

Feature

Developing raster management with Oracle 9i and MicroStation V8

by Stanislav Sumbera and Petr Vinduska

MicroStation plays important role in GIS, especially in Automated Mapping/Facility Management (AM/FM). In these areas, raster data such as satellite images, air photos, scanned topographical maps or other types of imagery are becoming increasingly popular. It has been estimated that these data may represents more than 90% of the average GIS data holding by volume.

MicroStation V8 comes with Raster Manager as a solution for united raster management of different raster formats. Oracle 9i brings its GIS technologies like Spatial option, *interMedia* or Database Workspace Manager for effective database storing of geographical-related data. Oracle database is used in number of organizations to manage all kinds of data across the enterprise.

In this article, we will join Oracle GIS technologies with MicroStation v8 to build up a prototype called "Image Storage" for storing rasters in Oracle9i database using Oracle *interMedia*, Oracle Spatial, Oracle Workspace Manager and finally displaying query results with help of Raster Manager and it's clipping capabilities. We are going to point out some interesting concepts and techniques in Image Storage.

Nowadays there are only a few solutions for storing rasters in databases, actually most of the interest in raster management lies in using compressed proprietary formats like Mr.SID or ECW. Note ECW is supported in v8.

Architecture

Figure 1 represents the architecture of the application prototype. Data tier utilizes Oracle 9i technologies for image store, spatial location and long transaction. Logical tier of Image Storage benefits from Java 2 environment available in MicroStation v8 and additional java classes provided by Oracle 9i for object manipulation.

Java class *ImageServer.class* provides all operations with Oracle 9i using Oracle JDBC and Oracle *interMedia* additional classes.

MicroStation v8 supports native code with full access to MicroStation Development Library, hence the presentation tier was designed in Visual C++ using MFC.

Communication between native and Java code is provided by Java Native Interface (JNI).

User interface is implemented in *ImageStorage.dll* native code. Displaying and clipping of rasters is realized by Raster Manager in MicroStation v8.

The architecture was designed to handle different tasks as simple as possible.

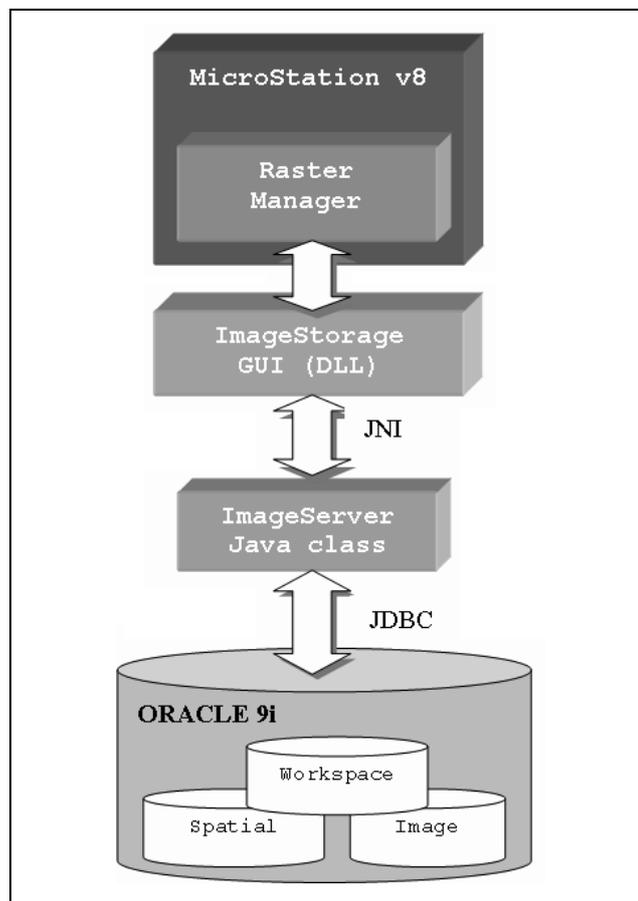


Figure 1 The Architecture of Image Storage

Using Oracle 9i technoloGIS

Conventional database management systems are designed to handle relation model and short-transactions. These technologies are insufficient for GIS applications where we work with spatial objects, images and transactions can take days to complete.

These types of problems require different technology. RDBMS needs an extension for long-transactions, the storage and management of image data together with other information.

The environment should facilitate manipulation and query with spatial features. This problem was recognized before couple of years and software companies started an extension their products by new features.

The examples of the extensions are workspace manager, spatial and interMedia options in Oracle 9i. This object-relational database management system and embedded functions can sharply help you to design GIS applications.

For our prototype, we will create a simple table with image and spatial objects. Each image storage table has suffix **_storage**. The table is created as follows:

```
create table img_storage
(ID number PRIMARY KEY not null,
NAME varchar2 (256),
MBR MDSYS.SDO_GEOMETRY,
-- Object defined in Spatial
IMAGE ORDSYS.ORDIMAGE);
-- Object defined in interMedia
```

Raster locating using Oracle Spatial

Oracle Spatial is a set of functions and definitions for a manipulation and maintenance of spatial objects. Current version supports two models, the relational and the object-relational. Both correspond to alternatives described in the OpenGIS specifications. Object-relational model is strongly recommended for all applications except replications and distributed database.

