A LINEAR ALGEBRA APPROACH TO OLAP

Rogério Pontes

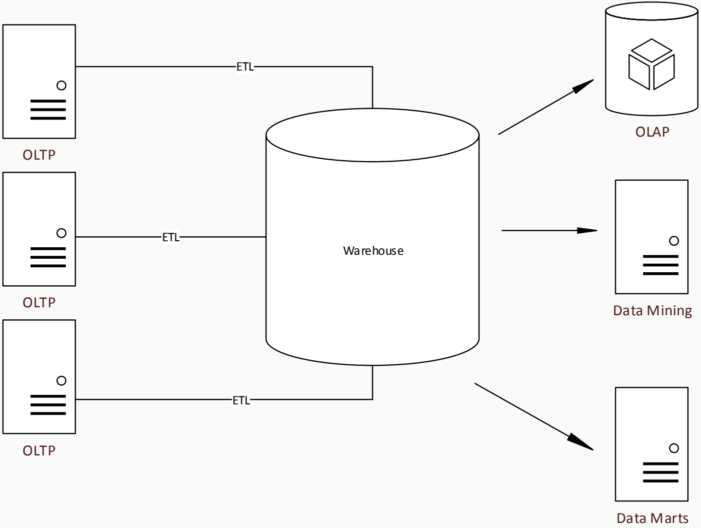
December 14, 2015

Universidade do Minho

HASLab HIGH-ASSURANCE SOFTWARE LABORATORY

INESCTEC TECHNOLOGY & SCIENCE ASSOCIATE LABORATORY PORTUGAL

DATA WAREHOUSE



OLAP

Online analytical processing (OLAP) systems, perform multidimensional analysis of business data and provides the capability for complex calculations, trend analysis, and sophisticated data modeling.

OLAP APPLICATIONS

- Multidimensional OLAP
  - Most efficient.
  - Require more storage.
  - Requires additional investment.
- Relational OLAP
  - Uses all the developments made on relational databases.
  - Least efficient.
  - One example is Hive.
- Hybrid OLAP
  - Has the advantage of both solutions.
  - It is more complex as it deals with MOLAP and ROLAP.

HASLab
HIGH-ASSURANCE SOFTWARE LABORATORY

Main Issue

All the solutions depend on Relational Algebra.

- Relational Algebra Lacks algebraic properties on the most common operations.
- Hard to frame OLAP operators in Codd's Relational Algebra.
- Relational Algebra does not provide qualitative and quantitative proofs for all the relational operator.

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

Typed Linear Algebra

An Algebra capable of calculation the OLAP operation with equations that can provide a formal proof both on the quantitative side and the qualitative side.

Projection Function creates the relation between the attributes and their original position in the table.

Model	*Year*	*Sales*
Chevy	1990	5
Chevy	1990	87
Ford	1990	64
Ford	1990	99
Ford	1991	8

$\xrightarrow{\textit{Convert}}$

t_{Model}	0	1	2	3	4
Chevy	1	1	0	0	0
Ford	0	0	1	1	1

Measure Matrices store the numeric values in a Diagonal Matrix.

Model	*Year*	*Sales*
Chevy	1990	5
Chevy	1990	87
Ford	1990	64
Ford	1990	99
Ford	1991	8

$\xrightarrow{\textit{Convert}}$

$[\![t]\!]_{Sales}$	0	1	2	3	4
0	5	0	0	0	0
1	0	87	0	0	0
2	0	0	64	0	0
3	0	0	0	99	0
4	0	0	0	0	8

HASLab
HIGH-ASSURANCE SOFTWARE LABORATORY

OLAP Operations in Linear Algebra

Projection functions (t_A), Measure Matrices ($[\![t]\!]_M$) and Matrix multiplications form the building blocks of the Typed Linear Algebra.

$$ctab^{Sales}_{Model \leftarrow Year}(T) : |Model| \leftarrow |Year|$$

$$ctab^{Sales}_{Model \leftarrow Year}(T) = t_{Model} \cdot [\![t]\!]_{Sales} \cdot t_{Year}$$

$t_{Model} \cdot [\![t]\!]_{Sales} \cdot t_{Year}$	1990	1991
Chevy	92	0
Ford	163	15

HASLab HIGH-ASSURANCE SOFTWARE LABORATORY

Main goal

Implement and evaluate the performance of a Typed Linear Algebra solution in a real world scenario. Can linear algebra provide a more efficient solution ?

Challenges

- Matrix sparsity. In the worst case only 0.1% of the matrix has non-zero values.
- Attribute-Range Problem. An attribute must always be in the same row in every matrix.
- Conversion of complex SQL queries to LA equations and implementation on a distributed system.

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

From the many standard sparse matrix representations, two of them were selected as the most appropriate to our problem.

- Compressed Sparse Column
  - Used to represent Projections Functions and Measure matrix.
  - Standard formats uses 3 or 4 arrays to store the data.
  - Due to the properties of linear algebra matrices we improved the format to use a single array.
- Coordinate Format
  - Used to exchange values between the computing nodes.

Problem

Typed linear algebra works with the premiss that the relation between Attribute and Matrix line number is bijective. How can two concurrent machines, that are generating a matrix, assign the same line number to the same attribute ?

Solution

Using a 64base encoding, its possible to assign a unique id to every Attribute.

