Plenario: An Open Data Discovery and Exploration Platform for Urban Science

Charlie Catlett[a,b] Tanu Malik[a,c] Brett Goldstein[a,b] Jonathan Giuffrida[b] Yetong Shao[a] Alessandro Panella[a] Derek Eder[3] Eric van Zanten[3] Robert Mitchum[1] Severin Thaler[c] Ian Foster[c]
[a] Urban Center for Computation and Data[1]
[b]Harris School of Public Policy[2]
[c]Department of Computer Science[2]
[1]Computation Institute of the University of Chicago and Argonne National Laboratory
[2]University of Chicago
[3]DataMade, LLC

Abstract

The past decade has seen the widespread release of open data concerning city services, conditions, and activities by government bodies and public institutions of all sizes. Hundreds of open data portals now host thousands of datasets of many different types. These new data sources represent enormous potential for improved understanding of urban dynamics and processes—and, ultimately, for more livable, efficient, and prosperous communities. However, those who seek to realize this potential quickly discover that discovering and applying those data relevant to any particular question can be extraordinarily difficult, due to decentralized storage, heterogeneous formats, and poor documentation. In this context, we introduce Plenario, a platform designed to automating time-consuming tasks associated with the discovery, exploration, and application of open city data—and, in so doing, reduce barriers to data use for researchers, policymakers, service providers, journalists, and members of the general public. Key innovations include a geospatial data warehouse that allows data from many sources to be registered into a common spatial and temporal frame; simple and intuitive interfaces that permit rapid discovery and exploration of data subsets pertaining to a particular area and time, regardless of type and source; easy export of such data subsets for further analysis; a user-configurable data ingest framework for automated importing and periodic updating of new datasets into the data warehouse; cloud hosting for elastic scaling and rapid creation of new Plenario instances; and an open source implementation to enable community contributions. We describe here the architecture and implementation of the Plenario platform, discuss lessons learned from its use by several communities, and outline plans for future work.

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

1 Plenario Context and Objectives: Open Data Discovery and Exploration

Over the past decade cities worldwide have adopted new policies for releasing municipal and goverment data, resulting in hundreds of open data portals and tens of thousands of datasets [1]. In the spirit of transparency, public access to city information, and collaboration with the community, cities such as Chicago, San Francisco, New York City, Barcelona, and Glasgow have launched online data portals containing datasets on a multitude of topics. Many of these portals include frequently updated data on crime, city contracts, business licenses, food safety inspections, service requests, traffic, energy usage, schools, and other data of importance to residents and researchers. Thus far, this data has been used by software developers to build new applications, by journalists to research stories and watchdog government activities, by researchers from many different disciplines (including sociology, education, economics, and behavioral sciences), and by policymakers to engage the public on new strategies and initiatives.

While this first wave of open data produced undeniable benefits, several issues have thus far prevented the movement from reaching its full potential. Most importantly, "open" does not always mean "usable." Finding relevant data in the many open data portals is largely a manual exercise requiring a high degree of experience and familiarity with portal technologies and their interfaces. Furthermore, most datasets are released in file formats and structures that make integration and analysis time-consuming even for skilled data analysts, and effectively out of reach to the general public. Even the most advanced portals release most datasets as massive spreadsheets or tables, putting the burden on users to extract, visualize, map, or combine data subsets of interest. Further, many of the information technologies and tools used to make data available were designed primarily to support the analysis of individual datasets rather than exploring relationships among datasets. These technical hurdles make asking even simple questions, such as "What datasets are available for the block on which I live?" or "What is the relationship between dataset A and dataset B?" immensely challenging. While data for answering such questions may have been released, in practice that data was often simply dumped without any descriptive metadata, in a wide range of formats, etc.—and is thus only intelligible to those possessing private knowledge.

This problem of combining datasets also exists within city government and has inspired novel solutions in the recent past. One such project, WindyGrid [2], was developed for internal use by the City of Chicago in anticipation of hosting the 2012 NATO Summit. WindyGrid organizes disparate datasets (both internal to the city and from public sources such as social networks) by their space and time coordinates using geospatial database technology. It thus allows city officials to gather multi-dimensional, real-time information about different areas of the city. This system was found to support much more informed and effective deployment and coordination of services, including emergency responses. After the summit, the city continued using WindyGrid, expanding its use by adding tools to analyze and improve city services.

In the same time period, the University of Chicago's Urban Center for Computation and Data (UrbanCCD) [3] organized the Urban Sciences Research Coordination Network (US-RCN) [4] to bring together scientists, policymakers, social service providers, and others to explore the use of open data for social science research. Disciplines represented in US-RCN range from sociology to economics; questions studied range from healthcare to education, crime, and employment. Interaction within this diverse community, along with lessons learned designing and using WindyGrid, revealed that a critical need for many data-enabled inquiries is to be able to easily find and access data about a particular place and for a particular window of time.

The common workflow required for such inquiries, shown in Figure 1(a), relies on the knowledge the investigator has about what datasets are available, and from what sources, as well as familiarity with the portal technologies and their internal search, refinement, and export functions. Additionally, the examination, refinement, and aligning and merging of datasets represents a significant cost in terms of labor and time given the diversity of spatial and temporal resolution and organization of data from multiple sources. The result is that effectively using open data requires both considerable knowledge and expertise in navigating and finding data as well as resources to evaluate and prepare the data.

The Plenario project began with a hypothesis that for open data is to have truly transformative impact, it

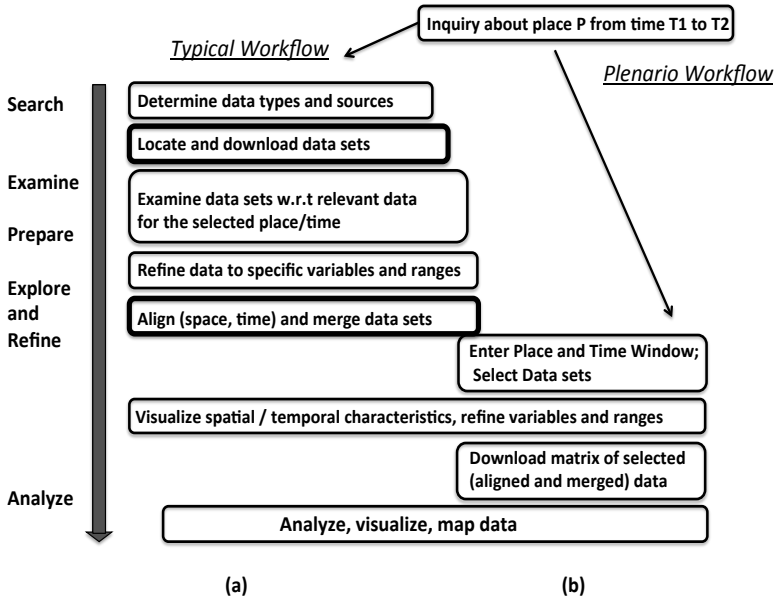

Figure 1: (a) Typical workflow for data discovery and exploration, involving many hours of data preparation and (b) the reformed, interactive space- and time-based exploration workflow using Plenario. Bold lined steps in (a) show steps that are particularly time-consuming, which Plenario eliminates by pre-integrating data from many sources.

