IBM RATIONAL APPLICATION DEVELOPER 6.0 – LAB EXERCISE

Building a Web Service with IBM Rational Application Developer v6.0

What this exercise is about

IBM Rational Application Developer v6.0 provides a complete set of tools and support for building and testing Web Service applications. This lab exercise is about creating a Web Service with IBM Rational Application Developer v6.0 for WebSphere Application Server v6.0.

Lab Requirements

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

- IBM Rational Application Developer v6.0 beta with embedded WebSphere Application Server v6.0 Test Environment installed with beta patches installed
- Lab source files (Labfiles60.zip) must be extracted to the local system (preferably C:\).

What you should be able to do

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

- Create a Web Service provider and test it with the Web Services Explorer included in IBM Rational Application Developer v6.0.

Introduction

In this lab, you will be using the WebSphereBank application. This J2EE 1.4 application is a simple banking application composed of EJB, Web, and Application Client modules. In the EJB module there is a session bean and multiple entity beans. Before being able to create a Web Service with this simple application, you will need to generate mappings for the entity bean to a Cloudscape data source. Also, you will generate deploy code used by the EJB container of the J2EE server to handle the calls to the entity bean and persisting of the data. Cloudscape is a small, in memory, object-relational database based purely on Java. This database needs no install. Therefore it can be set up very easily and you can quickly test your application to ensure it is working properly.

After completing the Lab Setup, you will create a Web Service and then test it with the Web Services Explorer. The Web Services Explorer acts as a Web Service client to access your Web Service when there is not a Web Service client created yet for quick testing to insure the Web Service works before going through the trouble of creating a Web Service Client. When finished, you will have a valid banking Web Service ready to be accessed by a true Web Service Client.

Exercise Instructions:

Because these instructions are not operating-system specific, the directory locations will be specified in the lab instructions using symbolic references, as follows:

Reference Variable	Windows Location	AIX/UNIX Location
<LAB_FILES>	C:\Labfiles60	/tmp/Labfiles60
<RAD_HOME>	C:\Program Files\IBM\Rational\SDP\6.0	/opt/IBM/RSDP/6.0

Part 1: Lab Setup

____ 1. Before you begin the lab, you will need to back up the test server. This backup will preserve the server and allow you to return to that configuration once you complete this exercise. Instructions on restoring the configuration are included at the end of this exercise.

__ a. Open a Windows **Command Prompt** and navigate to `<RAD_HOME>\runtimes\base_v6\bin\`

__ b. Backup the server by issuing the following command:

```
backupConfig "C:\Program Files\IBM\Backup.zip"
```

____ 2. Start IBM Rational Application Developer v6.0.

__ a. Select **Start** > **Programs** > **IBM Rational** > **IBM Rational Application Developer v6.0** > **Rational Application Developer**.

__ b. For workspace, specify **<LAB_FILES>\IRAD_WebServices\workspace**.

__ c. When IBM Rational Application Developer v6.0 opens, close the **Welcome page**.

NOTE: If you receive an Auto Launch Configuration Change Alert window, select **Yes** to change the auto- launch workspace to the path of the current lab.

____ 3. Import the WebSphereBank application into IBM Rational Application Developer.

__ a. Click **File > Import....**

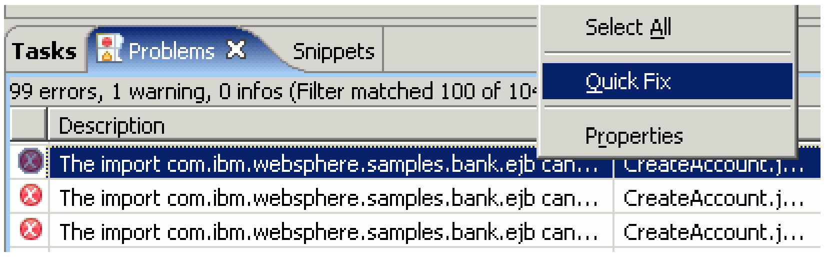
__ b. Click **EAR file** and select **Next.**

__ c. Click **Browse...** and navigate to **<LAB_FILES>\IRAD_WebServices\WebSphereBank.ear** and Click **Open**.

__ d. Click **Finish**.

____ 4. A number of items will be displayed in the Problems view. These errors should be resolved before continuing.

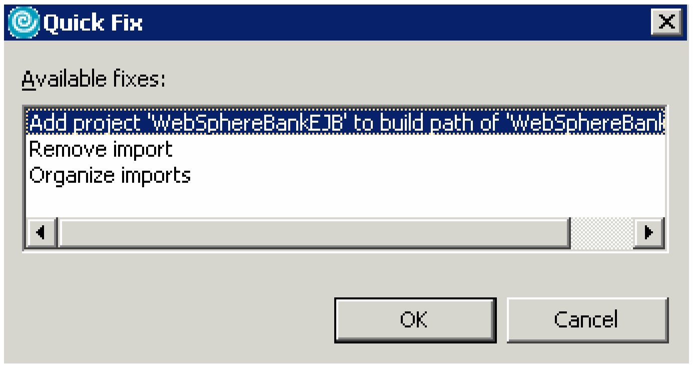
__ a. Right click on an error and select **Quick Fix.**



99 errors, 1 warning, 0 infos (Filter matched 100 of 10

Description	
The import com.ibm.websphere.samples.bank.ejb can...	CreateAccount.j...
The import com.ibm.websphere.samples.bank.ejb can...	CreateAccount.j...
The import com.ibm.websphere.samples.bank.ejb can...	CreateAccount.j...

__ b. A list of available fixes will appear. Select **Add project 'WebSphereBankEJB' to build path of 'WebSphereBankWeb'** and click **OK.**   The errors will be resolved.

