

## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

## Flat File 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 scenario .....	6
2.1. Prepare for the inbound scenario.....	7
2.2. Configure inbound using the external service wizard .....	12
2.3. Add Java component .....	17
2.4. Test inbound scenario.....	19
Part 3: Outbound scenario .....	21
3.1. Prepare for the outbound scenario .....	22
3.2. Configure outbound using external service wizard .....	24
3.3. Test outbound scenario .....	30
Solution instructions .....	39
What you did in this exercise .....	41
Task: Adding remote server to WebSphere Integration Developer test environment .....	42

### 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 V7.0 installed and updated with latest fixes
- WebSphere Process Server V7.0 Test Environment installed and updated latest fixes
- Extract Labfiles70.zip to your C:\ (your root) drive

## IBM WEBSPPHERE ADAPTER 7.0 – LAB EXERCISE

### What you should be able to do

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

- Import Flat File 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 Flat File 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 7.0 – LAB EXERCISE

Reference variable	Windows location	AIX® or UNIX® location
<WID_HOME>	C:\Program Files\IBM\WID7Beta	
<WPS_HOME>	C:\<WID_HOME>\runtimes\bi_v70	
<FFADAPTER_HOME>	<WID_HOME>\ResourceAdapters\FlatFile_7.0.0.0\deploy	
<LAB_FILES>	C:\Labfiles70	/tmp/Labfiles70
<WORKSPACE>	<LAB_FILES>\FlatFileCobol\workspace	
<EVENT_DIR>	<LAB_FILES>\FlatFileCobol\eventdir	
<ARCHIVE_DIR>	<LAB_FILES>\FlatFileCobol\archivedir	
<OUT_DIR>	<LAB_FILES>\FlatFileCobol\outdir	
<FFFFILES>	<LAB_FILES>\FFFfiles	
<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:\Labfiles70\ with C:/Labfiles70/

---

## IBM WEBSPHERE ADAPTER 7.0 – 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 following table is related to where you are running the remote test environment:

Reference variable	Example: Remote Windows test server location	Example: Remote z/OS <sup>®</sup> 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	

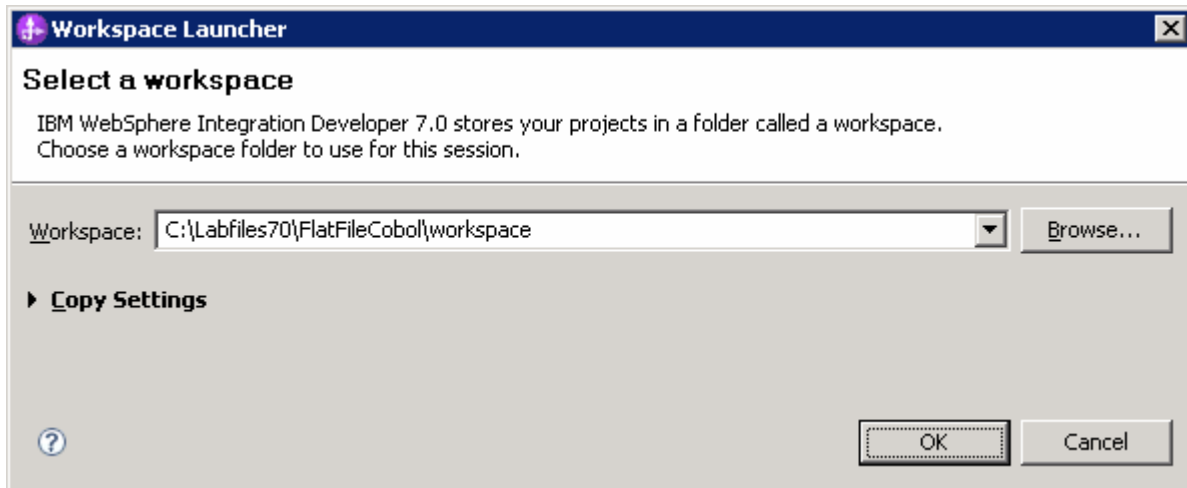
Instructions for using a remote testing environment, such as z/OS, AIX or Solaris, can be found at the end of this document, in the section [“Task: Adding remote server to WebSphere Integration Developer test environment”](#).


IBM WEBSHERE ADAPTER 7.0 – 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 V7.0 with a new workspace and then define required WebSphere Environment Variables using the administrative console of WebSphere Process Server V7.0

- \_\_\_ 1. Extract the provided Labfiles70.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. Start the WebSphere Integration Developer V7.0 with a new workspace
  - \_\_\_ a. Select **Start > All Programs > IBM WebSphere Integration Developer > IBM WebSphere Integration Developer V7.0 > WebSphere Integration Developer V7.0**
  - \_\_\_ b. From the Workspace Launcher window, enter **<WORKSPACE>** for the Workspace field



- \_\_\_ 3. Click the  button on the right corner to close the Welcome page and proceed with the workbench
- \_\_\_ 4. 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:

Name	Value
FF_EVENT	C:\Labfiles70\FlatFileCobol\eventdir
FF_ARCHIVE	C:\Labfiles70\FlatFileCobol\archivedir
FF_OUT	C:\Labfiles70\FlatFileCobol\outdir

## IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

---

### **Part 2: Inbound scenario**

In this part, you will import the Flat File 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.

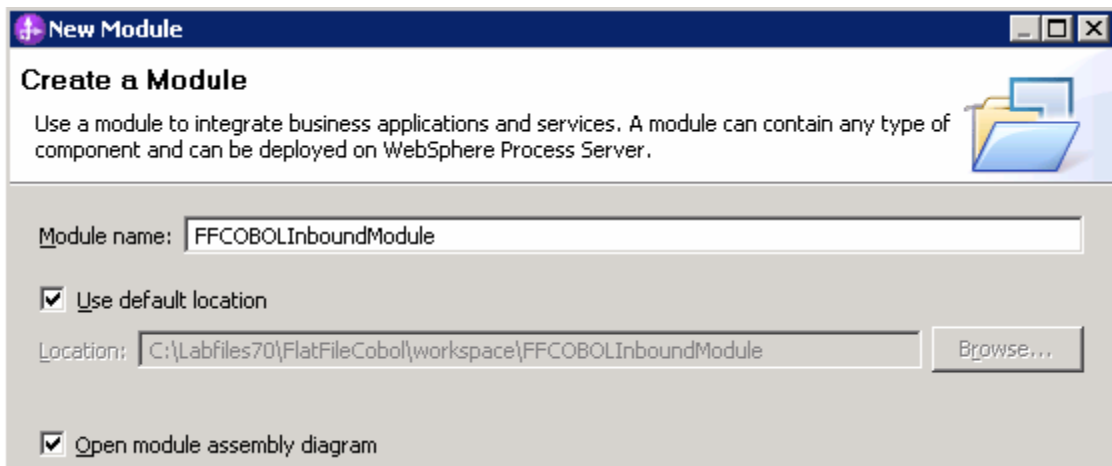
You will make use of the new 'COBOL, C, or PL/I data binding' introduced in V6.2, while running the External service wizard to generate the required artifacts to convert COBOL copybook files to business objects during inbound processing.

## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

## 2.1. Prepare for the inbound 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.

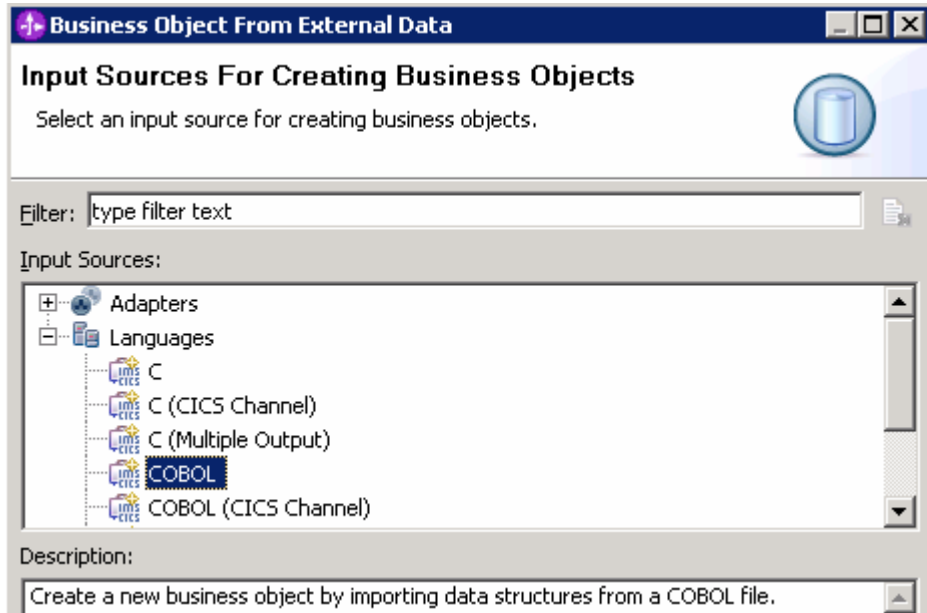
- \_\_\_ 1. Create FFCOBOLInboundModule
  - \_\_\_ a. From the Business Integration window, right-click and select **New > Module**
  - \_\_\_ b. From the New Module window, enter **FFCOBOLInboundModule** for the Module Name
  - \_\_\_ c. Ensure that the box next to **Open module assembly diagram** is checked



- \_\_\_ d. Click **Finish**
  - \_\_\_ e. You will now see a new module, FFCOBOLInboundModule, created in 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, FFCOBOLInboundModule, and select **New > Business Object from External Data**

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

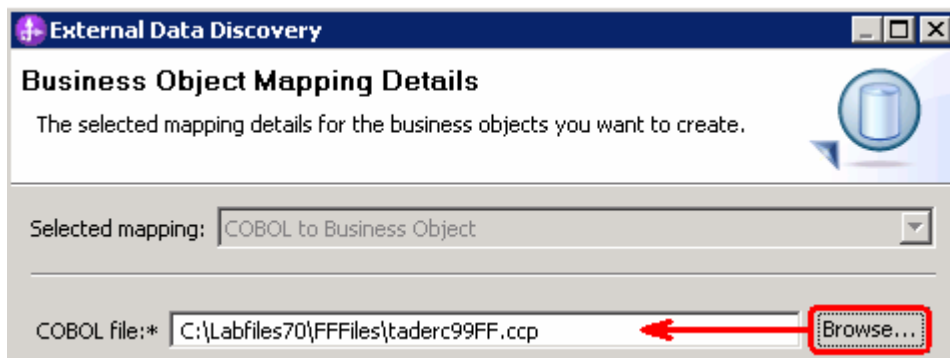
\_\_ b. From the Business Object from External Data window, select **Languages > COBOL**



\_\_ c. Click **Next**

\_\_ d. In the Business Object Mapping Details window,

- 1) Note that the **Selected mapping** value is **COBOL to Business Object**
- 2) For COBOL file, click **Browse...** and select the **traderc99FF.ccp** file located in **<FFFILES>**

