A Parking Asset Management System Implementation

EXECUTIVE SUMMARY

Nathalia Roberts
UW-MADISON GEOGRAPHY 778 | PRACTICUM IN GIS DEVELOPMENT

Introduction

This project aimed to demonstrate the primary use of open-source software for management and analysis of spatial data for parking-related assets in the city of Oakland, CA. In recent years, the city has experienced economic and population growth that has created an increase in demand for parking, affecting visitors and residents alike. In response, the city has implemented programs to help mitigate the issues that arise from high parking demand and limited parking supply in dense commercial and residential areas.

For example, programs such as Residential Permitted Parking (RPP) and Demand-Responsive Parking (DRP) are used to help increase parking availability. In neighborhoods with commercial districts and dense housing, residents can elect to create an RPP zone, which requires parking permits to park on the street for an extended time in the area. If a vehicle is parked beyond the posted time limit and lacks a permit, it is subject to citation.

In commercial areas with on-street parking meters, DRP is used to set parking meter rates at prices that encourage motorists to park for short periods of time and look elsewhere for longer-term parking options. The desired parking occupancy range for a given block is 70% - 85% and meter rates are adjusted over time to encourage this level of parking occupancy.

A parking asset management system can be used to help implement programs like these effectively and aid in planning decisions. Questions that require spatial analysis can be answered by structuring the data and modeling their relationships.

Audience

The targeted audience for this project are data managers, planners, and analysts for a city Department of Transportation that would find this workflow useful for data organization, maintenance, and analysis. Additionally, this may be useful for researchers or consultants that work with urban transportation issues and use city data for research.

Database Development | OakPark

A PostgreSQL database was developed using the PostGIS spatial extension for storing geographic data. Database design included the creation of an Entity-Relationship diagram and a Relational schema. Significant data cleaning was performed to have the entity attributes conform to the model design.

