



## Hadoop: Components and Working

Nitika Arora

Govt.College for Women, Karnal  
Assistant Professor in Computer Science  
India

**Abstract:** Nowadays, Companies need to process Multi Petabyte of data efficiently. The Data may not have schema for the large system. It has become costly to get reliability in each Application for processing Petabyte of datasets. If there is a problem of Nodes fails every day, some of the causes of failure may be expected, rather than exceptional. The number of nodes in a cluster is not constant. So there is a need for one common infrastructure to have Efficient, reliable, Open Source Apache License. The Hadoop platform was designed to solve problems where you have a lot of data perhaps a mixture of complex and structured data and it doesn't fit perfectly into tables. It's for situations where you want to go for analysis that is deep and computationally extensive, like clustering and targeting. That's exactly what Google was doing when it was indexing the web and examining user behavior to improve performance algorithms. Hadoop has its origins in Apache Nutch, an open source web search engine which is a part of the Lucene project. Building a web search engine from scratch was an ambitious goal, for not only is the software required to index websites complex to write, but it is also a challenge to run without a dedicated operations team, since there are so many moving parts

**Keywords:** Hadoop, different domains, volume, variety, velocity, value, veracity

### I. INTRODUCTION

With the growth of technological development and services, the large amount of data is formed that can be structured and unstructured from the different sources in different domains. Massive data of such sort is very difficult to process that contains the information of the records of million people that includes everyday massive amount of data from social sites, cell phones GPS signals, videos etc. Big data is a largest buzz phrases in domain of IT, new technologies of personal communication driving the big data new trend and internet population grew day by day but it never reach by 100%. The need of big data generated from the large companies like facebook, yahoo, Google, YouTube etc for the purpose of analysis of enormous amount of data which is in unstructured form or even in structured form. Google contains the large amount of information. So; there is the need of Big Data Analytics[2] that is the processing of the complex and massive datasets This data is different from structured data (which is stored in relational database systems) in terms of five parameters –variety, volume, value, veracity and velocity (5V's). The five V's (volume, variety, velocity, value, veracity) are the challenges of big data management are:

unstructured such as text, audio, videos, log files and more. The varieties are endless, and the data enters the network without having been quantified or qualified in any way.

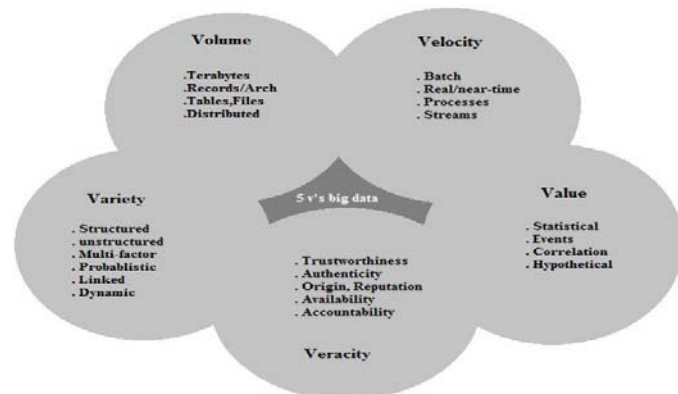


Fig. 1 Parameters of Big Data

- **Volume:** Data is ever-growing day by day of all types ever KB, MB, PB, YB, ZB, TB of information. The 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.
- **Variety:** Data sources (even in the same field or in distinct) are extremely heterogeneous. 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. The varieties are endless, and the data enters the network without having been quantified or qualified in any way.
- **Velocity:** The data comes at high speed. Sometimes 1 minute is too late so big data is time sensitive. Some organizations 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.
- **Value:** Which addresses the need for valuation of enterprise data? 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.
- **Veracity:** The increase in the range of values typical of a large data set. 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.

