Hadoop and Relational Database
The Best of Both Worlds for Analytics

Greg Battas
Hewlett Packard
The Evolution of Analytics

Questions

- Is Hadoop a replacement or compliment for R-DBMS today?
- How does this change in the future?
- Is the current vendor “co-processor” approach the right model?
- If so, how do these technologies best work together?
What we see today

- Vendors offering MPP “co-processors”
  - Vertica, Teradata, Greenplum, Netezza, etc + MapReduce
- Focus on good integration with Hadoop
  - Parallel Data Movement
  - Native access to each other’s data
  - Metadata Integration
What do you mean when you say “Analytics”

Query
OLAP
Reporting
EIS
Data Warehouse

Predictive Modeling
Clustering/Segmentation
Optimization
Affinity Analysis
Analytic Sandbox
Analytics and the Analytic Lifecycle

- **Understand Bus.**
  - Determine business objectives
  - Understand business requirements
  - Determine data mining goals

- **Understand Data**
  - Collect initial data
  - Describe data
  - Explore data
  - Verify data quality

- **Data Preparation**
  - Select data
  - Clean data
  - Construct data
  - Integrate data
  - Format data

- **Modeling**
  - Select modeling technique
  - Generate test design
  - Build model
  - Assess model

- **Evaluation**
  - Evaluate results
  - Review process
  - Determine next steps

- **Deployment**
  - Publish a report
  - Deliver a score file
  - Implement a model within an event processor
Testing classic BI queries on Hive

- Tested in HP’s Performance Labs
- Working with Distro vendors
- Comparable clusters
- Suite of 22 TPC-H queries
  - Not based on tuned, audited results

![Geomean Elapsed Time (sec)](chart)

- RDBMS1
- RDBMS2
- Hive

Geomean Elapsed Time (sec)
Digging a little deeper
Why the performance difference?

- **Startup Time**
  - Example Q2 – Very short runner
    - 121 seconds on Hive even with 2GB of data
    - <10 seconds on commercial DBMS’s with 1TB of data
    - Queries are turned into MapReduce jobs

- **Runtime Performance Tuning**
  - Example Q1 – Simple table scan
    - MPP Rowstore DBMS is 74x faster
    - Commercial DBMS’s are highly optimized for scans

- **Query Optimization**
  - Example Q7 – Nested loop join and 4 hash joins
    - MPP Rowstore DBMS is 64x faster
    - Lack of options and lack of stats makes optimization tricky
# 30 Years of Performance Optimization

<table>
<thead>
<tr>
<th>Hadoop</th>
<th>Commercial DBMS</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Optimizer</strong></td>
<td>Branch and bound, cost based, histogram stats,</td>
</tr>
<tr>
<td></td>
<td>Dynamic Recompile, etc</td>
</tr>
<tr>
<td><strong>Runtime</strong></td>
<td>Compiled runtime, generated bytecode, optimized</td>
</tr>
<tr>
<td></td>
<td>pseudocode, chip optimized, etc</td>
</tr>
<tr>
<td><strong>Access Methods</strong></td>
<td>B trees, columnar scans, zonemaps, bitmap indices,</td>
</tr>
<tr>
<td></td>
<td>binary searches, etc</td>
</tr>
<tr>
<td><strong>Storage</strong></td>
<td>Binary, Byte Aligned format</td>
</tr>
</tbody>
</table>

- **Hive Dynamic Programming**
- **Optimizer**
- **Generated MapReduce Java code**
- **Runtime**
- **Scan**
- **Access Methods**
- **HDFS semi-structured files**
- **Storage**
- **B trees, columnar scans, zonemaps, bitmap indices, binary searches, etc**
Analytics in Database

- MPP DBMS’s are adding the ability to execute analytic code within their runtimes
  - Parallel UDF Model
  - Executes inline with the query and close to the data
  - SQL Runtime can parallelize the functions
- Different than a co-processor model
  - Co-processor – SQL and Hadoop analytics side by side
  - Analytics in DB – Analytic functions parallelized inside of SQL
Testing Text Analytics “In Database”

