ISSN: 2393-8528

IJICSE
Open Access

Contents lists available at www.ijicse.in

International Journal of Innovative Computer Science & Engineering

Volume 1 Issue 2; Page No. 28-34

A Novel Approach towards Big Data Challenges

Rachana Sharma[1], Priyanka Sharma[2]

[1]Department of Computer Science and Engineering, Jayoti Vidhyapeeth Women's University, Jaipur, Rajasthan, India

rachana706@gmail.com

[2]Department of Computer Science and Engineering, Jayoti Vidhyapeeth Women's University, Jaipur, Rajasthan, India

piyusharma2812@gmail.com

ARTICLE INFO

Received 09 Oct. 2014
Accepted 20 Nov. 2014

Corresponding Author:

Rachana Sharma

1 Department of Computer Science and Engineering, Jayoti Vidhyapeeth Women's University, Jaipur, Rajasthan, India

Keyword: BigData, Hadoop, MapReduce, Big Data Analytics.

ABSTRACT

Big Data is a term for explosion of high quantity and diversity of high frequency digital data. In this era of "digital universe" we encounter data sets with high volume, variety, velocity and complexity. Data, information and knowledge are touching exponential powers of exabytes and zettabytes. The data generated is that immense that it cannot be efficiently processed by traditional database methods, tools and current technologies. The source of Big Data is sensor networks, call logs, retail transactions, user generated content that is producing structured and unstructured data. The problem does not only comprise these aspects, heterogeneity, privacy, data aggregation and transporting are also huge contributors. So today, all we require is to center on these challenges and revamping our approach towards the methodologies for Big Data representation, analysis and design. We require superior storage and management architectures that are fast, fault tolerant, flexible and scalable for analyzing data efficiently and extracting relevant data for decision-making. The focus of this paper is on giving a holistic view of Big Data, its challenges, how present technologies are dealing with these challenges and what is more to be explored as a solution to Big Data. Also to look over technologies like Hadoop, MapReduce, BigQuery and Apache Sparks.

©2014, IJICSE, All Right Reserved.

1. INTRODUCTION

Big Data has become an obscure for traditional computing infrastructure. According to what IDC estimated, in 2011 growth of data was 1.8 zettabyte, in 2012 it was 2.72 zettabyte and is now getting double every two years and will reach 8 ZB in 2015 [10]. Petabyte, exabyte, zettabyte and now how we are mounting will lead us to yottabyte in 10 more years. In this flood, data sets are getting larger in magnitude, arriving faster and getting diverse through machine generated data such as sensors, meters, RFID, GPS, computers, smart phones, digital TV, bank cards and data generated by humans such as documents, health records, social media contents including images videos, emails. Thus, most of the data we are tackling in this world is unstructured. The tweets we do, the blogs we write are structured, while images and videos are only structured to display, but are in unstructured from for semantic content and search [3]. This transformation of this to structured format is another confront. Further, to optimize value in Big Data we need to link data sets with each other, this data integration is much than the traditional mining approaches and when done properly can help us in decision-making. Relational database cannot scale to process such volcano of data. For this accelerating growth rate researchers and practitioners require overhauling the underlying algorithms for scalability and applying advance analytic techniques on it. So how Big Data and Analytics are teaming up to together create one of the most reflective trends in business intelligence [2]. For fetching value out of data requires transparency of data by making is available, supporting experimental analysis, supporting Real-time analysis applied to data sets based on customers and embedded sensors. As compared to the volume and complexity of data there is scarcity of tools and trained personnel for handling it.

This paper is organized as follows: The following section presents the importance of Big Data and also focuses on some of its characteristics. The issues faced by Big Data are given in Section 3. Then Section 4 represents the System design challenges of Big Data. Then handling of Big Data by the big Organizations is discussed in Section 5. Section 6 represents the Apache Hadoop architecture for Big Data solution. Section 7 gives a quick review of the flaws in MapReduce and how other technologies are taking over it and the final conclusion of this study is given in Section 8.

