

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

FTP adapter – Processing COBOL copybook files lab

What this exercise is about	1
Lab requirements	1
What you should be able to do	2
Introduction	2
Exercise instructions	2
Part 1: Initialize the workspace and prepare for the lab.....	5
Part 2: Inbound test.....	7
2.1. Prepare for the inbound scenario.....	8
2.2. Configure inbound using the external service wizard	13
2.3. Add Java component	18
2.4. Test inbound scenario.....	20
Part 3: Outbound scenario	22
3.1. Prepare for the outbound scenario	23
3.2. Configure outbound using external service wizard	25
3.3. Test outbound scenario	32
What you did in this exercise	39
Task: Adding remote server to WebSphere Integration Developer test environment	40

What this exercise is about

The objective of this lab is to provide you with an understanding of how to use the 'external data' wizard in WebSphere Integration Developer to generate business object definitions from a COBOL program source file. Then, the lab guides you through the configuration of inbound and outbound processing using these business object definitions to process COBOL copybook files.

Lab requirements

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

- WebSphere Integration Developer V6.2 installed and updated with latest fixes
- WebSphere Process Server V6.2 Test Environment installed and updated latest fixes
- Extract Labfiles62.zip to your C:\ (your root) drive

IBM WEBSPPHERE ADAPTER 6.2 – LAB EXERCISE

What you should be able to do

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

- Import FTP adapter RAR file into WebSphere Integration Developer
 - Use External Data wizard to generate business object and wrapper definitions from a COBOL program source file
 - Use External Service wizard to configure Activation Spec Properties, Resource Adapter Properties to generate Business Objects and other artifacts
 - Deploy the adapter application onto WebSphere Process Server
 - Test the deployed application using WebSphere Process Server test environment
 - Restore the server configuration
-

Introduction

Use the external data wizard to generate a business object definition for a COBOL program source file. After you have generated the business object definition, you can optionally rerun the external data wizard to generate a wrapper business object definition from the generated business object.

You will use external service wizard to configure FTP adapter to convert COBOL copybook files to business objects during the inbound processing and convert business objects to COBOL copybook files during outbound processing.

Exercise instructions

Some instructions in this lab are Windows operating-system specific. If you plan on running the lab on an operating-system other than Windows, you will need to run the appropriate commands, and use appropriate files (.sh or .bat) for your operating system. The directory locations are specified in the lab instructions using symbolic references, as follows:

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

Reference variable	Windows location	AIX® or UNIX® location
<WID_HOME>	C:\Program Files\IBM\WID62	
<WPS_HOME>	C:\<WID_HOME>\runtimes\bi_v62	
<FTPADAPTER_HOME>	<WID_HOME>\ResourceAdapters\FTP_6.2.0.0\deploy	
<LAB_FILES>	C:\Labfiles62	/tmp/Labfiles62
<WORKSPACE>	<LAB_FILES>\FTPCobol\workspace	
<EVENT_DIR>	<LAB_FILES>\FTPCobol\eventdir	
<ARCHIVE_DIR>	<LAB_FILES>\FTPCobol\archivedir	
<FTPFILES>	<LAB_FILES>\FTPFiles	
<TEMP>	C:\temp	/tmp

Windows users' note: When directory locations are passed as parameters to a Java™ program such as EJBdeploy or wsadmin, it is necessary to replace the backslashes with forward slashes to follow the Java convention. For example, replace C:\Labfiles62\ with C:/Labfiles62/

IBM WEBSphere ADAPTER 6.2 – LAB EXERCISE

Instructions if using a remote server for testing

Note that the previous table is relative to where you are running WebSphere Integration Developer. The table below is related to where you are running the 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	sssr011	
<WAS_HOME>	C:\Program Files\IBM\WebSphere\AppServer	/etc/sscell/AppServer	
<HOSTNAME>	Localhost	mvsxxx.rtp.raleigh.ibm.com	
<SOAP_PORT>	8880	8880	
<TELNET_PORT>	N/A	1023	
<PROFILE_NAME>	AppSrv01	default	
<USERID>	N/A	ssadmin	
<PASSWORD>	N/A	fr1day	

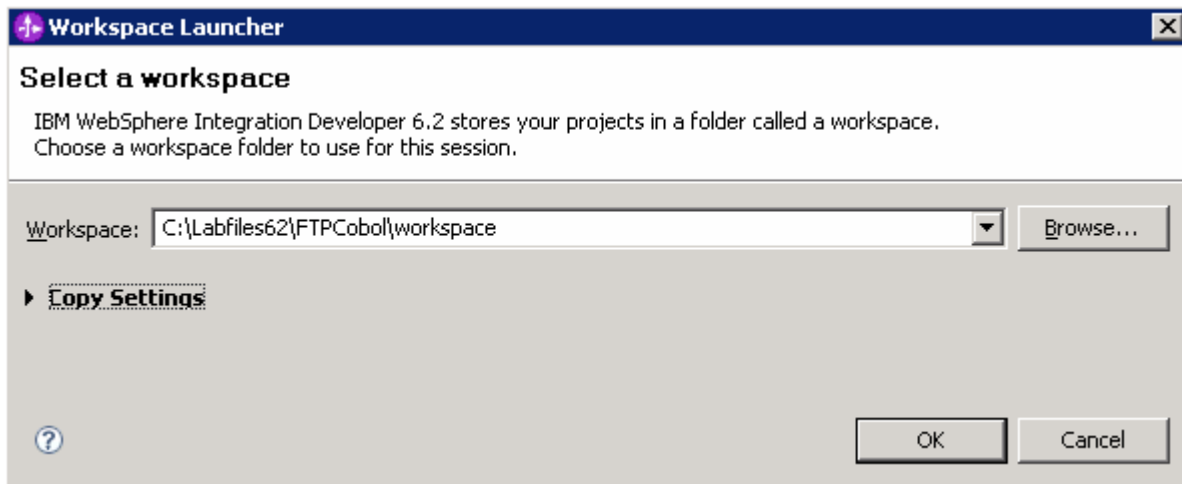
Instructions for using a remote testing environment, such as z/OS, AIX or Solaris, can be found at the end of this document, in the section [“Task: Adding remote server to WebSphere Integration Developer test environment”](#).


IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

Part 1: Initialize the workspace and prepare for the lab

This part of the lab, you will extract the required lab files and start the WebSphere Integration Developer V6.2 with a new workspace and then define required WebSphere Environment Variables using the administrative console of WebSphere Process Server V6.2

- ___ 1. Extract the provided Labfiles62.zip to your C:\ (root) drive, if you have not already done so. This will create the necessary subdirectory structure to complete the lab, and provides you with sample text files
- ___ 2. Create directory structure on your FTP Server
 - ___ a. Log onto FTP machine/FTP Server using your ftp user and its password
 - ___ b. Create these directories under the user's home directory:
 - 1) mkdir **EventDir**
 - 2) mkdir **ArchiveDir**
 - 3) mkdir **OutDir**
- ___ 3. Start the WebSphere Integration Developer V6.2 with a new workspace
 - ___ a. Select **Start > All Programs > IBM WebSphere Integration Developer > IBM WebSphere Integration Developer V6.2 > WebSphere Integration Developer V6.2**
 - ___ b. From the Workspace Launcher window, enter **<WORKSPACE>** for the Workspace field



- ___ 4. Click the  button on the right corner to close the Welcome page and proceed with the workbench

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

5. Follow the instructions of “**Define WebSphere variable**” lab to define the WebSphere variables listed in the table below, which are used later in this lab:

Note: In FlatFile adapter, you have to give directory values (Event directory, Archive directory, Output directory and so on, needed values in activation spec and managed connection factory) while running external service wizard. Presently these values can be changed in the module and re-deploy it (or) can be changed in deployed application from administration console. This is like hard coding the values of these properties. All local directory (including event and archive) properties in ActivationSpec can be filled with WebSphere environment variables. This way hard coding of directory paths for these properties can be avoided. In WebSphere Process Server V6.1 and above, you can declare these values as environment variables and specify the environment variable name in the external service wizard. So when you deploy this application, the environment variable name is replaced with actual value and used by the adapter. This is very helpful if the values of these properties need to be changed.

Name	Value
FTP_EVENT	/root/EventDir
FTP_ARCHIVE	/root/ArchiveDir
FTP_LOCAL_EVENT	C:\Labfiles62\FTPCobo\LocalEventDir
FTP_LOCAL_ARCHIVE	C:\Labfiles62\FTPCobo\LocalArchiveDir
FTP_OUT	/root/OutDir

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

Part 2: Inbound test

In this part, you will import the FTP adapter RAR into the new workspace and use the external data wizard to generate a business object definition for a COBOL program source file. After you have generated the business object definition, you can optionally rerun the external data wizard to generate a wrapper business object definition from the generated business object.

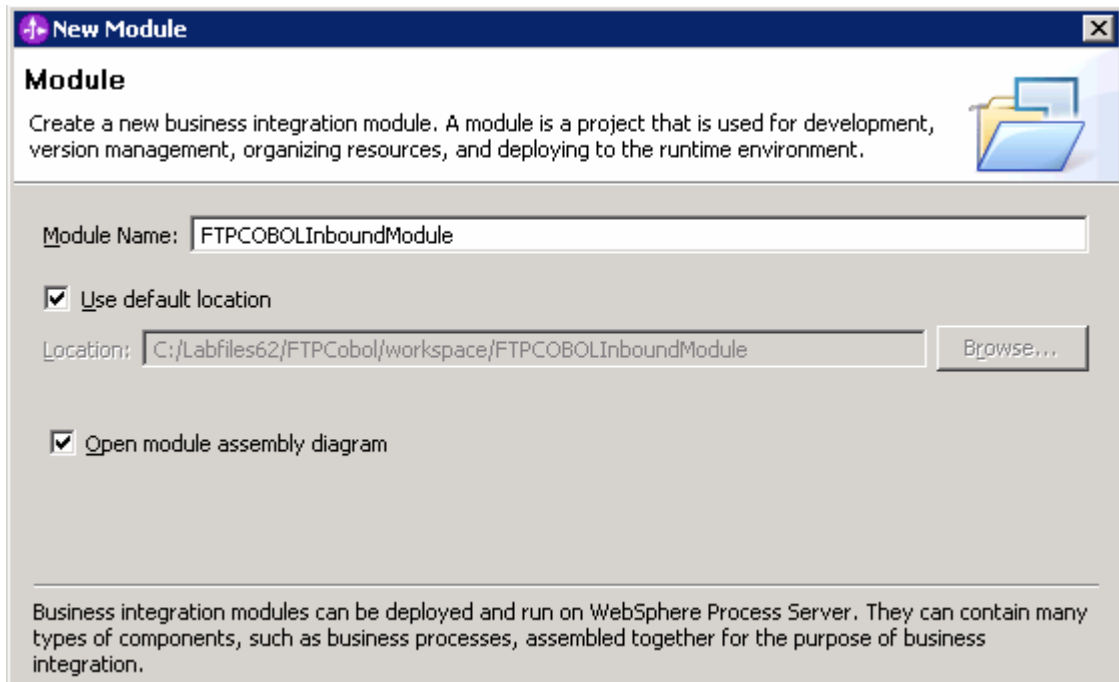
You will make use of the new 'COBOL, C, or PL/I data binding' introduced in V6.2, while running the External service wizard to generate the required artifacts to convert COBOL copybook files to business objects during inbound processing.

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

2.1. Prepare for the inbound scenario

In this part of the lab you will use the default function selector and data binding options from the External Service wizard and generate other required artifacts to test the outbound scenario.

