IBM WebSphere® Process Integration 6.0 – Lab Exercise

WebSphere Business Integration Adapters – Import/Export

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 v6.0.1 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 V6.0.1 installed
- WebSphere Process Server V6.0 test environment installed
- Sample code in the directory C:\Labfiles60 (Windows®) or /tmp/LabFiles60 (Linux®)
- WebSphere Business Integration Toolset installed

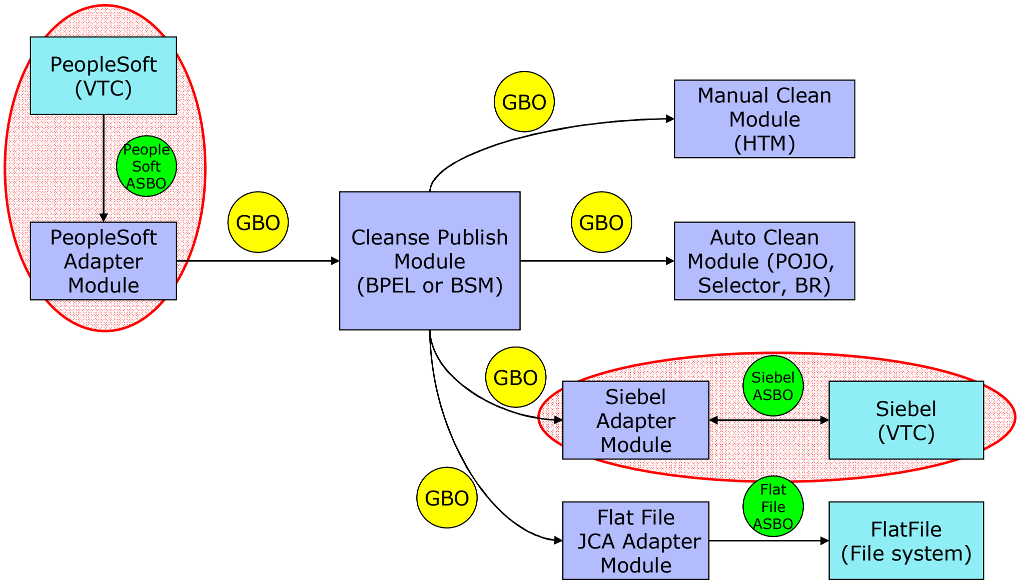
What you should be able to do

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

- Generate the artifacts required to use a WBI 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.

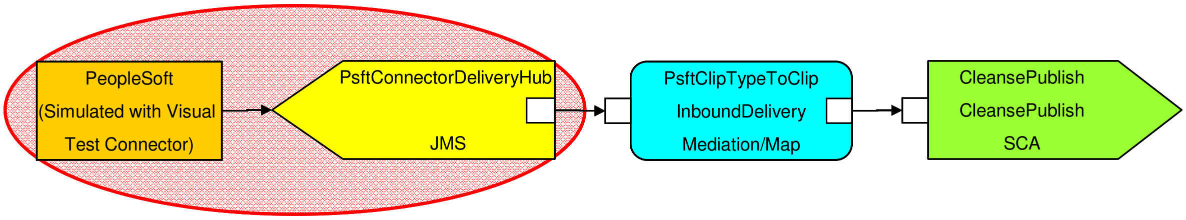

PeopleSoft (VTC)
People Soft ASBO
PeopleSoft Adapter Module
GBO
Cleanse Publish Module (BPEL or BSM)
GBO
Manual Clean Module (HTM)
GBO
Auto Clean Module (POJO, Selector, BR)
GBO
Siebel Adapter Module
Siebel ASBO
Siebel (VTC)
GBO
Flat File JCA Adapter Module
Flat File ASBO
FlatFile (File system)

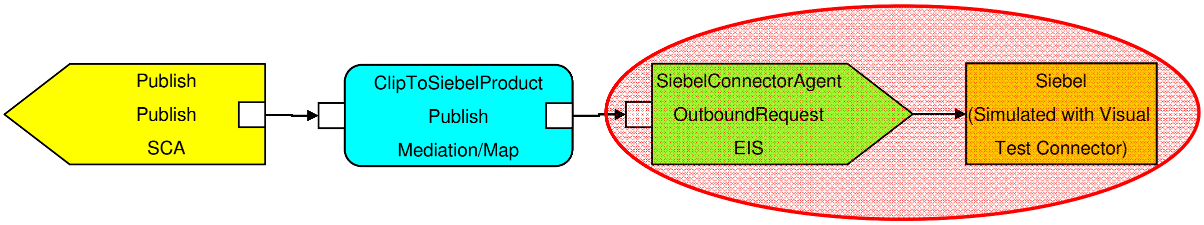
The process of converting the ASBOs and connector configuration is done using the Enterprise Service Discovery tool. It takes as input the .xsd files that define the ASBOs and the .cfg file that defines the connection between the WBI-Adapter and the existing broker. The following is produced as output:

- .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.

Please provide an introduction to the lab. Highlight major teach points and techniques used in this lab.

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:\Labfiles60	/tmp/Labfiles60
<WORKSPACE>	C:\Labfiles60\eXchange\WBIAdapters\workspace	/tmp/Labfiles60/eXchange/WBIAdapters/workspace
<TEMP>	C:\temp	/tmp
<SOLUTION>	C:\Labfiles60\eXchange\WBIAdapters\Solution	/tmp/Labfiles60/eXchange/WBIAdapaters/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

Follow the directions below, substituting the values listed for the variables.

<WORKSPACE>

C:\Labfiles60\eXchange\WBIAdapters\workspace

<PROJECT_INTERCHANGE>

C:\Labfiles60\eXchange\CleansePublishLibrary\import\CleansePublishLibrary_PI.zip

<MODULE> - (create both modules)

PeopleSoftAdapter

SiebelAdapter

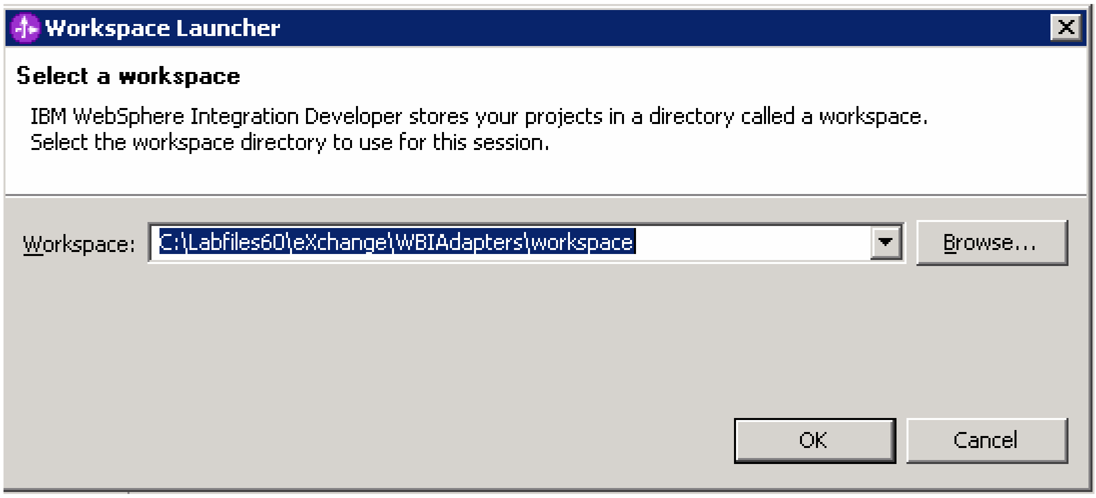
<DEPENDENT_LIBRARIES> - (this is a dependent library for each of the modules)

CleansePublishLibrary

____ 1. Start WebSphere Integration Developer V6 with a new workspace located at **<WORKSPACE>**.

__ a. Navigate to the **<WID_HOME>** directory from Windows Explorer and double click on wid.exe.

__ b. Enter the value for the **<WORKSPACE>** variable for this lab and click **OK, w**hen prompted.

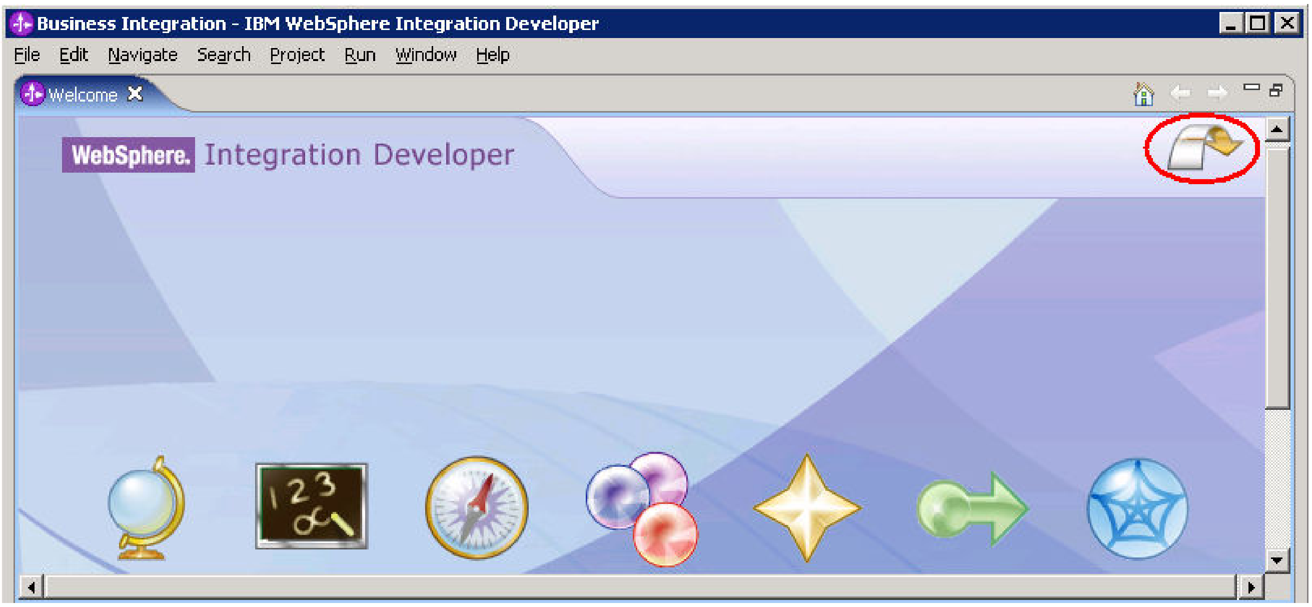


Select a workspace

IBM WebSphere Integration Developer stores your projects in a directory called a workspace.
Select the workspace directory to use for this session.

Workspace: C:\Labfiles60\eXchange\WBIAdapters\workspace

__ c. Close the **Welcome page** when WebSphere Integration Developer V6.0 opens by clicking on the Go to the workbench icon (bent over arrow at top-right).

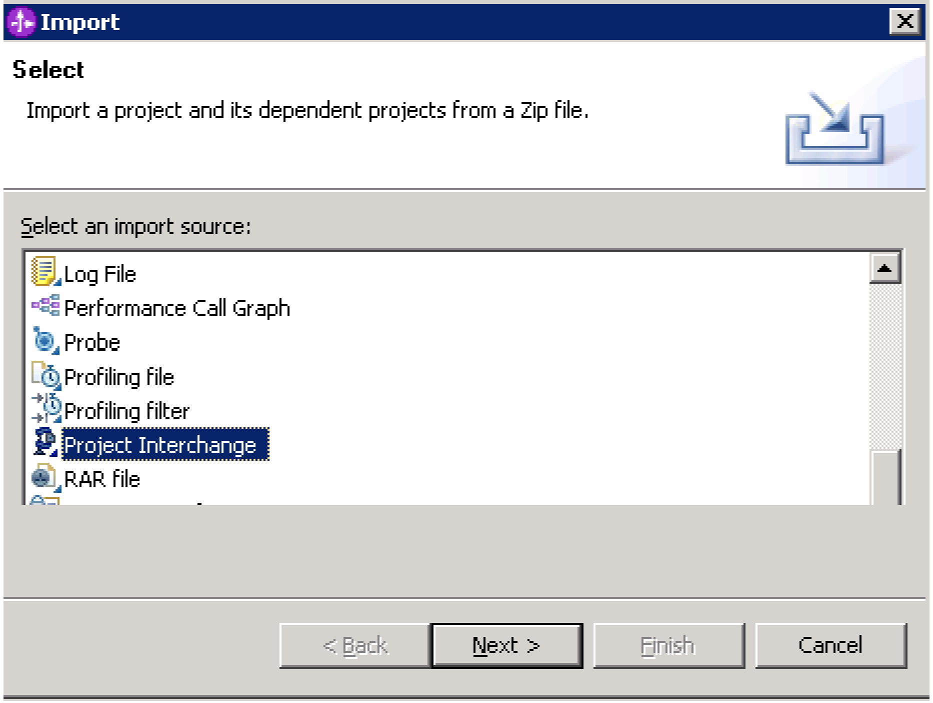

Business Integration - IBM WebSphere Integration Developer
File Edit Navigate Search Project Run Window Help
Welcome
WebSphere. Integration Developer

__ d. Ensure you are in the **Business Integration** perspective.

____ 2. If this lab requires you to import a project interchange file, setup the required libraries and modules for this lab by importing the project interchange file **<PROJECT_INTERCHANGE>.**

__ a. Select **File -> Import…** from the menu bar.

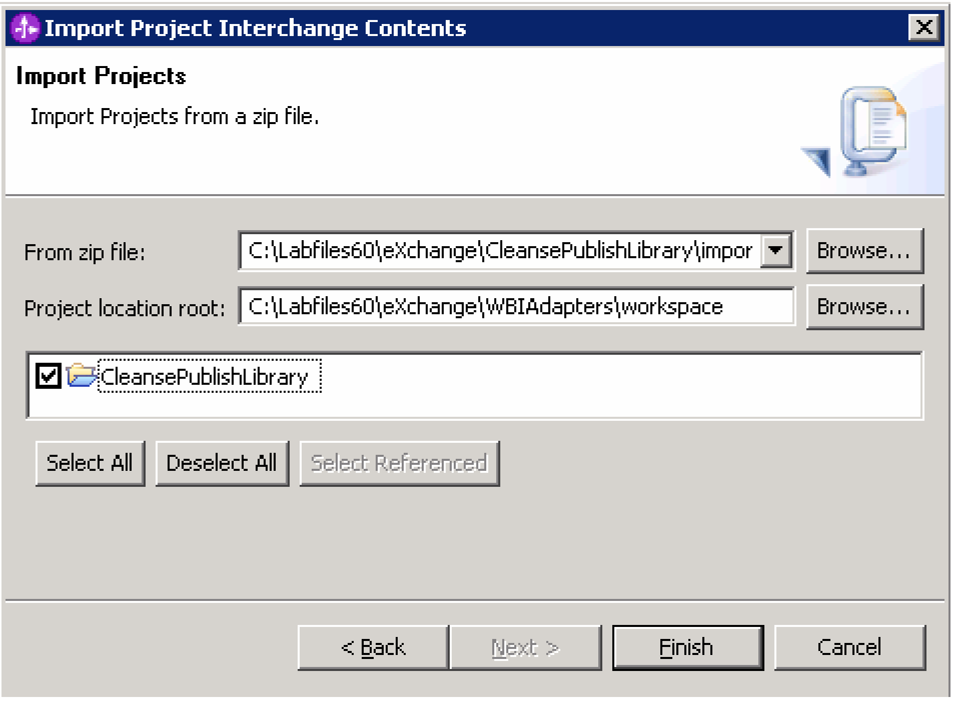
__ b. Scroll down and select **Project Interchange** in the Import dialog.



__ c. Click **Next.**

__ d. Initialize the From zip file: in the Import Projects dialog field to <**PROJECT_INTERCHANGE>.**

__ e. Click the **Select All** button.


Import Project Interchange Contents

Import Projects

Import Projects from a zip file.

From zip file: C:\Labfiles60\eXchange\CleansePublishLibrary\impor

Project location root: C:\Labfiles60\eXchange\WBIAdapters\workspace

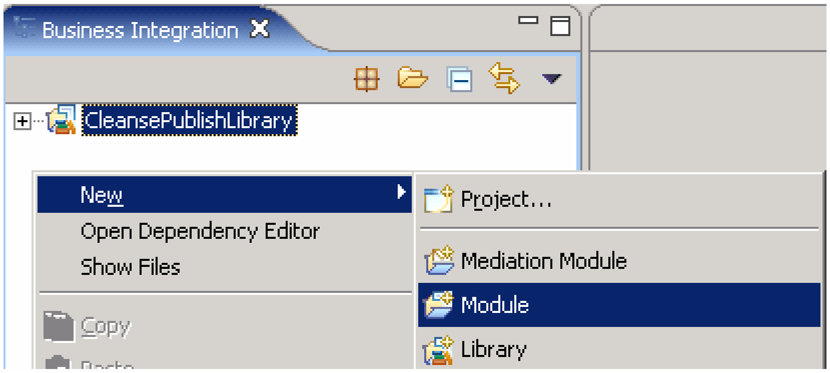
☑ CleansePublishLibrary

__ f. Click **Finish.**

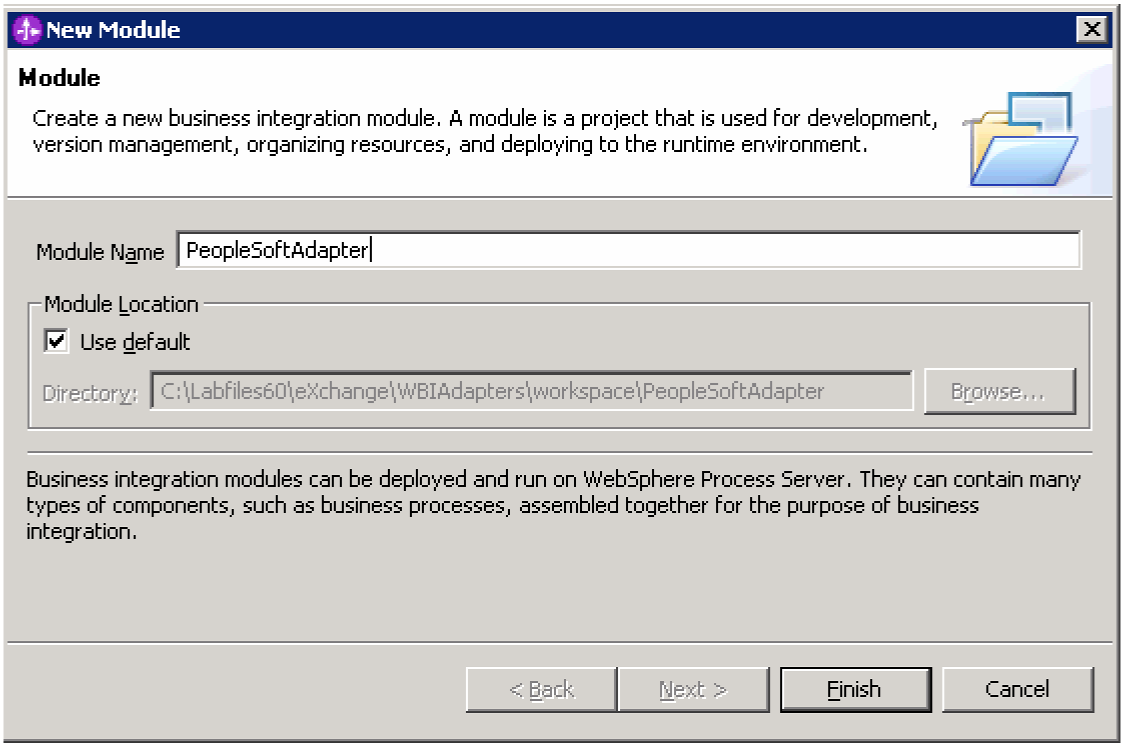
____ 3. If this lab requires that you create a Business Integration module called <**MODULE>**, complete these steps:

__ a. Right click on the background of the Business Integration view to access the pop-up menu.

__ b. Select **New** > **Module.**



__ c. Enter **<MODULE>** for the Module Name in the New Module dialog.



New Module

Module

Create a new business integration module. A module is a project that is used for development, version management, organizing resources, and deploying to the runtime environment.

