

GDAL Data Model

This document attempts to describe the GDAL data model. That is the types of information that a GDAL data store can contain, and their semantics.

Dataset

A dataset (represented by the [GDALDataset](#) class) is an assembly of related raster bands and some information common to them all. In particular the dataset has a concept of the raster size (in pixels and lines) that applies to all the bands. The dataset is also responsible for the georeferencing transform and coordinate system definition of all bands. The dataset itself can also have associated metadata, a list of name/value pairs in string form.

Note that the GDAL dataset, and raster band data model is loosely based on the OpenGIS Grid Coverages specification.

Coordinate System

Dataset coordinate systems are represented as OpenGIS Well Known Text strings. This can contain:

- An overall coordinate system name.
- A geographic coordinate system name.
- A datum identifier.
- An ellipsoid name, semi-major axis, and inverse flattening.
- A prime meridian name and offset from Greenwich.
- A projection method type (i.e. Transverse Mercator).
- A list of projection parameters (i.e. central_meridian).
- A units name, and conversion factor to meters or radians.
- Names and ordering for the axes.
- Codes for most of the above in terms of predefined coordinate systems from authorities such as EPSG.

For more information on OpenGIS WKT coordinate system definitions, and mechanisms to manipulate them, refer to the [osr_tutorial](#) document and/or the [OGRSpatialReference](#) class documentation.

The coordinate system returned by [GDALDataset::GetProjectionRef\(\)](#) describes the georeferenced coordinates implied by the affine georeferencing transform returned by [GDALDataset::GetGeoTransform\(\)](#). The coordinate system returned by [GDALDataset::GetGCPProjection\(\)](#) describes the georeferenced coordinates of the GCPs returned by [GDALDataset::GetGCPs\(\)](#).

Note that a returned coordinate system strings of "" indicates nothing is known about the georeferencing coordinate system.

Affine GeoTransform

GDAL datasets have two ways of describing the relationship between raster positions (in pixel/line coordinates) and georeferenced coordinates. The first, and most commonly used is the affine transform (the other is GCPs).

The affine transform consists of six coefficients returned by `GDALDataset::GetGeoTransform()` which map pixel/line coordinates into georeferenced space using the following relationship:

$$\begin{aligned} X_{\text{geo}} &= GT(0) + X_{\text{pixel}} * GT(1) + Y_{\text{line}} * GT(2) \\ Y_{\text{geo}} &= GT(3) + X_{\text{pixel}} * GT(4) + Y_{\text{line}} * GT(5) \end{aligned}$$

In case of north up images, the GT(2) and GT(4) coefficients are zero, and the GT(1) is pixel width, and GT(5) is pixel height. The (GT(0),GT(3)) position is the top left corner of the top left pixel of the raster.

Note that the pixel/line coordinates in the above are from (0.0,0.0) at the top left corner of the top left pixel to (width_in_pixels,height_in_pixels) at the bottom right corner of the bottom right pixel. The pixel/line location of the center of the top left pixel would therefore be (0.5,0.5).

GCPs

A dataset can have a set of control points relating one or more positions on the raster to georeferenced coordinates. All GCPs share a georeferencing coordinate system (returned by `GDALDataset::GetGCPProjection()`). Each GCP (represented as the `GDAL_GCP` class) contains the following:

```
typedef struct
{
    char        *pszId;
    char        *pszInfo;
    double      dfGCPPixel;
    double      dfGCPLine;
    double      dfGCPX;
    double      dfGCPY;
    double      dfGCPZ;
} GDAL_GCP;
```

The pszId string is intended to be a unique (and often, but not always numerical) identifier for the GCP within the set of GCPs on this dataset. The pszInfo is usually an empty string, but can contain any user defined text associated with the GCP. Potentially this can also contain machine parsable information on GCP status though that isn't done at this time.

The (Pixel,Line) position is the GCP location on the raster. The (X,Y,Z) position is the associated georeferenced location with the Z often being zero.

The GDAL data model does not imply a transformation mechanism that must be generated from the GCPs ... this is left to the application. However 1st to 5th order polynomials are common.

Normally a dataset will contain either an affine geotransform, GCPs or neither. It is uncommon to have both, and it is undefined which is authoritative.

