Hadoop Architecture and its Functionality

Dr. B V Ramana Murthy,
Department of CSE
Jyothishmathi College of Engg
and Technology, Shamirpet, India

Mr. V Padmakar
Department of CSE,
Guru Nanak Institutions
Technical Campus, Hyderabad

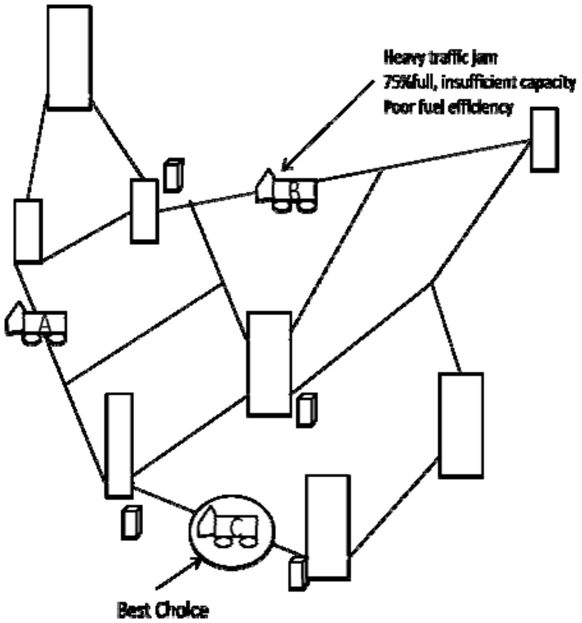
Mr. M Abhishek Reddy
Department of CSE,
Jyothishmathi College of Engg
Technology, Shamirpet, India.

Abstract:
Hadoop is nothing but a “framework of tools” and it is a java based programming framework (In simple terms it is not software). The main target of hadoop is to process the large data sets into smaller distributed computing. It is part of the Apache project sponsored by the Apache Software Foundation. As we observe in database management system, all the data are stored in organized form by following the rules like normalization , generalizations etc., and hadoop do not bother about the DBMS features as it stores large amount of data in servers. We are studying about Hadoop architecture and how big data is stored in servers by using this tools and the functionalities of Map Reduce and HDFS (Hadoop File System).

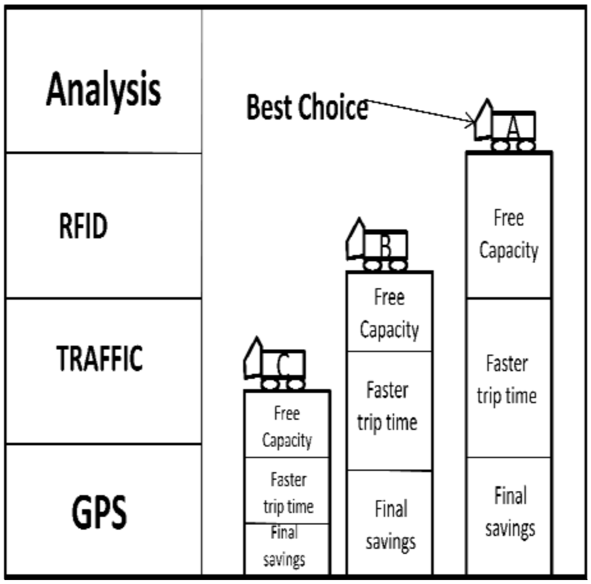
Keywords: Big Data, HDFS, Map Reduce Task Tracker, Job Tracker, Data Node, and Name Node.

Introduction-

Big Data Introduction: We probably heard about Big Data[1], but we may wondering what is it and why we should care. Ok, for starters big data is like data is getting bigger for a while now. From dawn of a time to less than a decade ago mankind generated about 5 Exabyte’s of data. In 2012 global data brought to 2.7 Zetta byte of data which is 500 times more data than all data ever generated in 2003. And it has grown 3 times bigger in 2015. Some of the reasons that data is getting bigger[2] is that continuously being generated more sources and more devices. Much back data is like videos, photos, comments and social media comments on web sites is unstructured. That means data is stored in structures pre defines tables, instead it’s often made up of volumes of text dates numbers in fact they are typically free from by nature. Certain data sources are arriving so fast not even a time to store the data before applying analytics to it. That is why traditional data management and analytics tools unable to store, process and analyze big data. So we could just ignore big data after all it’s worth the efforts? Turns out, it is. A recent study concluded only 10-15% organization would take full advantage of big data. In order to generate that level of insight and competitive advantage from big data innovative new approach and technologies are required because big data we looking at is like a mountain.



Imagine a logistic company mining data on truck pickup and delivering schedule on real time traffic patterns. The data they are using combines real time GPS speed from trucks. Public traffic pattern data or if I take cargos from data. Imagine they get a call from a new pickup, which truck should they send? The closest one right. So what if the route to the closets truck has heavy traffic jam? What if the cargo loaded on that truck doesn't allow space for new data? May be the route for that truck involve a series of great changes. In that case closest truck is not the best choice. They might b more costly less efficient or unable to service the customer needs. But the only way to arrival of optimal decision is to analyze multiple big data sources in real time.



Big Data = Big Impact

INTRODUCTION

As we see in our daily routine entire world has become an E-world (electronic world in common terms). So we can say increase in E-World in directly proportional to increase in data, so we required large no of servers to save the data.

To overcome this problem hadoop came into existence. Hadoop is so profound and powerful java tool which process large data into small data computations. Hadoop was created by Doug Cutting and Mike Cafarella in 2005 and Doug Cutting, who was working at Yahoo at the time named it after his son's toy elephant. And later they donated hadoop to apache so now we can say that hadoop is directed under the control of apache.

Hadoop architecture is playing a very important role in breaking of large data in to small data sets. In this paper we will know about architecture how the data[7] will get spitted and get computed and all its functionality.

Here comes a question in mind, how does Facebook, Google, Online marketing (retails), and all does store large amount of data?

The reason behind is all these frameworks uses hadoop system. The main reason hadoop came into existence of 3 factors.

They are velocity volume and verity. Velocity, large amount of data is coming with very high speed. Volume, large amount of data increasing day by day with huge volume. Verity, Data which are lots of verity. Ex: Audio, Video and etc.

Big data is creating large and growing files which are measured in terabytes (10^12) and petabytes (10^15) and the data is unstructured, we do not need relational models. This huge data is coming from tons of sources like users, applications like Facebook, yahoo, twitter etc, system, sensors and on and on.

The main problem hadoop is fixing is that in traditional hard disk transfer rate of data will be approx 60-100 MB/s and in hadoop there will be around 250-500 MB/s.

A. Reasons for hadoop evolution

Traditional Approach: when an enterprise will have a powerful computer it will process with very high speed it performance will be high and we can say computer is scalable. But there will be a certain point even a powerful computer cannot process Big Data. Now we can say computer is not scalable. This was one of the main reason hadoop came into existence.

Hadoop Approach: The main target of hadoop is to break Big Data[4] into smaller pieces and store into Commodity Hardware (Numerous Low Cost Computers known as Commodity Hardware). We do not require any powerful computers. At the same time all the computations are done on distributed system as well. All these computations are done at the same time and results send back to the application[6].

B. Hadoop simple Architecture

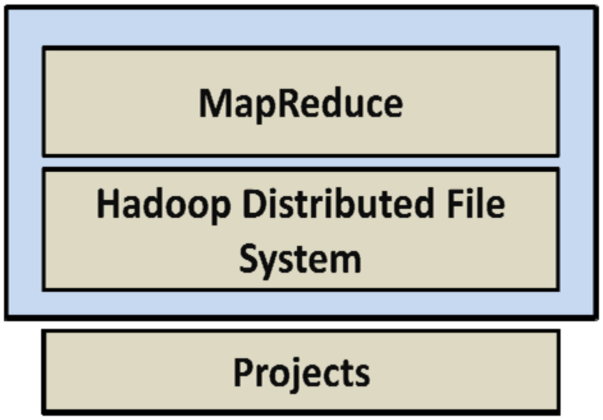


Figure 1: Hadoop Architecture

Hadoop Architecture consisting of three simple things i.e. MapReduce, HDFS, Projects. Hadoop MapReduce is a software framework for easily writing applications which process vast amounts of data (multi-terabyte data-sets) in parallel on large clusters (thousands of nodes) of commodity hardware in a reliable, fault-tolerant manner. Hadoop Distributed File System (HDFS) is a Java-based file system that provides scalable and reliable data storage that is designed to span large clusters of commodity servers. Finally the projects, as we said hadoop is framework of tools so all the tools come under this project. Some examples for projects are Hive, HBase, Mahout, Pig, Oozie, Flume, Sqoop etc.

Hadoop consisting of two nodes:-

1. Slave node
2. Master node.

1. Slave Node: Slave node are having two major components

- Task Tracker.
- Data Node.

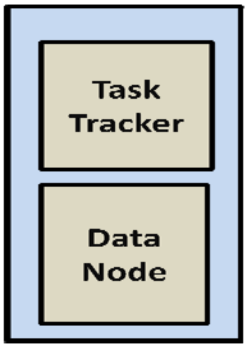


Figure 2: Slave node

1.1 Task Tracker: The job of task tracker is to processes the piece of task that has been given to this particular node.

1.2 Data Node: The job of data node is to manage piece of data that has been given to this particular node.

There can be n number of slave nodes. Here data is clustered in to these numerous slaves.

2. Master Node: The reason this is said to be a master node is that, master node having two another major components along with task tracker and data node.
1.1 Job Tracker.
1.2 Name Node.

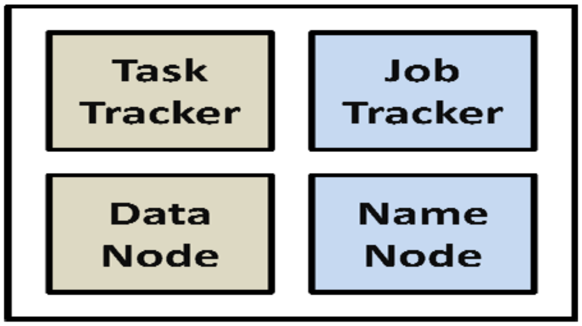


Figure 3: Master

1.1 Job Tracker: The role of job tracker component is to break higher task into smaller pieces and it will send each small computation to task trackers including its own. And after completing it will send back its results to the job tracker and it will combine the results and it will send back to application.

1.2 Name Node: It is responsible of keep an INDEX of which data is resigning on which data node.

Interaction between Master node and Slave node

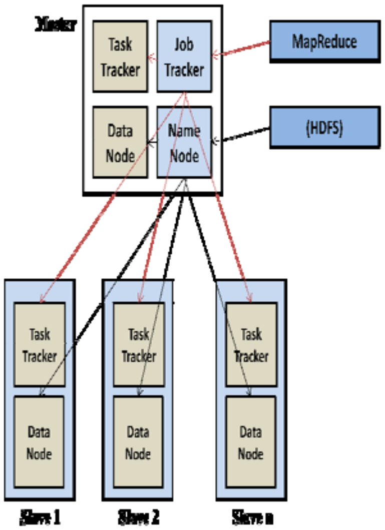


Figure 4: Interaction diagram

Job Tracker and Name Node functionality and interaction between them is observed in figure

MapReduce: Task Tracker and Job Tracker are the part of high level i.e. map reduce. So they all fall under the umbrella of map reduce.

File System: Data Node and Name Node are the part of high level i.e. map reduce. So they all fall under the umbrella of file system called HDFS.

Batch Processing:
One of the attribute of hadoop is that is a "Batch Processing" set of tools. So application would assign or provide a task for hadoop to in form of a QUEUE.

Once the task is completed it will inform application and results will be given back to application.

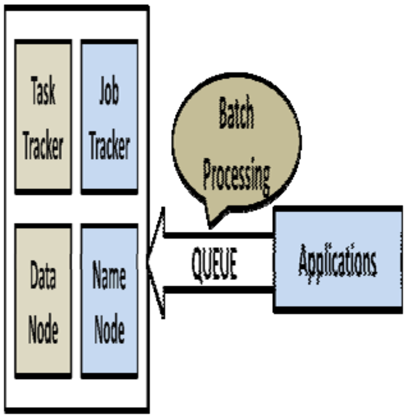


Figure 5: Batch Processing

Direction flow of data:

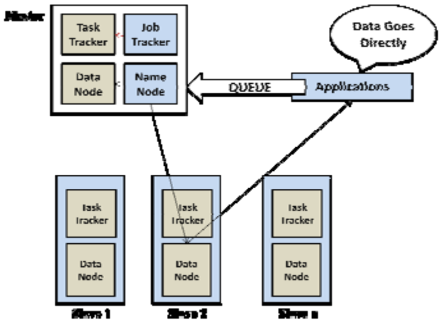


Figure 4: Data Flow Diagram

Here data flow directly when application comes in contact with master node and check the index on name node about the required information on which data node the data is residing. After the required information is gathered it directly goes to the application i.e. application doesn't wait for the name node to give back result. This is one of the important features is that; time optimizing in getting back result.

Fault Tolerance for Data and Master Backup:

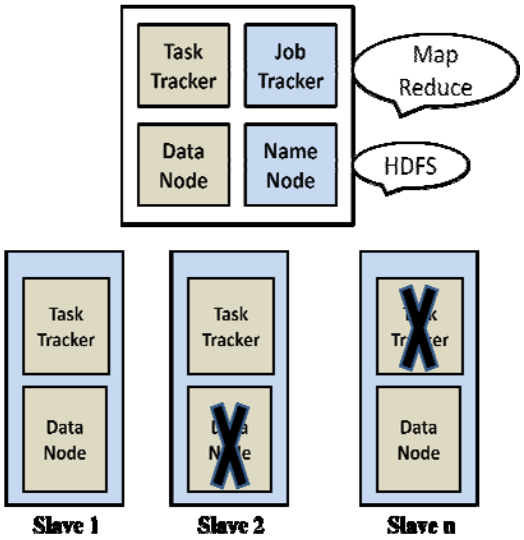


Figure 5: Fault Tolerance and Master Backup

One of the basic and important thing that hadoop keeps in mind is Fault Tolerance. If any of the Data Node gets failed, system doesn't go in to stop state by default Hadoop maintain 3 copies of each file and these copies are scattered along different computers. If any one of the task trackers gets failed to do its task, job tracker will detect the failure and it assigns the same task to other task tracker. When Master node gets failed then the tables that are maintained by name node which contain tables are backed up and copied over different computers. The enterprise version of hadoop also keeps two masters. One the main master and other the backup master.

Advantages of Hadoop:

One of the main advantages of hadoop to the programmers is

- Programmer need not worry about where the file is located name node will take care of it.
- Programmer need not worry about how to manage files; hadoop will take care of it.
- Programmer need not worry about how to break computations into pieces hadoop will take care of it.

- Programmer need not worry about writing the scalable programmers.

Consistency: - Component failures during execution of a job will not affect the outcome of the job.

Scalability: - Hadoop is highly scalable. As the no of slave nodes increases scalability also increases. Scalability of hadoop is linear, as we required processing speed to be increased then increase the no of computers.

Usage Areas:
There are tons of wide areas[3] where hadoop is used some of them are

- Social Media: Facebook, Twitter, yahoo, YouTube etc.
- Retail: e-Bay, Amazon etc.
- Searching Tools: Google.
- Companies: IBM etc.

And many more like American Airlines, The New York times and on and on. There are tons of users[5] who are using hadoop.

Conclusion

In this paper we have studied the entire architecture of hadoop and its functionality. It clearly explains that managing of big data in to clusters, how data is stored in numerous low cost computers (Commodity Hardware). Hadoop achieved Scalability and Consistency of data. As we seen in Database Management System we required organized data (following rows and columns) to store in server, we need follow normalizations techniques but where as in hadoop a programmer need not worry about relational data models.

Future Scope: - According to Yahoo point of view by the year 2015 50% of the enterprise[8] will processed by hadoop.

Biblography:-

[1] Advancing Discovery in Science and Engineering. Computing Community Consortium. Spring 2011.

[2] Drowning in numbers -- Digital data will flood the planet—and help us understand it better. The Economist, Nov 18, 2011. http://www.economist.com/blogs/dailychart/2011/11/big-data-0

[3] Computational Social Science. David Lazer, Alex Pentland, Lada Adamic, Sinan Aral, Albert-László Barabási, Devon Brewer,Nicholas Christakis, Noshir Contractor, James Fowler, Myron Gutmann, Tony Jebara, Gary King, Michael Macy, Deb Roy, and Marshall Van Alstyne. Science 6 February 2009: 323 (5915), 721-723.

[4] Big data: The next frontier for innovation, competition, and productivity. James Manyika, Michael Chui, Brad Brown, Jacques Bughin, Richard Dobbs, Charles Roxburgh, and Angela Hung Byers. McKinsey Global Institute. May 2011.

[5] Folowing the Breadcrumbs to Big Data Gold. Yuki Noguchi. National Public Radio, Nov. 29, 2011. http://www.npr.org/2011/11/29/142521910/the-digital-breadcrumbs-that-lead-to-bigdata

[6] The Search for Analysts to Make Sense of Big Data. Yuki Noguchi. National Public Radio, Nov. 30, 2011. http://www.npr.org/2011/11/30/142893065/the-search-for-analysts-to-make-sense-of-big-data

[7] The Age of Big Data. Steve Lohr. New York Times, Feb 11, 2012. http://www.nytimes.com/2012/02/12/sunday-review/big-datas-impact-in-the-world.html

[8] A Sustainable Future. Computing Community Consortium. Summer 2011.

[9] Windows Azure Platform: http://www.microsoft.com/windowsazure/

[10] Microsoft Service Bus and Access Control for Windows Azure platform AppFabric : http://www.microsoft.com/windowsazure/whitepapers/

[11] Windows Azure Tools – Constraints: http://msdn.microsoft.com/en-us/library/dd203058.aspx

[12] Microsoft Azure Comparison: http://cloudenterprise.info/2008/10/29/microsoft-azure-vs-amazon-google-and-vmware/

[13] Geneva Framework: http://download.microsoft.com/download/7/d/0/7d0b5166-6a8a-418a-addd-95ee9b046994/GenevaFrameworkWhitepaperForDevelopers.pdf

[14]SQL Azure: http://www.microsoft.com/windowsazure/sqlazure/

[15]WCF Data Services: http://msdn.microsoft.com/en-us/data/aa937697.aspx

[16]Windows Azure Platform AppFabric Services: http://msdn.microsoft.com/en-us/library/dd630576.aspx

Conclusion-

Windows Azure runs on machines in Microsoft data centers. Rather than providing software that Microsoft customers can install and run themselves on their own computers, Windows Azure is a service: Customers use it to run applications and store data on Internet-accessible machines owned by Microsoft. Those applications might provide services to businesses, to consumers, or both.

Dr. B. V.Ramana Murthy has done his PhD from Osmania University, presently he working as Professor in Computer Science and Engineering, has 18 years of experience in Teaching and R&D. His primary area of interest is Software Engineering & Web Engineering.

Mr. V Padmakar is pursuing PhD in CSE and has done his M Tech (CSE) from JNTUH, presently working as Professor in Computer Science and Engineering has 17 years of experience in Teaching and Industry. His primary area of interests is Software Engineering, Network Security and Data mining

Mr. Abhishek Reddy Mirupati is a Computer Science and Engineering student at Jyothishmathi College of Engineering and Technology pursuing his Bachelor of Technology in CSE. His primary area of interest is Object Oriented Programming and Data Base Management System.