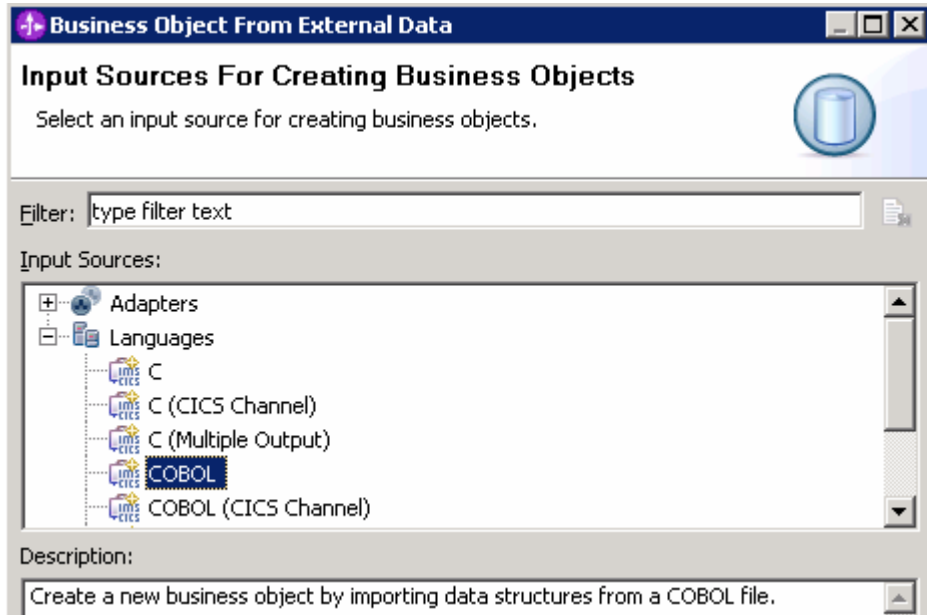
- ___ 1. Create FTPCOBOLInboundModule
 - ___ a. From the Business Integration window, right-click and select **New > Module**
 - ___ b. From the New Module window, enter **FTPCOBOLInboundModule** for the Module Name



- ___ c. Ensure that the box next to **Open module assembly diagram** is checked and then click **Finish**
 - ___ d. You will now see a new module, FTPCOBOLInboundModule, created from your Business Integration window
- ___ 2. Generate the business object definition for the COBOL program source file
 - ___ a. In the Business Integration section of the window, right-click the module, FTPCOBOLInboundModule, and select **New > Business Object from External Data**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

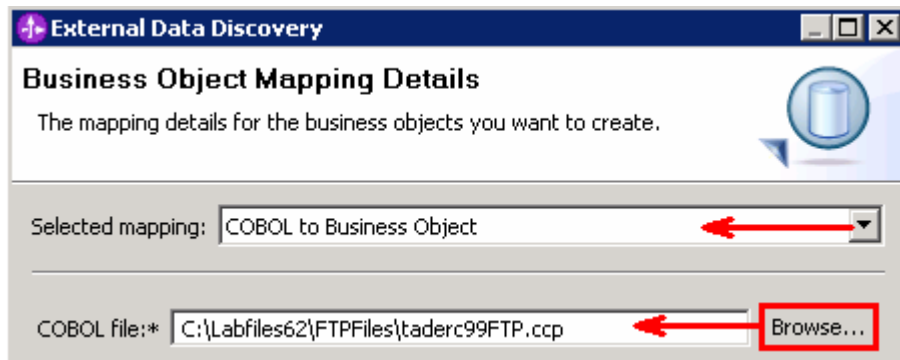
__ b. From the Business Object window, select **Languages > COBOL**



__ c. Click **Next**

__ d. In the Business Object Mapping Details window,

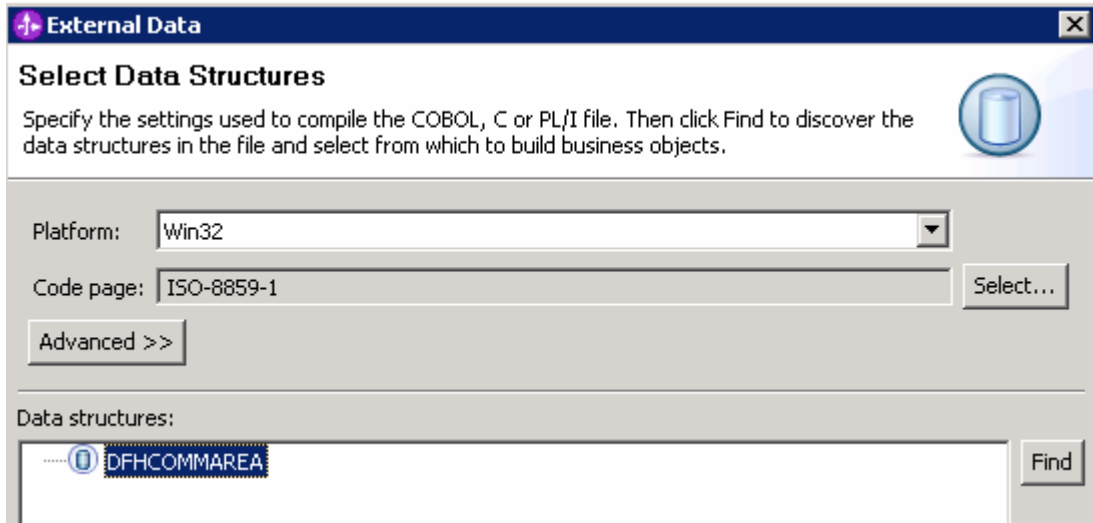
- 1) Ensure the **Choose mapping** value is **COBOL to Business Object**
- 2) Click **Browse...** and select the **traderc99FTP.ccp** file located in **<FTPFILES>**



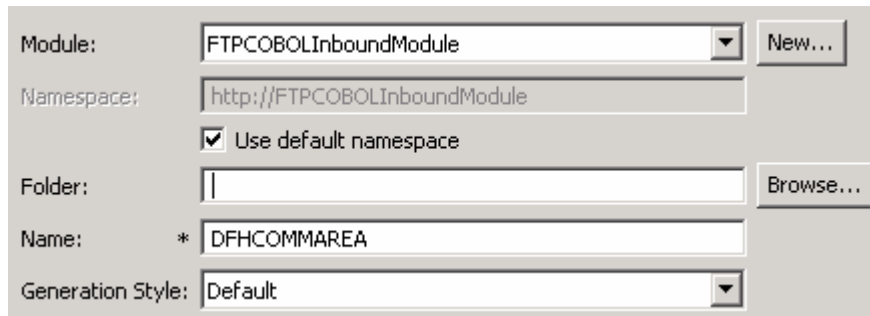
3) Click **Next**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

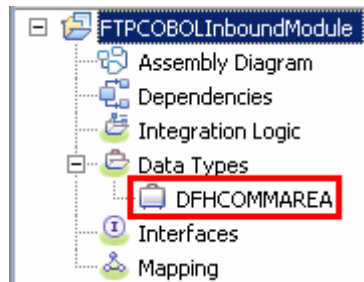
- ___ e. In the Select Data Structures window, the new business object, called **DFHCOMMAREA**, is displayed



- ___ f. Select **DFHCOMMAREA** and click **Next**
- ___ g. In the Generate Business Objects screen, ensure that the selected Module is **FTPCOBOLInboundModule** and click **Finish**



A business object, called DFHCOMMAREA, is created in the module

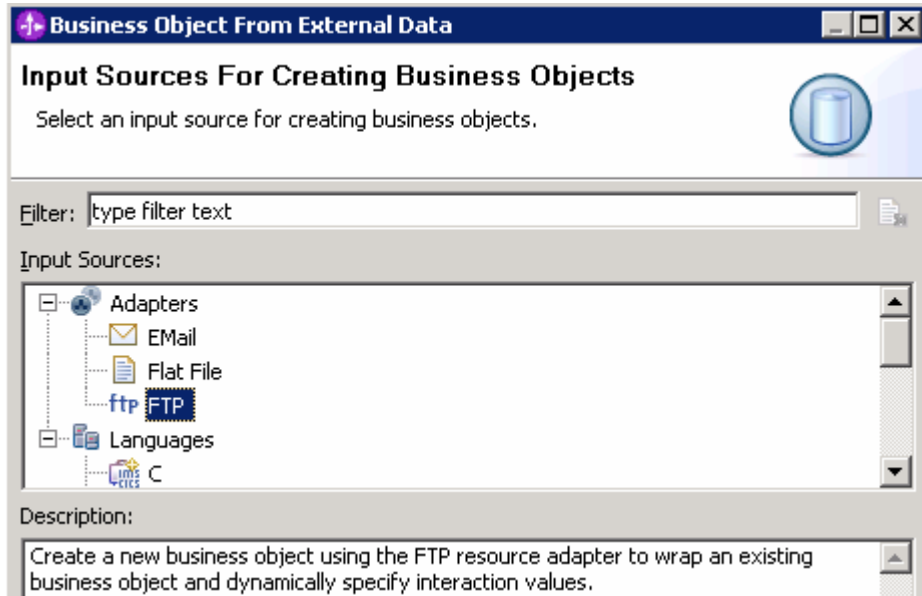


- ___ 3. Generate a wrapper business object definition. Wrapper business object definitions wrap existing business object definitions with additional function.

- ___ a. In the Business Integration section of the window, right-click the module and select **New > Business Object from External Data**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

___ b. From the External Data window, select **Adapters > FTP** under Available Types:

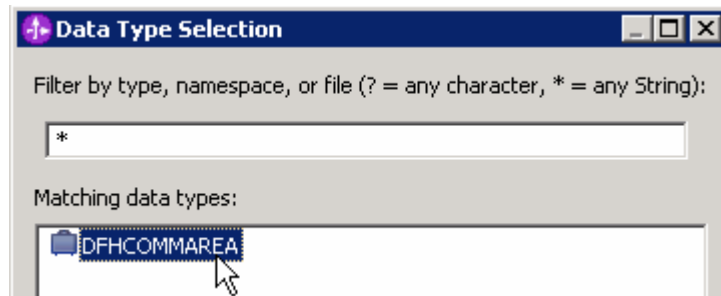


___ c. From the next screen, ensure that the correct module, **FTPCOBOLInboundModule**, is selected and click **Next**

Note: The resource adapter archive file is imported and a new connector project, **CWYFT_FTPFile**, is listed under Business Integration view.

___ d. Provide the below values in Business Object Properties screen:

- 1) Data type: click **Browse....** A Data Type selection window is opened
- 2) Select **DFHCOMMAREA** under Matching data types

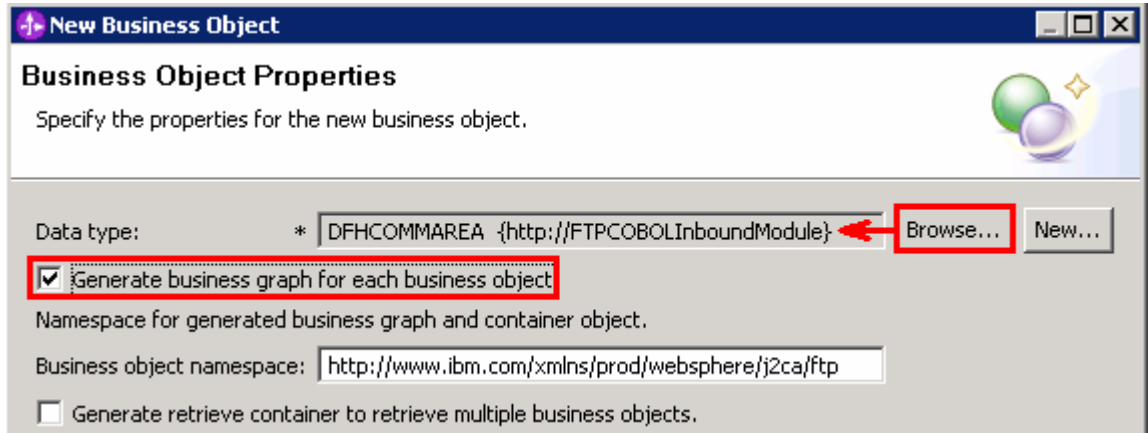


3) Click **OK**

___ e. To generate a business graph, select the **Generate business graph for each business object** check box

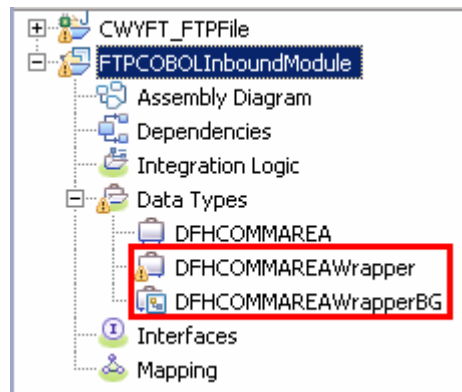
IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

