Evaluating Robustness of Workload-Aware Partitioning Schemes

Sumer Rathinam, Jake Bloomfeld, Keith Lovett
Overview

Introduction → Design → Evaluation

Background → Experiment → Conclusion
1. Introduction
How do Workloads Vary?

- By type...
- By number...
- By position...
Problem

Workload Statistic

Optimal Partitioning Scheme

Robustness of Solution
2. Background
Casper: Optimal Workload-Aware Column Layout for HTAP

A storage engine that navigates the design space of the leading column’s physical layout
3. Design
Design Process

Baseline

Input:
$W_1$
$P_1$
Output:
$C_1$

Noisy w/ Baseline

Input:
$W_2$
$P_1$
Output:
$C_2$

Noisy w/ Optimal

Input:
$W_2$
$P_2$
Output:
$C_3$
Effects of Noise

\[ \% \Delta = \frac{f(W_2, P_2) - f(W_2, P_1)}{f(W_2, P_2)} \]

Evaluates the % change of the cost of running the "noisy" workload with the baseline partitioning scheme, compared to running that same workload with its optimal partitioning scheme.
4. Experiment
Counts Experiments

Insert Queries:
- Increase total counts in 5% intervals
- Up to 25% of baseline
- Keep position the same

Point Queries:
- Increase total counts in 5% intervals
- Up to 25% of baseline
- Keep position the same
Position Experiments

Insert Queries:
- Increase total position ranges by 5%
- Decrease total position ranges by 5%
- Keep total counts the same

Point Queries:
- Increase total position ranges by 5%
- Decrease total position ranges by 5%
- Keep total counts the same
5. Evaluation
Insert Queries

Insert Query Counts Performance

Insert Query Position Performance
Point Queries

Point Query Counts Performance

Point Query Position Performance
6. Conclusion
Casper is more robust when dealing with a change in total number of a certain query rather than a change in where that query occurs.
Thanks!

Any questions?