Module Name: PeopleSoftAdapter

Module Location

☑ Use default

Directory: C:\Labfiles60\eXchange\WBIAdapters\workspace\PeopleSoftAdapter

Business integration modules can be deployed and run on WebSphere Process Server. They can contain many types of components, such as business processes, assembled together for the purpose of business integration.

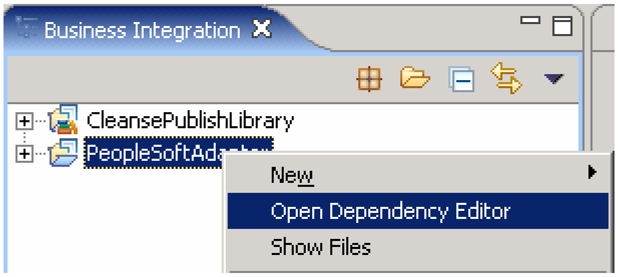
Note: For this lab you will have to create 2 modules – PeopleSoftAdapter and SiebelAdapter.

__ d. Click **Finish.**

____ 4. If this lab requires that **<MODULE>** needs any **<DEPENDENT_LIBRARIES>**, complete these steps:

__ a. In the Business Integration view, right click on the **<MODULE>** you just created to access the pop-up menu.

__ b. Select **Open Dependency Editor.**



__ c. Click the **Add…** button in the Dependency Editor.

__ d. Select from the list the **<DEPENDENT_LIBRARIES>** in the Library Selection dialog and click **OK**.

__ e. Repeat the above steps to create another module name SiebelAdapter

Note: For this lab you will have to add dependency library for the two models created.

__ f. Press **Ctrl+S** to save the dependencies for this module.

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 WBI-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 WBI-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

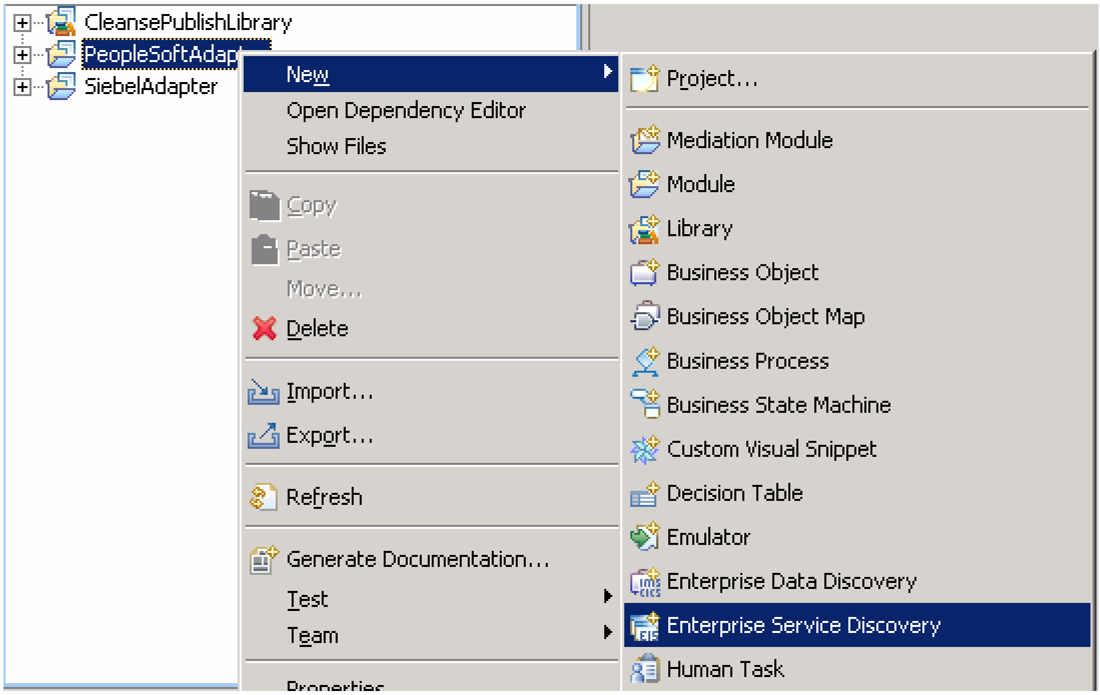
In this lab you will generate and test the artifacts for the PeopleSoft adapter and then repeat the process for the Siebel adapter. The following steps will lead you through the process.

Instructions for PeopleSoft

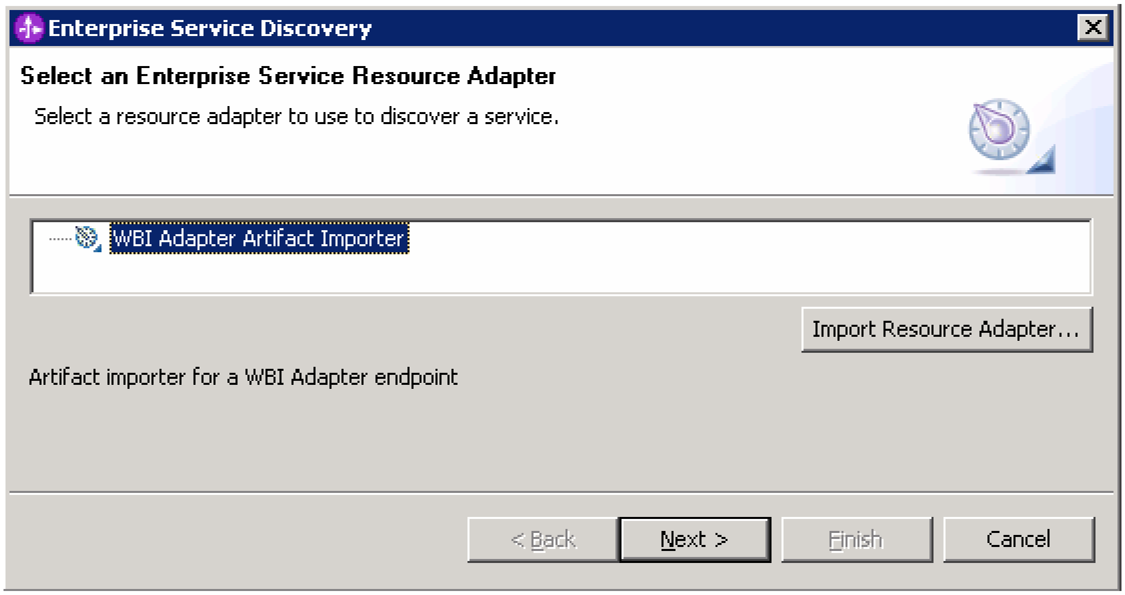
____ 1. Access the Enterprise Service Discovery wizard to generate the artifacts that will be used with the adapter for PeopleSoft.

__ a. Right click on the **PeopleSoftAdapter** module from the Business Integration perspective.

__ b. Select **New** > **Enterprise Service Discovery.**



__ c. Select **WBI Adapter Artifact Importer** in the **Enterprise Service Discovery** dialog.



Enterprise Service Discovery

Select an Enterprise Service Resource Adapter

Select a resource adapter to use to discover a service.

WBI Adapter Artifact Importer

Artifact importer for a WBI Adapter endpoint

__ d. Click **Next.**

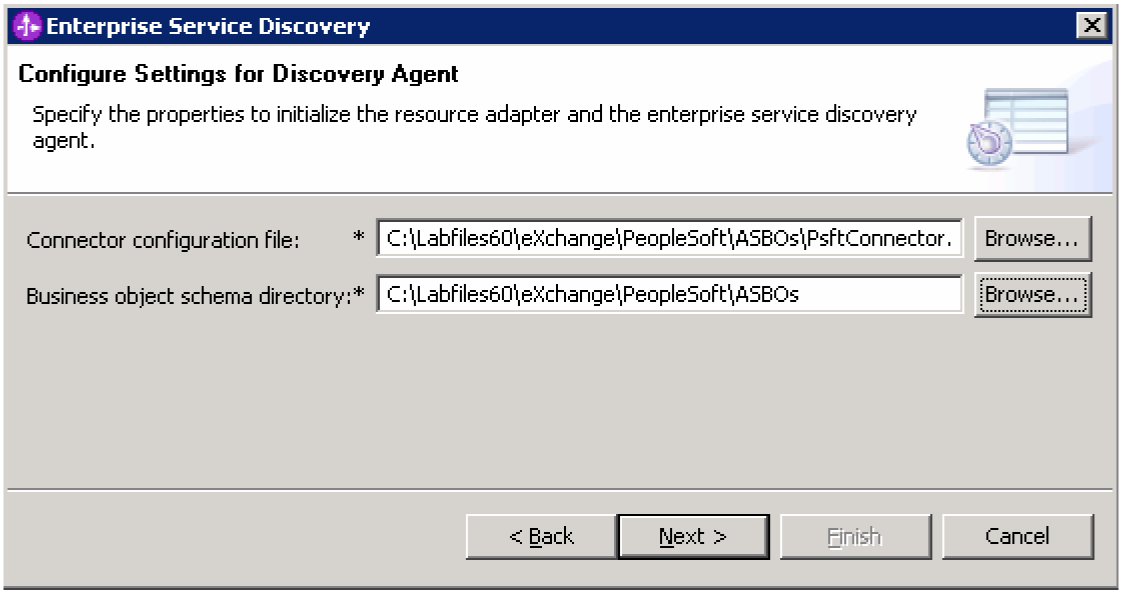
____ 2. Identify the specific configuration and location of the business objects to be discovered.

__ a. In the **Configure Settings for Discovery Agent** dialog, uses the **Browse…** or enter the fully qualified path to the connector configuration file as:

`<LAB_FILES>\eXchange\PeopleSoft\ASBOs\PsftConnector.cfg.`

and to the directory containing the ASBO definitions as:

`<LAB_FILES>\eXchange\PeopleSoft\ASBOs.`



Enterprise Service Discovery

Configure Settings for Discovery Agent

Specify the properties to initialize the resource adapter and the enterprise service discovery agent.

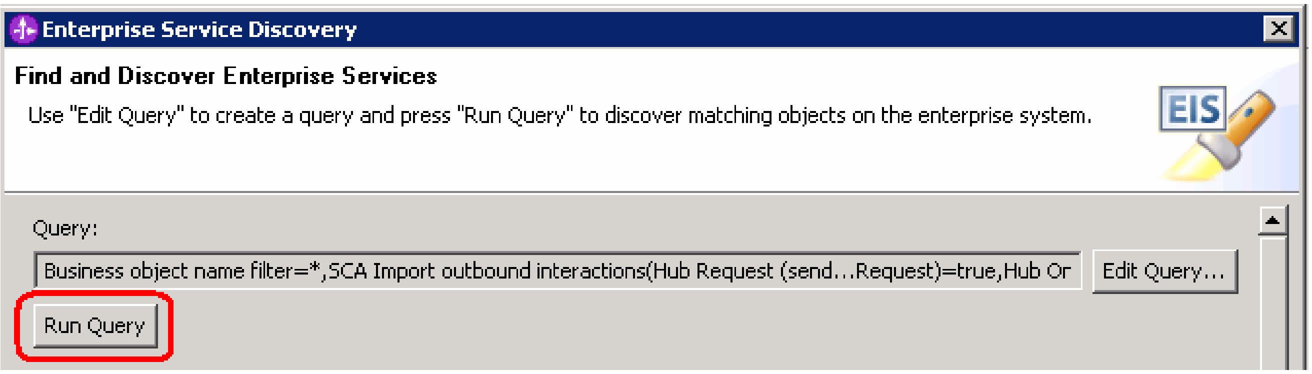
Connector configuration file: * C:\Labfiles60\eXchange\PeopleSoft\ASBOs\PsftConnector.

Business object schema directory:* C:\Labfiles60\eXchange\PeopleSoft\ASBOs

__ b. Click **Next.**

____ 3. Pick the specific objects from PeopleSoft to be processed.

__ a. Click the **Run Query** button on the **Find and Discover Enterprise Services** dialog to discover the PeopleSoft objects.



Enterprise Service Discovery

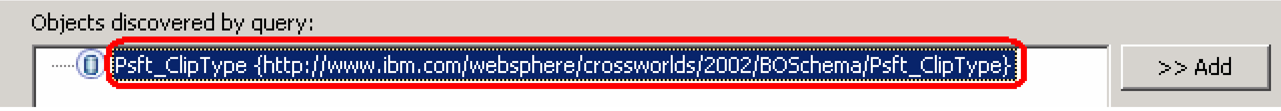
Find and Discover Enterprise Services

Use "Edit Query" to create a query and press "Run Query" to discover matching objects on the enterprise system.

Query:

Business object name filter=*,SCA Import outbound interactions(Hub Request (send...Request)=true,Hub Or

__ b. Select the business objects you want to process in the **Objects discovered by query** box.



Objects discovered by query:

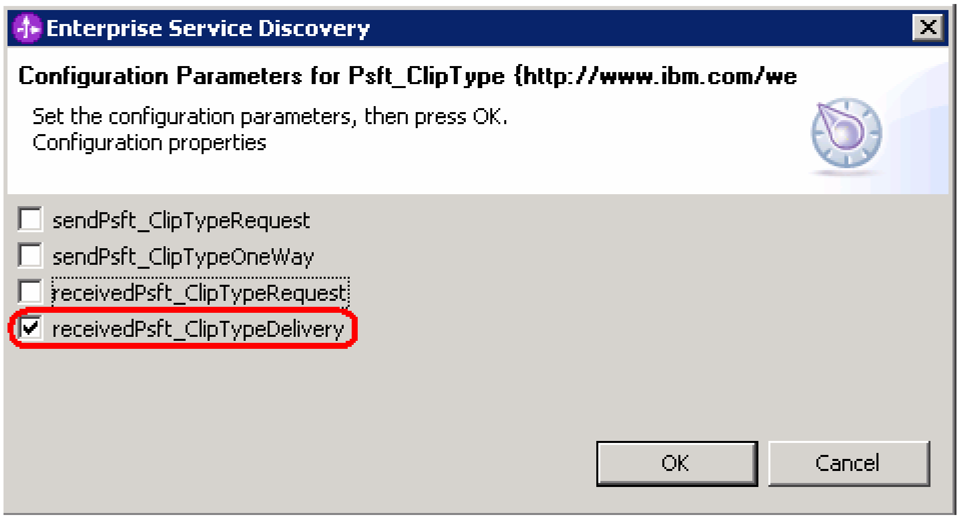
Psft_ClipType {http://www.ibm.com/websphere/crossworlds/2002/BOSchema/Psft_ClipType}

__ c. Click **>>Add** and you will be presented with the Configuration Parameters for Psft_ClipType dialog.

____ 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. For this scenario, you only need inbound requests from adapter to server without a response. Therefore, make sure that only **receivedPsft_ClipTypeDelivery** is checked, as shown below.



