Cloud architecture for analyzing real-time road traffic data

1st Boban Davidovic
Faculty of Organizational Sciences
Belgrade, Serbia
boban.da87@gmail.com

2nd Dusan Barac
Facutlty of Organizational Sciences
Belgrade, Serbia
dusan@elab.rs

Abstract—**With the expanded usage of IoT sensors in road traffic and an increasing number of vehicles on the road, analyzing road traffic data is something to be considered. Road traffic data is collected using IoT devices and these datasets are massive. Road traffic can be optimized in many ways by making the traffic flow better and "smarter". The main goal of this research is to architect a model which will provide a connection between road traffic sensors and traffic participants to build a centralized community where road traffic data can be collected and analyzed in real-time. The goal is to design a cloud architecture that provides an available, scalable, and reliable service that can collect road traffic data, analyze it and provide real-time results**.

Keywords—cloud, architecture, real-time, road traffic

I. INTRODUCTION

The current IoT systems are based on simple sensors that collect data to a cloud and actuators controlled by applications in the cloud [1]. The cloud provides access to virtually unlimited resources that can be programmatically provisioned with a pay-as-you-go pricing model, enabling applications to elastically adjust their deployment topology to match their current resource usage and according cost to the current request load. IoT cloud applications therefore must be designed to cope with issues arising from geographic distribution of edge devices, network latency and outages, as well as regulatory requirements, in addition to the traditional design considerations for cloud applications [2].

Internet of things is mainly based on smart objects working in a collaborative manner and interacting instantly with surrounding environment. The emergence of the internet of things and communication technologies allows collecting different types of information from sensors and surrounding systems [3].

IoT sensors are heavily used in road traffic. In some papers, IoT sensors are used for collecting data from roads, in order to prevent road traffic accidents [4]. The IoT cloud system they built is used for private drivers and drivers of critical helpful service such as ambulances. They proved with an experiment that the provided system had acceptable response times to allow drivers to receive alert message in useful time to avoid the risk of possible accidents. In other paper researchers were focused on empowering roads using IoTin order to prevent traffic congestion. They proposed model was divided into following layers: data collection by sensors, data receiving in the cloud, object and pre-processing layer and application layer. Application layer is separated into the performance layer and the prediction layer. In the prediction layer, SVM is used to assess the congestion and the performance layer evaluated the results produced by the prediction layer's analysis. With the help of RFID sensors, drivers will receive a notification about a congestion point [5].

The goal of this research is to design a cloud architecture that provides an available, scalable, and reliable service that can collect road traffic data.

II. EXISTING SOLUTIONS

Presently, numerous papers are available that are considering real-time processing of traffic data. In [9], the main focus is on deployment of the Internet of Vehicles (IoV), integrating Big Data analytics of the road network traffic measurements. This is measured for the city Mohammedia in Morocco. They presented architecture based on three layers: IoV, Fog computing and Cloud computing, with a specific focus on Fog computing layer, which is used for collecting and processing events generated by intelligent vehicles and traffic visualization per road section. However, in this project the scope is limited to intelligent vehicles for collecting road traffic data.

In [10], researchers have implemented a cloud-based platform (SMASH), specifically designed for handling near real-time traffic data. They showed how their platform can be used for analyzing social media data used for traffic jam identification. They used spatial and temporal clustering tweets on the road network and compare results with real-time traffic data from Sydney Coordinated Adaptive Traffic System. Real-time traffic is also in focus of [11]. Researchers have considered the real-time and accurate prediction of road traffic states as the basis of information service for road traffic participants. They presented an algorithm on kernel K-nearest neighbors (kernel-KNN) to predict road traffic states in time series. First, representative road traffic state data are extracted to build the road traffic running characteristics reference sequences. Then, kernel function of the road traffic state data sequence in time series is constructed. The current and referenced road traffic state data sequences are matched, based on which k nearest referenced road traffic states are selected and the road traffic states are predicted. They have considered data from roads in Beijing and their final experiments results prove that the road traffic states prediction approach based on kernel-KNN presented herein is feasible and can achieve a high level of accuracy.

