Oracle Fusion Middleware 11g MapViewer: 
*Deconstructing a Simple BI Application*

Dan Abugov  
NAVTEQ  
Daniel.Abugov@NAVTEQ.com
NAVTEQ Maps are Everywhere

NAVTEQ is the leading provider of the digital map, traffic and location data that power many of the world’s most innovative navigation applications.

In-Dash

Telematics

Internet

Portable

Cell Phones

Trip Planning

Fleet

Enterprise

Government
NAVTEQ Oracle Delivery Format (ODF)

Turnkey delivery specialized for Oracle

- In a standard Oracle format
  - Storage, Indexing, and Analysis (*Oracle Data Type – SDO_GEOMETRY*)
  - Map Display (*Oracle Fusion Middleware MapViewer*)
  - Geocoding (*Oracle Spatial Geocoder*)
  - Routing (*Oracle Spatial Routing Engine*)

- Delivered in Oracle Transportable Tablespaces
  - Install pre-created tables and indexes in minutes

- Any Oracle-based application can be location-enabled
  - Much can be done with Locator
  - Add FMW MapViewer for visualization

- MapViewer-based application: Low cost of deployment
NAVTEQ – Oracle Delivery Format (ODF)

Multiple Regions
“Simple” BI Application
BI Application Software and Content

Uses standard components from Oracle

- Database
- Application Server: MapViewer/Oracle Maps – JSP interface
- Geocoder to assign location associated with text address (Spatial Option of Enterprise Edition)

Content from NAVTEQ in Oracle Delivery Format (ODF)

- Mapping
- Geocoding
- INEGI Demographics
BI Application Options

- Add a Map to a Web App
- Heat Maps
- Zoom and Pan Controls
- Layer Controls
- Population Demographics
- Buffers
- Population Calculations
- Sales Mix
- Quarterly Sales
- Locator/Spatial Overview
- MapViewer Overview
- Map Builder
Heat Maps

- Heat map which changes to show icons when zoomed in
Start with Container Theme

- MapViewer Requirement
  - Generalized country boundaries (WOM AREA COUNTRY GEN)
Start with Container Theme

- No drawing style (C.CLEAR)
- Only uses Mexico for this application (name='MEXICO')
Use Advanced Style for the Heat Map

- Advanced Style
  - The container theme is set here
  - Other parameters for smooth drawing depending on density of data
Now Create the Theme

- Retailers table, geometria column
Now Create the Theme

- Style is the heat map style just created
How is this Coded?

Define the theme-based feature of interest layers

**Javascript**

```javascript
themebasedfoiallheat = new MVThemeBasedFOI('themebasedfoiallheat','odf_gbl.Z_MEXBIZHEAT');
themebasedfoiallheat.enableAutoWholeImage(true);
themebasedfoiallheat.setMaxVisibleZoomLevel(FOIHeatMaxZoom+2);
mapview.addThemeBasedFOI(themebasedfoiallheat);
themebasedfoiallheat.setVisible(false);

themebasedfoiall = new MVThemeBasedFOI('themebasedfoiall','odf_gbl.ZMEXICORETAILERS');
themebasedfoiall.setBringToTopOnMouseOver(true);
themebasedfoiall.setMinVisibleZoomLevel(FOIHeatMaxZoom+3);
mapview.addThemeBasedFOI(themebasedfoiall);
themebasedfoiall.setVisible(false);
```
How is this Coded?

**HTML**

```html
<input type="checkbox" name="allretailer" value="AllRetailers" id="allRetail" onclick="javascript: allFOI(this);">All Retailers</input>
```

**Javascript**

```javascript
function allFOI(checkBox)
{
    if (checkBox.checked)
    {
        themebasedfoiall.setVisible(true);
        themebasedfoiallheat.setVisible(true);
    }
    else
    {
        themebasedfoiall.setVisible(false);
        themebasedfoiallheat.setVisible(false);
    }
}
```
Zoom and Pan Controls
**Zoom and Pan Controls**

**HTML**

```
<font face="Arial, Helvetica, sans-serif" size=2><b>Map controls</b></font>

<input type="radio" name="zoom" value="Zoom" id="Zoom" onclick="javascript: marqueeZoomCustomStyle();" value="Zoom">Zoom

<input type="radio" name="zoom" value="Pan" id="Pan" checked onclick="javascript: noMarqueeZoom();" value="Pan">Pan
```
Zoom and Pan Controls