I. THE IMPORTANCE OF BIG DATA

The potential annual value of US health care is $300 billion more than the total annual health care spending in Spain. $600 billion potential annual consumer surplus from using personal location data globally [1], 1.5 million more data-savvy managers needed to take full advantage of Big Data in US, The departments of Defense and Energy and the Advance Research Projects Agency announced a joint R& D initiative in march2012 that will invest more than $200 to develop new tools and technologies for Big Data [1].

A. Characteristics of Big Data

- **Volume** The most visible aspect of data, data volume the amount of data available. The higher the volume, harder it is to find value. Human interpretation using traditional business intelligence and analytic environments does not scale to these new volumes, for example the social networking site are themselves producing data in order of terabytes and finding value in this amount is difficult.
- **Velocity** refers to the production rate, streaming and aggregation of data. Users not only want data today, but they want it as soon as possible. From real-time data, making decisions and maintaining a strategy in business is helpful. ECommerce is an example of increased data speed and richness of data used for business transactions.
- **Variety** refers to the various sources and forms of data being represented like image, text, audio, and video. Unstructured data, inconsistent semantics complicate the data analysis, mining, and storage. In analyzing this heterogeneous data from various resources creates a problem for data acquisition management phase.
- **Value** refers to what can be come out of the data which can be useful in making decisions and it can help the user to better results. It differentiates the demands of business leaders and IT professionals because for business leaders adding value to their profit is much important, unlike IT professional that have concerns with technologies of storage and processing
- **Complexity** refers to data quality in terms of correctness, noise, conflicts in the data sets interdependence of data. Data visualization will only prove to be a valuable tool if the quality is assured.

II. BIG DATA ISSUES

A. Storage and Transport Issues

As social media being a large part of this explosion we do not require any personal storage problem, cloud computing provides an on demand, pay as you go storage capacity. The amount of Amazon Web Storage S3 has jumped from 262 billion objects in 2012 to over 1 trillion objects at the end of first of 2012.But moving data in and out from cloud uses current communication network, for example, if we transfer 1 Exabyte of data through a network of 1 gigabyte per second will take about 2800 hours concluding data transfer is a greater problem than storing it and processing it [1]. This problem compels us to process data “in place” and then transferring only results.

A. Management Issues

The most difficult problem in Big Data is management is efficient organization, administration and governance of data sets. We need to follow protocols for ensuring data accuracy and validity. Finding outliers is a major task of the data qualification. The graphical representations of data through visualization tool can communicate trends and outliers faster than tables containing numbers. Validating each data set is impractical deed, thus new approaches for data qualification and validations are required. Big Data is a heterogeneous mix of both structured (traditional datasets-in rows and columns like DBMS tables, CSV’s and XLS;s) an unstructured data like email attachments, rich graphics like video and audio, PDF documents, manuals, medical records, ECG and MRI images, contacts, forms and other documents. 80% of enterprise data is unstructured, so managing it is a huge concern [3]. For query processing, there are various platforms available, for processing structured data advance SQL is widely used, for unstructured data there are many NoSQL platforms used providing dynamic flexibility based on the type of data modeling we require such as BigTable, HBase and Cassandra etc.

B. Processing Issues

For processing Big Data, timeliness is an important point to be considered. We require parallel processing and new analytic algorithms for resulting actionable information. Parallel processing techniques that now directly apply for intra-node parallelism, which means hardware resources such as processor caches and processors memory channels are shared across a single node. There are many situations where the result of the analysis is required immediately. For example, if a fraudulent credit card transaction is suspected, it should ideally be detected before the transaction is completed for potentially preventing the transaction from taking place at all. Obviously, it is not practical to analyze user’s real purchase in advance. Rather, partial results can be developed in advance so that a small

amount of incremental computation with new data can be used to arrive at a quick determination [3].

