# **FTP Adapter Outbound Lab**

What this exercise is about
Lab Requirements1
What you should be able to do1
Introduction
Exercise Instructions
Part 1: Initialize workspace and Import RAR file into WebSphere Integration Developer
Part 2: Use ESD Wizard to Generate Business Objects and other Artifacts
Part 3: Test the Adapter Application using WebSphere Process Server Test Environment
Part 4: Restore Server Configuration
What you did in this exercise

# What this exercise is about

The objective of this lab is to provide you with an understanding of the WebSphere<sup>®</sup> Adapter for FTP and outbound processing. In this lab you will deploy the WebSphere Adapter for FTP using WebSphere Integration Developer, and integrate it into an SCA application that processes outbound requests to the remote file system.

# Lab Requirements

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

- WebSphere Integration Developer V6.0.2 installed
- WebSphere Process Server V6.0 Test Environment installed
- WebSphere Adapter for FTP V6.0.2 installed
- Unzip LabFiles602.zip to your C:\ (your root) drive
- FTP server installed and configured

# 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 Enterprise Service Discovery wizard to configure the FTP Managed Connection Factory Properties and Resource Adapter Properties to generate Business Objects and other artifacts

2007 March, 07

Page 1 of 30

- Deploy the adapter application onto the WebSphere Process Server test environment
- Test the above deployed application using WebSphere Process Server test environment
- Restore the server configuration

# Introduction



The J2EE Client makes a SCA call by giving the outbound operation name and the input data object and then the custom data binding is called and it invokes other content-specific data binding based on the ContentType set in the child data object.

The protocol specific properties like directory Name, file name which are required during outbound operation are populated in the wrapper data object present in BG. These values are set on to the FTPFileUnstructuredRecord in the custom data binding and sent to the adapter.

# **Exercise Instructions**

Some instructions in this lab may be Windows<sup>®</sup> 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 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 <sup>®</sup> /UNIX <sup>®</sup> Location
<wid_home></wid_home>	C:\Program Files\IBM\WebSphere\6.0	
<wps_home></wps_home>	<wid_home>\runtimes\bi_v6</wid_home>	
<ffadapter_home></ffadapter_home>	C:\Program Files\IBM\ResourceAdapters\FTP\adapter\FTP	
<lab_files></lab_files>	C:\Labfiles602	/tmp/Labfiles60
<workspace></workspace>	C:\LabFiles602\FTPOutbound\workspace	
<temp></temp>	C:\temp	/tmp

**Windows users note**: When directory locations are passed as parameters to a Java<sup>™</sup> program such as EJBdeploy or wsadmin, it is necessary to replace the backslashes with forward slashes to follow the Java convention. For example, C:\LabFiles602\ would be replaced by C:/LabFiles602/

# Part 1: Initialize workspace and Import RAR file into WebSphere Integration Developer

This part of the lab will guide you through the steps for starting WebSphere Integration Developer with a new workspace, and then import the connector file **CWYFT\_FTPFile.rar** into your new workspace.

- 1. Start WebSphere Integration Developer V6.0.2 with a new workspace
  - Select Start > Programs > IBM WebSphere > Integration Developer V6.0.2
     > WebSphere Integration Developer V6.0.2
  - From the Workspace Launcher window, enter **WORKSPACE>** for the Workspace field

Workspace Launcher	>
<b>Select a workspace</b> IBM WebSphere Integration Developer stores your projects in a directory of Select the workspace directory to use for this session.	alled a workspace.
Workspace: C:\LabFiles602\FTPOutbound\workspace	▼ <u>B</u> rowse
Use this as the default and do not ask again	OK Cancel

• Click on the the button on the right hand corner to close the Welcome page and proceed with the workbench

🥵 Business Integration - IBM WebSphere Integration Developer	
Eile Edit <u>N</u> avigate Se <u>a</u> rch Project <u>R</u> un <u>Wi</u> ndow <u>H</u> elp	
Helcome X	$\land \leftarrow \Rightarrow \neg = =$
WebSphere. Integration Developer	

- \_\_\_\_ 2. Import FTP Adapter RAR file
  - From main menu, select **File > Import...**

