OpenStreetMap Sample Project - Data Wrangling with MongoDB

1. **Documentation of the work**
   1. [**mapparse.py**](#id.z17m47rq7k8m)
      1. [**Code**](#id.x8cdwd1fv1rp)
      2. **Result**
   2. [**tags.py**](#id.txff5f2dcbi9)
      1. [**Code**](#id.cwg2hb26ev7x)
      2. **Result**
   3. [**users.py**](#id.wy2phhdam604)
      1. [**Code**](#id.tzg7thr6ht5m)
      2. **Result**
   4. [**audit.py**](#id.7s7nby6omsr3)
      1. [**Code**](#id.11c6xdvid4tm)
      2. **Result**
   5. [**data.py**](#id.2jt5kg33wihc)
      1. [**Code**](#id.f9jne712m73w)
      2. **Result**
2. **Dataset**
   1. [**Sample of Dataset**](https://drive.google.com/file/d/0BwlRKjgcoEkIazVjelJwZ3kwZkE/view?usp=sharing)
3. [**Using MongoDB**](#id.rk28fccgr30o)
4. [**Additional data exploration using MongoDB queries**](#id.83ybh7e32xw9)
5. [**Source**](#id.2fzl1vlhij1s)

I used Debian OS on Google Cloud Platform and Windows Vista

|  |
| --- |
| mapparse.py: |

First error message:

File "mapparser.py", line 1, in <module> import xml.etree.Elements as ETImportError: No module named Elements

My first approach:

#import xml.etree.Elements as ET

Running again the code, i get

Killed

That problem was solved by using xml.dom.pulldom (1)

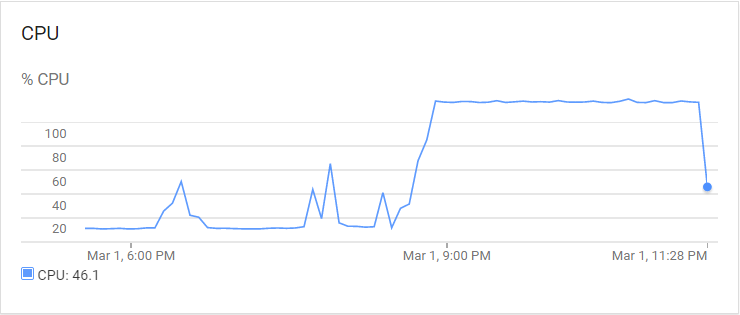
Code mapparser:

[**mapparser.py**](https://drive.google.com/file/d/0BwlRKjgcoEkIUkc4SF96RzJOX28/view?usp=sharing)

Result:

{u'bounds': 1, u'member': 236352, u'nd': 10523262, u'node': 8577720, u'osm': 1, u'relation': 19446, u'tag': 5899588, u'way': 1194243}

Time needed for the calculation



|  |
| --- |
| tags.py |

**Result:**

{'lower': 4275274, 'lower\_colon': 1592778, 'other': 31327, 'problemchars': 209}

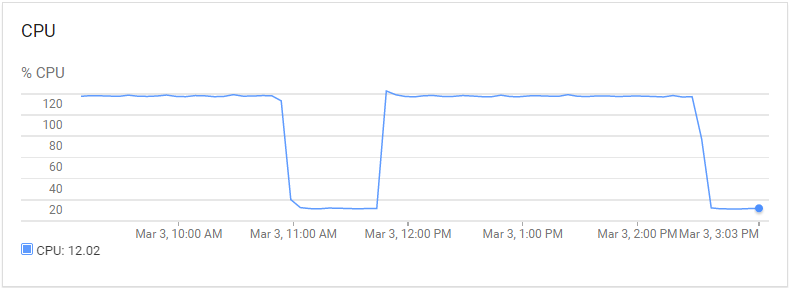
**Code tags:**

[**tags.py**](https://drive.google.com/file/d/0BwlRKjgcoEkISU9OWm5fUUR6Q2s/view?usp=sharing)

Time needed for the calculation



|  |
| --- |
| users.py |



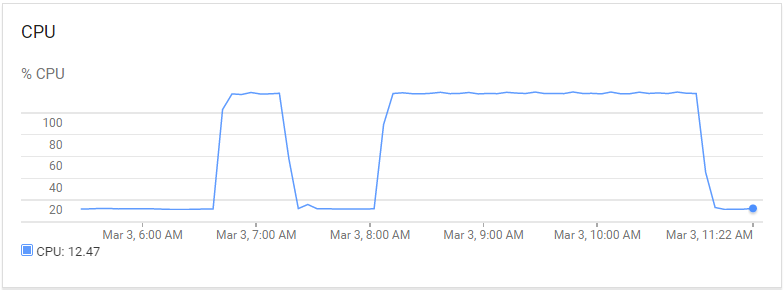
**Result:**

[**way\_vector**](https://drive.google.com/file/d/0BwlRKjgcoEkIWENDSU04dFlwckU/view?usp=sharing)

**Code users:**

[**users.py**](https://drive.google.com/file/d/0BwlRKjgcoEkIZVZmZFl3UmpSVFE/view?usp=sharing)

|  |
| --- |
| audit.py |



After running the code i noticed that the output is very big, so i decided to print the result into a text file. By open of the txt file you can see very quick that the text is not readable.