Javascript

function marqueeZoom()
{
    mapview.stopMarqueeZoom();
    mapview.startMarqueeZoom(mode, new Object());
}

function noMarqueeZoom()
{
    mapview.stopMarqueeZoom();
}
Zoom and Pan Controls

Javascript

function marqueeZoomCustomStyle() {
    var rectStyle = new Object();
    rectStyle.borderColor = "#0000FF";
    rectStyle.borderWidth = 2;
    rectStyle.borderDashed = false;
    rectStyle.fillColor = "#FF0000";
    rectStyle.fillOpacity = 0.5;
    mapview.stopMarqueeZoom();
    mapview.startMarqueeZoom('continuous', rectStyle);
}
Layer Controls

- Walmart
- Comercial Mexicana
- Pop. Demographics
- All Retailers

Location Analytics with NAVTEQ
Layer Controls

Same for theme-based feature-of-interest (FOI) and dynamic (JDBC) FOI

HTML

<input type="checkbox" name="popdemog" value="PopDemographics" id="popDemog" onclick="javascript: popFOI(this);">Pop. Demographics

Javascript

themebasedfoipop = new MVThemeBasedFOI('themebasedfoipop','odf_gbl.Z_MEX_AGE_THEME');
mapview.addThemeBasedFOI(themebasedfoipop);
themebasedfoipop.setVisible(false);
Layer Controls

Javascript

function popFOI(checkBox)
{

    if (checkBox.checked)
    {
        themebasedfoipop.setVisible(true);
        mdpoptheme.setVisible(true);
    }
    else
    {
        themebasedfoipop.setVisible(false);
        mdpoptheme.setVisible(false);
    }
}

Population Demographics

NAVTEQ BI Maps in Oracle Tutorial

[Map with demographics data]

Population: 14837

- Male Pop: 7183
- Female Pop: 7654

© 2009 NAVTEQ™ and © 2009 Oracle Corp.
Population Demographics

Create advanced style
Map Builder has automated tools to build
Population Demographics

Create theme
- Assign table and column to render
Population Demographics

Create theme

1. Assign column for thematic map and value when clicked
2. Assign advanced style
3. Assign label for flyover
Population Demographics

Create theme

- Assign columns for clickable values
Population Demographics

HTML

<input type="checkbox" name="popdemog" value="PopDemographics" id="popDemog" onclick="javascript: popFOI(this);">Pop. Demographics</input>

Javascript

// theme

themebasedfoipop = new
MVThemeBasedFOI('themebasedfoipop','odf_gbl.Z_MEX_AGE_THEME');

themebasedfoipop.setMaxWholeImageLevel(16);

mapview.addThemeBasedFOI(themebasedfoipop);

themebasedfoipop.setVisible(false);

// legend

mdpoptheme=new MVMapDecoration("<img src="http://localhost:8888/mapviewer/omserver?ds=ODF_GBL&sty=V.MEXICO_POP&w=100&h=75">",0.01,
0.65,0,0);

mdpoptheme.setOffset(0,50)

mdpoptheme.setVisible(false);

mapview.addMapDecoration(mdpoptheme)
Layer Control for Population Demographics

Javascript

function popFOI(checkBox)
{
    if (checkBox.checked)
    {
        themebasedfoipop.setVisible(true);
        mdpoptheme.setVisible(true);
    }
    else
    {
        themebasedfoipop.setVisible(false);
        mdpoptheme.setVisible(false);
    }
}
Buffers
Buffers

- Shown as dynamic feature of interest (FOI) layers
  - No pre-created theme or style definitions

HTML

Walmart Buffer Distance: <input type="text" id="buffer1Distance"
value="1" size="2" maxlength="3" > km
<br>
<input type="button" name="updateBuffer1" value="Update" onclick="updateBuffer1()">
<input type="button" id="showHideBuffer1" value="Show Buffer" onclick="toggleBuffer1()">
Buffers