III. BIG DATA SYSTEM DESIGN CHALLENGES

Designing a system for Big Data requires both understanding the need of users and technologies for solving problems arising with increase of scale and development of new analytics. Different Big Data requires a different design solution. For example, Business data is analyzed for many purposes, a company may perform system log analytics and social media analytics for risk assessment, customer retention, and brand management and so on to maintain their growth rate [3]. Traditional dimensional modeling and online Analytical processing (OLAP) has failed to access such large amount of structured and unstructured data. Another consideration for designing is the fact that sequential access to memory is much faster than random access thus we require a rethinking of how we design storage subsystem for data processing systems. How much quantity is requisite for extracting quality information? Deciding data relevance, reliability, accuracy and data adequacy to make predictions is also a challenge. So the dare here is not to build a system which is ideal for processing all tasks. Indeed, the need for underlying system architecture is to be flexible enough that the components built on top of it for expressing the various kinds of processing tasks with different workloads [3].

A. Compliance and security issues

Privacy has been a huge concern in context of increasing amount of data, for domains such as social media and heath records, data accumulated about a particular human is very important for analyzing trends but sometimes they can violate one's privacy. International data corporation (IDC) coined the term "digital shadow" which is a picture formed by analyzing, aggregating an individual's data [1]. Clearly, some Big Data must be secured with respect to privacy and security laws and regulations and finalizing these rules and regulation is a bigger task. As there is no better storage solution for Big Data than Cloud, which is itself struggling for its security concerns and downtime of cloud is another concern. Cloud providers have to find a way to combine security and performance, by putting appropriate policies and procedures, enforcing more focus on information security to make a best of both worlds. Technologies like Apache Accumulo or .20.20x version of Hadoop or above proves to be better than other technologies in fighting with these challenges, further technologies like Cloudera Sentry or DataStax enterprise enhance application layer security. In the NoSQL database role based access control can be supported by Accumlo and Sentry. Apache Oozie helps in consistently maintaining, monitoring and analyzing audit logs.

IV. HOW ORGANISATIONS ARE HANDLING BIG DATA

Yahoo is one of the biggest users for Hadoop and companies like Facebook, Twitter, LinkedIn and many more uses Hadoop as a solution for Big Data. For example, EBay, which is an ecommerce company, uses 532 Hadoop clusters used for search optimization and research. HPCC (high performance computer cluster) System, by LexisNexis Risk is an open source platform and is an alternative to Hadoop used by several leading banks, insurance companies, federal government and law enforcing agencies [9]. Google itself uses its Big Query which is based on Dremal which is a data analysis tool built by Google itself [4]. There are also many software solutions available which can be used by small to medium size enterprise for Example Amazon web services, Microsoft Azure, 1010data for complex business analysis. Taking an example An IBM solution platform for Big Data is IBM InfoSphere which is an ETL (extract-transform-load) tool for data integration warehousing, management and information governance for information-intensive projects, providing high performance, scalability, reliability for Big Data solution.

V. BIG DATA TECHNOLOGIES

A. Hadoop

Hadoop is an open source, Linux based Virtual Grid operating system architecture licensed by Apache for data storage and processing. It runs on commodity hardware (low cost numerous computers), it uses HDFS (Hadoop Distributed File System) which is fault-tolerant high-bandwidth clustered storage architecture and runs MapReduce for distributed data processing and can structured and unstructured data. There also exists many Apache projects started under the umbrella of Hadoop and are used to give assistance to Hadoop system management. The architecture of Hadoop consists of Software Stack is HDFS (Hadoop Distributed File system) at the bottom layer, which provide a distributed storage in which each file appears as a (very large) contiguous and randomly addressable sequence of bytes. Middle layer Hadoop architecture is Hadoop MapReduce System, which provides batch processing for computations. . At the top, the HBase store is available as a key-value layer in Hadoop stack, for applications needing basic key based record management operations. Many years of Hadoop stack prefer the use of declarative language over the MapReduce programming model, Hadoop project tools include Pig (for creating MapReduce programs in a language called Pig Latin which is similar to SQL) and Hive (a data warehouse architecture which provide data summarization, query and analysis) for this. Oozie, Scoop, Mahout, Flume are other project tools used by Hadoop to add a certain value to its core functionalities.

