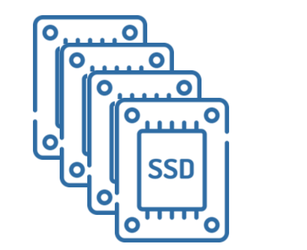
HPCache: Memory-Efficient OLAP through Proportional Caching

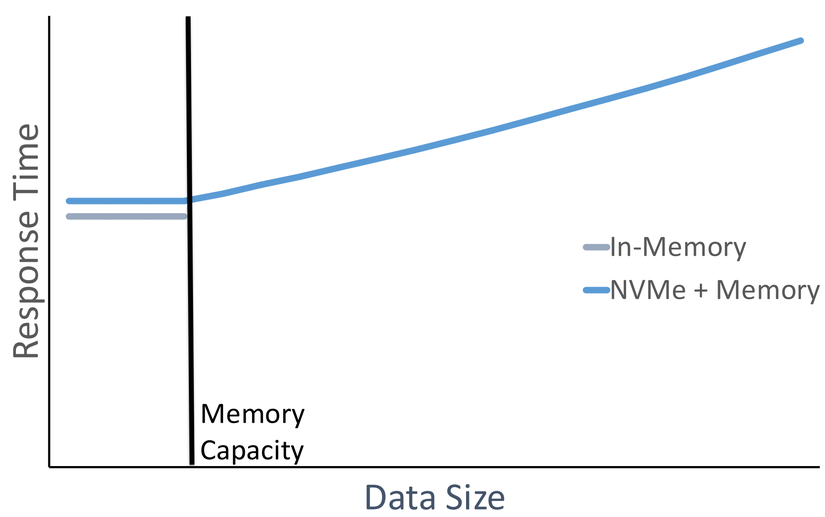
Hamish Nicholson, Periklis Chrysogelos and Anastasia Ailamaki

