

Study of HIVE Tool for Big Data Used in Facebook

Mr. Chandole N. S.¹ Miss. Kulkarni C.S.² Ms. Surwase M.D.³ Ms. Shelake S.M⁴

^{1,2,3,4}Assistant Professor

^{1,2,3,4}Department of Computer Science and Engineering

^{1,2,3,4}FTCCOER

Abstract— The time period ‘massive information’ is used for collections of large datasets that encompass big extent, excessive velocity, and a diffusion of data that is increasing day by day. The usage of traditional facts control systems, it is hard to technique huge data. Consequently, the Apache software basis brought a framework known as Hadoop to resolve huge statistics control and processing challenges. Apache Hive is a widely used records warehouse machine for Apache Hadoop, and has been followed via many companies for various huge statistics analytics programs. carefully running with many customers and companies, we have identified several shortcomings of Hive in its document formats, query making plans, and question execution, which are key factors determining the performance of Hive to be able to make Hive continuously fulfill the requests and requirements of processing an increasing number of excessive volumes information in a scalable and efficient way, we have set desires related to storage and runtime overall performance in our efforts on advancing Hive.

Key words: HIVE, Facebook, Apache Hadoop

I. INTRODUCTION

Hive is an information distribution center foundation instrument to handle organized information in Hadoop. It dwells on top of Hadoop to abridge Huge Information, and makes questioning and dissecting simple. At first Hive was created by Facebook, later the Apache Programming Establishment took it up and created it further as an open source under the name Apache Hive. It is utilized by various organizations. For instance, Amazon utilizes it in Amazon Versatile MapReduce.

II. COMPONENTS OF HIVE

It stores pattern in a database and handled information into HDFS.

It is intended for OLAP.

It gives SQL sort dialect to questioning called HiveQL or HQL.

It is natural, quick, versatile, and extensible.

In Hive, tables and databases are made first and after that information is stacked into these tables.

- Hive as information distribution center intended for overseeing and questioning just organized information that is put away in tables.
- While managing organized information, Outline doesn't have advancement and ease of use elements like UDFs yet Hive structure does. Inquiry streamlining alludes to a compelling method for question execution regarding execution.
- Hive's SQL-motivated dialect isolates the client from the multifaceted nature of Guide Diminish programming. It reuses recognizable ideas from the social database world,

for example, tables, lines, sections and pattern, and so forth for simplicity of learning.

- Hadoop's customizing takes a shot at level documents. Along these lines, Hive can utilize index structures to "parcel" information to enhance execution on specific inquiries.
- A new and important component of Hive i.e. Metastore used for storing schema information. This Metastore typically resides in a relational database

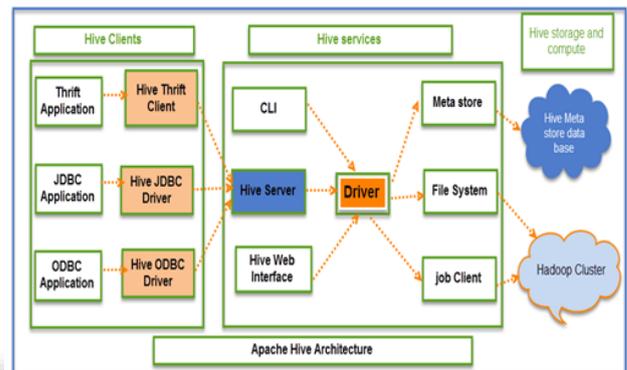


Fig. 1: General Hive System Architecture

The Hive Consists of Mainly 3 core parts

- 1) Hive Clients
- 2) Hive Services
- 3) Hive Storage and Computing

A. Hive Clients:

Hive provides different drivers for communication with a different type of applications. For Thrift based applications, it will provide Thrift client for communication. For Java related applications, it provides JDBC Drivers. Other than any type of applications provided ODBC drivers. These Clients and drivers in turn again communicate with Hive server in the Hive services.

B. Hive Services:

Client interactions with Hive can be performed through Hive Services. If the client wants to perform any query related operations in Hive, it has to communicate through Hive Services. CLI is the command line interface acts as Hive service for DDL (Data definition Language) operations. All drivers communicate with Hive server and to the main driver in Hive services as shown in above architecture diagram. Driver present in the Hive services represents the main driver, and it communicates all type of JDBC, ODBC, and other client specific applications. Driver will process those requests from different applications to meta store and field systems for further processing.

C. Hive Storage and Computing:

Hive services such as Meta store, File system, and Job Client in turn communicates with Hive storage and performs the following actions

- Metadata information of tables created in Hive is stored in Hive "Meta storage database".
- Query results and data loaded in the tables are going to be stored in Hadoop cluster on HDFS.