Huge volume of data (both structured and unstructured) is management by organization, administration and governance. Unstructured data is a data that is not present in a database. Unstructured data may be text, verbal data or in another form. Textual unstructured data is like power point presentation, email messages, word documents, and instant messages. Data in another format can be .jpg images, .png images, audio files (.mp3, .wav, .aiff) and video files that can be in flash format, .mkv format or .3gp format. According to the “IDC Enterprise Disk Storage Consumption Model” report[4] released in year 2009, in which the transactional data is proposed to raise at a composite yearly growth rate (CAGR) of 21.8%, it’s far outpaced by a 61.7% CAGR calculation for unstructured data.

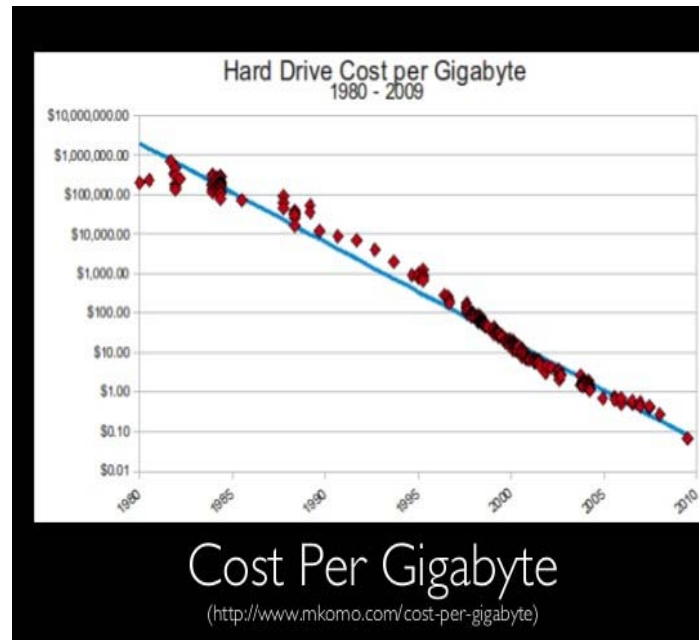


Fig 2: Cost per Gigabytes

From the figure 2 we conclude that cost of hard drive per gigabyte is increasing and usage of hard drive will prove to be cost ineffective. From last twenty years, the data is mounting day by day across the world in every domain. Some distinct facts about the data are, there are about 277,000 tweets per minute, 2 million queries approximately on Google every minute in all domains, 75 hours of new videos in different formats are uploaded to YouTube, More than 100 million emails are sent via Gmail, yahoo, rediff mail and many more, 350 GB of data is dealing out on facebook every day and more than 576 websites are created every minute. During the year 2012, 2.5 quintillion bytes of data were created every day. Big data and its depth analysis[3] is the core of modern science, research area and business areas. Huge amount of data is generated from the

distinct various sources either in structure or unstructured form. Such form of data stored in databases and then it become very complex to extract, transform and make in use. IBM indicates that 2.5 Exabyte data is created everyday which is very difficult to analyze in various aspects. The estimation about the generated data is that till year 2003 it was represented about 5 Exabyte, then until year 2012 is 2.7 Zettabyte and till 2015 it is expected to boost up to 3 times. From figure 3 we conclude that unstructured data[9] is exploding at large rate so there is need to manage structured/Unstructured data by Hadoop[9].

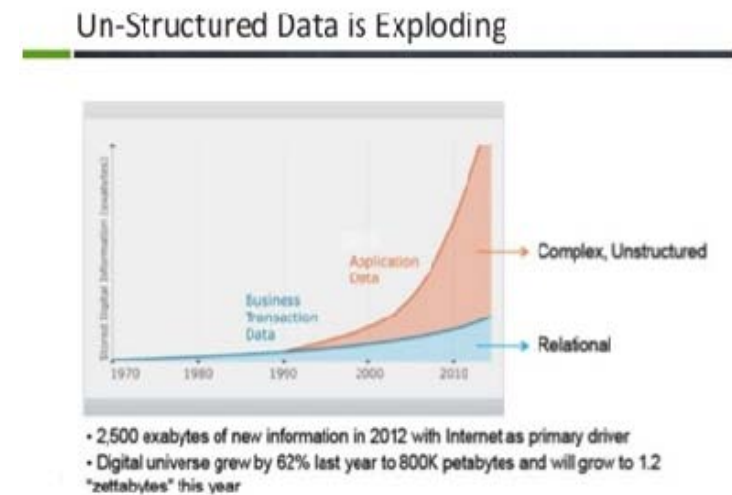


Fig 3: Explosion of Unstructured data

## II.APACHE HADOOP

Apache Hadoop[12] is an open source framework [1] for developing distributed applications that can process very large amounts of data. It is a platform that provides both distributed storage and computational capabilities. Hadoop has two main layers:

*A.Computation layer: The computation tier uses a framework called **Map Reduce**.*

*B.Distributed storage layer: A distributed file system called **HDFS** provides storage*

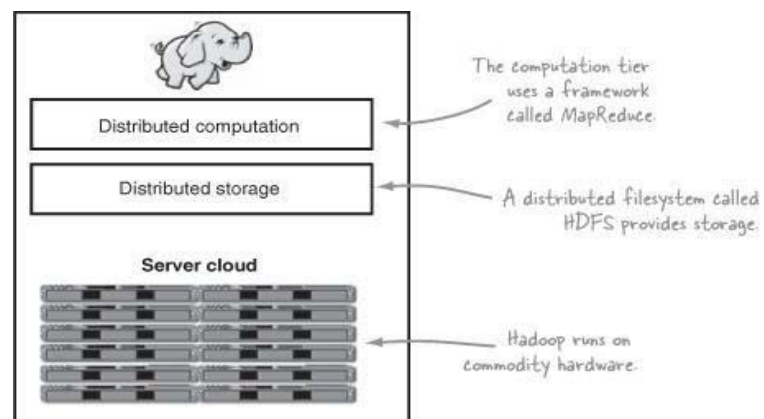


Fig 4: Layers of Hadoop

In a Hadoop cluster, data is distributed to all the nodes of the

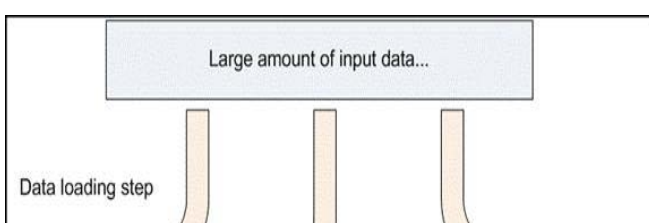
cluster as it is being loaded in. The Hadoop Distributed File System (HDFS) will split large data files into chunks which are managed by different nodes in the cluster. In addition to this each chunk is replicated across several machines, so that a single machine failure does not result in any data being unavailable. An active monitoring system then re-replicates the data in response to system failures which can result in partial storage. Even though the file chunks are replicated and distributed across several machines, they form a single namespace, so their contents are universally accessible. Data is conceptually **record-oriented** in the Hadoop programming framework. Individual input files are broken into lines or into other formats specific to the application logic. Each process running on a node in the cluster then processes a subset of these records.

The Hadoop framework then schedules these processes in proximity to the location of data/records using knowledge from the distributed file system. Since files are spread across the distributed file system as chunks, each compute process running on a node operates on a subset of the data. Which data operated on by a node is chosen based on its locality to the node: most data is read from the local disk straight into the CPU, alleviating strain on network bandwidth and preventing unnecessary network transfers. This strategy of **moving computation to the data**, instead of moving the data to the computation allows Hadoop to achieve high data locality which in turn results in high performance. Hadoop is written in the Java programming language and is an Apache top-level project being built and used by a global community of contributors. Hadoop and its related projects (Hive, HBase, Zookeeper, and so on) have many contributors from across the ecosystem. Though Java code is most common, any programming language can be used with "streaming" to implement the "map" and "reduce" parts of the system. It provides a distributed file system that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. Both map/reduce and the distributed file systems are designed so that node failures are automatically handled by the framework. It enables applications to work with thousands of computation-independent computers and pet bytes of data. The entire Apache Hadoop "platform" is now commonly considered to consist of the Hadoop kernel, Map Reduce and Hadoop Distributed File System (HDFS), as well as a number of related projects – including Apache Hive, Apache HBase, and others.

