



COG : Cloud Optimized Geotiff

Cabieces Julien (Oslandia)

 @troopa81

 @CabiecesJ

FOSS4G Buenos Aires, 30/09/2021



g Cloud Optimized Geotiff

- g Raster file format
- g Optimized for cloud storage
- g

g What is it ?

- g Tiled
- g Support overviews
- g Rely on HTTP Get Range Request
- g GDAL can read/write



What use cases ?

g Relevant if ...

- g Lots of raster to store (and share)
 - g Reduce storage cost
- g Retrieve only a subset of raster (bands, extent)
- g Access to raw data
- g You don't want to care about infrastructure

g

g It's like...

- g TMS but with raw data
- g WCS without server

g



How it works ?

- g We generate a COG file

- g gdal_translate with **-of COG**

- g Push thanks to a cloud specific API (or Web interface)

- g

- g Cloud

- g Object Storage : Store/Fetch file with URLs (GET/POST HTTP)

- g Amazon, Google, Azure...

- g And OpenStack

- g Open source

- g Several cloud providers propose it (OVH for instance) ~ Standard

- g



Let's try it 1/3

g Minio

- g Cloud Object Storage OpenSource
- g Install with Docker

```
docker run \  
-p 9000:9000 \  
-p 9001:9001 \  
-e "MINIO_ROOT_USER=admin" \  
-e "MINIO_ROOT_PASSWORD=mypasswd" \  
quay.io/minio/minio server /data --console-address ":9001"
```

Let's try it 2/3

g Generate the COG

- g Example with a **Sentinel 2** LA image (20 meter resolution)
- g Merge **several bands** file in one file

```
gdal_merge.py -separate -o T31TCJ_20210828T104619_20m.tif *.jp2
```

- g → 13 bands / 366 MB of JPEG2000 → After merge : 863 MB
 - g JPEG2000 is an effective lossless compression

g Generate COG

```
gdal_translate T31TCJ_20210828T104619_20m.tif \ T31TCJ_20210828T104619_20m_cog.tif -of COG
```

- g → 1.3 GB
 - g Bigger because overviews (mostly)
- g Upload it on Minio



Let's try it (3/3)

g QGIS configuration

g Set env

```
export AWS_S3_ENDPOINT=172.17.0.4:9000
export AWS_REGION=us-east-1
export AWS_SECRET_ACCESS_KEY=myspasswd
export AWS_ACCESS_KEY_ID=admin
export AWS_VIRTUAL_HOSTING=FALSE
export AWS_HTTPS=NO
```

g Add Raster with HTTP protocol / Select Amazon S3

Gestionnaire des sources de données | Raster

Type de source

Fichier Protocole: HTTP(S), cloud, etc.

Protocole

Type: AWS S3

Compartiment ou conteneur: mybucket

Clé d'objet: T31TCJ_20210828T104619_20m_cog.tif

D'autres options de certificat sont requises, tel que documenté [ici](#).

Options

Explorateur
Vecteur
Raster
Mesh
Nuage de points
Texte Délimité
GeoPackage
GPS
SpatiaLite

g Demo time !!

- g Navigate
- g Change bands
- g Compute an **Ndvi index** (vegetation index)

```
( "T31TCJ_20210828T104619_20m_cog@8" -  
"T31TCJ_20210828T104619_20m_cog@4" ) /  
( "T31TCJ_20210828T104619_20m_cog@8" +  
"T31TCJ_20210828T104619_20m_cog@4" )
```


Limitations

g Performance will depend on

- g Number of selected bands
- g DataType (Number of byte per pixel) : Uint16, Float32...
- g Bandwidth
- g Your cloud provider
- g

g

Beware with processings

- g No extent parameters → download the whole image
 - g Example : Contour, Aspect
 - g Solution : clip raster by extent beforehand





QUESTIONS ?!

Blog post

Cabieces Julien,

 @troopa81

 @CabiecesJ

FOSS4G 2021, 30/09/2021



FOSS4G
BUENOS AIRES 2021 