# Enhancing the Traditional File System to HDFS: A Big Data Solution

Himani Saraswat
M.Tech
Noida International University
Greater Noida. UP

Neeta Sharma, PhD
Assistant Professor
Noida International University
Greater Noida. UP

Abhishek Rai
Administrator-BigData
Wipro.Technologies
Gurgaon.

## ABSTRACT
We are in the twenty-first centuries also known as the digital era, where each and every thing generates a data whether it's a mobile phone, signals, day to day purchasing and many more. This rapidly increases in amount of data; Big data has become a current and future frontier for researchers. In big data analysis, the computation is done on massive heap of data sets to extract intelligent, knowledgeable and meaningful data and at the same time the storage is also readily available to support the concurrent computation process. The Hadoop is designed to meet these complex but meaningful work. The HDFS (Hadoop Distributed File System) is highly fault-safe and is designed to be deployed on low cost hardware. This paper gives out the benefits of HDFS given to the large data set; HDFS architecture and its role in Hadoop.

## General Terms
Distributed File System, Architecture, analysis

## Keywords
BigData, HDFS, clusters, Nodes, Hadoop, Architecture.

## 1. INTRODUCTION
Big data is a catch line in nowadays digital world. Analysis is being performed on cyclopean amount of data. This data can be structured, unstructured or semi-structured, it is so huge that it is difficult to process them from the traditional relational data base methods. In the present endeavor scenario, this data is so huge that it sometimes exceeds the processing capacities of some database methods. The Apache Hadoop is a useful framework to overcome the breaking issues of data sets. Its main focus is to execute this large data set into cluster form and execute within the time constrain. The domains which have large dataset to work on are: - the metrological researches, Internet searches, social networking, medical researches etc. The main challenge of these are to store, share search and capture the data .and then do an analysis over it. The main benefit to have Hadoop is that it overcome these challenges. The major approach of this paper is to describe the distributed file system and its architecture along with the little insight of ecosystem of Hadoop.

## 2. HADOOP
The Apache technologies have introduced its major project Hadoop in January 2008. After Jan 2008, Hadoop is being used in many streams like financial services, government services, telecom, advertising and many search engines like Google. It is an open source software framework for the reliable, scalable, distributed computing. It is designed to connect from single server to many thousand machines in which, each offers the local computation and storage. There are two most important components of Hadoop: HDFS and MapReduce.

### 2.1 MapReduce
IT is a processing unit of Hadoop. The basic idea behind is to have two different modules which has its own results; Map & Reduce. They run in a streamline or serializing manner. They do N' number of things like sorting, searching, shuffling, partitioning of data, and many more of its kind. After mapping it reduces the process data into desired result.

### 2.2 HDFS (Hadoop Distributed File System):
It is like a bluestockings of Hadoop. It accumulates the data in multiple nodes in a distributed manner. The main idea is to have a distributed file system where data get stored in DataNode. HDFS has a golden rule "Read many times write once. A file can be read many times on a HDFS file system but it can only be written once.

### 2.3 Hadoop Ecosystem
Ecosystem of Hadoop has a top down approach. In earlier release of Hadoop 1.x there are only two units which actually works i.e. MapReduce and HDFS. MapReduce process the data and result automatically get stored in the HDFS. But in later version of Hadoop 2.x(see Fig 1) the new component is added in its ecosystem i.e. YARN (Yet Another Resource Negotiator). The YARN is same as the MapReduce but the way of processing is different. YARN processes the data in container. The containers are the logic units which consist of resource and task itself (here resource is DataNode). With the presence of YARN the deadlock situation which usually use to appear in 1.x are minimized in 2.x.

In Hadoop eco system every data breakdown into the data chunks also known as the data blocks.

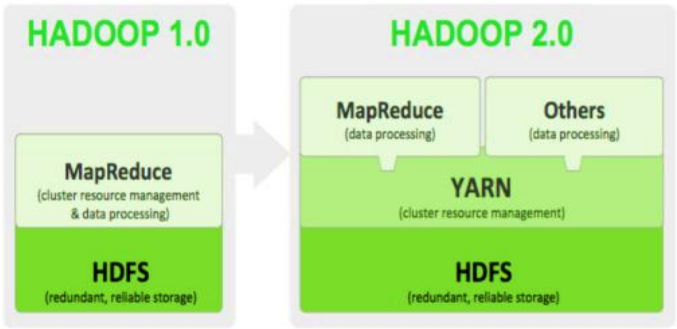


**Figure 1: Hadoop 1.0 and Hadoop 2.0 Ecosystem [7]**

# 3. HDFS

## 3.1 Why the concept of HDGFS came into existence?

We are aware that; to store a data for analysis at a single place is quite impossible also costly. Apache software 0 foundation comes with the concept to analyze the data without capping that data at single place. They term it as distributed file system. They integrate this in Hadoop framework to make it more useful for the BA's and the data analysts. In Hadoop framework, the distributed file system termed as the Hadoop Distributed file system (HDFS). HDFS is like the brain of Hadoop which stores the data in a distributed manner i.e. the migration to actual data physically is not required. To configure this there is a need of multiple machine cluster, which is connected over LAN. These machines can be a commodity hardware also for creating a cluster. What HDFS do it get the instances or the collection of the meta data of, actual data. If the User or Master gives an instruction from the Master Node, HDFS knows where the data is residing in real, and it starts processing on that machine where the real data resides. In persistence, the master node will be free from the burden and can do rest of the things of cluster. After processing, the final output is stored over the HDFS. This is how the HDFS works in Hadoop. In addition, the cost of migration of data is saved for the industries who generates Petabytes of data quarterly basis.

