Oracle Maps Installation and Integration in APEX

An Oracle White Paper

January, 2007

Table of Contents:

1. Purpose of this document

Have a „Quick Installation Guide“ to integrate Oracle Maps and APEX.

2. Prerequisites

1. Oracle Database Server 10g(10.1.3) or 9i (9.2.0.3)
2. Oracle Spatial functionality installed
3. APEX 2.2
4. Oracle Application Server 10.1.3 Java Edition (in Quickstart Kit)
5. Oracle Application Server Mapviewer 10.1.3 (in Quickstart Kit)
6. MVDEMO Demo Dataset for Mapviewer
7. Java JDK 1.5

The esteemed result will look as follows:

3. Installation of Oracle Application Server Mapviewer Quickstart Kit

1. Please go to the Oracle AS Mapviewer/Software Download page (http://www.oracle.com/technology/software/products/mapviewer/index.html)
2. Download following sources for the latest version 10.1.3.1:
3. MapViewer Quickstart kit—The MapViewer Quickstart kit includes a ready-to-run standalone OC4J with MapViewer 10.1.3.1 pre-deployed and configured (Zip - 89.6MB)
4. Demo Data Set—A MapViewer demo dataset (MVDEMO) specifically prepared for this release (Zip - 4.3MB)
5 View/Download the Users Guide—An updated PDF version of the MapViewer User's Guide (4.9MB)
6. Readme (Text)
Optional: Map Builder kit—The new Map Builder desktop application (a tool to edit Themes/Layers and spatial attributes)

Step 1:
Prerequisite: Oracle Database Server 10g or 9i with the Spatial Functionailty.
You can check, if it is already installed as follows:
Try to login as user MDSYS/MDSYS or install as follows:
SQL> connect SYS as SYSDBA
SQL> @?/md/admin/mdinst.sql

Step 2:
Unzip the "MVDEMO" Demo Dataset (as per 4.) and install it as per the Readme.txt using SQL-Plus into the database. A set of Demodata like tables, maps and layers, themes, useful spatial attributes etc.will be installed and indizes initialized; i.e. the spatial functions of the spatial schema MDSYS will e made available for the Demo Dataset. (note: the MD of MDSYS stands for "Media", as Spatial is a part of the Media (-Text) functionality of the database.

Step 3:
Unzip the Mapviewer Quickstart Kit and make shure you have Java SDK 1.5 installed/available on that machine; then you need to adjust the path of the batchfile start.bat or start.sh to start the OC4J and Mapviewer;
an example batchfile looks as follows:

```
C:
cd C:\mv10131_qs\oc4j\j2ee\home
java -server -Xmx384M -jar
oc4j.jar
```

(start.bat oder start.sh – Startbatchfile for the Oracle Application Server)

Step4:
Start the OC4J (Oracle Containers for Java, the J2EE Version of the Oracle Application Servers) - Doubleclick on "start.bat"

