IBM WEBSPHERE PROCESS INTEGRATION 6.0 – LAB EXERCISE

Adapter for Flat Files Inbound Lab

What this exercise is about

The objective of this lab is to provide you with an understanding of the WebSphere Adapter for Flat Files and inbound event processing.

Lab Requirements

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

- WebSphere Integration Developer V6 installed

- WebSphere Process Server V6 test environment installed

- WebSphere Adapter for Flat Files V6 installed

- Sample code in the directory C:\Labfiles60\FlatFiles (Windows) or /tmp/LabFiles60/FlatFiles (Linux)

What you should be able to do

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

- Install and deploy the Adapter for Flat Files and integrate it into an SCA application for inbound event processing.

Introduction

In this lab, you will install and deploy the WebSphere Adapter for Flat Files and create an SCA application that polls for and processes inbound events from the file system.

Exercise Instructions

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

Reference Variable	Windows Location	AIX/UNIX Location
<LAB_NAME>	FFInBound	
<WID_HOME>	C:\Program Files\IBM\WebSphere\ID\6.0	
<WPS_HOME>	C:\<WID_HOME>\runtimes\bi_v6	
<FFADAPTER_HOME>	C:\Program Files\IBM\ResourceAdapters\FlatFiles	
<LAB_FILES>	C:\Labfiles60	/tmp/Labfiles60
<WORKSPACE>	C:\Labfiles60\FlatFiles\workspace	
<SOLUTION>	C:\Labfiles60\FlatFiles\Solution	
<TEMP>	C:\temp	/tmp

Windows users please 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, C:\LabFiles60\ would be replaced by C:/LabFiles60/

Part 1: Prepare Database and Subdirectory Structure

This part describes the steps for creating the FFTEST database in Cloudscape which will contain the Event Distribution Table, EDTTABLE. The EDTTABLE will be created automatically for you during the install and deployment of the application containing the Flat Files adapter to the WebSphere Process Server. Inbound processing also requires that an event directory exists and optionally, an archive directory. By unzipping the Labfiles60Adapter.zip, these subdirectories, c:\Labfiles60\FlatFiles\eventdir and c:\Labfiles60\FlatFiles\archivedir, have already been created for you

_____ 1. Start Cloudscape Cview Graphical User Interface (GUI) by running the cview.bat.

    __ a. Open a command prompt window, navigate to the following subdirectory, and run the cview.bat program.

```
<WPS_HOME>\cloudscape\bin\embedded>cview
```

_____ 2. Using the CView GUI, create the FFTEST Database Select **File -> New -> Database** c:\<WPS_HOME>\cloudscape\databases\FFTEST. Click **OK**
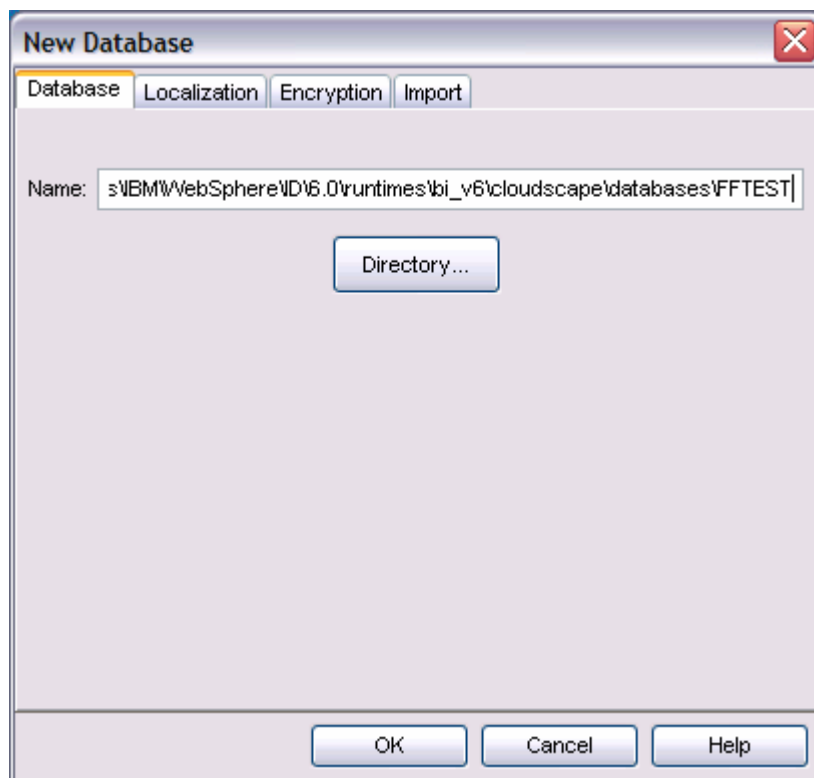
_____ 3. Close the Database and exit Cview GUI.

_____ 4. Check to ensure the event and archive subdirectories already exist.

Note: By unzipping the Labfiles60.zip, these subdirectories, c:\Labfiles60\FlatFiles\eventdir and c:\Labfiles60\FlatFiles\archivedir, have already been created for you.


    __ a. Open a command prompt window

__ b. Check for the following subdirectories. If they do not exist, create them by typing the commands:

```
C:\> MD c:\Labfiles60\FlatFiles\eventdir  (enter)
C:\> MD c:\Labfiles60\FlatFiles\archivedir (enter)
```

Part 2: Initialize the Workspace for this Lab Exercise

In this part, you will start WebSphere Integration Developer and create a separate workspace in which to work.

