Survey Of Technologies, Tools, Concepts And Issues In Big Data

Chitrakant Banchhor, N.Srinivasu

Abstract— This paper surveys tools and techniques used in the Big-data and Big-data computing. The principles upon which Big-data and data-intensive computations work have been explored with some outlines provided by various researchers in the field. The paper starts with the traditional cluster computing and further explores about the Hadoop system as a tool to solve Big-data issues. It also covers the significant growth in cloud computing towards hosting MapReduce to use for Data-intensive computations as one of the services available through clouds. It has brief coverage about Microsoft Azure and Amazon clouds for MapReduce services to be provided through Internet.

Index Terms-- Big data, Hadoop, MapReduce, data-intensive computing, high-performance computing.

1 INTRODUCTION

The first three science fields, namely empirical science, theoretical science and computational science have already showed their importance in terms of research gates for various innovations. Along with these three another science fields is emerging as the fourth scientific paradigm termed as Data-intensive science. The first paradigm emerged in the beginning stage of science discovery thousand years ago. In this first paradigm scientists described the natural phenomenon only based upon human empirical evidences and therefore it is called as empirical science [1]. The second paradigm named as theoretical science emerged hundred years ago, and most of the science laws described in this paradigm such as Newton's laws of motion. Scientists had found most of the principle requires using scientific simulations because of the complexities associated with the theoretical analysis. The third paradigm is emerged as computational branch of studies because of the simulations involved in most of the discovery process[2]. The current situation experiences very large amounts of generated data in various applications and systems. These data are used to explore further growth in the business and market situations through scientific analysis and studies. The technologies and techniques used in this kind of analysis are totally different from previous three paradigms. This is termed as Data-intensive science because of huge amounts of data involved in the computations. Therefore, Data-intensive science is viewed as a new fourth science paradigm for scientific discovery. This paper is a survey of various studies and researches have been done and going on in the field of Data-intensive computations by various researchers. In this paper the survey starts from traditional cluster computing systems and Big Data technologies and tools. The open source communities Hadoop system is explored which includes high-performance computing as well as Hadoop and its tools included in cloud computing environment also. This paper is organized as follows. Section-II provides basics require for today's Hadoop cluster through detail treatment on traditional cluster computing systems. It includes architecture and computations performed through message passing approach. This covers a brief description based on an example of distributed matrix multiplication on cluster computing environment. Section-III outlines Data-intensive computing with brief description about some techniques associated with it. Section-IV is devoted to Big-data and computing. This covers basic concepts associated in Big-data computing with challenges and tools used to deal with the Big-data and its issues. Section-V of the paper is based on Hadoop an open source system which includes various tools and techniques for dealing with the Big-data. The section starts with the overview of Hadoop and later goes into details about the components of the Hadoop system such as HDFS and MapReduce. This covers working of the Hadoop system with HDFS and MapReduce working approaches with their technical descriptions. Section-VI is the brief description about the association of Hadoop system in the field of High-performance computing. Finally the Section-VII of the paper deals about Hadoop and cloud computing including different cloud technologies with their features comparisons. It also focuses on MapReduce services through cloud computing platforms, and two such platforms have been outlined: Azure MapReduce and Amazon Elastic mapReduce. The paper concludes with some outlines about the Big data computing and tools and technologies used in the field.

- *Chitrakant Banchhor is currently pursuing PhD in Computer Science & Engg. in Dept. of Computer Science & Engg., Koneru Lakshmaiah Education Foundation, Vaddeswaram, A.P. India, E-mail: cobanchhor@gmail.com*
- *Dr. N.Srinivasu, is currently working as professor in Computer Science & Engg. in Dept. of Computer Science & Engg., Koneru Lakshmaiah Education Foundation, Vaddeswaram, A.P. India*

2 CLUSTER COMPUTING

A group of computers configured using networking and used to solve problems by means of parallel processing is typically termed as cluster computing. Cluster computing can be described as a fusion of the fields of parallel, high-performance, distributed, and high-availability computing [3].

2.1 Architecture of Cluster Computing systems

The typical architecture of the cluster computing system shown in the figure below includes various components with their interactions and tasks accomplishments together [3]. The figure – 1 depicts a typical cluster system consists of various hardware and software components which include independent computers including PCs, workstations, servers etc., and other communication components for interactions among nodes of the system. The software side includes network operating systems, message passing interface and other tools for application developments.

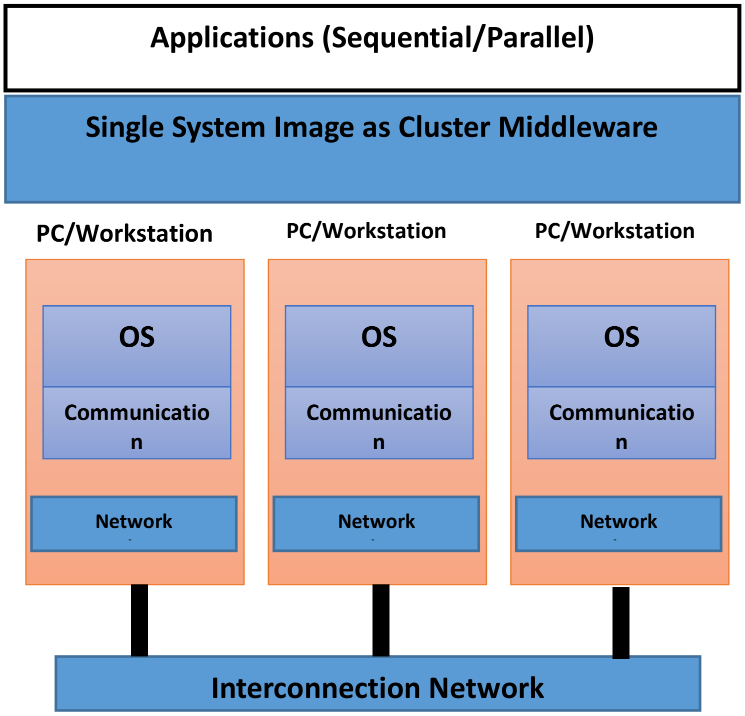

Fig.1. Cluster Architecture [3]

Master/Slave Architecture for cluster computing systems cluster computing is typically based on master/slave arrangement of included nodes in the system. One machine is designated as a master and it acts as an overall controller of all the nodes in the system. Master machine's main tasks include allocation of jobs to various slave nodes and get inputs from the user typically by means of input files. Slave nodes are also called as compute nodes because these are responsible for computations only; this means all computations are performed as slave nodes sides.