Fig 5: Data loading process

### III. HOW IT WORKS

Hadoop limits the amount of communication which can be performed by the processes, as each individual record is processed by a task in isolation from one another. While this sounds like a major limitation at first, it makes the whole framework much more reliable. Hadoop will not run just any program and distribute it across a cluster. Programs must be written to conform to a particular programming model, named "MapReduce." In MapReduce[8], records are processed in isolation by tasks called Mappers. The output from the Mappers is then brought together into a second set of tasks called Reducers, where results from different Mappers can be merged together. Separate nodes in a Hadoop cluster still communicate with one another. However, in contrast to more conventional distributed systems where application developers explicitly marshal byte streams from node to node over sockets or through MPI buffers, communication in Hadoop is performed implicitly. Pieces of data can be tagged with key names which inform Hadoop how to send related bits of information to a common destination node. Hadoop internally manages all of the data transfer and cluster topology issues. By restricting the communication between nodes[7], Hadoop makes the distributed system much more reliable. Individual node failures can be worked around by restarting tasks on other machines. Since user-level tasks do not communicate explicitly with one another, no messages need to be exchanged by user programs, nor do nodes need to roll back to pre-arranged checkpoints to partially restart the computation. The other workers continue to operate as though nothing went wrong, leaving the challenging aspects of partially restarting the program to the underlying Hadoop layer.





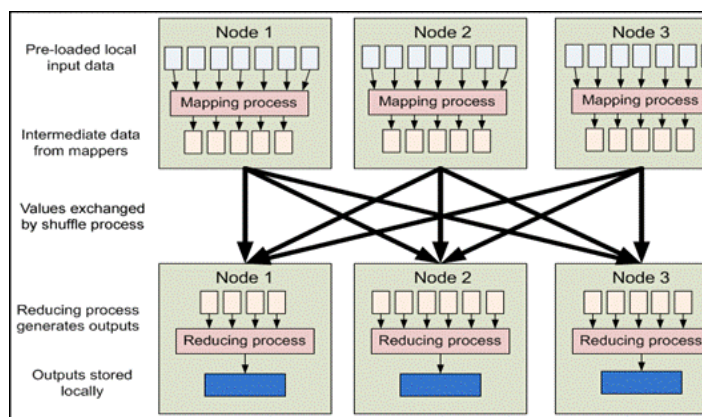


Fig 6: MapReduce Process

#### IV.HADOOP COMPONENTS

HDFS[10], the storage layer of Hadoop, is a distributed, scalable, Java-based file system adept at storing large volumes of unstructured data. Map Reduce[5] is a software framework that serves as the compute layer of Hadoop. MapReduce jobs are divided into two (obviously named) parts. The “Map” function divides a query into multiple parts and processes data at the node level. The “Reduce” function aggregates the results of the “Map” function to determine the “answer” to the query. Hive is a Hadoop-based data warehousing-like framework originally developed by Facebook. It allows users to write queries in a SQL-like language called HiveQL, which are then converted to Map Reduce. This allows SQL programmers with no Map Reduce experience to use the warehouse and makes it easier to integrate with business intelligence and visualization tools such as Micro strategy, Tableau, Revolutions Analytics, etc. Pig Latin is a Hadoop-based language developed by Yahoo. It is relatively easy to learn and is adept at very deep, very long data pipelines (a limitation of SQL.)

HBase is a non-relational database that allows for low-latency, quick lookups in Hadoop. It adds transactional capabilities to Hadoop, allowing users to conduct updates, inserts and deletes. EBay and Facebook use HBase heavily. Flume is a framework for populating Hadoop with data. Oozie is a workflow processing system that lets users define a series of jobs written in multiple languages – such as Map Reduce, Pig and Hive -- then intelligently link them to one another. Oozie allows users to specify, for example, that a particular query is only to be initiated after specified previous jobs on which it relies for data are completed. Flume is a framework for populating Hadoop with data. Ambari is a web-based set of tools for deploying, administering and monitoring Apache Hadoop clusters. Its development is being led by engineers from Hortonworks, which include Ambari in its Horton works Data Platform. Avro is a data serialization system that allows for encoding the schema of Hadoop files. It is adept at parsing data and performing remote procedure calls. Mahout is a data mining library. It takes the most popular data mining algorithms for performing clustering, regression testing and statistical modeling and implements them using the Map Reduce model. Sqoop is a connectivity tool for moving data from non-Hadoop data stores – such as relational databases and data warehouses – into Hadoop. HCatalog is a centralized metadata management and sharing service for Apache Hadoop. Big Top is an effort to create a more

formal process or framework for packaging and interoperability testing of Hadoop’s sub -projects and related components with the goal improving the Hadoop platform as a whole.








Hadoop Components and Sub projects	Comparing with	Reason for the specific animal	Image
1.Hadoop distributed file system (HDFS)	Elephant	Memory of an elephant is compared with huge data storage of HDFS	
2.MapReduce	Mammoth	Mammoth means enormous, huge, massive and immense. A Mammoth's task is compared with a programming model for performing the tasks with huge volumes of data	
3.Hive	Honey Bee	Storage area of the honey in wax honeycombs inside the beehive is compared with data warehouse for storing the data in the format of table.	
4.Hbase	Horse	Running Speed of the horse indicates the real time read/write access from HBase	
5.Pig	Pig	Pigs are omnivores animals which means they can consume both plants and animals. This PIG consumes any type of data whether structured or unstructured or any other machine data and helps processing the same.	
6.Hue	Elephant foot prints	Elephant foot print is compared with “TREAD ON” Hadoop and explore it.	
7.Beeswax	Beeswax	Data storage happens in a structured way as honey stored in Beeswax	

Table 1: Comparison of Hadoop Components

#### V.WORKING OF HADOOP ARCHITECTURE

Hadoop is designed to run on a large number of machines that don’t share any memory or disks. That means you can buy a whole bunch of commodity servers, slap them in a rack, and run the Hadoop software on each one. When you want to load all of your organization’s data into Hadoop, what the software does is bust that data into pieces that it then spreads across your different servers. There’s no one place where you go to talk to all of your data; Hadoop keeps track of where the data resides. And because there are multiple copy stores[10], data stored on a server that goes offline or dies can be automatically replicated from a known good copy.

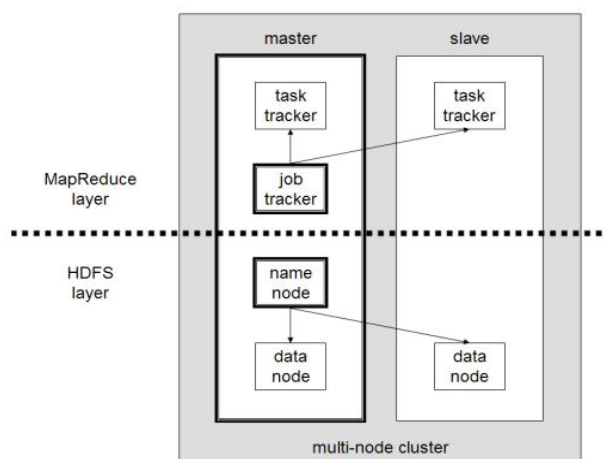


Fig 7: Multimode cluster

In a centralized database system[4], you’ve got one big disk connected to four or eight or 16 big processors. But that is as much horsepower as you can bring to bear. In a Hadoop cluster, every one of those servers has two or four

or eight CPUs. You can run your indexing job by sending your code to each of the dozens of servers in your cluster, and each server operates on its own little piece of the data. Results are then delivered back to you in a unified whole. That's Map Reduce[6] you map the operation out to all of those servers and then you reduce the results back into a single result set. Architecturally, the reason you're able to deal with lots of data is because Hadoop spreads it and the reason you're able to ask complicated computational questions is because you've got all of these processors, working in parallel, harnessed together. Hadoop implements a computational paradigm named Map/Reduce, where the application is divided into many small fragments of work, each of which may be executed or re-executed on any node in the cluster. In addition, it provides a distributed file system (HDFS) that stores data on the compute nodes, providing very high aggregate bandwidth across the that node failures are automatically handled by the framework. Hadoop Common is a set of utilities that support the other Hadoop subprojects. Hadoop Common includes File System, RPC, and serialization libraries.