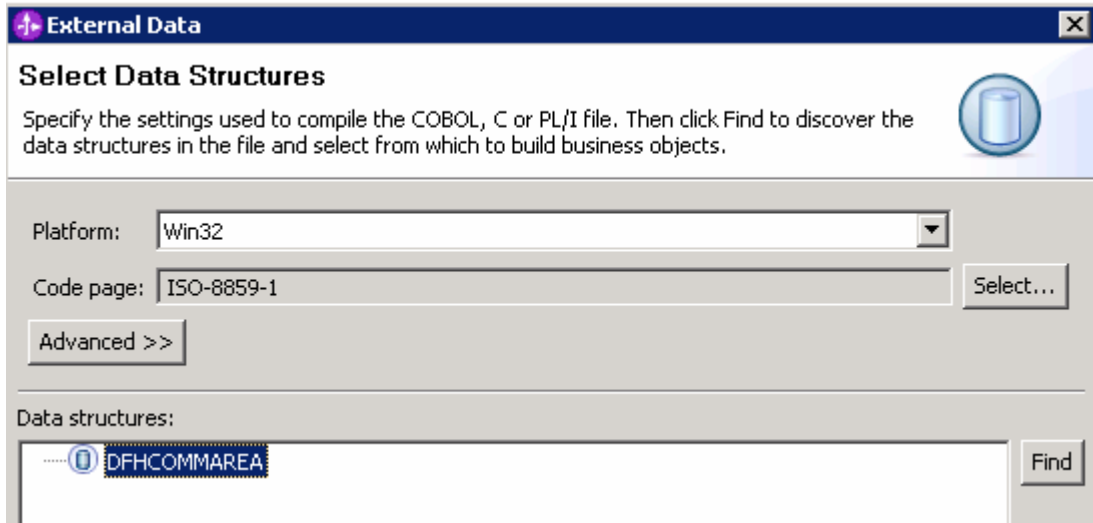


3) Click **Next**

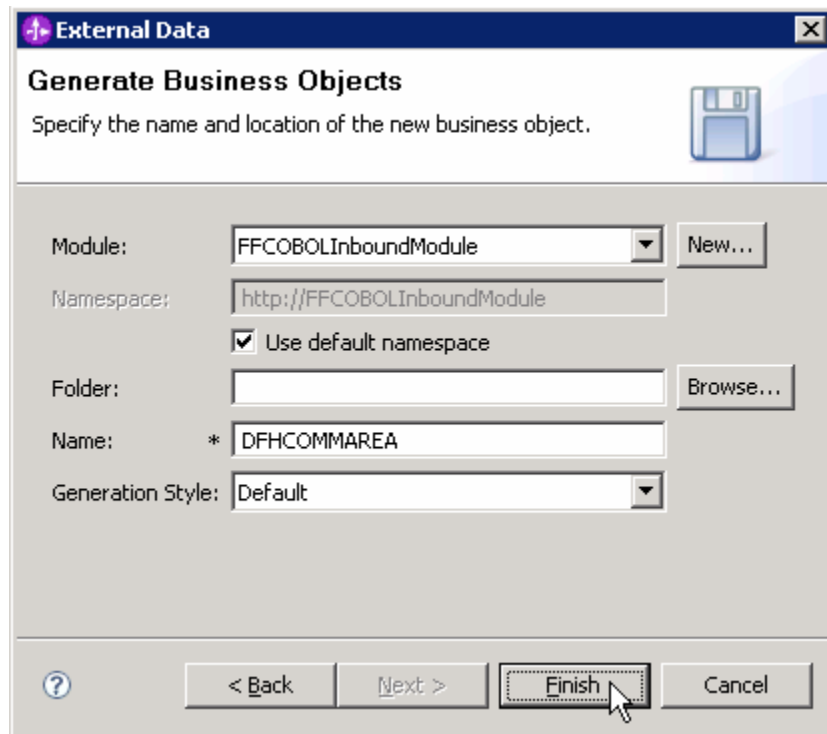


IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

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

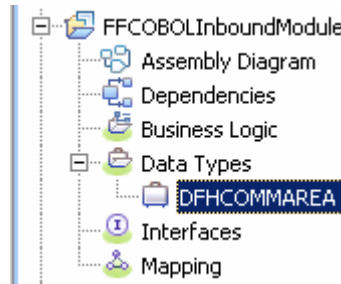


- \_\_\_ f. Select **DFHCOMMAREA** and click **Next**
- \_\_\_ g. In the Generate Business Objects screen, ensure that the selected Module is **FFCOBOLInboundModule** and click **Finish**

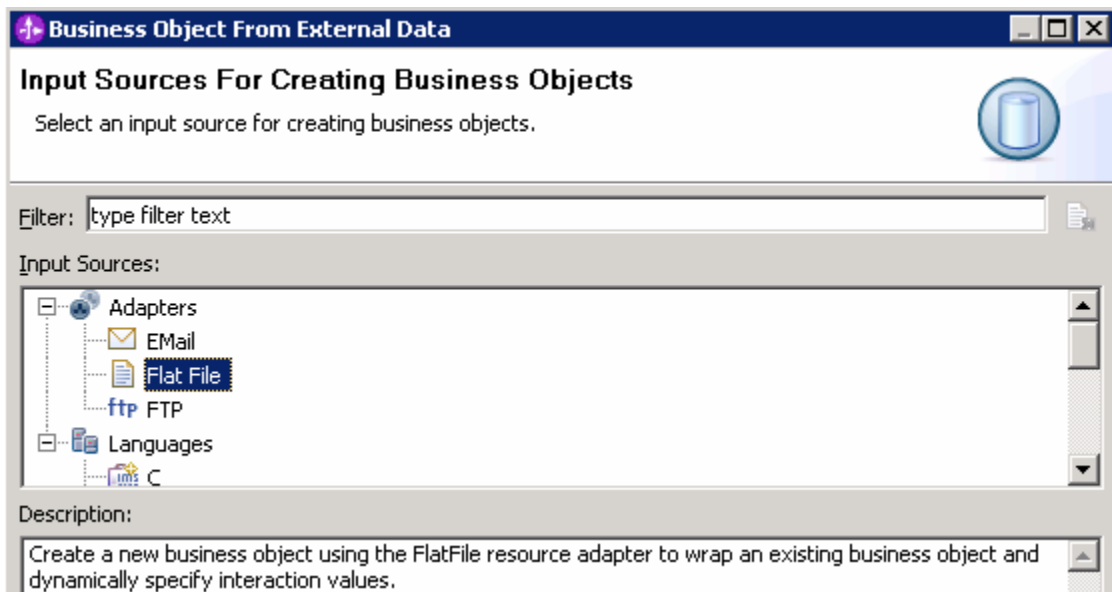


IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

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**
  - \_\_\_ b. From the External Data window, select **Adapters > Flat File** under Available Types:



- \_\_\_ c. From the next screen, ensure that the correct module, **FFCOBOLInboundModule**, is selected and click **Next**

---

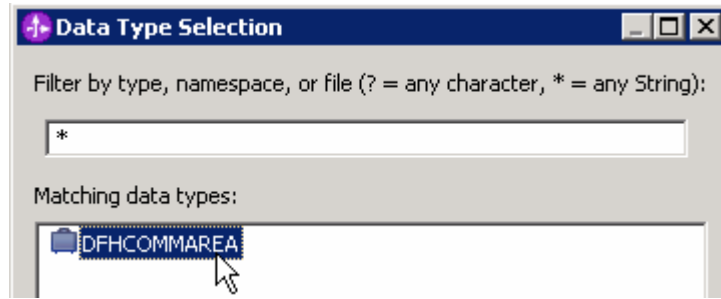
**Note:** The resource adapter archive file is imported and a new connector project, **CWYFF\_FlatFile**, is listed under Business Integration view.

---

- \_\_\_ d. Provide the below values in Business Object Properties screen:
  - 1) Data type: click **Browse....** A Data Type selection window is opened

## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

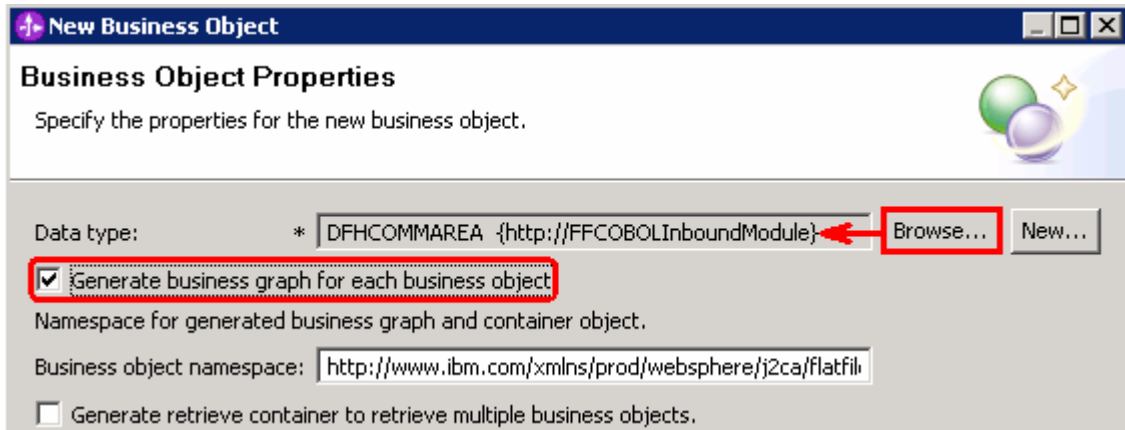
2) Select **DFHCOMMAREA** under Matching data types



3) Click **OK**

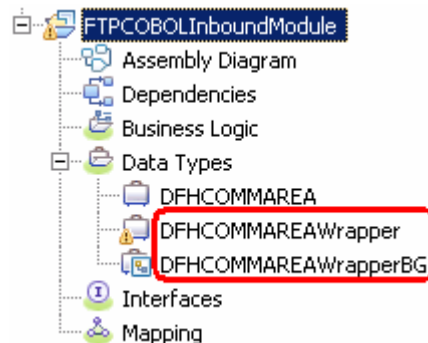
\_\_\_ e. To generate a business graph, select the **Generate business graph for each business object** check box

\_\_\_ 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

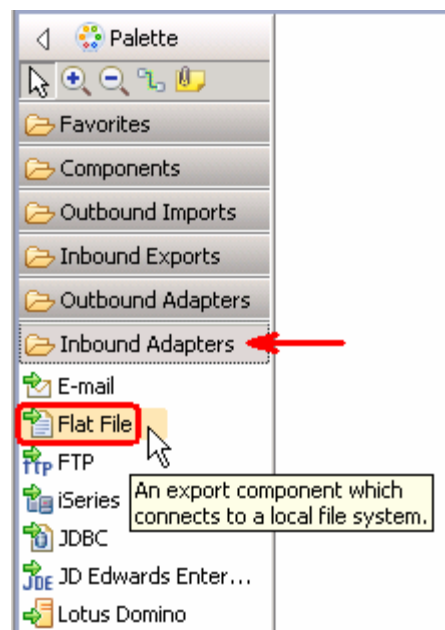


## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

## 2.2. Configure inbound using the external service wizard

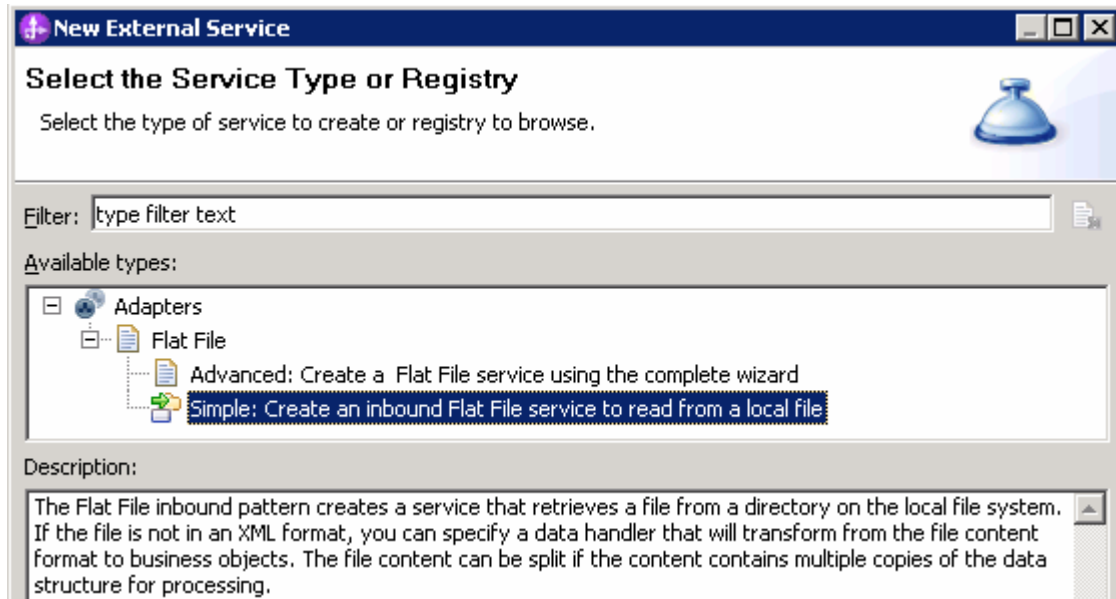
In this part of the lab you will use the new external service wizard and use the new 'COBOL, C, or PL/I data binding' to create the required artifacts to test the inbound scenario.

