ORACLE

An Oracle White Paper
April 2014

Oracle Application Express Plug-in for Ajax Maps: Map Integration the Easy Way

Introduction

This hands-on lab will show how to integrate an Oracle Maps (MapViewer) application into APEX 4.0 with minimal coding effort using the APEX plugin framework. The lab will describe how to set up and configure MapViewer and install it as an APEX plugin so that a map can be a Region Type in an APEX application. The latter portion will deal with extending the basic functionality to add interactivity, event handling, and communication between various components on the page. We'll add a tabular view and connect up the regions so that when a user clicks on a row on the table the corresponding location, associated with that row, is highlighted on the map.

Attendees are expected to be familiar with Application Express, SQL, PLSQL, and JavaScript.

Pre-requisites and what to expect in this lab

- Basic familiarity with Oracle Spatial and MapViewer technology is required.
- Basic familiarity with APEX will be useful.
- Understanding of JavaScript will also be useful.
- This lab is staged on a single database instance and APEX instance. Please make sure to follow the instructions carefully on creating the workspaces and user so that your work does not conflict with others in the lab.
- Spatial data required for this lab needs to be staged on the database and the MapViewer themes you need to build the sample application should be created. These theme definitions are also provided in the Appendix at the end of this document for your reference. You need to install the MapViewer and the corresponding MVDEMO schema (http://www.oracle.com/technetwork/middleware/mapviewer/downloads/index.html).

Creating a user and workspace for application development

The same DB and APEX install is used for this hands-on-lab. So it is very important that each user create a unique workspace and user for this lab. Hint: Use your first name or last name as part of the workspace name and user name below.

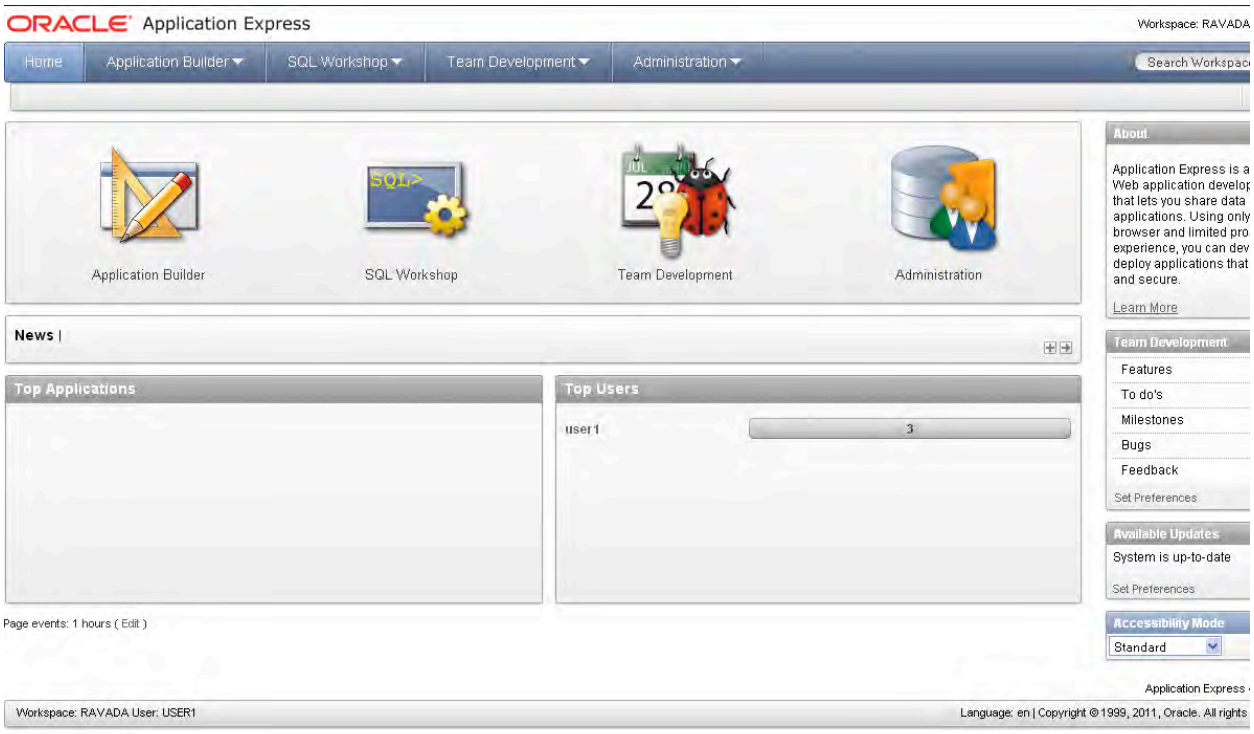
Login to APEX as the admin user and create a workspace and a user which will be used for developing the sample applications. The database schema used for the application is APEX. The host where APEX is accessed via http is called APEX-host in the rest of this document.

- Go to http://APEX-host/APEX/APEX_admin and connect as Admin/<admin_password>
- Click on "Create Workspace" and enter a name with your name as a prefix for the workspace name. And click "Next."
- Pick "Yes" for the "Re-use existing schema" option. And for the schema name, pick "APEX" from the list of values. Click "Next."
- User "ADMIN" for the administrator name and password: Spatial2!. Use a dummy email address for the email field and click "Next."
- And click on "Create Workspace" to create the workspace and click on "Done."
- Click on "Manage Workspace" tab at the top.
- In the "Workspace Actions" click on "Manage Developers and Users."
- Click on "Create User."
- Pick a unique username (preferably with your first name and last name) and use a dummy email address.

- In "Account privileges" section, pick the name of the workspace you just created.
- For "Default Schema" pick APEX.
- Next move to the "Password" section. And enter a password. For the "Require change of password on First use", pick "No".
- Click on "Create User" and then end the session by logging out of the admin user.

Logging into the APEX application development portal

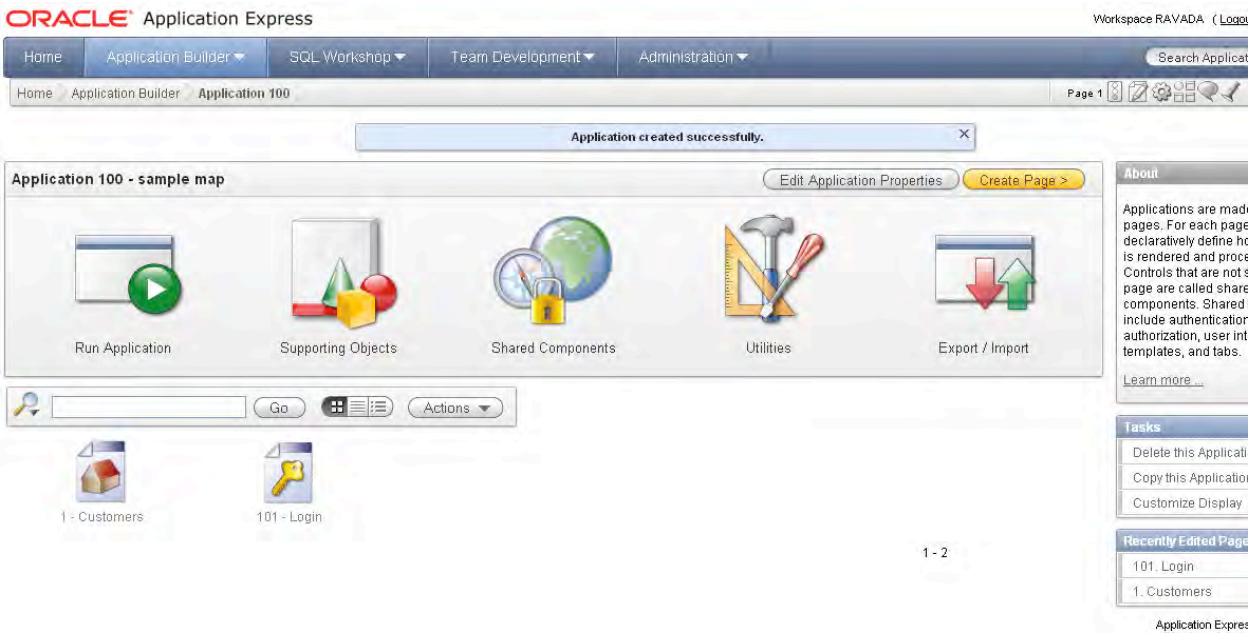
Go to http://APEX-host/APEX/ and enter the workspace name, username and password you just created. You will be taken to this page as shown below. You are all set to start creating web applications.