Communication technologies among nodes various cluster nodes need to communicate with each other require fast networking devices to provide high-bandwidth and low latency in communication between nodes. The parameters influence the selection of interconnection networking devices are compatibility with the operating system and hardware included in the cluster as well as price and performance. The bandwidth and latency are the main metrics for performance measurements of the interconnecting devices. The amount of data can be transmitted over the network in a fixed period of time whereas latency is referred as the time required for transmission of data from source node to destination node.

Single System Image (SSI) The Single System Image ability is provided to hide distributed nature of the components included in the distributed system. These are termed as transparency issues in the distributed systems; one such transparency is about hiding different locations of components included in the system. Cluster computing system acts as a distributed system because it does provide transparencies expected in the distributed system, we can consider location transparency as an example, here it is achieved by hiding various nodes to be considered as a single concrete system. SSI can be achieved by various mechanisms implemented at various levels like hardware, operating systems, middleware, and applications.

2.2 Programming model for cluster computing systems
The popular programming approach for the cluster system created using message passing libraries such as PVM and MPI is master/slave based distributed programming paradigm. In this section the overview of parallel programming is covered thereafter the programming for cluster computing system is described in with practical considerations.

Parallel and Distributed programming Parallel programming approach is actually different from the sequential programming. Writing parallel program increases complexity in terms of number of instructions executed simultaneously on many processors. It also provides great level of synchronization and communication issues among the modules running simultaneously in different processors. Parallel programming approach can be divided into two types:

– **Function level parallelism**
  The overall program is divided into number of modules or functions where each function performs its task individually. During executions each module is loaded into its own machine and all modules execute simultaneously. In theory it looks simple to write such programs but it is actually complicated and provides technological limitations in terms of finding independent steps in each module to be executed.
– **Data parallelism**
  Data parallelism style of programming is most suitable for scientific computations and suits distributed computing environments like cluster systems. This approach deals with the division of overall data among number of chunks and gives these into different processors for processing simultaneously. After computations results of all processed chucks will be combined and produced the final outcome of the program.

Master/slave programming style for parallel and distributed systems The overall programming task is divided into two different programs this means instead of writing a single program one has to write two programs one is termed as master and another is a slave. The following table outlines the steps performed by the master and slave programs.

Table 1. Steps of Master and Slave

Master	Slave
Accept input from the user	
Start executions of slave programs in required slave machines	
Divide input data into number of chunks based upon number of slaves started	Wait for the data for computations
Distribute data chunks to slaves	Get data chunk from the master
Wait for intermediate results from slaves	Performs computations of data chunk
	Send intermediate result to master
Combine all intermediate results and produce the final output	

– **Example**

Consider distributed matrix multiplication for cluster computing environment. Assume we have a cluster computing system comprises of four machines. In this arrangement three machines are given exclusive responsibilities to acts as slaves whereas one machine acts as a master as well as involve in computations in terms of slave also. Also assume two matrices A and B of size 100X100 for multiplications. The whole matrix can be divided into four chunks with 25 elements in each because four slave nodes are available for computations. Matrix A is divided into four chunks as a1[25][25], a2[25][25], a3[25][25] and a4[25][25] where a1 contains first 25 elements, a2 contains second 25 elements, a3 contains third part of 25 elements and a4 contains last 25 elements. Similarly matrix B is divided into four chunks of 25 elements as b1, b2, b3 and b4. Here one has to write the master and slave programs. In this example the master is responsible to accept whole matrices A and B and divide these into chunks. The multiplication steps are written in the slave program. The executable copies of these programs are installed in all the nodes before starting the task. In this arrangement three copies of slave program are installed in three individual slave nodes and master node will contain master program and it also has one copy of executable slave program. The user starts execution of the master program in master node and master program indirectly starts executions of all four slave programs in four nodes. Once all the slaves have been started, master distributes the chunks of matrices A and B to all the slaves. The slave programs in different nodes perform the multiplications of different sets of data simultaneously and produces there part results back to the master. Now master is responsible to combine all these intermediate results and finally produces the output of the multiplication.

Role of message passing systems in cluster computing
We have covered the example of distributed programming with practical description about placements of various modules and their executions. Two popular message passing libraries PVM and MPI are used in clustered systems. The message passing library provides APIs for communications among nodes in the systems and also provides mechanisms to transfer the data from one node to another. In matrix multiplication program master uses API to start executions of slaves in different machines in a similar fashion it uses API to transfer data from itself to slave nodes after dividing the whole matrix into chunks.

Cluster computing system architectures The cluster system architecture is classified in two broad categories [4]:

– **High Performance Computing**

The jobs to be performed in High Performance Computing (HPC) clusters are of various sizes, types and most importantly require much more CPU power and performance. The design and implementation goals of HPC cluster are to provide the computing environments for the compute intensive problems, where multiple modules can execute simultaneously in multiple nodes and reduces their turnaround times [5]. A HPC cluster is realized to include hardware and software parts, and the components collectively contribute to solve compute intensive problems.

– **High Availability Computing**

The cluster system configured as High availability (HA) clusters are used to provide uninterrupted services to the end users for a particular service with its instances running in many nodes of the cluster. This way it is able to provide fault tolerance also in a sense that if one node fails it can be served by another instance from other compute nodes [5].

Operating Systems for Cluster Computing As a general principle any network operating system can be used to manage or work with the cluster computing systems. Linux along with other scientific software is most widely used operating system in High Performance Computing. Linux operating system features such as compatibility with different architectures and networks suggest it as an operating system for the choice of HPC clusters. It has been experienced by many users and researchers that Linux is more stable and scalable than other platforms. Open source community developers contribute significantly in terms of new and modified software according to the current need makes Linux as an operating system of anybody's choice to use in cluster computing [5].