Javascript

function updateBuffer1()
{
    var radius = document.getElementById('buffer1Distance').value;
    bufferStyle = "ODF_GBL:C.TRANSPARENT_BLUE";
    mapview.removeThemeBasedFOI(bufferthemel);
    baseQuery = "select sdo_geom.sdo_buffer(A.geometria, "+radius+", 0.005, 'unit=km arc_tolerance=0.005') location " +
    " from retailers A where (cadena = 'WALMART') " +
    " and
    sdo_filter(geometria,sdo_geometry(2003,8307,null,sdo_elem_info_array(y(1,1003,3),"+"
    " sdo_ordinate_array(?,?,?,?)),'querytype=window')='TRUE'";
Buffers

Javascript (continued)

```javascript
var theme = '/themes/<theme name="JDBC_THEME" >' +
    '<jdbc_query asis="true" spatial_column="location"
jdbc_srid="8307" ' +
        'render_style="'+bufferStyle+'"
datasource="odf_gbl">" + baseQuery +
        '</jdbc_query></theme></themes>' ;
buffertheme1 = new MVThemeBasedFOI('buffertheme1',theme);
buffertheme1.setBringToTopOnMouseOver(true);
mapview.addThemeBasedFOI(buffertheme1);
// place the buffer theme beneath the customer theme
mapview.setThemeIndex(buffertheme1, 1) ;
// buffer visible so set button to hide
document.getElementById('showHideBuffer1').value='Hide Buffer';
buffer1IsVisible = true ;
```
Buffers

Javascript (continued)

function toggleBuffer1() {
    if(!buffer1IsVisible) {
        buffer1IsVisible = true;
        document.getElementById('showHideBuffer1').value='Hide Buffer';
        buffertheme1.setVisible(true);
    } else {
        buffer1IsVisible = false;
        document.getElementById('showHideBuffer1').value='Show Buffer';
        buffertheme1.setVisible(false);
    }
}
Population Calculation
Population Calculation

- Style definition for thematic display
Population Calculation

- Style definition for markers in thematic map
Population Calculation

- Shown as dynamic feature of interest (FOI) layers
  - Uses pre-created style definitions

**HTML**

Walmart Buffer Distance: `<input type="text" id="buffer1Distance" value="1" size="2" maxlength="3" > km`

`<input type="button" id="showHidePop1" value="Show Pop." onclick="togglePop1()">`

**Javascript for legend**

```
mdpopjdbctheme=new MVMapDecoration("<img src="http://localhost:8888/mapviewer/omserver?ds=ODF_GBL&sty=V.Z_MEX_AVG_POP_RETAIL&w=85&h=100"">",0.01,0.42,0,0);`n
    mdpopjdbctheme.setOffset(0,50)`n`    mdpopjdbctheme.setVisible(false);`n`    mapview.addMapDecoration(mdpopjdbctheme)`n```
Population Calculation

Javascript

function togglePop1()
{
    if(!pop1IsVisible)
    {
        var radius = document.getElementById('buffer1Distance').value;
        bufferStyle = "ODF_GBL:V.Z_MEX_AVG_POP_RETAIL";
        mapview.removeThemeBasedFOI(foiaggrpop1);
        baseQuery = "select round(population/(3.14159*"+radius+"*"+radius+"))" +
                    "pop_normalized, population, location from(" +
                    "select round(sdo_sam.aggregates_for_geometry(" +
                    "'DEMOGRAPHICS_MEX','geometry','sum','col2', " +
                    "sdo_geom.sdo_buffer(a.geometria,"+radius+",0.5, " +
                    "'arc_tolerance=0.01 unit=km'))) " +
                    "population,a.geometria location " +
                    "from retailers a where (cadena = 'WALMART') " +
                    "and sdo_filter(geometria,sdo_geometry(2003,8307,null," +
                    "sdo_ELEM_INFO_ARRAY(1,1003,3),sdo_ORDINATE_ARRAY(? ,?,? ,?)," +
                    "'querytype=window')='TRUE')";
Population Calculation

Javascript

var theme = '<themes><theme name="JDBC_THEME">
  <jdbc_query asis="true" spatial_column="location">
    jdbc_srid="8307" render_style="'+bufferStyle+'" datasource="odf_gbl">' + baseQuery + '  </jdbc_query></theme></themes>';

foiaggrpop1 = new MVThemeBasedFOI('foiaggrpop1',theme);
foiaggrpop1.setBringToTopOnMouseOver(true);
mapview.addThemeBasedFOI(foiaggrpop1);
pop1IsVisible = true;
mdpopjdbctheme.setVisible(true);
document.getElementById('showHidePop1').value='Hide Pop.';
}
else
{
  pop1IsVisible = false;
document.getElementById('showHidePop1').value='Show Pop.';
foiaggrpop1.setVisible(false);
if (!pop2IsVisible)
  mdpopjdbctheme.setVisible(false);
}
}
Sales Mix
Sales Mix

- Style definition for thematic display
Sales Mix

- Shown as dynamic feature of interest (FOI) layers
  - Uses pre-created style definitions

**HTML**