____ 1.   Start WebSphere Integration Developer v6 with a new workspace

   __ a. From the start menu select **Start > Programs > IBM WebSphere > Integration Developer V6.0 > WebSphere Integration Developer V6.0 Beta**

   __ b. When prompted enter **<LAB_FILES>\FlatFiles\ffinbound\workspace** for your workspace and click **OK**
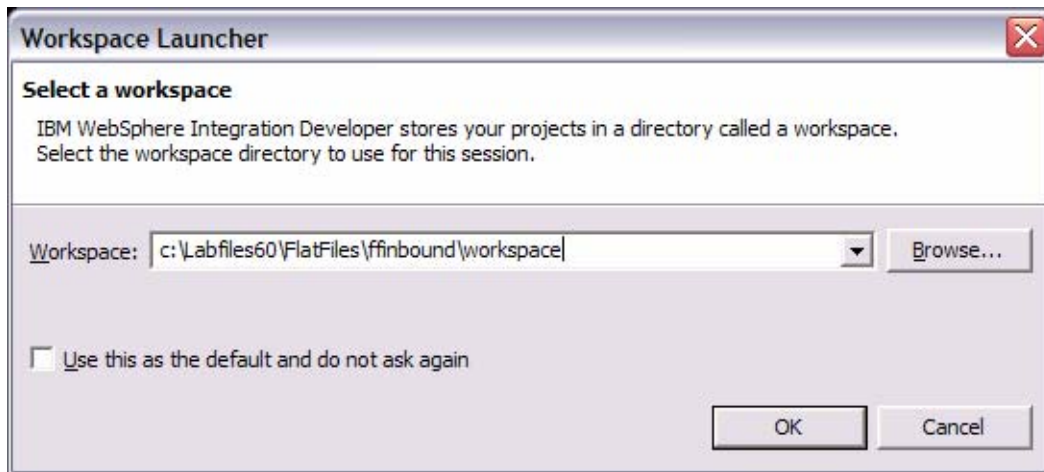
   __ c. When WebSphere Integration Developer v6.0 opens, close the **Welcome page**

WPSWIDv6_FFInboundLab
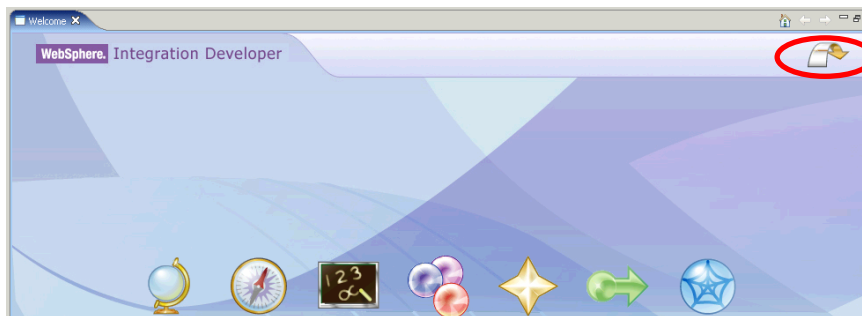
Part 3: Create the FFInbound Application

In this part you will run Enterprise Service Discovery to import the WebSphere Adapter for Flat Files, input the property values to build the Service Description (used to create the Activation Specification), create the necessary SCA artifacts, and assemble them into an SCA application. Since the Flat Files Adapter has a single, pre-defined business object structure, there is no Enterprise Service Discovery for "object discovery" of business objects.
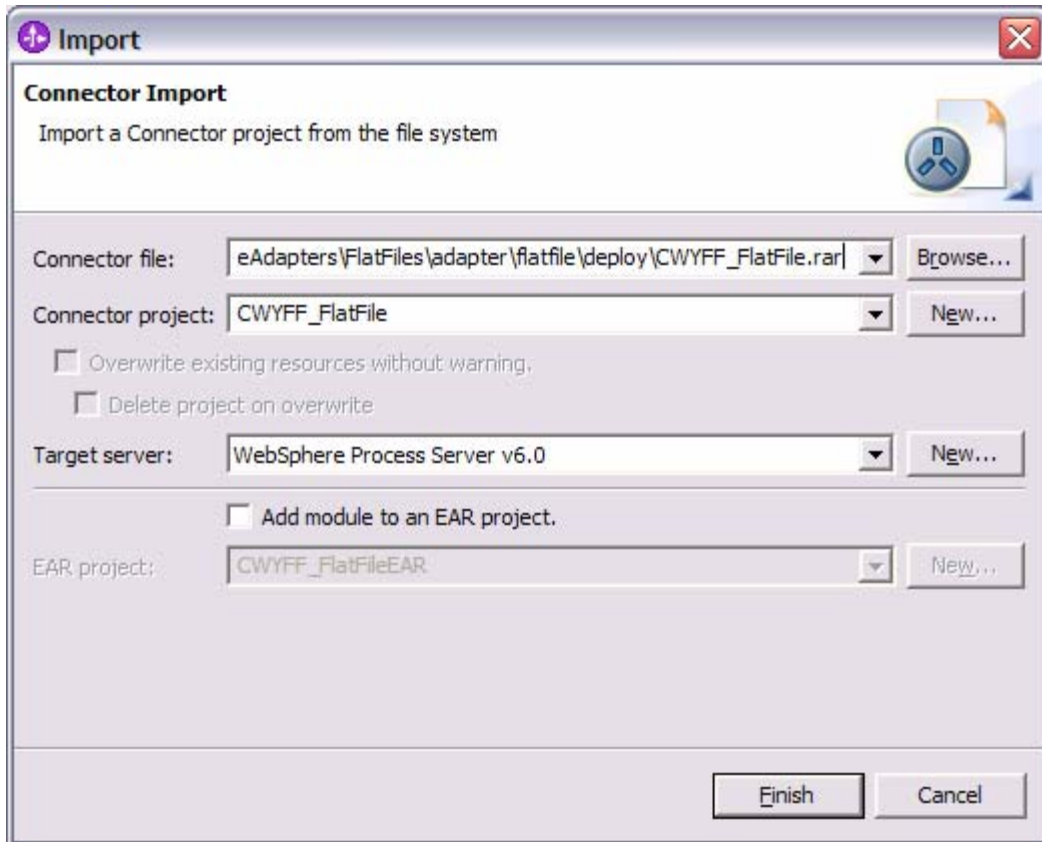
____ 1.   Start the Enterprise Service Discovery process and import the Flat Files Adapter Resource Archive (RAR) file. This will create a J2EE Connector Project.

