An Empirical Study on Data Processing Using Hadoop

Mrs. Aparna Abhijit Kale[#1], Mrs. Asawari Anant Sawant[*2]

#Assistant Professor, K.B.Joshi IIT BCA College, Pune

Abstract— **Data has turn out to be a very central part in almost every field. But the factual hurdle in this is which the expedient data is and where to store this huge data. In this scenario, data processing and storage comes in picture. Data is nothing but collection of meaningful items, these items are processed and analysed using 3 major steps: processing of raw data, normalization and statistical analysis. And the storage of this analysed data can be done either manually or using the new emerging and growing technologies like Big Data, Cloud computing, Hadoop etc.In Hadoop, the giant data collected, is divided into numerous clusters, and each cluster is individually processed on distributed servers and finally executed using distributed analysis application on each cluster. The biggest advantage of using Hadoop is that, even though the individual cluster collapse, the process of analysis continues.**

Keywords— **Data analysis, data processing, data storage, Big Data cluster, Hadoop, HDFS. Map Reduce, YARN Framework.**

I. Introduction

In today's world processing Big Data has become a crucial task. The data storage capacity has moved from peta bytes to zeta bytes. Every day, the data is collected from variety of sources like posts on social media, images, videos, etc. The data processing of such a Big Data is a very crucial task. Data processing is nothing but mining important data gathered from various sources like posts on social media, images, videos, etc and removing the unnecessary data. Lots of technologies are emerging in the market today for efficient data processing; one of them which is a boom in market is Hadoop.

II. Hadoop components

Hadoop is open source commodity software developed by Apache. This software was mainly written in Java language. In Hadoop, the Big Data is collected from variety of sources like posts on social media, images, videos, etc. is divide into different clusters of uniform sized block of size 64 MB. These clusters are stored on distributed file system on different machines rather than storing it on main memory of single standalone server. These clusters are then processed individually to get the required data. The Hadoop architecture comprises of four main components.

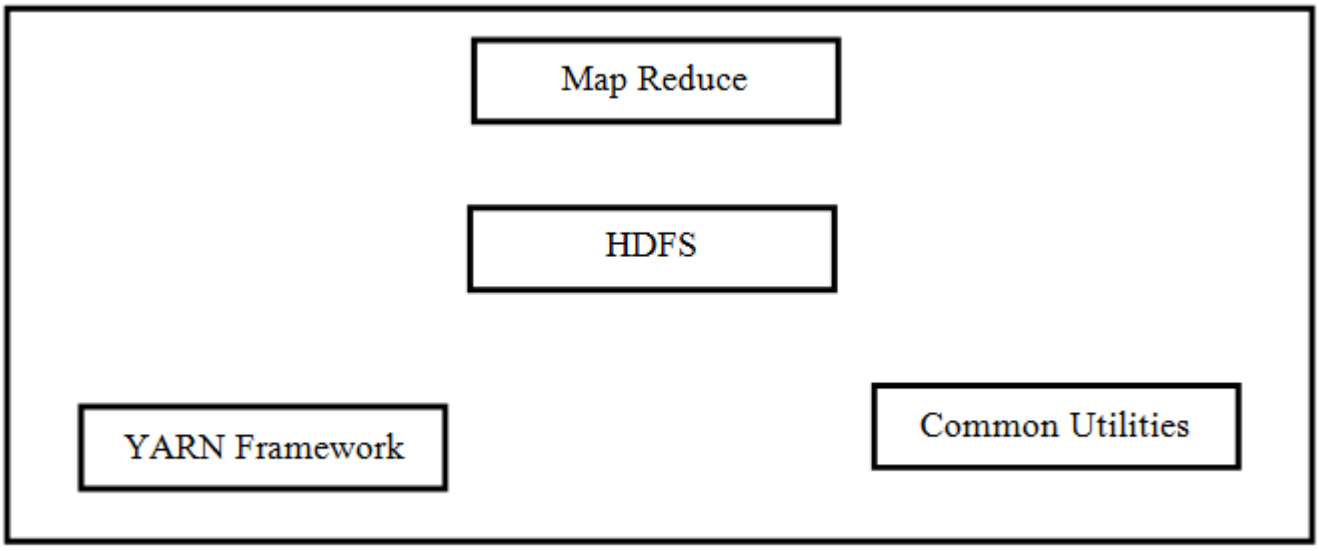

Fig.1 Hadoop Architecture

The 4 components in Hadoop architecture are:

1. COMMON UTILITIES:
   As the name implies, this component of Hadoop provides all the necessary services or functions that are required for the proper functioning of all the other Hadoop components which include HDFS, Map Reduce, and YARN framework. These utilities are in the form of java library files and scripts.

