International Journal of Engineering Research and Reviews ISSN 2348-697X (Online)
Vol. 7, Issue 3, pp: (6-10), Month: July - September 2019, Available at: www.researchpublish.com

Query optimization techniques in Hive: Comparative Analysis

Khadeeja Alsolami[1] and Fahad Alqurashi[2,*]

[1] Computer Science Department, King Abdulaziz University

[2] Information Technology for Infrastructure, King Abdulaziz University, Saudi Arabia

*Corresponding author E-mail: fahad@kau.edu.sa

Abstract: **Hive is an open source is designed to handle a large amount of data. It is an open source is built on top of Hadoop. It stores data at tables like a relational database management system. Today, a many organization use Apache Hive to process their data. Hive is begin increasingly used in the many organizations so, a more efficient and flexible technique is needed to improve the query performance in Hive. The goal of this paper to conduct a comparative analysis between a two of a query optimization technique that used by Hive. These techniques are map-reduce and cost-based optimization. In the paper, we determine the methodology to perform the test. After performing the test, we conclude that the map-reduce was best response time of a query in the hive**.

Keywords: **Hive; query optimization; map-reduce; cost-based optimization.**

1. INTRODUCTION

Hive,[1], Hive is a data warehouse infrastructure. It is an open source built on top of Hadoop framework It provides data summary, query, and analysis. Hive gives a SQL-like interface to inquiry information stored in different databases and file system that include with Hadoop.

The technology initially developed by Facebook. Today, many of the organization using Apache Hive to Analysis and process their data in a familiar way[2].

Hive is being increasingly implemented in a wide range of organizations, so a more efficient and flexible technique is needed to optimize the performance of queries. Optimizing the queries is directly related to data size, data organization, infrastructure, storage formats, and processor readers.

Recently, there are many query optimization techniques introduced into the HIVE, but still, there is no clarity of the most effective technique, which can give more efficient query processing time with a lower latency rate[3].

The purpose of this research is to conduct a comparative analysis between map-reduce and cost-based optimization technique. After that, determine which has low latency in response time.

This paper is divided into ten sections as follows: Section 2 presents related work, while Section 3 presents hive definition, Section 4 presents Hive Architecture, Section 5 presents query optimization, section6 presents map-reduce, section7 presents cost based optimizer, Section 8 presents methodology, Section 9 present experiment result and Section 10 presents conclusion and future work

2. RELATED WORK

In recent years, some researchers have applied some approaches to optimize the query in the hive and reduce response time.

Gruenheid et al. [4] Developed an approach to improve the performance of queries executed under a hive. They use the column statistics to optimize the query in the hive. After the experiment, they concluded that a small number of column statistics can effectively promote the arrangement of joining in a hive.

Ala [5] made a comparison between two types of optimization technique. After that, she evaluates the performance of the partition table to Hive by used TCP-H.

In [6], the authors proposed to use of Multi-Query Optimization (MQO) technique to improve the performance of Hive. The author classifies sets of queries and then combines these queries into the Optimized HiveQL statements. From this approach, the author has shown that performance improvements can be achieved at the hive.

The author at [3] conducts a comparative analysis of different technique to explore the most efficient technique in Apache Hive query execution engine, and which technique can facilitate to improve the response time and decrease a job load. These techniques are MapReduce, Optimized Row Columnar File(ORCF), Vectorization and Cost-Based Optimization. The author concluded from the experiment map-reduce with ORC was the best at response time of different query

3. HIVE

Hive is a data warehouse infrastructure. It is an open source built on top of Hadoop framework. It provides data summary, query, and analysis. Hive has an interface to inquiry information stored in different databases and file system. Query language in the hive called HiveQL. It is like SQL query which is utilized to sort and query data stored in Hive. The most important functions that are supported by Selective statements in HiveQL are to join tables on a common key, filter data by using row selection techniques and project columns. The Hive supports the execution of multiple HiveQL statements during a single operation. But it does not support cross products, unlike database management systems [1],[4].