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

Problem

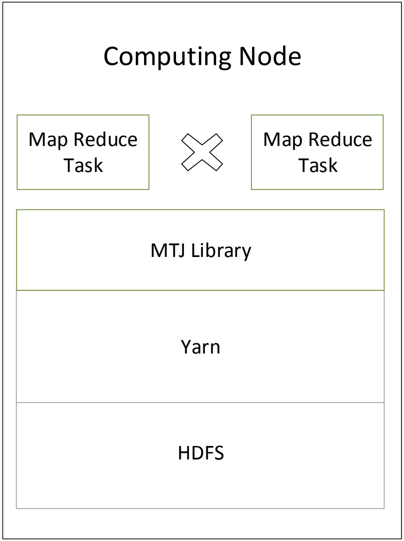
Typed linear algebra initial conception only operated with queries on a Single Table. Real Word query benchmarks are far more complex.

Solution

Typed linear algebra was extended to support the relational algebra projection, restriction and joins.

HASLab
HIGH-ASSURANCE SOFTWARE LABORATORY

ARCHITECTURE



CLUSTER SETUP

The cluster is made of 5 Machines, one server is dedicated to manage the cluster, the other four are the computing nodes. Each machine is running Ubuntu 14.04 64 bit on Intel Core i3-3240 @ 3.40 Ghz, 3K cache and 8GB of RAM.

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

Objective

We seek to evaluate the job latency time and resource usage of each computing node (CPU, Memory, Disk and Network).

Procedure

Adapt TPC-H queries, generate data with scale factors from 2 to 32 and calculate the average job latency of 30 runs.

Baseline

Hive, an Hadoop application that converts SQL to MapReduce jobs is used as a baseline of comparison.

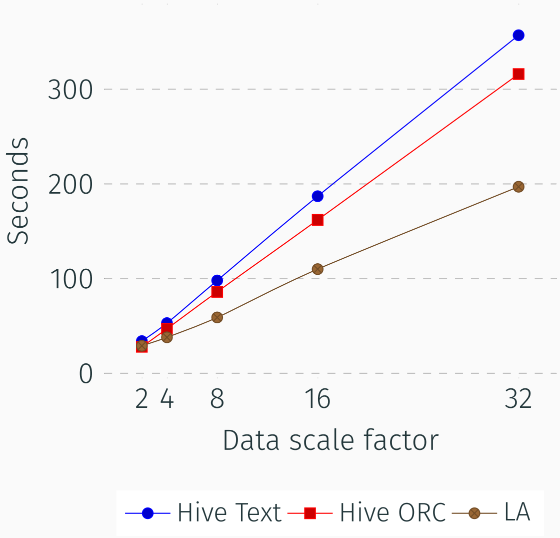
QUERY1 (ADAPTED FROM TPC-H)

```
SELECT RETURNFLAG, LINESTATUS, sum(QUANTITY)
FROM LINEITEM
WHERE SHIPDATE >= 1998-08-28
AND SHIPDATE <= 1998-12-01
GROUP BY RETURNFLAG, LINESTATUS
```

$$L_{ReturnFlag} \triangledown L_{LineStatus} \cdot [L]^{>=1998-08-28}_{Shipdate} \cdot [L]^{<=1998-12-01}_{Shipdate} \cdot [\![L]\!]_{Quantity} \cdot !^{\circ}$$

HASLab
HIGH-ASSURANCE SOFTWARE LABORATORY

QUERY1 JOB LATENCY



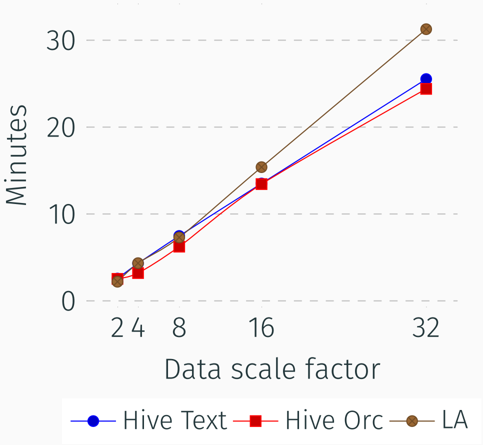
QUERY 3 (ADAPTED FROM TPC-H)

```
SELECT  L_SHIPMODE,
        O_ORDERSTATUS,
        SUM(O_TOTALPRICE * L_QUANTITY)
FROM LINEITEM AS L, ORDERS AS O
WHERE L_ORDERKEY = O_ORDERKEY
GROUP BY L_SHIPMODE, O_ORDERSTATUS.
```

$$(L_{shipmode} \cdot [\![L]\!]_{quantity} \cdot L^{\circ}_{orderkey}) \cdot (O_{orderkey} \cdot [\![O]\!]_{totalprice} \cdot O_{orderstatus})$$

HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

QUERY 3 JOB LATENCY



HASLab
HIGH-ASSURANCE
SOFTWARE LABORATORY

- LA has an improved performance of 38% on queries that work on a single Table.
- It has a decrease of performance of 29% on queries that involve a join.
- A large subset of Relational Algebra can be converted to LA equations.
- Matrix kernel libraries can be improved to handle sparse matrices.

FUTURE WORK

- With the extension of the algebra its possible to benchmark a larger set of TPC-H queries.
- Create an execution plan from the conversion of SQL to LA.
- Prove correctness of conversion from SQL to LA.
- Other areas worth exploring are:
    - Improve matrix libraries.
    - Extend other types of joins (*OuterJoins*, *AntiJoin*)

HASLab
HIGH-ASSURANCE SOFTWARE LABORATORY

A LINEAR ALGEBRA APPROACH TO OLAP

Rogério Pontes

December 14, 2015

Universidade do Minho