Oracle Database 10*g*

Managing Geographic Raster Data Using GeoRaster

An Oracle Technical White Paper
December 2003

Table Of Contents

1 GeoRaster

1.1 Introduction

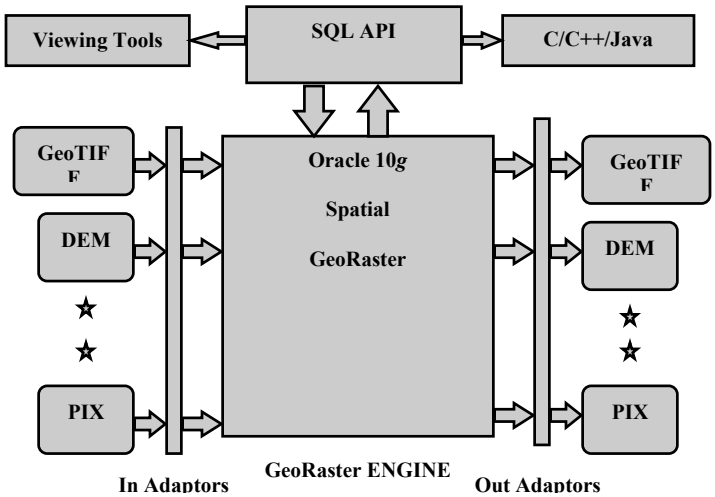
GeoRaster is a feature of Oracle Spatial in Oracle Database 10*g* that lets you store, index, query, analyze, and deliver GeoRaster data, that is, image and gridded raster data and its associated metadata. GeoRaster provides Oracle Spatial data types and an object-relational schema. You can use these data types and schema objects to store multidimensional gridded data and raster layers that can be referenced to positions on the Earth's surface or in a local coordinate system. If the data is georeferenced, you can find the location on Earth for a cell in an raster; or given a location on Earth, you can find the cell in a raster layer associated with that location.

GeoRaster is designed to deliver enterprise-class data management capability to large image processing solutions. It is now possible for developers to integrate this powerful data management technology with the leading image processing and raster/grid analysis tools.

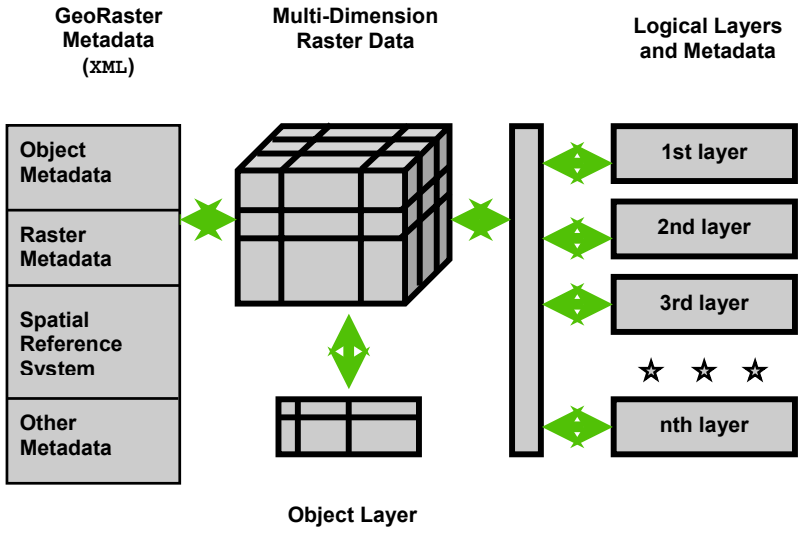
1.2 Architecture

The GeoRaster architecture provides the core functionality needed to support the use of image or grid-based raster data in Oracle Database 10*g*. At a very high level of abstraction, the GeoRaster architecture includes six basic components:

1. GeoRaster Engine – Core GeoRaster functionality includes data, metadata, methods and indexing.
2. SQL API – SQL access to the raster and grid-based data in GeoRaster.
3. C/C++/Java – OCI, OCCI, and Java access to the raster and grid-based data in GeoRaster with or without calling the GeoRaster API
4. Viewing Tools: A variety of third party viewing and analysis tools now support GeoRaster. In addition, a free downloadable viewer is available from Oracle.
5. Input [data] adapters – Facilitate loading raster data from well-known file formats into GeoRaster.
6. Output [data] adapters – Facilitate unloading raster data into well-known file formats from GeoRaster.

