Asymptotically Better Query Optimization Using Indexed Algebra

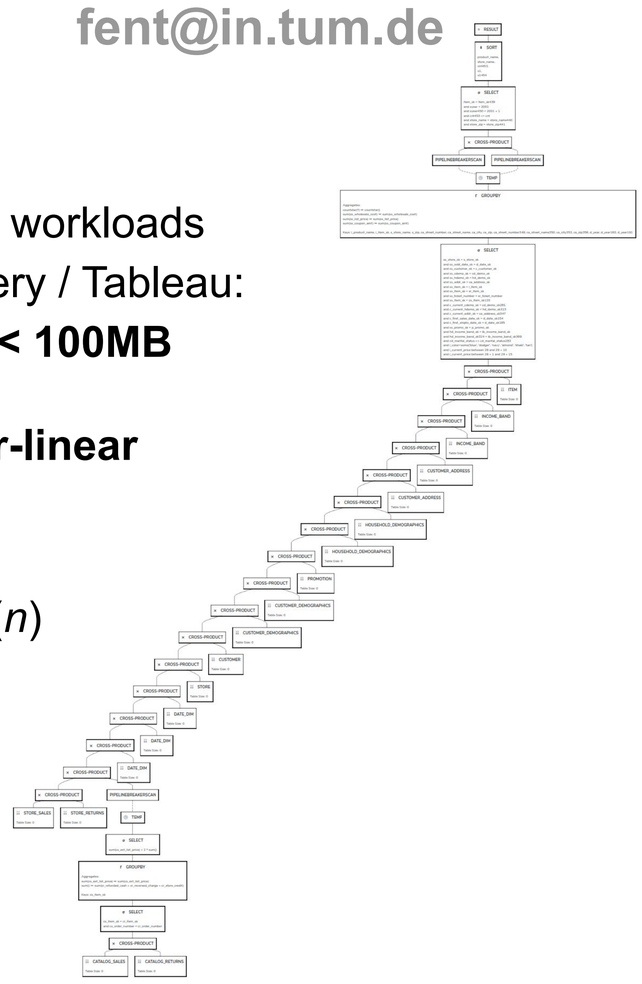
Philipp Fent, Guido Moerkotte, Thomas Neumann
fent@in.tum.de

Motivation

Complex queries on small workloads are common. E.g., BigQuery / Tableau:
90% of queries process < 100MB

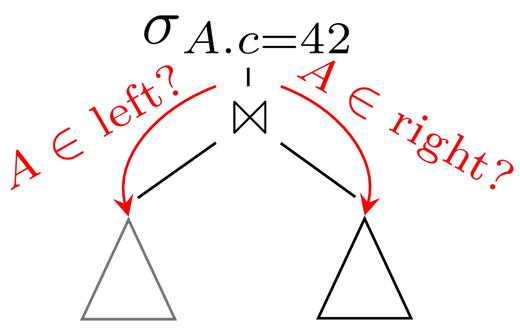
Optimization time is **super-linear** with algebra depth!

Algebra trees can have $O(n)$ Depth. Especially before Optimization.
E.g., TPC-DS Q64 →