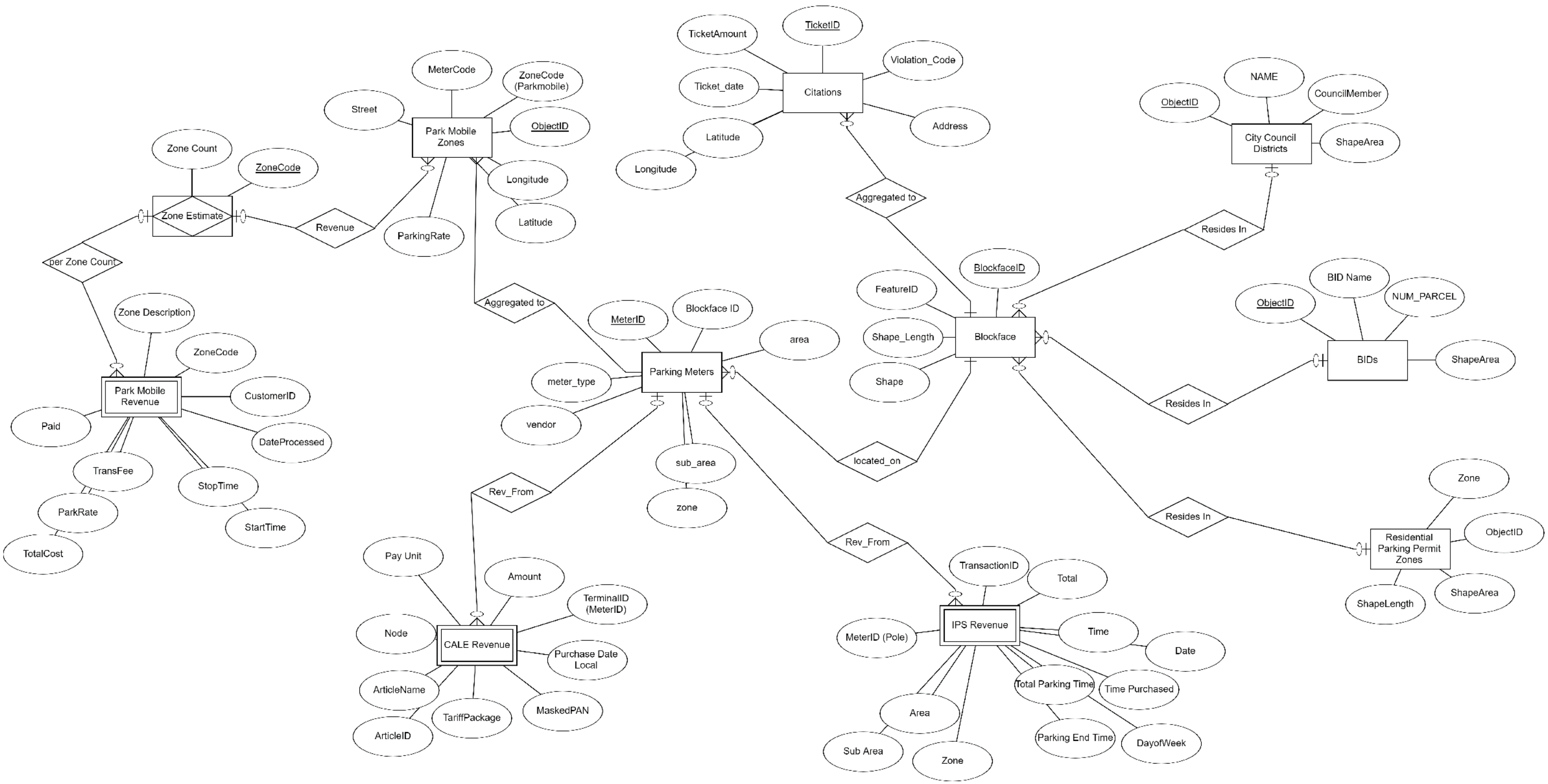

Figure 1. Entity-Relationship Diagram

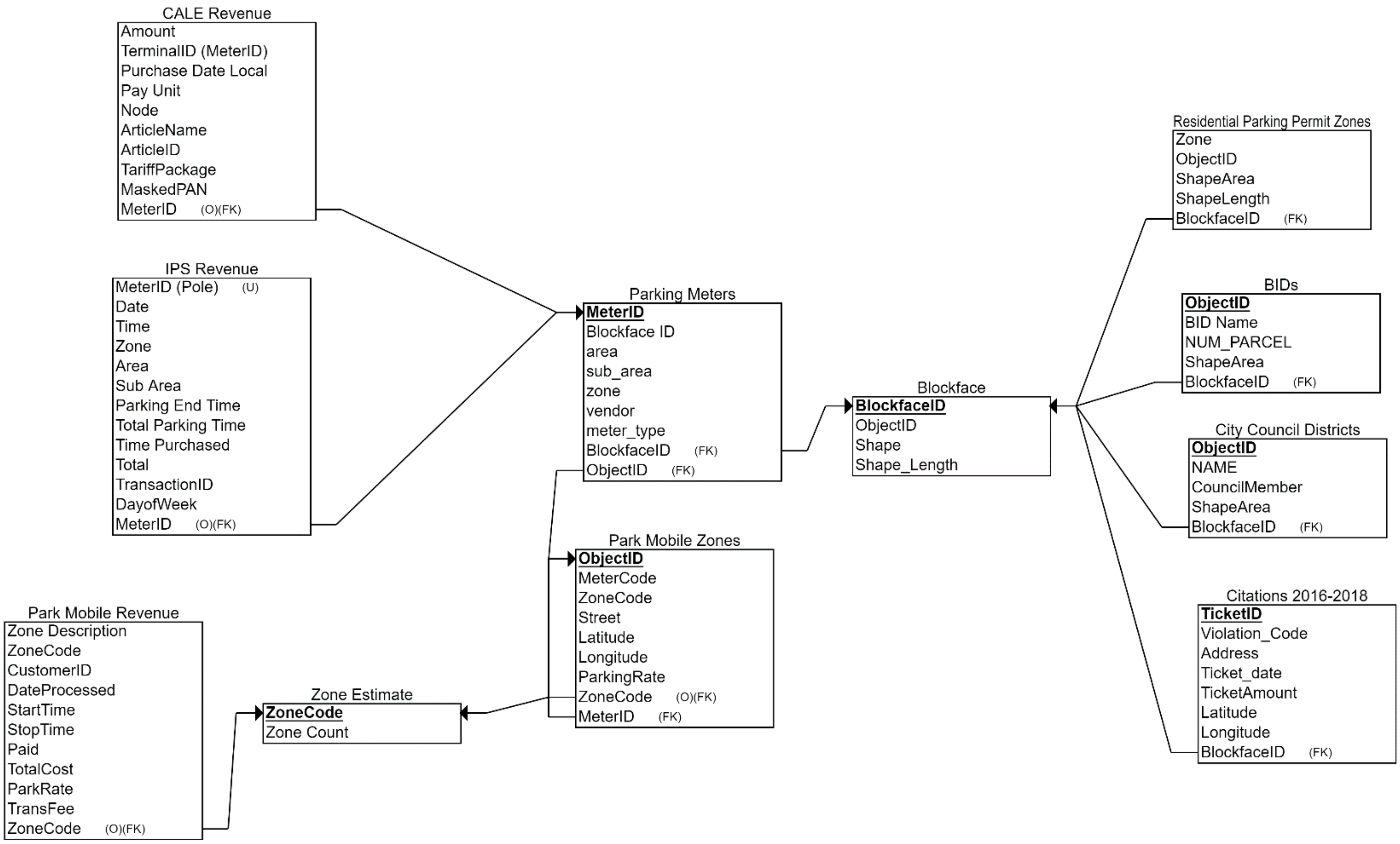

Figure 2. Relational Schema

Because the key purpose of this database is to allow the archival storage of parking meter revenue from multiple vendors, it is technically more appropriate to refer to it as a data warehouse. A data warehouse is denormalized, whereas a database is normalized for optimal performance. One example where optimal database performance is essential would be for an online payment system that processes transactions in real-time. In contrast, a data warehouse is for historical data and pulls from many different sources within an organization for reporting and analysis.

For this report, database, and data warehouse will be used interchangeably but note that because of the disparate data from the parking meter vendors IPS, CALE and Parkmobile, formal database normalization is not practical.

In Figure 1, the majority of the relationships throughout the model was many-to-one, such as many transactions for one parking meter, and many meters to one blockface. An exception is the Zone Estimate entity which models a many-to-many relationship between Parkmobile zones, their associated revenue, and their relationship to parking meters and blockfaces. Unlike parking meters that can be tied to a specific block, individual Parkmobile zones can be tied to multiple blocks which makes it more difficult to calculate the revenue for the zone at the level of the blockface. Approximately 60% of Parkmobile zones for the City of Oakland cover two or more blockfaces.

To ensure that revenue was not double counted at a blockface level, Zone Estimate functioned as an associative entity with the Parkmobile zones (Zonecode) as the primary key. A field that counted the total instances of Parkmobile zones (Zonecount) was created based on analysis from the Parkmobile spatial data provided by the city. After Parkmobile's quarterly revenue was grouped by Zonecode it was averaged by Zonecount which was ultimately joined to the blockface.

Primary keys were absent from original vendor revenue but were added after the data was loaded to the database. Spatial data was loaded with the PostGIS Shapefile Import/Export Manager. SRID 2227 was specified to maintain projection, and spatial indexes were created upon upload.