🚯 Busine:	ss Integra	ition - IB	M WebSpl
<u>F</u> ile <u>E</u> dit	<u>N</u> avigate	Se <u>a</u> rch	Project E
New		Alt+Sh	nift+N 🕨 🕨
⊆lose		Ctrl+F	4
Close A	sil	Ctrl+S	hift+F4
_ <u>S</u> ave		Ctrl+S	
Save A	ş		
Sav <u>e</u> A	.[]	Ctrl+S	hift+S
Rever <u>t</u>			
Mo <u>v</u> e			
Rena <u>m</u>	e.,,	F2	
Refres	ħ	F5	
Print		Ctrl+P	
Switch Open E	<u>W</u> orkspace External File	 	
🚵 <u>I</u> mport			
🛃 Exp <u>o</u> rt			

o Select RAR file from the Import window and then click Next

🚯 Import	×
Select Import an external Connector RAR file into a Connector project	Ľ
Select an import source: Log File Performance Call Graph Probe Profiling file Profiling filter Project Interchange RAR file Security Certificate	
< <u>B</u> ack <u>N</u> ext > Einish	Cancel

- Click on the Browse... button next to the Connector file field to select the CWYFT\_FTPFile.rar
- Uncheck the check box next to Add module to and EAR project and click Finish

🚯 Import		×
Connector Import Import a Connecto	t or project from the file system	æ
Connector file:	esourceAdapters\FTP\adapter\FTP\deploy\CWYFT_FTPFile.rar	Browse
Connector project:	CWYFT_FTPFile	N <u>e</u> w
🔲 Overwrite exi	isting resources without warning,	
🔲 Delete proj	ect on overwrite	
Target server:	WebSphere Process Server v6.0	N <u>e</u> w
-	OAdd module to an EAR project.	
EAR project:	CWYFT_FTPFileEAR	Ne <u>w</u> ,
	< <u>B</u> ack Mext > Einish	Cancel

 Click on No from Confirm Perspective Switch window to continue with the Business Integration perspective



- 3. Create directory structure on your FTP Server
  - o Log onto your FTP Server using your ftpuser and password
  - o Create an Output directory named **Outputdir** under the user's home directory
    - mkdir Outputdir

# Part 2: Use ESD Wizard to Generate Business Objects and other Artifacts

- 1. Launch the Enterprise Service Discovery wizard
  - \_\_\_\_a. Select File > New > Enterprise Service Discovery



\_\_ 2. Select IBM WebSphere Adapter for FTP (version 6.0.1) from the 'CWYFT\_FTPFile' Connector Project and click Next

🚯 Enterprise Service Discovery			×
Select an Enterprise Service Resource	Adapter		
Select a resource adapter to use to discover	a service.		۵,
IBM WebSphere Adapter for FTP (ve ) IBM WebSphere Service Registry an ) WBI Adapter Artifact Importer	ersion 6.0.2) from the 'CV d Repository	VYFT_FTPFile' Connecto	or Project
IBM WebSphere Adapter for FTP		[Import Res	burce Adapter
	< <u>B</u> ack <u>N</u> ext	> Einish	Cancel

3. Configure settings for the Discovery agent

You will specify the properties to initialize the Resource Adapter and Enterprise Service Discovery agent

\_\_\_\_a. Click Browse... button next to the Folder Name field and select the folder <FTPADAPTER\_HOME>\samples that contains the XSD file for Customer Business Object

## Note: For your convenience, the Customer.xsd is also placed under <LAB\_FILES>\FTPFiles.

- \_\_\_\_b. Select Big5 from the drop down list for the Character Set field
- \_\_\_\_ c. Select **text/xml** for the **Content Type** field. Once the content type is selected, the **DataBinding Type** will be automatically set to **XMLBOSerializerDataBinding**
- \_\_\_\_\_d. Click the **Show Advanced** >> button to see the Log file location and Logging level options for discovery log and click **Next** leaving the default log file location