There are four part of spatial:

- **Scheme with information about storage, syntax and semantic of supported data geometry types.**
- **Spatial index mechanism.** It is a key feature, the choice of type of spatial index has an influence on a performance and every developer would recognize those mechanisms. Today, you can use Oracle spatial advisor to reduce an expense and time to solve this problem.
- **A set of operations and functions for performing area-of-interest and spatial join queries.** Spatial uses two-tier mode for processing of queries. First tier only reduce set of data (approximation comparison) second tier (filter) exactly compute objects that result from primary processing
- **Administration functions**

The column MBR in table `img_storage` represents minimal boundary rectangle of the raster stored in spatial object and will serve for locating purpose.

Suppose we have populated data into database. Example of spatial query used for locating rasters in download follows. Note the coordinates are taken from fence placed in MicroStation and passed into Java code for processing.

```
select id from img_storage
where SDO_FILTER (img_storage.mbr,
MDSYS.SDO_GEOMETRY (2003,NULL,NULL,
MDSYS.SDO_ELEM_INFO_ARRAY (1,1003,1),
MDSYS.SDO_ORDINATE_ARRAY
(-794227608,-962012845,-763500066,-962012845
,-763500066,-996137402,-794227608,-996137402
,-794227608,-962012845)),
'querytype=window')= 'TRUE'
```

Raster storage using Oracle *interMedia* image object

interMedia enables Oracle 9i to manage and retrieve images and others heterogeneous data. The technology is based on object relational database system, which provides support for definition of object types and methods that work on them.

The objects of Oracle *interMedia* are stored in binary large objects (BLOB) in Oracle9i or in file-based large objects (BFILE) in file system of operation system. Applications can interact with database objects through modern Java library. or traditional PL/SQL and Oracle call interface. Images are stored in **ORDImage** object that supports two-dimensional, static, digitalized raster images in binary representation.

The object can store and retrieve image data in any format and supports among others methods for inserting a row using BLOB images, populating a row using BLOB images, copying an image, converting an image format, extending *interMedia* with new image types and many others.

An example of using *interMedia* image object will be an upload of image into Oracle and download image from database to file system. An upload consists of select query on row where the image is going to be uploaded via `OrdImage` object.

Know Your Freebies

Fromto.ma

Aside from cell, it also allows you to re-map design file elements from one Level, CO, WT, LS, and/or Cellname to another, based on an Attribute Configuration File.

The user can Add copies and delete elements from the design file. And most important of all it can be used in batch file. You can replace cells A1 to Cell A2 in all drawings without opening one by one. Check out:

Systems hk

<http://home.attbi.com/~hksys>

CAD

Upload of image

```
// Java code fragment of uploading an image into database
String sql = "select IMAGE from IMG_STORAGE where NAME = '"+name+"' for update";
Statement s1 = con.createStatement();
OracleResultSet rs1 = (OracleResultSet) s1.executeQuery(sql);
rs1.next();
OrdImage imgObj = (OrdImage)
rs1.getCustomDatum(1, OrdImage.getFactory());
imgObj.loadDataFromFile(FileName);
sql = "update img_storage set image = ? where NAME = '"+ name+"'";
stmt = (OraclePreparedStatement)con.prepareStatement(sql);
stmt.setCustomDatum(1, imgObj);
stmt.execute();stmt.close();
```

Download of image

```
// Java code fragment of downloading an image from database
String select = "select IMAGE,NAME from img_storage";
Statement s1 = con.createStatement();
OracleResultSet rs1 =(OracleResultSet) s1.executeQuery(select);
while(rs1.next()) {
OrdImage imgObj= (OrdImage)
rs1.getCustomDatum(1, OrdImage.getFactory());
String fileName = rs1.getString(2);
imgObj.getDataInFile(fileName);
}
rs1.close();s1.close();
```

Database Workspace Manager

Workspace Manager can hold different versions of the same record in one or more workspaces. This feature is called versioning. There are two fundamental advantages of this system:

- Concurrent access to the database. In the database without versioning, users can change same record only when other user has not locked it yet. The versioning ability relaxes this rule and improves concurrent access to the database.
- What-if analyses can be run simultaneously. Each analysis works on a separate version of the data. After the completing of analyses, the results can be stored in the database for lookup.

A database table is base element for versioning, which means that all rows in the version-enabled table can now support multiple versions of data. The infrastructure is not visible to the users of the database. The system offers set of functions for manipulating with results.

Since workspace manager creates tree of spaces, all functions are fitted for it. For example the MERGE propagates all changes from child workspace to the parent workspace. The REFRESH has opposite effect.

Figure 2 (on the next page) shows the various image storage GUI.