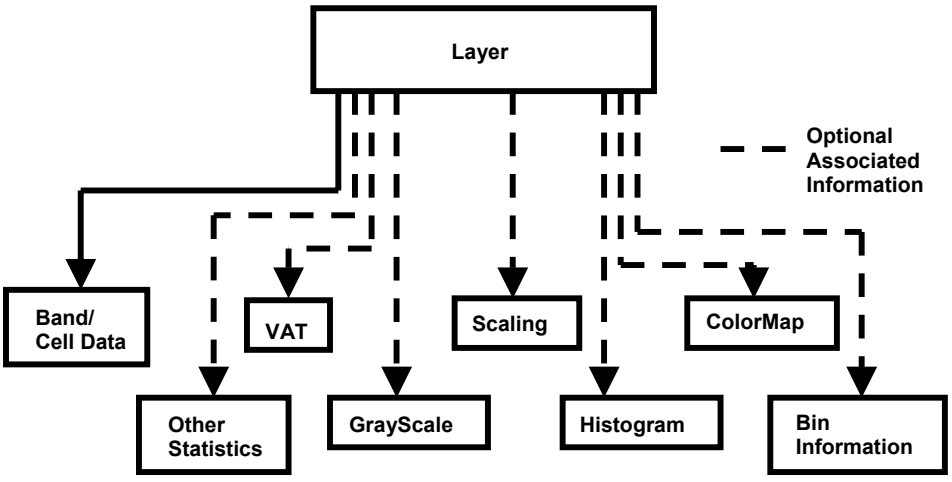


At a somewhat less abstract level the GeoRaster architecture is a blend of cell data plus GeoRaster Metadata including layer metadata. The cell data contains the actual raster

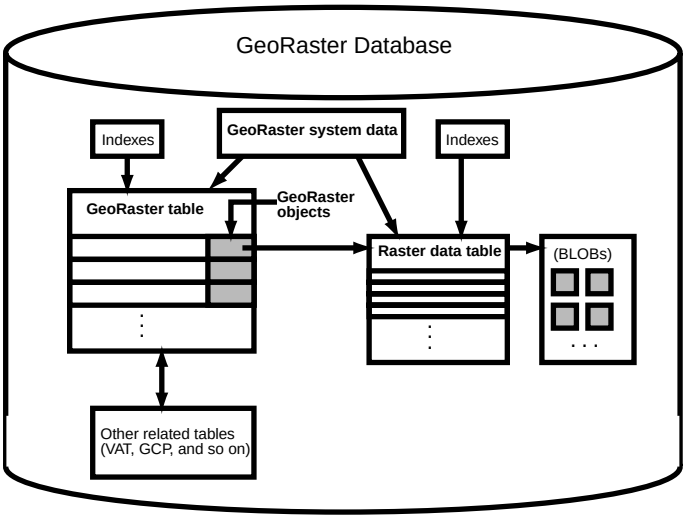


data itself. The GeoRaster metadata contains XML representations of metadata that describes the raster data. This metadata includes object metadata such as description and version information. GeoRaster also includes raster metadata for cell depth (1BIT, 32BIT_S, or 64BIT_REAL), dimensionality, blocking, interleaving and other information. Additionally, Spatial Reference System metadata, containing information for the affine transformation required for georeferencing, can also be stored.

Additional metadata, such as metadata pertaining to each layer in a GeoRaster object, can be stored in the database as well. If the data is grid data, one or more Value Attribute Tables (VAT) can be used to maintain information about the values stored in each layer (e.g. elevation value, saturation level etc.). In addition, there is a comprehensive suite of metadata used to capture and track image/cell attribution, scaling factors, color related information (color map, grayscale), histogram and other layer-based attributes essential for image management and use by client applications.