Fast Analytics Demands Memory

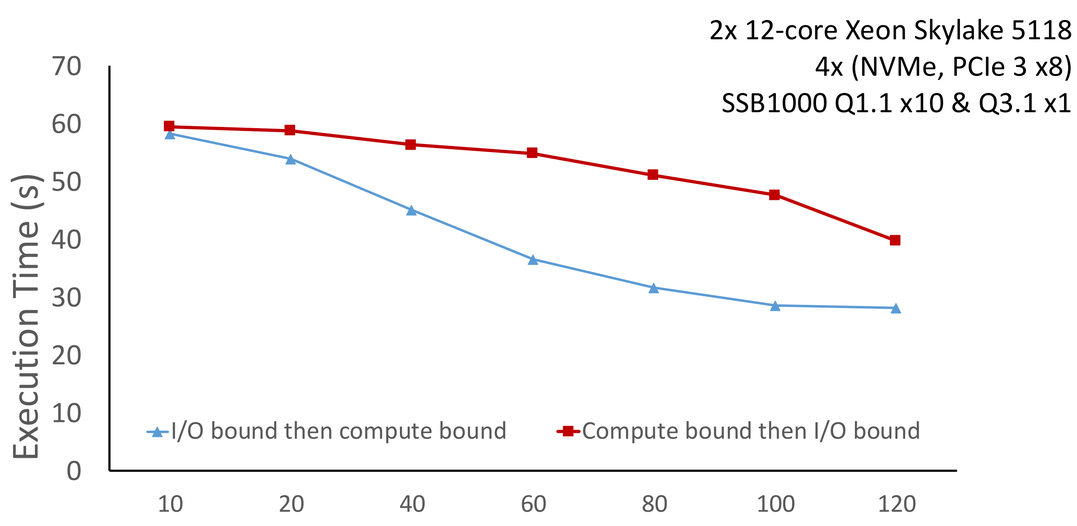


~10-100s GB/s

~100s GB/s

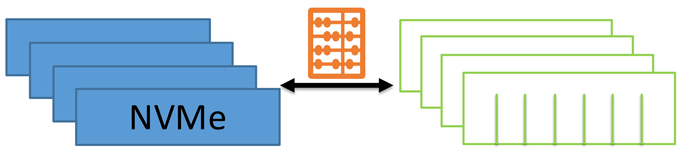


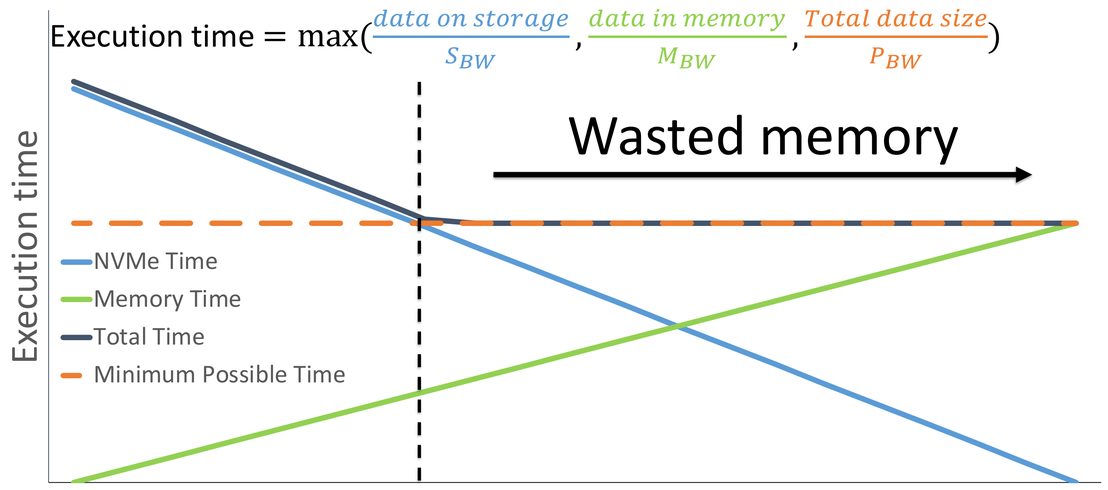
NVMe Requires New Caching Policies



