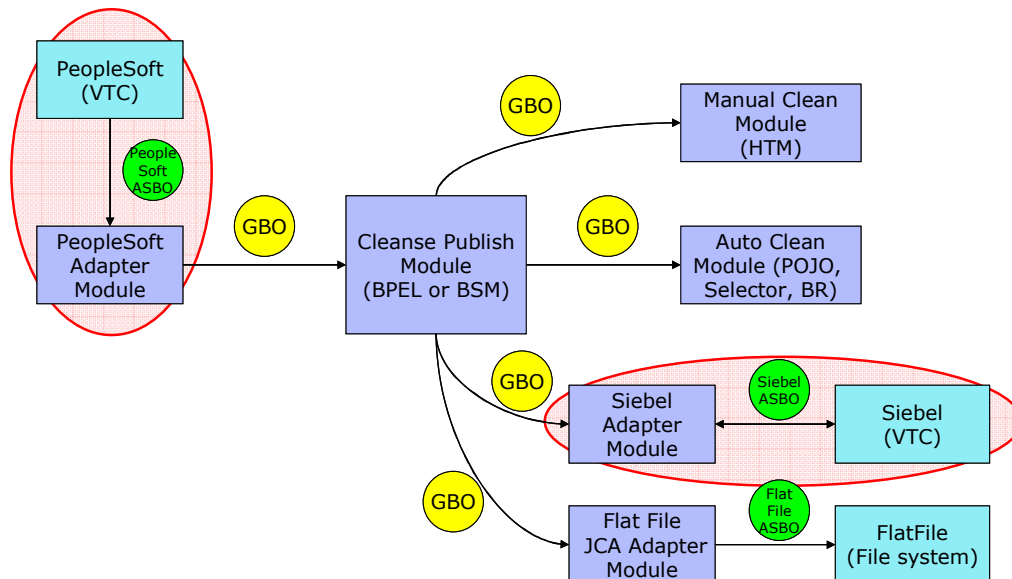
What you should be able to do

At the end of this lab you should be able to:

- Generate the artifacts required to use a WebSphere Business Integration Adapter with WebSphere Process Server
- Configure the adapter configuration files to be able to define how the adapter is to communicate with the server
- Assemble the artifacts in an SCA wiring diagram
- Use the test framework to test both the outbound and inbound use of the adapters.

Introduction

The following diagram highlights the parts of the overall scenario that will be addressed in this lab. You will work on both the PeopleSoft and Siebel modules, but will only partially implement and test each of the modules. The main focus will be converting existing Application Specific Business Objects (ASBOs) and Connector configurations for PeopleSoft and Siebel so that they can be used with the WebSphere Process Server. You will then test the ability to create a connection between the Visual Test Connector and WebSphere Process Server for sending in a PeopleSoft ASBO and for sending out a Siebel ASBO.

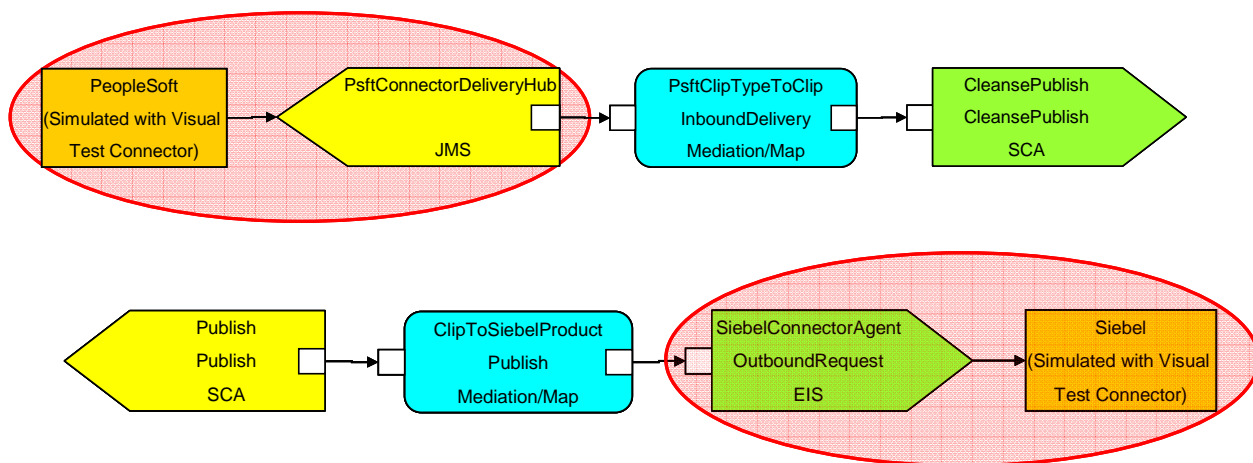


The process of converting the ASBOs and connector configuration is done using the Enterprise Service Discovery tool. It takes as input the .xsd files that define the ASBOs and the .cfg file that defines the connection between the WebSphere Business Integration Adapter and the existing broker. The following is produced as output:

- .xsd files defining the ASBOs in a format used by WebSphere Process Server
- SCA Imports and Exports (or both) specifically configured for use with this adapter and these ASBOs
- SCA Interfaces used for Inbound and Outbound communication with this adapter with these ASBOs
- A properties file used to configure WebSphere Process Server to be able to communicate with this adapter

Description of the module

The following two diagrams show the PeopleSoftAdapter and SiebelAdapter modules. They highlight the parts of the modules that will be addressed in this exercise. The mediation and maps, which are used to convert from PeopleSoft ASBOs to GBOs and from GBOs to Siebel ASBOs, will be added in a subsequent exercise.



Business objects

The following business objects will be used in this exercise.

- Psft_ClipType – The parent business object describing characteristics of a particular clip.
- Psft_RetailItem – The child business object describing a retail packing option for this type of clip
- Siebel_ProductType – The parent business object describing characteristics of a particular type of product
- Siebel_ProductPackage – The child business object describing a retail packaging option for this type of product. .

All four of these business objects will be used both in their original format and in the format required for use with WebSphere Process Server.

In addition to these business objects, there will be four business graphs generated to contain these business objects.

- Psft_ClipTypeBG
- Psft_RetailItemBG
- Siebel_ProductTypeBG
- Siebel_ProductPackageBG

Although all four of these are generated by the tool, it is only the two associated with the parent business objects that are needed. The business graphs for the child business objects are not used in the scenario.

Interfaces

The following interfaces are generated by the Enterprise Service Discovery tool.

- InboundDelivery – Generated interface used to pass in a Psft_ClipTypeBG business graph from the PeopleSoft adapter to an SCA component running in the server.
- OutboundRequest – Generated interface used to pass out a Siebel_ProductTypeBG business graph from an SCA component running in the server to the Siebel adapter.

Exercise instructions

Some instructions in this lab might be specific for Windows platforms. If you run the lab on a platform other than Windows, you will need to run the appropriate commands, and use appropriate files (for example .sh in place of .bat) for your operating system. The directory locations are specified in the lab instructions using symbolic references as follows:

Reference variable	Windows location	Linux location
<WID_HOME>	C:\Program Files\IBM\WebSphere\ID\6.0	/opt/IBM/WebSphere/ID/6.0
<WPS_HOME>	<WID_HOME>\runtimes\bi_v6	<WID_HOME>/runtimes/bi_v6
<LAB_FILES>	C:\Labfiles602	/tmp/Labfiles602
<WORKSPACE>	C:\Labfiles602\eXchange\WBIAdapters\workspace	/tmp/Labfiles602/eXchange/WBIAdapters/workspace
<TEMP>	C:\temp	/tmp
<SOLUTION>	C:\Labfiles602\eXchange\WBIAdapters\Solution	/tmp/Labfiles602/eXchange/WBIAdapater s/Solution

Windows users' note: When directory locations are passed as parameters to a Java™ program such as wsadmin, you must replace the backslashes with forward slashes to follow the Java convention. For example, C:\LabFiles60\ would be replaced by C:/LabFiles60/.

Note that the previous table is relative to where you are running WebSphere Integration Developer. The following table is related to where you are running remote test environment:

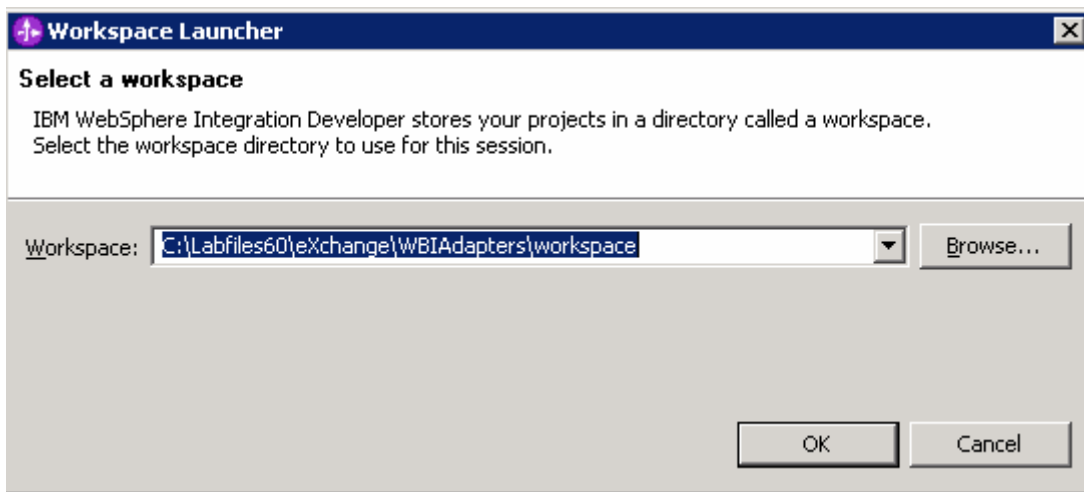
Reference Variable	Example: Remote Windows test server location	Example: Remote z/OS® test server location	Input your values for the remote location of the test server
<SERVER_NAME>	server1	cl1sr01	
<WAS_HOME>	C:\Program Files\IBM\WebSphere\AppServer	/etc/cl1cell/AppServerNode1	
<HOSTNAME>	localhost	mvsxxx.rtp.raleigh.ibm.com	
<BOOTSTRAP_PORT>	2809	2809	
<TELNET_PORT>	N/A	1023	
<PROFILE_NAME>	AppSrv01	default	
<USERID>	N/A	cl1admin	
<PASSWORD>	N/A	fr1day	

Instructions for using a remote testing environment, such as z/OS, AIX® or Solaris, can be found at the end of this document, in the section [“Task: Adding remote server to WebSphere Integration Developer test environment”](#).

Part 1: Initialize the workspace for this lab exercise

In this section of the lab, you will be importing a library project part of the CleansePublishLibrary_PI.zip project interchange file into your workspace and eventually create two modules PeopleSoftAdapter and SiebelAdapter.

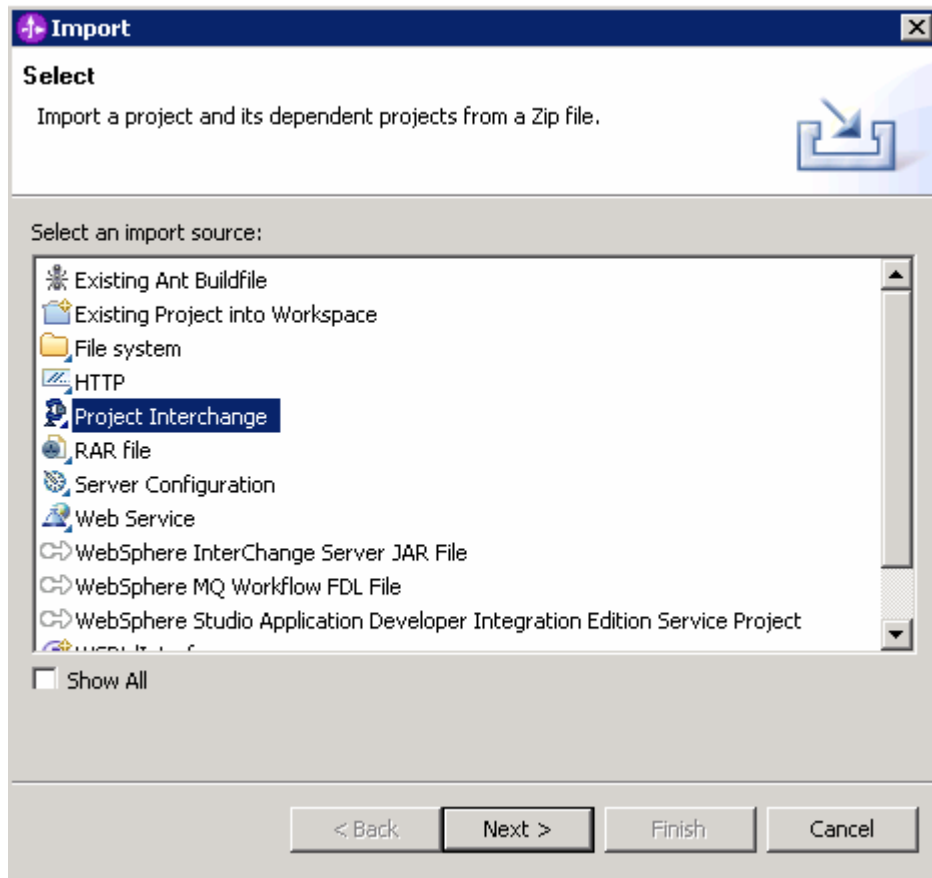
- ___ 1. Start WebSphere Integration Developer V6.0.2 with a workspace location of **<WORKSPACE>**, that is **<LAB_FILES>\eXchange\WBIAapters\workspace**
 - ___ a. From Windows Explorer, navigate to the **<WID_HOME>** directory and double click on wid.exe
 - ___ b. When prompted for workspace, enter the value provided by the **<WORKSPACE>** variable for this lab and click **OK**



- ___ c. When WebSphere Integration Developer V6.0.2 opens, click the curved arrow at top right to **go to Business Integration perspective**

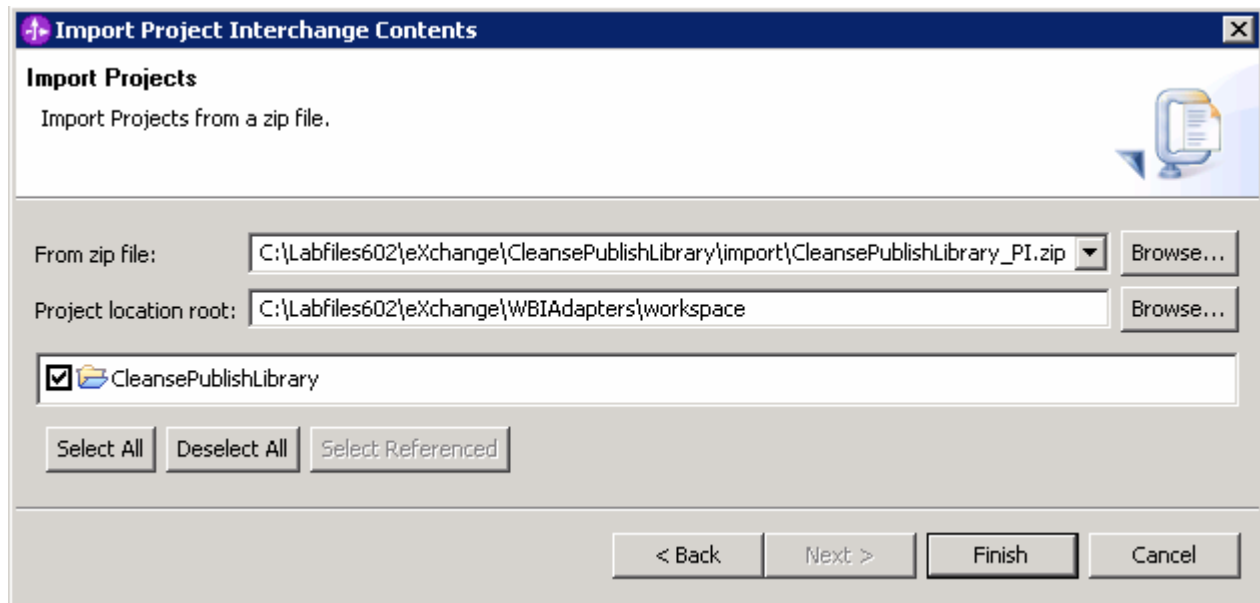


- ___ d. Ensure you are in the **Business Integration** perspective
- ___ 2. Import Project Interchange file, **CleansePublishLibrary_PI.zip** located at **<LAB_FILES>\eXchange\CleansePublishLibrary\import**
 - ___ a. Right-click inside **Business Integration View** (top left view in the Business Integration Perspective) and select **Import** from the context menu
 - ___ b. Select **Project Interchange** listed in the import dialog

