DeepOLA: Online Aggregation for Deeply Nested Queries

Nikhil Sheoran
University of Illinois at Urbana-Champaign

Introduction

- AQP techniques provide fast but "accurate" enough results for aggregate queries.
- Online Aggregation (OLA) is an AQP technique where estimates continuously improve.
- Eventually provides correct results.
- Lack of general framework to handle OLA for complex deeply nested queries.
- We provide DeepOLA, a generalized framework and implementation to perform Online Aggregation for arbitrarily nested queries.

DeepOLA: Framework

- Incremental Dataframe
  - Append (DA): Rows are only appended.
  - Merge (DM): Rows can be update/append.
  - Complete (DC): No change.
- Incremental Operations
  - Define merge operation for each operation R
  - $R(I \cup \Delta I) = merge(R(I), R(\Delta I), c)$

Operation	Input	Output
Filtering (Π)	{DA, DM, DC}	{DA, DM, DC}
Projection (σ)	{DA, DM, DC}	{DA, DM, DC}
JOIN or UNION	DA, {DA, DM, DC}	{DA, DM, DA}
JOIN or UNION	DM, {DM, DC}	{DM, DM}
JOIN or UNION	DC, DC	DC
GROUP BY AGG	{DA,DM,DC}	{DM,DM,DC}

DeepOLA: Query Processing

- Parse the input query into a DAG.
- **Mergeable Nodes**
  - A node n is mergeable if no node in the paths from the root nodes (including) to that node (excluding) are of type DM. For example:
  - FROM, WHERE, GROUP-BY-AGG are mergeable.
  - SELECT is not mergeable.
- **Query Processing**
  - Compute operations on the new partition of data, ΔT till the *last mergeable* node.
  - Merge the *previous result*, $R(I)$ and result on ΔT, $R(\Delta T)$ at the *last mergeable* node.
  - Recompute operations of all nodes after this node.
- A *nested sub-query* is a *sub-graph* of the DAG.
  - The evaluation methodology remains the same.
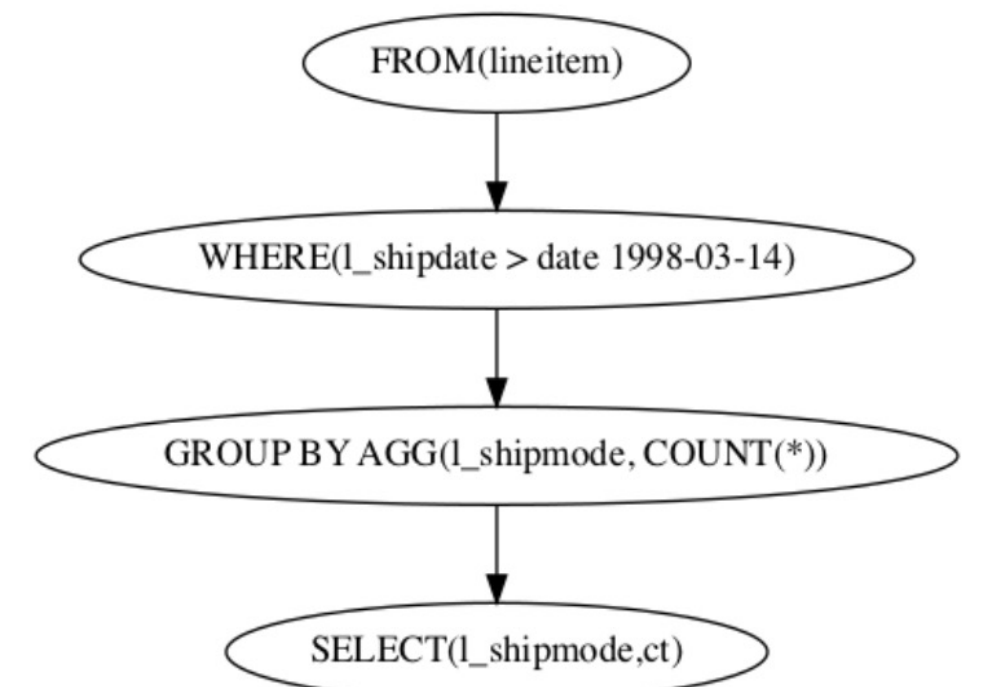
Experiment: Results

- TPC-H Dataset. Scale = 1.
- Compared against Postgres with indexes.
- Optimizations
  - Re-use of hash-table across partitions.
  - Merge join of ordered tables.
  - Concurrent execution.
- Analysis of Query 5
  - 4 Hash Joins: Re-use of hash-tables across partitions.
  - 1 Merge Join
- 1st Result is the result after processing first partition (of fixed size of 100K rows)

Query	Postgres (in ms)	DeepOLA (in ms)	DeepOLA 1st Result (in ms)	Relative Error 1st Result (%)
q1	2307	3560	203	1.65%
q5	13521	2620	226	10.1%
q6	1668	2490	72	5.68%
q12	765	2470	79	7.76%
q14	640	2550	238	11.8%