Data Integration Platform for Smart and Connected Cities

Austin Harris†
Center for Urban Informatics and Progress (CUIP)
Department of Computer Science and Engineering
University of Tennessee at Chattanooga
Chattanooga, TN, USA
apaulh1992@gmail.com

Mina Sartipi
Center for Urban Informatics and Progress (CUIP)
Department of Computer Science and Engineering
University of Tennessee at Chattanooga
Chattanooga, TN, USA
mina-sartipi@utc.edu

ABSTRACT

By exploiting IoT technologies and smart sensors, smart and connected cities acquire massive amounts of urban data critical for various urban operations, applications, and services. Data allows cities to identify the underlying urban rhythm in real time and understand the history of the urban transformation. Cities can use this data to determine the priorities and challenges that once addressed can improve the citizen's quality of life. Implementing reliable infrastructure to ingest data produced within these cities creates many challenges such as scalability, reliability, availability, and fault-tolerance. Using these challenges as design considerations, we propose a data integration architecture design that has been implemented on our urban testbed and is currently in use. Our contributions will accelerate further urban development initiatives by providing a integration platform that can easily be deployed within other testbeds.

CCS Concepts

• **Computer Systems Organization→Real-time systems;** • **Computer Systems Organization→Distributed Architecture;** • **Information Systems →Database Management Systems;** *Stream Management;*

Keywords

Connected and Smart Cities, Architecture, Stream Processing, Distributed Systems, Data Management.

1. INTRODUCTION

It is expected that two thirds of global citizens will live in urban environments by 2050 [1]. This urbanization in combination with aging infrastructure will present new complex challenges that affect the quality of services provided by cities such as transportation, health, security, and sanitation [2, 3]. Currently, a wide variety of initiatives have been proposed that define and conceptualize the notation of smart cities in terms of sensors, IoT devices, and infrastructure that can help to overcome these challenges [4].

Data is the key component in smart and connected cities. By using data from the dynamic urban environment and big data analytics, smart cities aim to improve sustainability, create economic development and efficiencies, and enhance quality of life for citizens. We need a powerful data integration platform to support data storage and data analysis. The sensors and IoT devices generate large quantities of heterogeneous data in continuous data streams. Integrating this data into a single usable platform promotes the goal of connected cities to ultimately improve the everyday life of its citizens by developing intelligent transport systems, waste and water management systems, smart grids and energy networks. Monitoring, automating, and controlling these services will allow cities to be operated and managed more effectively than ever before. However, without the proper data management strategy and software infrastructure smart and connected cities cannot achieve this goal.

2. RELATED WORKS

Smart city architecture is not clearly defined by any one definition. A variety of research has been conducted in various domains of smart city architecture. In [5], three core components of smart city architecture are defined as: storage, application, and user-interface. The storage component is a data store that is responsible for storing all heterogeneous data generated within the smart city. The application component provides functionality that defines the services required for specific user groups. Lastly, the user interface component that is used to expose the application functionality to the end users. A similar approach is used by others to define and classify core components and functionality required by smart city architecture [6, 7, 8].

Research concentrated on the challenges and fundamental requirements of architectural models has also been a large focus of recent studies. Network infrastructure, security, data management and scalability repeatedly appear as key challenges in the development of smart city architecture [3, 4, 5, 7].

SCOPE'19, April 15, 2019, Montreal, QC, Canada.