must be accessible to non-data scientists, by individuals without a priori familiarity with the growing collection of open data portals (or their user interfaces), and it must be possible to explore the data without first investing weeks or months in data preparation. Plenario is a prototype platform developed to test this hypothesis by bringing many open datasets together, integrating them, and presenting a map-based interface for users to discover datasets relevant to a particular place over a period of time, and to examine the potential for interdependencies among those datasets.

1.1 Plenario: An Overview

Plenario exploits the fact that the vast majority of open data portals are implemented using one of two platforms, each with an API for accessing and downloading data—Socrata Open Data API (SODA) [5] and Comprehensive Knowledge Archive Network (CKAN) [6]. Each platform offers various internal search capabilities, visualization tools, and APIs for external application development. Yet at present there is no federation between these platforms or global search capabilities across portals implementing either of the platform. In part the lack of search capabilities reflects the fact that the data is highly diverse, from text to spreadsheets to shapefiles, and it is not clear how one might search such a collection of sources—keyword? Full text? Based on interactions with the US-RCN community and the experience with WindyGrid in the City of Chicago, Plenario is designed to support place and time inquiries, and consequently uses a map interface with time specifications to implement search. This user interface replaces the first two steps in typical workflows—the "Discover" phase shown in Figure 1.

Beyond the need for search, open data sources are diverse with respect to their spatial and temporal organization, resolution, units of measure, and other factors. Plenario imports datasets and integrates them into a single geospatial database, performing the alignment and merger of the datasets, eliminating the need for the user to do so, as shown in the "Explore and Refine" phase of Figure 1. Moreover, the Plenario workflow does not rely on the knowledge of the user to determine where relevant data might exist. In cases where the use may

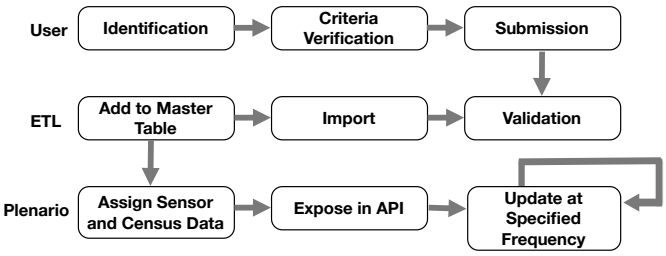

Figure 2: The path from identification of an appropriate dataset to making it available in the Plenario API. This process takes less than 24 hours for all but the largest datasets; we aim to improve this efficiency much further.

be aware of data that is not within Plenario, a request for data import is provided through a web form.

Figure 1(b) illustrates the Plenario workflow, with its new open data discovery capability and automation of several of the most costly steps in the data analysis workflow—notably the "Locate and download" and "Align and Merge" steps in Figure 1(a). Instead of searching for and combing through a multitude of potential open data sources and datasets to find data of interest for a particular location, the user specifies geographical boundaries and instantly receives all of the data available for that location (Figure 3). The labor-intensive work of combining and aligning the various spatial and temporal attributes of datasets is already done as part of the data import functions of Plenario, significantly shortening the path from question to discovery.

Plenario also provides basic data visualization, mapping, and time series creation to give users a snapshot of data before they decide to download it (Figure 4). When datasets are listed in response to a user query, each includes not only basic information and links to provenance and meta data, but a simple time series graph that provides the user with an indication as to the overall signal of the dataset, for instance whether it might provide relevant information for the particular temporal query. datasets can be selected for a map-based view showing spatial density of the data, and the user can modify the aggregation density anywhere from 100 meters to 1 kilometer. Finally, the user can refine the view of a selected dataset by specifying fields and values or ranges of interest. All of these tools enable the user to examine each dataset to determine its relevance to the research questions before exporting the integrated datasets of interest.

Furthermore, the platform helps avoid duplication of effort by providing a space for users to collaborate on cleaning data and exploring datasets. Every dataset only needs to be imported once but can be used by all (or all authorized users if the dataset is not open); similarly, the provenance of how data was cleaned is available for all users.

2 Plenario Architecture and Implementation

The Plenario software platform is open source and available on GitHub [7]. It is implemented using Amazon Web Services commercial cloud services. This facilitates replication in several important ways. First, governments and other organizations can readily create and potentially customize their own instance of Plenario, including both open and internal data. Second, organizations can provide open data for integration into an existing Plenario instance, such as the one operated by the University of Chicago at http://plenar.io. In either case, they can then use Plenario's analysis tools and API to power their own applications. The architecture easily allows data providers to choose which datasets are available to the public and which should remain "closed," available only for internal use for authorized users. The San Francisco Plenario instance, detailed in Section 4.2, is being used to explore functions for aggregating sensitive data prior to presenting to the end user.

Here we describe the Plenario architecture. We describe, in particular, the following features: (a) data import via an automated Extract-Transform-Load (ETL) builder, (b) integration using a geospatial database, and (c) special cases for common datasets such as weather and census data. In brief, an automated extract-transform-