__ a. From the top Menu bar, select **File > New > Enterprise Service Discovery**

__ b. From the Select an Enterprise Service Resource Adapter panel, at the bottom right, click **Import Resource Adapter…**
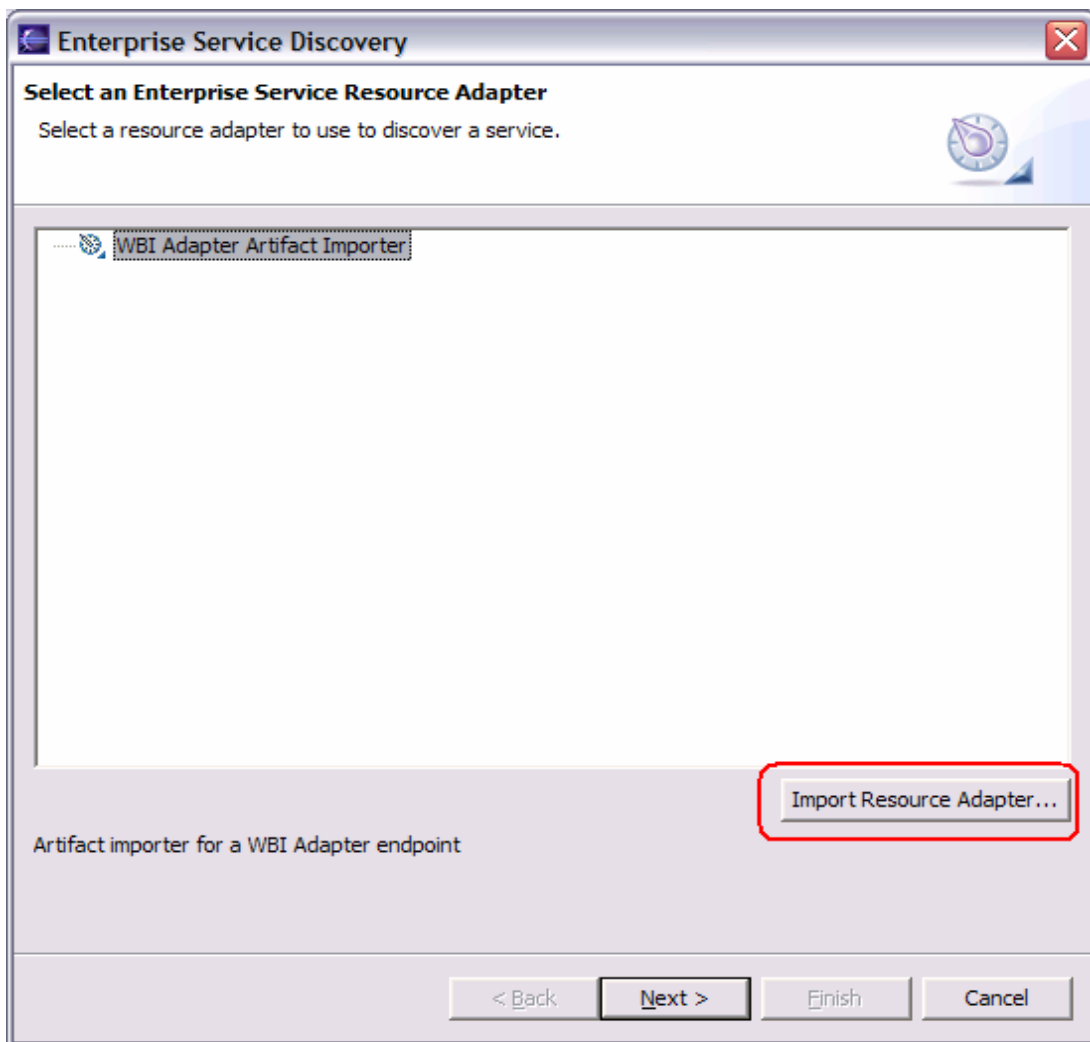
__ c. Complete the Connector Import panel

1) Connector file: **Browse…** to the location of the CWYFF_FlatFile.rar adapter file and select it, for example: C:\<FFADAPTER_HOME>\adapter\flatfile\deploy\CWYFF_FlatFile.rar (or wherever you previously installed the adapter using the WebSphere Adapter for Flat Files installation program).

