**PROJECT DRAFT**

**Project Topic**: Big Data Analysis on H1-B Visa Petitions 2011-16 using Hive , SparkSQL and MapReduce and their performance comparisons

**Introduction:**

Nowadays the challenges of collecting, storing and analysing data are addressed by big data systems. One such famous one is Hive that supports HiveQL which compiles MapReduce, another one is SparkSQL that has aced the big data systems by using in memory distributed computing. MapReduce is a programming model and an implementation for processing and generating large data sets. This project is focused on comparing these big data systems and their query performances for a given data set and analysing the results.[2]

The SQL engines execute queries for getting specific insights to answer the test cases or provide them with findings which can help to do so. It also compares the EMR data nodes and sees if increased the processing speed and query performance. This literature review compares Hive, SparkSQL, Presto and Phoenix. Their uses, architecture. The objectives of the project, the design of the data processing work and the results obtained.

**List of big data systems:**

* Apache Hive
* Apache Spark
* Apache Presto
* Apache Phoenix

**Apache Hive**

Hive is a data warehouse that is built on top of Hadoop and does easy data summarization, ad-hoc queries, analysis of large datasets and integrate them with Hadoop. It integrates with data center technologies using JDBC/ODBC interface.[8] It has SQL-like interface called as HiveQL that can execute MapReduce or Apache Spark which makes simple for more users to process and analyse unlimited amounts of data. The shared data structures allow users to easily read and write data without worrying about where the data is stored, what format it is, or redefining the structure for each tool. There is faster batch processing in Hive and has performance boosts over MapReduce.[9]

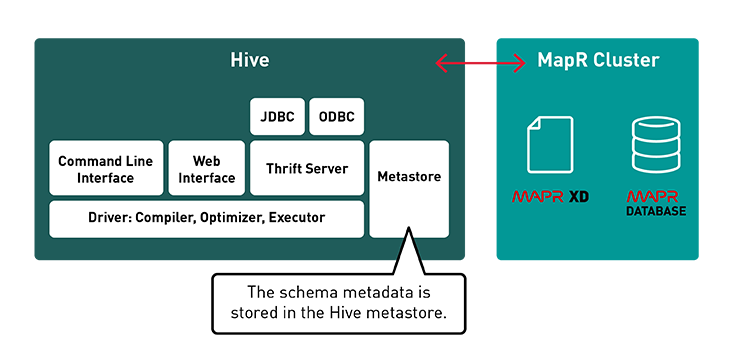


Figure 1: Hive Metastore and working [8]

**Apache Spark:**

Spark is an open source community framework of Hadoop that is on the top of Hadoop Distributed File System (HDFS). Spark is 100 times faster as compared to MapReduce. Big data systems have been proposed to address the challenges of big data such as collecting, storing, and analysing data. There are several studies have been performed to evaluate these two frameworks and showed that in most cases Spark SQL is faster than Hive on MapReduce, but in some cases related to joining large tables Spark SQL is slower than

Hive on MapReduce [1]

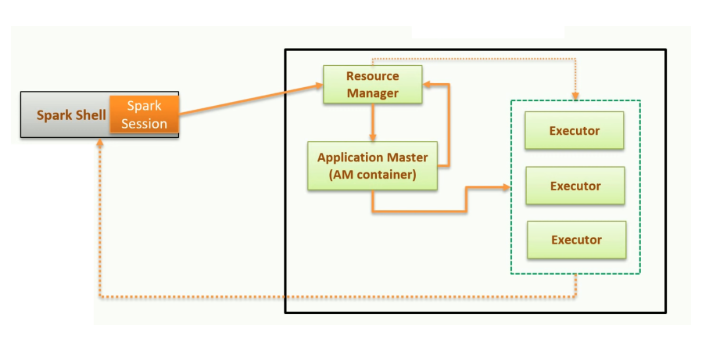


Figure 2: Spark Architecture [10]