2. YARN FRAMEWORK(YET ANOTHER RESOURCE NEGOTIATOR):

This component of Hadoop deals 2 main tasks i.e. job scheduling and resource management.

Job scheduling means, when more than one job arrives at the server machine this sub component assigns the priority to each job and depending upon that the job with highest priority is serviced first and later the least priority jobs are serviced. The sub component is resource management as the name implies it deals with allocating the required amount of resources to each job and if excess resources are allocated to the particular job deallocating those resources.

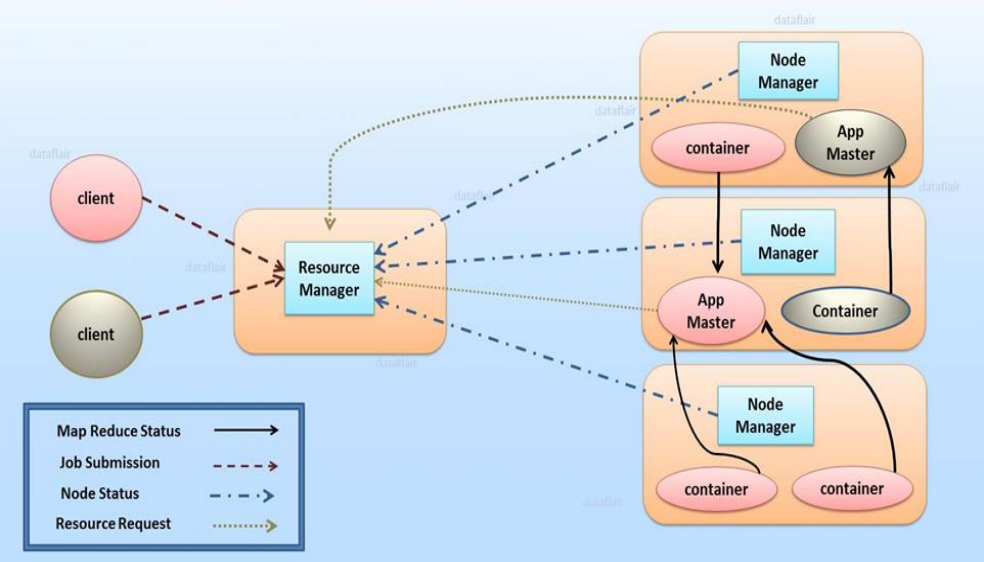

Fig.2 YARN Framework Architecture

3. HDFS(HADOOP DISTRIBUTED FILE SYSTEM):

HDFS is based on GFS (Google File System). This is the main component of Hadoop architecture. As it is a file system it is used to store the data but in this case the Big Data which can be in structured, semi-structured or unstructured format and it is stored in distributed manner. HDFS has 2 main sub modules i.e. name node and data node. These sub components interact with each other by following master slave architecture, where there is a single name node which will act as a master and there can be many data nodes which will act as slaves.

- NAME NODE:

For every single file which is received as an input from the user is taken by the name node. The name node stores Meta data about the file i.e. its size, location, etc. The name node acts as a master who divides the inputted file into number of uniform sized blocks of 64 MB and maps it to the data nodes which will be acting as slaves. The name node has an authority to create a replica of specific block if that block holds any important data. If any of the data node damages, the name node will distribute that task among the remaining data nodes. The name node also carries out the tasks like opening, closing, renaming files and directories.