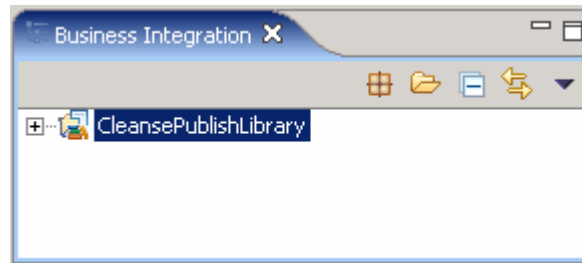


__ c. Click **Next**

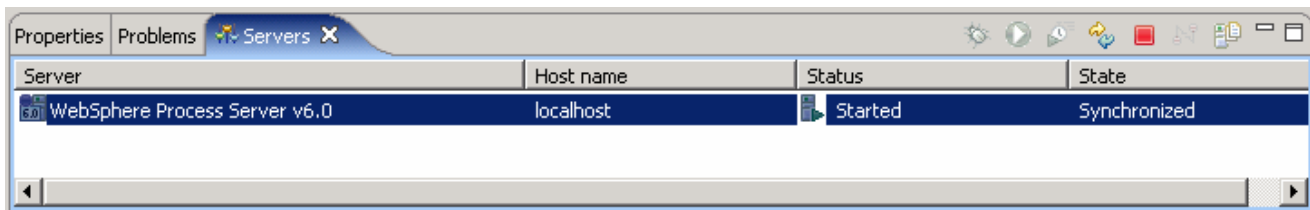
__ d. Click the **Browse** button for “**From zip file**” and navigate to **<LAB_FILES>eXchange\CleansePublishLibrary\import\CleansePublishLibrary_PI.zip** and hit **Open**



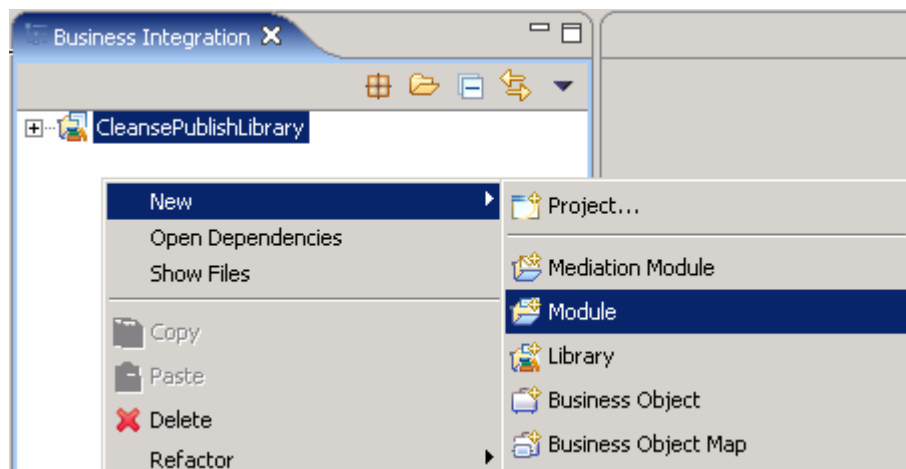
- ___ e. Select the check box next to **CleansePublishLibrary** and click **Finish**
- ___ f. Verify you have **CleansePublishLibrary** module listed in the Business Integration view



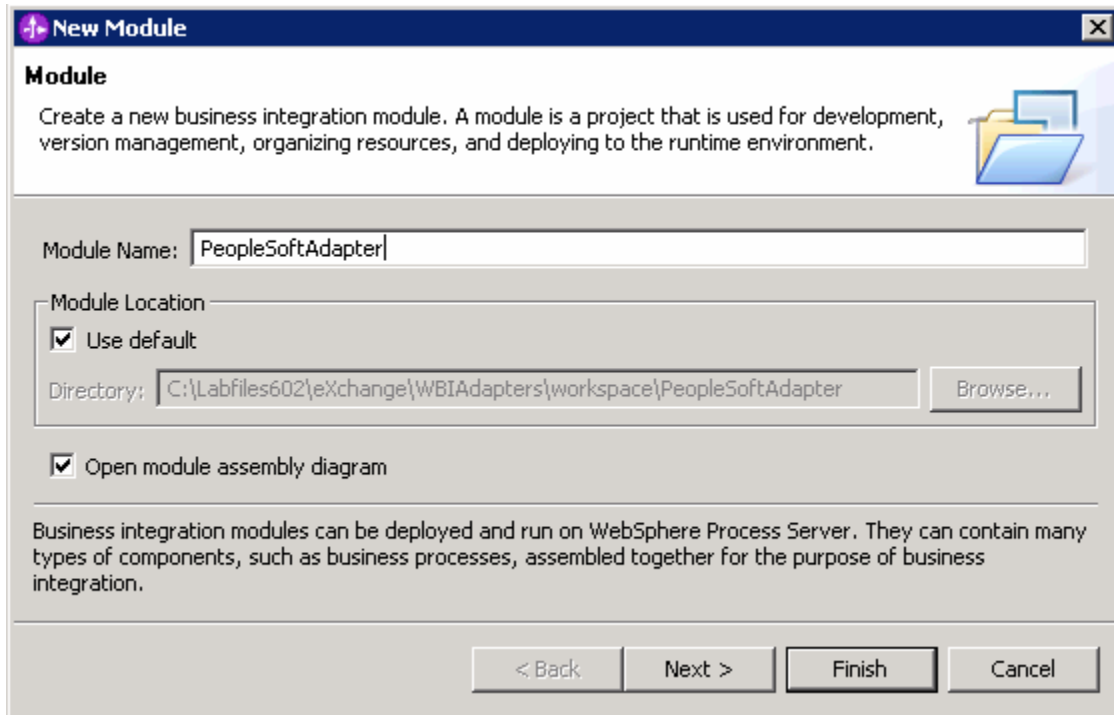
- ___ g. Verify you have WebSphere Process Server V6.0 listed in your Servers view



- ___ 3. Create a new module named, **PeopleSoftAdapter**
 - ___ a. Right-click in Business Integration view and select **New → Module** from the context menu

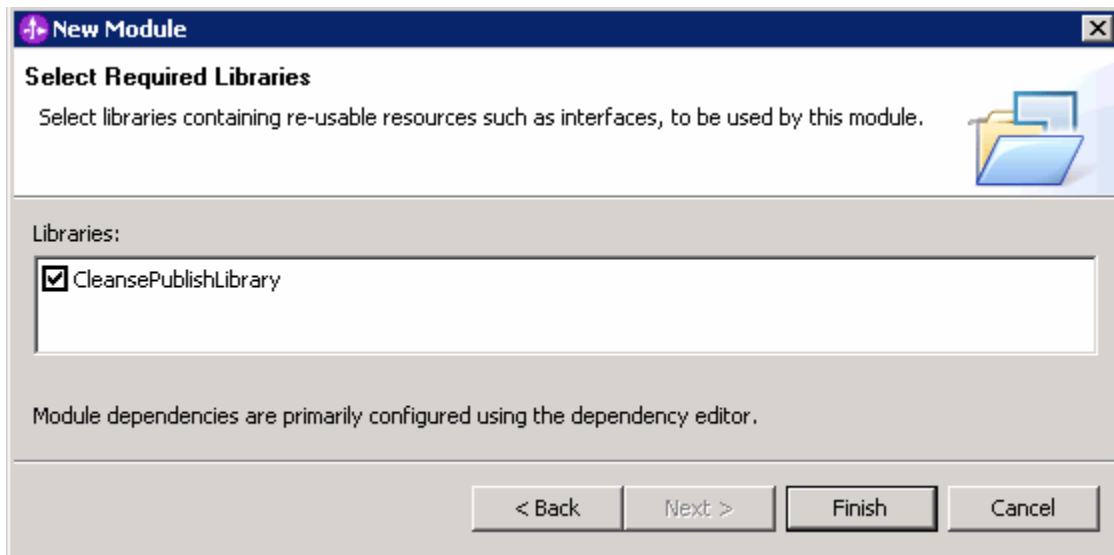


- ___ b. Enter the Module Name as **PeopleSoftAdapter**



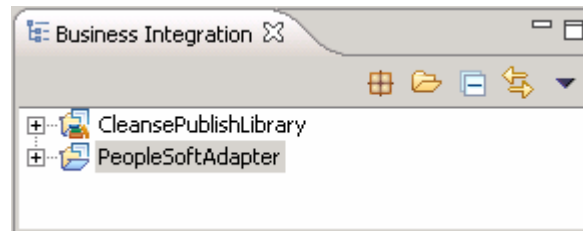
__ c. Click **Next**

__ d. In the following panel, select the check box next to **CleansePublishLibrary**. This adds the CleansePublishLibrary as a dependant library for the ManualClean module

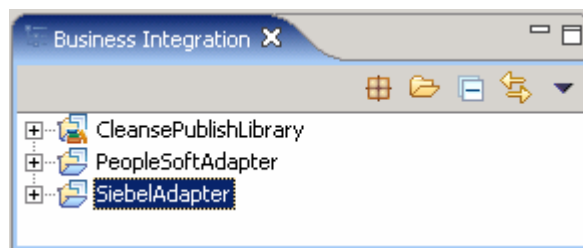


__ e. Click **Finish**

__ f. Verify you have **PeopleSoftAdapter** module listed in the Business Integration view



- ___ 4. Create a another module named, **SiebelAdapter**
 - ___ a. Right-click in Business Integration view and select **New → Module** from the context menu
 - ___ b. Enter the Module Name as **SiebelAdapter**
 - ___ c. Click **Next**
 - ___ d. In the following panel, select the check box next to **CleansePublishLibrary**. This adds the CleansePublishLibrary as a dependant library for the ManualClean module
 - ___ e. Click **Finish**
 - ___ f. Verify you have **SiebelAdapter** module listed in the Business Integration view



Part 2: Generate artifacts required for using the adapter with WebSphere Process Server

In this part, you will convert the PeopleSoft and Siebel business objects and configuration from the format used with WebSphere InterChange Server to the format used by the WebSphere Process Server. This must be done for any adapter configuration and business objects when you plan to continue to use the WebSphere Business Integration Adapter with the EIS application but will start using the WebSphere Process Server as the broker.

The process is done using Enterprise Service Discovery which is provided in WebSphere Integration Developer. It takes the .xsd files, which define the ASBOs, and the .cfg file, which defines the connection between the WebSphere Business Integration Adapter and the existing broker, as input. The following is produced as output:

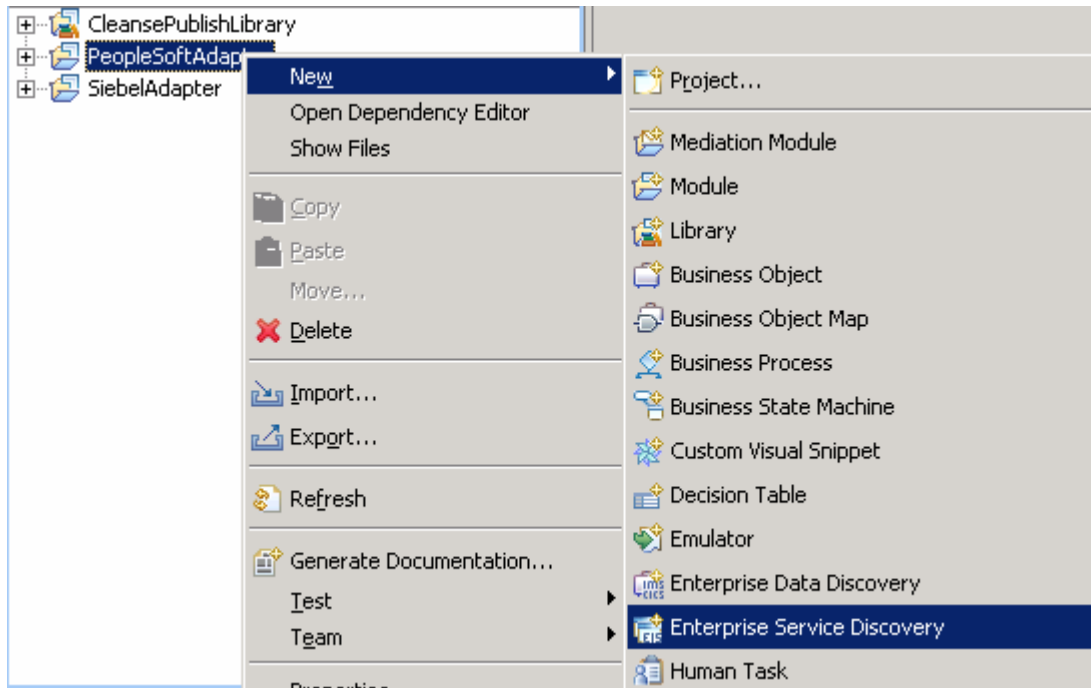
- .xsd files defining the ASBOs in a format used by WebSphere Process Server
- SCA Imports or Exports(or both) specifically configured for use with this adapter and these ASBOs
- SCA Interfaces used for Inbound and Outbound communication with this adapter with these ASBOs
- A properties file used to configure WebSphere Process Server to be able to communicate with this adapter

In this lab you will generate and test the artifacts for the PeopleSoft adapter and then repeat the process for the Siebel adapter. The following steps will lead you through the process.

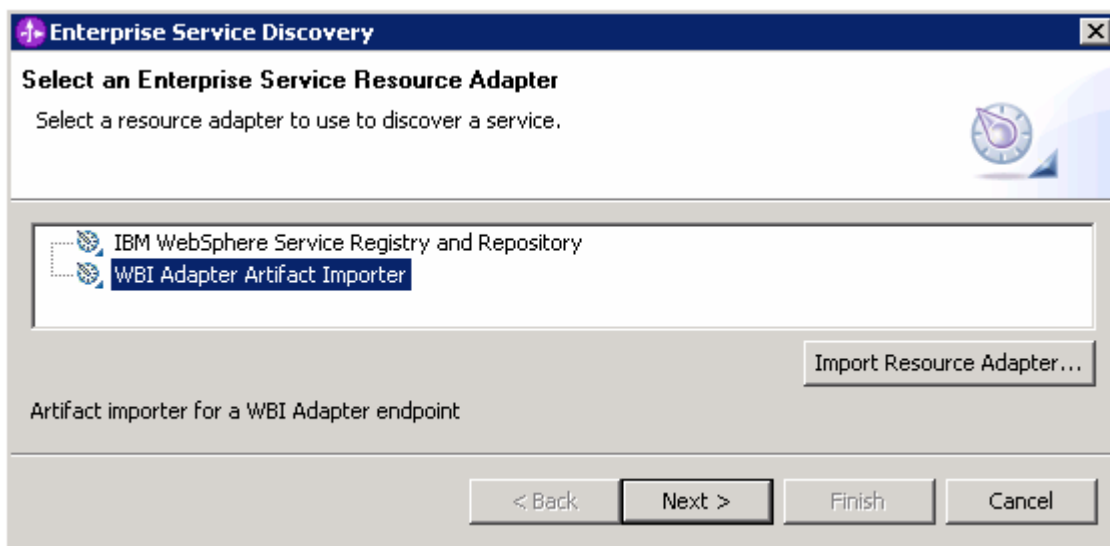
Instructions for PeopleSoft

1. Access the Enterprise Service Discovery wizard to generate the artifacts that will be used with the adapter for PeopleSoft.
 - a. Right click on the **PeopleSoftAdapter** module in the Business Integration perspective.