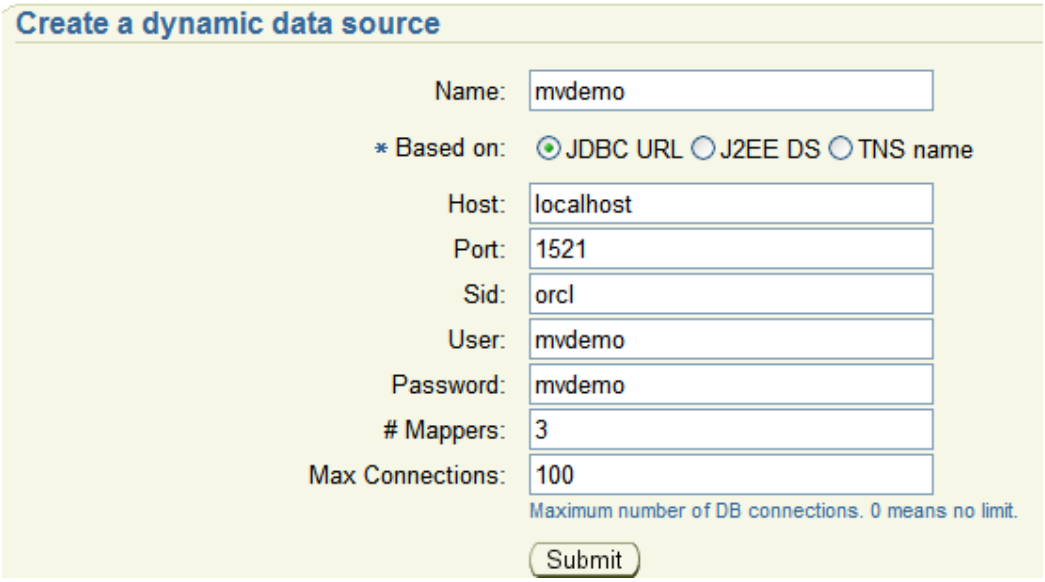
Step5:
Start the Mapviewer in the browser:
http://localhost:8888/mapviewer or
http://localhost:8888/mapviewer/faces/home.jspx

Now the Admin API for the OC4J/Mapviewer is shown:

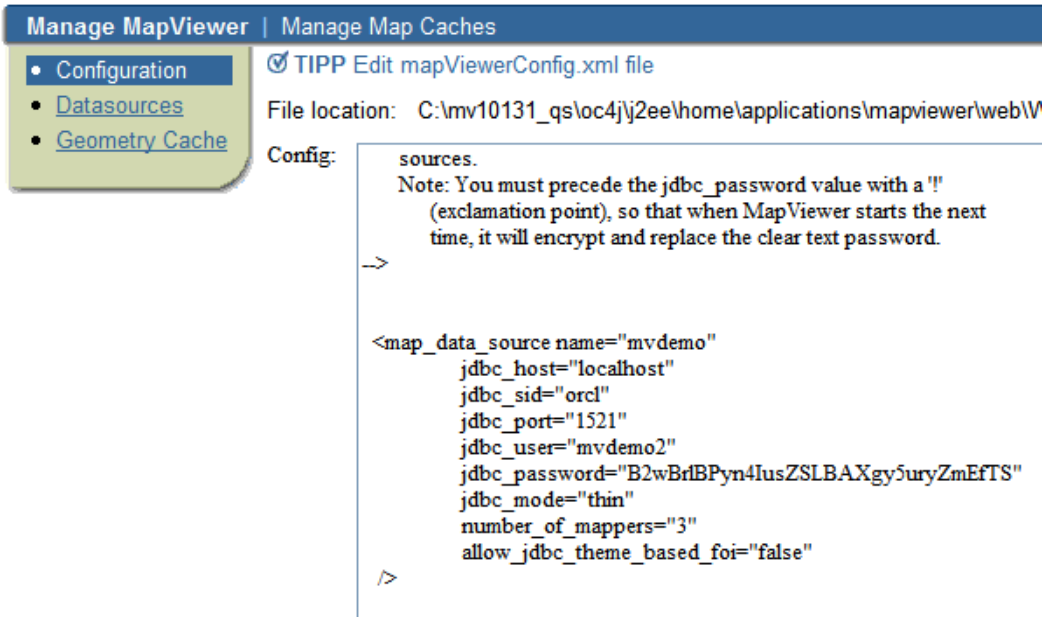
At the bottom of the page you will see: ***Oracle Maps Tutorial***.

Step6:
We enter the Oracle Maps Tutorial and start reviewing Chapter 3 Demo Setup:
Tutorial Demo Setup Step 1 - Import of the Demo Datasets MVDEMO (you will have done already). So you will only need to drop the user MVDEMO, if it existed from previous installations.
Please follow with step 2 accessing the ADMIN area (Button top right in the header) and define a Datasource as per the instructions (connection between the OC4J and the database).
You may login in as „oc4jadmin/oc4jadmin“ (if you have not changed it).