- \_\_\_ 1. Ensure that the FFCOBOLInboundModule assembly diagram is opened in Assembly editor
  - \_\_\_ a. In the Business Integration view, expand FFCOBOLInboundModule 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**:
- \_\_\_ 3. Under Inbound Adapters, click the **Flat File** and then click the empty canvas of the assembly diagram



- \_\_\_ 4. From the New External Service window, expand **Adapters > Flat File** and select **Simple: Create an inbound Flat File service to read from a local file**

## IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE



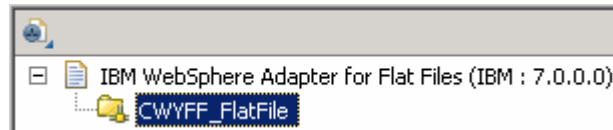
\_\_ a. Click **Next**

**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 > Flat File** and click **Next**

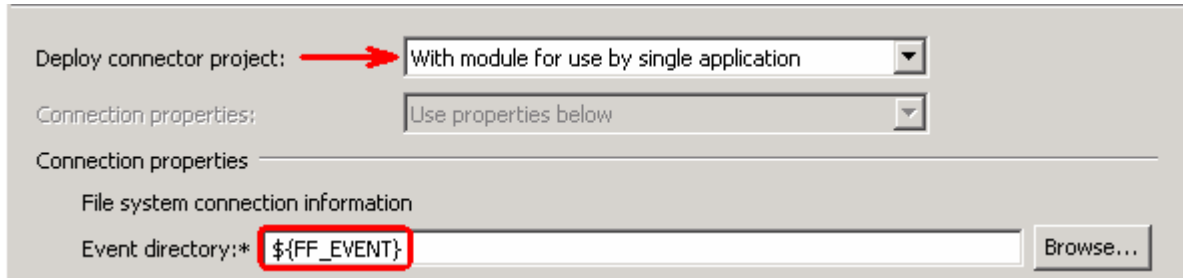
- \_\_\_ 5. On the Select an Adapter screen, select **IBM WebSphere Adapter for Flat Files (IBM : 7.0.0.0) > CWYFF\_FlatFile** and click **Next**



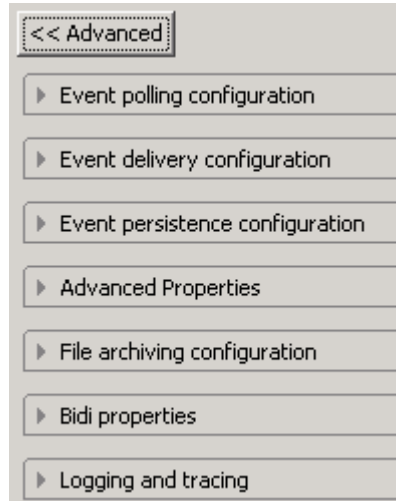
- \_\_\_ 6. Service Configuration Properties:
- \_\_ a. Deploy connector project: ensure that the default option **With module for use by single application** is selected
- \_\_ b. Under Connection Configuration, for Event directory, enter **\${FF\_EVENT}**:

**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. From WebSphere Process Server V6.1, 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.

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE



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



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

\_\_ d. Event polling configuration: This has all the polling configuration details and for this lab, you can accept the defaults.

\_\_ e. Event delivery configuration:

- 1) **Ensure once-only event delivery:** You should check this box only if you are using data source and table name in the Event persistence configuration (below). If this property is set to true, while using in-memory capability (explained below), the adapter will log a warning message. By default this is selected and you can accept the default selection.

\_\_ f. 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".

---

\_\_ g. Advanced properties:

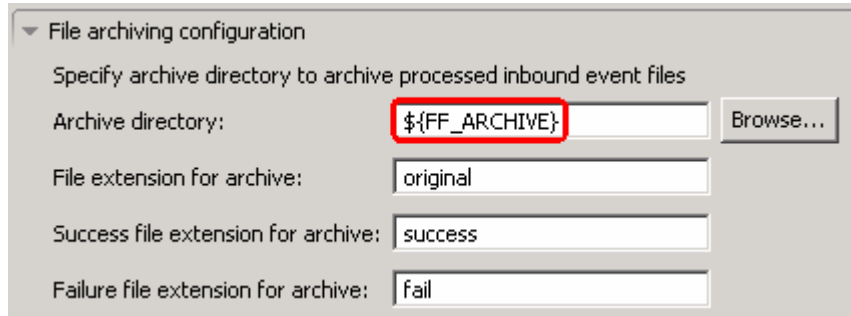
- 1) **Poll subdirectories in event directory:**

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

**Note: ‘Poll subdirectories in event directory’** is the new Boolean property added in V6.2. This property is used for recursive polling. Flat File adapter polls the given 'Event directory' and all the sub-directories present in the 'Event directory'. Refer to **‘Log and confidential trace lab’** for more details on this new feature

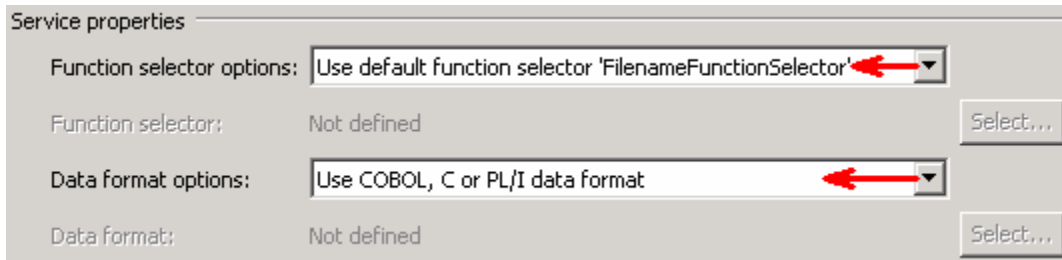
\_\_ h. File archiving configuration:

1) Archive directory: **`\${FF\_ARCHIVE}`**



\_\_\_ 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

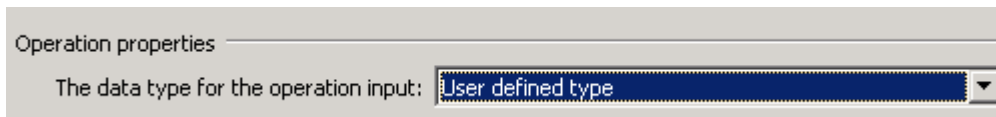


\_\_\_ 9. Check the box next to **Change logging properties for wizard** to view the output location of the log file and the logging level and click **Next**

**Define emitCOBOLFile 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**

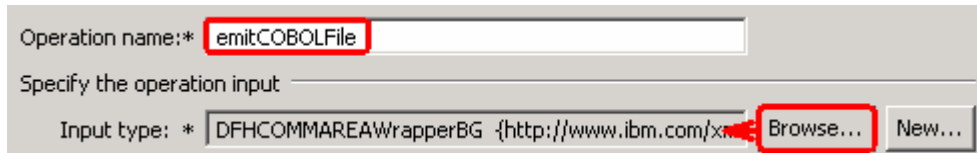


\_\_\_ 11. For **Operation name**, enter any name, for Ex: **emitCOBOLFile**

### IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

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

**Note:** You can also define the business object from this screen by selecting **New**, but you can not be able to define the wrapper or business graph. So, for this driver, you should define the wrapper using External Data wizard as explained in Part 2.1 and select it here.

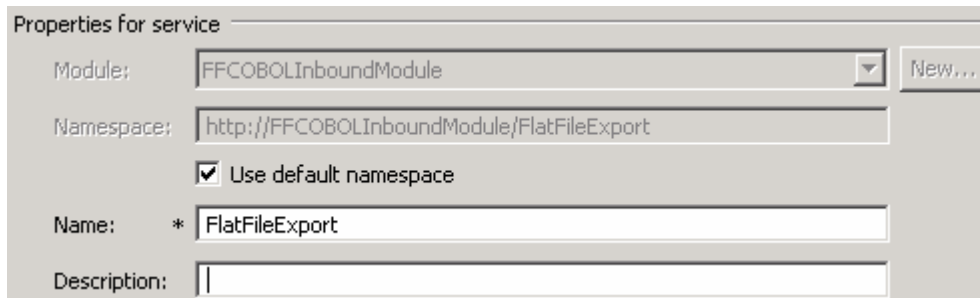


- \_\_\_ a. Click **Finish**. The defined operation, **emitCOBOLFile**, is populated under Operations list



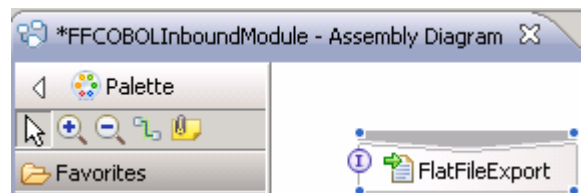
- \_\_\_ b. Click **Next** from Operations screen

- \_\_\_ 13. From Generate Service screen, accept the default value, **FlatFileExport**, for **Name**



- \_\_\_ a. Click **Finish**

- \_\_\_ 14. The Assembly diagram for FFCOBOLInboundModule is opened with an Export component, **FlatFileExport**:



- \_\_\_ a. Save (**Ctrl + S**) changes to your assembly diagram

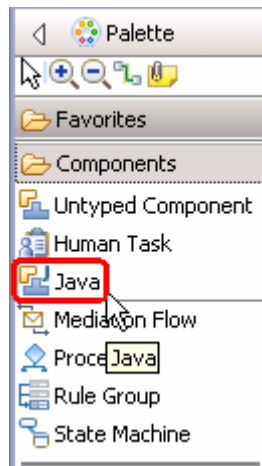



## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

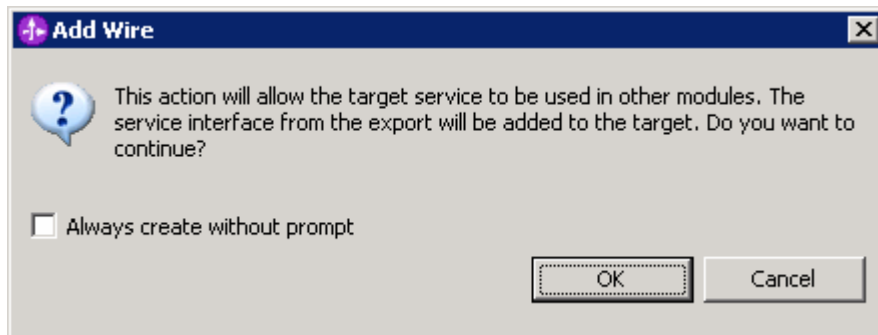
## 2.3. Add Java component

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 FFCOBOLInboundModule (if it is already not open)
  - \_\_\_ a. From the business integration view, expand **FFCOBOLInboundModule** and double click **Assembly diagram**
- \_\_\_ 2. Drop a Java component to onto the assembly diagram
  - \_\_\_ a. From the **Palette**, click **Components** to expand it



