IJSTE - International Journal of Science Technology & Engineering | Volume 5 | Issue 1 | July 2018
ISSN (online): 2349-784X

Big Data Analysis using Hadoop Technologies

Aiswariya M.
UG Student
Sri Krishna Arts & Science College, India

Swathi V.
UG Student
Sri Krishna Arts & Science College, India

Vivekavarthini K.
UG Student
Sri Krishna Arts & Science College, India

Brindha K.
UG Student
Sri Krishna Arts & Science College, India

Nanthini S.
UG Student
Sri Krishna Arts & Science College, India

Abstract

Big data is a term that describes the large volume of data. That contains data in the form of both structured and un-structured data. These data sets are very large and complex so that it becomes difficult to process using traditional data processing applications. To process this enormous amount of data Hadoop can be used. Hadoop is an open source software project that enables the distributed processing of large data sets across clusters of commodity servers. It is designed to scale up from a single server to thousands of machines, with a very high degree of fault tolerance. The technologies used by big data application to handle the massive data are Hadoop, Map Reduce, Apache Hive, No SQL and HPCC. These technologies handle massive amount of data in KB, MB, GB, TB, PB, EB, ZB, YB and BB.

Keywords: Big Data, Volume, Variety, Velocity, Value, Veracity, Hadoop, HDFS, Map Reduce, Hadoop-Eco System

I. INTRODUCTION

Big data: The whole world has a tendency to produce quintillion bytes of data. This much amount of data comes from everywhere. Big Data is as a collection of large dataset that cannot be processed using traditional computing techniques. Big Data is not merely a data rather it has become a complete subject which involve various tools, techniques and framework. The need of big data generated from the large companies like Facebook, yahoo, Google, YouTube etc. And also Google contains large amount of information. Whose volume (size), complexity and rate of growth (velocity) make them difficult to be captured, managed, processed or analyzed by conventional technology and tools such as relational databases. There are various technologies in the market from different vendors including Amazon, IBM, Microsoft, etc., to handle such big data, the data in it will be of three types.

- Structured data: Relational data.
- Semi Structured data: XML data.
- Unstructured data: Word, PDF, Text, Media Logs.

A. Parameters of Big Data

1) Volume

Data is growing day by day of all types ever MB, PB, YB, ZB, KB, TB of information. These data results into large files. Excessive volume of data is main issue of storage. This main issue is resolved by reducing storage cost. Data volumes are expected to grow 50 times by 2020.

2) Variety

Variety makes the data too big. The files comes in various formats and of any type, it may be structured or unstructured such as text, audio, videos, log files and more variety of data.

3) Velocity

Velocity refers to the speed at that new data is generated and the speed at that data moves around. In Some organisation's data velocity is main challenge. The social media messages and credit card transactions done in millisecond and data generated by this putting in to databases.

4) Value

It is a most important v in big data. Value is main buzz for big data because it is important for businesses, IT infrastructure system to store large amount of values in database.

5) Veracity

Veracity refers to the messiness or trustiness of the data. When we dealing with high volume, velocity and variety of data. The all of data are not going 100% correct, there will be dirty data. Big data and analytics technologies work with these types of data.

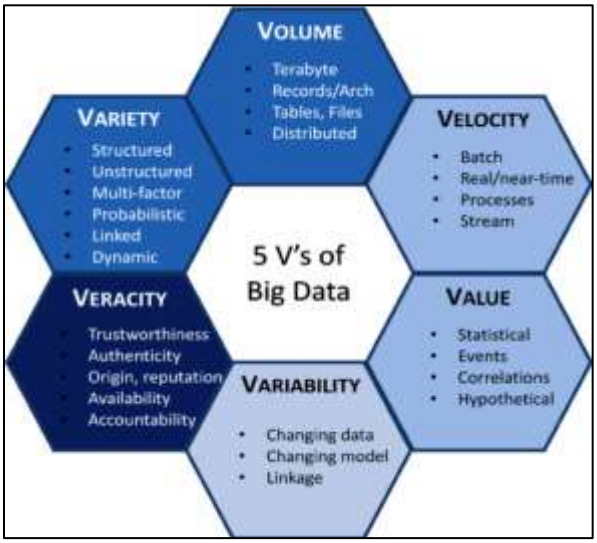

Fig. 1: Five V's of Big Data

II. HADOOP

Hadoop is an Apache open source framework written in Java that allows distributed processing of large dataset across cluster of computers using simple programming model. Hadoop creates cluster of machines and coordinates the work among them. It is designed to scale up from single servers to thousands of machines, each offering local computation and storage.