```html
Sales Mix <input type="button" id="showHideSalesMix" value="Show" onclick="toggleSalesMix()">
```

**Javascript for legend**

```javascript
mdproduct = new MVMapDecoration("<img src="http://localhost:8888/mapviewer/omserver?ds=ODF_GBL&sty=V.Z_RETAIL_MIX&w=115&h=170">", 0.01, 0.07, 0, 0);
mdproduct.setOffset(0, 50)
mdproduct.setVisible(false);
mapview.addMapDecoration(mdproduct)
```
Sales Mix

Javascript

function toggleSalesMix()
{
  if(!salesMixIsVisible)
  {
    bufferStyle = "ODF_GBL:ALL_INVISIBLE";
    mapview.removeThemeBasedFOI(foisalesmix);
    baseQuery = "select CANASTA_BASICA_PCT, TECNOLOGIA_PCT, "+
      "ROPA_PCT, FARMACIA_PCT, "+
      "OTROS_PCT,Cadena,geometria location,ID "+
      "from retailers "+
      "where sdo_filter(geometria,sdo_geometry(2003,8307,null,"+
      "sdo_elem_info_array(1,1003,3), "+
      "sdo_ordinate_array(?,?,?,?)),'querytype=window')='TRUE' "+
      " and cadena='WALMART'";
}
Sales Mix

Javascript

```javascript
var theme = '<themes><theme name="JDBC_THEME" >
  '<jdbc_query asis="true" spatial_column="location"
  jdbc_srid="8307" key_column="ID" 
  render_style="'+bufferStyle+'" datasource="odf_gbl">'

  '+baseQuery+

  '</jdbc_query>'+

  '</themes>' ;
foisalesmix = new MVThemeBasedFOI('foisalesmix',theme);
foisalesmix.setBringToTopOnMouseOver(true);
mapview.addThemeBasedFOI(foisalesmix);
salesMixIsVisible = true;
mdproduct.setVisible(true);
document.getElementById('showHideSalesMix').value='Hide';
```
Sales Mix

Javascript

```javascript
else {
    salesMixVisible = false;
    document.getElementById('showHideSalesMix').value='Show';
    foisalesmix.setVisible(false);
    mdproduct.setVisible(false);
}
```
### Quarterly Sales

**Location Analytics with NAVTEQ**

<table>
<thead>
<tr>
<th>Name</th>
<th>VTAS_Q1</th>
<th>VTAS_Q2</th>
<th>VTAS_Q3</th>
</tr>
</thead>
<tbody>
<tr>
<td>WALMART</td>
<td>101.34</td>
<td>120.56</td>
<td>154.3</td>
</tr>
</tbody>
</table>

![Map Image](attachment:image.png)
Quarterly Sales

- Style definition for thematic display
Quarterly Sales

- Style definition for thematic display
Quarterly Sales

- Theme definition assigns style to content (assign table/column)
Quarterly Sales

- Theme definition assigns style to content (columns to bar chart)
Quarterly Sales

- Theme definition assigns style to content (clickable content)
Quarterly Sales

- Shown as dynamic feature of interest (FOI) layers
  - Uses pre-created style and theme definitions

**HTML**

```html
Quarterly <input type="button" id="showHideQuarterly" value="Show" onclick="toggleQuarterly()">
```

**Javascript for legend**

```javascript
mdquarterly=new MVMapDecoration("<img src="http://localhost:8888/mapviewer/omserver?
ds=ODF_GBL&sty=V.Z_MEX_QUARTERLY_INCOME&w=50&h=50">",0.01,0.03,0,0);

mapview.addMapDecoration(mdquarterly);
mdquarterly.setVisible(false);
mapview.addMapDecoration(mdquarterly)
```
Quarterly Sales

Javascript to add theme to map

define function quarterlySales