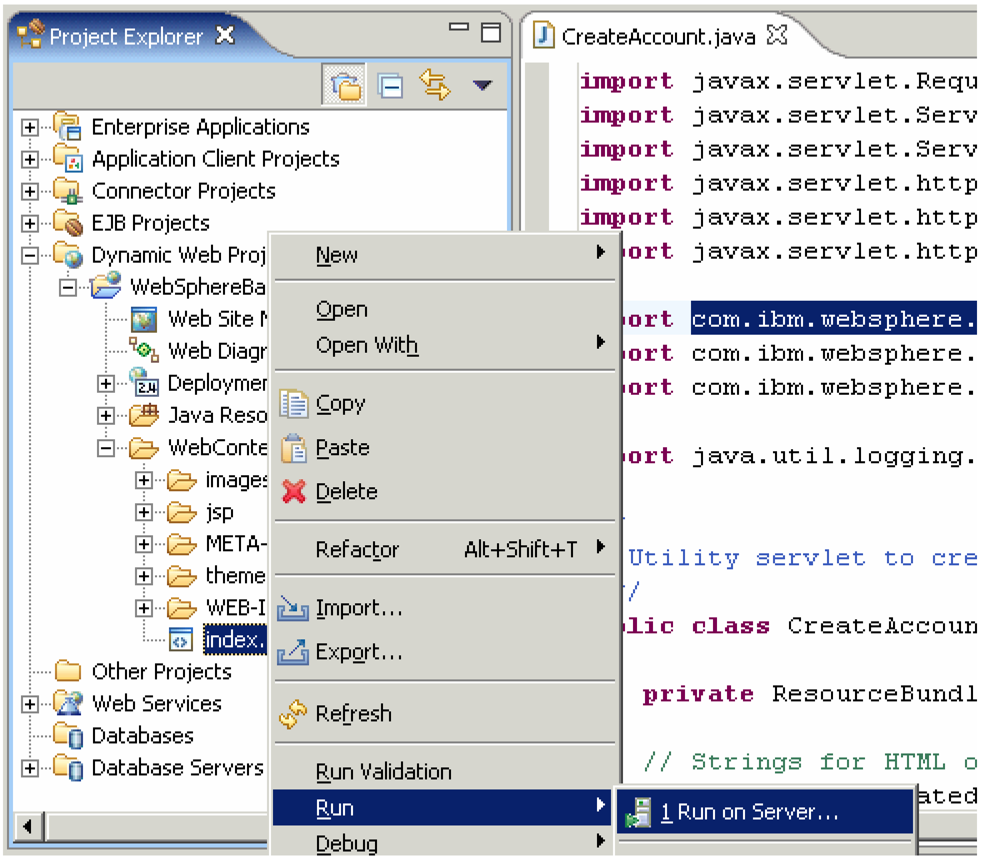

Quick Fix
Available fixes:
Add project 'WebSphereBankEJB' to build path of 'WebSphereBank
Remove import
Organize imports
OK
Cancel

__ c. Close any Java files which are opened in the process.

Part 2: Initializing database components

_____ 1. Start the server with the WebSphereBank project while initializing the database and datasource.

__ a. In the Project Explorer view, navigate to **Dynamic Web Projects** > **WebSphereBankWeb** > **WebContent** and right click on index.html.

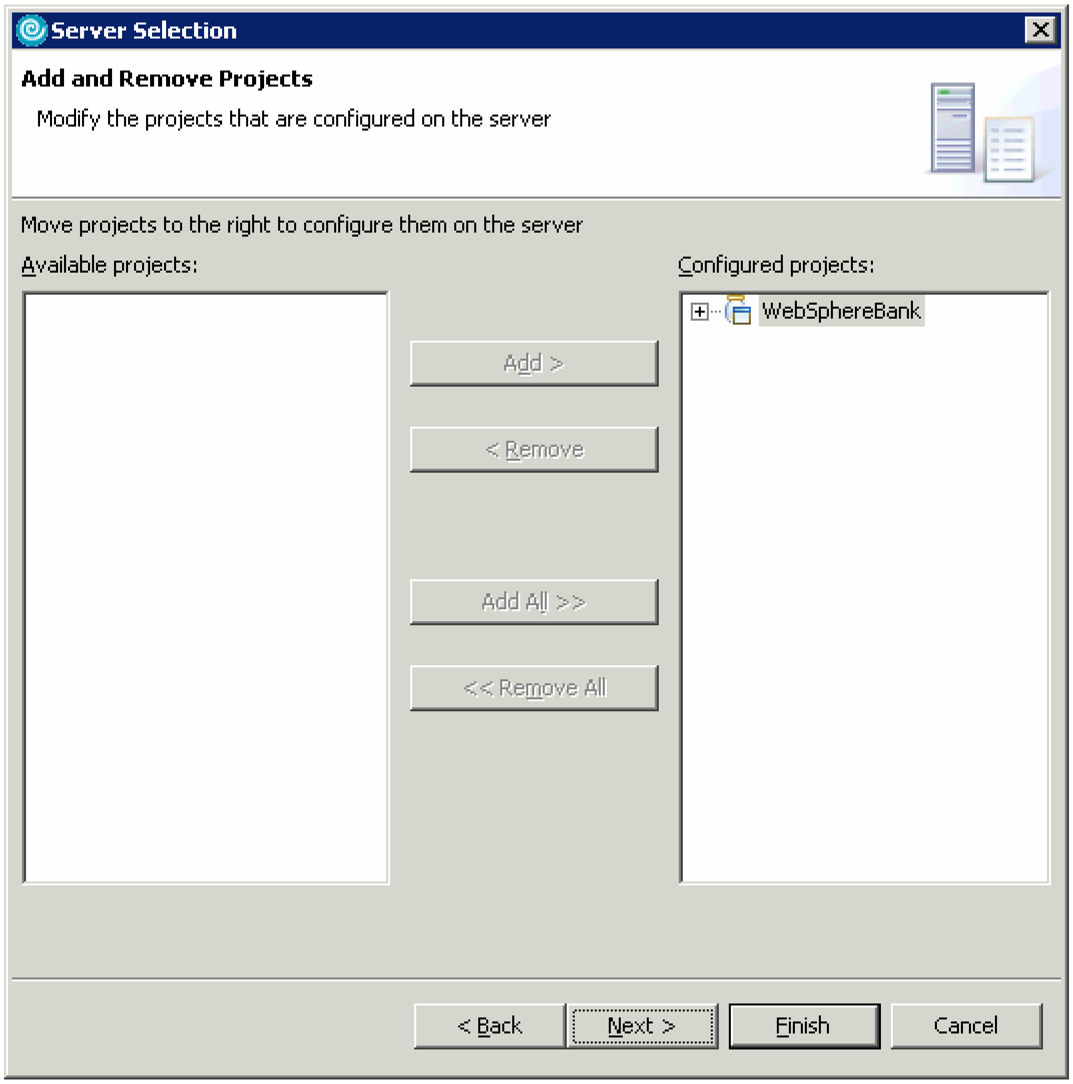


```
import javax.servlet.Requ
import javax.servlet.Serv
import javax.servlet.Serv
import javax.servlet.http
import javax.servlet.http
ort javax.servlet.http

ort com.ibm.websphere.
ort com.ibm.websphere.
ort com.ibm.websphere.

ort java.util.logging.

Utility servlet to cre
/
lic class CreateAccoun

 private ResourceBundl

 // Strings for HTML o
ated
```