In [12], researchers are focused on predicting a crash frequency based on real-time data. They are considering types of collision on Expressways. They have analyzed weather, speed, traffic volume and density as parameters to run the prediction. Due to the impact of real-time driving environment data such as traffic flow on traffic accidents, and with the technology progress of traffic data detection and storage, real-time crash risk assessment has become a research hotspot in the field of traffic safety. Other studies on traffic crash prediction mainly focus on the crash frequency and crash severity of freeways or arterials [12].

In [13], researchers are focused on analyzing and describing deep learning architectures in traffic domain. They were describing a deep learning architecture based on road traffic data as the graph - GNN (graph neural network). They provide a comprehensive and clear picture of such emerging trend. Their survey examines various graph-based deep learning architectures in many traffic applications. In their paper, they first give guidelines to formulate a traffic problem based on graph and construct graphs from various kinds of traffic datasets. Then they decomposed graph-based architectures to discuss shared deep learning techniques, clarifying the utilization of each technique in traffic tasks.

Also, as an important topic for tracking road traffic data, visualization is used. In [14], researchers did a structured survey of the state of the art in the visualization of traffic data. First, they reviewed traffic data visualization methods: WebVRGIS based traffic analysis and visualization system, TripMiner, IoV distributed architecture, SMASH architecture, and LDA-based topic modelling. They analyzed the traffic datasets that applied in each method and did summarize methods from following aspects: scalability, data storage, data update, interactivity, reliability, data anomaly detection, and spatiotemporal visualization. They did a detailed comparative analysis of the key capabilities of representative traffic data visualization methods in processing traffic big data and they conclude that the SMASH architecture performs better in processing high speed and large flow traffic data.

Researchers are in general focused on specifics of the road traffic data - how this data could be analyzed, visualized or fed into prediction algorithm. In this paper, we are considering different approach by looking at the big picture of planning a informatics system that will be able to collect huge amount of sensor data, process this data and return analyses report to other traffic participants.

III. Methodology

With the rapid development of cities, massive and complex traffic data is being generated and collected. The traffic data is not intuitive and cannot highlight key information about urban traffic conditions [14].

Road traffic systems are complex and highly variable and traffic states prediction based on big-data driven methods is becoming an increasingly important concept [11].

Big data from the road traffic is gathered and processed in the cloud systems where this data is stored, processed and analyzed.

This data is processed and stored in the cloud. Systems designed in the cloud are: available, scalable, reliable, are able to collect and visualize traffic data, and are able to analyze traffic road data.

A. Designing an available cloud system

When we talk about cloud systems, the first that we consider is availability. Cloud systems provide availability zones. Availability zone is a unique physical location within a region. Each availability zone is made up of one or more datacenters equipped with independent power, cooling, and networking. To ensure resiliency, there's a minimum of three separate zones in all enabled regions. The physical separation of Availability Zones within a region protects applications and data from datacenter failures [6]. To ensure best availability several availability zones should be used (one is the main one and others are replications). If one availability zone fails, the other will be accessible.

B. Designing a scalable cloud system

Scalability is the core of each cloud system. In the cloud systems it is very convenient to scale both horizontally and vertically. Vertical scaling is also called scale up and down and it is considering when the size of the virtual machine is increased/decreased. Horizontal scaling is when number of virtual machines is increased/decreased and they are working together. It is considered for horizontal scaling to be more flexible, since it can easily spin hundreds of virtual machines. All popular cloud provides autoscaling option, which can be configured. Auto scaling is always using horizontal scaling and it can trigger automatically, based on the defined measures (CPU, RAM usage and similar).

C. Desigining a reliable cloud system

The probability that a system is operational in a time interval without any failures is represented as the system reliability. Cloud systems use fast and real-time failure detection to identify or predict a failure in the early stages is one of the most important principles to achieving high availability and reliability in cloud systems. A service level agreement is often defined between a company that builds a service and the owner of the service. It is simply defined as a part of a standardized service contract where a service is formally defined. It is an agreement about the quality of a provided service. This is referring to system reliability [7].

