IBM WEBSPHERE 5.0 Skills Transfer - LAB EXERCISE

# **Developing Web Services - Part 1**

### What This Exercise is About

WebSphere Studio Application Developer has tools for working with each part of the Web Services programming model. For the Service Provider, there is a wizard for turning applications into Web Services. For the Service Requester, there is a wizard for creating client applications to access Web Services. Finally, for the Service Registry, there is a Web Services explorer for viewing UDDI public or private registries. This exercise will feature some of the Web Service tools available in WebSphere Studio Application Developer.

### **User Requirement**

User must have IBM DB2 Universal Database (version 7.2) and IBM WebSphere Studio Application Developer Version 5.0.1 installed on a Windows 2000 workstation with Service Pack 3. Internet Explorer version 5.5 is also required. The WebSphere Studio Application Developer should be installed at C:\Program Files\IBM\WebSphere Studio. To complete all parts of the lab, the WebSphere Application Server Version 5.0.1 and WebSphere Network Development Version 5.0.1 packages are required. The lab source files (LabFiles50.zip) must be extracted to the root directory (e.g. C:\). Experience with previous versions of WebSphere Studio Application Developer and the J2EE programming model are also required. Throughout this lab, a userid profile named wsdemo with a password of wsdemo1 is assumed to be created on the system.

NOTE: This lab exercise makes the following assumptions: DB2 is installed at C:\SQLLIB; WebSphere Application Server Version 5.0,1 is installed at: C:\WebSphere\AppServer; WebSphere Network Deployment Version 5.0.1 is installed at:

C:\WebSphere\DeploymentManager; WebSphere Studio Application Developer is installed at C:\Program Files\IBM\WebSphere Studio; and the LabFiles50 is extracted to C:\. Throughout this lab, batch files are used to simplify some selected operations. If the software is installed in different locations other than those specified above, you can modify the batch files, however, extensive testing has not been completed to verify all possible combinations. To accommodate different installed locations of the required products above, a batch file called setupVARs.bat has been created. It is located in \LabFiles50. You may update this file according to the instructions contained therein - specifying the drive letters and/or paths of the installed products. All subsequent batch files used throughout this lab calls the setupVARs batch file first. Effectively, you have a single location to make updates to product installation drives and directories.

# What You Should Be Able to Do

In this lab, the focus is on getting the WebSphere Studio Application Developer workbench environment setup to develop a Web Service. Upon completion of all three parts to this lab exercise, you will have created a Web Service, published it to the Unit Test UDDI Registry and written a sample application to act as a Service consumer.

# **Background Information**

There are three main roles and functions in the Web Services model. The Service Provider provides applications as Web Services and publishes them to the Service Registry. They accept requests, process data, and return the result to Service Requesters. The Service Requester makes these requests (of Service Providers) and receives the result back. The Service Registry is used by both the Service Provider and the Service Requester. The Service Registry lists the different services available by Service Providers and the Service Requester. All of this is done in a loosely-coupled environment as the technical implementation between the different players is based solely on Open Standards (SOAP, XML, WSDL, UDDI), avoiding rigid proprietary implementations. Each role in the Web Services model focuses their efforts on their part of the model, shielded from the other's by the Open Standards.

The following graphic further illustrates the roles and functions in the Web Services model.



### Introduction

In Part One, you will complete some necessary steps for the Lab Setup. This lab makes the assumption that you are beginning with the WebSphere configuration files that were backed up after the WebSphere Application Server was installed. This was done in the Installation lab. In addition, the BankData database is created (if it doesn't exist) and the necessary tables are created.

During Part Two, you will create a server in WebSphere Application Studio Developer that is a Test Server running within the workbench environment.

In Part Three, you install the IBM Unit Test UDDI Registry in the workbench. You will use Cloudscape as the backend database for the Registry.

### **Exercise Instructions**

#### Part One: Lab Setup

- \_\_\_1. Restore the original WebSphere configuration files.
  - \_\_\_\_\_a. At a Command Prompt, **CD \WebSphere\AppServer\bin**.
  - \_\_\_\_b. Type **restoreConfig.bat baseWASnode.zip**. This will recover the contents of several directories and configuration files from the **baseWASnode.zip** file, which was created as part of the Installation lab.
- \_2. The MyBank application (the application used throughout this lab) uses a database called BankData. This database needs to be created. At a Command Prompt, CD \LabFiles50\WebServices\Bankdata. Then type BankData.bat wsdemo wsdemo1 Table.ddl.