3 Data Intensive Computing

Compute-intensive fields deal with the allocations of applications times for executions and computational requirements with small volume of data, whereas data intensive computing applications deal with large volumes of data and their executions. Data intensive computations require to spend their large amount of time for I/O processing and manipulation of data. In a nutshell one can say about data intensive computing is the field where computing applications are I/O bound. The editorial column titled Special issue on Data Intensive Computing in ScienceDirect [6] mentioned about the enormous growth on data volumes, it has been predicted till 2011 the digital data would reach nearly **1.8 zetta – bytes** where **zetta = 10^{21}**. In recent developments in processor technology (faster CPUs, GPUs, etc) as well as other computing and electronic devices are the prime components for explosion of data because large groups of people involved in the computing field by some means such as social networking sites and many other related fields. The scientific communities also contribute in growth of large volumes of datasets from applications which include modeling and simulation, sensor data and other high throughput instruments.

3.1 Phases of Data-intensive computing

Data-intensive computing is mainly deal with the processing of very large amount of data and get useful information from it which is based on some analytical principles. The applications in Data intensive computing uses filtering process to be applied on large data and get highly valuable information from the data. The overall processes of Data intensive computing is divided into different steps such as capturing, managing, accessing, analyzing, and understanding vast amounts of data.

3.2 Data-intensive applications

The application in the data intensive field is created for processing and analysis of large amounts of data for further use in the society and commercial fields. Data intensive application development focuses on processing of large data and analyzes it by means of some analytical tools for knowledge discovery rather than simulating physical process.

Applications use various tools dynamically and determine which data sets are suitable for processing; such tools are called as information discovery interfaces. The applications are mainly I/O bound and therefore, its largest fraction of execution time is devoted to data movements. The computer's memory acts as a cache during data movement to reduce the disk bandwidth requirements. However, applications are typically developed to work in network environment, one can obviously questions it's suitably; because its bandwidth requirements in network environments will be much higher than local disk access [7].

3.3 Distributed processing and caching

In distributed processing of data intensive applications, the application is moved for executions towards the data residence and data movement is minimized to avoid the bandwidth requirements. Distributed caching deals with the movement of data towards High Performance Computing sites where applications already installed and waiting for the availability of the data. The uses of these two techniques depends upon some parameters such as network latency, network protocol overhead, computational and network bandwidth and the amount of data to be accessed.

4 BIG DATA AND BIG DATA COMPUTING

The huge amount of electronic data generated by various sources such as social networking sites, sensor data etc., and actually difficult to store for further processing and getting meaningful information from it is termed as big data [8]. Though any very large collection of data which is difficult to store and process by traditional RDBMS is considered as Big Data, but following are the types of the data involve [9]:

Traditional enterprise data: data gathered from CRM (Customer Relationship Management) systems, ERP data obtained from transactions, Web store transactions, and general ledger data.

Machine-generated/sensor data: includes Call Detail Records (CDR), weblogs, smart meters, sensors, logs and trading systems data.

Social data: includes data from micro-blogging sites like Twitter, and social media platforms like Facebook, etc.

4.1 3Vs in Big Data

To define Big Data various researchers used different terms from 3Vs to 4Vs to characterize Big Data from data growth to its processing. Doug Laney used **volume**, **velocity** and **variety**, known as 3Vs [10][38]. For the definition of the Big Data, there are various different explanations from 3Vs to 4Vs. Doug Laney used volume, velocity and variety, known as 3Vs.

Volume Big Data features have its size attribute to be considered as one of the characteristics [10] in Big Data termed as its volume. The size of data plays significant role in storage requirements of Big data field. The volume of the data becomes very large because of the applications generating data dynamically in less amount of time. For example, a single jet engine can produce 10 terabytes of data in half an hour of flying time.

Velocity Typical velocity associated with Big Data is how quickly data have arrived and been stored in an enterprise [10]. The velocity of Big Data factor is used to describe the unprecedented growth of data which will be stored and processed in real time basis. The immediate consequences of the growth of such a large amount of data require the tools to process data and extract information from it must have the ability to handle it dynamically. The infrastructure should have the facility to provide high performance computing for processing.

Variety Variety represents all types of data from traditional data used in RDBMS to unstructured and raw data include images, videos or anything in the context. The variety of Big Data introduces other challenges in processing during different phases such as merging and fusion of different types and formats of data [10][30].

4.2 Big Data Challenges

In one side such a huge amount of data plays vital role for organizations' growth through analytical planning and strategy making for enhancement of customer satisfaction. Another side is difficulties in handling and managing Big Data. The typical steps require for Big Data problem are data capture, data storage, processing and analysis [11]. Each of these steps requires high end systems in terms of hardware and software to provide the solutions. The biggest challenge is inherent to the computer architectures. The Big Data technology is associated with data intensive computing and CPU speed increases every time but I/O activities are not coping with the CPU power. The challenges arise in Big Data processing are outlined in the following with steps comprising of capture, storage, transmission, analysis and visualization.

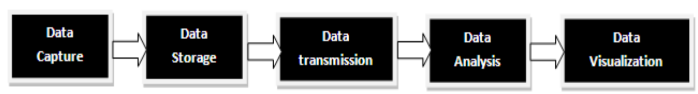

Fig. 2. Big-Data challenges [11]

Data capture and storage Data grows rapidly because of the data considered to be included in Big Data field is about unstructured and semi-structured data generated from various types of devices and web sites [12]. There are 2.5 quintillion bytes of data created every day, and data increases exponentially day by day, the gathering of such a huge data require very high end technologies [13]. The traditional processes of data capture and storage have been changed for Big Data gathering because of the size and formats of the data. The storage requirements for Big Data are much more, even no one had predicted in recent past. Big Data deals with heavily on I/O bound instead of compute bound, the dynamic data capture mechanisms and its storage techniques have been taken care by the researchers and market analyzers. Different storage technologies have come into picture to fulfill the storage requirements of Big Data such as solid-state drive (SSD) [14] and phase-change memory (PCM) [15]. These storage technologies are partially functioning and are still in development stages and can be used in near future as Big Data storage mediums. HDDs are still used as a permanent storage for almost all types of data storage facilities. HDDs are being replaced by other technologies such as SSD because HDDs have slower random I/O performance than sequential I/O performance [14][15]. The systems for Big Data processing are large-scale distributed systems, and commonly used enterprise storage architectures such as Direct-attached storage (DAS), network-attached storage (NAS), and storage area network (SAN) have drawbacks and limitations with such systems. The data access platforms are used in commercial applications for further development. Various such platforms are CASTOR, dCache, GPFS etc. have been employed in Big Data filed to provide satisfaction in storage requirements [16].

