Hadoop, Map Reduce and HDFS-A Review

Aditya Gupta, Gaurav Aggarwal, Akash Kumar

Abstract—Right now we are living in data world, so everywhere we are seeing is only data so the important thing is how to store the data and how to process the data. In this paper we will focus on two relatively new developments: Hadoop (Map Reduce and HDFS). The common goal is to understand and to explore the concept of what is called “Big Data”, i.e., data which is beyond to the storage capacity and beyond to the processing power (hundreds or thousands of terabytes to petabytes or more in size).

Index Terms – Big Data, Hadoop, Map Reduce, HDFS, Name Node, Secondary Name Node, Job Tracker, Data NodeTask Tracker

1 Introduction

Hadoop is an open source technology or framework which is written in java by Apache software foundation. This framework is used to write software applications which requires to process vast amount of data (typically terabytes of data). This framework functions in parallel on large clusters and each cluster may have thousands of nodes. Hadoop processes the data very reliably and in a fault tolerant manner using simple programming models. Hadoop is designed to scale up from a single server to thousands of the machines, each offering us local computation and storage [1].

There are two core concepts in Hadoop i.e., HDFS (Hadoop distributed file system) and Map-Reduce. Hadoop distributed is provided as file system which is capable of storing huge amount of data. The Map-Reduce technology was introduced for processing of such a huge data. So Hadoop is a combination of HDFS and Map-Reduce. HDFS can also be defined as a specially designed file system for storing huge data sets with cluster of commodity hardware and with streaming access patterns.

As Java uses the slogan “Write once run anywhere”, which means program written in Java can be executed on any platform provided there is a Java environment on that platform. HDFS also uses a slogan “Streaming access patterns” which means write once, read any number of times and don’t try to change the contents of file, once you are keeping data in HDFS [2].

Figure 1.1 Hadoop Logo

Hadoop operates on massive datasets by horizontally scaling (aka scaling out), the processing across very large numbers of servers through an approach called Map Reduce. Vertical scaling (aka scaling up), i.e., running on the most powerful single server available, is both very expensive and limiting. There is no single server available today or in the foreseeable future that has the necessary power to process so much data in a timely manner[4].

Map-Reduce is a frame work for processing such a vast amount of data by assigning data to number of different processors which works in parallel and gives the result in a timely manner.

2 History of Hadoop

We are all aware of google ,a great web search engine in web world .as these google people have done a great work in 1990s ,they had to come up with more data that time they started thinking that how to store huge data and how to process it ,so to get proper solution for that it has been taken 13 years for them and in the tear 2003 they had given one conclusion to store the data as GFS called as Google file system, a technique to store the data and in the year 2004 they came up with one more technique called as Map Reduce[5] .As GFS is a technique to store so much of huge data ,Map Reduce is a technique to process that much of huge data but the problem with Google exactly! is they had just given these techniques as description in some white paper but never implemented that. Later yahoo, a largest search engine in web world introduced a technique called HDFS (Hadoop distributed file system) by using the concept of Google file system in the year 2006 and Map reduce in the year 2007.

Before understanding Hadoop and its core concepts (HDFS and Map-Reduce), we need to have some knowledge about Big Data.

3 Big Data

Right now we are living in data world, so everywhere we are seeing is only data so the important thing is how to store the data and how to process the data. Exactly to say what is

Big Data?

We can define big data as data which is beyond to the storage capacity and which is beyond to the storage power, that data we are calling here is the big data. In other words Big Data is nothing but an assortment of such a complex and huge data that it becomes tedious to capture, store process, retrieve and analyse it with the help of traditional data base management techniques.

How we are getting such data?

There are different data generators like sensors,CCTV, online shopping, airlines, hospitality data, social networks like Facebook, twitter, linked in, and e bloggers and so on

There are many real examples of such a huge data.

Social media such as Facebook generates more than 500 TB of data on a single day, New York Stock Exchange generates more than 1 TB data per day. These are some of examples of Big Data.

If we are living in 100 percent of data world, 90 percent of data has been generated for the last two years and the remaining 10 percent of the data has been generated for the long back when these systems were getting introduced[6].

In fact, big data is about more than just the "bigness" of the data. Its key characteristics, coined by industry analysts are the "Three V's," which include volume (size) as well as velocity (speed) and variety (type). As far as we are getting so much of big data, we must be in a position to process that much of huge data in less time. With the time as data has increased but processing speed has not been increased to synchronise with such a data.

So our processing power must be equalise to our big data, in that sense Hadoop has been introduced as a best solution to big data. Hadoop knows very well how to store and process huge data in less time.

4 HDFS (Hadoop Distributed File System)

HDFS is an important component of Hadoop.HDFS is a specially designed file system for storing huge datasets with a cluster of commodity hardware and with streaming access patterns .Here commodity hardware refers to the cheap hardware. HDFS Uses a block size of 64 MB that can be extended up to 128 MB depending upon the need and type of applications. Normally file systems uses a block size of 4 KB which results in a loss of memory, HDFS by default uses 64 MB .Another reason for using 64 MB block is that meta data would be increased if 4 KB block is used .For Example if we want to store 200 MB of data, whole data will be splitted into 4 files, three files of 64 MB and a single file of 8 MB.

HDFS uses five type of services-
Name Node
Secondary Name Node
Job Tracker
Data Node
Task Tracker

Name Node, Secondary Name Node, Job Tracker are also called as Master Services or Master Daemons or Master Nodes and Data Node , Task Tracker are called as Slave Services or Slave Nodes or Slave Daemons[7].