ACM 978-1-4503-6703-5/19/04$15.00
https://doi.org/10.1145/3313237.3313301

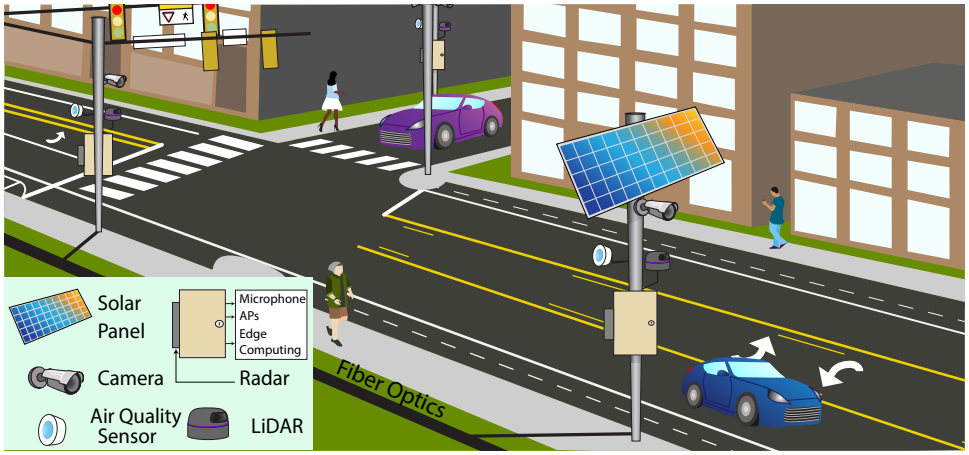

Figure 1: Visualization of MLK Smart Corridor

3. BACKGROUND

The City of Chattanooga is a pioneer in urban renewal and sustainable development. It is internationally recognized as one of the most innovative smart cities, partly because of the contributions of University of Tennessee at Chattanooga (UTC) and Chattanooga's locally owned electric distribution and communication provider, Electric Power Board (EPB). In 2009, EPB deployed a 600-square mile fiber optic network that provides up to 10Gbps Internet service to over 87,000 households and businesses. The Center for Urban Informatics and Progress (CUIP) with initial internal investment from the University of Tennessee at Chattanooga and UT Systems, operational support from Chattanooga Department of Transportation (CDOT), communications infrastructure from EPB, and design support from The Enterprise Center, has launched an urban testbed in downtown Chattanooga named MLK Smart Corridor shown in Figure 1. This corridor covers about 1.5 miles and consists of 16 poles each containing some combination and permutation of the following technologies.

- Sensors:
    - 1080p cameras
    - 4k cameras
    - Purple Air II Air Quality Sensor
    - Audio Sensors
- EDGE Computing:
    - Nvidia Jetson TX2
    - Raspberry Pi

MLK Smart Corridor is built modular and programmable to ensure additional sensors and capabilities can be augmented easily. The poles are connected to EPB's existing gigabit fiber network, allowing a backhaul for data transmission at the low latency and high throughput needed to make real-time decisions. This testbed is available and currently being used by researchers from UTC, other academic institutions, high-schools, and national labs. In this paper, we will explain the details of implementation of such integration platform that can not only be used for existing MLK Smart Corridor, but also, applied to other testbeds and sensor networks.

4. CONNECTED CITY ARCHITECTURE

Through research it was clear that our platform must support a wide range of sensors, communication methods, and external frameworks [12]. Thus, a large development effort was put forth to achieve objectives such as compatibility and interoperability to reduce the number of challenges that organizations and institutions must overcome to integrate and consume data.

Bearing this in mind, we disregarded the traditional approach of developing point-to-point data pipelines unique to each data source or application. These designs become difficult to manage and complex as they scale. Instead, this research proposes a centralized data integration platform that ingests from all source systems and devices. All systems and services that need access to this data communicate with our platform directly. Our platform lies at the end of every data pipeline whether it be source or sync. As a result, architectural maintenance and complexity is reduced.

4.1 Data Hub

At the core of our smart city platform is the Data Hub. The hub is a distributed, horizontally scalable, publish subscribe event platform. It is a centralized location for all events generated by devices within a connected ecosystem. This design creates a single point of ingest for all data. Analytics platforms, external systems, data stores, control systems, and monitoring applications access data stored in the Data Hub. As shown in Figure 2, instead of complex, unmanageable, dedicated pipelines between each system, all systems and services communicate with the data hub directly. Applications that need to share data with other applications push data into the hub and the target system is notified. This creates a single point of scalability.