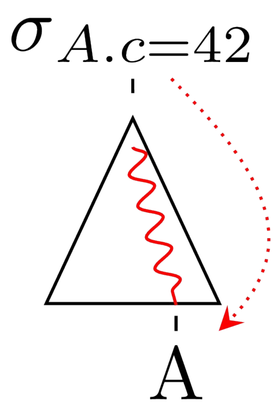


Query Optimization

Analyze algebra to find optimization opportunities



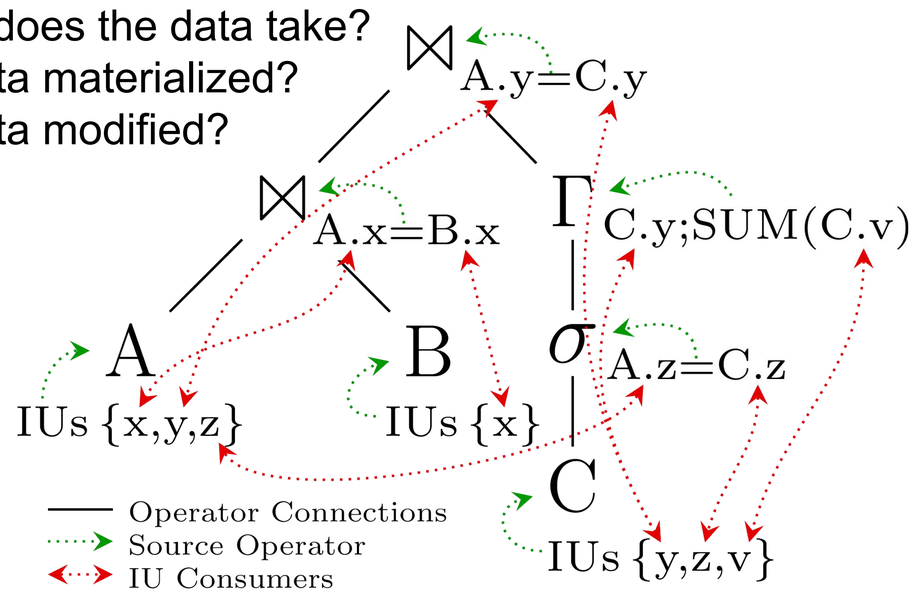
Operator-centric
$O(n^2)$



Path-centric
$O(n \log(n))$
with Indexed Algebra

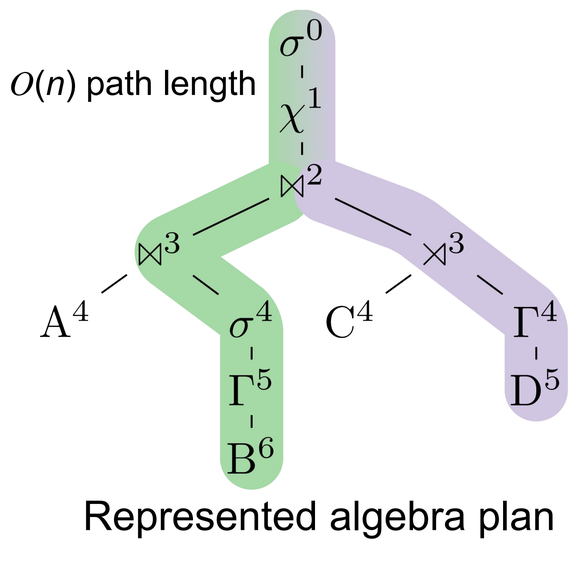
Algebra to analyze data-flow for optimization

Which path does the data take?
Where is data materialized?
Where is data modified?