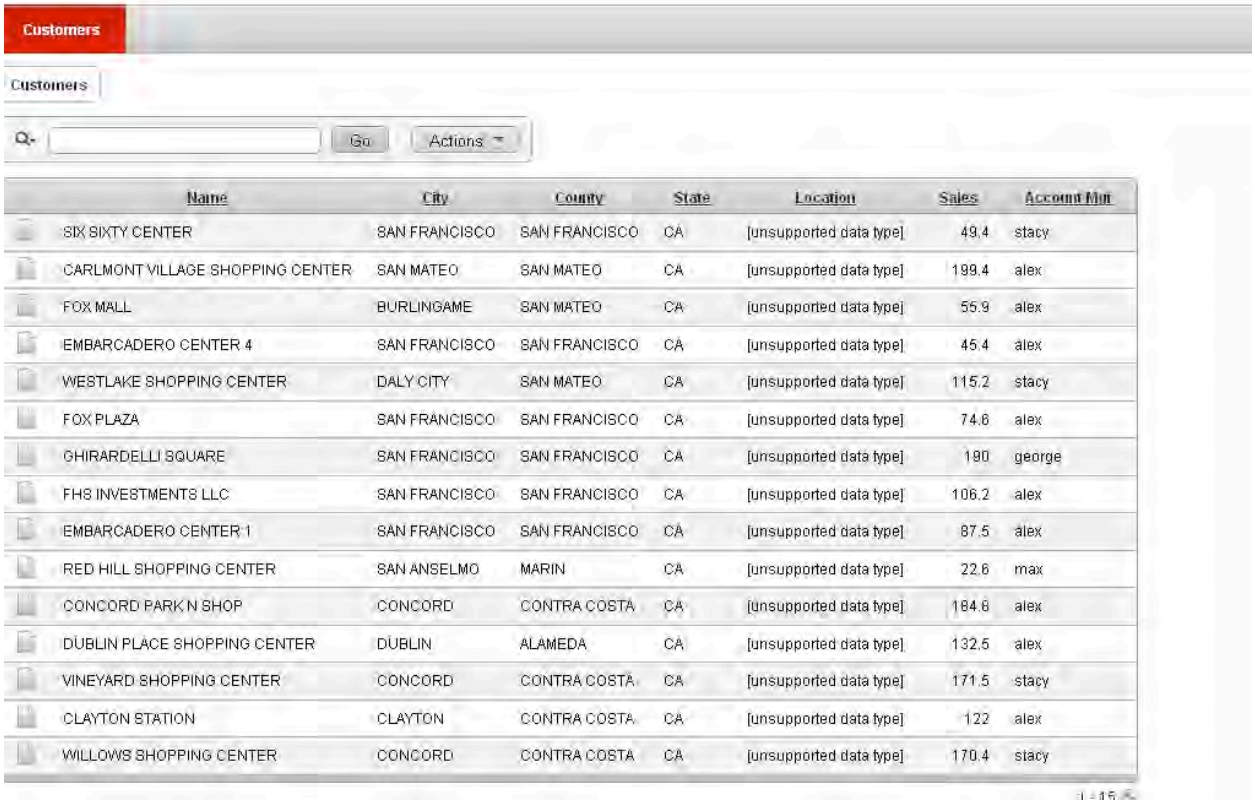
Building a simple reporting application

- Click on "Application Builder" and then click on "Create" to start creating a simple application.
- Pick "Database" for application type and click on "Next".
- Pick "From Scratch" and click on "Next".

- Give a name to the application and pick an ID (any integer number). Pick from "Scratch" and click on "Next".
- On this screen, you need to pick a type for the application page you are about to create. For this, pick "Report" since we want to create a simple reporting application. So pick "report" and it will show a list box for the source of the report. Pick "Table" and pick "Customers" for the table name. Then click on "Add Page".
- Click on "Create" to create the page and then again "Create" to create the application.
- You will a screen like the one below at this time.



- Click on "Run Application" to run the application and you will see this report.