```javascript
let themebasedfoiQuarterly = new
MVThemeBasedFOI('themebasedfoiQuarterly',
'odf_gbl.Z_MEX_QUARTERLY_INCOME');
themebasedfoiQuarterly.setBringToTopOnMouseOver(true);
mapview.addThemeBasedFOI(themebasedfoiQuarterly);
themebasedfoiQuarterly.setVisible(false);
mapview.setThemeIndex(themebasedfoiQuarterly, 10);
```

end function quarterlySales
Quarterly Sales

Javascript

function toggleQuarterly()
{
    if(!quarterlyIsVisible)
    {
        quarterlyIsVisible = true;
        document.getElementById('showHideQuarterly').value='Hide';
        themeBasedFoiQuarterly.setVisible(true);
        mdQuarterly.setVisible(true);
    }
    else
    {
        quarterlyIsVisible = false;
        document.getElementById('showHideQuarterly').value='Show';
        themeBasedFoiQuarterly.setVisible(false);
        mdQuarterly.setVisible(false);
    }
}

Every Oracle database includes Oracle Locator

- Oracle Express and Oracle Standard, Oracle Standard One, and Oracle Enterprise Editions
- Oracle Locator includes:
  - Data type – `SDO_GEOMETRY`
  - Index type – Spatial Index
  - SQL query capabilities
- With Oracle Locator you can store and analyze information based on where things are
- Most information in a database has a location component
  - Customers, suppliers, assets, inventory, employees
Oracle Has a Location Data Type

SDO_GEOMETRY

- Oracle tables can easily incorporate a location column to hold geospatial information
- Simple to add, or can create table with the column