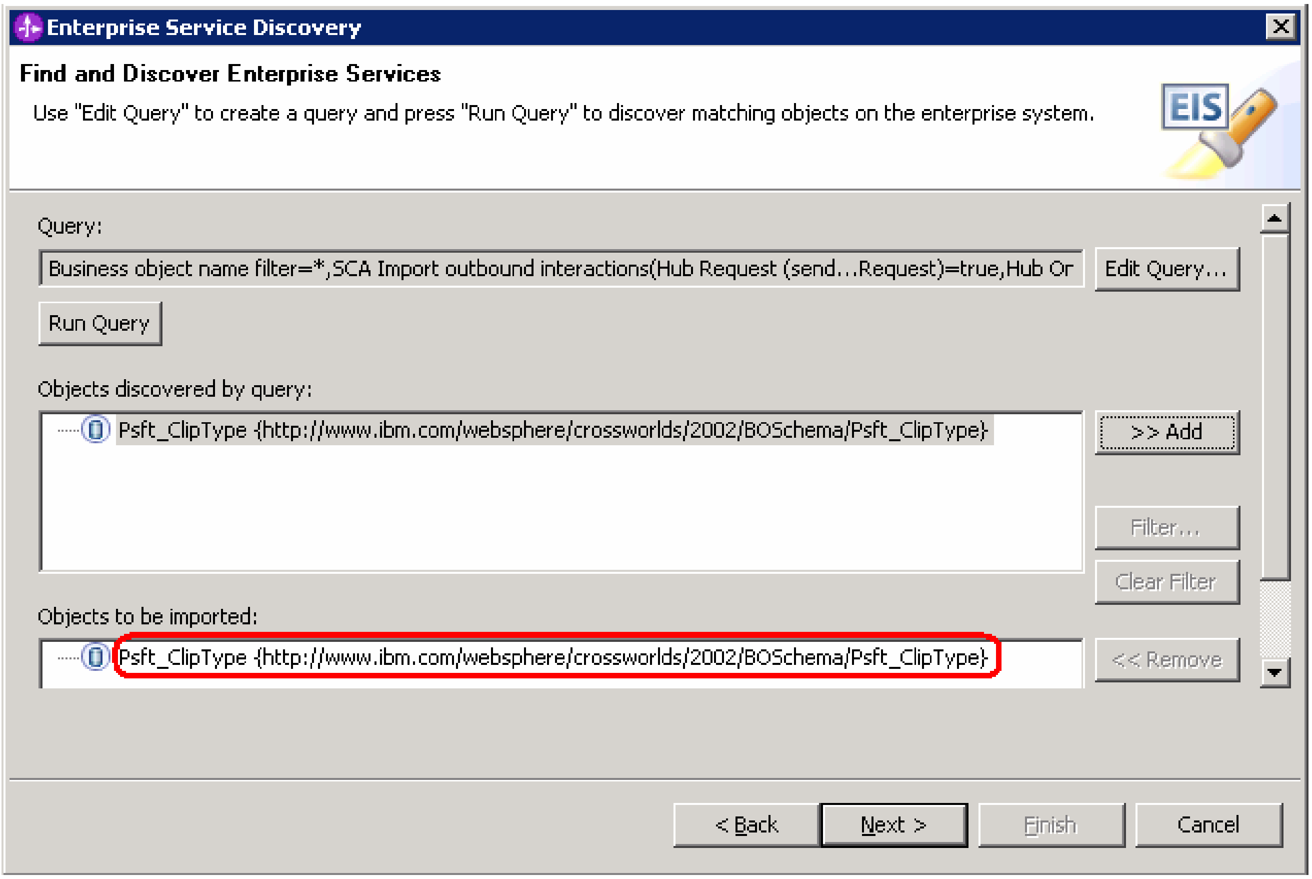
Enterprise Service Discovery

Configuration Parameters for Psft_ClipType {http://www.ibm.com/we

Set the configuration parameters, then press OK.
Configuration properties

□ sendPsft_ClipTypeRequest
□ sendPsft_ClipTypeOneWay
□ receivedPsft_ClipTypeRequest
☑ receivedPsft_ClipTypeDelivery

__ b. Click **OK** and you will see the following message under **Objects to be imported**:



Use "Edit Query" to create a query and press "Run Query" to discover matching objects on the enterprise system.

Query:

Business object name filter=*,SCA Import outbound interactions(Hub Request (send...Request)=true,Hub Or

Objects discovered by query:

Psft_ClipType {http://www.ibm.com/websphere/crossworlds/2002/BOSchema/Psft_ClipType}

Objects to be imported:

Psft_ClipType {http://www.ibm.com/websphere/crossworlds/2002/BOSchema/Psft_ClipType}

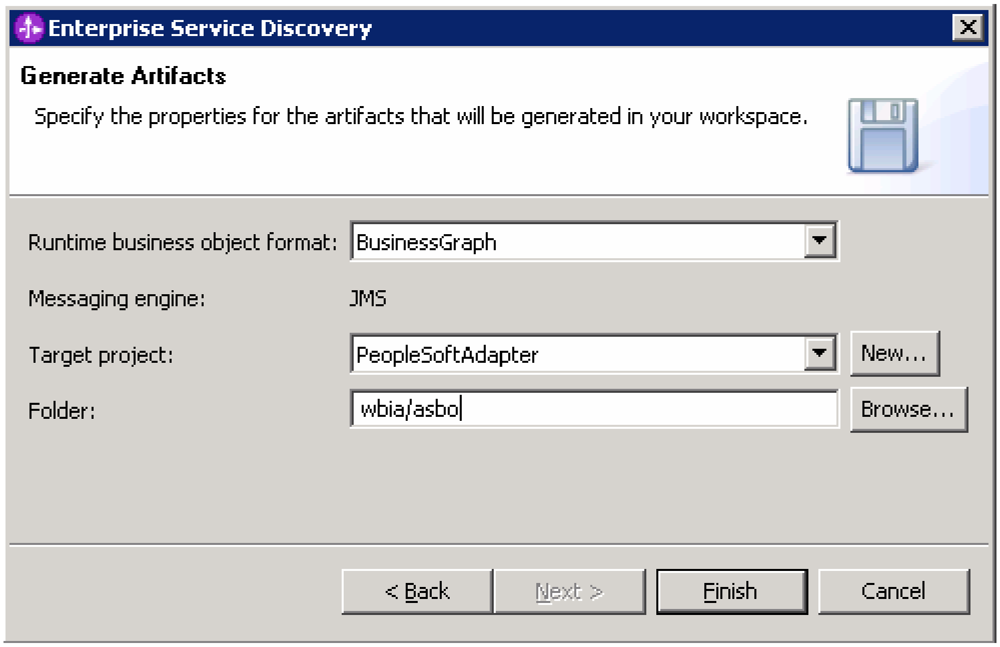
__ c. Click **Next.**

____ 5. Save the generated artifacts.

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

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

__ c. Enter the name of the folder as **wbia/asbo.**



Enterprise Service Discovery

Generate Artifacts

Specify the properties for the artifacts that will be generated in your workspace.

Runtime business object format: BusinessGraph

Messaging engine: JMS

Target project: PeopleSoftAdapter

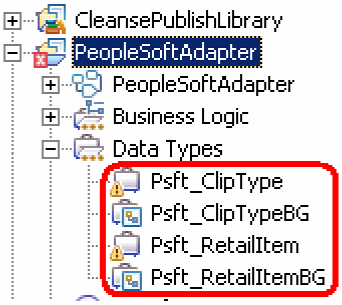
Folder: wbia/asbo

__ d. Click **Finish.**

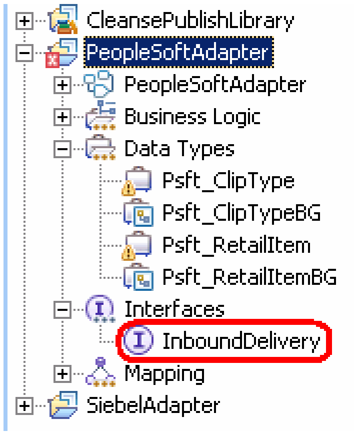
____ 6. Examine the generated artifacts by expanding the tree in the Business Integration view.

__ a. Expand **PeopleSoftAdapter.**

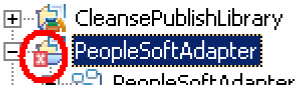
__ b. Expand **Data Types**. You will see four new business objects that provide the WebSphere Process Server form of the two business objects which were defined for use with the PeopleSoft adapter. For each of the adapters business objects, there is a business object generated with the same fields and 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.



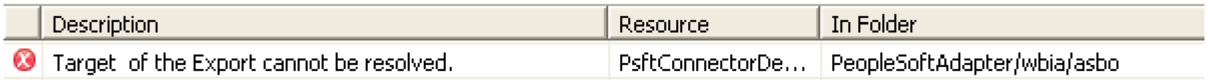
__ c. Expand **Interfaces**. You will see the InboundDelivery interface. This is the interface through which the Psft_ClipTypeBG will be passed into the server.



__ d. Expand **PeopleSoftAdapter**. 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 might notice that the **PeopleSoftAdapter** assembly is marked with an error. If so, it is because 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.



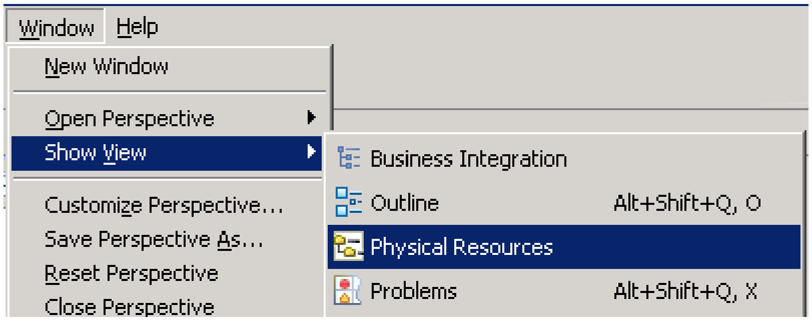
You might also see the error in the Problems view.



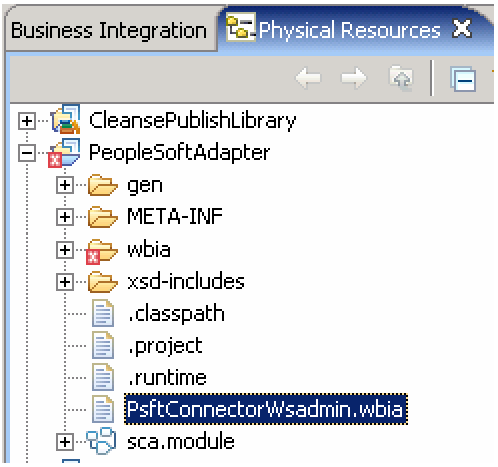
	Description	Resource	In Folder
⊗	Target of the Export cannot be resolved.	PsftConnectorDe...	PeopleSoftAdapter/wbia/asbo

____ 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 by selecting **Window > Show View > Physical Resources**.



__ b. Expand **PeopleSoftAdapter**. You will see the **PsftConnectorWsadmin.wbia**. This contains information about the queues that are needed to configure the server to communicate with the adapter.



__ c. If you want to examine these definitions, double-click on **PsftConnectorWsadmin.wbia** and the file will be opened in the text editor for you to examine.

____ 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.

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.

____ 9. Access the Enterprise Service Discovery wizard to generate the artifacts that will be used with the adapter for Siebel.

__ a. From the Business Integration perspective, right click on the **SiebelAdapter** module.

__ b. Select **New > Enterprise Service Discovery.**

____ 10. 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** dialog.

__ b. Click **Next.**

__ c. In the **Configure Settings for Discovery Agent** dialog, use the **Browse…** dialog or enter the fully qualified path to the connector configuration file as:

```
<LAB_FILES>\eXchange\Siebel\ASBOs\SiebelConnector.cfg.
```

and to the directory containing the ASBO definitions as:

```
<LAB_FILES>\eXchange\Siebel\ASBOs.
```

__ d. Click **Next.**

____ 11. Pick the specific objects from 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** box.

__ c. Click **>>Add** and you will be presented with the Configuration Parameters for Siebel_ProductType dialog.

____ 12. 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. For this scenario, only outbound requests from server to adapter with a response are needed. Therefore, ensure that only **sendSiebel_ProductTypeRequest** is checked.

__ b. Click **OK**.

__ c. Click **Next.**

____ 13. Save the generated artifacts.

__ 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.**

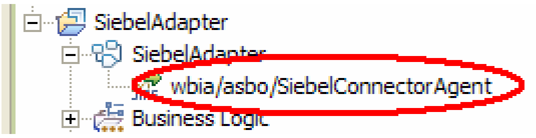
____ 14. Examine the artifacts that were generated for you by expanding the tree in the Business Integration view.

__ a. Expand **SiebelAdapter.**

__ b. Expand **Data Types** and examine the generated business objects.

__ c. Expand **Interfaces**. You will see the **OutboundRequest** interface. This is the interface through which the Siebel_ProductTypeBG will be passed out to the adapter.

__ d. Expand **SiebelAdapter**. 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.



____ 15. 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. Expand **SiebelAdapter**. You will see the **SiebelConnectorWsadmin.wbia** file. 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.

____ 16. Return to the Business Integration view in the upper left panel by clicking on the **Business Integration** tab at the top of the panel.

Part 3: Configure the Connector Configuration Files

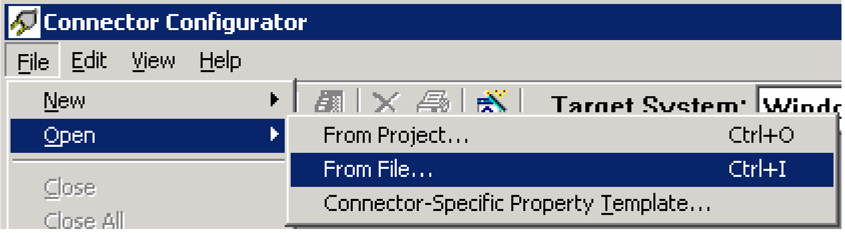
The PeopleSoft and Siebel adapters make use of a connector configuration file to define the configuration of the broker they will be communicating with. You already have the connector configuration files that were defined for use with a broker other than WebSphere Process Server. You used these configuration 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.**

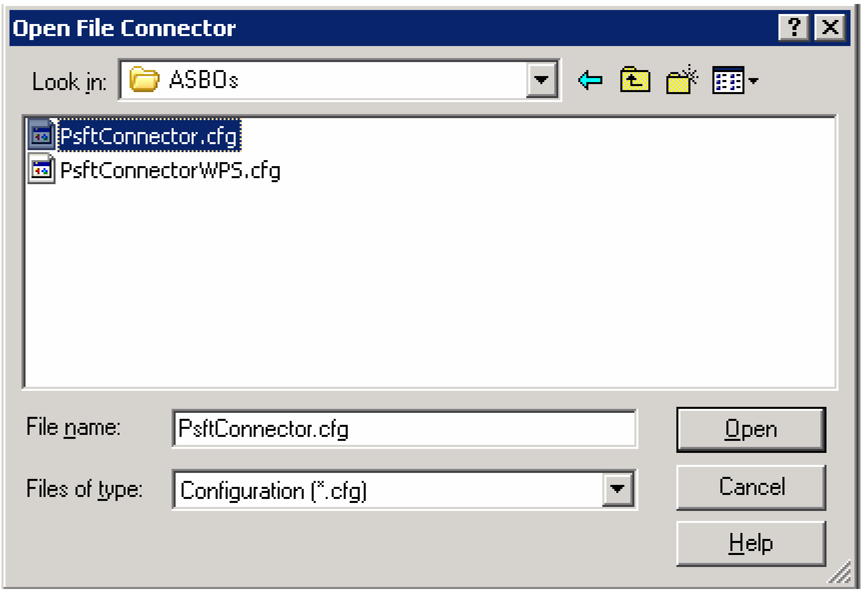
____ 2. Open the original connector configuration file for PeopleSoft which was used with the previous broker.

__ a. Select **File > Open > From File…** in the Connector Configurator.



__ b. Use the open dialog to navigate to **<LAB_FILES>\eXchange\PeopleSoft\ASBOs**.

__ c. Select **PsftConnector.cfg.**



File name: PsftConnector.cfg

Files of type: Configuration (*.cfg)

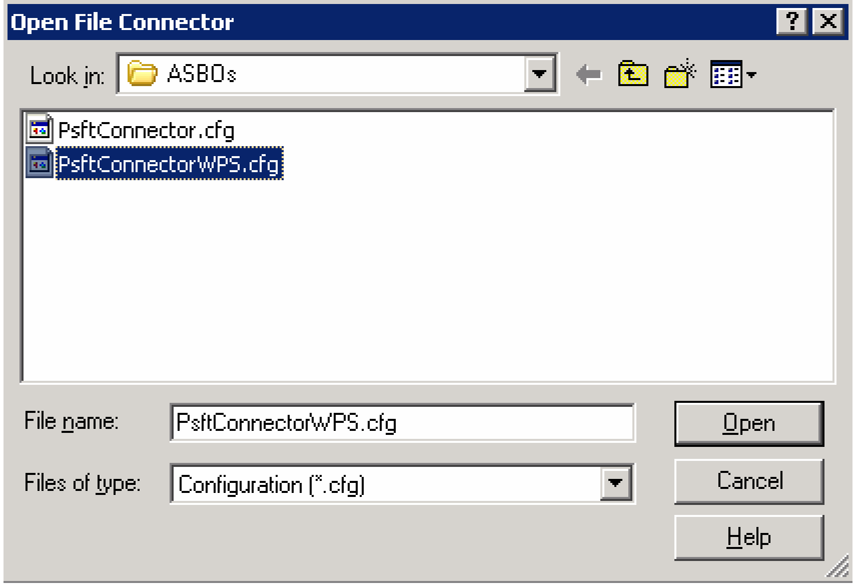
__ d. Click **Open.**

____ 3. Open the updated connector configuration file for PeopleSoft which will be used when communicating with WebSphere Process Server.

__ a. Select **File > Open > From File…** in the Connector Configurator.

__ b. Use the open dialog to navigate to **<LAB_FILES>\eXchange\PeopleSoft\ASBOs**.

__ c. Select **PsftConnectorWPS.cfg.**



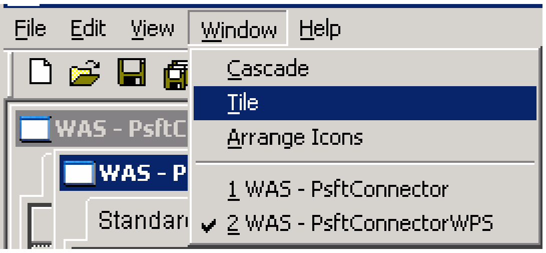
File name: PsftConnectorWPS.cfg

Files of type: Configuration (*.cfg)

__ d. Click **Open.**

____ 4. Arrange the two connector configuration files so they can be viewed together.

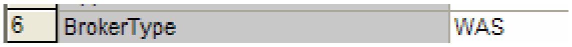
__ a. Select **Window > Tile** in the Connector Configurator.



____ 5. Examine the key fields in original and updated Connector Configuration files to see which values needed to be changed so that the adapter can connect to WebSphere Process Server.

__ a. **BrokerType** – The broker type must be set to **WAS** (in this case, the original broker type was also WAS, so this did not need to be changed).

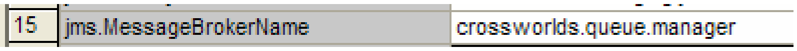
1) Old and New value.



