# Transform GeoTIFF file to COG.

The folder contains three files: two Python scripts and one Jupyter Notebook.

1. tif\_to\_COG\_translate.py transforms a GeoTIFF to a Cloud-Optimized-GeoTIFF (COG).
2. tif\_to\_COG\_translate\_notebook.ipynb is the Jupyter Notebook version of the script.
3. validate\_cloud\_optimized\_geotiff.py is a Python script for check if a .tif file is or not a COG.

REQUIREMENTS:

1. GDAL - <https://gdal.org/index.html>
2. PROJ - <https://proj.org/about.html>

The script tif\_to\_COG\_translate.py transforms a GeoTIFF to COG using GDAL translator library and OS system call.

Parameters of function are:

1. Path of input file (e.g., *C:/User/……/input\_file.tif*)
2. Path of output file (e.g., *C:/User/……/output\_file.tif*)
3. Include internal overviews parameter.
4. Compression method. Typically, no compression, DEFLATE or LZW can be used for lossless, or JPEG for lossy. (Note that DEFLATE while more efficient than LZW can cause compatibility issues with some software packages)

tif\_to\_COG\_translate.py can be use in this way:

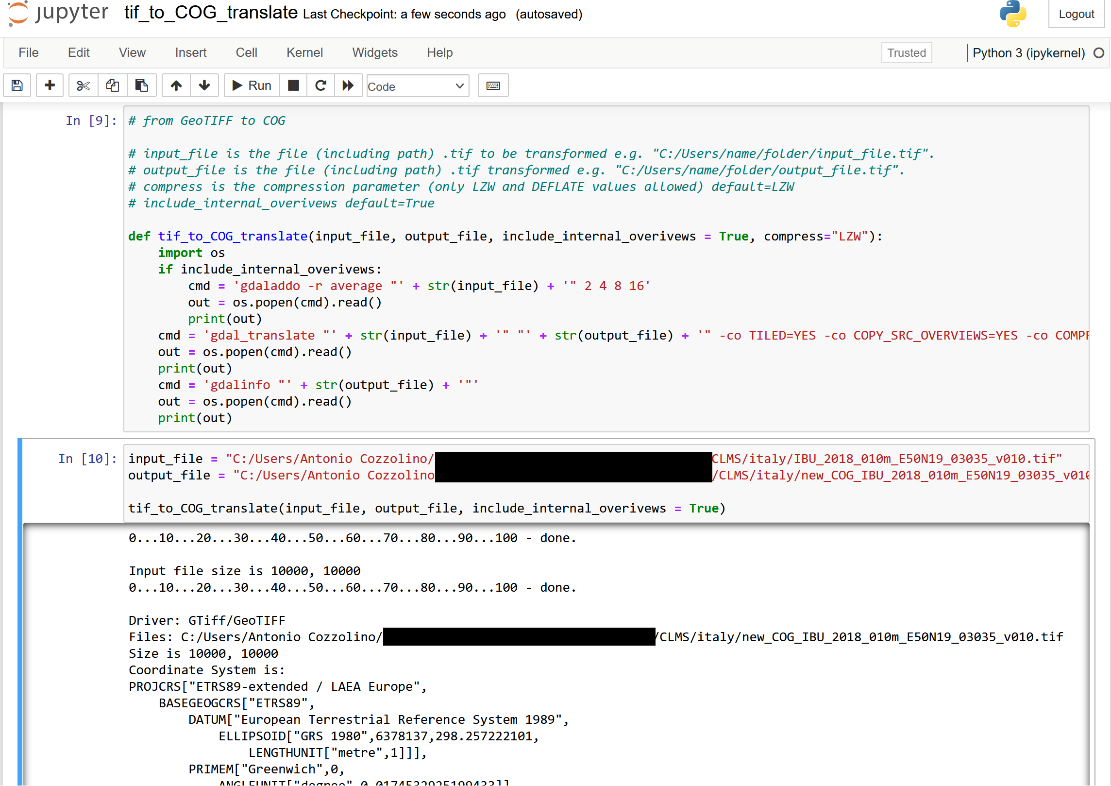


Figure - Example of use

## STEP BY STEP TRASFORMATION

This example was done using a GeoTIFF file downloaded from CLMS.   
The steps performed are:

1. Viewing the initial .tif file in QGIS, in order to make a comparison with the respective COG obtained.
2. Checking whether the initial file is COG or not.
3. Transformation of the initial file into COG.
4. Displaying the COG .tif file in QGIS in order to compare it with the respective initial file.
5. Verify whether the resulting file is indeed COG.