The data hub consists of a cluster of machines called brokers. Brokers are responsible for providing distributed data storage that is reliable, fault-tolerant, and scalable while providing high-throughput and low-latency availability via client application programming interfaces (API). Data within the cluster is stored in immutable ordered commit logs that are partitioned across

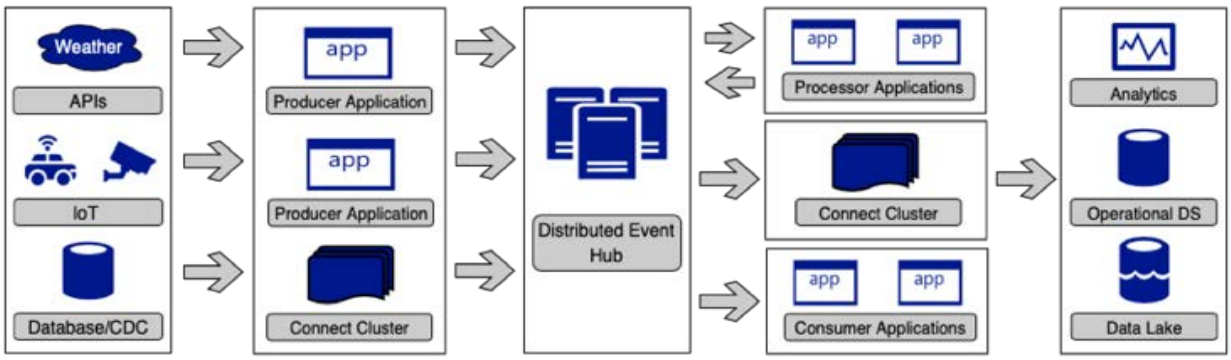

Figure 2: Architectural Diagram of MLK Smart Corridor

multiple brokers. Effectively this design concept acts as a load balancing mechanism that increases the throughput of the data by parallelizing read and writes across the cluster. Data is replicated across the cluster providing data redundancy.

The Data Hub is the central component of our data infrastructure. All other components rely on it and therefore, must have the capabilities to interface will other systems. Systems and applications communicate with our Data Hub via our client APIs. Client applications that publish data are known as producers and applications that subscribe to our system are known as consumers. Producer and consumer applications are developed using lightweight APIs that expose functionality to communicate with the cluster. The framework supports many of the common programming languages such as Java, Python, C++, and Scala. It was chosen because of its simplicity and support. Developers can embed producer applications into existing systems that need to integrate data with our platform. Additionally, all other systems that cannot communicate using these APIs can integrate via our REST API. This API exposes simple producer and consumer capabilities via REST functionality in order to maximize system compatibility.

The exchange of real-time data between systems is crucial for entities and services within the city to access and process data. Therefore, external framework, indexes, and file system integration was a necessary component of our architecture. We address this issue via the Connect framework. This framework provides prebuilt connectors that require minimal configuration to move data in and out of our platform. There are two types of connectors: sync and source. Source connectors are used to import data and sync connectors are used to export data. Connect applications are downloadable, runnable applications that run in two modes: standalone and distributed. Standalone mode consists of a single node that executes a single task. A task has a single data source and a single destination. In distributed mode, many connect instances are combined into a cluster that can execute multiple tasks. A REST interface accepts configuration files that determine what tasks the cluster should execute. A single connect cluster can export high volumes of data in real-time to many external frameworks simultaneously.

Now that we have explained the architectural components role within our platform, the remainder of this section discusses how these components address the critical challenges and capabilities of smart city architecture discussed earlier in the paper.

4.2 Information Architecture

Events are represented as a single message consisting of a key-value pair. Messages are published to a categorized stream of data called a topic. Producer applications publish messages to a topic. While consumer applications subscribe to topics. Consumer applications subscribe to a topic. For example, all data from a single air pollution sensor may publish its data to a single topic called “AirSenor_24432”. A single topic consists of a single event type. Topic schemas are stored in the schema registry which can be accessed through a REST API. Providing this functionality allows developers to more efficiently exchange data across the platform.

4.3 Scalability

As discussed in the previous section scalability is an important feature of smart city architecture. Our Data Hub cluster is horizontally scalable and can easily be scaled up by provisioning a new machine and adding the new machines/nodes address to the configuration file. Scaling the cluster can be accomplished without the need to bring the cluster offline improving availability. The cluster contains an auto-rebalancing technique that is used when nodes are added or removed from the cluster. This rebalancing technique redistributes data evenly across all machines within the cluster.

