

IBM RATIONAL APPLICATION DEVELOPER 6.0 – LAB EXERCISE

# JavaServer Faces Client Components

What this exercise is about .....	1
Lab Requirements.....	1
What you should be able to do .....	1
Introduction .....	2
Exercise Instructions .....	2
Part 1: Setup Development Environment.....	3
Part 2: Create Supplier Administration Application.....	4
Part 3: Test Supplier Administration Application.....	19
Part 4: Restore the Server Configuration.....	22
What you did in this exercise .....	23
Solution Instructions.....	24

## What this exercise is about

The objective of this lab is to provide you with an understanding of several of the more common Faces client components. This lab primarily focuses on providing the details necessary to properly configure and use these components. Specifically, this lab will showcase the Data Grid, Tree, and Graph View components.

## Lab Requirements

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

- IBM Rational Application Developer v6.0 with embedded WebSphere Application Server v6.0 Test Environment 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 application that uses Faces client components
- Configure several common Faces client components
- Use Faces client components with a relational record list (SDO technology)

- Develop code to handle updates to client component data
- Create and configure a test environment for running a Faces application

## Introduction

Most people would agree that the current state of Web UI technology is in many ways a step backward from more traditional thick client applications in terms of usability. One of the most noticeable differences is that Web UIs are typically not as responsive or interactive. The reason for this comes from that fact that each user interaction typically results in a full page refresh. This situation gives rise to a performance gap that is undoubtedly perceived by the user. Additionally, from both the user and developer points of view, the number of UI components that can be used to develop web UI is generally lacking when compared to thick clients.

In response to these types of usability issues, IBM Rational Application Developer v6.0 supports a client-side framework that is aimed at improving the responsiveness and interactability of web-based UIs. The primary idea behind this technology is to create web pages that contain all of the data necessary for the client to work with, and only go back to the server when absolutely necessary (for example, on updates, refresh views, etc). To make this all work a Model-View-Controller (MVC) program is created right inside the browser page that is built upon an advanced usage of Javascript. Each page contains a working set of data along with all of the necessary UI controls right in the browser page. This is different from traditional web pages because rather than embedding the data as text in the html tags, the data is sent back to the browser in a structured format such that it can be reused by the various components on the client side.

In order to make this framework more accessible to a broader range of developers, this technology has been designed to provide JSF support to enable rapid application development. This lab will introduce the basic process required to create and test a web application that supports Faces client components. Specifically, you will build an administrative application that will be used to manage the supplier component of the Plants By WebSphere (PBW) sample. This application has been designed to allow supplier administrators the ability to view information and statistics on all orders for each customer, in addition to the ability to create, update, and delete customer records in the supplier database. This application will make use of the Data Grid, Tree View, and Graph client components.

## 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
<LAB_NAME>	IRAD_FacesClientComponents	

## Part 1: Setup Development Environment

In this part of the lab you will run a script that will create the Supplier database you will be accessing throughout this exercise. After setting up the database you will open Application Developer with a blank workspace.

- \_\_\_ 1. Create the supplier database. This database holds information to be displayed with the Faces client components.
  - \_\_\_ a. Open a command window and navigate to: <LAB\_FILES>\IRAD\_FacesClientComponents\  
\_\_\_ b. Enter the following command: **setupSupplierDB.bat**
- \_\_\_ 2. Start IBM Rational Application Developer v6.0 with a new workspace.
  - \_\_\_ a. Select **Start > Programs > IBM Rational > IBM Rational Application Developer v6.0 > Rational Application Developer.**
  - \_\_\_ b. When prompted enter <LAB\_FILES>\IRAD\_FacesClientComponents\workspace for your workspace.
  - \_\_\_ c. Click **OK.**
  - \_\_\_ d. When IBM Rational Application Developer v6.0 opens, close the **Welcome page**

---

**NOTE:** If the Auto Launch Configuration Change Alert window appears click the **Yes** button to change the auto launch eclipse instance to use when opening IBM Rational Software Development Platform in the future.

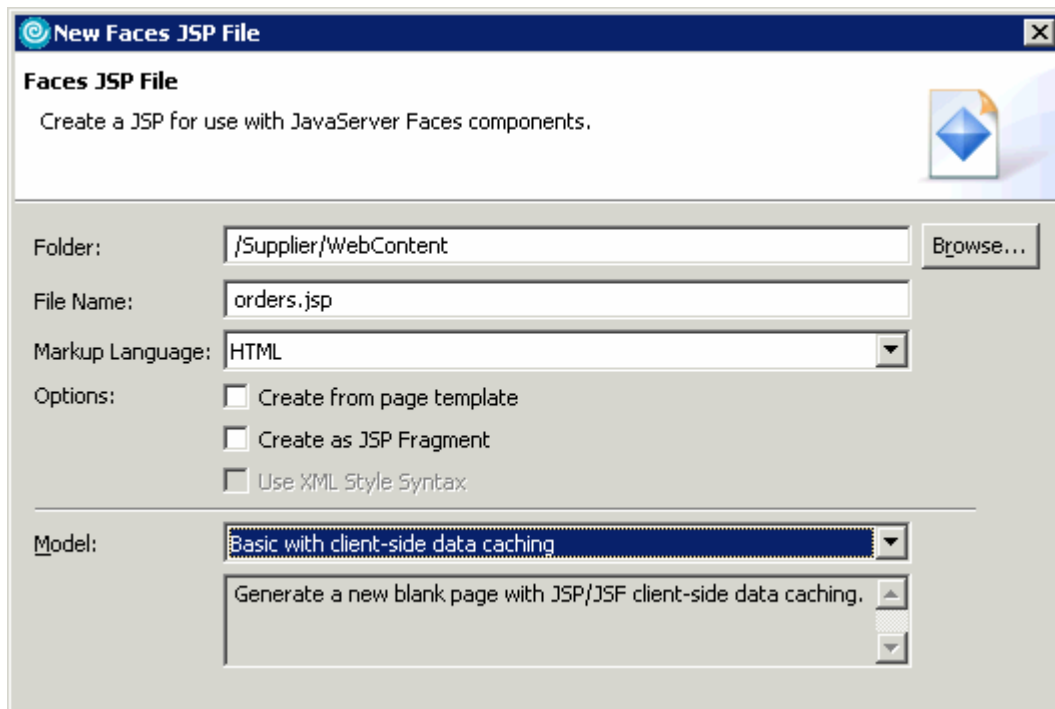
---

- \_\_\_ 3. Create a new Dynamic Web Project
  - \_\_\_ a. Select **File > New > Dynamic Web Project.**
  - \_\_\_ b. Enter **Supplier** for the Name.
  - \_\_\_ c. Click the **Show Advanced** button and **uncheck** the check box for **Add support for annotated Java classes** if it is not already unchecked.
  - \_\_\_ d. Click **Finish.**
  - \_\_\_ e. When prompted to switch to the Web perspective, click **Yes.**

## Part 2: Create Supplier Administration Application

In this part of the exercise you will begin building the Supplier administration application. Specifically, in this section you will be building and configuring the data and components necessary to view the customer and order information stored in the Supplier database. The instructions for doing this take a bottom-up approach by first defining the appropriate page data (in this case a relational record list) and then dropping specific Faces client components onto the page and binding these components to the page data.