2) Deselect the "**Add module to an EAR project**" if it is selected, and click **Finish**
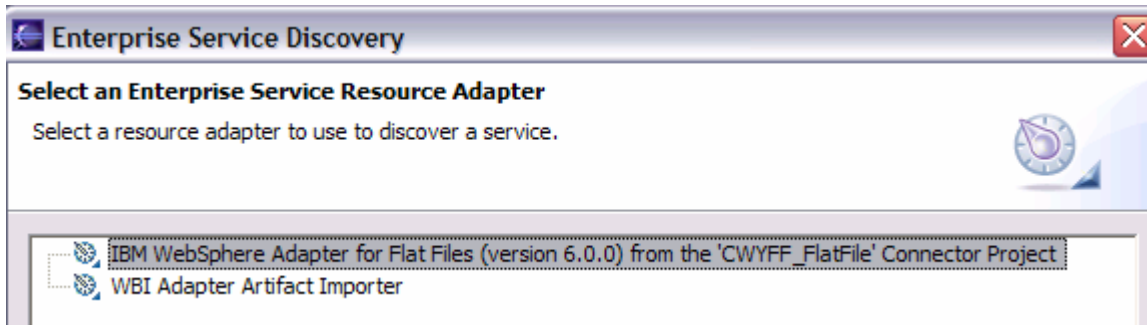

__ d. At the Confirm Perspective Switch prompt, select **No**
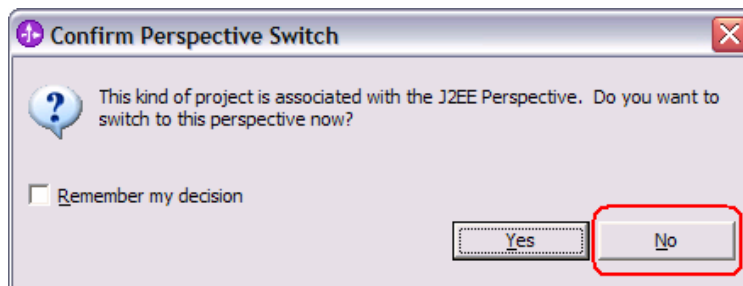
____ 2. You will return to the Select an Enterprise Service Resource Adapter panel. Continue through the remaining Enterprise Service Discovery panels.
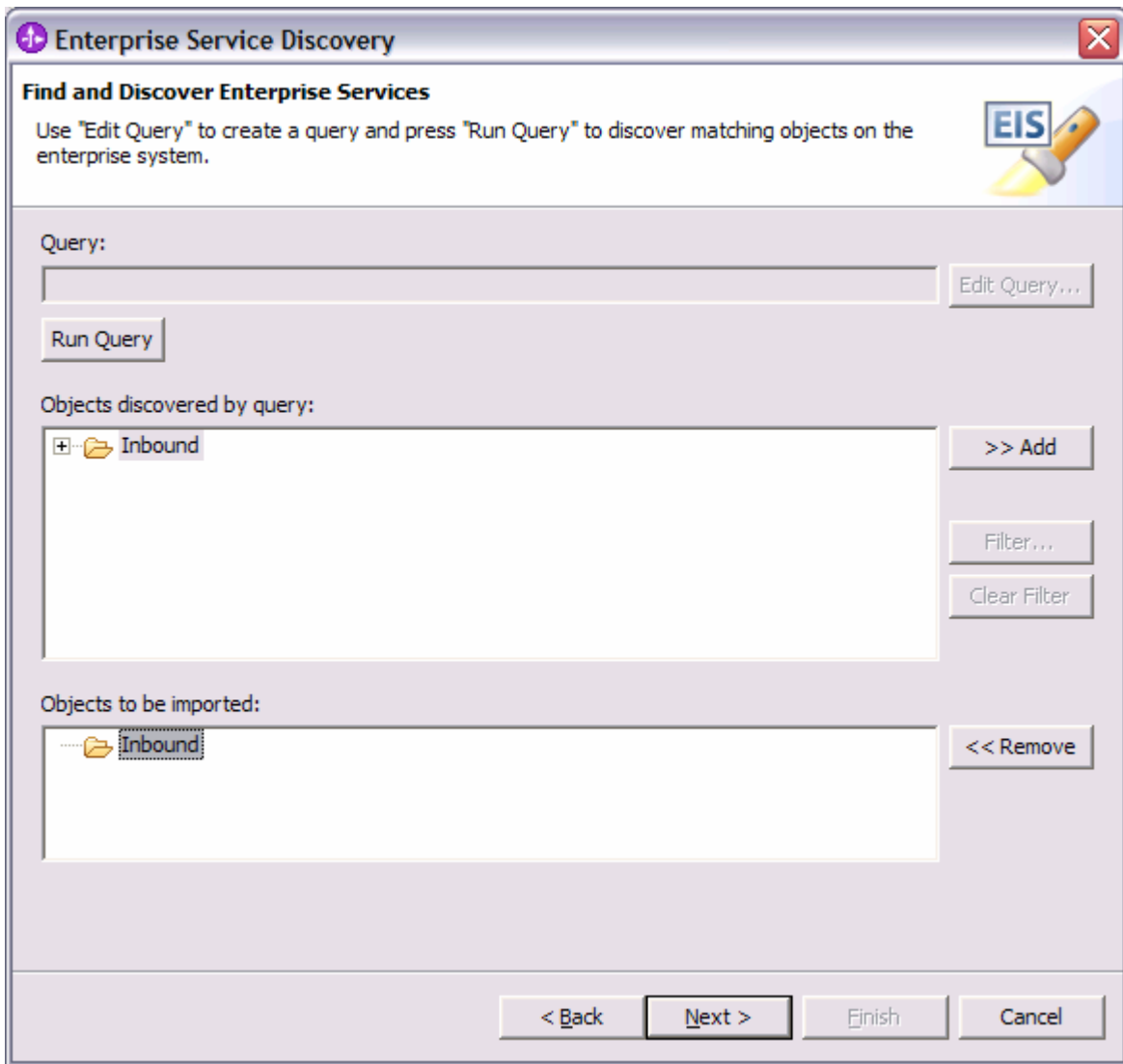
   __ a. Highlight the **IBM WebSphere Adapter for Flat Files** and click **Next.**