__ b. Select **New** → **Enterprise Service Discovery** from the context menu



__ c. From the Enterprise Service Resource Adapter panel, select **WBI Adapter Artifact Importer**



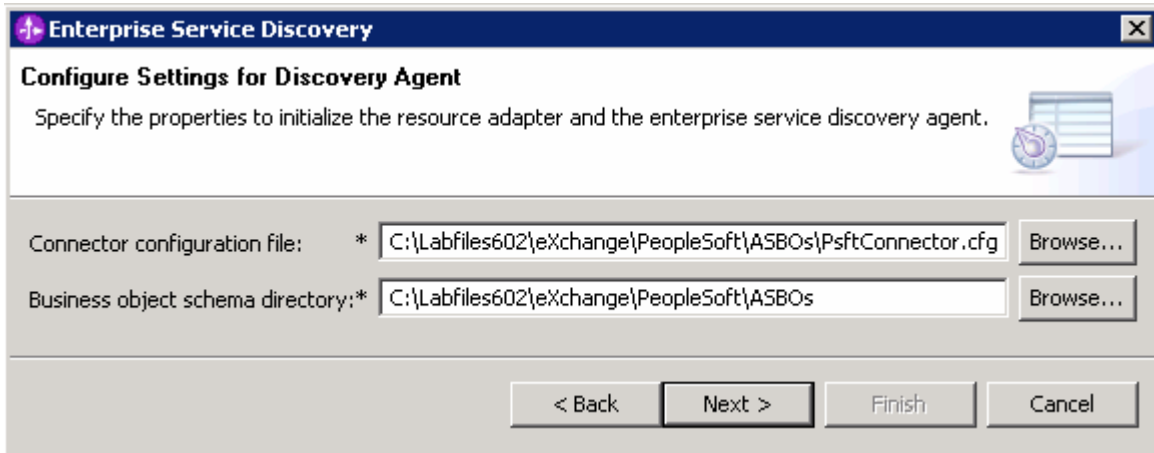
__ d. Click **Next**

___ 2. In the following panel, identify the specific configuration and location of the business objects to be discovered

__ a. In the **Configure Settings for Discovery Agent** panel,

- 1) Click the Browse for Connector configuration file and locate PsftConnector.cfg file at **<LAB_FILES>\eXchange\PeopleSoft\ASBOs\PsfConnector.cfg**

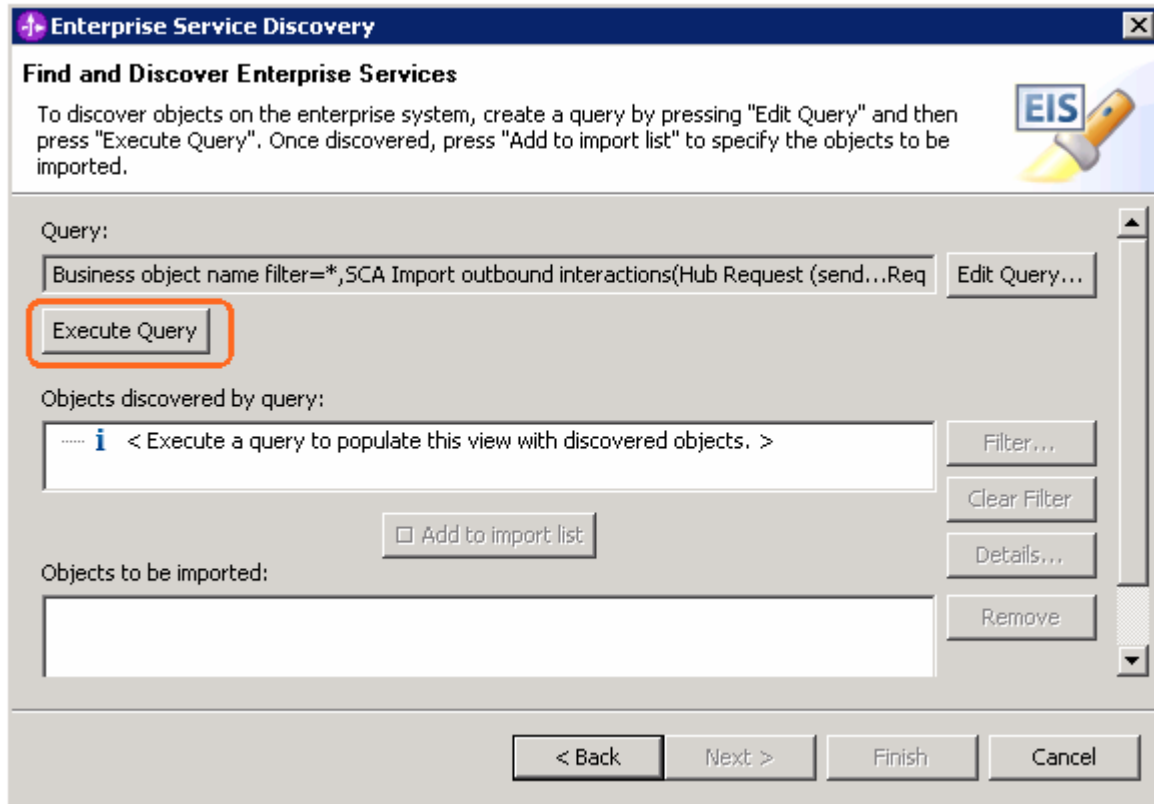
- 2) Click the **Browse** button for Business object schema directory locate the directory named **ASBOs** at <LAB_FILES>\eXchange\PeopleSoft\ASBOs



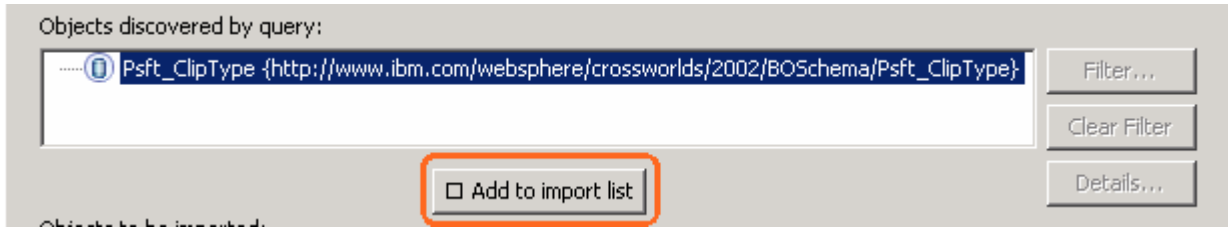
___ b. Click **Next**

- ___ 3. Pick the specific objects for PeopleSoft to be processed

___ a. In the following **Find and Discover Enterprise Services** panel, click the **Run Query** button on to discover the PeopleSoft objects.



___ b. Select the business objects you want to process listed in the **Objects discovered by query** text area and click the **Add to import list** button

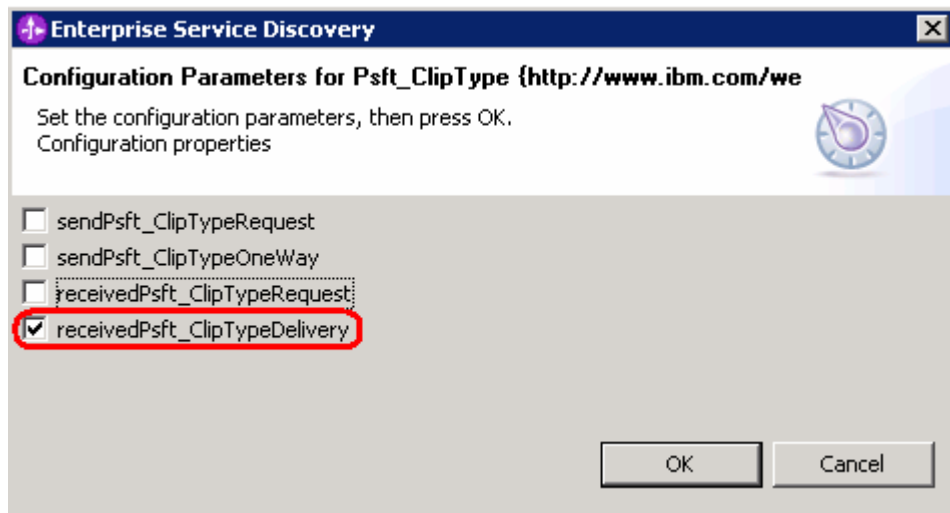


__ c. The Computation Parameters Psft_ClipType dialog pops-up

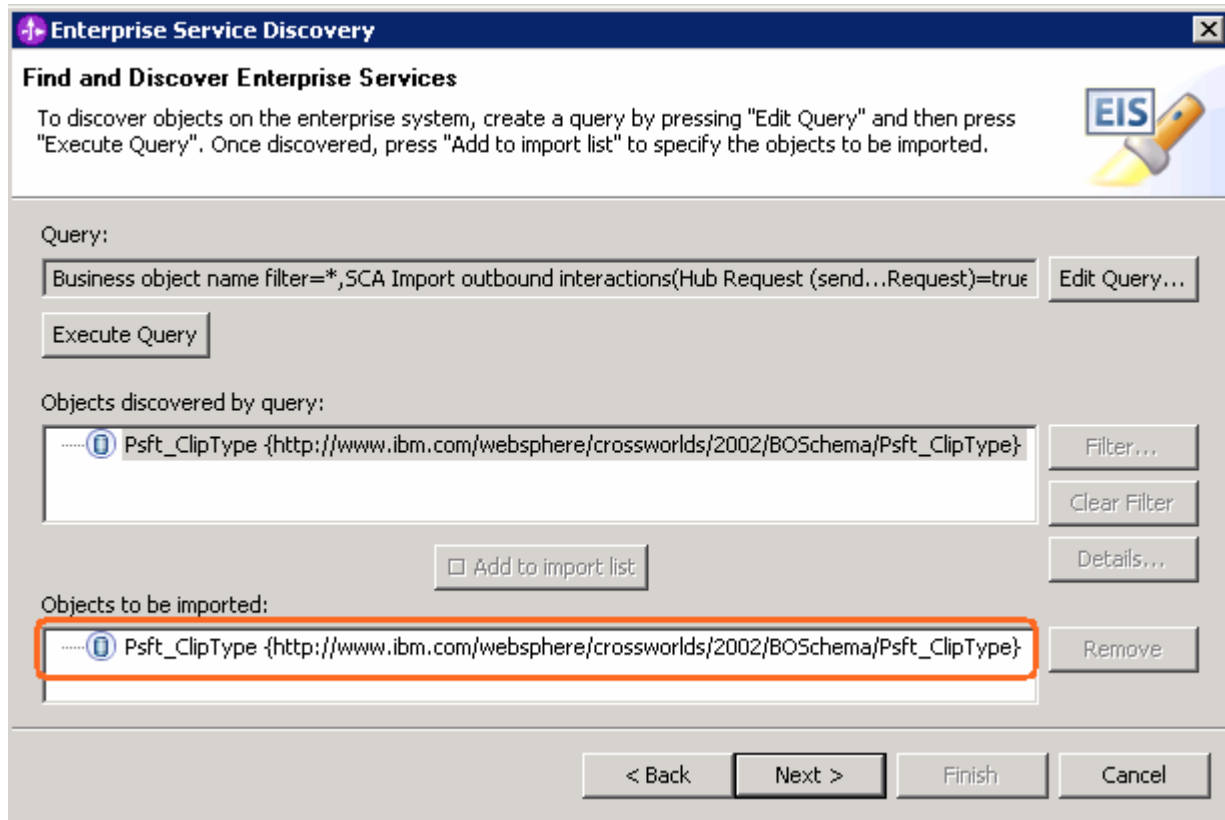
___ 4. In the Configuration Parameters for Psft_ClipType dialog, select the kinds of interactions you will need between the adapter and the server. This will determine which interfaces will be generated for the selected business objects. The choices you have are:

- sendPsft_ClipTypeRequest – outbound requests from server to adapter with response
- sendPsft_ClipTypeOneWay - outbound requests from server to adapter without response
- receivedPsft_ClipTypeRequest – inbound requests from adapter to server with response
- receivedPsft_ClipTypeDelivery – inbound requests from adapter to server without response

___ a. In this lab scenario, you only need inbound requests from adapter to server without a response. Therefore, ensure that only the check box next to **receivedPsft_ClipTypeDelivery** is checked, as shown below:



___ b. Click **OK** and you will see the object you had imported populated in the **Objects to be imported** text area as shown below:



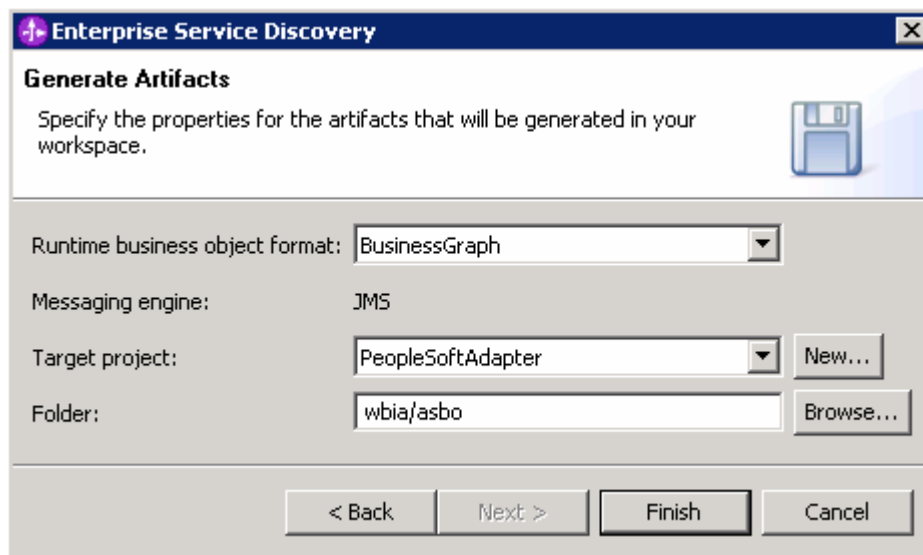
__ c. Click **Next**

___ 5. Save the generated artifacts. In the following Generated Artifacts panel, provide the following information:

__ a. Ensure Runtime business object format is set to **BusinessGraph**

__ b. Ensure the Target project is set to **PeopleSoftAdapter**

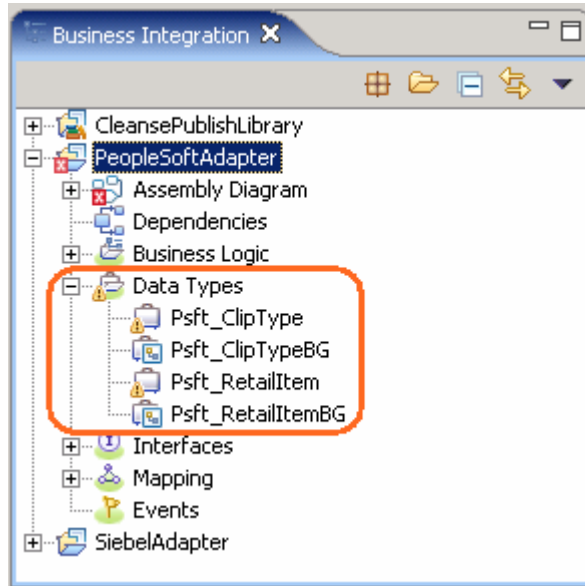
__ c. Enter **wbia/asbo** for folder



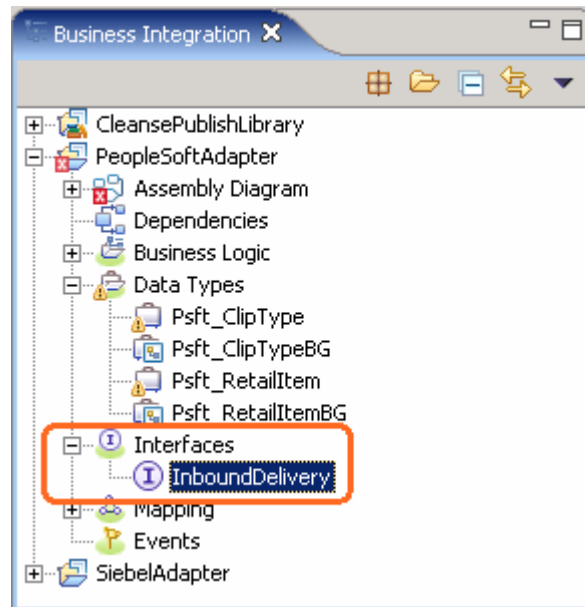
___ d. Click **Finish**

___ 6. Examine the generated artifacts by expanding the **PeopleSoftAdapter** module tree in the Business Integration view.