6	BrokerType	WAS

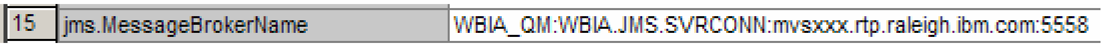
__ b. **jms.MessageBrokerName** – The name of the message broker must be **WBIA_QM:WBIA.JMS.SVRCONN:<HOSTNAME>:5558**.

1) Old value



15	jms.MessageBrokerName	crossworlds.queue.manager

2) New value

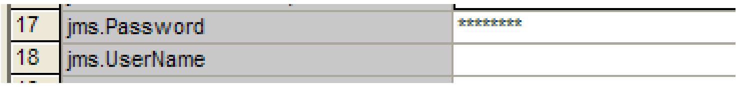


15	jms.MessageBrokerName	WBIA_QM:WBIA.JMS.SVRCONN:mvsxxx.rtp.raleigh.ibm.com:5558

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 hostname and the configured SIB_MQ_ENDPOINT_ADDRESS.

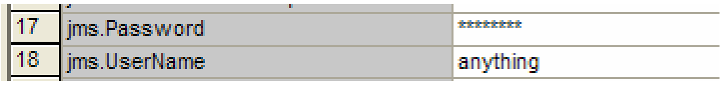
__ c. **jms.UserName** and **jms.Password** – This is the username and password used to authenticate for JMS communication with WebSphere Process Server.

1) Old value



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

2) New value

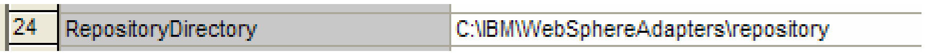


17	jms.Password	********
18	jms.UserName	anything

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.

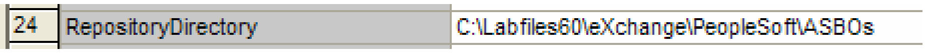
__ d. **RepositoryDirectory** – The repository directory must point to where the business objects used by the adapter are located, in this case **<LAB_FILES>\eXchange\PeopleSoft\ASBOs.**

1) Old value



24	RepositoryDirectory	C:\IBM\WebSphereAdapters\repository

2) New value



24	RepositoryDirectory	C:\Labfiles60\eXchange\PeopleSoft\ASBOs

____ 6. **Optional:** Examine the equivalent changes have been made to the connector configuration file for Siebel.

__ 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
jms.MessageBrokerName
jms.UserName
jms.Password
RepositoryDirectory
```

____ 7. Close the Connector Configurator window.

__ a. Select **File > Exit**.

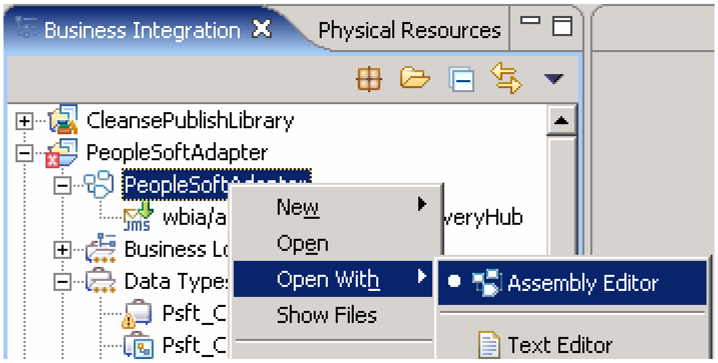
Part 4: Assemble and Test the PeopleSoftAdapter Module

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.

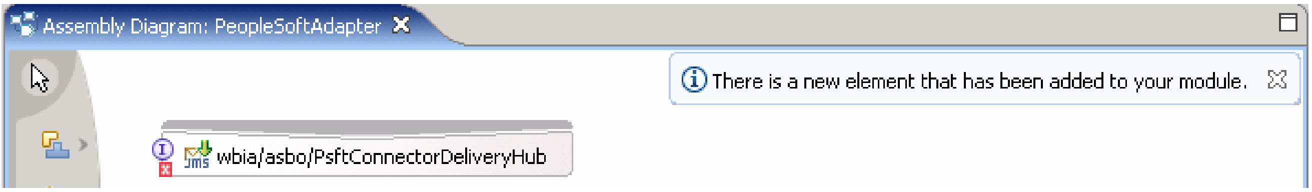
____ 1. Open the PeopleSoftAdapter in the Assembly Editor.

__ a. Expand the **PeopleSoftAdapter** sub-tree In the Business Integration view by clicking on the **+**.

__ b. Right click on the **PeopleSoftAdapter** module assembly and select **Open With > Assembly Editor.**



__ c. In the Assembly Editor, you will see a message that there is a new element in the module and also 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.

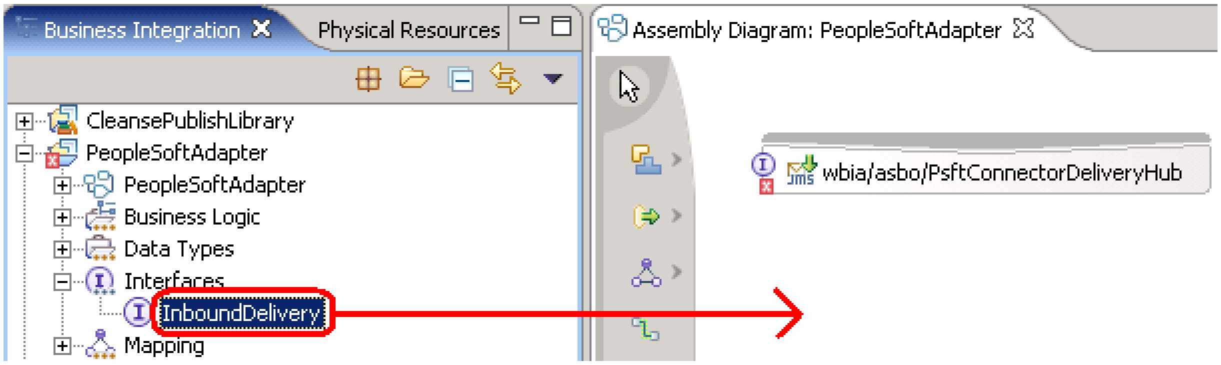

Assembly Diagram: PeopleSoftAdapter
wbia/asbo/PsftConnectorDeliveryHub

There is a new element that has been added to your module.

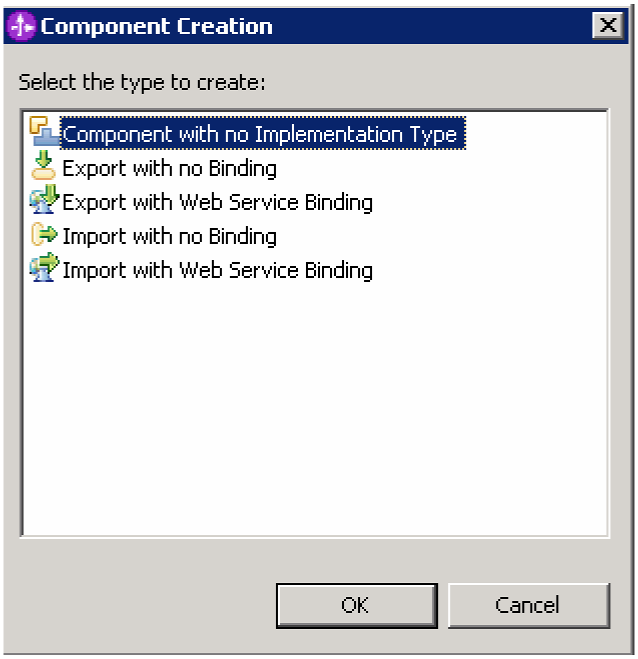
____ 2. The Test Component cannot examine the data coming in through an Export unless the Export is wired to something. Therefore, you need to create a component that the Export can be wired to. To do this, the component must support the same interface as the Export. However, the component does not need 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. Expand the interfaces by clicking on the **+** in front of **Interfaces**.

__ b. Drag the **InboundDelivery** interface to the assembly editor by clicking on it and dragging it while you hold down the mouse button.



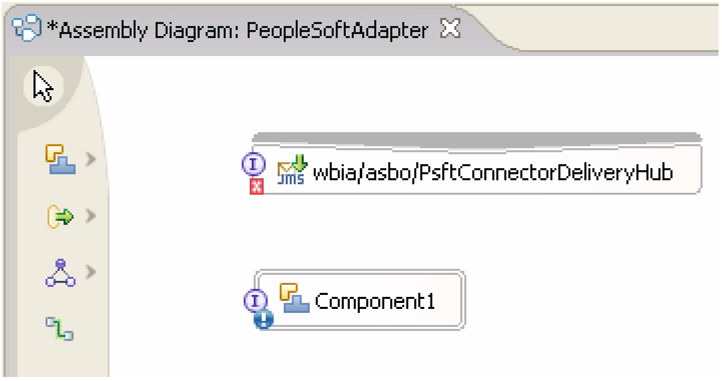
__ c. Release the mouse button to drop it on the assembly editor and you will see the Component Creation dialog.



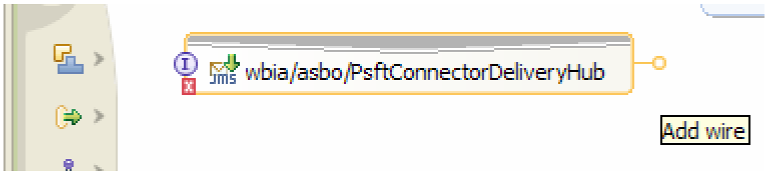
Select the type to create:

- Component with no Implementation Type
- Export with no Binding
- Export with Web Service Binding
- Import with no Binding
- Import with Web Service Binding

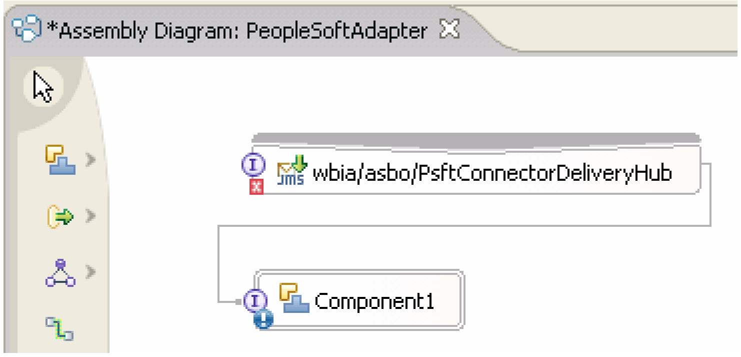
__ d. Select **Component with No Implementation Type** and click **OK**. The assembly editor should now look similar to the following screen capture:


*Assembly Diagram: PeopleSoftAdapter
wbia/asbo/PsftConnectorDeliveryHub
Component1

__ e. Place the mouse pointer (hover) over the far right side of the Export. This will cause the Add Wire construct appear.



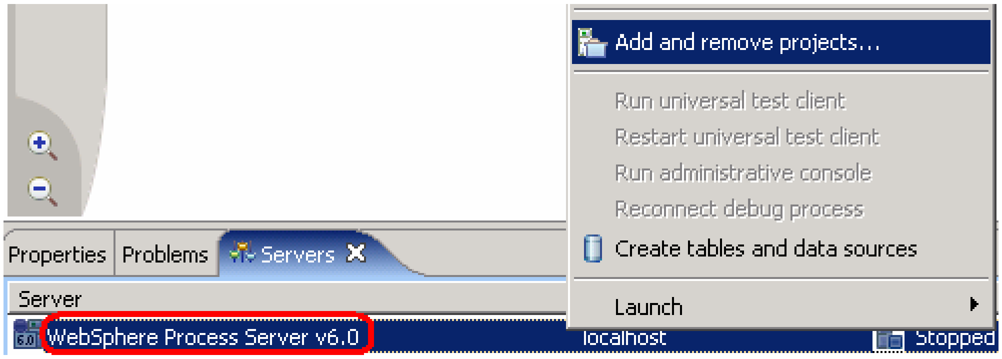
__ f. Draw a wire between the Export and the Component by clicking and holding the mouse button on the circle of the Add Wire construct and dragging it to the circle containing the Ⓘ on the Component1. The result will look like this:


*Assembly Diagram: PeopleSoftAdapter
wbia/asbo/PsftConnectorDeliveryHub
Component1

__ g. Click **Ctrl+S** to save the wiring diagram.

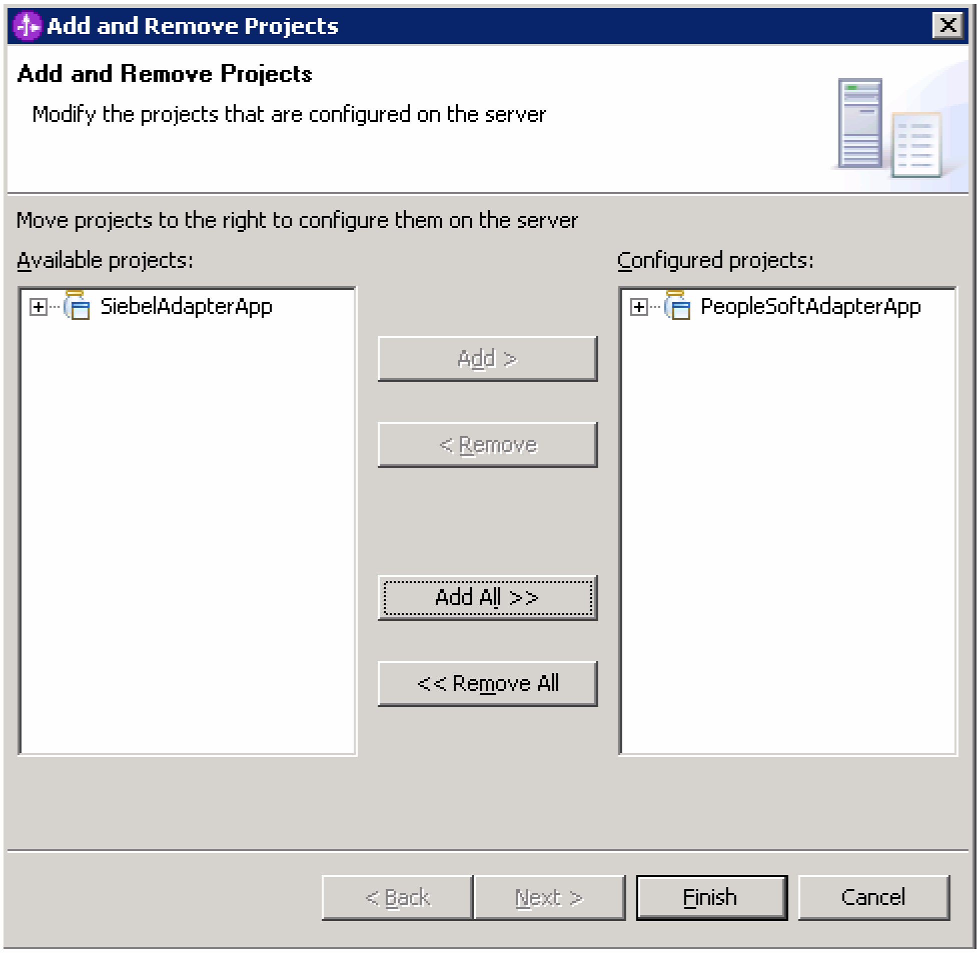
____ 3. Add the PeopleSoftAdapter module to the UTE server so that it can be tested.

__ a. From the Server view, right click on the server and select **Add and Remove Projects…** from the popup menu.


Add and remove projects...
Run universal test client
Restart universal test client
Run administrative console
Reconnect debug process
Create tables and data sources
Launch
Properties
Problems
Servers
Server
WebSphere Process Server v6.0
localhost
Stopped

__ b. In the Add and Remove Projects dialog, select the **PeopleSoftAdapterApp** project from the Available projects panel.

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



Add and Remove Projects

Add and Remove Projects

Modify the projects that are configured on the server

Move projects to the right to configure them on the server

Available projects:

- SiebelAdapterApp

Configured projects:

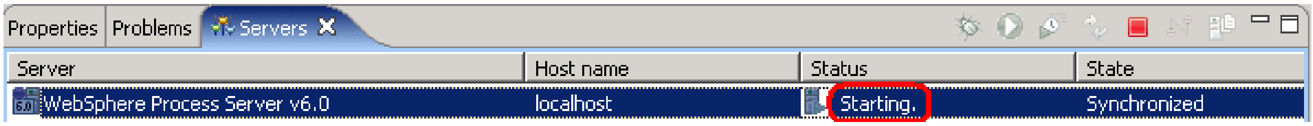
- PeopleSoftAdapterApp

__ d. Click **Finish.**

__ e. 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 add a server to the WebSphere Integration Developer test environment and start it. This is especially true for z/OS, AIX, Solaris remote test environment, where the WebSphere Integration Developer will be remote to the test environment.

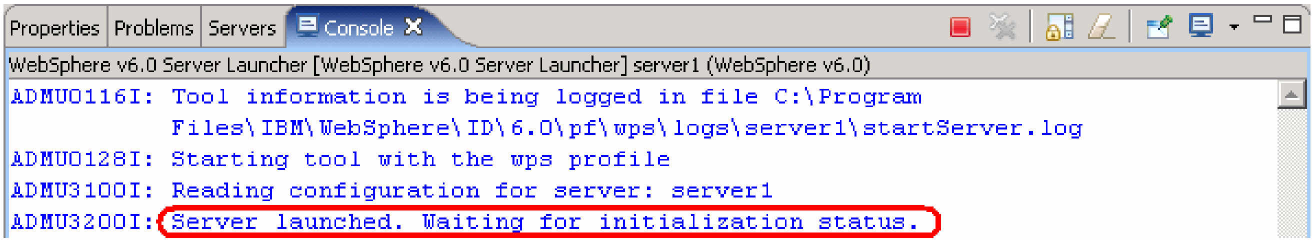
If using a local testing environment, the server will automatically begin to start so that the module can be added to the server.

__ f. You will at first see the following in the Servers view in the lower right panel.



Server	Host name	Status	State
WebSphere Process Server v6.0	localhost	Starting.	Synchronized

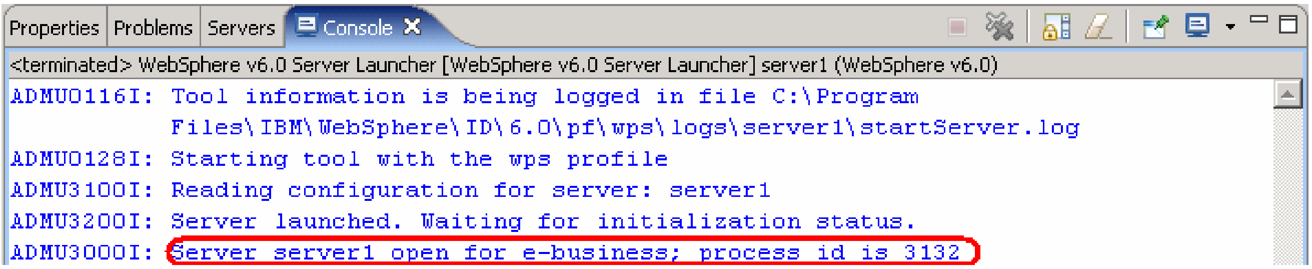
__ g. You will then see the Console view appear with something similar to the following:



WebSphere v6.0 Server Launcher [WebSphere v6.0 Server Launcher] server1 (WebSphere v6.0)