Data transmission In distributed systems and cloud storage, network bandwidth capacity shows significant bottlenecks because the volume of data communication is very large. Data transmission mainly deals about the security issues provided while transmitting the data. Researchers have proposed new algorithms to provide the security and other integrity related issues about data in cloud computing fields [17][18].

Data Analysis Big Data analysis faces biggest challenge in terms of scalability because of the volume of the data. Volume of the data increases so frequently and therefore researchers have paid more attentions to accelerate analysis algorithms to handle such a huge volume of data according to the enhancements of CPU speed. Real-time Big data applications such as social networks, finance, biomedicine, intelligent transport systems, internet of things where everything should happen in timely basis. It is a real challenge to provide the response in time basis especially when the volume of data is huge. It is still a big challenge for stream processing involved by Big Data. Big data has changed the hardware development according to the requirements of such voluminous data, the way software has been developed for its analysis has also changed the development directions.

Data Visualization Data visualization deals with the representation of knowledge effectively by using different graphs. The knowledge hidden in the large scale and unstructured data sets are extracted, and can be conveyed in the better way if it is represented in terms of proper functionality and aesthetic form. Various Big Data Visualization tools are used by organizations to get instant review and feedbacks from their customers to improve their service for better marketplace. Online marketplace eBay uses a tool named as Tableau [11] for Big Data visualization, which has the capability to transform large datasets to intuitive pictures. Tableau produces such results which can be handled interactively by eBay employees to get current customers feedback and improve their quality of service in real time basis also.

4.3 Big Data Techniques

In Big Data processing specialized techniques are required to process large amounts of data efficiently within limited run times. For market analysis different organizations use techniques to provide higher competitiveness in goods pricing strategies and marketing purposes. Marketing great Wal-Mart applies machine learning and statistical techniques for getting pattern from their large volumes of transactional data [11]. Multiple disciplines such as statistics, data mining, machine learning, neural networks, social network analysis, signal processing, pattern recognition, optimization methods and visualization approaches are included in Big Data techniques. Many specific techniques of these disciplines involved to provide the overall techniques in Big Data analysis and visualizations.

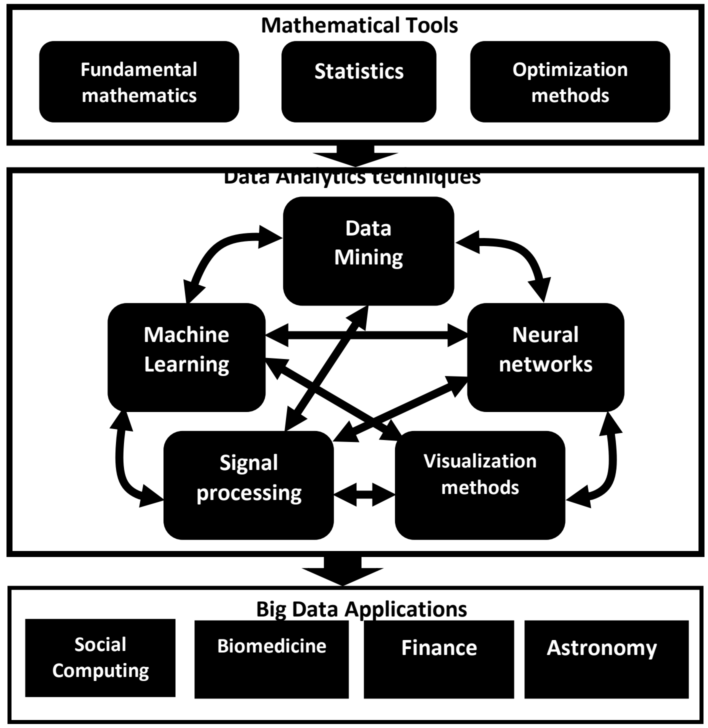

Fig. 3. Big-Data Tools and Techniques [11]

4.4 Big Data tools

Big data contains lots of hidden information; its processing gives knowledge about the things can be extracted from it. Actually various tools are required to make sense from Big data extraction. Current Big Data tools are divided into three classes, namely, batch processing tools, stream processing tools, and interactive analysis tools. Mahout and Dryad are batch processing tools based on Hadoop infrastructure; actually most of the batch processing tools are Hadoop based. Various tools are available for real-time analytic for stream data applications such as Storm and S4. In an interactive environment, data processed in interactive fashion where user is directly connected to the computer and interact to the data in real-time. Users can reviewed, compare and analyze data in tabular and graphic formats in such environments [11].

5 BIG DATA USING HADOOP

5.1 Apache Hadoop

Hadoop is a flexible and freely available framework for developing distributed applications to process large amount of data termed as Big Data in cluster computing environments. It is an open source implementation of Google's proprietary framework called as MapReduce. Hadoop is written in java and runs on large clusters of commodity machines or on services provided through cloud computing such as EC2. Hadoop system is scalable in the sense that new nodes can be added at any time to process large amounts of data. Hadoop is designed to handle failures automatically because basically it is created for commodity hardware systems [19]. Hadoop allows storing petabytes of data on tens of thousands of nodes of servers. Hadoop has scalability benefits to add many nodes dynamically when required without affecting the resettlements of existing setups [20].

5.2 Core components of Hadoop system

Hadoop is a leading Big Data analytics engine that is designed to extracts meaningful information from the large datasets on large cluster computing platforms. It has many components to provide services effectively, but its core components are Hadoop Distributed file system (HDFS) for storage and Hadoop mapreduce for distributed computations [21].

HDFS HDFS is a block based file system and works at the top of multiple nodes in a cluster to store user data in files. HDFS design is so simple from users' perspective that most of the traditional file system commands (create, rename, remove, delete) work in the similar manner. Other advanced file system operations like setting links (hard, soft), seek operations (to particular blocks) and overwrite files are not supported by HDFS because of the purpose for which it has been actually designed and implemented. The mount operation used in other file systems to attach one file system into another does not work in HDFS because it requires programmatic access. HDFS communications are based on the top of TCP/IP protocol. Working of HDFS is described in D part of this section in more details.

Hadoop MapReduce MapReduce is the distributed computing framework in Hadoop system. It provides APIs for writing applications to process large amounts of data in the cluster computing environment. Part E of this section describes Hadoop MapReduce working in detail.

