



Using MapViewer Heatmap in OBIEE

Chris Hughes, Senior Sales Consultant, Oracle Public Sector



- Chris Hughes is a Senior Sales Consultant for the Oracle Public Sector organization supporting clients and prospects in evaluating and deploying Oracle solutions.
- He has had multiple roles in Public Sector Sales and IT over the past 20 years and acts as a Spatial Technical Evangelist within the Oracle and Business Intelligence communities.
- He is a two time graduate of the Pennsylvania State University

A. Review Existing Technology

- Standard OBIEE Mapping component
- Methodologies for integrating Javascript APIs in OBIEE
- Newer analytical features in Oracle Maps HTML5 API

B. Step By Step: Heatmap in OBIEE

- Build a working heatmap in OBIEE using Answers and Oracle Maps HTML5 API
- Making Cupcakes!



1. Understand Spatial Visualization in OBIEE
2. Learn the new features of Oracle Maps for analytics
3. Understand some of the integration points for OBIEE
4. Learn a Recipe

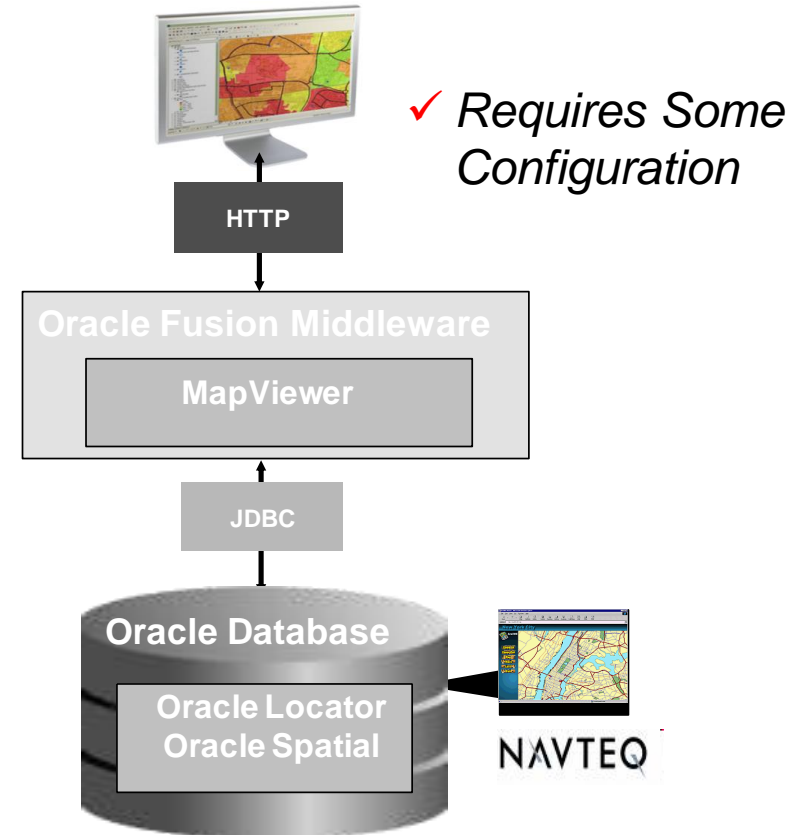
Using MapViewer Heatmap in OBIEE



A. REVIEW EXISTING TECHNOLOGY

Out of the Box OBIEE Map Component

- **Oracle Locator:** Feature of Oracle Database XE, SE, EE
- **Oracle Spatial:** Priced option to Oracle Database EE
- **MapViewer:** Java application and map rendering feature of Oracle Fusion Middleware
- **Workspace Manager:** Long transactions feature of Oracle Database SE, EE
- **Bundled Map Content:** Major roads, administrative boundaries (city, county, state, country) - worldwide coverage from Navteq