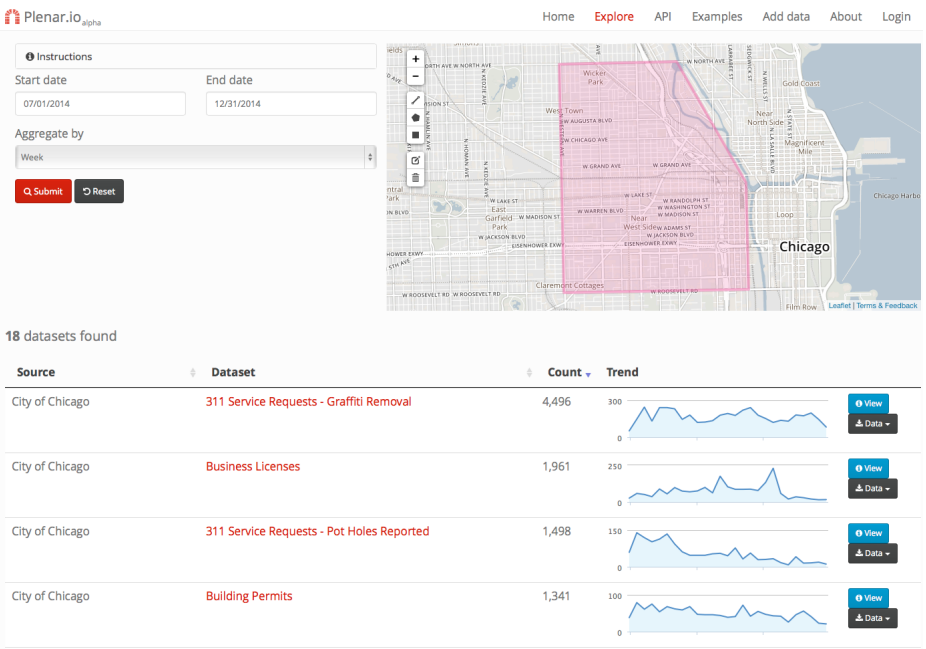

Figure 3: An example search using the Plenario portal. The search panel, at top, specifies the desired time period (the second half of 2014), aggregation (weekly), and spatial extent (the polygon). The results panel, truncated here to the first four of 18 matching data sources, includes not only basic metadata but also time series graphs as an indication of temporal dynamics.

load (ETL) builder imports each dataset and inserts the table into a PostgreSQL database. The ETL builder includes a specification for update frequency, so that Plenario updates the dataset at the same frequency as the source dataset is updated. Besides creating a table for each dataset, every record is represented as a pointer on a single "Master Table," so that all data is indexed on the same spatial and temporal database indices to improve performance (Figure 2). Additionally, the platform joins each observation to a set of commonly used datasets including sensor and place-based data. Finally, the datased is exposed in the API, making it accessible from the Plenario web portal and other clients.

2.1 Data Import: Automated ETL Builder

A dataset can be submitted to Plenario as a URL that points to a publicly available table in CSV format. This approach supports, for example, datasets on a Socrata or CKAN platform, as well as direct links to CSV files. Plenario's automated ETL builder scans the dataset, gets meta-information if available from the source platform, infers field types, and checks for common issues of data integrity. As Plenario currently focuses on spatio-temporal datasets, the user is then asked to identify which fields in the source data contain the required spatial information (location), temporal information (timestamp), and unique identifier, and how frequently the dataset is updated. (Future improvements to Plenario will remove one, two, or all three of these requirements: see discussion in Section 5.) The user can also add information regarding the dataset's provenance, description, and license if these are not automatically populated (as they are with Socrata datasets).

Following a basic check for URL stability and malware, an ETL worker process begins importing a local copy of the dataset as a new table in the PostgreSQL database. After import, Plenario inserts a row into the Master Table for every row in the new dataset, containing the dataset name and dataset-specific identifier (foreign key), the row identifier (primary key), and the spatial and temporal fields. The dataset is then made available via

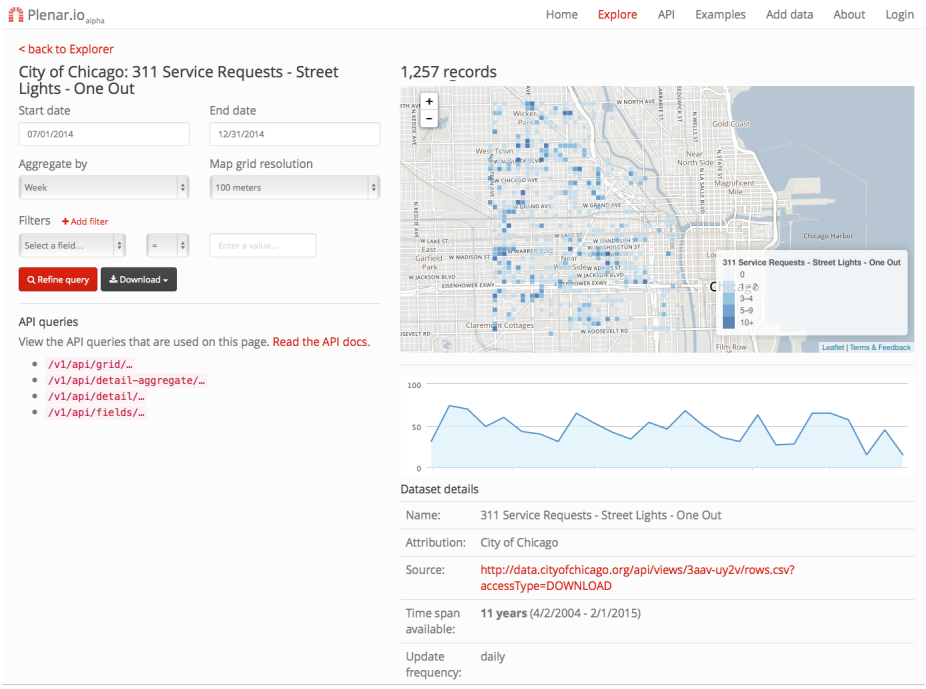

Figure 4: Plenario dataset view. Selected from the first results screen, this view allows the user to view the spatial distribution of a given dataset and provides links to the data and associated metadata. This screen also allows the user to change the temporal and spatial resolution of the query and to refine the dataset by selecting and specifying values or ranges for individual record values.

a RESTful API with endpoints for raw data, data aggregation by time and space, metadata, and weather-specific data (weather is one of several special case base datasets discussed in Section 2.3). Tasks are automatically scheduled to update the dataset according to the refresh frequency of the source dataset, using the unique identifier to avoid re-populating the entire table. Datasets can be imported and updated simultaneously using multiple ETL workers.

2.2 Core Database: Single Spatio-Temporal Index and PostgreSQL Schema

Plenario achieves the workflow optimizations discussed in Section 1 and illustrated in Figure 1 by organizing all records using common spatial and temporal indices in the Master Table (Figure 5). This method has several important implications.

First, data is automatically organized in an intuitive and coherent manner that can be easily searched and accessed by the user. In addition to API access, Plenario includes a portal interface (Figure 3) that allows users to search for datasets by drawing polygons or paths on a map and selecting start and end dates.