🚯 Enterprise Service Disco	wery	×
Configure Settings for Disc	covery Agent	
Specify the properties to initia	lize the resource adapter and the enterprise service discovery agent.	0
Connection Configuration		
Folder Name:	C:\Program Files\IBM\ResourceAdapters\FTP\adapter\FTP\Samples	Browse
Charater Set:	(Big5)	
Content Type:	(text/xml)	
DataBinding Type:	(XMLBOSerializerDataBinding)	
Specify BO Properties		
BiDi Properties		
🔲 Bidi transformatio	n	
Bidi ordering schema:	Implicit	
Text direction:	LTR	
🔽 BiDi SymmetricSw	apping	
Bidi shaping;		
Bidi numeric shaping:	Nominal	
Hide Advanced <<		
Logging options		
Log file output location:*	C:\Labfiles602\FTPInbound\workspace\.metadata\FTPFileMetadataDiscoveryImpl.log	Browse
Logging Level:	SEVERE	
	< <u>B</u> ack <u>N</u> ext > <u>Finish</u>	Cancel

- \_\_\_\_4. Find and discover the enterprise services. In this step you will select the business objects and services to be used with the adapter
  - \_\_\_\_a. From the Enterprise Service Discovery window, click on **Execute Query** button. You will see a **Customer** business object under **Objects discovered by query**
  - \_\_\_\_b. Select **Customer** business object and click **Add to import list** button. The Customer business object will now be displayed under the **Objects to be imported**

🚯 Enterprise Service Discover	У	×
Find and Discover Enterprise To discover objects on the enterp press "Execute Query". Once disc imported.	<b>Services</b> rise system, create a query by pressing "Edit Query" and the overed, press "Add to import list" to specify the objects to be	
Query: < execute default query > Execute Query Objects discovered by query:		Edit Query,
Objects to be imported:	□ Add to import list	Filter Clear Filter Details
(1) Customer		Remove
	< <u>Back</u> <u>Next</u> Finish	Cancel

- \_\_\_ c. Click Next
- 5. Configure the objects that will be imported by the discovery agent
  - \_\_\_\_a. From the Configure objects window, select **Outbound** from the dropdown menu for the **Service Type**. Note the operations available for the selected Service Type

\_\_\_\_b. Enter the **BO Location** as **FTPOutBO** and click **Next** 

🚯 Enterprise Serv	vice Discovery	×
Configure Object	\$	
Specify the proper	ties for the objects that will be imported by the discovery agent.	
ServiceType:	Outbound	
Namespace: *	http://www.ibm.com/xmlns/prod/websphere/j2ca/ftp	
Operations:		
Create Append Overwrite Delete Exists List Retrieve ExecuteFTPScript ServerToServerFile	eTransfer	Add Remove
BO Location:	FTPOutBO	
Function Selector:	WBIFunctionSelector	
	< <u>B</u> ack <u>N</u> ext > Einish	Cancel

- \_\_\_\_6. Specify the properties for the artifacts that will be generated in your workspace
  - \_\_\_\_a. Create a new module
    - 1) Click on the New... button next to the Module field

2) From the New Integration Project window, ensure that the radio button next to **Create a** module project is selected and click **Next** 

🚯 New Integration Project	×
Integration Project Select the type of integration project to create.	
<ul> <li>Create a module project.</li> <li>Create a mediation module project.</li> </ul>	
< Back Next > Einish	Cancel

3) Enter FTPOutboundModule for the Module Name field and click Finish

🚯 New Module				×
Module				
Create a new busine used for developmer and deploying to the	ss integration m nt, version mana runtime enviror	odule. A modu Igement, orga Iment.	ule is a project than nizing resources,	at is
Module Name FTPO	DutboundModule	•		
Module Location				
🔽 Use <u>d</u> efault				
Directory; C:\Lab	Files602\FTPOut	bound',worksp	ace\FTPOutbou	Browse
© Open module as Business integration ( They can contain mar	ssembly diagram modules can be on ny types of comp	deployed and ponents, such	run on WebSpher as business proc	e Process Server. esses, assembled
together for the purp	ose of business	integration.		
			1	<b>1</b> (
They can contain mai together for the purp	ny types of comp nose of business	integration.	as business proc	esses, assembled

\_\_\_\_ b. The module which is created above will appear under the Module field of the Generate Artifacts window