Fig. 2: Big Data Hadoop Architecture

Hadoop consists of two components [3]. The core of Apache Hadoop consists of a storage part, known as Hadoop Distributed file system (HDFS), and the processing part is called Map Reduce. [6]

B. HDFS (Hadoop Distributed File System) [2]

HDFS is a file system designed for storing very large files with streaming data access pattern, running clusters on commodity hardware. HDFS manages storage on the cluster by breaking incoming files into pieces called 'blocks' and storing each blocks redundantly across the pool of the server. HDFS stores three copies of each file by copying each piece to three different servers. Size of each block 64MB. HDFS architecture is broadly divided into following three nodes which are Name Node, Data Node, and HDFS Clients/Edge Node.

1) Name Node

It is centrally placed node, which contains information about Hadoop file system. The main task of name node is that it records all the metadata & attributes and specific locations of files & data blocks in the data nodes. Name node acts as the master node as it stores all the information about the system .and provides information which is newly added, modified and removed from data nodes.

2) Data Node

It works as slave node. Hadoop environment may contain more than one data nodes based on capacity and performance. A data node performs two main tasks storing a block in HDFS and acts as the platform for running jobs.

3) HDFS Clients/Edge node

HDFS Clients sometimes also known as Edge node. It acts as linker between name node and data nodes. Hadoop cluster there is only one client but there are also many depending upon performance need.

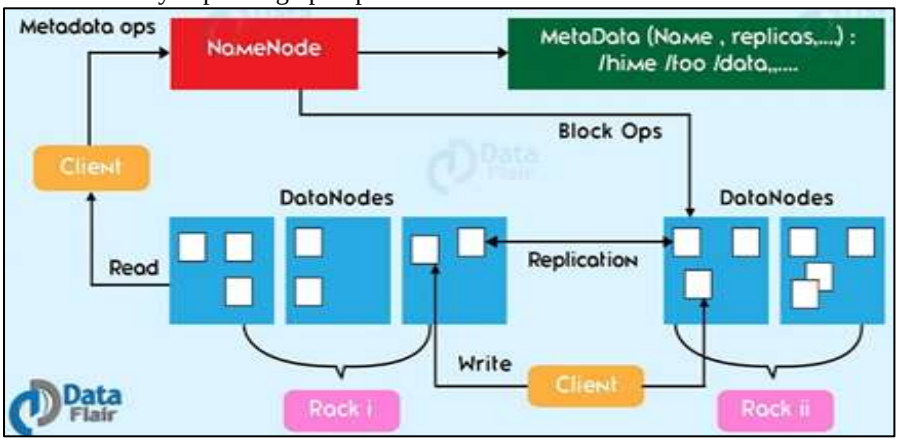

Fig. 3: HDFS Architecture

4) Map Reduce

MapReduce [1]: It is a programming model introduced by Google in 2004 for easily writing applications which processes large amount of data in parallel on large clusters of hardware in fault tolerant manner. This operates on huge data set, splits the problem and data sets and run it in parallel. Two functions in MapReduce are as following:

a) Map

The Map function always runs first typically used to filter, transform, or parse the data. The output from Map becomes the input to Reduce.

b) Reduce

The Reduce function is optional normally used to summarize data from the Map function.

C. *Master & Slave*

A Map Reduce framework is based on a master-slave architecture where one master node handles a number of slave nodes. Map Reduce works by first dividing the input data set into even-sized data blocks for equal load distribution. Each data block is then assigned to one slave node and is processed by a map task and result is generated. The slave node interrupts the master node when it is idle. The scheduler then assigns new tasks to the slave node. The scheduler takes data locality and resources into consideration when it disseminates data blocks. [4] The slave node processes the base problem and passes the result back to the master Node. The Map Reduce system arrange together all intermediate pairs based on the intermediate keys and refer them to reduce() function for producing the final output. Reduce function works as the master node collects the results from all the sub problems and combines them together to form the output.

