* Big Data Paper Summary

Hive – A Petabyte Scale Data Warehouse Using Hadoop

Facebook Data Infrastructure Team: 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

Andrew Pavlo, Brown University, pavlo@cs.brown.edu
Erik Paulson, University of Wisconsin, epaulson@cs.wisc.edu
Alexander Rasin, Brown University, alexr@cs.brown.edu
Daniel J. Abadi, Yale University, dna@cs.yale.edu
David J. DeWitt, Microsoft Inc., dewitt@microsoft.com
Samuel Madden, M.I.T. CSAIL, madden@csail.mit.edu
Michael Stonebraker, M.I.T. CSAIL, stonebraker@csail.mit.edu

By: Joseph Schmidt
11/23/2014

Main Idea- *Hive*

- The Facebook Data Infrastructure team needed a way to run HaDoop, an open source framework for handling large data-sets, which was easier for the end user. Previously, the end user would have to write a program for even simple tasks and data analysis. The team came up with the idea of Hive to make it easier to perform tasks by making it familiar with tables, columns, rows, and other SQL-like structure.

Implementation- *Hive*

- The combination of Hive and HaDoop contributes to the data processing that Facebook has to do on a daily basis. There are a variety of jobs that Hive runs daily which include summarization jobs and machine learning algorithms. It has made ad hoc analysis simpler, and most of the workload is ad hoc queries, while the rest is for report dashboards. To summarize, Hive provides data processing for analysts and engineers in a more efficient manner than traditional methods.

Analysis- *Hive*

- Based on the paper, there was certainly a need for a way to easily work with HaDoop without having to program each time you wanted to do even a small amount of analysis, which sounds like the old flat file system that we talked about in class, where you had to write a program if you wanted to access data (and that is a bit absurd). The use of almost all SQL syntax makes Hive easy for the end user that is familiar with SQL, and easier to learn in general, since SQL syntax is not too difficult. Hive seems to be a flexible format, and conceptually can be grasped if one has knowledge of Map Reduce and RDBMS (which Hive uses in its metastore process).

Comparison of *Hive* to comparison paper

* Ideas: The second paper, *A Comparison of Approaches to Large-Scale Data Analysis*, seems to agree on the point that Map Reduce can be effective if there is no sharing needed between programmers, but you need to write a program every time you want to manipulate data sets. This shows that if there is a need to share, that RDBMS is the way to go, and removes the need to write a program every time you want to manipulate data, but instead, just write a query. They also agree that higher level interfaces like Hive make it much easier to run complex tasks on data sets with Map Reduce software like Hadoop.
* Implementation: The benchmarks for Hadoop in this paper show that it has better load times, though is slow with a small amount of data and across all cluster scaling levels, and has lower performance with compression and task set up. Additionally, Hadoop was easier to configure and performance was either increased or not affected by certain parameters, and getting Hadoop running was not labor intensive. Essentially, the paper shows the reader that Hadoop is popular because of its ease of set up and usage, but lacks behind in performance compared to RDBMSs like DBMS-X or Vertica since it does not have a schema.

Advantages/Disadvantages of *Hive* in context of comparison paper

* Advantages: In context of the second paper, the main idea of *Hive* has the benefit of using a higher level interface to accomplish common tasks on data sets through Hadoop, which sort of combines the ability to use SQL and some parts of regular RDBMSs and Map Reduce . Further, the Hive's SQL like syntax is good for those who are familiar with SQL syntax, since it makes it more usable. The Hive also has a way to share between programmers, although it is very difficult, and is still being worked on as a feature, but this could be very beneficial since Map Reduce software alone typically cannot do this (e.x. Hadoop).
* Disadvantages: The Hive is still not relational, so even though it introduces SQL like syntax, it is still based on a "schema later" or "schema never" format since it works with Hadoop, which means that performance will not be as good as compared to a relational format. Additionally , programmers who are used to languages like Java may not necessarily be familiar with a declarative language like SQL (which part of Hive is using SQL like syntax).