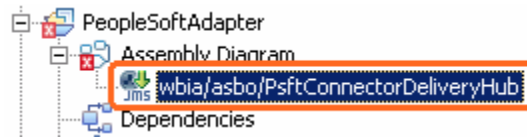
___ a. Expand **PeopleSoftAdapter** → **Data Types**. You will see four new business objects that provide the WebSphere Process Server form of the two business objects which were defined for use with the PeopleSoft adapter. For each of the adapters business objects, there is a business object generated with the same fields and a business graph object that wrappers it. In this scenario there is no need for the Psft_RetailItemBG because the Psft_RetailItem is contained within Psft_ClipType and therefore does not require a business graph when passed across an interface



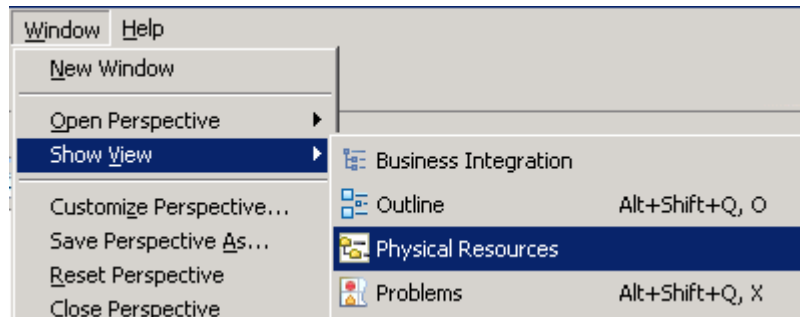
___ b. Now expand **PeopleSoftAdapter** → **Interfaces**. You will see the InboundDelivery interface. This is the interface through which the Psft_ClipTypeBG will be passed into the server



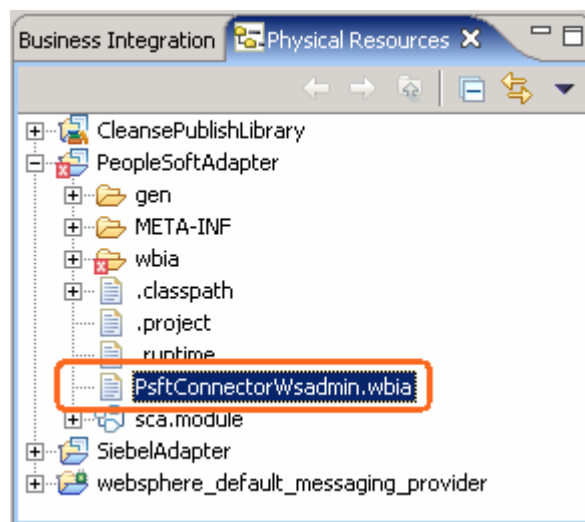
- ___ c. Expand **PeopleSoftAdapter** → **Assembly Diagram**. You will see the Export that is used to define the interface and JMS data binding, which will be used by the adapter to communicate with the server. You will notice that the **PeopleSoftAdapter** assembly is marked with an error as the export is not yet wired to any SCA component, which is an invalid state for an Export. Later in this exercise you will wire this Export to a component that supports the InboundDelivery interface.



- ___ 7. Examine the properties file that was generated for you. The system will later use this file to configure the server so that it can communicate with this adapter.
 - ___ a. From the main menu, obtain the Physical Resources view by selecting **Window** → **Show View** → **Physical Resources**. Click **OK** on the warning dialog



- ___ b. In the Physical Resources view, expand **PeopleSoftAdapter**. Ensure the properties file, **PsftConnectorWsadmin.wbia** you are looking for is listed. This contains information about the queues that are needed to configure the server to communicate with the adapter.

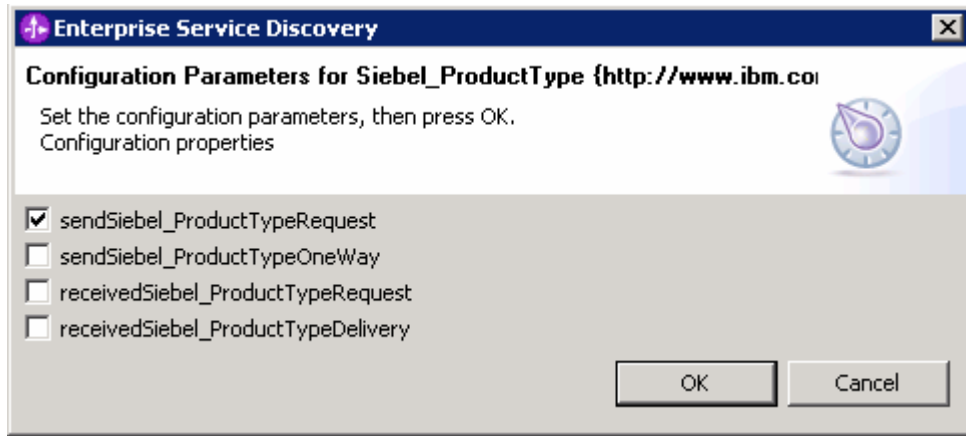


- ___ c. If you would like to examine the property definitions, double-click on **PsftConnectorWsadmin.wbia** and the file will be opened in the text editor.
- ___ 8. Now return to the Business Integration view in the upper left panel by clicking on the **Business Integration** tab at the top of the panel.

Instructions for Siebel

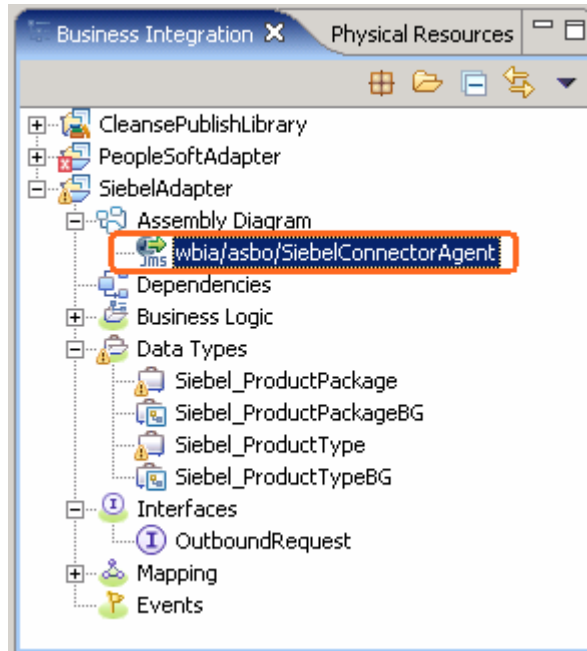
In this section you will perform essentially the same steps that were done for PeopleSoft. The entire step by step instructions will be provided, but without the extensive use of screen captures.

- ___ 1. Access the Enterprise Service Discovery wizard to generate the artifacts that will be used with the adapter for Siebel.
 - ___ a. Right click on the **PeopleSoftAdapter** module in the Business Integration perspective.
 - ___ b. Select **New → Enterprise Service Discovery** from the context menu
- ___ 2. In the **Enterprise Service Discovery** panel, identify the specific configuration and location of the business objects to be discovered.
 - ___ a. Select **WBI Adapter Artifact Importer** in the **Select an Enterprise Service Resource Adapter** panel
 - ___ b. Click **Next**
 - ___ c. In the following **Configure Settings for Discovery Agent** panel,
 - 1) Click the Browse for Connector configuration file and locate PsftConnector.cfg file at **<LAB_FILES>\eXchange\Siebel\ASBOs\SiebelConnector.cfg**
 - 2) Click the **Browse** button for Business object schema directory locate the directory named **<LAB_FILES>\eXchange\Siebel\ASBOs**
 - ___ d. Click **Next**
- ___ 3. Pick the specific objects for Siebel to be processed.
 - ___ a. Click the **Run Query** button to discover the Siebel objects on the **Find and Discover Enterprise Services** dialog.
 - ___ b. Select the business objects that you want to process in the **Objects discovered by query** text area.
 - ___ c. Click the **Add to import list** button and you will be presented with the Configuration Parameters for Siebel_ProductType dialog.
- ___ 4. In the Configuration Parameters for Siebel_ProductType dialog, you select the kinds of interactions you will need between the adapter and the server. This will determine which interfaces will be generated for the selected business objects. The choices you have are:
 - sendSiebel_ProductTypeRequest – outbound requests from server to adapter with response
 - sendSiebel_ProductTyeOneWay - outbound requests from server to adapter without response
 - receivedSiebel_ProductTypeRequest – inbound requests from adapter to server with response
 - receivedSiebel_ProductTypeDelivery – inbound requests from adapter to server without response
 - ___ a. In this lab scenario, only outbound requests from server to adapter with a response are needed. Therefore, ensure that only the check box next to **sendSiebel_ProductTypeRequest** is selected.



- ___ b. Click **OK**
- ___ c. Click **Next**
- ___ 5. Save the generated artifacts. In the following Generated Artifacts panel, provide the following information:
 - ___ a. Ensure Runtime business object format is set to **BusinessGraph**
 - ___ b. Ensure the Target project is set to **SiebelAdapter**
 - ___ c. Enter the name of the folder as **wbia/asbo**
 - ___ d. Click **Finish**
- ___ 6. Examine the artifacts that were generated for you by expanding the **SiebelAdapter** module tree in the Business Integration view
 - ___ a. Expand **SiebelAdapter** → **Data Types** and examine the generated business objects

- ___ b. Expand **SiebelAdapter** → **Interfaces**. You will see the **OutboundRequest** interface. This is the interface through which the Siebel_ProductTypeBG will be passed out to the adapter
- ___ c. Expand **SiebelAdapter** → **Assembly Diagram**. You will see the Import that defines the interface and the JMS data binding which will be used by the server to communicate with the adapter

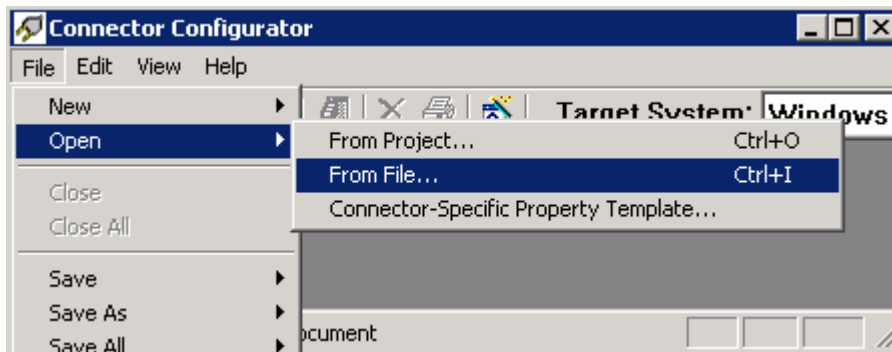


- ___ 7. Examine the properties file that was generated for you. The system will later use this file to configure the server so that it can communicate with this adapter
 - ___ a. Obtain the Physical Resources view in the upper left panel by clicking on the **Physical Resources** tab at the top of the panel
 - ___ b. In the Physical Resources view, expand **SiebelAdapter**. Ensure the properties file, **SiebelConnectorWsadmin.wbia** you are looking for is listed. This contains information about the queues that are needed to configure the server
 - ___ c. Double click on **SiebelConnectorWsadmin.wbia** if you want to examine the definitions
- ___ 8. Return to the Business Integration view in the upper left panel by clicking on the **Business Integration** tab at the top of the panel
- ___ 9. Save all work by **File** → **Save All** or **Ctrl + Shift + S**

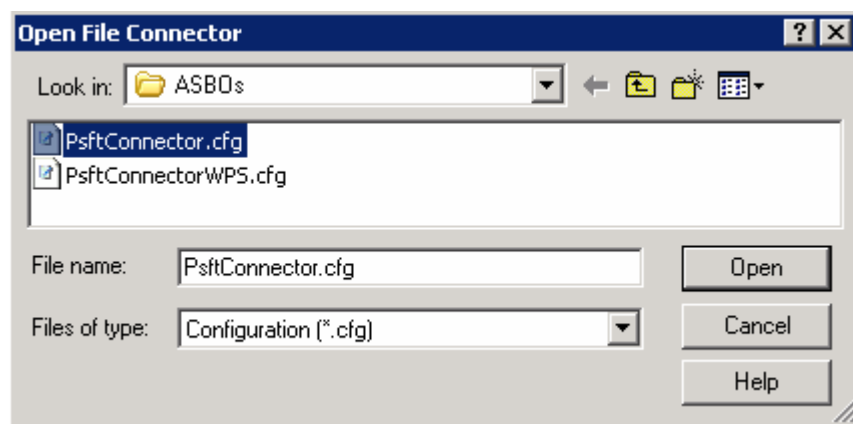
Part 3: Configure the connector configuration files

The PeopleSoft and Siebel adapters make use of a connector configuration file to define the configuration of the broker they will be communicating with. You already have the connector configuration files that were defined for use with a broker other than WebSphere Process Server. You used these configuration files when you ran the Enterprise Service Discovery earlier in this exercise as it provided information that was used to create the properties file that is used to configure WebSphere Process Server. There are some modifications that must be made to these files in order for the adapters to communicate with WebSphere Process Server. In this part you will examine these modifications. The modified connector configuration will be used in later parts to allow the Visual Test Connector to communicate with WebSphere Process Server to simulate the PeopleSoft and Siebel adapters.