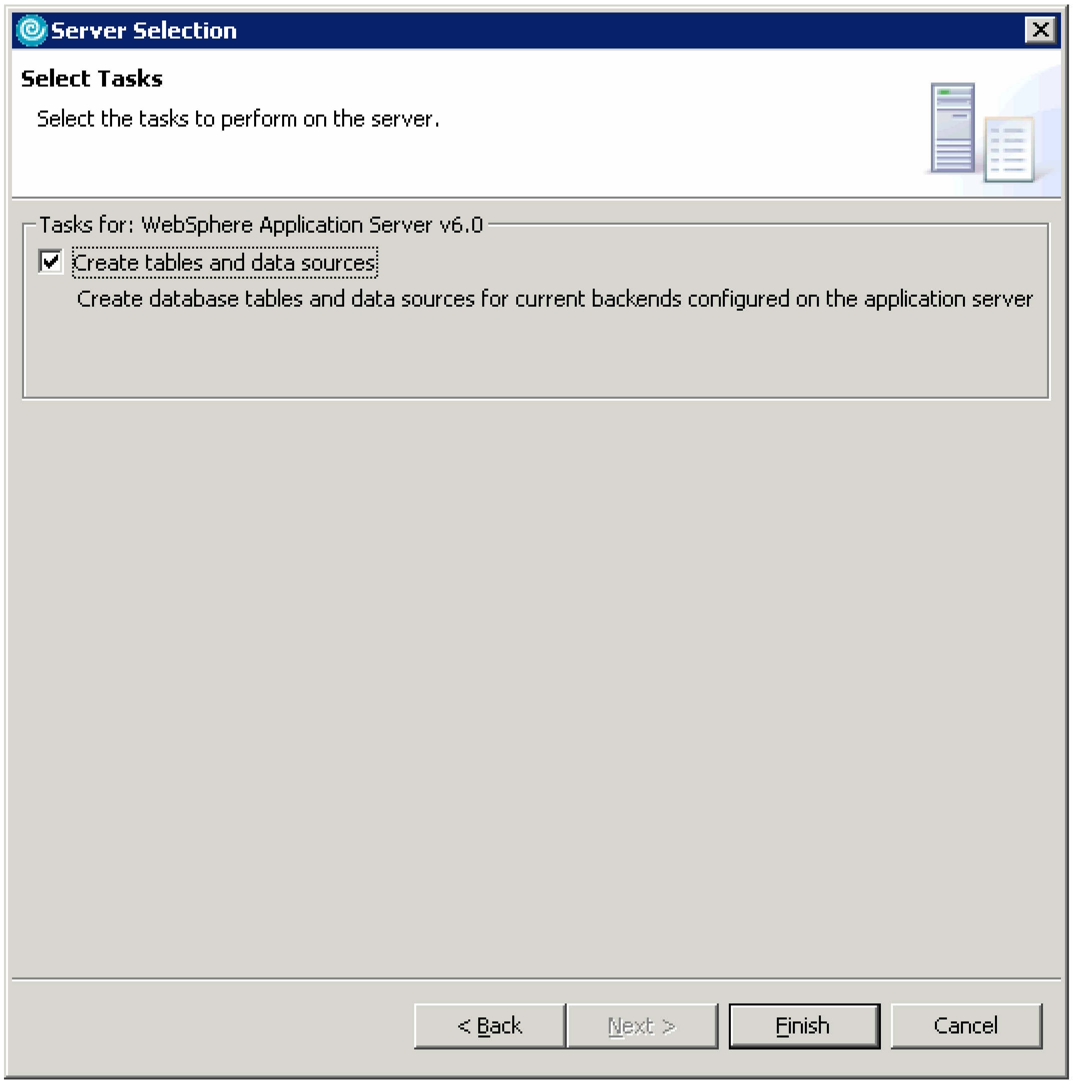
__ b. Select the **Run** > **Run on Server** option to open the Server Selection window.

__ c. On the Define a New Server page of the Server Selection window make sure the **Choose an existing server** option is selected as well as the server WebSphere Application Server v6.0. Then click on the **Next** button.

__ d. On the **Add and Remove Projects** page, make sure that WebSphereBank is added to the Configured projects list on the right side of the window. Then click on the **Next** button.



__ e. On the **Select Tasks** page, click on the Create tables and data sources checkbox. Then click the **Finish** button.



Server Selection

Select Tasks

Select the tasks to perform on the server.

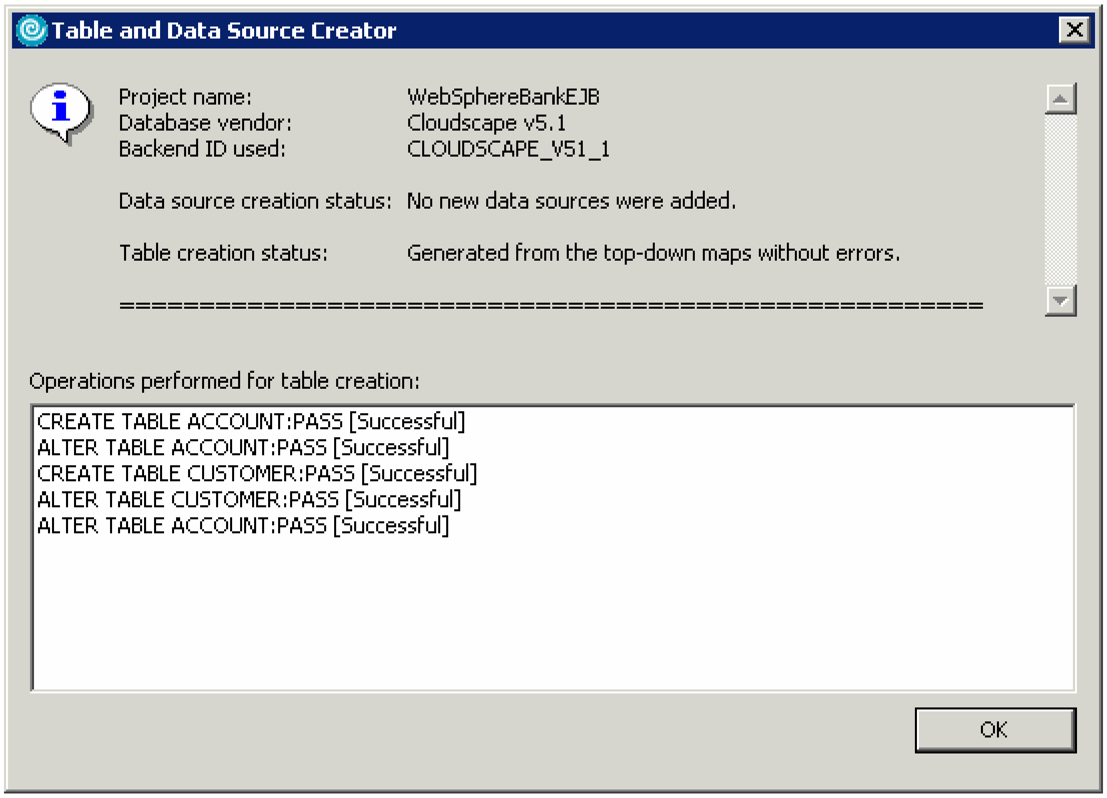
Tasks for: WebSphere Application Server v6.0

☑ Create tables and data sources

Create database tables and data sources for current backends configured on the application server

