1. Map Area

The map chosen was that of the greater Guadalajara metropolitan area in Jalisco, Mexico. The data was aquired through the Overpass API, the coordinate ranges for the latitude and longitude are [20.9884, 20.3639] and [-103.8606, -102.8348] respectively.

2. Data Problems in the Map

The language difference along with the accents used in the Spanish language made this data particularly messy. There were however some fields such as 'addr:country' and 'addr:state' which contained no errors. As for the other fields with inconsistencies, I chose to focus on those encountered in 'addr:city', 'addr:postcode', 'addr:street', 'amenity', and 'cuisine'.

City Names

Some city names were completely illegitimate such as 'Morelia' which is not in the state of Jaliso, while others were inconsistent with spelling, accents, capitalization and/or included the state name along with the city name. For example, 'GUADALAJARA JALISCO', 'Guadalajara Jalisco', and 'Guadalajara, Jalisco'. These types of errors occurred in only a few of the main municipalities in the Guadalajara area and were corrected with a simple search and replace function. To make sure that other city names were legitimate, I scraped data from a wikipedia page containing city names in the state of Jalisco and compared them with those encountered in the OpenStreetMap file. If the city name was not in this list, it was omitted.

```
# Sort city names by count, descending

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

```
[{u'_id': u'Zapopan', u'count': 84},
 {u'_id': u'Guadalajara', u'count': 31},
 {u'_id': u'Tlaquepaque', u'count': 28},...]
```

These results don't really reflect the area that each municipality covers in the metropolis. My suspicion is that the amount of entries corresponds with the wealth in each area. From my experience I know that most wealthy communities are in Zapopan and it is likely that here more people can afford expensive equipment and the time needed to contribute to OpenStreetMap data.

Postal Codes

Several postal codes in the data had trailing zeros or whitespace, and others were not valid postal codes for the state of Jalisco, Mexico. I confirmed valid postal codes via wikipedia, those that are valid begin with numbers 44 through 48 and are 5 digits long in total. Any postal codes not beginning with these numbers were left out. For those with trailing zeros or whitespace, it was assumed that these characters were mistakes and they were removed from the rest of the postal code.

Street Names

There was a lot of mess in the street name fields, it is unclear if some street names left out street types inadvertently and in fact were another street name already in the data such as 'Av. Vallarta' and 'Vallarta'. It is entirely possible that some of these names without street types were different from their apparent counterparts so I chose not to make any assumptions. I did however expand any street abbreviations to the full word such as 'av' to 'Avenida', 'esq' to 'Esquina', and 'prol' to Prolongación.

```
# Sort street names by count, descending, limit 6
>db.guad.aggregate([{"$match":{"address.street":{"$exists":1}}}, {"$group":{"_id":"$address.street", "count":{"$sum":1}}}, {"$sort":{"count":-1}}, {"$limit": 6}])

[{u'_id': u'Avenida del Bosque', u'count': 7},
 {u'_id': u'Juarez', u'count': 7},
 {u'_id': u'Intel', u'count': 7}]
```

There is a large Intel office in Guadalajara. That this smaller street has as many entries as some of the larger avenues gives us an idea of where some of the contributors might work.

Amenity Types

For the amenity attribute in the field tag, there were some inconsistencies such as 'parking' and 'parking_space'. Any seemingly interchangeable values were simplified to a basic derivative such as 'parking'.

```
# Sort amenities by count, descending, limit 10
>db.guad.aggregate([{"$match":{"amenity":{"$exists":1}}}, {"$group":{"_id": "$amenity", "count": {"$sum":1}}}, {"$sort": {"count": -1}}, {"$limit": 10}])

[{u'_id': u'school', u'count': 144},
 {u'_id': u'fuel', u'count': 135},
 {u'_id': u'parking', u'count': 123},
 {u'_id': u'restaurant', u'count': 107},
 {u'_id': u'bank', u'count': 100},
 {u'_id': u'place_of_worship', u'count': 94},
 {u'_id': u'fast_food', u'count': 83},
 {u'_id': u'waste_basket', u'count': 81},
 {u'_id': u'pharmacy', u'count': 64},
 {u'_id': u'hospital', u'count': 56}]
```

I am surprised that the most popular amenity is a school. In my experience driving around Guadalajara, I would say that the most frequently seen amenities are convenience stores, pharmacies, banks, and gas stations. Because of their commonplace I could imagine people not thinking to include them however, and instead opting to document more scarce but socially important places.

Cuisine Types

The cuisine attribute contained several different types of inconsistencies. Naming issues such as 'Hot_Dogs', 'hot_dogs', or 'Hot_Dogs_Gourmet', were all updated to simple derivative such as 'hot_dogs'. There were also some problems with primary language, as we had fields with 'seafood' and 'mariscos'. Because most values were in English (also values are in English even in the Spanish translated OSM wiki), I chose to change anything redundant in Spanish to it's English counterpart. Some values had more than one cuisine type such as 'sushi,_burgers'. These were split up into a list for the 'cuisine' field.