4. HIVE ARCHITECTURE

The components of Hive architecture [7], are:

- **User Interface**: it is a command line interface that allows the user to enter HiveQL and shows the result of the query.
- **Compiler** It executes a compilation of the query and converts the query to the execution plan.
- **Driver**: It acts as a controller which Receives the queries statements from user interface then start execution the statements.
- **MetaStore**: It stores metadata for each table and partitions. Such as schema and location which helps the driver to track the progress of dataset distributed on the cluster.
- **Execution Engine**: After Compilation and Optimization, the Executor executes the generated plan.

Figure 1 shows Hive Architecture

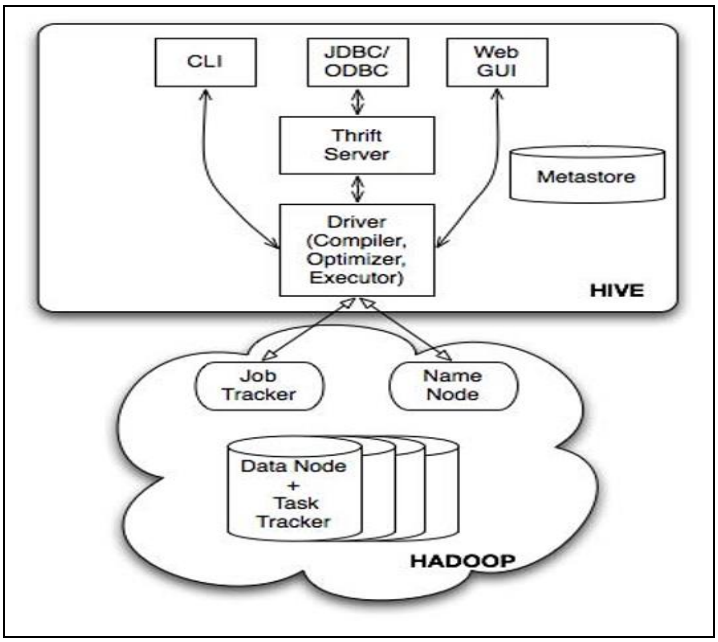

Figure1: Hive Architecture [7]

5. QUERY OPTIMIZATION

The Query optimizer tries to determine the most effective way to execute a particular query by taking into account the possible query plans [8]

Generally, the query optimizer cannot be accessed by users. First, the queries are sent to the database server and analyzed by the parser. After that, the queries are passed to the query optimizer where optimization occurs. When you submit a query to the database, the query optimizer evaluates and corrects some of the possible plans to execute the query and return what is considered the best option [8],[9].We will discuss in next two sections two type of query optimization in Apache Hive.

6. MAP REDUCE

MapReduce is a parallel programming model. It is suitable for big data processing[10]. MapReduce divide the task into smaller and assign them to many computers. It contains two important tasks Map and Reduces. Map phase takes a set of data and transforms it into another set of data. Reduce phase takes the output from the map as input and combine them into a smaller set of tuples[11].The figure2 display different phase of MapReduce. Input phase records the data and sends the analyzed data to the mapper in the form of key-value pairs. Key pairs created by the mapper are known as intermediate keys. Then reduce phase take the output of map phase as input.