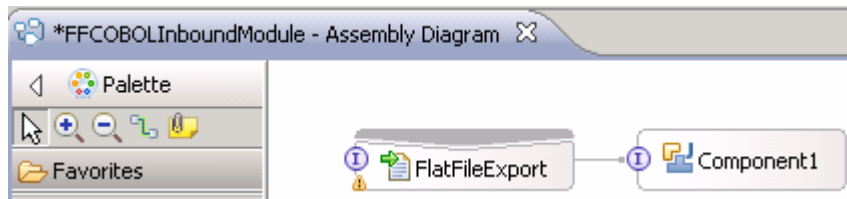
- \_\_\_ b. Click **Java** and then click the empty space of FFCOBOLInboundModule assembly diagram. This will place a new component, **Component1** on the assembly diagram.
- \_\_\_ 3. Wire the FlatFileExport to the Component1
  - \_\_\_ a. Select the **wire** () icon from the Palette
  - \_\_\_ b. Click **FlatFileExport** 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

## IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

Your assembly diagram for FFCOBOLInboundModule will look like this:



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

---

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

---

```

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

```

- \_\_ h. Save (**Ctrl + S**) and close Component1Impl.java
- \_\_ i. Save (**Ctrl + S**) and close Assembly diagram: FFCOBOLInboundModule

## IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

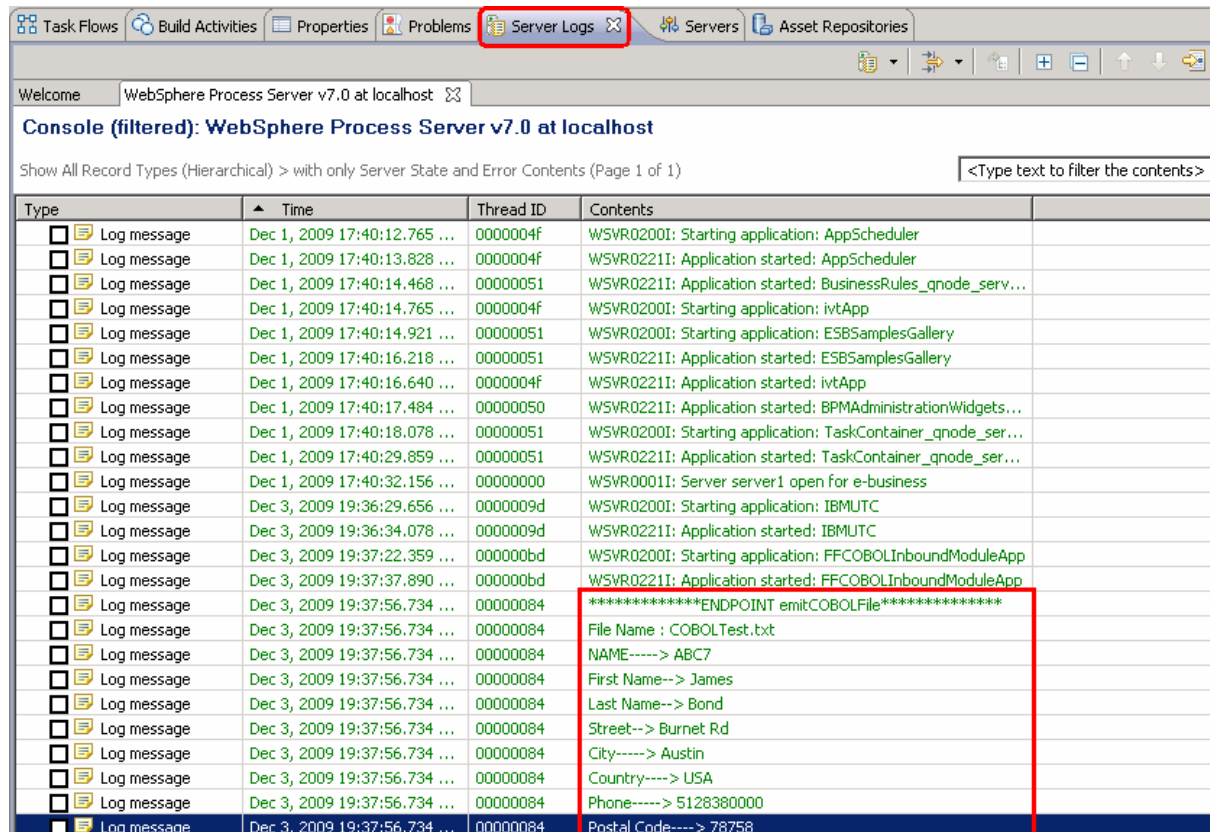
## 2.4. Test inbound scenario

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 v7.0** under the Servers view and select **Add and remove projects...** from the pop-up menu
  - \_\_\_ b. From the Add and Remove Projects window, select **FFCOBOLInboundModuleApp** under Available projects panel and click **Add >**
  - \_\_\_ c. You will now see the **FFCOBOLInboundModuleApp** added to the **Configured projects**
  - \_\_\_ d. Click **Finish** and wait until the project is being published onto the server. The server will start in Debug mode if it is not already started before
- \_\_\_ 2. Test the application by providing input files:

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

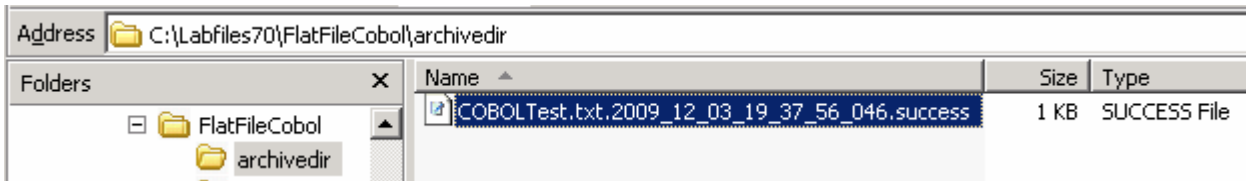
- \_\_\_ a. Copy the **COBOLTest.txt** file from **<FFFILES>** to **<EVENT\_DIR>**. The adapter will poll the copied file from the event directory and will transfer it to the archive directory
- \_\_\_ b. It will pass through the **emitFlatFileBG** method and you should see this message in your **Server Logs** view (or SystemOut.log):



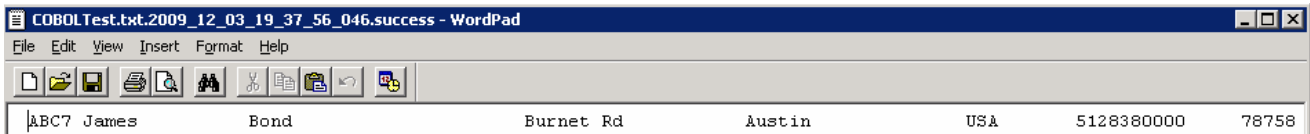
Type	Time	Thread ID	Contents
Log message	Dec 1, 2009 17:40:12.765 ...	0000004f	WSVR0200I: Starting application: AppScheduler
Log message	Dec 1, 2009 17:40:13.828 ...	0000004f	WSVR0221I: Application started: AppScheduler
Log message	Dec 1, 2009 17:40:14.468 ...	00000051	WSVR0221I: Application started: BusinessRules_qnode_serv...
Log message	Dec 1, 2009 17:40:14.765 ...	0000004f	WSVR0200I: Starting application: ivtApp
Log message	Dec 1, 2009 17:40:14.921 ...	00000051	WSVR0200I: Starting application: ESBSamplesGallery
Log message	Dec 1, 2009 17:40:16.218 ...	00000051	WSVR0221I: Application started: ESBSamplesGallery
Log message	Dec 1, 2009 17:40:16.640 ...	0000004f	WSVR0221I: Application started: ivtApp
Log message	Dec 1, 2009 17:40:17.484 ...	00000050	WSVR0221I: Application started: BPMAdministrationWidgets...
Log message	Dec 1, 2009 17:40:18.078 ...	00000051	WSVR0200I: Starting application: TaskContainer_qnode_ser...
Log message	Dec 1, 2009 17:40:29.859 ...	00000051	WSVR0221I: Application started: TaskContainer_qnode_ser...
Log message	Dec 1, 2009 17:40:32.156 ...	00000000	WSVR0001I: Server server1 open for e-business
Log message	Dec 3, 2009 19:36:29.656 ...	0000009d	WSVR0200I: Starting application: IBMUTC
Log message	Dec 3, 2009 19:36:34.078 ...	0000009d	WSVR0221I: Application started: IBMUTC
Log message	Dec 3, 2009 19:37:22.359 ...	000000bd	WSVR0200I: Starting application: FFCOBOLInboundModuleApp
Log message	Dec 3, 2009 19:37:37.890 ...	000000bd	WSVR0221I: Application started: FFCOBOLInboundModuleApp
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	*****ENDPOINT emitFlatFileBG*****
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	File Name : COBOLTest.txt
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	NAME----> ABC7
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	First Name--> James
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	Last Name--> Bond
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	Street--> Burnet Rd
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	City----> Austin
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	Country----> USA
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	Phone----> 5128380000
Log message	Dec 3, 2009 19:37:56.734 ...	00000084	Postal Code----> 78758

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

- \_\_\_ c. To verify your test results, check the <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



- \_\_\_ d. Open the file using note pad or word pad and verify the contents of it



\_\_\_ 3. Restore the Sever Configuration

- \_\_\_ a. Right-click **WebSphere Process Server v7.0** under the Servers view and select **Add and remove projects...** from the pop-up menu
- \_\_\_ b. Select **FFCOBOLInboundModuleApp** 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 7.0 – 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 WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

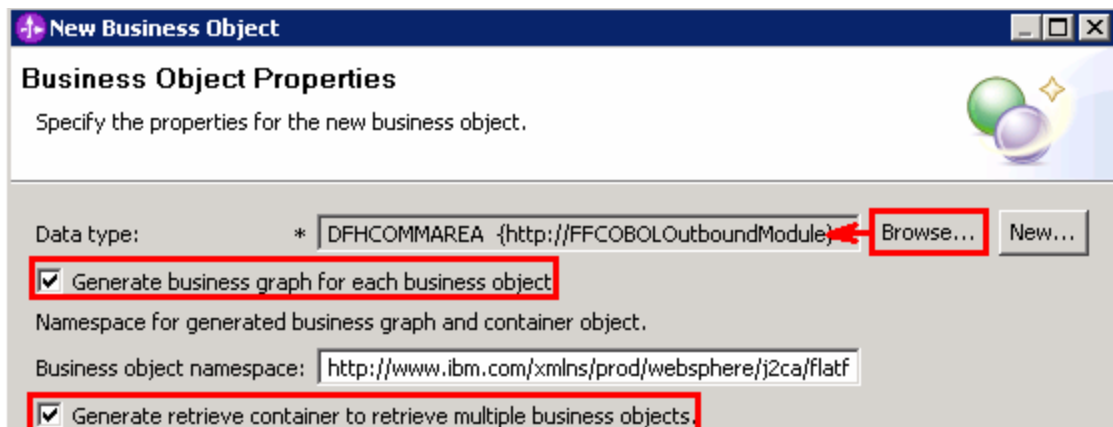
### 3.1. Prepare for the outbound scenario