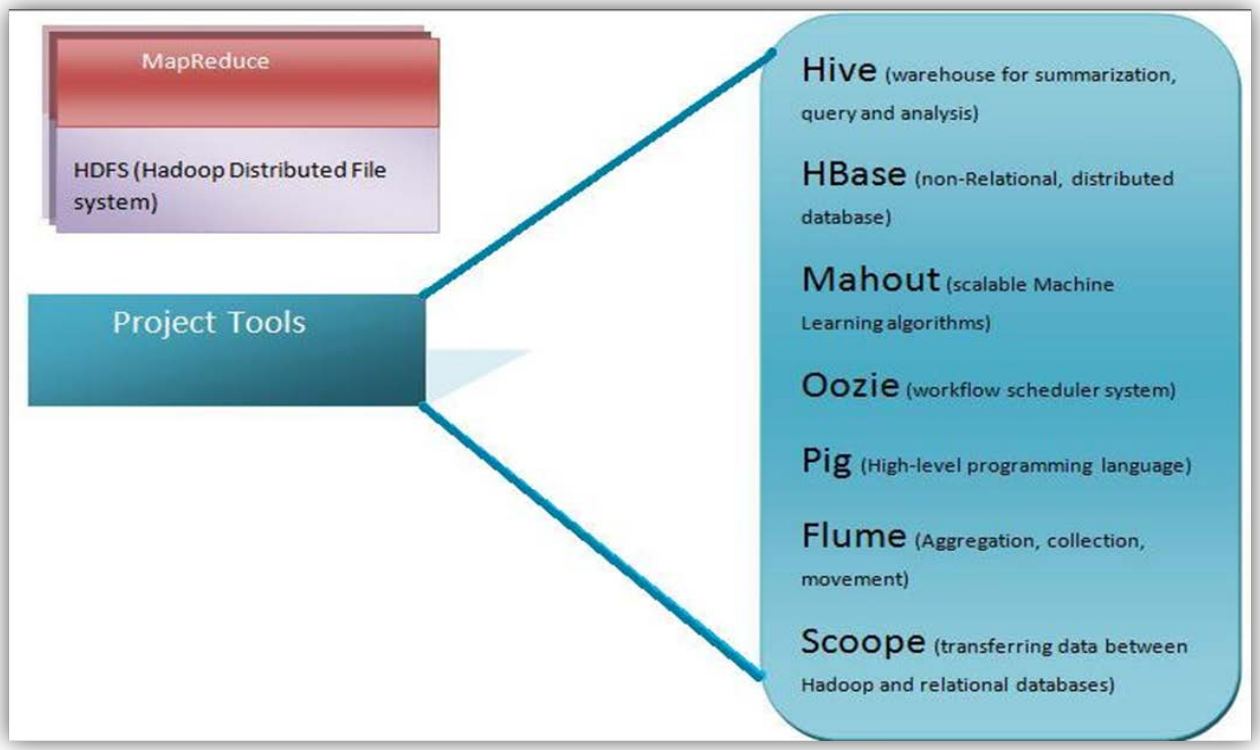

Figure 1: Hadoop System Architecture

1) MapReduce:

MapReduce programming model and a software framework which enables proceeding large data collections by writing two user functions, map and reduce, that the map() function takes an input key/value pair and produces a list of intermediate key/value pairs, the MapReduce runtime system groups together all intermediate pairs based on the intermediate keys and passes them to reduce() function for producing final results. MapReduce was first used by Google to process computations on its GFS(Google file system) and then after it was used by developers named Doug cutting and Michael Camarilla to create Hadoop.

Map (inky, in value) --->list(out key, intermediate value).

Reduce (out_key, list (intermediate_value)) -- ->list (out_value).

The signatures of Map () and Reduce () are as follows:

Map (k1,v1)! List (k2, v2) and Reduce (k2, list (v2))! List (v2)