```
# Sort cuisines by count, descending, limit 5
>db.guad.aggregate([{"$match":{"cuisine":{"$exists":1}}}, {"$group":{"_id": "$cuisine", "count":
{"$sum":1}}}, {"$sort": {"count": -1}}, {"$limit": 5}])

[{u'_id': u'mexican', u'count': 27},
 {u'_id': u'pizza', u'count': 18},
 {u'_id': u'burger', u'count': 15},
 {u'_id': u'ice_cream', u'count': 7},
 {u'_id': u'seafood', u'count': 5}]
```

There is definitely no surprise that the number 1 cuisine is Mexican, however I don't think many people would realize how common it is to eat burgers and pizza in Guadalajara.

3. Data Overview

File Sizes
guad.osm … 59.5 MB
guad.json … 67.5 MB

```
# Number of documents
> db.guad.find().count()
320485

# Number of nodes
> db.guad.find({"type":"node"}).count()
276498

# Number of ways
> db.guad.find({"type":"way"}).count()
43986

# Number of unique users (not command line)
len(db.guad.distinct("created.user"))

#Top 1 Contributing user
> db.guad.aggregate([{"$group":{"_id": "$created.user", "count":{"$sum":1}}}, {"$sort": {"count":
-1}}, {"$limit": 1}])

{ "_id" : "minoxfilm", "count" : 200766 }

# Number of users appearing only once
> db.guad.aggregate([{"$group":{"_id": "$created.user", "count":{"$sum":1}}}, {"$group":{"_id":
```

```
"$count", "num_users":{"$sum":1}}}, {"$sort": {"_id": 1}}, {"$limit": 1}])
[{u'_id': 1, u'num_users': 79}]
```

4. Additional Ideas

Suggestion for Further Cleaning and Non-English Standards

For the 'addr:street' field it seems that there is still some cleaning that could be done. Mostly these are spelling errors or absence of the full street name, but there is also a need for consistency of accent marks. A large street in Guadalajara named 'Niños Héroes' is found in the data with and without accent marks. Of course these accents make it especially difficult because some are in unicode and others are coded as UTF-8. I think it would be helpful if the OSM wiki had some standards for contributing data with accent marks.

Another issue with clean data is perhaps a cultural one. It is common in Mexico for some addresses to contain 'Esquina' which means 'corner of'. These addresses include two street names and inputting this type of data into MongoDB for later queries can be problematic. In this case we have to use a $regex query to find the street name rather than a simple field-value match.

```
>db.guad.find({"address.street" : {$regex : ".*Esquina.*"}}).pretty()
```

The problem with suggesting this type of change to contributors is that a contributor might have to make up an address that isn't necessarily accurate in order to avoid this 'Esquina' problem.

Another issue I noticed were redundant field values across languages. For example, 'Abarrotes' is most closely related to a 'convenience' store (you could argue that it is a 'general' store but for this data I did not find 'general' store as a value even though this value is listed in the OSM wiki). I could see why there may be some confusion on the part of a contributor... what is the proper translation? In any case, having field values that are in different languages but generally representing the same word makes things redundant. Perhaps there should be a reference for each country/region on what to do in these culturally specific situations.

Additional data exploration using MongoDB queries

```
#Most popular sports
>db.guad.aggregate([{"$match":{"sport":{"$exists":1}}}, {"$group":{"_id": "$sport", "count":
{"$sum":1}}}, {"$sort": {"count": -1}}, {"$limit": 5}])

[{u'_id': u'soccer', u'count': 188},
 {u'_id': u'basketball', u'count': 65},
 {u'_id': u'tennis', u'count': 21},
 {u'_id': u'baseball', u'count': 13},
 {u'_id': u'swimming', u'count': 8}]
```

As anticipated, soccer is the top sport. I did however expected to see baseball higher than basketball. Although there are more basketball courts, which are smaller in area than baseball fields, it doesn't necessarily mean that the sport itself is more popular. I have often seen kids playing baseball on the street.

```
#Most popular shops
>db.guad.aggregate([{"$match":{"shop":{"$exists":1}}}, {"$group":{"_id": "$shop", "count":
{"$sum":1}}}, {"$sort": {"count": -1}}, {"$limit": 4}])
```

```
[{u'_id': u'convenience', u'count': 196},
 {u'_id': u'supermarket', u'count': 72},
 {u'_id': u'mall', u'count': 31},
 {u'_id': u'car_repair', u'count': 27}]
```

Convenience stores are everywhere in Guadalajara and Mexico for that matter. The top two are definitely as expected.

```
#Most popular religions
>db.guad.aggregate([{"$match":{"religion":{"$exists":1}}}, {"$group":{"_id": "$religion", "count":
{"$sum":1}}}, {"$sort": {"count": -1}}, {"$limit": 2}])
```

```
[{u'_id': u'christian', u'count': 77}, {u'_id': u'jewish', u'count': 1}]
```

There are only two different religions in the data, and as we can see there is a landslide in favor of Christianity.

Conclusions

The Guadalajara OSM compared to other metropolitan areas is still very new. There are many more data points to be added to this area in the future. Because of the inconsistencies that I've seen in this data set and the potential for much more dirty data to be added I think two things in particular would help ensure clean data going forward. One is some standard with regard to accents (or their absence) and their encoding. The other, a reference for common problems encountered (such as multiple street names) and what to do in such situations.