

Map Area

I chose sendai, japan for this OpenStreetMap case study.

Sendai is my second hometown that I spent most of my time in college.

- <https://www.openstreetmap.org/relation/4135014>

1. Problems Encountered in the Map

Postal Codes

When I check what kind of postal codes in the osm file for sendai, I found many postal codes with hyphen. Using hyphen in postal code makes it readable for people, but it is convenient for data wrangling with no hyphen numbers(just 7digits).

The result was below.

```
# count each postal code in osm file
{'980-0065': 1,
 '980-0802': 1,
 '980-0822': 1,
 '980-0862': 1,
 '980-8671': 1,
 '981-0122': 1,
 '981-0952': 2,
 '981-1211': 3,
 '981-1224': 6,
 '981-1231': 1,
 '981-1292': 1,
 '9811224': 1,
 '9820003': 1,
 '9820011': 3,
 '9830852': 3,
 '985-0002': 2,
 '985-0016': 2,
 '985-0052': 1,
 '985-8510': 1,
 '989-3128': 1}
```

So, I modified all of postal codes to no-hyphen digits before convert to json file.

Sort postcodes by count, descending

```
>db.sendai.aggregate([{"$match":{"address.postcode":{"$exists":1}}}, {"$group":{"_id":"$address.postcode", "count":{"$sum":1}}}, {"$sort":{"count":-1}}])
```

```
{u'_id': u'9811224', u'count': 7}
{u'_id': u'9830852', u'count': 3}
{u'_id': u'9811211', u'count': 3}
{u'_id': u'9820011', u'count': 3}
{u'_id': u'9810952', u'count': 2}
{u'_id': u'9850002', u'count': 2}
{u'_id': u'9850016', u'count': 2}
```

```
{u'_id': u'9811292', u'count': 1}
{u'_id': u'9810122', u'count': 1}
{u'_id': u'9800822', u'count': 1}
{u'_id': u'9820003', u'count': 1}
{u'_id': u'9808671', u'count': 1}
{u'_id': u'9811231', u'count': 1}
{u'_id': u'9850052', u'count': 1}
{u'_id': u'9800802', u'count': 1}
{u'_id': u'9800065', u'count': 1}
{u'_id': u'9858510', u'count': 1}
{u'_id': u'9893128', u'count': 1}
{u'_id': u'9800862', u'count': 1}
```

As you can see, all postcodes are converted to just 7digits.

2. Data Overview

This section will analyse basic statistics about sendai osm data.

File Size

sendai_japan.osm : 200MB

sendai_japan.osm.json : 282MB

Number of documents

```
>db.sendai.find().count()
1057125
```

Number of ways

```
>db.sendai.find({"type":"way"}).count()
96668
```

Number of nodes

```
>db.sendai.find({"type":"node"}).count()
960457
```

Number of unique users

```
>len(db.sendai.distinct("created.user"))
486
```

Top 3 contributing user

```
>db.sendai.aggregate([{"$match": {"created.user": {"$exists": 1}}}, {"$group": {"_id":
"$created.user", "count": {"$sum": 1}}}, {"$sort": {"count": -1}}, {"$limit": 3}])
```

```
{u'_id': u'Tom_G3X', u'count': 149406}
{u'_id': u'nori_u', u'count': 144625}
{u'_id': u'ikiya', u'count': 140709}
```

Number of changeset for each year

```
>pipeline = [
```

```

    {"$match": {"created.timestamp": {"$exists": 1}}},
    {"$group": {"_id": "$created.changeset",
                "timestamp": {"$first": "$created.timestamp"}}},
    {"$project": {"_id": "$_id",
                  "time-range": {
                      "time-range": {
                          "$concat": [
                              {"$cond": [{"$gte": ["$timestamp", '2016-01-01T00:00:00Z']], "2016", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2015-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2016-01-01T00:00:00Z']}}], "2015", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2014-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2015-01-01T00:00:00Z']}}], "2014", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2013-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2014-01-01T00:00:00Z']}}], "2013", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2012-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2013-01-01T00:00:00Z']}}], "2012", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2011-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2012-01-01T00:00:00Z']}}], "2011", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2010-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2011-01-01T00:00:00Z']}}], "2010", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2009-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2010-01-01T00:00:00Z']}}], "2009", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2008-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2009-01-01T00:00:00Z']}}], "2008", ""}],
                              {"$cond": [{"$and": [{"$gte": ["$timestamp", '2007-01-01T00:00:00Z']], {"$lt": ["$timestamp",
'2008-01-01T00:00:00Z']}}], "2007", ""}],
                          ]
                      }
                  }
    },
    {"$group": {"_id": "$time-range",
                "count": {"$sum": 1}}},
    {"$sort": {"_id": 1}}
]
>db.sendai.aggregate(pipeline)

```

```

{'u'_id': u'2007', u'count': 1}
{'u'_id': u'2008', u'count': 6}
{'u'_id': u'2009', u'count': 43}
{'u'_id': u'2010', u'count': 242}
{'u'_id': u'2011', u'count': 2962}
{'u'_id': u'2012', u'count': 533}
{'u'_id': u'2013', u'count': 1199}
{'u'_id': u'2014', u'count': 770}
{'u'_id': u'2015', u'count': 1295}
{'u'_id': u'2016', u'count': 135}

```

3. Additional Ideas

In data exploration process above, I found that a lot of data points were added to OpenStreetMap for sendai in 2011. This is because of Great East Japan Earthquake which was happened in March 11, 2011. Sendai was hit by it directly, so many things such as roads, buildings, harbor broke into pieces. Therefore, a lot of data was rewritten as information about sendai changed.

But I think the number of changesets are not enough to describe current situation in sendai. As the interest of Great East Japan Earthquake decreases, OpenStreetMap for sendai seems to be less modified.

So I suggest that modifying the OpenStreetMap to be incorporated into one of the tasks of disaster restoration work. If it works, the nearest people to restoration site can modify up to date information to the map.