- DATA NODE:

The data node stores the actual data from the file in the form of blocks. The data node acts as a slave which follows all the instructions from the name node and acts accordingly. It is the job of data nodes to inform the name node about their current status. The data nodes carries out the tasks like read and write requests from the client's file systems, and it also performs block creation, deletion and replication upon the instructions received from the name node.

The name node and data nodes are commodity software which run on GNU/Linux operating system and is written in Java language which makes HDFS a portable distributed file system. The system is designed in such a way that the data from client's file never gets store in the name node. The name node gives instructions to the data nodes to create replicas of significant data for fault tolerance.

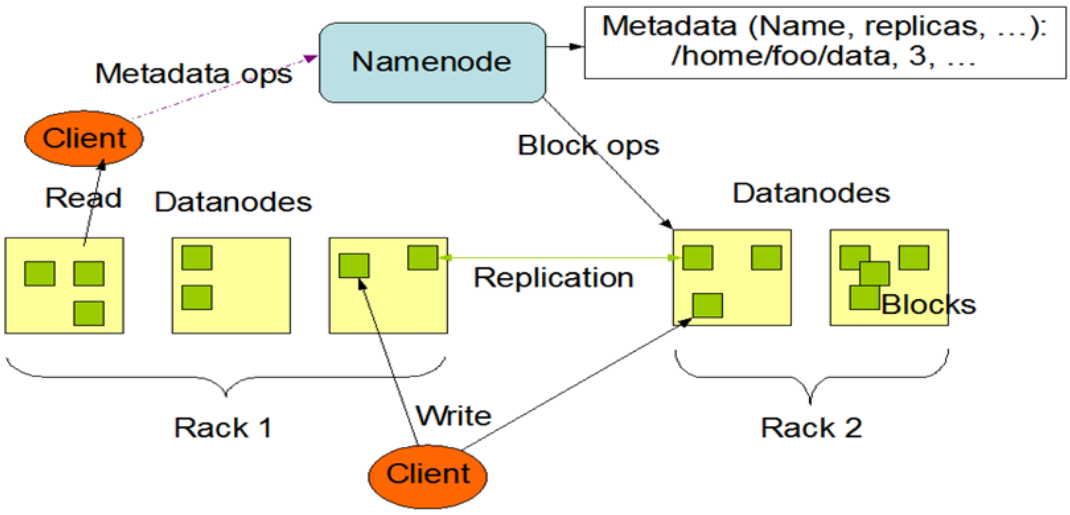

Fig.3 HDFS Architecture

4. MAP REDUCE

This is the last component of Hadoop architecture which deals with actual processing of the Big Data. The data processing is done in distributed and parallel manner which reduces the access time. This component is also based on master-slave architecture. Map Reduce basically follows two phases:

- MAP:

The Map phase accepts the input from HDFS. The input is nothing but the set of data sets which are divided into tuples of key-value pair. Each tuple is then send to distributed machines in a cluster for processing. All the clusters are then processed in parallel. The function that all these tasks are known as map function.These intermediate key-value pairs are then shuffled to provide input to the reduce phase. The process of transferring data from the map function to reduce function is known as shuffling which acts as an input to the reducer.

- REDUCE:

The reduce phase accepts the input from the map function. The input is nothing but the intermediate key-value pairs which are grouped together into another set of tuples based on the keys of the tuples.Each tuple is then send to distributed machines in a cluster for processing.All the clusters are then processed in parallel. The function that all these tasks are known as reduce function. The output of reduce function is send to the output function which automatically sorts the key-value pairs and finally send for displaying it to the client. The reduce function stores its output in HDFS.

