Enterprise Platform and Integration Concepts
Research Group of Prof. Dr. h.c. mult. Hasso Plattner

Master's Project Winter Term 2022/2023

Extend Your Own Database

Larger-Than-Memory Data Management for an In-Memory Database System

Description

Relational in-memory database systems achieve a high query processing performance by storing all their data in DRAM, which provides a lower data access latency than disks. However, DRAM is still relatively expensive compared to other storage technologies such as modern SSDs. Therefore, for cost-effectiveness and to avoid potential DRAM capacity limitations, we may want to store some parts of the data on secondary storage devices, resulting in larger-than-memory database systems. Two common approaches for implementing larger-than-memory databases are either having a buffer manager or using memory-mapped file I/O, e.g., via the OS-provided `mmap` command.

For small and mid-size data sets, the performance of Hyrise is competitive with that of comparable systems such as MonetDB, DuckDB, HyPer, and Umbra. Now we want to move toward processing terabytes of data. In this project, we will extend our database system Hyrise from a pure main memory to a larger-than-memory database system using the memory-mapped file I/O approach. After you have been introduced to the most important components of Hyrise by your supervisors and have familiarized yourself with the codebase, we will first focus on implementing a mechanism to persist table data on SSDs and load the stored data into the main memory efficiently. Second, we will evaluate different libraries for memory-mapped file I/O, including their page fault handling, to identify a particularly well-suited library for the targeted database workloads.

We aim for results that can be integrated into the main code base and push forward the open-source Hyrise project. After this project, there will be research and engineering opportunities to dive deeper into identified issues in the form of student assistantships, master's theses, and Ph.D. positions.

Existing Infrastructure

We will not waste any time in setting up dependencies, as most of the setup needed for this project already exists. This includes

- a database system that can execute most queries out of the box,
- a benchmark framework, which automatically executes queries in parallel, tracks their execution time and reports the results,

- scripts for comparing multiple benchmark runs, e.g., for tracking the improvement made by a commit; as well as scripts that plot the throughput with varying number of CPUs or the improvement over time,
- a code base with a high degree of test coverage (>90%) and a CI server that enforces code quality for all pull requests, and
- a mechanism to load and store tables as binary data, which might be partially reused for persisting table data on an SSD

Learning Goals

Through successful completion of this project, you will:

- Improve your programming and teamwork skills
- Learn to familiarize yourself with and work on an existing large software project
- Learn to identify and eliminate performance bottlenecks
- Learn to perform experimental evaluations
- Deepen your database and memory management knowledge
- Improve your research methodology and academic writing

Prerequisites

Prior understanding of the fundamentals of databases (e.g., from the database systems lecture or the *Develop your own Database* seminar) is expected as well as knowledge of C++.

Initial References

- Radu Stoica and Anastasia Ailamaki: *Enabling Efficient OS Paging for Main-Memory OLTP Databases*. DaMoN 2013
- Lin Ma et al.: *Larger-than-Memory Data Management on Modern Storage Hardware for In-Memory OLTP Database Systems*. DaMoN 2016
- Andrew Crotty et al.: *Are You Sure You Want to Use MMAP in Your Database Management System*? CIDR 2022
- Ivy Peng et al.: *UMap: Enabling Application-driven Optimizations for Page Management*. MCHPC 2019
- Anastasios Papagiannis et al., *Memory-Mapped I/O on Steroids*. EuroSys 2021

Contact

You are welcome to contact one of us via mail (<firstname>.<lastname>@hpi.de) or visit us in the villa.

Marcel Weisgut, Stefan Halfpap, Martin Boissier, Dr. Michael Perscheid