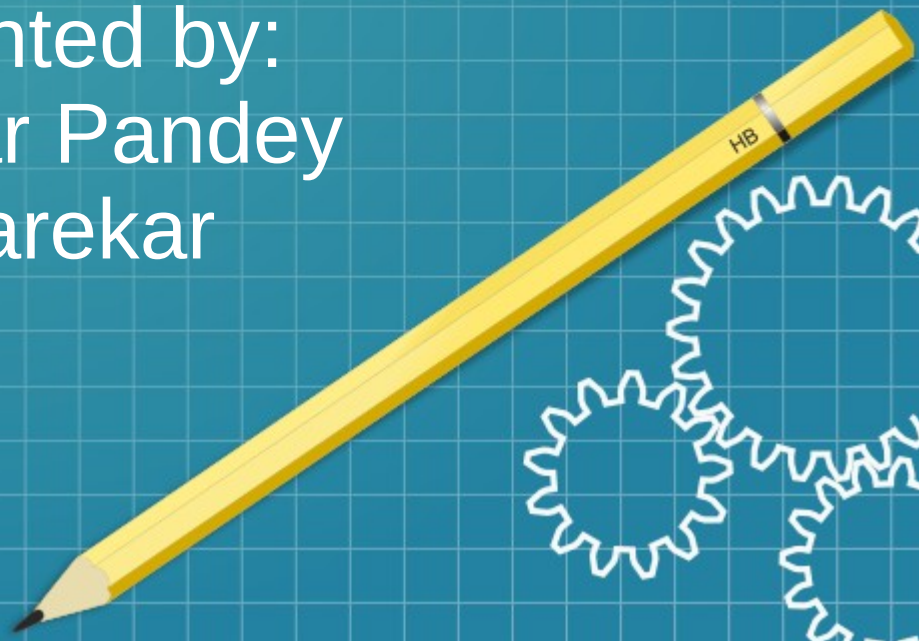
TCUP Hackathon 2023

Solution presented by:
Mr. Manishkumar Pandey
Mr. Atharv Darekar

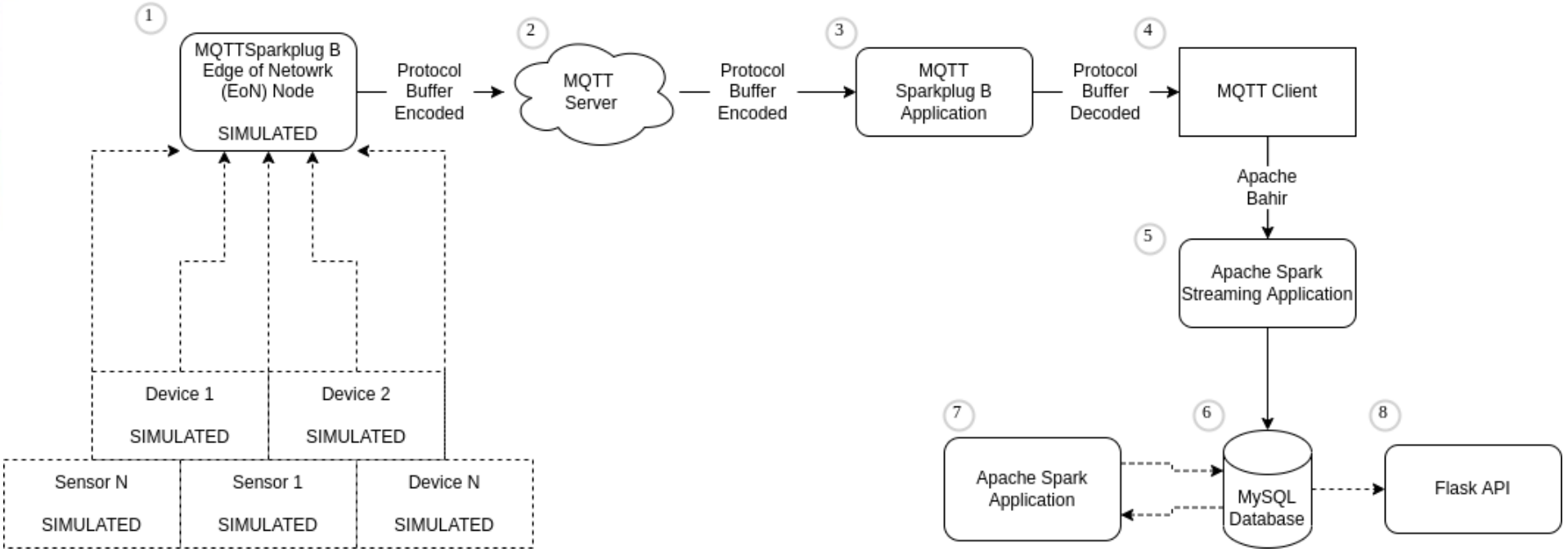


Use Case: Ice Plant

- This solution provides a working model of real time event processor for IoT devices in an ice plant.

- The solution provided is generic in nature and can be easily extended to any IoT application use case.

Architecture



1. MQTT Sparkplug B Edge of Netowrk (EoN) Node

- It simulates the data generated by various sensors and devices present in the monitoring system of the plant.
- It is developed using the mqtt-spb-wrapper version 1.0.16
- The generated data is compliant with the Sparkplug B schema and is encoded using Protocol Buffer.
- The program is configurable.

2. MQTT Server

- The system uses Mosquitto MQTT broker version 1.6.9-1.
- It receives the data generated by the simulator.

3. MQTT Sparkplug B Application

- This is the standard component of the Sparkplug B infrastructure responsible for subscribing to the various topics to receive the data published by the edge nodes.
- The data received from the MQTT topic is decoded and sent to the MQTT client for ingestion into Apache Spark Streaming application.

4. MQTT Client

- It is a simple program to feed the message received from the topic to the Apache Spark Streaming application.
- It is developed using paho-mqtt version 1.6.1.

5. Apache Spark Streaming application

- Developed using Apache Spark version 2.4.0, it ingest the data into MySQL server, version 8.0.32-0ubuntu0.20.04.2 by imposing a proper schema on the raw data received from the client.
- Apache Spark uses Apache Bahir version 2.11 to connect to the MQTT client

6. MySQL Database

- MySQL database is ideal for online transaction processing (OLTP) system and can easily handle the data write operations done by Apache Spark.

7. Apache Spark application

- This is a batch processing application configurable to process the streaming data as per the users requirement.
- It consumes the data stored in the MySQL database, processes it and writes back to MySQL database for report generation and future use.

8. Flask API

- The database is exposed for further activities like monitoring, analysis, visualization and reporting using the API.

Features

- The system is modular and scalable in nature, each component can be replaced with suitable tool as per the use case, for example
  - MQTT client can be replaced by Apache Kafka
  - Apache Spark can be replaced by Apache Flink
  - MySQL database can be replaced by Apache Hive
- The pipeline is developed using a layered approach which helps isolate and fix issues in case of failures.

Technology Used

- Flask 2.2.3
- Flask-SQLAlchemy 3.0.3
- Java 2:1.11-72
- Mosquitto 1.6.9-1
- MySQL Connector j-8.0.32
- MySQL Server 8.0.32-0ubuntu0.20.04.2
- PySpark 2.4.0
- Python 3.7
- Ubuntu 20.04
- mqtt-spb-wrapper 1.0.16
- mysql-connector-python 8.0.32
- paho-mqtt 1.6.1

THANK YOU…!