Finally, looking at the GeoRaster architecture from a much less abstract perspective the physical schema for storing and managing raster or grid-based data in the database can be examined. A GeoRaster data table has data column of type SDO_GEORASTER. SDO_GEORASTER objects include information about how to retrieve to GeoRaster cell data that is stored in a raster data table, otherwise known as an object table of type SDO_RASTER. The SDO_RASTER type includes a BLOB column called RASTERBLOCK, which stores the real raster blocks. Other information associated with the GeoRaster data table can be stored in separate columns or tables, such as a Value Attribute Table (VAT).

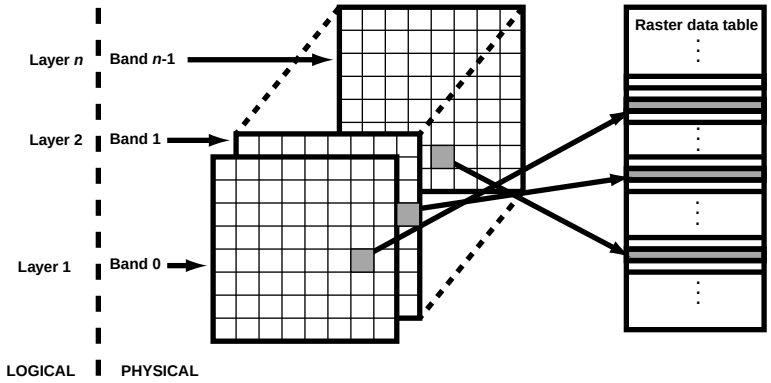


The specifics of the GeoRaster data model and how this architecture is implemented in Oracle Database 10*g* are provided in the sections below.

1.3 Data Model

GeoRaster uses a generic raster data model that is component-based, logically layered, and multidimensional. The core data in a raster is a multidimensional matrix of raster cells. Each cell is one element of the matrix, and its value is called the cell value. The matrix has a number of dimensions, a cell depth, and a size for each dimension. The cell depth is the data size of the value of each cell. It also defines the range of all cell values. The cell depth applies to each single cell, not an array of cells. This core raster data set can be blocked for optimal storage, retrieval and processing.

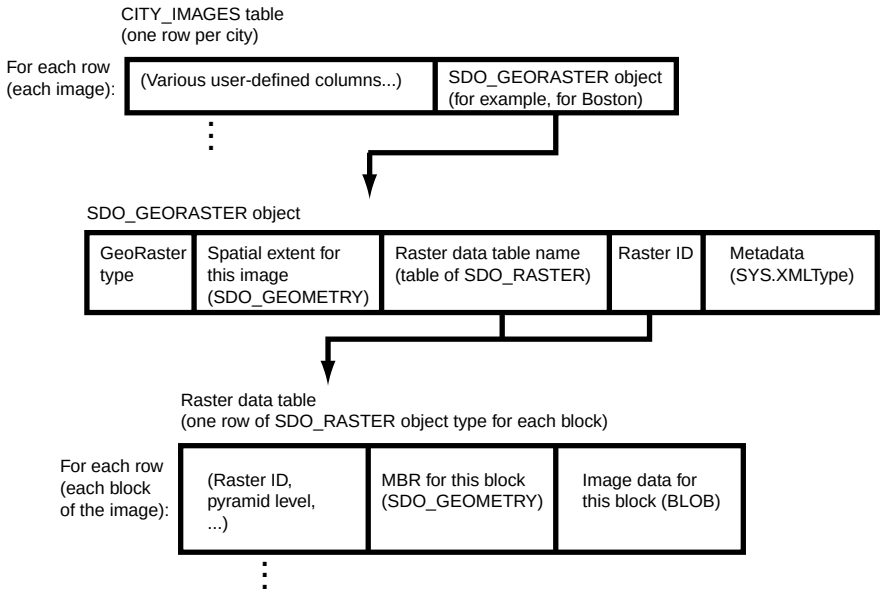
The data model is logically layered. The core data is called the object layer or layer 0, and consists of one or more logical layers (or sublayers). For example, for multi-channel remote sensing imagery, the layers are used to model the channels or bands of the imagery. In GeoRaster, each layer is a two-dimensional matrix of cells that consists of the row dimension and the column dimension. The relationship between logical layers in the GeoRaster data model and the physical bands or channels of the source image data is depicted as follows.