- ___ 1. Start the Connector Configurator (this could vary for different versions of Windows – the instructions provided are consistent with Windows XP)
 - ___ a. From the start menu, select **All Programs → IBM WebSphere Business Integration Adapters → IBM WebSphere Business Integration Toolset → Connector Configurator**
- ___ 2. Open the original connector configuration file for PeopleSoft which was used with the previous broker
 - ___ a. Select **File → Open → From File...** in the Connector Configurator

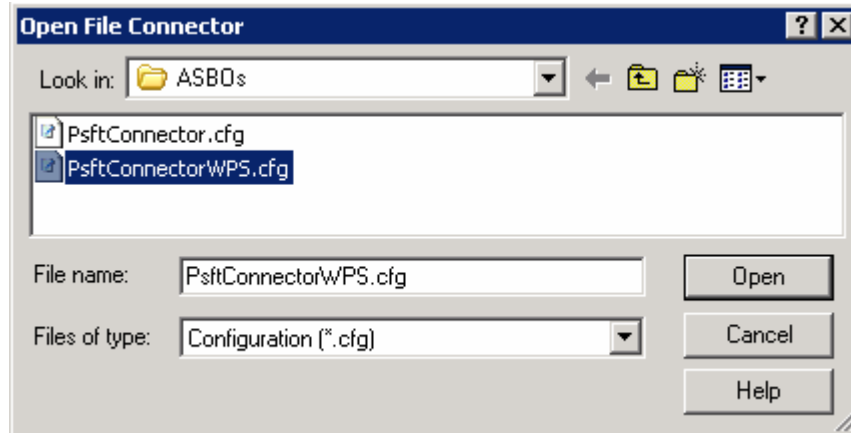


- ___ b. The Open File Connector pops-up. Navigate to **<LAB_FILES>\eXchange\PeopleSoft\ASBOs** and select **PsftConnector.cfg** file

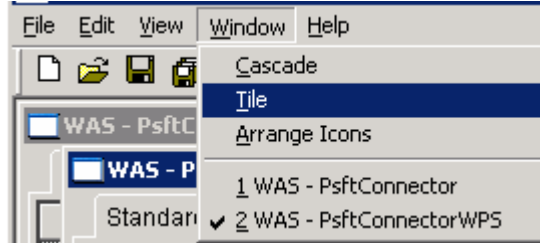


- ___ c. Click **Open**

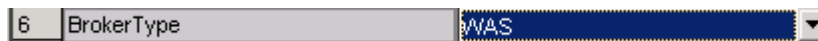
- ___ 3. Now open the connector configuration files for PeopleSoft which will be used when communicating with WebSphere Process Server
 - ___ a. Select **File → Open → From File...** in the Connector Configurator
 - ___ b. When the Open File Connector pops-up, navigate to **<LAB_FILES>\eXchange\PeopleSoft\ASBOs** and select **PsftConnectorWPS.cfg** file



- ___ c. Click **Open**
- ___ 4. Arrange the two connector configuration files that you opened, so they can be viewed together
 - ___ a. From the main menu, select **Window → Tile** in the Connector Configurator



- ___ 5. Examine the key fields in the Connector Configuration files to see which values needed to be updated based on your environment so that the adapter can connect to WebSphere Process Server
 - ___ a. **BrokerType** – The broker type must be set to **WAS**



- ___ b. **jms.MessageBrokerName** – The name of the message broker must be **WBIA_QM:WBIA.JMS.SVRCONN:<HOSTNAME>:5558**



Note: **localhost: 5558** assumes that (1) the adapter and WebSphere Process Server are running on the same machine and (2) the server has been configured with the default port assignments. These must be the WebSphere Process Server host name and the configured SIB_MQ_ENDPOINT_ADDRESS.

- ___ c. **jms.UserName** and **jms.Password** – This is the user name and password used to authenticate for JMS communication with WebSphere Process Server

17	jms.Password	*****
18	jms.UserName	anything

Note: In this case, WebSphere Process Server is not secured and therefore you do not need a valid username and password. However, the values cannot be left blank or errors will occur.


- ___ d. **RepositoryDirectory** – The repository directory must point to where the business objects used by the adapter are located, in this case **<LAB_FILES>\eXchange\PeopleSoft\ASBOs**

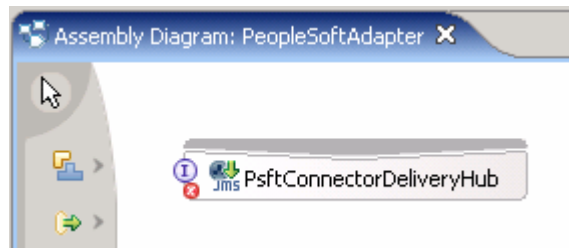
24	RepositoryDirectory	C:\Labfiles602\eXchange\PeopleSoft\ASBOs
----	---------------------	--

- ___ 6. If updated the connector files, from the main menu select **File → Save All → To File** to save the changes
- ___ 7. Now open the connector configuration files for Siebel which will be used when communicating with WebSphere Process Server
 - ___ a. Open the following two files in the Connector Configurator:
 - <LAB_FILES>\eXchange\Siebel\ASBOs\SiebelConnector.cfg
 - <LAB_FILES>\eXchange\Siebel\ASBOs\SiebelConnectorWPS.cfg
 - ___ b. Compare the values for the following fields
 - BrokerType: WAS
 - jms.MessageBrokerName: WBIA_QM:WBIA.JMS.SVRCONN:<HOSTNAME>:5558
 - jms.Password: *****
 - jms.UserName: anything
 - RepositoryDirectory : <LAB_FILES>\eXchange\Seibel\ASBOs
 - ___ c. Save the changes
- ___ 8. Close the Connector Configurator window
 - ___ a. From the main menu, select **File → Exit**

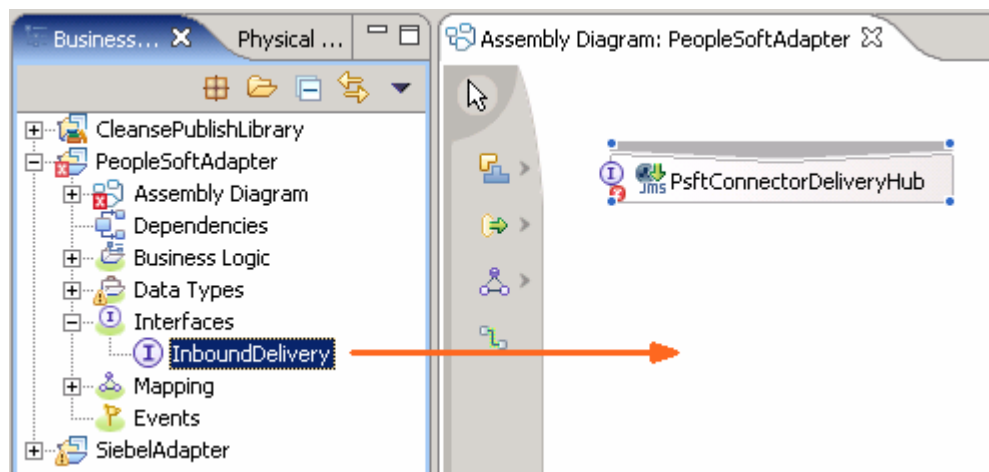
Part 4: Assemble and test the PeopleSoftAdapter module

In this part you will open the Assembly Editor for the PeopleSoftAdapter module. Using the Component Test capability in WebSphere Integration Developer and the Visual Test Connector, you will test the Export to ensure it can be used to receive a business object from the adapter.

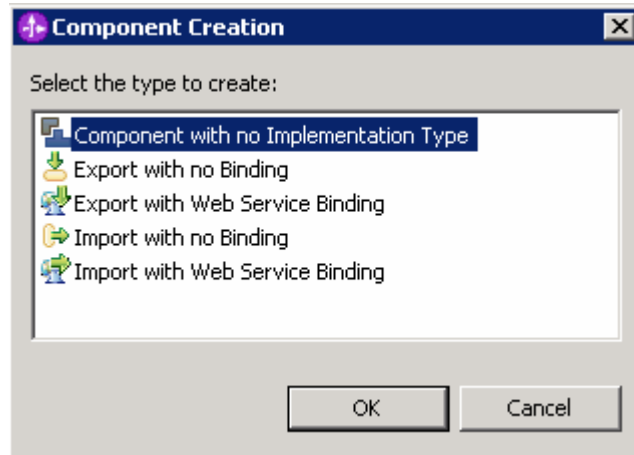
- ___ 1. Open the PeopleSoftAdapter in the Assembly Editor
 - ___ a. Expand **PeopleSoftAdapter** in the Business Integration view and double click on **Assembly Diagram** ( **Assembly Diagram**) to open it in an Assembly Diagram editor
 - ___ b. In the Assembly Diagram Editor, you will see a message that there is a new element in the module and see an error indicator at the lower left of the PsftConnectorDeliveryHub. This is because the PsftConnectorDeliveryHub does not have anything wired to it to send output. You will fix this in subsequent steps



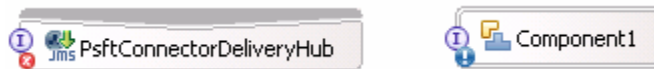
- ___ 2. The Test Component cannot examine the data coming in through an Export unless the Export is wired to something. Therefore, you need to create a component that the Export can be wired to. To do this, the component must support the same interface as the Export. However, the component does not need to be implemented and the test component will emulate its operation. This setup enables you to examine the business graph sent in from the Visual Text Connector through the Export.
 - ___ a. In the Business Integration view, expand **PeopleSoftAdapter** → **Interfaces**
 - ___ b. Drag the **InboundDelivery** interface to the assembly editor canvas



- ___ c. On releasing the mouse button to drop it on the assembly editor, you will see the Component Creation dialog pop-up



- ___ d. From the Component Creation dialog, select **Component with No Implementation Type** and click **OK**. The assembly editor should now look similar to the following screen capture:



- ___ e. Wire the Export to the unimplemented Component

- 1) Click on the Export **PsftConnectorDeliveryHub** and drag a wire to **Component1**
- 2) Your Assembly Editor should now look something like the picture below:

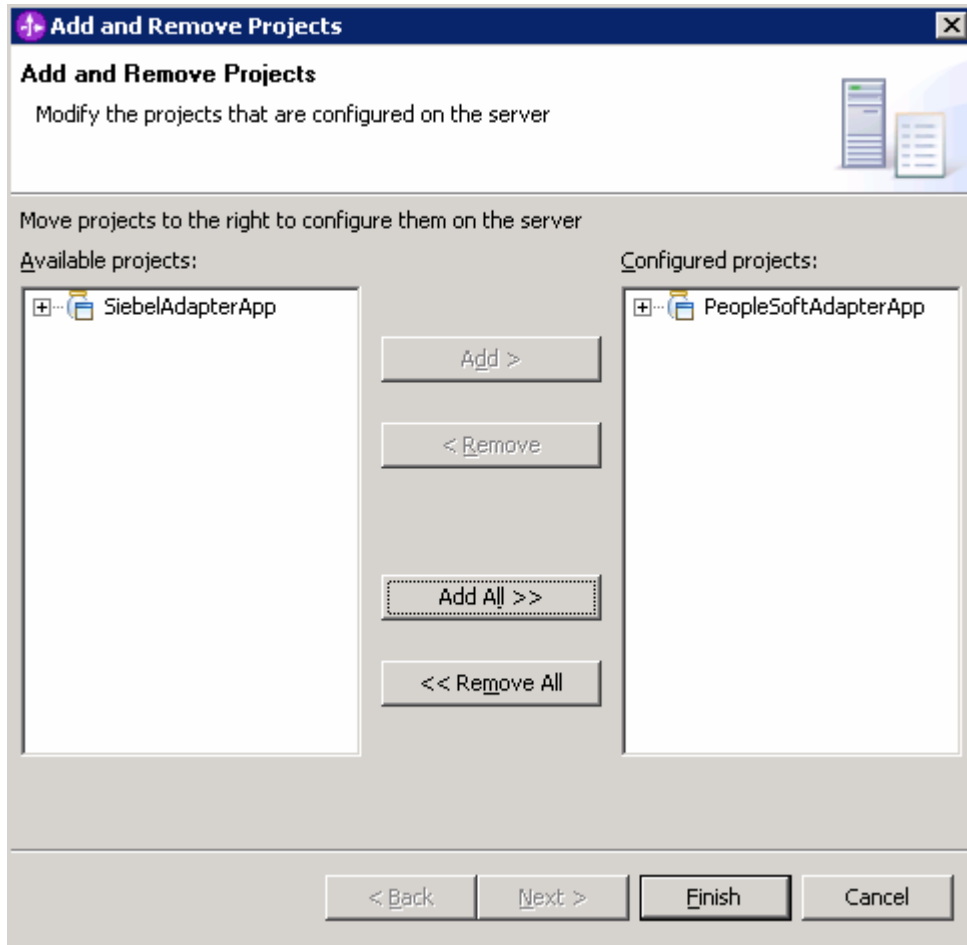


- ___ f. Save all work by **File → Save All** or **Crtl + Shift + S**

Note: There must not be any errors reflected in the Problems view at this time.

- ___ 3. Add the PeopleSoftAdapter module to the WebSphere Process Server Test Environment
 - ___ a. From the Server view, right click on WebSphere Process Server V6.0 and select **Add and Remove Projects...** from the context menu
 - ___ b. In the Add and Remove Projects dialog, select only the **PeopleSoftAdapterApp** project from the Available projects panel

- ___ c. Click **Add >** to add it to the Configured projects panel



- ___ d. Click **Finish**. If using a local testing environment, the server automatically starts if it is not started at this time

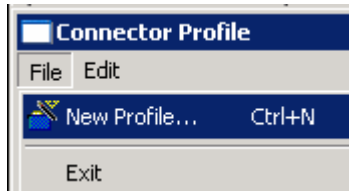
If using a remote testing environment, follow the directions provided in [Task: Adding Remote Server to WebSphere Integration Developer Test Environment](#) (at the end of this document) to restart a server

- ___ 4. Connect the Visual Test Connector (from now on VTC) to the WebSphere Process Server test environment. Using the connector configuration at **<LAB_FILES>\eXchange\PeopleSoft\ASBOs\PstfConnectorWPS.cfg** follow the instructions below to use the Visual Test Connector:
 - ___ a. Ensure the WebSphere Process Server is active (the server is the JMS Broker). The VTC requires an active JMS Broker to start its connection
 - ___ b. Start the Visual Test Connector
 - ___ c. From the **start** menu, select **Programs → IBM WebSphere Business Integration Adapters → IBM WebSphere Business Integration Toolset → Visual Test Connector**. This opens a command window in the background where the process runs and the Visual Test Connector Interface in the foreground. Do not close the command window as this will cause the VTC to exit

___ d. From the main menu of the VTC interface, select **Create/Select a Profile**. The Connector Profile panel opens

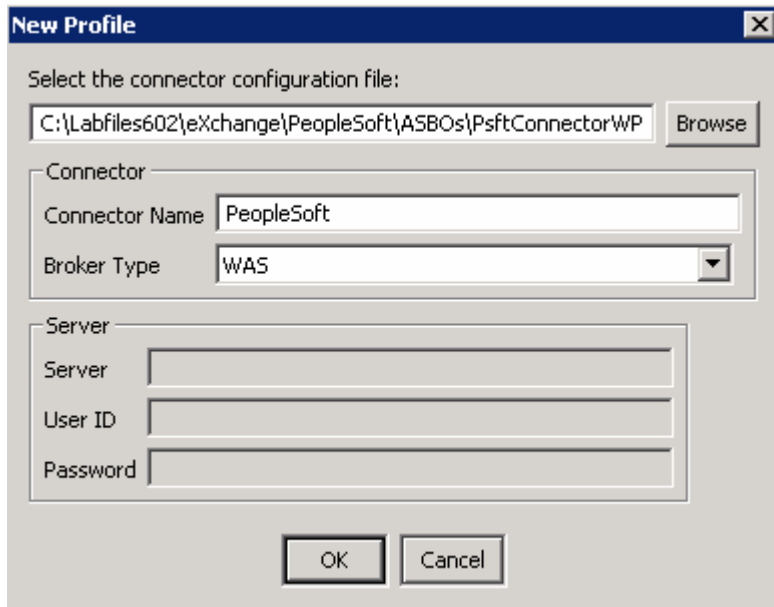


___ e. Create a new PeopleSoft profile by selecting **File → New Profile...** from the main menu of the Connector Profile panel