**NOTE:** There is a dependency in the loadBankDataBase.bat batch file that the BankData.bat batch file calls. If you do not have DB2 installed at C:\SQLLIB, you should update the setupVARs.bat file according to the instructions specified in the User Requirement section.

- \_\_\_\_a. This batch file drops the ACCOUNT and CUSTOMER tables in a database called BankData, re-creates the ACCOUNT table, and populates the ACCOUNT table with a few records.
- \_\_\_\_b. After the BankData batch file completes, you should see the following message:

BankData database has been created and loaded.

#### Part Two: Create a WebSphere Test Server and Server Configuration with Application Developer

- \_\_\_1. Start WebSphere Studio Application Developer.
  - \_\_\_\_a. Select the menu Start > Programs > IBM WebSphere Studio and select Application Developer 5.0.
  - b. A dialog box will be displayed allowing you to select the location where you would like the workspace directory to be stored. Enter
     C:\LabFiles50\WebServices\workspace for the location and select OK. Application Developer will start with an empty workspace. An empty workspace will leave your existing workspace untouched and help avoid name conflicts between what you may already have in your workspace and what you will be creating in this lab.
- \_2. When WebSphere Application Studio Developer becomes active, you will need to switch to a J2EE perspective. If the J2EE perspective is not already opened, select Window > Open Perspective > Other and select J2EE. Click OK.
- \_\_3. Import the MyBank EAR file.

\_\_\_\_a. Select File > Import... > EAR File.

| elect   |                   |       |        |       |
|---|-------------------|-------|--------|-------|
| (mport an external EAR file into an Enterprise  | e Application pro | iject |        | Ľ     |
| Select an import source:  |                   |       |        |       |
| App Client JAR file   |                   |       |        |       |
| 🤹 EJB JAR file  |                   |       |        |       |
| Existing Project into Workspace   |                   |       |        |       |
| External Plug-ins and Fragments   |                   |       |        |       |
| ftp.FTD   |                   |       |        |       |
| httehttp  |                   |       |        |       |
| http<br>HTTP Recording  |                   |       |        |       |
| 📃 Logging Utilities XML Log File  |                   |       |        |       |
| 🔆 Profiling file  |                   |       |        |       |
| In a start of the start of |                   |       |        |       |
| Server Configuration  |                   |       |        |       |
| egoymptom Database File   |                   |       |        |       |
| wAR file  |                   |       |        |       |
| Keb Services  |                   |       |        |       |
| 🧓 WebSphere Application Server Log File   |                   |       |        |       |
| 🛃 Zip file  |                   |       |        |       |
|   |                   |       |        |       |
|   |                   |       |        |       |
|   | z Baali           | News  | Fields | Grand |

- \_\_\_\_b. Click Next.
- \_\_\_\_\_c. Click **Browse...** to the right of EAR file.
- \_\_\_\_d. Navigate to \LabFiles50\WebServices\startEAR. Select MyBank.ear and click Open.

\_\_\_\_e. Type **MyBank** for the Project Name.

| Interprise Application Import                                    | 8       |
|--|---------|
| Import an Enterprise Application project from the file System.   |         |
| FAR File: C:\LabFiles50\WebServices\startEAR\MvBank.ear          | Browse. |
|  |         |
| where do you want the imported resources to go?                  |         |
| Project name:   MyBank   |         |
| ✓ Use default  |         |
| New project location; c:\LabFiles50\WebServices\workspace\MyBank | Browse  |
| Options  |         |

- \_\_\_\_f. Click Finish.
- \_\_\_4. Create a WebSphere Test Environment Server.
  - \_\_\_\_a. First, you will configure a test server to test our Web Services within WebSphere Studio Application Developer. Select **Window > Open Perspective > Other...** and select **Server**. Click **OK**.
  - \_\_\_\_b. Next select File > New > Server and Server Configuration.

