

More Options For Loading and Exporting SDO_GEORASTER

- Oracle Spatial's minimum support for loaders and exporters include:
 - TIFF / GeoTIFF in Oracle 11g and newer
 - JPEG
 - JPEG 2000 in Oracle 11g and newer
 - GIF
 - BMP
 - PNG
- Oracle Spatial relies on partners to import or export data in many data formats to or from the SDO_GEORASTER data type.
- GDAL, an open source tool for importing/exporting many raster formats, now includes GeoRaster support.

GDAL Open Source Raster Loader/Exporter Includes SDO_GEORASTER Support

- GDAL's natively supports importing and exporting many formats, to/from SDO_GEORASTER, including JPEG, TIF, GeoTiff and more.
- Plugins are available to extend the formats natively supported by GDAL.
 - For example, ERDAS offers an ECW JPEG 2000 Codec SDK that can plug into GDAL for reading and writing ECW and JPEG 2000 formats.
 - No cost for reading ECW or JPEG 2000, cost for writing.
 - URL to licensing details of ERDAS JDK in slide notes.
- GDAL is written in C++, and runs much faster than Oracle's JAVA GeoRaster loader/exporter. Oracle's loader/exporter is based on SUN's JAI libraries.

How to compile GDAL

- Included are abridged instructions on how to compile GDAL on Linux, UNIX and Windows.
- A Windows precompiled executable for GDAL is also available through the GDAL OSGeo4w tool.
- Instructions on how to install OSGeo4w are included in the file that describes compiling GDAL for Windows.

Using GDAL utilities

- Two key executables are:
 - `gdalinfo` – utility to view information about a raster. For example, image size, coordinate system, georeference and color map information.
 - `gdal_translate` – utility to translate a raster from one format to another, for example:
 - GeoTiff to SDO_GEORASTER
 - SDO_GEORASTER to GeoTiff
 - To/from formats supported by GDAL

Calling gdalinfo On A GeoTiff File

```
CMD> gdalinfo sf1.gtiff
```

```
Size is 4299, 4299  
Coordinate System is:
```

Displays image size and coordinate system (including SRID)

```
PROJCS["NAD83 / California zone 3",  
  GEOGCS["NAD83",  
    DATUM["North_American_Datum_1983",  
      SPHEROID["GRS_1980",6378137,298.2572221010042,  
        AUTHORITY["EPSG","7019"]],  
      AUTHORITY["EPSG","6269"]],  
    PRIMEM["Greenwich",0],  
    UNIT["degree",0.0174532925199433],  
    AUTHORITY["EPSG","4269"]],  
  PROJECTION["Lambert_Conformal_Conic_2SP"],  
  PARAMETER["standard_parallel_1",38.43333333333333],  
  PARAMETER["standard_parallel_2",37.06666666666667],  
  PARAMETER["latitude_of_origin",36.5],  
  PARAMETER["central_meridian",-120.5],  
  PARAMETER["false_easting",2000000],  
  PARAMETER["false_northing",500000],  
  UNIT["metre",1,  
    AUTHORITY["EPSG","9001"]],  
  AUTHORITY["EPSG","26943"]]
```

Calling gdalinfo On A GeoTiff File Continued

```
Origin = (1828467.919737499905750,646446.278891499969177)
Pixel Size = (0.304800999999999,-0.304800999999999)
Metadata:
  AREA_OR_POINT=Area
Image Structure Metadata:
  INTERLEAVE=PIXEL
Corner Coordinates:
Upper Left  ( 1828467.920,  646446.279)
Lower Left  ( 1828467.920,  645135.939)
Upper Right ( 1829778.259,  646446.279)
Lower Right ( 1829778.259,  645135.939)
Center      ( 1829123.089,  645791.109)
Band 1 Block=4299x1 Type=Byte, ColorInterp=Red
Band 2 Block=4299x1 Type=Byte, ColorInterp=Green
Band 3 Block=4299x1 Type=Byte, ColorInterp=Blue
```

- Displays georeference and color map information too.

Calling gdal_translate On A GeoTiff File

The following is an example executed from Windows. The continuation character in Windows is ^, in UNIX it's \.

```
CMD> gdal_translate -of georaster sf1.tif ^
      georaster:student/student,,table_name,raster_column ^
      -co "DESCRIPTION=(ID NUMBER,raster_column SDO_GEOASTER)" ^
      -co "INSERT=VALUES(101, SDO_GEO.INIT('student_rdt_01',1))" ^
      -co "BLOCKXSIZE=512" -co "BLOCKYSIZE=512"
```

- -of (output format) is `georaster`
- Connection information specified as “`user_name/ password,optional tnsname`”. It’s faster to load rasters without going over SQL*Net (no tnsname).
- -co (create options)
 - `Description` – If specified, creates a table for you. Ignored if table exists.

Calling `gdal_translate` On A `GeoTiff` File Continued

- `-co` (create options, continued)
 - `Insert`
 - Inserts the specified values for a row.
 - As parameters to `SDO_GEOR.INIT`, pass in a raster data table name, along with a unique raster id
 - Raster id is system generated if not specified.
 - `BlockXsize` and `BlockYsize`
 - X and y pixel sizes to use when blocking the raster
 - `BlockBsize`
 - Number of bands per block when blocking the raster
 - `Interleave`
 - How to interleave when blocking the raster.
 - Possible values are: `Band`, `Pixel`, `Line`, `BSQ`, `BIP`, `BIL`
(default `BSQ`)

Calling `gdal_translate` On A `GeoTiff` File Continued

- `-co` (create options, continued)
 - `SRID`
 - Model SRID of the raster.
 - Don't have to specify this create option for `GeoTiffs`.
 - If the raster model SRID does not match the indexed `spatialExtent` SRID, the `spatialExtent` will be set to `NULL`. In that case, in a post process step, you can set the `NULL` `spatialExtents` appropriately.