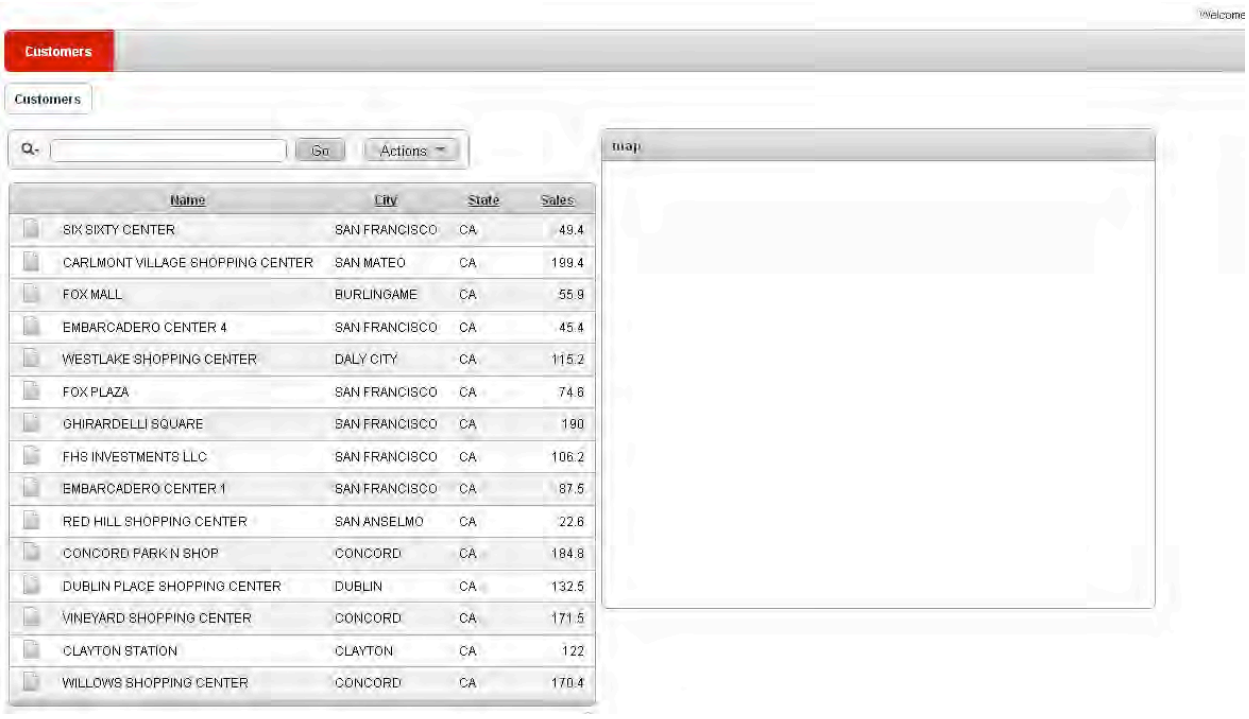


Name	City	County	State	Location	Sales	Account Mgr
SIX SIXTY CENTER	SAN FRANCISCO	SAN FRANCISCO	CA	[unsupported data type]	49.4	stacy
CARLMONT VILLAGE SHOPPING CENTER	SAN MATEO	SAN MATEO	CA	[unsupported data type]	199.4	alex
FOX MALL	BURLINGAME	SAN MATEO	CA	[unsupported data type]	55.9	alex
EMBARCADERO CENTER 4	SAN FRANCISCO	SAN FRANCISCO	CA	[unsupported data type]	45.4	alex
WESTLAKE SHOPPING CENTER	DALY CITY	SAN MATEO	CA	[unsupported data type]	115.2	stacy
FOX PLAZA	SAN FRANCISCO	SAN FRANCISCO	CA	[unsupported data type]	74.6	alex
GHIRARDELLI SQUARE	SAN FRANCISCO	SAN FRANCISCO	CA	[unsupported data type]	190	george
FHS INVESTMENTS LLC	SAN FRANCISCO	SAN FRANCISCO	CA	[unsupported data type]	106.2	alex
EMBARCADERO CENTER 1	SAN FRANCISCO	SAN FRANCISCO	CA	[unsupported data type]	87.5	alex
RED HILL SHOPPING CENTER	SAN ANSELMO	MARIN	CA	[unsupported data type]	22.6	max
CONCORD PARK N SHOP	CONCORD	CONTRA COSTA	CA	[unsupported data type]	184.6	alex
DUBLIN PLACE SHOPPING CENTER	DUBLIN	ALAMEDA	CA	[unsupported data type]	132.5	alex
VINEYARD SHOPPING CENTER	CONCORD	CONTRA COSTA	CA	[unsupported data type]	171.5	stacy
CLAYTON STATION	CLAYTON	CONTRA COSTA	CA	[unsupported data type]	122	alex
WILLOWS SHOPPING CENTER	CONCORD	CONTRA COSTA	CA	[unsupported data type]	170.4	stacy

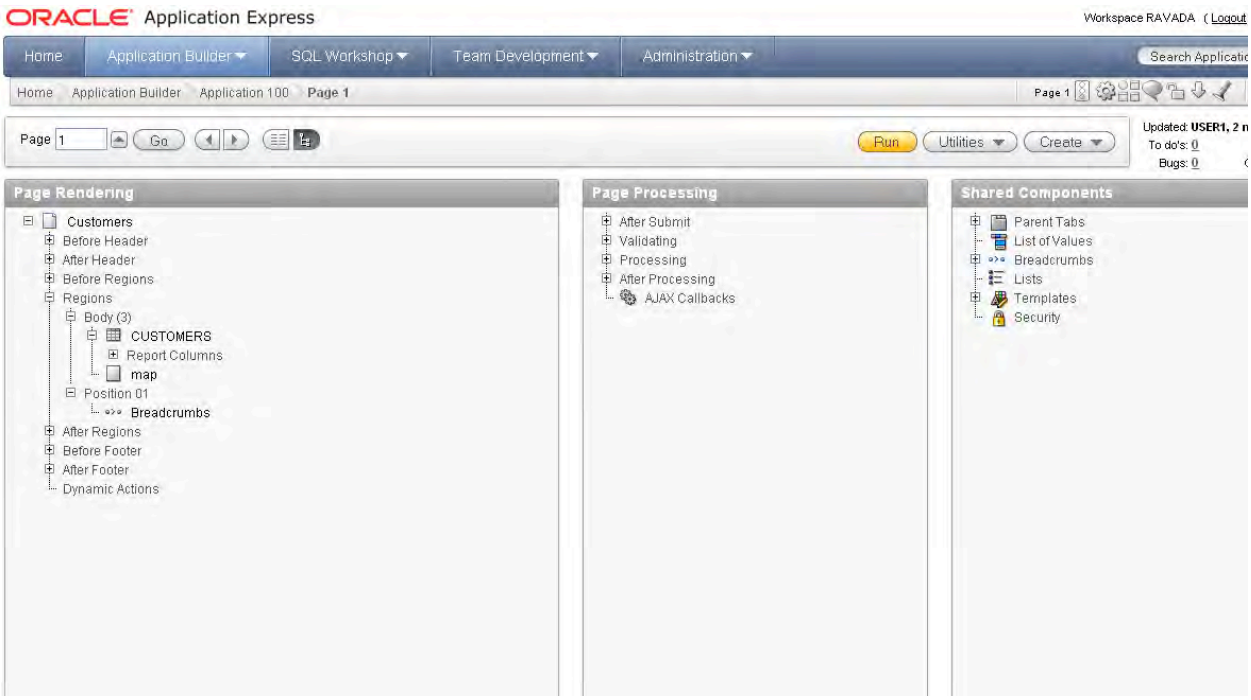
- In this report, the "Location" value is displayed as "unsupported data type" instead of the sdo_geometry values. Since this information is not suitable for the report, we can filter this column out of the report. To reduce the size of the report, let's remove a few columns from the report so that we will have space to add a map later to this application.
- Click on "Actions" and pick "select columns" and move "Location", "Account Manager" and "County" to the "Do Not Display" panel and click "Apply."
- Now you will see only four columns displayed in the report.

Add a simple map to the application

- Run the application to display the report.
- Click on “Create” at the bottom of the screen and pick “Region on this page” and click “Next.”
- Pick “HTML” for the region type, click “Next” and pick “HTML” again for the type of HTML used for the region and click “Next”.
- Add “Map” for the title and pick “2” for “Column” option. Leave the rest as defaults. Click “Next.”
- On the next screen, you need to enter the top level HTML used to define the region we just created. Enter this text: in the text box and click “Next“. This is specifying that we want to create an HTML region that is 500 pixels wide and 400 pixels high.
- Then run the application and you will see the report and an empty box next to it.



- Now click on “Edit Page 1” at the bottom of the screen to bring you to this page.



- Right click on "Customers" and "Edit".
- Now we need to specify the JavaScript library we want to use to display the map. Pick the "Header and Footer" tab and paste the following text into the "Header Text" box.