| Eile     | <u>E</u> dit     | <u>N</u> avigate | Se <u>a</u> rch | Project | Profile | <u>R</u> un    | <u>W</u> indow   | Help     |                                 |        |
|----------|------------------|------------------|-----------------|---------|---------|----------------|------------------|----------|---------------------------------|--------|
|          | <u>N</u> ew      |                  |                 |         |         |                |                  | 2 📩      | Project                         |        |
|          | ⊆lose<br>Close / | 41               |                 |         |         | Ctrl+<br>Ctrl+ | -F4<br>-Shift+F4 | <u>e</u> | Server Project                  |        |
| -        | -1               |                  |                 |         |         |                | 21113111         | - 🔝      | Server and Server Configuration |        |
|          | Save             |                  |                 |         |         | Ctrl+          | -5               | Ť        | Server                          |        |
| <b>.</b> | Save (           | <u>1</u> 5       |                 |         |         |                |                  | 1        | Server Configuration            |        |
| 6        | Sav <u>e</u> A   | sill.            |                 |         |         | Ctrl+          | -Shift+S         | Å        | Folder                          |        |
|          | Rever            | <u>i</u>         |                 |         |         |                |                  | _ 🖹      | File                            |        |
|          | Mo <u>v</u> e,   |                  |                 |         |         |                |                  |          |                                 |        |
|          | Renam            | je.,,            |                 |         |         |                |                  | <u> </u> | Other                           | Ctrl+N |

- \_\_\_\_ c. On the Create a new server and server configuration panel, type **TestServer** for the Server name and **TestServer** for the Folder.
- \_\_\_\_d. For the Server type, expand **WebSphere version 5.0** and select **Test Environment**.
- \_\_\_\_e. Click Finish.
- \_\_\_5. Create a Data Source for the Test Environment server.
  - \_\_\_\_a. In the Server Configuration pane (lower left corner), expand **Servers**. Right-click **TestServer** and select **Open**.

| Server Configurati | on                   | • | × |
|--------------------|----------------------|---|---|
| 🕀 🚳 Unused Server  | New                  | - |   |
| · ·                | Open                 |   |   |
|                    | Delete               |   |   |
|                    | Control              | • |   |
|                    | Switch Configuration | → |   |
|                    | Add                  | → |   |
|                    | 👷 Debug on Server    |   |   |
|                    | KI Run on Server     |   |   |
|                    | 👌 Profile on Server  |   |   |

\_\_\_\_b. Select the **Security** tab as shown Below.

Server Configuration | Paths | Environment | Web | Data source | Ports | Variables | Trace Security EJB | J2C | JMS

\_\_\_\_\_c. Click the **Add** button to the right of JAAS Authorization Entries.

| Security Options                                 |                          |                 |
|--|--------------------------|-----------------|
| Scope: localhost/localhost/ser                   | ver1                     |                 |
| <ul> <li>Cell Settings</li> </ul>                |                          |                 |
| Enable and setup security.                       |                          |                 |
| Enable security (Not support<br>Enable security) | rted on Windows 98 and W | indows ME)      |
|  |                          |                 |
| Local OS Authentication:                         |                          |                 |
| Server ID:                                       |                          |                 |
| Server password:                                 |                          |                 |
| Confirmed password:                              |                          |                 |
|  |                          |                 |
| Enforce Java 2 security                          |                          |                 |
|  |                          |                 |
| JAAS Authentication Entries:                     |                          |                 |
| Alias  | User ID                  | Description Add |
|  |                          |                 |
|  |                          | Laultyrr        |
|  |                          | Remove          |

\_\_\_\_d. On the Add JAAS Authentication Entry panel, type **MyBankAlias** for the Alias name, type **wsdemo** for the Userid, type **wsdemo1** for the Password, and any text for the description.

| 🕀 Add JAA        | 5 Authentication Entry 🛛 🗙 |
|------------------|----------------------------|
| <u>A</u> lias:   | MyBankAlias                |
| <u>U</u> ser ID: | wsdemo                     |
| Password:        | *****                      |
| Description:     | The description.           |
|                  | OK Cancel                  |

\_e. Click OK.

\_\_\_\_f. Select the **Data source** tab as shown below.

|  | S  |  |                                      |
|--|--|--|--------------------------------------|
| ope: localhost/                                      | /localhost/server1   |  |                                      |
| Node Settings  |  |  |                                      |
| eate and manage                                      | e data sources.  |  |                                      |
| JDBC provider list                                   | t:   |  |                                      |
| Name   | Implementation class na  | me                                     | Add                                  |
|  |  |  | Edit                                 |
|  |  |  | Remove                               |
|  |  |  |                                      |
| Data source defir                                    | ned in the JDBC provider selected -  | above:                                 |                                      |
| Data source defir<br>Name                            | ned in the JDBC provider selected -<br>JNDI Name                                       | above:                                 | Add                                  |
| Data source defii<br>Name                            | ned in the JDBC provider selected . JNDI Name  | above:<br>Type                         | Add<br>Edit                          |
| Data source defii<br>Name                            | ned in the JDBC provider selected -  | above:<br>Type                         | Add<br>Edit<br>Remove                |
| Data source defii<br>Name                            | ned in the JDBC provider selected -  | above:<br>Type                         | Add<br>Edit<br>Remove                |
| Data source defii<br>Name                            | ned in the JDBC provider selected<br>JNDI Name   | above:<br>Type                         | Add<br>Edit<br>Remove                |
| Data source defi<br>Name<br>Resource propert<br>Name | ned in the JDBC provider selected<br>JNDI Name<br>ties defined in the data source sele | above:<br>Type<br>scted above:<br>Type | Add<br>Edit<br>Remove                |
| Data source defi<br>Name<br>Resource propert<br>Name | ned in the JDBC provider selected<br>JNDI Name   | above:<br>Type                         | Add<br>Edit<br>Remove<br>Add<br>Edit |