Partitioned topics distributed across brokers introduce a scaling mechanism for consumer and producer applications. Producer applications may write to the same topic without the need for coordination. The consumer framework introduces a consumer group functionality that distributes messages across many consumer applications to avoid the re-processing of a single event. Discovery services enable consumers to join and leave groups as needed. The partitions of a topic are distributed throughout the consumers within the group for processing. High-throughput topics can utilize this scalability to decrease end-to-end latency.

4.4 Low-Latency

Real-time cyber-physical systems within smart cities require low-latency response times in order to effectively provide solutions [8]. As previously mentioned, our network layer provides low-latency and high-throughput capabilities. Generally, in data processing systems one of the major inefficiencies is the serialization and deserialization of data as it moves across the network.

Within our ecosystem IoT devices, real-time monitoring and processing applications produce and consume data from the Data Hub via the network. The frameworks we utilized to implement our infrastructure uses a standardized binary format for producers, brokers and consumers. Brokers, producers, and consumers All data is transmitted in this manner. This results in the ability to use a zero-copy protocol that allows data to be called directly from the kernel page cache instead of being copied into user space to be modified. Therefore, data can be passed without the need for modification to the network buffer. Latency overhead is also reduced on the brokers as they can store data and fulfill client requests using this same technique.

Additionally, micro-batching is used between all components to push messages over the network. This mechanism minimizes network overhead by combining messages into single requests. Thresholds are configurable for batch size and the amount of time a message is waiting to be delivered. If the configured batch size is not reached the data is delivered at the configured time interval. These design considerations enable low-latency data transfer required by smart city applications to make decisions with millisecond latency, and the infrastructure to ingest tens of thousands of events per second.

4.5 Availability

Critical systems such as smart city infrastructure cannot afford to go offline [3]. Therefore, availability and reliability was a key design aspect of our data platform. The underlying technologies of our platform were chosen due to there high availability and reliability. Distributed systems provide availability in case of events that may cause machines to go offline such as flooding or fires. In the case of such events, other mechanisms are implemented to reduce data loss. In addition to a distributed architecture, the Data Hub is resilient to node failure and supports automatic recovery further ensuring that the data is never lost or inaccessible.

4.6 Security and Privacy

Our platform currently supports over the wire encryption, authorization, and access control logs (ACLs). The combination of these provides a reliable security protocol for data across all components allowing for secure data transfer, storage, and access. In the future, we will consider Blockchain to ensure security for access to deployed devices. Blockchains are distributed databases that can be securely and iteratively updated [13]. Although the concept has been around since the early 1990s, only recently applications employing blockchains have been developed, primarily to facilitate secure, private financial transactions and, as a specific implementation, as Bitcoin, which has made it a well-known technology. We will consider blockchain technology to improve the security of data management for IoT devices in smart cities.

4.7 Data Management

Our architecture provides event-level data segmentation in the form of real-time categorized event streams. These streams give users the option to subscribe and monitor the events based on their use case. This in combination with our Processing API provides a means of creating applications that process events in real-time and push enriched data to topics designated for certain systems, services, organizations, or entitles. A variety of dedicated service management streams can easily be deployed and integrated using our framework. This will enable more efficient management, planning, and utilization throughout the city over time.

4.8 Stream Processing

In smart city ecosystems, sensors generate thousands of records simultaneously. Stream processing mechanisms enable the ability to react as soon as the data is produced, as opposed to traditional queries for static data in conventional databases. Due to the volume of data, the processing layer must be scalable in order to meet the low-latency requirements.

For real-time stream processing, we utilize an elastically scalable Processor API. The API has the ability to horizontally scale depending on the throughput desired. Smart city applications aimed at improving service response time require low-latency feed back in order to be effective. Utilizing the processor API services will enable real-time analysis and monitoring.

Using the Processor API, a developer can deploy as many instances of the processing application as required to achieve the desired throughput of the topic. The only requirement is that the application ID within the configuration file is the same. The Data Hub will automatically recognize that these applications belong to the same processing group and will begin distributing data to the applications in a round robin manner. Due to the elastic scalability nature of these applications, instances can be taken offline and deployed online depending on the throughput of the pipeline at any given time. This has no impact on the performance of the overall system.