5.3 Hadoop Architecture

Physical architecture Hadoop is design and implemented for any kind of machines and systems. It works on the cluster computing systems where one machine is designated as master and rest of the machines are called as slaves. As we have detail discussion on cluster computing in section-II where we considered about the system where master slave arrangement of machines are used for compute- incentive tasks. Hadoop is a software framework for such type of cluster system which provides various components to handle the tasks efficiently for the user in fewer amounts of times. Hadoop provides the facility to store petabytes of data on tens of thousands of machines and at the same time it shows the performance for suitable adoptability in today's computing environments to process the data in real-time for the online users [20][22].

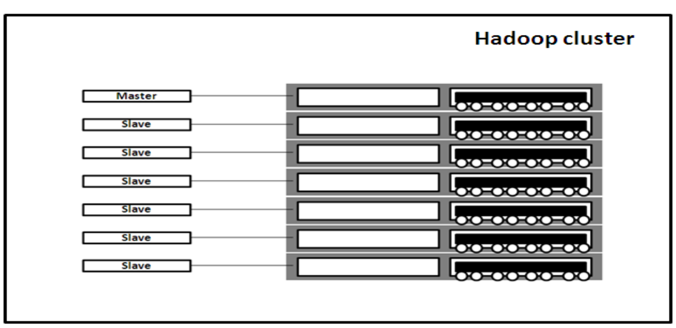

Fig. 4. Hadoop Cluster Architecture

Logical architecture In its top view, the hadoop software system is in master-slave architecture. Hadoop system consists of the two core components HDFS for storage and MapReduce for computations. As we have already discussed the master/slave architecture in terms of machine arrangements, the Hadoop provides software components for each machine to be managed in a distributed environment. It has master/slave software components of storage and computations. Hadoop storage HDFS consists of basic master/slave components as NameNode and DataNode, similary MapReduce computation also has master/slave basic components termed as JobTracker and TaskTracker [23][24].

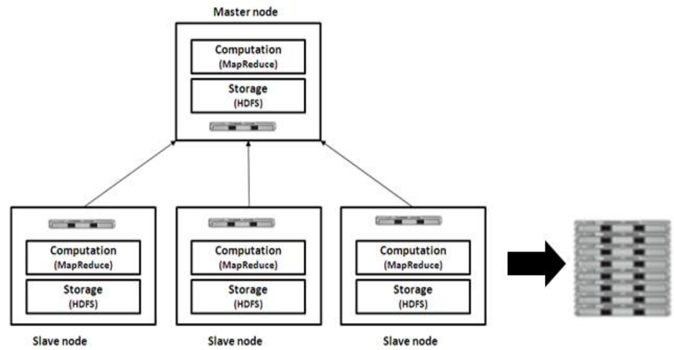

Fig. 5. High-level Hadoop architecture [24]

5.4 HDFS working

The following fig shows logically how components (NameNode and DataNode) of HDFS interact with each other to provide services to the user in terms of user applications.

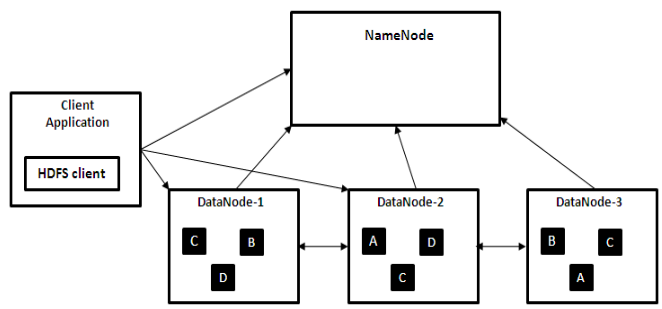

Fig. 6. HDFS architecture shows an HDFS client communicating with the master NameNode and slave DataNodes [24]

HDFS DataNodes are components runs in slave machines to provide the storage for the file system. The overall storage is distributed at different DataNode sides whereas NameNode maintains records about all the nodes. NameNode maintains information about the overall data maintained and being processed by the system. When any user application intends to read data from the HDFS, requests first goes to the Namenode because it maintains metadata about the entire file system in hadoop. NameNode verifies the validity and locations of the data in different dataNodes and thereafter it returns it to the application for reading the contents. Similar steps are performed while writing the file also in this case n NameNode decides the destination DataNodes and returns information to the application and accordingly updates information in its repository to maintain the metadata up- to-date. In fact all DataNodes send block reports containing information about different blocks presents to the NameNode after certain intervals, this happens periodically so that NameNode is able to maintain current details about the data in the system [24].

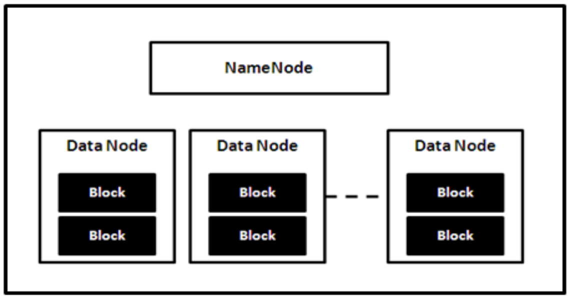

Fig. 7. HDFS blocks in DataNodes [20]

HDFS is block oriented distributed file system which divides large files into number of blocks typically of size 64 MB or 128 MB. These blocks are replicated into three or more servers. HDFS provided APIs are used by MapReduce applications to read and write data in parallel [20].

5.5. MapReduce working

Hadoop MapReduce is also designed in the similar fashion like HDFS; it has also the same model as master-slave architecture. The following figure shows logical architecture of MapReduce with its main components included in it.

