

# Création de tuiles à partir de la base de données OSM

## Matériel

- Processeur : Intel(R) Core(TM) i7-3630QM CPU @ 2.40GHz (8 cœurs)
- Mémoire : 16Go Ram
- OS : OpenSuse 42.1
- Postgresql : 9.4
- Postgis : 2.1

## Création des dossiers pour les sources

- cd /home/jpm/Stockage/osm/
- mkdir src downloads tiles

## Installation de Postgresql 9.4 & Postgis 2.1 + Configuration

- sudo zypper in postgresql94 postgresql94-postgis postgresql94-contrib
- systemctl start postgresql
- systemctl status postgresql
- su - postgres
- createuser osm
- createdb -E UTF8 -O osm osm
- createlang plpgsql osm
- psql osm
  - ALTER USER postgres PASSWORD '<password>';
  - ALTER USER osm WITH PASSWORD '<password>';
  - CREATE EXTENSION postgis;
  - CREATE EXTENSION hstore;
  - ALTER TABLE geometry\_columns OWNER TO osm;
  - ALTER TABLE spatial\_ref\_sys OWNER TO osm;
  - \q
- En root:
  - vi /var/lib/pgsql/data/pg\_hba.conf
  - Modifier la ligne local all all peer par local all all md5

## Installation de osm2pgsql

- Info ligne de commande osm2pgsql :  
<http://www.volkerschatz.com/net/osm/osm2pgsql-usage.html>
- Ajouter le dépôt de paquets Geo d'OpenSuse
- zypper in make cmake gcc-c++ boost-devel libboost\_system1\_54\_0 \

```
libboost_filesystem1_54_0 libexpat-devel zlib-devel \ libbz2-devel
libpqxx-devel libproj-devel lua \ lua-devel geos-devel postgresql-devel
• cd /home/jpm/Stockage/osm/src
• git clone https://github.com/openstreetmap/osm2pgsql.git
• cd osm2pgsql
• mkdir build && cd build
• cmake ..
• make
• sudo make install
```

## Télécharger les données OSM et les charger dans Postgresql

- cd /home/jpm/Stockage/osm/downloads
- wget <http://download.geofabrik.de/europe/france/languedoc-roussillon-latest.osm.pbf>
- wget <http://download.geofabrik.de/europe/france/midi-pyrenees-latest.osm.pbf>
- Télécharger le fichier de style pour osmfr-cartocss :
  - wget <http://osm13.openstreetmap.fr/~cquest/default.style> -O pg-osmfr-default.style
- Dans les lignes de commande ci-dessous utiliser l'un des 2 styles :
  - ./src/osm2pgsql/default.style : pour le projet openstreetmap-cartocss
  - ./downloads/pg-osmfr-default.style : pour le projet osmfr-cartocss
- L'ajout de données via osm2pgsql étant trop lent, on peut à la place fusionner les fichiers pbf et faire un import initial :
  - osmconvert languedoc-roussillon-latest.osm.pbf --out-05m | osmconvert - midi-pyrenees-latest.osm.pbf -o=occitanie.pbf
- Importer les données initiales :
  - Si assez de mémoire : osm2pgsql --create --database osm --username osm --password --prefix osm -E 3857 --style ./src/osm2pgsql/default.style --hstore --unlogged occitanie.pbf
  - Si pas assez de mémoire vive utiliser le cache : osm2pgsql --create --database osm --username osm --password --prefix osm -E 3857 --slim --style ./src/osm2pgsql/default.style --hstore --number-processes 6 --cache 2048 occitanie.pbf
- Pour ajouter de nouvelles données, il faut obligatoirement utiliser la méthode "slim" :
  - osm2pgsql -l --append --database osm --username osm --password --prefix osm -E 3857 --slim --style ./src/osm2pgsql/default.style --hstore --number-processes 6 --cache 2048 midi-pyrenees-latest.osm.pbf

## Ajouter des index à la base osm

- psql osm
- Puis exécuter les requêtes suivantes :

```
CREATE INDEX osm_point_tags ON osm_point USING GIN(tags);
CREATE INDEX osm_line_tags ON osm_line USING GIN(tags);
```

```
CREATE INDEX osm_polygon_tags ON osm_polygon USING GIN(tags);
```

## Table "params" manquante

```
CREATE TABLE params ( num INTEGER, KEY text);
INSERT INTO params (num, KEY) VALUES (0, 'buffer');
INSERT INTO params (num, KEY) VALUES (0, 'x_bleed');
INSERT INTO params (num, KEY) VALUES (0, 'y_bleed');
```

## Solution basée sur CartoCSS & Mapnik

Fonctionne ! Voir : [https://medspx.fr/kb/GIS/generate\\_osm/](https://medspx.fr/kb/GIS/generate_osm/)