D. Design a system that collects and visualize traffic data

Traffic data consists of a large amount of data that comes in streams directly from IoT sensors. Various sensors can be used, but they all provide large number of data points which are used. For accepting streaming data, normal REST API is not something that is recommended. This can easily become a bottleneck. When designing a system that should take large amount of data that is being sent simultaneously it is always first thing to consider data streams.

Many relevant IoT applications can take advantage of streaming data, as i) those based on distributed monitoring systems using embedded devices with limited processing capacity (e.g., environmental analysis), and ii) real-time data stream analytics, processing a large volume of data (e.g., e-health and driving assistance systems) [8].

Largest cloud providers have included services that can process hundreds of thousands records per second. These records are looked as data streams. Benefit of data stream is that it can analyze data on the fly and it allows real-time insights from the sensor data to look for patterns and take actions on them. Cloud providers do allow writing queries directly on the real-time data that is streamed. This data can also be visualized [15]. "Table I" shows the list of actions that could be done against the streaming data in the cloud.

TABLE I. ACTIONS AGAINST THE STREAMING DATA IN THE CLOUD

Action name	Description
Real-time analysis	Streaming data is processed real time
Batch processing data	Grouping data before processing with short latency intervals
Visualize data	Real-time data is populated to a visualization service
Data analytics	Advances analytics of raw and processed data

Both real-time data and batch processed data can be used for analytics and visualization. "Table II" shows what is the difference between batch and real-time processing in the cloud.

TABLE II. COMPARISON BETWEEN BATCH AND STREAM PROCESSING

Batch processing	Stream processing
Processing over all the data in the dataset	Processing over the most recent data records
Large batches of data	Micro batches consisting of a few records
Latencies in minutes to hours	Latency in the order of seconds or milliseconds
Complicated analytics	Simple aggregates and rolling metrics

E. Design a system architecture for collecting road traffic data

The idea of the designed system is to be able to collect road traffic data, prepare and send it to the cloud. Data is collected and processed on the cloud. Processing data will give us more insight about it and we will be able to make some conclusions. These conclusions should be shared to end users in traffic. High level architecture of the system is presented on the figure 1.

Road traffic data is collected using IoT sensors. There are many different IoT sensor types that are used in traffic monitoring, for example light detectors, radars, cameras, humidity sensor, accelerometer, fire sensor, etc. Those sensors collect road traffic data, which are gathered in the IoT gateway. The IoT gateway is used as a bridge between IoT sensors and the cloud. It is used for initial data filtering. IoT gateway is also handling checks if devices are healthy, if they are sending good data, etc. The IoT gateway is sending data from sensors to the cloud through streaming platform. IoT streams look a lot like web server log events. Events are being generated at high volumes, and they need to be processed and either made available to downstream consumers and stored in databases. Road traffic data is high priority for members in the traffic and require very low latency. These data streams have widely varying data formats. Streaming platform is able to receive large amount of data. Received data is guided to one of the following: device tracking (for real-time processing), log analytics (batch processing) and predictions service (near real-time). Prediction service communicates to traffic participants and notifies them when there are some changes.

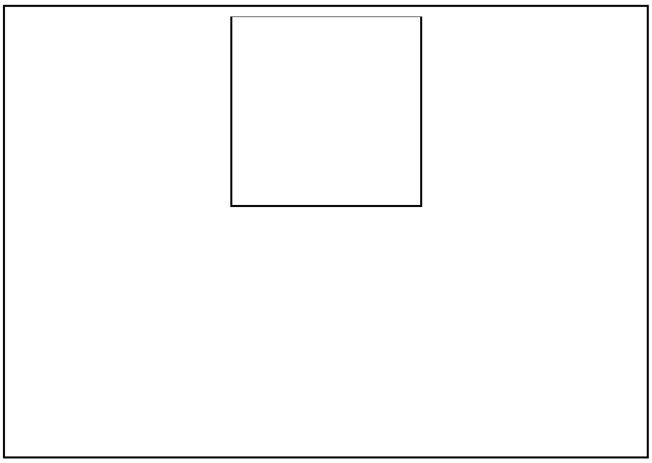

Fig. 1. High level architecture

The most important part of this architecture is the stream processing. When the data quantity is coming from IoT devices, it is the first thing to expect that it will be massive. Therefore, instead of using standard Rest APIs, using streaming engine is crucial. In our case, we are using Kafka as streming engine. Sensor data flow architecture is presented on Figure 2.