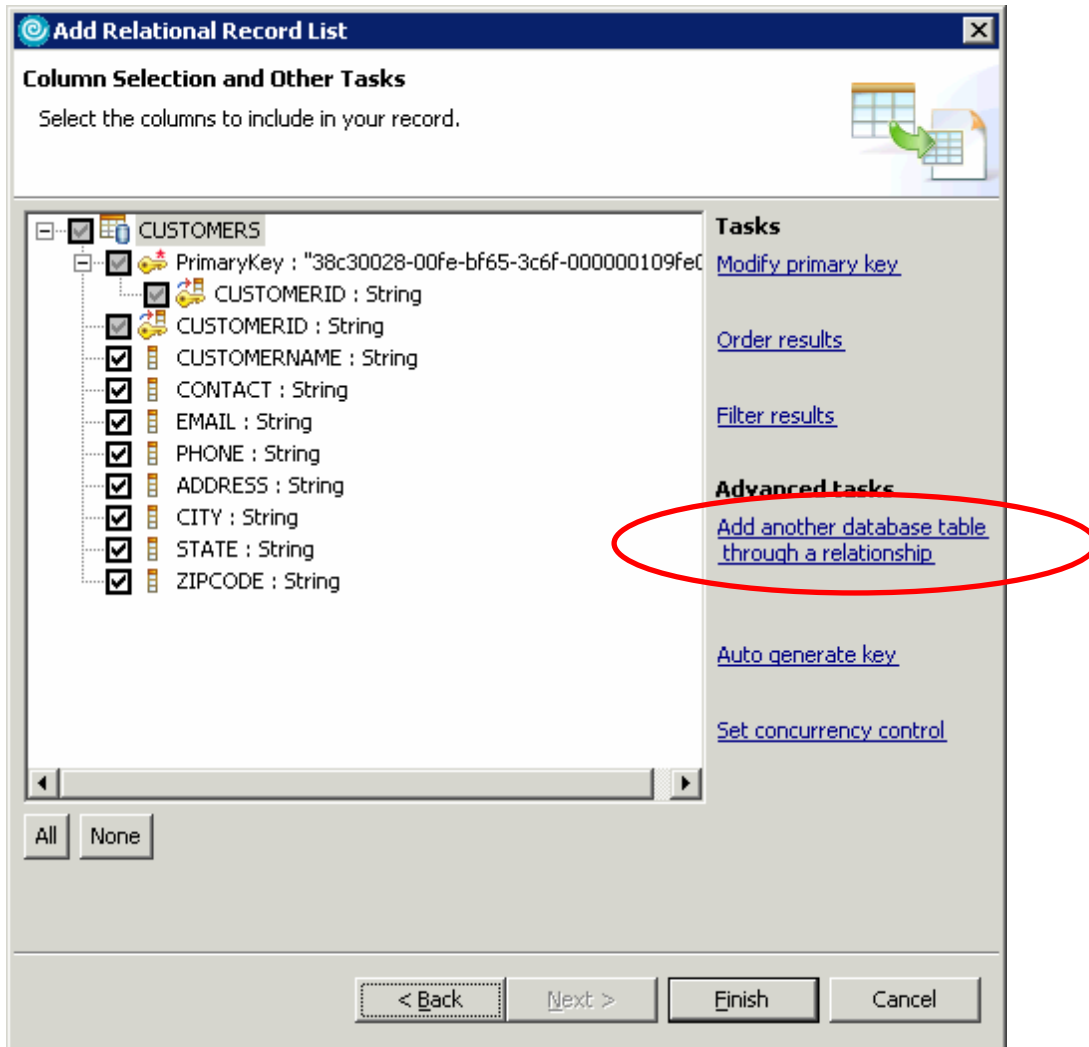
- \_\_\_ 1. Create a new Faces JSP page.
  - \_\_\_ a. In the Project Explorer view, right click on the **Supplier** project, found under Dynamic Web Projects, and select **New > Faces JSP File**.
  - \_\_\_ b. Enter **orders.jsp** for the File Name.
  - \_\_\_ c. Select **Basic with client-side data caching** from the Model drop down list.



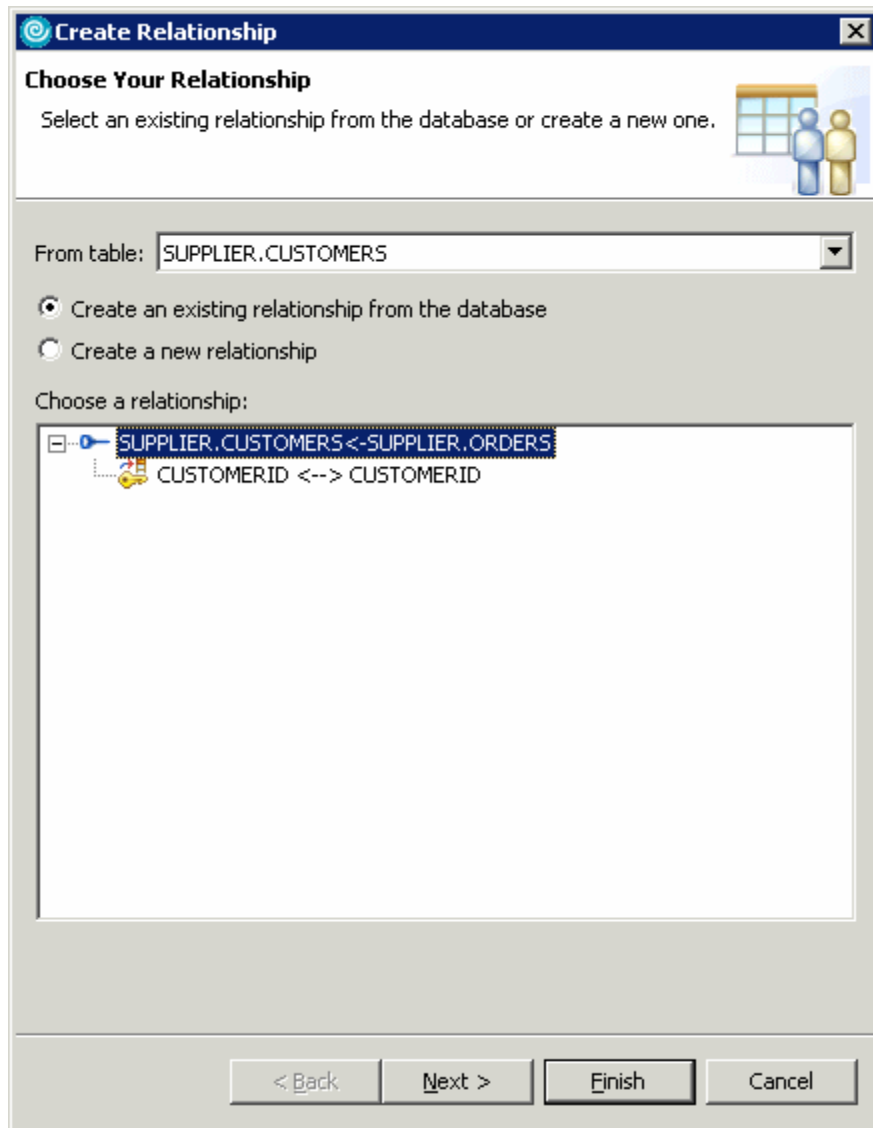
- \_\_\_ d. Click **Finish**.
- \_\_\_ 2. Specify the title for the orders page.
  - \_\_\_ a. The orders page will be displayed in the editor. Select the entire text **Place content here.** and replace it with **Orders**.
  - \_\_\_ b. From the menu select **Format > Paragraph > Heading 1**.
  - \_\_\_ c. From the menu select **Format > Align > Horizontal Center**.
- \_\_\_ 3. Add a Relational Record List to the page data for the orders.jsp file.
  - \_\_\_ a. In the Page Data view, right click and select **New > Relational Record List**.
  - \_\_\_ b. Enter **orderInfo** for the Name, and click **Next**.