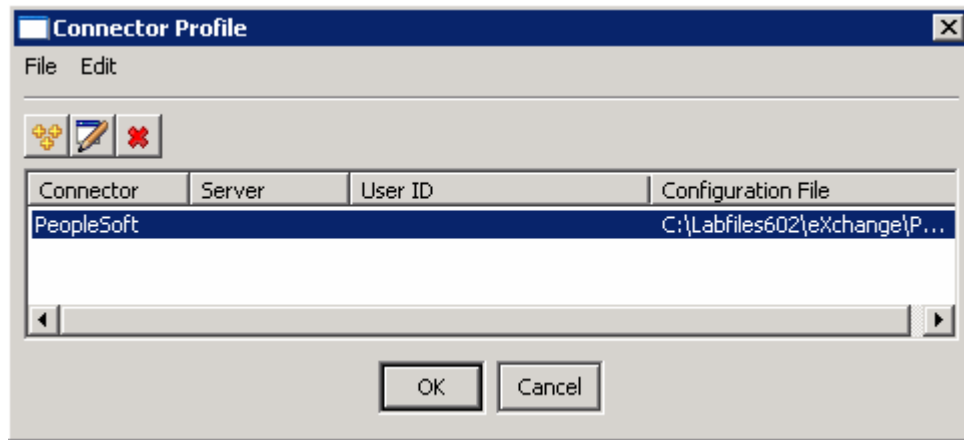
___ f. Enter the following information in the New Profile panel:

- 1) Click the **Browse** button for “Select the connector configuration file:” field and select **<LAB_FILES>\eXchange\PeopleSoft\ASBOs\PsfConnectorWPS.cfg** file
- 2) Connector Name : **PeopleSoft**
- 3) Broker Type : **WAS**



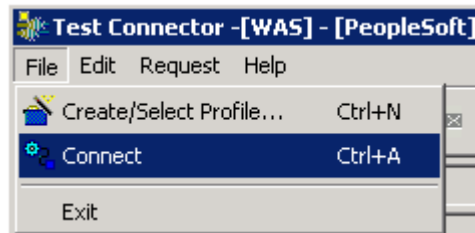
___ g. Click **OK**

___ h. Select the PeopleSoft profile you had created as shown below:

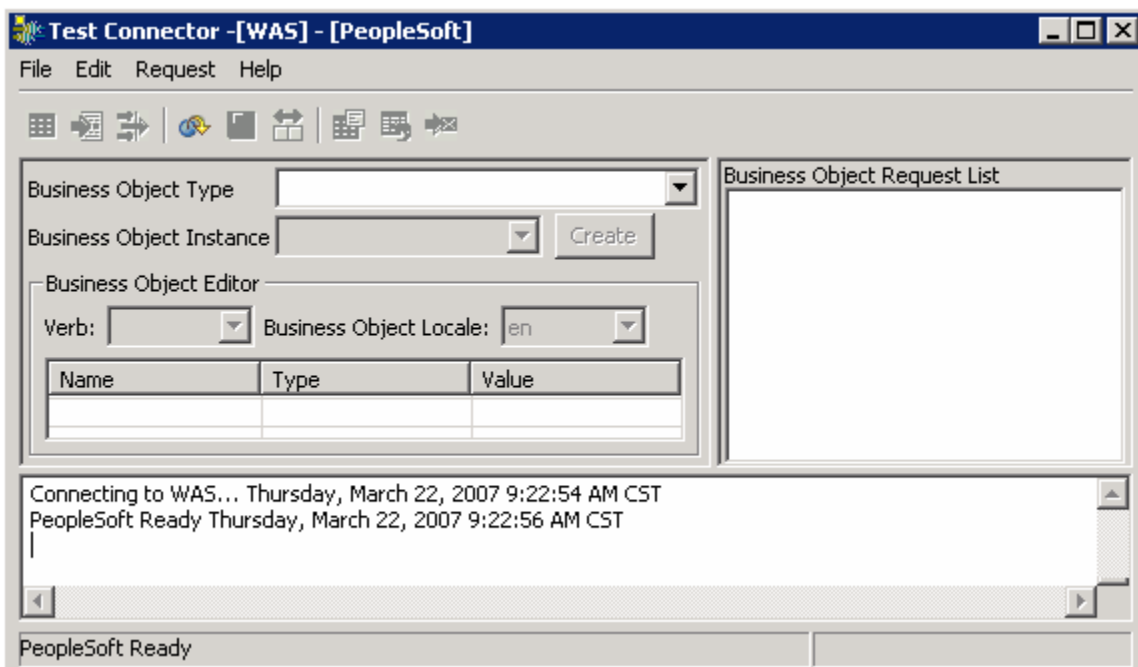


__ i. Click **OK**

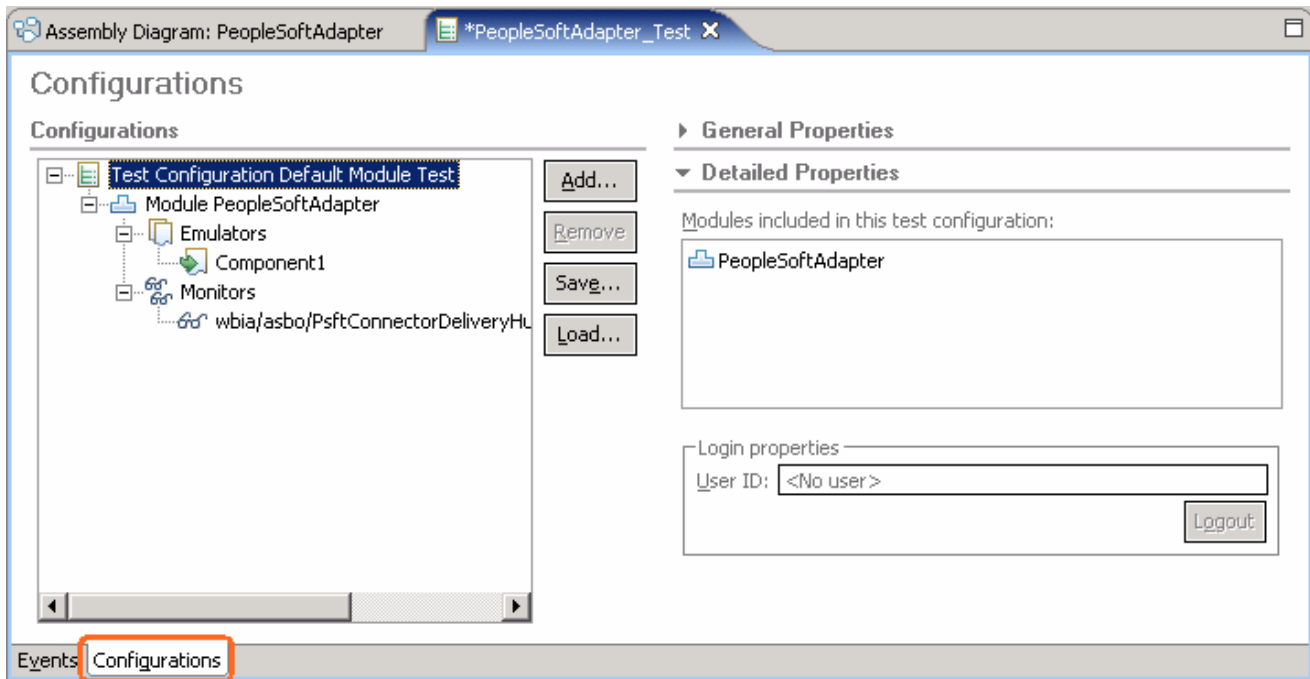
__ j. Now in the Test Connector interface, select **File** → **Connect** from the main menu



__ k. Ensure that a successful connection is made. You can confirm this by watching the trace messages displayed at the bottom frame of the Test Connector interface as shown below:

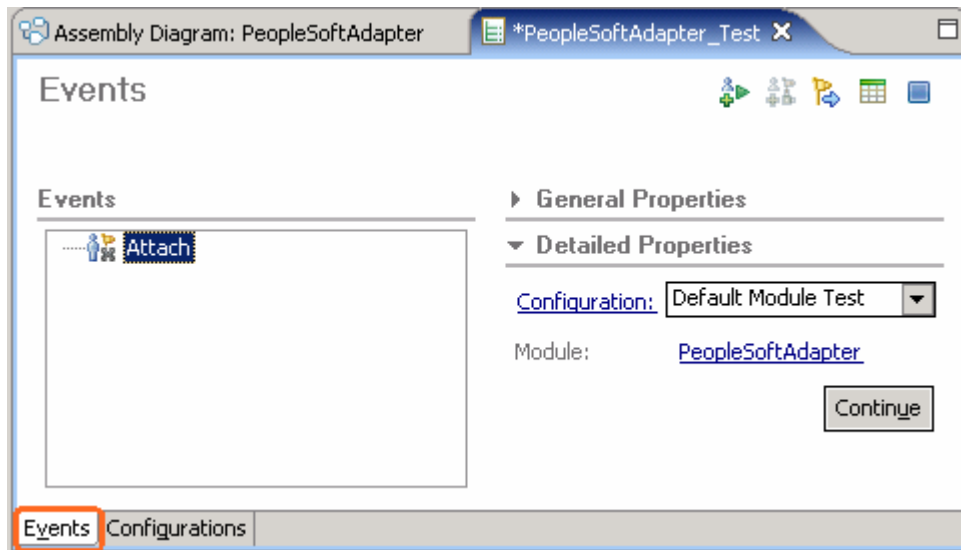


5. In the WebSphere Integration Developer, attach the test component to the PeopleSoftAdapter module
 - a. In the Business Integration view, right click on the **PeopleSoftAdapter** and select **Test** → **Attach** from the context menu. The PeopleSoftAdapter Test Component editor opens
 - b. Click on the **Configurations** tab at the bottom of the **Testing Module: PeopleSoftAdapter** view. You should see something like in the following image:

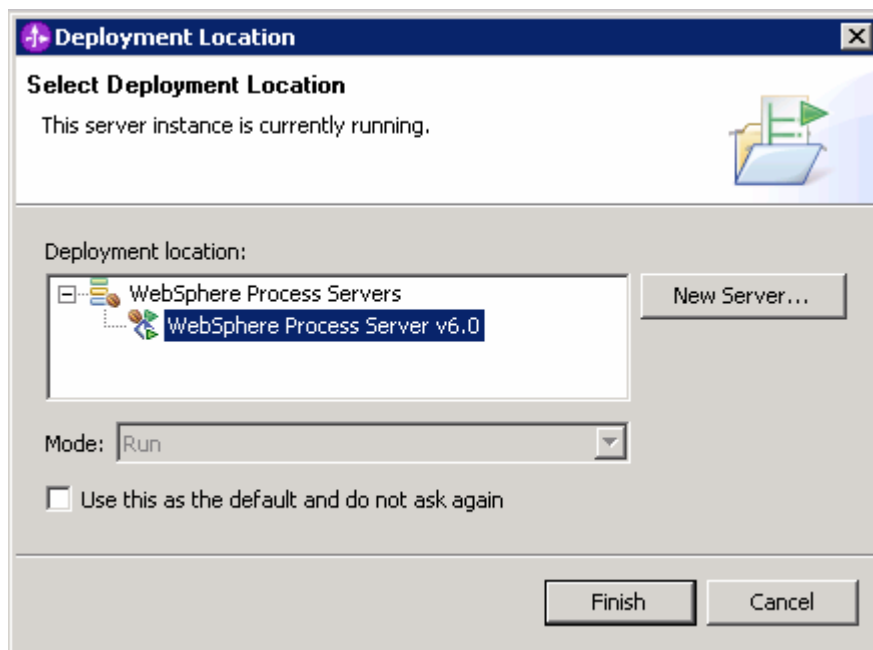


- c. Notice that **Component1** is listed under the **Emulators**. This means that Component1, which is not implemented, will be emulated when it is called. This will allow you to see the business graph that results from the input from the Visual Test Connector.
- d. Notice that **PsftConnectorDeliveryHub.<export>** → **Component1** is listed under Monitors. This means that anything that flows over that wire will result in an event which can then be interrogated to see what passed through the wire.

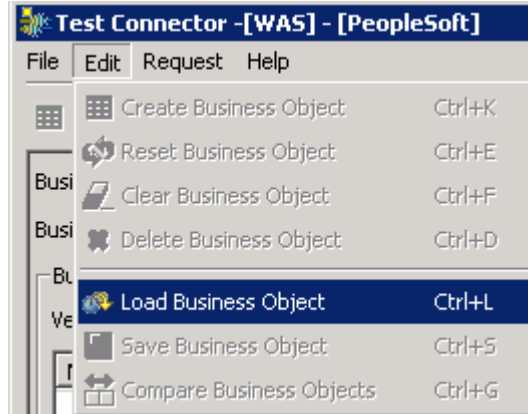
- ___ e. Click on the **Events** tab at the bottom of the **Testing Module: PeopleSoftAdapter** view. You should see something like the following screen capture. This is where the results will be shown when the application is invoked.



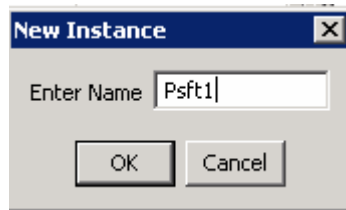
- ___ f. Click **Continue**
- ___ g. Select the **WebSphere Process Server V6.0** server in the **Choose a deployment location** dialog



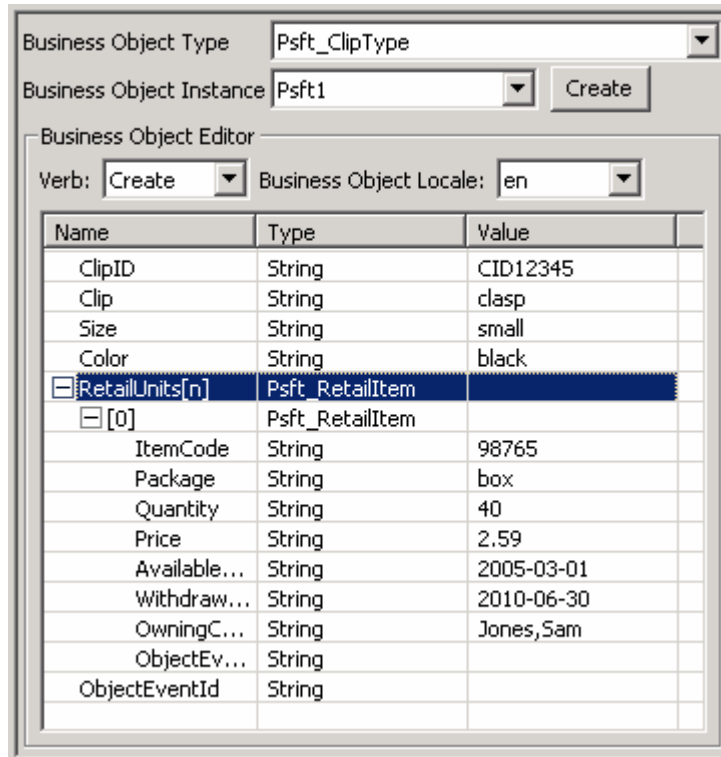
- ___ h. Click **Finish**
- ___ 6. Load a predefined business object into the Visual Test Connector
 - ___ a. In the main menu of the Visual Test Connector interface, select **Edit → Load Business Object**



- ___ b. In the Open dialog, navigate to <LAB_FILES>\eXchange\PeopleSoft\ASBOInstances and select **PsftASBO1.bo**
- ___ c. Click **Open**
- ___ d. In the New Instance dialog, enter the name to be assigned to this instance as **Psft1**