__ f. Click **Finish**.

__ g. You should see a window like the one below if the database has been setup successfully.



```
Project name:               WebSphereBankEJB
Database vendor:            Cloudscape v5.1
Backend ID used:            CLOUDSCAPE_V51_1

Data source creation status: No new data sources were added.

Table creation status:      Generated from the top-down maps without errors.

=================================================================

Operations performed for table creation:
CREATE TABLE ACCOUNT:PASS [Successful]
ALTER TABLE ACCOUNT:PASS [Successful]
CREATE TABLE CUSTOMER:PASS [Successful]
ALTER TABLE CUSTOMER:PASS [Successful]
ALTER TABLE ACCOUNT:PASS [Successful]
```

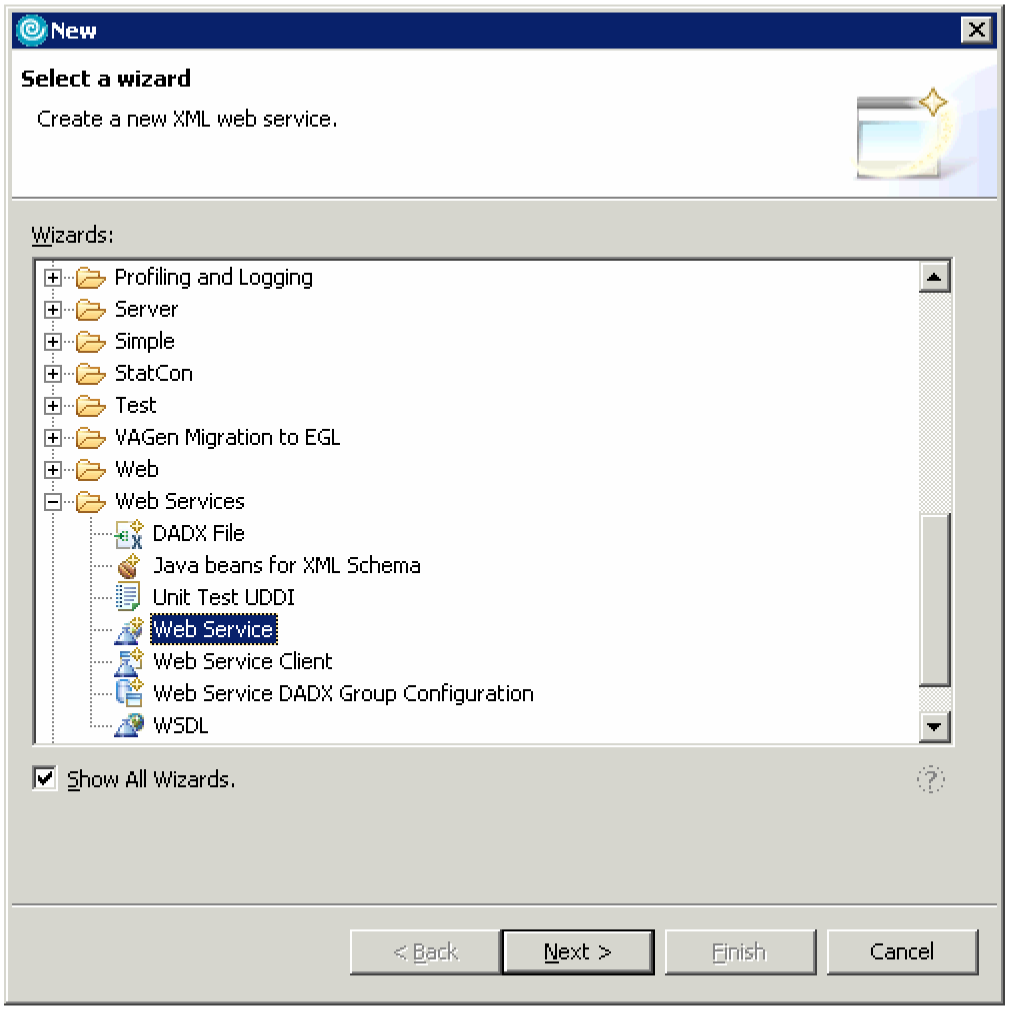
NOTE: After the code has been generated, browse to EJBProjects > WebSphereBankEJB > ejbModule. Notice the classes and the packages which were created. In the com.ibm.websphere.samples.bank.ejb.websphere_deploy package are common classes used to support persisting entity beans running in WebSphere regardless of the data source. The com.ibm.websphere.samples.bank.ejb.websphere_deploy.CLOUDSCAPE_V5_1 classes are specific for persisting entity beans to a Cloudscape database.

__ h. Click **OK**.

Part 3: Create a Web Service provider in WebSphere Bank

____ **1.** Now that you have the workspace and test environment prepared you can create a Web Service.

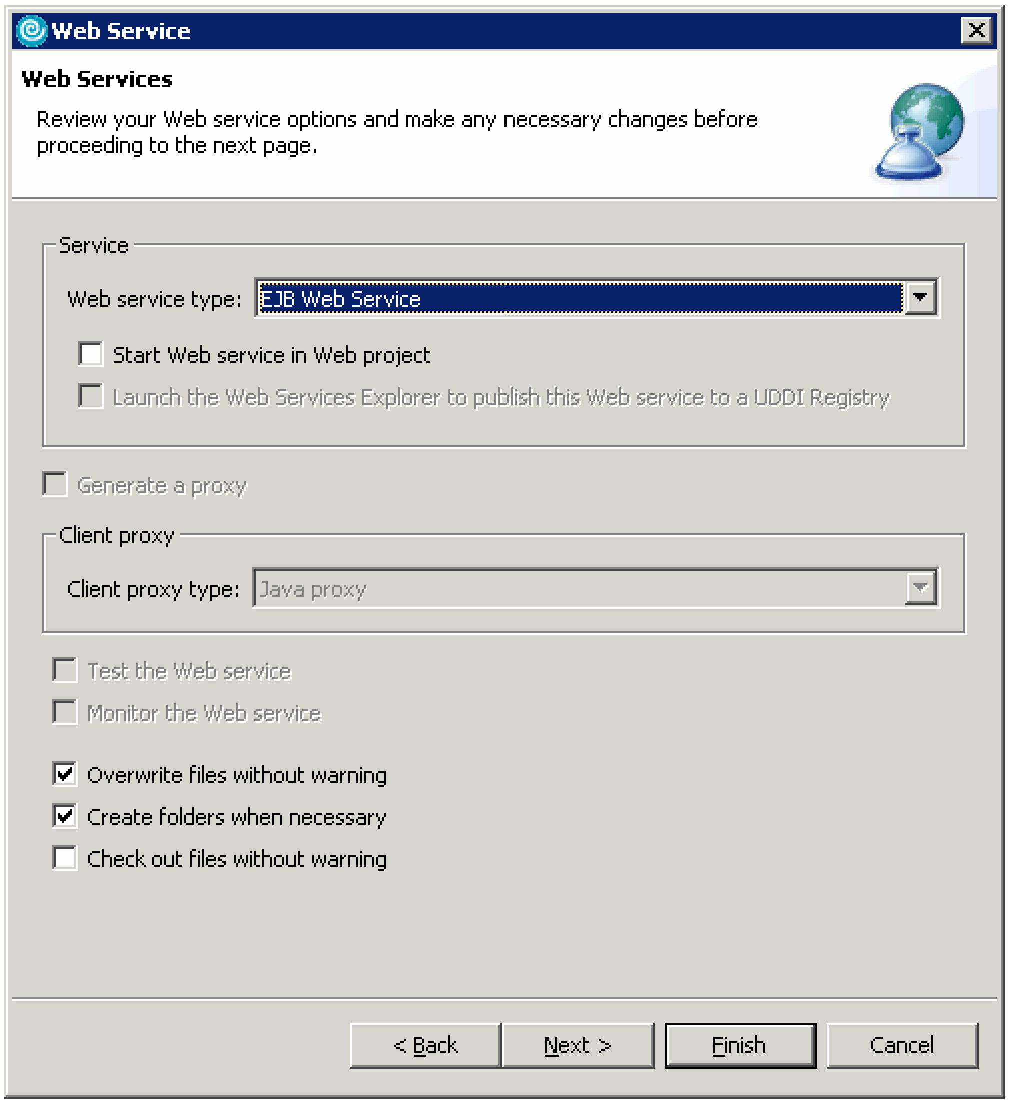
__ **a.** To start creation of a Web Service select **File > New > Other…**. Click checkbox to **Show All Wizards**. Expand **Web Services** and select **Web Service**. Click **Next**.