**Result in \*.txt File**

[**streetname.txt**](https://drive.google.com/file/d/0BwlRKjgcoEkIbk9lc3BqRDRDMTA/view?usp=sharing)

small summary from the result:

*u'Adolf-Sch****\xe4****rf-Stra\xdfe': set([u'Dr. Adolf-Sch\xe4rf-Stra\xdfe']),  
 u'Adolf-Sterz-Stra****\xdf****e': set([u'Adolf-Sterz-Stra\xdfe']),  
 u'Adolf-Weber-Gasse': set([u'Adolf-Weber-Gasse']),  
 u'Adolfine-Malcher-Gasse': set([u'Adolfine-Malcher-Gasse']),  
 u'Adolfstorgasse': set([u'Adolfstorgasse']),  
 u'Adyho': set([u'Adyho']),  
 u'Ad****\xe1****miho': set([u'Ad\xe1miho']),*

It looks like that some words in German like “ä”, “ü” have some special unicode character

|  |  |
| --- | --- |
| Zeichen | in Unicode |
| ä | \xe4 |
| ö | \xf6 |
| ü | \xfc |
| Ä | \xc4 |
| Ö | \xd6 |
| Ü | \xdc |
| ß | \xdf |
| € | \u20ac |

I found the function (street\_name.decode('utf8')) to return a string that solve that kind of problem. By analyzing the streetname.txt document I saw that python dictionary do not save f.e. Straße, it saves the ascii code Stra\xdfe. So i do not reach the goal yet.

Later I saw that it would be nicer to display the street names only in roman Unicode because we have german language and slovakian language this languages have some special characters that romanian languages are not using.

To do so i used the package unidecode from python, for more information see source nr.7. It would also be possible to define a dictionary f.e {ß:ss} and replace all character.

**Result after translation:**

[**streetname\_translate.txt**](https://docs.google.com/document/d/1BoyvTYcogQy6Xbuj9X22EUTQoJLB3qfW-sM8k7l9D5g/edit?usp=sharing)



After some extra fees to google they increased the computer capacity and we run the code approximately for an hour instead of three.

**Code audit.py**

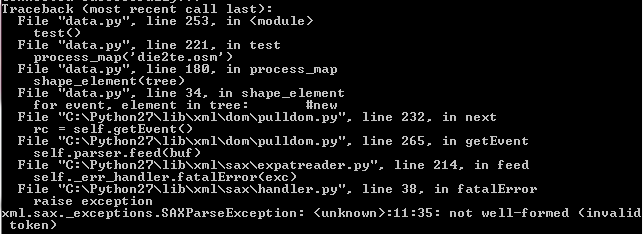
[**audit.py**](https://drive.google.com/file/d/0BwlRKjgcoEkIQk5TWHpMRUZ4MlU/view?usp=sharing)

|  |
| --- |
| data.py |

It was not easy to get an result from this function because the calculation time was more than six hours so my connection to the server broke twice.

So I decided to install everything (MongoDB and Python) onto my Windows machine. The calculation was done in approximately 3 hours. The most work on that project was the cleaning of the raw data. My first challenge, bringing der German and Slovakian Words to ‘UTF-8’ convention f.e. ä->a,ü->u,ö->o,ß->ss as described in audit.py.

That was not my last challenge because when I try to insert my json data into MongoDB I got this nice error message twice.



The reason for this I will explain in the example:

By inserting this node I got the error message

<node id="2613120756" version="1" timestamp="2014-01-07T08:08:46Z" uid="302363" user="fkv" changeset="19858523" lat="47.6562351" lon="15.8724352">

<tag k="name" v="Das Schildwirtshaus **>>**Zur Goldenen Krone**<<**"/>

<tag k="tourism" v="information"/>

<tag k="board\_type" v="history"/>

<tag k="information" v="board"/>

</node>

MongoDB has a problem by inserting

**“>>” and “<<”**

People uses special character for highlighting the name of their company.The most inconsistent spelling I found in the names.