🚯 Enterprise Service Dis	overy	×
Generate Artifacts		
Specify the properties for the artifacts that will be generated in your workspace.		
Properties for Interface —		
Module:	FTPOutboundModule	
Namespace:	http://FTPOutboundModule/FTPFileOutboundInterface	
	🔽 Use Default Namespace	
Folder:	Browse	
Name:	* FTPFileOutboundInterface	
Description:		
	Edit operation names	

- \_\_\_\_\_ c. From the Generate Artifacts window, select the radio button next to Use discovered connection properties. This will make the FTP Adapter Managed Connection Factory and Resource Adapter Properties visible
- \_\_\_\_ d. Enter the following fields under the FTP Adapter Managed Connection Factory Properties:
  - 1) FtpUrl: ftp://<FTP Server IP Address>
  - 2) UserName: **<user>** (username using which you connect to your FTP machine)
  - 3) Password: **<password>** (password for the above user to connect to your FTP machine)
  - 4) EisEncoding: **Big5** (from the drop down menu)

\_\_\_\_e. Enter the following fields under Resource Adapter Properties:

1) LogFilename:	C:\FTPRA\outlog.txt
-----------------	---------------------

2) TraceFilename: C:\FTPRA\outtrace.txt

Deployment properties	
Deploy connector with module	
J2C Authentication Data Entry:	
Specify the connection properties which will be used	o connect to the Enterprise Information System at runtime:
O Use connection properties specified on server	
<ul> <li>Use discovered connection properties</li> </ul>	
Connection properties	ti
FIP Adapter ManagedConnection Factory Proper	des
Ftp Url: * Ttp://iocalnost	
User Name: root	
Password:	
Staging Directory:	
Custom Parser Class Name:	
EIS Encoding: Big5	
Second Server Directory:	
Second Server User Name:	
Second Server Password:	
Socks Proxy Host:	
Socks Proxy Port: 1080	
Socks Proxy User Name:	
Socks Proxy Password:	
ResourceAdapter Properties	
Logging and Tracing	
Adapter ID [String]: * ResourceAdap	ter
Log file size [Integer]: 0	
Log file name [String]: C:\FTPRA\Ou	log.txt
Log Files [Integer]:	
Trace file size [Integer]: 0	
Trace file name [String]: C:\FTPRA\Ou	trace.txt
Trace files [Integer]: 1	

## \_\_\_ f. Click Finish

- 7. You can also configure/change the adapter properties using Assembly Editor
  - \_\_\_\_a. Change to the Business Integration perspective if not open
    - 1) Select Window > Open Perspective > Other....



2) From the Select Perspective window, select Business Integration (default) and click OK



\_ b. Expand FTPOutboundModule and double-click FTPOutboundModule to open it in Assembly Editor



- \_\_\_\_ c. Click on FTPFileOutboundInterface from the Assembly Editor and select Properties tab from the bottom
- \_\_\_\_\_d. Select **Binding** under Properties and select **Endpoint Configuration** under Binding itself and then select the **Connection** tab

 Ensure the radio button next to Specify properties for pre-configured new J2EE Connector Architecture recourse is selected and then click on Managed Connection Factory Properties to expand it. You can change the properties that you entered using the ESD wizard in the previous steps and save those changes before you deploy the application onto the server

Properties X Problem	s Servers Console	
Description	📑 Import: FTPFileOutboundInterf	ace (EIS Binding)
Details	Connection Resource Adapter	
Binding  End-point configuration  Method bindings  Security attributes	Select configuration view option: O Specify JNDI name for pre-configu- Specify properties for configuring	ured J2EE Connector Architecture resource new J2EE Connector Architecture resource
	<ul> <li>Managed Connection Factory Pro</li> </ul>	perties
	Managed connection factory class name:	com.ibm.j2ca.ftp.FTPFileManagedConnectionFactory
	Managed Connection Factory Properties	
	Ftp Url: *	ftp://localhost
	User Name:	root
	Password:	*******
	Staging Directory:	
	Custom Parser Class Name:	
	EIS Encoding:	Big5 <
	Second Server Directory:	
	Second Server User Name:	
	Second Server Password:	
	Socks Proxy Host:	
	Socks Proxy Port:	1080
	Socks Proxy User Name:	
	Socks Proxy Password:	

