Evaluating the Vertical and Horizontal Read and Write Scalability of MobilityDB and CrateDB

Bachelor's Thesis Exposé

Eryk Karol Kściuczyk

Spatio-temporal databases are increasing in popularity due to increased production of such data by numerous IoT devices. As the need for such databases increases, developers try to reach for familiar and open-source solutions. MobilityDB, built on top of the spatial extension PostGIS, extends PostgreSQL's capabilities regarding spatio-temporal aspects. Previous research has shown that the MobilityDB extension can be partially used in conjunction with Citus, a PostgreSQL extension, which provides horizontal scalability options. CrateDB, on the other hand, is a distributed SQL database that is designed for scalability and high performance, making it a potential candidate for comparison.

While specific queries runtimes of BerlinMOD benchmark have been evaluated on 4 and 28 node clusters, horizontal and vertical scalability of the platform has not yet been fully explored, especially ingestion performance. BerlinMOD queries are not designated for distributed MOD and different set of queries is required to further assess its performance. Moreover, benchmarking with a wider range of cluster sizes is necessary to establish the scalability pattern. Additionally, a comparison between MobilityDB+Citus and CrateDB in terms of horizontal scalability has not been conducted, which could provide valuable insights into their respective strengths and weaknesses.

In this thesis we aim to benchmark the vertical and horizontal scalability using generated spatio-temporal data to measure metrics like request latency, query runtime, and read/write throughput under different data sizes. We will generate data for the benchmark using a custom data generation tool specifically developed for this study. The tool will simulate the movement of e-scooters, chosen due to their rising popularity, movements within urban areas, different varying speeds and use times. The benchmarking setup for MobilityDB can be seen on figure 1.

We will compare the performance of MobilityDB combined with Citus and CrateDB on different hardware configurations using ~~Google Compute Engine service on~~ Google Cloud Platform. Additionally, we will evaluate several scenarios, including vertically scaling a single node and scaling out by distributing both solutions across several instances. Afterwards, we will compare scalability trade-offs between those two approaches.

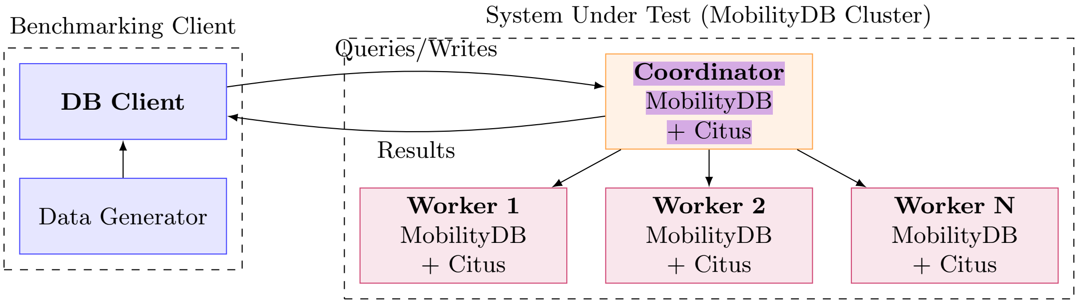

Figure 1: Benchmarking client will write generated e-scooter movement data and execute the workload against System Under Test (MobilityDB cluster). The cluster consists of a coordinator and multiple workers, each running PostgreSQL with MobilityDB and Citus extensions.

We will measure query latencies and throughput using confidence intervals (95% CI) to account for performance variability in distributed systems. To visualize the distribution of latencies across different scaling configurations, we will utilize Empirical Cumulative Distribution Function (ECDF) plots, to reveal performance patterns and tail latencies. ~~In order to validate the statistical significance of performance differences between vertical and horizontal scaling approaches, we will conduct paired t-tests where appropriate.~~ We are going to run each benchmark configuration multiple times to ensure statistical reliability.