Choose „Datasources“ and enter details (this as an example only) as follows:



Note:
This dynamic datasource is only valid as long as the OC4J is running;
If you like to define a *permanent datasource*, go to „Configuration“ and edit the XML file by means of this API:
Tipp: when entering the jdbc_password you have to put an „!“ in front of the password (like !mvdemo) enabling the OC4J to encrypt it when restarting.
Press „Save & Restart“ at the bottom of the screen.

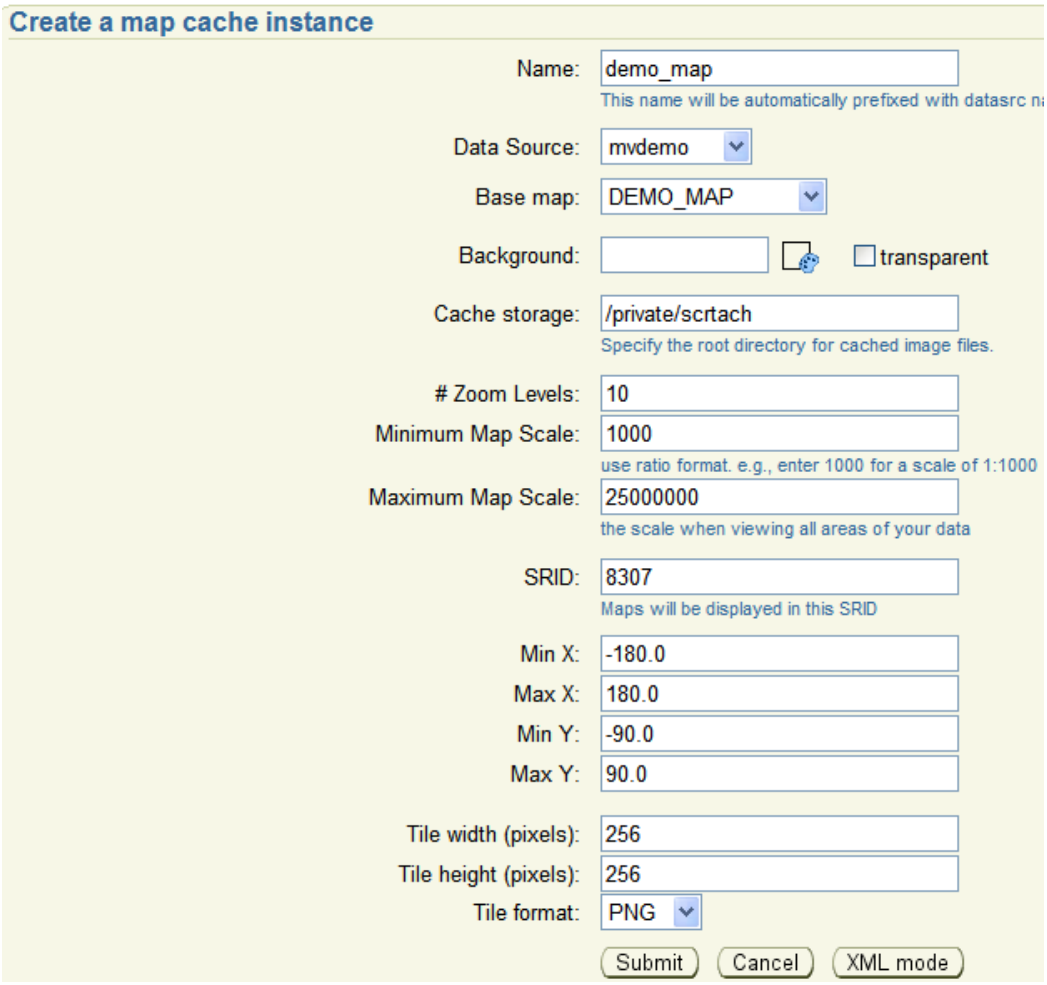


Re Step 3) of the tutorial
Ofcourse you will need to run the sript "mcsdefinition.sql" as explained.

