REVIEW PAPER ON HADOOP AND MAP REDUCE

Nishant Rajput[1], Nikhil Ganage[2], Jeet Bhavesh Thakur[3]

[1]*Department of Electronics and Telecommunication, Manav Rachna International University*
[2]*Department of Information Technology, MIT Pune, Pune University*
[3]*Final year Student Information Technology, Fr CRCE, Mumbai University*

Abstract

Big data is a term that clarifies the procedures and advancements to deal with expansive volume of data which could be both structured and unstructured. Big data can be dissected for experiences that prompt better choices and vital business moves and in addition for everyday utilization the review paper is presented in two of the most fundamental steps required to adhere Big Data solutions in order to provide a deep insight for all the new students and industry experts to understand this world of Big Data in a better and joyful manner.

Keywords: *Hadoop, map reduce, Big data, Hadoop architecture, MapReduce Architecture.*

----------***----------

1. INTRODUCTION

Big Data is a data whose scale, decent variety, and intricacy require new engineering, methods, calculations, and investigation to oversee it and concentrate esteem and concealed learning from it. Hadoop is the center stage for organizing Big Data, and takes care of the issue of making it valuable for investigation purposes. Hadoop is an open source programming venture that empowers the appropriated handling of vast data sets crosswise over bunches of item servers. It is intended to scale up from a single server to thousands of machines, with a high level of adaptation to internal failure. Since the big data is included principally with the overwhelming or gigantic measure of data , the speed to deal with the tremendous measure of data comes in to the photo. Subsequently the meaning of big data as the three Vs could be clarified like:

Volume – Collection of data from an assortment of sources, including B2B, B2C exchanges, online networking and data from sensor or machine-to-machine data. To store such enormous measure of data and make a data warehouse was getting troublesome until the point that Big Data came into picture to improve the execution of the data warehousing. When we begin discussing volume, we're discussing madly a lot of data. As we push ahead, we will have increasingly enormous accumulations.

Facebook, for instance, stores photos. That announcement doesn't start to boggle the psyche until the point that you begin to understand that Facebook has a bigger number of clients than China has individuals. Each of those clients has put away a mess of photos. Facebook is putting away approximately 250 billion pictures. Velocity– It alludes to the speed of handling the data. How about we take a case from communication industry, Service supplier mammoths like Vodafone,Airtel have billions of clients worldwide and envision each individual making at least 10 calls every day, or perusing web with various rates now to deal with such colossal measure of data and give offers accordingly, after dissecting the data of billions of records quickly gets troublesome thus the vector speed assumes an imperative part in the photo. Variety – Data comes in a wide range of organizations – from organized, numeric data in conventional databases like Oracle to unstructured content records, email, video, sound, stock ticker data and money related exchanges. for instance, email messages. A legitimate disclosure process may require filtering through thousands to a large number of email messages in an accumulation. Not one of those messages will be precisely similar to another. Everyone will comprise of a sender's email address, a goal, in addition to a period stamp. Each message will have human-composed content and possible attachments.

2. HADOOP: SOLUTION FOR BIG DATA PROCESSING

• What is Hadoop and why Hadoop?

Hadoop is a product framework that can be introduced on a ware Linux cluster to allow substantial scale distributed data investigation. Hadoop gives the strong Hadoop Distributed File System (HDFS) and in addition a Java-based API that permits parallel processing over the nodes of the cluster. Programs utilize a Map/Reduce execution engine which works as a fault-tolerant distributed figuring system over huge data sets - a strategy advanced by use at Google. There are separate Map and Reduce steps, each progression done in parallel, each working on sets of key-value sets. Processing can be parallelized more than a huge number of nodes chipping away at terabyte or bigger measured data sets. The Hadoop framework naturally plans map undertakings near the data on which they will work, with "close" which means a similar node or, at any rate, a similar rack. Node disappointments are likewise taken care of consequently

Hadoop is an open source tool from the ASF – Apache Software Foundation. Its capacity is to storing data and running applications on clusters of product equipment. It

gives huge capacity to any sort of data. As it is Open source project it implies it is uninhibitedly accessible and even its source code can be changed according to the necessities.

Hadoop make it a unique platform:
• Flexibility to store and mine any kind of data whether it is organized, semi-organized or unstructured. It is not limited by a solitary outline.
• Excels at processing data of complex nature, its scale-out architecture isolates workloads over various nodes. Another additional favorable position is that its adaptable file-system wipes out ETL bottlenecks.
• Scales monetarily, as examined it can be conveyed on item equipment. Aside from this its open-source nature makes preparations for merchant bolt.
• Architecture of Hadoop.
In Hadoop architecture you need to learn 3 things.
1. HDFS
2. MapReduce
3. Yarn

Hadoop distributed file system-HDFS is the world's most dependable storage system. HDFS stores extensive files running on a cluster of commodity hardware. It takes a shot at the guideline of capacity of less number of huge files as opposed to the colossal number of little files. HDFS stores data dependably even on account of equipment disappointment. It gives high throughput by giving the data access in parallel.