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

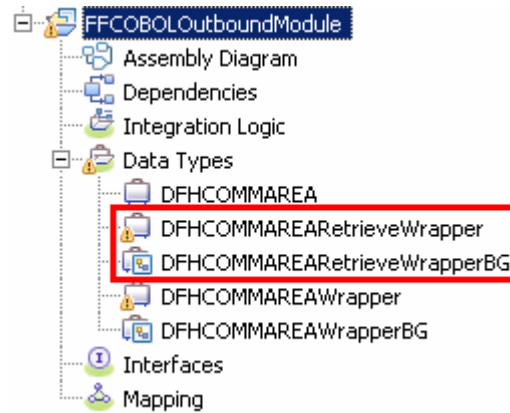
You will now see a new module, FFCOBOLOutboundModule, created from your Business Integration window

- \_\_\_ 2. Follow the instructions in **Steps 2 and 3 of Part 2.1** to generate business object and a wrapper definition under **FFCOBOLOutboundModule**
- \_\_\_ 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 Objects From External Data**
  - \_\_\_ b. From the External Data window, select **Adapters > Flat File** under Available Types:
  - \_\_\_ c. From the next screen, ensure that the correct module, **FFCOBOLOutboundModule**, is selected and click **Next**
  - \_\_\_ 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 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



## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

- \_\_\_ 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.

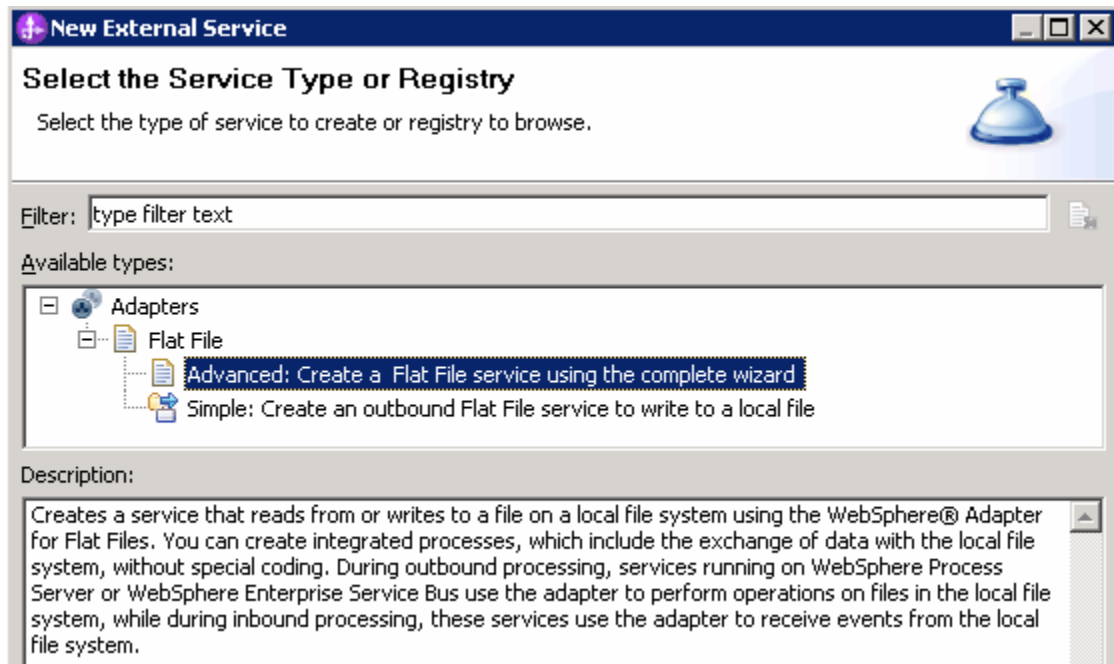


## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

### 3.2. Configure outbound using 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 outbound scenario.

- \_\_\_ 1. Ensure that the FFCOBOLOutboundModule assembly diagram is opened in Assembly editor
  - \_\_\_ a. In the Business Integration view, expand FFCOBOLOutboundModule 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 **Outbound Adapters**
  - \_\_\_ b. Under Outbound Adapters, click the **Flat File** and then click the empty canvas of the assembly diagram. The New Flat File Service wizard is opened
- \_\_\_ 3. From the New External Service window, expand **Adapters > Flat File** and select **Advanced: Create a Flat File service using the complete wizard**



- \_\_\_ 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 > Flat File** and click **Next**.

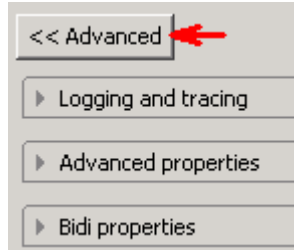
---

- \_\_\_ 4. On the Select an Adapter screen, expand **IBM WebSphere Adapter for Flat Files (IBM : 7.0.0.0)** and select **CWYFF\_FlatFile**
  - \_\_\_ a. Click **Next**



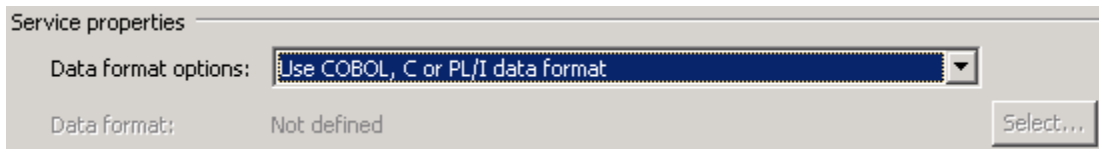
IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

- \_\_\_ 5. Service Configuration Properties:
  - \_\_\_ a. Deploy connector project: ensure that the default option **With module for use by single application** is selected
  - \_\_\_ b. For **Output Directory** enter **\${FF\_OUT}**
  - \_\_\_ c. Click **Advanced >>** to see the hidden advanced properties that can be configured:



You can click each of the configurations and review the options available under it. This lab guides you through some of the important and required configurations.

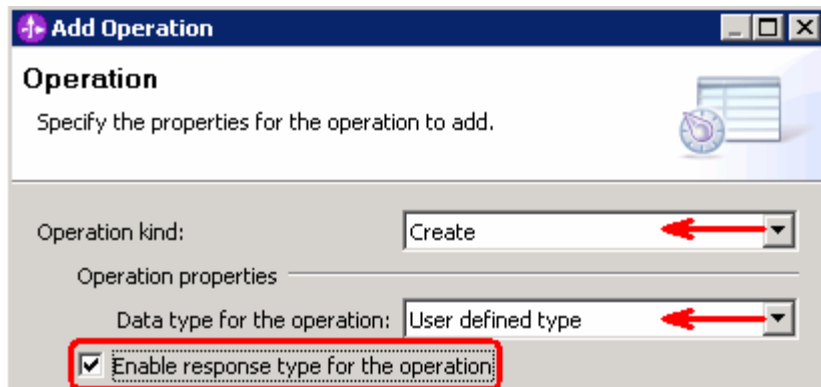
- \_\_\_ 6. For **Data format options**, select **Use COBOL, C or PL/I data format** from the drop down list



- \_\_\_ 7. 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 createCOBOL operation:**

- \_\_\_ 8. From the Operations screen, click **Add...**
  - \_\_\_ a. In the Add Operation window, select **Create** from the drop down menu for Operation kind
  - \_\_\_ b. Select **User Defined Type** for the Data type
  - \_\_\_ c. Select the check box next to **Enable response type for the operation**



- \_\_\_ d. Click **Next**

IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

- \_\_\_ 9. For **Operation name**, enter any name, for Ex: **createCOBOL**
- \_\_\_ 10. For Input type, click **Browse** and select **DFHCOMMAREAWrapperBG** from the Data Type Selection window

Operation name: \* createCOBOL

Specify the operation input

Input type: \* DFHCOMMAREAWrapperBG {http://www.ibm.com/xml...} Browse... New...

Specify the operation output

Output type: CreateResponse {http://www.ibm.com/xmlns/prod/we...} Browse... New...

- \_\_\_ a. Click **Finish**. The defined operation, **createCOBOL**, is populated under Operations list

Operations:

createCOBOL ({http://www.ibm.com/xmlns/prod/websphere;j2ca/flatfile/dfhcommareawrapperb...} Add...

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

Operation name: \* appendCOBOL

Specify the operation input

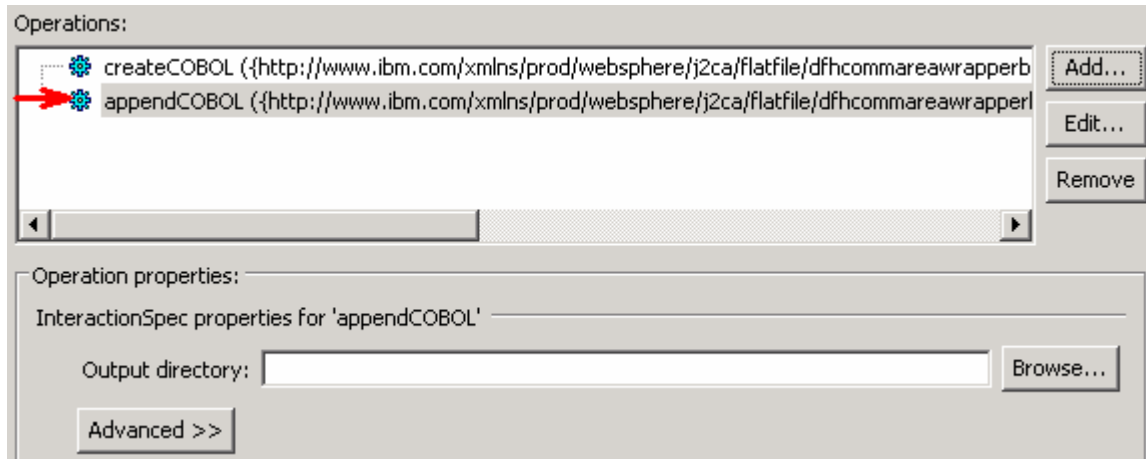
Input type: \* DFHCOMMAREAWrapperBG {http://www.ibm.com/xml...} Browse... New...

Specify the operation output

Output type: AppendResponse {http://www.ibm.com/xmlns/prod/w...} Browse... New...

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

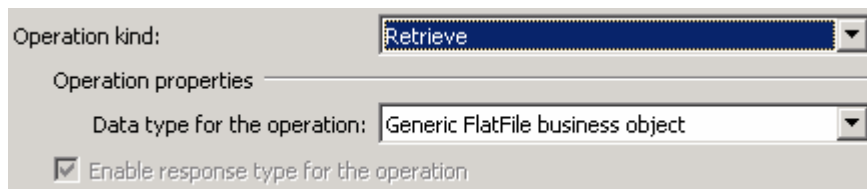
\_\_ g. Click **Finish**. The defined operation, **appendCOBOL**, is populated under Operations list



**Define retrieveCOBOL operation:**

\_\_\_ 12. Click **Add...** to open Add Operation window

- \_\_ a. For **Operation kind**, select **Retrieve** from the drop down list
- \_\_ 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 **FlatFile** (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).