Re Step 4) of the tutorial - Mapcache
In order to gain performance especially for Oracle Maps a separate cache is used.

Please procede as per the tutorial: (Text from the Tutorial for easier reference)

Create a map cache instance. Go to the 'Management' tab, then choose "Manage Map Caches". Now click on "Create", select 'Internal' then submit. In the ensuing page, fill in the form for creating a new map cache instance. You should use "demo_map" as the name of the new cache, and select "mvdemo" as the data source, then choose "demo_map" from the base maps drop down list. You can keep all the other fields as is. Optionally, enter a proper value for cache storage, so that it points to a desired directory on the same host OC4J/MapViewer is running. It does not have to be under the MapViewer directory tree. Now click the Submit button. This will tell MapViewer to create a new Map Cache instance for the basemap "demo_map" in the datasource "mvdemo". The newly created map cache is automatically renamed to 'mvdemo.demo_map'.



Note: If the Cache storage root directory is incorrect, an error (warning) is raised when starting the OC4J, which you can disregard. The OC4J will choose a default location for these cache data in the OC4J directory.

Then you can switch to "Running the Demos" and have a look at the first Oracle Maps.

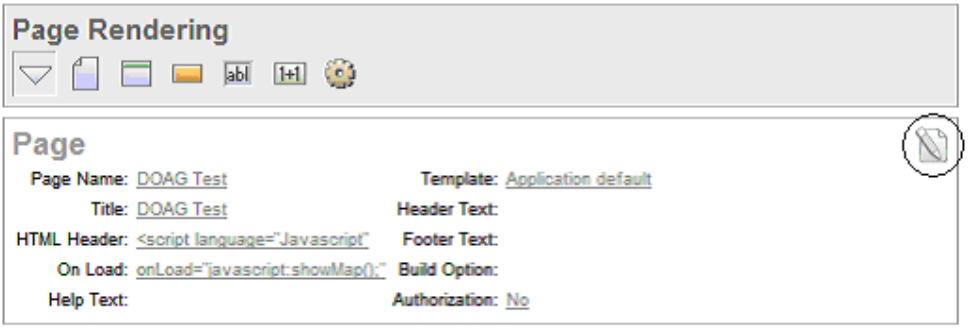
Step 7:
In the Oracle Maps Tutorial you will find a TOC top left with: ***Getting Started***.
3 steps to build an Oracle Maps API are described:
The essential steps to building any Oracle Maps application or mashup using the JavaScript API are:
(1) Loading the Oracle Maps JavaScript library in the beginning of a web page
(2) Placing an HTML DIV component that serves as the master map container on the page
(3) Writing custom JavaScript code to set up initial map contents (base-map, FOI layers, etc) and implement application specific logic.