## 3.2 HDFS Architecture

HDFS or Hadoop Distribute File System is the bluestockings of Hadoop. As a human brain accumulate data in a very distinctive manner. HDFS do the same to Hadoop, it stock-up data in a distributive manner over the multiple nodes. HDFS provides high throughput access to application data and is Suitable for applications that have large data sets. The priority thing in HDFS is; (see Fig 2) it stores the data in distributed way to multiple nodes over the cluster (set of multiple machines connected over a LAN or MAN connection. There are components in Hadoop; NameNode, DataNode. NameNode stores the metadata (like name, path and so on) and helps the client to commute with the Hadoop infrastructure. DataNode stores the actual data and with the help of processing unit it actually processes the data. All the machines which are in cluster interact with each other based on SSH and TCP/IP protocol. It basically works on master-slave architectural concepts.

The client i.e. the human who commute with the NameNode, (the master NameNode), who stores the Metadata, (replicas, and the path of the file, the client query for the file through the NameNode i.e. read operation). NameNode initiate a block operation to generate the query to find the exact location of the file over the HDFS. As we know on HDFS everything lies there as a block of data. So NameNode find that data block and return it to Client. The same happens for the write operation; just the difference is that, instead of querying we write the data on the HDFS only once, which is done by the NameNode.

It supports the hierarchical file organization system. A user can create directories and store files inside them. The NameNode maintains the file system namespace. Any changes to the file- system namespace or its properties is recorded by the NameNode. An application can specify the number of replicas of a file that should be maintained by HDFS. The number of copies of a file is called the replication factor of that file. This information is stored by the NameNode.

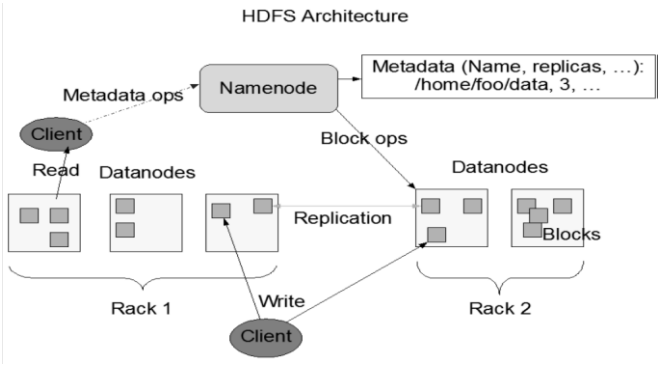


**Figure 2: HDFS Architecture [8]**

## 3.3 Command – Line Interface

HDFS has many interfaces of interacting but the command line is the way, which is more convenient and most widespread way of interacting with the kernel of the Linux system by researchers and developers.

**FS Shell**: The File System Shell is the interactive way to communicate with the file system in Hadoop. It accesses the HDFS with some Linux based 'root' commands which help to do particular task over the HDFS. Few of the commands are as follows:

(1) hadoop fs –ls /: to access the HDFS and list all the files/ directory present over the HDFS.

(2) hadoop fs –ls /user: to get into the user directory over the HDFS.

(3) hadoop fs –cat /user/myfile.txt: to read the content of file (myfile.txt) present in the user directory over HDFS.

(4) Hadoop fs –put /filesource /final destination over the HDFS: this command is used to put any local file to HDFS

# 4. CONCLUSION

The paper hence explains the benefits and role of HDFS. As the HDFS is a new world distributed file system. It not only gives a beneficial stake to Hadoop but also a low cost, fault-tolerant system. HDFS is a boon to BI/BA who wanted to not only secure data as well as wanted to have extensive data access.

# 5. ACKNOWLEDGEMENT

Our sincere thanks to Dr. Neeta Sharma and Mr. Abhishek Rai for this research work and the development of this project. Mrs. Neeta Sharma always supported me with her guidance on project by providing research papers and other materials to enhance my knowledge. Mr. Abhishek Rai being a working professional he helped me in my project development and teach me the Linux as well as the Hadoop so that I can easily work on Hadoop. There efforts made this project and this research valuable.

## 6. REFERENCES

[1] Tom White, "Hadoop The Definitive Guide", 4th Edition 2015.3 by O'reilly.

[2] Alexdro Labrindis, H.V.Jagdish, Challenges and Opportunities with Big Data", Proceedings of the VLDB Endowment,Vol.05,No.12,States.http://cra.org/ccc/docs/init/bigdatawhitepaper.pdf, Mar 2012.

[3] Dr. PelleJakovitis, Reducing Scientific Computing, Master Thesis, University of Tartu,2010.

[4] Girish Prasad Patro, "A Novel Approach for Data Encryption in Hadoop", Department of Computer Science and Engineering National Institute of Technology Rourkela, www.nitrkl.ac

[5] Puneet Singh Duggal,Sanchita Paul," Big Data Analysis: Challenges and Solutions. International Conference on Cloud, Big Data and Trust 2013, Nov 13-15, RGPV, At RGPV,Bhopal, India

[6] Gloria-Phillips-Wren,"Business Analytics in the Context of Big Data: A Roadmap for Research", Loyola University-Maryland,

[7] Hadoop 2 vs. Hadoop 1 [Image] http://www.tomsitpro.com/articles/hadoop-2-vs-1,2-718.html

[8] HDFS Architecture Guide [Image] https://hadoop.apache.org/docs/r1.2.1/hdfs_design.html