- \_\_\_ c. From the Record List Properties page you will need to create a new database connection. Click the **New** button next to Connection name.
- \_\_\_ d. Accept the default connection name, and verify that the **Use Live Connection** radio button is selected and click the **Create New DB Connection** button.
- \_\_\_ e. Click **Next**.
- \_\_\_ f. On the Specify connection parameters page, enter  
<LAB\_FILES>\IRAD\_FacesClientComponents\databases\SUPPLIERDB for the Database location field.
- \_\_\_ g. Click **Finish**.
- \_\_\_ h. After the database connection has been successfully created, you will be returned to the Select A Database page. Click **Finish**.
- \_\_\_ i. On the Record list properties page select the **SUPPLIER.CUSTOMERS** table and click **Next**.

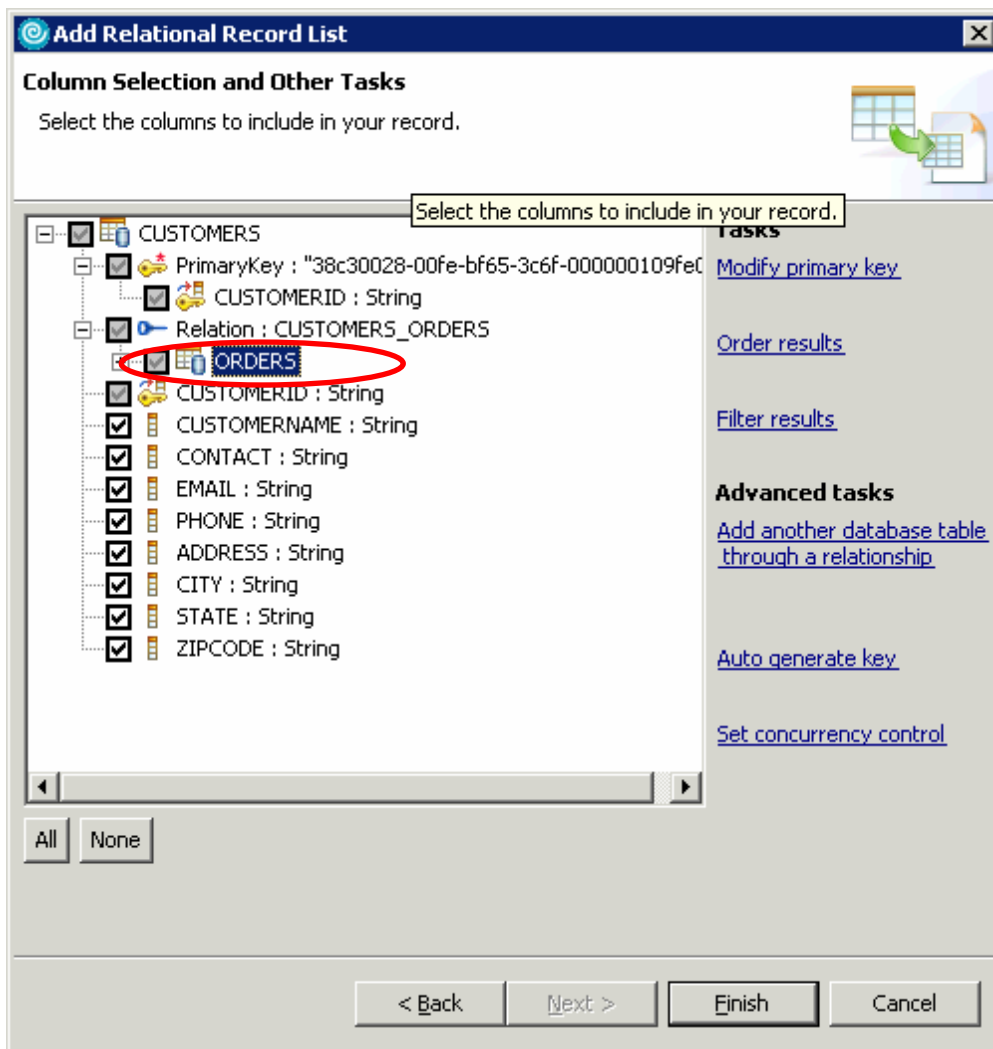
- \_\_\_ j. On the Column Selection and Other Tasks page, click the **Add another database table through a relationship** link on the right under **Advanced Tasks**.



- \_\_\_ k. On the Choose Your Relationship page, verify that the **Create an existing relationship from the database** radio button is selected. Select the **SUPPLIER.CUSTOMERS<-SUPPLIER.ORDERS** entry in the Choose a relationship area, and click **Finish**.



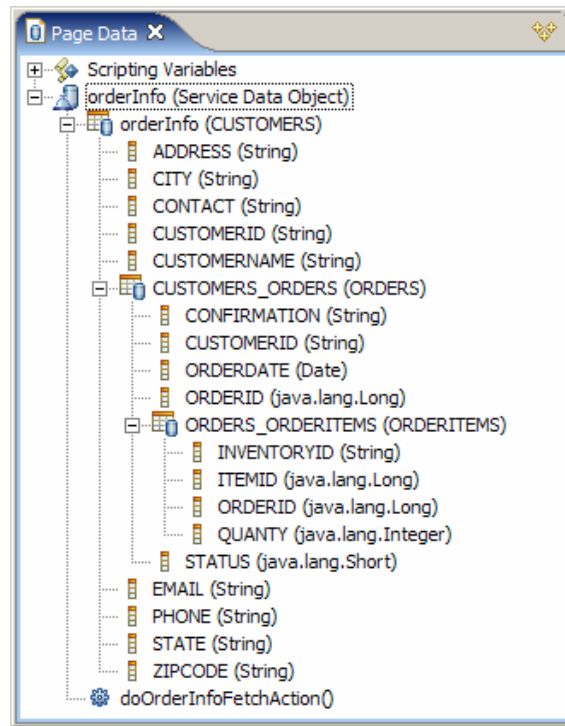
- \_\_ l. Expand the **Relation: CUSTOMERS\_ORDERS** entry and select the **ORDERS** table.