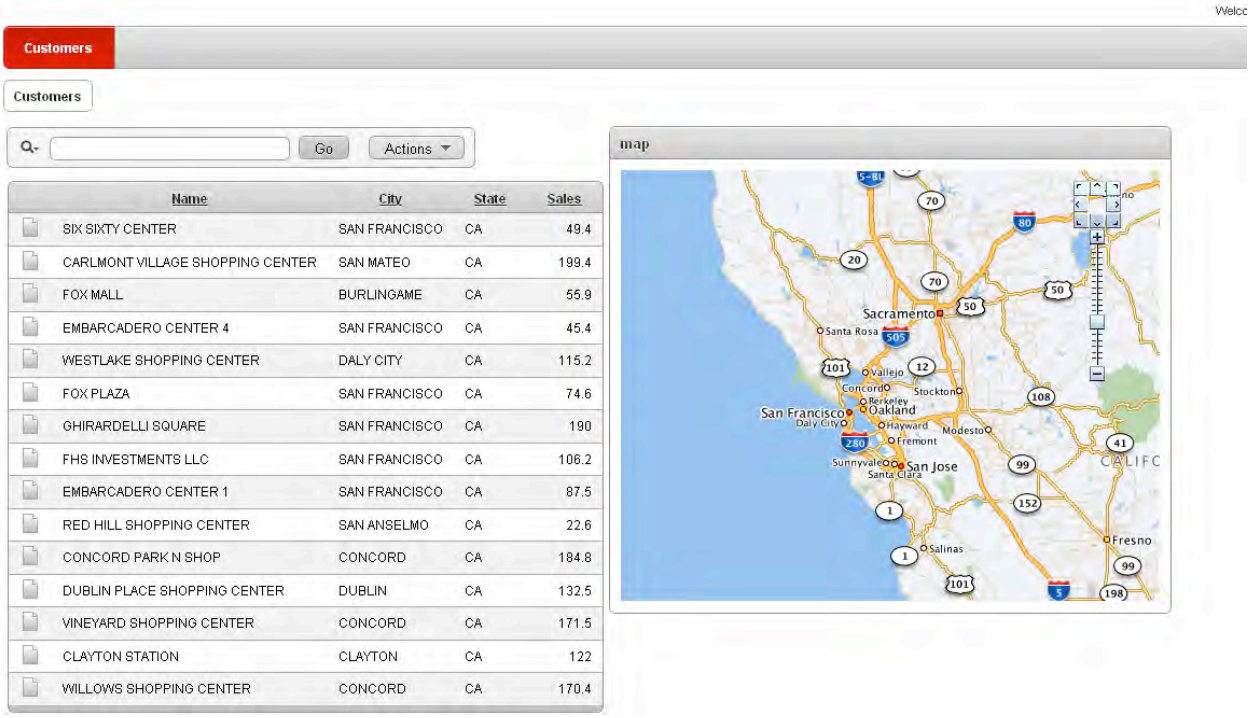
  <script language="JavaScript" src="http://elocation.oracle.com/mapviewer/fsmc/jslib/oraclemaps.js"></script>

- Next we need to specify the JavaScript code we want to run in the HTML region we created. For this, click on "JavaScript" tab and enter the following code into the "Function and Global Variable Declaration" text box.

```
var eLocURL = "http://elocation.oracle.com/mapviewer/mcserver";
var baseURL = "http://local-mapviewer-host/mapviewer";
var mapview;
 function display_map() {
    mapview = new MVMapView(document.getElementById("map"), baseURL);
  mapview.addMapTileLayer(new MVMapTileLayer("elocation_mercator.world_map",
eLocURL));
```

```
    mapview.setCenter(MVSdoGeometry.createPoint(-122, 38, 8307));
    mapview.setZoomLevel(6);
    mapview.addNavigationPanel("east");
    mapview.display();
}
```

- Next, click on the "HTML Header and Body Attribute" and past the following into the text box for "Page HTML Body Attribute": onLoad="display_map()"
- Next go to the "Display Attributes" tab and pick "Do not Focus Cursor" for the "Cursor Focus" option.
- Click on "Apply Changes" and "Run" the application.



Add customer theme to the map

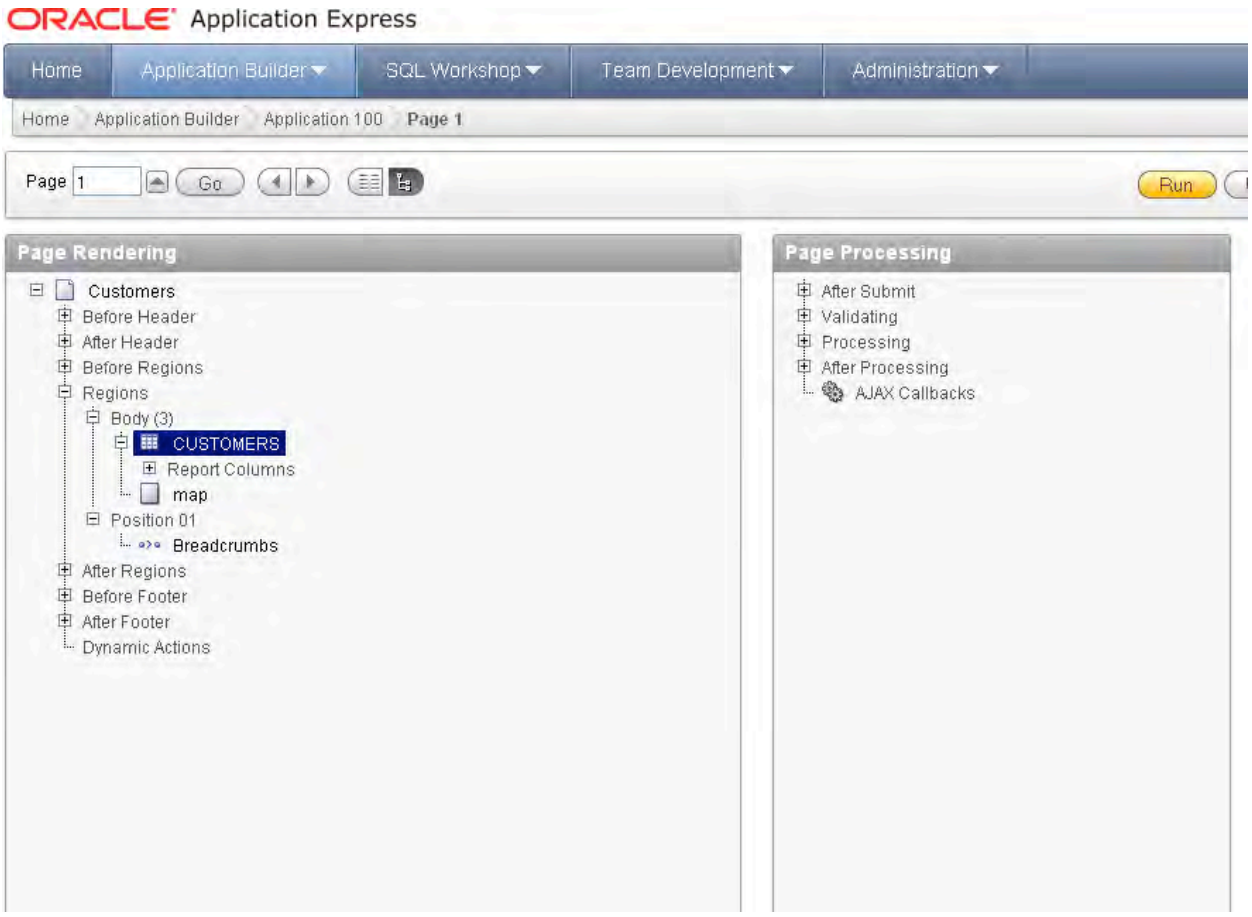
- Run the current application.
- Click on "Edit Page 1" at the bottom of the screen.
- Right click on "Customers" and pick "Edit."
- Go to the "JavaScript" tab and replace the JavaScript code there with this one. The new code added here is highlighted in red color.

```
var eLocURL = "http://elocation.oracle.com/mapviewer/mcserver";
var baseURL = "http://local-mapviewer-host/mapviewer";
var mapview;
 function display_map() {
     mapview = new MVMapView(document.getElementById("map"), baseURL);
  mapview.addMapTileLayer(new MVMapTileLayer("elocation_mercator.world_map",
eLocURL));
     mapview.setCenter(MVSdoGeometry.createPoint(-122, 38, 8307));
     mapview.setZoomLevel(6);
     mapview.addNavigationPanel("east");
     var foi = new MVThemeBasedFOI("custfoi", "mvdemo.CUSTOMERS");
     foi.enableAutoWholeImage(true);
     foi.setMinVisibleZoomLevel(5);
     mapview.addThemeBasedFOI(foi);
     mapview.display();
 }
```

