1. OpenDEM: SRTM DTM Processing

This Howto explains how to compute an approximately Digital Terrain Model from the SRTM dataset and OSM data land-use/-cover classes for correction. The test dataset covers the whole area of Europe.

There are many ways to achieve this goal. I would prefer to do this all with PostGIS 2.0, but I was not successful. If somebody could explain me how to do this I would be very pleased.

Software:

- **GRASS (http://grass.osgeo.org)**
- **QGIS (http://www.qgis.org)**
- **GDAL (optional) (http://www.gdal.org)**

Download data

First **Download** the original SRTM dataset. Version 2.1 is the most advanced dataset:

USGS → SRTM 3: http://dds.cr.usgs.gov/srtm/version2_1/

Make a correction of the voided areas explained in the Howto „OpenDEM: SRTM DSM Processing".

Download the OSM dataset features:
Landuse: forest
Natural: wood
building
highway: pedestrian
highway: residential
highway: living_street

You could download ready to use OSM Shape Files from the Geofabrik (http://download.geofabrik.de/).

If you have a huge area like Europe it is less work to import the desired OSM Data with the program OSMOSIS. Go to the chapter "2. OpenDEM: OSM Data Processing".

Processing with QGIS

Merge the datasets with the same geometry type and correction factor:

1. forst & wood
2. pedestrian & residential &highway: living_street

SRTM DTM PROCESSING

QGIS: Vector → Geoprocessing Tools → Union

Convert the OSM shape files into a raster image:
QGIS: Raster → Conversion → Rasterize (Vector to raster)

GRASS has also a conversion tool for vector to raster processing, but I was not able to use this for huge datasets.

Processing with GRASS

Import the raster data into GRASS.

GRASS: File → Import raster data → Common import formats [r.in.gdal]

Process the data with the Raster map calculator.

GRASS: Raster → Raster map calculator

Example for forest layer:

if ((isnull(poly_raster@PERMANENT)), srtm_koeln_all_cor@PERMANENT, (srtm_koeln_all_cor@PERMANENT - 7))

srtm_koeln_all_cor@PERMANENT = original SRTM data
poly_raster@PERMANENT = forest OSM data

If no "NULL" Values are available you could compute them with GRASS: Raster → Develop Raster Map → Manage NULL values (r.null). Or simply use the value 0, which is OK in our case:

```
if(streets_81_69@PERMANENT ==0, europe_81_69@PERMANENT,(europe_81_69@PERMANENT - 7))
```

Do the raster calculation for every OSM layer and you are finished.

The raster calculator works only with Tif files < 2GB. You have to split the Tifs before processing when they are bigger, e.g. with gdal:

gdal_translate -a_srs EPSG:4326 -a_ullr -11.0 81.0 41.0 34 original.tif subseted.tif

Merge the images when you are finished, e.g. with gdal_merge.py.

Of course you could also use GRASS for this concern:

Subsetting: Raster → Develop raster map → Region boundaries
Mosaicing: Raster → Overlay rasters → Patch raster maps

2. OpenDEM: OSM Data Processing

- **OSMOSIS (http://wiki.openstreetmap.org/wiki/Osmosis)**
- **Postgres & PostGIS (http://www.postgresql.org)**
- **7Zip (http://www.7-zip.org) for WINDOWS or BUNZIP2 for LINUX**
- **OSM2PGSQL (http://wiki.openstreetmap.org/wiki/Osm2pgsql)**

Download the latest OSM Planet file: http://planet.osm.org/

To avoid heap Space Errors tune your OSMOSIS file:
http://wiki.openstreetmap.org/wiki/Osmosis/Tuning

Extract the desired data from the OSM planet file, e.g. for the forested areas (example for Windows7):

"YourPath\7-Zip\7zG.exe" e -so YourPath\planet.osm.bz2 | YourPath\osmosis.bat --rx file="-" --way-key-value keyValueList="landuse.forest,natural.wood" --bounding-box top=81.0 left=-11.0 bottom=34.0 right=41.0 --used-node --write-xml file="YourPath\planet_forest.osm"

Be careful with copy & paste because of character encoding. For LINUX you have to use BUNZIP2 for the data streaming.

Do the same for the buildings and the roads.

building.yes

highway.pedestrian
highway.residential
highway.living_street

Load the data in a PostGIS database via OSM2PGSQL (e.g.):

osm2pgsql.exe -c -s -l -d gis -U postgres -W -H localhost -P 5432 -S D:\opentopomap\osm2pgsql\default.style E:\planet_building.osm

Do the same for the buildings and the roads.

To correct unclosed polygons and other errors of the geometries use the PostGIS function ST_BUFFER with 0 as parameter (e.g.):

CREATE TABLE buildings_cor AS SELECT st_buffer(way, 0) as the_geom, osm_id from planet_osm_polygon;

Create a shape from the table via PGSQL2SHP (e.g.):

pgsql2shp.exe -k -u postgres -p 5432 gis buildings_cor -f building_cor.shp

PGSQL2SHP is available in your Postgres bin folder.

Now you have your Shapefile. Shapefiles have a limit of 2 GB. If your Shapefile is to big subset the dataset (e.g.):

pgsql2shp.exe -k -u postgres -p 5432 gis "SELECT (st_intersection(the_geom, GeometryFromText('POLYGON((-11.0 53.5, 41.0 53.5, 41.0 34.0, -11.0 34.0, -11.0 53.5))', 4326))),osm_id FROM buildings_cor WHERE ST_INTERSECTS (the_geom,GeometryFromText('POLYGON((-11.0 53.5, 41.0 53.5, 41.0 34.0, -11.0 34.0, -11.0 53.5))', 4326))" -f building_sub.shp