Map(in_key,in_value)--- >list(out_key,intermediate_value)Reduce(out_key,list(intermediate_value))---
>list(out_value)

The parameters of map () and reduce () function is as follows:

map (k1,v1) ! list (k2,v2) and reduce (k2,list(v2)) ! list (v2)

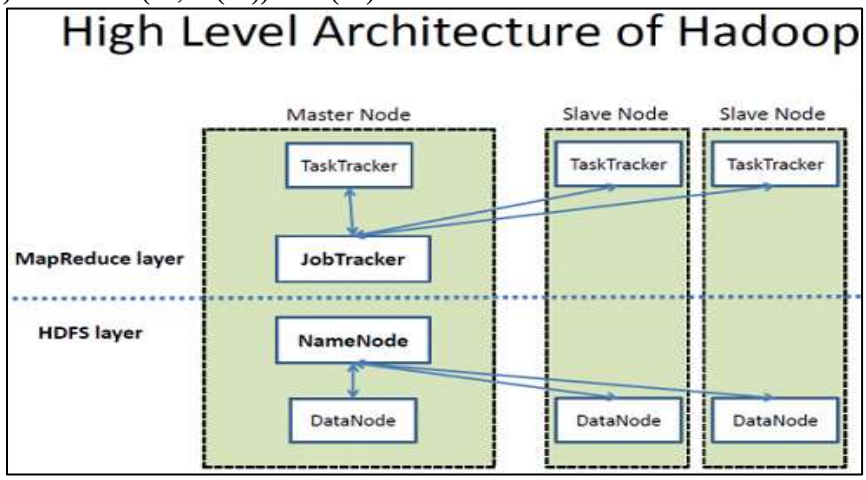

Fig. 4: Map Reduce: Master & Slave Architecture

D. Usag

Assuming HADOOP_HOME is the root of the installation and HADOOP_VERSION.
1) Hadoop Version Installed, Compile WordCount.java and create a jar:
$ mkdir wordcount_classes
$ javac -classpath ${HADOOP_HOME}/Hadoop
${HADOOP_VERSION}-core.jar -d wordcount_classes WordCount.java
$ jar -cvf /usr/joe/wordcount.jar -C
wordcount_classes/ .
2) Assuming That
• /usr/joe/wordcount/input - input directory in HDFS
• /usr/joe/wordcount/output - output directory in HDFS
3) Sample Text-Files as Input
$ bin/hadoop dfs -ls /usr/joe/wordcount/input/ /usr/joe/wordcount/input/file01
/usr/joe/wordcount/input/file02 $ bin/hadoop dfs -cat /usr/joe/wordcount/input/file01 Hello World Bye World (input file text) $ bin/hadoop dfs -cat /usr/joe/wordcount/input/file02 Hello Hadoop Goodbye Hadoop
4) Run The Application
$ bin/hadoop jar /usr/joe/wordcount.jar org.myorg.WordCount /usr/joe/wordcount/input /usr/joe/wordcount/output
5) Output
$ bin/hadoop dfs -cat /usr/joe/wordcount/output/part-00000
Bye 1
Goodbye 1
Hadoop 2
Hello 2
World 2

III. HADOOP ECO-SYSTEM

A. HBase

HBase is distributed column oriented database where as HDFS is file system. But it is built on top of HDFS system. HBase is a management system that is open-source, versioned, and distributed based on the Big Table of Google. It is Non-relational, distributed database system written in Java. It runs on the top of HDFS. It can serve as the input and output for the MapReduce. For example, read and write operations involve all rows but only a small subset of all columns. [8]

B. Avro

Avro is data serialization format which brings data interoperability among multiple components of apache Hadoop. Most of the components in Hadoop started supporting Avro data format. It works with basic premise of data produced by component should be readily consumed by other component Avro has following features Rich data types, Fast and compact serialization, Support many programming languages like java, Python.

C. Pig

Pig is platform for big data analysis and processing. Pig adds one more level abstraction in data processing and it makes writing and maintaining data processing jobs very easy. Pig. can process tera bytes of data with half dozen lines of code.