- ___ f. **Do not** select the **Generate retrieve container to retrieve multiple business objects** check box. This new options is used and explained in outbound scenario



- ___ g. Click **Finish**

A wrapper business object and a business graph, called **DFHCOMMAREAWrapper** and **DFHCOMMAREAWrapperBG**, as shown in the figure below, are listed for the current module in the Business Integration window

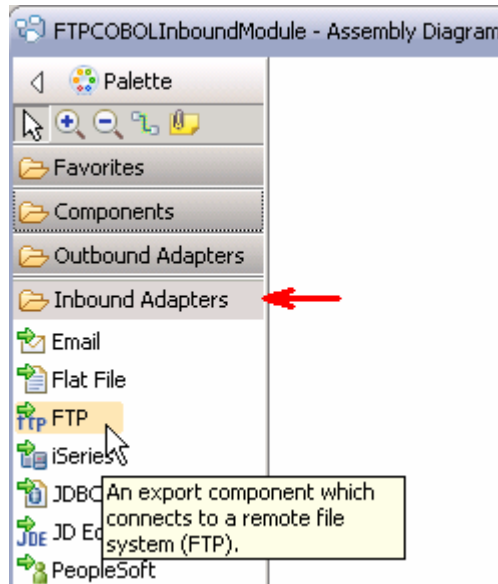


IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

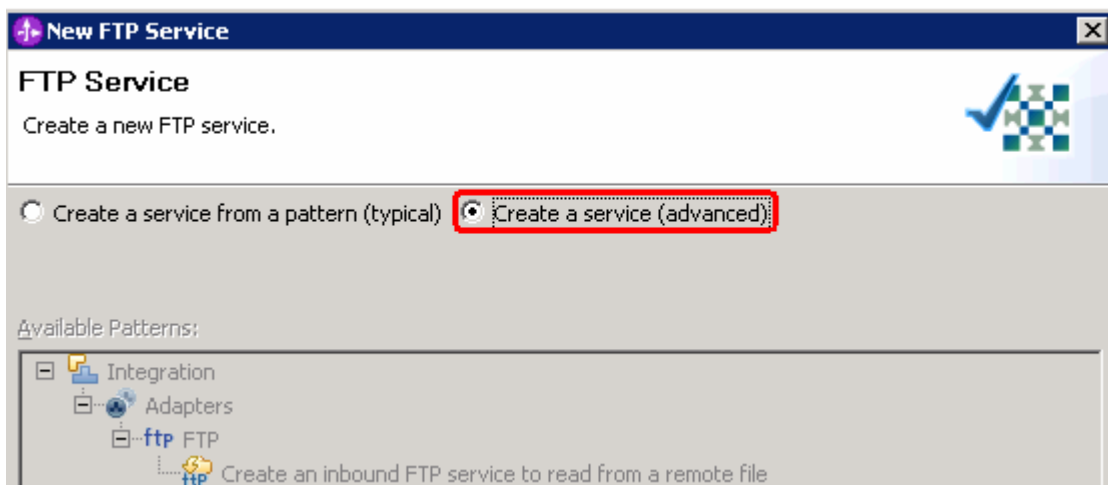
2.2. Configure inbound using the external service wizard

In this part of the lab you will use the new external service wizard and use the new 'COBOL, C, or PL/I data binding' to create the required artifacts to test the inbound scenario.

- ___ 1. Ensure that the FTPCOBOLInboundModule assembly diagram is opened in Assembly editor
 - ___ a. In the Business Integration view, expand FTPCOBOLInboundModule and double-click Assembly Diagram
- ___ 2. To start External Service from the Palette:
 - ___ a. From the **Palette** on the left side of Assembly Diagram, click **Inbound Adapters**:



- ___ b. Under Inbound Adapters, click the **FTP** and then click the empty canvas of the assembly diagram. The New FTP File Service wizard is opened
- ___ 3. From the FTP Service screen, select **Create a service (advanced)**



IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

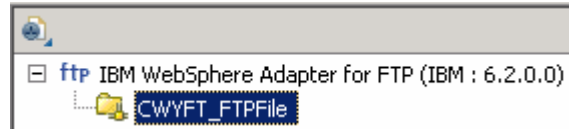
__ a. Click **Next**

Note: You can also start the External Service from the **File menu** option:

From the main menu, select **File > New > External Service**. This opens an External Service wizard that helps you obtain a service which establishes connectivity with other systems.

Select **Adapters > FTP** and click **Next**

- ___ 4. On the Select an Adapter screen, select **IBM WebSphere Adapter for FTP (IBM : 6.2.0.0) > CWYFT_FTFile** and click **Next**



- ___ 5. Service Configuration Properties:
- __ a. Deploy connector project: ensure that the default option **With module for use by single application** is selected
- __ b. Enter these for FTP system connection information:

1) Host name: **<FTP_Machine_Name>** (or IP Address of the machine that has FTP Server),
for Ex: wsbeta159.austin.ibm.com

Note: Alternatively you can also define a WebSphere variable for the host name and replace it here.

2) Remote directory: **`\${FTP_EVENT}`**

Note: This is the directory from where adapter gets the event files.

3) Local directory: **`\${FTP_LOCAL_EVENT}`**

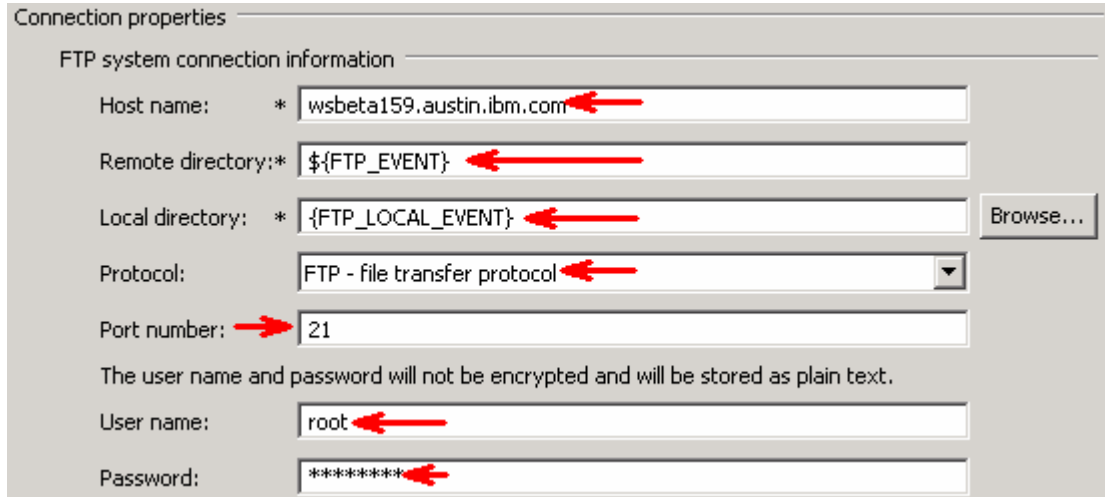
4) Protocol: **FTP – file transfer protocol**

5) Port number: **21** (default)

6) **User name:** username using which you connect to your FTP server (for Ex: **root**)

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

7) **Password:** password for the above user to connect to your FTP server



Connection properties

FTP system connection information

Host name: * wsbeta159.austin.ibm.com

Remote directory:* \${FTP_EVENT}

Local directory: * {FTP_LOCAL_EVENT} Browse...

Protocol: FTP - file transfer protocol

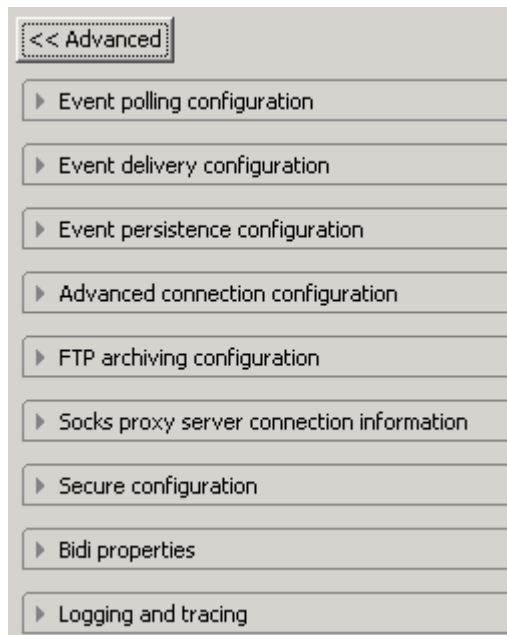
Port number: 21

The user name and password will not be encrypted and will be stored as plain text.

User name: root

Password: *****

__ c. Click **Advanced >>** to see the hidden advanced properties that can be configured:



<< Advanced

- ▶ Event polling configuration
- ▶ Event delivery configuration
- ▶ Event persistence configuration
- ▶ Advanced connection configuration
- ▶ FTP archiving configuration
- ▶ Socks proxy server connection information
- ▶ Secure configuration
- ▶ Bidi properties
- ▶ Logging and tracing

You can click each of the configurations and review the options available under it. For this lab, you will need only some of these properties.

__ d. Event persistence configuration: In this part of the lab, you will not use any JNDI instead use adapter's in-memory representation of event table to store all the necessary information

Note: The Event recovery data source (JNDI) name is **not mandatory** from V6.1. Now, the adapter can use **in-memory representation** of event table to store all the necessary information. Adapter uses this feature when event database information is not configured during inbound event polling. This feature will not support the capability of handling "Ensure once-only event delivery".

__ e. FTP archiving configuration:

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

- 1) Local Archive directory: click **Browse...** and select **#{FTP_LOCAL_ARCHIVE}**
- 2) Remote archive directory: **#{FTP_ARCHIVE}**

- ___ 6. For this lab, you are not going to use the J2C authentication. So, **uncheck** the box next to **Specify a Java Authentication and Authorization Services (JAAS) alias security credentials**.
- ___ 7. Under Service properties, for **Function selector options**, select **Use default function selector 'FilenameFunctionSelector'** from the drop down list
- ___ 8. For **Data format options**, select **Use COBOL, C or PL/I data format** from the drop down list

- ___ 9. Check the box next to **Change logging properties for wizard** to view the output location of the log file and the logging level and click **Next**

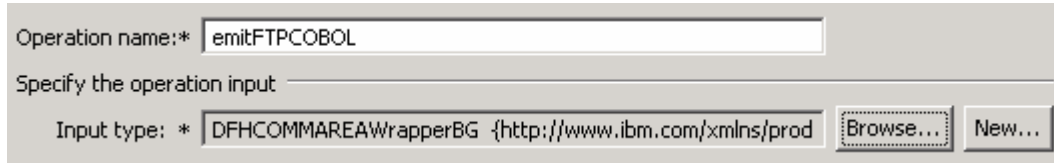
Define emitFTPCOBOL operation:

- ___ 10. From the Operations screen, click **Add...**
 - ___ a. Add Operation window is opened. Select **User Defined Type** for the Data type and click **Next**

- ___ 11. For **Operation name**, enter any name, for Ex: **emitFTPCOBOL**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

- ___ 12. For **Input type**, click **Browse** and select **DFHCOMMAREAWrapperBG** from the Data Type Selection window