    __ b. From the Configure Settings for Discovery Agent panel, change the **ServiceType:** from Outbound to **Inbound**. Click the **Show Advanced >>** button to note the Log file output location and Logging Level options for discovery log, leave the default values, and click **Next**
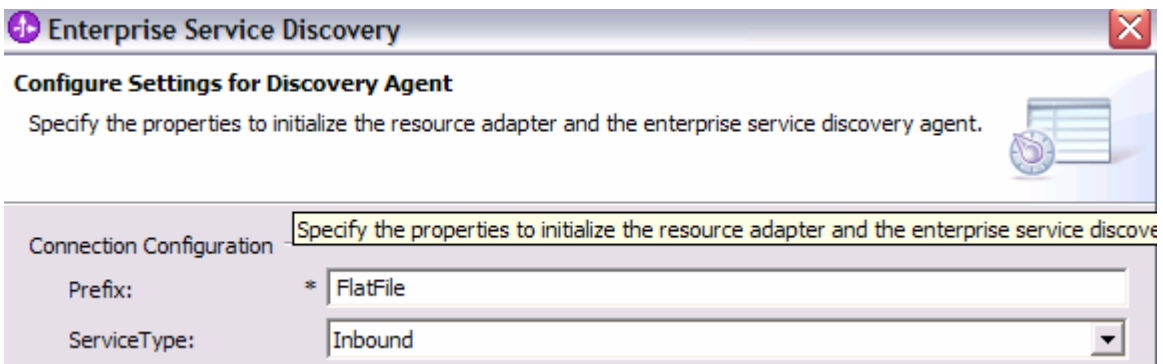
__ c. From the Find and Discover Enterprise Services panel, click the **Run Query** button. An **Inbound** folder will appear in the Objects discovered by query: window. **Select** the **Inbound Folder, click** the **>> Add** button. The **Inbound** folder is now displayed in the Objects to be imported: window. Click **Next**

Enterprise Service Discovery

Find and Discover Enterprise Services

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

EIS

Query:

[   ] Edit Query...

Run Query

Objects discovered by query:

⊞ 📁 Inbound

>> Add

Filter...

Clear Filter

Objects to be imported:

📁 Inbound

<< Remove

< Back   Next >   Finish   Cancel

WPSWIDv6_FFInboundLab

____ d. On the Configure Objects panel, enter **com/test/flat** for **BO Location:** Notice the operation available for inbound is **READ.** Click **Next.**
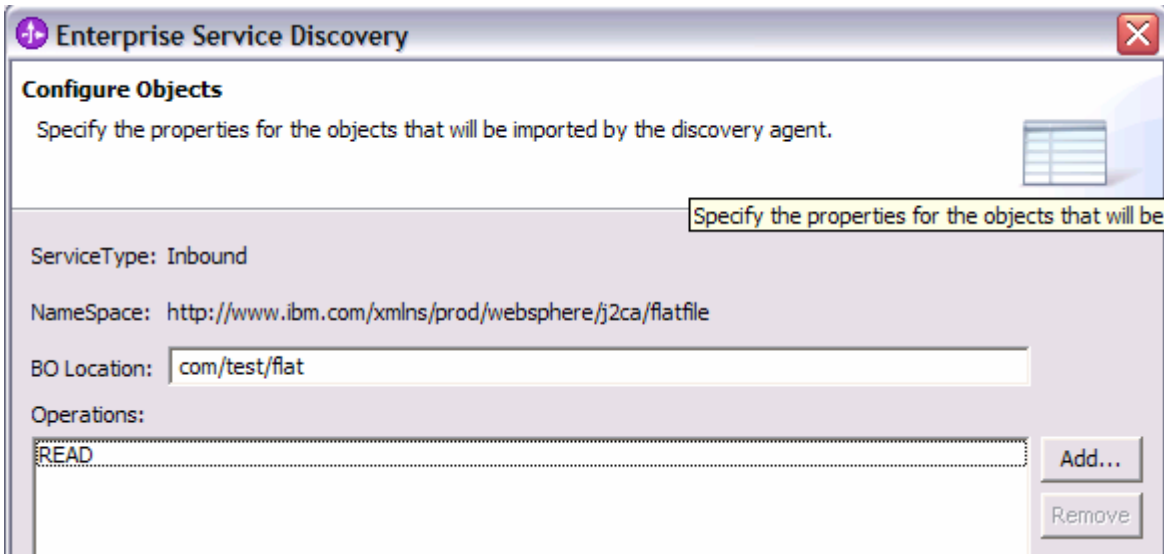
____ e. On the Generate Artifacts panel, create a new Module. Click the **New…** button to the right, enter a Module name of **FFTestInbound**, Click **Finish.**
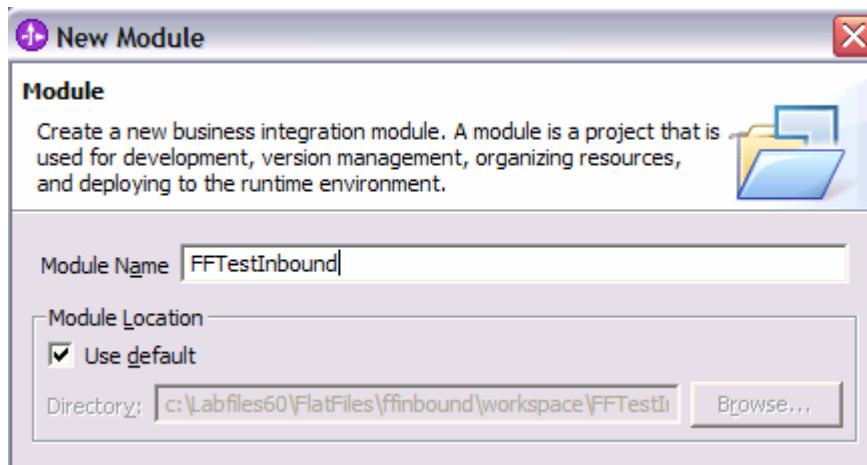
____ f. Enter **Folder:** value of **com/test/flat.**