**Apache Presto:**

Presto is an open source, distributed SQL query engine that can query any size of data. It supports non-relational sources like HDFS, Amazon S3, Cassandra, MongoDB etc. along with relational data sources like MySQL, PostgreSQL and  [Amazon Redshift](https://aws.amazon.com/redshift/). It can query data where it is stored without needing to move data into a separate system. It is used in Facebook, Netflix, Airbnb etc. As it runs on top of Hadoop it uses a massively parallel processing (MPP). It has one coordinator node working in synch with multiple worker nodes. Users submit their SQL query to the coordinator which uses a custom query and execution engine to parse, plan, and schedule a distributed query plan across the worker nodes.[11]

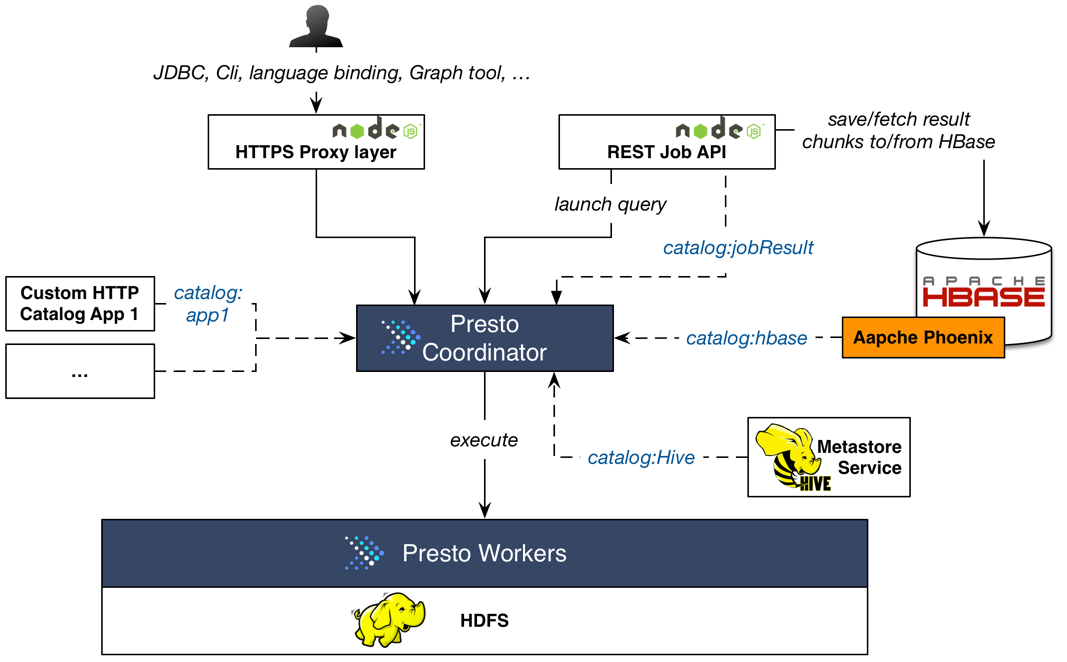


Figure 3: Apache Presto Architecture [5]

**Apache Phoenix:**

Apache Phoenix is an add-on for Apache HBase which provides a programmatic ANSI SQL interface. It is used for deploying machine learning-based applications, web scale and mobile apps, customer-facing dashboards, fraud analysis etc. The data querying can be done using SQL. Phoenix does a full scan although working faster as it chunks up query using guidePosts, it uses more threads for a single chunk. The queries are run in parallel on the client using configurable number of threads. Aggregation is done in a coprocessor on the server-side, reducing the amount of data that is returned to the client.[12]

