BIG DATA PAPER SUMMARY

Kayla Mesmain
October 30,2017

HIVE- A PETABYTE SCALE DATA WAREHOUSE USING HADOOP

Thusoo, Ashish, et al. “Hive - a petabyte scale data warehouse using Hadoop.” *2010 IEEE 26th International Conference on Data Engineering (ICDE 2010)*, 2010, doi:10.1109/icde.2010.5447738.

A COMPARSION OF APPROACHES TO LARGE-SCALE DATA ANALYSIS

Pavlo, Andrew, et al. “A comparison of approaches to large-Scale data analysis.” *Proceedings of the 35th SIGMOD international conference on Management of data - SIGMOD 09*, 2009, doi:10.1145/1559845.1559865.

MICHEAL STONEBRAKER ON HIS 10-YEAR MOST INFLUENTIAL PAPER AWARD

One Size Fits All- An Idea whose Time has Come and Gone, 2005, at ICDE 2015

MAIN IDEA OF HIVE

- In 2008, Facebook data warehouse was using a RDBMS. Due to increase use in Facebook there was a need in a flexible infrastructure for larger data that was cost effective.
- Hadoop- is a map-reduce open source for storing large data like petabyte.
- Hive- uses Hadoop, thus is an open source warehouse solution. It translates queries into map-reduce.
- Facebook used Hadoop as a faster alternative which required to use hive because it uses a query language called HiveQL. This helps incorporate data to table without translation which saves time.

IMPLEMENTATION OF HIVE

- Hives is compiled into a implementation for processing big data called map-reduce which is then executed with Hadoop.
- Hives uses a metastore.
- Hives stores in a table with rows and columns which have a primitive and complex type.
- Hive is supports integers, doubles, floats and arrays.
- Hives compiler is able to edit the hdfs directory needed for data and uses a bucket.
- It uses SerDe (Serializer and Deserializer) and objectInspector interface for data formatting and encoding used in a table.
- Hives is interchangeable when using clauses in sub-queries.
- Important parts of hive:
  - Metastore-stores the system catalog
  - Driver – controls the lifecycle
  - Query Compiler- compiles hiveQL to map reduce
  - Execution Engine-executes the plan to the Hadoop
  - HiveServer – helps hive be adaptable, provides the interface and JDBC/ODBC server

ANALYSIS

- I think that Hive is more cost efficient system that geared to handle larger clusters of data to help grow with the productivity of Facebook being one of the largest social media websites. Thus, this helps Facebook be more efficient for users.
- Helps reduce the capacity of workload for Facebook programmers due to modifications and manageable tasks.
- The ability to use hiveQL as a sql function for Hadoop makes this an easier way to handle large data.

MAIN IDEA OF COMPARISION PAPER

- The fundamental points of the comparison paper was to compare parallel SQL Database Management Systems (DBMS) and Map-Reducers (MR). The performances of these large scale analysis are different.
- There was a test performed between MR and DBMS with Hadoop, DBMS-X, and Vertica. These performances of the test prove the outcome of the analysis.
- The results in the paper proved that DBMS was more efficient. DBMS performance of DBMS-X and Vertica results were significantly quicker computing and less code. But the data had to conform to a well defined schema data.
- Map-reducer is much simpler and enables data to be in any format. We have found that it is less efficient because it works better with smaller amounts of programmers and application domains compared to larger size.

IMPLEMENTATION OF COMPARISON

- In Map-reducer there are two processes the:
  - Map –data records being split partitions into buckets
  - Reducer – the files are sent to multiple clusters of nodes

- During computation MR is better equipped for node failures because of the split partition
- Can conform to any form
- Hadoop represented the MR for the results system test in which the loaded time was quicker because it reused the task

- In parallel DBMS the process are :
  - The clustered nodes are partitioned
  - The optimizer is being used by translating sql to query
  - Divided into multiple nodes

- DBMS used DBMS-X and Vertica in which to prove that the performance was better
- The execution of the systems required them to load the commands parallel on each node and using a specific form
- Although it was slower the results proved performance was better

ANALYSIS

- I found it interesting that DBMS preformed better. In the journal it stated that it only uses 100 nodes compared to MR which uses 1000 nodes. Although 100 nodes seems like a very low amount its found that many companies actually use less than 100 nodes.
- DBMS is better than MR in the test due to the queries on the nodes go through local tables and take specific fields. Map – reduce needs both elements in order to perform faster.
- Due to MR being able to be flexible by forming in the system catalog the necessary steps needed is lacked when using data referential integrity.
- Thus, DBMS is scaled in the diagrams as a better use of larger data than MR.

BOTH PAPERS COMPARISON

Hive

- In the Hive's journal it was found the writers stressed the importance that hives and hadoop was an upgrade system for storing large data for Facebook. These map-reduce systems made it easier for employees and faster.
- The flexible query language makes it more helpful. There are a two step process when using mapreduce which can be a disadvantage or advantage to there being less failure.

A comparison of approaches to large scale data analysis

- Tested the performances of both systems and found that DBMS had a better performance. The multiple tests of the performance were proved and shown in the table.
- MR had a complete scan while DBMS used the clustered indexes.
- In the article it states that DBMS is faster in range of 3.1 to 6.5 even though there is less nodes than in MR.

Both papers prove that there is an need for taking on larger scale data. Although, some companies such as Facebook found that MapReduce is a better alternative the other paper tested that DBMS has a higher performance than MR. Therefore, both targets different audiences.

MAIN IDEA OF STONEBRAKER TALK

- Stonebraker address that one size fits all, but in reality one size does not fill all in relational database. Many people believe that DBMS is the answer for everything and its not.
- He believes that oracle and DB-2 is good for nothing
- It can be considered a universal type data traditional uses will be non existent
- Data warehouse, complex and analytics are transitioning to columns store.
- For example, Business Intelligence will perform basic procedures
- For example, OLTP stimulates column store instead of row storage

ADVANTAGES AND DISADVANTAGES OF THE THREE SOURCES

Advantages

- Cost Effective
- Its simplicity in design makes installation and implementation easier to the user as opposed to other
- The Hive query language makes it easier for users programming when compiled
- It is less likely to have system failure due to the process during mapreduce
- One system is able to form to many other uses

- Disadvantages
- It is not a practical use for all forms of using large data
- The Insert is not functional it overwrites over existing data.
- Doesn't provide an built in index and transformation.
- Slower process for Mapreduce
- It only enables equality predicates for join