D. Facebook uses HIVE:

The entire information processing infrastructure in Facebook prior to 2008 was built around a records warehouse constructed using an industrial RDBMS. The facts that we had been generating became developing very fast - as an instance we grew from a 15TB information set in 2007 to a 700TB facts set nowadays. The infrastructure at that time changed into so inadequate that a few day by day data processing jobs have been taking greater than an afternoon to procedure and the situation turned into just getting worse with every passing day. Hadoop changed into no longer smooth for end users, mainly for the ones customers who were not familiar with map reduce. Quit users had to write map-lesser programs for simple obligations like getting raw counts or averages. Hadoop lacked the expressiveness of famous question languages like square and as a end result customers ended up spending hours (if now not days) to put in writing packages for even easy evaluation. It turned into very clean to us that so that you can simply empower the corporation to analyze this statistics more productively, we needed to enhance the query abilities of Hadoop. Bringing these records towards customers is what inspired us to construct Hive in January 2007. Our imaginative and prescient was to bring the acquainted concepts of tables, columns, partitions and a subset of sq. to the unstructured global of Hadoop, whilst still retaining the extensibility and flexibility that Hadoop loved. Hive changed into open sourced in August 2008 and since then has been used and explored through a quantity of Hadoop users for his or her facts processing needs. Hive systems information into the nicely-understood database standards like tables, columns, rows, and partitions. It supports all of the essential primitive sorts - integers, floats, doubles and strings - in addition to complex kinds which includes maps, lists and structs. The latter may be nested arbitrarily to construct more complicated types. Similarly, Hive lets in users to extend the device with their very own kinds and functions. The query language may be very much like sq. and therefore can be without difficulty understood with the aid of each person acquainted with square. There are some nuances within the records version, type machine and HiveQL that are special from conventional databases and that have been inspired with the aid of the reports won at facebook. We are able to spotlight those and other info in this segment.

The following components are the main building blocks in Hive:

Metastore - The component that stores the system catalog and metadata about tables, columns, partitions etc.

Driver - The component that manages the lifecycle of a HiveQL statement as it moves through Hive. The driver also maintains a session handle and any session statistics.

Query Compiler - The component that compiles HiveQL into a directed acyclic graph of map/reduce tasks.

Execution Engine - The component that executes the tasks produced by the compiler in proper dependency

order. The execution engine interacts with the underlying Hadoop instance.

Hive Server- The component that provides a thrift interface and a JDBC/ODBC server and provides a way of integrating Hive with other applications.

Client's components like the Command Line Interface (CLI), the web UI and JDBC/ODBC driver.

Extensibility Interfaces which include the SerDe and ObjectInspector interfaces already described previously as well as the UDF (User Defined Function) and UDAF (User Defined Aggregate Function) interfaces that enable users to define their own custom functions.

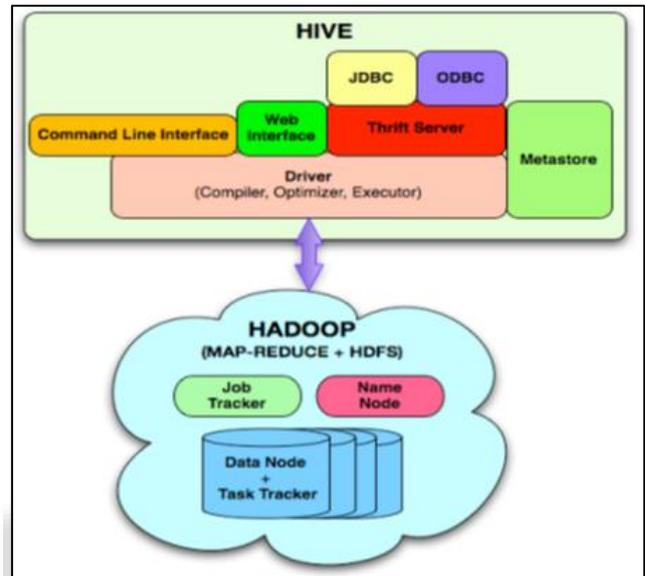


Fig. 2: Hive System Architecture for facebook

Metastore-The Metastore acts as the system catalog for Hive. It stores all the information about the tables, their partitions, the schemas, the columns and their types, the table locations etc. This information can be queried or modified using a thrift ([7]) interface and as a result it can be called from clients in different programming languages.

Query Compiler -The metadata stored in the Metastore is used by the query compiler to generate the execution plan. Similar to compilers in traditional databases, the Hive compiler processes HiveQL statements in the sequence of Parse, Type checking and Semantic Analysis and optimization.

Execution Engine- Finally the tasks are executed in the order of their dependencies. Each dependent task is only executed if all of its prerequisites have been executed.