4. Installation of Oracle Maps in APEX

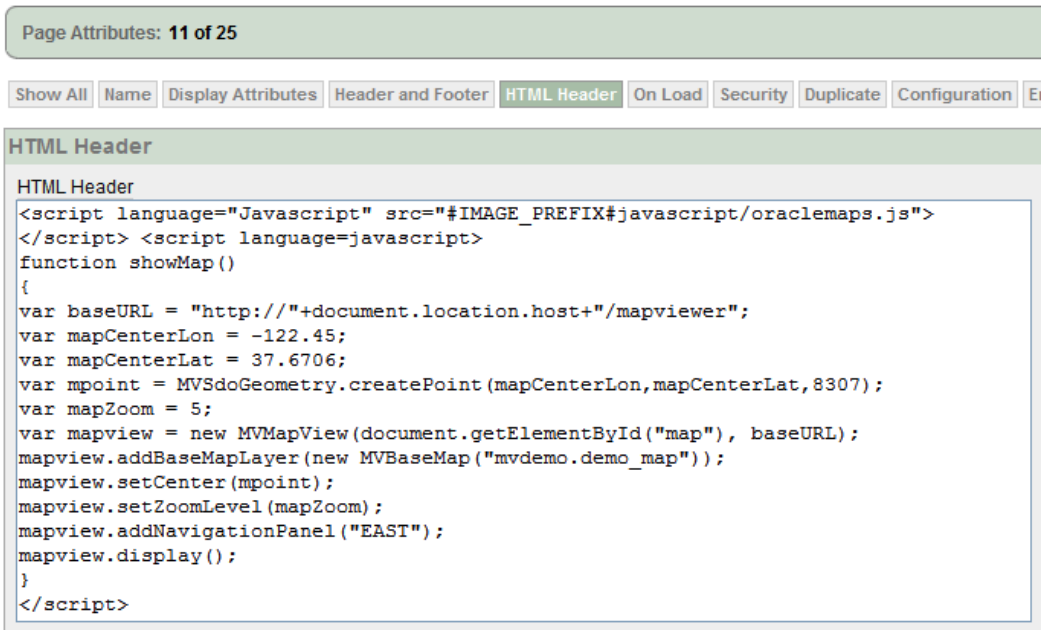
So where do we place the maps in APEX?

Step 1 – loading of the Oracle Maps Javascript library
We define a new APEX application with a blank page.

Re (1) – We go to the blank page and look at the page attributes. In the section „page rendering“ we edit the page attributes by clicking at the top right icon.

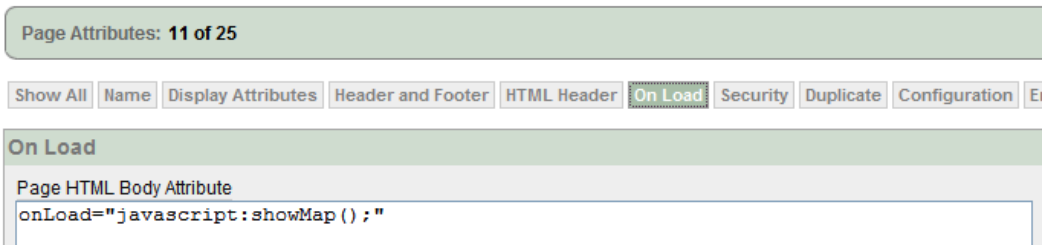


On the page attribute page we chose the HTML Header and copy the sourcecode as per Step 1 and Step 3 (of the tutorial) and paste it into the „HTML Header“ field.:
(ex Step 1)
<script language="Javascript" src="/mapviewer/fsmc/jslib/oraclemaps.js"></script>



and copy/paste the the large block as per Step 3 (only the large block) as well.
Please make sure the "Cursor Focus" amongst the Display Attributes has to be set to: "Do not focus Cursor"
Next go to the page attribute section "On Load" (below the "HTML Header") and copy/paste the code:

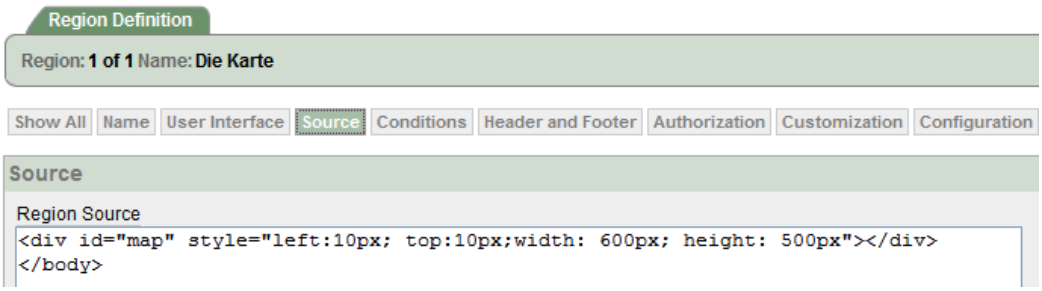
```
<body onLoad="javascript:showMap();">
```



Now the javascript of Oracle Maps will be loaded at the beginning of this page.