Need Efficient Memory Use to Scale to Large Datasets

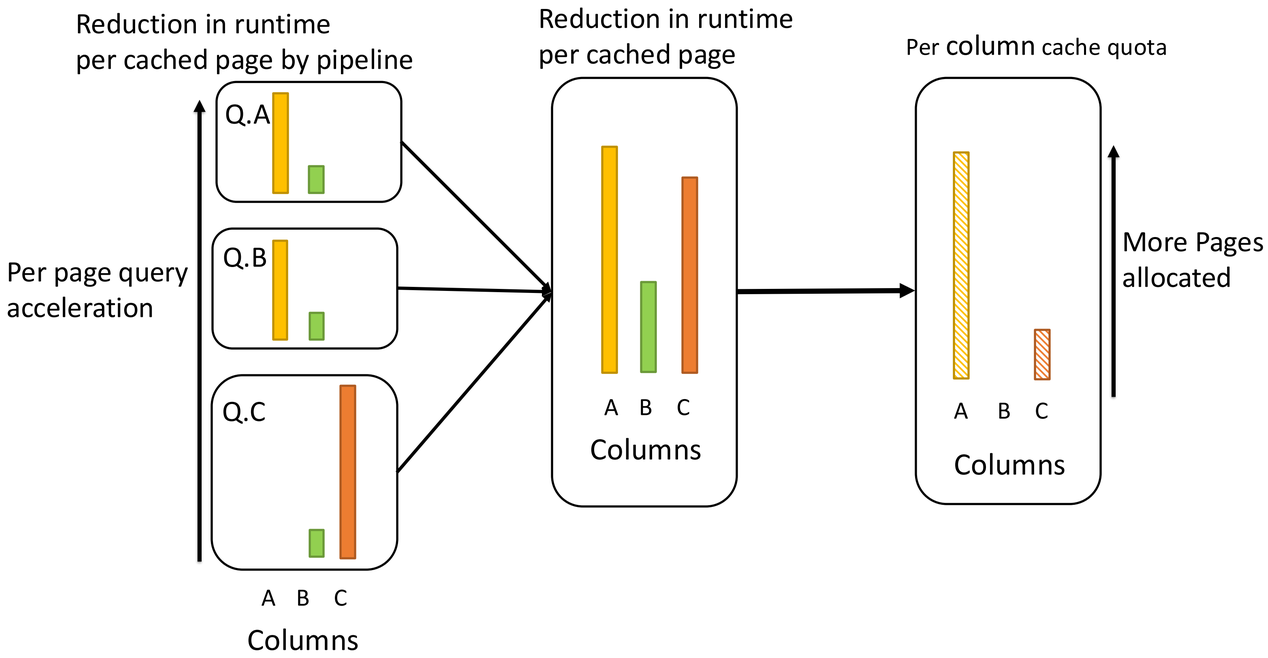
Match Scan Time to Processing Time





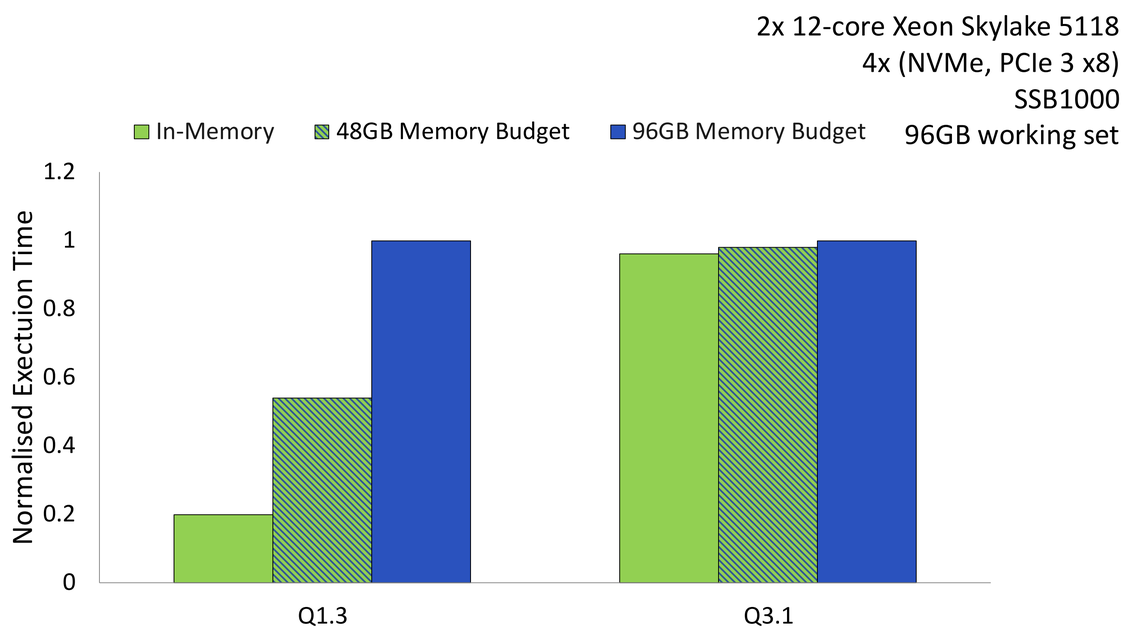
Cache Less Data for the Same Performance

