Poster: Data Hub Architecture for Smart Cities

Jason Koh[1*], Sandeep Sandha[2*], Bharathan Balaji[2], Daniel Crawl[3], Ilkay Altintas[3], Rajesh Gupta[1], Mani Srivastava[2] *

[1]University of California San Diego, [2]University of California Los Angeles, [3]San Diego Supercomputer Center

{jbkoh,rgupta}@ucsd.edu,{sandha,bbalaji,mbs}@cs.ucla.edu,{crawl,altintas}@sdsc.edu

ABSTRACT

Today large amount of data is generated by cities. Many of the datasets are openly available and are contributed by different sectors, government bodies and institutions. The new data can affect our understanding of the issues faced by cities and can support evidence based policies. However usage of data is limited due to difficulty in assimilating data from different sources. Open datasets often lack uniform structure which limits its analysis using traditional database systems. In this paper we present *Citadel*, a data hub for cities. Citadel's goal is to support end to end knowledge discovery cyber-infrastructure for effective analysis and policy support. Citadel is designed to ingest large amount of heterogeneous data and supports multiple use cases by encouraging data sharing in cities. Our poster presents the proposed features, architecture, implementation details and initial results.

CCS CONCEPTS

• **Information systems** → **Spatial-temporal systems**; **Decision support systems**; • **Security and privacy** → *Access control*;

KEYWORDS

smart cities, data hub, big data, spatio-temporal

ACM Reference format:

Jason Koh[1*], Sandeep Sandha[2*], Bharathan Balaji[2], Daniel Crawl[3], Ilkay Altintas[3], Rajesh Gupta[1], Mani Srivastava[2]. 2017. Poster: Data Hub Architecture for Smart Cities. In *Proceedings of SenSys '17, Delft, Netherlands, November 6–8, 2017,* 2 pages.
https://doi.org/10.1145/3131672.3137001

1 INTRODUCTION

Urban sectors, where 54% of the world's population lives [1], consist of many different systems such as transport, energy, safety, and water systems. Each of the system generates huge amount of sensor data and have the potential to produce meaningful knowledge and applications for the public. For example, weather, safety and traffic information can be used to find an optimal path for wild fire evacuation. Building occupancy and traffic data can help with scheduling of electric vehicle charging stations.

However, the various datasets are underexploited due to several reasons. First, each dataset has a unique interface and their data model is not interoperable with other datasets. Some datasets are provided as CSV files and others with REST API. Same information can be represented in multiple ways: temperature can be written as *Temp* or *T*, it may have different units, and different resolution of data. Secondly, users need infrastructure to ingest high frequency data and tools to clean the data, query across datasets and create new data using analytics. Lastly, city datasets may contain sensitive information and hence need protection mechanisms to ensure access to authorized users, compliance with data use agreements and maintain provenance of data flows across different parties.

There are several existing works providing a data service in urban sectors prior to Citadel. Plenario[6] provides a common API for spatio-temporal data for heterogeneous datasets but it lacks access control and rich analytics support. Datahub provides a version control system for datasets but does not provide APIs for real-time interaction [5]. New York City has been dedicated towards sharing data publicly [2]. It has abundant types of datasets but their system does not consider security features as the shared datasets are public.

We design Citadel as a unified but flexible real-time data hub for heterogeneous urban sectors while providing both framework for computational workloads and secured access control. Data publishers can push their data to Citadel and users can retrieve/query proper data in real time. Our poster will present Citadel architecture, key features and initial performance results.

2 CITADEL: DESIGN AND IMPLEMENTATION

2.1 Key Features

2.1.1 Data Model. It is vital to have a common understanding of data for users to store and query data. Datasets produced by different actors are inherently heterogeneous. Users cannot query and process the data without a common data model. Thus, we normalize the different data models into our system-agnostic schema, and then query, store and optimize over the schema. Data crawlers normalize the metadata of target data and store them into Citadel.

We model a dataset as a collection of data streams, where each data stream is an array of (latitude, longitude, time, value) tuple. It thus captures dynamic spatio-temporal data generated across a city. Each data stream's contextual information such as its owner (e.g., a bus company) or the source of data (e.g., a bus) are represented as metadata. We currently model metadata as tag-value pairs per stream. Going forward, we will adopt a graph model to describe relationships between entities such as a bus, stops and sensors.

2.1.2 Analytics. Supporting analytics across multiple datasets is one of the goals of Citadel. We consider data streams generated from

*Sandeep Sandha and Jason Koh contributed equally as first authors.

SenSys '17, November 6–8, 2017, Delft, Netherlands

ACM ISBN 978-1-4503-5459-2/17/11…$15.00
https://doi.org/10.1145/3131672.3137001

analysis of different data streams as a *virtual sensor*. For example, temperature data stream can be used to define a new data stream with a different spatial resolution both of which can be considered as a separate virtual sensors. Virtual sensors can be computed both using APIs as well as using real-time stream processing. For real-time, each virtual sensor computation will be registered by the user in Citadel and executed as a spark streaming job. In order to have a modular design, we use Kafka based pub-sub model for data exchange between different spark streaming jobs. Users can also subscribe to Kafka and get real-time updates. We will also explore use of trusted computing methods in future for users who would like to deploy proprietary compute in Citadel.

