International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 3 Issue 9, September- 2014

A STUDY ON BIG DATA

Indumathi V[1]
[1]Computer science and engineering,
Prathyusha institute of technology and management,
Anna University, India

Divya U[2]
[2](Computer science and engineering,
Prathyusha institute of technology and management, Anna University, India

Vishnu Priya P[3]
[3]Computer science and engineering,
Prathyusha institute of technology and management,
Anna University, India

***Abstract*—Big data is the term for a collection of data sets so large and complex that it becomes difficult to process using on-hand database management tool or traditional data processing applications. The challenges include capture, curation, storage, search, sharing, transfer, analysis and visualization. Big data sizes are constantly moving target as 2012 ranging from a few dozen terabytes to petabytes of data in a single data set. Big data is high volume, high velocity and high variety and veracity information assets that requires new form of processing to enable enhanced decision making. We have proposed a study about Big data on Hadoop and Map-Reduce. As a result of this study gives an idea of pros and cons of Big data and helps in implementation of Hadoop and Map-Reduce in future.**

Index terms*—*Big data, Hadoop, HDFS, Map-Reduce, Job tracker, Task tracker.

I.INTRODUTION

Big data is a popular term used to describe the exponential growth and availability of data, both structured, semi-structured and unstructured.[4] Big data involves several new issues to consider such as discovery, iteration, flexibility capacity, mining and predicting and decision management. [1] Articulated mainstream definition of big data as three Vs- velocity, volume and variety. There are five key approaches to analyzing big data and generating insight namely discovery tool, BI (Business Intelligence)tool, In-data analytics, Hadoop and decision management. In this paper, we mainly focus on integrated use of Hadoop. Big data has incredibly successful framework named as Map-Reduce. Map-Reduce is a programming model and associated implementation for processing and generating large data sets. Implementation of Map-Reduce framework was adopted by an Apache open source project named Hadoop for reliable, scalable and distributed computing.

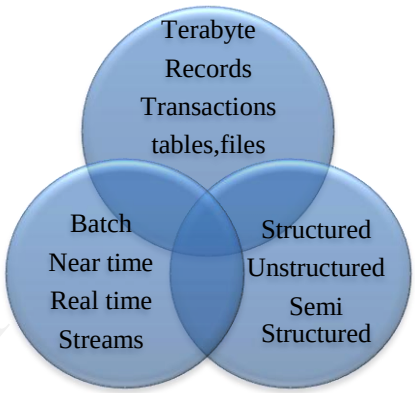

Fig 1: Big data as 3V's

Hadoop is an Apache top level project being built and used by a global community of contributors and users. It is licensed under the Apache license 2.0 which is written in Java programming language, with some native code in C and command line utilities written as shell-script.

The Apache Hadoop framework is composed of the following modules:

- Hadoop Common - contains libraries and utilities needed by other Hadoop modules
- Hadoop Distributed File System (HDFS) - a distributed file-system that stores data on commodity machines, providing very high aggregate bandwidth across the cluster.
- Hadoop YARN - a resource-management platform responsible for managing compute resources in clusters and using them for scheduling of users' applications.
- Hadoop Map-Reduce - a programming model for large scale data processing.[3]

Hadoop-compatible file system should provide location awareness: the name of the rack where a worker node is. Hadoop applications can use this information to run work on the node where the data is on the same rack/switch, reducing backbone traffic. HDFS uses this method when replicating data to try to keep different copies of the data on different racks.

The goal is to reduce the impact of a rack power outage or switch failure, so that even if these events occur, the data may still be readable.Map-Reduce runs on a large cluster of commodity machines and is highly scalable: a typical map reduce computation process many terabytes of data on thousands of machines.

Programmer finds the system easy to use: the hundreds of map reduce program have been implemented and one thousand map reduce jobs are executed on Google's cluster every day. The ultimate goal of the study is to analyze the history of Big data and software's (Hadoop) used in it with functions (Map Reduce).

II.BACKGROUND

HFDS can be part of a Hadoop cluster or can be a stand-alone general purpose distributed file system. An HFDS cluster primarily consists of

- NameNode that manages file system metadata
- DataNode that stores actual data

HDFS stores very large files in blocks across machines in a large cluster. It has data awareness between nodes and designed to be deployed on low-cost hardware. Typical Hadoop cluster integrates the MapReduce and HDFS. Hadoop architecture has Master node and Slave node. Master node contains Job tracker node and Task tracker node which are in MapReduce Layer whereas Name node and Data node are present in HDFS layer. Slave node contains Task tracker node and Data node present in Map Reduce layer and HDFS layer.[3] as shown in fig 2.

Map reduce framework has large number of cluster nodes,for each cluster node a single job tracker per master will be responsible for scheduling and monitors the slave process and then re-executes the tasks when it fails and a single task tracker per slave will execute the task as directed by the masters. The MapReduce programming paradigm executes a job in two phases: *Map* and *Reduce*. In map step the master node takes large problem input and sliced it into smaller sub problems; distributes these to worker nodes. The worker node will do the same as master node and creates a multi-level tree structure. The worker processes smaller problem and hands back to master. In reduce step the master node takes the answer to the sub problem and combines them in a predefined way to get output to the original problem.

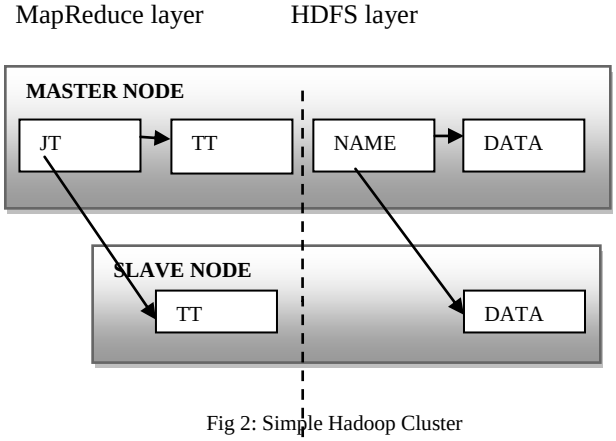

Fig 2: Simple Hadoop Cluster

III.METHODOLOGY

Map reduce core functionality:

In this section we are going to see about key functions of MapReduce framework which are as follows: 1.Input reader which divides input into appropriate size splits, which gets assigned to Map functions. 2. Map function maps the file data to smaller, intermediate <key, values> pairs. 3. Partition function finds the correct reducer given the key and number of reducer, return to the desired Reduce node. 4. Compare function input for Reduce is pulled from the Map intermediate output and sorted according to the Compare function. 5. Reduce function takes the intermediate values and reduced to a smaller solution and handed back to the framework. 6. Output Writer writes file output.[1]

A. Input reader:

Map Reduce framework requires parsing each record at reading inputs. The input reader reads data from stable storage (typically distributed file systems) and generates <key, values> pairs. The Key and Value classes have to serializable by the framework and hence need to be implemented by writable interface. The key classes have to implement the writable comparable interface to facilitate sorting by the framework.

B. Map function:

Map function takes a series of <key, value>pair and process each then generate zero or more output <key, value> pair. The input and output types of the map can be (and often are) different from each other. If application is doing word count, the map function would break the lines into the words and output a <key, value> pairs for each word. Each output pair would contain the word as the key and the number of instance o that word in the line as the value.

As we know, the mapper functions produce intermediate records and those record need not to be in same type as input record. All intermediate values associated with a given output key are subsequently grouped by the framework, and passed to the Reducer to determine the final output.

C. Partition function:

Each Map function output is allocated to particular Reducer by the application partition function for sharding process. The key value and the number of reducer

are given as input to the partition function which in turn returns the index of the desired reducer. The key is used to derive the partition, typically by a hash function.

D. Compare function:

Compare function has 2 primary phases: Shuffle and Sort. After partitioning the same key are shuffled among data sets and sorted. The shuffle and sorting phases occurs simultaneously when the Map-output are being fetched.

E. Reduce function:

In this phase the method is called for each <key, (list of values)> pair in the grouped inputs. The lists of values are summed and write it to the value of the key. The output of this function is not sorted.

F. Output writer:

At the end the output of the reducer function will write to the stable storage such as Distributed file system.

KEY PROS OF THE MAP REDUCE FUNCTION:

Fault Tolerance: Map Reduce is highly fault tolerant even thoughfailures are common phenomenon in the large scale distributed computing and it includes master failure and slave failure.

Slave failure: The master monitors every mapper and reducer periodically. If no response received from the slave for certain amount of time is marked as failed. The ongoing task on the failed machine will be mapped to another mapper and executes from the beginning. The completed reduce task need not to be re-executed since the output are stored in the global file systems.

*Master failure:*Since the master node is single node the probability of failure will be very less. If the master node fails it restarts the entire job.

Flexibility: As we know the Map Reduce is Schema free and Index free it does not have any dependency on data model and schema.

High Scalability: As the name itself indicates, a large amount of data can be stored and processed and hence it is highly scalable.[10]

IV.EMERING TECHNOLOGY

1. Column-oriented databases

Traditional, row-oriented databases are excellent for online transaction processing with high update speeds, but they fall short on query performance as the data volumes grow and as data become more unstructured. Column-oriented databases store data with a focus on columns, instead of rows, allowing for huge data compression and very fast query times.[8]

2. Schema-less databases, or nosql databases

There are several database types that fit into this category, such as key-value stores and document stores, which focus on the storage and retrieval of large volumes of unstructured, semi-structured, or even structured data.

They achieve performance gains by doing away with some (or all) of the restrictions traditionally associated with conventional databases, such as read-write consistency, in exchange for scalability and distributed processing[8].

3. Mapreduce

This is a programming paradigm that allows for massive job execution scalability against thousands of servers or clusters of servers.

Any MapReduce implementation consists of two tasks:

- The "Map" task, where an input dataset is converted into a different set of key/value pairs, or tuples;
- The "Reduce" task, where several of the outputs of the "Map" task are combined to form a reduced set of tuples (hence the name).

4. Hadoop

Hadoop is an open source software project that enables the distributed processing of large data sets across clusters of commodity servers. It is designed to scale up from a single server to thousands of machines, with a very high degree of fault tolerance. [7]

5. Hive

Hive is a "SQL-like" bridge that allows conventional BI applications to run queries against a Hadoop cluster. It was developed originally by Facebook, but has been made open source for some time now, and it's a higher-level abstraction of the Hadoop framework that allows anyone to make queries against data stored in a Hadoop cluster just as if they were manipulating a conventional data store.[8]

6. Pig

PIG is tool that tries to bring Hadoop closer to the realities of developers and business users, similar to Hive. Unlike Hive, however, PIG consists of a "Perl-like"(Latin) language that allows for query execution over data stored on a Hadoop cluster, instead of a "SQL-like" language. PIG was developed by Yahoo!, and, just like Hive, has also been made fully open source.

7. Wibidata

WibiData is a combination of web analytics with Hadoop, being built on top of HBase, which is itself a database layer on top of Hadoop. It allows web sites to better explore and work with their user data, enabling real-time responses to user behaviour, such as serving personalized content, recommendations and decisions.

8. Platfora

Platfora's software works with the open-source software framework Apache Hadoop, when a user queries a database, the product delivers answers in real time via a graphical user interface. Bloomberg BusinessWeek called it "Big Data for Dummies.PLATFORA is a platform that turns user's queries into Hadoop jobs automatically, thus creating an abstraction layer that anyone can exploit to simplify and organize datasets stored in Hadoop.

9. Storage technologies

As the data volumes grow, so does the need for efficient and effective storage techniques. The main evolutions in this space are related to data compression and storage virtualization.

10. Sky tree

Sky Tree is a high-performance machine learning and data analytics platform focused specifically on handling Big Data. Machine learning, in turn, is an essential part of Big Data, since the massive data volumes make manual

exploration, or even conventional automated exploration methods unfeasible or too expensive.[9]

V.APPLICATIONS

The term 'Big Data' is a massive buzzword at the moment and to show how big data is used today to add real value.

