Industrial IoT Devices Analytical Reporting and Monitoring

Capstone Public Project Summary

Student Name Student ID Company Name Project Category Faculty Supervisor Ali Shahzad 1902213 ExxonMobil Analytics and Monitoring Dashboard Gergely Daroczi

Background:

ExxonMobil, an Oil & Gas corporation, is one of the world's largest publicly traded energy providers and chemical manufacturers that uses next-generation technologies to identify, analyze and solve everyday problems to meet the world's ever-growing need for energy. Oil & Gas companies need constant monitoring and instant feedback from all on-field operations in order to effectively mitigate any unpredictable events. The company has some thousands of Industrial IoT sensors that report real time Up-stream and Down-stream data. Due to a complex interconnected graph network of devices, keeping the devices functional (average availability) and minimizing the number of outages (devices malfunctioning) is one of the main goals of the IoT Edge Delivery Team at ExxonMobil.

Business Problem:

As Internet of Things has been around only for the past twenty years, the continual maturation phase of IoT solutions can have numerous issues related to security, connectivity and data transfer etc. This project aims to discuss the connectivity issues and devises a strategy to aid in its mitigation processes.

One way to regulate the network availability of these devices is to dedicate a server that continually pings these devices and awaits a response in return. A successful response registers the device as available and a no-response triggers an outage event. The company currently uses OpenNMS to implement the above-mentioned methodology, which is an open-source tool that continually pings the Industrial IoT sensors and registers the response in a database. The software effectively monitors the devices however it lacks severely at reporting the collected metrics to the end user which limits the actual usability of the collected information.

The current reporting methodology involves manually downloading CSV files from the OpenNMS portal, cleaning and transforming the data to create some performance graphs and then emailing the insights to the users every week. A methodology to automate this entire process was the question at hand.

Project Scope and Workflow:

The project aimed at building an end-to-end solution that would execute every day to collect new data, clean and transform it, store it in a database and finally display it over the web so that users from within the organization could choose to display it at any point in time. Hence, this meant that it wasn't a stand-alone solution but instead had to be deployed in a production environment to automate the entire data pipeline.



The scope was thus defined as a combination of three separate python applications that would be containerized and deployed on Red Hat's OpenShift Container Platform. Each of these applications had a separate purpose that would play a key role in the solution proposed:

- App 1: Availability and Outages data collection via Python Get requests to the OpenNMS API. Pandas was used for data cleansing and filtering
- App 2: Data storage to a containerized instance of MongoDB on the OpenShift platform
- App 3: Python Dash Application that queries the database for data and displays queried aggregates of data based in user input via callbacks

All development and coding were done on Microsoft VS code due to its enhanced interoperability with Git, as an internal GitLab server was used for all code review and deployment.

Data Collection:

As an important part of the project was to create a fully automated data pipeline, the OpenNMS API played a key role in almost all my data collection exercises. Being an open source technology, OpenNMS has a well-versed publicly available API documentation. The Real Time Console Data endpoint was used to collect past 24 hours availability and outage metrics every day.

Data Storage:

As the application was deployed on OpenShift, I used a MongoDB instance on OpenShift for document data storage as it allowed frequent and fluid data retrieval without defining complex data types and schemas etc. The collected data was stored to MongoDB using a POST request in Python.

Web Dashboard (Overview):

The analytical dashboard was the most critical part of the project as it is here where all the analytics, aggregations and visualizations were performed. The dashboard was built using the Python Dash framework and was made interactive to display information based on the user selections on the dashboard. It had three main purposes to serve:

Overall Performance of Devices per Site

The first graph shows the user an overview of the performance of all the devices on each site highlighting more the bad performers.

Performance Timeline for a User Selected Site

The second graph shows the performance of a single site the user may be interested in based on the performance depicted by the first graph.

Gateway Level Performance on Each Site

As each site had multiple IoT devices installed, the final graph would allow the user to see the performance of each individual device on a selected site for a specified date for troubleshooting

The snapshot to the right aims to provide a high-level view of the dashboard. Sensitive information from the axes has been hidden to ensure the company's privacy. The snapshot captures a demo troubleshooting exercise.

It shows that the user was interested in the site US-*** depicted by the yellow indicator. A click on the bar caused the second graph to show а performance timeline of the selected Site. The user then clicked on April 19th to see how individual the gateways performed at US-*** on the 19th. Conclusively, the sixth gateway was the one that malfunctioned.



Conclusion:

The project served to be my first endeavor to build an end-to-end data science and engineering solution. I was new to almost all the tools and technologies used hence the project provided a steep learning curve for me. Working at ExxonMobil in an agile environment was perhaps the highlight of this project as it forced me to develop production ready code and applications.