Intelligent Architecture for the Data-Driven Business
Making the Most of Hadoop with Optimized Data Compression
(and Boost Performance)

Mark Cusack
Chief Architect
RainStor
Agenda

- Importance of Hadoop + data compression
- Data compression techniques
- Compression, projection and filtering for faster analytics
- Boost performance by avoiding the Map-Reduce layer
Introducing Hadoop...

- Hadoop is a scalable and reliable platform for storing and processing Big Data

Hive

Pig

SQL

HBase

Map-Reduce

HDFS
Multi-Structured Data Management

- Tabular
- Key Value
- Blob/Clob

**Hadoop**

- Relational
- Event logs
- XML Documents
- Binary Objects
Hadoop - The Price of Scalability and Reliability

- 3x data replication for availability
- I/O bound operations, and CPU under-utilization
- Heavyweight Map-Reduce framework
- Larger cluster sizes
  - More Admin & More Cost
Why Data Compression is Important

- Reduces cluster size and boosts efficiency
  - Reduces network and disk IO
  - Increases CPU utilization

- Lowers impact of replication and shuffle
  - Compress data at rest in HDFS
  - Compress data between Map-Reduce tasks
  - Compress output result sets
Compression Techniques

- General purpose compression libraries:
  - Bzip2, Gzip, LZO, Snappy, LZ4
  - Applies to all data types

- Column-oriented compression:
  - Hive columnar file format
  - Proprietary data de-duplication
  - Applies to structured data
Column-Oriented Compressed File Formats

- Structured data stored in a column-wise order

- Values in the same column are often related
  - Related values are localized on disk
  - Generic compression algs are more effective
  - Column-level de-duplication can be applied

- Direct access to columns of interest
  - Projections
  - Reduces IO overhead for analytics workloads
Example: Compression of FS Stock Quote Data

Ratios vs uncompressed source data

- LZO: 6X
- GZIP: 11X
- BZIP2: 14X
- Structured data de-duplication: 23X
Structured Data De-duplication

• No re-inflation
Hadoop Use Case: Data Warehouse Archiving

- Large bank manages Petabyte data on Hadoop for lower cost analysis and compliance
- Data warehouse too costly to scale (2 years of data = 500TB)
- Eliminate Offline tape
Boosting Analytics Performance using Filtering

- Data often has a natural order:
  - Database transactions ordered by timestamp

- Or data may have an order imposed on it:
  - Stock quotes partitioned by ticker symbol

- Order can be used to filter out tasks
  - Only process data in the window of interest
Filter for Performance

Pig statement
FILTER quotes BY symbol == 'GOOG' AND exchange == 'P';

HDFS

Bloom Filter

Pig statement
FILTER quotes BY symbol == 'GOOG' AND exchange == 'P';

HDFS

Bloom Filter
Map-Reduce Framework is Heavyweight

- Hadoop is best suited to batch analytics
  - Task scheduling takes a few seconds

- HBase achieves fast performance by avoiding Map-Reduce layer

- Some SQL DBs can also run on HDFS
  - Implement their own compute frameworks
  - HDFS used as a scalable storage platform
Hadoop and Data Warehouses are Converging

- Low Latency Analytics
- Costly to Scale
- Built upon SQL Standards

- Batch Analytics
- Low Cost
- Immature BI Support

Hybrid SQL-Hadoop Technologies Bridge the Gap
Example: FS Stock Quote Data Analysis

**Query Test:** Calculate the average daily quote price for ticker symbols for one day across 1.5 billion quotes
Summary

- Structured data compression increases storage capacity and cluster efficiency
  - Compression is an IO bandwidth multiplier

- Filtering and projection can significantly boost the performance of Map-Reduce tasks
  - Only load and process the data required

- The new breed of SQL on Hadoop databases provide:
  - Faster query response times than standard Hadoop
  - Full access to enterprise-grade Business Intelligence tools
Questions?

Mark Cusack
mark.cusack@rainstor.com
Intelligent Architecture for the Data-Driven Business