Second, data is organized, and can be searched for and accessed, without relying upon user knowledge of the existence of the data or its sources. Any point or polygon, and any time period, can be associated with data from multiple datasets, from multiple government agencies or organizations. This data can then be returned as a result of a search without the user needing to specify the data source. Thus, for example, a query for data points from Midtown Manhattan during June 2013 will return data from the City of New York, New York State, federal government, and numerous local or national organizations and surveys, including sources of which the user is unaware.

The third implication is that data for any arbitrary geography can be readily organized as a time series

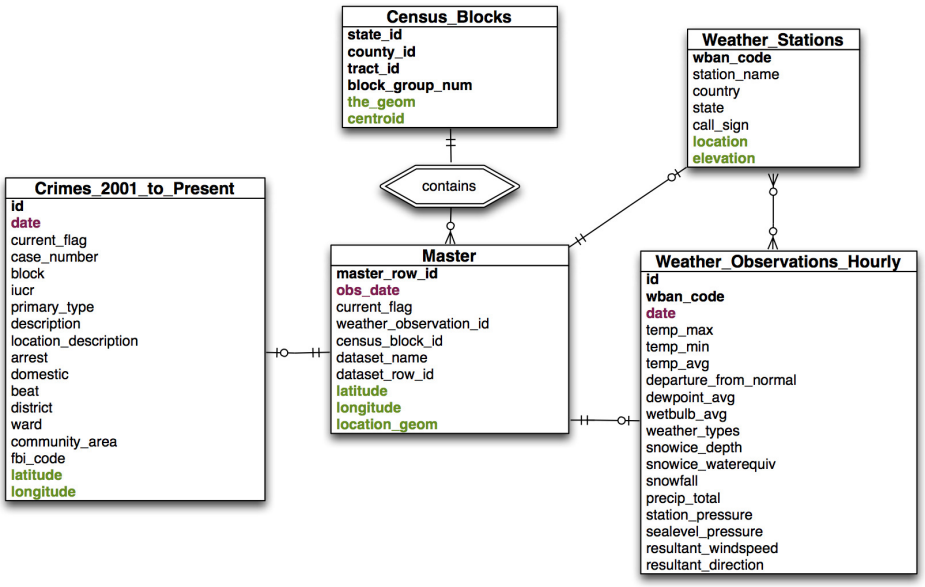

Figure 5: PostgreSQL Schema, showing how a sample dataset (Chicago crimes) feeds into the Master Table, which in turn links to spatial data like Census blocks and sensor data like weather observations through the `census_block_id` and `weather_observation_id` fields. There is one row in the Master Table for every row in a source dataset like Crimes, but a many-to-many relationship exists between the Master Table and Census blocks or weather observations. Note that the `Weather_Observations_Hourly` table (which contains no spatial information) is filtered through the `Weather_Stations` table (which contains no temporal information).

containing counts (or other variables) of observations in each contained dataset. Plenario enables one-click download of such time series matrices—instant snapshots of a geography over time—with any temporal resolution from hours to decades. Plenario thus eliminates the tedious work of data compilation and aggregation along identical temporal and spatial units and allows users to begin simple analyses immediately.

2.3 Special Cases: Commonly Used datasets

Plenario was optimized to support not only particular datasets related to a specific topic but to enable investigations in the context of widely used data sources, such as weather observations (with corresponding station position obtained from shapefiles) and aggregated census information. Once a dataset is imported and inserted into the Master Table, Plenario enriches it with other data relevant to the same geography, including sensor and location-specific data. Below we discuss *sensor data* (time series such as weather) and *local data* (relatively static data about the geography), which are also shown in Figure 5.

Sensor data, which records one or more variables at regular intervals in fixed locations, usually along a network with high coverage (such as weather stations), is important both for tracking environmental variables over time and for enhancing human-generated data (such as noise complaints) with objective recordings from the same time and location (such as noise levels).

Weather data in particular is important to many questions about place, from healthcare studies to traffic or flooding analysis. We thus include NOAA hourly and daily weather weather station data as an integral element of the Master Table scheme. To date, we have loaded into Plenario all U.S. hourly and daily weather station data since 2011. We also assign to every record in the Master Table the identity of the weather station that is closest to it; thus, we can respond to any query requesting weather information about a record by retrieving

the observation from that "nearest" weather station for the time closest to the record's timestamp. The process is efficient because all weather data is stored in only one place, and because the closest weather stations are pre-calculated when a dataset is imported.

This integration of weather station data provides an initial example of how sensor data adds to the open data landscape. Further plans for sensor data include incorporating data from the Array of Things [8] project in Chicago, which will report measures such as air pollution and noise at a much higher resolution, both temporally (every 30–60 seconds) and spatially (sensors will be deployed throughout Chicago, with densities ranging from one per square block to one per square kilometer). This data source will further illustrate the value of sensor data to municipal open datasets, enabling investigations such as the spatial and temporal characteristics of air quality in the context of vehicle flow and weather, or the interrelationships between hyperlocal weather and crime.

Local data refers to data aggregated at a regional (not individual) level, containing variables that are relatively static over time, such as demographic data and local economic data. The Chicago instance of Plenario incorporates a prototypical example of local data, which is data from the United States Census: every row in the Master Table is coded with its FIPS Census block identifier, which allows for easy enhancement with open data from other sources tied to that Census block, Census tract, county, state, etc., all of which can be determined from the Census block identifier.

2.4 Components and AWS Implementation

Plenario is built entirely from open source tools and is released under the MIT license. All code is in a GitHub repository [7], making the platform easy to fork and clone. By default, all source datasets remain under their original license; most are released under the MIT license or are unlicensed.

The data backend of Plenario is built as a PostgreSQL relational database, with PostGIS geospatial extension. SQL Alchemy [9] is used as the object relational mapper with the GeoAlchemy 2 extension. The web application with API was developed using Flask [10], with mapping capabilities provided by Leaflet [11] and Open Street Map [12]. The ETL process uses Celery [13] for logging, and Redis [14] is available for caching support when high loads are anticipated.

We host Plenario on Amazon Web Services (AWS)'s Elastic Cloud Compute (EC2) infrastructure. We currently use four virtual servers within a Virtual Private Cloud (VPC): one web server, one database server, one ETL worker server, and one gateway server. Due to its elastic nature, the EC2 server resources can be upgraded in real time as traffic load and data footprint increase. For larger deployments, the database server can be sharded and replicated for more capacity using the AWS Relational Database Service (RDS). Amazon Machine Images (AMI) can be used to snapshot the various server configurations for easy re-deployability.