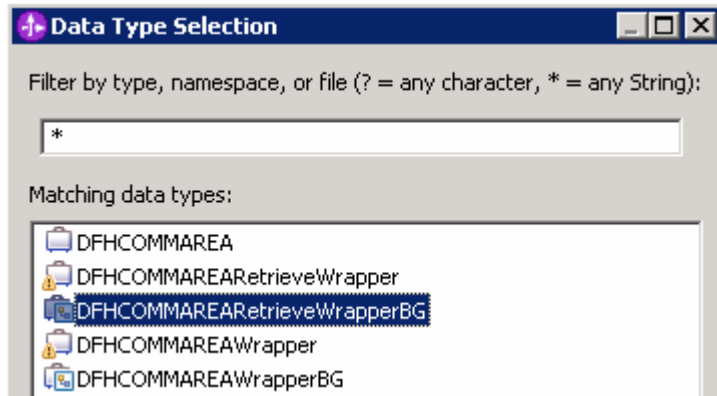
\_\_ e. For Operation name, enter **retrieveCOBOL**

\_\_ f. Define Output type:

- 1) Under **Specify the operation output**, click **Browse...** next to **Output type** to open a Data Type Selection window

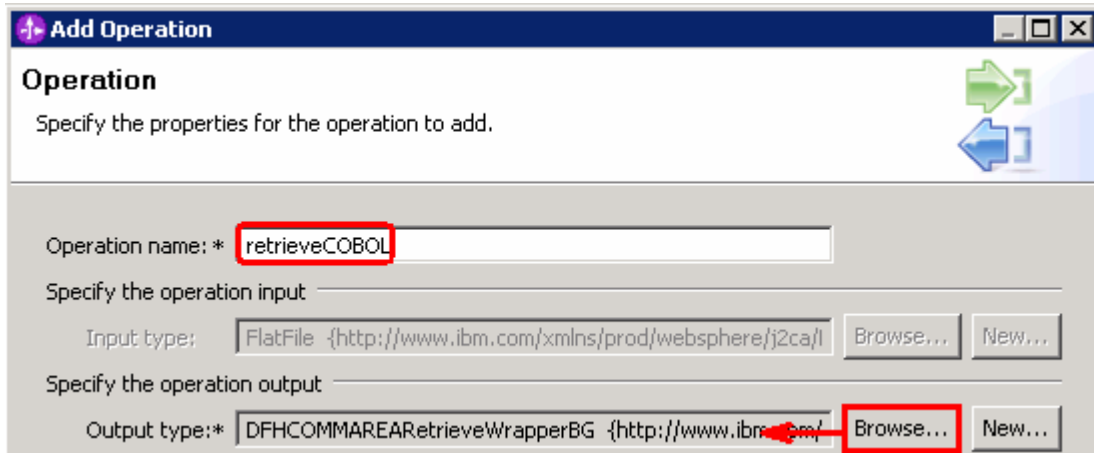
### IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

- 2) From the Data Type Selection window, select **DFHCOMMAREARetrieveWrapperBG** under Matching data types



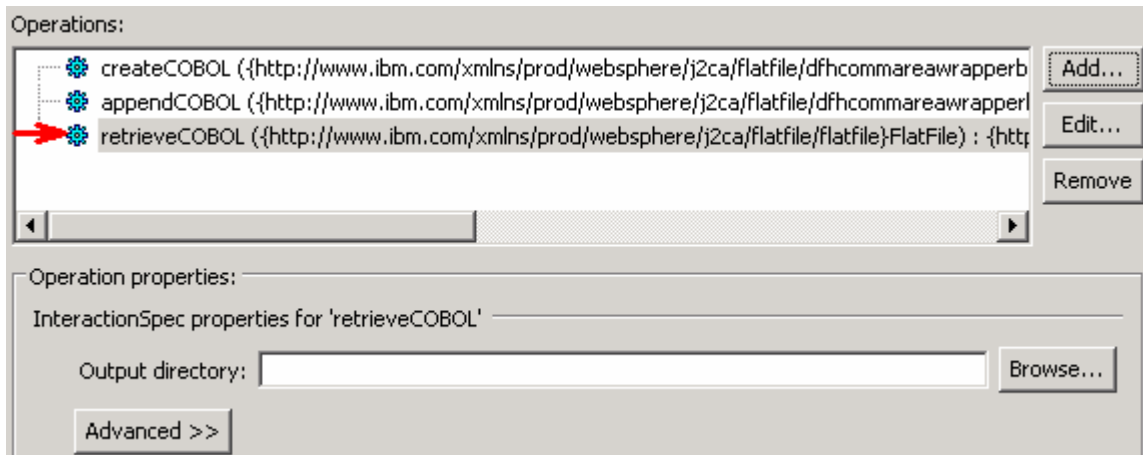
- 3) Click **OK**

\_\_ g. You should now see this:



- \_\_ h. Click **Finish** from the Add Operation window

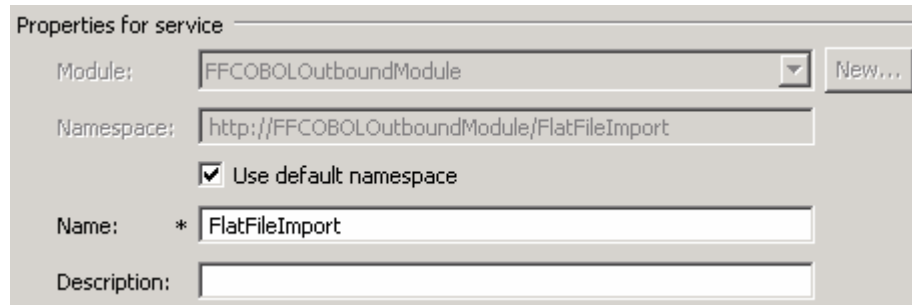
- \_\_ i. You can click **Advanced >>** under 'InteractionSpec properties for retrieveCOBOL' to review the properties available at Interaction spec level



## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

\_\_\_ j. Click **Next** from Operations screen

\_\_\_ 13. From Generate Service screen, accept the default value, **FlatFileImport**, for **Name**



Properties for service

Module: FFCOBOLOutboundModule New...

Namespace: http://FFCOBOLOutboundModule/FlatFileImport

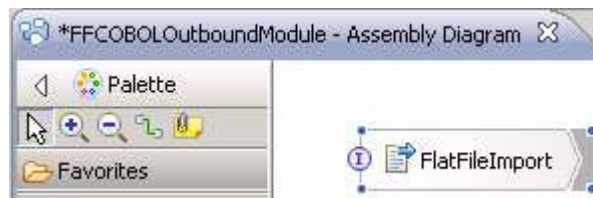
Use default namespace

Name: \* FlatFileImport

Description:

\_\_\_ a. Click **Finish**

\_\_\_ 14. The Assembly diagram for FFCOBOLOutboundModule is opened with an Export component, **FlatFileImport**



\_\_\_ 15. Save (**Ctrl + S**) changes to your assembly diagram

## IBM WEBSHERE ADAPTER 7.0 – 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. Start WebSphere Process Server (if not started already)
  - \_\_\_ a. From the **Servers** view of WebSphere Integration Developer, right click **WebSphere Process Server v7.0** and select Start from the pop-up menu
  - \_\_\_ b. Wait until the server status shows as **Started**
- \_\_\_ 2. Add the project to the WebSphere test environment server
  - \_\_\_ a. Right-click **WebSphere Process Server v7.0** under the Servers view and select **Add and remove projects...** from the pop-up menu
  - \_\_\_ b. From the Add and Remove Projects window, select **FFCOBOLOutboundModuleApp** under Available projects panel and click **Add >**
  - \_\_\_ c. You will now see the **FFCOBOLOutboundModuleApp** 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
- \_\_\_ 3. Open the test client for the module
  - \_\_\_ a. From the Business Integration perspective, right-click the **FFCOBOLOutboundModule** and select **Test > Test Component**
  - \_\_\_ b. The **FFCOBOLOutboundModule\_Test** window is opened in the Assembly editor

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

- createCOBOL
- appendCOBOL
- retrieveCOBOL

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

**Test Create operation:**

- \_\_\_ 4. Under **Detailed Properties**, note that the Operation is **createCOBOL**
  - \_\_\_ a. Fill out the fields for Initial request parameters:
  - \_\_\_ b. For **fileName**, enter any name, for Ex: **COBOLTest.txt**
  - \_\_\_ c. For **includeEndBODElimiter**, enter **###**

► **General Properties**

---

▼ **Detailed Properties**

---

Specify the component, interface, operation, and input parameter values for the Invoke event, then click the Continue icon in the Events area to run the test. [More...](#)

**Configuration:** Default Module Test

**Module:** FFCOBOLOutboundModule

**Component:** FlatFileImport

**Interface:** FlatFileImport

**Operation:** createCOBOL

Initial request parameters:

Value editor  XML editor

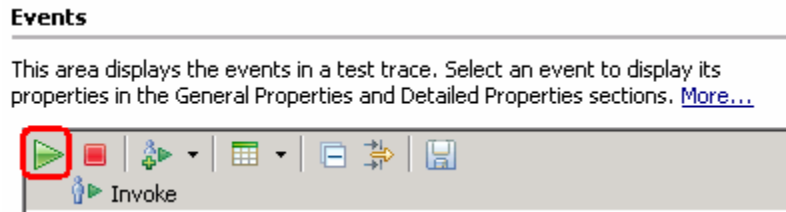
Name	Type	Value
createCOBOLInput	DFHCOMMAREAWrapperBG	[ab]
verb	verb<string>	[ab] CREATE
DFHCOMMAREAWrapper	DFHCOMMAREAWrapper	[ab]
directoryPath	string	[ab]
fileName	string	[ab] COBOLTest.txt
chunkFileName	string	[ab]
fileContentEncoding	string	[ab]
includeEndBODElimite	string	[ab] ###

- \_\_\_ d. Enter the values of your choice for the fields under **Content**

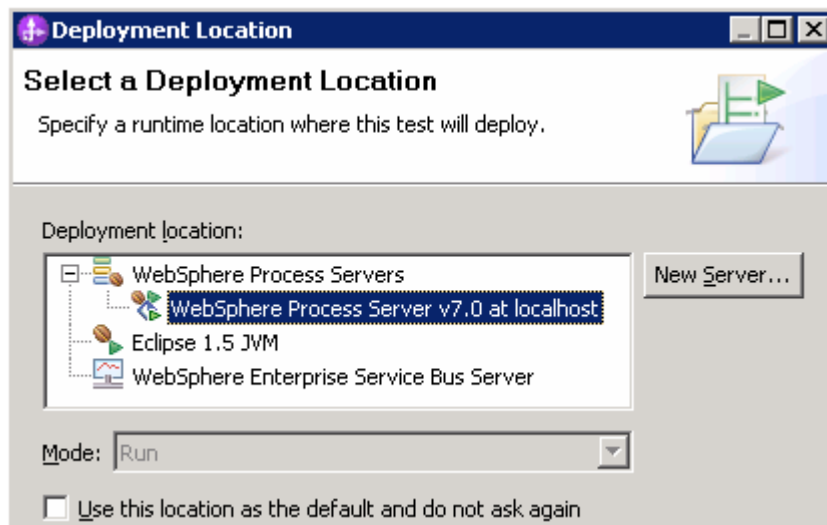
Content	DFHCOMMAREA	[ab]
CustomerNumber *	CustomerNumber<string>	[ab] ABC7
FirstName *	FirstName<string>	[ab] James
LastName *	LastName<string>	[ab] Bond
Street *	Street<string>	[ab] Burnet Rd
City *	City<string>	[ab] Austin
Country *	Country<string>	[ab] USA
Phone *	Phone<string>	[ab] 5128380000
PostalCode *	PostalCode<string>	[ab] 78758

## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

\_\_ e. Click **Continue** button under Events

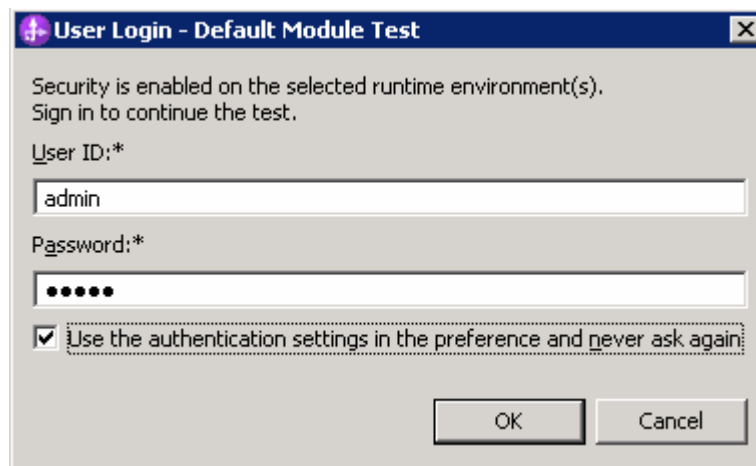


\_\_ f. From Deployment Location window, select **WebSphere Process Servers > WebSphere Process Server v7.0 at localhost** and click **Finish**



\_\_ g. 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





IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

\_\_ h. Verify the results:

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

The screenshot shows the Event Viewer interface. On the left, a tree view shows the event sequence: Invoke (FlatFileImport:createCOBOL), Invoke started, Invoke (FlatFileImport:createCOBOL), Return (FlatFileImport:createCOBOL), and Invoke returned. The 'Return' event is selected. On the right, the 'General Properties' section shows: Module: FFCOBOOutboundModule, Component: FlatFileImport, Interface: FlatFileImport, and Operation: createCOBOL. Below this, the 'Detailed Properties' section shows a table of return parameters:

Name	Type	Value
createCOBOLOutput	CreateResponse	[abi]
filename	string	[abi] COBOLTest.txt

- 2) To verify your test results, check the <OUT\_DIR> directory which should contain a file with the name **COBOLTest.txt**

The screenshot shows a Windows Explorer window with the address bar set to C:\Labfiles70\FlatFileCobol\outdir. The file list shows a folder named FlatFileCobol and a file named COBOLTest.txt with a size of 1 KB and type of Text Document.

- 3) Double-click to open the file, **COBOLTest.txt** and it should contain the text entered under Content field in your test client

The screenshot shows a Notepad window titled 'COBOLTest.txt - Notepad'. The text content is as follows:

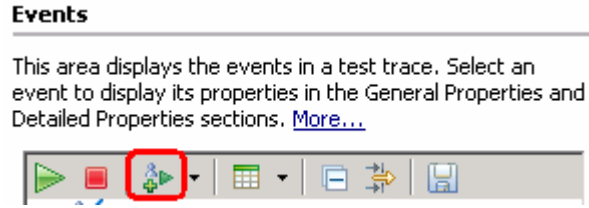
```

ABC7 James          Bond          Burnet Rd          Austin          USA
5128380000         78758   ###
    
```

IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

**Test Append operation:**

- \_\_\_ 5. Add invoke
  - \_\_\_ a. Click the invoke button under Events to add a new invoke



- \_\_\_ 6. Under **Detailed Properties**, note that the Operation is **appendCOBOL**
  - \_\_\_ a. Fill out the fields for Initial request parameters:
  - \_\_\_ b. For **fileName**, enter any name, for Ex: **COBOLTest.txt**
  - \_\_\_ c. For **includeEndBODElimiter**, enter **###**

► **General Properties**

---

▼ **Detailed Properties**

Specify the component, interface, operation, and input parameter values for the Invoke event, then click the Continue icon in the Events area to run the test. [More...](#)

**Configuration:** Default Module Test

**Module:** FFCOBOLOutboundModule

**Component:** FlatFileImport

**Interface:** FlatFileImport

**Operation:** **appendCOBOL**

Initial request parameters:

- Value editor
- XML editor

Name	Type	Value
appendCOBOLInput	DFHCOMMAREAWrap...	[ab]
verb	verb<string>	[ab] CREATE
DFHCOMMAREAWrap	DFHCOMMAREAWrap...	[ab]
directoryPath	string	[ab]
fileName	string	[ab] <b>COBOLTest.txt</b>
chunkFileName	string	[ab]
fileContentEncodi	string	[ab]
includeEndBODElir	string	[ab] <b>###</b>

- \_\_\_ d. Enter the values of your choice for the fields under **Content**

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

Content	DFHCOMMAREA	[ab]
CustomerNumber *	CustomerNumber <string>	[ab] XYZ9
FirstName *	FirstName <string>	[ab] James
LastName *	LastName <string>	[ab] Doe
Street *	Street <string>	[ab] Duval Rd
City *	City <string>	[ab] Austin
Country *	Country <string>	[ab] USA
Phone *	Phone <string>	[ab] 5128380000
PostalCode *	PostalCode <string>	[ab] 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**

This area displays the events in a test trace. Select an event to display its properties in the General Properties and Detailed Properties sections. [More...](#)

**General Properties**

**Detailed Properties**

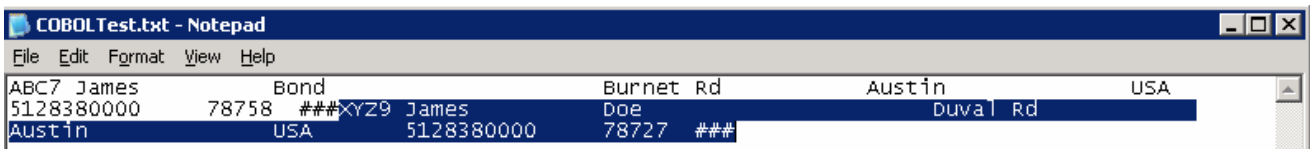
Module: [FFCOBOLOutboundModule](#)  
 Component: [FlatFileImport](#)  
 Interface: [FlatFileImport](#)  
 Operation: [appendCOBOL](#)

Return parameters:

Value Editor | XML Source

Name	Type	Value
appendCOBOLOutput	AppendResponse	[ab]
filename	string	[ab] COBOLTest.tx...

- 2) To verify your test results, open the COBOLTest.txt file from <OUT\_DIR> directory 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.



**Test Retrieve operation:**

7. Click the invoke button under Events to add another invoke
8. Under **Detailed Properties**, note that the Operation is **retrieveCOBOL**
  - a. Fill out the fields for Initial request parameters:
  - b. For **filename**, enter **COBOLTest.txt**
  - c. **splitFunctionClassName**: **com.ibm.j2ca.utils.filesplit.SplitByDelimiter**
  - d. **splitCriteria**: **###**

IBM WEBSPPHERE ADAPTER 7.0 – LAB EXERCISE

Name	Type	Value
retrieveCOBOLInput	FlatFile	[ab]
directoryPath	string	[ab]
fileName	string	[ab] COBOLTest.txt
chunkFileName	string	[ab]
fileContentEncoding	string	[ab]
includeEndBODelimiter	string	[ab]
stagingDirectory	string	[ab]
chunkNumber	string	[ab]
generateUniqueFile	boolean	[ab] false
createFileIfNotExists	boolean	[ab] false
splitFunctionClassName	string	[ab] com.ibm.j2ca.utils.filesplit.SplitByDelimiter
splitCriteria	string	[ab] ###
deleteOnRetrieve	boolean	[ab] false
archiveDirectoryForDeleteOnRetrieve	string	[ab]
Content	UnstructuredContent	[ab]
ContentType	string	[ab]
ObjectName	string	[ab]
AsText	string	[ab]
AsBinary	hexBinary	[ab] 00

\_\_ e. Click **Continue** button under Events

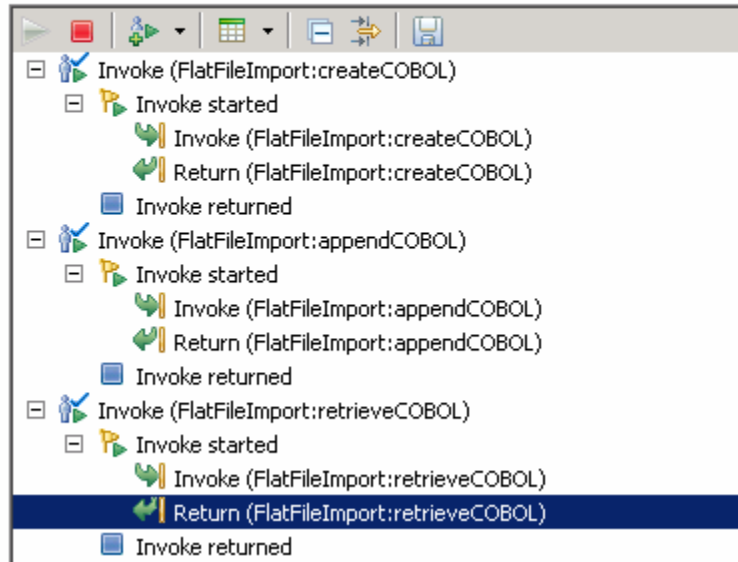
## IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

\_\_\_ f. Verify the results:

- 1) There is a new return under retrieveCOBOL invoke

### Events

This area displays the events in a test trace. Select an event to display its properties in the General Properties and Detailed Properties sections. [More...](#)



- 2) Select the Return and notice the return parameters , which contains the data you just entered in the previous steps:

## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

## Return parameters

Name	Type	Value
retrieveCOBOLOutput	DFHCOMMAREARetrieveWrapperBG	[ab]
verb	verb <string>	[ab]
DFHCOMMAREARetrieveWrapper *	DFHCOMMAREARetrieveWrapper	[ab]
Content	DFHCOMMAREA[]	[ab]
Content[0] ←	DFHCOMMAREA	[ab]
CustomerNumber *	CustomerNumber <string>	[ab] ABC7
FirstName *	FirstName <string>	[ab] James
LastName *	LastName <string>	[ab] Bond
Street *	Street <string>	[ab] Burnet Rd
City *	City <string>	[ab] Austin
Country *	Country <string>	[ab] USA
Phone *	Phone <string>	[ab] 5128380000
PostalCode *	PostalCode <string>	[ab] 78758
Content[1] ←	DFHCOMMAREA	[ab]
CustomerNumber *	CustomerNumber <string>	[ab] XYZ9
FirstName *	FirstName <string>	[ab] James
LastName *	LastName <string>	[ab] Doe
Street *	Street <string>	[ab] Duvan Rd
City *	City <string>	[ab] Austin
Country *	Country <string>	[ab] USA
Phone *	Phone <string>	[ab] 5128380000
PostalCode *	PostalCode <string>	[ab] 78727