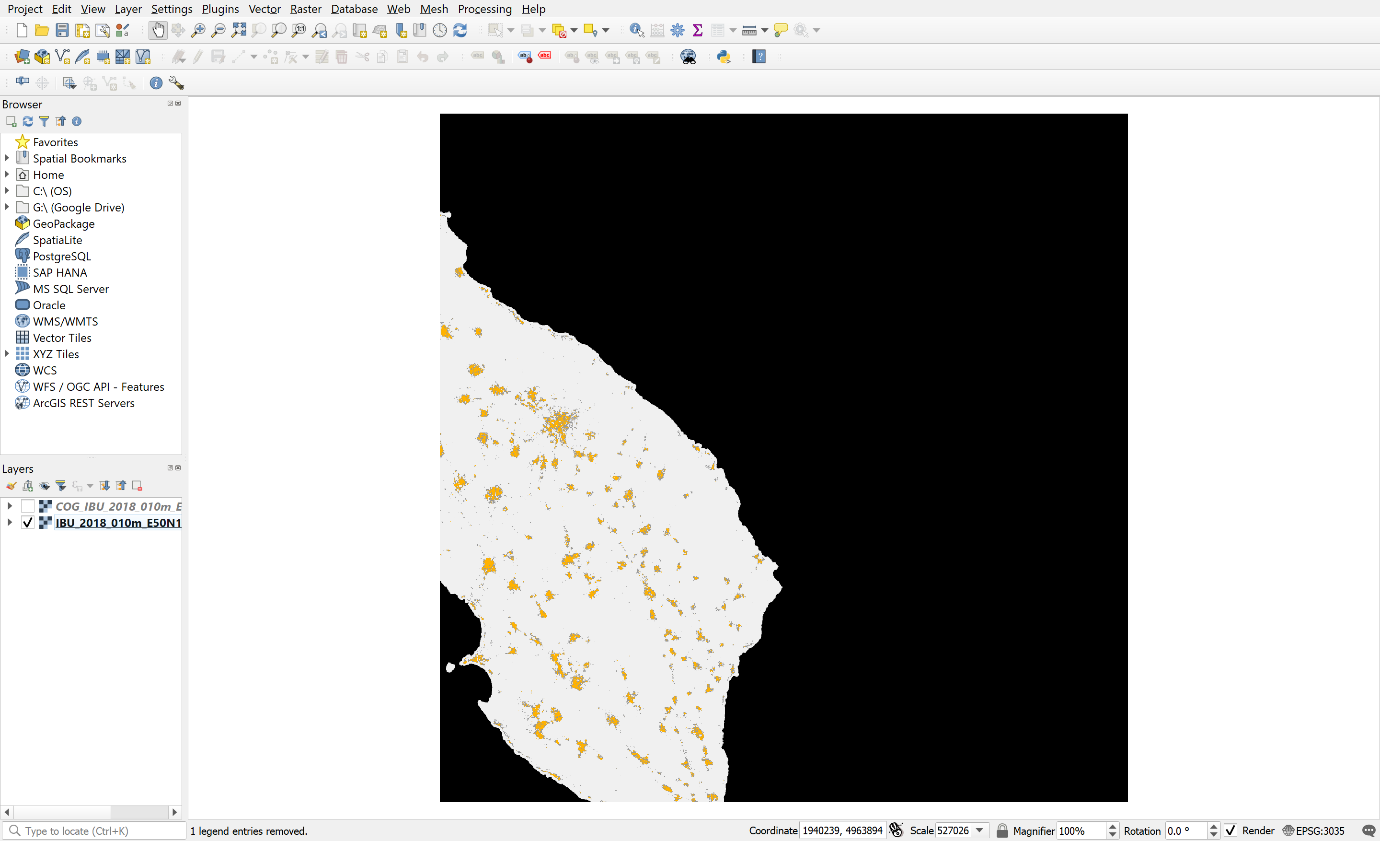


Figure - NO-COG file in QGIS

Graphical user interface, text

Description automatically generated

Figure - NO-COG information

Graphical user interface, text, application, email

Description automatically generated

Figure - NO-COG CRS

Now, using validate\_cloud\_optimized\_geotiff.py, we can see if the GeoTIFF file is really NO-COG.

Text

Description automatically generated

The file is not COG, let us now proceed with the transformation, using the script tif\_to\_COG\_translate.py.

Parameters used *are include\_internal\_overivews = True, compress="LZW".*

We obtained a new file which, when viewed in QGIS, appears:

Graphical user interface, application

Description automatically generated

Figure – COG file in QGIS

Graphical user interface, text, email

Description automatically generated

Figure - COG file information

Graphical user interface, text, application, email

Description automatically generated

Figure - COG file CRS

Now, using the script using validate\_cloud\_optimized\_geotiff.py we can check whether the resulting file is really COG.

Text

Description automatically generated

The file we obtained is COG!