____ g. **Select** the radio button to the left of **Use discovered connection properties**. Additional Connection properties and Resource Adapter properties will become visible. Leave the defaults except as follows, and click **Finish** (type the full path to the database FFTEST, remember to include the space between the words Program and Files).

EDTDatabaseName: c:\Program Files\IBM\WebSphere\ID\6.0\runtimes\bi_v6\cloudscape\databases\FFTEST
EDTDriverName: com.ibm.db2j.jdbc.DB2jXADataSource
**EDTTableName: EDTTABLE**
EDTUserName: enter a valid username
**EDTPassword: enter a valid password**
Leave the **AutoCreateEDT** box checked and a table with the name specified in **EDTTableName** will be created.

EventDirectory: c:\Labfiles60\FlatFiles\eventdir
ArchiveDirectory: c:\Labfiles60\FlatFiles\archivedir

_____ 3. You will return to the Business Integration perspective. Use the Assembly Diagram to wire the **FlatFileInboundInterface** to a Java Component.

__ a. From the **Business Integration** perspective, expand the **FFTestInBound** folder.

__ b. Double click the **FFTestIntbound** module. This will open the module in the Assembly Diagram. You will see a message that there is one new element added to the module.
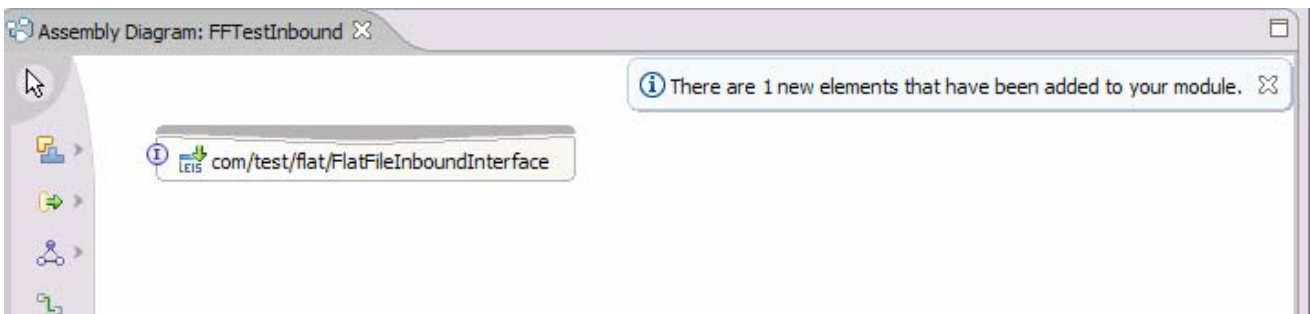
__ c. From the palette, select the **Component (with no implementation type)** icon, then select the **Java Component** and click in the Assembly Diagram drop it on the Assembly Diagram.
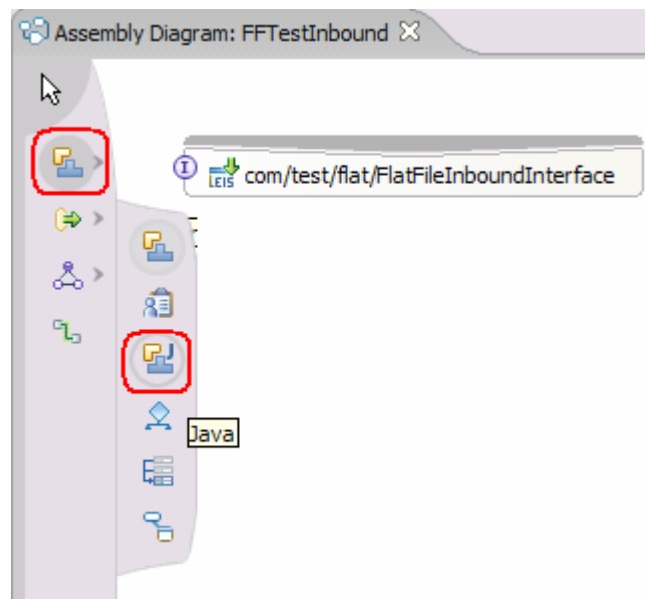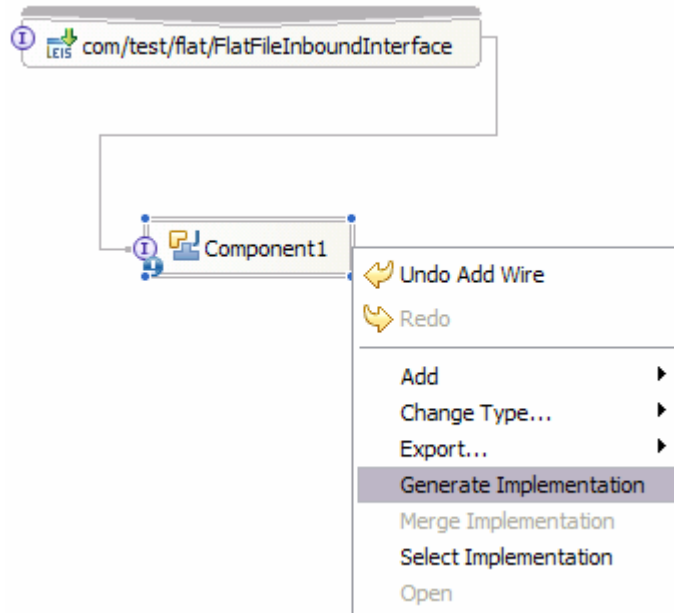
__ d. Wire the **com/test/flat/FlatFileInboundInterface** to the **Java Component.** At the Add Wire popup window, select **OK** in response:

_____ 4.   Generate the implementation for the Java Component

__ a. Right click the **Java Component1**, select **Generate Implementation** from the context menu

__ b. Select the package of **com.test.flat** Select **OK**

__ c. The Java Editor will open with the **Component1Impl.java** file. Scroll down and locate the **READ (DataObject rEADInput)** method that needs to be implemented. Paste the following code into the method so the complete method looks as follows

NOTE: For your convenience, the following code snippet of System.out.println statements, can be found in **<LAB_FILES>\FlatFiles\snippets\javacomp1.txt.**

```
public void READ(DataObject rEADInput) {
System.out.println("readInput is:"+rEADInput);
System.out.println("BO is:  "+rEADInput.getDataObject("FlatFile"));
System.out.println("inbytes is:  "+rEADInput.getDataObject("FlatFile").get("inputBytes"));
System.out.println("outbytes is:  "+rEADInput.getDataObject("FlatFile").get("outputBytes"));
System.out.println("outstr is:  "+rEADInput.getDataObject("FlatFile").get("outputString"));
System.out.println("dir is:  "+rEADInput.getDataObject("FlatFile").get("directoryPath"));
System.out.println("fn is:  "+rEADInput.getDataObject("FlatFile").get("fileName"));
DataObject bo=rEADInput.getDataObject("FlatFile");
```

```
        byte []  bar=bo.getBytes("inputBytes");
        System.out.println("array size is:  "+bar.length);
        }
```

___ d. Following is a screen print of the same complete method for easier reading.

```
public void READ(DataObject rEADInput) {
    System.out.println("readInput is:"+rEADInput);
    System.out.println("BO is:   "+rEADInput.getDataObject("FlatFile"));
    System.out.println("inbytes is:   "+rEADInput.getDataObject("FlatFile").get("inputBytes"));
    System.out.println("outbytes is:   "+rEADInput.getDataObject("FlatFile").get("outputBytes"));
    System.out.println("outstr is:   "+rEADInput.getDataObject("FlatFile").get("outputString"));
    System.out.println("dir is:   "+rEADInput.getDataObject("FlatFile").get("directoryPath"));
    System.out.println("fn is:   "+rEADInput.getDataObject("FlatFile").get("fileName"));
    DataObject bo=rEADInput.getDataObject("FlatFile");
    byte []  bar=bo.getBytes("inputBytes");
    System.out.println("array size is:  "+bar.length);

}
```

___ e. Save your work by selecting **File -> Save** from the top menu, or using the shortcut key sequence **Ctrl + S.**   Close the file.

___ f. Close the Assembly Diagram and save any changes.

Part 4: Test the Application using the WebSphere Test Environment (WTE)

In this part you will use the WebSphere Test Environment to test the SCA application with Flat Files Adapter event processing by simply starting the application and verifying that it polls for and picks up the ffinboundtest.txt file event.