```
ALTER TABLE customers
ADD (location SDO_GEOMETRY);
```

- Adds a column called LOCATION to the CUSTOMERS table
  - Column of type SDO_GEOMETRY
  - Column naming follows normal Oracle column rules
What is **SDO_GEOMETRY**?

- An Oracle data type that defines the location on the Earth of a feature
  - It can define a *point* feature such as an address, an automobile location, a point of interest, a survey marker, etc.
What is SDO_GEOMETRY?

- An Oracle data type that defines the location on the Earth of a feature
  - It can define a *linestring* to represent a linear feature such as a road, a river, or a railway line
What is SDO_GEOMETRY?

- An Oracle data type that defines the location on the Earth of a feature
  - It can define a polygon to represent an area feature such as a country, state, park, lake, harbor, etc.
What is `SDO_GEOMETRY`?

For mapping purposes, the same feature may have different representations depending on the “zoom scale”

Above, San Diego International Airport is appropriately rendered as a point when zoomed out, but when we zoom in it is rendered as a polygon.
Querying by Location

- There are four commonly used location queries:
  - Relationship (*SDO_RELATE*)
  - Within Distance (*SDO_WITHIN_DISTANCE*)
  - Nearest Neighbor (*SDO_NN*)
  - Index only (*SDO_FILTER*)

- All four require a spatial index
- All four associate data with a feature or location of interest
Querying by Location

Relationship
- Who are my customers with income > $67,000?

Within distance
- Which hotels are within two miles of my proposed hotel location

Nearest neighbor
- What is the closest ATM to my current location

All location analysis queries use simple SQL which includes location as a search in the where clause.
Using Oracle for Location Analysis

- Most data includes a location component
- Information can be derived from the location component using location queries
- You can easily add location information into your existing tables
- You can index location data
- You can query by location
Oracle Spatial

- Priced option for Oracle Enterprise Edition
- Lots of features
  - Geocoding
  - Imagery storage and retrieval
  - Spatial mining
  - Linear referencing
  - Routing
  - Network analysis
  - Topology
  - Spatial analysis and aggregation
  - Web Services
  - ... and much more
Oracle Fusion Middleware MapViewer

- No cost feature of the Oracle Application Server (including WebLogic Server)
- Supports rendering of location data out of the database
- Easily publishes spatial data to the web
- Centralized management of symbology, annotation and map definition rules
  - Map authoring tool is Map Builder
- Provides an XML API, Java API, easy-to-use JSP interface and an OGC WMS interface
MapViewer Architecture

- **Client**
  - Browser/Apps
    - XML/HTTP, Java/JSP
    - OGC WMS APIs

- **Middle-tier**
  - Oracle Application Server
    - MapViewer
    - Optional

- **Database**
  - Oracle Spatial/Locator
    - Map Content (NAVTEQ)
    - Map metadata
  - Map Cache
    - Map Builder
MapViewer

Renders maps

- Visual representation of location data stored in the database
- Maps are composed of one or more themes
  - Each theme is location information from a table in the database with drawing instructions
  - The drawing instructions are styling information that include color, line thickness, markers, label text, font, and more
MapViewer Metadata

- **Styles**
  - Can control fill color, border color, line thickness, line style, text styles, and so on
  - Stored as XML in view `USER_SDO_STYLES`

- **Themes**
  - A geometry column associated with a style
  - A text label column can also be associated with a theme. The label column has its own style (size, font, color, and other attributes)
  - Stored as XML in view `USER_SDO_THEMES`

- **Maps**
  - One or more themes
  - Stored as XML in view `USER_SDO_MAPS`
Oracle Map Builder

Oracle Map Builder is Oracle’s map authoring tool created to manage map metadata

- Creates and manages XML metadata associated with styles, themes, and maps (plus a lot of other things)
- Simple GUI

<?xml version="1.0" standalone="yes"?>
<svg width="1in" height="1in">
  <desc/>
  <g class="color" style="stroke:#403E3E;fill:#B0AEAE">
    <rect width="50" height="50"/>
  </g>
</svg>
Connect to a Database

After startup, you need to connect to a database to read existing map metadata, and/or to store new map metadata

- Click Connection, choose Load/Add/Remove
Connect to a Database

After startup, you need to connect to a database to read existing map metadata, and/or to store new map metadata

- Click Connection, choose Load/Add/Remove
- Click Add
Connect to a Database

After startup, you need to connect to a database to read existing map metadata, and/or to store new map metadata

- Click Connection, choose Load/Add/Remove
- Click Add
- Enter the connection information
Connect to a Database

After startup, you need to connect to a database to read existing map metadata, and/or to store new map metadata

- Click Connection, choose Load/Add/Remove
- Click Add
- Enter the connection information
- Click OK
Map Creation

To create a map using Oracle Map Builder:
- Create styles that will define how features will look on the map
Map Creation

To create a map using Oracle Map Builder:

- Create styles that will define how features will look on the map
- Create themes, which associate features with the style you have created
Map Creation

To create a map using Oracle Map Builder:

- Create styles that will define how features will look on the map
- Create themes, which associate features with the style you have created
- Create a map, which is a set of themes and instructions for when to turn themes on and off
Style Information

Areas
- A fill pattern for an area

Colors
- The fill color of an area; can also include stroke color and thickness (the line surrounding the area), and transparency

Lines
- The color, transparency, thickness, wing lines, center lines, fencing, and end caps associated with a line

Markers
- Icons usually representing point features or labels on a map

Texts
- The size, font, color, and halo of text

Advanced
- A way of assigning different styles based on column values
Creating a Style

To create a style, right click on the type of style you want to create, then click “Create … Style”
Creating a Style

Name the style

Choose a fill color

Fill -> Color (Swatches, RGB, HSB)

If you want a see-through color, decrease the opacity
Creating a Style

Add a stroke color and width if desired
  - Stroke is the boundary of an area feature
Click Preview
When complete, click Save
Themes

Assign style information to features for rendering location data

Color or fill patterns are assigned to area features

Line styles such as color, thickness, wing line, and more are assigned to linear features

Markers are assigned to point features, and can also be assigned for labeling purposes (such as highway markers)

Text styles are assigned for labeling purposes

Advanced styles are assigned when the feature drawing style is dependent on one or more values in the table

Location data in a table + rendering style = Theme
Creating a Theme (1)

To create a theme from scratch, right click the type of theme (Geometry), then click Create Geometry Theme
Name the theme, chose a table and an associated geometry column
Creating a Theme (2)

Choose a color style from the colors style picker (area).

If you assign an Advanced style, in the lower box choose the column whose value defines the style.
Creating a Theme (3)

Check the Label Style check box if you want to label this theme, then choose the text style

Click Attributes to choose the column with the label text
Creating a Theme (4)

Most often there will be no Query Condition
In this case, we choose fresh water features (lakes, rivers, canals), but not bays
Theme Preview

When we are done creating a theme we can preview it.
Maps

A map is a collection on one or more themes

When associating a theme with a map, Oracle Map Builder gives the map author a lot of flexibility beyond what has already been discussed. The author can:

- Define the order in which themes render
- Define the map scale at which themes are turned on and off
- Define the map scale at which labels for a theme are turned on and off
- Allow MapViewer to simplify data when rendering
- Force labels to always appear
- Eliminate duplicate labels
- Stop insignificant features from being fetched from the database
- …and much more
Creating a Map

To create a map, right click Base Map in the Navigation Panel on the left, and choose Create Base Map.

Next, name the map.

Then add one or more themes to the map.
Adding Themes to a Map

Click on the Editor tab, then click the plus sign to add a new row. Next, click on a theme to add.
Adding Themes to a Map

Click on the theme you just added, then click the pencil (to edit)

In the scale section of the properties tab, change scale mode from `MAPVIEWER_NATIVE` to `RATIO`

The scale is the denominator in map scale, i.e. 1/200000 is 200000

Minimum scale is the scale at which the theme turns on when zooming in

Maximum scale is the scale at which the theme turns off when further zooming in

Minimum scale will always be greater than maximum scale

Empty scales mean infinity (minimum scale) and 0 (maximum scale)
Adding Themes to a Map

In the general properties tab:

- **Allow naked points** – Points can be rendered even if the label or marker they are associated with cannot be rendered (due to label or marker conflict)
- **Fast Geometry Unpickling** – Uses special MapViewer code to unpickle data. Incurs very slight loss of precision, seldom a problem for read applications
- **Force labeling** – MapViewer will always label the theme when map rendering
- **No repetitive labels** – Ensures label will only be rendered once in a theme

- **Simplify Shapes** – MapViewer will simplify data to improve rendering speed
- **Minimum pixels** – MapViewer asks the database not to return data that will not render over at least the given number of display pixels
- **Timeout** – Used for WMS/WFS
Add a Map to a Web App
Add a Map to a Web Application

