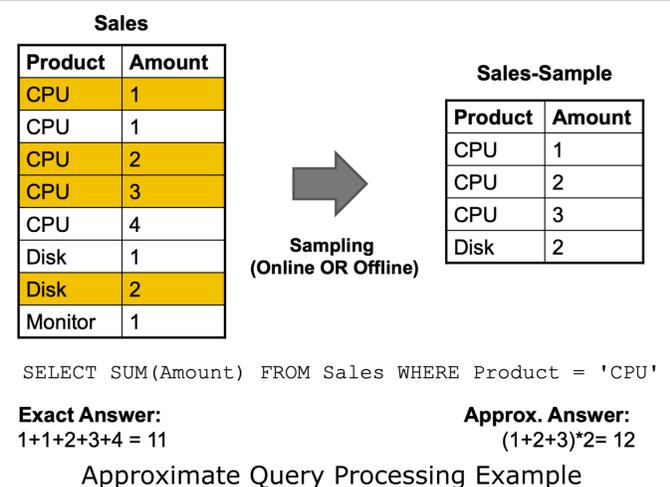# Approximate Query Compiler – Research Proposal

Abstract

Interactive data analytics allows users with limited technical knowledge to explore large sets of data interactively to gain insights. To interactively dive into data, the results for queries need to be available right away. This poster proposes a combination of a query compiler with an approach to achieve interactively results by approximating the result from a subset of a given dataset, decreasing the response time compared to traditional approaches.

Goal

Approximate Query Processing (AQP) provides a fast approximated answer for a query on large data sets, by sampling it into smaller datasets calculating the results and scaling it up. The goal of this research project is to create an AQP pipeline based on a query compiler. The pipeline compiles a SQL query into a native binary while adding the approximation functionality. The proposed solution aims to be faster then existing AQP pipelines.

Sales

Product	Amount
CPU	1
CPU	1
CPU	2
CPU	3
CPU	4
Disk	1
Disk	2
Monitor	1

Sampling
(Online OR Offline)

Sales-Sample

Product	Amount
CPU	1
CPU	2
CPU	3
Disk	2

```
SELECT SUM(Amount) FROM Sales WHERE Product = 'CPU'
```

Exact Answer:
1+1+2+3+4 = 11

Approx. Answer:
(1+2+3)*2= 12

Approximate Query Processing Example

Problem

Vizdom, an existing tool for interactive data analytics, allows users with limited technical knowledge to create visually complex workflows for machine learning and statistics. Studies [1] showed that a response time higher than 500ms already limits the exploration space and productivity of the user. To accomplish this on big datasets, Vizdom currently uses **IDEA**, an AQP middleware on top of Big Data engines. While IDEA works well for simple SQL queries, it struggles to achieve the required speed for more complex queries.

Solution

Approximate Query Compiler (AQC), a drop-in replacement for IDEA, compiles a SQL query into a native binary, adding approximation functionality. Based on the approach presented in [2] an SQL interpreter creates a Scala query. The **Approximation Query Optimizer** adds the required approximation functionality into the generated Scala query, which is then compiled into C with the open-source LMS Framework (Lightweight Modular Staging). AQC sends the compiled C query to the database, which sends the approximated result to Vizdom.

AQC in Action
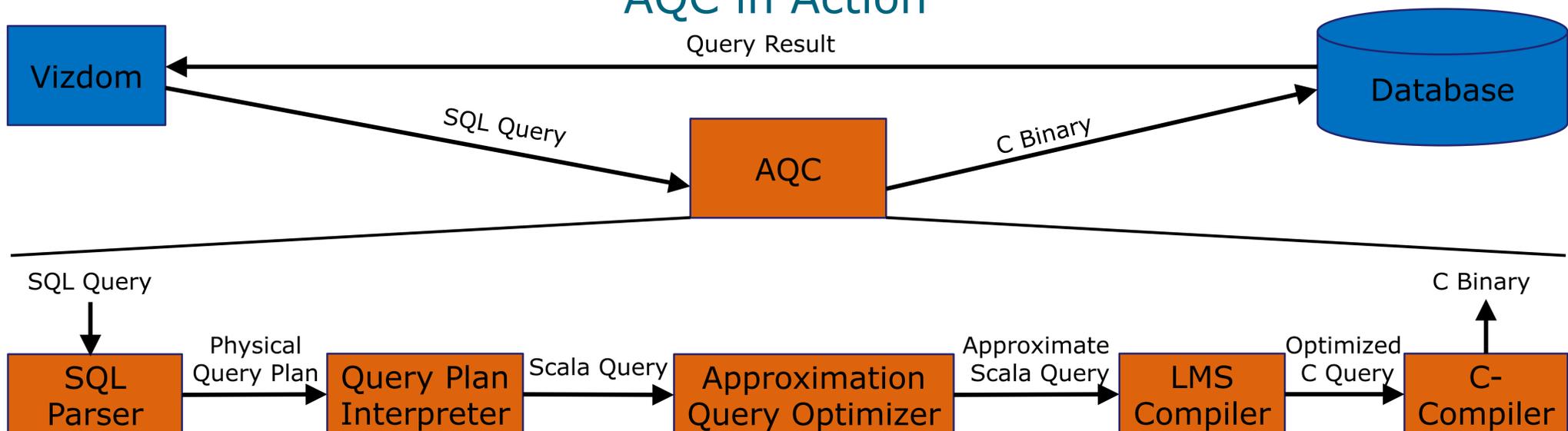
Maximilian Schall

Master IT-Systems Engineering
Hasso Plattner Institute, Potsdam, Germany

E-Mail: Maximilian.Schall@student.hpi.de
Poster for Lecture Series in Practical Data Engineering
Prof. Dr. Tilmann Rabl, Prof. Dr.Felix Naumann

References:

[1] Liu, Z., & Heer, J. (2014). The Effects of Interactive Latency on Exploratory Visual Analysis. IEEE Transactions on Visualization and Computer Graphics, 20, 2122-2131.
[2] Tiark Rompf and Nada Amin. 2015. Functional pearl: a SQL to C compiler in 500 lines of code. SIGPLAN Not. 50, 9 (August 2015), 2–9.
DOI:https://doi.org/10.1145/2858949.2784760

Connection to the Lecture:

Vizdom, IDEA, AQP, AQP Example: Towards Interactive Data Analytics, Carsten Binnig, TU Darmstadt
Query Compiler and LMS Framework: A Programming Language and Compiler View on Data Management and Machine Learning Systems, Tiark Rompf, Purdue University

HPI Hasso Plattner Institut