Kafka is an open-source distributed event streaming platform. It is used for high-performance data pipelines, streaming analytics, data integration, and mission-critical applications. It is scalable, has high availibity and has high throughput [16]. All sensor data is stored in Kafka cluster.

As an intermediary between IoT sensors and Kafka MQTT servers are used, accessible through MQTT load balancer. This will allow easy horizontal scaling of MQTT servers.

MQTT is a standard messaging protocol for IoT. It is designed as a lightweight publish/subscribe messaging transport. It is ideal for connecting remote devices with a small code footprint and minimal network bandwidth.

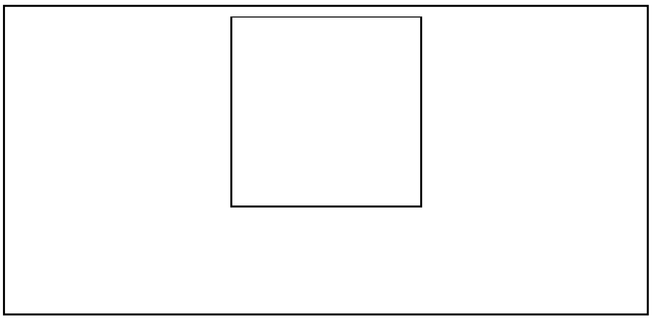

Fig. 2. Sensor data flow architecture

For communication between MQTT server and Kafka we are using Kafka Connect. Kafka Connect is a free, open-source component of Apache Kafka that works as a centralized data hub for simple data integration between databases, key-value stores, search indexes, and file systems. Kafka Connect can collect metrics from all your application servers into Kafka topics, making the data available for stream processing with low latency [18]. Transfering from MQTT to Kafka is shown on the figure 3.

Fig. 3. Data flow diagram - MQTT to Kafka (using Kafka connect)

On the figure 3 we can see how data changes from MQTT broker (message broker) to Kafka (inbound flow) and from Kafka to S3 object storage (outbound flow). In the inbound data flow MQTT Source Connector connects to a MQTT broker and subscribes to the specified topics. Then Avro Converter converts data to Kafka connect to Avro format. In the outbound data flow S3 Sink Connector export data from and to Kafka topics to S3 and data is saved into S3 object storage.

After the data has been processed on the cloud, the predictions service is ready to communicate the results to the traffic participants. On the figure 4 this communication is shown.

Fig. 4. Data flow diagram - send data to road traffic participants

After machine learning model finishes up data processing and returns some results / recommendations, we are using notification system to notify diffent traffic participants on the updated results / recommendations. This is separated into two notification queues: for vehicles and for users. Notifications for vehicles should show up directly in the car, while notifications for users can come in different formats: SMS, email and push notification. Using notification system we are assuring that vehicles and users in the traffic are notified as soon as there is some new information to be shared.

IV. Conclusion

The main focus of the paper is on preparing infrastructure that can handle large amount of road traffic data that is collected by using IoT devices. The presented cloud architecture is scalable, reliable and available. Since the amount of data coming from traffic roads is huge, it is very important to consider that as the most important system requirement. Communication stream goes from IoT devices that are connected to the vehicle or that are present on the road to the cloud. This data travels back to the traffic road consumer and makes a complete circle – data ends where started. The system collects the data, process it, analyze it and return suggestion to traffic road consumers. The benefit of this architecture is that it is using streaming system for collecitng road traffic data. This architecture can easily be scaled on both ends. On receiving end, Kafka streaming platform is using MQTT scalable server using load balancer. On sending end, notification queues are being used for sending notfications. These queues can also handle large amount of data.

This architecture can be used with various sensors and for different analysis. It can return various recommendation to the road traffic users. Future extension of the architecture includes in depth definition of data analytics and describing procedures for data manipulation.