```
ADMU0116I: Tool information is being logged in file C:\Program
           Files\IBM\WebSphere\ID\6.0\pf\wps\logs\server1\startServer.log
ADMU0128I: Starting tool with the wps profile
ADMU3100I: Reading configuration for server: server1
ADMU3200I: Server launched. Waiting for initialization status.
```

__ h. Wait for the server to complete startup. This may take several minutes. Look for the “Server server1 open for e-business” message in the console.



<terminated> WebSphere v6.0 Server Launcher [WebSphere v6.0 Server Launcher] server1 (WebSphere v6.0)

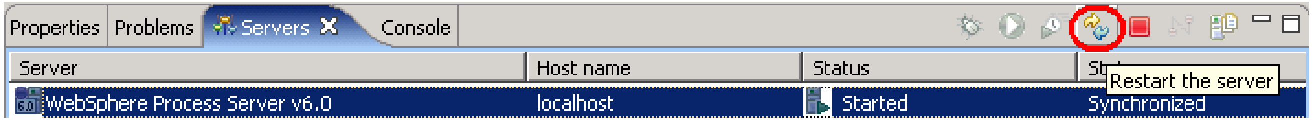
```
ADMU0116I: Tool information is being logged in file C:\Program
           Files\IBM\WebSphere\ID\6.0\pf\wps\logs\server1\startServer.log
ADMU0128I: Starting tool with the wps profile
ADMU3100I: Reading configuration for server: server1
ADMU3200I: Server launched. Waiting for initialization status.
ADMU3000I: Server server1 open for e-business; process id is 3132
```

____ 4. The server must be restarted at this point for the messaging engine to be fully initialized and be ready to receive input from the Visual Test Connector.

NOTE: The requirement for a server restart after the installation of an application containing a WBI Adapter Import or Export is a one time requirement. After this has been done the first time, the required messaging configuration is defined to the server. Additional applications using WBI Adapters can be installed without requiring a server restart. Therefore, if you have already done the End-To-End lab, you do not have to do this server restart.

__ a. 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.

If using a local testing environment, click the **Restart the server** icon.



Server	Host name	Status	St
WebSphere Process Server v6.0	localhost	Started	Synchronized

__ b. Wait for the server restart to complete.

____ 5. Connect the Visual Test Connector to the UTE Server.

__ a. Using the connector configuration at **<LAB_FILES>\eXchange\PeopleSoft\ASBOs\PsftConnectorWPS.cfg** follow the instructions below to use the Visual Test Connector.

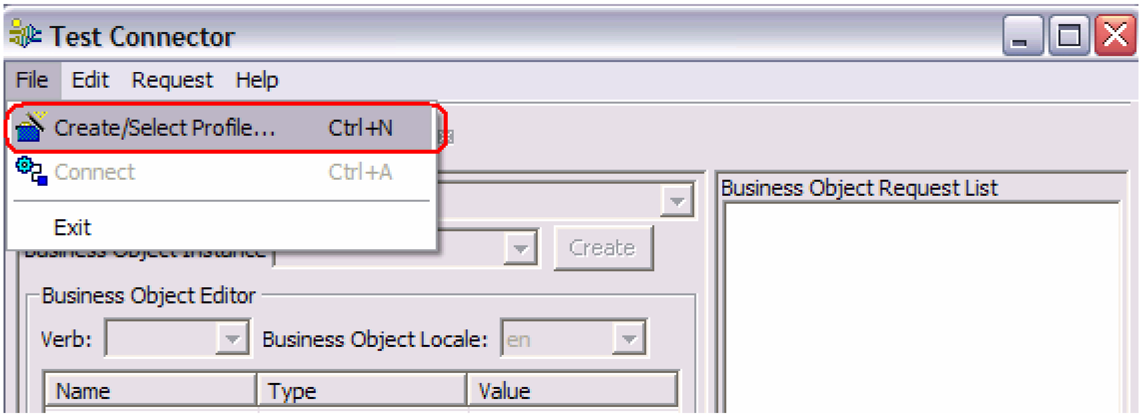
____ 6. Ensure the WebSphere Process Server is active (the server is the JMS Broker). The VTC requires an active JMS Broker to start its connection.

____ 7. Start the Visual Test Connector.

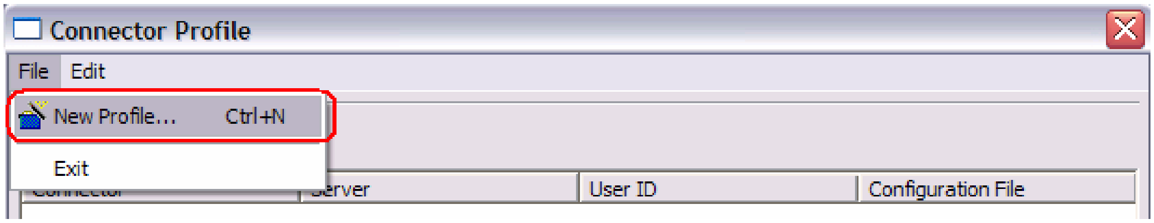
__ a. From the **start** menu, select **Programs > IBM WebSphere Business Integration Adapters > IBM WebSphere Business Integration Toolset > Visual Test Connector** (VTC). This opens a command window in the background where the process runs and the Visual Test Connector Interface in the foreground. Do not close the command window as this will cause the VTC to exit.

____ 8. Create/Select a Profile.

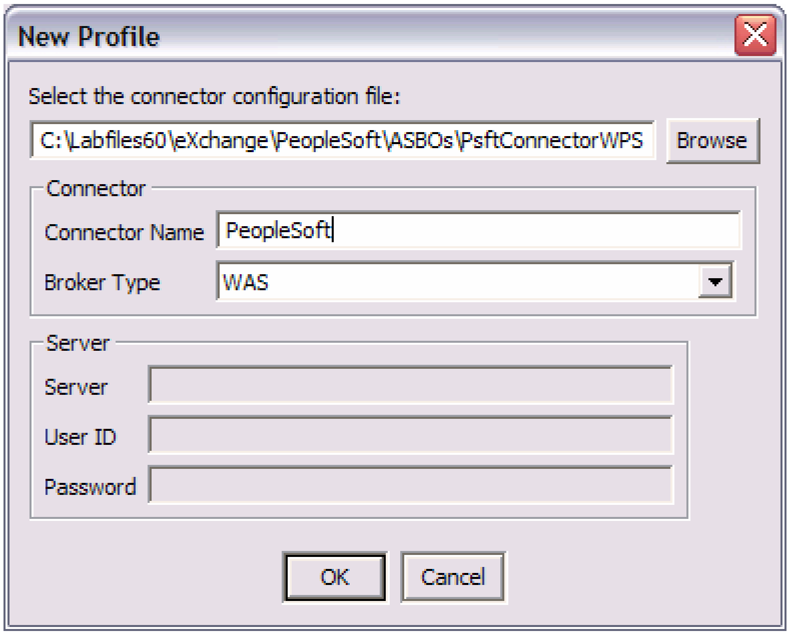
__ a. From the VTC interface, select **File > Create/Select Profile.**



__ b. If a profile does not already exist, create one by selecting **File > New Profile …** If a profile does already exist skip to step d.



__ c. Browse to and select the connector configuration file, type in a Connector Name for the Profile, and select WAS for the Broker Type. Click **OK**.

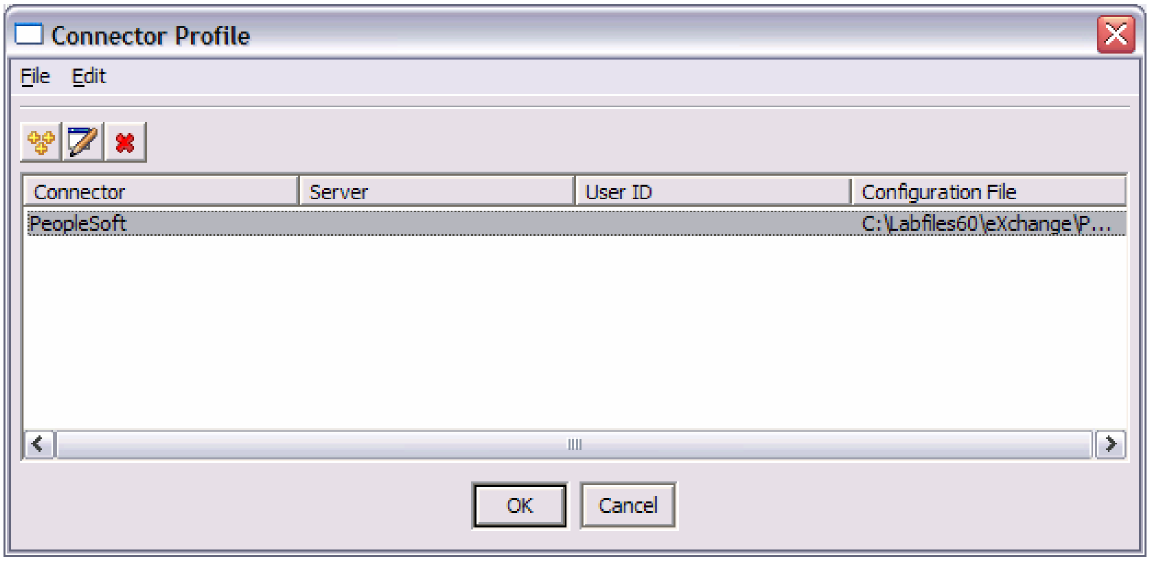

New Profile

Select the connector configuration file: C:\Labfiles60\eXchange\PeopleSoft\ASBOs\PsftConnectorWPS

Connector Name: PeopleSoft

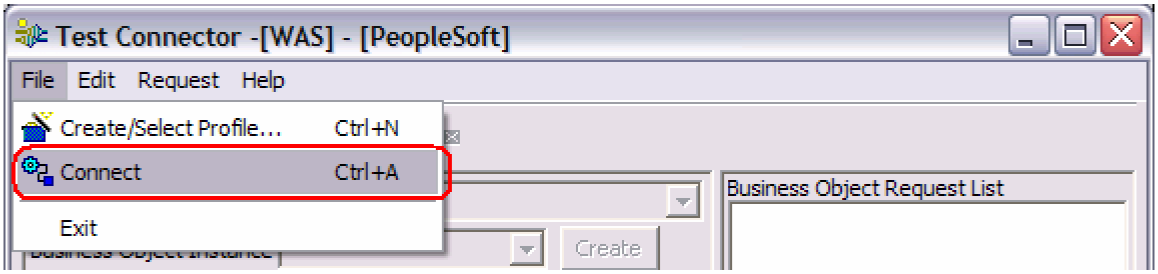
Broker Type: WAS

__ d. Select and highlight the profile and click **OK**



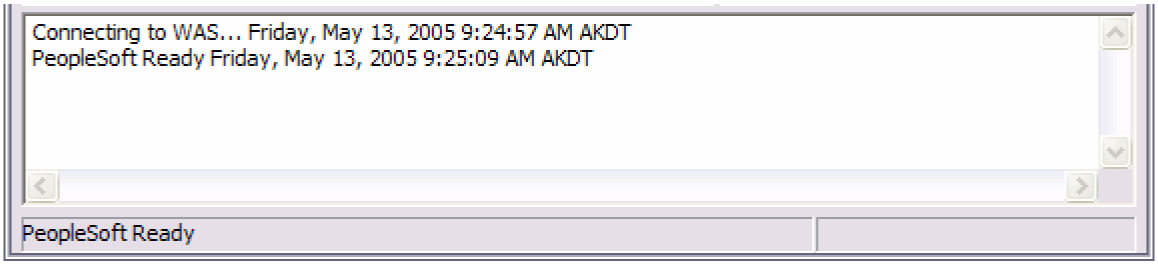
Connector	Server	User ID	Configuration File
PeopleSoft			C:\Labfiles60\eXchange\P...

__ e. Select **File > Connect.**



Test Connector -[WAS] - [PeopleSoft]
File Edit Request Help
Create/Select Profile... Ctrl+N
Connect Ctrl+A
Exit
Business Object Request List
Create

__ f. View messages in the lower window.

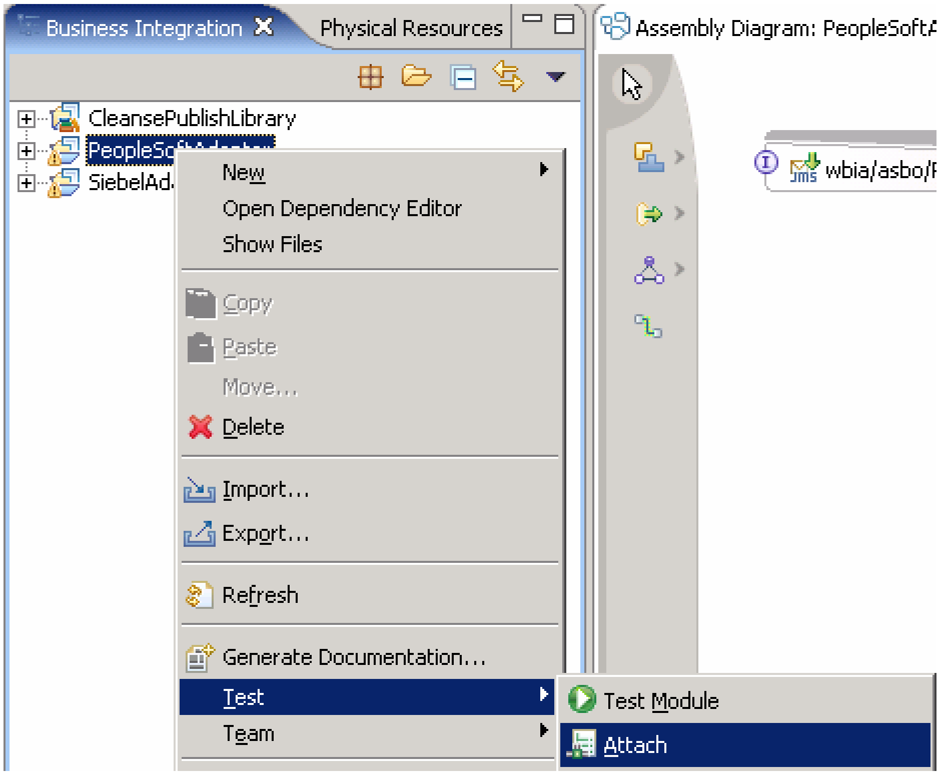


