Information Systems
Research Group of Prof. Felix Naumann

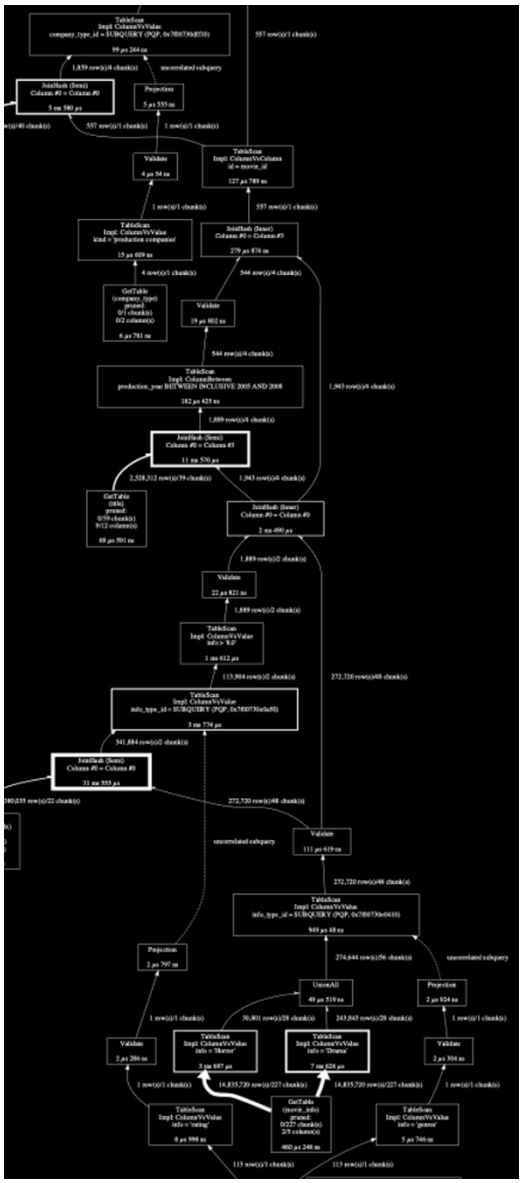
Master's Project Summer Term 2023

Boost Query Performance with Dynamic Pruning

Accelerating Query Execution for Data-induced Predicates and Dependency-based Optimization

Description

State-of-the-art database management systems (DBMSs) efficiently process large amounts of data by storing and processing them purely in memory, applying sophisticated query optimization techniques, using massive parallelization, or a combination of these approaches. The queried datasets are often large and contain *data dependencies*. An example of such a data dependency is the unique column combination (UCC) on the students' matriculation number: by definition, there are no duplicates. While research proposes numerous dependency-based optimization techniques, DBMSs actually apply only few.



This project will provide a deeper insight into database internals and dependency-based query optimization. *Pruning*, i.e., excluding irrelevant data from processing, saves I/O and computation. Especially in cloud environments, transferring data is time-consuming and costly. Dependency-based optimizations can, e.g., rewrite joins to linear scans on the result of a subquery. However, the subquery's result is not known beforehand, and *dynamic pruning* during the query execution is required. Together with *data-induced predicates* (diPs) that pre-filter joined tables, dynamic pruning accelerates query execution and speeds up dependency-based optimizations. Thus, we will investigate in which cases and workloads diPs are valuable or where they have drawbacks, adding dependencies to the execution order of operators.

We shall implement and measure the impact of dynamic pruning and data-induced predicates with and without dependency-based optimization techniques in the in-memory DBMS Hyrise.

The Hyrise DBMS

Your experiments will be implemented for and evaluated with the open-source in-memory DBMS Hyrise. Most of the setup needed for this project already exists. This includes:

- A database system that can execute most queries of industry benchmarks out of the box, which has functionality to handle data dependencies in query plans and automatically use and discover UCCs for query optimization in a workload-driven fashion
- A benchmark framework that automatically executes queries in parallel, tracks their execution time and reports the results, including scripts for comparing multiple benchmark runs
- A code base with high test coverage (>90%) and a CI pipeline that enforces code quality for all pull requests.

Learning Goals

Through successful completion of this project, you will:

- Pass the entire cycle of a research project, from formulating initial research questions, to prototyping and conducting experimental results, to deriving novel insights and formulating them in a scientific text
- Improve your programming and teamwork skills
- Learn to familiarize yourself with and work on an existing large software project with a focus on data systems and high-performance data processing
- Deepen your database and query optimization knowledge
- Improve your research methodology and academic writing

At the end of the project, we will condense our insights and results into a research submission to be submitted to a top database conference.

Prerequisites

- Prior understanding of the fundamentals of databases (e.g., from the database systems lectures or the *Develop your own Database* seminar)
- Familiarity with (modern) C++

Initial Related Work

[1] Laurel J. Orr et al. *Pushing Data-Induced Predicates Through Joins in Big-Data Clusters*. PVLDB 13(3) (2019). DOI: 10.14778/3368289.3368292

[2] Jan Kossmann et al. *Data dependencies for query optimization: a survey*. VLDBJ 31(1) (2022). DOI: 10.1007/s00778-021-00676-3

[3] Jan Kossmann et al. *Workload-driven, Lazy Discovery of Data Dependencies for Query Optimization*. CIDR 2022. URL: cidrdb.org/cidr2022/papers/p70-kossmann.pdf

[4] Markus Dreseler et al. *Hyrise Re-engineered: An Extensible Database System for Research in Relational In-Memory Data Management*. EDBT 2019. DOI: 10.5441/002/edbt.2019.28

Contact

You are welcome to contact Prof. Dr. Felix Naumann (felix.naumann@hpi.de), Daniel Lindner (daniel.lindner@hpi.de), or Martin Boissier (martin.boissier@hpi.de) or to visit us in the F building, second floor (on Campus II).