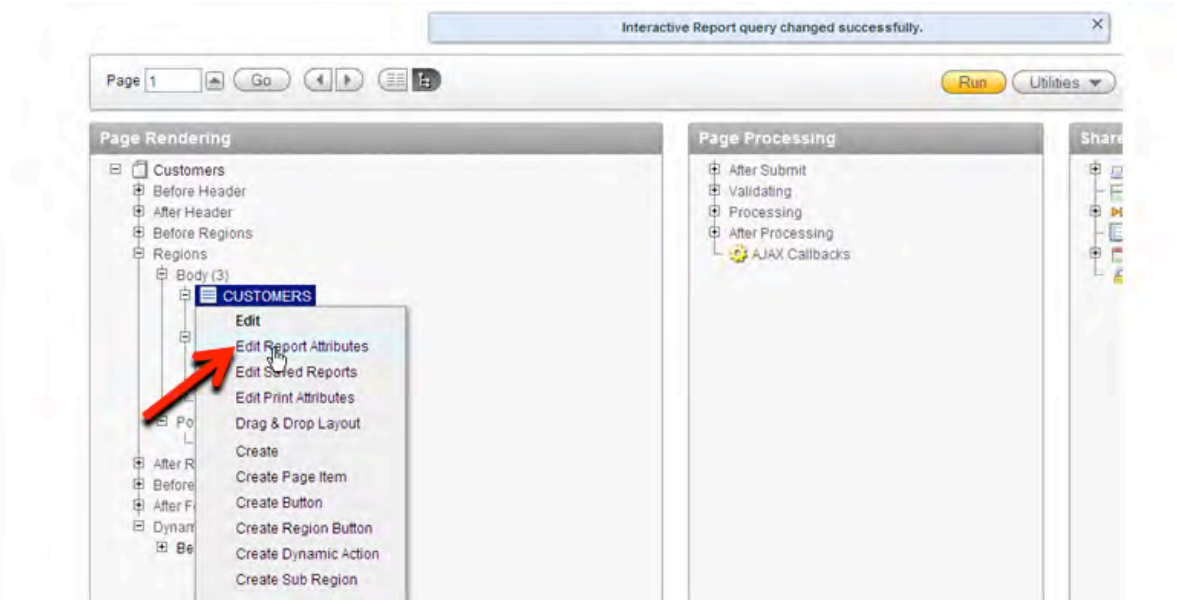
- Apply changes and "Run" the application again. This will now show the map with a list of customers on the map.
- At this time, there is no link between the report and map displayed on the page. Next we will add code to link the report and map display so that we can interact with each other.

Add interaction between the report and the map

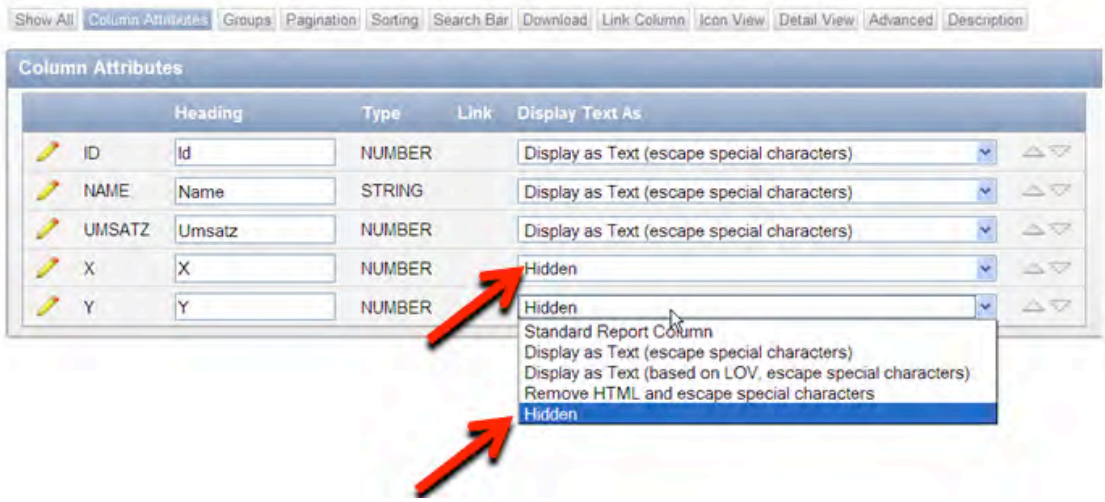
- Run the current application.
- Click on "Edit Page 1", and right click on "Customers" and "Edit".



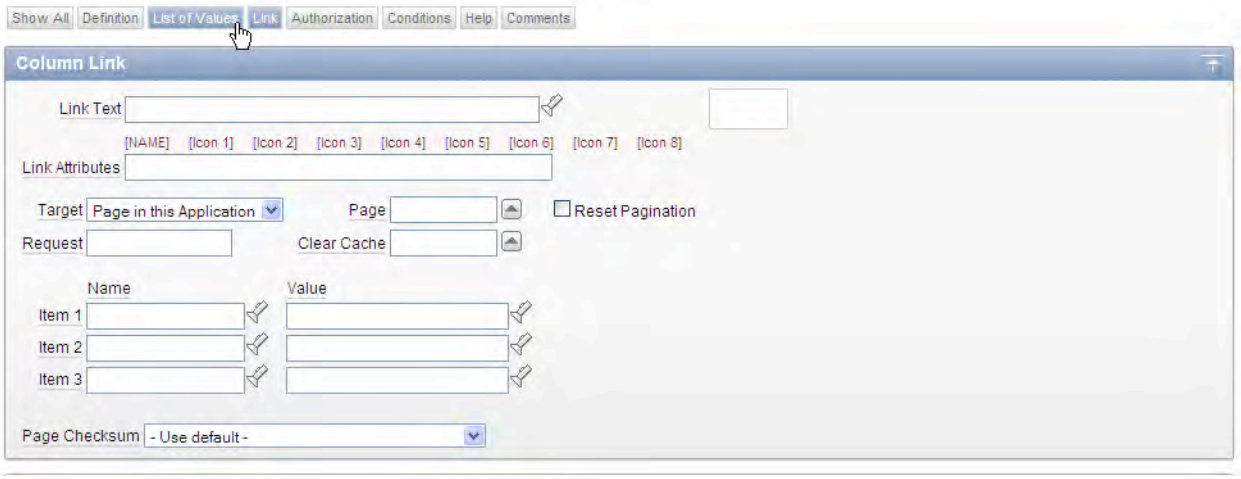
- In the "Region Source" box, replace the existing SQL with this new SQL.
- select "NAME", "CITY", "COUNTY", "STATE", "LOCATION", "SALES","ACCOUNT_MGR", round(c.location.sdo_point.x * 10000) x, round(c.location.sdo_point.y * 10000) y from "CUSTOMERS" c
- Click "Apply Changes" and again click "Apply Changes" on the next screen.
- Next we need to edit the report attributes to make the X and Y hidden.



