Sean Huban
Database Systems
Alan Labouseur
March 7th, 2017

Big Data: Analyzing and Comparing Database Systems

•••

Bibliography:

Hive- A Petabyte Scale Data Warehouse Using Hadoop

Ashish Thusoo, Joydeep Sen Sarma, Namit Jain, Zheng Shao, Prasad Chakka, Ning Zhang, Suresh Antony, Hao Liu, Raghotham Murthy

A Comparison of Approaches to Large-Scale Data Analysis

Andrew Pavlo, Erik Paulson, Alexander Rasin, Daniel J. Abadi, David J. Dewitt, Samuel Madden, Michael Stonebraker

Main ideas (Hive...)

- Facebook was created initially on an RDBMS database
  - As business quickly scaled upwards, it became too slow for their needs
- After researching many options for alternatives they decided that Hadoop would be a good foundation for what they were looking for
  - Hadoop is open source, and was already used in the world at petabyte scale, much larger than facebook at the time
  - But Hadoop on its own required hours of coding for even the most basic queries and analysis
- In order to combat the pitfalls of Hadoop, Facebook created Hive in 2007
  - Hive was created with the end user in mind, seeking ease of use
  - It was built on top of Hadoop using the Hadoop File System (HDFS), and uses its own language HiveQL
  - Hive structures data into well understood database concepts like tables ,rows, columns, and partitions.

Implementation of Hive

- The Goal of Hive was to become a system that followed the simplest and most traditional databases
- Hive stores its data in a combination of tables, rows, and columns
  - The columns support a various number of data types
  - Primitive Data Types
    - Integers (int, smallint, tinyint), Floating Point Numbers (float, double), and Strings
  - Complex Data Types
    - Associative Arrays (map), Lists (list), Structs (struct)
- The Query language used is a subset of SQL that they have created called HiveQL
  - They noted that they have added extensions that they view as very useful in their environment
  - In HiveQL, you cannot use inserts, it will override the data that was there and replace it
- The data for tables in Hive are stored in an HDFS directory
  - The data's primary units are: Tables, Partitions, and Buckets
  - It is important to note that Hadoop files can be stored in different formats and Hive does not impose restrictions on this

Implementation of Hive (cont.)

- The Building Blocks of Hive
  - These important ‘building blocks’ are what makes up the way Hive runs
- Metastore
  - The component that stores the system catalog and metadata about tables, columns, etc.
- Driver
  - The component that manages the lifecycle of a HiveQL statement as it moves through the Hive
- Query Compiler
  - Compiles HiveQL into a directed graph of map/reduce tasks
- Execution Engine
  - Executes tasks given from the compiler
- Hive Server
  - Provides and interface and provides a way for integrating Hive with other applications
- Command Line interface (and various other interfaces including UDF, UDAF)
  - Allow users to define their own custom functions

Analyzing Hive

- Hive was created in order for Facebook to have a scalable and efficient way to turn their data into analyzable information
  - In this regard they were very successful, as Hive is efficient and loading and sorting data, as well as having the ability to scale larger and larger as facebook inevitably continues to grow
  - Hive always seeks to optimize whatever task it is given so that it can be completed as efficiently as possible
- Hive created with the end user in mind seeking to eliminate unneeded complexity
  - By using HiveQL, a language so similar to SQL, familiarity for new users of Hive will allow them to work at their most efficient levels much quicker and giving these new users the best experience possible
- Hive is a great program to be used in systems that follow RDBMS and is tweaked specifically for an environment dealing with large amounts of data containing special extensions
  - The only consideration is that it does alter the INSERT and join commands which could be a deal breaker for certain databases wishing to implement Hive

Main Ideas (A Comparison of Approaches…)

- This study looks to compare 2 systems of large-scale data analysis to see which was more efficient in completing tasks: MapReduce paradigm or Parallel DBMS?
- MapReduce contains only two functions, ‘map’ and ‘reduce’ which is simple to implement and allows for programmers to try different code
- Parallel DBMS uses relations to structure the data while MapReduce has no real structure at all
- It was found that DBMS performed faster than MapReduce in testing

Implementation of Comparisons

- Both systems were tested equally on multiple tasks
- The first test was scanning 100-byte records for 3 character patterns
  - DBSM perfomed the best in this test
- The second test was 4 tasks related to HTML document processing
  - DBSM also performed best in this test

Analyzing Comparisons

- All of the tests that were run to compare these two systems were very large scale for large business application
- DBMS looks to optimize through structuring and indexing it’s data
  - This makes it the most practical choice for large businesses
- MapReduce is open source, and therefore can be easier to understand and implement in special situations
  - Just because it was slower in this testing than DBMS does not mean that it has no purpose. The open source provides for a large number of possibilities in specific scenarios that require large amounts of experimentation and customization

Comparing Ideas and Implementations: ‘Hive’ vs ‘Large Scale Data’

Ideas:

- ➔ MapReduce is similar to Hive in that it seeks simplicity. Keeping the functions simple allows for coders to experiment within their own environments and leave the system open to more unique possibilities similar to how Hive does not restrict the types of data that can be used.
- ➔ DBSM was created with the performance aspect in mind, detailing a specific structure for data that results in faster loading times and task completion
- ➔ Hive seems to be a mix between the two of these, remaining open source to allow for new code and experimentation, while also containing structure that will optimize large amounts of data and keep it organized in an effective manner

Comparing Ideas and Implementations: ‘Hive’ vs ‘Large Scale Data’ (cont.)

Implementations:

- ➔ MapReduce does nothing to implement the data is it given in any specific way
- ➔ DBMS uses a relational paradigm to store and sort its data
- ➔ Hive uses some of the most well understand relational concepts (RDBMS) to organize its data such as tables, columns and rows
- ➔ The combination of open source code and performance enhancing implementation of Hive makes it a great combination of these two systems, making it very versatile and suitable for a wide range of users.

Main Ideas (Stonebraker)

- ➔ The initial databases were thought to have been standardized with Relational Database Management storage (RDBMS)
  - ◆ By 2005, it was realized that RDBMS was becoming obsolete in many markets
- ➔ In 2015,'One Size Fits None', arguing that 10 years later RDBMS is entirely obsolete
  - ◆ Most major venues will soon use column stores which has been proven to be faster
  - ◆ Transaction Processing(OLTP): makes row stores obsolete with little memory requirements
  - ◆ NoSQL Market: over 100 vendors with no standards to them but it is a growing market
  - ◆ Complex Analytics are business intelligence products allowing for all kind of predictions that would be way too slow if simulated in SQL
- ➔ Many other new markets and implementation being created every day that hold further potential that will move us past RDBMS and will affect majority market shareholders today as the "elephants try to adapt without losing market share"

Advantages and Disadvantages of Hive

- Advantages:
  - Hive is a mix of both DBMS and MapReduce, combining the best features of each to eliminate some of the problems that they have individually, creating a versatile system
    - Using a language similar to SQL makes Hive easy to use
  - Is perfect for companies sorting through large amounts of data like Facebook who are looking for enhanced performance
- Disadvantages:
  - Hive is a RDBMS which according to Stonebraker is a model that is not only becoming obsolete, but is already obsolete when looking at the future from today.
    - Hive would not be useful for any industry looking for column stores or business analytical graphs and predictions
  - It is modified in ways that only makes it the best choice in certain applications of business