Metadata

GDAL metadata is auxiliary format and application specific textual data kept as a list of name/value pairs. The names are required to be well behaved tokens (no spaces, or odd characters). The values can be of any length, and contain anything except an embedded null (ASCII zero).

The metadata handling system is not well tuned to handling very large bodies of metadata. Handling of more than 100K of metadata for a dataset is likely to lead to performance degradation.

Some formats will support generic (user defined) metadata, while other format drivers will map specific format fields to metadata names. For instance the TIFF driver returns a few information tags as metadata including the date/time field which is returned as:

```
TIFFTAG_DATETIME=1999:05:11 11:29:56
```

Metadata is split into named groups called domains, with the default domain having no name (NULL or ""). Some specific domains exist for special purposes. Note that currently there is no way to enumerate all the domains available for a given object, but applications can "test" for any domains they know how to interpret.

The following metadata items have well defined semantics in the default domain:

- **AREA_OR_POINT:** May be either "Area" (the default) or "Point". Indicates whether a pixel value should be assumed to represent a sampling over the region of the pixel or a point sample at the center of the pixel. This is not intended to influence interpretation of georeferencing which remains area oriented.
- **NODATA_VALUES:** The value is a list of space separated pixel values matching the number of bands in the dataset that can be collectively used to identify pixels that are nodata in the dataset. With this style of nodata a pixel is considered nodata in all bands if and only if all bands match the corresponding value in the NODATA_VALUES tuple. This metadata is not widely honoured by GDAL drivers, algorithms or utilities at this time.
- **MATRIX_REPRESENTATION:** This value, used for Polarimetric SAR datasets, contains the matrix representation that this data is provided in. The following are acceptable values:
 - SCATTERING
 - SYMMETRIZED_SCATTERING
 - COVARIANCE
 - SYMMETRIZED_COVARIANCE
 - COHERENCY
 - SYMMETRIZED_COHERENCY
 - KENNAUGH
 - SYMMETRIZED_KENNAUGH
- **POLARIMETRIC_INTERP:** This metadata item is defined for Raster Bands for polarimetric SAR data. This indicates which entry in the specified matrix representation of the data this band represents. For a dataset provided as a scattering matrix, for example, acceptable values for this metadata item are HH, HV, VH, VV. When the dataset is a covariance matrix, for example, this metadata item will be one of Covariance_11, Covariance_22, Covariance_33, Covariance_12, Covariance_13, Covariance_23 (since the matrix itself is a hermitian matrix, that is all the data that is required to describe the matrix).
- **METADATATYPE:** If IMAGERY Domain present, the item consist the reader which processed the metadata. Now present such readers:
 - DG: DigitalGlobe imagery metadata
 - GE: GeoEye (or formally SpaceImaging) imagery metadata
 - OV: OrbView imagery metadata
 - DIMAP: Pleiades imagery metadata
 - MSP: Resurs DK-1 imagery metadata

- ODL: Landsat imagery metadata
- **CACHE_PATH**: A cache directory path. Now this metadata item sets only by WMS driver. This is useful when dataset deletes with cached data or when external program need to put tiles in cache for some area of interest.

SUBDATASETS Domain

The SUBDATASETS domain holds a list of child datasets. Normally this is used to provide pointers to a list of images stored within a single multi image file.

For example, an NITF with two images might have the following subdataset list.

```
SUBDATASET_1_NAME=NITF_IM:0:multi_1b.ntf
SUBDATASET_1_DESC=Image 1 of multi_1b.ntf
SUBDATASET_2_NAME=NITF_IM:1:multi_1b.ntf
SUBDATASET_2_DESC=Image 2 of multi_1b.ntf
```

The value of the **_NAME** is the string that can be passed to **GDALOpen()** to access the file. The **_DESC** value is intended to be a more user friendly string that can be displayed to the user in a selector.

Drivers which support subdatasets advertize the **DMD_SUBDATASETS** capability. This information is reported when the **--format** and **--formats** options are passed to the command line utilities.

Currently, drivers which support subdatasets are: ADRG, ECRGTOC, GEORASTER, GTiff, HDF4, HDF5, netCDF, NITF, NTV2, OGD, PDF, PostGISRaster, Rasterlite, RPFTOC, RS2, WCS, and WMS.

