Project Summary

Area of study:

- Location: New Delhi,India
- Open Street Map URL
- Mapzen URL

Objective: Audit and clean the data set, converting it from XML to CSV format.

Problems Encountered in the Map

`mapparser.py` was used to count occurrences of each tag, with a result:

- `bounds: 1`
- `member: 16633`
- `nd: 300677`
- `node: 248773`
- `osm: 1`
- `relation: 653`
- `tag: 60324`
- `way: 45384`
- `note: 1`
- `meta: 1`

Additional functionality was added to `mapparser.py` to examine the keys stored in each `tag` element, in the `k` attribute. Unexpectedly, the 20 most common key values were:

[('building', 34566), ('highway', 7856), ('name', 3034), ('oneway', 1229), ('railway', 891), ('amenity', 822), ('landuse', 762), ('type', 655), ('boundary', 526), ('admin_level', 514), ('power', 449), ('gauge', 436), ('electrified', 414), ('leisure', 393), ('layer', 386), ('barrier', 353), ('bridge', 309), ('addr:housenumber', 301), ('natural', 295), ('voltage', 265)].

data.py was used to import the xml file to the csv file. Five csv file was created namely
nodes.csv,nodes_tags.csv,ways.csv,ways_tags.csv,ways_nodes.csv.
A database delhi.db was created by using csv file. The table name used in the database is same as the csv file name. Querying the database reveals some of the problems which is stated below:

1. In the node tag, the value of the key ‘state’ were inconsistent. Delhi state was referred to as ‘Delhi’,’NCR’, and ’DL’. All values were changed to ‘Delhi’.

2. Misspelling in the node tag where key is ‘source’ and value is ‘sourvey’. It was corrected.

3. There were two fields for the pincode(‘postal_code’ ,’postcode’) in both node tag as well as in way tag. ‘postal_code’ was changed to postcode.

4. In the country field, generally ISO standard for countries short code is used. But in some fields full name was used. Full names was converted into the ISO standard.

These problems in the node tag were addressed by the function audit_node_tags in the audit.py.

5. Inconsistency in the way tag where key is ‘source’ and value is ‘bing’,’Bing’,’Bing 2012’. All were changed to ‘Bing’.

6. Street name was inconsistent.(eg. Both extn. and extension was used). Here, we changed extn. to extension, Delhi. to Delhi and many more edits to make name more consistent.

These problems in the ways tags were addressed by the function audit_ways_tags in the audit.py.

Data Overview

This section contains basic statistics about the dataset and the SQL queries used to gather them.

File Included

delhi.osm .... 55.7 MB

mapparser.py

data.py

audit.py

schema.py

delhi.db

nodes.csv

nodes_tags.csv

ways.csv

ways_tags.csv

ways_nodes.cv

nodes_tags_updated.csv

ways_tags_updated.csv

Number of nodes

```
sqlite> SELECT COUNT(*) FROM nodes;
```

```
248774
```

Number of ways

```
sqlite> SELECT COUNT(*) FROM ways;
```

```
45385
```

Number of unique users

```
sqlite> SELECT COUNT(DISTINCT(e.uid))
FROM (SELECT uid FROM nodes UNION ALL SELECT uid FROM ways) e;
```

```
402
```

Top 10 contributing users

```
sqlite> SELECT e.user, COUNT(*) as num
FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways) e
GROUP BY e.user
ORDER BY num DESC
LIMIT 10;
```

```
saikumar   42331

kranthikumar   23216

premkumar  21178

bindhu   20953

harisha   18049

PlaneMad  15163

Naresh08   14483

Oberaffe   12551

n'garh  11111

sramesh   9635
```

Number of users appearing only once (having 1 post)

```
sqlite> SELECT COUNT(*)
FROM
   (SELECT e.user, COUNT(*) as num
    FROM (SELECT user FROM nodes UNION ALL SELECT user FROM ways) e
    GROUP BY e.user
    HAVING num=1)  u;
```

```
92
```

Additional Ideas

We can observe that there was more of a focus on the study of node ‘places’ rather than ways and their respective node ‘waypoints’. Given that an overwhelming number of nodes (85%) do not include addresses and the large number (300677) of nd reference tags, an interesting exercise would be to compress this data by removing any way tags and waypoint-like nodes. Naturally, this decision falls on the application. This could potentially reduce the database size by a factor of 10. Furthermore, if ways were still needed, it would still be possible to remove any ‘orphaned’ nodes that were only referenced in the removed relation and member tags.

Additional Data Exploration

Top 10 appearing amenities

```
sqlite> SELECT value, COUNT(*) as num
FROM nodes_tags
WHERE key='amenity'
```

```
GROUP BY value
ORDER BY num DESC
LIMIT 10;
```

```
restaurant  47
atm          45
fast_food        40
place_of_worship        37
embassy              36
school          34
fuel       30
toilets                 24
bank            23
cafe            18
```

Biggest religion

```
sqlite> SELECT nodes_tags.value, COUNT(*) as num
FROM nodes_tags
    JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value='place_of_worship') i
    ON nodes_tags.id=i.id
WHERE nodes_tags.key='religion'
GROUP BY nodes_tags.value
ORDER BY num DESC
LIMIT 1;
```

```
hindu 571
```

Most popular cuisines

```
sqlite> SELECT nodes_tags.value, COUNT(*) as num
FROM nodes_tags
    JOIN (SELECT DISTINCT(id) FROM nodes_tags WHERE value='restaurant') i
    ON nodes_tags.id=i.id
WHERE nodes_tags.key='cuisine'
```

```
GROUP BY nodes_tags.value
ORDER BY num DESC limit 5;
```

```
indian	3
North_Indian	2
thai	2
Chinese_and_North_Indian	1
asian	1
```

Conclusion

After this review of the data it’s obvious that the Delhi area is largely incomplete.There are still several opportunities for cleaning and validation that I left unexplored, though I believe it has been well cleaned for the purposes of this exercise.the data set is populated only from one source: OpenStreetMaps. While this crowdsourced repository pulls from multiple sources, some of data is potentially outdated. It would have been an interesting exercise to validate and/or pull missing information from the Google Maps API, since every node has latitude-longitude coordinates.