A GeoRaster object has specific metadata associated with it. Each layer of the GeoRaster data can have its own metadata and attributes. In the GeoRaster data model, metadata is everything but the core cell matrix and it is stored as an XML type in the database. GeoRaster metadata is further divided into different components that contain, but are not limited to, the following information:

- Object information
- Raster information
- Spatial reference system information
- Date and time (temporal reference system) information
- Spectral (band reference system) information
- Layer information for each layer

The physical structure of the GeoRaster model illustrates how the data interacts, as shown in the following figure.



As shown in the preceding figure, GeoRaster tables, created by a user, can contain attribute data (just like any other Oracle table). However, a GeoRaster table will also contain a column of object type SDO_GEORASTER. The SDO_GEORASTER object type has several attributes, two of which are needed to store and select cell data from the SDO_GEORASTER object: RASTERDATATABLE and the RASTERID provide the information required to store and retrieve the raster cell data.

1.4 GeoRaster Object

As discussed above GeoRaster data consists of a multidimensional matrix of cells and the GeoRaster metadata that is stored as an XML document using the Oracle XML data type. The metadata is defined according to the published GeoRaster metadata XML schema. The spatial extent (footprint) of a GeoRaster object is part of the metadata, but it is stored separately as an attribute of the GeoRaster object. This allows GeoRaster to take advantage of the spatial geometry type and related capabilities, such as using R-tree indexing on GeoRaster objects.

The multidimensional matrix of cells in the GeoRaster object is blocked (or tiled) into small subsets for optimal storage and retrieval. Blocking also allows GeoRaster objects to be large scale in size. Each block is stored in a table as a binary large object (BLOB), and a geometry object (of type SDO_GEOMETRY) is used to define the precise extent of the block. Each row of the table stores only one block and the blocking information related to that block. This same blocking scheme is used to support pyramids as well.

2 Features of GeoRaster

GeoRaster provides a rich set of foundation methods in addition to providing both a logical model and a physical model that facilitate data management for raster information in the Oracle Database 10*g*. This section provides a general overview of the basic functional infrastructure available in GeoRaster.

GeoRaster methods can be grouped into the following categories:

- Database Administration (Create table, Load and Export, etc.)
- Data Manipulation (Copy, Modify, Index, Subset, Transform)
- Cell Data and Metadata Update and Query

2.1 Database Administration

Database Administration includes functions performed to instantiate the GeoRaster environment in a database. Key methods targeting database administration for GeoRaster include:

- SDO_GEOR.init - Initializes an empty GeoRaster object, which will be registered by GeoRaster in the xxx_SDO_GEOR_SYSDATA views.
- SDO_GEOR.createBlank - Creates a blank GeoRaster object, in which all cells have the same value.
- SDO_GEOR.copy - Makes a copy of an existing GeoRaster object.

- SDO_GEOR.importFrom - Imports an image or gridded raster data in a file or BLOB object into a GeoRaster object stored in the database.
- SDO_GEOR.exportTo - Exports a GeoRaster object or a subset of a GeoRaster object to a file or to a BLOB object.
- SDO_GEOR.validateGeoraster - Validates a GeoRaster object.
- SDO_GEOR.schemaValidate - Validates a GeoRaster object's metadata against the GeoRaster XML schema.