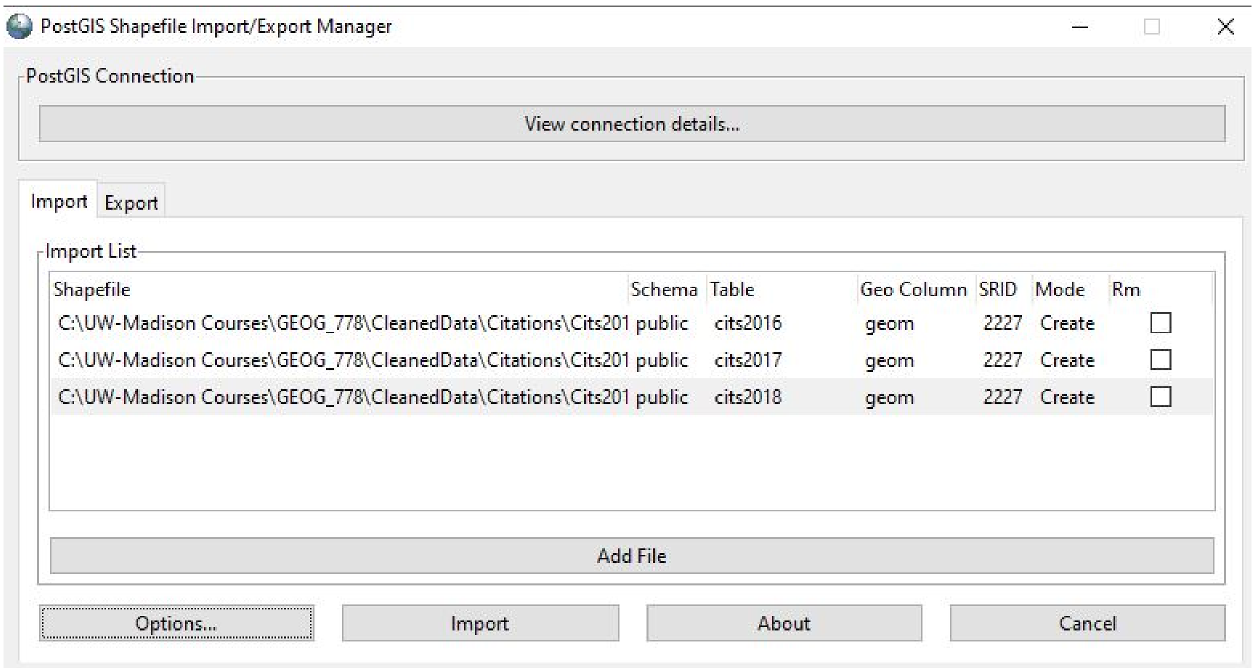

Figure 3. PostGIS spatial data loader

Foreign key constraints could not be formally added to IPS or CALE revenue data because the parking meter spatial data is a common record of both vendors. This created foreign key constraint violations.

Parking meter citations were geocoded using a geocoder that was built in ArcGIS and converted to shapefiles based on citation year. Citations were ultimately broken down by year because of file size constraints of the shapefile data format. Approximately 27 million records were loaded for parking meter revenue and parking citations. Two schemas were used in the database. Parking meter transaction tables and their related query tables were stored in the Revenue schema and spatial data was kept in the Public schema. Quarterly parking meter revenue was calculated for all of 2017 and 2018 for IPS, CALE, and Parkmobile transactions which resulted in the creation of 35 tables in the Revenue schema. There are 14 tables in the Public schema.

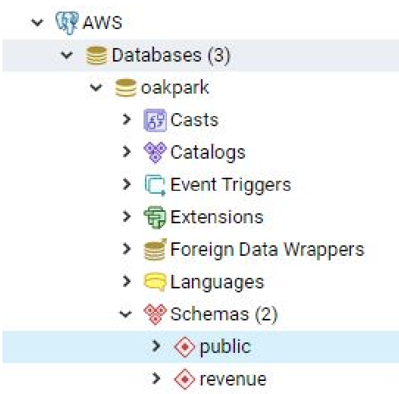

Figure 4. PostgreSQL Database Schema

For greater accessibility, cloud hosting was implemented. A PostgreSQL database instance was created using the AWS Amazon Relational Database Service (RDS). Amazon offers a free 1-year trial, afterward, the service is fee-based.

Host name: pamsdemo.cbpltcizfdj7.us-west-1.rds.amazonaws.com
Username: postgres
Password: nitehawks1942
Port: 5435

Major OakPark Tables

Name	Description
Blockfaces	Line feature class representing city blocks with parking meters. Blockface ID is the unique ID tied to parking meters.
Parking Meters	Point feature class representing multispace and single space parking meters. Meters are on-street and off-street (parking lots). Meter ID is the unique identifier.
Parkmobile	Point feature class representing Parkmobile zones.
Parkmobile Zone Count	Parkmobile Zones are the unique ID.
Blockfaces Quarterly Revenue (blockfaces_rev_qtr)	Quarterly parking meter revenue totals aggregated to the blockface.
Parking Meter Quarterly Revenue (parking_rev_qtr)	Quarterly parking meter revenue totals.
Citations (cits2016, etc.)	Point feature class of parking citations.
RPP Zones	Polygon feature class representing boundaries of Residential Parking Permit Zones.

QGIS Analysis

QGIS is an open-source desktop GIS software. Analysts that are comfortable with ArcGIS Desktop can easily familiarize themselves with the QGIS interface. An advantage of QGIS software is the lack of licensing requirements for database connectivity and editing. Using it to connect to the OakPark database enabled analysis that answered a question posed during the early stages of the project:

For a given blockface or area, what has generated more revenue over time? Fines or fees?
Using College Avenue as the area and 2018 Q4 as the time point, revenue generated by all of the parking payment vendors was aggregated to the blockface. QGIS filtered citations and revenue to Q4 data and created a 20ft. buffer around the blockfaces. All citations that fell within the buffer distance were spatially selected. *Fines in the area totaled ~ $31,499 while revenue was ~$141,382. Parking meter revenue is approximately 4.5 times the amount of citation revenue on College Avenue for 2018 Q4.*