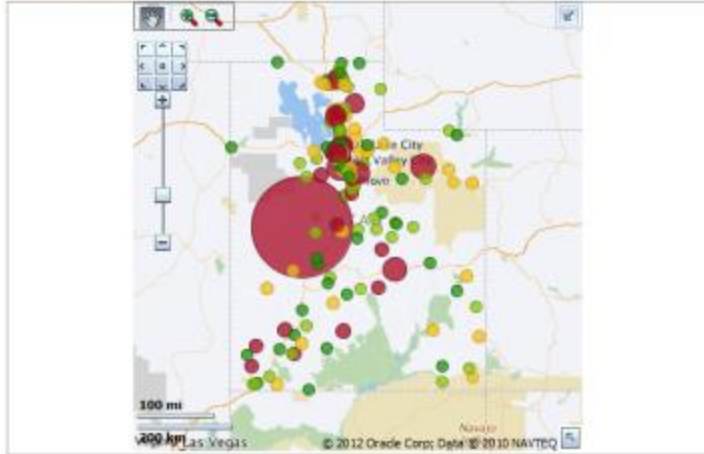


OBIEE Map Component



ORACLE Business Intelligence

Top 10 Fatal Crash Areas by Year
crossTab Crashes Points



[Refresh](#) - [Print](#) - [Export](#) - [Add to Briefing Book](#) - [Copy](#)

Rt Dir Id	Route Fullname	2005		2006 Total		2007		2007 Total		2008		2008 Total		2009		2009 Total		2010		2010 Total		2011		2011 Total		Num Crashes
		Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	Num Crashes	
310.00	US 40	31	12	43	22	8	30	16	8	24	14	10	24	20	4	24	12	3	24	12	3	15	160			
318.00	SR 71	29	3	32	34	2	36	23	1	24	8	2	10	18	3	21	12	3	21	12	3	15	138			
321.00	I-80 EB	54	15	69	35	14	49	36	23	59	30	11	41	23	7	30	23	20	30	23	20	33	281			
403.00	US 189	24	4	28	26	5	31	13	6	19	13	1	14	5	2	7	11	1	12	11	1	12	111			
418.00	I-215 CW	27	8	35	23	3	26	17	5	22	10	3	13	12	2	14	7	5	12	7	5	12	122			
1332.00	US 6	28	6	34	19	6	25	10	5	15	16	6	22	13	10	23	11	3	14	11	3	14	133			
1333.00	I-15 NB	217	31	248	165	37	202	128	35	163	126	32	158	112	29	141	89	29	118	89	29	118	1030			
1368.00	SR 68	25	4	29	29	4	33	26	6	32	22	2	24	19	4	23	24	7	31	24	7	31	172			
1370.00	I-70 EB	27	5	32	21	6	27	18	7	25	5	8	13	15	9	24	16	6	22	16	6	22	143			
1384.00	US 89	136	24	160	126	18	144	203	12	115	79	16	95	66	20	76	69	8	77	69	8	77	667			

ORACLE Business Intelligence Search All [v] [x] Advanced Administration Help Sign Out

Administration Home Catalog Favorites Dashboards New Open Signed In As weblogic

Manage Map Data

Manage map components and associate geographic layers to BI data [Back]

Layers Background Maps Images [x] [v] [p]

Name	Description	Location
NEWER_MIDDLE_THEME		sdsud/NEWER_MIDDLE_THEME
SCHOOLBOARD_DIST		sdsud/SCHOOLBOARD_DIST
UDOT_DISTRICTS		udot/UDOT_DISTRICTS
UDOT_ROUTES		udot/UDOT_ROUTES
UTAH_COUNTIES		udot/UTAH_COUNTIES
UTAH_CRASHES		udot/UTAH_CRASHES
UTAH_ROADS		udot/UTAH_ROADS

Oracle Maps HTML5 API (QuickStart)



ORACLE MapViewer 11g Samples

Oracle Maps HTML5 Documentation

The Oracle Maps HTML5 Library is a modern JavaScript framework that helps building rich Internet mapping applications. It provides a HTML5 based JavaScript API that lets you quickly develop a rich mapping application based on a variety of map tile servers.