```
Connecting to WAS... Friday, May 13, 2005 9:24:57 AM AKDT
PeopleSoft Ready Friday, May 13, 2005 9:25:09 AM AKDT
```

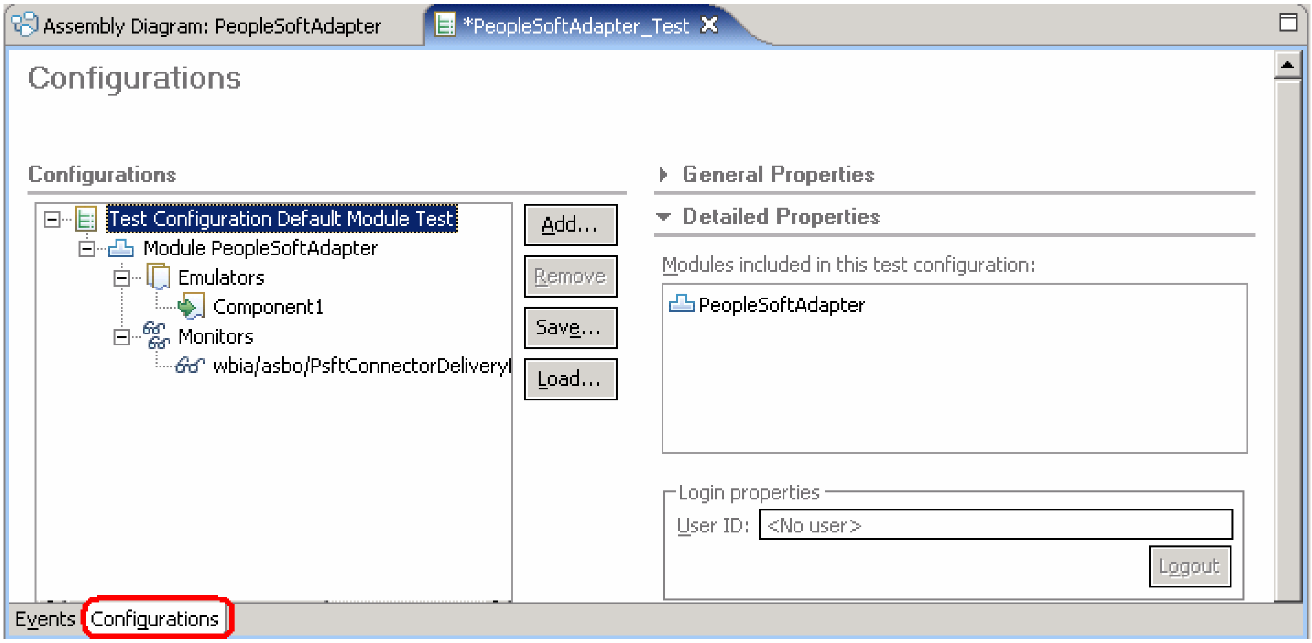
PeopleSoft Ready

____ 9. Attach the test component to the PeopleSoftAdapter module.

__ a. In the Business Integration view, right click on the **PeopleSoftAdapter** and select **Test > Attach.**



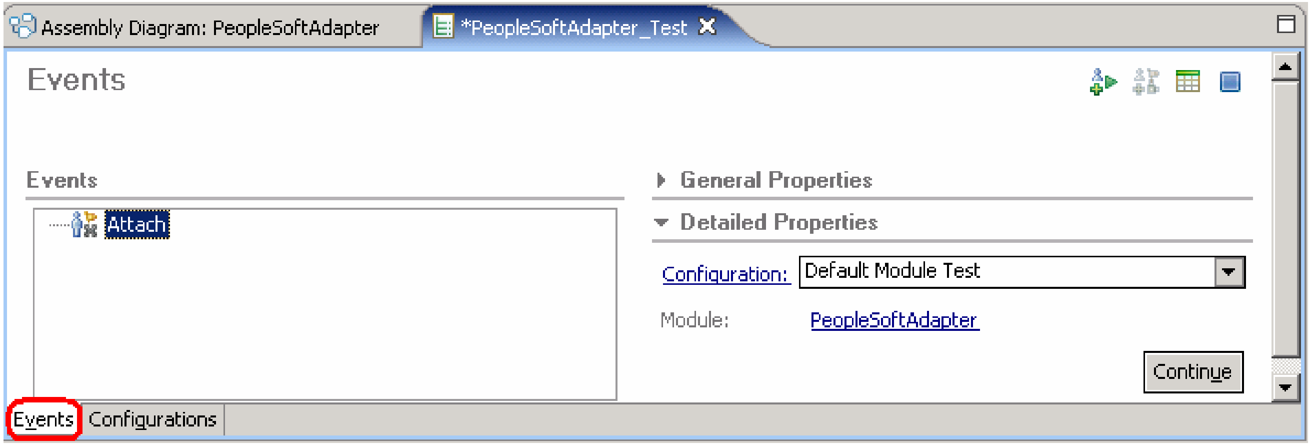
__ b. Click on the **Configurations** tab at the bottom of the **Testing Module: PeopleSoftAdapter** view. You should see something like the following:



__ 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.

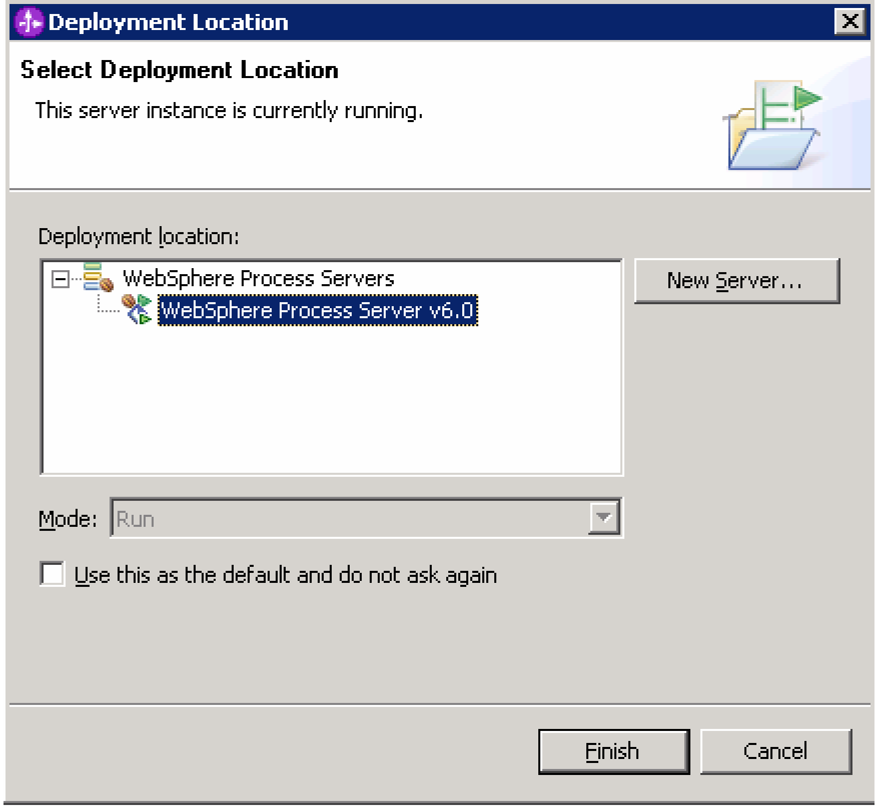
__ d. Notice that **PsftConnectorDeliveryHub.<export> -> Component1** is listed under Monitors. This means that anything that flows over that wire will result in an event which can then be interrogated to see what passed through the wire.

__ 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 on **Continue**.

__ g. Select the **WebSphere Process Server v6.0** server in the **Choose a deployment location dialog**.



Deployment Location

Select Deployment Location

This server instance is currently running.

Deployment location:

WebSphere Process Servers

WebSphere Process Server v6.0

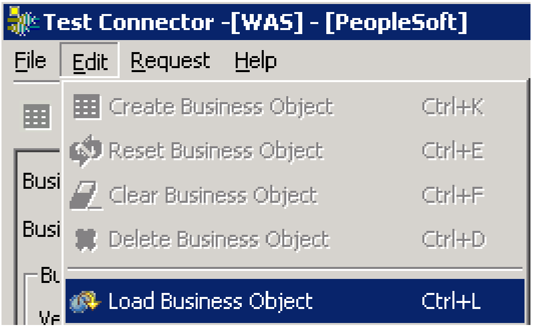
Mode: Run

□ Use this as the default and do not ask again

__ h. Click **Finish.**

_____ 10. Load a predefined business object into the Visual Test Connector.

__ a. Select **Edit > Load Business Object** in the VTC window.



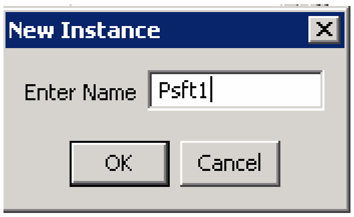
Test Connector -[WAS] - [PeopleSoft]

File
Edit
Request
Help
Create Business Object Ctrl+K
Reset Business Object Ctrl+E
Clear Business Object Ctrl+F
Delete Business Object Ctrl+D
Load Business Object Ctrl+L
Busi
Busi
Bu
Ve

__ b. In the Open dialog, navigate to **c:\Labfiles60\eXchange\PeopleSoft\ASBOInstances** and select **PsftASBO1.bo**.

__ c. Click **Open.**

__ d. Enter a name to be assigned to this instance, such as **Psft1**.



Enter Name Psft1

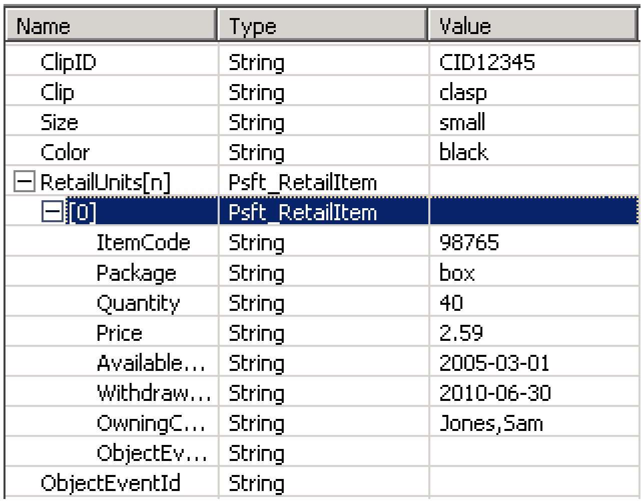
__ e. Click **OK.**

_____ 11. Examine the data in the business object.

__ a. Look at the values assigned to the Psft_ClipType fields.

__ b. Expand **RetailUnits[n]** to expand the Psft_RetailItem array.

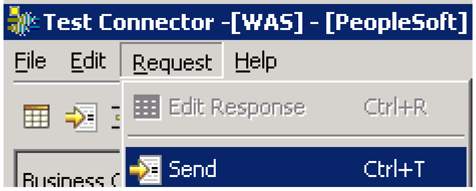
__ c. Expand **[0]** to see an instance of Psft_RetailItem.



Name	Type	Value
ClipID	String	CID12345
Clip	String	clasp
Size	String	small
Color	String	black
⊟ RetailUnits[n]	Psft_RetailItem	
⊟ [0]	Psft_RetailItem	
ItemCode	String	98765
Package	String	box
Quantity	String	40
Price	String	2.59
Available...	String	2005-03-01
Withdraw...	String	2010-06-30
OwningC...	String	Jones,Sam
ObjectEv...	String	
ObjectEventId	String	

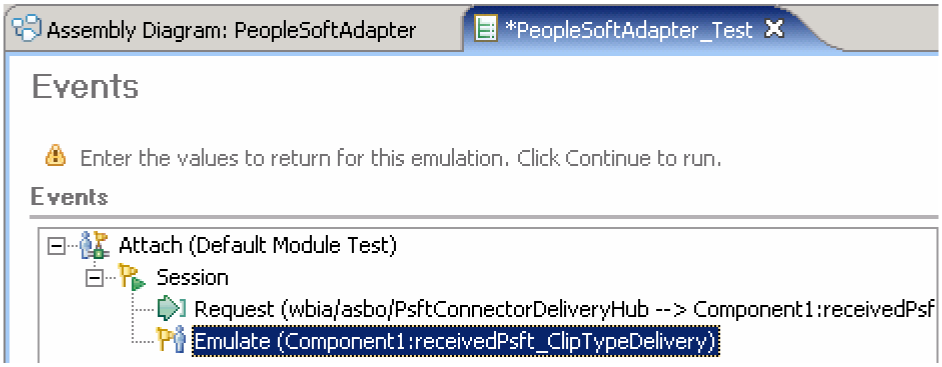
_____ 12. Send the business object to the server.

__ a. Select **Request > Send.**



_____ 13. Examine the resulting 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.



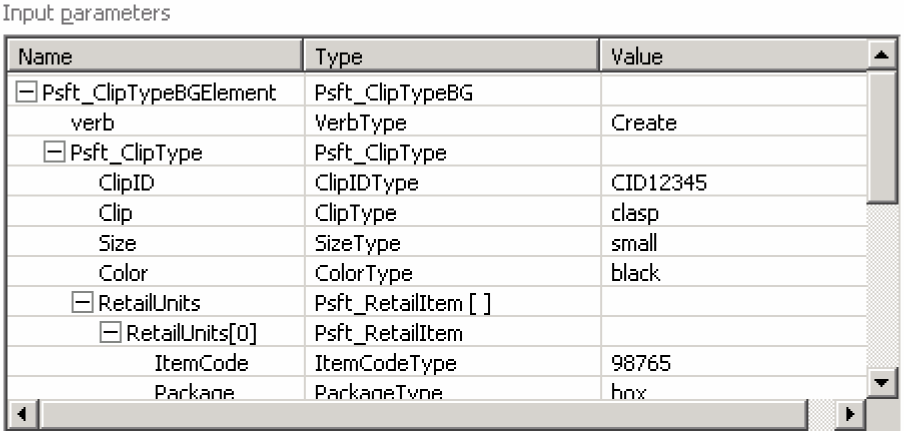
Events

Enter the values to return for this emulation. Click Continue to run.

Events

- Attach (Default Module Test)
  - Session
    - Request (wbia/asbo/PsftConnectorDeliveryHub --> Component1:receivedPsf
    - Emulate (Component1:receivedPsft_ClipTypeDelivery)

__ 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.



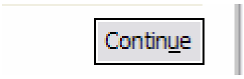
Input parameters

Name	Type	Value
Psft_ClipTypeBGElement	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
Package	PackageType	box

__ 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.



_____ 14. Stop and close the test component.

__ a. Click the **Stop** icon at the top right of the Events pane.

__ b. Click **X** to Exit **PeopleSoftAdapter_Test** .

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

_____ 15. Close the Visual Test Connector.

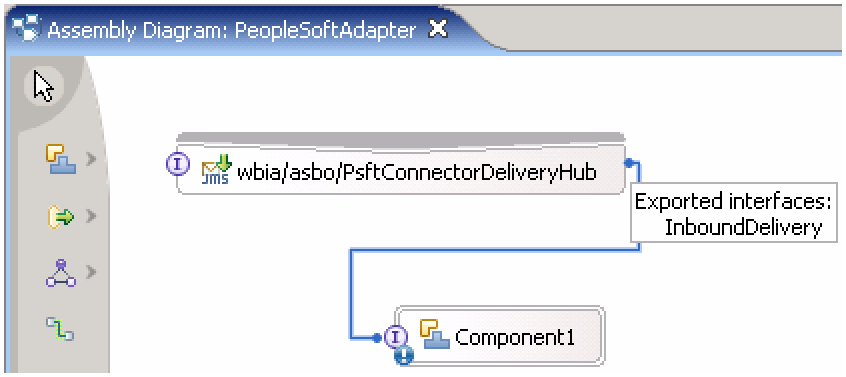
__ a. Select **File > Exit** in the VTC window.

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

__ c. Press the **enter** key In the Visual Test Connector command window to close it.

____ 16. Remove the unimplemented component. It is not needed in subsequent labs.

__ a. Click on the **wire** between PsftConnectorDeliveryHub and Component1 to select the wire. The result should look like this:


Assembly Diagram: PeopleSoftAdapter
wbia/asbo/PsftConnectorDeliveryHub
Exported interfaces:
InboundDelivery
Component1

__ b. Press the **Delete** key.

__ c. Right click on **Component1** and select **Delete** from the menu.

__ d. Press **Ctrl-s** to save the Assembly Diagram.

__ e. Close the Assembly Diagram.

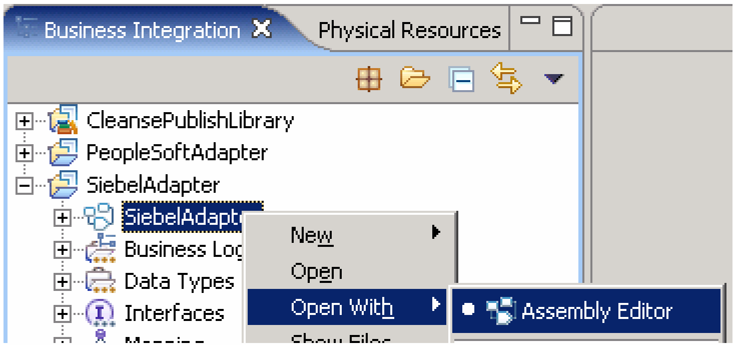
Part 5: Assemble and Test the SiebelAdapter Module

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.

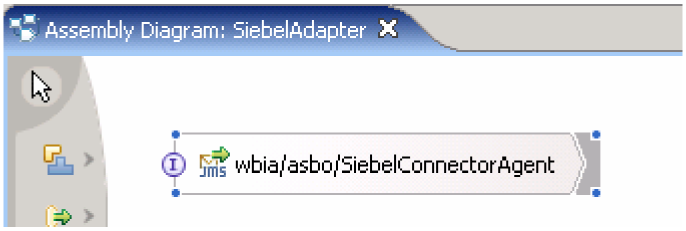
____ 1. Open the SiebelAdapter in the Assembly Editor.

In the Business Integration view, expand **SiebelAdapter.**

__ a. Right click on the **SiebelAdapter** module assembly and select **Open With > Assembly 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.


Assembly Diagram: SiebelAdapter
wbia/asbo/SiebelConnectorAgent

____ 2. Remove the PeopleSoftAdapter module and add the SiebelAdapter module to the UTE server so that the SiebelAdapter can be tested.

__ a. Right click on the server from the Server view and select **Add and Remove Projects…** from the popup menu.

__ b. Select the **PeopleSoftAdapterApp** project from the Configured projects list.

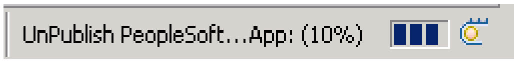
