Data Processing Automation with Python and PostgreSQL

Irish OSGeo Symposium

May 26th 2017

Heikki Vesanto

Lexis Nexis

Background

Heikki Vesanto

- Scottish QGIS User Group:
 - Presented at the 1st, 2nd and 4th user group meetings
- > QGIS Plugins:
 - Multi Ring Buffer Plugin
 - Select Within Plugin
- Top 10% contributer on GIS.StackExchange
- Lexis Nexis
 - GIS Administrator and Developer



GIS Software

From Oracle + esri ArcGIS + Python







© pentaho® A Hitachi Group Company

Why Open Source?

Flexibility

- ➤ Want to do x analysis, you will need y extension.
- > Adding RAM to your server? License cost will double.
- Integration
 - > Open standards
 - QGIS / PostGIS seamless integration
 - \succ One place for data
- > Cost
 - ➢ Free
- Function
 - \succ Has the functionality we require, or allows us to build it

PostgreSQL + PostGIS

> Database:

- ➤ 2500 GB of data
- ➢ 60 Schemas
- ➤ 1500 Tables
- 300 Custom Functions
- > Interaction:

> QGIS

- ➢ GDAL & OGR − Command line tools for data loading/translation
- > Python GDAL/OGR, psycopg2
- ➢ pgAdmin − SQL queries
 - Data visualisation
- pentaho Open Source ETL tools Spoon/Kettle

Case study

- Calculate drive times for each address in the United States to the nearest fire stations
- ➢ 200,000,000 addresses
- ➢ 60,000 fire stations
- OpenStreetMap road network

- > What is the drive time to the nearest fire station?
- ➢ How many fire stations within a 6 minute drive time?

Case study

- Calculate drive times for each address in the United States to the nearest fire stations
- ➤ 200,000,000 addresses
- ➢ 60,000 fire stations
- OpenStreetMap road network

- > What is the drive time to the nearest
- > How many fire stations within a 6 mi



Fire stations



Addresses

Addresses

Too Many for QGIS to Render

Process

- Calculate drive time from each fire station to each intersection on the network within 30 minutes
 - Using intersections allows us to calculate drive times for 5 million intersections instead of 200 million addresses
 - Done using a python script to launch pgRouting functions
 - > Python allows us to use parallel processing



Process

- Join each address to the nearest road, check for fire station drive times at each intersection
- > Linear referencing for the distance between the address and the intersection
- Calculation done in a PostGIS function, but run from Pentaho, allowing for multiple instances of each query



Pentaho Workflow:



North Dakota – Drive time to nearest fire station



North Dakota – Drive time to nearest fire station



North Dakota – Drive time to nearest fire station



Conclusion

- Successful calculation for our 200 million addresses
 - > Run time around two weeks
 - > 20 hours for aggregation/linear referencing
- Automation of most workflows
 - Data loading
 - Data delivery
 - Data processing
- PostgreSQL for data storage
- Python for processing
- > Pentaho for stringing the two together and controlling flows

Thank You

Heikki Vesanto GIS Administrator and Developer Heikki.Vesanto@LexisNexisRisk.com

@HeikkiVesanto

Due to the nature of the origin of public record information, the public records and commercially available data sources used in reports may contain errors. Source data is sometimes reported or entered inaccurately, processed poorly or incorrectly, and is generally not free from defect. This product or service aggregates and reports data, as provided by the public records and commercially available data sources, and is not the source of the data, nor is it a comprehensive compilation of the data. Before relying on any data, it should be independently verified. LexisNexis and the Knowledge Burst logo are registered trademarks of Reed Elsevier Properties Inc., used under license. All rights reserved.