Every Master Service can talk to each other, similarly every Slave Service can talk to each other.

Name Node talks to Data Node and Job Tracker talks to Task Tracker no more combinations of talking between these possible.

Data Node is a commodity hardware and it is a cheap hardware, we need not to implement Data Node as hardware of high quality as HDFS by default makes 3 replicas of each file and there is a no need to worry about file loss. Name Node is a highly reliable hardware as it acts as master and handles all the data nodes.

When a client needs to store the data in HDFS, it approaches Name Node and asks for the space. Name Node also maintains a Meta data which contains all the information about data, space allotted to client for storage, which replica is stored in which data node, file size and so on. This Meta data a wide role to play in HDFS. Name Node then assigns data nodes for storage and maintains by default 3 replicas and the complete information is stored in Meta data file. Each data node gives block report and heartbeat to name node to make sure that data nodes are alive and working properly. If data node gives no block report toname node it is considered dead and the data is maintained at other data node and related information is stored in Meta data. If Name Node fails the whole system would be damaged that is why highly reliable hardware is used for name node and it is called as single point of failure[8].

4.1 Features of HDFS

1. It is suitable for the distributed storage and processing.
2. To interact with HDFS, there is a command line.
3. The built-in servers of namenode and datanode help users to easily check the status of cluster.
4. Streaming access to file system data.
5. HDFS provides file permissions and authentication.

For accessing of data stored in HDFS, Map-Reduce comes in picture which is discussed after HDFS.

- *Aditya Gupta is currently working in Shivalik College of Engineering Dehradun, India. E-mail: Aditya.gupta@sce.org.in*
- *Gaurav Aggarwal is currently working in Shivalik College of Engineering Dehradun, India. E-mail: gaurav.aggarwalcoer@gmail.com*
- *AkashKumar is currently pursuing B.Tecch in Shivalik College of Engineering Dehradun, India. E-mail: akashkumar.ak775@gmail.com*

5 MAP REDUCE

Map reduce is a technique for processing the huge data stored in Hadoop distributed file system. The Map Reduce algorithm contains two important tasks, namely Map and Reduce. The component Map takes a data set and converts it into another data set where individual elements are splitted into key, value pairs then the reducer comes in picture whose task is to take the output from maps as input and combine those inputs to generate final output. The number of maps will be equal to the number of input splits[9].

There are basically four formats of a file:

1 TextInput Format
2 KeyValueTextInput Format
3 SequencefileInput Format
4 SequencefileAsTextInputFormat .

5 TextInput Format is the default format and the other three are explicitly specified in driver code for record reader understanding. If file format is TextInput Format then the record reader reads one line at a time from its corresponding input split and it is converted into Block offset, entire line pair as key, value pair. If file format is KeyValueTextInput Format then it splits that key as per the basis of tab character.

6 ANALYSIS ON RELATED WORK

It is also important to secure the data stored in Hadoop Distributed File system as the data stored in that architecture may be sensitive and may create security issues if not properly secured. The different pillars of security are authentication, authorisation and audit. is important which determines control access to the cluster. Authorisation restricts access to explicit data and audit maintains files which determines who did what?

But it is not an easy task to maintain proper security as stream of data is increasing day by day.

Another issue that may be addressed is to reduce the cost of saving huge amount of data and to improve the scality so that most of ted ta cab be stored in single cluster that may reduce the cost of hardware implemented.

7 CONCLUSIONS

Now we are in an environment of big data.In this paper we reviewed the concepts of big data, its characteristics that is variety, velocity and volume also called as 3 v's of big data. Then we learnt about how to handle that is how to store such a huge data and how to process it in an efficient manner. The main problem was to process the data in a timely manner with main focus to reduce the total time for processing .Hadoop which is an open source framework, solves the problem in an effective manner.

REFERENCES

[1] International Journal of Scientific & Engineering Research, Volume 5, Issue 6, June-2014 138 ISSN 2229-5518 IJSER © 2014 http://www.ijser.org

[2] Konstantin Shvachko, HairongKuang, Sanjay Radia, Robert Chansler, "The Hadoop Distributed File System", Shv, Hairong, SRadia, Chansler@YahooInc.com, IEEE 2010

[3] "Big Data - Solutions for RDBMS Problems – A Sur vey", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2

[4] Contributing Authors, "Big Data Spectrum", Infosys Limited Bangalore India, 2012

[5] Papineni Rajesh, Y. MadhaviLatha, "HADOOP the Ultimate Solution for BIG DATA Problems", IJCTT, Vol-4 Issue-4,April 2013

[6] AzzaAbouzeid, KamilBajdaPawlikowski, Daniel Abadi, AviSilberschatz, Alexander Rasin (August 2009), "HadoopDB: An Architectural Hybrid of MapReduce and DBMS Technologies for Analytical Workloads" VLDB '09, Lyon, France.

[7] Jeffrey Dean, Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", Google,Inc. , OSDI 2004

[8] White Paper Big Data Analytics, "Extract,Transform and Load Big Data with Apache Hadoop", Intel, 2013

[9] Umesh V. Nikam, Anup W. Burange, Abhishek A. Gulhane, "Big Data and HADOOP: A Big GameChanger", International Journal of Advance Research in Computer Science and Management Studies, Volume 1, Issue 7, ISSN: 2321-7782, DEC 2013