



# State of GDAL
# GDAL 3.4 & 3.5

Even Rouault
*SPATIALYS*

August 26th 2021

# GDAL/OGR : Introduction

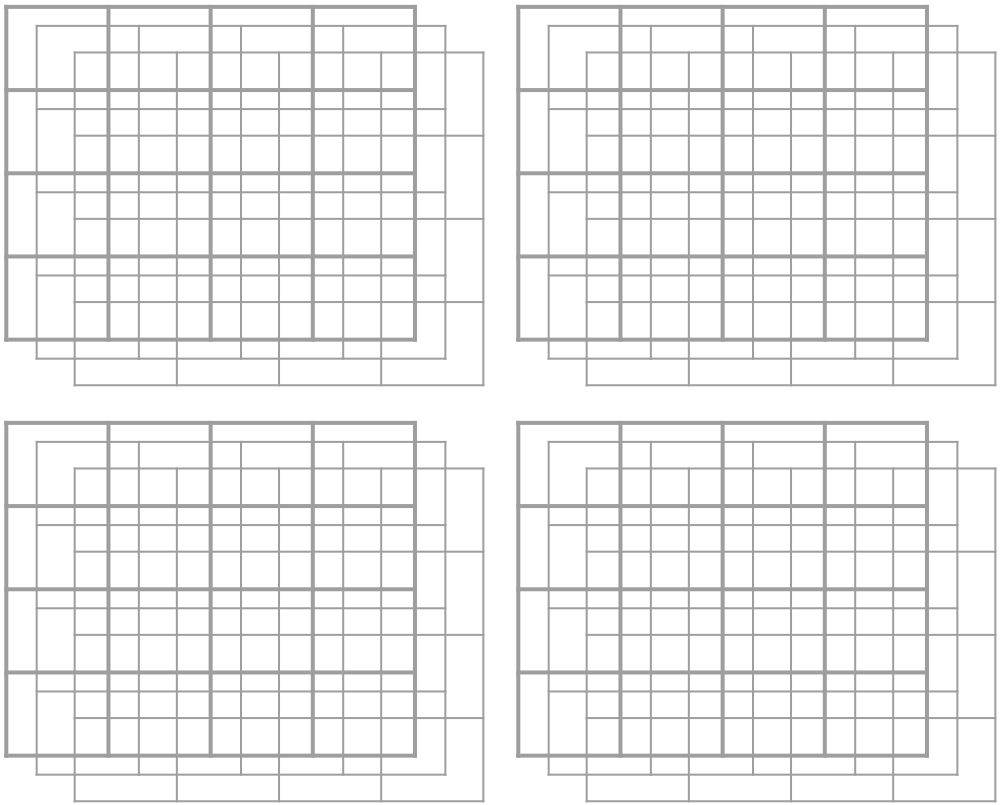
- GDAL? Geospatial Data Abstraction Library. The swiss army knife for geospatial.
- Read and write Raster (GDAL) and Vector (OGR) datasets
- 250 (mainly) geospatial formats and protocols.
- Widely used


(> 100 http://trac.osgeo.org/gdal/wiki/SoftwareUsingGdal)

- MIT Open Source license (permissive)

# GDAL 3.4: Zarr format


- Zarr: cloud-oriented format for storage of chunked, compressed, N-dimensional arrays



# GDAL 3.4: Zarr format


- Hierarchical organization of arrays in groups
- Data types: numeric, strings, compound
- Metadata in .json files
- Each chunk in a separate data file
- Several compression methods:
  ZLIB, GZIP, LZMA, ZSTD, LZ4, BLOSC
- Filters: delta, …
- Non-geo native. Some practice borrowed from netCDF-CF (Climate & Forecast) conventions
- Zarr V2 submitted as a candidate for a OGC community standard

# GDAL 3.4: Zarr driver


- Read/write capabilities
- Implements the GDAL multidimensional API
- Implements the GDAL “classic” 2D API
- Handles most data types
- Works with local and remote stores (AWS S3, Google Cloud Storage, MSFT Azure, …)
- Handles the Zarr V2 and V3 specifications
- Add CRS as an extension (WKT/PROJJSON)
- Some multi-threaded capabilities

# GDAL 3.4: Zarr. More reading…


- Format specification and Python reference implementation: https://zarr.readthedocs.io/en/stable/
- GDAL Zarr driver documentation: https://gdal.org/drivers/raster/zarr.html
- OGC Testbed 17: COG/Zarr Evaluation Engineering Report: http://docs.opengeospatial.org/per/21-032.html

# GDAL 3.4: raster STACIT driver

- STACIT = Spatio-Temporal Asset Catalog ITems
- Using the projection extension specification: https://github.com/stac-extensions/projection to add info about projection, size in pixels, resolution and geospatial extent.
- Uses VRT internally
- Exposes each asset type as a GDAL subdataset