__ **b.** If this is the first time running Web Services, you will get a **Confirm Enablement** alert window. Check the **checkbox** to always enable capabilities and click **OK** to get to continue the Web Service Wizard.

__ **c.** At the first page of the Web Services wizard, select **EJB Web Service** from the Web Service Type drop down menu.

__ **d.** Uncheck the box to **Start Web service in Web project**. This reduces the time the wizard needs to complete. Click **Next**.



Web Service

Web Services

Review your Web service options and make any necessary changes before proceeding to the next page.

Service

Web service type: EJB Web Service

[ ] Start Web service in Web project

[ ] Launch the Web Services Explorer to publish this Web service to a UDDI Registry

[ ] Generate a proxy

Client proxy

Client proxy type: Java proxy

[ ] Test the Web service

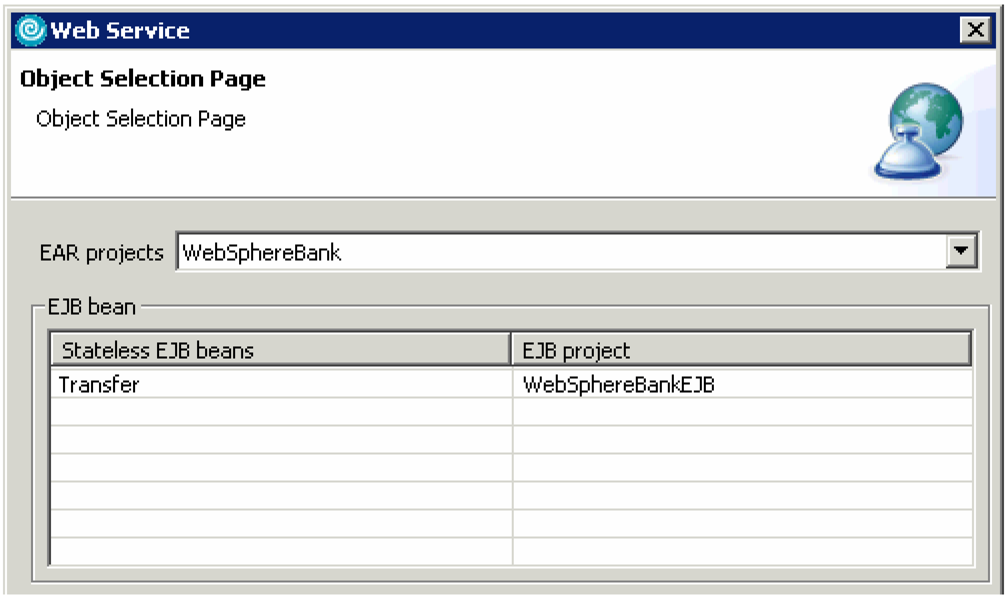
[ ] Monitor the Web service

[✓] Overwrite files without warning

[✓] Create folders when necessary

[ ] Check out files without warning

__ **e.** On the Object Selection Page, select the Transfer Stateless EJB Bean. Click **Next**.



Web Service

Object Selection Page

Object Selection Page

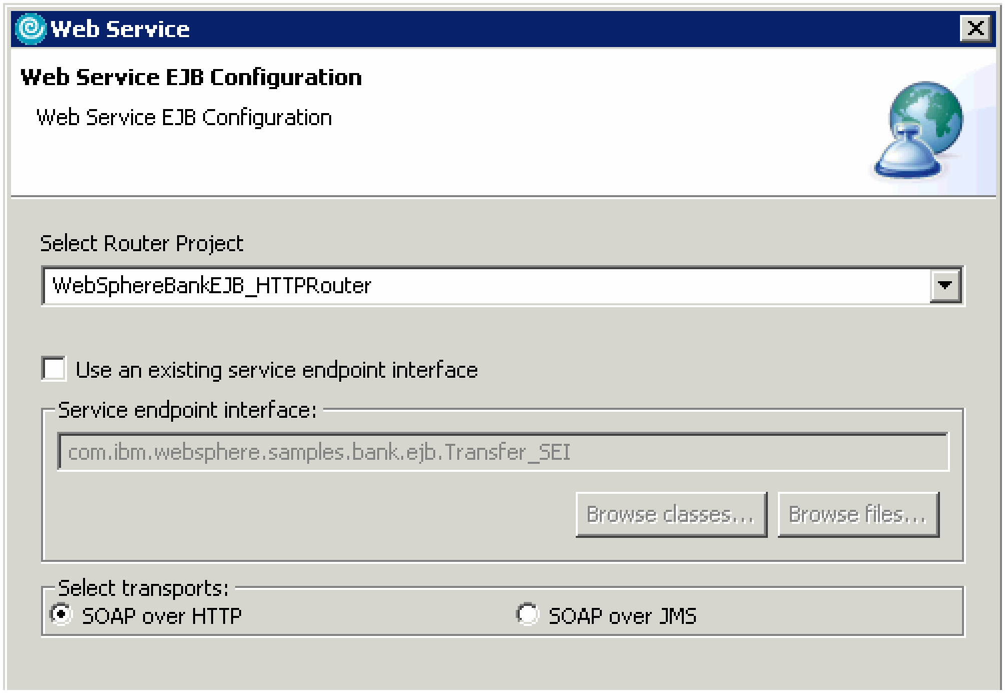
EAR projects: WebSphereBank

EJB bean

Stateless EJB beans	EJB project
Transfer	WebSphereBankEJB

__ **f.** On the Service Deployment Configuration page, accept the defaults and click **Next**.

__ **g.** Under Select Router Project, instead of the router project **WebSphereBankWeb** type in **WebSphereBankEJB_HTTPRouter**. Keep all other defaults values here, especially **SOAP over HTTP** selected for the transport. Click **Next**.