- \_\_ m. Click the **Add another database table through a relationship** link on the right under **Advanced Tasks**.
- \_\_ n. On the Choose Your Relationship page, verify that the **Create an existing relationship from the database** radio button is selected, select the **SUPPLIER.ORDERS←SUPPLIER.ORDERITEMS** entry in the Choose a relationship area, and click **Finish**.
- \_\_ o. Click **Finish** from the Column Selection and Other Tasks page.



\_\_ p. Verify that the following entry appears in the Page Data view.



\_\_\_ 4. Add the Relational Record List to the Client Data view.

\_\_ a. From the **Client Data** view, right click and select **New > Client Data**.

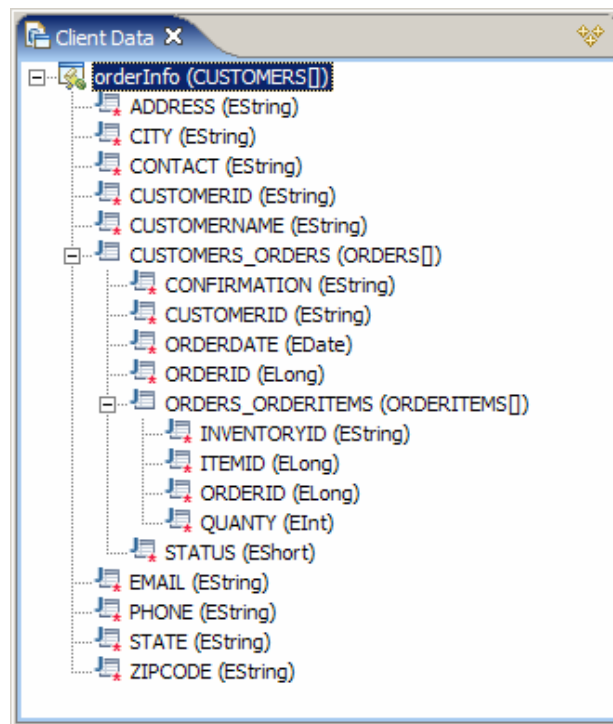
---

**Note:** The Client Data view is located in the same area of the workspace as the Page Data view (lower left hand corner of the workspace). To surface the Client Data view click on the >> icon next to the Page Data view and select Client Data from the drop down list.

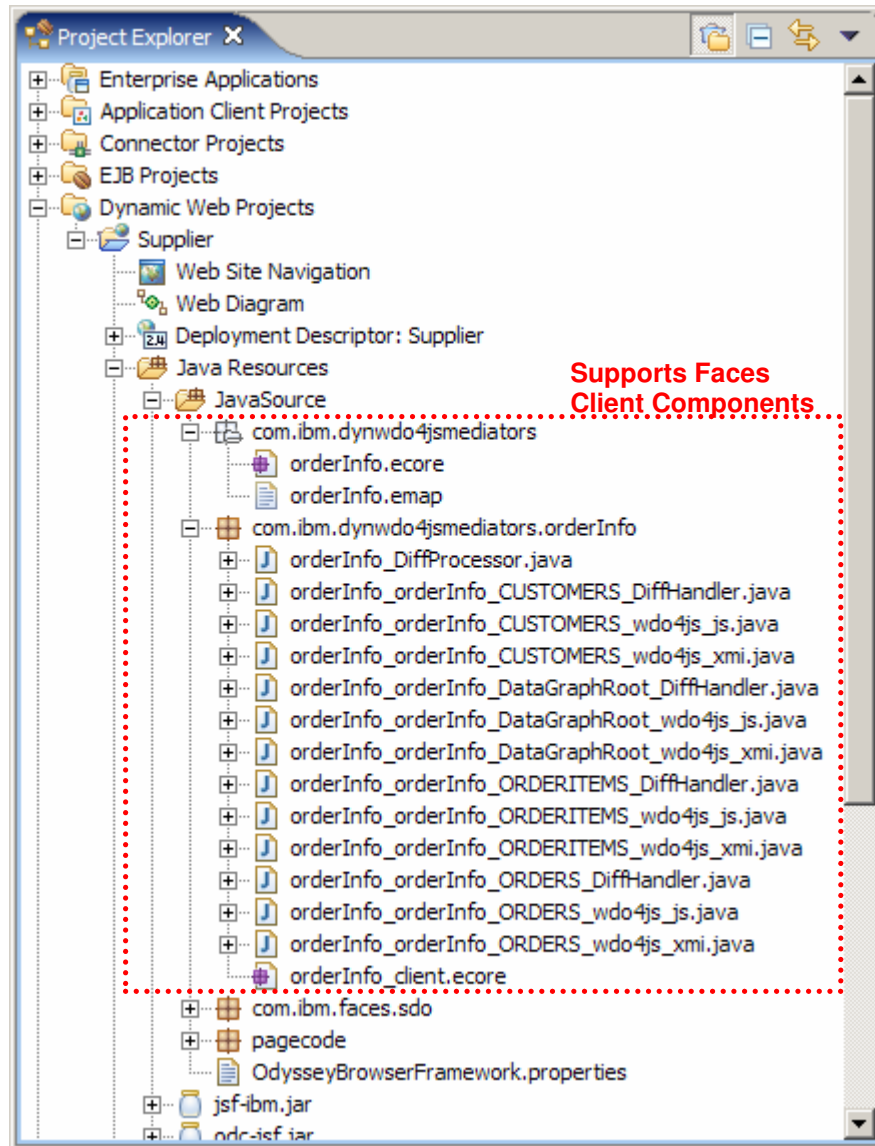
---

\_\_ b. In the **Server Data Model** area, select **orderInfo (CUSTOMERS)** and click **OK**.

\_\_ c. Verify that the following entry appears in the Client Data view.



5. Notice the files that have been automatically generated by Rational Application Developer to support the client data underneath the **Java Resources** folder in the Project Explorer view.



6. Add an HTML table component to format the page data nicely.
- a. From the Palette view expand the **HTML Tags** drawer.
  - b. Select the **Table** component and drag it onto the page.
  - c. From the Insert Table dialog, specify 1 row and 2 columns for this table. Set the Table width to 100 and select % from the drop down list.
  - d. Still on the Insert Table dialog, select the **Apply pattern** as indicated below and click **OK**.



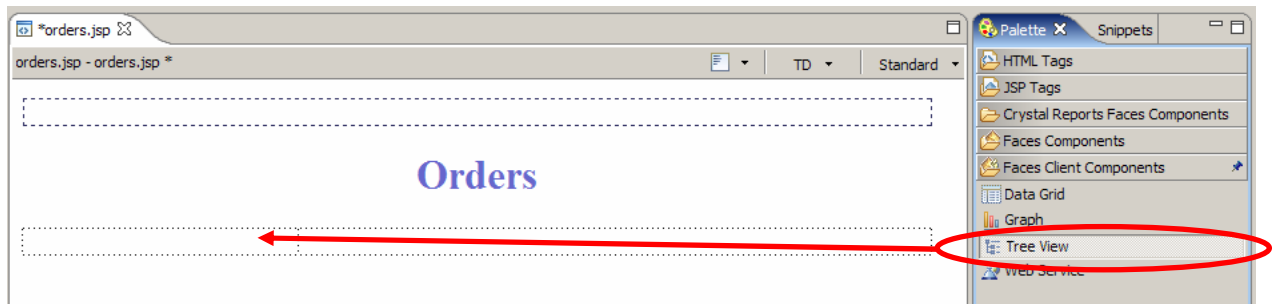
\_\_\_ e. Click inside the left column of the table and go to the **Properties** view. Enter **30** for the Width field and select **%** from the drop down list. Set the vertical alignment to **Top**.