5. Future Work

The platform described throughout this paper is a building block to a complete smart city platform that will provide automation, monitoring, and control capabilities. Providing easily accessible infrastructure to other institutions is an important goal of our testbed. To provide the functionality for external users to deploy sensors and execute their applications, the Data Hub needs additional tools such as self-service portals, orchestration infrastructure, and monitoring capabilities.

6. Conclusion

We proposed a platform for data integration to be used in smart and connected cities. This platform was implemented on the urban

testbed launched by CUIP in Chattanooga urban environment. This platform, easily replicated to other smart cities, provides access to valuable data in real-time that was previously non-existent and facilitate citywide strategies that increase public safety, promote active transportation, improve air-quality, and reduce emissions. Analysis of this data that is being made available due to our platform will be used to develop future urban infrastructures to advance health equity among citizens in the near future. Design decisions such as data integration, data management, types of sensors, and data access will be available to other cities. Our proposed system solves core data challenges such as fault tolerance, low-latency, scalability, and availability.

7. ACKNOWLEDGMENTS

The project team would like to thank EPB Fiber Optics, the City of Chattanooga, and the Enterprise Center of Chattanooga for their help in constructing the MLK Smart Corridor used in this project. Also, we extend our gratitude to the NSF US Ignite (Award# 1647161) and the University of Tennessee Foundation (UCF) for partially funding this project.

8. REFERENCES

[1] World Urbanization Prospects.2018. United Nations Department of Economic and Social Affairs. https://population.un.org/wup/publications/Files/WUP2018-PopFacts_2018-1.pdf

[2] Mathieu Daquin, John Davies, Enrico Motta. 2015. *Smart Cities Data: Challenges and Opportunities for Semantic Technologies*. IEEE Internet Computing 19, 6 (2015), 66-70.

[3] Narmeen Zakaria, Jawwad Shamsi. 2015. *Smart City Architecture: Vision and Challenges*. International Journal of Advanced Computer Science and Applications 6, 11 (2015).

[4] Aditya Gaur, Bryan Scotney, Gerard Parr, Sally McClean. 2015. *Smart City Architecture and its Applications Based on IoT.* Procedia Computer Science 52 (2015), 1089-1094.

[5] Nicos Komninos. 2006. *The architecture of Intelligent Cities: Integrating Human, Collective, and Artificial Intelligence to enhance knowledge and Innovation*. 2nd IET International Conference on Intelligent Environment's (IE 06) (2006).

[6] Mahmoud Al-Hader, Ahmad Rodzi, Abdul Rashid Sharif, Noordin Ahmad. 2009. *Smart City Components Architecture*. 2009 International Conference on Computational intelligence, Modeling and Simulation (2009).

[7] Andres Monzon. 2015. *Smart Cities Concept and Challenges: Based for the Assessment of Smart City Projects*. Communications in Computer Science and Information Science Smart Cities, Green Technologies, and Intelligent Transport Systems (2015) 17-31.

[8] Giovanni Merlino, Dario Bruneo, Salvatore Distefano, Franceso Longo, and Antonio Puliafito. 2014. *Stack4Things: Integrating IoT with OpenStack in a Smart City Context*. 2014 International Conference on Smart Computing Workshops (2014).

[9] Adel Elmaghraby. 2013. *Smart City Security and Privacy In The Smart City,* In the Proceedings of the 6th Ajman International Urban Planning Conference AIUPC 6.

[10] Mohammad Abu-Matar, John Davies. 2017. *Data driven reference architecture for smart city ecosystems*. 2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advance & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation.

[11] George Kakarontzas, Leonidas Anthopoulos, Despoina Chatzakou, and Athena Vakali. 2014. *A Conceptual Enterprise Architecture Framework for Smart Cities – A Survey Based Approach*. Proceedings of the 11th International Conference on e-Business (2014).

[12] Leonidas Anthopoulos, Panos Fitsilis. 2010. *From Digital to Ubiauitous Cities: Defining a Common Architecture for Urban Development*. 2010 Sixth International Conference on Intelligent Environments (2010).

[13] Kotobi, Khashayar,, Mina Sartipi. *Efficient and Secure Communications in Smart Cities using Edge, Caching, and Blockchain*.4th IEEE Annual International Smart Cities Conference (ISC2 2018).