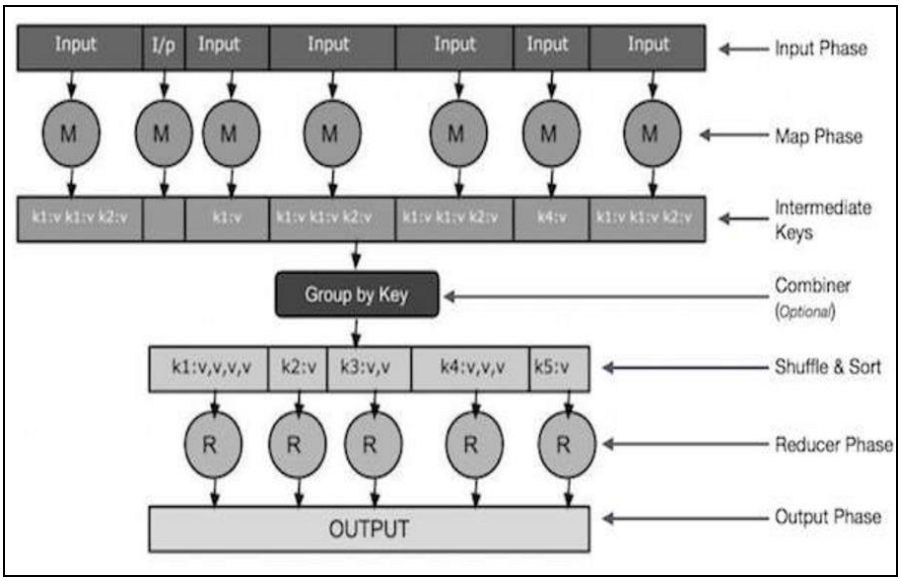

Figure2: MapReduce Phase [11]

7. COST BASED OPTIMIZER

Query optimization tools tend to have the greatest impact on performance in the data warehouse system because creating the correct (or wrong) execution plan will be affected at the time the query is executed[12].

The main objective of the CBO is to generate effective execution plans by examining the tables and conditions specified in the query, in the end, reducing the query execution time and limiting the use of resources. Calcite has an efficient pruner plan that can determine a cheaper query plan. All queries are converted by Hive to a physical trigger tree and converted to Tez/MapReduce jobs and then executed on the Hadoop cluster.

In CBO, a query passes through four phases: Parse and validate a query, Generate possible execution plans, assign a cost for each logically equivalent plan and Select the plan with the lowest estimated cost[13],[12].

8. METHODOLOGY

Figure 2 shows a main step in the methodology. First, we search for the suitable dataset and download it. The data set that we downloaded from website contains three tables which have many records. Then upload dataset in Hive MetaStore to do some queries later. After that, we design a different query and perform a test using two technique. At the end, we will Compare the results that we obtained. Figure 3 shows research methodology.