For data integrity and redundancy, every raw dataset processed by the Plenario ETL worker is saved as a snapshot and stored on Amazon's Simple Storage Service (S3). This approach allows for data integrity checks, ETL fault tolerance, and history tracking on every dataset Plenario ingests.

Plenario can be used in several ways: a user can fork the GitHub code to develop a separate project, copy the entire project via a machine image on AWS, or feed data into the web portal supported by the University of Chicago at http://plenar.io. Each of these modalities of use has been seen since Plenario's alpha launch in September 2014, including a repurposing of the core API to power the City of San Francisco's Sustainable Systems Framework initiative, as detailed below.

The web portal interface described above is in fact an application that accesses a Plenario instance running on AWS, via the Plenario API. This modular approach enables other front-end frameworks to be built to use the API, ranging from custom mobile and web applications (of which http://plenar.io is an example) to a complex analytics system such as WindyGrid, which uses commercial mapping and user interface software such as ESRI.

3 Evaluating Plenario's Architecture

The key architectural feature of the Plenario system is its spatio-temporal database, which is hosted on the cloud. Users can upload new datasets into the hosted system, and query datasets for discovery and exploration using a RESTful API. The system, given its need to support open-data for all, sets no limits, both in terms of the number of uploads and the size of the upload. The open upload feature raises scalability concerns, especially when performing exploratory spatial querying, which depending upon spatial-extents itself can be I/O-intensive. A complementary, but related concern to scalability is the cost of hosting an open-data service such as Plenario on the cloud. Given the volume of anticipated data, hosting Plenario on high-end instances seems natural but if most uploads are small and queries retrieve small spatial regions then high-end instances do not provide sufficient cost-benefit advantage.

To evaluate our choices, we performed a thorough performance evaluation to evaluate our database choice, and to determine which type of cloud instance provides the best cost-benefit ratio. To conduct the experiments, we developed a benchmark based on available Plenario queries and logs. We evaluated the benchmark workload with open-source relational, NoSQL and array database systems to determined which database system exhibits highest performance for concurrent uploads and query. Finally, we instantiated Plenario's relational database on different cloud-instances to determine the best cost-benefit ratio in terms of transactions per second per dollar spent.

3.1 The Plenario Benchmark

We developed a benchmark specific to Plenario because unlike other geospatial benchmarks [20], Plenario has no separate database instantiation and query generation phase; database tables and queries are determined by an incoming user-workload leading to an openly-writable spatial storage system. To simulate the discovery and exploration phases in Plenario, we simulated a closed loop Markov-chain synthetic workload generator in Python. The Markov-chain consists of two states, `add_data` and `query`, with the query state having five sub-states, `initial_query`, `expand_time`, `narrow_time`, `expand_geography`, and `narrow_geography`. The current state is updated after each query and the system chooses between `query` with probability 0.85 and `add_data` with probability 0.15. Within the query phase, expansion and reduction of spatial attributes occurs with equal probability, and spatial attributes are chosen over time attributes by 0.6 to 0.4. We chose these probabilities based on user query logs and estimated patterns.

As part of a session, users often change spatial and time attributes by either expanding them or narrowing them from an initial specification. Expansion and reduction of the time attribute is simply achieved by changing both the starting and ending dates by an equal amount such that the new date range is a factor k times of that of the old date range. We choose k so as to reflect a daily, weekly, or monthly search pattern. To expand and narrow spatial boundaries such that the new query is again a convex polygon, we follow a simple single parametric method. (A convex polygon is not a requirement of the Plenario API. However logs show that all users to date have indeed specified convex polygons.) For expansion, given an N-sided polygon with coordinates $(x_1, y_1), \ldots, (x_n, y_n)$,

1. Find the smallest rectangle that exactly covers all the vertices of the polygon by finding the maximum and minimum of the coordinates of all the vertices;
2. Fix the center of the rectangle and expand it outwards by an expansion factor of k;
3. Divide the rectangle into four equal sub-rectangles: top-left, top-right, bottom-left and bottom-right;
4. For each vertex of the polygon, identify the sub-rectangle it is located in, and create a new point in the region of the box further from the center than the vertex is;
5. The newly simulated points form the vertices of our new polygon.

The same algorithm works for narrowing the polygon except instead of expanding by a factor k, we narrow by k. The algorithm is guaranteed to produce a convex polygon since all the new points are simulated randomly in four different sub-rectangles, so it's impossible that they lie on a line. Finally, in the `add_data` state, data is generated for a given spatial with a skew ranging from 0.1–0.3 and with the size of the data chosen from a Zipf distribution varying from a few kilobytes to a few gigabytes.

3.2 Evaluation

To evaluate the query and upload workload from the Markov-based generator, we chose three databases: (a) PostGIS, (b) Accumulo, and (c) SciDB. Plenario is currently hosted on PostGIS, which is a traditional RDBMS with spatial index support. RDBMs can encounter scaling problems when dealing with Terabytes of data or thousands of tables. We examine Accumulo and SciDB as alternative database systems that support scaling out onto multiple nodes. Accumulo is a key-value store, and SciDB a multi-dimensional array store.

To correctly ingest two-dimensional geospatial data into the one-dimensional Accumulo key-value store, we create a geohash [18] of each latitude-longitude pair, hence mapping each two-dimensional coordinate into a scalar value. Geohashes have the property that points near to each in space other have geohashes with the same prefix, which improves locality significantly when points are stored by order of geohash. Besides, geohash can obtain good precision in a small scale of bit length. While there are other schemes of linearizing spatial data that are more precise than geohash in terms of maintaining the locality of spatial points, geohash is simple to implement and is used significantly in key-value systems for spatial indexing. In SciDB we consider latitude and longitude as two different dimensions of a grid that is divided into cells. Since SciDB supports only integer dimensions and Plenario's data has spatial coordinates of arbitrary precision, to consider latitude and longitude as dimensions, we perform linear scaling of the spatial coordinates, and round off the resulting values. Using these dimensions, the size of the grid is decided based on the cross-product of the number of latitude and longitude points, respectively, in the data. As not every combination of a latitude and longitude cell may have a corresponding data point in the dataset, the cross-product results in a two-dimensional sparse array in SciDB.