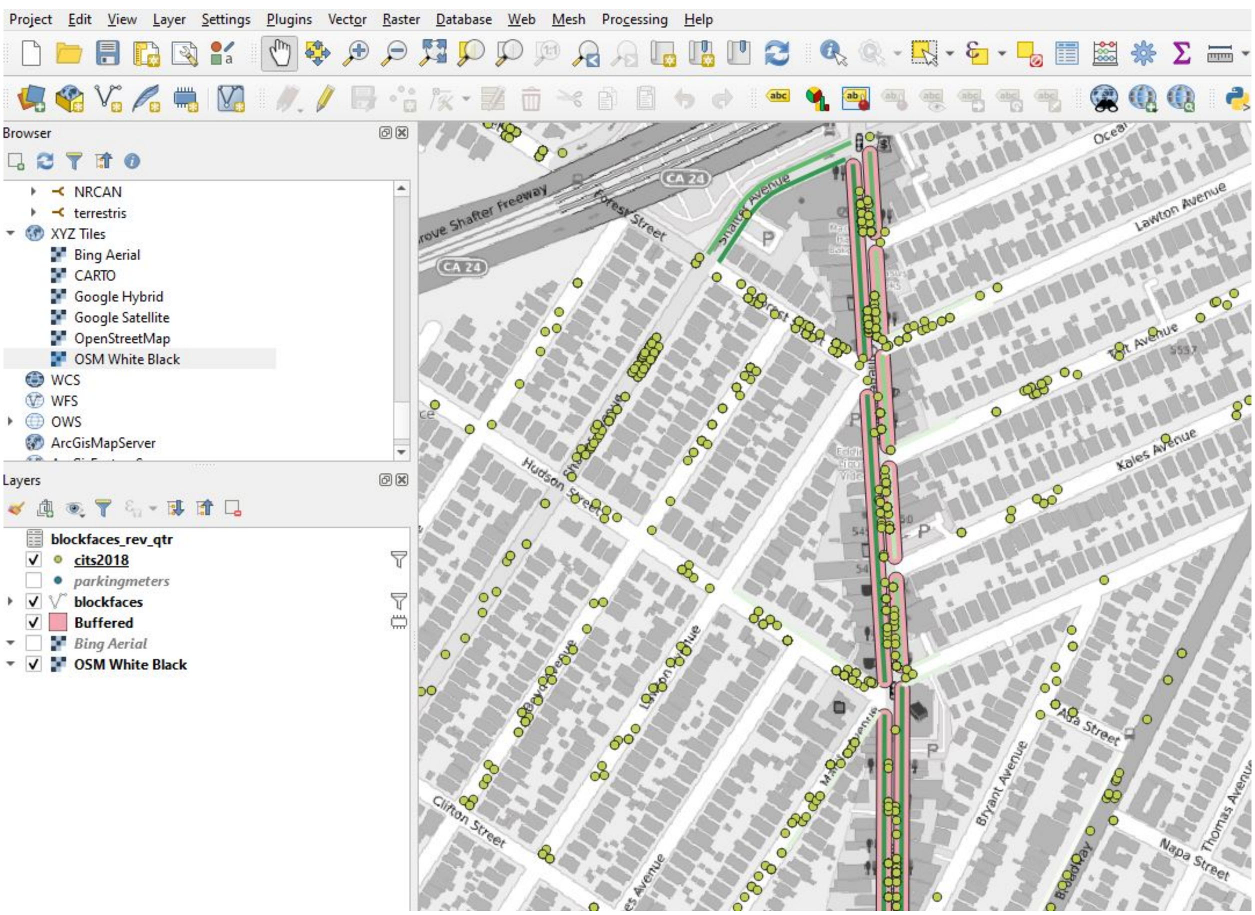

Figure 5. Revenue analysis in QGIS.

Jupyter Notebook Analysis using Python

Using Jupyter Notebooks for analysis was valuable because functions or queries were scripted into the notebook and could generate the output data, provide a narrative description of that data, and generate visual aids for display. It is described as an open-source, web-based, interactive computing notebook environment.

A demonstration notebook was created for the project that connects to the OakPark database and a series of basic analysis questions were posed. Figure 6 displays a simple bar chart generated using the Pandas and Matplotlib libraries for Python.

What are the 10 most profitable parking meters in Oakland for Q3 2018? What is the year over year change compared with 2017?