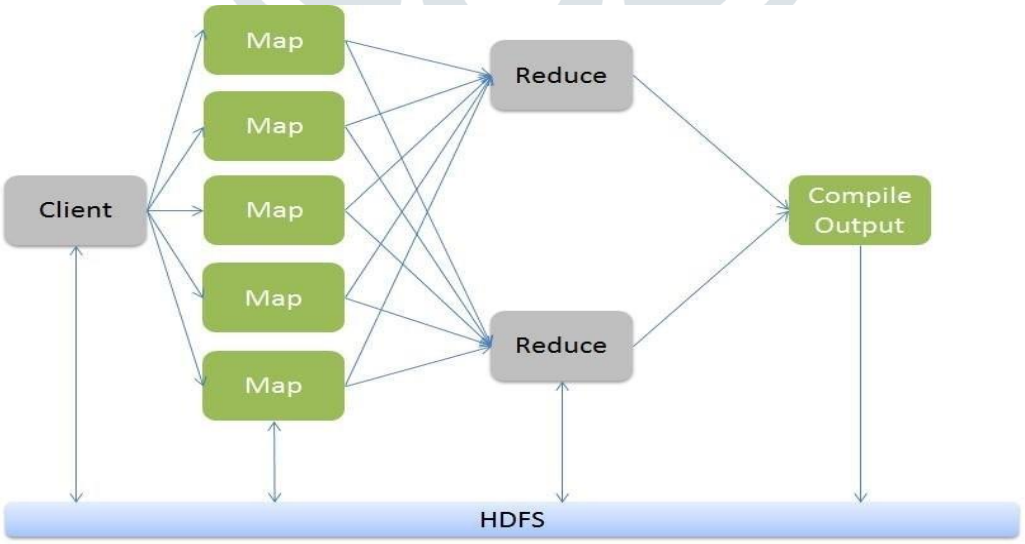

Fig.4 Map Reduce Architecture

HOW MAP REDUCE WORKS?

Let's understand this with an example –Consider you have following input data for your MapReduce Program
Deer, Bear, River
Car, Car, River
Deer, Car, Bear

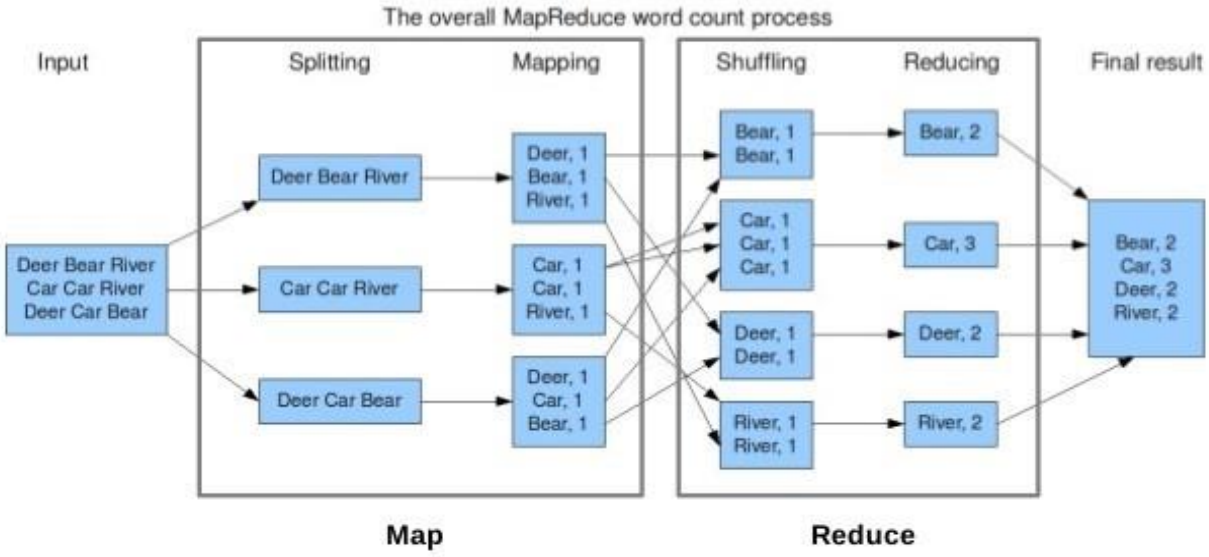

Fig.5 Map Reduce Example

III. Different big data processing tools

HPCC is a big data tool developed by LexisNexis Risk Solution. It delivers on a single platform, a single architecture and a single programming language for data processing.

A. *Storm:*
Storm is a free and open source big data computation system. It offers distributed real-time, fault-tolerant processing system. With real-time computation capabilities.

B. *Qubole:*
Qubole Data is Autonomous Big data management platform. It is self-managed, self-optimizing tool which allows the data team to focus on business outcomes.

