<https://docs.geonode.org/en/master/install/basic/index.html>

DD/MM/YYYY

**Objetivo**:

**Proceso**:

**Resultado**:

**Observaciones**:

28/10/2024

**Objetivo**:

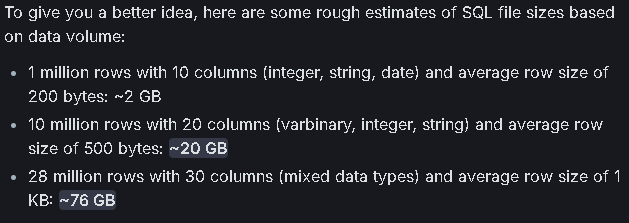
Subir capa raster a geoserver 🡪 crear el mapa base propio de la muni

**Proceso**:

1-Teastear subiendo un .tif… usado un .tif de 1,35 GB

2-SIN POSTGIS: <https://www.youtube.com/watch?v=m99XngrsUbo> video de como subir un raster .tif local

CON POSTGIS: esto para subirlo a postgis:  
~$> raster2pgsql.exe -s 5347 -I -M -C -F -t 250x250 "C:\loma.tif"> "C:\loma.sql"

tardó una hora aprox en crear, partiendo de un .tif de 1,35 GB, un .sql de 12,0 GB 

3-como buscar el problema de fondo en google/GPT: “How to efficiently create millions of overlapping raster tiles”

**Resultado**:

se pudo crear una capa raster base exportable a postgis…

falta optimizar, zoom muy lento

**Observaciones**:

Cambiar formato de imagen… a jpeg

29/10/2024

**Objetivo**:

Subir capa raster a geoserver OPTIMIZADA

**Proceso**:

continuando lo de ayer, se pueden subir capas raster, pero no optimizadas y le pesan a la bd

1- JPEG COMPRESSION en cortar raster por extension

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

2- prepárate para estar un rato largo:

Texto

Descripción generada automáticamente

3- raster2pgsql es extensión de postgres, mientras que gdal\_translate es extension de qgis…. En la sección observaciones esta el log de la operación:

**Resultado**:

el raster obtenido sigue siendo inutilizable… la calidad de compresión es mala y aun así pesa 1GB la imagen

**Observaciones**:

Siguiente pagina

Versión de QGIS: 3.34.8-Prizren

Revisión del código de QGIS: 91642333

Versión de Qt: 5.15.13

Python version: 3.12.4

Versión de GDAL: 3.9.0

Versión de GEOS: 3.12.2-CAPI-1.18.2

Versión de PROJ: Rel. 9.4.0, March 1st, 2024

PDAL version: 2.6.3 (git-version: b5523a)

El algoritmo comenzó en: 2024-10-29T09:02:41

**Algoritmo 'Importar ráster' comenzando…**

Parámetros de entrada:

{ 'DATABASE' : 'db\_repo\_sig', 'HOST' : '200.58.127.135', 'NAMECOL' : False, 'OVERVIEW' : '2,4,8', 'PORT' : '5433', 'RASTER' : 'C:/Users/Usuario/AppData/Roaming/QGIS/QGIS3/profiles/default/processing/outputs/prueba\_upload\_raster/loma\_compress\_10percent.tif', 'SCHEMA' : 'rasters', 'TABLE' : 'zzz\_loma\_10perc', 'TILING' : '256x256', 'TYPE' : 0, 'USER' : 'editor', 'VERSION' : 8 }

Dominio:raster2pgsql -s 5347 -t 256x256 -d -l 2,4,8 -I -C -M "C:/Users/Usuario/AppData/Roaming/QGIS/QGIS3/profiles/default/processing/outputs/prueba\_upload\_raster/loma\_compress\_10percent.tif" rasters.zzz\_loma\_10perc | psql -U editor -d db\_repo\_sig -h 200.58.127.135 -p 5433

Importando ráster a la base de datos...

¡La operación se realizó con éxito!

Leandro Franca - Ingeniero Cartográfico

Execution completed in 9100.05 segundos (2 horas 31 minutos 40 segundos)

Resultados:

{}

Cargando las capas resultantes

Algoritmo 'Importar ráster' finalizado

30/10/2024

**Objetivo**:

Aporte de Cynthia. Grazie. Entender el OL3ditor de ATER.

**Proceso**:

1-bajar el repo de github walkermatt/ol3-popup y programmerg/ol3editor

2- notar que tienen mucho de openlayers que no se

3- aprender openlayers : <https://openlayers.org/doc/quickstart.html>

3.5- OL usa Vite… aprender Vite rapidin : <https://www.youtube.com/watch?v=KCrXgy8qtjM&pp>

4- seguir el tutorial hasta llegar a “npm run build” y ejecutar ese comando

5- cambiar ubicación de /assets en “index.html” DENTRO DE /dist: 

6- subir a produccion

**Resultado**:

Captado el proceso de como levantar el mapa tutorial y subirlo a produccion

**Observaciones**:

Todo

31/10/2024

**Objetivo**:

**Proceso**:

**Resultado**:

**Observaciones**: