A Big Data Hadoop Architecture for Online Analysis.

Suresh Lakavath, CSIR-URDIP,Pune, India.
Ramlal Naik .L, Acme Tele Power LTD,Haryana, India.

Abstract
Big Data is a collection of data that is large or complex to process using on-hand database management tools or data processing applications. Big Data has recently become one of the issues important in the networking world. Hadoop is a distributed paradigm used to manipulate the large amount of data. This manipulation contains not only storage as well as processing on the data. Hadoop is normally used for data intensive applications. It actually holds the huge amount of data and upon requirement perform the operations like data analysis, result analysis, data analytics etc. Now a day's almost every online users search for products, services, topics of interest etc. not only in Google and other search engines, but also more importantly on site itself (For example, in ecommerce site Amazon.com, search is the top product finding method used by site visitors).

Keywords
Hadoop, Big Data, Map Reduce, Facebook,HDFS.

1. Introduction

BIG data is the name used every where now a days in distributed paradigm on web. As the name shows it is the collection of sets of very huge amount of data in terabytes, pet bytes etc. associated with systems as well as algorithms used to analyze this massive data [5]. Sometimes we call big data as a phenomenon as this contains not only massive data also techniques to analyze and perform some actions or to get information from that data. In the last ten years of 19th century Google introduced this big data because at that time distributed computation was started. So, in early of 2000 Google introduced a new distributed file system named as GFS [6]. GFS was a great discovery in to handle massive data in the fashion to store, retrieve, and process and analyze [3] [4]. The major issue with this was it was not open source and many of the users demanded to use it. So, on the same technique of GFS, there was a new invention in big data handling named as Hadoop. Hadoop is used by many organizations either they are business oriented or social media, for example Yahoo, Facebook, Bit.ly, LinkedIn etc. Online users search for products, services, topics of interest etc. not only in Google and other search engines, but also more importantly on site itself (For example, in ecommerce site Amazon.com, search is the top product finding method used by site visitors). Facilitating searchers by providing relevant search results is something online search providers like Google, Bing and also site search providers continuously optimize and calibrate. Each data node sends a heartbeat to name node periodically. There are many causes that the data nodes fails of the heartbeat signal do not received by the name node e.g. network connectivity, data node over loaded or crashed etc. Upon the death of data node, name node has to balance the load of data and processing on rest of the nodes. But for the backup Hadoop contains a block replication factor. Upon the death of data node replication value surely increased and the information is passed to all the nodes that which block is going to replicate.

2. Hadoop in Death

Hadoop in fact have two core components, HDFS (Hadoop Distributed file system) use for storing the data and Map Reduce to process the data which is stored on its file system. Hadoop is pretty different from other distributed file systems like it has high fault tolerance and its design in fact made to commodity computers via simple programming model. If some of the machine is failed, its backup is always available to avoid the pause in service.
Hadoop is a free, Java-based programming framework that supports the processing of large data sets in a distributed computing environment [1]. It is part of the Apache project sponsored by the Apache Software Foundation. Hadoop makes it possible to run applications on systems with thousands of nodes involving thousands of terabytes. Its distributed file system facilitates rapid data transfer rates among nodes and allows the system to continue operating uninterrupted in case of a node failure this approach lowers the risk of catastrophic system failure, even if a significant number of nodes become inoperative.
Hadoop was inspired by Google's MapReduce, a software framework in which an application is broken down into numerous small parts. Any of these parts (also called fragments or blocks) can be run on any node in the cluster. The Hadoop framework is used by major players including Google, Yahoo and IBM, largely for applications involving search engines and advertising. The preferred operating systems are Windows and Linux but Hadoop can also work with BSD and OS X. HDFS is purely a distributed file system provides the high throughput and access the data in efficient manner. It has number of replicas to get data without any issue and quickly return to the user. One of the major goals to create these replicas is

to provide the availability all the time and also if some node is failed, nothing should be stopped. So, in simple words we can say that in Hadoop every block of data must have the replicas of itself.

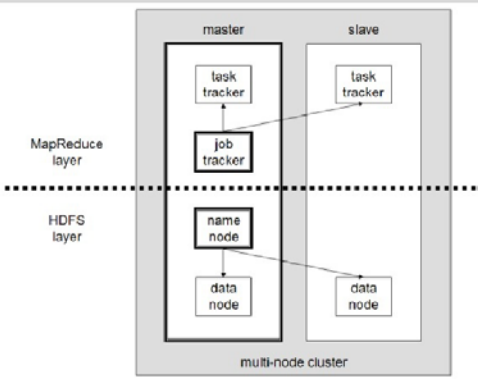

Fig 1. Multi Load Cluster Layers.

Map Reduce is another major component of Hadoop. It is the component use to perform processing on the data kept in HDFS. It gets the values in key/value pairs and after processing produces the result in the form of a value or set of some value or may also be in key/value pairs. MapReduce is a programming model and an associated implementation for processing and generating large data sets with a parallel, distributed algorithm on a cluster. MapReduce contains two major functions first is Map and the other one is Reduce. These two are the only functions perform processing on the data. These are the logical functions and the developers have to create the logic according to the required processing job. In the Map step, master node take the input and divide into its smaller sub problems and spread all these amongst the worker which actually gather the data as per required and after the gathering of data from all workers send it to Reduce workers which now perform the operation of shuffling and analysis n the base of gathered data.

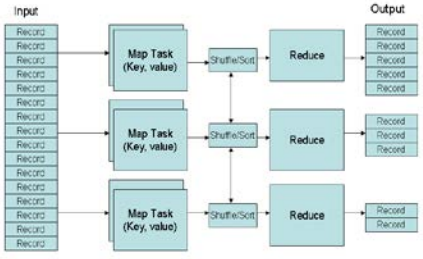

Fig 2. Map Reduce.

Deciding on what will be the key and what will be the value ➔ developer's responsibility
First of all, we have to decide what are the parties among to which results will be derived.

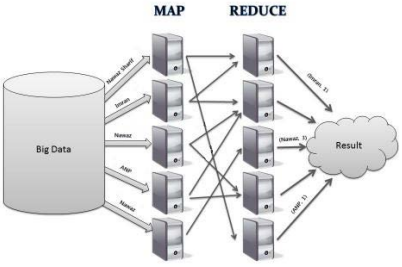

Fig 3 Big Data Transformation by Map Reduce.

In the above figure 1.0, data from server is like lines that contain Nawaz, ANP and Imran etc. Map function will map and after reduce it will give us the final result in the form of key/value pairs e.g. (Nawaz, 3), (Imran, 1) etc. Its programming logic would be as follows [9].

function map (String name, String document):
// name: blog name
// document: document contents
for each word w in document:
output (w, 1)
function reduces (String word, Counts):
// word: given name
// Counts: a list of aggregated counts
sum = 0
for each pc in Counts:
sum += pc
output (word, sum)

In above code, Mapper is just getting the input and making ready for reducer. Reducer in the end giving a value via omit in the form of key/value pair.

3. How HDFS Works

An HDFS cluster is comprised of a NameNode which manages the cluster metadata and DataNodes that store the data. Files and directories are represented on the NameNode by inodes [2]. Inodes record attributes like permissions, modification and access times, or namespace and disk space quotas.

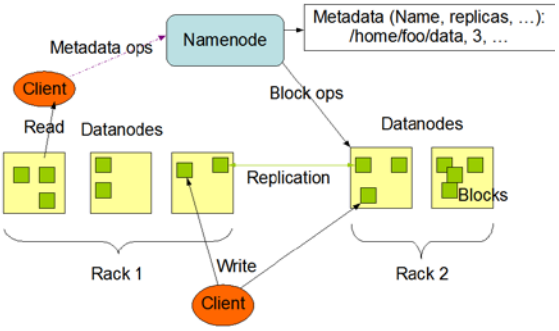

Fig 4 HDFS Architecture.

The file content is split into large blocks (typically 128 megabytes), and each block of the file is independently replicated at multiple DataNodes [8]. The blocks are stored on the local file system on the datanodes. The Namenode actively monitors the number of replicas of a block. When a replica of a block is lost due to a DataNode failure or disk failure, the NameNode creates another replica of the block. The NameNode maintains the namespace tree and the mapping of blocks to DataNodes, holding the entire namespace image in RAM. The NameNode does not directly send requests to DataNodes. It sends instructions to the DataNodes by replying to heartbeats sent by those DataNodes. The instructions include commands to: replicate blocks to other nodes, remove local block replicas, re-register and send an immediate block report, or shut down the node.

Within HDFS, a given name node manages file system namespace operations like opening, closing, and renaming files and directories. A name node also maps data blocks to data nodes, which handle read and write requests from HDFS clients. Data nodes also create, delete, and replicate data blocks according to instructions from the governing name node.

This is where Big Data Analytics solutions come in. In this above example, a typical Architecture to support Big Data Analytics is solution using open source Apache Hadoop framework. In Hadoop architecture - big volumes, variety and velocity of online data are collected and then stored in HDFS file system. Hadoop architecture also provides RDBMS like databases such as HBase, for storing big data in traditional style, particularly useful for beginners and new users of these Big Data Architectures. As we can see in this example, a big data landing zone is set up on a Hadoop cluster to collect big data, which is then stored in HDFS file system.

4.Big Data Hadoop Application

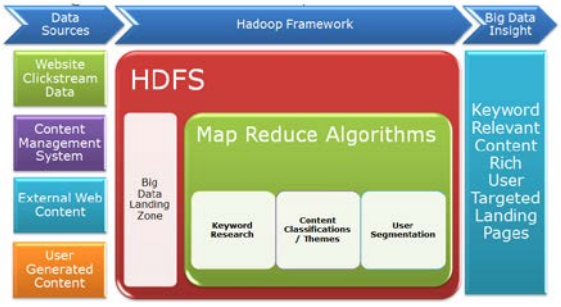

Fig 5 Big Data Hadoop Architecture.

Online users search for products, services, topics of interest etc. not only in Google and other search engines, but also more importantly on site itself (For example, in eCommerce site Amazon.com, search is the top product finding method used by site visitors). Facilitating searchers by providing relevant search results is something online search providers like Google, Bing and also site search providers continuously optimize and calibrate.

From an Online Marketing perspective, once the searchers click through the search results and arrive at the website (if coming through external search like Google) or arrive at the product or topic page they were searching internally on the site, that page of arrival from a search result, called as landing page in Online Marketing terminology, is very important for:

- Improving Conversion Rate (%) of the site.
- Traffic dispersion to subsequent stages of the site.
- Improving site engagement for the users

Delivering dynamic and search relevant landing pages is very important, particularly for large websites like eCommerce stores, Music & Movie download sites, Travel websites etc. While delivering keyword or search relevant landing pages dynamically across thousands of keywords, perhaps across hundreds of thousands of keywords for large websites, itself is a big challenge; even bigger challenge is to deliver these dynamic, search relevant landing pages targeted to each of different user segments. As already discussed previously, luckily Big Data Analytics solutions are available now to solve these Big Data challenges in Online Marketing.

Large websites generate and also need to process, huge volumes of different varieties of data as below:

- Website clickstream data collected through Web Analytics applications like Omniture and from web server logs.
- The website content such as product content, marketing content, navigation etc. in various formats like text, images, videos etc. which is available in the web content management systems.
- External web content typically collected by web crawlers, which includes content such as
    - Product content from competitor websites
    - Marketing collaterals from external industry websites etc.
- User generated content such as product reviews, user survey feedback, social media posts, online discussions, tweets, blog posts, online comments, Wiki articles etc.

Most of the above varieties of data are unstructured or semi-structured, and hence cannot be collected and processed in traditional RDBMS databases like Oracle or MySQL.

For large websites, it is not just important to collect large volumes of variety of data as shown above, but it is also important to handle the velocity at which all these data is getting generated online, particularly clickstream data and user generated content.

This is where Big Data Analytics solutions come in. In this above example, a typical Architecture to support Big Data Analytics is solutioned using open source Apache Hadoop framework. In Hadoop architecture - big volumes, variety and velocity of online data are collected and then stored in HDFS file system. Hadoop architecture also provides RDBMS like databases such as HBase, for storing big data in traditional style, particularly useful for beginners and new users of these Big Data Architectures. As we can see in this example, a big data landing zone is set up on a Hadoop cluster to collect big data, which is then stored in HDFS file system.

Using Map-Reduce programming method, Online Marketing Analysts or Big Data Scientists or Analysts develop and deploy various algorithms on a Hadoop cluster for performing Big Data Analytics [7]. These algorithms can be implemented in standard Core Java programming language which is the core programming language used for executing various services for collecting, storing and analyses of big data in Hadoop architecture. Additional programming languages like Pig, Hive, Python or R can be used to implement the same algorithms with less number of lines of code to be deployed. However code written in any of these additional languages would still is compiled into Core Java code by Java Compilers for execution on Big Data Hadoop Architectures.

Some of the use cases of Online Marketing Algorithms which can be implemented on Hadoop Architecture for deriving Analytics are shown in the same example. All these algorithms are deployed using the Map-Reduce programming method.

- **Keyword Research**: Counting the number of occurrences in content and search for hundreds of thousands of keywords across the diverse variety of data collected into Hadoop and stored in HDFS. This algorithm would help identify top keywords by volume, and also the long tail of hundreds of thousands of keywords searched by users. Even new hidden gems among keywords can be discovered using this algorithm to deploy in SEM/SEO campaigns.
- **Content Classifications / Themes**: Classify the user generate content and also web content into specific themes. Due to huge processing capabilities of Hadoop Architecture, huge volumes of content can be processed and classified into dozens of major themes and hundreds of sub themes.
- **User Segmentation**: Individual user behavior available in web clickstream data is combined with online user generated content and further combined with user targeted content available in web content management systems to generate dozens of user segments, both major & minor segments. Further this algorithm would identify the top keywords and right content themes targeted for each of the dozens of user segments, by combining the output from other algorithms used for Keyword Research and Content Classifications.

Also, since the Hadoop Architecture is running on clusters of computers, all the above algorithms can not only process huge voluminous amounts and varieties of data, but can handle data in motion which keeps coming into the Hadoop Big Data landing zone in near real time. This would enable the Online Marketing Campaigns to be tweaked in near real time to derive better ROIs from Online Marketing spends. In the example illustrated above, the output from the 3 algorithms running in parallel is dynamic Keyword Relevant Content Rich User Targeted Landing Pages generated in near real time, for hundreds of thousands of keywords, across dozens of content themes and targeted across dozens of user segments. This output would be integrated with eCommerce platforms or Web Content Management Systems or with Web Portals for creation, production & delivery of Keyword Relevant Content Rich User Targeted Landing Pages in near real time.

5.short comings in hadoop

No doubt Hadoop is using by every big organization in this

World and also using in cloud computing but still there are Some issues or shortcomings in Hadoop.

- Low level programming paradigm and schema
- Strictly batch processing
- Incremental Computation
- Improving Conversion Rate (%) of the site.
- Traffic dispersion to subsequent stages of the site.
- Improving site engagement for the users.
- Time Skew

If we go in-depth of Hadoop we can easily observe that it's schema and its programming paradigm is made in low level languages. In MapReduce, if process been mapped and after reduce its output must be saved in local storage

or we can say it is just materialized approach. We cannot use them further until we save it on disk.
For small change MapReduce have to run whole MapReduce application, this is very costly in time and resources [10].

6.conclusion

In this paper we have discussed big data, Hadoop, its overall architecture, technical details. It's usage in social media and Online Marketing and highlighted its shortcomings.
This manipulation contains not only storage as well as processing on the data. Hadoop is normally used for data intensive applications. It actually holds the huge amount of data and upon requirement perform the operations like data analysis, result analysis, data analytics etc. Now a day's almost every online users search for products, services, topics of interest etc. not only in Google and other search engines, but also more importantly on site itself. Future work will consist in implementing the same system with on-line analysis big data framework such as Spark Streaming [11].

References

[1] Hadoop, http://hadoop.apache.org/.
[2] HDFS,http://hadoop.apache.org/docs/r1.2.1/hdfs_design.html
[3] T. White, Hadoop: the Definitive Guide, O'Reilly, 3rd ed., 2012.
[4] Hadoop, http://delicious.com/jhofman/hadoop.
[5] Y. Lee and Y. Lee, Detecting DDoS Attacks with Hadoop, ACM CoNEXT Student Workshop, Dec.2011.
[6] Intro to Big Data using Hadoop by Sergejus Barinovas.
[7] RIPE Hadoop PCAP, https://labs.ripe.net/Members/wnagele/large-scalepcap-data-analysis-using-apache-hadoop, Nov. 2011.
[8] The Hadoop Distributed File System: Architecture and Design by Apache.
[9] Impact of Big Data: Networking Considerations and Case Study.Vol.12 No.12, pp- 30-34 2012.
[10] Incremental Page Rank for Twitter Data Using Hadoop by Ibrahim Bin Abdullah.
[11] M. Zaharia, T. Das, H. Li, S. Shenker, and I. Stoica, "Discretized streams: An efficient and fault-tolerant model for stream processing on large clusters," in Proceedings of the 4th USENIX Conference on Hot Topics in Cloud Ccomputing, ser. HotCloud'12. Berkeley, CA, USA: USENIX Association, 2012.