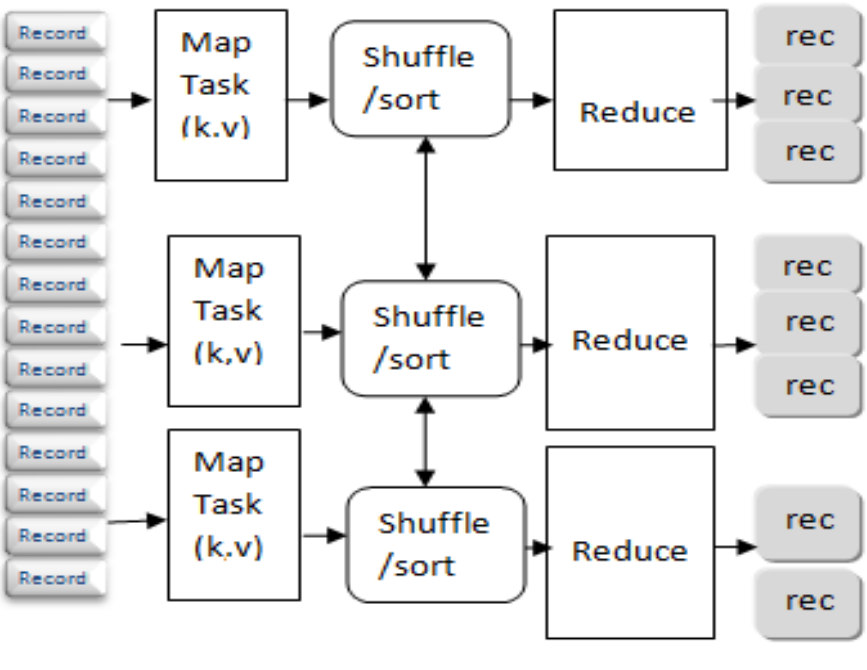

Figure 2: MapReduce framework

A MapReduce cluster employs a master-slave architecture where one master node manages a number of slave nodes schedule jobs to them, monitor them and execute failed tasks. In the Hadoop, the master node is called JobTracker and the slave node is called TaskTracker. For a map and reduce task, master stores the state (idle, in-progress, or completed) of TaskTrackers. MapReduce works by first splitting the input dataset into even-sized independent data blocks. Each data block is then scheduled to one TaskTracker node and it parses key/value pairs out of the input data and passes each pair to the user designed map function. Map tasks of all the TaskTrackers run in parallel. When the map () functions complete, the runtime system groups all intermediate pairs and launches a set of reduce tasks to produce the final results. Task Tracker and JobTracker comes under MapReduce framework while Data node and name nodes come under HDFS. Data node manages the piece of data given to a particular node. A name node in master keeps the index to which data is kept by which node.

Map and reduce functions are implemented through interfaces called Mapper and Reducer.