D. Hive

Hive is a data ware housing framework on top of Hadoop. Hive allows to write SQL like queries to process and analyse the big data stored in HDFS. It is Data warehousing application that provides the SQL interface and relational model. Hive infrastructure is built on the top of Hadoop that help in providing summarization, query and analysis. [7]

E. Sqoop

Sqoop is tool which can be used to transfer the data from relational database environments like oracle, my SQL and into Hadoop environment Sqoop is a command line interface platform that is used for transferring data between relational databases and Hadoop.

F. Zookeeper

Zookeeper is a distributed coordination and governing service for Hadoop cluster. It is a centralized service that provides distributed synchronization and providing group services and maintains the configuration information etc. In Hadoop, this will be useful to track if particular node is down and plan necessary communication protocol around node failure.

G. Mahout

Mahout is a library for machine-learning and data mining. It is divided into four main groups: collective filtering, categorization, clustering, and mining of parallel frequent patterns. The Mahout library belongs to subset that can be executed in a distributed mode and can be executed by MapReduce.

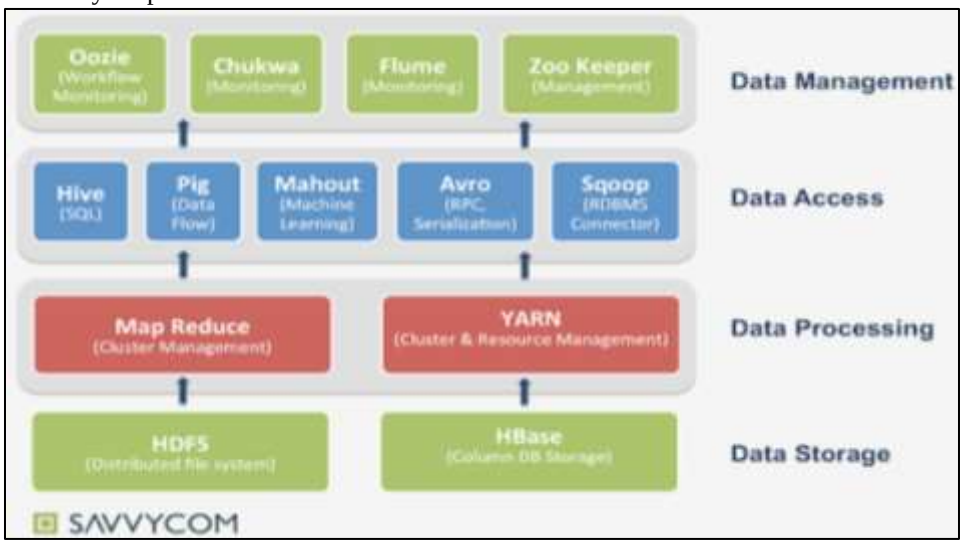

Fig. 5: Hadoop Ecosystem

IV. CONCLUSION

This paper describes about big data using Hadoop Framework and its components, HDFS and Map reduce. Hadoop plays an important role in Big data here I have entered an era of Big Data. The paper describes the concept of Big Data along with Operational vs. Analytical Systems of Big Data. The paper also focuses on Big Data processing problems. These technical challenges must be addressed for efficient and fast processing of Big Data. In this paper we have tried to cover all detail of Hadoop and Hadoop component and future scope.

REFERENCES

[1] Mrigank Mridul, Akashdeep Khajuria, Snehasish Dutta, Kumar N " Analysis of Bidgata using Apache Hadoop and Map Reduce" in International Journal of Advance Research in Computer Science and Software Engineering, Volume 4, Issue 5, May 2014.
[2] Varsha B.Bobade, International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056 Volume: 03 Issue: 01 | Jan-2016 www.irjet.net
[3] Apache Hadoop: http://Hadoop.apache.org.
[4] International Journal of Advanced Research in Computer Science and Software Engineering. www.ijarcsse.com Survey Paper on Big Data C. Lakshmi*, V. V. Nagendra Kumar MCA Department, RGMCET, Nandyal, Andhra Pradesh, India.
[5] Hadoop Distributed File System, http://hadoop.apache.org/hdfs
[6] HadoopTutorial: http://developer.yahoo.com/hadoop/tutorial/module1.html
[7] Apache Hive. Available at http://hive.apache.org
[8] Apache HBase. Available at http://hbase.apache.org