DIAS: Data-Intensive Applications and Systems Laboratory
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne
Building BC, Station 14
CH-1015 Lausanne
URL: http://dias.epfl.ch/

Cost models & tuning the query optimizer of a JIT analytical engine

Keywords: Query optimization, cost models, relational operators

Problem: Analytical DMBS engines rely on their query optimizer to pick an efficient query plan for execution. To select the more appropriate query plan, query optimizers apply a set of optimization rules to transform the candidate plans into more efficient ones. Usually, the efficiency is evaluated based on cost models for each node of the plan.

Project: The goad of this project is for the student to become familiar with Calcite [1] a query optimizer used by many DBMS systems and model the different operators of Proteus [2], an in-house DBMS engine for analytical queries. The student will create cost models for different operators used in Proteus and evaluate the performance of the selected queries, the accuracy of the models as well as the performance of the query optimization itself.
As an outcome of the project, the cost models incorporated in the query optimizer should make it able not only to argue about the physical operators but also about the parallelization and resource allocation for the query. The student, will teach the optimizer through the cost models to make decisions about the parallelization of the query. In addition, the models should be flexible enough to take into consideration the constant variation in available resources, which result from other concurrently running workloads. Lastly, as the characteristics of modern servers greatly vary from machine to machine due to the many different configurations, the last part of the project will focus on creating an auto-tuning infrastructure that will be able to select the hyper-parameters of the cost-models for Proteus to adapt to unknown machines.

Plan:

1. Become familiar with Calcite [1]
2. Write cost models for simple queries and make then consistent across different small variations of the queries
3. Expand into cost models that take into consideration heterogeneity and/or parallelism
4. Work on auto-tuning the cost models for different machines

Supervisor: Prof. Anastasia Ailamaki, anastasia.ailamaki@epfl.ch
Responsible collaborator(s): Hamish Nicholson, hamish.nicholson@epfl.ch
Duration: 14 weeks

References

[1] E. Begoli et al. Apache calcite: a foundational framework for optimized query processing over heterogeneous data sources. In SIGMOD, pages 221–230. ACM, 2018.
[2] M. Karpathiotakis, I. Alagiannis, and A. Ailamaki. Fast Queries Over Heterogeneous Data Through Engine Customization. PVLDB, 2016.