__ c. Click < **Remove**.

__ d. Select the **SiebelAdapterApp** project from the Available projects panel.

__ e. Click **Add** > to add it to the Configured projects panel.

__ f. Click **Finish.**

__ g. Wait for the server to complete the Add and Remove Projects processing (see messages in lower right hand corner of the window.


UnPublish PeopleSoft…App: (10%)

____ 3. Connect the Visual Test Connector to the UTE Server using the connector configuration at **<LAB_FILES>\eXchange\Siebel\ASBOs\SiebelConnectorWPS.cfg.**

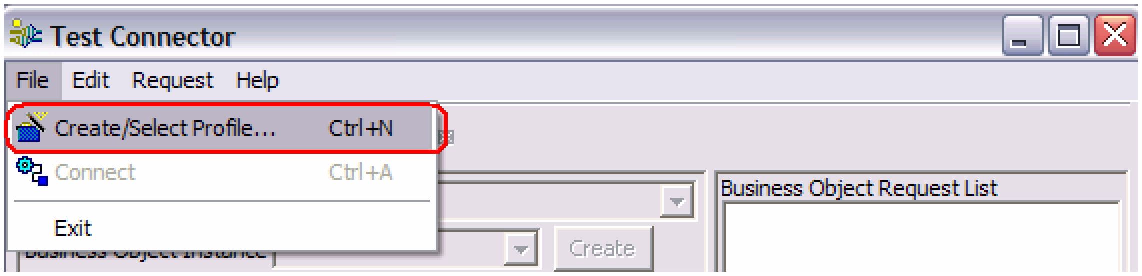
____ 4. Ensure the WebSphere Process Server is active (the server is the JMS Broker). The VTC requires an active JMS Broker to start its connection.

____ 5. Start the Visual Test Connector.

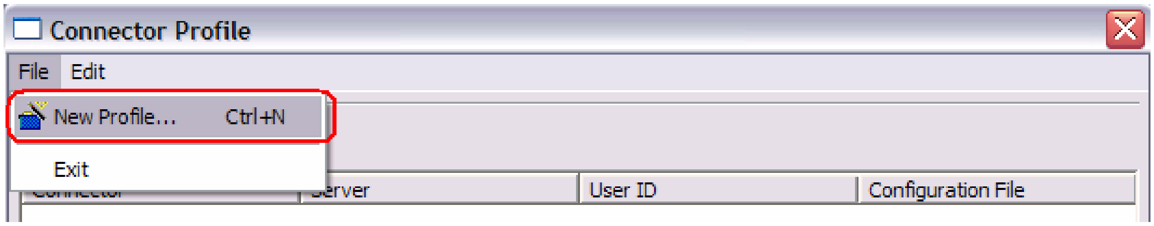
__ a. From the **start** menu, select **Programs > IBM WebSphere Business Integration Adapters > IBM WebSphere Business Integration Toolset > Visual Test Connector** (VTC). This opens a command window in the background where the process runs and the Visual Test Connector Interface in the foreground. Do not close the command window as this will cause the VTC to exit.

____ 6. Create/Select a Profile.

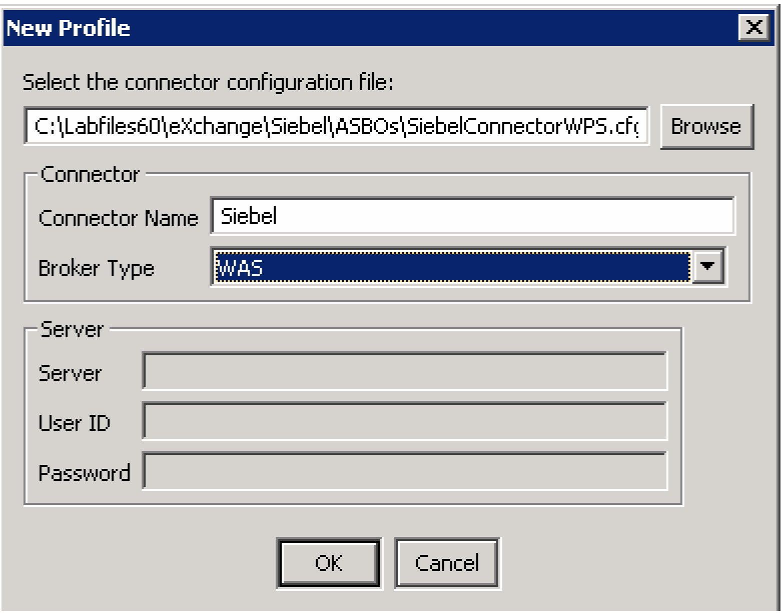
__ a. From the VTC interface, select **File > Create/Select Profile.**



__ b. If a profile does not already exist, create one by selecting **File > New Profile…** if a profile does already exist skip to step d.



__ c. Browse to and select the connector configuration file, type in a Connector Name for the Profile, and select WAS for the Broker Type.   Click OK.



New Profile

Select the connector configuration file: C:\Labfiles60\eXchange\Siebel\ASBOs\SiebelConnectorWPS.cfg

Connector

Connector Name: Siebel

Broker Type: WAS

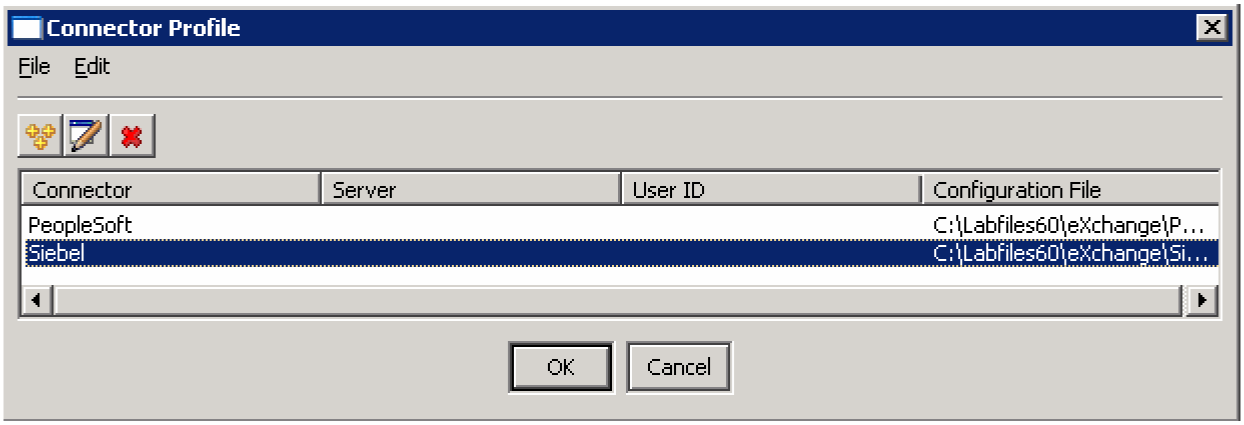
Server

Server:

User ID:

Password:

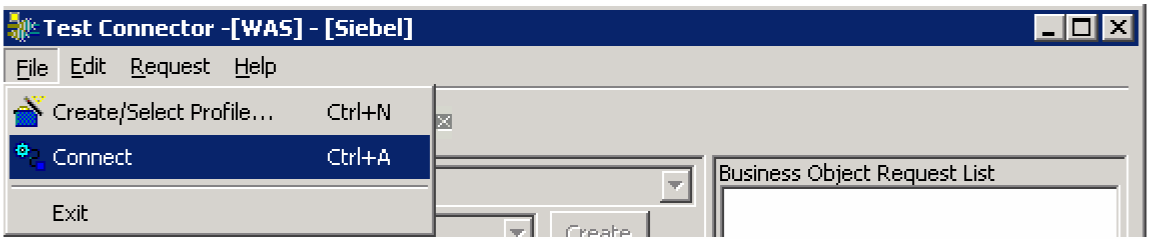
__ d. Select and highlight the profile and click **OK.**



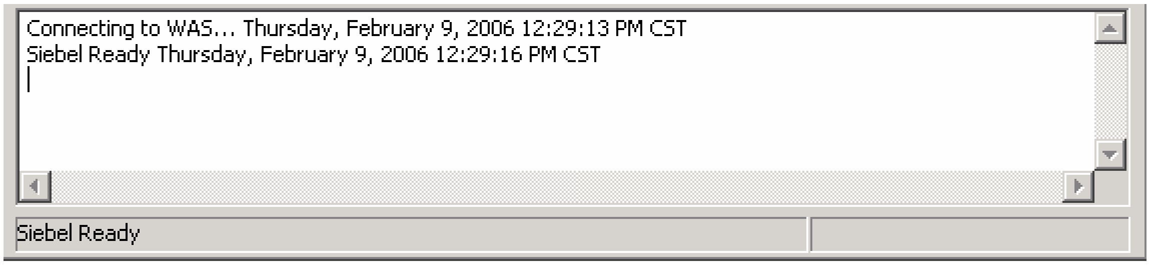
Connector Profile

Connector	Server	User ID	Configuration File
PeopleSoft			C:\Labfiles60\eXchange\P…
Siebel			C:\Labfiles60\eXchange\Si…

__ e. Select **File > Connect.**


Test Connector -[WAS] - [Siebel]
File
Edit
Request
Help
Create/Select Profile... Ctrl+N
Connect Ctrl+A
Exit
Business Object Request List
Create

__ f. View messages in the lower window.



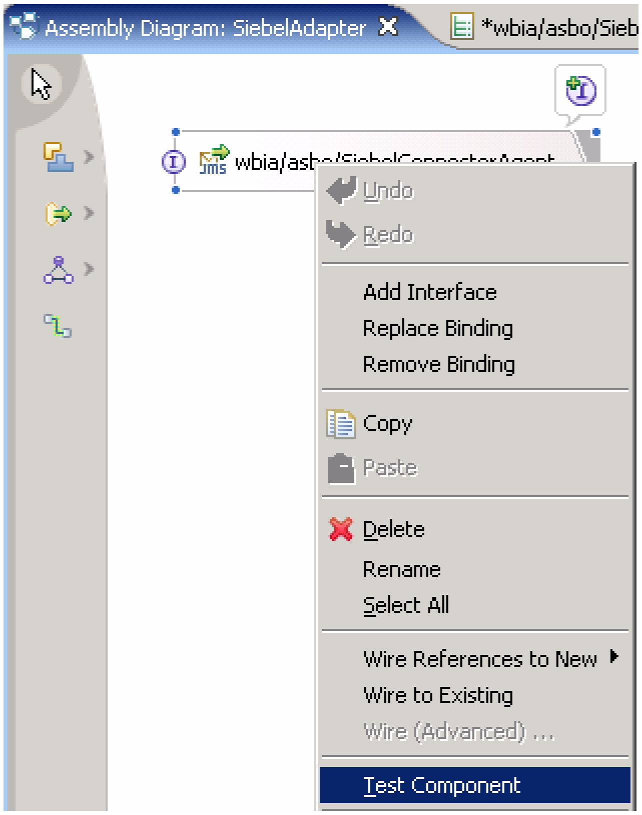
```
Connecting to WAS... Thursday, February 9, 2006 12:29:13 PM CST
Siebel Ready Thursday, February 9, 2006 12:29:16 PM CST
```

Siebel Ready

____ 7. Open the test component.

__ a. Right click on the **SiebelConnectorAgent** import in the Assembly editor.

__ b. Select **Test Component** from the pop-up menu.



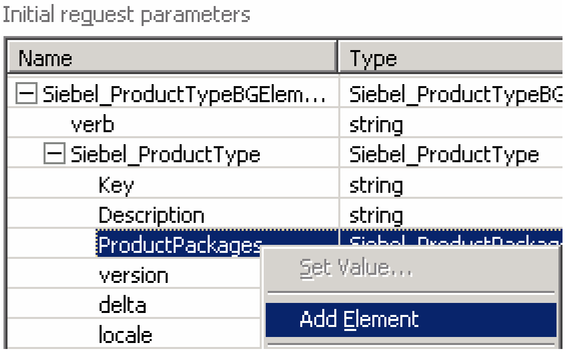
____ 8. Fill in the data fields of the Initial request parameters.

__ a. Use the drop down box to set the value for the verb to **Create**.

__ b. Leave **Key** blank.

__ c. For **Description,** enter **small black clasp.**

__ d. Right click on **ProductPackages** and select **Add element** from the pop-up menu.


Initial request parameters

Name	Type
⊟ Siebel_ProductTypeBGElem...	Siebel_ProductTypeBG
verb	string
⊟ Siebel_ProductType	Siebel_ProductType
Key	string
Description	string
ProductPackages	Siebel_ProductPackag
version	
delta	
locale	

__ e. For **GlobalIdentifier,** enter **12345678901234.**

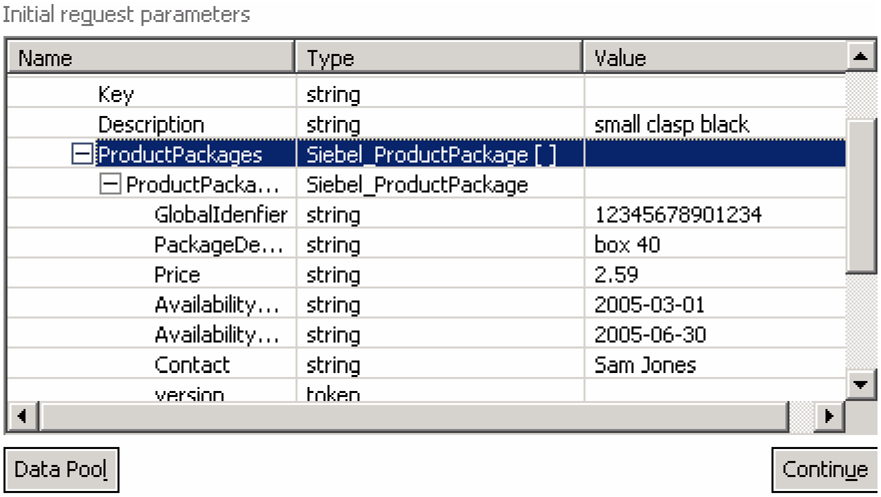
__ f. For **PackageDescription**, enter **box 40.**

__ g. For **Price**, enter **2.59.**

__ h. For **AvailabilityStart**, enter **2005-03-01.**

__ i. For **AvailabiltyEnd**, enter **2010-06-30.**

__ j. For **Contact**, enter **Sam Jones.**

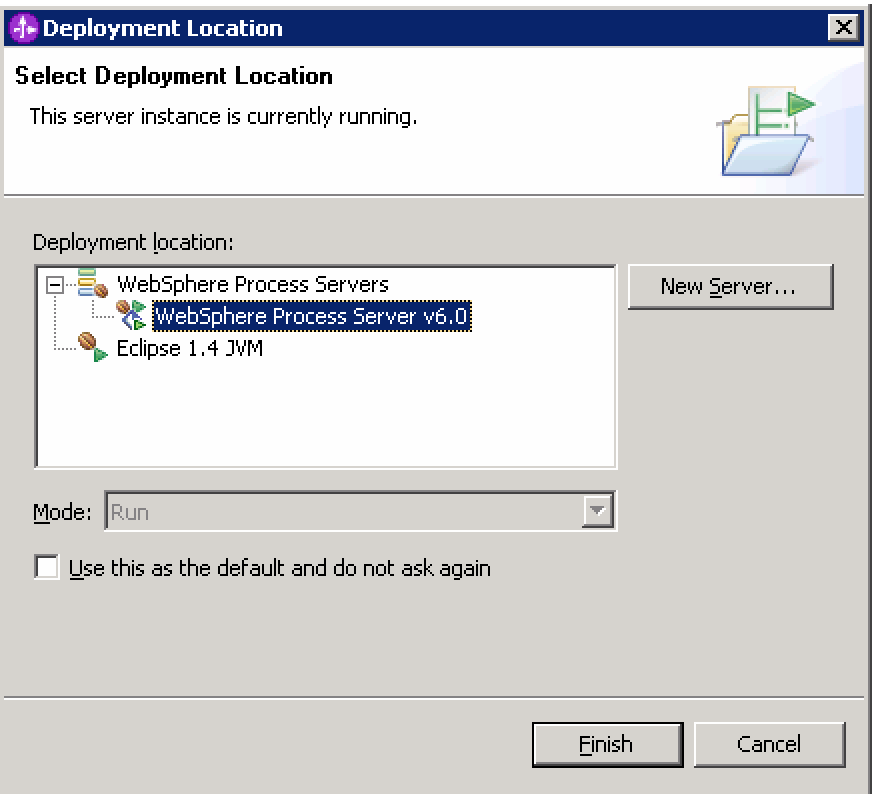

Initial request parameters

Name	Type	Value
Key	string	
Description	string	small clasp black
⊟ ProductPackages	Siebel_ProductPackage [ ]	
⊟ 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
version	token	

____ 9. Run the test.

__ a. Click the **Continue** button.

__ b. In the Deployment Location dialog, select the appropriate WebSphere Process Server. This is especially true if you are using a remote server for testing



Select Deployment Location

This server instance is currently running.

Deployment location:

- WebSphere Process Servers
  - WebSphere Process Server v6.0
- Eclipse 1.4 JVM

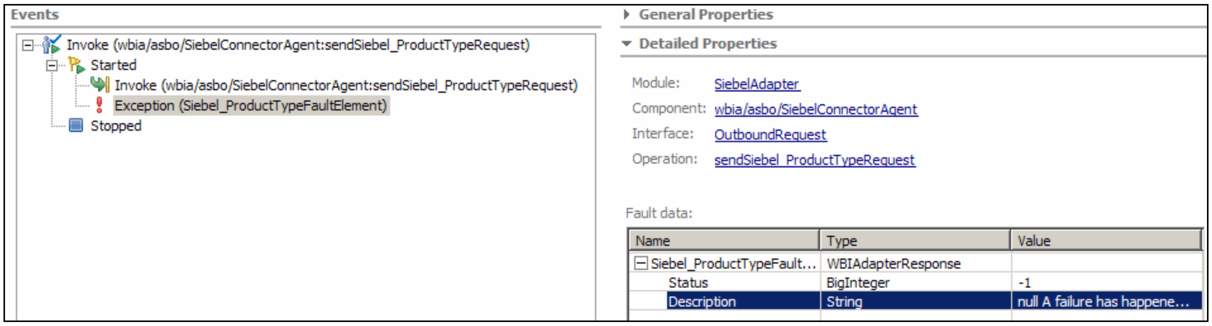
Mode: Run

☐ Use this as the default and do not ask again

__ c. Click **Finish.**

__ d. Wait for the test client to start.

__ e. You might see the following exceptions in the Test Component. This should not affect the outcome of this lab.



- Invoke (wbia/asbo/SiebelConnectorAgent:sendSiebel_ProductTypeRequest)
  - Started
    - Invoke (wbia/asbo/SiebelConnectorAgent:sendSiebel_ProductTypeRequest)
    - Exception (Siebel_ProductTypeFaultElement)
  - Stopped

Module: SiebelAdapter

Component: wbia/asbo/SiebelConnectorAgent

Interface: OutboundRequest

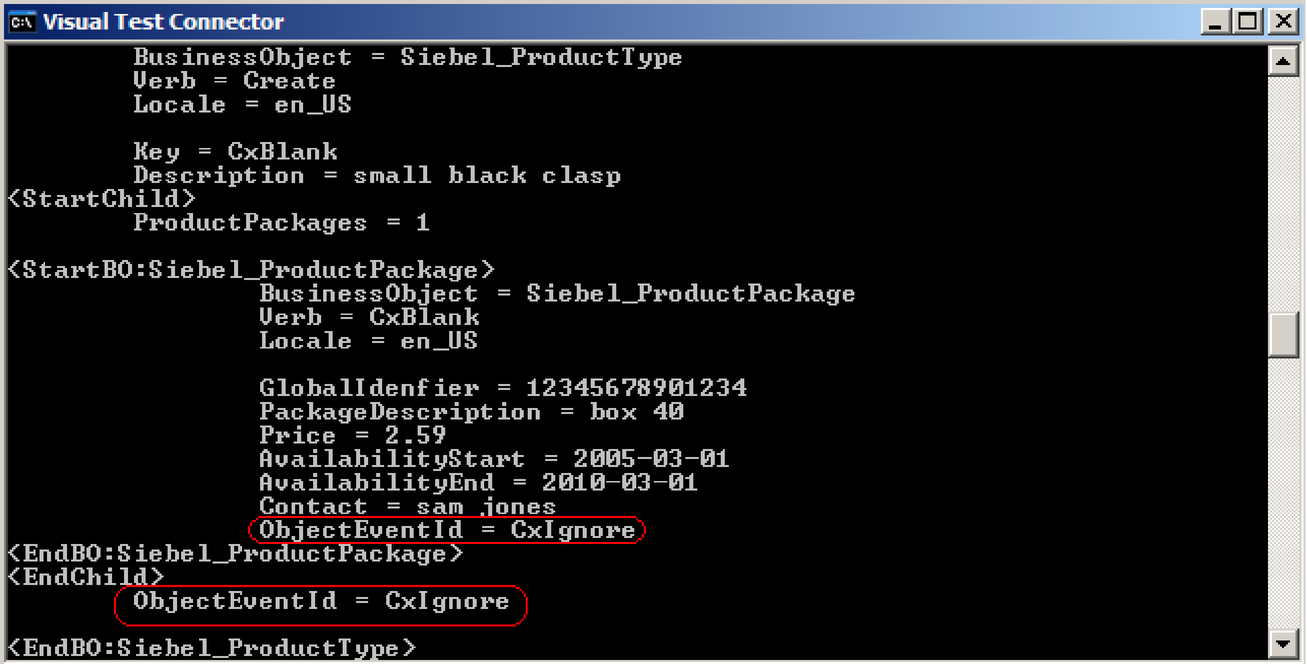
Operation: sendSiebel_ProductTypeRequest

Fault data:

Name	Type	Value
Siebel_ProductTypeFault...	WBIAdapterResponse	
Status	BigInteger	-1
Description	String	null A failure has happene...

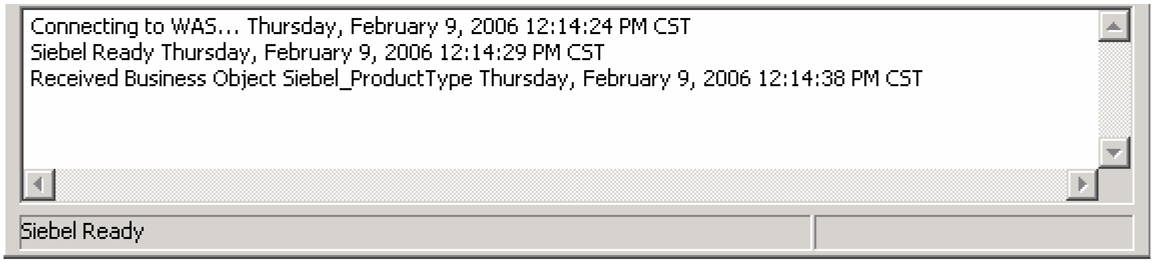
The reason for this error is because of using the test component to test the module. When the BO is passed to the adapter from test component, the “ObjectEventId” for the BO “Siebel_ProductType” and its child BO “Siebel_ProductPackage” are not being set. But in real world scenario this ObjectEventId will be set when you specify the mapping.

You can look at the values in the Visual Test Connector Command prompt.