1.*Understanding and Targeting Customers:*
Big data is used to better understand customers and their behaviors and preferences. For example the US retailer will easily predict their customer expectation. Then incase of Telecom service companies can predict customer churn and the car insurance companies can better understand that how well their customer actually drive. Even government election campaigns can be optimized using big data analytics.
2.Understanding and Optimizing Business Processes:
Big data is also increasingly used to optimize business processes. Retailers are able to optimize their stock based on predictions generated from social media data, web search trends and weather forecasts.
3.Personal Quantification and Performance Optimization:
Big data is not just for companies and governments but also for all of us individually. We can now benefit from the data generated from wearable devices such as smart watches or smart bracelets.
4.Improving Healthcare and Public Health
The computing power of big data analytics enables us to decode entire DNA strings in minutes and will allow us to find new cures and better understand and predict disease patterns. By recording and analyzing every heart beat and breathing pattern of every baby, the unit was able to develop algorithms that can now predict infections 24 hours before any physical symptoms appear.
5. Improving Sports Performance:
Most elite sports have now embraced big data analytics. The IBM Slam Tracker tool is used for tennis tournaments; and video analytics that track the performance of every player in a football or baseball game, and sensor technology in sports equipment such as basket balls or golf clubs allows us to get feedback via smart phones and cloud servers and also provides guidelines about how to improve it.
6. Improving Science and Research:
CERN, the Swiss nuclear physics lab with its Large Hadron Collider, the world's largest and most powerful particle accelerator. Experiments to unlock the secrets of our universe – how it started and works - generate huge amounts of data. The CERN data center has 65,000 processors to analyze its 30 petabytes of data.
7. Optimizing Machine and Device Performance:
Big data analytics help machines and devices become smarter and more autonomous. For example, big data tools are used to operate Google's self-driving car. The Toyota Prius is fitted with cameras, GPS as well as powerful computers and sensors to safely drive on the road without the intervention of human beings.
8. Security Improvement and Law Enforcement:
Big data is applied heavily in improving security and enabling law enforcement. For example In U.S, National Security Agency (NSA) uses big data analytics to foil terrorist plots. Big data techniques are also used to detect and prevent cyber-attacks. Police forces use big data tools to catch criminals and even predict criminal activity and credit card companies use big data use it to detect fraudulent transactions.
9. Improvement of Optimizing Cities and Countries:
Big data allows cities to optimize traffic flows based on real time traffic information as well as social media and weather data.
10. Trades in financial sector:
High-Frequency Trading (HFT) is an area where big data finds a lot of use today. Here, big data algorithms are used to make trading decisions.[6]

VI. CONCLUSION

As a conclusion of this study, we analyzed that how the big data becomes a key factor in our day to day life. And we have seen that how this Big data made an high Impact on the recent technology. Big data must support the different sectors like search, development, governance and analytics services for all data types—from transaction and application data to machine and sensor data to social, image and geospatial data, and more.

REFERENCE:

[1]Dean and S.Ghemawat,"Mapreduce: Simplifieddata processing on large clusters "in Proceedings of (OSDI'04, 2004, pp. 137–150).
[2] http://www.sas.com/en_us/insights/Big-data.
[3] "Apache hadoop,"http://hadoop.apache.org/.
[4]"IBM What is big data?- Bringing big data to the Enterprise" www.ibm.com.Retrived 2013-08-26.
[5] "Core Techniques and Technologies for Advancing Big Data Science and Engineering (Big data),"Program Solicitation NSF 12-499).
[6] "Manufacturing: Big Data Benefits and Challenges" TCS Big Data Study .Mumbai, India: (Tata consultancy service limited .Retrieved 2014-06-03
[7] "IBM What is big data? — Bringing big data to the Enterprise". www.ibm.com.(*Retrieved 2013-08-26)*
[8] http://www.techrepublic.com/blog/big-data-analytics/10-emerging-technologies-for-big-data/.
[9]Michael Minelli, Michelle Chambers, and Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses", (*Wiley, 2013).*
[10]G. Mackey, S. Sehrish, J. Bent, J. Lopez, S. Habib, and J. Wang, "Introducing Map-reduce to High End Computing", in Petascale Data Storage Workshop", 2008. Pdsw, 08. 3rd, 2008, pp. 1-6.
[11]C. Lynch, "Big Data: How do your data grow?" Nature, Vol. 455, No. 7209, pp. 28-29, 2008.
[12]White, Tom (10 May 2012). Hadoop: The Definitive Guide. O'Reilly Media. p. 3. ISBN 978-1-4493-3877-0.