Step 2 – Presenting the map in a page region:
So where will we position the map?
For presentation purpose we will define a region of type HTML and copy/paste the following source code as per Step 2 into „Sources":

```
<div id="map" style="left:1%; top:1%; width:99%; height:99%"></div>
```



Explanation: A so called DIV Container named "map" will receive all data and details the Oracle Maps javascript and the Mapviewer is generating to present the map on the page.
As the copied source as ment for a pure HTML page for APEX we have to adjust the position of the map/container to read like:

```
<div id="map" style="left:10px;top:10px;width:600px; height:500px"></div>
```

We still have to make following small adjustment in the HTML Header of the page attributes; you will recall that the original sources read as follows:

src="/mapviewer/fsmc/jslib/oraclemaps.js

and it has to read:

"src="#IMAGE_PREFIX#javascript/oraclemaps.js"

As #IMAGE_PREFIX# maps to the apex\images directory of the Apache/http server; that inturn only makes sense, if we make sure, that the "*oraclemaps.js*" is pasted into the APEX „images/javascript“ directory

...??\?apex?\images\javascript

Therefore please check for the "*oraclemaps.js*" file in the directory where the OC4J has been unzipped/installed:
You will find the „oraclemaps.js“file here:

..??\oc4j\j2ee\home\applications\mapviewer\web\fsmc\jslib

and you copy it to here:

..??\?apex?\images\javascript

Note:
You can also store the „oraclmaps.js“ into the APEX/shared component/static files facility and use: "src="#WORSPACE_IMAGES#oraclemaps.js" in the HTML Header.

So now we have the last step for you – then we are set:

As APEX and Oracle Maps both make use of javascript and "Cross-Scripting" of

APEX and OC4J is not allow we have to tell the Apache/http server and the proxy settings how to overcome that problem: We go to the "*httpd.conf*" in:

...??\htmldb\Apache\Apache\conf

and revise following section to read as follows:

```
#
# Proxy Server directives. Uncomment the following lines to
# enable the proxy server:
#
ProxyRequests On
ProxyPass /mapviewer/ http://localhost:8888/mapviewer/
ProxyPassReverse /mapviewer/ http://localhost:8888/mapviewer/
NoCache *
#
```

Now you should be able to see your first map based on Oracle Maps in APEX.
If you review the Oracle Maps tutorial in more detail you will find about 30 to 40 features which can easily be added to this simple map.

5. Our map in APEX looks as follows

6. Web resources

1. Oracle Spatial : http://www.oracle.com/technology/products/mapviewer/index.html
2. MapViewer: http://www.oracle.com/technology/products/mapviewer/index.html
3. Mapviewer Quickstart Kit Download:
http://www.oracle.com/technology/software/products/mapviewer/index.html
3. APEX:
http://www.oracle.com/technology/products/database/application_express/index.html

Compiled by

BERNHARD FISCHER-WASELS
Principal Salesconsultant
EMEA IPC leader Ports & Maritime
STCC Server Technologies Competence Center Nord
ORACLE Deutschland GmbH
Kühnehöfe 5
22761 Hamburg
Germany
Phone: +49 40 89091 128
Fax: +49 40 89091 250
Mobile: +49 177 5949 128
e-Mail: bernhard.fischer-wasels@oracle.com
APEX (HTMLDB) Blog (in German) http://htmldb-de.blogspot.com

Oracle Maps Installation and Integration in APEX
January 2007
Author: Bernhard Fischer-Wasels
Oracle Deutschland GmbH
Niederlassung Hamburg
Kühnehöfe 5
D-21692 Hamburg Germany
Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com
Oracle Corporation provides the software
that powers the internet.