IMAGE_STRUCTURE Domain

Metadata in the default domain is intended to be related to the image, and not particularly related to the way the image is stored on disk. That is, it is suitable for copying with the dataset when it is copied to a new format. Some information of interest is closely tied to a particular file format and storage mechanism. In order to prevent this getting copied along with datasets it is placed in a special domain called **IMAGE_STRUCTURE** that should not normally be copied to new formats.

Currently the following items are defined by [RFC 14](#) as having specific semantics in the **IMAGE_STRUCTURE** domain.

- **COMPRESSION**: The compression type used for this dataset or band. There is no fixed catalog of compression type names, but where a given format includes a **COMPRESSION** creation option, the same list of values should be used here as there.
- **NBITS**: The actual number of bits used for this band, or the bands of this dataset. Normally only present when the number of bits is non-standard for the datatype, such as when a 1 bit TIFF is represented through GDAL as **GDT_Byte**.
- **INTERLEAVE**: This only applies on datasets, and the value should be one of **PIXEL**, **LINE** or **BAND**. It can be used as a data access hint.
- **PIXELTYPE**: This may appear on a **GDT_Byte** band (or the corresponding dataset) and have the value **SIGNEDBYTE** to indicate the unsigned byte values between 128 and 255 should be interpreted as being values between -128 and -1 for applications that recognise the **SIGNEDBYTE** type.

RPC Domain

The RPC metadata domain holds metadata describing the Rational Polynomial Coefficient geometry model for the image if present. This geometry model can be used to transform between pixel/line and georeferenced locations. The items defining the model are:

- **ERR_BIAS:** Error - Bias. The RMS bias error in meters per horizontal axis of all points in the image (-1.0 if unknown)
- **ERR_RAND:** Error - Random. RMS random error in meters per horizontal axis of each point in the image (-1.0 if unknown)
- **LINE_OFF:** Line Offset
- **SAMP_OFF:** Sample Offset
- **LAT_OFF:** Geodetic Latitude Offset
- **LONG_OFF:** Geodetic Longitude Offset
- **HEIGHT_OFF:** Geodetic Height Offset
- **LINE_SCALE:** Line Scale
- **SAMP_SCALE:** Sample Scale
- **LAT_SCALE:** Geodetic Latitude Scale
- **LONG_SCALE:** Geodetic Longitude Scale
- **HEIGHT_SCALE:** Geodetic Height Scale
- **LINE_NUM_COEFF (1-20):** Line Numerator Coefficients. Twenty coefficients for the polynomial in the Numerator of the r_n equation. (space separated)
- **LINE_DEN_COEFF (1-20):** Line Denominator Coefficients. Twenty coefficients for the polynomial in the Denominator of the r_n equation. (space separated)
- **SAMP_NUM_COEFF (1-20):** Sample Numerator Coefficients. Twenty coefficients for the polynomial in the Numerator of the c_n equation. (space separated)
- **SAMP_DEN_COEFF (1-20):** Sample Denominator Coefficients. Twenty coefficients for the polynomial in the Denominator of the c_n equation. (space separated)

These fields are directly derived from the document prospective GeoTIFF RPC document (http://geotiff.maptools.org/rpc_prop.html) which in turn is closely modeled on the NITF RPC00B definition.

The line and pixel offset expressed with **LINE_OFF** and **SAMP_OFF** are with respect to the center of the pixel.

IMAGERY Domain (remote sensing)

For satellite or aerial imagery the IMAGERY Domain may be present. It depends on the existence of special metadata files near the image file. The files at the same directory with image file tested by the set of metadata readers, if files can be processed by the metadata reader, it fill the IMAGERY Domain with the following items:

- **SATELLITEID:** A satellite or scanner name
- **CLOUDCOVER:** Cloud coverage. The value between 0 - 100 or 999 if not available
- **ACQUISITIONDATETIME:** The image acquisition date time in UTC

xml: Domains

Any domain name prefixed with "xml:" is not normal name/value metadata. It is a single XML document stored in one big string.

Raster Band

A raster band is represented in GDAL with the [GDALRasterBand](#) class. It represents a single raster band/channel/layer. It does not necessarily represent a whole image. For instance, a 24bit RGB image would normally be represented as a dataset with three bands, one for red, one for green and one for blue.