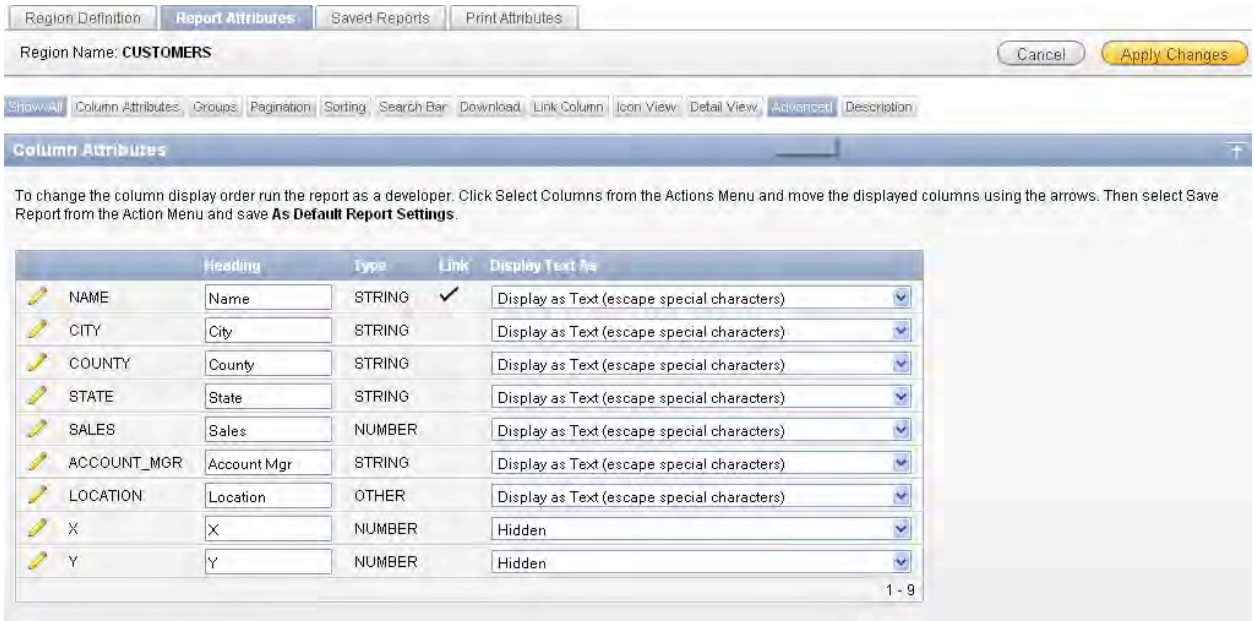
- Change the X and Y to be hidden columns.



- Next, make a link between the map and the report using the "NAME" column of the report.
- Click on the icon next to the "NAME" attribute and it will take you to this screen.

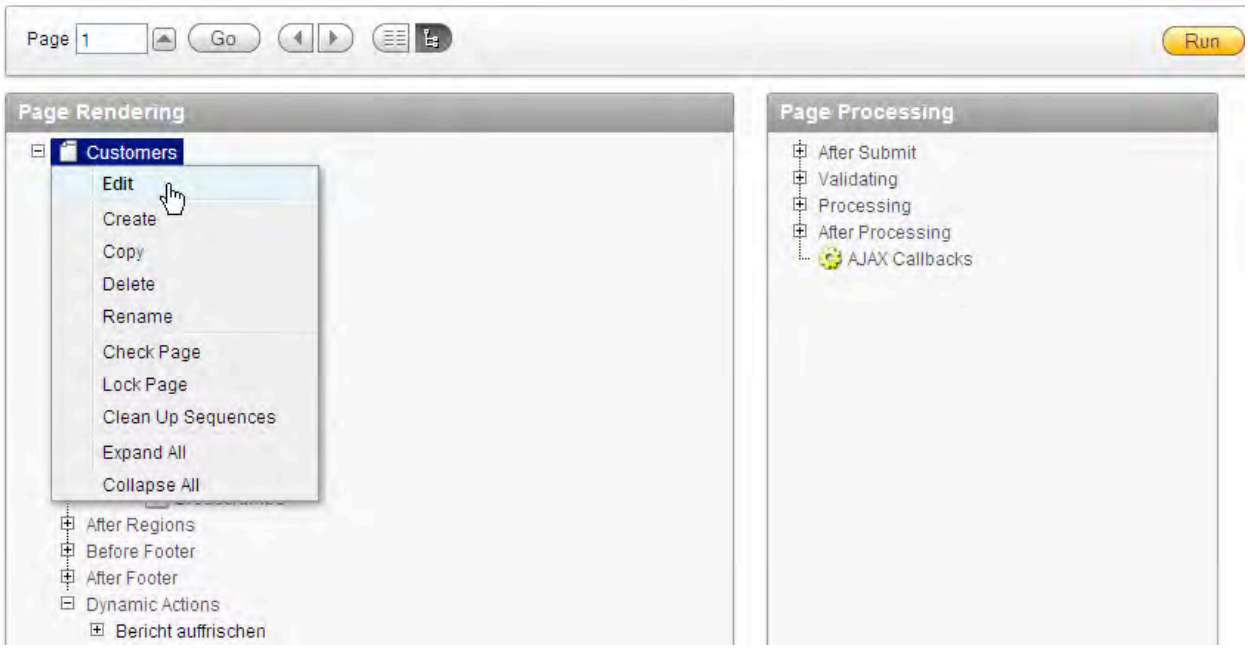


- Click on "Link" tab and add "#NAME#" for the "Link Text" attribute.
- For the "Target" pick "URL" and it will add a URL text box to the page.
- In the URL text box, add: **JavaScript:setMapCenter(#X#, #Y#);**
- Now you will be back to this screen with NAME checked.



- Click "Apply Changes".

- Edit the customer page again to edit the java script code to center the map to the X and Y passed in from the report.