# 2) Click on Resource Adapter tab to view/change the Resource Adapter Properties

Properties X Problem	is Servers Console	
Description	📄 Import: FTPFileOutbou	ndInterface (EIS Binding)
Details	Connection Resource Adapter	
Binding	Resource adapter name:	FTPOutboundApp.IBM WebSphere Adapter for FTP
End-point configuration	Resource adapter class name:	com.ibm.j2ca.ftp.FTPFileResourceAdapter
<ul> <li>Method bindings</li> </ul>	Resource Adapter Bean Properti	es
<ul> <li>Security attributes</li> </ul>	Logging and Tracing	
	Adapter ID [String]: *	ResourceAdapter
	Log file size [Integer]:	0
	Log file name [String]:	C:\FTPRA\Outlog.txt
	Log Files [Integer]:	1
	Trace file size [Integer]:	0
	Trace file name [String]:	C:\FTPRA\Outtrace.txt
	Trace files [Integer]:	1
	EIS Encoding:	▼

# Part 3: Test the Adapter Application using WebSphere Process Server Test Environment

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

FTP adapter supports the following outbound operations:

**Create** –file with the specified name is created in the specified directory of the FTP Server. The content of the file can be given as either a part of the request from the client of it can be picked from the local file system where the adapter is running according to the configuration. If the file to be created already exists, an FTPFileCreateException is thrown. Existing file is not overwritten.

**Append** – the file with specified name in the specified directory of the FTP Server is appended with the content sent across in the request. If the file to be appended doesn't exist, an FTPFileAppendException is thrown to the calling component.

**Delete** – deletes the file from the specified directory in the FTP Server. If the file to be deleted doesn't exist, an FTPFileDeleteException is thrown to the calling component.

**Retrieve** – response returns the content of the file(s) which is/are specified in the request. The file names are input as comma separated in a Filename attribute. If the file doesn't exist, an FTPFileRetrieveException is thrown to the calling component.

**Overwrite** – overwrites the file in the directory with the content specified in the request. If the file to be updated doesn't exist, an FTPFileOverwriteException is thrown to the calling component.

**Exists** - checks the existence of a file. Returns true if the file name in the request exists in the specified directory and returns false if either directory or file name doesn't exist.

**List** - returns all the file names in the all directories specified in the request. If the directory specified doesn't exist, an FTPFileListException is thrown to the calling component.

**ServerToServerFileTransfer** –transfers the specified file from a directory of one FTP server to a directory of another FTP server. If all the details of the two servers are not specified, an FTPFileServerToServerFileTransferException is thrown.

**ExecuteFTPScript** –executes the commands present in a FTP script file, which exists on the local machine where adapter is installed. The location of the script file is given in DirectoryPath and Filename attributes of the FTPFileUnstructuredRecord. This operation executes only the commands supported by FTP Server.

- \_\_\_\_\_1. Add the project to the server for the WebSphere Process Server Test Environment
  - \_\_\_\_a. Right-click on WebSphere Process Server v6.0 under the server view and select Add and remove projects... from the context menu

Properties Problems 👫 Servers 🗙			
Server	Host name		Status
🛅 WebSphere ESB Server v6.0	localhost		
WebSphere Process Server v6.0	localhost	New	•
		Open	
		Delete	
		🎋 Debug	
		🜔 Start	
		🔊 Profile	
		Restart	+
		📕 Stop	
		🔄 Disconnect	
		追 Publish	
		Monitoring	•
		🏪 Add and remov	e projects

\_\_\_ b. From the Add and Remove Projects window, select FTPOutboundModuleApp under Available projects panel and click Add >

🚯 Add and Remove Projects			×
Add and Remove Projects			
Modify the projects that are config	ured on the server		
Move projects to the right to configu	ire them on the server		
<u>Available projects:</u>		Configured projects:	
	A <u>d</u> d >		

\_\_\_\_ c. You will now see the **FTPOutboundModuleApp** added to the **Configured projects**. Click **Finish** 

🚯 Add and Remove Projects		×
Add and Remove Projects Modify the projects that are configu	red on the server	
Move projects to the right to configure	e them on the cerver	
<u>Available projects:</u>		Configured projects:
	A <u>d</u> d >	
	< <u>R</u> emove	
	Add Alj >>	
	<< Remove All	
,		,
	< <u>B</u> ack <u>N</u> ext >	Einish Cancel

Wait for the project to be added to the server and the application to start. The server will be started in Debug mode if it is not already started before.

