MongoDB for IoT Systems in Manufacturing

Manufacturing companies face intense challenges in sensorizing and connecting equipment to collect and analyze IoT data. IoT applications need scalable databases with flexible schemas, high availability, excellent compression, and data tiering capabilities to manage massive amounts of time series data. Additionally, real time data analysis is crucial for many manufacturing IoT use cases such as condition monitoring and anomaly detection.

MongoDB enables flexible and scalable IoT applications

MongoDB embraces the variety and volume of IoT data without compromising on performance. Through its rich document model and purpose built time series collections, MongoDB eliminates data movement and blends time series with the rest of the enterprise data in a single Developer Data Platform - MongoDB Atlas.

Every IoT application has three major components:

- Data Collection
- Data Analysis
- Data Management

Each of these components come with its unique challenges. How does MongoDB address them?

Data Collection

As the first step towards building an IoT application, it is necessary to provision devices, machines, and gateways and define relationships between them. We also need to collect and visualize raw data from devices and, optionally, control devices by sending commands from the IoT application to devices.

Challenges and how MongoDB helps

As an example, a factory floor has diverse types of sensors installed on different machines/equipment. Each sensor may have its own unique data fields that need to be captured and stored. MongoDB's document model enables data of any structure to be stored and processed. The schema can be adapted by adding new fields as they are needed without downtime, making it simple to handle rapidly changing data of IoT applications. Not to mention, the document model is an excellent choice to create hierarchical models that show parent, child relationship between a production line and the machines on the line for example.

IoT data is typically categorized as time series data which needs to be combined with other operational data such as maintenance records, inventory levels, and quality control metrics to reveal inefficiencies or issues in production operations. That said, setting up a niche time series database and synchronizing data between time series database and operational database increases architectural complexity. Using MongoDB's Time Series Collections, an organization can effectively and securely manage both time series and operational data within one single, versatile database. Time series collections use a highly optimized columnar storage format that provides excellent compression, along with fast query processing. Finally, the MongoDB Kafka Connector, or a variety of third party connectors (MQTT, OPC), can be used to send commands and messages back to end devices if required.

Data Management

It's not uncommon for IoT projects to fail because they don't cater for scalability and data tiering. Alongside that, we need to ensure easy and flexible deployment in a variety of different environments (edge, on-prem, cloud). Post deployment, an IoT application needs to keep a robust audit log for continuous improvement.

Challenges and how MongoDB helps

In a customer facing IoT application, anticipating and accommodating fluctuations in IoT workloads is challenging. Instead of just adding bigger storage, servers and increasing cost along the way, MongoDB offers horizontal scalability along with auto-scaling features that allow the IoT database cluster to auto-scale in response to workload changes. Atlas Online Archive automatically archives aged data to cheaper storage to balance cost and performance. This eliminates the need to manually migrate or delete data. MongoDB works on the principle of *freedom to run anywhere!* allowing users to deploy MongoDB either on the edge (via MongoDB Enterprise Advanced) or as a fully managed service on any cloud provider (AWS, Azure or GCP), giving a true multi-cloud experience. This ensures global scalability and reachability. Finally, MongoDB provides fine-grained auditing capabilities that enable users to track and record operations and changes made to the database. This helps organizations not just detect and investigate security incidents but also improve the IoT application performance.

Data Analysis

Once a robust IoT data infrastructure is established, the data needs to be aggregated, analyzed, searched through and actions need to be taken based on analysis.

Challenges and how MongoDB helps

Aggregation of data coming from different IoT sensors requires the creation of a unified and coherent data model. Such a model is extremely difficult to create using a relational database. The document data model with its inherent flexibility is tailor made for such a scenario. To meet strict real time requirements of an IoT application, data often needs to be analyzed in motion. Atlas Stream Processing, a part of MongoDB Atlas platform, enables continuous querying, analysis, and reaction to streaming IoT data without the delays inherent to batch processing or the need for a separate stream processing engine. Once the data lands in MongoDB, robust analytical pipelines can be created using window functions and other data aggregation operators. Finally, IoT applications need advanced search capabilities for not just the users but also to generate context for Large Language Models in Generative AI applications. MongoDB Atlas integrates the database, search engine (Atlas Search), semantic search (Atlas Vector Search), and sync mechanism into a single, unified, and fully managed platform. This reduces architecture complexity and increases developer productivity. This allows developers to focus on innovation and not worry about IoT data management and the challenges that come alongside it.