```html
<html> <head>
<META http-equiv="Content-Type" content="text/html" charset=UTF-8">
<TITLE>NAVTEQ BI Tutorial</TITLE>
<script language=javascript src="jslib/oraclemaps.js"></script>
<script language=javascript>

var baseURL  = "http://"+document.location.host+"/mapviewer";
var mapUser = "ODF_GBL";
var vectorMapCache = "WORLD_MAP";
var mapview;
var oraBasemap;
var copyRight = null ;

function showMap()
{
    mapview = new MVMapView(document.getElementById("map"), baseURL);
    oraBasemap = new MVMapTileLayer(mapUser+"."+vectorMapCache);
    mapview.addMapTileLayer(oraBasemap);
    mapview.enableMapWrapAround(true);
    oraBasemap.setVisible(true);
    callBack();
}
```
Add a Map to a Web Application

```javascript
function callBack()
{
    var mapCenterLon = -99.134;
    var mapCenterLat =  19.432;
    var mapZoom      =  10;
    var mpoint = MVSdoGeometry.createPoint(mapCenterLon,mapCenterLat,8307);
    mapview.setCenter(mpoint);
    mapview.setZoomLevel(mapZoom);

    mapview.addNavigationPanel("EAST");

    copyRight=new MVMMapDecoration( "&©2009 NAVTEQ©8482; and &©2009 Oracle Corp",1, 1,400,20);
    copyRight.setOffset(-600, -20);
    mapview.addMapDecoration(copyRight);

    mapview.addScaleBar();
}
```
Add a Map to a Web Application

```javascript
ovcontainer = new MVMapDecoration(null, null, null, 200, 150);
ovcontainer.setCollapsible(true, true);
ovcontainer.setTitleBar("Overview Map", "/mapviewer/fsmc/images/overviewicon.png", "Overview map");
mapview.addMapDecoration(ovcontainer);
var over = new MVOverviewMap(ovcontainer.getContainerDiv(), 3);
mapview.addOverviewMap(over)

mapview.display();
}

</script>
</head>

<body onload="javascript:showMap();">
...

  <div id="map" style="width:100%;height:90%"></div>
...

</body>
</html>
```
Click to continue