The databases are set up on an AWS T2.Extra Large instance on Ubuntu 14.04. We used Accumulo version 1.7.0 with a single master and tablet server. The accompanying Hadoop version is 2.6. configured for one name node, data node, and secondary node respectively. Hadoop uses Zookeeper v3.4.6 with one server. We used version 14.12 of SciDB with four instances one SciDB master and one worker. With this setup, we use our experiments to answer four important questions regarding Plenario's architecture: (1) Which data model (grid, geohashed key-value, or R-tree-based relational) provides the best performance for retrieving the relevant datasets from the Master table, given a spatial polygon? (2) Given that Plenario accepts CSV data, how do different database systems compare in terms of ingest costs? (3) When different users upload and query datasets concurrently, how do systems compare in terms of the transaction rates that they supported? (4) What is the economic cost of hosting Plenario on different EC2 instances?

Figure 6 compares the query performance on the three different databases. The query workload consists of 4–6 sided spatial polygons. An initial specification of points is chosen randomly. The query may expand or narrow as described in Section 3.1. We measure the average response time of the workload against the same data loaded in each of the database system. In PostGIS, each spatial query uses the spatial index to retrieve the objects; in Accumulo, given a spatial polygon, we calculate the minimum and maximum geohashs across all vertices. We use these geohash values as the range to query to perform a batch scan and determine if the scanned point is in the polygon to get an exact answer. In SciDB there is no index, so a two-dimensional polygon query is approximated to the nearest grid rectangle to obtain all the points from that grid and then each point is checked for containment. As we see, we get best performance from PostGIS for querying but Accumulo is not far off. Since we are only dealing with point data with arbitrary precision, so far geohashing is comparable to R-tree.

We compare the three databases for ingesting data (Table 1). Note that Plenario intends to expand its Chicago instance to thousands of datasets so fast ingest is important. While we have already described a formal ETL

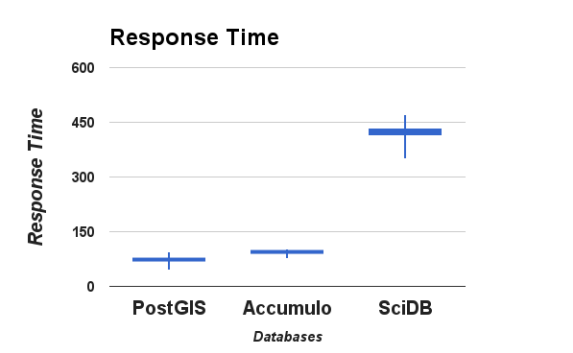

Figure 6: Response Time Comparison

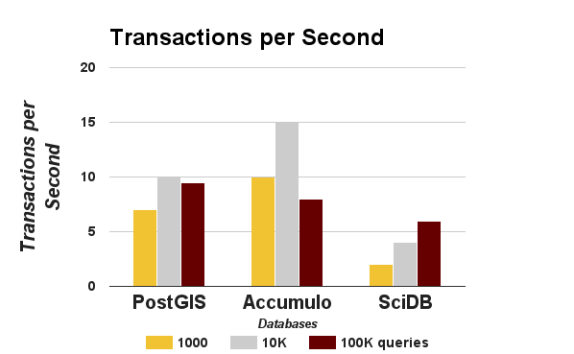

Figure 7: Throughput Comparison

Table 1: Ingest Time (in seconds)

Data Size	PostGIS	Accumulo	SciDB
1	120	65	300
10	800	630	NA
20	1800	1260	NA

Table 2: Cost of hosting Plenario

Machine Size	Transactions per $
Small	9.8
Large	13.6
XL-HM	12.8
2XL-HM	12.3

process for ingesting public datasets, in this experiment we compare in how must time cleaned CSV datasets are ingested. In Accumulo ingest implies using geohashing to first convert data and then using batch mode to ingest the newly formatted data into an Accumulo table. Accumulo loads data at a constant rate into this table, which is approximately at 3700 entries/second or 200 KB/second. In SciDB, the CSV file is first ingested as a single dimensional array and then redimensioned into a two-dimensional grid. In SciDB, while ingesting into single dimensional array is straightforward, re-dimensioning is a memory intensive operation which often fails or is extremely slow due to thrashing for fairly modest size of data such as 8GB. In PostGIS, data is simply ingested into the relational table using bulk copy.

Figure 7 compares the concurrent cost of the querying and update in terms of transactions per second. Given the fast uploads Accumulo is able to achieve higher number of transactions per second, even though it is not running the queries against a spatial index. This is indeed surprising and exciting result, which demonstrates the benefits of using the Accumulo instance especially as thousands of datasets are added to Plenario and the system needs to scale out. In RDBMSs, R-trees (and other spatial indices) are external to the data and must be updated and computed when new features are added. This is computationally intensive and is affecting performance adversely as data size increases.

Finally, in Table 2 we compare the cost of hosting Plenario on different size instances AWS. Given that Chicago Plenario instance is 100GB in size, and most queries are fetching small datasets of a few kibobytes to maximum of 1GB, currently even a large instance is sufficient to host Plenario in terms of the cost benefit ratio. While currently sustainable, this result must be examined in lieu of projected growth and other similar studies [19]

4 Plenario Use Cases and Early Lessons Learned

We have discussed a number of advantages the Plenario project was designed to provide for researchers, government employees, developers, journalists, and citizen users. Here we discuss several specific use cases where Plenario is being used today to support social and economic science research (The Chicago Plenario instance)

and for community engagement on urban sustainability, resilience, and livability goals (The San Francisco Plenario instance).

4.1 Supporting Research: The Chicago Instance

The Chicago instance of Plenario, active at http://plenar.io, has been implemented with data from multiple cities, and with a focus in particular on Chicago, to support research into urban science and computational approaches to public policy, identified through the US-RCN, by supporting rapid data discovery, helping researchers, journalists, and residents identify datasets that interest them regardless of the original source. For example, Goldstein leads a team investigating the interrelationship between local weather and violent crime, which involves all of the base sensor and local data described in Section 2.3 as well as urban crime data, 311 service call data, and other datasets. Without Plenario, this research would begin with an attempt to find data of interest from many independent sources, many of which the user might not know exist. For Chicago these include open data portals operated by the City of Chicago, Cook County, the State of Illinois, the federal government, and data from hundreds of separate government departments such as the Illinois Department of Transportation or the Chicago Department of Public Health. In addition to these sources, relevant data has been imported into http://plenar.io from the National Oceanic and Atmospheric Administration (NOAA) and the U.S. Census Bureau.

As the Chicago instance has grown from dozens of data sets to over 150, we have found that Plenario's automatic data set summary feature has led users to identify a range of previously undetected data quality problems. For example, a single error in the date field is immediately apparent when a summary suggests that Plenario contains data from prehistoric times, or far into the future. We intend to incorporate consistency checks to flag obvious errors of this kind (such as impossible dates), but we note that not all errors are readily flagged by algorithms. Errors and holes in data are inevitable. Thus it will be important both to work with data providers to fix obvious errors and to provide Plenario users with mechanisms to discover and flag errors.