3. HADOOP ARCHITECTURE

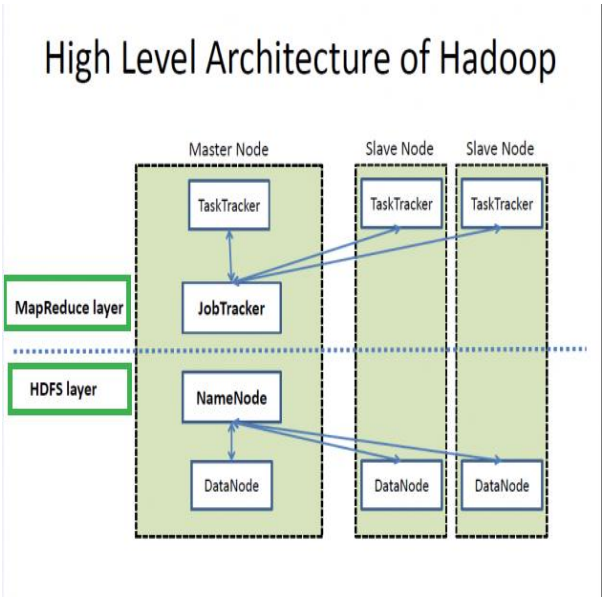

Fig 1: Architecture

NameNode

It is the master daemon that keeps up and deals with the DataNodes (slave nodes). It records the metadata of the considerable number of files put away in the cluster, e.g. location of blocks stored, the size of the files, permissions, hierarchy and so on. It records every single change that happens to the file system metadata.

For example, if a file is deleted in HDFS, the NameNode will immediately record this in the EditLog. It regularly receives a Heartbeat and a block report from all the DataNodes in the cluster to ensure that the DataNodes are live. It keeps a record of all the blocks in HDFS and in which nodes these blocks are stored.

DataNode

These are slave daemons which runs on each slave machine. The actual data is stored on DataNodes. They are responsible for serving read and write requests from the clients. They are also responsible for creating blocks, deleting blocks and replicating the same based on the decisions taken by the NameNode.

For processing, we utilize YARN(Yet Another Resource Negotiator). The segments of YARN are ResourceManager and NodeManager.

Resource Manager

It is a cluster level (one for each cluster) component and keeps running on the master machine. It manages resources and schedule applications running over YARN.

NodeManager

It is a node level component (one on every node) and keeps running on each slave machine. It is responsible for managing containers and monitoring resource utilization in each container. It also keeps track of node health and log management. It continuously communicates with Resource Manager to remain up-to-date.

You can perform parallel processing on HDFS utilizing MapReduce.

4. MAPREDUCE

MapReduce is the processing layer of Hadoop. MapReduce is a programming model intended for processing substantial volumes of data in parallel by partitioning the work into an arrangement of free undertakings. You simply need to put business rationale in the way MapReduce works and rest things will be taken care by the framework. Work (complete job) which is put together by the client to master is partitioned into little works (tasks) and doled out to slaves. MapReduce is the most complex piece of Hadoop as it needs programming. It is the center component of processing in a Hadoop Ecosystem as it gives the rationale of processing. At the end of the day, MapReduce is a product framework which helps in composing applications that procedures expansive data sets utilizing distributed and parallel calculations inside Hadoop condition. In a MapReduce program, Map() and Reduce() are two capacities. The Map work performs activities like sifting, gathering and arranging. While Reduce work totals and outlines the outcome created by map work. The outcome produced by the Map work is a key value combine (K, V) which goes about as the contribution for Reduce function. A typical MapReduce application has two capacities, a Mapper

and a Reducer. Mappers and Reducers will be keep running as tasks on nodes in the cluster. The Mapper capacities sort out pieces of data in a way that enables the data to be amassed and sent to Reducer capacities for any sort of total rationale. For instance, in the Word Count application the Mapper capacities read pieces of content and yield each word they find. On the off chance that few Mappers yield a similar word, each of those yields are accumulated to a solitary Reducer which can check the quantity of times it has gotten a

4.1 MapReduce Architecture

The figure shown below illustrates the various parameters and modules that can be configured during a MapReduce operation**:**

