Big Data Summary

Hive - A Petabyte Scale Data Warehouse Using Hadoop

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

A Comparison of Approaches to Large-Scale Data Analysis

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

Michael Stonebraker on his 10-Year Most Influential Paper Award at ICDE 2015

Presentation by: Nick Comito
10/20/16

Hive - Main Idea

- Traditional warehouse solutions are becoming extremely expensive for the business intelligence industry as the size of data sets continue to grow.
- Companies such as Facebook and Yahoo have been using Hadoop to store large data sets.
  - Hadoop is is an open source MapReduce implementation but MapReduce models are hard to maintain and reuse since the developer needs to write custom code.
- Hive is a relatively new model that expands on Hadoop used to manipulate data. It allows users to write MapReduce scripts into queries making it easier for end users.

Implementation

- Hive provides a querying language similar to SQL for individuals who are not used to the language used in MapReduce models, specifically Hadoop.
    - The queries written are still being executed in Hadoop
    - Currently Hive does not accept all SQL functions.
- HiveQL supports all primitive and complex data types.
- Hive contains a Metastore which is similar to the systems catalog of a relational database model.
    - The metastore contains all information regarding tables, schema, location, data types, and any other internal information.
    - “Metastore is very critical for Hive. Without the system catalog it is not possible to impose a structure on hadoop files”

Analysis

- HiveQL is an effective system that helps solve the end user problem seen with MapReduce models. Hive gives control to the ender users of the database since they can now manipulate the data with queries.
- Being that HiveQL is similar to SQL, it “enables anyone familiar with SQL to start a hive cli(command line interface) and begin querying the system right away”
- The addition of Hive to Hadoop has proven to be a positive combination for Facebook
  - Hive has reduced costs for providing data processing services to engineers and analysts
- Facebook believes the performance of Hive and the ability of Hadoop to scale large commodities will allow them to be able to handle the growth of big data in the future.

A Comparison of Applications to Large-Scale Analysis

- The purpose of this paper is to compare the MapReduce model and the parallel DBMSs.
    - The paper compares the two by looking at their performance as well as developmental complexity
- MapReduce is a relatively new large-scale data model that is gaining attention. It is called MapReduce because of its functionality.
    - The Map function reads a set of records from inputs, performs any desired tasks such as filtering, then returns a set of intermediate records as the output
    - The Reduce function performs summary operations regarding the inputs
- The parallel DBMSs have been the standard database systems used since the 1980's and support all standard relational tables as well as SQL.

Implementation of the Comparison

- In order to compare the two models, 5 tasks were executed on MapReduce systems and parallel DBMS's
    - MR system that was tested was Hadoop
    - DBMS's systems tested were DBMS - X and Vertica
- The five tasks performed were the original MR task (Grep Task), selection, data loading, aggregation, and UDF aggregation.
- For each of these tasks, the systems were tested three separate times and the average was recorded. The tasks were executed on different cluster sizes in order to show the ability to scale for each system.

Analysis of MR vs. Parallel DBMS

- The overall results of the comparison show that the parallel DBMSs outperformed the Hadoop MR system.
    - The averages between the 5 tasks and different tests within each task show that on a 100 node scale, DBMS - X and Vertica were 3.2x and 2.3x faster than Hadoop, respectively.
- Hadoop MR cost a lot of energy compared to the parallel systems due to the fact that MR must scan the entire input file to start a query
    - Parallel DBMS use cluster indexes to save time and energy
- Besides the results of the test, three things in the paper stuck out to me.
    - Fault Tolerance: MR has more tolerance to node failure than parallel DBMS
    - Flexibility: Even though parallel DBMS have increased generality, MR provides more
    - Indexing: MR does not have built-in indexes like DBMS making it difficult for programmers to develop indexes to increase speed and efficiency.

Comparing Hive and Comparison Paper

- The ideas and implementations regarding the subject matter in both papers differ
- Hive paper:
  - Hive was a system built to improve on the Hadoop MR system that was being used for large data
  - The purpose of Hive is to provide users that are experienced in SQL a way to transition and manipulate data in an MR system. Since MR systems are designed by the developer, they are hard to maintain and reuse
- Comparison paper:
  - This paper's main objective was to illustrate the difference in performance between the standard DBMS and the relatively new MR systems.
  - The paper shows the results of testing to compare the performance of the two systems, ultimately stating that the DBMS outperforms the the MR.
- Both papers do a good job in detailing the architecture of the systems they are describing, giving their audience a better understanding Big Data Systems.

Stonebreaker Discussion

- One size fits none
  - DBMS are being replaced with newer systems and techniques to handle big data faster
- Column Stores are taking over
  - Vendors are using column stores instead of row stores to increase speed.
  - The warehouse market will become entirely column stores.
- Analytics market is becoming complex
  - Soon data scientists will replace business analysts and want to conduct more complex analytics such as regressions and data clustering
  - Can be simulated on SQL but it is extremely slow
- Opportunity for new ideas
  - Processor diversity will increase for vendors.
  - Stonebreaker expects to see a large amount of new implementations
  - Current bottleneck for multi-node DBMS implementation is networking. High speed network systems have the potential to fill this market gap.
- How will old school vendors adapt without losing market share?

Pros & Cons of Hive

Pros

- Hive is a column store database which functions much faster than row based
- HiveQL is is an easy transition from SQL to MapReduce, which will help the "elephants" adapt to new technology without losing market share.
- Hive allows end users to manipulate data

Cons

- Hive is stacked on Hadoop which is a MR system. As seen in the comparison paper, MR systems run significantly slower than parallel DBMS systems.
  - With an emphasis on complex analytics this may pose an issue.
- Hive does not currently accept all functions in SQL
  - Even though complex analytic simulation is slow in SQL, Hive could potentially slow the process even further.