Web Service

Web Service EJB Configuration

Web Service EJB Configuration

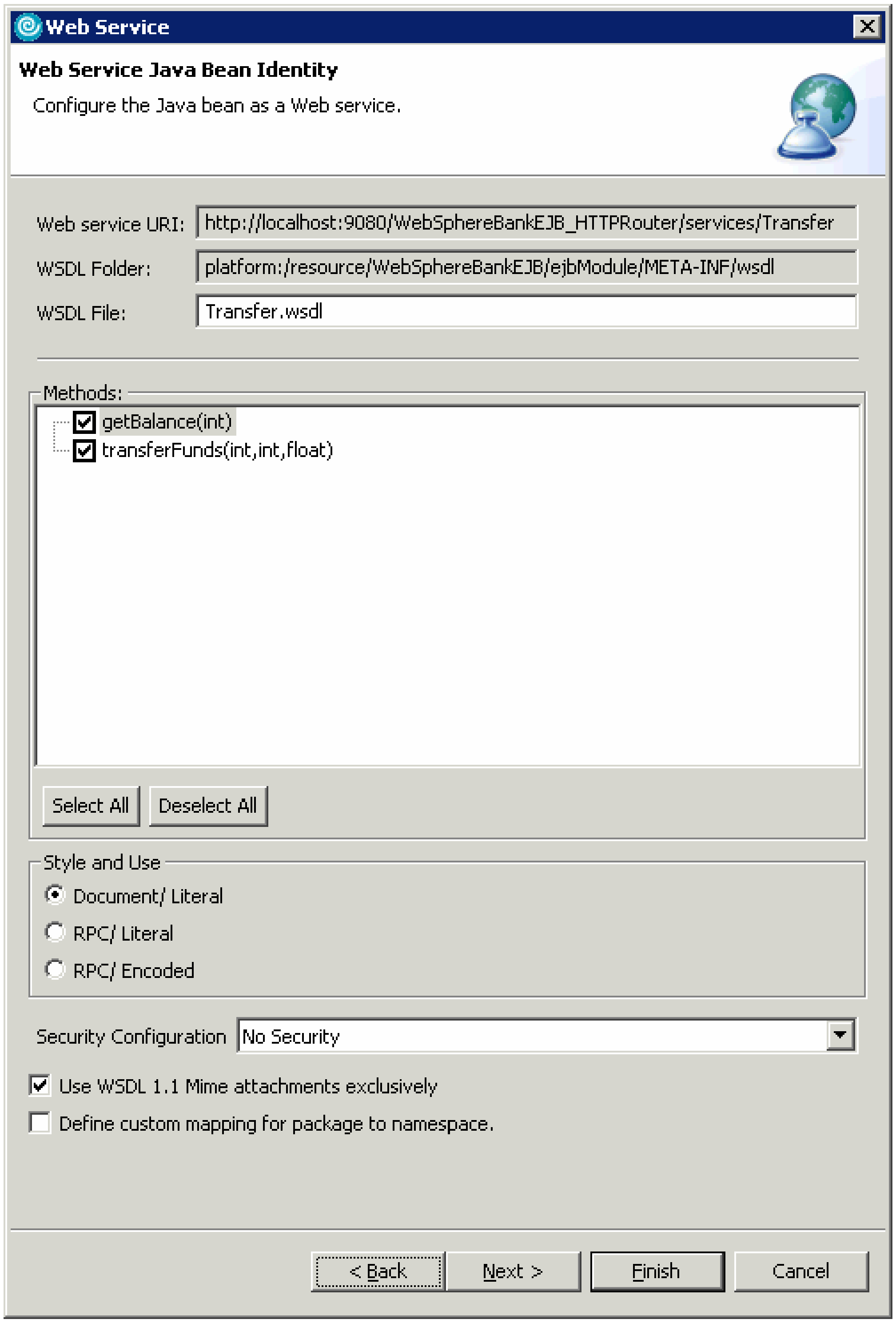
Select Router Project: WebSphereBankEJB_HTTPRouter

☐ Use an existing service endpoint interface

Service endpoint interface: com.ibm.websphere.samples.bank.ejb.Transfer_SEI

Select transports: [✓ SOAP over HTTP] SOAP over JMS

__ **h.** At the Web Service Java Bean Identity page, keep all defaults. Notice where the WSDL folder and WSDL file is located in EJB Project. You will be going to this location later. Make sure **getBalance** and **transferFunds** are checked. Style should be **Document** / **Literal**. This Web Service will have **No Security**. Click **Next.**



Web Service Java Bean Identity

Configure the Java bean as a Web service.

Web service URI: http://localhost:9080/WebSphereBankEJB_HTTPRouter/services/Transfer

WSDL Folder: platform:/resource/WebSphereBankEJB/ejbModule/META-INF/wsdl

WSDL File: Transfer.wsdl

Methods:
- ✓ getBalance(int)
- ✓ transferFunds(int,int,float)

Style and Use
- ● Document/ Literal
- ○ RPC/ Literal
- ○ RPC/ Encoded

Security Configuration: No Security

- ✓ Use WSDL 1.1 Mime attachments exclusively
- ☐ Define custom mapping for package to namespace.

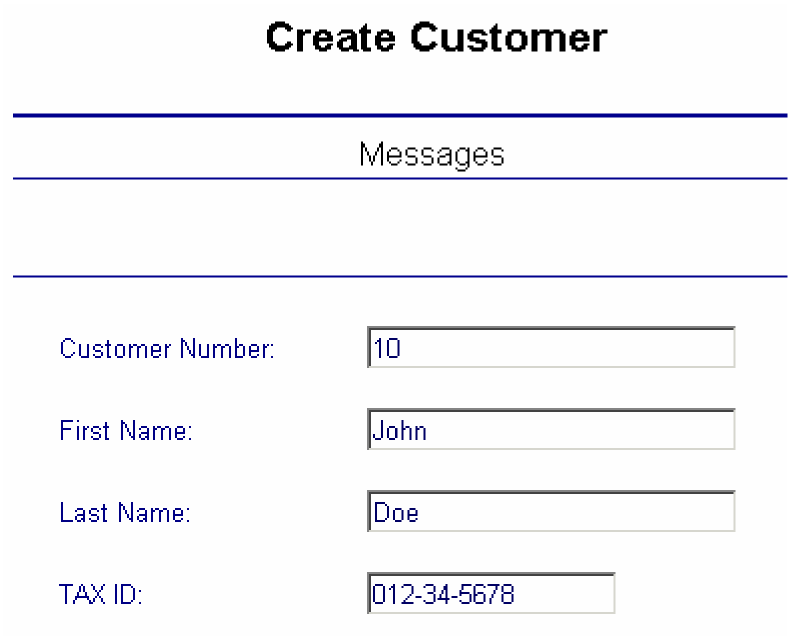
__ **i.** At the last screen of the Web Service wizard, there are UDDI commands. Keep these unchecked and click **Finish.** After you click finish, Rational Application Developer will finalize the building of the projects and necessary files in those projects. You have just created a Web Service using the Rational Application Developer's Web Service wizard.