Final words

This example presented how to store in Oracle database also rasters with usage of interMedia, Workspace Manager and Spatial option. Native code was compiled in Microsoft Visual C++ .NET v.7.0. The code was tested on MicroStation v 8.0 running on Windows XP. Oracle database was 9.0.

About The Author

Stanislav Sumbera, Ph.D. is a LIDS and MicroStation software developer for the Berit group (www.berit.com). You may contact Stanislav Sumbera at ssumbera@berit.cz. The code for this article may be downloaded form www.sumbera.com.

Petr Vinduska is Oracle database developer for APP Czech s.r.o specialized in GIS and data warehouse solutions. You may contact Petr Vinduska at pvindusk@appg.com.

CAD

If you have MicroStation tips,
please email details to

penbrush@ozemail.com.au

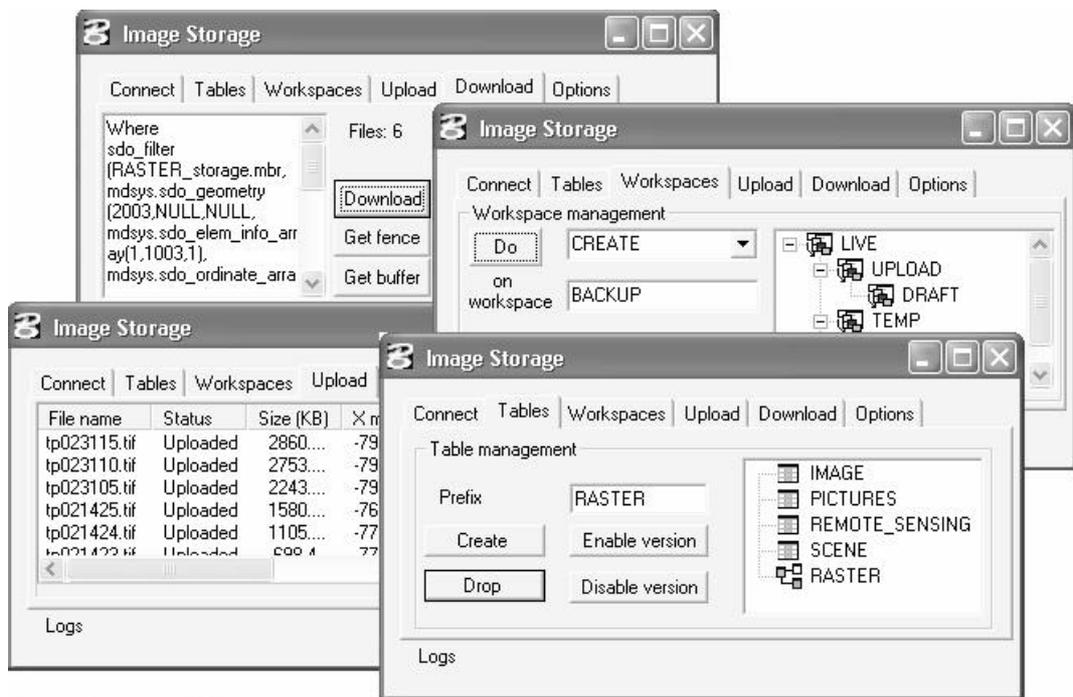


Figure 2 Image Storage GUI

Tips

Changing Readout Format

How do I change from DD.MM.SS to DD.MM on text that has already been placed throught out the design plane?

Use the FIND/REPLACE tool.

- In the Find field, put ". ." (thats 2 full stops and the seconds symbol).
- Leave the Change Replace field blank.
- Check the Regular Expression checkbox.
- Press the Change All button the seconds should all disappear.

Read the FIND/REPLACE help file for other useful tricks.

CAD

Tips

Scaling Custom Linestyles

Is there a way to change custom linetypes that were drawn at a scale in one file and needs to be referenced to another file that needs to be shown at another scale and the linetypes need to be adjusted for the different scale. The reference scale cannot be changed.

Custom Linestyles are usually scaled at the time of placement. there is a tool called "modify linestyle attributes" this will allow you to modify the scale of the linestyle after it has been already placed.

CAD

Tips

Save The Pre-processing

When rendering using Ray Tracing, is there a way to save the pre-processing to use when I come back to rendering a file at a later date?

-
- In V8, go to *Settings>Rendering>Ray Trace* and the Render dialog box will open.
 - Click the Settings button to open the Ray Tracing dialog box.
 - Under the File pulldown menu, select *Save> Solution*.

Save the pre-processing as a *.rad file which can be loaded later or during an *Animation>Fly-through*. Remember that if any material or lighting setting changes, then the pre-processing has to be recalculated.

CAD

Tips

Activate Key-in Browser

How do I activate the Key-In Browser with some shortcuts?

If you open the default menu with *Workspace>Customize* (and go to the Menu Bar tab), *Utilities>Key-in* invokes the command dialog *cmdbrowse*.

CAD