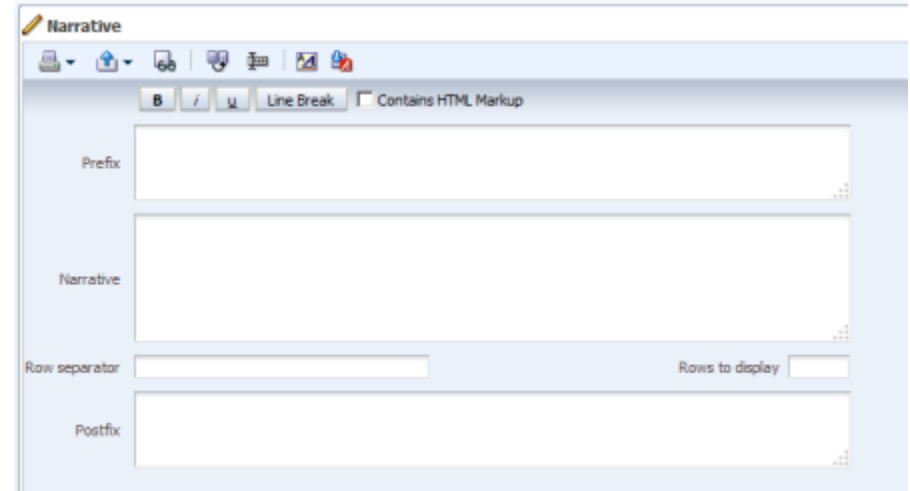
Theme Tips

1. Create a Sample that limits rows for Testing
2. Create a Production Layer with business driven Binding Variables
3. Remember order of variables

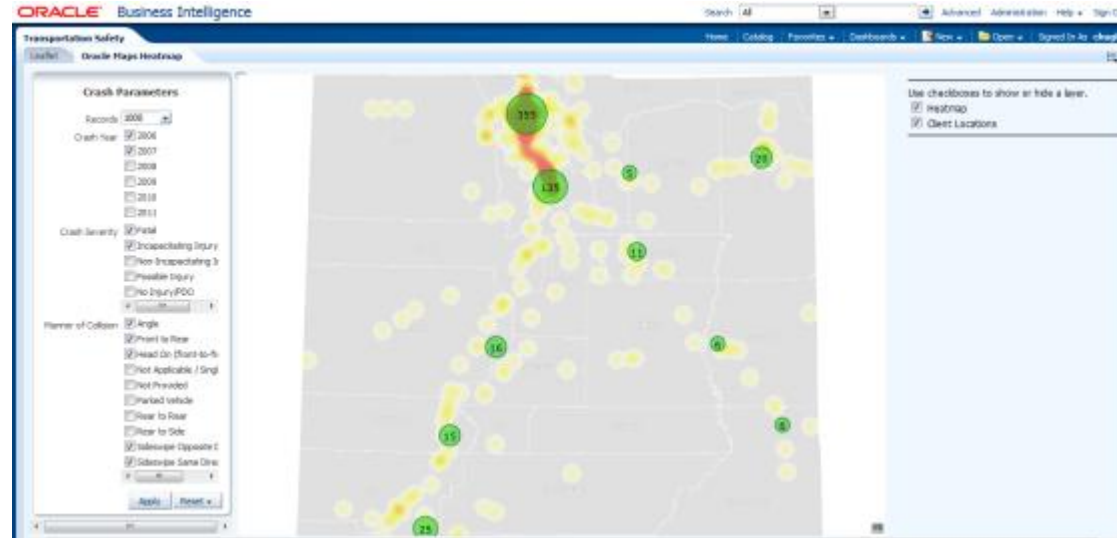
Easiest OBIEE JS Integration Points



- Narrative Component
- Dashboard Text Component
- Embedded (IFrame)
- Prefer Dashboard Text component
 - Pass in variables
 - easier to work with
 - can code in complete HTML and JS in an IDE



Using MapViewer Heatmap in OBIEE



B. STEP BY STEP: HEATMAP IN OBIEE

Problem Definition (why do this?)