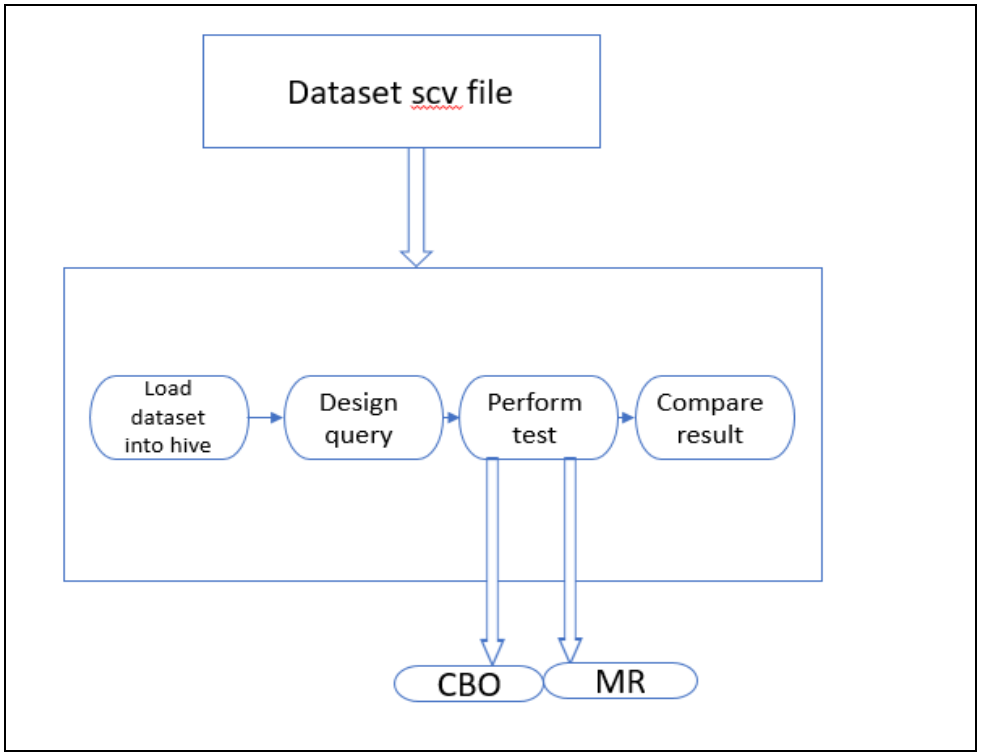

Figure3: methodology

9. EXPERIMENT RESULT

To evaluate how query optimization technique can reduce response time on the hive. We performed an experiment comparing between map-reduce and cost-base. We design two types of query three simple query and three simple join query on the dataset. We run the query on Hive at CLOUDERA VM[14]. After that, we compared the performance as shown in Tabel1.

Tabel1: Execution time1

Query	Execution time(sec)	
Technique	Map-reduce	Cost-based
Q1	.01	.2
Q2	.01	.3
Q3	1	2.3
Q4	137.105	200
Q5	1.102	3
Q6	50	69

10. CONCLUSION AND FUTURE WORK

In this paper, we did a comparative analysis between two type of query optimization technique in the hive. By applying query optimization and execute set of a query on Apache Hive. The Map reduces technique was better than cost-base technique. It speed-up performance with low latencies. Also, average latency rate of map-reduce faster than the Hive default execution engine.

Finally, in future work, we will test other techniques and try to make a combination between some of them to see if it speeds up the response time of hive query.

REFERENCES

[1] "Apache Hive TM." [Online]. Available: http://hive.apache.org/. [Accessed: 18-May-2017].

[2] "What is Apache Hive? - YouTube." [Online]. Available: https://www.youtube.com/watch?v=WkuIWJNjtDE. [Accessed: 18-May-2017].

[3] "Query optimization techniques in Apache Hive." [Online]. Available: https://www.slideshare.net/ZaraTariq/query-optimization-techniques-in-apache-hive. [Accessed: 18-May-2017].

[4] A. Gruenheid, E. Omiecinski, and L. Mark, "Query Optimization Using Column Statistics in Hive," *ACM*, 2011.

[5] A. Marashdeh, "Comparison query optimization technique in hive." [Online]. Available: http://www.academia.edu/14173420/Comparison_query_optimization_technique_in_hive. [Accessed: 18-May-2017].

[6] V. Garg, "Optimization of Multiple Queries for Big Data with Apache Hadoop/Hive," presented at the International Conference on Computational Intelligence and Communication Networks, 2015.

[7] A. Thusoo and J. Sen Sarma, "Hive - A Warehousing Solution Over a Map-Reduce Framework."

[8] "Query optimization - Wikipedia." [Online]. Available: https://en.wikipedia.org/wiki/Query_optimization. [Accessed: 19-Apr-2017].

[9] "Chapter 4. Query Optimization - Hive Performance Tuning." [Online]. Available: https://docs.hortonworks.com/HDPDocuments/HDP2/HDP-2.4.0/bk_performance_tuning/content/ch_query_optimization_hive.html. [Accessed: 19-Mar-2017].

[10] john Wiley, "Data Science & Big Data Analytics," .

[11] "MapReduce Quick Guide." [Online]. Available: https://www.tutorialspoint.com/map_reduce/map_reduce_quick_guide.htm. [Accessed: 21-Mar-2017].

[12] "HIVE 0.14 Cost Based Optimizer (CBO) Technical Overview," *Hortonworks*, 02-Mar-2015. [Online]. Available: https://hortonworks.com/blog/hive-0-14-cost-based-optimizer-cbo-technical-overview/. [Accessed: 21-May-2017].

[13] "Cost-based optimization in Hive - Apache Hive - Apache Software Foundation." [Online]. Available: https://cwiki.apache.org/confluence/display/Hive/Cost-based+optimization+in+Hive. [Accessed: 21-Apr-2017].

[14] "Download QuickStarts for CDH 5.10." [Online]. Available: https://www.cloudera.com/downloads/quickstart_vms/5-10.html. [Accessed: 25-May-2017].