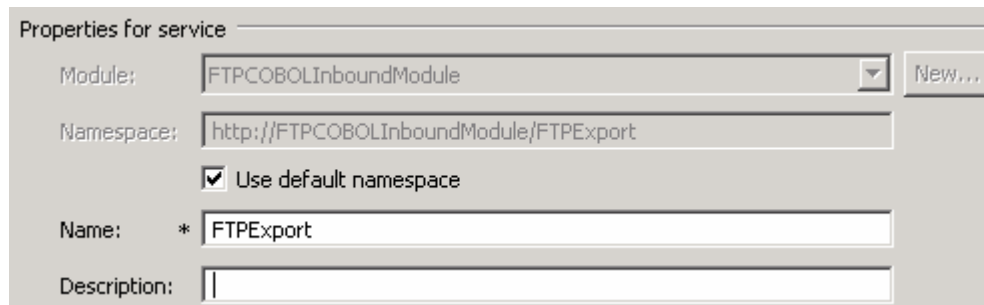


- ___ a. Click **Finish**. The above defined operation, **emitFTPCOBOL**, is populated under Operations list



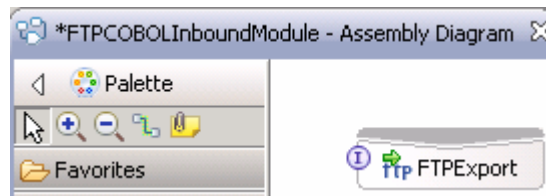
- ___ b. Click **Next** from Operations screen

- ___ 13. From Generate Service screen, accept the default value, **FTPEExport**, for **Name**



- ___ a. Click **Finish**

- ___ 14. The Assembly diagram for FTPCOBOLInboundModule is opened with an Export component, **FTPEExport**:





- ___ 15. Save (**Ctrl + S**) changes to your assembly diagram

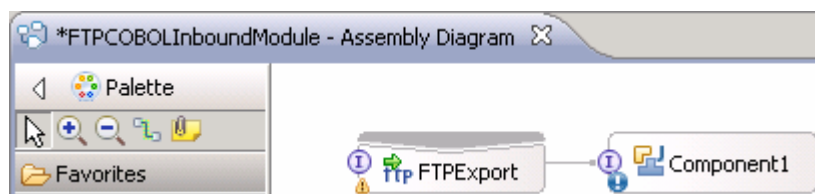
IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

2.3. Add Java component

In this part of the lab, you will add a Java component and then wire the component to the existing Export interface. The Java component is your endpoint.

- ___ 1. Open the assembly diagram for FTPCOBOLInboundModule (if it is already not open)
 - ___ a. From the business integration view, expand **FTPCOBOLInboundModule** and double click **Assembly diagram**
- ___ 2. Drop a Java component to onto the assembly diagram
 - ___ a. From the **Palette**, click **Components** to expand it
 - ___ b. Click **Java** and then click the empty space of FTPCOBOLInboundModule assembly diagram. This will place a new component, **Component1** on the assembly diagram.
- ___ 3. Wire the FTPEXport to the Component1
 - ___ a. Select the **wire** () icon from the Palette
 - ___ b. Click **FTPEXport** and then click **Component1** to wire them together
 - ___ c. Select **OK** for the Add Wire pop-up window:
 - ___ d. From the top of the Palette, click the **Selection Tool** icon () to get back to the normal cursor mode
 - ___ e. Right-click the empty space of the Assembly diagram and select **Arrange Contents Automatically** from the pop-up menu

Your assembly diagram for FTPCOBOLInboundModule will look like this:



- ___ f. Right-click **Component1** and select **Generate Implementation** from the pop-up menu
- ___ g. On the **Generate Implementation** panel, select **default package**, and click **OK**
- ___ h. **Component1Impl.java** is opened in Assembly editor. Scroll down to the method **emitFTPCOBOL** that needs to be implemented and add this code under that method:

Note: The Java code is also available at **<FTPFILES>\ COBOLJavaCode.txt**

```
System.out.println("*****ENDPOINT
emitCOBOLFile*****");
DataObject wrapper =
emitFTPCOBOLInput.getDataObject("DFHCOMMAREAWrapper");
String filename = wrapper.getString("Filename");
System.out.println("File Name : "+filename);
DataObject customer = wrapper.getDataObject("Content");
```

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

```
String customernumber = customer.getString("CustomerNumber");
System.out.println("NAME-----> "+customernumber);
String firstname = customer.getString("FirstName");
System.out.println("First Name--> "+firstname);
String lastname = customer.getString("LastName");
System.out.println("Last Name--> "+lastname);
String street = customer.getString("Street");
System.out.println("Street--> "+street);
String city = customer.getString("City");
System.out.println("City-----> "+city);
String country = customer.getString("Country");
System.out.println("Country-----> "+country);
String phone = customer.getString("Phone");
System.out.println("Phone-----> "+phone);
String postalcode = customer.getString("PostalCode");
System.out.println("Postal Code-----> "+postalcode);
```

__ i. Save (**Ctrl + S**) and close Component1Impl.java

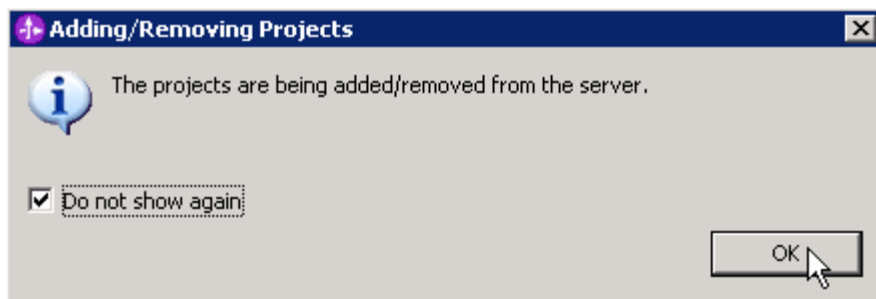
__ j. Save (**Ctrl + S**) and close Assembly diagram: FTPCOBOLInboundModule

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

2.4. Test inbound scenario

In this part of the lab, you will use the WebSphere Process Server Test Environment to test the SCA application inbound processing for the COBOL copybook scenario.

- ___ 1. Add the project to the WebSphere Test Environment server
 - ___ a. Right-click **WebSphere Process Server v6.2** under the Servers view and select **Add and remove projects...** from the pop-up menu
 - ___ b. From the Add and Remove Projects window, select **FTPCOBOLInboundModuleApp** under Available projects panel and click **Add >**
 - ___ c. You will now see the **FTPCOBOLInboundModuleApp** added to the **Configured projects**
 - ___ d. Click **Finish** and wait until the project is being published onto the server. The server is started in Debug mode if it is not already started before
 - ___ e. Click **OK** from Adding/Removing Projects pop-up. Optionally, you can select 'Do not show again' so that you are not required to do this next time when you start the test client



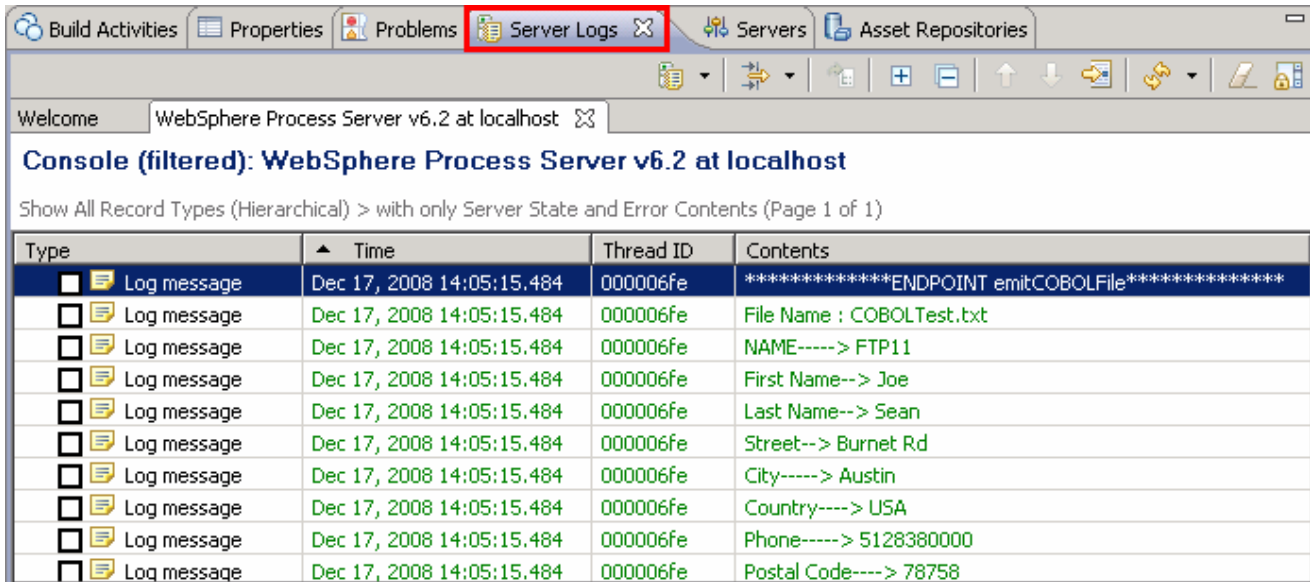
- ___ 2. Test the application by providing input files:

Note: For your convenience, **COBOLTest.txt** is placed in **<FTPPFILES>**.

- ___ a. On the machine where the FTP Server is running, put **COBOLTest.txt** file in the **EventDir**. The adapter will poll the copied file from the event directory and will transfer it to the archive directory

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

- ___ b. The file will pass through the only existing method, **emitFTPCOBOL**, and you should see this message in your **Server Logs** view (or SystemOut.log):



Verify the results:

- ___ c. Check the **<LOCAL_EVENT_DIR>** on your local machine. The file is quickly moved from this directory to the local archive directory
 - ___ d. Check the **ArchiveDir** of your FTP server which should contain the same file name appended with year, month, date, system time, and processed as you have given
 - ___ e. Check the **<LOCAL_ARCHIVE_DIR>** subdirectory which should contain an archive of the event file, with the same file name appended with year, month, date, system time, and success
- ___ 3. Restore the Sever Configuration
- ___ a. Right-click **WebSphere Process Server v6.2** under the Servers view and select **Add and remove projects...** from the pop-up menu
 - ___ b. Select **FTPCOBOLInboundModuleApp** under Configured projects and click **< Remove**
 - ___ c. Click **Finish** after you see the application moved to Available projects. Wait until the application is being unpublished

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

Part 3: Outbound scenario

In this part, you will use the external data wizard to generate a business object definition for a COBOL program source file. After you have generated the business object definition, you can optionally rerun the external data wizard to generate a wrapper business object definition from the generated business object.

You will make use of the new 'COBOL, C, or PL/I data binding' introduced in V6.2, while running the External service wizard to generate the required artifacts to convert business objects to COBOL copybook files during outbound processing.