2.2 Data Manipulation

Data manipulation highlights operations to optimally manage GeoRaster data in support of various application requirements. Important concepts include:

Georeferencing - GeoRaster currently supports six-parameter affine transformation that georeferences two-dimensional raster data. The affine transformation is a special type of the Functional Fitting polynomial model. If an affine transformation is provided and is valid in the metadata, the GeoRaster object is considered georeferenced. This is validated when the isReferenced value in the SRS metadata is specified as TRUE..

Pyramiding - Pyramid levels represent reduced or increased resolution of a raster object that require less or more storage space, respectively. (GeoRaster currently supports only reduced resolution pyramids.) A pyramid level of 0 indicates the original raster data. In other words, there is no reduction in the resolution and no change in the storage space required. Values greater than 0 (zero) indicate increasingly reduced levels of resolution and reduced storage space requirements.

Change Format - Under the GeoRaster architecture, all raster cell data is stored in the Raster Data Table. How the cell data is physically organized in the Raster Data Table depend on cell depth, blocking sizes and interleaving types. These parameters can be flexibly changed for any existing GeoRaster object. Application developers can use these parameter functions as a tuning device for physically adjusting the GeoRaster object to addresses various application requirements. This eliminates the need for off-line re-adjustment and loading of the original raster data, or readjusting settings of existing application software.

Subsetting – Performs either or both of the following operations: (1) spatial crop, cut, clip, or (2) layer or band subset or duplicate.

Scaling – Enlarges or reduces a GeoRaster object while having a choice of resampling methods (nearest neighbor, bilinear interpolation, cubic convolution, average) using four or six neighboring cells.

The following is a list of key methods provided by GeoRaster:

- SDO_GEOR.changeFormat - Changes the storage format of an existing GeoRaster object, including changes of blocking, cell depth and/or interleaving.

- SDO_GEOR.changeFormatCopy - Makes a copy of an existing GeoRaster object using a different storage format.
- SDO_GEOR.generateSpatialExtent - Generates a Spatial geometry that contains the spatial extent of the GeoRaster object.
- SDO_GEOR.georeference - Georeferences a GeoRaster object using specified cell-to-model transformation coefficients.
- SDO_GEOR.generatePyramid - Generates pyramid data for a GeoRaster object.
- SDO_GEOR.deletePyramid - Deletes pyramid data for a GeoRaster object.
- SDO_GEOR.subset - Performs either (or both) of the following operations: (a) spatial crop, cut, or clip, or (b) layer or band subset.
- SDO_GEOR.scale - Scales (enlarges or reduces) a GeoRaster object.
- SDO_GEOR.scaleCopy - Scales (enlarges or reduces) a GeoRaster object and inserts the result into a new object that reflects the scaling.
- SDO_GEOR.mosaic - Mosaics seamless GeoRaster objects into one GeoRaster object.

2.3 Cell Data and Metadata Update and Query

GeoRaster cell data and metadata update and query are crucial to successful use of GeoRaster in the Oracle Database 10*g* environment. Many methods are provided for these purposes, and some key methods include:

- SDO_GEOR.getID - Returns the user-defined identifier value associated with a GeoRaster object.
- SDO_GEOR.setID - Sets a user-defined identifier to be associated with a GeoRaster object, or deletes the existing value if you specify a null id parameter.
- SDO_GEOR.getVersion - Returns the user-specified version of a GeoRaster object.
- SDO_GEOR.setVersion - Sets the user-specified version of a GeoRaster object.
- SDO_GEOR.getInterleavingType - Returns the interleaving type for a GeoRaster object.
- SDO_GEOR.getSpatialDimNumber - Returns the number of spatial dimensions of a GeoRaster object.
- SDO_GEOR.getSpatialDimSizes - Returns the number of cells in each spatial dimension of a GeoRaster object.
- SDO_GEOR.getTotalLayerNumber - Returns the total number of layers in a GeoRaster object.
- SDO_GEOR.getBlockSize - Returns the number of cells for each dimension in each block of a GeoRaster object as an array showing the number of cells for each row, column, and (if relevant) band.
- SDO_GEOR.isSpatialReferenced - Returns TRUE if the GeoRaster object is spatially referenced, or FALSE if the GeoRaster object is not spatially referenced.
- SDO_GEOR.setSpatialReferenced - Specifies whether or not a GeoRaster object is spatially referenced, or deletes the existing value if you specify a null isReferenced parameter.