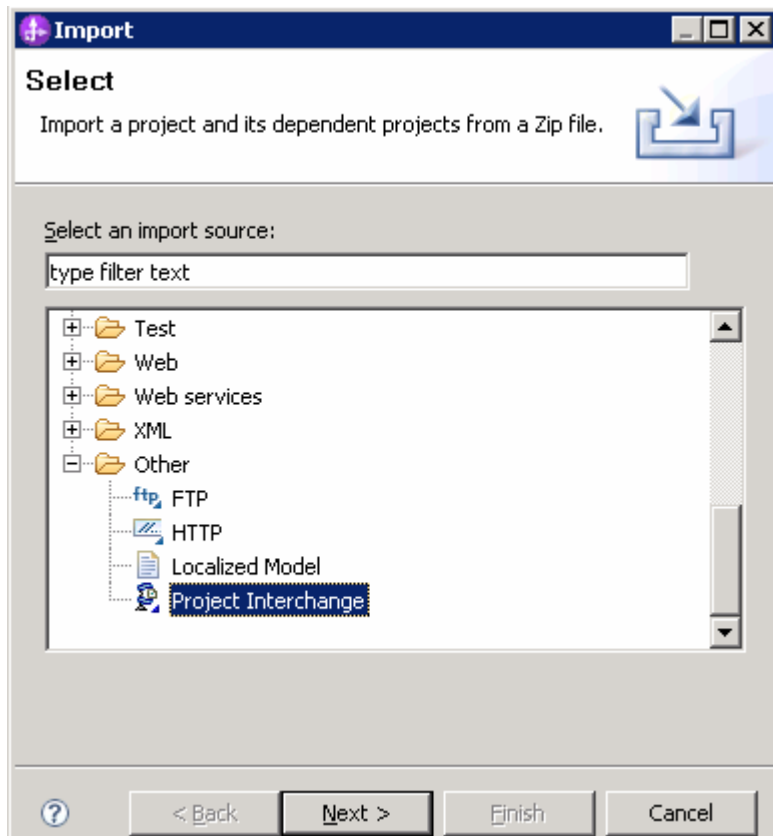
## 9. Restore the Sever Configuration

- \_\_\_ a. Close the **FFCOBOLOutboundModule\_Test** window and click **No** for the Save Resources window
- \_\_\_ b. Right-click **WebSphere Process Server v7.0** under the Servers view and select **Add and remove projects...** from the pop-up menu
- \_\_\_ c. Select **FFCOBOLOutboundModuleApp** under Configured projects and click **< Remove**
- \_\_\_ d. Click **Finish** after you see the application moved to Available projects. Wait until the application is being unpublished

## IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

### Solution instructions

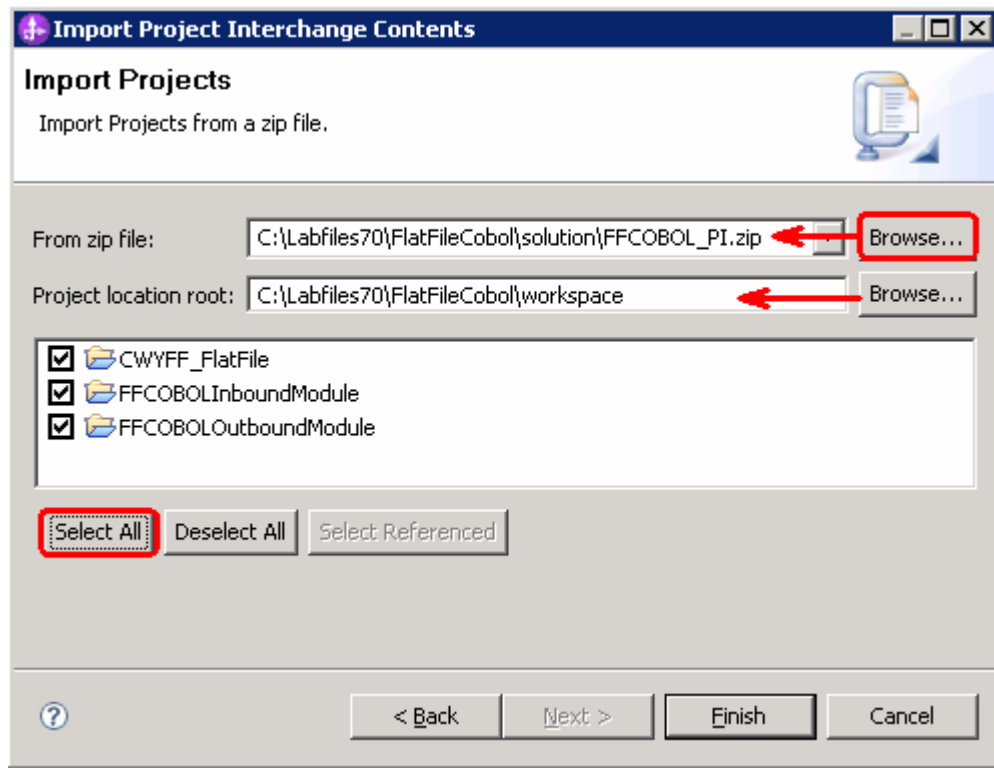
- \_\_\_ 1. Start WebSphere Integration Developer V7.0 with a new workspace
  - \_\_\_ a. Follow the instructions outlined in **Part 1** of this exercise
- \_\_\_ 2. Import the Project Interchange
  - \_\_\_ a. Import the project interchange file **FFCOBOL\_PI.zip** from **<LAB\_FILES>\FlatFileCobol\solution** directory
  - \_\_\_ b. Select **File → Import** from the menu
  - \_\_\_ c. Select **Other → Project Interchange** in the **Import** dialog and click **Next**



- \_\_\_ d. For the **From zip file**, click on the **Browse** button and select the **FFCOBOL\_PI.zip** in the **<LAB\_FILES>\ FlatFileCobol\solution** directory
- \_\_\_ e. Enter **<LAB\_FILES>\FlatFileCobol\workspace** for the **Project location root**

## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

- \_\_\_ f. Click the **Select All** button. This will select all the modules: **CWYFF\_FlatFile**, **FFCOBOLInboundModule**, and **FFCOBOLOutboundModule**



- \_\_\_ g. Click **Finish**

- \_\_\_ 3. Continue with **Part 2.4** of this lab to test the inbound scenario
- \_\_\_ 4. Continue with **Part 3.3** of this lab to test the outbound scenario



## IBM WEBSPHERE ADAPTER 7.0 – LAB EXERCISE

### **What you did in this exercise**

In this lab, you started with importing the Flat File 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.

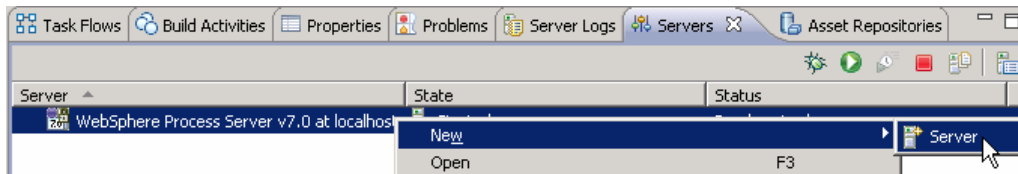
At the end of each part, you deployed and tested the adapter application for operations defined during the external service wizard.

## IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

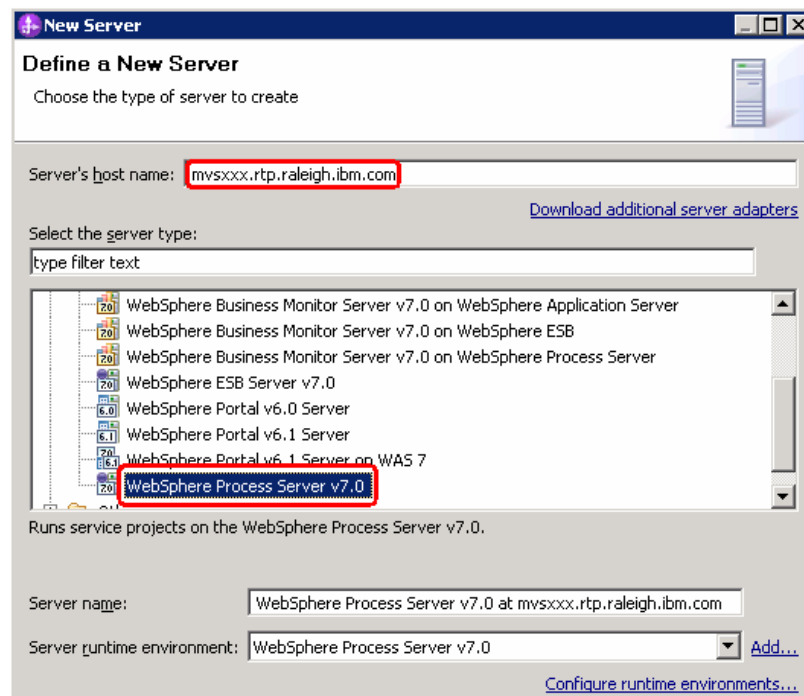
## Task: Adding remote server to WebSphere Integration Developer test environment

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**.



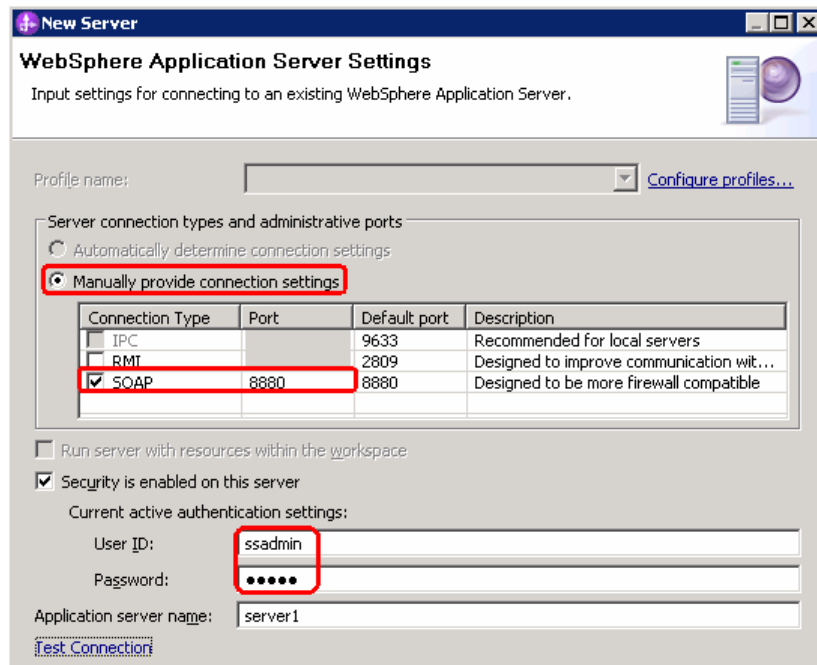
- \_\_\_ c. In the New Server dialog, specify the remote server's host name, **<HOSTNAME>**.
- \_\_\_ d. Ensure that the appropriate server type, **'WebSphere Process Server v7.0'** or **'WebSphere ESB Server v7.0'**, is highlighted in the server type list



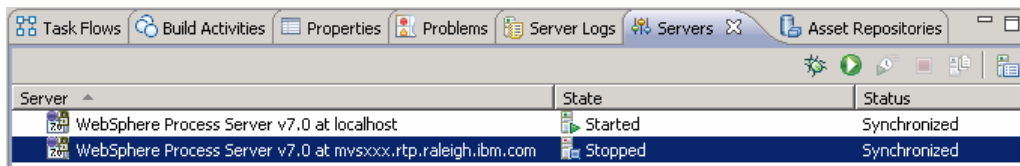
- \_\_\_ e. Click **Next**

IBM WEBSHERE ADAPTER 7.0 – 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 WEBSHERE ADAPTER 7.0 – 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
```

IBM WEBSHERE ADAPTER 7.0 – LAB EXERCISE

This page is left intentionally blank.