\_\_\_ f. Repeat Step 6e with the right column in the table, but this time set the width at **70%**.

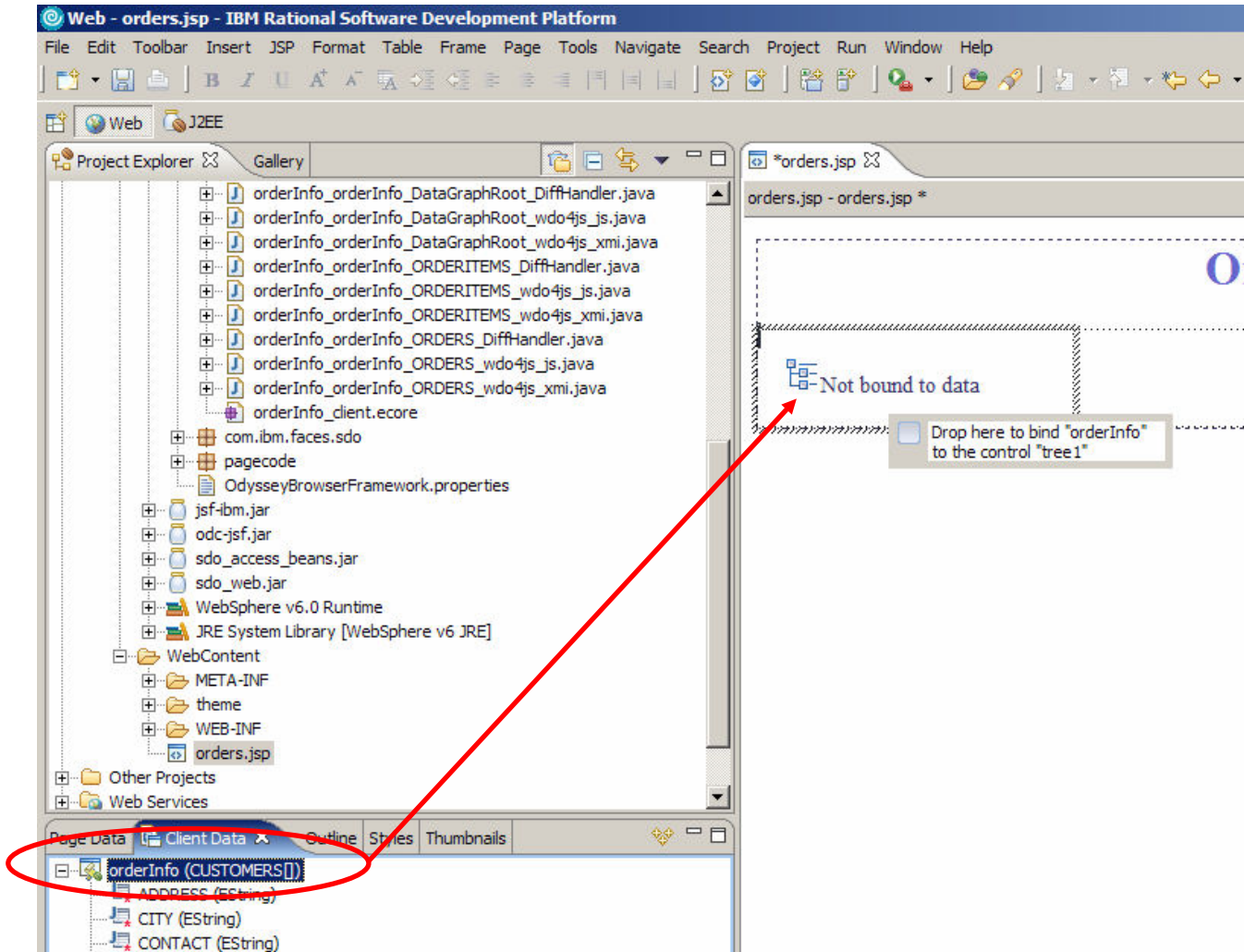
\_\_\_ 7. Create the Tree View

\_\_\_ a. From the Palette view expand the **Faces Client Components** drawer.

\_\_\_ b. Select the **Tree View** component and drag it onto the left most cell of the HTML table



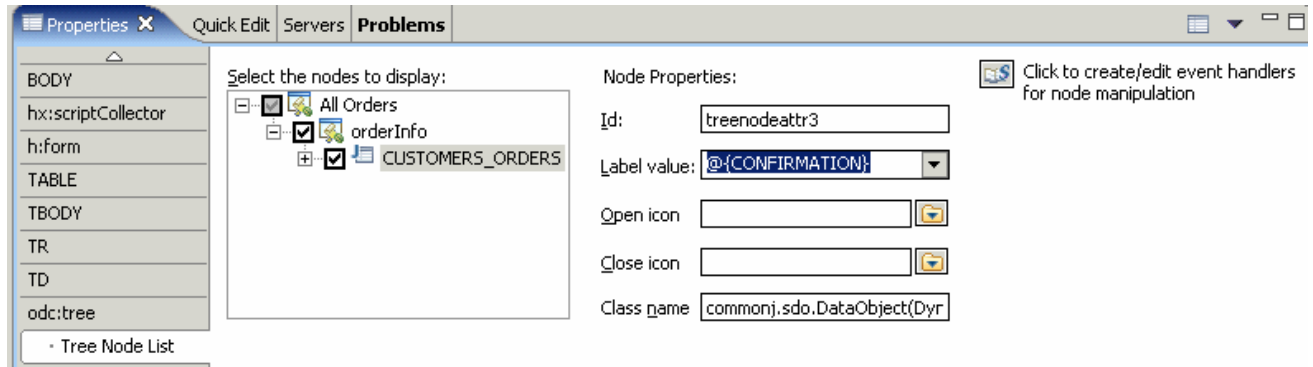
- \_\_ c. From the Client Data view, select **orderInfo(CUSTOMERS [])** and drag it onto the tree view component.



NOTE: When positioned correctly, you should see a fly over message that says "Drop here to bind "orderInfo" to the control "tree1" "

- \_\_ d. Switch to the Properties view for the tree component and select the **Tree Node List** tab.
- \_\_ e. Select the Root node and enter **All Orders** for the Label value.
- \_\_ f. Expand the Root node (All Orders) and make sure the check box next to **orderInfo** is checked. Click on **orderInfo** and select **@{CUSTOMERNAME}** for the Label value.

- \_\_\_ g. Expand the **orderInfo** node and check the checkbox next to **CUSTOMERS\_ORDERS**. Click on **CUSTOMERS\_ORDERS** and select **@{CONFIRMATION}** for the Label value.



