Blue Zoo Data Platform

I was very interested when I came across the Disparate Data Challenge late last month because I've been thinking about a solution right along those lines and looking for an opportunity to develop my ideas further. I'm a software entrepreneur and have previously founded two successful software products. This challenge gave me the incentive to spend more time exploring my ideas with some actual working code. My initial experience has been positive and I look forward to pushing the idea further.

My background is in computer science. I have a doctorate degree in programming languages and have been working the past several years in the domain of data science. I've spent a lot of this year getting involved in the Internet of Things. The remainder of this document describes the vision, the work done to date, and future directions.

Design

My concept is not to create another database, but rather create an interactive run-time environment that enables users to pull together and compose data from many sources. The guiding principles of the design are:

- No ingestion – data is queried in its original format, possibly remotely
- Interactive – data is incrementally collected and composed into an in-memory working set
- Schema free – data is a heterogeneous mixture of objects with different schema
- Intelligent[1] – expression/language variation can be minimized though NLP/ontologies

Prototype

In the past few weeks, I've put together a prototype with as many features as possible to demonstrate the guiding principles. The main features of the prototype are:

- Heterogeneous formats – 3 formats completed (wms, csv, shapefile). Limited only by time, additional formats are easily implemented such as json, xslx or geotiff.
- CKAN connector – provides discovery and download of the thousands of files available from CKAN catalogs such as data.humdata.org.
- Data discovery – layers are searched by keywords and concepts via an ontology.
- Schema discovery – layer previews and data summarization results provide schema exploration.
- Original format search – searches are performed on local files in their original format and on WebMapping services remotely via the WMS API.
- Heterogeneous workspace – multiple layers are searched to build a collective result set of features with varying schemas.
- Aggregate related results – group by and aggregation functions collect information about related objects in the result set.

[1] Well Artificially Intelligent anyway

- Geocoding – all objects including csv records with lat/lon columns are reverse geocoded so that that they can be grouped by area at various administration levels.
- Visualization – a visualization tool provides drag-n-drop visualization design.

The following short video demonstrates some of these features:

https://youtu.be/mw74b34bGss

Future Direction

This section discusses several features that I envision in an eventual product that didn't make it into the first prototype. My goal would be to develop *some* of these for stage two of the challenge if it passes stage one. These ideas represent several different directions. Ideally, there would be some kind of feedback on which areas are of most interest to the agency.

Spatial features. 1) The prototype uses proximity-based reverse geocoding which not accurate and needs to be replaced with point-in-polygon search for better accuracy. 2) It should also support spatial filtering for search, and the creation of spatial filters from the current result set. 3) See also clustering below.

Visualization. The prototype tool only includes a handful of advanced visualizations. It should also include the more basic ones (e.g. histograms). There is also no map-based visualization, but there are pre-existing solutions that could be integrated to provide that.

Clustering. The group by function partitions objects into groups based on the value of some field. That's a very specific instance of the more general idea of grouping objects together that are similar in some way. Group by could be extended to include different types of similarity (exact or proximity) and along different dimensions such as location or time.

Imagery. The prototype does not handle any image or point cloud formats. Basic support for this type of data might be to convert them to multi-polygon features by color (e.g. colored maps with regions) or by binning (e.g. elevation value of LIDAR data). Another way to use this data might be to calculate summary statistics over given regions like average, min/max, variation.

Knowledge base. There now exists a number of large knowledge bases that contain a lot of information about people, places and things. This knowledge could be incorporated into the tool in several ways such as 1) associating facts to result objects as fields (e.g. population), 2) identifying field types (e.g. does a field contain peoples' names, restaurants, etc.), 3) clustering similarity could be based on concept similarity, 4) identifying topics of text documents.

Core functionality. The prototype includes only minimal functionality for demonstration. The core functionality needs to be extended with all the features found in languages like SQL such as renaming fields, data type conversion, arbitrary expressions etc.

Thank You

Thank you for sponsoring this challenge. I'm new to the area of geospatial processing and really enjoyed putting this solution together.