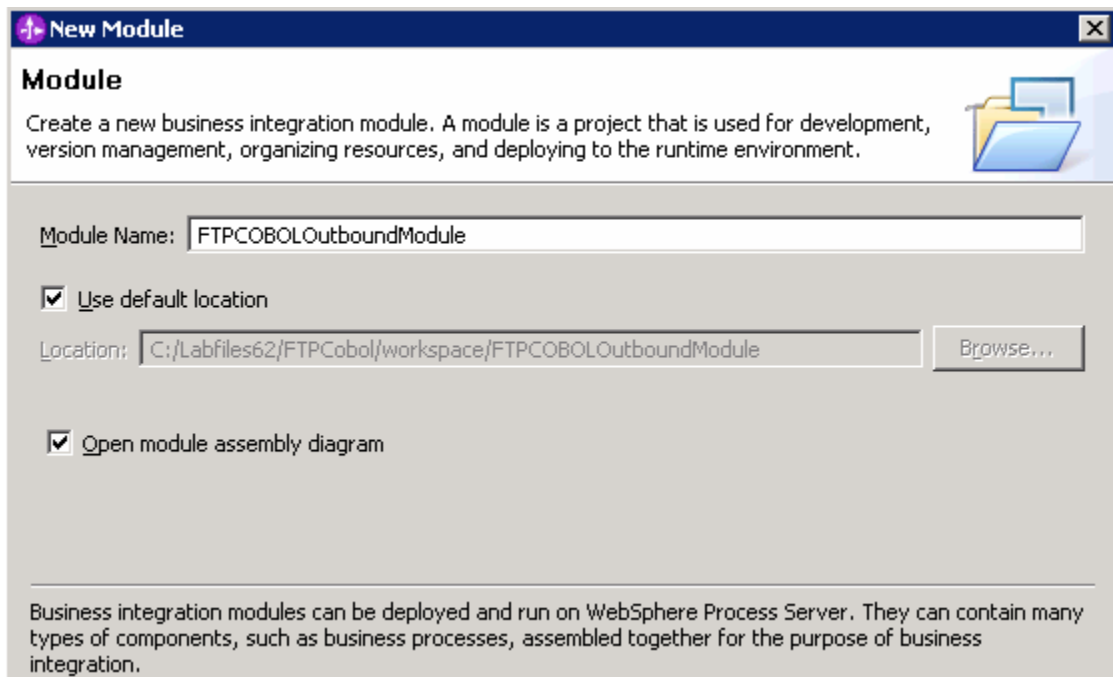
IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

3.1. Prepare for the outbound scenario

In this part of the lab you will create a module, use the external data wizard to generate a business object definition for a COBOL program source file. After you have generated the business object definition, you can optionally rerun the external data wizard to generate a wrapper business object definition from the generated business object.

___ 1. Create FTPCOBOLOutboundModule

- ___ a. From the Business Integration window, right-click and select **New > Module**
- ___ b. From the New Module window, enter **FTPCOBOLOutboundModule** for the Module Name
- ___ c. Ensure that the box next to **Open module assembly diagram** is checked



- ___ d. Click **Finish**

You will now see a new module, FTPCOBOLOutboundModule, created from your Business Integration window

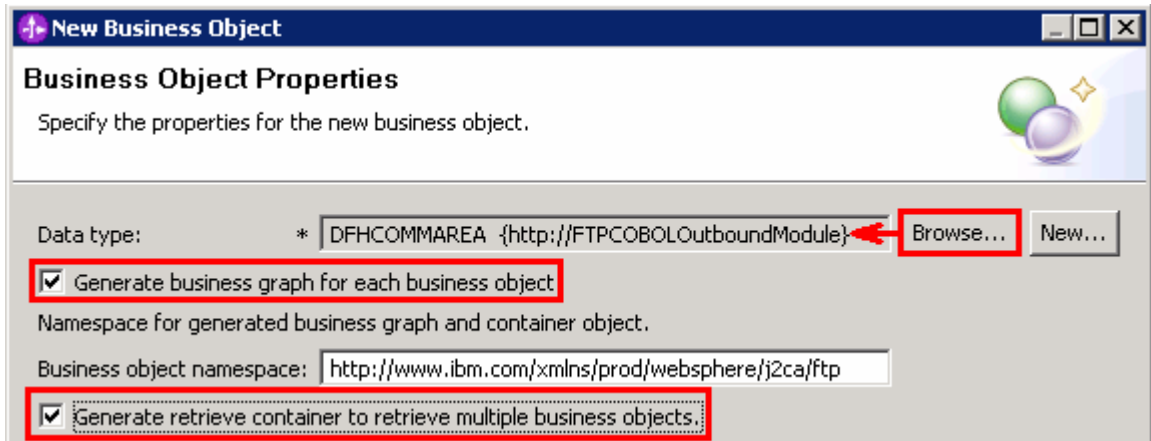
___ 2. Follow the instructions in **Steps 2 and 3 of Part 2.1** to generate business object and a wrapper definition under **FTPCOBOLOutboundModule**

___ 3. Generate a wrapper business object definition for retrieve operation.

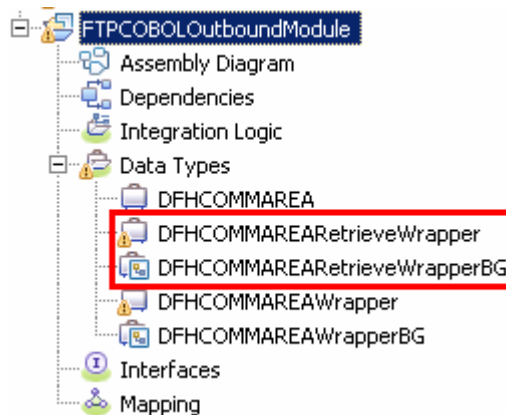
- ___ a. In the Business Integration section of the window, right-click the module and select **New > Business Object From External Data**
- ___ b. From the External Data window, select **Adapters > FTP** under Available Types:
- ___ c. From the next screen, ensure that the correct module, **FTPCOBOLOutboundModule**, is selected and click **Next**
- ___ d. Provide the below values in Business Object Properties screen:

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

- 1) Data type: click **Browse...**. A Data Type selection window is opened
 - 2) Select **DFHCOMMAREA** under Matching data types and click **OK**
- ___ e. To generate a business graph, select the **Generate business graph for each business object** check box
- ___ f. Select the **Generate retrieve container to retrieve multiple business objects** check box



- ___ g. Click **Finish**
- ___ h. A wrapper business object and a business graph, called **DFHCOMMAREARetrieveWrapper** and **DFHCOMMAREARetrieveWrapperBG**, as shown in the figure below, are listed for the current module in the Business Integration window. This **DFHCOMMAREARetrieveWrapperBG** is used during External Service wizard in the next part for retrieve operation.

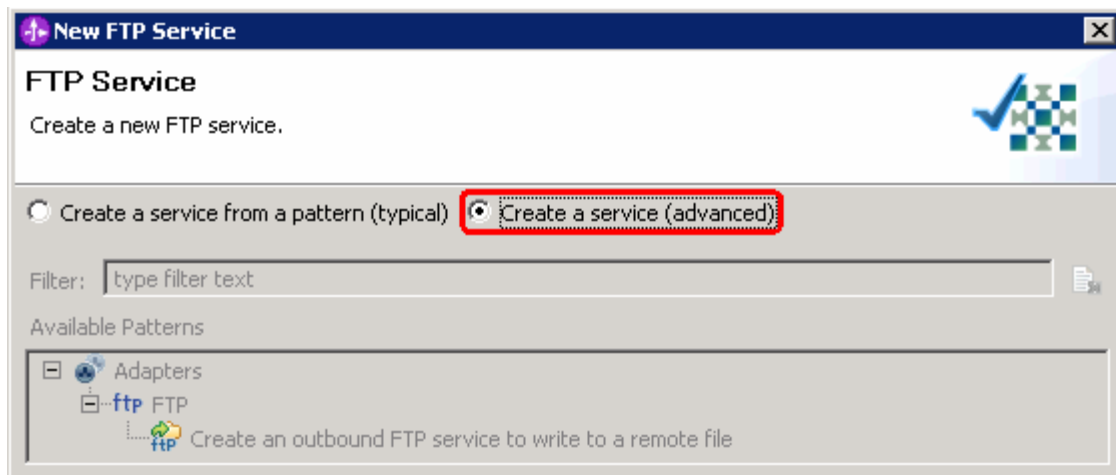


IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

3.2. Configure outbound using external service wizard

In this part of the lab you will use the default function selector and data binding options from the External Service wizard and generate other required artifacts to test the outbound scenario.

- ___ 1. To start the external service wizard from the Palette:
 - ___ a. From the **Palette** on the left side of Assembly Diagram, click **Outbound Adapters**:
 - ___ b. Under Outbound Adapters, click the **FTP** and then click the empty canvas of the assembly diagram. The New FTP Service wizard is opened
- ___ 2. From the FTP Service screen, select **Create a service (advanced)**



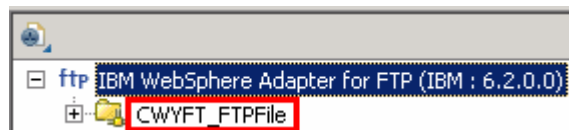
- ___ a. Click **Next**

Note: You can also start the external service wizard from the **File** menu option:

From the main menu, select **File > New > External Service**. This opens an external service wizard that helps you obtain a service which establishes connectivity with other systems.

Select **Adapters > FTP** and click **Next**

- ___ 3. On the Select an Adapter screen, expand **IBM WebSphere Adapter for FTP (IBM : 6.2.0.0)** and select **CWYFT_FTPFile**



- ___ a. Click **Next**

- ___ 4. Service Configuration Properties:
 - ___ a. Deploy connector project: ensure that the default option **With module for use by single application** is selected
 - ___ b. Enter these for FTP system connection information:

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

- 1) Host name: **<FTP_Machine_Name>** (or IP Address of the machine that has FTP Server),
for Ex: wsbeta159.austin.ibm.com
- 2) Directory: **#{FTP_OUT}**

Note: This is the folder where the adapter will create the file.

- 3) Protocol: **FTP – file transfer protocol**
- 4) Port number: **21** (default)
- 5) User name: **username using which you connect to your FTP server** (for Ex: wsbeta)
- 6) Password: **password for the above user to connect to your FTP server**

Deploy connector project: With module for use by single application

Connection properties: Use properties below

Connection properties

FTP system connection information

Host name: * wsbeta159.austin.ibm.com

Directory: * #{FTP_OUT}

Protocol: FTP - file transfer protocol

Port number: 21

The user name and password will not be encrypted and will be stored as plain text.

User name: root

Password: *****

5. Optionally, click **Advanced >>** to see the hidden advanced properties that can be configured:
6. Click **Logging and tracing**

Note: Refer to 'WBIV62_AdapterLogAndTraceLab' for more details on this feature

7. For this lab, you are not going to use the J2C authentication. So, **uncheck** the box next to **Specify a Java Authentication and Authorization Services (JAAS) alias security credentials**.
8. For **Data format options**, select **Use COBOL, C or PL/I data format** from the drop down list

Service properties

Specify a Java Authentication and Authorization Services (JAAS) alias security credential.

J2C authentication data entry:

Data format options: Use COBOL, C or PL/I data format

Data format: Not defined

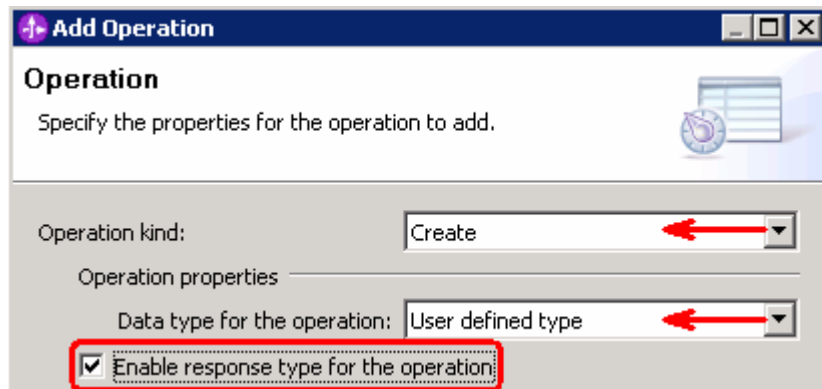
Select...