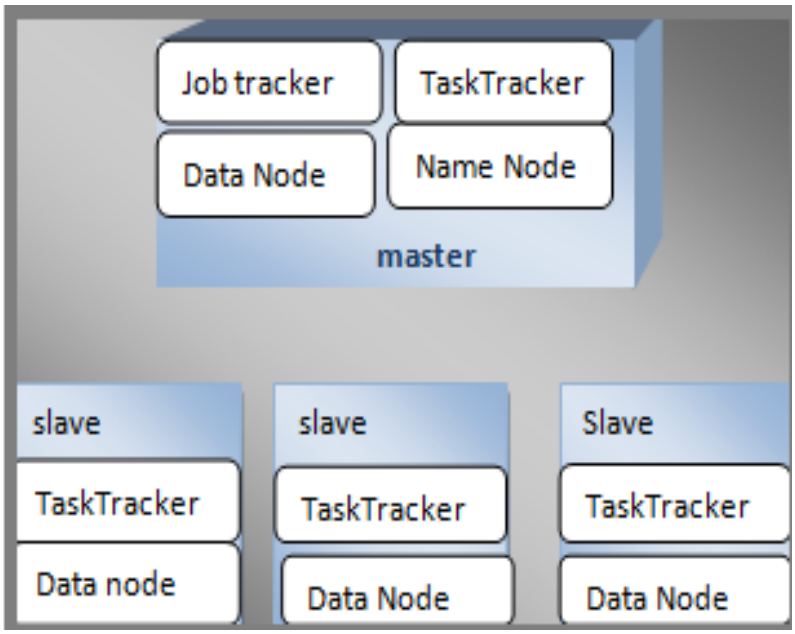

Figure 4: Hadoop concept

- **Mapper**: Mapper performs map task of organizing the input data so that it can be easily processed by reduce task. The input key/value pairs are transformed to intermediate key/value pairs. The number of maps is determined by the total size of input, i.e. the total no. of blocks of the input files.
- **Reducer**: The intermediate records which share the same key are reduced to a smaller set of value by Reducer.
- **Shuffle:** The output from Mapper requires arrangement before they are partitioned to the Reducer. Input to the Reducer is sorted output of Mappers.
- **Sort**: Sorting groups the output of Mapper on the bases of intermediate keys. The shuffle and sort occur simultaneously; while map outputs are being fetched they are merged.
- **Secondary Sort:** A comparator is used for the determination of the order in which key/values are fed to the reduce function. Secondary sort is used by programmer to control this order.
- **Reduce:** Each reduce function process its input pairs in parallel manner, there is one to one mapping between keys and Reducers. Increasing the no. of Reducers makes the system fault tolerant, but increases the framework overhead.
- **Partitioner:** partitioning of key to Reducer is done by Partitioner. A hash function is used to determine the partition by using keys or the subset of key.
- **Reporter**: A reporter is used to report the progress of the application, the amount of time the framework using for processing key/value pairs. Counters are also updated using reporter.
- **Output collector:** collects final and intermediate outputs from Mapper and Reducer.

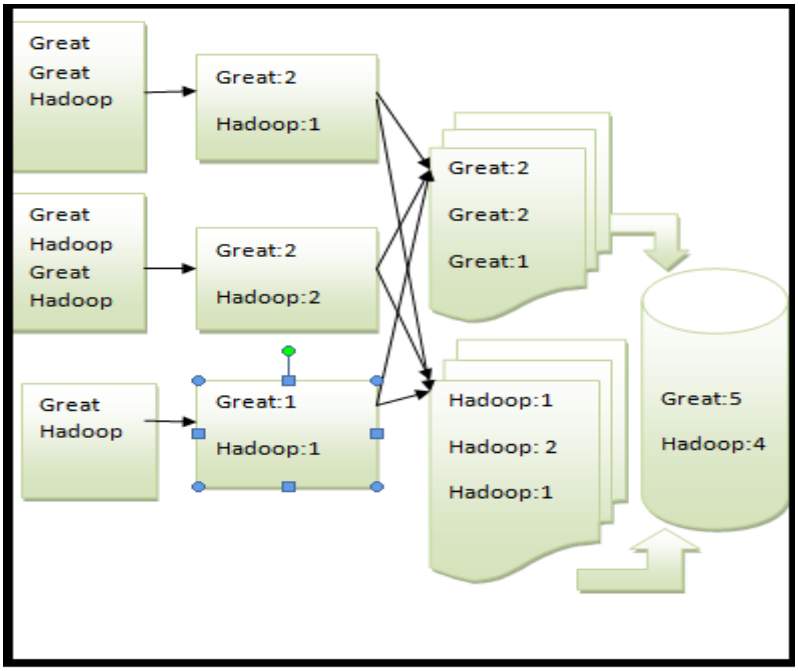

Figure: 3 Example of MapReduce.