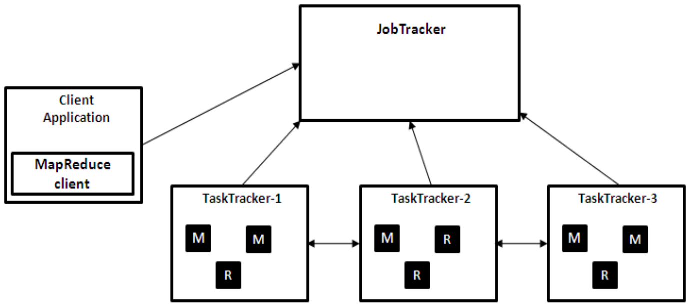

Fig. 8. MapReduce logical architecture [24]

The Hadoop MapReduce consists of two core components named as JobTracker and TaskTracker.The master/slave architecture of the MapReduce allows one master and many slaves in the system. In MapReduce framework single JobTracker process acts as a master and runs in master machine, whereas there are many TaskTrackers processes run in all slave machines. The JobTracker is responsible to coordinate the activities across all TaskTracker processes. JobTracker accepts job requests from the client applications and schedules various map and reduce tasks on TaskTrackers to perform the actual work [24].

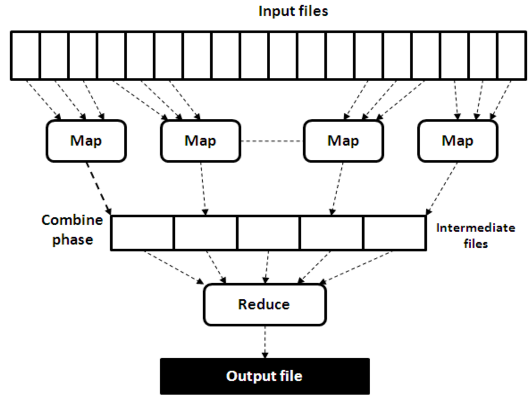

Fig. 9. Map Reduce Data Flow [23]

MapReduce programming framework provides working environments for various map functions to work in parallel fashion in different nodes of the cluster. The outputs of all the map operations are collected and combined using combine function and then another operation called as reduce is performed on intermediate results. The output(s) of the reduce operation is the output of the system and stored in the persistent storage [23]. Programmers use MapReduce framework to solve data-parallel problems. In Hadoop MapReduce the overall data termed as dataset is divided into small subparts and processed independently in distributed computing environment [20].

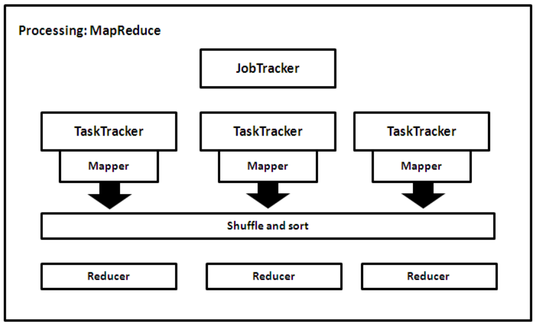

Fig. 10. Map Reduce processing [20]

MapReduce processing works on input splits; these are created after splitting the input dataset into multiple chunks. Each chunk is assigned to map task and process the data parallel. Input splits are read as sets of (key, value) pairs by each map task and apply the instructions defined in its scope. Each map processes its part of data and produces a transformed set of (key, value) pairs as the output. The other operations such as shuffles and sorting of map tasks outputs are performed by the MapReduce framework and it is considered as intermediate results. These intermediate results are send to the reduce task for further processing. MapReduce framework services JobTracker and TaskTracker are used to scheduling and monitoring of tasks in the system [23].

5.6. MapReduce programming model

The MapReduce processing is divided into Map and Reduce steps. Map phase takes key/value pair as Input and it generates key/value pairs as intermediate outputs. In the Reduce phase of operation all values associated with the same keys are merged and finally it generates the output.

Real world tasks such as search operations can be expressed in this model. Following example map and reduce functions are used to count the words in the documents [25].

```
Map(String key, String value) {

    // key : document name
    // value : document contents

  for each word W in value {
      EmitIntermediate(W, 1);
  }
}
```

Fig. 11. Map function for counting

One document is considered for processing at a time in Map function and the count is emitted, just '1' in this simple example. In the reduce function the count of each word is sums together and emitted. These two phases can be expressed in the following manner.

map: (k1, v1) → list(k2, v2)
reduce: (k2, list(v2)) → list(v2)

5.7. Understanding the MapReduce Job Life Cycle

This part covers brief description about the life cycle of a MapReduce job also the focus is given the roles of the actors involved in the life cycle. In almost all types of cluster computing systems, commonly we consider one master and many slave nodes. In Hadoop cluster configuration, a single JobTracker process runs in a single master node whereas many TaskTrackers run in multiple slave nodes. Following steps are typically performed when a MapReduce job is submitted into Hadoop [26]:

1. The local job client submits a job to the JobTracker.
2. The JobTracker now does scheduling for the submitted job and distribute the map work amongst all TaskTrackers for parallel processing.
3. In turn each TaskTracker spawns a Map task and inform the progress to the JobTracker.
4. Now when JobTracker knows the availability of results of map operations, it distribute the reduce work amongst the TaskTrackers for parallel processing.
5. Each TaskTracker spawns a Reduce task to perform the work and inform its progress to JobTracker.

6 HIGH PERFORMANCE COMPUTING AND HADOOP

High-performance computing (HPC) is the platform to provide computing environments for application programs require to be executed efficiently, reliably and quickly. HPC uses parallel processing to run highly compute intensive tasks in less time [27]. Highly performance oriented systems are used to solve advanced computations problems. Mainly HPC uses cluster computing systems to provide computation infrastructures for solving computing problems. While solving the problems, applications along with the data moved towards available computation resources for executions [28]. The powerful infrastructures used for HPC are highly efficient to perform compute intensive data movements. The current trend in HPC is showing performance of computation in teraflops and much more because of availability of highly powerful systems [27].

6.1. High Performance Computing and Big Data

The basic comparison of HPC and Big data solution reveals that the HPC is used for the problems that require a lot of CPU power. For the solutions of Big Data problems the storage capacity in terms of RAM and HDD must be very high. Big Data problems also require the framework where everything is available in terms of computations and file systems. Hadoop and Google mapreduce frameworks have successfully used by commercial organizations for their compute intensive and storage activities. In addition Big data solutions based much more on disk IO performance with CPU power [29].

6.2. High Performance Computing with Hadoop

Linux cluster systems were extensively used as the platform for high performance computing and alternative to supercomputers. High performance computing uses various specialized software packages for services such as job control and scheduling. Packages like SGE and Torque are commonly used for the task to be performed as a Queue system. A framework is required to coordinate a parallel program in to provide HPC solutions on the cluster computing environments. Typically message passing libraries are used for communications among the computing nodes; MPI and PVM are two popular frameworks for this task. Hadoop with its own distributed file system (HDFS) and distributed computing framework (MapReduce) provides the similar kind of solutions for High performance and scientific computations [30].