4.2 Enabling Community-Driven Urban Sustainability, Resilience, and Livability: The San Francisco Instance

The Plenario platform has also been deployed experimentally as part of the City of San Francisco's Sustainable Development initiative [15]. This project has motivated important Plenario enhancements, including support for additional data types such as geographical information in the form of ESRI shapefiles. It has also spurred the development of new features to enable the use of Plenario as a community "dashboard," whereby the visual interface is the primary use. (In contrast, the Chicago Instance is mostly used to refine and export data for advanced data analytics.) Several enhancements driven by the San Francisco implementation have already been incorporated into the core Plenario code base. Others will be incorporated after further evaluation in the San Francisco instance.

The San Francisco Plenario instance contains datasets pertaining to a wide variety of sustainability indices, ranging from community structure accessibility to green space, canopy cover, water consumption, and energy use. Having the ability to instantly access datasets of this kind by spatial-temporal queries empowers institutions and communities to assess the status quo and plan future efforts in sustainable development. Thus, for example, the framework is to be used in the sustainable development of the South of Market (SoMa) ecodistrict.

The data needed for the applications that the San Francisco Instance is designed to support are highly heterogeneous in both content and form. For example, quantifying access to green spaces—the vicinity of parkland to residents—requires analysis of geographic information regarding the location and shape of each park, which cannot be treated simply as a point in space. Similarly, a community center is an entity that exists over a certain time span, in contrast to much place-based urban data such as crime or inspections, which are "events" that each occur at a specific instant. To incorporate these and other types of data, Plenario's database schema was extended and more ETL functions were added. Moreover, new types of queries were developed and implemented

efficiently, aimed at questions of the type "What is the average distance for residents of a given area to the closest farmer's market, at any point in time and in a given range?"

In the San Francisco Plenario instance approaches to support a mix of open and sensitive data are being explored. As with the Census data in the Chicago instance, some of the data made available from City of San Francisco is not public and thus must be carefully aggregated to protect privacy. Utilities data is one such kind of available data in which privacy must be protected. One algorithm to protect privacy that is commonly used for utilities datasets is the "15/15 rule," which requires that no aggregation sample may contain less than 15 data points, and any point in any aggregation sample cannot represent more than 15% of the measure for that sample (The "15/15 Rule" was adopted by the California Public Utilities Commission in Decision D.97-10-031.). The methodology being explored in the San Francisco project is to host raw data securely in the Plenario instance, and then to implement the query- and data-specific privacy-preserving aggregations as a function of the particular search, view, and/or data export process.

5 Lessons, Challenges, and Opportunities

With the two large-scale Plenario instances described above, we have identified a number of challenge areas that will be essential to address in order to move Plenario from an alpha platform to a fully supported and sustainable resource. We group these as issues related to data, scaling, and architecture.

5.1 Data Issues

Data is often collected for different purposes and thus in different ways across juridisctions. Even datasets with similar purposes, such as 311 service requests or food safety inspection reports, can rarely be merged across jurisdictions, effectively limiting research to a focus on one particular city rather than incorporating and studying multiple cities at once. These barriers can exist at the metadata level (different variables recorded), in the resolution of the data (spatial and temporal), and even at the level of individual data points and fields (semantics and ontology). For example, a crime classified as "assault" in New York City crime data would be classified as a "battery" in crime data from Chicago, which may mislead a researcher attempting to compare violent crime in the two cities or compile a large dataset of crime in the United States.

We have also encountered the common challenge of poor data quality and documentation. Because all data in Plenario ultimately refers to a source dataset hosted by a municipality, the remedy is limited to either cleaning the data upon insertion into Plenario or providing feedback to the data providers. Data cleaning at insertion would accelerate cleaning in comparison to relying on data providers, but would also require that the platform understand in each case what is "correct." Ultimately this balance might be encoded into the ETL process in a similar fashion to the update frequency. Finally, the lack of unique identifiers on many datasets also means that updating datasets requires a full refresh of the entire dataset, which increases load but more importantly introduces data consistency issues that will impact the applications using the datasets, particularly those aimed at real-time capabilities.

5.2 Scaling Issues

Plenario was designed with consideration regarding scale, given the enormity of the open data landscape and the rapid pace with which open datasets are being released. Nevertheless, as the experiments show the Master Table approach introduces scaling challenges, particularly as the table grows to billions of rows. The team has explored a variety of approaches including partitioning the table along the temporal index, with mixed results. In particular, the number of NOAA's hourly observations for all 2,200+ weather stations since 1997 in the United States was deemed too large to import in its entirety, while maintaining a reliably responsive API. To

work around this limitation, only observations from weather stations within a certain radius of each dataset's bounding box were added.

The sensor data also contributes to scaling challenges. Though the closest weather station to every record is identified upon insertion into the Master Table, the platform executes the join between the Master Table and the weather table at the time of request rather than as part of the insertion process. This has significant impact on query performance but the alternative would exacerbate scaling issues with the Master Table by making it extremely wide. Furthermore, sensor data needs to be spatially smoothed to avoid sharp boundaries in the data such as when two neighboring weather stations record significantly different values for a given variable. To reduce computational load, sensor data is organized spatially using a Voronoi diagram [16] without spatial smoothing.

5.3 Architecture and Data Semantics Issues

Plenario's original purpose as a platform for spatio-temporal data discovery and exploration brings into question what variables count as "space" and "time." For example, should 311 data reflect the location of the caller or the location of the problem reported? How should the location of non-spatial crimes, like fraud or online crimes, be reported? And how should Plenario represent records missing a spatial or temporal value? How, too, could unstructured data be supported in Plenario—especially when the location and datestamp of such data are uncertain?

We have also encountered challenges with respect to how to treat data that lacks resolution in spatial and temporal data. For instance, how do we present city budget data that covers an entire city for the period of one year—and make this data discoverable in typical user searches? Should a query across multiple years return multiple city budgets, ones wholly contained in the temporal arguments, or none at all? How should shapes like parks, streets, and parcel lots be dated? Some of these challenges are being highlighted in the San Francisco Plenario instance, as discussed earlier.