A raster band has the following properties:

- A width and height in pixels and lines. This is the same as that defined for the dataset, if this is a full resolution band.
- A datatype (GDALDataType). One of Byte, UInt16, Int16, UInt32, Int32, Float32, Float64, and the complex types CInt16, CInt32, CFloat32, and CFloat64.
- A block size. This is a preferred (efficient) access chunk size. For tiled images this will be one tile. For scanline oriented images this will normally be one scanline.
- A list of name/value pair metadata in the same format as the dataset, but of information that is potentially specific to this band.
- An optional description string.
- An optional single nodata pixel value (see also NODATA_VALUES metadata on the dataset for multi-band style nodata values).
- An optional nodata mask band marking pixels as nodata or in some cases transparency as discussed in [RFC 15: Band Masks](#) and documented in [GDALRasterBand::GetMaskBand\(\)](#).
- An optional list of category names (effectively class names in a thematic image).
- An optional minimum and maximum value.
- Optional statistics stored in metadata:
 - STATISTICS_MEAN: mean
 - STATISTICS_MINIMUM: minimum
 - STATISTICS_MAXIMUM: maximum
 - STATISTICS_STDDEV: standard deviation
 - STATISTICS_APPROXIMATE: only present if GDAL has computed approximate statistics
 - STATISTICS_VALID_PERCENT: percentage of valid (not nodata) pixel
- An optional offset and scale for transforming raster values into meaning full values (i.e. translate height to meters).
- An optional raster unit name. For instance, this might indicate linear units for elevation data.
- A color interpretation for the band. This is one of:
 - GCI_Undefined: the default, nothing is known.
 - GCI_GrayIndex: this is an independent gray-scale image
 - GCI_PaletteIndex: this raster acts as an index into a color table
 - GCI_RedBand: this raster is the red portion of an RGB or RGBA image

- GCI_GreenBand: this raster is the green portion of an RGB or RGBA image
 - GCI_BlueBand: this raster is the blue portion of an RGB or RGBA image
 - GCI_AlphaBand: this raster is the alpha portion of an RGBA image
 - GCI_HueBand: this raster is the hue of an HLS image
 - GCI_SaturationBand: this raster is the saturation of an HLS image
 - GCI_LightnessBand: this raster is the hue of an HLS image
 - GCI_CyanBand: this band is the cyan portion of a CMY or CMYK image
 - GCI_MagentaBand: this band is the magenta portion of a CMY or CMYK image
 - GCI_YellowBand: this band is the yellow portion of a CMY or CMYK image
 - GCI_BlackBand: this band is the black portion of a CMYK image.
- A color table, described in more detail later.
 - Knowledge of reduced resolution overviews (pyramids) if available.

Color Table

A color table consists of zero or more color entries described in C by the following structure:

```
typedef struct
{
    /*- gray, red, cyan or hue -/
    short      c1;

    /*- green, magenta, or lightness -/
    short      c2;

    /*- blue, yellow, or saturation -/
    short      c3;

    /*- alpha or black band -/
    short      c4;
} GDALColorEntry;
```

The color table also has a palette interpretation value (GDALPaletteInterp) which is one of the following values, and indicates how the c1/c2/c3/c4 values of a color entry should be interpreted.

- GPI_Gray: Use c1 as gray scale value.
- GPI_RGB: Use c1 as red, c2 as green, c3 as blue and c4 as alpha.
- GPI_CMYK: Use c1 as cyan, c2 as magenta, c3 as yellow and c4 as black.
- GPI_HLS: Use c1 as hue, c2 as lightness, and c3 as saturation.

To associate a color with a raster pixel, the pixel value is used as a subscript into the color table. That means that the colors are always applied starting at zero and ascending. There is no provision for indicating a pre-scaling mechanism before looking up in the color table.

Overviews

A band may have zero or more overviews. Each overview is represented as a "free standing" **GDALRasterBand**. The size (in pixels and lines) of the overview will be different than the underlying raster, but the geographic region covered by overviews is the same as the full resolution band.

The overviews are used to display reduced resolution overviews more quickly than could be done by reading all the full resolution data and downsampling.

Bands also have a `HasArbitraryOverviews` property which is `TRUE` if the raster can be read at any resolution efficiently but with no distinct overview levels. This applies to some FFT encoded images, or images pulled through gateways (like OGDII) where downsampling can be done efficiently at the remote point.