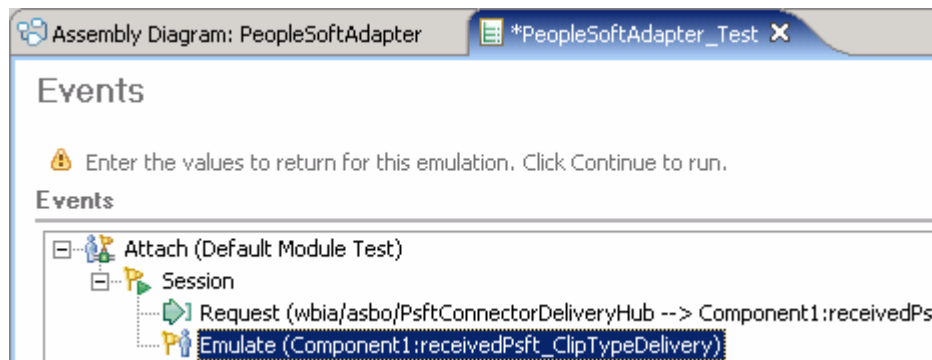
- ___ e. Click **OK**
- ___ 7. Review the data in the business object
 - ___ a. Review the values assigned to the Business Object, **Psft_ClipType**
 - ___ b. Expand **RetailUnits[n]** to expand the **Psft_RetailItem** array
 - ___ c. Expand **[0]** to view the instance of **Psft_RetailItem**



- ___ 8. Send the business object to the server
 - ___ a. From the main menu of the Visual Test Connector, select **Request → Send**



- ___ 9. Review the resultant business object in the server test component
 - ___ a. In the Events panel, you should see the Emulate of **Component1**, indicating it is being called at the **receivedPsft_ClipTypeDelivery** operation



- ___ b. If the Emulate is not highlighted, click on it so that the input parameter to the method is displayed on the right side of the panel

Input parameters

Name	Type	Value
[-] Psft_ClipTypeBGEElement	Psft_ClipTypeBG	
verb	VerbType	Create
[-] Psft_ClipType	Psft_ClipType	
ClipID	ClipIDType	CID12345
Clip	ClipType	clasp
Size	SizeType	small
Color	ColorType	black
[-] RetailUnits	Psft_RetailItem []	
[-] RetailUnits[0]	Psft_RetailItem	
ItemCode	ItemCodeType	98765
Parkane	ParkaneType	hnx

- ___ c. Examine the values on the input parameter (scroll down as required). The data field values should match those sent in from the Visual Test Connector
- ___ d. Notice that there are no Output parameters because you generated your PeopleSoft adapter to use the input with no response operation.
- ___ e. Click the **Continue** button to complete the transaction


___ 10. Stop and close the test component

- ___ a. Click the **Stop** (■) icon at the top right of the Events pane
- ___ b. Close the **PeopleSoftAdapter_Test** component editor
- ___ c. Click **No** in the Save Resource dialog

___ 11. Close the Visual Test Connector

- ___ a. Select **File → Exit** in the VTC interface
- ___ b. Select **Yes** in the Shutdown dialog
- ___ c. Press **Enter** key In the Visual Test Connector command window to close it

___ 12. Remove the unimplemented component that is the **Component1**. It is not needed in subsequent labs

- ___ a. In the WebSphere Integration Developer, expand **PeopleSoftAdapter** in the Business Integration view and double click on **Assembly Diagram** ( **Assembly Diagram**) to open it in an Assembly Diagram editor
- ___ b. Right click on **Component1** and select **Delete** from the context menu
- ___ c. The resultant Assembly diagram looks like the following screen capture:




- ___ d. Save all work by **File → Save All** or **Ctrl + Shift + S**

- ___ e. Close the Assembly Diagram
- ___ 13. Remove the PeopleSoftAdapter module from the WebSphere Process Server test environment
 - ___ a. Right click on WebSphere Process Server V6.0 from the Server view and select **Add and Remove Projects...** from the context menu
 - ___ b. Select the **PeopleSoftAdapterApp** project from the Configured projects list
 - ___ c. Click **< Remove**
 - ___ d. Click **Finish**

Part 5: Assemble and test the SiebelAdapter module

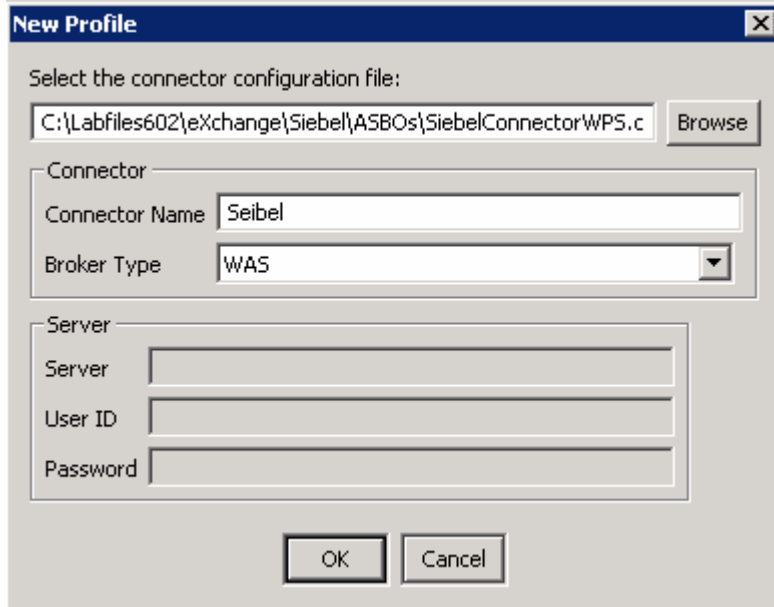
In this part you will open the Assembly Editor for the SiebelAdapter module. Using the Component Test capability in WebSphere Integration Developer and the Visual Test Connector, you will test the Import to ensure it can be used to send a business object to the adapter.

- ___ 1. Open the SiebelAdapter in the Assembly Editor
 - ___ a. In the WebSphere Integration Developer, expand **SiebelAdapter** in the Business Integration view and double click on **Assembly Diagram** ( **Assembly Diagram**) to open it in an Assembly Diagram editor
 - ___ b. In the Assembly Editor, you will see a message that there is a new element added to the module. This is the Import that was generated using Enterprise Service Discovery in a previous part of this lab.

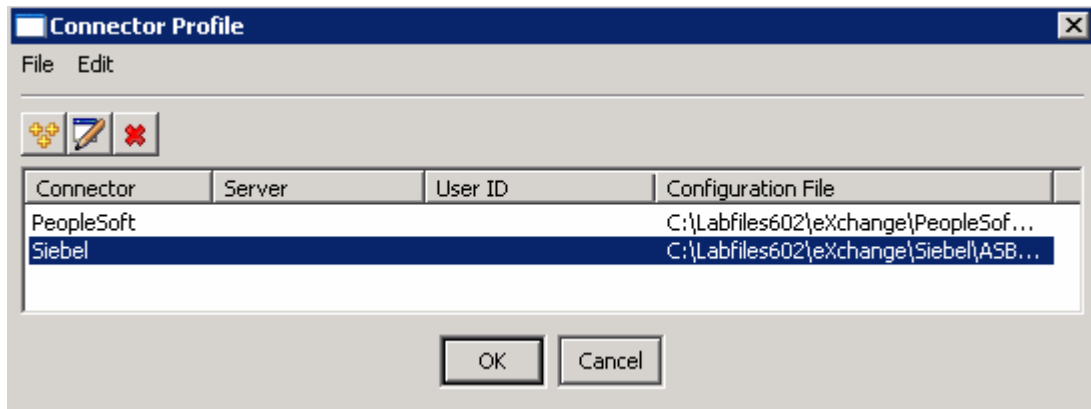


- ___ 2. Add the SiebelAdapter module to the WebSphere Process Server test environment
 - ___ a. Right click on WebSphere Process Server V6.0 from the Server view and select **Add and Remove Projects...** from the context menu
 - ___ b. Select only the **SiebelAdapterApp** project from the Available projects panel
 - ___ c. Click **Add >** to add it to the Configured projects panel
 - ___ d. Click **Finish**
- ___ 3. Connect the Visual Test Connector to the WebSphere Process Server test environment. Using the connector configuration at **<LAB_FILES>\eXchange\Siebel\ASBOs\SiebelConnectorWPS.cfg** and follow the instructions below to use the Visual Test Connector:
- ___ 4. Ensure the WebSphere Process Server is active (the server is the JMS Broker). The VTC requires an active JMS Broker to start its connection.
 - ___ a. Ensure the WebSphere Process Server is active (the server is the JMS Broker). The VTC requires an active JMS Broker to start its connection
 - ___ b. Start the Visual Test Connector
 - ___ c. From the **start** menu, select **Programs → IBM WebSphere Business Integration Adapters → IBM WebSphere Business Integration Toolset → Visual Test Connector**. This opens a command window in the background where the process runs and the Visual Test Connector Interface in the foreground. Do not close the command window as this will cause the VTC to exit
 - ___ d. From the main menu of the VTC interface, select **Create/Select a Profile**. The Connector Profile panel opens
 - ___ e. Create a new Seibel profile by selecting **File → New Profile...** from the main menu of the Connector Profile panel
 - ___ f. Enter the following information in the New Profile panel:

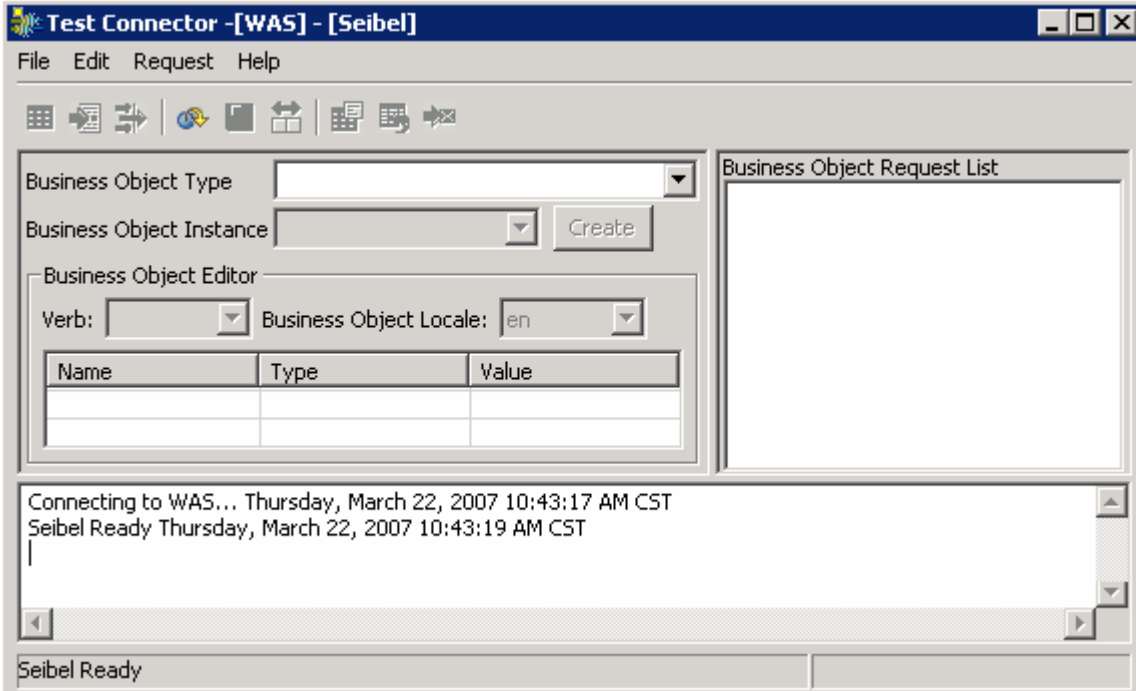
- 1) Click the **Browse** button for “Select the connector configuration file:” field and select <LAB_FILES>\eXchange\Siebel\ASBOs\SiebelConnectorWPS.cfg file
- 2) Connector Name : **Siebel**
- 3) Broker Type : **WAS**




- ___ g. Click **OK**
- ___ h. Select the Siebel profile you had created as shown below:



- ___ i. Click **OK**
- ___ j. Now in the Test Connector interface, select **File → Connect** from the main menu
- ___ k. Ensure that a successful connection is made. You can confirm this by watching the trace messages displayed at the bottom frame of the Test Connector interface as shown below:



- ___ 5. In the WebSphere Integration Developer, expand **SiebelAdapter** in the Business Integration view and double click on **Assembly Diagram** ( **Assembly Diagram**) to open it in an Assembly Diagram editor
 - ___ a. Right click on the **SiebelConnectorAgent** import in the Assembly editor and select **Test Component** from the context menu. The **SiebelConnectorAgent** test component editor opens
- ___ 6. Enter the following values for Initial Request parameters:
 - ___ a. verb : Select **Create** from the drop down list
 - ___ b. Description : **small black clasp**
 - ___ c. Right click on **ProductPackages** and select **Add element** from the context menu

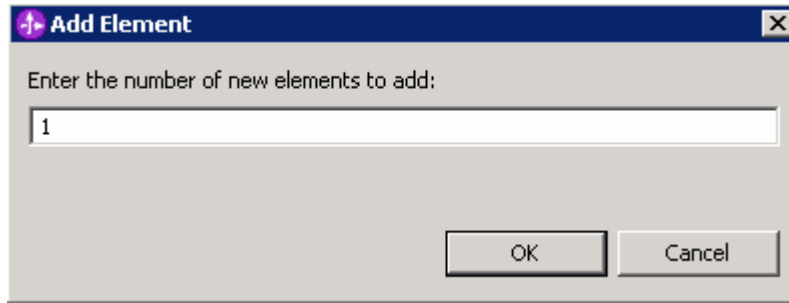
Initial request parameters

Name	Type
<input type="checkbox"/> Siebel_ProductTypeBGElem...	Siebel_ProductTypeBC
verb	string
<input type="checkbox"/> Siebel_ProductType	Siebel_ProductType
Key	string
Description	string
ProductPackages	Siebel_ProductPackag
version	
delta	
locale	

Set Value...