```
        BusinessObject = Siebel_ProductType
        Verb = Create
        Locale = en_US

        Key = CxBlank
        Description = small black clasp
<StartChild>
        ProductPackages = 1

<StartBO:Siebel_ProductPackage>
                BusinessObject = Siebel_ProductPackage
                Verb = CxBlank
                Locale = en_US

                GlobalIdenfier = 12345678901234
                PackageDescription = box 40
                Price = 2.59
                AvailabilityStart = 2005-03-01
                AvailabilityEnd = 2010-03-01
                Contact = sam jones
                ObjectEventId = CxIgnore
<EndBO:Siebel_ProductPackage>
<EndChild>
        ObjectEventId = CxIgnore

<EndBO:Siebel_ProductType>
```

__ f. Look in the Visual Test Connector window to see if a message has been received.

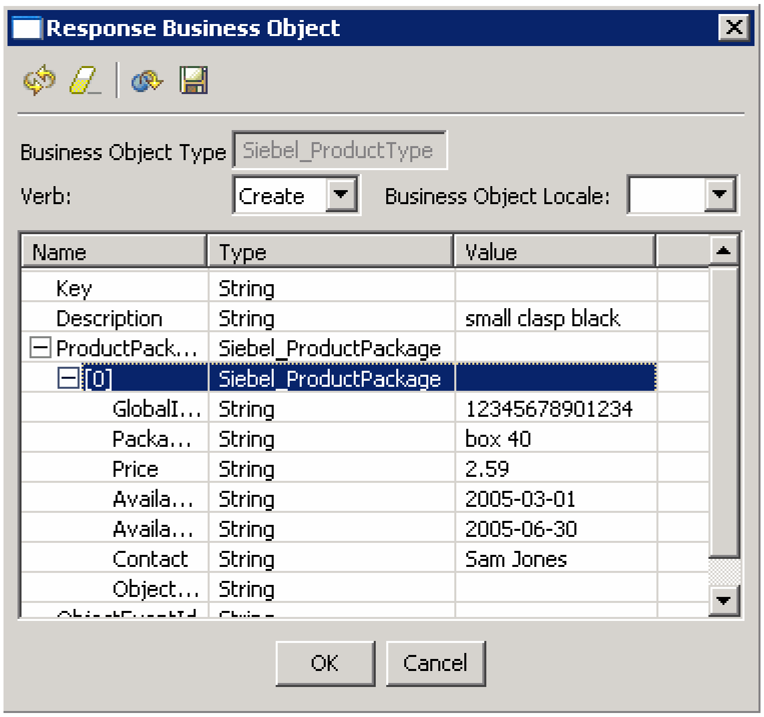


```
Connecting to WAS... Thursday, February 9, 2006 12:14:24 PM CST
Siebel Ready Thursday, February 9, 2006 12:14:29 PM CST
Received Business Object Siebel_ProductType Thursday, February 9, 2006 12:14:38 PM CST
```

Siebel Ready

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

__ h. The business object should look like this:



Business Object Type: Siebel_ProductType

Verb: Create

Business Object Locale:

Name	Type	Value
Key	String	
Description	String	small clasp black
⊟ ProductPack...	Siebel_ProductPackage	
⊟ [0]	Siebel_ProductPackage	
GlobalI...	String	12345678901234
Packa...	String	box 40
Price	String	2.59
Availa...	String	2005-03-01
Availa...	String	2005-06-30
Contact	String	Sam Jones
Object...	String	
ObjectEventId	String	

____ 10. Stop and close the test component.

__ a. Click the **Stop** icon at the top right of the Events pane.

__ b. Click the X on the SiebelConnectorAgent_Test tab.

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

____ 11. 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.

____ 12. Clean up test environment.

__ a. Right-click on appropriate WebSphere Process Server on which you deployed the applications in the Servers view and select Add and remove projects…

__ b. Click << Remove All.

__ c. Click Finish.

____ 13. 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:\Labfiles60\eXchange\WBIAdapters\workspace

<PROJECT_INTERCHANGE>

C:\Labfiles60\eXchange\WBIAdapters\Solution\ImportExport_PI.zip

<MODULE>

n/a

<DEPENDENT_LIBRARIES>

n/a

____ **2.** Continue with **Part 4: Assemble and Test the PeopleSoft Module.**

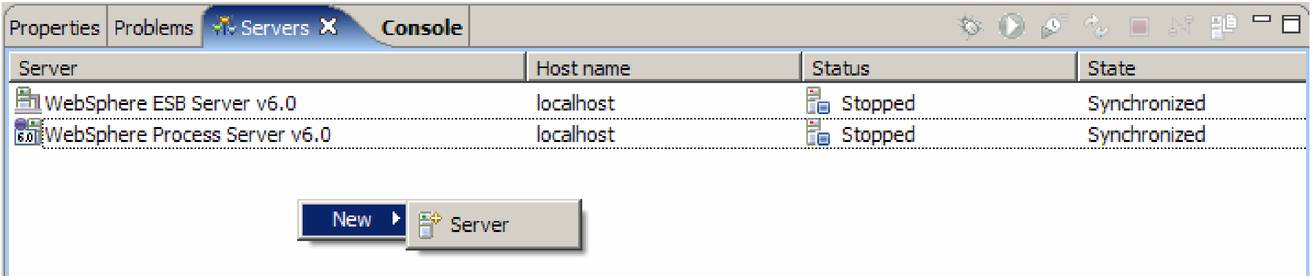
Task: Adding Remote Server to WebSphere Integration Developer Test Environment

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

____ 1. Create a new remote server.

__ a. Right click on the background of the Servers view to access the pop-up menu.

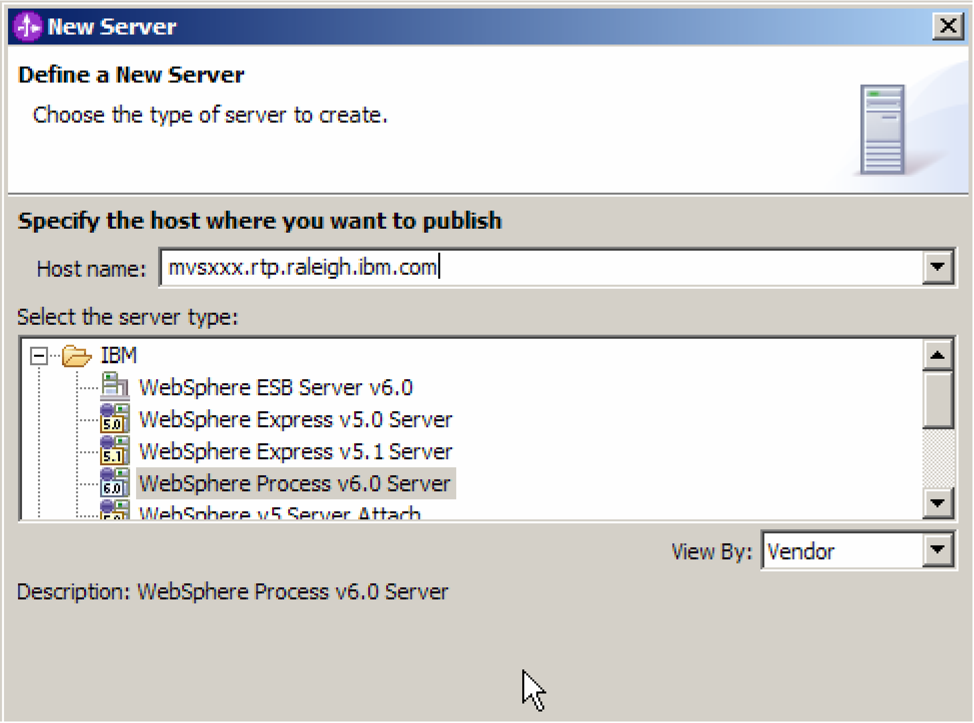
__ b. Select **New > Server.**



Server	Host name	Status	State
WebSphere ESB Server v6.0	localhost	Stopped	Synchronized
WebSphere Process Server v6.0	localhost	Stopped	Synchronized

__ c. Specify hostname to the remote server, **<HOSTNAME>**.

__ d. Ensure that '**WebSphere Process v6.0 Server**' is highlighted in the server type list.



Define a New Server

Choose the type of server to create.

Specify the host where you want to publish

Host name: mvsxxx.rtp.raleigh.ibm.com

Select the server type:

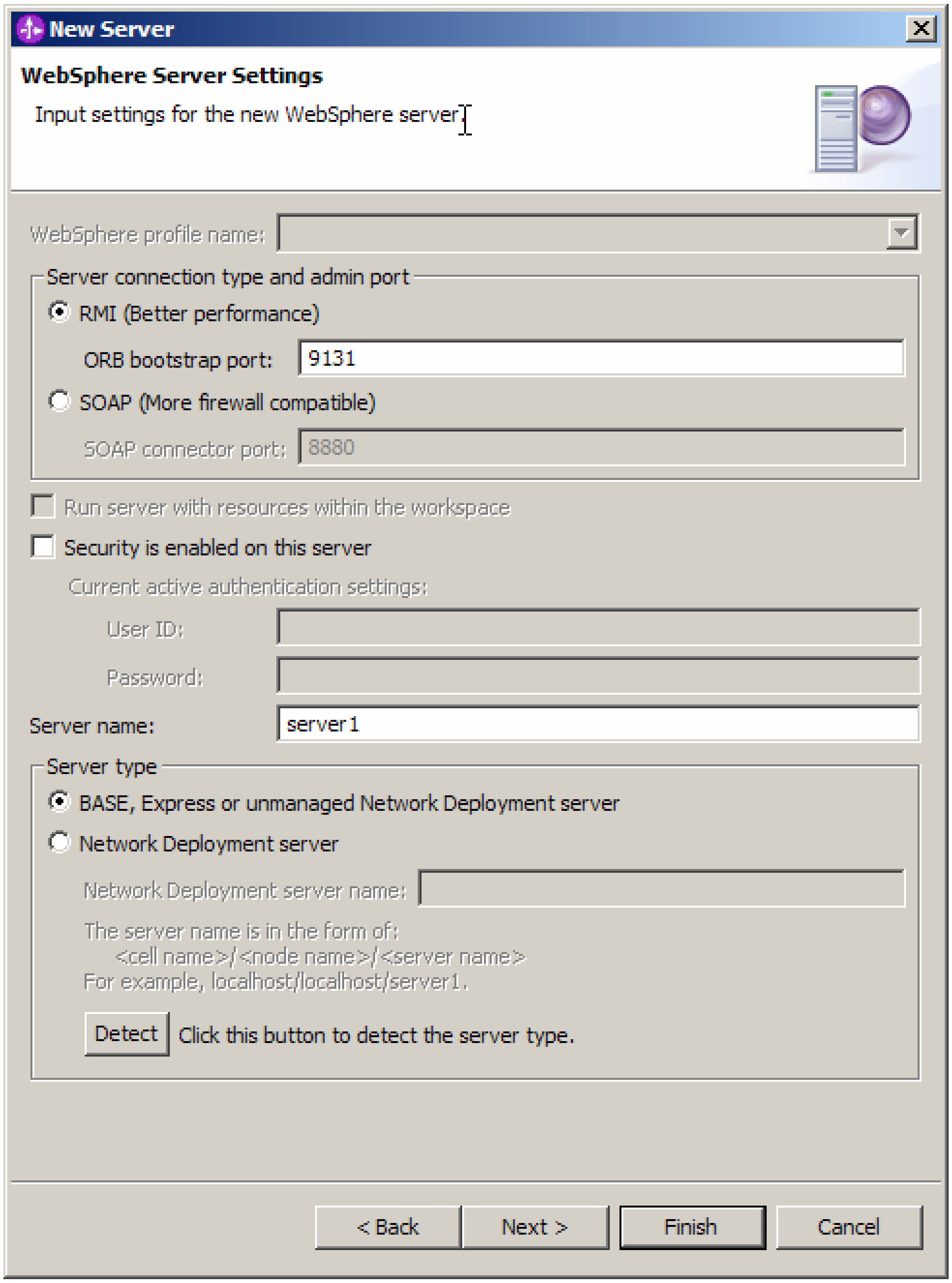
- IBM
  - WebSphere ESB Server v6.0
  - WebSphere Express v5.0 Server
  - WebSphere Express v5.1 Server
  - WebSphere Process v6.0 Server
  - WebSphere v5 Server Attach

View By: Vendor

Description: WebSphere Process v6.0 Server

__ e. Click **Next.**

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



New Server

WebSphere Server Settings

Input settings for the new WebSphere server

WebSphere profile name:

Server connection type and admin port

● RMI (Better performance)

ORB bootstrap port: 9131

○ SOAP (More firewall compatible)

SOAP connector port: 8880

□ Run server with resources within the workspace

□ Security is enabled on this server

Current active authentication settings:

User ID:

Password:

Server name: server1

Server type

● BASE, Express or unmanaged Network Deployment server

○ Network Deployment server

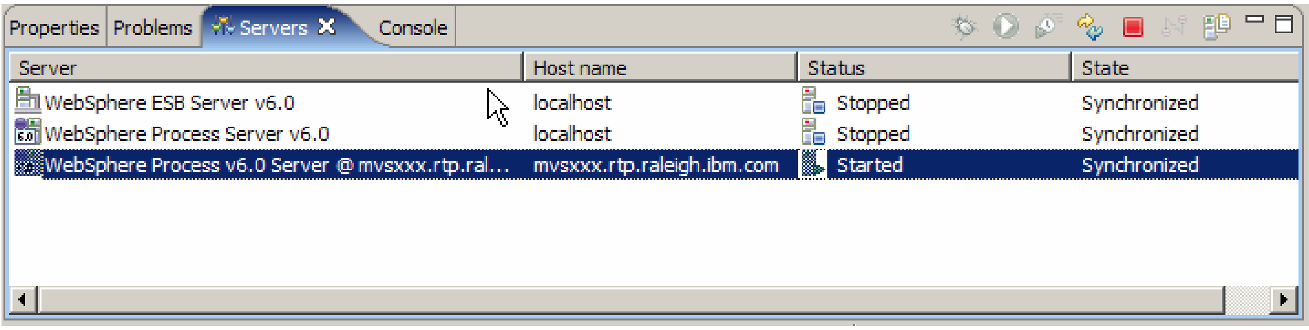
Network Deployment server name:

The server name is in the form of:
<cell name>/<node name>/<server name>
For example, localhost/localhost/server1.

Detect Click this button to detect the server type.

__ g. Click **Finish**.

__ h. The new server should be seen in the Server view.



Server	Host name	Status	State
WebSphere ESB Server v6.0	localhost	Stopped	Synchronized
WebSphere Process Server v6.0	localhost	Stopped	Synchronized
WebSphere Process v6.0 Server @ mvsxxx.rtp.ral...	mvsxxx.rtp.raleigh.ibm.com	Started	Synchronized

____ 2. Start the remote server if it is not already started. WebSphere Integration Developer does not support starting remote servers from the Server View.

__ a. From a command prompt, telnet to the remote system if needed:

‘**telnet <HOSTNAME> <TELNET_PORT>**’

userid : **<USERID>**

pw : **<PASSWORD>**

__ b. Navigate to the bin directory for the profile being used:

cd <WAS_HOME>/profiles/<PROFILE_NAME>/bin

__ c. Run the command file to start the server: **./startServer.sh <SERVER_NAME>**

__ d. Wait for status message indicating server has started:

```
ADMU3200I: Server launched. Waiting for initialization status.

ADMU3000I: Server cl1sr01 open for e-business; process id is 0000012000000002
```