9. Check the box next to **Change logging properties for wizard** to view the output location of the log file and the logging level and click **Next**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

Define createFTPCOBOL operation:

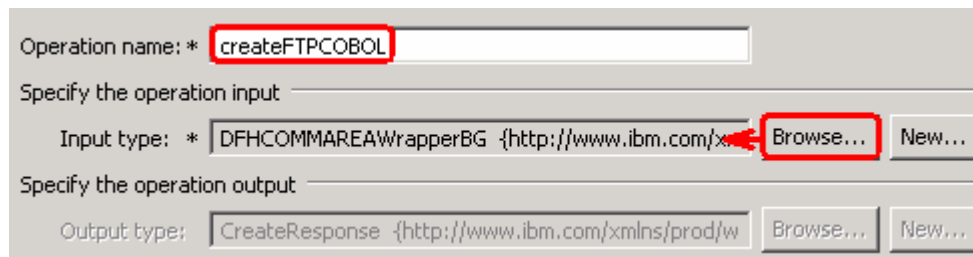
- ___ 10. From the Operations screen, click **Add...**
 - ___ a. For **Operation kind**, select **Create** from the drop down list
 - ___ b. For **Data type for the operation input**, select **User defined type** from the drop down list
 - ___ c. Select the check box next to **Enable response type for the operation**



- ___ d. Click **Next**

The Data type for input and output are populated based on the selection of the Data type for the operation in the previous step. Since you have chosen User defined type, the **Input type** is **blank** and because you have selected Output required box, the **Output type** is **CreateResponse**

- ___ e. For **Operation name**, enter any name, for Ex: **createFTPCOBOL**
- ___ f. For **Input type**, click **Browse** and select **DFHCOMMAREAWrapperBG** from the Data Type Selection window



- ___ g. Click **Finish**. The above defined operation, **createFTPCOBOL**, is populated under Operations list

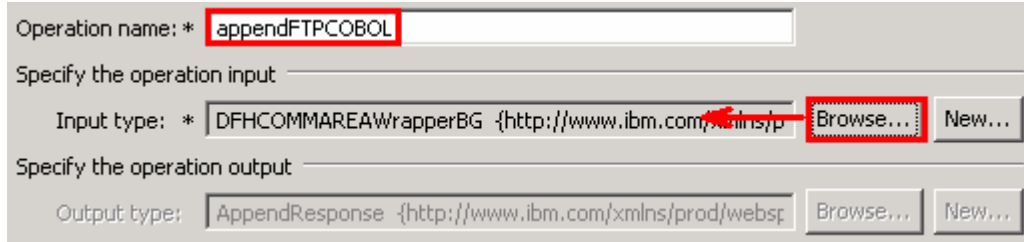


Define appendFTPCOBOL operation:

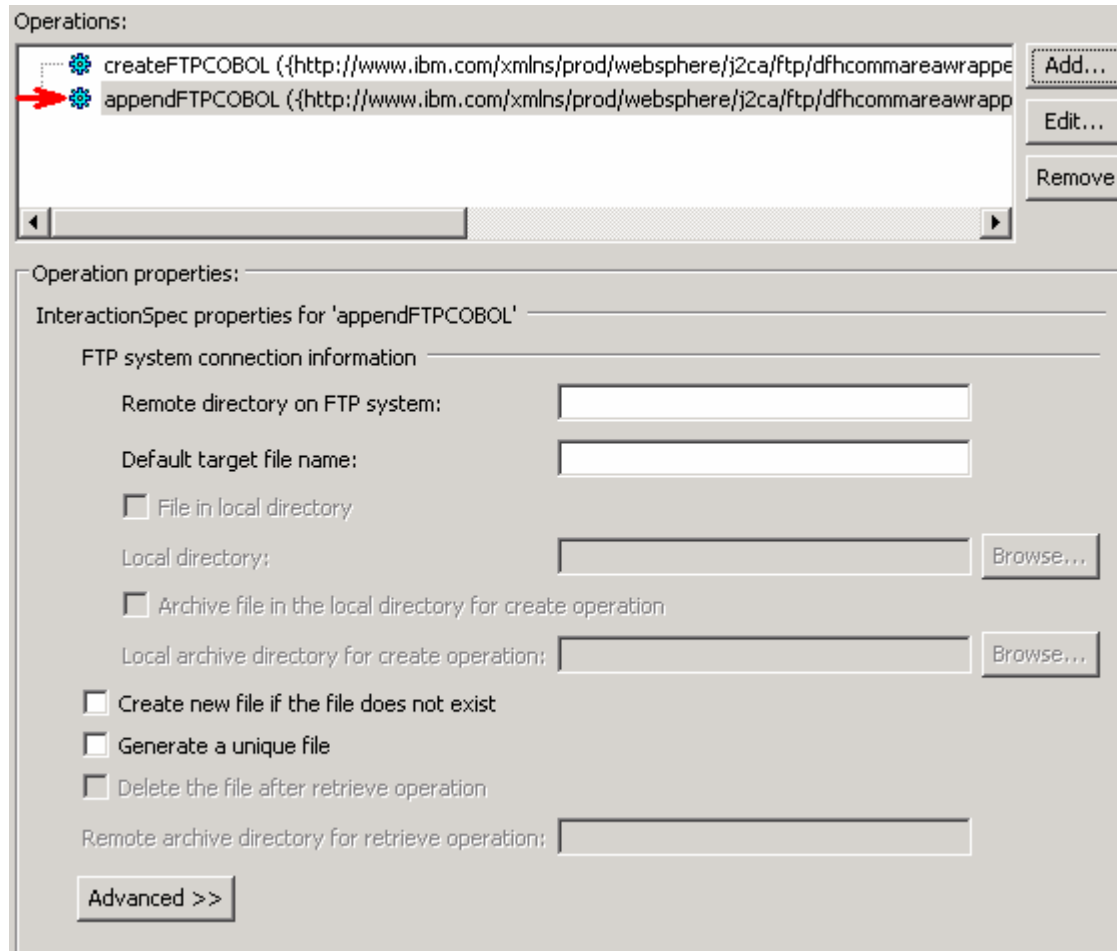
- ___ 11. From the Operations screen, click **Add...**
 - ___ a. In the Add Operation window, select **Append** from the drop down menu for Operation kind
 - ___ b. Select **User Defined Type** for the Data type

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

- ___ c. Select the check box next to **Enable response type for the operation** and click **Next**
- ___ d. For **Operation name**, enter **appendFTPCOBOL**
- ___ e. For Input type, click **Browse** and select **DFHCOMMAREAWrapperBG** from the Data Type Selection window



- ___ f. Click **Finish**. The above defined operation, **appendFTPCOBOL**, is populated under Operations list



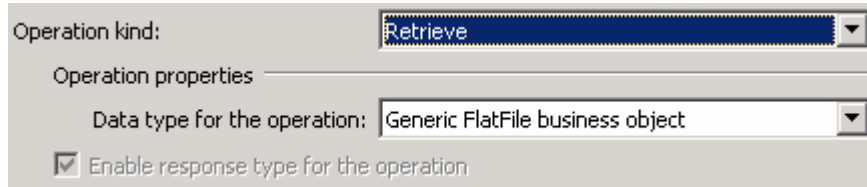
Define retrieveFTPCOBOL operation:

- ___ 12. Click **Add...** to open Add Operation window
 - ___ a. For **Operation kind**, select **Retrieve** from the drop down list

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

__ b. For **Data type for operation**, select **Generic FlatFile business object** from the drop down list

__ c. Note that the box next to **Enable response type for the operation** is checked by default



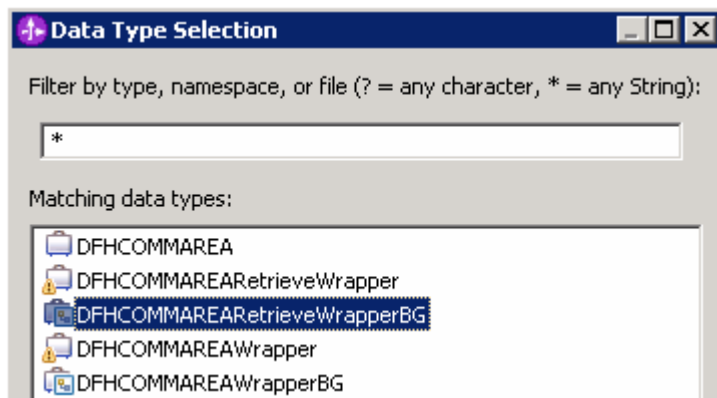
__ d. Click **Next**

In the Add Operation window, under Specify the operation input, you will see the Input type **FTPFile** (because you have selected **not to** have business graph (BG)) and you will also see the Output type **RetrieveResponseWrapper** under Specify the operation output (because the Output was selected by default).

__ e. For Operation name, enter **retrieveFTPCOBOL**

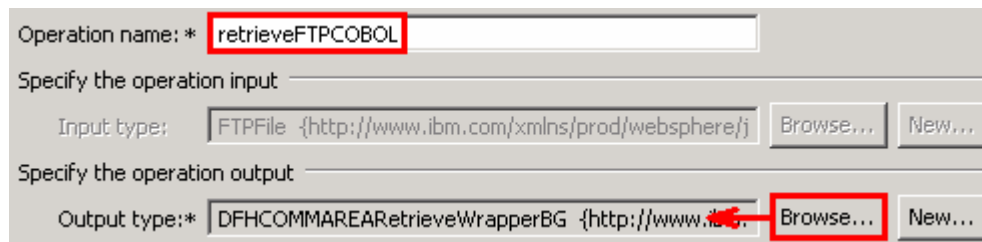
__ f. Define Output type:

- 1) Under **Specify the operation output**, click **Browse...** next to **Output type** to open a Data Type Selection window
- 2) From the Data Type Selection window, select **DFHCOMMAREARetrieveWrapperBG** under Matching data types



3) Click **OK**

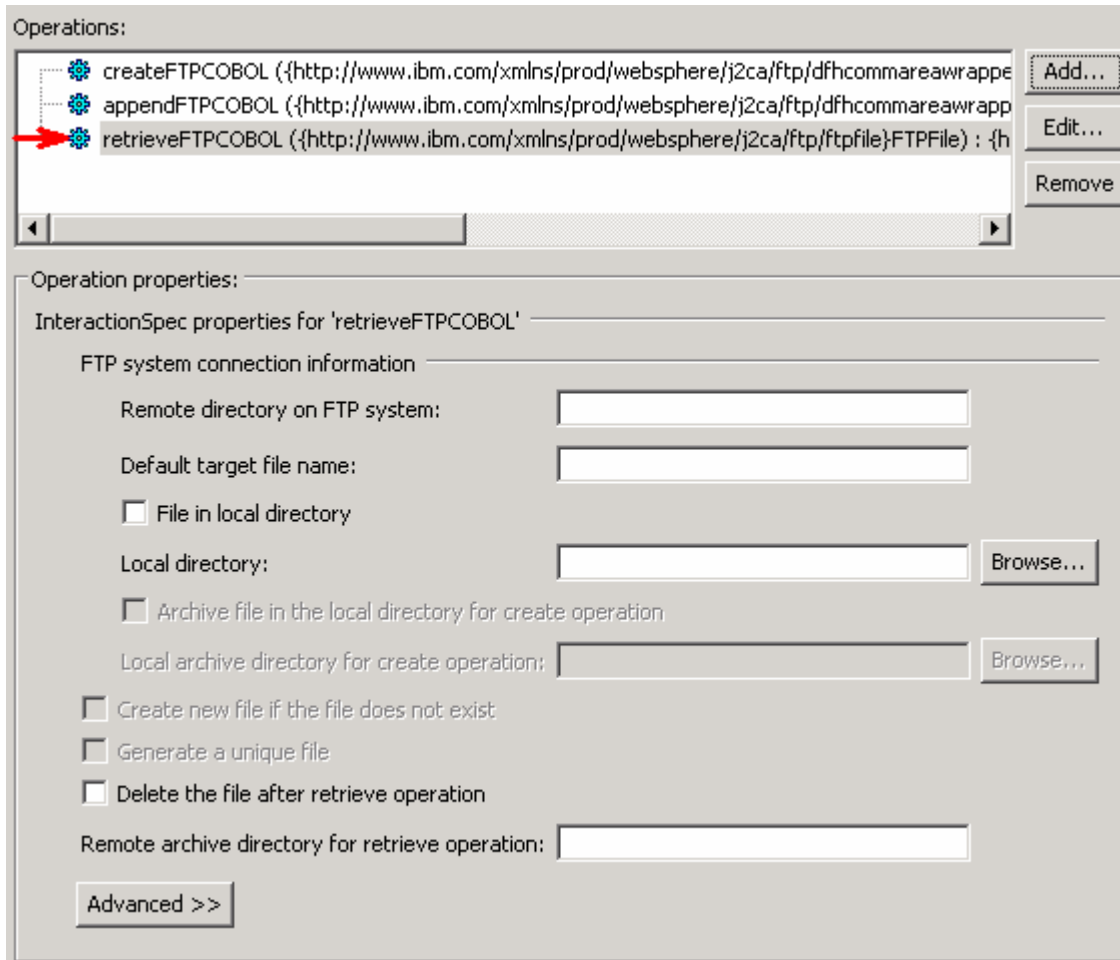
__ g. You should now see this:



__ h. Click **Finish** from the Add Operation window. The operation, retrieveFTPCOBOL, will now be displayed under Operations list.