III. HIVE USAGE IN FACEBOOK

Hive and Hadoop are used extensively in Facebook for different kinds of data processing. Currently our warehouse has 700TB of data (which comes to 2.1PB of raw space on Hadoop after accounting for the 3 way replication). We add 5TB (15TB after replication) of compressed data daily. Typical compression ratio is 1:7 and sometime more than that. On any particular day more than 7500 jobs are submitted to the cluster and more than 75TB of compressed data is processed every day. With the continuous growth in the Facebook network we see continuous growth in data. At the same time as the company scales, the cluster also has to scale with the growing users. More than half the workload is on adhoc queries whereas the rest is for reporting dashboards.

Hive has enabled this kind of workload on the Hadoop cluster in Facebook because of the simplicity with which adhoc analysis can be done. However, sharing the same resources by the adhoc users and reporting users presents significant operational challenges because of the unpredictability of adhoc jobs. Many times these jobs are not properly tuned and therefore consume valuable cluster resources. This can in turn lead to degraded performance of the reporting queries, many of which are time critical. Resource scheduling has been somewhat weak in Hadoop and the only viable solution at present seems to be maintaining separate clusters for adhoc queries and reporting queries. There is also a wide variety in the Hive jobs that are run daily. They range from simple summarization jobs generating different kinds of rollups and cubes to more advanced machine learning algorithms. The system is used by novice users as well as advanced users with new users being able to use the system immediately or after an hour long beginners training. A result of heavy usage has also lead to a lot of tables generated in the warehouse and this has in turn tremendously increased the need for data discovery tools, especially for new users. In general the system has enabled us to provide data processing services to engineers and analysts at a fraction of the cost of a more traditional warehousing infrastructure Different modes of Hive. Hive can operate in two modes relying on the scale of facts nodes in Hadoop.

These modes are,

- Local mode
- Map reduce mode

While to use local mode: •If the Hadoop installed underneath pseudo mode with having one information node we use Hive on this mode

- If the statistics length is smaller in time period of restrained to single local device, we are able to use this mode
- Processing will be very rapid on smaller information units present in the local system

When to use Map reduce mode:

- If Hadoop is having multiple data nodes and data is distributed across different node we use Hive in this mode
- It will perform on large amount of data sets and query going to execute in parallel way
- Processing of large data sets with better performance can be achieved through this mode

In Hive, we can set this property to mention which mode Hive can work? By default, it works on Map Reduce mode and for local mode you can have the following setting.

Hive to work in local mode set

```
SET mapred.job.tracker=local;
```

From the Hive version 0.7 it supports a mode to run map reduce jobs in local mode automatically.

IV. CONCLUSION

We have entered an era of Big Data. The paper describes the concept of Big Data along with 3 Vs., Volume, Velocity and variety of Big Data. The paper also focuses on Big Data processing. . Hive is a work in progress. It is an open-source project, and is being actively worked on by Facebook as well as several external contributors. Hive currently has a naive rule-based optimizer with a small number of simple rules. We

plan to build a cost based optimizer and adaptive optimization techniques to come up with more efficient plans

ACKNOWLEDGMENTS

We would like to thank the Hive development community. In recent years, more than 100 developers have made technical efforts to improve Hive on more than 3000 issues. We thank those individuals who have operated, tested, supported and documented Hive in various platforms and applications. We are grateful to the Hive user community who has provided us with numerous valuable feedbacks that drive the development and advancements of Hive. We thank anonymous reviewers for their constructive comments.

REFERENCES

- [1] Apache Hadoop .Available at <http://wiki.apache.org/hadoop>.
- [2] Facebook Lexicon at <http://www.facebook.com/lexicon>.
- [3] Hive wiki at <http://www.apache.org/hadoop/hive>.
- [4] Hadoop Map-Reduce Tutorial at http://hadoop.apache.org/common/docs/current/mapred_tutorial.html.
- [5] Hadoop HDFS User Guide at http://hadoop.apache.org/common/docs/current/hdfs_user_guide.html.
- [6] Mysql list partitioning at <http://dev.mysql.com/doc/refman/5.1/en/partitioning-list.html>.
- [7] Apache Thrift. Available at <http://incubator.apache.org/thrift>.
- [8] Data Nucleus Available at <http://www.datanucleus.org>.
- [9] S.Vikram Phaneendra & E.Madhusudhan Reddy "Big Data- solutions for RDBMS problems- A survey" In 12th IEEE/IFIP Network Operations & Management Symposium (NOMS 2010) (Osaka, Japan, Apr 19{23 2013).
- [10] Kiran kumara Reddi & Dnysl Indira "Different Technique to Transfer Big Data : survey" IEEE Transactions on 52(8) (Aug.2013) 2348 { 2355 }