V. References

[1] F. Ahmadighohandizi, and K. Systä, “Application Development and Deployment for IoT Devices,” Advances in Service-Oriented and Cloud Computing, 2016, pp. 74-85.

[2] M. Vögler, J. M. Schleicher, C. Inzinger and S. Dustdar, “DIANE – Dynamic IoT Application Deployment,” 2015 IEEE International Conference on Mobile Services, 2015, pp. 298-305.

[3] M. Nahri, A. Boulmakoul, L. Karim, and A.Lath, “IoV distributed architecture for real-time traffic data analytics,” The 9th International Conference on Ambient Systems, Networks and Technologies, Procedia Computer Science, 2018, pp. 480-487.

[4] A. Celesti, A. Galleta, L. Carnevale, M. Fazio, A. Lay-Ekuakille, and M. Villari, “An IoT Cloud System for Traffic Monitoring and Vehicular Accidents Prevention Based on Mobile Sensor Data Processing,” IEEE Sensors Journal, vol. 18(12), pp. 4795–4802

[5] A. Ata, M. A. Khan. S. Abbas, M. S. Khan, and G. Ahmad, “Adaptive IoT Empowered Smart Road Traffic Congestion Control System

Using Supervised Machine Learning Algorithm, ” The Computer Journal

[6] https://docs.microsoft.com/en-us/azure/availability-zones/az-overview, “Availability zones overview,” [Accessed 14.05.2021]

[7] M. R. Mesbahi, A. M. Rahmani, and M. Hosseinzadeh , “Reliability and high availability in cloud computing environments: a reference roadmap, ” Human-centric Computing and Information Sciences vol. 8 (20), 2018.

[8] R. Cárdenas, P. Arroba, R. Blanco, P. Malagón, J. L. Risco-Martín, and J. M. Moya, “Mercury: a Modeling, Simulation, and Optimization Framework for Data Stream-Oriented IoT Applications,” Simulation Modelling Practice and Theory, vol. 101, 2020.

[9] M. Nahria , A. Boulmakoula, L. Karima and A. Lbath, “IoV distributed architecture for real-time traffic data analytics,” Procedia Computer Science, vol. 130, 2018, pp. 480 –487

[10] Y. Gong., P. Rimba., and R. Sinnott. “A Big Data Architecture for Near Real-time Traffic Analytics,” Companion Proceedings of The 10th International Conference on Utility and Cloud Computing, 2017.

[11] D. Xu, Y. Wang, P. Peng, S. Beilun, Z. Deng, and H. Guo, “Real-time road traffic state prediction based on kernel-KNN,” Transportmetrica A: Transport Science, pp. 1–15. doi:10.1080/23249935.2018.1491073

[12] H. Zhang, S. Li, C. Wu, Q. Zhang, nd Y. Wang, “Predicting Crash Frequency for Urban Expressway considering Collision Types Using Real-Time Traffic Data,” Journal of Advanced Transportation, vol 2020, 2020.

[13] J. Ye, J. Zho, K. Ye, and C. Xu, “How to Build a Graph-Based Deep Learning Architecture in Traffic Domain: A Survey,” IEEE Transactions on Intelligent Transportation Systems, pp. 1-21, 2020.

[14] F. Qian,, “Visualization of Traffic Data: A Survey of Methods and Datasets,”, IEEE 6th International Conference on Computer and Communications (ICCC), 11-14 Dec. 2020, China. DOI: 10.1109/ICCC51575.2020.9345004

[15] https://docs.microsoft.com/en-us/azure/stream-analytics/stream-analytics-get-started-with-azure-stream-analytics-to-process-data-from-iot-devices, “Process real-time IoT data streams with Azure Stream Analytics,” [Accessed 14.05.2021.]

[16] https://kafka.apache.org, “Apache Kafka”, [Accessed: 15.06.2021.]

[17] https://mqtt.org, “MQTT: The Standard for IoT Messaging”, [Accessed: 15.06.2021.]

[18] https://docs.confluent.io/platform/current/connect/index.html, “Kafka Connect”, [Accessed: 15.06.2021.]