- \_\_\_ h. Expand the **CUSTOMERS\_ORDERS** node and check the checkbox next to **ORDERS\_ORDERITEMS**. Click on **ORDERS\_ORDERITEMS** and select **@{INVENTORYID}** for the Label value.

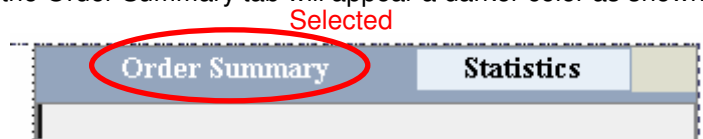
\_\_\_ 8. Create and configure a Tabbed Panel.

- \_\_\_ a. From the Palette view expand the **Faces Components** drawer.
- \_\_\_ b. Select the **Panels – Tabbed** component and drag it onto the right most cell of the table created earlier.
- \_\_\_ c. Select the **tabbed panel** component in the Properties view.
- \_\_\_ d. In the Properties view for the tabbed panel enter **100 %** for the Width field (this property is found on the odc:tabbedPanel tab).
- \_\_\_ e. Remain in the Properties view for the tabbed panel, and switch to the **Panel List** tab. Change the Label of bfpanel1 from Tab1 to **Order Summary**, and the Label for bfpanel2 from Tab2 to **Statistics**.
- \_\_\_ f. For each panel, set the **Show Finish Btn** property to **true**.
- \_\_\_ g. Verify that the labels for the tabs are updated appropriately and that a Finish button appears for each panel.

\_\_\_ 9. Create an output component to be associated with a selected node in the tree view.

- \_\_\_ a. Select the **Order Summary** tab on the tabbed panel.

**Note:** When selected, the Order Summary tab will appear a darker color as shown below:



- \_\_\_ b. From the Palette view expand the **Faces Components** drawer.
- \_\_\_ c. Select the **Output** component and drag it into the interior of the Order Summary tab.

- \_\_\_ d. Repeat the previous step so that 2 Output components have been added to the tabbed panel.
- \_\_\_ e. Select the left Output component and go to the Properties view and enter the text **Order summary for:** into the Value field.
- \_\_\_ f. In the Client Data view select **CUSTOMERNAME** under orderInfo(CUSTOMERS []) and drag it onto the 2<sup>nd</sup> output component created earlier.

**Note:** When positioned correctly, you should see a fly over message that says “Drop here to bind “CUSTOMERNAME” to the control “text2””

- \_\_\_ 10. Create a Data Grid on the “Order Summary” tab.
  - \_\_\_ a. Select the **Order Summary** tab on the tabbed panel.
  - \_\_\_ b. From the Palette view expand the **Faces Client Components** drawer.
  - \_\_\_ c. Select the **Data Grid** component and drag it onto the middle of the tabbed panel.
  - \_\_\_ d. In the Client Data view, expand the **orderInfo (CUSTOMERS [])** node and select **CUSTOMERS\_ORDERS (ORDERS[])**. Drag this data onto the Data Grid on the **Order Summary** tab.

## Orders

Order Summary		Statistics		
Order summary for <sup>abc</sup> {CUSTOMERNAME} <sup>abc</sup>				
ORDERID	CONFIRMATION	CUSTOMERID	STATUS	ORDERDATE
1	1	1	1	1
2	2	2	2	2
3	3	3	3	3
4	4	4	4	4
5	5	5	5	5

**Note:** When positioned correctly, you should see a fly over message that says “Drop here to bind “CUSTOMERS\_ORDERS” to the control “datagrid1””

- \_\_\_ e. In the Properties view for the Data Grid component, select the **odc:dataGrid** tab and enter **5** for the Height (rows per page) field.
- \_\_\_ f. Check the box next to **Rows can be added/deleted** (still on the odc:dataGrid tab of the Properties view)
- \_\_\_ g. Switch to the Columns tab in the Properties view (right underneath odc:dataGrid) and select **ORDERDATE** in the table. Switch to the Column details tab and select **Date/Time** for the Format for the ORDERDATE column.
- \_\_\_ h. Switch to the Columns tab in the Properties view (right underneath odc:dataGrid) and select **STATUS** in the table. Switch to the Column details tab and check the box next to **Column content is editable**.

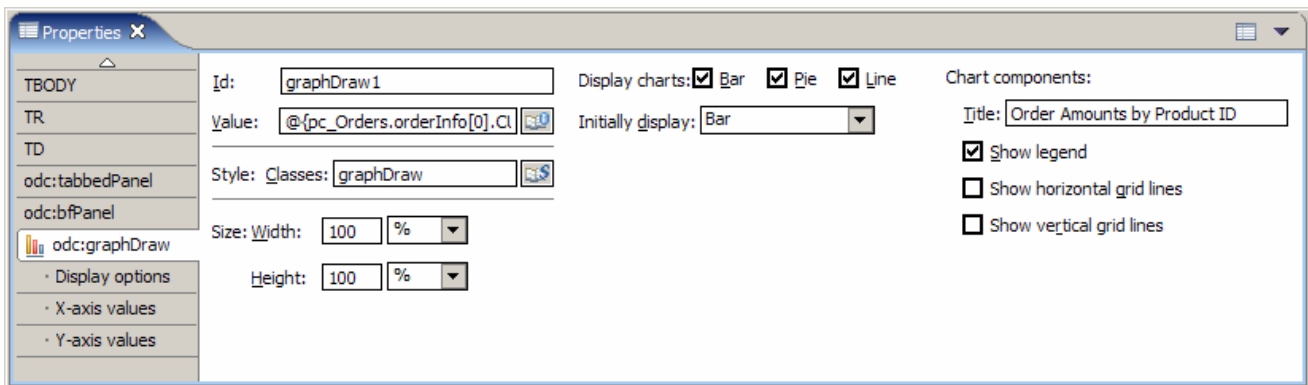
- \_\_\_ 11. Create a Graph component on the Statistics tab.
- \_\_\_ a. Select the **Statistics** tab on the tabbed panel.
  - \_\_\_ b. From the Palette view expand the **Faces Client Components** drawer.
  - \_\_\_ c. Select the **Graph** component and drag it onto the middle of the tabbed panel.
  - \_\_\_ d. In the Client Data view, expand the **orderInfo (CUSTOMERS []) > CUSTOMERS\_ORDERS** node and select **ORDERS\_ORDERITEMS (ORDERITEMS [])**. Drag this data onto the Graph on the “Statistics” tab.

---

**Note:** When positioned correctly, you should see a fly over message that says “Drop here to bind “ORDERS\_ORDERITEMS” to the control “graphDraw1”

---

- \_\_\_ e. In the Properties view for the Graph component, do the following on the **odc:graphDraw** tab:
  - 1) For **Display charts**, check the checkboxes next to Bar, Pie, and Line.
  - 2) For the **Initially display**, select **Bar** from the drop down list.
  - 3) Set the Title to **Order Amounts by Product ID**.
  - 4) Set the **Width** and **Height** to 100 % for both.

