Big Data Paper

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 and Raghotham Murthy

A Comparison of Approaches to Large-Scale Data Analysis – Andrew Pavlo, Erik Paulson, Alexander Rasin, Daniel Abadi, David DeWitt, Samuel Madden, Michael Stonebraker

Michael Stonebraker On His 10-Year Most Influential Paper Award- ICDE 2015

Cory Lang, March 7, 2017

Hive – Main Idea

- Issues Addressed in the Paper
  1. Rapid Expansion of the amount of data that needs to be stored and collected
  2. Facebook had 15TB of data to store in 2007, 700TB today, all built upon outdated infrastructure
  3. As a solution, the authors looked at Hadoop and Hive (two open sourced frameworks used to process and store large amounts of data
  4. Their goal was to bring the familiar concepts of tables, columns, and subsets of SQL to the unstructured world of Hadoop

Using Hive, they can build on Hadoop to optimize the best of both programs, in turn reducing query times and database efficiency altogether

In summary, the main goal was to establish easy-to-use infrastructure that was able to scale in size and need with the amount of data they were dealing with

How is Hive Implemented?

- Hive is an open sourced program that supports traditional database concepts as well as complex ones such as maps, lists, and structs.
- Hive stores data in tables. These tables have rows and columns and the cells have specified types.
- Using Hive, we are given the ability to incorporate the data into a table without having to transform the data.
- Hive is built on top of Hadoop. Using this structure, the authors of the paper can use the advantages of both programs to effectively run the database.
- They introduced Hadoop, and they were able to execute jobs in a few hours that would normally take them a few days.
- However, Hadoop lacked the expressiveness of popular query languages, and Hive was able to make up for this.

Hive Analysis

- The introduction of Hadoop was necessary for companies like Facebook and Yahoo to sort through the data they were collecting. It was crucial for them to secure quality infrastructure and software that was able to scale with their needs.
- Hadoop provided a good job at being able to execute their commands, but again lacked the ease-of-use off traditional SQL.
- Due to the expected future growth of Facebook, one can except the data to grow with it. This means implementing Hadoop, a program that can scale with the amount of data, was a good choice.
- Hive is not perfect- going ahead they can improve on the efficiency of the Hive queries and improved scan performance. This will all come in time, but it was a good step.
- Implementing a project like Hive and Hadoop was necessary and a solid foundation for Facebook.
- The data they are going to have to analyze will only grow exponentially, so it is important to have this solid foundation on which they can improve.

Comparison Paper- Main Idea

- The main idea of the comparison paper was to provide the reader an in-depth analysis of a MapReduce program and a traditional DBMS. In this analysis, they were put to the test in parallel with each other and their results were compared.
- Using these results, the authors are able to identify where each of the models excels and where they are lacking. They are also able to attribute some of those strengths and weaknesses to certain developments and inventions over the years.
- Both Hadoop and the DBMS have their pros and cons, and each one is better at something the other lacks.
- Using this analysis, the author presents a clear analysis of why certain interfaces like PIG and Hive are being built on top of MR engines like Hadoop.
- They are looking for the advantages of both engines.

Comparison Paper- Implementation

- In this paper, the MapReduce engine and DBMS conduct several tests and their results are compared against each other. The results clearly show the strengths and weaknesses of each engine, and they can be attributed to how long those engines have been in existence.
- For Example, Hadoop was much easier to initially implement because of how new it was in comparison to the DBMS. The DBMS was much harder to implement and this was a major reason why Hadoop has such popularity amongst new customers.
- On the other hand, Hadoop's speed in execution clearly lagged behind that of the DBMS. The DBMS has gone through different technology improvements like B-tree indices, aggressive compression techniques, and column orientation in order to speed up jobs. Hadoop also used much more energy to do the same job because of how spread out the job was being executed.
- Using this analysis, we are able to identify why certain customers are choosing a MapReduce engine over a traditional DBMS and vice versa.
- This also gives birth to why people are designing interfaces over Hadoop (such as Hive). It allows the user to have the SQL strengths of a traditional DBMS but the complex computing of a MapReduce engine.

Comparison Paper- Analysis

- The analysis of the comparison paper shows a clear shift in the focus of these individual engines. It is clear that there are certain strengths and weakness to both, and with this we can make the assumption that the open-sourced programs are going to be more popular.
- One way to maximize the efficiency and ease-to-use is to implement interfaces like Hive over Hadoop. It is clear that these systems, alone, are not going to accomplish as much as they would when they are used in conjunction with each other.
- There is going to be a movement towards integrating multiple engines into one user interface. With this model, the user will have access to all of the strengths of each engine with the coding benefits of SQL.

Ideas and Implementations of the Two Papers

- The Hive paper and comparison paper have several things in common. They both implement the idea of having a higher level interface that mitigates the weaknesses of Hadoop.
- The Hive paper supports the ideas presented in the comparison paper because comparison paper identifies a general movement within the industry. The Hive example is using this general movement and specifically testing it.
- The general movement from individual engines to integrated programs with multiple engines are going to be the way to maximize the potential of these Big Data systems.
- As we move forward with these integrations, there are going to be flaws, and the Hive paper recognizes this. Now that they have recognized where they are weak, they are going to move forward and further develop the model.

Stonebraker Talk

- The main idea behind the Stonebraker talk was the fact that RDBMS people thought the answer to any question was a RDBMS. The traditional table engine was so effective in the 80's and 90's that they essentially thought all it needed to be better was a little fine tuning.
- Using the phrase, "One size fits all," they quickly realized that the RDBMS was not the answer to everyone's question. This phrase transformed into "One size fits none".
- Using an analysis of several different markets including data warehousing, transaction processing, NoSql, complex analytics, streaming market and they graph market, Stonebraker was able to show how useless a traditional RDBMS was in comparison to more targeted engines.
- Using this conclusion that RDBMS were not in the future of these markets, he stressed the importance of engines being specifically geared towards certain applications.
- Due to the new ideas and new applications in the world of Big Data, it is important to realize that RDBMS will not have a huge market share in any of these markets.
- A RDBMS is well rounded enough to adapt to any job, but it is not specialized enough to compete with the engines that are specifically designed to do a certain job.

Advantages and Disadvantages

- The implementation of Hive built over Hadoop is going to provide the employees at Facebook and Yahoo the complex analytics of MapReduce with the simplicity of SQL.
- Stonebraker addressed the issues with traditional RDBMS and how they are being faded out of todays markets.
- This improvement on Hadoop is a good move as it allows for the scaling of needs with the amount of data they store.
- In order to stay competitive and retain market share, it is important that the engines they are using are designed specifically to the application they are used for.
- One of the disadvantages of this implementation with Hadoop is the fact that it may have to be replaced by a more complex data analytics engine.
- Traditional RDBMS used to be thought of as the “one size fits all” answer, but now it is clear that as applications are becoming more and more specialized, they are going to require different engines