This changes in the text was made with the function toascii.py before I saved the data into mongoDB with the function data.py

**Code toascii.py:**

[**toascii.py**](https://drive.google.com/file/d/0BwlRKjgcoEkIRk8ydWRkczI0YlU/view?usp=sharing)

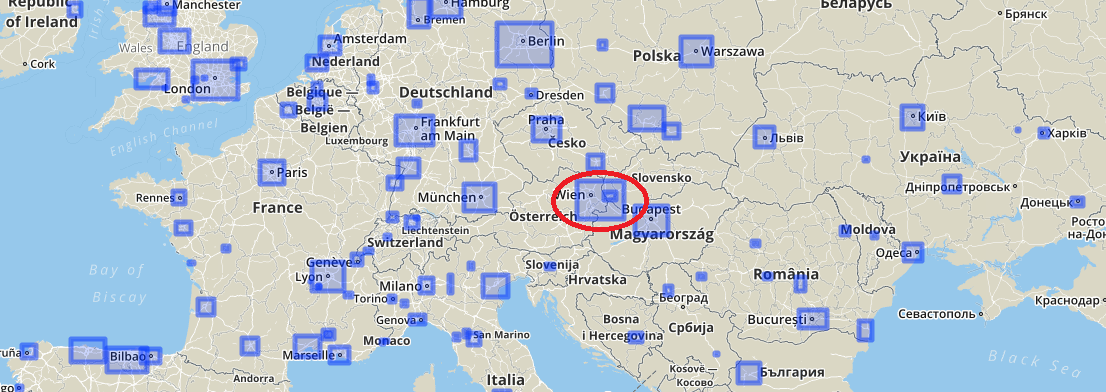
**Code data.py**

[**data.py**](https://drive.google.com/file/d/0BwlRKjgcoEkIUmxjNXRoY0g0Mk0/view?usp=sharing)

|  |
| --- |
| MongoDB - Data |



The size of the data is 139,6 MB



I was using this data because I am living and working in this area

**Number of documents:**



**Number of nodes:** (by checking my first answer mapparser.py I saw that I get by searching the nodes in the osm file 8577720). I was double checking the website openstreetmap.org and I found the answer to my problem. I download my file from that [LINK](https://mapzen.com/metro-extracts/), they make an automatically update with the latest data each week. I started my Project on Google Cloud Platform, after a few week with lots of problems I reuploaded the data onto my windows machine. So thats the reason for the mismatches.



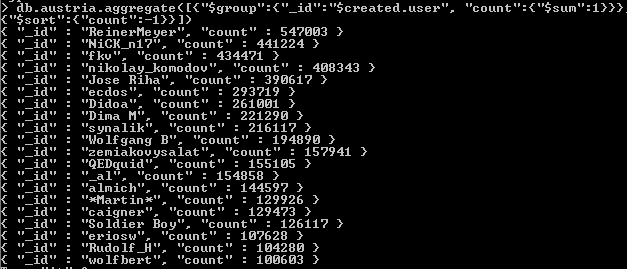
**Number of ways:**



**Number of unique users:**



**Top 20 contributing user:**

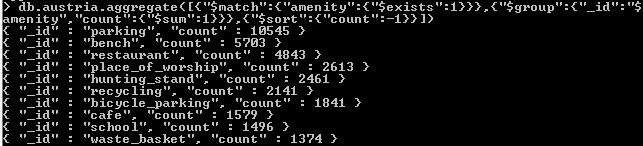


**Number of users appearing only once (having 1 post):**

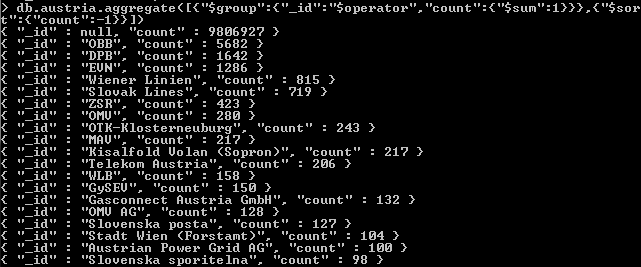


|  |
| --- |
| **Additional data exploration using MongoDB queries** |

**Top 10 appearing amenities:**

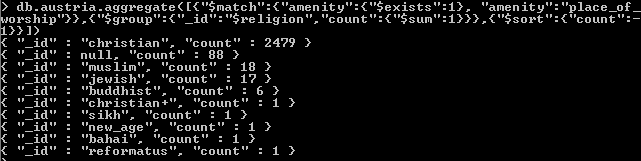


## Biggest operators:



It looks likes like that big infrastructure companies are represented most.