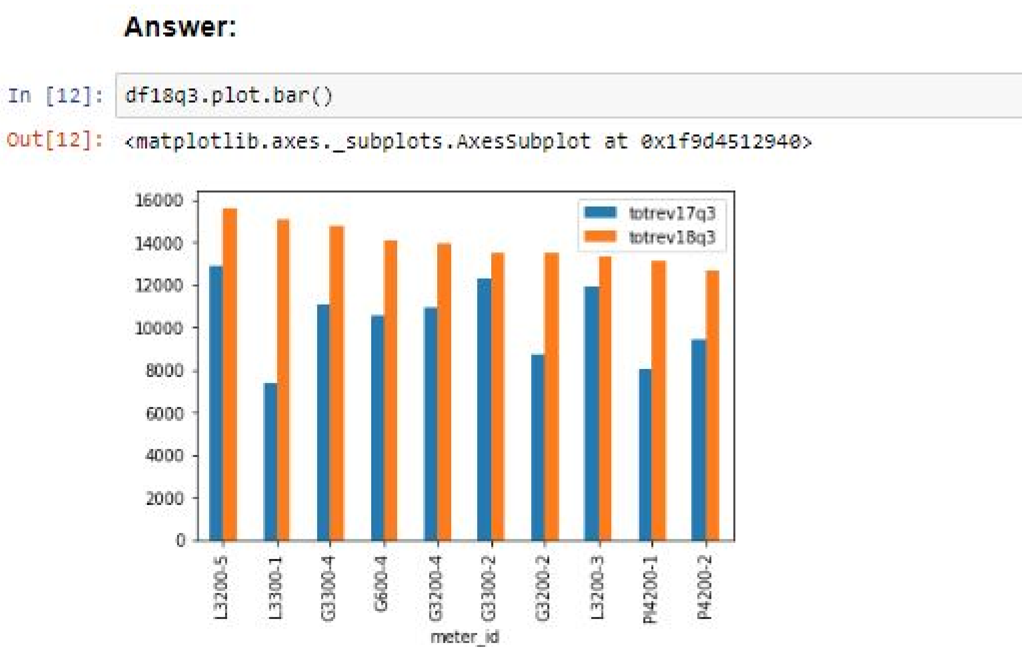

Figure 6. Jupyter Notebook Data Analysis

Going Forward

This project has demonstrated the robust capability of open-source software for data modeling, management, and analysis of parking asset data for the City of Oakland. The following are additional recommendations:

- Summary statistic examples are shown in the workflows, but other modeling is now possible. For example, parking meter transactions can be segmented by the time of day to see what variations exist. Such information can be used to help implement varying meter rates based on peak parking demand.

- More data can be added for curb management, such as curb regulation data and street signage related to parking. This can enable a more focused analysis of curb regulations and their relationship to parking citations throughout the city.
- To simplify the relationship model for Parkmobile zones, more unique zone IDs should be created that tie to the blockface. As of mid-2019, 40% of Parkmobile zones were already at this level of analysis.

Project Research Sources:

1. *San Francisco Adopts Demand-Responsive Pricing Program to Make Parking Easier.* SFMTA, Dec. 2017.
   Retrieved from: https://www.sfmta.com/blog/san-francisco-adopts-demand-responsive-pricing-program-make-parking-easier

2. *Downtown Oakland Parking Management Report.* City of Oakland/ Metropolitan Transportation Commission, June 2016.
   Retrieved from:
   http://www2.oaklandnet.com/oakca1/groups/pwa/documents/report/oak059357.pdf
   http://sfpark.org/resources/docs_pilotevaluation/

3. City of Oakland Parking Permits. Sept. 2019.
   Retrieved from: https://www.oaklandca.gov/topics/parking-permits

4. *2018 Annual Paid Parking Study.* Seattle Department of Transportation, Oct. 2018.
   Retrieved from:
   https://www.seattle.gov/Documents/Departments/SDOT/ParkingProgram/PaidParking/SDOT_AnnualReport2018.pdf

5. *The Difference Between a Data Warehouse and a Database.* Panoply, 2018.
   Retrieved from:
   https://panoply.io/data-warehouse-guide/the-difference-between-a-database-and-a-data-warehouse/

6. Data was provided from the following Oakland Department of Transportation data portals:
   https://data.oaklandnet.com | https://oakgis.maps.arcgis.com/home/index.html
   https://www.oaklandca.gov/resources/parking-mobility-related-maps-and-data