7. CLOUD COMPUTING AND HADOOP

Internet-based computing to share computing resources, software and other information to computers and other devices on demand is called as cloud computing [27]. A technical definition [31] is "a computing capability that provides an abstraction between the computing resource and its underlying technical architecture (e.g., servers, storage, networks), enabling convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction." According to this definition clouds essential characteristics are on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service.

7.1. Cloud services modes

The U.S. National Institute of Standards and Technology Mell and Grance, have mentioned most popular types of cloud services are Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS) [32].

SaaS Various software applications and services can be accessed from providers to clients on demand basis. Such applications are from different areas include email client, text editors and online storage.

PaaS Cloud computing also provides complete environment for user applications executions, for example user can deploy .NET applications onto Windows Azure cloud instances. (Windows Azure website, 2012)

IaaS Infrastructural resources are provided through cloud computing in terms of virtual machines. Once the infrastructural resources are available, consumers can install and run any software which includes operating systems and applications. Amazon Elastic Compute Cloud (Amazon EC2 website, 2012) is one of the well-known IaaS vendors.

7.2. Cloud Technologies

Cloud technologies have been around us for different types of services to be provided on demand. Many popular cloud technologies include Google MapReduce, Hadoop, Microsoft Dryad, and CGL-MapReduce is being used by various organizations as well as other consumers for the solution of their tasks [27]. Google MapReduce, Hadoop and Dryad technologies have created different trends in parallel programming field. Cloud technologies have provided the facilities due to which the computations can be moved to large sets of data and execute the tasks where datasets are available, this concept eliminate the bottlenecks in network bandwidth during data movement. The comparison of features [33] supported by three popular cloud technologies are shown in table 2.

7.3. Cloud based services for MapReduce computing

Amazon Web Services (AWS) AWS is a cloud computing services available on-demand offered by Amazon. AWS offers services for computing, storage and communications. AWS services are specifically named as Elastic Compute Cloud (EC2), Elastic MapReduce (EMR), Simple Storage Service (S3) and Simple Queue Service (SQS) [34]. The EC2 provides Xen based virtual machines instances through Internet for user activities. Once users fulfill the initial steps, EC2 dynamically provides resizable virtual cluster for computations. EC2 directly offers infrastructure as a service through cloud computing platform [35].

Amazon Elastic Map Reduce (EMR) Amazon EMR is used to analyze and process vast amounts of data through distributing the computational work across a cluster of virtual servers running in the Amazon cloud [36]. This cluster is managed by the Hadoop framework. In fact EMR is a hosted Apache Hadoop MapReduce framework utilizing Amazon EC2 for computing power and Amazon S3 for data storage [35].

Microsoft Azure Platform Microsoft Azure platform also offers a set of cloud computing services similar to Amazon Web Services. Windows virtual machine instances are available for user's computational tasks through Windows Azure platform [37]. Microsoft Azure provides .net runtime as the platform as a service for deploying users programs as an Azure deployment package through web application.

Azure MapReduce Windows Azure has a distributed decentralized MapReduce runtime called as Azure MapReduce. It is developed using Azure cloud infrastructure services [35]. Availability issues have been taken care in AzureMapReduce by retrying and the system is designed in such a way that it does not rely on the immediate availability of data to all the worker tasks.

7.4. High Performance Computing With Clouds

Batch processing and analytics jobs analyzes very large amount of data (terabytes) and take usually hours to finish executions. The applications to be executed must have sufficient scopes for data parallelism for executions in parallel into different nodes of the system in short time periods. Google MapReduce and Hadoop provide programming environments for the programmer to write the code which can execute in parallel across hundreds of cloud computing servers. This hides the internal complexities associated with allocation of resources to program modules. Some commercial HPC applications for instance Server Labs, Pathwork Diagnostics, Cycle computing and Atbrox have been deployed by researchers and enterprises into clouds [27]. These applications show benefits in the cloud environment and provide scope for further studies in the HPC field.

Table 2. Feature Comparisons of three Cloud Technologies

Feature	Hadoop	Dryad & DryadLINQ	CGL-MapReduce
Programming model	MapReduce	DAG based execution flows	MapReduce with Combine phase
Data handling	HDFS	Shared directories/ Local disks	Shared file systems/ local disks
Intermediate data communication	HDFS/ Point-to-point via HTTP	Files/TCP pipes/ Shared memory FIFO	Content Distribution Network
Scheduling	Data locality/ Rack aware	Data locality/ Network topology based run time graph optimizations	Data locality
Failure handling	Persistence via HDFS Re-execution of Map and Reduce tasks	Re-execution of vertices	Currently not implemented (Re-executing map tasks, redundant reduce tasks)
Language support	Implemented using Java Other languages are supported via Hadoop Streaming	Programmable via C# DryadLINQ provides LINQ programming API for Dryad	Implemented using Java Other languages are supported via Java wrappers

8. CONCLUSION

The computing and its technical growth have been entered in the field and era of Big-data. It has large number of opportunities to deal with, and this seems to be the starting of new revolution for innovation. Fortunately we have tools available with us to analyze our needs and accordingly can explore for further growth in the recent developments. This survey paper gives the brief overview of Big-data problems, including tools and techniques dealing with the Big-data computing. It has also the coverage of High-performance computing and cloud computing included with MapReduce access through services and platform available. These technological descriptions can suggests for further developments in terms of tools enhancements and applications developments. It has been clearly observed that most of the Big-data technology and tools are still in their initial stages but we can have confidence to get new solutions and beneficial growth in the field in near future.

ACKNOWLEDGMENT

We would like to acknowledge our thanks to all directly or indirectly helped us in dealing with the things which came across while doing the work. We also thank to the Koneru Lakshmaiah Education Foundation officials and staff members for their cooperation and suggestions in finding out the directions in right time.