Constrained Memory Budgets



High-bandwidth Storage → Opportunity for Better Caching

Match Processing Time for In-Memory and On-Storage inputs

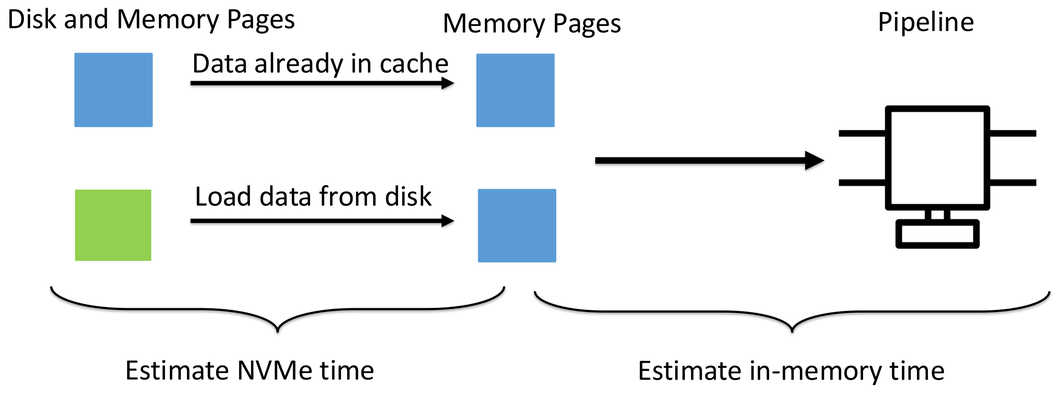


Caching without Query-Info Wastes Performance

Determine Pipeline Throughput at Runtime

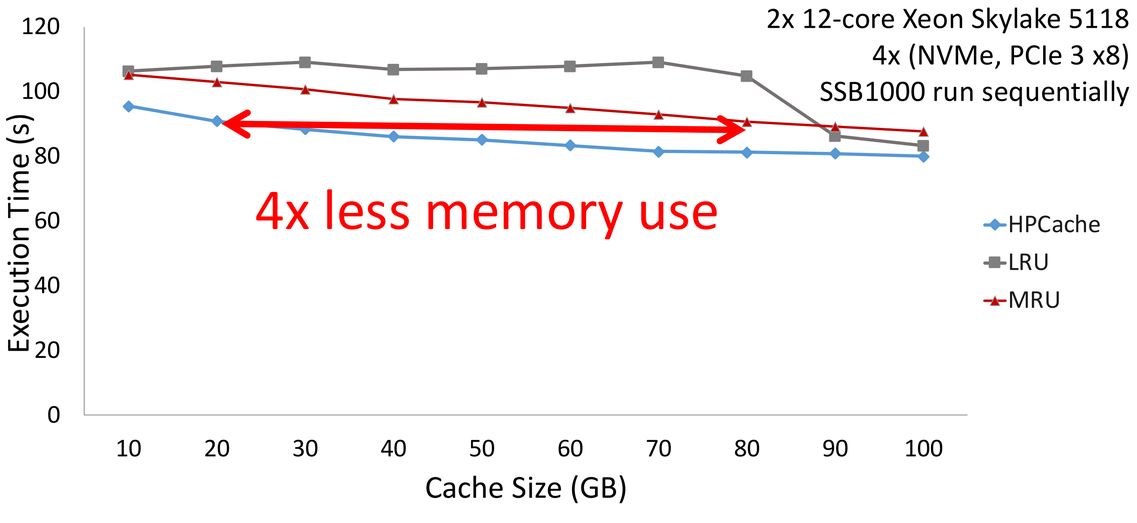
Observe pipeline invocation time

- Derive good estimate of in-memory execution time
- Mostly independent of current data placement



Cache Data to Minimize Workload Execution Time

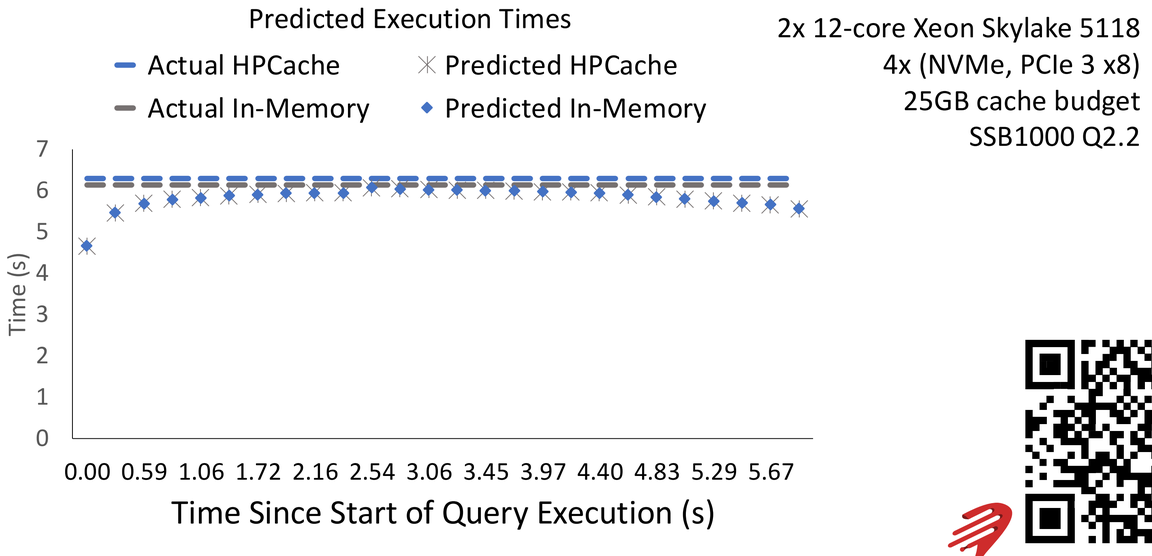
Improved Memory Efficiency



75% Less Memory Use Through Proportional Caching

Predict In-Memory Performance for Any Data Placement

Accurate Runtime Performance Predictions



Average Prediction Within 15% Through Runtime Prediction

proteusdb.com