#### \_\_\_\_g. Scroll down until you see Server Settings.

# \_\_\_\_h. Click the Add button to the right of JDBC provider list.

| reate and manage data so | urces.   |        |
|--------------------------|--|--------|
| JDBC provider list:      |  |        |
| Name                     | Implementation class name                      | Add    |
| Cloudscape JDBC          | com.ibm.db2j.jdbc.DB2jConnectionPoolDataSource | Edit   |
| 🗓 Default DB2 JDBC       | COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource   | Luit   |
|                          |  | Remove |

\_\_\_\_\_i. On the Create a JDBC Provider panel, select **IBM DB2** for the Database type and **DB2 JDBC Provider** for the JDBC provider type.

| Create a JDBC Prov                        | vider   |
|---|---|
| Create a JDBC Pro<br>Select the type of J | <b>vider</b><br>DBC provider to create.   |
| <u>D</u> atabase type:                    | User-defined<br>IBM DB2<br>Cloudscape<br>Informix<br>Sybase<br>Create   |
| JDBC <u>p</u> rovider type:               | DB2 JDBC Provider     DB2 JDBC Provider     DB2 JDBC Provider (XA)     DB2 for OS/400 JDBC Driver (Native)     DB2 for OS/400 JDBC Driver (Native-XA)     DB2 for OS/400 Pre-VSR2 JDBC Driver (Native)     DB2 for OS/400 Pre-VSR2 JDBC Driver (Native) |
| Description:                              | DB2 JDBC2-compliant Provider  |

\_\_\_j. Click Next.

\_\_\_\_k. On the second panel (properties of the JDBC provider), type MyBank.JDBC.Provider for the Name. Accept the defaults for the remaining items.

| reate a JDBC Provider          |  | 0                  |
|--------------------------------|--|--------------------|
| Enter the properties of the JE | BC provider.                                 |                    |
| N                              | MuRaph IDPC Described                        |                    |
| Na <u>m</u> e:                 |  |                    |
| Description:                   | DB2 JDBC Provider                            |                    |
| Implementation class name:     | COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource |                    |
| Class <u>p</u> ath:            | 5 \${DB2_JDBC_DRIVER_PATH}/db2java.zip       | Add External JARs. |
|                                |  | Add Path           |
|                                |  | Remove             |
|                                |  |                    |
| <u>N</u> ative path:           |  | Add External JARs. |
|                                |  | Add Path           |
|                                |  | Remove             |

#### \_\_ I. Click Finish.

\_\_\_\_m. Click **Add** to the right of the Data source defined in the JDBC provider selected above.

| eate and manage data so            | urces.                                       |                           |              |
|------------------------------------|--|---------------------------|--------------|
| IDBC provider list:                |  |                           |              |
| Name                               | Implementation class name                    | Implementation class name |              |
| Cloudscape JDBC                    | com.ibm.db2j.jdbc.DB2jConnectionPo           | olDataSource              | Edit         |
| Default DB2 JDBC                   | COM.ibm.db2.jdbc.DB2ConnectionPoo            | olDataSource              |              |
| WyBank.JDBC.Pro                    | COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource |                           | Remove       |
|                                    | · · · · · · · · · · · · · · · · · · ·        |                           |              |
| Data source defined in the         | e JDBC provider selected above:              |                           |              |
| Data source defined in the<br>Name | e JDBC provider selected above:<br>JNDI Name | Туре                      | Add          |
| Data source defined in the<br>Name | e JDBC provider selected above:<br>JNDI Name | Туре (                    | Add<br>Edit, |