# GDAL 3.4: STACIT driver

gdalinfo 'STACIT:"https://planetarycomputer.microsoft.com/api/stac/v1/search?collections=naip&bbox=-100,40,-99,41&datetime=2020-08-17T00:00:00Z%2F..":asset=image'

Driver: VRT/Virtual Raster*
Files:
/vsicurl?pc_url_signing=yes&pc_collection=naip&url=https%3A//naipeuwest.blob.core.windows.net/naip/v002/ne/2020/ne_060cm_2020/40099/m_4009928_nw_14_060_20200904.tif
[... snip …]
/vsicurl?pc_url_signing=yes&pc_collection=naip&url=https%3A//naipeuwest.blob.core.windows.net/naip/v002/ne/2020/ne_060cm_2020/40099/m_4009904_ne_14_060_20200904.tif

Size is 27580, 81670

Coordinate System is:

PROJCRS["NAD83 / UTM zone 14N",
[... snip …]

ID["EPSG",26914]]

Origin = (441600.000000000000000,4539144.000000000000000)

Pixel Size = (0.600000000000000,-0.600000000000000)

[... snip …]

Band 1 Block=128x128 Type=Byte, ColorInterp=Red

Description = Red

Band 2 Block=128x128 Type=Byte, ColorInterp=Green

Description = Green

Band 3 Block=128x128 Type=Byte, ColorInterp=Blue

Description = Blue

Band 4 Block=128x128 Type=Byte, ColorInterp=Undefined

Description = NIR

# GDAL 3.4: Coordinate epoch

- Coordinates of ground points in non plate-fixed CRS like WGS 84 (G1762), ITRF2014, ATRF2014, … move over time (plate tectonics)

  ⇒ need to be qualified with coordinate epoch (!= observation collection)
- Modeling in GDAL: one optional coordinate epoch attribute attached to a OGRSpatialReference object
- Used by OGRCoordinateTransformation class (when time not provided per-coordinate).

  ⇒ PROJ restriction: only static←→dynamic transformations supported. Not dynamic←→ dynamic currently

# GDAL 3.4: Coordinate epoch

- ogr2ogr/gdal_translate/gdalwarp: new options to set source/target coordinate epoch
- Formats updated to store coordinate epoch:
  - GeoTIFF
  - GeoPackage
  - FlatGeoBuf
  - JPEG2000 (through GeoTIFF encoding)
  - Persistent Auxiliary Metadata (.aux.xml)
  - GDAL VRT

# GDAL 3.5: CMake build system

- Aim: add a CMake build system, and remove existing autoconf & nmake systems
- Why?
  - Unification of build process rather than having 2 different ones for Windows vs Unix
  - No consistent capabilities and option naming
  - Non-optimal parallel builds with existing builds
  - Lacking: no header dependency tracking, …
  - Very good tooling for CMake (Visual Studio, qtcreator, …)
  - Users having been crying for CMake GDAL for years

# GDAL 3.5: CMake build system

- Plan & schedule:
  - GDAL 3.5: addition of CMake build system (CMake 3.9 minimum version). autoconf/nmake kept but deprecated
  - GDAL 3.6: only CMake. autoconf/nmake removed
- Credits:
  - Hiroshi Miura for the bootstrapping with cmake4gdal repository !
  - GDAL sponsorship program: for funding all the fine tuning & integration effort

# GDAL 3.5: GeoParquet & GeoArrow vector drivers



- Parquet is an open source, column-oriented data file format designed for efficient data storage and retrieval.
- Column-oriented = information for a given attribute is grouped by many rows
- Data analysis focused databases/systems: Snowflake, Google BigQuery, etc.
- GeoParquet 0.4.0 extension: defines metadata (CRS, etc.) and geometry encoding (WKB)
- Doc: https://gdal.org/drivers/vector/parquet.html and https://gdal.org/drivers/vector/arrow.html

# GDAL 3.5: Miscellaneous

- JPEG-XL codec for (Geo)TIFF (libjxl + internal libtiff copy of GDAL required). JPEG-XL:
  - “Next-gen” codec from the JPEG group
  - Lossless and lossy profiles
  - Many channels/bands
  - Up to 24-bit integer / 32-bit floating point
  - Libjxl: BSD 3-clause ref. implementation
- 64-bit integer data types for raster
- OGR SAP Hana vector driver (requires close source ODBC driver)
- Removal of a few legacy/unmaintained drivers

# GDAL 3.6 preview

- Column-oriented read API for vector layers, using Arrow array stream interface
  ⇒ https://gdal.org/development/rfc/rfc86_column_oriented_api.html
- Full open-source built-in support for creation / update support of (vector) Esri FileGeodatase (.gdb) ⇒ mostly make FileGDB closed-source SDK useless
- New drivers: JPEGXL, KTX2, BASISU

# Thanks to GDAL sponsors!

- Gold level:


- Silver level:


- Bronze level:


- Supporter level:

Kaplan Open Source Consulting  Umbra  Space Intelligence  Myles Sutherland


# Questions?

Links:

http://gdal.org/

Contact: even.rouault@spatialys.com