Part 4: Test Web Service provider with Web Services Explorer

____ 1. Create a customer.

__ a. Select the **Create Customer** link.

__ b. Enter Customer Number, Name and Tax ID. Example below. Select **Create**.



Create Customer

Messages

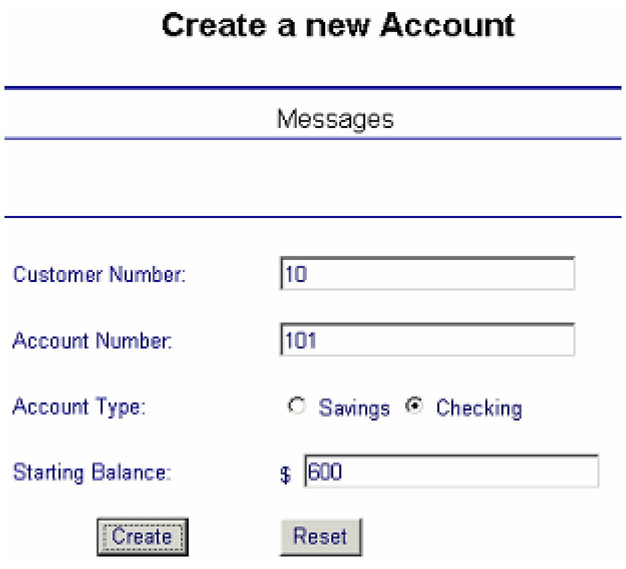
Customer Number:	10
First Name:	John
Last Name:	Doe
TAX ID:	012-34-5678

__ c. You will see details for customer created.

____ 2. Create two accounts for the customer.

__ a. Select the **Create Account** button.

__ b. Enter **101** for the Account Number, select **checking** for account type, and **600** for the starting balance. Select **Create**.



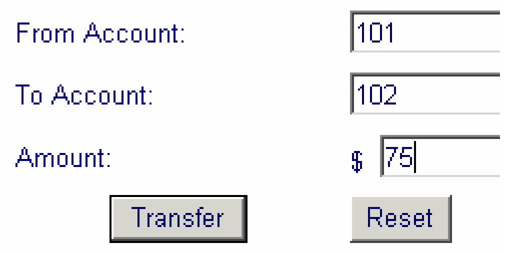
Create a new Account

Messages

Customer Number:	10
Account Number:	101
Account Type:	Savings [✓ Checking]
Starting Balance:	$ 600

__ c. Now create a **savings account** with a number of **102** and a balance of **400**. With two accounts created, you can now transfer funds between these two accounts.

__ d. Select the **Transfer Funds** link. For the From Account enter **101,** for the To Account enter **102**, and for the amount enter **75**.



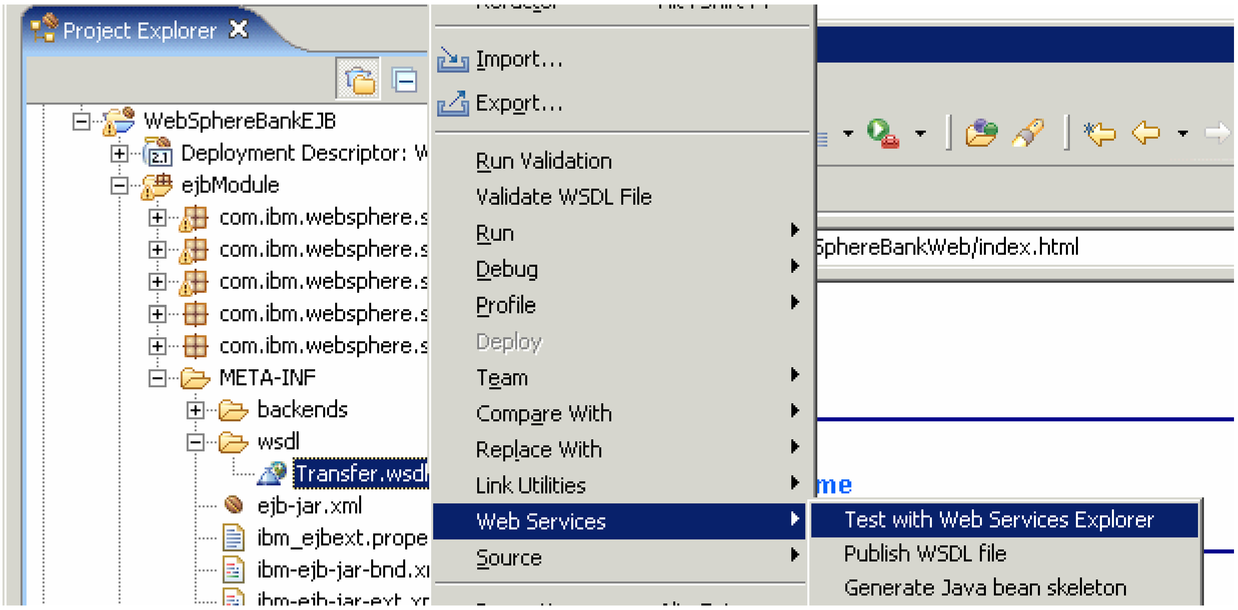
From Account:	101
To Account:	102
Amount:	$ 75

__ e. Select **Transfer**. The success message will be displayed. The updated balance will also be displayed.

____ 3. With the regular Web interface working for the application, you can test the Web Service you created with the Web Services Explorer. The Web Services Explorer only mimics what a Web Service client would do, and therefore, you may get different results between the Web Services Explorer and a real Web Service client.