- \_\_\_\_n. Select Version 5.0 data source. Click Next.
- \_\_\_\_\_o. On the Modify Data Source panel, type MyBankDS for the Name. Type jdbc/MyBank for the JNDI Name. Ensure the Data source help class name is com.ibm.websphere.rsadapter.DB2DataStoreHelper. Select MyBankAlias for the Component-managed authentication alias. Ensure the checkbox to Use this data source in container managed persistence (CMP) is selected.

| Modify Data Source                      |   |
|---|---|
| Modify Data Source                      |   |
| Edit the settings of the data source.   |   |
|   |   |
|   |   |
| Name:                                   | MyBankDS  |
| JNDI name:                              | jdbc/MyBank                                       |
| Description:                            | New JDBC Datasource                               |
| ⊆ategory:                               |   |
| Statement cache size:                   | 10  |
| Data source helper class name:          | com.ibm.websphere.rsadapter.DB2DataStoreHelper    |
| Connection timeout:                     | 1800  |
| Maximum connections:                    | 10  |
| Minimum connections:                    | 1   |
| <u>R</u> eap time:                      | 180   |
| Unused timeout:                         | 1800  |
| Aged timeout:                           | 0   |
| Purge policy:                           | EntirePool  |
| Component-managed authentication alia   | MyBankAlias                                       |
| Container-managed authentication aljas: |   |
| Jse this data source in container manag | ged persistence (CMP)                             |
| $\mathbf{\overline{\mathbf{C}}}$        |   |
|   |   |
|   |   |
|   | < <u>Back</u> <u>N</u> ext > <u>Finish</u> Cancel |

### \_\_\_\_p. Click Next.

\_\_\_\_q. On the Modify Resource Properties panel, select databaseName and type **BankData** for the Value.

| odify Resource Properties              |   |                         |
|--|---|-------------------------|
| dit the resource properties for this c | lata source.  | <u> </u>                |
| Resource Properties:                   |   |                         |
| Name                                   | Description   |                         |
| 📑 databaseName                         | This is a required property. The database name. Fo  | r example, enter sa     |
| 🔒 description                          | The description of this datasource.                 |                         |
| 🔚 portNumber                           | The TCP/IP port number where the jdbc Provider re   | sides.                  |
| 🔚 connectionAttribute                  | The connection attributes. Refer to the DB2 referen | nce for the list of co  |
| loainTimeout                           | The maximum time to attempt to connect a databas    | e. If this value is no. |
| Name: databaseName                     |   |                         |
| Type: java.lang.String                 |   |                         |
| Required: Yes                          |   |                         |
| Value: BankData                        |   |                         |
|  |   |                         |

### \_\_\_ r. Click Finish.

s. The TestServer editor window should now look similar to the following graphic.

| ate and manage data :                          | sources.  |  |        |
|--|---|--|--------|
| DBC provider list:                             |   |  |        |
| Name   | Implementation class name                             |  | Add    |
| 🗓 Cloudscape JDBC                              | com.ibm.db2j.jdbc.DB2jConnectionPoolDataSource        |  | Edit   |
| 🗓 Default DB2 JDBC                             | COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource          |  | Eult   |
| 👤 MyBank. JDBC. Pro                            | . COM.ibm.db2.jdbc.DB2Conn                            | COM.ibm.db2.jdbc.DB2ConnectionPoolDataSource |        |
|  | JNDI Name   | Iype   | iAdd   |
| Name   | JNDI Name   | Туре   | Add    |
| 🚺 MyBankDS                                     | jdbc/MyBank   | V5   | Edit   |
|  |   |  |        |
|  |   |  | Demove |
|  |   |  | Remove |
|  |   |  | Remove |
| esource properties def                         | ined in the data source selected                      | above:                                       | Remove |
| esource properties def                         | ined in the data source selected                      | above:                                       | Remove |
| esource properties def<br>Name<br>databaseName | ined in the data source selected<br>Value<br>BankData | above: Type java.lang.String                 | Add    |

- \_\_\_\_\_t. Save the Server Configuration, by typing **Ctrl+S**.
- \_\_\_\_u. Next, **Close** the Server Configuration by clicking on the X on the right.

| TestServer | х |
|------------|---|
|------------|---|

\_\_6. Add the MyBank application to the Test Environment Server Configuration.