- Dataset – about 100K Wikipedia articles
- Four Text queries
  - Q1 requires a machine learning algorithm (CRFs) written in a 3GL
  - Q2-4 can be expressed as SQL with text operators

<table>
<thead>
<tr>
<th></th>
<th>Vertica</th>
<th>Hadoop</th>
</tr>
</thead>
<tbody>
<tr>
<td>Q1</td>
<td>447 seconds (SQL/UDF)</td>
<td>447 seconds (MapReduce)</td>
</tr>
<tr>
<td>Q2</td>
<td>0.356 seconds (SQL)</td>
<td>127 seconds (via PigLatin)</td>
</tr>
<tr>
<td>Q3</td>
<td>0.541 seconds (SQL)</td>
<td>172 seconds (via PigLatin)</td>
</tr>
<tr>
<td>Q4</td>
<td>24 seconds (SQL)</td>
<td>279 seconds (via PigLatin)</td>
</tr>
</tbody>
</table>
Comparing the Strengths

**Hadoop**
- Schemaless Model
- Human query optimization
- The ability to create complex dataflows with multiple inputs and outputs
- Parallelize many Analytic Functions
- Adept at unstructured classification and feature extraction

**MPP DBMS**
- Performance
- Machine query optimization
- Mature workload management
- High concurrency interactive query processing
A Framework for Discussing Data

Real Time

Traditional Apps
- Unstructured Apps
- App Server/Web Server
- RDBMS

Event Processing/Web Apps
- CEP/Stream Processing
- NoSQL/NewSQL

Data Manipulation
- Data Manipulation
- ETL
- Unstructured Feature Extraction/Classification

Batch

Staging Area
- Staging Area

Query/Analysis

Business Intelligence
- EDW
- Data Marts
- Query, Reporting and ROLAP

External Events and Data
(eg. twitter feeds, sensor data, clickstream)

Profiles/Models/Scoring
- Profiles/Models/Scoring
A Framework for Discussing Data

Real Time

- Traditional Apps
  - App Server/Web Server
  - RDBMS

- Unstructured Apps
  - File System

- Event Processing/Web Apps
  - CEP/Stream Processing
  - NoSQL (KVS/Document)

Batch

- Data Manipulation
  - ETL
  - Staging Area
  - Unstructured Feature Extraction/Classification

- Analytic Sandbox
  - File System
  - Analytic R-DBMS
  - Tools/Data Prep

Query/Analysis

- Business Intelligence
  - EDW
  - Data Marts
  - Query, Reporting and ROLAP

- MPP RDBMS

External Events and Data

(eg. twitter feeds, sensor data, clickstream)
Leveraging a Co-Processor Approach

Understand Bus.
- Determine business objectives
- Understand business requirements
- Determine data mining goals

Understand Data
- Collect initial data
- Describe data
- Explore data
- Verify data quality

Data Preparation
- Select data
- Clean data
- Construct data
- Integrate data
- Format data

Modeling
- Select modeling technique
- Generate test design
- Build model
- Assess model

Evaluation
- Evaluate results
- Review process
- Determine next steps

Deployment
- Publish a report
- Deliver a score file
- Implement a model within an event processor

DBMS
Hadoop or Analytic DBMS
Hadoop or Analytic DBMS

Key - with the right architecture it is cheaper to move the data once

HBase
Cassandra
Mongo
Membase
etc.
1. The enemy of great is good enough…
The rules of my industry

1. The enemy of great is good enough…

2. And the enemy of good enough is free
How might this change in the future

- Query Optimization Improvements in Hive
  - Statistics, better join ordering, more join types, etc
- Startup Time Improvements through Drill
  - Static Runtime like HBase rather than Mapreduce
  - Nested Model means simpler query plans to pass out
- Runtime Performance Improvements through Drill
  - Nested model eliminates many joins and flattens query trees
  - Columnar, compressed storage optimizes access methods

*Hive/Drill could evolve to be an interesting analytic co-processor*