1. Client wants to build a record detail map visualization
2. OBIEE works in aggregates (mainly)
3. MapViewer Oracle Maps v1 FOI system requires some babysitting in OBIEE (embedded in OBIEE widget)
4. MapViewer Oracle Maps v1 FOI system has a practical limit of 100-200 objects



Steps in Building a Heatmap in OBIEE



1. Define Your Geometry Themes (one test, one prod, use parameters)
2. Configure the MapViewer 11.1.1.7 JSON dataservice
3. Troubleshooting the dataservice
4. Deploy analyticsRes add libraries and HTML test apps
5. Test using Leaflet consuming the Mapviewer data
6. Build the HTML/JS Framework as Text in Dashboard
7. Build Dashboard Prompt in OBIEE
8. Add parameters to the JS code (matching geometry theme)
9. The cupcakes are done!
 - Add OBIEE Content as Desired (sprinkles)



1) Use Map Builder to Define Your Geometry Themes

Oracle Map Builder

File Edit View Tools Window Help

Connection: udot

SAMPLE_CRASHES

Name: SAMPLE_CRASHES

Description:

Theme Options

Basic Information

Styling Rules

Advanced

Other Properties

Custom Tags

Styling Rules

Rendering

Labeling

Columns: CITY,CITY_CODE,COUNTY_ID,CRAS...

Style: M.CIRCLE.GREEN.15

Query: (rownum < NVL(:1,1200))

Column:

Style:

Function: -1

Editor XML Preview

MapViewer Scale (per screen inch): 147,487.76 Ratio Scale: 1:5,806,605 (X: -108,294.984122) (Y: 4,080,559.905612)

Theme Tips

1. Create a Sample that limits rows for Testing
2. Create a Production Layer with business driven Binding Variables
3. Remember order of variables

Production Query (CRASH_SUM_FILTERED)

```
(CRASH_SEVERITY IN (select column_value
from Table(:Sev)) AND CRASH_YEAR IN
(select column_value from Table(:Year))
AND rownum < :limit)
```



2) Configure the Mapviewer Instance for dataserver

1

http://localhost:9704/mapviewer/faces/admin/admin.jspx

ORACLE MapViewer Administration Console 11g

Admin | Editor | Logout | Help

Home Management Meta data About

Manage MapViewer

- Configuration
- Datasources
- Geometry Cache
- Create Tile Layer
- Manage Tile Layers
- Monitoring
- View Logs

TIP Edit mapViewerConfig.xml file

File location: D:\fmw\Oracle_BI1\bifoundation\jee\mapviewer.ear\web.war\WEB-INF\conf\mapViewerConfig.xml

Config:

```
(exclamation point), so that when MapViewer starts the next time, it will encrypt and replace the clear text password.
-->
<map_data_source name="udot"
  host="localhost"
```

Steps

1. Set the data source in MV configuration
2. Ensure data streaming is enabled in mds.xml

2

D:\fmw\Oracle_BI1\bifoundation\jee\mapviewer.ear\web.war\WEB-INF\conf\mds.xml

Organize Include in library Share with New folder

Name	Date modified
foi_proc.id	1/13/2014 6:45 PM
mapViewerConfig.xml	1/10/2014 5:54 PM
mds.xml	12/30/2013 11:54 PM
mv_proc.id	1/13/2014 6:45 PM

```
<data_source name="udot">
  <allow_predefined_themes>true
</allow_predefined_themes>
<allow>
  <theme>*</theme>
</allow>
</data_source>
```



3) Test and troubleshoot Mapviewer dataserver

Access the JSON Data Server here:

<http://localhost:9704/mapviewer/dataserver/<datasource>?help=y>

t: theme name (mandatory)

bbox: bounding box, such as `bbox=-122.0,24.6,-100,45`

bbox_srid: srid of the bounding box, if different from the data source (optional)

to_srid: data should be transformed into this SRID before rendering (optional)

dadp: all coordinates should have this number of digits after the decimal point (optional, default is 5)

include_style_info: if set to false, then no styling information is included in the response (optional)

sql: (only used when requesting dynamic query-based data) complete SQL query. (mandatory)

asis: (only used when requesting dynamic query-based data) query should be executed as-is by the server. (optional)

```
{"mds_error":
  {"message":"Data server
streaming request
validation error. Root
cause: This data source
does not allow streaming
access."},
  "details":"Check server
log for details."} }
```

**If you receive this message
you have not set the tags
needed in mds.xml and
restarted mapviewer**

3a) Additional Mapviewer datasever options



paramnum: specifies the number of binding variables (to be used for a pre-defined theme that has binding variables in its query cond) included in the request.

