QGIS stylesheets and code to render OSM data as a scalable map

Copyright 2017 Industrial Data & Analytics Ltd. Distributed under the GPLv3 license. See the statement at the end of this document for more information. For support, please use GitHub tickets / bug tracker. For other contact, email info@dataandanalytics.co.uk.

The goal of this project is to produce beautiful, modern map styles or OpenStreetMap (OSM) data in QGIS. I will enable anyone to easily create professional quality maps without needing to look into tile servers and other technicalities. Simply load the OSM data into QGIS, apply the stylesheets and let it render.

If you improve on these maps, please upload your changes for the project to commit. Full attribution to you will be given, subject to the GPLv3 license under which this is distributed.

OpenStreetMap (openstreetmap.org) is a project to free the information about our world and ensure it belongs to all, and not a small group (eg Google, Microsoft etc who may ‘be evil’ at some point in the future).

The steps to your beautiful, scalable map in QGIS are below.

## QUICK START GUIDE

To get started quickly these are the steps:

Note: recommended to untick “Render” to speed creation.

1) load your OSM data into QGIS as Postgres vector layers.

One layer should be created for each table of planet\_osm\_roads, planet\_osm\_point, planet\_osm\_line, planet\_osm\_polygon (in rendering order last→first ie, in the layers panel, roads should be on top of point, on top of line etc).

2) Apply stylesheets. There is one stylesheet per layer with the same name eg planet\_osm\_roads.qml should be applied to planet\_osm\_roads

3) Set the ocean back ground. Project→Project Properties→Background color (click the color bar). Set to colour #b2ddf2.

4) Apply national parks. Database→DB Manager→PostGIS→(select db name eg osm)→(select login role)→SQL Window

Enter:

with "dumped\_polygonised" as

(

SELECT

"polygonised"."name" as "name",

"polygonised"."boundary" as "boundary",

(ST\_Dump("polygonised"."polycoll"))."geom" As "polygon\_geom"

FROM (

SELECT

string\_agg(distinct "name",' ') as "name",

"boundary",

ST\_Polygonize("way") as "polycoll"

FROM "planet\_osm\_line"

WHERE "boundary" = 'national\_park' and "highway" IS NULL

group by "osm\_id","boundary"

) as "polygonised"

)

SELECT row\_number() over(order by "name") as "id",

"name",

"boundary",

ST\_Multi(ST\_Collect("polygon\_geom")) as "polygon\_geom\_multi"

FROM "dumped\_polygonised"

GROUP BY "name", "boundary"

Check “Load as new layer”. Check “Column(s) with unique id” and select id. For the Geometry column select polygon\_geom\_multi. For layer name add osm\_polygons\_from\_lines. Click “Load Now!”.

Your map will now have national parks.

5) Country outlines.

WG84, faster render / smaller scales, download from here: <http://data.openstreetmapdata.com/land-polygons-complete-4326.zip>

WG84, Slower render / more detailed / larger scales, download from here:

<http://data.openstreetmapdata.com/land-polygons-complete-4326.zip>

Others, eg Mercator, can be downloaded from here. Just select the one you want:

<http://openstreetmapdata.com/data/land-polygons>

Import as shapefile into QGIS and apply land.qml stylesheet.

You’re done. Tick “Render” and you now have a beautiful, scalable map.

Step-by-step guide:

## GET THE OSM DATA

For this to work, you need to have the OpenStreetMap.org data in a Postgres DB.

A quick search should tell you where to find it.

If you only want a section of the world, [http://download.geofabrik.de](http://download.geofabrik.de/) is a good option. Download the pbf file desired.

In linux, I used this to get European data:

nohup wget -O europe-latest.osm.pbf "http://download.geofabrik.de/europe-latest.osm.pbf" &

**Load data into Postgres**

I used osm2pgsql which is hosted here <https://github.com/openstreetmap/osm2pgsql>

If you only need a subset of the data, then use a bounding box to save time. This will load data for the UK only (replace DATABASE with the name of the db you have created and USERNAME with the name of the login role. It will ask you for the password as -W is set):

osm2pgsql --bbox -14.106,48.720,2.505,62.063 --number-processes=4 --latlong --keep-coastlines --hstore --slim -C 30000 -d DATABASE -U USERNAME -W -S /usr/share/osm2pgsql/default.style europe-latest.osm.pbf

Otherwise, this will load whatever data from OSM you have in the osm.pbf file:

osm2pgsql --latlong --keep-coastlines --hstore --slim -C 30000 -d DATABASE -U USERNAME -W -S /usr/share/osm2pgsql/default.style osm.pbf

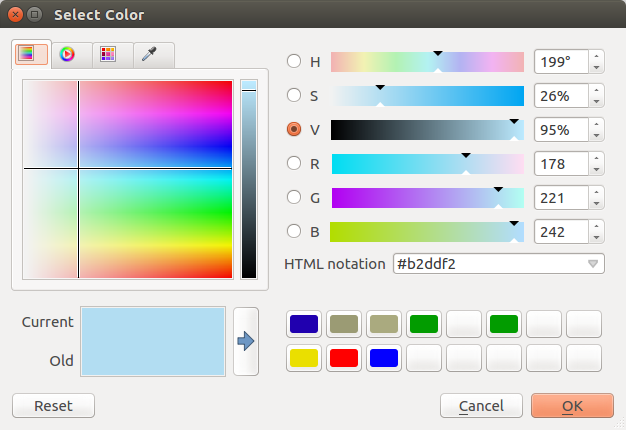
Please refer to the osm2pgsql documentation at GitHub for further information.

## FIRST STEPS

Untick “render” on the bottom toolbar of the QGIS window.

**Set the map background to an ocean color**

Project→Project Properties→Background color (click the color bar).



In the Select Color window, write in the HTML notation #b2ddf2 (or our ocean colour).

Click Ok.

## **Add the country outlines**

So far your map only contains ocean. Beautiful but not so useful to a land dwelling race. Adding the countries from the osm data is surprisingly complicated requiring intersections of administrative boundaries and coastlines.

Of course, being the open source philanthropist project it is, the problem has been solved for you for free (I donate £2 per year to every project I use. I encourage you to do the same. It costs me about £50 per year but because I make sure it doesn’t all happen at once, I don’t notice it).

Download the shapefile you want from here <http://openstreetmapdata.com/data/land-polygons>

If you are not sure which one you want, use this one: <http://data.openstreetmapdata.com/land-polygons-complete-4326.zip>

And if it looks bad, you’re probably zooming in too much so use this: <http://data.openstreetmapdata.com/land-polygons-complete-4326.zip>

Unzip the .shp file into your project folder and delete the zip.

Import as shapefile into QGIS and apply *land.qml* stylesheet. Double click on the layer→Style→Load Style→”Load from file...” and select the land.qml from the folder you downloaded the stylesheets to.

You now have countries.

## **Load the osm layers**

Create a new postgres vector layer by clicking the  button on QGIS.

Add the connection to your db. Select each of the main tables:

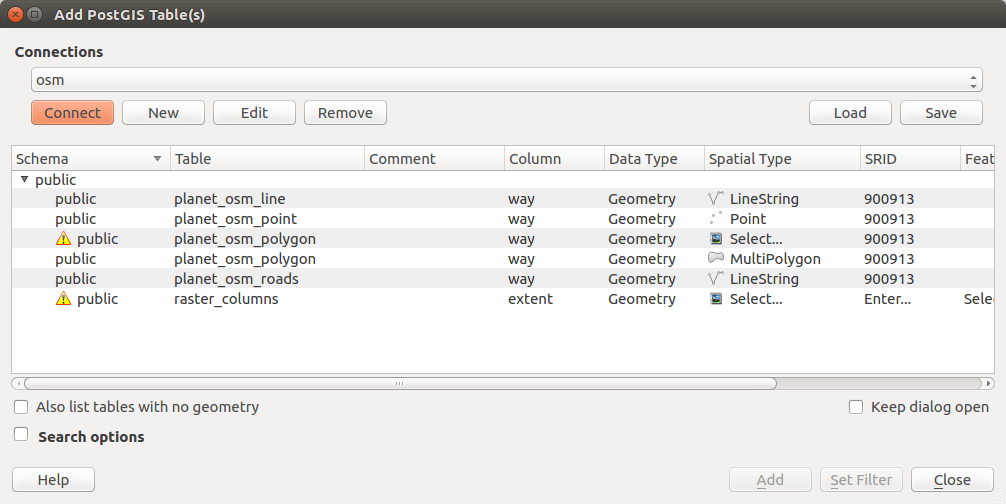
planet\_osm\_roads

planet\_osm\_point

planet\_osm\_line

planet\_osm\_polygon

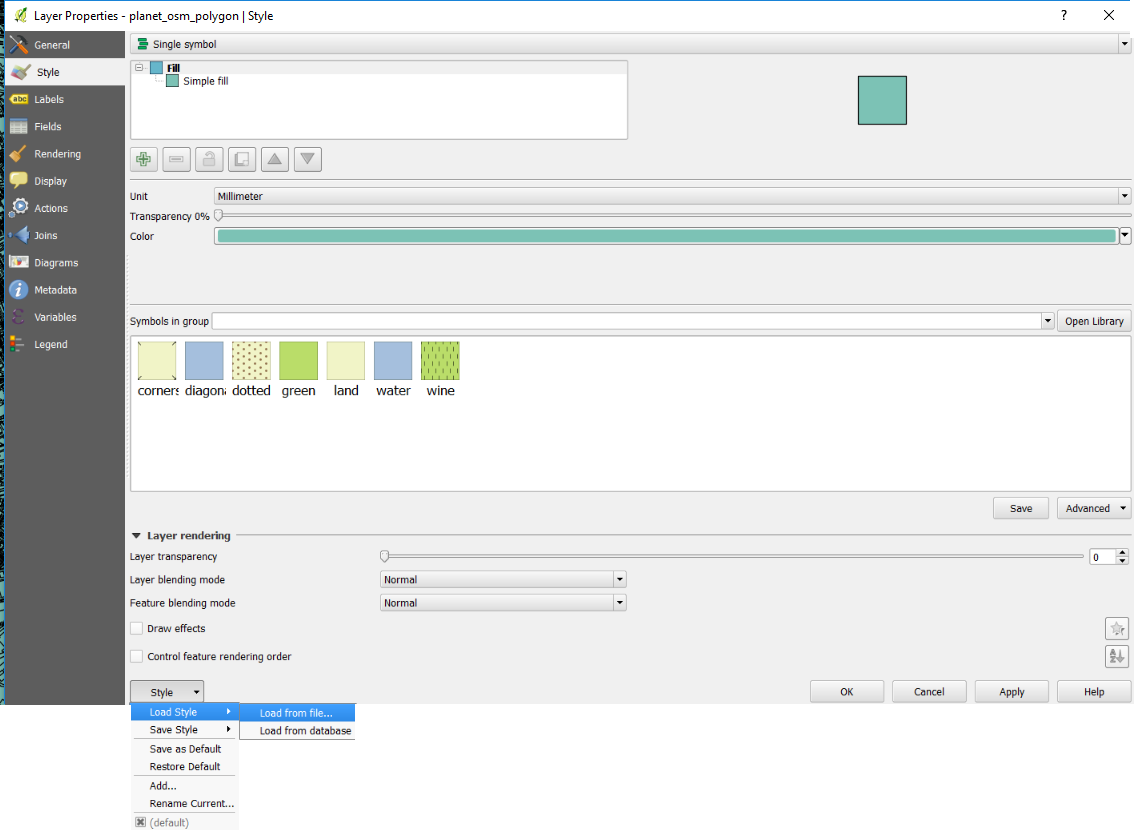
(Note in QGIS they should be listed in the order above).



Click the “Add” button.

## Load Style sheets

Right click->Layer Properties (or double click on the layer). Click the “Style” dropdown. Follow “Load file” and select “Load from file...”



Go to your stylesheets folder and chose the style sheet that has the same name as the layer.

Select “Open”

## national parks

Natoinal parks will be missing from your maps. To apply them, select from the QGIS menu Database→DB Manager→PostGIS→(select db name eg osm)→(select login role)→SQL Window

And enter:

with "dumped\_polygonised" as

(

SELECT

"polygonised"."name" as "name",

"polygonised"."boundary" as "boundary",

(ST\_Dump("polygonised"."polycoll"))."geom" As "polygon\_geom"

FROM (

SELECT

string\_agg(distinct "name",' ') as "name",

"boundary",

ST\_Polygonize("way") as "polycoll"

FROM "planet\_osm\_line"

WHERE "boundary" = 'national\_park' and "highway" IS NULL

group by "osm\_id","boundary"

) as "polygonised"

)

SELECT row\_number() over(order by "name") as "id",

"name",

"boundary",

ST\_Multi(ST\_Collect("polygon\_geom")) as "polygon\_geom\_multi"

FROM "dumped\_polygonised"

GROUP BY "name", "boundary"

Check “Load as new layer”.

Check “Column(s) with unique id” tickbox and select “id”.

For the Geometry column select “polygon\_geom\_multi”.

For layer name add “osm\_polygons\_from\_lines” or choose your layer name.

Click “Load Now!”

Your map will now have national parks.

And you’re done for a beautiful, scalable map in QGIS using OSM data.

## KNOWN HACKS & ISSUES

The planet\_osm\_roads stylesheet contains a hack to limit the frequency of roadshields displayed. By default, so many roadshields are displayed at high scales that it can be hard to see the roads. The stylesheet only displays the roadshield if the feature’s osm\_id is divisible by 40 with no remainder (using modulo ie if osm\_id % 40 = 0). This is completely arbitrary and a hack. Feel free to improve it if you know how.

## Design notes

### PLANET\_OSM\_point.qml:

This style sheet is designed for OSM planet\_osm\_point data, it shows/labels the place name, bus stop and symbol of amenities.

Figure: Legend of amenity



#### Place

It is shows some important places name:  
- locality. A named place that has no population.

- hamlet. A smaller rural community, typically with fewer than 100-200 inhabitants, and few infrastructure.

- village. A smaller distinct settlement, smaller than a town with few facilities available with people traveling to nearby towns to access these.

- farm. An individually named farm.

- isolated dwelling. The smallest kind of settlement (1-2 households).

- suburb. A part of a town or city with a well-known name and often a distinct identity.

- town. An important urban centre, between a village and a city in size.

- neighbourhood. A neighbourhood is a smaller named, geographically localised place within a suburb of a larger city or within a town or village.

- island. Any piece of land that is completely surrounded by water and isolated from other significant landmasses.

- city. The largest urban settlement or settlements within the territory.

- country.

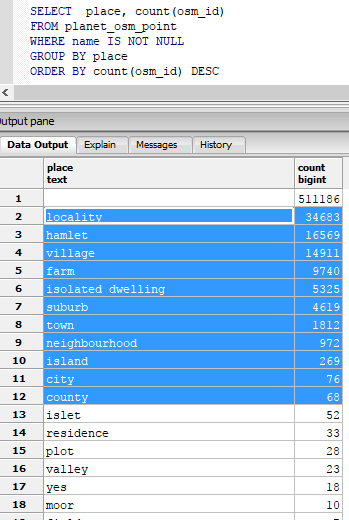
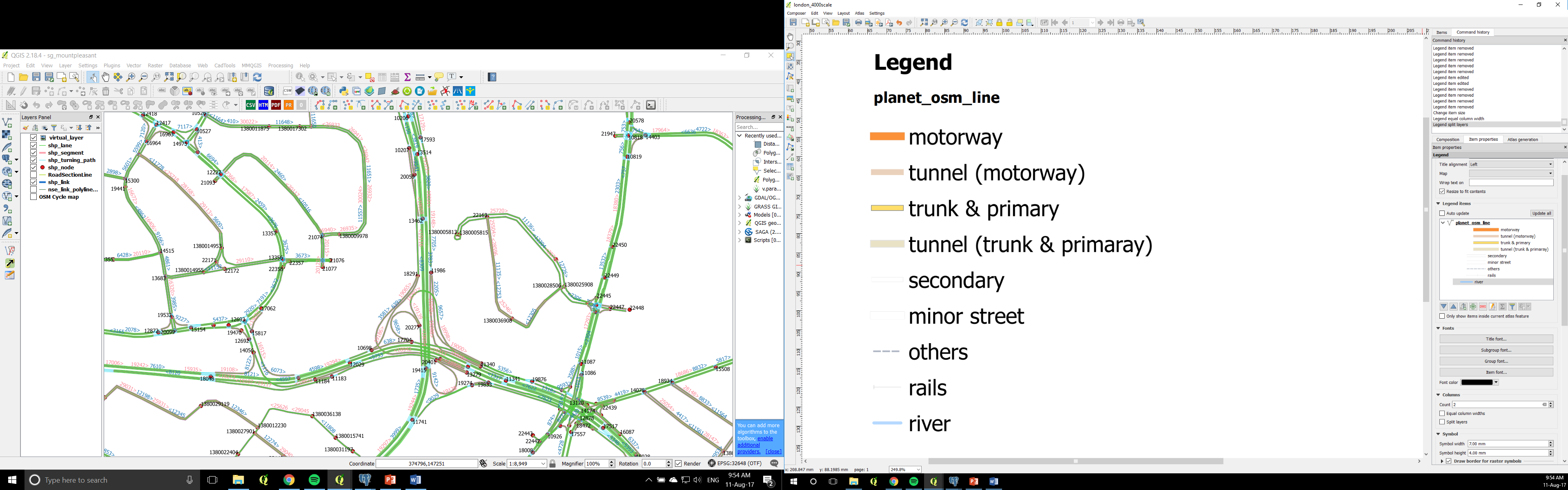


Figure: SQL Query: Get the number of objects (place) from planet\_osm\_point

### planet\_osm\_line.qml:

This style sheet is designed for OSM planet\_osm\_line data and present the osm lines in different styles (width, color) based on their functional class, e.g. highway, trunk, secondary street, footway, river, etc. \*\* This style sheet can be used by planet\_osm\_roads which only included major roads.

Figure: Legend of roads, river and others.



### planet\_osm\_polygon.qml: used for planet\_osm\_polygon

This style sheet is designed for OSM planet\_osm\_polygon. It classifies the osm polygon in several styles based on land use (park, industrial, water, commercial, retail, etc.) and building types (school, hospital, residential, etc.)

Figure: Legend of land use

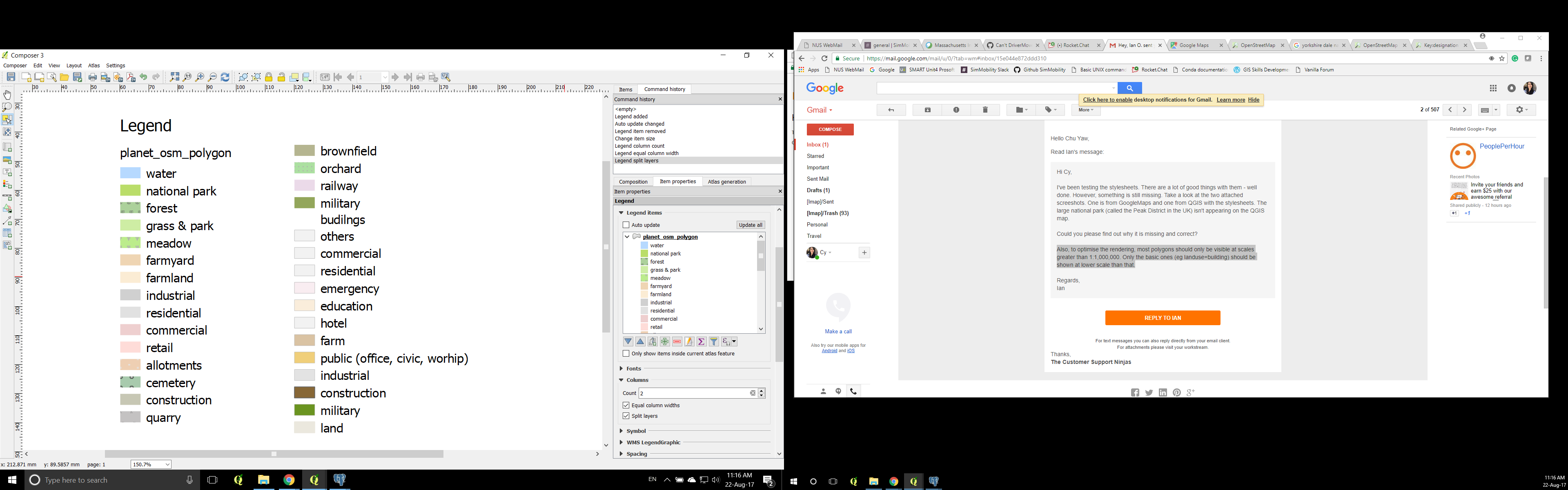
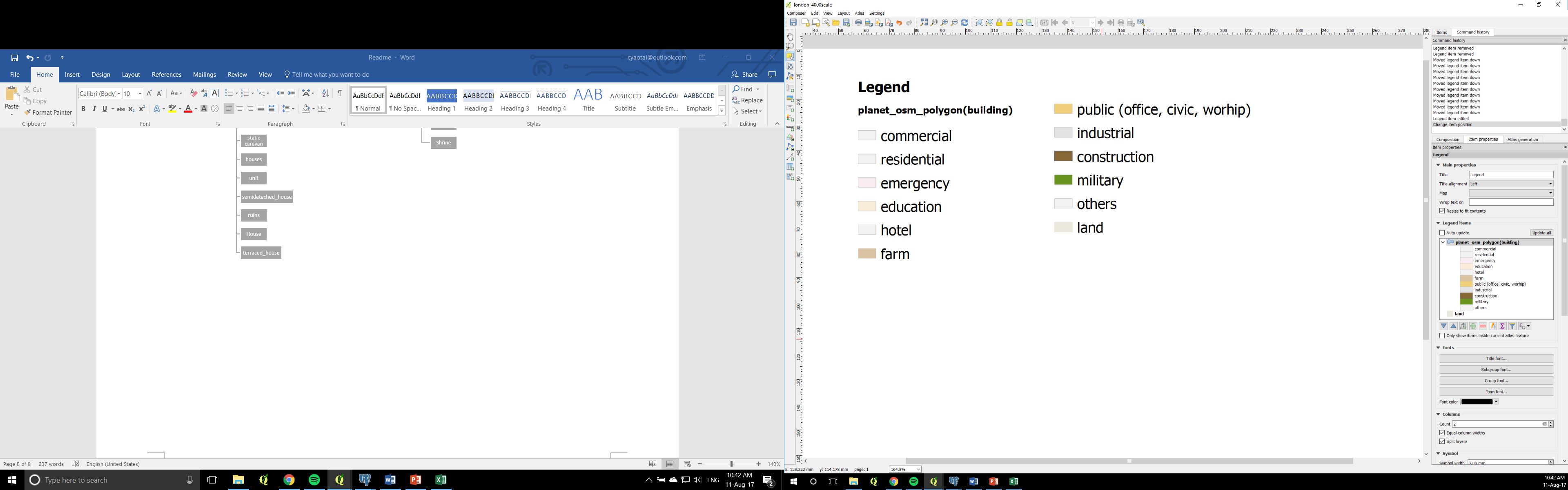


Figure: Legend of buildings



## Copyright and license

Copyright 2017 Industrial Data & Analytics Ltd. Released under the GPLv3 license.

This file is part of "QGIS stylesheets and code to render OSM data as a scalable map".

"QGIS stylesheets and code to render OSM data as a scalable map" is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

"QGIS stylesheets and code to render OSM data as a scalable map" is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with "QGIS stylesheets and code to render OSM data as a scalable map" in a file called COPYING.txt and/or COPYING.license-gpl-3.0.odt. If not, see <<http://www.gnu.org/licenses/>>.

For support, please use GitHub tickets / bug tracker. For other contact, email info@dataandanalytics.co.uk.