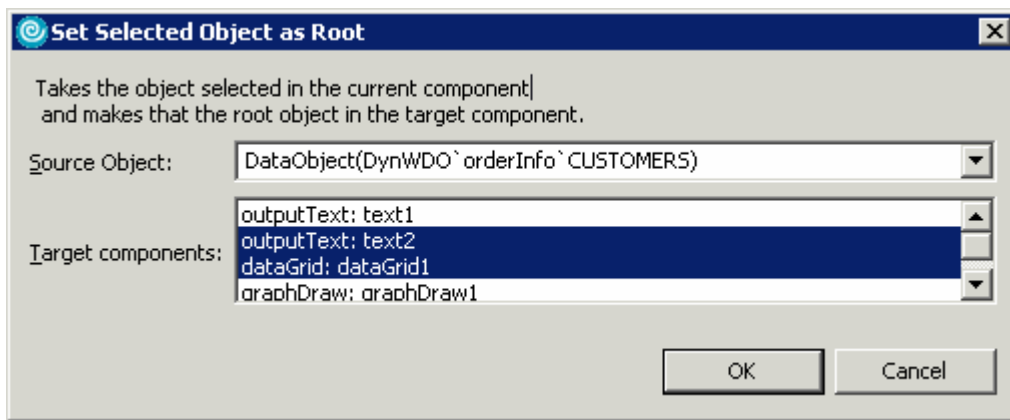


- \_\_\_ f. On the **Display options** tab in the Properties view set the Y-axis title to **Quantity** and the X-axis title to **Inventory ID**.
  - \_\_\_ g. On the **X-axis values** tab of the Graph Properties view select **INVENTORYID** from the drop down list for the value plotted on the X-axis.
  - \_\_\_ h. On the **Y-axis values** tab of the Graph click the **Add Series** button. From the drop down list for the new series, select **QUANTITY** from the value drop down list and enter **Quantity** for the Series Name.
- \_\_\_ 12. Bind orderInfo node in tree view to data grid, and output text components
- \_\_\_ a. Select the **Tree View** in left column of the table.
  - \_\_\_ b. Select **orderInfo** in the graphical hierarchy of the Tree View (not the Properties view). When orderInfo is selected it will be bolded.





- \_\_\_ c. Go to the **Quick Edit** view and click on the **onhighlight** event.
- \_\_\_ d. Right click in the editor area, of the onhighlight event, and select **Insert Snippet > Set selected object as root in target component**.
- \_\_\_ e. In the **Set Selected Object as Root** dialog, select the **DataObject(DynWDO`orderInfo`CUSTOMERS)** from the Source Object drop down list, and select all of the following in the Target components list: **dataGrid1**, and **text2**. Then click **OK**.



- \_\_\_ 13. Bind CUSTOMERS\_ORDERS node in tree view to data graph component.
  - \_\_\_ a. Click on the **CUSTOMERS\_ORDERS** node in the Tree Component in the Left column of the table. Make sure the text in the CUSTOMERS\_ORDERS node is now **bold**. Go to the **Quick Edit** view and click on the **onhighlight** event.
  - \_\_\_ b. Right click in the editor area, of the onhighlight event, and select **Insert Snippet > Set selected object as root in target component**.
  - \_\_\_ c. In the **Set Selected Object as Root** dialog, select the **DataObject(DynWDO`orderInfo`ORDERS)** from the Source Object drop down list, and select all of the following in the Target components list: **graphDraw1**. Then click **OK**.
- \_\_\_ 14. Add action code to handle updates to Orders table
  - \_\_\_ a. Click on the Finish button at the bottom of the Tabbed Panel
  - \_\_\_ b. From the quick edit view select the Command item in the left pane.
  - \_\_\_ c. Add the following code to the editor area for the quick edit view. For your convenience, this code can be found in <LAB\_FILES>\IRADv6\_FacesClientComponents\snippets\snippet1.txt.

```

boolean success = true;
try {
    DataObject root =
        (DataObject) ((EcoreEList) getOrderInfo()).getEObject();
    getOrderInfoMediator().applyChanges(root);
    doOrderInfoFetchAction();
}

```

```
    } catch (Throwable e) {  
        logException(e);  
        success = false;  
    }  
  
    return success ? "success" : "false";  
}
```

\_\_\_ d. Save the **orders.jsp** file.

\_\_\_ 15. After adding the method as indicated in the previous step, you should notice an error flagged in the Quick Edit view. This error is the result of not having the appropriate import statement for the `EcoreEList` class. Resolve this issue before continuing.

\_\_\_ a. Go to the Problems view and right click on the error. Select Quick Fix from the context menu.

\_\_\_ b. Select **Import 'EcoreEList' (org.eclipse.emf.ecore.util)** from the available fixes list.

\_\_\_ c. Save and close the `Orders.java` file.

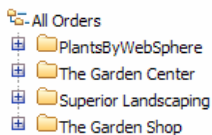
\_\_\_ 16. **Save** changes to `orders.jsp`.

## Part 3: Test Supplier Administration Application

In the last section you built the Supplier administration UI. In this part of the lab you will test and explore the UI you have created with the Faces Client Components by testing it with the WebSphere version 6.0 Test Environment. When the application comes up in the browser, you will notice how selecting a node in the tree causes changes to other components on the page.

- \_\_\_ 1. Backup your server configuration. This will preserve your current server configuration. You will restore your server configuration at the end of the lab exercise.
  - \_\_\_ a. Open a Windows **Command Prompt** and navigate to the following directory:  
`<RAD_HOME>\runtimes\base_v6\bin`
  - \_\_\_ b. Backup the server configuration by issuing the following command:  
`backupConfig "C:\Program Files\IBM\Backup.zip"`
- \_\_\_ 2. Start the application on the server.
  - \_\_\_ a. Locate the orders.jsp file in the Project Explorer view. Right click on this file and select **Run > Run on Server...**
  - \_\_\_ b. In the Define a New Server window, accept the server listed in the view and click **Next**.
  - \_\_\_ c. Verify the SupplierEAR is listed under the Configured projects list. Click **Finish**.
  - \_\_\_ d. A message box may be displayed indicating that the server can not be connected. Click **Yes** to have the server started.
  - \_\_\_ e. If a window pops up asking if you want to install Macromedia Flash, click **Yes**.
- \_\_\_ 3. When the browser opens verify that your page looks like the following:

### Orders



ORDERID	CONFIRMATION	CUSTOMERID	STATUS	ORDERDATE
0	PBW001-0	PBW001	0	Jun 30, 2004
3	PBW001-3	PBW001	0	Jun 25, 2004
5	PBW001-5	PBW001	0	Jul 1, 2004
6	PBW001-6	PBW001	0	Jul 9, 2004
7	PBW001-7	PBW001	0	Jul 10, 2004

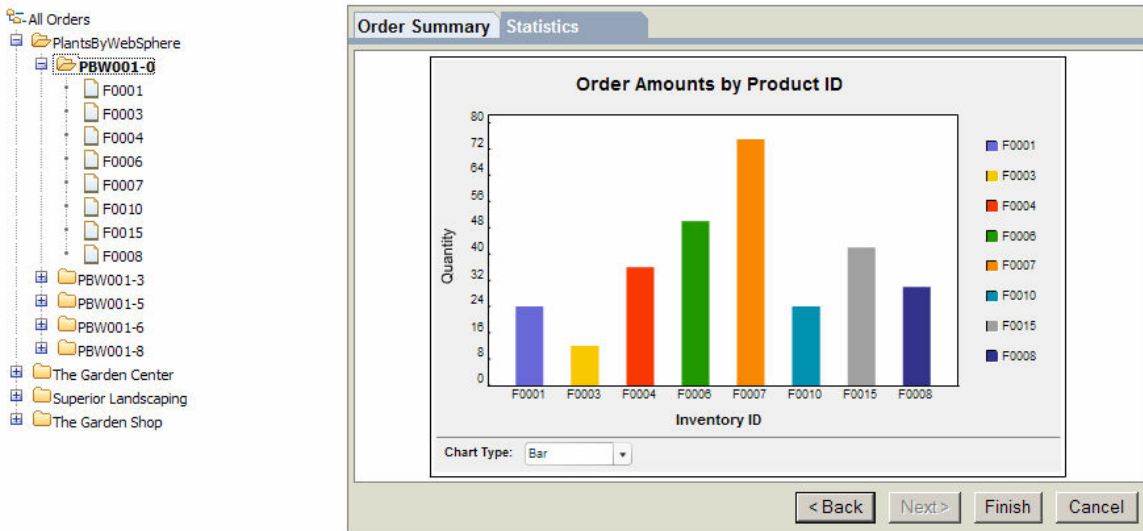
Page 1 of 2    Jump to page:

< Back    Next >    Finish    Cancel

- \_\_\_ 4. Expand the tree node for customer **PlantsByWebSphere** and verify that the orders listed under the node match those in the table shown on the Order Summary panel.
- \_\_\_ 5. The orders for customer PlantsByWebSphere exceed the page/item size for the data grid. Use the paging links at the bottom of the data grid to page through the data.

- \_\_\_ 6. Click on any of the column header labels on the data grid. Notice that this action will sort the items in the table.
- \_\_\_ 7. Initially, the data shown on the Summary tab matches the data associated with the first customer in the tree view (in this case PlantsByWebSphere). Now, highlight one of the other customers in the tree view. Notice that the data shown on the Order Summary tab changes when different customers in the tree are highlighted.
- \_\_\_ 8. Select the Statistics panel. Verify that your page looks like the following when Order **PBW001-0** (under PlantsByWebSphere) is highlighted in the tree view.

## Orders



- \_\_\_ 9. Select another order node in the tree (for any customer) and verify that the graphs change for different customers.
- \_\_\_ 10. Verify that you can switch to the different types of graphs by selecting the appropriate type in the drop down list below the graph (Pie, Bar, Line).
- \_\_\_ 11. Return to the Order Summary tab and delete an order listed for the PlantsByWebSphere customer.
  - \_\_\_ a. On the tree view click on PlantsByWebSphere.
  - \_\_\_ b. Click the delete button to the right of the Order with the OrderID of 7

The screenshot shows the 'Order Summary' tab for 'PlantsByWebSphere'. The table below contains the following data:

ORDERID ^	CONFIRMATION	CUSTOMERID	STATUS	ORDERDATE	
0	PBW001-0	PBW001	0	Jun 30, 2004	X
3	PBW001-3	PBW001	0	Jun 25, 2004	X
5	PBW001-5	PBW001	0	Jul 1, 2004	X
6	PBW001-6	PBW001	0	Jul 9, 2004	X
7	PBW001-7	PBW001	0	Jul 10, 2004	X

At the bottom of the table, there is a pagination control showing 'Page 1 of 2' and a 'Jump to page:' field.

- \_\_\_ c. Click the Finish button at the bottom of the Tabbed panel to submit the changes to the server.
  - \_\_\_ d. Verify that the order has been deleted.
- \_\_\_ 12. Update the status for an order listed for the PlantsByWebSphere customer.
- \_\_\_ a. On the tree view click on PlantsByWebSphere.
  - \_\_\_ b. For Order 3 change the STATUS from 0 to 1.
  - \_\_\_ c. Click the Finish button at the bottom of the Tabbed panel to submit the changes to the server.
  - \_\_\_ d. Verify that the status has been updated.
- \_\_\_ 13. When you have completed testing / exploring the client components, close the browser window and stop the server.
- \_\_\_ 14. Remove the Supplier application from the application server
- \_\_\_ a. From the Server view right click on the server and select Add and Remove projects
  - \_\_\_ b. In the Configured projects list select SupplierEAR and click the Remove button
  - \_\_\_ c. Click **Finish**
- \_\_\_ 15. Stop the server
- \_\_\_ a. From the Server view, right click on the server and select Stop from the context menu

## Part 4: Restore the Server Configuration

\_\_\_ 1. Restore your server configuration. This will return your server configuration to its original state.

\_\_\_ a. Open a Windows **Command Prompt** and navigate to the following directory:

<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

In this lab exercise you built and tested an administrative UI for the Supplier database associated with the PlantsByWebSphere application. The application you built in this lab demonstrates how to configure and use various Faces Client Components. Additionally this lab shows how the Faces Client Components can be used in conjunction with SDO technologies.

## 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 for examples of how the data components look and function in the final application.

- \_\_\_ 2. Start with a clean workspace and follow the steps to create the SupplierDB database in Part 1

---

**NOTE:** If the Auto Launch Configuration Change Alert window appears click the **Yes** button to change the auto launch eclipse instance to use when opening IBM Rational Software Development Platform in the future.

---

- \_\_\_ 3. Import the **Faces Client Components application** into Rational Application Developer for testing.

\_\_\_ a. Select **File > Import...**

\_\_\_ b. Select **Project Interchange** and select **Next**.

\_\_\_ c. Select **Browse...** and navigate to  
**<LAB\_FILES>\IRAD\_FacesClientComponents\solution\solutionFacesClientComponents.zip** and select **Open**.

\_\_\_ d. Click **Select All**.

\_\_\_ e. Click **Finish**.

\_\_\_ f. In the explorer view, navigate to **Supplier>WebContent> order.jsp** and double click the file.

- \_\_\_ 4. Switch to the **Web Perspective** view.

\_\_\_ a. From the menu select **Window > Open Perspective > Other....**

\_\_\_ b. Select **Web** and click **OK**.

- \_\_\_ 5. Now you can test the solution following the steps outlined in part 3.



## Trademarks and Disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	iSeries	OS/400	Informix	WebSphere
IBM(logo)	pSeries	AIX	Cloudscape	MQSeries
e(logo)business	xSeries	CICS	DB2 Universal Database	DB2
Tivoli	zSeries	OS/390	IMS	Lotus

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. Microsoft, Windows, Windows NT, and

the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both. Intel, ActionMedia, LANDesk, MMX, Pentium and ProShare are

trademarks of Intel Corporation in the United States, other countries, or both. UNIX is a registered trademark of The Open Group in the United States and other countries. Linux is a registered trademark of Linus Torvalds. Other company, product and service names may be trademarks or service marks of others.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements and/or changes in the product(s) and/or program(s) described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (e.g., IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.

This page is intentionally left blank.