- \_\_\_\_\_ 2. **Test Scenario**: outbound for **createCustomer** operation (Non\_PassThrough)
  - \_\_\_\_a. Right-click the FTPOutboundModule and select Test > Test Module from the context menu

Business Integration	×	Assembly Diagram: FTPOutboundMode
	🕀 🕞 🗐	× 1
E-12 FTPOutboundM		
ETPOutbour	Ne <u>w</u>	
+- 🚝 Business Lo	Open Dependency Edito	
🕀 🔂 Data Types	Show Files	
+ 🗊 Interfaces		×
🗄 🖓 👗 Mapping	🛅 Сору	& >
	Paste	9.
	Move	
	💥 Delete	
	··· =	
	🚵 Import	
	🛃 Export	
	🐑 Refresh	
	🚭 Coperate Documentatio	
	Test	🕨 🥬 Test <u>M</u> odule

\_\_\_\_b. The FTPOutboundModule\_Test editor will be opened in Assembly Editor

Assembly Diagram: FTPOutboundModule	it X						
Events				å⊳ ⊕⊳	å 🖽		<b>_</b>
	dial. Cashierra ha						
<ul> <li>Select the component, interface, and operation you would like to invoke.</li> </ul>		run.					
Events		opercies					
ĝ▶ Invoke	<ul> <li>Detailed Pr</li> </ul>	operties					
	Configuration:	Default I	Module Test			•	
	Module:	FTPOut	oundModule			•	
	Component:	FTPFileC	outboundInterface			•	
	Interface:	FTPFileC	outboundInterface			•	
	Operation:	append	Iustomer			•	
	Initial reguest p	arameter	s				
	Name		Туре	Value			
	🖃 appendCus	tomer	CustomerWrapperBG				
	verb		string	<null></null>			
	Custome	erWra	CustomerWrapper				
	Direct	toryPath	string				
	Filena	ame	string				
	Data	Conne	string				
	FileTr	anste	string				
	Secor	ndSer	string				
	Secor	ndSer	string				
	Secor	nd5er	string			-	
	1 Secon	III DHE	SCOOL				
	Data Poo <u>l</u>				Contin	nye	
Fuente Configurations							

\_\_\_ c. Under **Detailed Properties**, for the **Operation** field, select **createCustomer** from the drop down list

## \_\_\_\_d. Fill out the fields for **Initial request parameters**:

# 1) DirectoryPath: **<Outputdir>**

## Note: Outputdir is the output directory that you created in Step 3 of Part 1.

## 2) Filename: Customer.xml

3) Give any values for the fields (CustomerName, Address, City, and State) under Content

## General Properties

<ul> <li>Detailed Pr</li> </ul>	operties	
Configuration:	Default Module Test	•
Module:	FTPOutbound	•
Component:	FTPFileOutboundInterface	•
Interface:	FTPFileOutboundInterface	•
Operation:	(createCustomer)	-

Initial reguest parameters

Name	Туре	Value
createCustomerInput	CustomerWrapperBG	
verb	string	<null></null>
<ul> <li>CustomerWrapper</li> </ul>	CustomerWrapper	
DirectoryPath	string	/root/OutputDir
Filename	string	Customer.xml
DataConnectionMode	string	
FileTransferType	string	
SecondServerDirec	string	
SecondServerUser	string	
SecondServerPass	string	
FileContentEncoding	string	
IncludeEndBODeli	string	
FileInLocalDirectory	boolean	false
LocalDirectoryPath	string	
LocalArchivingEna	boolean	false
LocalArchiveDirFor	string	
StagingDirectory	string	
<ul> <li>Content</li> </ul>	Customer	
CustomerName	string	ABC
Address	string	11501 Burnet Rd
City	string	Austin
State	string	ТХ
Data Pool		Continue