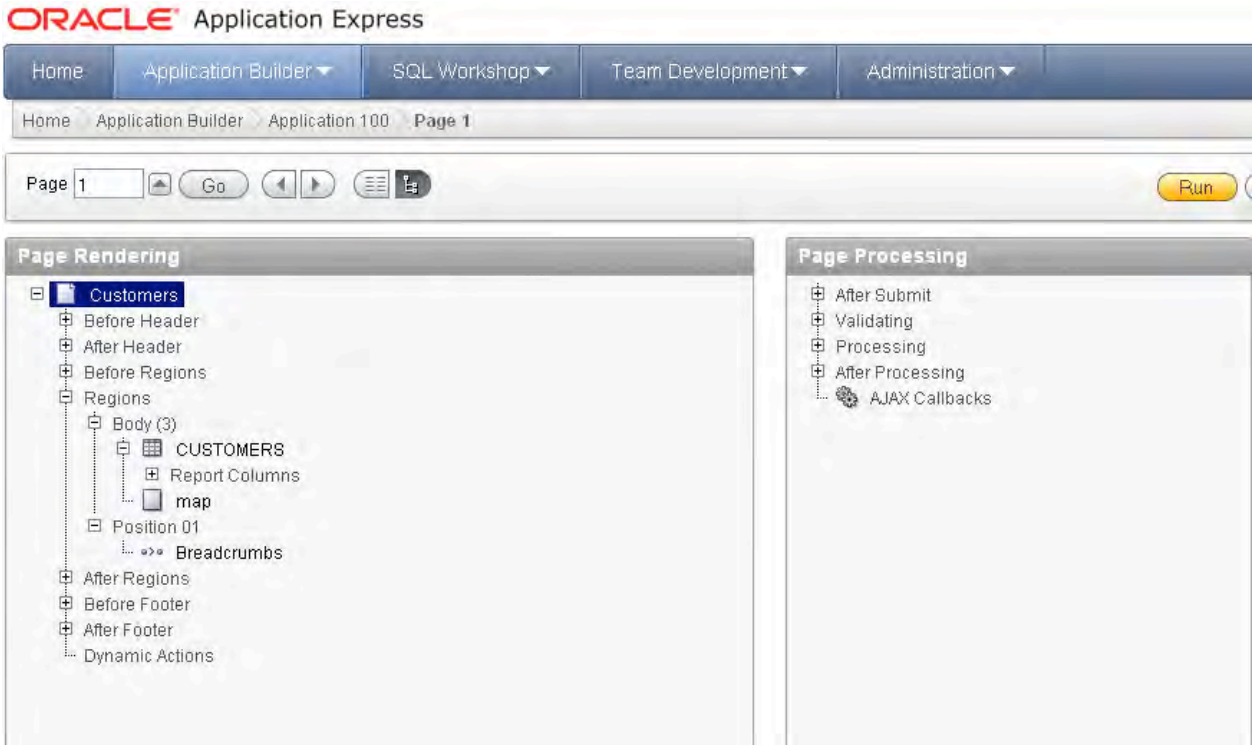
- Click on the JavaScript tab and add the following code to the end of the JavaScript code that is already there.

```
function setMapCenter(x, y) {
  mapview.setCenterAndZoomLevel(
    MVSdoGeometry.createPoint(x / 10000, y / 10000, 8307),     11  );
}
```

- Click on "Apply Changes" and run the application.
- Now if you click on a row on the report, the map will center to the location of the clicked row.

Add a theme to display hotels near a customer

- Run the current application.
- Click on "Edit Page 1" at the bottom of the screen and then right click on "Customers" and pick "Edit."'



- Go to the "JavaScript" tab and edit the java script code to add a new theme to the map. This new theme is called "hotels". This theme is defined with bind variables so that at run time we can pass in query parameters to it and the results based on the bind variables are displayed on the map. The query for the theme looks like this:

  select NAME, ACTUAL_ADDRESS from

  MAP_POI_HOTEL

  where sdo_within_distance(GEOMETRY, sdo_geometry(2001, 8307,

  sdo_point_type(:long, :lat,null),null,null), 'distance=3000 unit=meter') = 'TRUE';

- At run time, the X and Y we get from the selected customer is passed to this theme and we query for all hotels which are within 3KM from this location.
- Now replace the JavaScript that is already there with this new code.

```
var eLocURL = "http://elocation.oracle.com/mapviewer/mcserver";
var baseURL = "http://local-mapviewer-host/mapviewer";
var mapview;
var nearestHotelsTheme;
 function display_map() {
     mapview = new MVMapView(document.getElementById("map"), baseURL);
  mapview.addMapTileLayer(new MVMapTileLayer("elocation_mercator.world_map",
eLocURL));
     mapview.setCenter(MVSdoGeometry.createPoint(-122, 38, 8307));
     mapview.setZoomLevel(6);
     mapview.addNavigationPanel("east");
     var foi = new MVThemeBasedFOI("custfoi", "mvdemo.CUSTOMERS");
     foi.enableAutoWholeImage(true);
     foi.setMinVisibleZoomLevel(5);
     mapview.addThemeBasedFOI(foi);
     mapview.display();
 }
function setMapCenter(x, y, name) {
 if(nearestHotelsTheme)
   {
      mapview.removeThemeBasedFOI(nearestHotelsTheme);
   }
   nearestHotelsTheme = new MVThemeBasedFOI('hotels-theme',  'mvdemo.hotels');
   //set the binding variable for this theme.
   var x1 = x/10000;  // :1
   var y1 = y/10000;  // :2
```

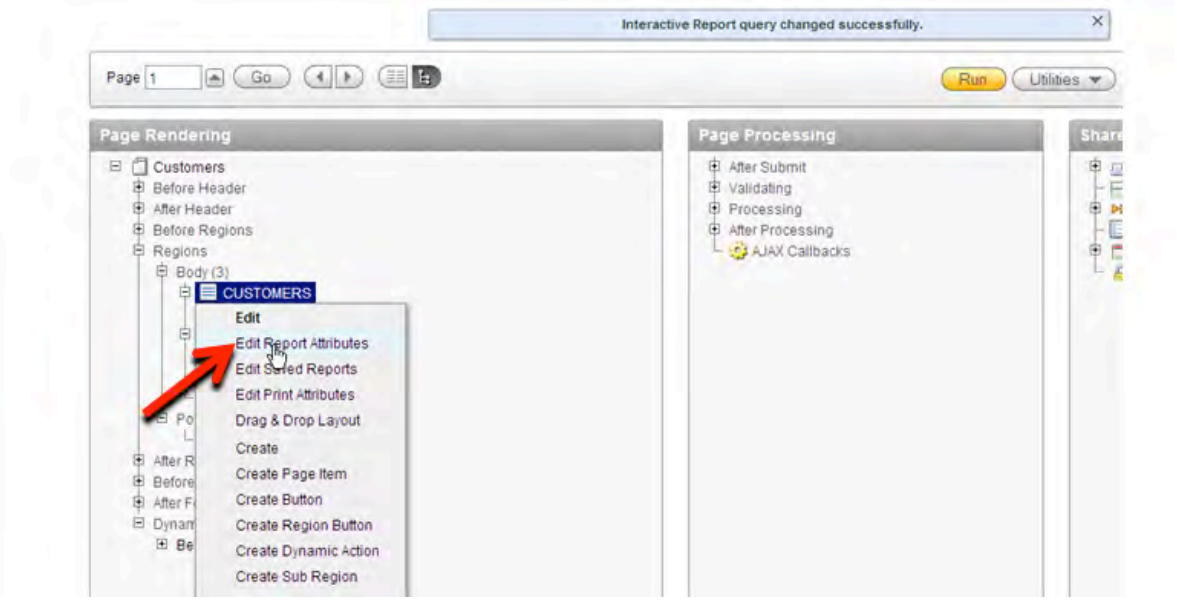
```
    nearestHotelsTheme.setQueryParameters(x1,y1);

    nearestHotelsTheme.setBringToTopOnMouseOver(true);
    mapview.addThemeBasedFOI(nearestHotelsTheme);
  mapview.setCenterAndZoomLevel(
    MVSdoGeometry.createPoint(x/10000, y/10000, 8307), 12);
}
```

- Click on "Apply Changes" and "Run" the application. This should now center the map when you click on a row in the report and also add a new theme showing the hotels within 3K from the selected customer location.