Add Element

- ___ d. Enter **1** for the “Number of New Elements to Add” in the New Element dialog



__ e. Now enter the following values or of your choice under ProductPackages

- 1) GlobalIdentifier: **12345678901234**
- 2) PackageDescription: **box 40**
- 3) Price: **2.59**
- 4) AvailabilityStart: **2007-03-23**
- 5) AvailabiltyEnd: **2010-06-30**
- 6) Contact: **Sam Jones**

Initial request parameters

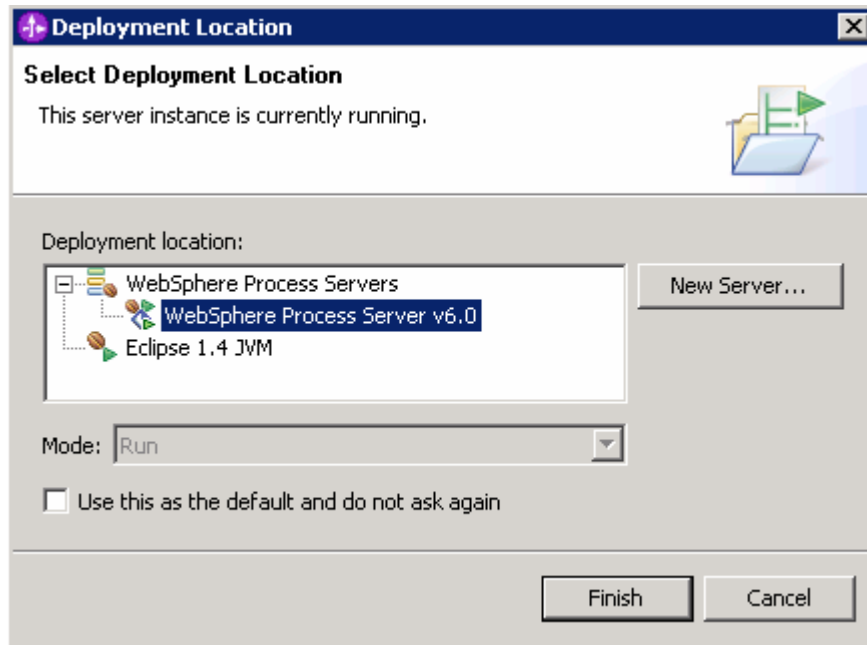
Name	Type	Value
Key	string	
Description	string	small clasp black
<input checked="" type="checkbox"/> ProductPackages	Siebel_ProductPackage []	
<input type="checkbox"/> ProductPacka...	Siebel_ProductPackage	
GlobalIdenfier	string	12345678901234
PackageDe...	string	box 40
Price	string	2.59
Availability...	string	2005-03-01
Availability...	string	2005-06-30
Contact	string	Sam Jones
versinn	token	

Data Pool Continue

___ 7. Run the test

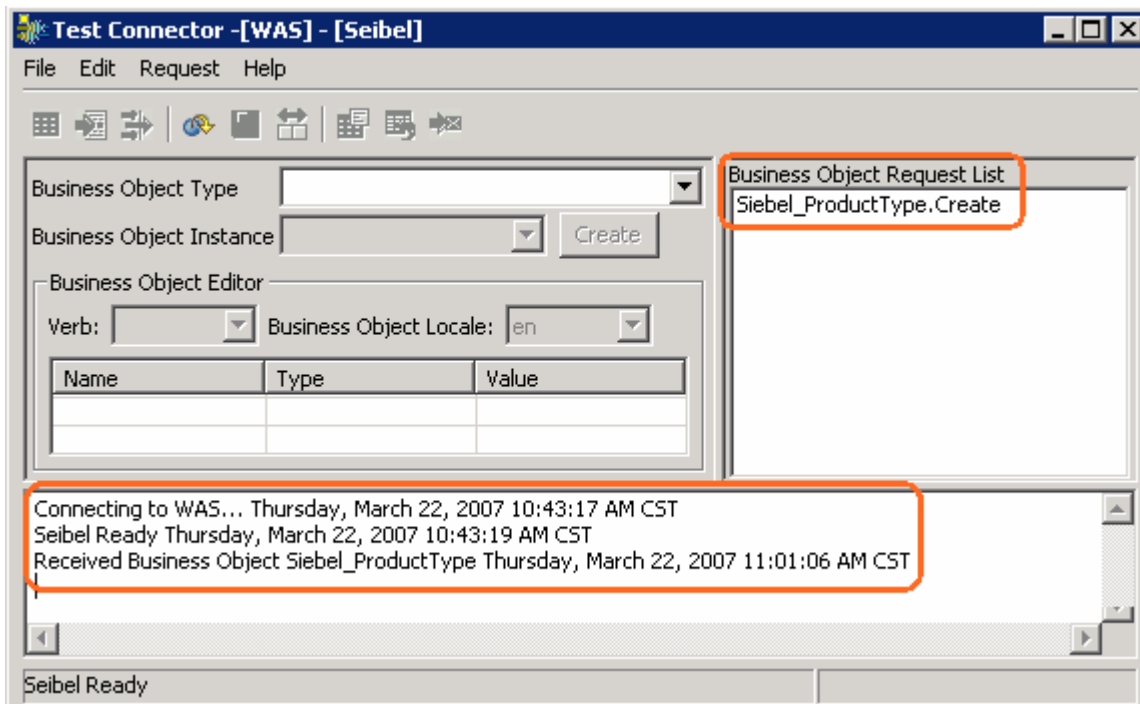
__ a. Click the **Continue** button

- __ b. Select the **WebSphere Process Server V6.0** server in the **Choose a deployment location** dialog



- __ c. Click **Finish**

- __ d. Review the Visual Test Connector window to see if a message has been received



- __ e. Double click on **Siebel_ProductType.create** in the Business Object Request List to view the business object received by the Visual Test Connector

___ f. The business object should look like this:

Name	Type	Value
Key	String	
Description	String	small black clasp
ProductPack...	Siebel_ProductP...	
[0]	Siebel_ProductP...	
GlobalI...	String	12345678901234
Packa...	String	box 40
Price	String	2.59
Availa...	String	2007-03-23
Availa...	String	2010-06-30
Contact	String	Sam Jones
Object...	String	CxCommon.BusinessObject@106...
ObjectEventId	String	CxCommon.BusinessObject@106...

___ 8. Stop and close the test component

___ a. Click the **Stop** (■) icon at the top right of the Events pane

___ b. Close the SiebelConnectorAgent_Test component editor

___ c. Click **No** in the Save Resource dialog

___ 9. Close the Visual Test Connector

___ a. Select **File → Exit** in the VTC window

___ b. Select **Yes** in the Shutdown dialog

___ c. Click the **Enter** key in the Visual Test Connector command window to close it

___ 10. Clean WebSphere Process Server test environment

___ a. Right-click on WebSphere Process Server V6 from the Servers view and select **Add and remove projects...** from the context menu

___ b. Click **<< Remove All**

___ c. Click **Finish**

___ 11. Stop the Server

___ a. Right click on WebSphere Process Server V6.0 server from the Servers view and select **Stop** from the context menu

What you did in this exercise

In this chapter you started with configuration information and business object definitions used with a WebSphere Interchange Server and PeopleSoft and Siebel adapters. You used the Enterprise Service Discovery tool to do the conversions required to allow the adapters to be used with the WebSphere Process Server. Using a properties file generated from ESD, you configured the WebSphere Process Server to have the queues required for communication with the adapters. Using the Export and Import generated from ESD, you constructed wiring assemblies for the two modules. Finally, using the Test Component and the Visual Test Connector the modules were tested.

Solution instructions

- ___ 1. Follow the directions in the task [Initialize the Workspace for a Lab Exercise](#), using the following values:

<WORKSPACE>

C:\Labfiles602\exchange\WBIAdapters\workspace

<PROJECT_INTERCHANGE>

C:\Labfiles602\exchange\WBIAdapters\Solution\ImportExport_Pi.zip

<MODULE>

n/a

<DEPENDENT_LIBRARIES>

n/a

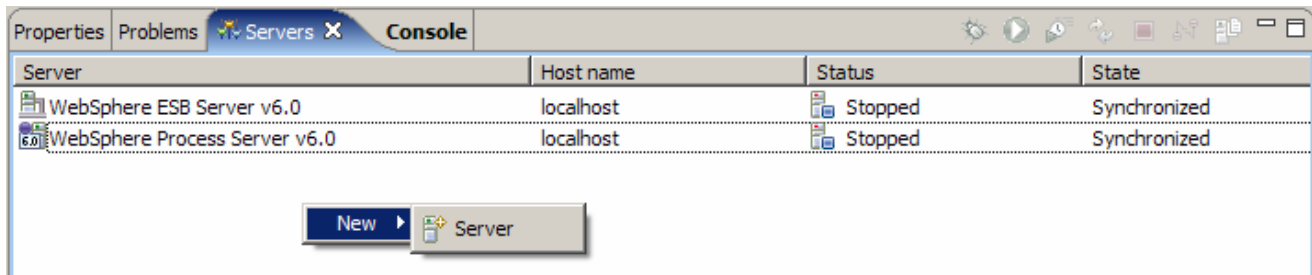
- ___ 2. Continue from **Part 4: Assemble and Test the PeopleSoft Module**

Task: Adding remote server to WebSphere Integration Developer test environment

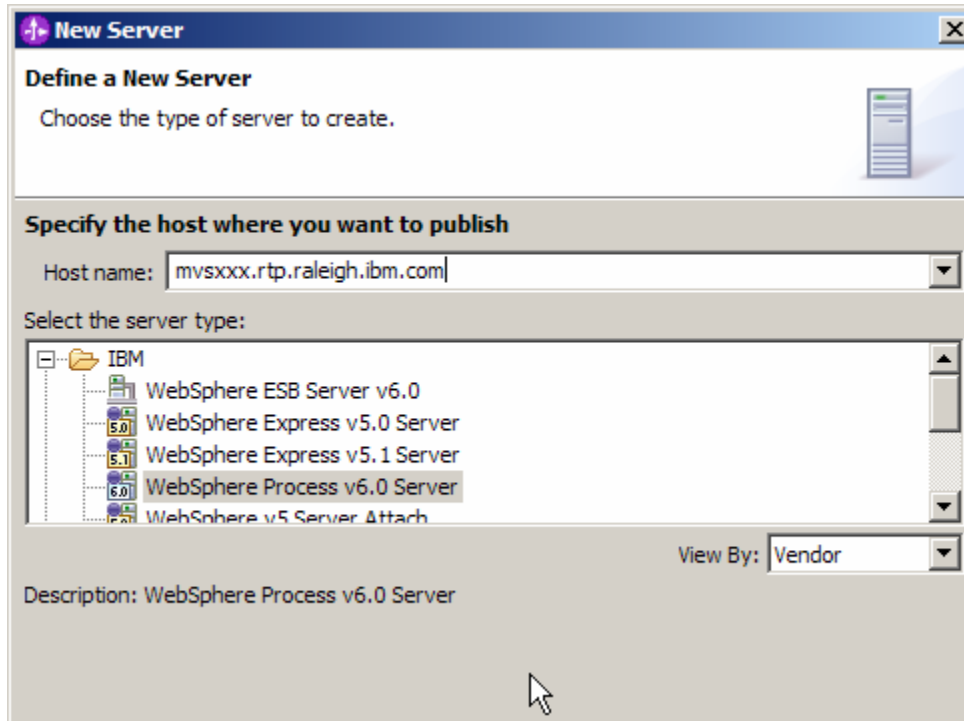
This task describes how to add a remote server to the WebSphere Integration Developer Test environment. The sample you will use is a z/OS machine.

Create a new remote server.

- ___ 1. Right click on the background of the Servers view to access the pop-up menu.
- ___ 2. Select New → Server.

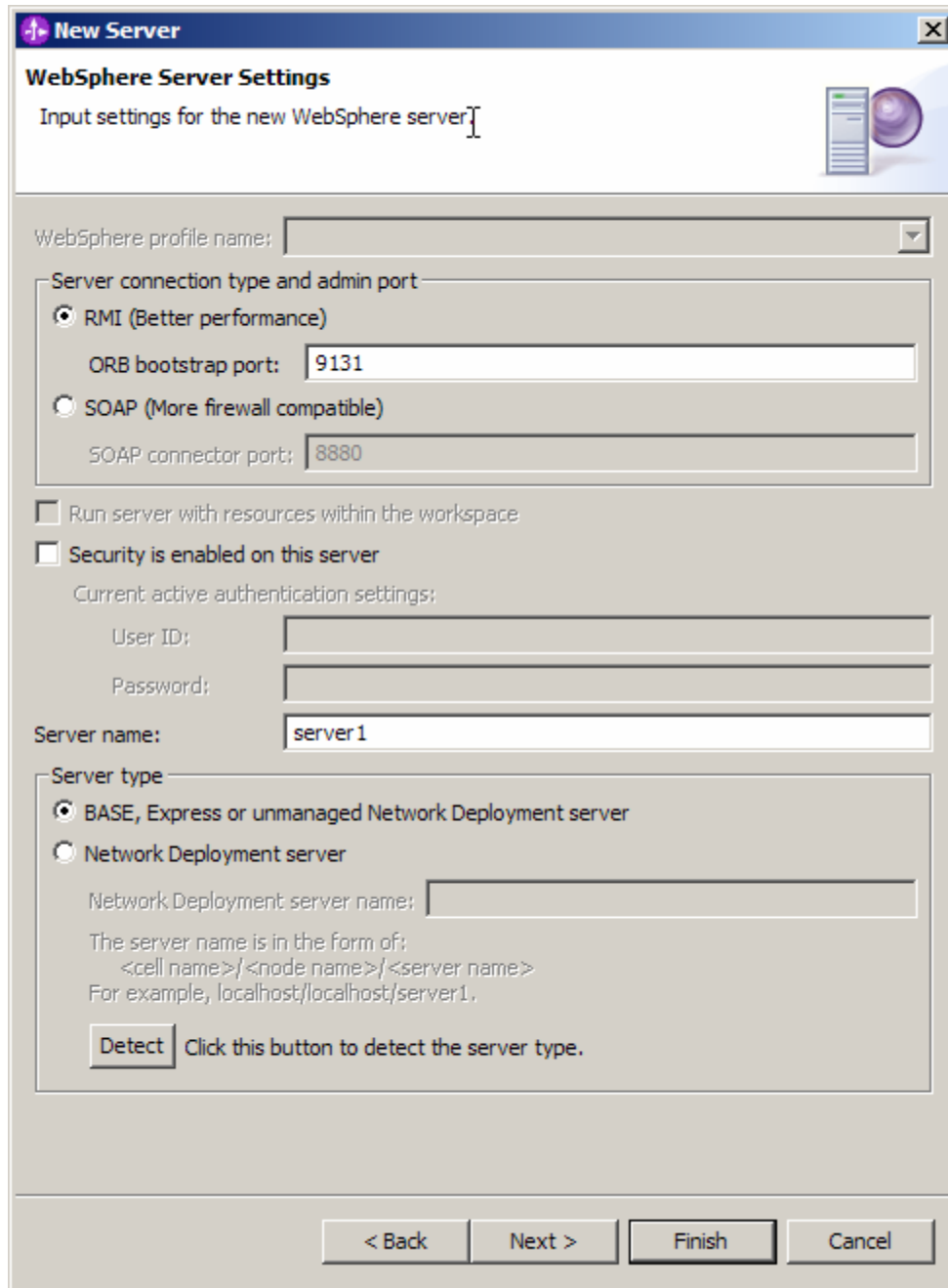


- ___ 3. Specify host name to the remote server, <HOSTNAME>.
- ___ 4. Ensure that 'WebSphere Process V6.0 Server' is highlighted in the server type list.

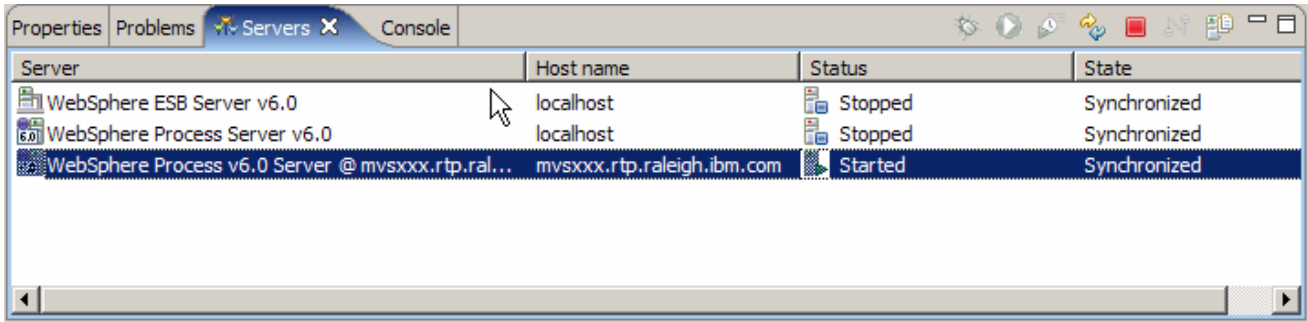


- ___ 5. Click **Next**.

- 6. On the WebSphere Server Settings page, select the radio button for **RMI** and change the ORB bootstrap port to the correct setting (<BOOTSTRAP_PORT>).



- 7. Click **Finish**.
- 8. The new server should be seen in the Server view.



- ___ 9. Start the remote server if it is not already started. WebSphere Integration Developer does not support starting remote servers from the Server View.
- ___ 10. From a command prompt, telnet to the remote system if needed:
- ```
'telnet <HOSTNAME> <TELNET_PORT>'
 userid : <USERID>
 pw : <PASSWORD>
```
- \_\_\_ 11. Navigate to the bin directory for the profile being used:
- ```
cd <WAS_HOME>/profiles/<PROFILE_NAME>/bin
```
- ___ 12. Run the command file to start the server: `./startServer.sh <SERVER_NAME>`
- ___ 13. Wait for status message indicating server has started:

```
ADMU3200I: Server launched. Waiting for initialization status
ADMU3000I: Server c11sr01 open for e-business; process id is 0000012000000002
```