Ultimately these challenges suggest exploration into the optimal approach to support the integration of spatial/temporal data with data that is primarily "entity" based. In some cases, such as with census data, spatial and temporal mapping can be done in concert with data aggregation as is necessary for privacy protection. In other cases, particularly with organizations whose data includes internal private data about businesses and individuals, such mapping is less straightforward. Plenario currently supports questions such as "where were the automobile accidents in mid-town Manhattan during heavy rainstorms in 2014?" but is not organized in order to refine this query to show only those accidents involving cars greater than 10 years old, or male drivers aged 18-24.

Finally, Plenario is currently designed as a portal for open data, which is only a subset of data useful for urban science and research, for policy development, or for many areas of improved urban operations. There are known solutions to challenges to multiple levels of authorization, and it will be important to integrate these into the platform. The San Francisco Plenario instance supports sensitive data by aggregating at the time of query, presenting the aggregated data to the end user. The Chicago Plenario instance uses pre-aggregated census data, eliminating the need to aggregate at query time. While this improves query performance and reduces the sensitivity of the data stored in Plenario, it also requires that the aggregation algorithm is defined a priori, where different aggregation schemes may be more or less optimal for different types of inquiry.

6 Conclusions and a Plenario Roadmap

The Plenario team has begun to develop a 12–18 month roadmap based on input from early users. A rigorous set of performance scaling tests is being developed to explore the architecture issues noted above; the results of these tests may lead us to revisit various design decisions, ranging from the underlying database to the Master Table. Several features requested by researchers are under consideration for this roadmap, including automated

time series analysis to identify correlations between datasets: for instance, identifying subsets of 311 data that are lagged by violent crime in various neighborhoods of a city.

Of particular interest to many place-based investigations is the identification of urban areas that function as units. Traditional boundaries such as neighborhoods or districts often do not reflect the underlying social or economic structure, in part because many such boundaries were drawn generations in the past and/or through political processes. The rapidly expanding variety of data being integrated into Plenario is resulting in increased opportunity to understand what differentiates one neighborhood from another and to use spatial units defined by current data, not solely by a 20^{th} century surveyor's pen. Concurrently, support for place-based research will require more powerful tools for specifying spatial aggregation of data (where Plenario has already provided flexibility in temporal aggregation), necessary to address the Modifiable Area Unit Problem [17]: that is, the fact that the results of spatial analysis are often highly dependent on the spatial units used.

Today's open data landscape largely resembles the Internet of the 1980s when data was shared through anonymous file transfer servers, which were useful only to those with inside knowledge of their locations and contents. The advent of HTTP and web browsers led to today's powerful search and integration capabilities, including those that Plenario uses to import data. An underlying objective of the Plenario project is to contribute to these benefits extending to open data.

The first step toward this vision has been to implement the Plenario platform as a means to reduce or eliminate many of the challenges of working with open data, beginning with discovery, exploration, and integration across many data sources. Addressing these challenges provides increased incentives for governments to release data, reducing the need to develop their own custom data portals and providing the basic tools to start extracting insight and return on investment from their data. By building and encouraging a collaborative open data ecosystem at every stage, from identifying datasets to building third-party tools, Plenario helps push the full potential of this movement closer to realization.

Acknowledgments

We thank Mengyu Zhang for help with experiments, and anonymous reviewers for comments on the paper.
The Plenario project is funded by the John D. and Catherine T. MacArthur Foundation and the National Science Foundation via an NSF Early-Concept Grant for Exploratory Research (EAGER) for software development (award number 1348865), while the interaction capabilities were driven by the Urban Sciences Research Coordination Network, created with an NSF Building Community and Capacity for Data-Intensive Research in the Social, Behavioral, and Economic Sciences and in Education and Human Resources (BCC-SBE/EHR) award.

References

[1] Maksimovic, M.D., Veljkovic, N.Z., and Stoimenov, L.V., "Platforms for open government data," Telecommunications Forum (TELFOR), 2011 19th, vol., no., pp.1234,1237, 22-24 Nov. 2011. doi: 10.1109/TELFOR.2011.6143774

[2] "Chicago's WindyGrid: Taking Situational Awareness to a New Level." `http://datasmart.ash.harvard.edu/news/article/chicagos-windygrid-taking-situational-awareness-to-a-new-level-259` [Accessed July 7, 2015]

[3] The Urban Center for Computation and Data, at the Computation Institute of the University of Chicago and Argonne National laboratory. `http://www.urbanccd.org` [Accessed July 7, 2015]

[4] NSF 1244749, "BCC-SBE: An Urban Sciences Research Coordination Network for Data-Driven Urban Design and Analysis. PI Catlett, C., University of Chicago. 2012-2015.

[5] `http://www.socrata.com/` [Accessed July 7, 2015]

[6] http://ckan.org/ [Accessed July 7, 2015]

[7] https://github.com/UrbanCCD-UChicago/plenario [Accessed July 7, 2015]

[8] Moser, W, "What Chicago's 'Array of Things' Will Actually Do," Chicago Magazine, January 27, 2014. See also http://ArrayofThings.github.io [Accessed July 7, 2015]

[9] http://www.sqlalchemy.org/ [Accessed July 7, 2015]

[10] http://flask.pocoo.org/ [Accessed July 7, 2015]

[11] http://leafletjs.com/ [Accessed July 7, 2015]

[12] http://www.openstreetmap.org/about [Accessed July 7, 2015]

[13] http://www.celeryproject.org/ [Accessed July 7, 2015]

[14] http://redis.io/ [Accessed July 7, 2015]

[15] "The Sustainable Development Program." http://www.sf-planning.org/index.aspx?page=3051 [Accessed July 7, 2015]

[16] Voronoi, G., Nouvelles applications des paramétres continus á la théorie des formes quadratiques. Deuxième mémoiure: recherches sur les parallèloedes primitifs, J. reine angew. Math. 134, 198-287 (1908)

[17] Wong, D., "The modifiable areal unit problem (MAUP)", In Fotheringham, A Stewart; Rogerson, Peter. *The SAGE handbook of spatial analysis*. pp. 105–124 (2009)

[18] "Geohash," https://en.wikipedia.org/wiki/Geohash [Accessed July 7, 2015]

[19] Malik, T., Chard, K., and Foster, I., "Benchmarking cloud-based tagging services". In *In IEEE 30th International Conference of Data Engineering Workshops (ICDEW)*, pp. 231-238 (2014)

[20] Ray, S., Simion, B., and Brown, A. D., "Jackpine: A benchmark to evaluate spatial database performance". In IEEE 27th International Conference on Data Engineering (ICDE), pp. 1139-1150, IEEE. (2011).