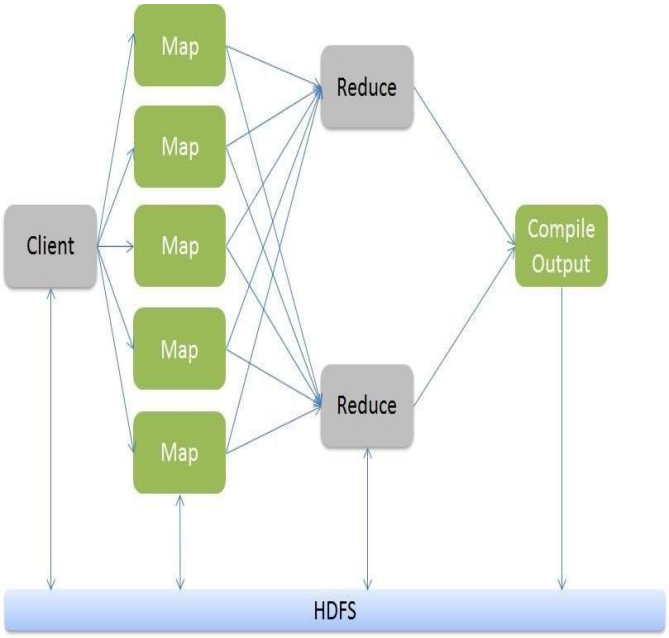

Fig 2: MapReduce architecture

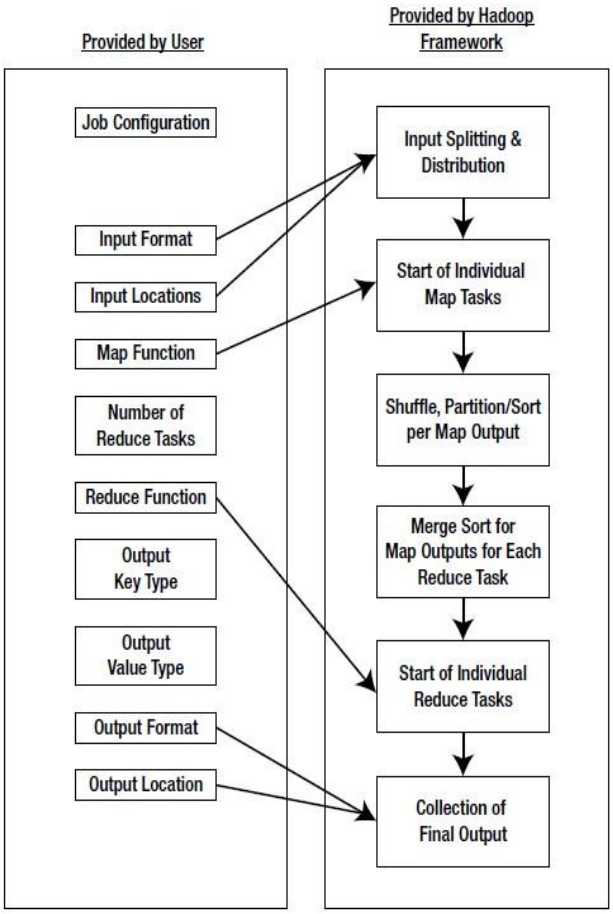

Fig 3: Architecture of Map Reduce.

The number of Map and Reduce nodes can likewise be characterized. You can set Partitioner work which partitions and transfer the tuples which as a matter of course depends on a hash work. At the end of the day we can set the choices with the end goal that a particular arrangement of key value sets are transferred to a particular lessen undertaking. For instance if your key value comprises of the year it was recorded, at that point we can set the parameters with the end goal that all the keys of particular year are transferred to a specific reduce task. The Hadoop framework comprises of a single master and multiple slaves. Each master has JobTracker and each slave has TaskTracker. Master disseminates the program and data to the slaves. Task tracker, as the name recommends monitor the undertaking coordinated to it and transfers the data to the JobTracker. The JobTracker screens all the status reports and re-starts the failed tasks assuming any.

Combiner class are keep running on map undertaking nodes. It takes the tuples radiated by Map nodes as info. It essentially reduces operation on the tuples discharged by the map node. It resembles a pre-decrease assignment to spare a ton of bandwidth. We can likewise pass worldwide constants to every one of the nodes utilizing 'Counters'. They can be utilized to monitor all occasions in map and lessen tasks. For instance we can pass a counter to ascertain the insights of an occasion past a specific edge.

5. CONCLUSION

Hadoop MapReduce programming worldview and HDFS are progressively being utilized for processing vast and unstructured data sets. Hadoop empowers collaborating with the MapReduce programming model while concealing the intricacy of conveying, arranging and running the product components in the general population or private cloud. Hadoop empowers clients to make cluster of item servers. MapReduce has been displayed as a free platform-asa-benefit layer appropriate for various necessity by cloud suppliers. It additionally empowers clients to comprehend the data processing and investigating.