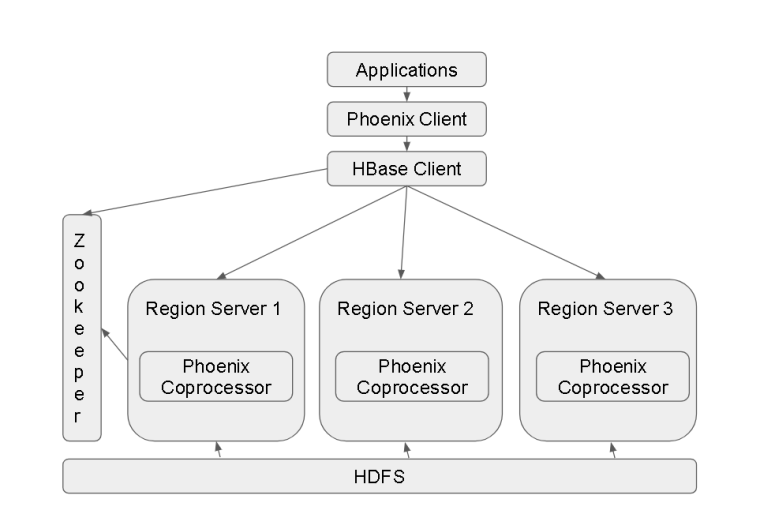
****

Figure 4: Phoenix Architecture [6]

**Project Description:**

The project would solve the following test cases:

1. Maximum, Minimum, Average based on the columns.
2. Which worksite has the popular job title?
3. Which job title has the case status as “Certified”, “Certified -withdrawn”, “Denied”, “Withdrawn”?
4. How many H1-B filers are “Certified” having the fulltime status as “Yes”?
5. What is the general trend of the employer’s H1-B status in the period 2011-16?
6. What is the general trend of the employer name with respect to the year?
7. Is the number of petitions with Data Engineer job title increasing over time?
8. Which part of the US has the most Data Engineer jobs?
9. Which industry has the greatest number of Data Scientist positions?
10. Which employers file the most petitions each year?

The project would be on a dataset that contains 100,000 to 500,000 records and performance analysis would be done using Hive with Tez and MapReduce. It would also compare the performance with SparkSQL. The project would also analyse the performance if we increase the EMR data nodes from 2 to 3 or use 5. It would load the data from the csv file and store it into the hive table where the processing would be carried out.

**Conclusion:**

Thus, the project helps in understanding and analysing the job trends for foreigners in US. It shows that how many cases were certified with H1-B and which is the trending field for H1-B. The exploratory data analysis will also show a graphical representation of the data in Tableau for providing an insight to the analysis.

**References:**

1. <https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3365403>
2. <https://www.researchgate.net/publication/320577378_Performance_Evaluation_between_Hive_on_MapReduce_and_Spark_SQL_with_BigBench_and_PAT>
3. <https://www.tutorialspoint.com/hadoop/hadoop_introduction.htm>
4. <https://www.google.com/search?q=diagram+for+spark,+hive+presto+and+phoenix&rlz=1C1EKKP_enIN781IN781&sxsrf=ACYBGNSSDck5eCzOSsdvmWpzXu4Obu4Teg:1573365650708&tbm=isch&source=iu&ictx=1&fir=JsacuxlZgI0iqM%253A%252CdRZfeBRWP7YAjM%252C_&vet=1&usg=AI4_-kRlW5C7CMytrRzDZwxJrvPUamqk2w&sa=X&ved=2ahUKEwjX-djl-97lAhUiwFkKHScaCqkQ9QEwAHoECAMQBg#imgrc=4WRGAXigpOx6KM:&vet=1>
5. <https://labs.gree.jp/blog/2014/12/12838/>
6. <https://mohdnaeem.wordpress.com/2018/01/24/apache-phoenix-another-query-engine-with-a-sql-interface-fine-tuned-for-performance-with-hbase/>
7. <https://arxiv.org/pdf/1804.00224.pdf>
8. <https://mapr.com/products/apache-hive/>
9. <https://www.cloudera.com/products/open-source/apache-hadoop/apache-hive.html>
10. <https://towardsdatascience.com/getting-started-with-apache-spark-ad9d59e71f6f>
11. <https://aws.amazon.com/big-data/what-is-presto/>
12. <https://www.cloudera.com/products/open-source/apache-hadoop/apache-phoenix.html>