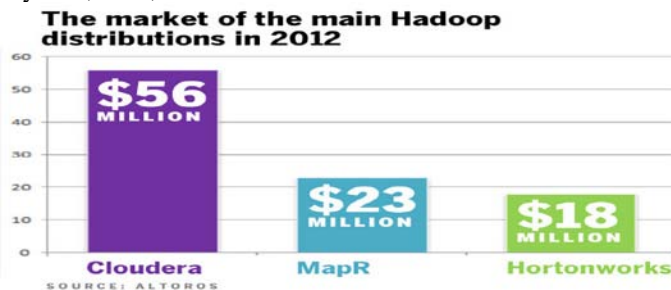


Fig 8. The market of the main Hadoop distributions in 2012, \$ million

Three of the top Hadoop distributions are provided by Cloudera[12], MapR and Hortonworks. The chart below illustrates the results of the market research "Big Data Vendor Revenue and Market Forecast 2012–2017." It compares the revenue of these major Hadoop vendors in 2012. While Cloudera and Hortonworks claim they are 100% open source, MapR adds some proprietary components to the M3, M5, and M7 Hadoop distributions to improve the framework's stability and performance.

Along with Cloudera, MapR and Hortonworks, Hadoop distributions are available from IBM, Intel, Pivotal Software, and others. These distributions may even be shipped as a part of a software suite (e.g., IBM's distribution), or designed to solve specific tasks (e.g., Intel's distribution optimized for the Xeon microprocessor).

## VI. CONCLUSION

Hadoop is establishing a foothold in enterprises. It is maturing and gradually becoming a key piece of the enterprise data infrastructure at many organizations. Until recently, mainstream companies experimented with Hadoop to learn its functionality and capabilities and better understand the role it could play in their existing analytical ecosystems. Today, many of these organizations are moving Hadoop in production to support data processing and analytical requirements or to offload workloads from existing data warehouses to save money. IT and data management professionals who have implemented Hadoop give it high ratings. The software is evolving fast, thanks to the hard work of many vendors in the space—as well as practitioners who contribute code to the Apache Foundation. Although it remains to be seen whether Hadoop can fulfill its promise as the enterprise operating system for data analytics, it's moving in the right direction.

## VII. REFERENCES

- [1] Welcome to Apache™ Hadoop®! (n.d.). Retrieved July 2, 2015.
- [2] Rajasekar1, D. (2015). A Survey on Big Data Concepts and Tools. International Journal of Emerging Technology and Advanced Engineering, 5(2), 80-81. Retrieved September 1, 2015, from [www.ijetae.com](http://www.ijetae.com).
- [3] Big Data in Hadoop – How? What is it? (2013, February 15). Business Analytics and Intelligence.
- [4] Yuri Demchenko "The Big Data Architecture Framework (BDAF)" Outcome of the Brainstorming Session at the University of Amsterdam 17 July 2013.
- [5] Sagioglu, Sinanc, "Big Data: A Review", 978-1-4673-6404-1/13 IEEE.
- [6] Sabia, Arora, "Technologies to Handle Big Data: A Survey".
- [7] Shilpa, Manjit Kaur, "Big Data and Methodology – A review", Volume 3, Issue 10, October 2013.
- [8] Aditya B. Patel, Manashvi Birla, Ushma Nair, "Addressing Big Data Problem Using Hadoop and Map Reduce", Aug, 2012.
- [9] <http://www.slideshare.net/shilpasoi/v3-i10-0415>.
- [10] White paper – BigData-as-a service, A Market & Technology Perspective-EMC Solution group
- [11] [http://en.wikipedia.org/wiki/Big\\_data](http://en.wikipedia.org/wiki/Big_data).
- [12] <http://hadoop.apache.org/>