For each binding variable, you must supply the following with n starting from 1 through 'paramnum':

param<n>: the value of the binding variable. (mandatory)

paramtype<n>: used to specify an array type or a geometry type: narray (numeric array), sarray (string array), or geometry. (optional)

sqltype<n>: the sql type for the corresponding narray or sarray params, such as MV_NUMBERLIST or MV_STRINGLIST; not needed for any other param types. (optional)

workspace: specifies the workspace to use when retrieving data

For 12c (planned for this calendar year):

simplify: indicates whether geometry should be simplified

threshold: if simplify is true, this value specifies the reduction percentage (value must be 1 through 99).



4) Deploy analyticsRes and any libraries or apps you might want

The screenshot shows the Oracle WebLogic Server Administration Console interface. The browser address bar shows the URL: localhost:7001/console/console.portal?_nfpb=true&_pageLabel=WebApplicationOverviewPage&W. The page title is "ORACLE WebLogic Server® Administration Console".

The main content area displays the "Settings for analyticsRes" page. The "Overview" tab is selected. The settings are as follows:

Name:	analyticsRes	The name of this application deployment. More Info...
Context Root:	/analyticsRes	The specific path at which this web application is found by a servlet. More Info...
Path:	D:\fmw\instances\instance1\bifoundation\OracleBIPresentationServicesComponent\coreapplication_obips1\analyticsRes	The path to the source of the deployable unit on the Administration Server. More Info...
Deployment Plan:	(no plan specified)	The path to the deployment plan document on Administration Server. More Info...
Staging Mode:	nostage	The mode that specifies whether an application's files are copied from a source on the Administration Server to the Managed Server's staging area during application preparation. More Info...

Red circles with numbers 1, 2, 3, and 4 are overlaid on the screenshot to indicate key steps: 1. Points to the Path field. 2. Points to the Overview tab. 3. Points to the Domain Structure tree. 4. Points to the Settings for analyticsRes title.

Steps

1. Create a folder or war file with WEB-INF and proper deployment descriptors
2. Create a sample html for testing
3. Create a deployment in WLS console
4. Add libraries as needed (ensure you stop / start after adding files)



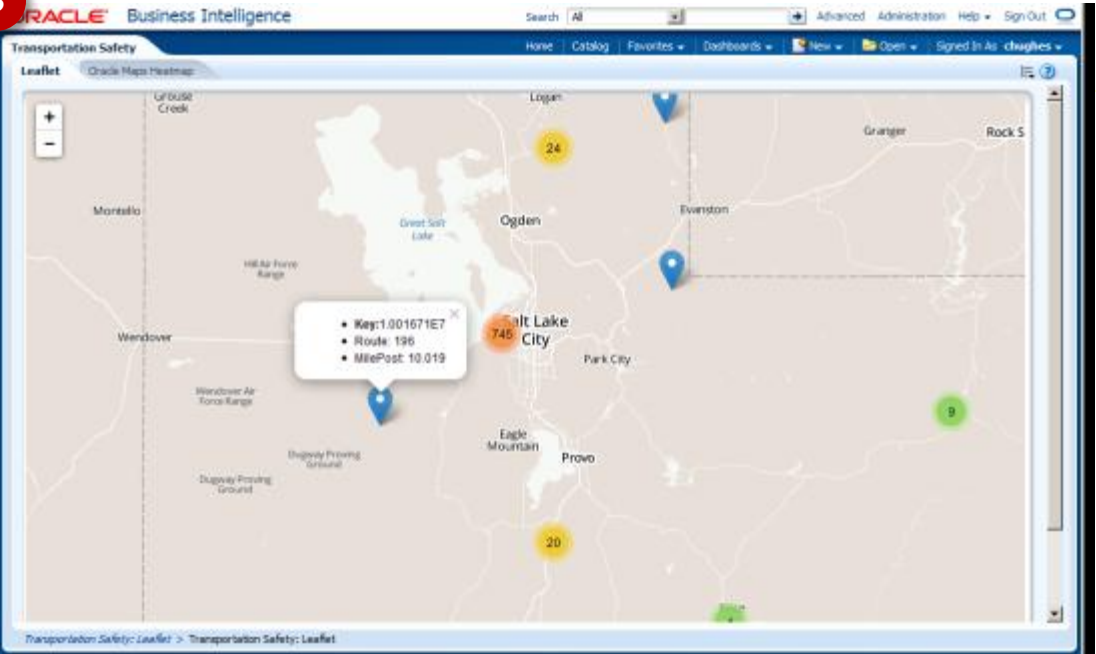
5) Test using Leaflet consuming the Mapviewer data