# o Click Continue

 From Deployment Location window, select WebSphere Process Servers > WebSphere Process Server v6.0 and click Finish

🚯 Deployment Location	×
<b>Select Deployment Location</b> This server instance is currently running.	
Deployment location:	
WebSphere Process Servers  KebSphere Process Server v6.0  Eclipse 1.4 JVM	New Server
Mode: Run	<u>v</u>
Use this as the default and do not ask again	
	<u>Einish</u> Cancel

\_\_\_\_e. Click on the **Invoke (FTPFileOutboundInterface:createCustomer)** and you will see a window similar to the following, that contains the data you just entered in the previous steps:

lencs	General Properties				
🖃 🕌 Invoke (FTPFileOutboundInterface:createCustomer)	ustomer)				
Started	Configuration:	Default Module	Default Module Test		
Stopped	Module:	FTPOutbound			
		FTPFileOutboundInterface FTPFileOutboundInterface			
	Component:				
	Interface:				
	Operation:	createCustome	=[		
	Initial reguest p	parameters			
	Name		Туре	Value	
	🖃 createCust	omerInput	CustomerWrapperBG	1000	
	verb		string	<null></null>	
	CustomerWrap DirectoryPat Filename	rWrapper	CustomerWrapper		
		toryPath	string	/root/OutputDir	
		ame	string	Customer.xml	
	Data	ConnectionMode	string	1. Sec	
	FileTr	ansferType	string		
	SecondS SecondS FileConte IncludeEr FileInLoc LocaDire	ndServerDirec	string		
		ndServerUser	string		
		ndServerPass	string		
		ontentEncoding	string		
		deEndBODeli	string		
		LocalDirectory	boolean	false	
		DirectoryPath	string		
	Local	ArchivingEna	boolean	false	
	Local/	ArchiveDirFor	string		
	Stagir	ngDirectory	string		
	- Conte	ent	Customer		
	Cu	stomerName	string	ABC	
	Ad	ldress	string	11501 Burnet Rd	
	Cit	cy .	string	Austin	
	Str	ate	string	TX	

\_\_\_\_\_f. To verify the results, check the **Outputdir** folder on the machine where you have FTP server. You should have a **Customer.xml** file created under that directory with the following contents:

<pre></pre> // version="1.0" encoding="UTF-8"?>
<customer:customer xmlns:customer="http:/&lt;/td&gt;&lt;/tr&gt;&lt;tr&gt;&lt;td&gt;/www.ibm.com/xmlns/prod/websphere/j2ca/ftp/customer" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="customer:Customer"></customer:customer>
<customername>ABC</customername>
<address>11501 Burnet Rd</address>
<city>Austin</city>
<state>TX</state>

3. **Test Scenario**: outbound for **createFTPFile** operation (PassThrough)

- \_\_\_\_a. Click on Invoke button <sup>\*</sup> at the top from the FTPOutboundModule\_Test window
- \_\_\_\_b. FTPOutbpoundModule\_Test window will be opened in Assembly Editor
- \_\_\_\_ c. Under **Detailed Properties**, for the **Operation** field, select **createFTPFile** from the drop down menu
- \_\_\_\_d. Fill out the fields for **Initial request parameters**:
  - 1) DirectoryPath: <Outputdir>

Note: Outputdir is the output directory that you created in Step 3 of Part 1.

## 2) Filename: ftptest.txt

 On the row that contains Content, under Type and select browse button and then select UnstructuredContent from the Data Type Selection window

# 3) Enter some string data for **AsText** field

#### General Properties

Ŧ	Detai	ed	Pro	perties	
---	-------	----	-----	---------	--

Configuration:	Default Module Test	•
Module:	FTPOutbound	T
Component:	FTPFileOutboundInterface	•
Interface:	FTPFileOutboundInterface	•
Operation:	createFTPFile	Ŧ