____ 1.   Add the project to the server for the WebSphere Test Environment.

__ a. Right click on the server in the server view and select **Add and remove projects** …
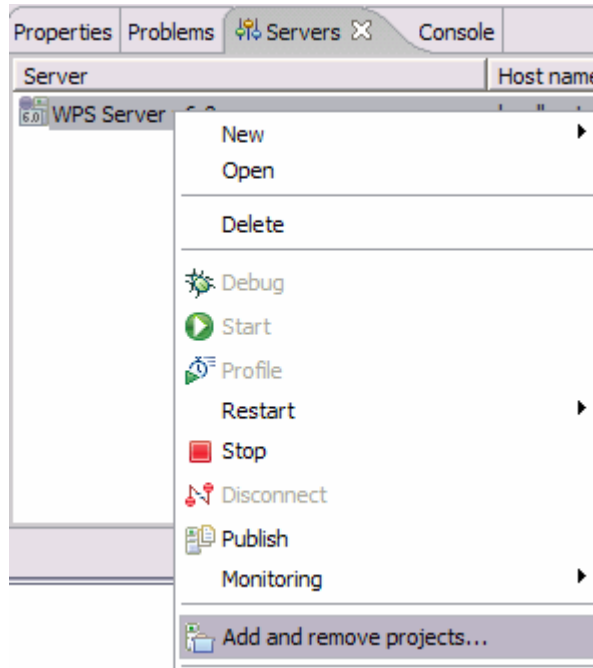
__ b. In the Add and Remove Projects dialog, select the **FFTestInboundApp** project from the Available projects panel. Click **Add >** to add it to the Configured projects panel. Click **Finish**
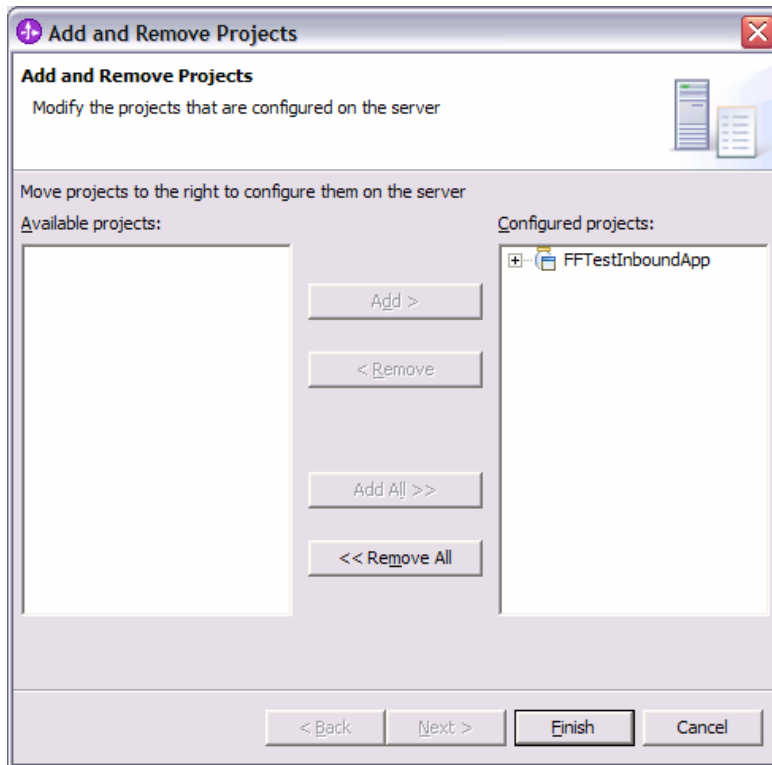
__ c. Wait for the project to be added. In the Console view, you will see messages that the application has successfully started.

____ 2. Copy any test file to the c:\Labfiles60\FlatFiles\eventdir> subdirectory. As the file is copied to the eventdir subdirectory, it will be processed by the adapter and moved to the c:\Labfiles60\FlatFiles\archivedir>.

NOTE: For your convenience, a test file named ffinboundtest.txt file can be found in **<LAB_FILES>\FlatFiles\snippets\ffinboundtest.txt**

__ a. Copy the test file and check the c:\Labfiles60\FlatFiles\eventdir – the subdirectory should be empty as the file has been processed as an event. Check the c:\Labfiles60\FlatFiles\archivedir – the subdirectory should contain an archive of the event file, with the same filename appended with the year, month, date, system time, and PROCESSED. For example:

Directory of C:\Labfiles60\FlatFiles\eventdir (Note that it's empty)
07/21/2005 01:01 PM   <DIR>   .
07/21/2005 01:01 PM   <DIR>   ..
            0 File(s)   0 bytes
            2 Dir(s) 14,498,340,864 bytes free

Directory of C:\Labfiles60\FlatFiles\archivedir (Note that the event file ffinboundtest.txt is now archived here)
07/21/2005 01:01 PM   <DIR>   .
07/21/2005 01:01 PM   <DIR>   ..
07/21/2005 12:43 PM   52 ffinboundtest.txt.2005_07_21_13_01_44_371.PROCESSED

__ b. Check the c:\wid\prof\logs\server1\systemOut.log for the print statements similar to:

```
[7/25/05 19:34:18:321 AKDT] 00000063 SystemOut    O inbytes is:  [B@ec00ec0
[7/25/05 19:34:18:321 AKDT] 00000063 SystemOut    O outbytes is:  null
[7/25/05 19:34:18:321 AKDT] 00000063 SystemOut    O outstr is:  null
[7/25/05 19:34:18:321 AKDT] 00000063 SystemOut    O dir is:  null
[7/25/05 19:34:18:321 AKDT] 00000063 SystemOut    O fn is:  null
[7/25/05 19:34:18:321 AKDT] 00000063 SystemOut    O array size is:  52
```

__ c. Try copying additional files to the c:\ffdir\eventdir if you'd like.

____ 3.   Clean up test environment.

__ a. In the Servers view, right-click on WPS Server v6.0 and select **Add and remove projects…**.

__ b. Select **FFTestInboundApp** and click **< Remove**. Click **Finish**.

__ c. Right-click on **WPS Server v6.0** and select **Stop**

What you did in this exercise

In this exercise you installed and deployed the WebSphere Adapter for Flat Files into an SCA application. You also configured the adapter to poll for and process events on the file system.