1 http://localhost:9704/mapviewer/dataserver/udot?t=SAMPLE_CRASHES&to_srid=8307¶mnum=1¶m1=12

2

```
//handle json response  
function handleJson(data) {  
  for (var i=0; i<data.length; i++) {  
    var a = data[i];  
    var in = a["Route"];  
    var ma = a["MilePost"];  
    marker = new L.Marker(a["Route"], {  
      icon: new L.Icon.Default(),  
      title: a["Route"] + " at MilePost: " + a["MilePost"]  
    });  
    markers.addLayer(marker);  
  }  
  markers.addTo(map);  
  map.fitBounds(markers.getBounds());  
}
```

3



- ### Steps
1. Test your layer URL
 2. Write Simple HTML file
 3. Deploy in a Dashboard as a Text Component



6) Build the MapView HTML/JS Framework as Text in Dashboard

```
var tilelayer = null;
var heatlayer = null;

var center = new OM.geometry.Point(-122,38,8307);
var utcenter = new OM.geometry.Point(452962,4347062,26912);
```

```
var zoomLevel = 3;
$(document).ready(function()
{
```

4

```
limitPrm = @{{variables.pv_limit}}{5};
```

```
var map = new OM.Map(document.getElementById('map'),{mapviewerURL: baseUrl});
tilelayer = new OM.layer.OSMTileLayer("tilelayer");
//local tile if needed
//tilelayer = new OM.layer.TileLayer("udotbase", {Source:"udot". tileLayer:"LOCALTILELAYER", tileServerUrl: baseUrl
```

```
map.addLayer(tilelayer) ;
```

```
vectorlayer1 = new OM.layer.VectorLayer("vectorlayer1");
vectorlayer1.setLabelsVisible(false);
vectorlayer1.enableClustering(true,{clusterStyle:circle});
vectorlayer1.setQueryParameters(sevPrms, yearPrms );
map.addLayer(vectorlayer1) ;
```

```
heatlayer = new OM.layer.VectorLayer("heatlayer",
{ def: layerdef, renderingStyle: new OM.style.HeatMap,
  spotlightRadius:12,
  colorStops:["#ffffff", "#ffff00", "#ff8800",
  opacity:0.5
```