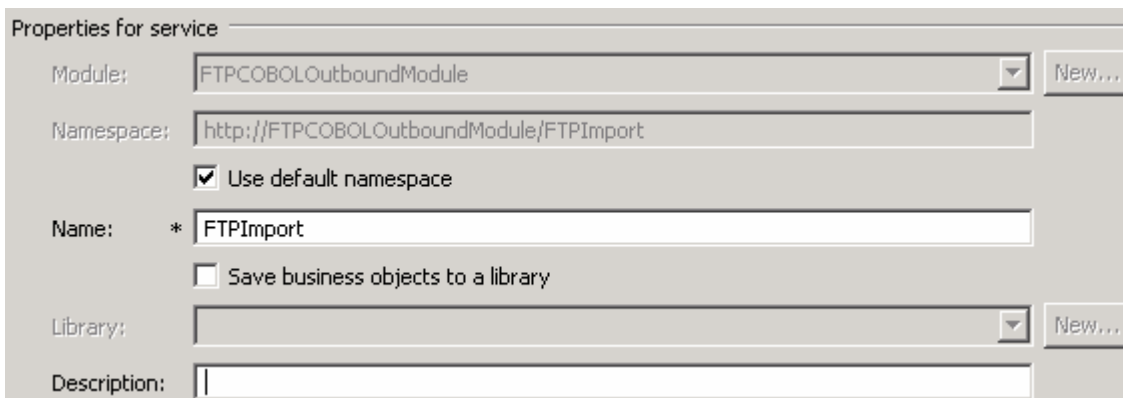
IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

- ___ i. You can click **Advanced >>** under 'InteractionSpec properties for retrieveCOBOL' to review the properties available at Interaction spec level



- ___ j. Click **Next** from Operations screen

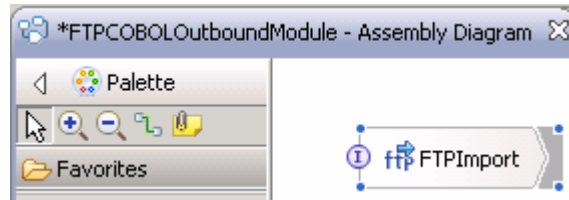
- ___ 13. From Generate Service screen, accept the default value, **FTPImport**, for **Name**



- ___ a. Click **Finish**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

- ____ 14. The Assembly diagram for FTPCOBOLOutboundModule is opened with an Import component, **FTPImport**:



- ____ 15. Save (**Ctrl + S**) changes to your assembly diagram

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

3.3. Test outbound scenario

In this part of the lab, you will use the WebSphere Process Server Test Environment to test the SCA application outbound processing for the COBOL copybook create scenario.

- ___ 1. Add the project to the WebSphere Test Environment server
 - ___ a. Right-click **WebSphere Process Server v6.2** under the Servers view and select **Add and remove projects...** from the pop-up menu
 - ___ b. From the Add and Remove Projects window, select **FTPCOBOLOutboundModuleApp** under Available projects panel and click **Add >**
 - ___ c. You will now see the **FTPCOBOLOutboundModuleApp** added to the **Configured projects**
 - ___ d. Click **Finish** and wait until the project is being published onto the server. The server is started in Debug mode if it is not already started before
- ___ 2. Open the test client for the module
 - ___ a. From the Business Integration perspective, right-click the **FTPCOBOLOutboundModule** and select **Test > Test Module**
 - ___ b. The **FTPCOBOLOutboundModule_Test** window is opened in the Assembly editor

You have three operations that were defined in the previous part in this module:

- createFTPCOBOL
- appendFTPCOBOL
- retrieveFTPCOBOL

Test Create operation:

- ___ 3. Under **Detailed Properties**, for the **Operation** field, select **createFTPCOBOL** from the drop down menu

▶ General Properties	
▼ Detailed Properties	
<u>Configuration:</u>	Default Module Test
<u>Module:</u>	FTPCOBOLOutboundModule
<u>Component:</u>	FTPImport
<u>Interface:</u>	FTPImport
<u>Operation:</u>	createFTPCOBOL

Fill out the fields for Initial request parameters:

- ___ a. For **Filename**, enter any name, for Ex: **FTPCOBOLTest.txt**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

__ b. For **includeEndBODelimiter**, enter **###**

Initial request parameters

Name	Type	Value
createFTPCOBOLInput	DFHCOMMAREAWrapperBG	✓
verb	verb<string>	✓ CREATE
DFHCOMMAREAWrapper	DFHCOMMAREAWrapper	✓
DirectoryPath	string	✓
Filename	string	✓ FTPCOBOLTest.txt
ChunkInfo	string	✓
FtpServerHostName	string	✓
FtpServerEventDirectory	string	✓
DataConnectionMode	string	✓
FileTransferType	string	✓
SecondServerDirectory	string	✓
SecondServerUsername	string	✓
SecondServerPassword	string	✓
FileContentEncoding	string	✓
IncludeEndBODelimiter	string	✓ ###
FileInLocalDirectory	boolean	✓ false

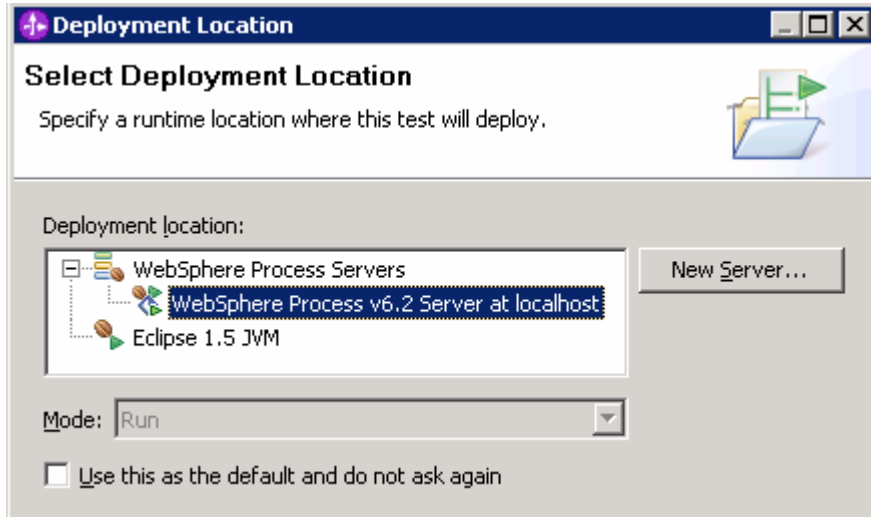
__ c. For the fields under **Content**, enter any values:

Content	DFHCOMMAREA	✓
CustomerNumber	CustomerNumber <string>	✓ FTP11
FirstName	FirstName <string>	✓ Joe
LastName	LastName <string>	✓ Sean
Street	Street <string>	✓ Burnet Rd
City	City <string>	✓ Austin
Country	Country <string>	✓ USA
Phone	Phone <string>	✓ 5128380000
PostalCode	PostalCode <string>	✓ 78758

__ d. Click **Continue** button under Events

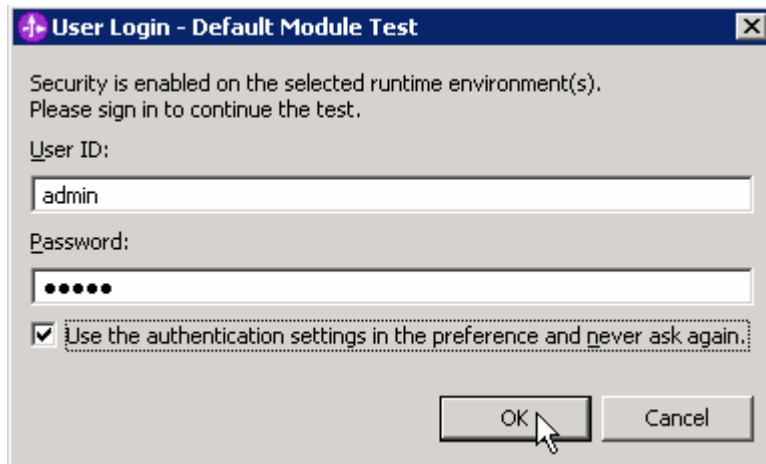
IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

- ___ e. From Deployment Location window, select **WebSphere Process Servers > WebSphere Process Server v6.2 at localhost** and click **Finish**



- ___ f. Provide Administrator User ID and Password

- 1) Optionally, select the box 'Use the authentication settings in the preference and never ask again' so that you do not have to enter the credentials next time when you start the test client



- ___ g. Click **OK** from Adding/Removing Projects pop-up, if asked. Optionally, you can select 'Do not show again' so that you are not required to do this next time when you start the test client

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

___ h. Verify the created file and its contents:

- 1) You should see a window similar to this, which contains the data you just entered in the previous steps:

The screenshot shows the 'Events' pane on the left with a tree view containing 'Invoke (FTPImport:createFTPCOBOL)', 'Invoke started', 'Invoke (FTPImport:createFTPCOBOL)', 'Return (FTPImport:createFTPCOBOL)' (highlighted with a red box), and 'Invoke returned'. The 'Detailed Properties' pane on the right shows the following information:

- Module: [FTPCOBOLOutboundModule](#)
- Component: [FTPImport](#)
- Interface: [FTPImport](#)
- Operation: [createFTPCOBOL](#)

Return parameters:

Name	Type	Value
createFTPCOBOLOutput	CreateResponse	✓
Filename	string	✓ FTPCOBOLTest.txt

- 2) You will see a new file, **FTPCOBOLTest.txt** created under **OutDir** on your FTP server. Open that file to verify the content

Test Append operation:

- ___ 4. Under **Detailed Properties**, note that the Operation is **appendFTPCOBOL**
 - ___ a. Fill out the fields for Initial request parameters:
 - ___ b. For **fileName**, enter **FTPCOBOLTest.txt**
 - ___ c. For **includeEndBODElimiter**, enter **###**

appendFTPCOBOLInput	DFHCOMMAREAWrapperBG	✓
verb	verb<string>	✓ CREATE
DFHCOMMAREAWrapper	DFHCOMMAREAWrapper	✓
DirectoryPath	string	✓
Filename	string	✓ FTPCOBOLTest.txt
ChunkInfo	string	✓
FtpServerHostName	string	✓
FtpServerEventDirectory	string	✓
DataConnectionMode	string	✓
FileTransferType	string	✓
SecondServerDirectory	string	✓
SecondServerUsername	string	✓
SecondServerPassword	string	✓
FileContentEncoding	string	✓
IncludeEndBODElimiter	string	✓ ###

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

__ d. Enter the values of your choice for the fields under **Content**

Content	DFHCOMMAREA	✓
CustomerNumber	CustomerNumber <string>	✓ XYZ9
FirstName	FirstName <string>	✓ James
LastName	LastName <string>	✓ Doe
Street	Street <string>	✓ Duval Rd
City	City <string>	✓ Austin
Country	Country <string>	✓ USA
Phone	Phone <string>	✓ 5128380000
PostalCode	PostalCode <string>	✓ 78727

__ e. Click **Continue** button under Events

__ f. Verify the results:

- 1) You should see a window similar to this, which contains the data you just entered in the previous steps:

Events

- Invoke (FTPImport:createFTPCOBOL)
 - Invoke started
 - Invoke (FTPImport:createFTPCOBOL)
 - Return (FTPImport:createFTPCOBOL)
 - Invoke returned
- Invoke (FTPImport:appendFTPCOBOL)
 - Invoke started
 - Invoke (FTPImport:appendFTPCOBOL)
 - Return (FTPImport:appendFTPCOBOL)
 - Invoke returned

General Properties

Detailed Properties

Module: [FTPCOBOLOutboundModule](#)
 Component: [FTPImport](#)
 Interface: [FTPImport](#)
 Operation: [appendFTPCOBOL](#)

Return parameters:

Name	Type	Value
appendFTPCOBOLOutput	AppendResponse	✓
Filename	string	✓ FTPCOBOLTest.txt

__ g. To verify your test results, open the FTPCOBOLTest.txt file from **OutputDir** of your FTP server and it should contain the text entered under content field in your test client. The highlighted text is the appended content and note the end BO delimiter, **###**, at the end of each COBOL BO

FTP11Joe	Sean	Burnet Rd	Austin	USA	5128380000	78758	###
XYZ9	James	Duval Rd	Austin	USA	5128380000	78758	###

Test Retrieve operation:

5. Under **Detailed Properties**, note that the Operation is **retrieveCOBOL**
 - a. Fill out the fields for Initial request parameters:
 - b. For **filename**, enter **FTPCOBOLTest.txt**
 - c. **LocalDirectoryPath**: provide absolute path to any local directory
 - d. **splitFunctionClassName**: **com.ibm.j2ca.utils.filesplit.SplitByDelimiter**

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

__ e. **splitCriteria: ###**

retrieveFTPCOBOLInput	FTPFile	✓
DirectoryPath	string	✓
Filename	string	✓ FTPCOBOLTest.txt
ChunkInfo	string	✓
FtpServerHostName	string	✓
FtpServerEventDirectory	string	✓
DataConnectionMode	string	✓
FileTransferType	string	✓
SecondServerDirectory	string	✓
SecondServerUsername	string	✓
SecondServerPassword	string	✓
FileContentEncoding	string	✓
IncludeEndBODelimiter	string	✓
FileInLocalDirectory	boolean	✓ false
LocalDirectoryPath	string	✓ C:\Labfiles62\FTPCobol\LocalEventDir
LocalArchivingEnabledForCreate	boolean	✓ false
LocalArchiveDirForCreate	string	✓
StagingDirectory	string	✓
GenerateUniqueFile	boolean	✓ false
CreateFileIfNotExists	boolean	✓ false
ScriptFileParameters	string[]	60
SplittingFunctionClassName	string	✓ com.ibm.j2ca.utils.filesplit.SplitByDelimiter
SplitCriteria	string	✓ ###
DeleteOnRetrieve	boolean	✓ false
ArchiveDirectoryForRetrieve	string	✓

__ f. Click **Continue** button under Events

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

__ g. Verify the results:

- 1) You should see a window similar to this, which contains the data you just entered in the previous steps:

retrieveFTPCOBOLOutput	DFHCOMMAREARetrieveWrapperBG	✓
verb	verb <string>	✓
DFHCOMMAREARetrieveWrapper	DFHCOMMAREARetrieveWrapperBG	✓
verb	verb <string>	✓
DFHCOMMAREARetrieveWrapper	DFHCOMMAREARetrieveWrapper	✓
Content	DFHCOMMAREA[]	6
Content[0]	DFHCOMMAREA	✓
CustomerNumber	CustomerNumber <string>	✓ FTP11
FirstName	FirstName <string>	✓ Joe
LastName	LastName <string>	✓ Sean
Street	Street <string>	✓ Burnet Rd
City	City <string>	✓ Austin
Country	Country <string>	✓ USA
Phone	Phone <string>	✓ 5128380000
PostalCode	PostalCode <string>	✓ 78758
Content[1]	DFHCOMMAREA	✓
CustomerNumber	CustomerNumber <string>	✓ XYZ9
FirstName	FirstName <string>	✓ James
LastName	LastName <string>	✓ Doe
Street	Street <string>	✓ Duval Rd
City	City <string>	✓ Austin
Country	Country <string>	✓ TX
Phone	Phone <string>	✓ 5128380000
PostalCode	PostalCode <string>	✓ 78727

___ 6. Restore the Sever Configuration

- ___ a. Right-click **WebSphere Process Server v6.2** under the Servers view and select **Add and remove projects...** from the pop-up menu
- ___ b. Select **FTPCOBOLOutboundModuleApp** under Configured projects and click **< Remove**
- ___ c. Click **Finish** after you see the application moved to Available projects. Wait until the application is being unpublished

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

What you did in this exercise

In this lab, you started with importing the FTP Adapter RAR file into your WebSphere Integration Developer new workspace. Next, you made use of the External Data wizard to generate business object and wrapper definitions from a COBOL program source file. Then you continued to with External Service wizard available in WebSphere Integration Developer to generate other artifacts for inbound and then for outbound.

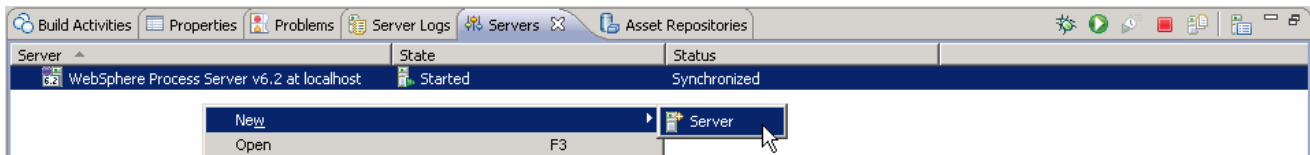
At the end of each part, you deployed and tested the adapter application for operations defined during the external service wizard.

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

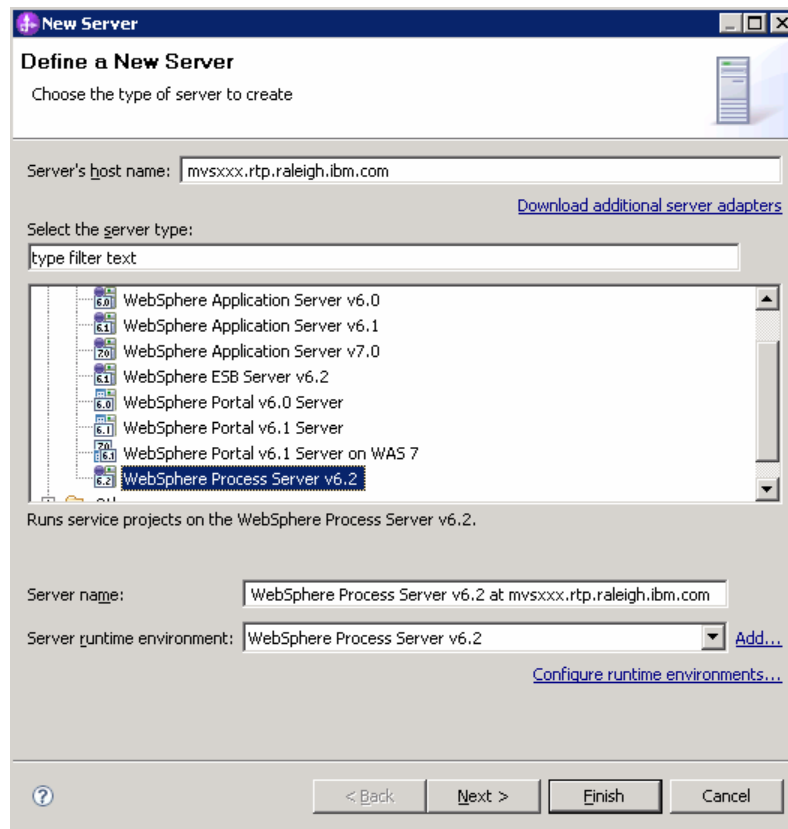
Task: Adding remote server to WebSphere Integration Developer test environment

This task describes how to add a remote server to the WebSphere Integration Developer Test environment. This example uses a z/OS machine.

- ___ 1. Define a new remote server to WebSphere Integration Developer.
 - ___ a. Right click the background of the Servers view to access the pop-up menu.
 - ___ b. Select **New → Server**.



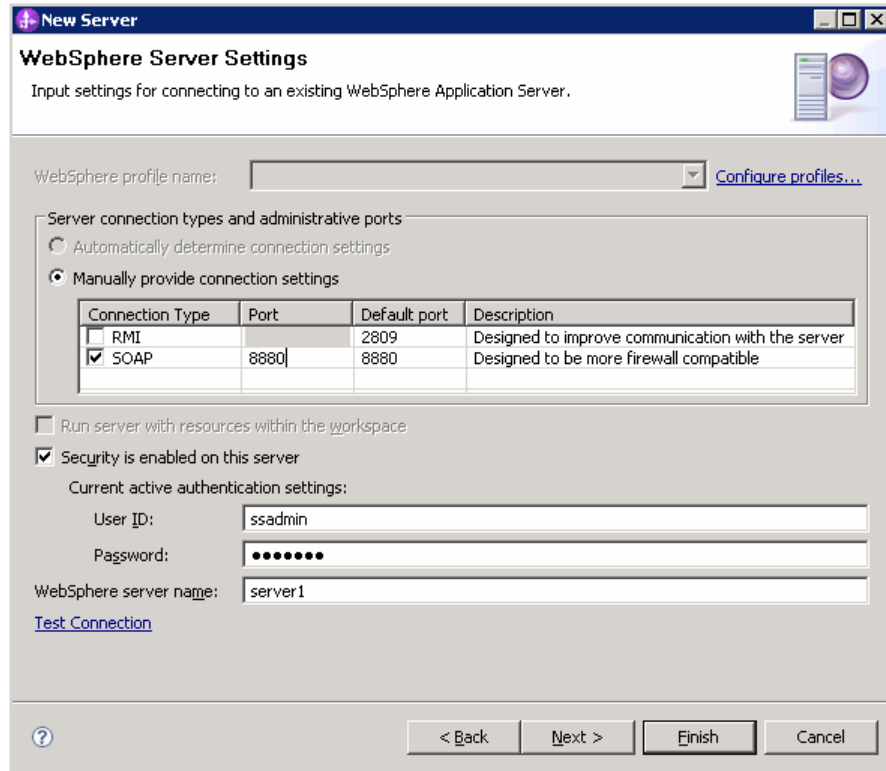
- ___ c. In the New Server dialog, specify the remote server's host name, **<HOSTNAME>**.
- ___ d. Ensure that the appropriate server type, **'WebSphere Process Server v6.2'** or **'WebSphere ESB Server v6.2'**, is highlighted in the server type list



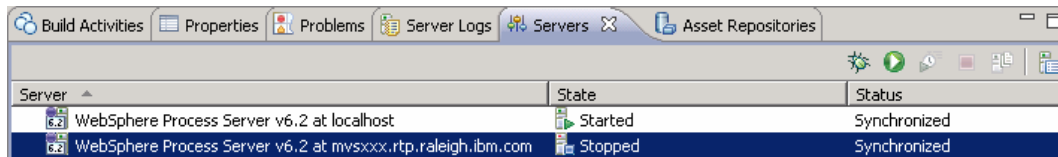
- ___ e. Click **Next**.

IBM WEBSHERE ADAPTER 6.2 – LAB EXERCISE

- ___ f. On the WebSphere Server Settings page, leave the radio button for **Manually provide connection settings** selected, and select the box for SOAP
- ___ g. Enter the correct setting (<SOAP_PORT>) for **Port** column
- ___ h. If security is enabled on your server, select the box for **'Security is enabled on this server'** and enter <USERID> for the User ID and <PASSWORD> for the password.



- ___ i. Click **Finish**.
- ___ j. The new server should be seen in the Server view.



- ___ 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>'

User ID : <USERID>

Password : <PASSWORD>

IBM WEBSPHERE ADAPTER 6.2 – LAB EXERCISE

__ 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 sssr01 open for e-business; process id is 0000012000000002
```