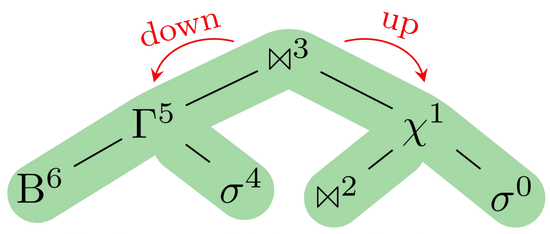
Indexed Algebra

Index of paths through the algebra

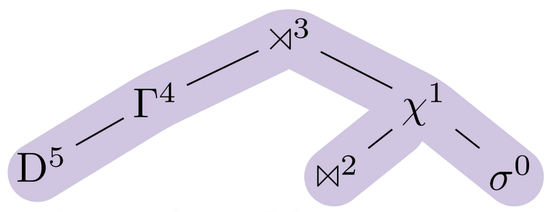


Represented algebra plan

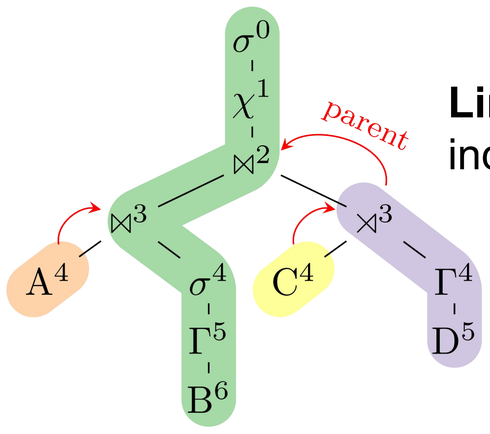
$O(\log(n))$ index height



Balanced binary index of the path from B^6 to the root



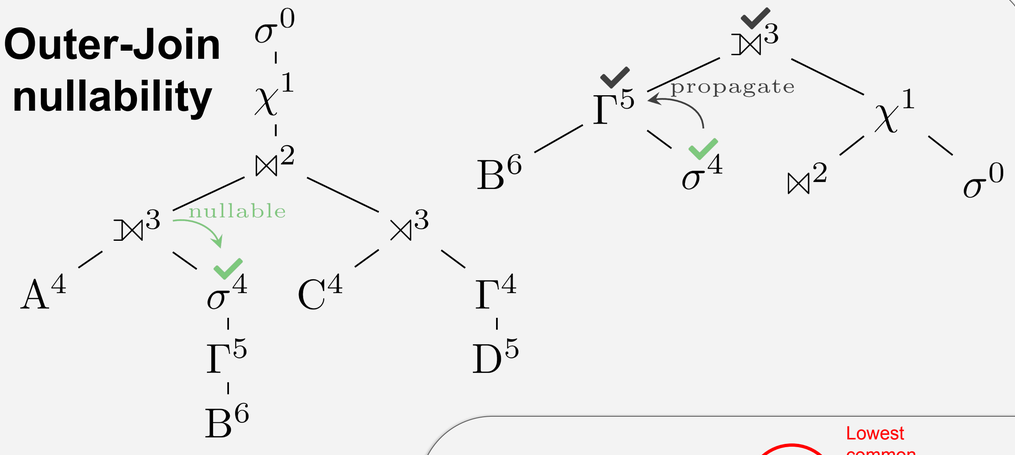
Index from D^5 to the root



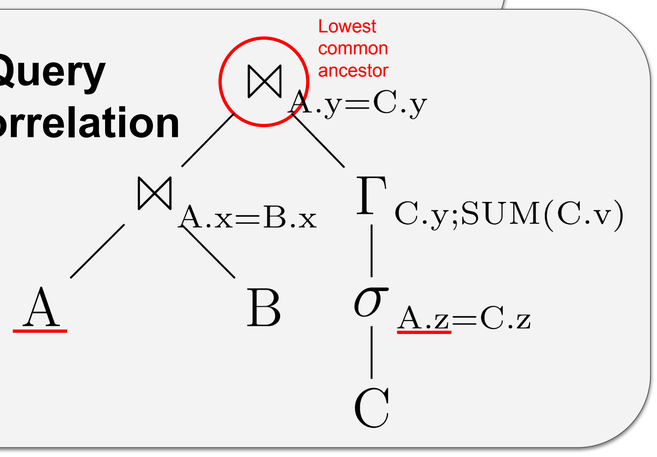
Link/cut trees make indexing efficient

Path-centric optimization

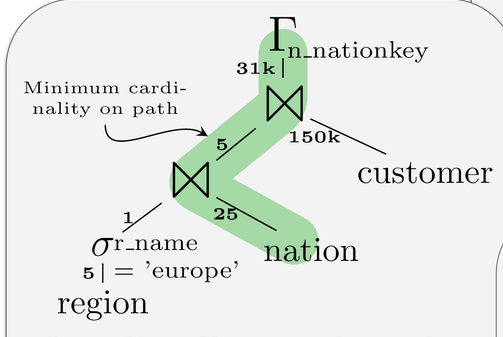
Outer-Join nullability



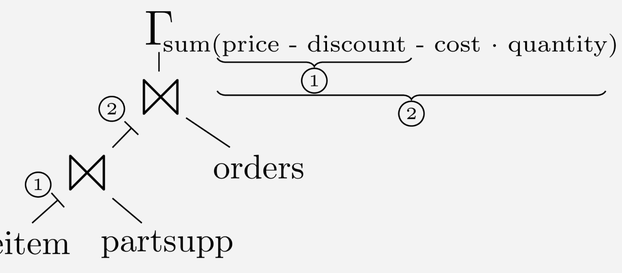
Query decorrelation



Cardinality estimation

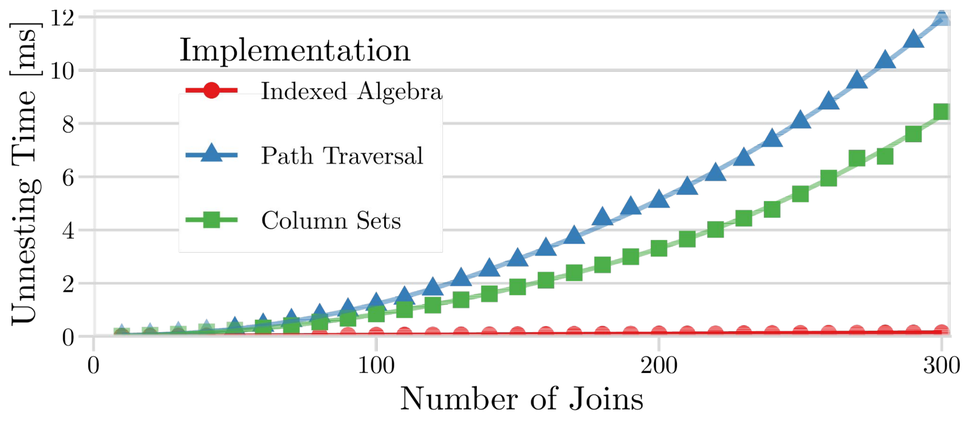


Expression evaluation

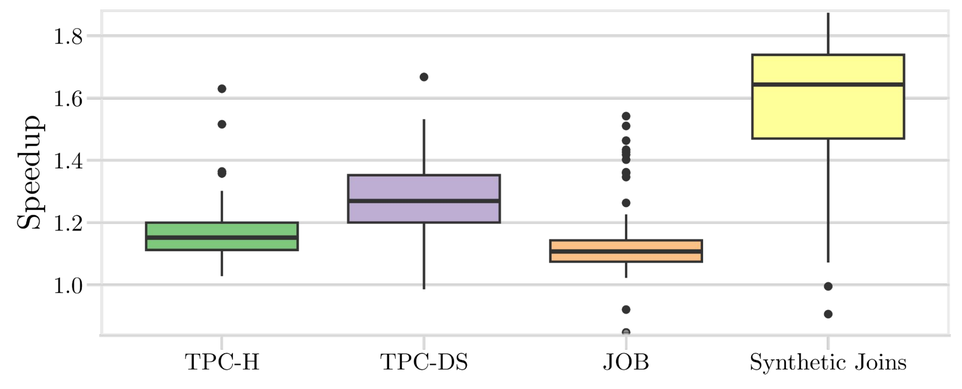


Evaluation

Asymptotic behaviour



Optimization time



VLDB2023 vancouver