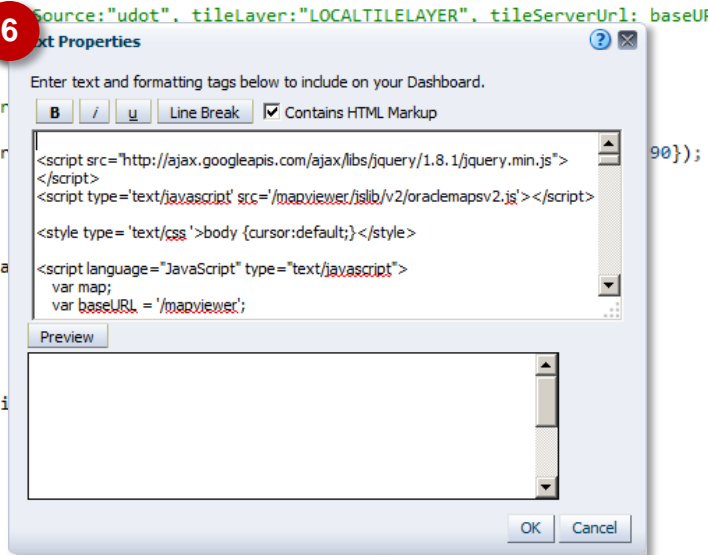
5

```
});
heatlayer.setQueryParameters(sevPrms, yearPrms, limitPrm
```

```
map.addLayer(heatlayer) ;
map.setMapCenter(utcenter);
map.setMapZoomLevel(zoomLevel) ;
map.init()
```

```
</script>
```

6



Steps

1. Use IDE of your Choice
2. Follow existing MVDemo samples
3. Start with your test Theme
4. Add your parameters as OBIEE substitution strings
5. Add Two layers
6. Copy and paste into Dashboard Text



7) Add Presentation Variables to Your Dashboard Prompt

Definition

Add prompts for users when they run this analysis.

Prompt Label
Page 1: Crash Parameters
Records
Crash Year
Crash Severity
Manner of Collision

Edit Prompt

Prompt for: Presentation Variable [dropdown] pv_severity

Label: Crash Severity

Description: Filter By Crash Severity

User Input: Check Boxes [dropdown]

Check Boxes Values: Specific Column Values [dropdown]

Column: "Crash"."Severity" [button: Select Column]

Fatal
Incapacitating Injury
Non-Incapacitating Injury
Possible Injury
No Injury/PDO

Options

Variable Data Type: Default (Text) [dropdown]

Limit values by All Prompts [dropdown]

Require user input

Default selection: Specific Values [dropdown]

Fatal
Incapacitating Injury

Check Boxes Width: Dynamic 120 Pixels

OK Cancel

Type	Prompt For	Description	Required	New Column
Page				
Variable value		Filter number of records for map layout		<input type="checkbox"/>
Variable value		Filter Data by Year		<input type="checkbox"/>
Variable value	"Crash"."Severity"	Filter By Crash Severity		<input type="checkbox"/>
Variable value	"Crash"."Collision Type"			<input type="checkbox"/>

Parameters Desired

1. Crash Severity
2. Crash Year
3. Limit of Num Recs

Display

Page: 1 [button]

Crash Parameters

Records: 1000 [dropdown]

Crash Year: 2006 2007 2008 2009 2010 2011



8) Add Parameters to JS Code on Dashboard

```
//parameter objects for calling mapviewer
var sevPrms = {value:["Fatal","Non-Incapacitating Injury"], type:"sarray", sqlType:"MV_STRINGLIST"};
var yearPrms = {value:[2007,2008], type:"narray", sqlType:"MV_NUMBERLIST"};
var manPrms = {value:["Head On","Angle"], type:"sarray", sqlType:"MV_STRINGLIST"};
var limitPrm = 1200;
```

```
var sevText, yearTxt, mannerTxt
```

```
var center = new OM.geometry.Point(-122,38,8307);
var utcenter = new OM.geometry.Point(452962,4347062,26912);
```

```
var zoomLevel = 3;
$(document).ready(function()
{
```

```
limitPrm = @{{variables.pv_limit}}{5};
sevText = "@{{variables.pv_severity}}{Fatal}";
yearTxt = "@{{variables.pv_year}}{2006,2007}";
mannerTxt = "@{{variables.pv_manner}}{RearEnd}";
```

```
yearPrms.value = yearTxt.split(',');
sevPrms.value = sevText.split(',');
manPrms.value = mannerTxt.split(',');
```

```
var map = new OM.Map(document.getElementById('map'),{mapviewerURL: baseUrl}) ;
tilelayer = new OM.layer.OSMTileLayer("tilelayer");
//tilelayer = new OM.layer.TileLayer("udotbase", {dataSource:"udot", tileLayer:"LOCALTILELAYER", tileServerUrl: baseUrl }

map.addLayer(tilelayer) ;
vectorlayer1 = new OM.layer.VectorLayer("vectorlayer1", { def:layerdef, renderingStyle: circleMarker});
vectorlayer1.setLabelsVisible(false);
vectorlayer1.enableClustering(true,{clusterStyle:circleMarker, minPointCount:5, maxClusteringLevel:8, threshold:90});
vectorlayer1.setQueryParameters(sevPrms, yearPrms, limitPrm);
map.addLayer(vectorlayer1) ;
```