REFERENCES

[1]. Apache Hadoop. http://hadoop.apache.org/

[2]. T. White, Hadoop: The Definitive Guide. O'Reilly Media, Yahoo! Press, June 5, 2009.

[3]. J. Venner, Pro Hadoop. Apress, June 22, 2009. [17]. S. Weil, S. Brandt, E. Miller, D. Long, C. Maltzahn, "Ceph: A Scalable, High-Performance Distributed File System," In Proc. of the 7th Symposium on Operating Systems Design and Implementation, Seattle, WA, November 2006.

[4]. B. Welch, M. Unangst,F Ahmad, S. Y. Lee, M. Thottethodi, T. N. Vijaykumar. PUMA: Purdue MapReduceBenchmarks Suite. ECE Technical Reports, 2012

[5]. Apache Hadoop NextGen MapReduce (YARN). http://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/YARN.html.

[6]. Jonathan Stuart Ward and Adam Barker "Undefined By Data: A Survey of Big Data Definitions" Stamford, CT: Gartner, 2012.

[7]. Ahmed Eldawy, Mohamed F. Mokbel "A Demonstration of SpatialHadoop:An Efficient MapReduce framework for Spatial Data" Proceedings of the VLDB Endowment, Vol. 6, No. 12 Copyright 2013 VLDB Endowment 21508097/13/10.

[8]. Jeffrey Dean and Sanjay Ghemawat "MapReduce: Simplified Data Processing on Large Clusters" OSDI 2010.

[9]. Niketan Pansare1, Vinayak Borkar2, Chris Jermaine1, Tyson Condie "Online Aggregation for Large MapReduce Jobs" August 29September 3, 2011, Seattle, WA Copyright 2011 VLDB Endowment, ACM.

[10]. Tyson Condie, Neil Conway, Peter Alvaro, Joseph M. Hellerstein "Online Aggregation and Continuous Query support in MapReduce" SIGMOD'10, June 6–11, 2010, Indianapolis, Indiana,USA. Copyright 2010 ACM 978-1-4503-0032-2/10/06.

[11]. Jonathan Paul Olmsted "Scaling at Scale: Ideal Point Estimation with 'Big-Data" Princeton Institute for Computational Science and Engineering 2014.

[12]. Jonathan Stuart Ward and Adam Barker "Undefined By Data: A Survey of Big Data Definitions" Stamford, CT: Gartner, 2012.

[13]. Balaji Palanisamy, Member, IEEE, Aameek Singh, Member, IEEE Ling Liu, Senior Member, IEEE" Cost-effective Resource Provisioning for MapReduce in a Cloud"gartner report 2010, 25

[14]. Mrigank Mridul, Akashdeep Khajuria, Snehasish Dutta, Kumar N " Analysis of Bidgata using Apache Hadoop and Map Reduce" Volume 4, Issue 5, May 2014" 27.

[15]. Kyong-Ha Lee Hyunsik Choi "Parallel Data Processing with MapReduce: A Survey" SIGMOD Record, December 2011 (Vol. 40, No. 4)

[16]. Chen He Ying Lu David Swanson "Matchmaking: A New MapReduce Scheduling" in 10th IEEE International Conference on Computer and Information Technology (CIT'10), pp. 2736–2743, 2010

[17]. Comparing the Hadoop Distributed File System (HDFS) with the Cassandra File System (CFS) White Paper BY DATASTAX CORPORATION August 2013

[18]. Matthew Edwards, Aseem Rambani, Yifeng Zhu*a, Mohamad Musavi "Design of Hadoop-based Framework for Analytics of Large Synchrophasor Datasets"1877-0509 © 2012 Published by Elsevier B.V. Selection and/or peer-review under responsibility of Missouri University of Science and Technology. doi: 10.1016/j.procs.2012.09.065

[19]. Jiong Xiea,b, FanJun Mengc, HaiLong Wangc, HongFang Panb, JinHong Chengb, Xiao Qina" Research on Scheduling Scheme for Hadoop clusters" 1877-0509 © 2013 The Authors. Published by Elsevier B.V. Selection and peer review under responsibility of the organizers of the 2013 International Conference on Computational Science. doi: 10.1016/j.procs.2013.05.423.

[20]. Samson Oluwaseun Fadiyaa*, Serdar Saydamb,Vanduhe Vany Zirac" Advancing big data for humanitarian needs" 1877-7058 © 2014 Published by Elsevier Ltd. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/3.0/).
Selection and peer-review under responsibility of the Organizing Committee of HumTech2014 doi:.1016/j.proeng.2014.07.043.

[21]. Chia-WeiLeea, Kuang-YuHsieha, Sun-YuanHsieha,b,∗, Hung-ChangHsiaoa "A Dynamic Data Placement Strategy for Hadoop in Heterogeneous Environments" Big Data Research 1 (2014) 14–22.