2.1.3 Access Control. City has multiple sectors generating and consuming data from each other. In this scenario, access control is a crucial component of a data hub as datasets may need to be selectively exposed to users or groups. For example, let us consider the 911 call data of police department, which it would like to share with city hospitals to help them predict the number of ambulances required in Los Angeles. The police department would like to provide access to data only belonging to Los Angeles region in the past one month and hide sensitive data streams as well as share data at a resolution of zip-code and per hourly basis. In order to implement such access control, we will exploit the concept of virtual sensor and rich metadata model where every data stream will have associated access control information. We propose a separate Spark job that monitors changes to the metadata to maintain and update access related information of data streams in real-time. In our previous example, the police department may define a virtual sensor with desired spatial and temporal resolution over the Los Angeles data of past one month and provide the access to multiple hospitals.

2.1.4 Provenance. Provenance keeps track of who created what data and who in turn reads this data to produce more data. It is essential when we want to assess spread of sensitive information, analyze error propagation or evaluate the usability of a dataset. For the ambulance prediction example, the results should be shared with the same level of access control for 911 data as it may contain the inherited sensitive data. For virtual sensors, we keep track of the inputs used for processing, the user requesting the process and the outputs generated. Any metadata change will be recorded as version controls. We will provide API to control provenance information.

2.2 Implementation

We propose an architecture as shown in Figure 1 to achieve the proposed features. We have implemented the solid boxes in the figure and plan to develop the dotted boxes. For the data storage, we propose to use Spark's in-memory capabilities to provide fast query on the recently ingested data. We use Geomesa as a distributed, spatio-temporal database due to its support for distributed storage and rich spatio-temporal query features [7]. Geomesa can run on top of HBase/Accumulo. It provides scalability and space-time indexing absent in traditional databases such as PostGIS or document stores like MongoDB. In the analytics module, the offline analytics will be supported using Spark batch processing. Kafka will act as a

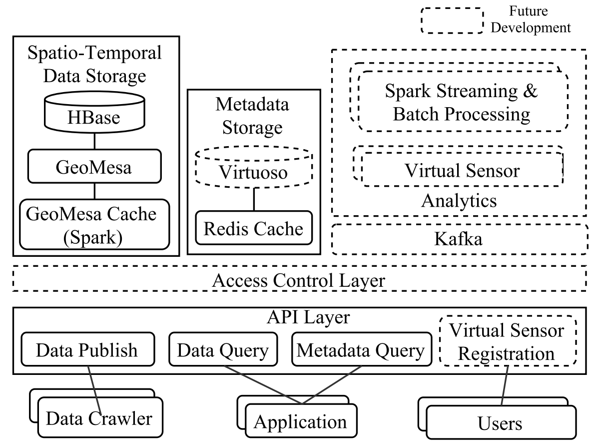

Figure 1: Citadel Architecture.

data source for different real-time analytics. The access control module governs the access of external applications/users to data streams and virtual sensors as well as access of virtual sensors to input data streams. For metadata, we currently use MongoDB to store tag-values and Redis to cache metadata with the most recent data. We will adopt a graph database, Virtuoso [3], for graph-structured metadata. Virtuoso supports a graph-querying language, SPARQL, over distributed databases in scale. We implement the entire service and REST API for data publications and querying on Vert.x, a Java framework. Vert.x natively provides asynchronous task processing for scalability and abstraction of microservice [4]. We design each component as a microservice to make them scalable and run independently.

3 ACKNOWLEDGEMENTS

This research is funded in part by the National Science Foundation under awards # IIS-1636916, IIS-1636879, IIS-1636936, OAC-1640813, CI-1331615 and CSR-1526841, and by the King Abdullah University of Science and Technology under KAUST Sensor Initiative. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the funding agencies.

REFERENCES

[1] [n. d.]. World's population increasingly urban with more than half living in urban areas. ([n. d.]). http://www.un.org/en/development/desa/news/population/world-urbanization-prospects-2014.html.
[2] 2016. NYC Analytics. (2016). https://opendata.cityofnewyork.us/.
[3] 2017. OpenLink Virtuoso. (2017). https://virtuoso.openlinksw.com/.
[4] 2017. Vert.x. (2017). http://vertx.io/.
[5] Anant Bhardwaj, Amol Deshpande, Aaron J Elmore, David Karger, Sam Madden, Aditya Parameswaran, Harihar Subramanyam, Eugene Wu, and Rebecca Zhang. 2015. Collaborative data analytics with DataHub. *Proceedings of the VLDB Endowment* 8, 12 (2015), 1916–1919.
[6] Charlie Catlett, Tanu Malik, Brett Goldstein, Jonathan Giuffrida, Yetong Shao, Alessandro Panella, Derek Eder, Eric van Zanten, Robert Mitchum, Severin Thaler, et al. 2014. Plenario: An Open Data Discovery and Exploration Platform for Urban Science. *IEEE Data Eng. Bull.* 37, 4 (2014), 27–42.
[7] James N Hughes, Andrew Annex, Christopher N Eichelberger, Anthony Fox, Andrew Hulbert, and Michael Ronquest. [n. d.]. Geomesa: a distributed architecture for spatio-temporal fusion. In *Geospatial Informatics, Fusion, and Motion Video Analytics V*.