Initial reguest parameters

Name	Туре	Value
	FTPFileBG	
verb	string	<null></null>
- FTPFile	FTPFile	
DirectoryPath	string	/root/OutputDi
Filename	string	ftptest.txt
ChunkInfo	string	
DataConnectionMode	string	
FileTransferType	string	
SecondServerDirec	string	
SecondServerUser	string	
SecondServerPass	string	
FileContentEncoding	string	
IncludeEndBODeli	string	
FileInLocalDirectory	boolean	false
LocalDirectoryPath	string	
LocalArchivingEna	boolean	false
LocalArchiveDirFor	string	
StagingDirectory	string	
<ul> <li>Content</li> </ul>	UnstructuredContent [any	
ContentType	string	
ObjectName	string	
AsText	string	FTP Testing
AsBinary	hexBinary	0
Data Pool		Continue

\_\_\_\_e. Click Continue

 Click on the Invoke (FTPFileOutboundInterface:createFTPFile) and you will see a window similar to the following, that contains the data you just entered in the previous steps:

Events +	General Properties		
Invoke (FTPFileOutboundInterface:createCustomer)	▼ Detailed Properties		
	Configuration: Default Module Test		
Invoke (FTPFileOutboundInterface:createFTPFile)	Component: <u>FTPFileOutbour</u> Interface: <u>FTPFileOutbour</u>	ndInterface ndInterface	
	invocation parameters		
	Name	Туре	Value
	createFTPFileInput	FTPFileBG	
	verb	VerbType	<unset></unset>
	<ul> <li>FTPFile</li> </ul>	FTPFile	
	DirectoryPath	String	/root/OutputDir
	Filename	String	ftptest.txt
	ChunkInfo	String	
	DataConnectionMode	String	
	FileTransferType	String	
	SecondServerDirec	String	
	SecondServerUser	String	
	SecondServerPass	String	
	FileContentEncoding	String	
	IncludeEndBODeli	String	
	FileInLocalDirectory	Boolean	false
	LocalDirectoryPath	String	
	LocalArchivingEna	Boolean	false
	LocalArchiveDirFor	String	
	StagingDirectory	String	
	<ul> <li>Content</li> </ul>	UnstructuredContent	
	ContentType	String	
	ObjectName	String	
	AsText	String	FTP Testing
	AsBinary	Bytes	[B@d877129

\_\_\_\_\_f. To verify the results, check the **Outputdir** folder on the machine where you have FTP server. You should have the **ftptest.txt** file created under that directory with the following contents:



# Part 4: Restore Server Configuration

\_\_\_\_\_1. Close the FTPOutboundModule\_Test window and click No for the Save Resources window



2. Right-click on WebSphere Process Server v6.0 under the Servers view and select Add and remove projects... from the context menu

Properties Problems 👫 Servers 🗙			
Server	Host name		Status
🛅 WebSphere ESB Server v6.0	localhost		
WebSphere Process Server v6.0	localhost	New	•
		Open	
		Delete	
		🎋 Debug	
		🜔 Start	
		🔊 Profile	
		Restart	+
		📕 Stop	
		🔄 Disconnect	
		担 Publish	
		Monitoring	•
		🏪 Add and remov	e projects

\_\_\_\_ 3. Select FTPOutboundModuleApp under Configured projects and click < Remove

🚯 Add and Remove Projects	×
Add and Remove Projects	
Modify the projects that are configu	red on the server
Move projects to the right to configur	e them on the server
<u>A</u> vailable projects:	Configured projects:
	Add >

- 4. Click **Finish** after you see the application moved to Available projects. Wait until the application is unpublished
- \_\_\_\_5. Right-click on **WebSphere Process Server v6.0** from the Servers view and select **Stop** from the context menu

Properties Problems 🛪 Servers 🗙 Console				
Server	Host name	Status		State
🛅 WebSphere ESB Server v6.0	localhost			
m WebSphere Process Server v6.0	localhost	🎝 Started	New	
			Open	
			Delete	
			🐝 Debug	
			D Start	
			🔎 Profile	
			Restart	+
			📕 Stop	

# What you did in this exercise

In this lab, you imported the FTP Adapter RAR file into your WebSphere Integration Developer workspace and integrated it into an SCA application that creates a file to the file system.

You made use of Enterprise Service Wizard available in WebSphere Integration Developer to specify Managed Connection Factory Properties and Resource Adapter Properties which, after deploying onto the server will generate Business Objects and other artifacts.

In the end you deployed and then tested the adapter application for the operations available.