Benchmarking Time Series Databases for Monitoring and Analytics: TimescaleDB vs. InfluxDB

A Comprehensive Comparison of Two Popular Time Series Databases for Energy Plant Data

Graduate

Lucien Hagmann

Initial Situation: The growing importance of time series (TS) data, driven by technological advances and the emergence of the Internet of Things, underscores the need for tools capable of handling the inherent challenges. Time series databases (TSDB) advertise their suitability in this regard by providing techniques catered to this data category. However, within this domain, disparities emerge between SQL-based solutions on one side and purpose-built NoSQL solutions on the other side. Their architectural differences must be considered when faced with a TS application. In energy plant monitoring, data is first gathered in a sensor fusion unit (SFU) and then transmitted to a remote data storage and processing unit (RDSPU) for further analytics (Fig. 1). Here, a TSDB must interface with various tools along the data pipeline and provide the means to efficiently manage the workload. This thesis analyses whether the PostgreSQL-based TSDB TimescaleDB can meet these requirements. Furthermore, TimescaleDB and the purpose-built TSDB InfluxDB are benchmarked.

Approach: The goal of this thesis is to provide a comprehensive comparison of TimescaleDB and InfluxDB. First, by describing their techniques, such as TimescaleDB's hypertable partitioning or InfluxDB's time-structured merge tree and columnar data structure. Both leverage caching and enable efficient data processing. Then, by implementing both tools in the context of an industry scenario. Thus, the key characteristics of TS, specific TSDB techniques, and benchmark methodologies are explored. Based on these findings, the framework for the benchmark is defined. As shown in Fig. 2 these findings precede the practical implementation. In this phase, a server-based test environment is created to emulate production conditions. Real energy plant data is collected for two months. Every 10s around 25k datapoints are written to the database, leading to a dataset that grows by 2GB each day. For the performance benchmark, disk usage, query execution time, CPU and memory resources are measured in several tests. In addition, a list of twelve key criteria for a TSDB in the given scenario (i.e. interoperability, or access management) is established based on literature research and direct exchange with industry representatives. For each performance test and criterion, the tools are rated on a scale from -2 to 2, resulting in a performance score and an overall product score.

Result: Despite their operational differences, this study provides a nuanced and transparent comparison of two popular TSDBs. It was possible to embed TimescaleDB in the current data pipeline and use its data for plant monitoring dashboards (Fig. 3). The performance tests show that InfluxDB, in its default configuration, exhibits superior query performance through its TS-optimized engine. In addition, InfluxDB uses less disk space by directly compressing data. However, with features like continuous aggregates, multi-indexing, or manually set compression policies, TimescaleDB can notably mitigate the performance gaps. Under the benchmark conditions created here, InfluxDB achieves a higher performance score (7) than TimescaleDB (3). TimescaleDB, on the other hand, offers more advanced database features, such as granular access management, due to its integration with PostgreSQL, while maintaining high performance for TS. Therefore, in the context of the given scenario and the established list of criteria, TimescaleDB achieves a higher overall product (17) score than InfluxDB (13).

Fig. 1: Schematic of an energy plant monitoring data pipeline with a remote time series database.
Own presentment

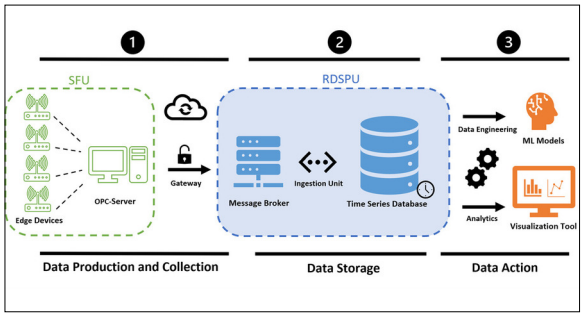


Fig. 2: Framework for benchmarking this thesis with phases 1-5 for efficient implementation and sound results.
Own presentment

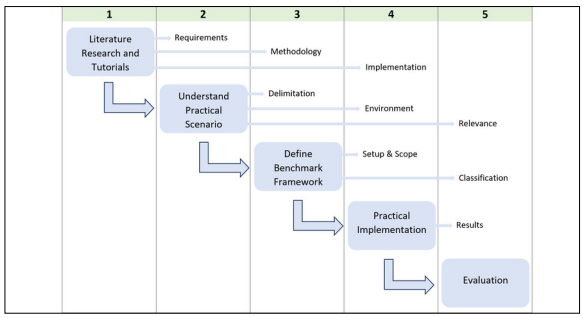
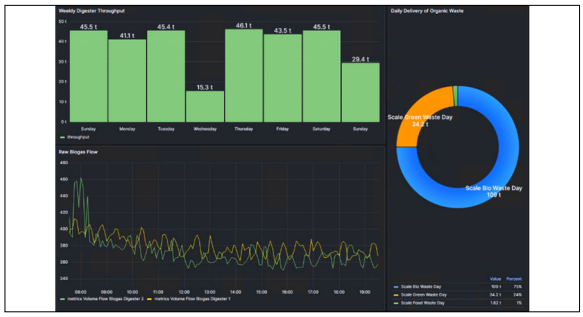


Fig. 3: Sample dashboard created in the Grafana tool showing near real-time plant data from TimescaleDB.
Own presentment



Advisor
Prof. Stefan F. Keller

Co-Examiner
Dr. Ralf Hauser, Zürich, ZH

Subject Area
Software and Systems, Computer Science, Data Science