C. *Cassandra:*
The Apache Cassandra database is widely used today to provide an effective management of large amounts of data.

D. *Statwing:*
Statwing is an easy-to-use statistical tool. It was built by and for big data analysts. Its modern interface chooses statistical tests automatically.

E. *CouchDB:*
CouchDB stores data in JSON documents that can be accessed web or query using JavaScript. It offers distributed scaling with fault-tolerant storage. It allows accessing data by defining the Couch Replication Protocol.

F. *Pentaho:*
Pentaho provides big data tools to extract prepare and blend data. It offers visualizations and analytics that change the way to run any business. This Big data tool allows turning big data into big insights.

G. *Flink:*
Apache Flink is an open-source stream processing Big data tool. It is distributed, high-performing, always-available, and accurate data streaming applications.

H. *Cloudera*:
Cloudera is the fastest, easiest and highly secure modern big data platform. It allows anyone to get any data across any environment within single, scalable platform.

I. *Rapidminer*:
RapidMiner is an open source big data tool. It is used for data prep, machine learning, and model deployment. It offers a suite of products to build new data mining processes and setup predictive analysis.

J. *Kaggle*:
Kaggle is the world's largest big data community. It helps organizations and researchers to post their data & statistics. It is the best place to analyse data seamlessly.

I .References:

- Article on Hadoop technology [1],[6]
- Article on Data processing [2],[5],[7]
- Journal on data processing [3]
- Research paper on Big Data [4],[8]
- Book on MapReduce algorithm for Big Data analysis [9]
- Article on MapReduce[10],[11],[12],[16],[17]
- Article on HDFS [13],[14],[15]
- Big Data processing tools[18],[19]
- Big Data Analytics: A Literature Review Paper[21]

IV .Conclusion

Now- a-days, every second data is produced from different sources and variety of rates. The processing of these huge amounts of data is a crucial problem today.In this paper we have discussed Hadoop tool for Big Data processing in detail. We have also discussed some Hadoop components which are used to support the processing of large data sets in distributed computing environments. In future we can use some clustering techniques and check the performance by implementing it in Hadoop.

Acknowledgment

We are very thankful to the online research articles and research papers without them the concepts would not have been that clear to us. We would also like to show our gratitude towards our institute and our colleagues to share their thoughts with us.

References

1 https://searchdatamanagement.techtarget.com/definition/Hadoop
2 https://planningtank.com/computer-applications/data-processing
3 https://pubs.acs.org/doi/abs/10.1021/pr500665j
4 https://www.cs.helsinki.fi/u/jilu/paper/BigdataSurvey02.pdf
5 https://www.slideshare.net/UdaybhaskarMogallapu/data-processing-and-report-writing
6 https://readwrite.com/2013/05/23/hadoop-what-it-is-and-how-it-works/
7 https://en.wikipedia.org/wiki/Data_processing
8 https://www.researchgate.net/publication/281202956_Big_Data_Analysis_using_Hadoop_A_Survey
9 https://link.springer.com/chapter/10.1007/978-3-642-37134-9_3
10 https://www.thegeekstuff.com/2014/05/map-reduce-algorithm/
11 https://www.tutorialspoint.com/map_reduce/map_reduce_algorithm.htm
12 https://data-flair.training/blogs/shuffling-and-sorting-in-hadoop/
13 https://intellipaat.com/blog/what-is-hdfs/
14 https://www.tutorialspoint.com/hadoop/hadoop_hdfs_overview.htm
15 https://searchdatamanagement.techtarget.com/definition/Hadoop-Distributed-File-System-HDFS
16 https://www.tutorialspoint.com/hadoop/hadoop_mapreduce.htm
17 https://www.dezyre.com/hadoop-tutorial/hadoop-mapreduce-tutorial-
18 https://www.guru99.com/big-data-tools.html
19 https://www.whizlabs.com/blog/big-data-tools/
20 https://analyticstraining.com/8-big-data-tools-need-know/
21 https://www.researchgate.net/publication/264555968_Big_Data_Analytics_A_Literature_Review_Paper