Highlight the selected customer

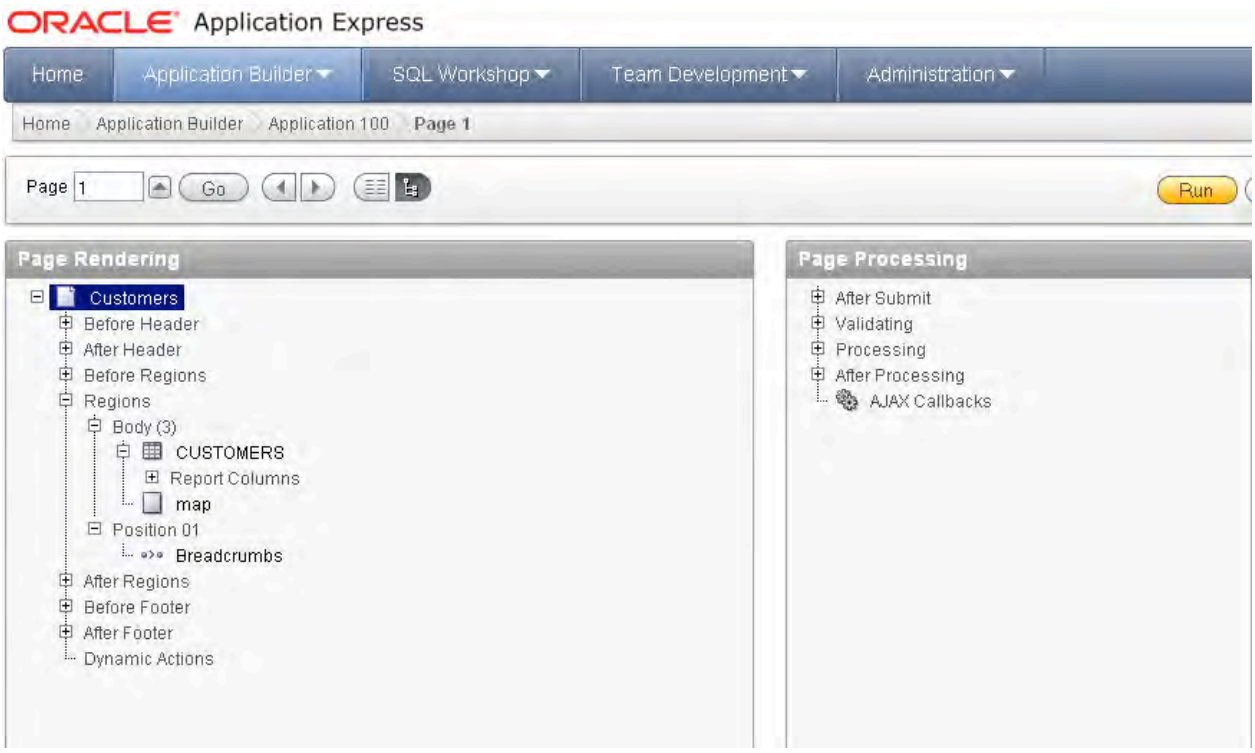
- When you select a customer on the map, the hotels near that customer are displayed on the map. But there is no way to tell which customer you selected by looking at the map. We now want to highlight the selected customer on the map so that the selected customer icon on the map changes to a different color.
- Run the application.
- Click on "Edit Pages 1" again and on the next screen, pick "Customers" from the "Regions" and pick "Edit Report Attributes." (This is similar to Step 6.5.)



- Click on the icon next to "NAME" and click on the "Link" tab.
- In the URL text box, replace what is there with this new code.

JavaScript:setMapCenter(#X#, #Y#,'#NAME#');

- We are now going to pass the NAME of the customer we picked to the java script function. Then we can pass that to a theme to highlight this particular customer on the map.
- Click "Apply Changes" and "Apply Changes" and this will bring you back to the main page.
- Now edit the "Customers" at the top of the list again.



- Go to the "JavaScript" tab and edit the code there.
- At the top, add this declaration.

```
var selectedCustomer;
```

- And replace the setMapCenter function with this new code.

```
function setMapCenter(x, y, name) {
```

```
if(nearestHotelsTheme)
  {
     mapview.removeThemeBasedFOI(nearestHotelsTheme);
  }
  nearestHotelsTheme = new MVThemeBasedFOI('hotels-theme',  'mvdemo.hotels');
  //set the binding variable for this theme.
  var x1 = x/10000;  // :1
  var y1 = y/10000;  // :2
  nearestHotelsTheme.setQueryParameters(x1,y1);
  nearestHotelsTheme.setBringToTopOnMouseOver(true);
  mapview.addThemeBasedFOI(nearestHotelsTheme);
if(selectedCustomer)
 {
   mapview.removeThemeBasedFOI(selectedCustomer);
 }
  selectedCustomer = new MVThemeBasedFOI('cust-theme',  'mvdemo.customer_byname');
  //set the binding variable for this theme.
  selectedCustomer.setQueryParameters(name);
  mapview.addThemeBasedFOI(selectedCustomer);
mapview.setCenterAndZoomLevel(
  MVSdoGeometry.createPoint(x/10000, y/10000, 8307), 12);
}
```

- Click on "Apply Changes" and then "Run" the application again.
- Now when you click on a row in the report, the corresponding customer icon changes to red in the map, in addition to displaying the nearest hotels.

APPENDIX

These are the themes used in building the sample applications using APEX.

1. CUSTOMER_BYNAME theme is based on the CUSTOMERS table.

```
<?xml version="1.0" standalone="yes"?>
<styling_rules>
  <hidden_info>
    <field column="NAME" name="NAME"/>
    <field column="SALES" name="SALES"/>
  </hidden_info>
  <rule>
    <features asis="true" style="M.STOPLIGHT_RED"> (name = :name) </features>
    <label column="NAME" style="T.COUNTRY_NAME_11"> 1 </label>
  </rule>
</styling_rules>
```

2. CUSTOMERS theme is based on the CUSTOMERS table.

```
<?xml version="1.0" standalone="yes"?>
<styling_rules>
  <hidden_info>
    <field column="NAME" name="NAME"/>
    <field column="SALES" name="SALES"/>
  </hidden_info>
  <rule>
    <features style="M.STOPLIGHT_GREEN"> </features>
    <label column="NAME" style="T.COUNTY_NAME"> 1 </label>
  </rule>
</styling_rules>
```

3. HOTELS theme is based on the MAP_POI_HOTEL table.

```
<?xml version="1.0" standalone="yes"?>
```

```
<styling_rules>
  <hidden_info>
    <field column="NAME" name=""/>
    <field column="ACTUAL_ADDRESS" name=""/>
  </hidden_info>
  <rule>
    <features asis="true" style="M.CUSTOMERS">
(sdo_within_distance(GEOMETRY, sdo_geometry(2001, 8307,
sdo_point_type(:1, :2,null),null,null), 'distance=3000') = 'TRUE') </features>
    <label column="NAME" style="T.S13_CITIES_L1_STREETS"> 1 </label>
  </rule>
  </rule>
</styling_rules>
```

Oracle Application Express Plug-in for Ajax Maps: Map Integration the Easy Way
March 2014
Author: Siva Ravada, LJ Qian

Oracle Corporation
World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200

oracle.com

Oracle is committed to developing practices and products that help protect the environment

Hardware and Software, Engineered to Work Together