\_\_\_\_a. In the Server Configuration panel, **expand Servers**. Right-click TestServer and select **Add**, then choose **MyBank**.

| 🛞 Server Configuration | ٦   |            |
|------------------------|---|------------|
| Servers                |   |            |
| 🔊 Unused Server C      | New   | •          |
| -                      | Open  |            |
|                        | Delete  |            |
|                        | Control   | •          |
|                        | Switch Configuration  | →          |
|                        | Add   | 🕑 👬 MyBank |
|                        |   |            |
|                        | 🐙 Debug on Server   |            |
|                        | ∦] Debug on Server<br>≰] Run on Server  |            |
|                        | ※] Debug on Server<br>홋리 Run on Server<br>장] Profile on Server  |            |
|                        | <ul> <li>֎ Debug on Server</li> <li>֎ Run on Server</li> <li>֎ Profile on Server</li> <li>Tgam</li> </ul>   | •          |
|                        | <ul> <li>Debug on Server</li> <li>Run on Server</li> <li>Profile on Server</li> <li>Tgam</li> <li>Compare With</li> </ul>   | Þ          |
|                        | <ul> <li>Debug on Server</li> <li>Run on Server</li> <li>Profile on Server</li> <li>Tgam</li> <li>Compare With</li> <li>Replace With</li> </ul>   | •          |
|                        | <ul> <li>Debug on Server</li> <li>Run on Server</li> <li>Profile on Server</li> <li>Tgam</li> <li>Compare With</li> <li>Replace With</li> <li>Link Utilities</li> </ul>                       | •          |
|                        | <ul> <li>Debug on Server</li> <li>Run on Server</li> <li>Profile on Server</li> <li>Tgam</li> <li>Compare With</li> <li>Replace With</li> <li>Link Utilities</li> <li>Web Services</li> </ul> | )          |

#### Part Three: Install the IBM Unit Test UDDI Registry within Application Developer

- \_\_1. Using a wizard, you can easily install the IBM Unit Test UDDI Registry.
  - \_\_\_\_a. Select the J2EE Perspective.
  - \_\_\_\_b. Select **File > New > Other...**. Then select the **Web Services** wizard in the left panel and **Unit Test UDDI** in the right Panel.

| New<br>Select<br>Wizard for deploying the unit test UDDI registry  | ***  |
|--|--|
| Component Test     Data     Data     EJB     J2EE     Java     Plug-in Development     Remote File Transfer     Server     Simple     Symptom Database     Web     Web     Web Services     XML     Examples | Web Service Client<br>Web Service<br>Web Service DADX Group Configuration<br>DADX<br>Unit Test UDDI<br>Java beans for XML Schema |

- \_\_\_\_ c. Click Next.
- d. On the Unit Test UDDI Registry Configuration panel, ensure the Private UDDI Registry For WAS v5 with Cloudscape is selected for the Private UDDI Registry Type.



- \_\_\_\_e. Click Next.
- \_\_\_\_f. Select **Use an existing server** and ensure the Server name reads **TestServer**. **Unselect** Launch the Web Services Explorer.

| Unit Test UDDI   |   |
|--|---|
| <b>Unit Test UDDI Registry with Cloudscape Configuration</b><br>Configure the Unit Test UDDI Registry with Cloudscape. | P |
| <ul> <li>Create a new server for unit test UDDI registry</li> <li>Use an existing server</li> </ul>                    |   |
| Server name: TestServer  | • |

- \_\_\_g. Click Finish.
- \_\_\_h. If prompted about Modified Context Roots, select **Yes**.

The Unit Test UDDI Registry will be installed and deployed -- it is a J2EE application. This entire operation will take some time. The UDDI database is being created and loaded in Cloudscape.

\_2. The IBM Unit Test UDDI Registry installation launches the WebSphere Test Server as part of its installation. This server should be stopped. Depending on how quickly you move from the previous step to this step, the TestServer may still be coming up. Regardless, you can stop the server. You <u>may</u> see a message like the following graphic, select OK to terminate.

| 🕀 Terminate Server  |           | ×      |  |
|---|-----------|--------|--|
| Server TestServer is not responding. Do you want to terminate this server? Click<br>OK to terminate the server or click Cancel to continue waiting. |           |        |  |
|   | <u>ОК</u> | Cancel |  |

\_\_\_\_\_a. Switch to the Server Perspective. Select TestServer from the Server tab. Right click on TestServer and select Stop. This will stop the WebSphere Test Environment server.

# What you did in this exercise

In this lab exercise, you configured a WebSphere Test Server within the WebSphere Studio Application Developer and configured a Data Source to interact with a DB2 database. You also installed the Unit Test UDDI Registry in the workbench environment using Cloudscape as the backend database. There was no configuration of Cloudscape necessary.