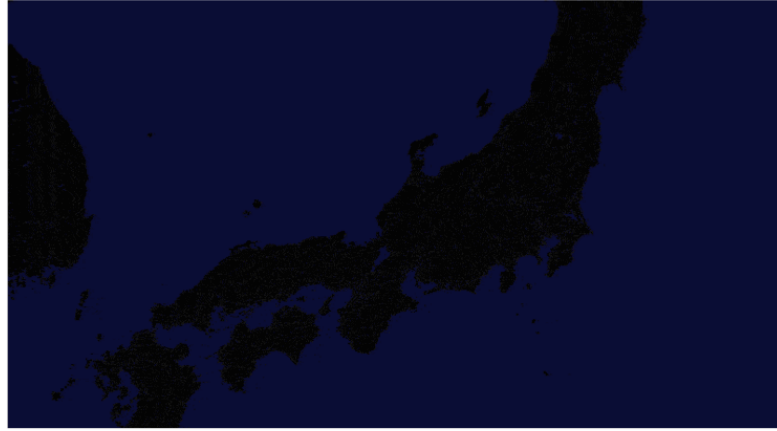


Easier OSM exports

Some simple tools from
@eldang

We <3 OpenStreetMap



GIF by Eric Fisher for Mapbox; data © OpenStreetMap contributors

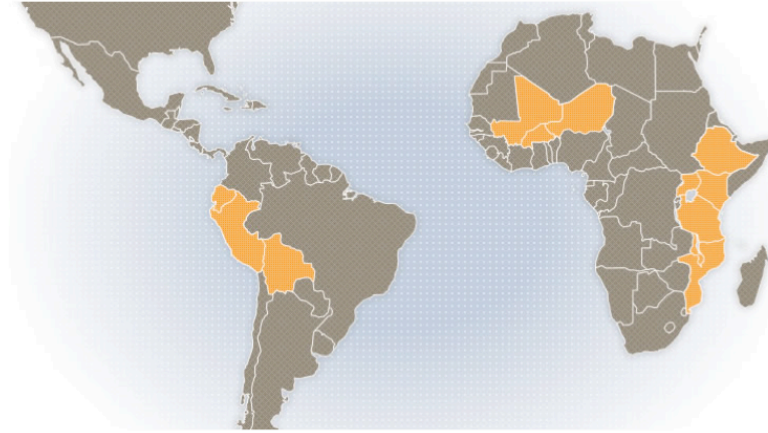
OSM is amazing. Given the audience I don't think I need to say anything more than that, but here's a little celebration.

Great tool ecosystem, but...



It can be pretty frustrating as a beginner. Like so much open source software, there's a certain amount that you just have to know to be able to get things done with it. I've made some little tools to help with that. I'll first give you some background on why I started, then tell you about the tools, and then ask for your help making them better.

Background: CCRP



Map from ccrp.org

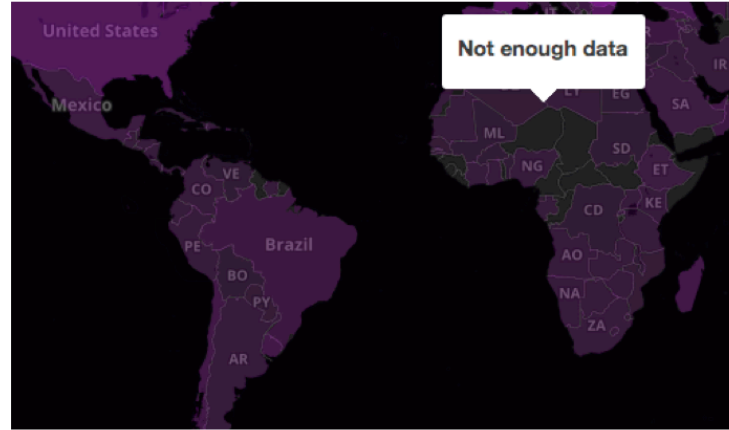
Karsten & Matt, who many of you know, are providing GIS support to the Collaborative Crop Research Program, which in turn supports on-the-ground research on locally suitable crop varieties and production techniques. Their maps can help people see similarities and differences between study sites, and understand why some things will generalise and others won't. However: The study areas tend to be those with the worst internet access in the world.

Karsten: <http://www.terragis.net/>

Matt: <http://coregis.net/>

<http://www.ccrp.org/>

Background: CCRP



Map from netindex.com

In fact, Niger and Burkina Faso have so few active internet users per month that netindex.com can't even calculate a speed estimate. So we need to be able to package data for offline use. And we can't just tell people to download the planet file because it's just too big. So they asked me to help build a pipeline to package up OSM data for relevant geographic areas. Here's what I made, with an emphasis on simplicity over flexibility – if you need power user features it's probably easier to learn how to use osm2pgsql directly than adapt these scripts.

Ingredients

GEOFABRIK  *downloads*

+

osm2pgsql

+

Python

+



Of course, I didn't have to start from scratch. Resources used:

<http://download.geofabrik.de/> for continent-level data

<https://github.com/openstreetmap/osm2pgsql> to import those to our own DB

Python scripts to glue it all together

<http://www.gdal.org/ogr2ogr.html> to make the exports

Basic use: importing

```
./import.py antarctica
```

```
./import.py africa,south-america
```

```
./import.py north-america/canada/ontario
```

That's it!

This is the part I'm happiest with – importing couldn't be simpler! You just need to follow the region naming syntax Geofabrik uses.

If you repeat an import, the script will apply changelists since it was run, rather than re-importing the whole thing.

Basic use: exporting

```
./export.py africa country 'burkina faso'  
burkina
```

```
./export.py africa province mwanza  
mwanza-province
```

```
./export.py africa province mwanza  
tanzania-mwanza -pcfn tanzania
```

```
./batch_export.py south-america countries
```

After one database setup step that really ought to be automated, exporting all data for a given geography is also fairly simple. This produces sqlite output by default. There's also an option to export ESRI shapefiles, like so:
./export.py africa province mwanza tanzania-mwanza -pcfn tanzania -f shp

Getting fancy: data subsets

```
{
  "comment": "Use this to include all waterways: natural, artificial, permanent or intermittent, but not pipelines, shorelines or wetlands",
  "includeByPresence": [
    "waterway",
    "water"
  ],
  "includeByValue": [
    {
      "tagName": "natural",
      "values": [
        "water",
        "wadi",
        "waterfall"
      ]
    },
    {
      "tagName": "bridge",
      "values": [
        "aqueduct"
      ]
    },
    {
      "tagName": "landuse",
      "values": [
        "reservoir"
      ]
    }
  ]
}
```

There's a simple system for specifying tags to include as JSON files. I think this example exports all waterways.

What next?

Use it! github.com/eldang/osm_export_import

Please check the tag sets - I don't have much OSM experience.

Please help me package the prerequisites & geometries for less-technical users.

Please suggest improvements.



First of all, I'm telling you about this so you can use it! But I could also use some specific help to make this as simple to use as it could be.

Thank you

github.com/eldang/osm_export_import

Contact me: [@eldang](#) ; eldang@gmail.com

Credits



CORE GIS
Mapping & Analysis