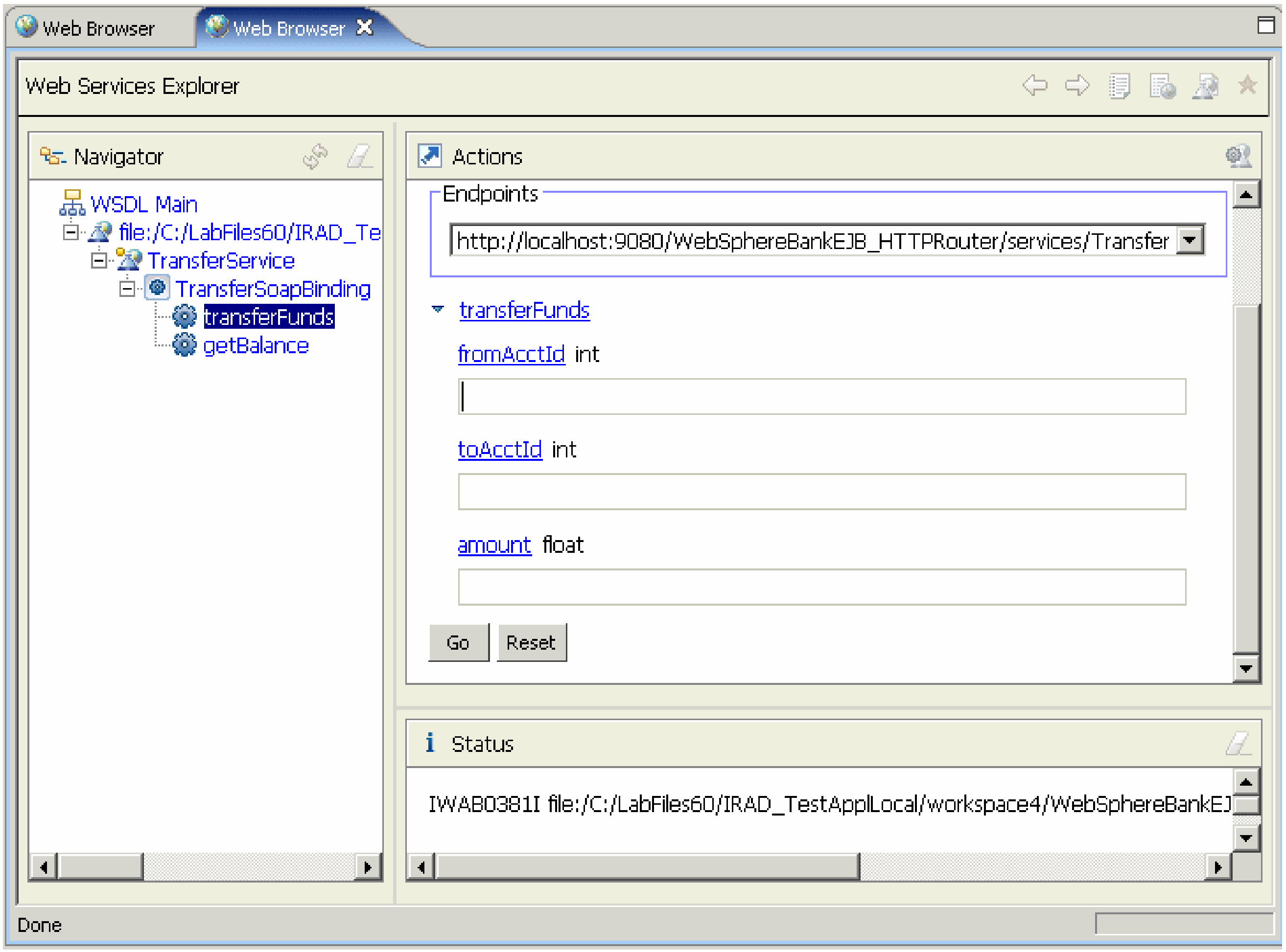
__ a. In the **Project Explorer** view, navigate to **EJB Projects** > **WebSphereBankEJB** > **ejbModule -> META-INF** > **wsdl**.

__ **b.** Right click on **Transfer.wsdl** and select **Web Services > Test with Web Services Explorer**.



__ **c.** This will start the WS Explorer in a browser. You should see the TransferSoapBinding binding in the Navigator section and the operations transferFunds and getBalance. Get the balance of account 101 by clicking **getBalance**.

__ **d.** In the Actions section of the WS Explorer enter **101** for Account ID to access your first account. Click the **Go** button**.**



IWAB0381I file:/C:/LabFiles60/IRAD_TestApplLocal/workspace4/WebSphereBankEJ

__ **e.** In the Status section, you will see the result of getBalance on the account ID you entered. Repeat the steps for the second account balance. Enter **102** for Account ID to access your second account (in the Actions section of the WS Explorer). Click **Go b**utton. You will see the balance for account 102 in the Status section of the WS Explorer.

__ **f.** Now to transfer some funds. Click on **transferFunds** in the Navigator section of the WS Explorer. In the Actions section of the WS Explorer, enter **101** for **fromAcctID**, enter **102** for **toAcctID**, and enter **77** for **amount**. Click the **Go** button.

__ g. You will not be shown the result like the servlet driven web application that we first used to create and enter customer information. However if you go back and click getBalance again and enter the account number in Account ID field and then click Go again, you will see the amounts have changed. At this point, you have successfully created and tested your Web Service with the Web Services Explorer. You now have a valid web service and will need to create a web service client in order to access / use it.

Part 5: Export WebSphereBank with Web Services as an EAR File

Applications destined for a production environment can be deployed directly from Rational Application Developer or can be exported out as an EAR file for a more managed installation. This part of the exercise describes the steps for exporting out the application

____ 1. Click on **File > Export** and select **EAR File** from the list. Click **Next.**

____ 2. Select **WebSphereBank** from the drop down menu for the Enterprise Application project .

____ 3. Browse to **<LAB_FILES>\IRAD_WebServices**\ and enter **WebSphereBankwithWS** for the name of the EAR file. Click **Save**.

____ 4. Check the **Export source files** and **Include project build paths and meta-data files** check boxes and click **Finish**.

Part 6: Restore your server

Skip this Part if you plan on completing the lab exercise **Building a Web Service Client with IBM Rational Application Developer v6.0**.

____ 5. Restore your server configuration. This will return your server configuration to its original state.

__ a. Open a Windows Command Prompt and navigate to `<RAD_HOME>\runtimes\base_v6\bin`.

__ b. Restore the server configuration by issuing the following command:

```
restoreConfig "C:\Program Files\IBM\Backup.zip"
```

What you did in this exercise

This lab exercise, you created a Web Service using IBM Rational Application Developer v6.0 wizards and ran this on a WebSphere Application Server v6.0 test environment.

Solution Instructions

If you are unable to get this lab to work, and would like to see the results, you can follow these steps to import the solution to this lab. You will then be able to browse the solution and view what was generated.

____ 1. Start IBM Rational Application Developer v6.0 with a clean workspace.

____ 2. Import WebSphereBank.ear with a Web Service Provider into the workspace.

__ a. Select **File > Import...**

__ **b.** Select **EAR file** and click **Next.**

__ c. Select **Browse**... and navigate to **<LAB_FILES>\IRAD_WebServices\solution\WebSphereBankwithWS.ear** and click **Open**.

__ d. Click **Finish.**

____ 3. When the import is complete, you will notice several errors in the Problems view. Resolve these errors.

__ a. Right click on one of the errors and select Quick Edit.

__ **b.** A list of available fixes will appear. Select **Add project 'WebSphereBankEJB' to build path of 'WebSphereBankWeb'** and click **OK.** The errors will be resolved.

____ 4. Browse the application and view the generated components or follow the instructions in Part 3 to create the database and test the web service.

This page intentionally left blank