Masterthesis@SAP: Shared SIMD Column Scan

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Motivation

With the SAP HANA® database, SAP has created a completely new database technology for the current generation of business applications that combines transactional (OLTP) and analytical (OLAP) data processing. Sharing Ressources is already established for OLAP queries [2, 1]. Though some literature looks into mixed OLAP/OLTP workloads (e.g. SharedDB), the focus has never been on OLTP queries, alone. Within this project, we would like to investigate into the novel field of resource sharing for OLTP queries and the limits of its possibilities: incoming queries are therefore batched into a big query, which is executed. The aggregated result is then split and in the parts are returned to the owners of the merged queries. Currently, splitting the result is done on top of the execution plan, the goal however, is to push the decision, which result-tuple belongs to which query, down into the data-access operator.

In this project, we would like to develop together with SAP a shared SIMD column scan operator for OLTP queries. A SIMD column scan operator loads and decompresses the values, compares it and returns a bitvector with bits indicating for each column, whether that result tuple is in the result set or not. By comparing the results with multiple attributes at once, loading the values into cache and decompressing can be shared. Multiple bitvectors (one for each shared query) are returned, instead.

Task and Requirements

- Goal: Investigate design options for a shared SIMD column scan operator for OLTP.
- Part 1: Design and implement different strategies for a shared SIMD column scan operator using Intel's AVX extensions.
- Part 2: Evaluate the performance using benchmarks such as YCSB.
- Requirements: Strong C/C++ programming skills, background in database internals and ideally AVX/SIMD instructions.

Contact

- Robin Rehrmann (SAP): r.rehrmann@sap.com
- Prof. Carsten Binnig (Data Management): carsten.binnig@cs.tu-darmstadt.de

Literatur

[1] O. Polychroniou et al. Rethinking SIMD vectorization for in-memory databases. In *ACM SIGMOD*, pages 1493–1508, 2015.

[2] T. Willhalm et al. Simd-scan: Ultra fast in-memory table scan using on-chip vector processing units. *PVLDB*, 2(1):385–394, 2009.