VI. MAPREDUCE GAINS AND SHORTCOMINGS OVER OTHER TECHNOLOGIES

Hadoop proves to be fault tolerant to processing and hardware failure by maintaining three copies of each file which are scattered across computers and is highly scalable by adding as much number of slave computers as required. Yahoo! Reported that their Hadoop gear could scale out of more than 4000 nodes in 2008. MapReduce framework involves distributed data computation which prevents network overloading due to local computation of data. It is very simple to use and best suited to batch process huge amount of data, which require large join operations. Task allotted in MapReduce are independent of each other so handling node failure is easy, it is much more flexible than the traditional database system as it is not dependent on data model or schema and is also independent of storage used that is, we can use Google's Big table or any other storage as a storage solution. MapReduce can be used for applications such as data mining where you need to apply complex statistical computation or data mining algorithms to a chunk of text or binary data. What lacks in MapReduce is that it's not suitable for Adhoc and trial-and-error data analysis. The turnaround time is too slow and doesn't allow programmers to perform iterative or one-shot analysis [4]. MapReduce faces lacks in the use of high level language in its framework like SQL in DBMS and also in query optimization technique. MapReduce cannot be used in streaming data, online transactions and also for real time processing executing a complex data mining on Big Data which requires multiple iterations and paths of data processing with programmed algorithms. Google's *Big Query* which is a fully-managed and cloud based interactive query service for massive data sets, in comparison to MapReduce is much faster, its columnar storage and tree architecture makes it to achieve higher compression ratio and quick aggregation of results respectively. It supports interactive query of large data set for quick analysis and troubleshooting and BigQuery is suitable for OLAP (Online Analytical Processing) or BI (Business Intelligence) [4]. But, it can only efficiently process structured data and only used for read only data not for updating it and one cannot program complex logic with it which gain a point for using MapReduce. *Apache Sparks* is a powerful open source processing engine for Hadoop data and runs applications faster than MapReduce especially for machine learning and the interactive analysis, it lets you write applications using Java, Sacla and Python. In addition to simple map and reduce functions Sparks supports SQL queries, stream processing, and complex analysis such as machine learning and graph algorithms. But MapReduce outperform sparks for ETL type computations where results sets are large and for the computations where requiring communication between the computing entities result set may need to be exchanged in intermediate steps for example fluid dynamic and some graph algorithms falls in this category. With the latest release of Hadoop 2.0 has made it faster and efficient due to use of YARN (Yet another Resource Negotiator) as a resource monitor to Hadoop.

VII. CONCLUSION:

Big Data is the new business and social science frontier. Hadoop has proven to be a full solution for Big Data up to some point, The MapReduce programming model has been successfully used at Google for different purposes and is used widely in a distributed pattern based searching, distributed sorting, machine

learning, document clustering, web-link-graph reversal etc. It will be erroneous to judge MapReduce as restrictive programming model because Pig provides a light weighted scripting languages for manipulating dataset by specifying data transformation (filtering, joining, grouping, etc) which is converted to MapReduce job by pig execution engine and Hive allows to issue query against large relational datasets stored in HDFS and HiveQL can be compiled down to Hadoop jobs by hive query engine providing user a comfort of relational databases. But MapReduce efficiency, I/O cost still needs to be addressed for successful implication. It is much slower to struggle with other technologies. We need to focus on the optimization of Mapper/Reducer parallel execution of nodes and reducing the amount of memory needed for execution of MapReduce task. Stream processing is very attractive for working with time-series data (news feeds, tweets, sensor reading, etc.), which is difficult in MapReduce due to its batch oriented design.

VIII. REFERENCES:

1. Stephen Kaisler, Frank Armour, J. Alberto Espinosa and William Money, "Big Data: Issues and Challenges Moving Forward", 2013 46th Hawaii International Conference on System Sciences.
2. Puneet Singh Duggal and Sanchita Paul "Big Data Analysis: Challenges and Solutions", International Conference on Cloud, Big Data and Trust 2013, Nov 13-15, RGPV.
3. "Challenges and Opportunities with Big Data", a community white paper developed by leading researchers across the United States.
4. "An Inside Look at Google BigQuery", White paper by Google.
5. "2013 Big Data Survey Research Brief", White paper by SAS.
6. Kyong-Ha Lee Yoon-Joon Lee, Hyunsik Choi Yon Dohn Chung, Bongki Moon," Parallel Data Processing with MapReduce: A Survey", SIGMOD Record, December 2011.
7. Jeffrey Dean and Sanjay Ghemawat," MapReduce: Simplified Data Processing on Large Clusters", to appear in OSDI 2004.
8. "Big Data: A New World of Opportunities", NESSI White Paper, December 2012.
9. Anthony M. Middleton, Ph.D. LexisNexis Risk Solutions, "HPCC Systems: Introduction to HPCC (High-Performance Computing Cluster)", white paper LexisNexis Risk Solutions, Date: May 24, 2011.
10. Dan Vesset, Benjamin Woo, Henry D. Morris, Richard L. Villars, Gard Little, Jean S. Bozman, Lucinda Borovick, Carl W. Olofson, Susan Feldman, Steve Conway, Matthew Eastwood and Natalya Yezhkova, "Worldwide Big Data Technology and Service 2012-20145 Forcast".