REFERENCES

[1] Gordon Bell, Tony Hey, Alex Szalay, Beyond the data deluge, Science 323 (5919) (2009) pp 1297–1298. www.cloudinnovation.com.au/Bell_Hey%20_Szalay_Science_March_2009.pdf

[2] Tony Hey, Stewart Tansley, Kristin Tolle, The fourth paradigm: data-intensive scientific discovery, Microsoft Research2009.http://research.microsoft.com/enus/collaboration/fourthparadigm/4th_paradigm_book_complete_lr.pdf

[3] R. Buyya, "*High Performance Cluster Computing: Architectures and System"s*, *vol. 1*, Prentice Hall, 1999.

[4] Semra AYDIN et al, "Building a high performance computing clusters to use in computing course applications", ScienceDirect, January 4, 2009.

[5] Morton, D., "High Performance Linux Clusters, *Linux Journal"*, Volume 2007, Issue 163.

[6] The editorial column titled Special issue on Data Intensive Computing in ScienceDirect, Surendra Byna, NEC Labs America Inc., 4 Independence Way, Suite 200, Princeton, NJ 08540, United States doi:10.1016/j.jpdc.2010.10.009

[7] Reagan W. Moore, "Digital Libraries and Data Intensive Computing", San Diego Supercomputer Center, http://dret.net/lectures/ppos-spring11/reading/CDLC-Moore.pdf

[8] Oracle Company, "Oracle Information Architecture: An Architect's Guide to Big Data", August 2012. http://www.oracle.com/technetwork/topics/entarch/articles/oea- big-data-guide-1522052.pdf

[9] Oracle Company, "Oracle: Big Data for the Enterprise", January 2012 http://www.oracle.com/us/products/database/big-data-for-enterprise-519135.pdf

[10] Doug Laney, 3d Data managment: controlling data volume, velocity and variety, Appl. Delivery Strategies Meta Group (949) (2001).

[11] C.L. Philip Chen, C.-Y. Zhang, Data-intensive applications, challenges, techniques and technologies: A survey on Big Data,Inform.Sci.(2014),http://dx.doi.org/10.1016/j.ins.2014.01.015

[12] Martin Hilbert, Priscila Lopez, The world's technological capacity to store, communicate, and compute information, Science 332 (6025) 2011 pp 60–65

[13] Simone Ferlin Oliveira et al, "Trends in computation, communication and storage and the consequences for data-intensive science, in: IEEE 14th International Conference on High Performance Computing and Communications, 2012

[14] Lee Hutchinson, "Solid-state revolution: in-depth on how ssds really work", Ars Technica 2012

[15] A. Pirovano et al, "Scaling analysis of phase-change memory technology", IEEE Int. Electron Dev. Meeting (2003) 29.6.1–29.6.4.

[16] David Leong, "A new revolution in enterprise storage architecture", IEEE Potentials 28 (6) (2009) pp 32–33.

[17] Alina Oprea et al, "Space efficient block storage integrity", in: Proc. 12th Ann. Network and Distributed System Security Symp. 1478 (NDSS 05), 2005.

[18] Qian Wang et al, "Enabling public auditability and data dynamics for storage security in cloud computing", IEEE Trans. Parallel Distrib. Syst. 22 (5) (2011) pp 847–859.

[19] Chuck Lam, "Hadoop in action", Manning Publications Co, 2011

[20] Intel White Paper on Big Data Analytics, "Extract, Transform, and Load Big Data with Apache Hadoop", https://software.intel.com/sites/default/files/article/402274/etl-big-data-with-hadoop.pdf

[21] White Paper, EMC Corporation, "HADOOP ON EMC ISILON SCALE-OUT NAS", December 2012 http://www.emc.com/collateral/software/white-papers/h10528- wp-hadoop-on-isilon.pdf

[22] Joey Jablonski, " Introduction to Hadoop", A Dell Technical White Paper, 2011 Dell Inc http://i.dell.com/sites/content/business/solutions/whitepapers/en/Documents/hadoop-introduction.pdf

[23] Grant Mackey et al, "Introducing Map-Reduce to High End Computing"http://www.pdsiscidac.org/events/PDSW08/resources/papers/mackeyMR_HEC.pdf

[24] Alex Holmes, "Hadoop in practice", Manning Publications Co, 2012

[25] AMIT GOYAL, "A Survey on Cloud Computing", in:University of British Columbia Technical Report for CS, 508 (2009)

[26] MarkLogic Connector for Hadoop Developer's Guide, MarkLogic 7—November, 2013, pp 28-31

[27] Ye Xiaotao1 et al, "Research of High Performance Computing with Clouds", ISBN 978-952-5726-10-7, Proceedings of the Third International Symposium on Computer Science and Computational Technology (ISCSCT '10) Jiaozuo, P. R. China, 14-15,August 2010, pp. 289-293

[28] WIKIPEDIA,"High-performance computing", http://en.wikipedia.org/wiki/High- performance_computing

[29] http://research.microsoft.com/en-us/projects/azure/high-perf-computing-on-Windows-azure.pdf

[30] http://www.splunk.com/web_assets/pdfs/secure/Discover_Hadoop.pdf

[31] Cloud Computing Definition, National Institute of Standards and Technology, Version 15, http://csrc.nist.gov/groups/SNS/cloud-computing/index.html

[32] Ling Chen et al, "Introducing Cloud Computing Topics in Curricula", Journal of Information Systems Education, Vol. 23(3) Fall 2012, pp 315-324

[33] Jaliya Ekanayake and Geoffrey Fox, "High Performance Parallel Computing with Clouds and Cloud Technologies", 1st International Conference on Cloud Computing, Oct 19-21, 2009

[34] Amazon Web Services. http://aws.amazon.com

[35] Thilina Gunarathne et al, "MapReduce in the Clouds for Science", CLOUDCOM '10 Proceedings of the 2010 IEEE Second International Conference on Cloud Computing Technology and Science, pp 565-572 http://cgl.soic.indiana.edu/publications/CloudCom2010-

MapReduceintheClouds.pdf
[36] http://docs.aws.amazon.com/ElasticMapReduce/latest/DeveloperGuide/emr-what-is-emr.html
[37] Windows Azure Platform, Retrieved April 20, 2010, from Microsoft: http://www.microsoft.com/windowsazure/
[38] Zaiying Liu et al, “A Sketch of Big Data Technologies”, 2013 Seventh International Conference on Internet Computing for Engineering and Science, 2013 IEEE, DOI 10.1109/ICICSE.2013.13, pp 26 – 29