Parameters Matched

1. Crash Severity
2. Crash Year
3. Limit of Num Recs



8a) Review Parameters posted from the Oracle Maps js client

Firebug - Oracle BI Interactive Dashboards - Transportation Safety

Console HTML CSS Script DOM Net Cookies YSlow

XHR	Clear	Persist	All	HTML	CSS	JavaScript	XHR	Images	Plugins	Media	Fonts
+	GET	Messages_e	404 Not Found	http	192.168.136.136:9704	1.1 KB	192.168.136.136:9704				9ms
+	GET	Messages_e	404 Not Found	http	192.168.136.136:9704	1.1 KB	192.168.136.136:9704				8ms
+	POST	mserver	200 OK	http	192.168.136.136:9704	3.0 KB	192.168.136.136:9704				14ms
+	POST	udot	200 OK	http	192.168.136.136:9704	30.5 KB	192.168.136.136:9704				444ms
-	POST	udot	200 OK	http	192.168.136.136:9704	30.5 KB	192.168.136.136:9704				704ms

Headers **Post** Response XML JSON Cache

Parameters application/x-www-form-urlencoded

```
bbox_srid 26912
include_label_box true
param1 "Fatal","Incapacitating Injury"
param2 2006,2007
param3 1000
paramnum 3
paramtype1 sarray
paramtype2 narray
refresh 71811
sqltype1 MV_STRINGLIST
sqltype2 MV_NUMBERLIST
t CRASH_SUM_FILTERED
to_srid 26912
```

Source

```
t=CRASH_SUM_FILTERED&paramnum=3&param1=%22Fatal%22%2C%22Incapacitating+Injury%22&sqltype1=MV_STRINGLIST
&paramtype1=sarray&param2=2006%2C2007&sqltype2=MV_NUMBERLIST&paramtype2=narray&param3=1000&include_label_box
=true&to_srid=26912&bbox_srid=26912&refresh=71811
```

+	POST	omserver	200 OK	http	192.168.136.136:9704	336 B	192.168.136.136:9704				15ms
-	POST	omserver	200 OK	http	192.168.136.136:9704	487 B	192.168.136.136:9704				17ms

Headers **Post** Response XML Cache



9) End Result with Heatmap and Clustered Symbols

Crash Parameters

Records: 1000 [dropdown]

Crash Year

- 2006
- 2007
- 2008
- 2009
- 2010
- 2011

Crash Severity

- Fatal
- Incapacitating Injury
- Non-Incapacitating Injury
- Possible Injury
- No Injury/PDO

Manner of Collision

- Angle
- Front to Rear
- Head On (front-to-front)
- Not Applicable / Single
- Not Provided
- Parked Vehicle
- Rear to Rear
- Rear to Side
- Sideswipe Opposite Direction
- Sideswipe Same Direction

[Apply] [Reset]



Use checkboxes to show or hide a layer.

- Heatmap
- Client Locations

Demonstrations



- Review prompts/parameters in JavaScript
- Firebug the XMLHTTP
- Lets Try and break the Demo!

Key Takeaways



- New Oracle Maps HTML5 API allows better and faster spatial data retrieval and manipulation
- Embedding other APIs in OBIEE is a good way to meet needs when OOTB OBIEE wont cut it
- Develop HTML/JS separately and then bring into OBIEE
- As always: Firebug (or Chrome Dev Tools) is your friend



CONCLUSION