- sudo zypper in gdal mapnik python-shapely
- cd ~/src
- Styles pour le rendu Mapnik :
  - style officiel *openstreetmap-carto* (<http://www.openstreetmap.org/>) : git clone <https://github.com/gravitystorm/openstreetmap-carto>
  - style alternatif *osm-fr* (<http://tile.openstreetmap.fr/>) : git clone <https://github.com/cquest/osmfr-cartocss.git>
- cd openstreetmap-carto
  - wget [http://svn.openstreetmap.org/applications/rendering/mapnik/generate\\_tiles\\_multiprocess.py](http://svn.openstreetmap.org/applications/rendering/mapnik/generate_tiles_multiprocess.py)
  - chmod -x generate\_tiles\_multiprocess.py
  - ./get-shapefiles.sh
  - Modification du fichier *project.mml* pour le style :
    - *openstreetmap-carto* :
      - sed -i '"/dbname": "gis"/c \\t"dbname": "osm"' project.mml
      - sed -i '"/dbname": "osm"/ i \\t"user": "osm", \n\t"host": "localhost",' project.mml
    - *osm-fr* :
      - sed -i 's/"user": "fr"/"user": "osm"/g' project.mml
      - sed -i 's/"password": "m4d31nfr4nc3"/"password": "osm"/g' project.mml
      - sed -i 's/"srs-name": "900913"/"srs-name": "3857"/g' project.mml
      - sed -i 's/"srid": "900913"/"srid": "3857"/g' project.mml
      - sed -i 's/900913/3857/g' project.mml
      - sed -i 's/planet\_osm/\_osm\_/\_g' project.mml
      - sed -i 's/\\data\\work\\mapnik\\data\\data/g' project.mml
    - Layers osmfr, 2 solutions :
      - Supprimer l'objet faisant référence au *layers/z7.tif* dans *project.mml* ligne 77 à 95.
      - Télécharger le fichier correspondant :
        - mkdir layers
        - wget <http://osm13.openstreetmap.fr/~cquest/z7.tif>

-0 layers/z7.tif

- Installer *carto*, le parseur de CartoCSS :
  - Instaler *nvm* : wget -q0-
   
<https://raw.githubusercontent.com/creationix/nvm/v0.31.2/install.sh>
  
| bash
  - Utiliser node v4.4.x : nvm use 4.4
  - Installer *carto* : npm install carto
- Transformer le format *.mml* en *.xml* pour Mapnik : ./node\_modules/.bin/carto project.mml > mapnik.xml
- Installer les Polices :
  - sudo zypper in free-ttf-fonts gnu-unifont-bitmap-fonts gnu-free-fonts dejavu-fonts google-droid-fonts sil-padauk-fonts khmeros-fonts indic-fonts thai-fonts tibetan-machine-uni-fonts noto-sans-devanagari noto-sans-bengali noto-sans-tamil
  - S'il reste des polices non supportée, supprimer les lignes qui pausent problème dans le fichier *mapnik.xml*
- Aller sur <http://www.openstreetmap.org/> :
  - Afficher dans le navigateur la zone correspondant à vos données (visible à l'écran) puis cliquer sur "Exporter", les coordonnées max de la zone (bbox) vont alors s'afficher.
    - Pour plus de facilité, copier le lien *Api Overpass*, qui contient la bbox **(-0.027,42.155,8.597,45.325)** :
    - <http://overpass-api.de/api/map?bbox=-0.027,42.155,8.597,45.325>
    - Extraire dans l'url le zoom minimum (**8**) :
    - <http://www.openstreetmap.org/export#map=8/43.761/4.285>
  - Zoomer au maximum voulu puis extraire dans l'URL le zoom max (**17**) :
   
<http://www.openstreetmap.org/export#map=17/44.34644/3.95358>
- Éditer le fichier *generate\_tiles\_multiprocess.py* : vi generate\_tiles\_multiprocess.py
  - Remplacer les dernières lignes par le contenu ci-dessous où l'on aura remplacer les valeurs *mapfile*, *tile\_dir*, *minZoom*, *maxZoom* et *bbox* par celle relevées sur Openstreetmap.
  - Donner les droits d'exécution au script : chmod +x generate\_tiles\_multiprocess.py
- Lancer le script : ./generate\_tiles\_multiprocess.py&
  - Laisser tourner le script pour générer l'ensemble des tuiles. Cela peut prendre vraiment très longtemps, suivant l'étendue des de la zone à couvrir et le zoom maximum demandé.

### Dernière ligne du fichier *generate\_tiles\_multiprocess.py* :

```
if __name__ == "__main__":
    home = os.environ['HOME']
    try:
        mapfile = os.environ['MAPNIK_MAP_FILE']
    except KeyError:
        mapfile = home + "/src/openstreetmap-carto/mapnik.xml"
    try:
        tile_dir = os.environ['MAPNIK_TILE_DIR']
    except KeyError:
        tile_dir = home + "/web/tiles(bitmap)/mapnik/"

    if not tile_dir.endswith('/'):
        tile_dir += '/'
```

```
tile_dir = tile_dir + '/'  
  
#-----  
--  
#  
# Change the following for different bounding boxes and zoom levels  
minZoom = 8  
maxZoom = 17  
bbox = (-0.027, 42.155, 8.597, 45.325)  
render_tiles(bbox, mapfile, tile_dir, minZoom, maxZoom)
```

## Gpsd, Nodejs et Leaflet

- NodeJs module faisant le pont avec Gpsd : <https://github.com/pdeschen/bancroft>

From:  
<https://memos.clapas.org/> - **Memos**



Permanent link:  
<https://memos.clapas.org/osm/creation-de-tuiles-raster>

Last update: **2020/02/09 20:48**