## Biggest religion:

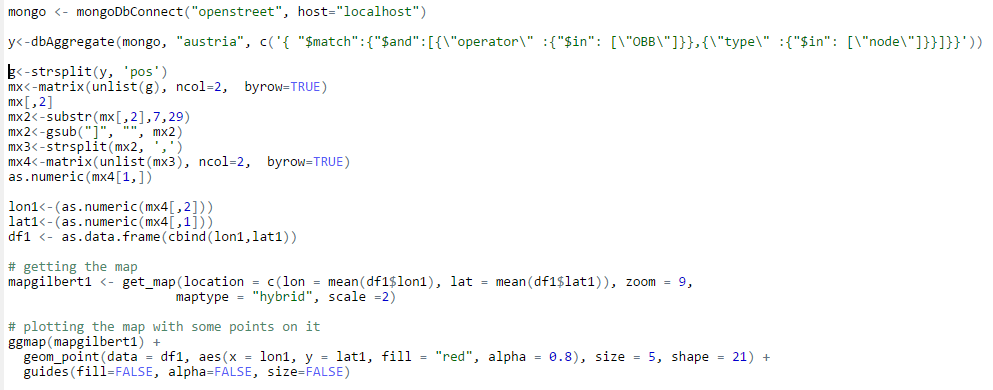


As a next milestone I was very interesting to work with the data in an effective way, not only make a query on the database.Because of the reason that I installed mongoDB onto my local machine and worked with some query makes me not satisfied to understand the data. So my next plan, bringing the data with mongoexport/mongoimport onto a VM (Virtual machine) in my Google Cloud.

In R you have two packages RMongo and rmongodb to import the data.

We have seen that OBB (the austrian railways company) is the biggest operator in the jusen dataset. To get a better view I filter all buildings (notes) and plot it into a graph from the operator OBB.

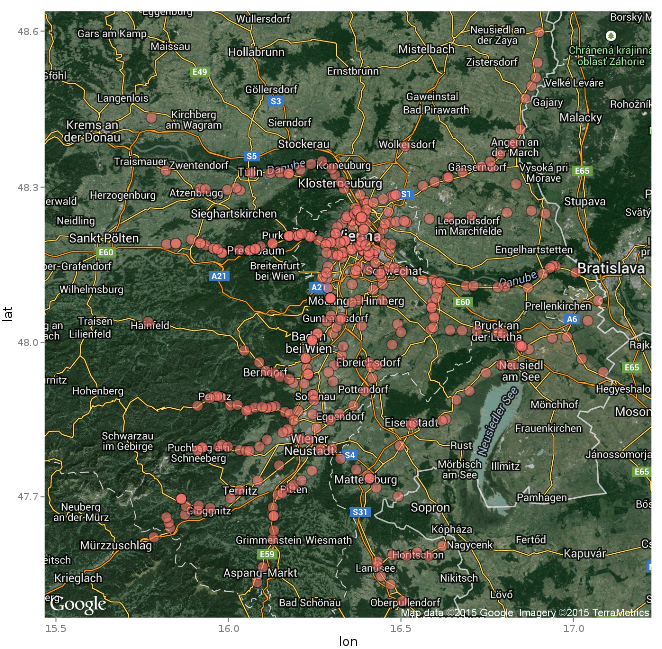
Here is the used R code:



I was using RMongo because it is very easy to handle, the code imports the mongodb data into R, make some transformation on it and plots the lon and lat parameters from all nodes into a map.

This code produces the following chart:

The red points represents nodes where the operator is OBB. Most of the buildings are railways stations. What you also can see very nice is the train track from the red points.



This project was great, to the beginning of the course I decided to work with Google Cloud Platform. Turing the working I had a lot of troubles with the performance with the Cloud, I was not able to run the program data.py to the end after 6 hours and more I killed the process.It takes me a lot of time and financial cost I got a bill of 115 USD from Google by using their cloud.

On my windows vista machine everything went fine to complete the project.

Note: the programs are very short so my comments on it are very rare, most of them are self-explanatory.

|  |
| --- |
| **Source** |

1. <http://www.redmountainsw.com/wordpress/257/pulldom-with-python-tutorialdocumentation-or-code-sample/>
2. <https://docs.python.org/3/library/xml.dom.pulldom.html>
3. <https://docs.python.org/2/library/xml.dom.html>
4. <http://docstore.mik.ua/orelly/other/python/0596001886_pythonian-chp-23-sect-3.html>
5. <http://www.wspiegel.de/tkinter/tkinter01.htm>
6. <http://www.joelonsoftware.com/articles/Unicode.html>
7. <https://packages.debian.org/sid/python/python-unidecode>
8. <https://pypi.python.org/pypi/Unidecode/>
9. <http://markedmondson.me/run-r-rstudio-and-opencpu-on-google-compute-engine-free-vm-image>
10. <http://stackoverflow.com/questions/23130604/r-plot-coordinates-on-map>