- SDO_GEOR.getSRS - Returns information related to the spatial referencing of a GeoRaster object.
- SDO_GEOR.setSRS - Sets the spatial reference information of a GeoRaster object, or deletes the existing information if you specify a null srs parameter.
- SDO_GEOR.getModelSRID - Returns the coordinate system (SDO_SRID value) associated with the model (ground) space for a GeoRaster object.
- SDO_GEOR.setModelSRID - Sets the coordinate system (SDO_SRID value) for the model (ground) space for a GeoRaster object, or deletes the existing value if you specify a null srid parameter.
- SDO_GEOR.getBeginDateTime - Returns the beginning date and time for raster data collection in the metadata for a GeoRaster object.
- SDO_GEOR.setBeginDateTime - Sets the beginning date and time for raster data collection in the metadata for a GeoRaster object, or deletes the existing value if you specify a null beginTime parameter.
- SDO_GEOR.hasPseudoColor - Checks if a layer contains pseudocolor information (colormap).
- SDO_GEOR.getColorMap - Returns the colormap representing pseudocolor display of a layer.
- SDO_GEOR.setColorMap - Sets the colormap for a layer in a GeoRaster object, or deletes the existing value if you specify a null colorMap parameter.
- SDO_GEOR.getVAT - Returns the name of the value attribute table (VAT) associated with a layer.
- SDO_GEOR.setVAT - Sets the name of the value attribute table (VAT) associated with a layer of a GeoRaster object, or deletes the existing value if you specify a null vatName parameter.
- SDO_GEOR.getPyramidMaxLevel - Returns the level number of the top pyramid of a GeoRaster object.
- SDO_GEOR.getModelCoordinate - Returns the coordinates in the model (ground) coordinate system associated with the point at the specified cell (raster) coordinates.
- SDO_GEOR.getCellCoordinate - Returns the coordinates in the cell (raster) coordinate system associated with the point at the specified model (ground) coordinates.
- SDO_GEOR.getCellValue - Returns the value of a single cell located anywhere in the GeoRaster object by specifying its row, column, and band number in its cell coordinate system, or by specifying a point geometry in the model coordinate system and its logical layer number.
- SDO_GEOR.changeCellValue - Updates the value of raster data cells in a specified window and specified bands of a GeoRaster object.
- SDO_GEOR.getRasterSubset – A generic query resulting in a single BLOB object containing all cells of a specified pyramid level that are inside or on the boundary of a specified window and containing only the specified layers.
- SDO_GEOR.getRasterData - Creates a single BLOB object that contains all raster data of the input GeoRaster object at the specified pyramid level.

Many of the functions discussed above are extended, augmented or leveraged by partner technologies delivered as extract/transform/load (ETL) tools, comprehensive remote sensing and image processing client tools, or in the form of visualization engines built on top of the GeoRaster model.

3 Summary

The introduction of GeoRaster in Oracle Database 10*g* creates significant new capacity for managing large volumes of raster data. Oracle is the only provider of commercial database management software that can store raster data and grid-based geospatial information in the database as a named type. Moreover, a full range of tools to populate, manage and manipulate both the cell-based data and the essential metadata is provided as part of the GeoRaster infrastructure. It is now possible for users of file-based image processing and raster data applications to benefit from the scalability, security and performance of Oracle Database 10*g* to support the mission critical applications.

Oracle Database 10*g*
Managing Geographic Raster Data Using GeoRaster
An Oracle Technical White Paper
Authors: Jim Farley and Jeffrey Xie
Contributing Author: Xavier Lopez

Oracle Corporation World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065
U.S.A.
Phone 650.506.7000
Fax 650.506.7200

International Inquiries:
Phone 44.932.872.020
Telex 851.927444(ORACLEG)
Fax 44.932.874.625

http://www.oracle.com