High Performance and Fault Tolerant Distributed File System for Big Data Storage and Processing using Hadoop

E.Sivaraman
Ph.D Research Scholar
Research Department of Computer Science
NGM College (Autonomous), Pollachi
Coimbatore, India
sivaraman.eswaran@gmail.com

Dr.R.Manickachezian
Associate Professor
Department of Computer Science
NGM College (Autonomous), Pollachi
Coimbatore, India
chezian_r@yahoo.co.in

***Abstract*—Hadoop is a quickly budding ecosystem of components based on Google's MapReduce algorithm and file system work for implementing MapReduce algorithms in a scalable fashion and distributed on commodity hardware. Hadoop enables users to store and process large volumes of data and analyse it in ways not previously possible with SQL-based approaches or less scalable solutions. Remarkable improvements in conventional compute and storage resources help make Hadoop clusters feasible for most organizations. This paper begins with the discussion of Big Data evolution and the future of Big Data based on Gartner's Hype Cycle. We have explained how Hadoop Distributed File System (HDFS) works and its architecture with suitable illustration. Hadoop's MapReduce paradigm for distributing a task across multiple nodes in Hadoop is discussed with sample data sets. The working of MapReduce and HDFS when they are put all together is discussed. Finally the paper ends with a discussion on Big Data Hadoop sample use cases which shows how enterprises can gain a competitive benefit by being early adopters of big data analytics.**

Keywords— Big data; Analytics; Hadoop; Hadoop Distributed File System (HDFS); Hype cycle; MapReduce; Replication; Fault-tolerance; Unstructured data

I. INTRODUCTION

Recent applications such as index web searches, social networking, banking transactions, recommendation engines genome manipulation in life sciences and machine learning produce huge amounts of data in the form of logs, blogs, email, and other technical structured and unstructured information streams. These data needs to be stored, processed and associated to gain close view into today's business processes. Also, the need to keep both structured and unstructured data to fulfill the government regulations in certain industry sectors requires the storage, processing and analysis of large amounts of data.

While a haze of excitement often envelops the universal discussions of Big Data, a clear agreement has at least combined around the definition of the term. The term "Big Data" is typically considered to be a data collection that has grown so large it can't be affordably or effectively managed using conventional data management tools such as traditional relational database management systems (RDBMS) or conventional search engines, based on the task at hand. Another buzzing term "Big data Analytics" is where advanced analytic techniques are made to operate on big data sets. Thus, big data analytics is really about two things namely, big data and analytics and how the two have coalesced up to create one of the most philosophical trends in business intelligence (BI) today. There are several ways to store, process and analyze large volumes of data in a massively parallel scale. Hadoop is considered as a best example for a massively parallel processing system.

II. WHAT IS HADOOP?

Hadoop is an open source Apache software framework that evaluates gigabytes or petabytes of structured or unstructured data and transforms it into a more manageable form for applications to work with. As a budding technology solution, Hadoop design concerns are new to most users and not common knowledge. MapReduce framework launched by Google by leveraging the concept of map and reduce functions are well known used in Functional Programming. Even though the Hadoop framework is written in Java language, it allows developers to deploy custom written programs coded in Java or any other language to process data in a parallel manner across thousands of commodity servers. It is optimized for adjacent read requests (based on streaming reads), where as processing consists of scanning all the data. Based on the complexity of the process and the volume of data, response time can vary from minutes to hours. Hadoop can process the given data speedy, and it is considered as the key advantage for massive scalability. Hadoop is depicted as a solution to abundant applications in visitor behavior, image processing, web log analysis, search indexes, analyzing and indexing textual content, for research in natural language processing and machine learning, scientific applications in physics, biology and genomics and all forms of data mining [1]. Hadoop emerged as a distributed software platform for transforming and managing large quantities of data, and has grown to be one of the most popular tools to meet many of the above mentioned needs in a cost-effective manner. By summarizing away many of the high availability (HA) and distributed programming

978-1-4799-3966-4/14 $31.00 © 2014 IEEE
DOI 10.1109/ICICA.2014.16

issues, Hadoop allows developers to focus on higher level algorithms. Hence Hadoop is intended to run on a large cluster of commodity servers and to scale to hundreds or thousands of nodes.

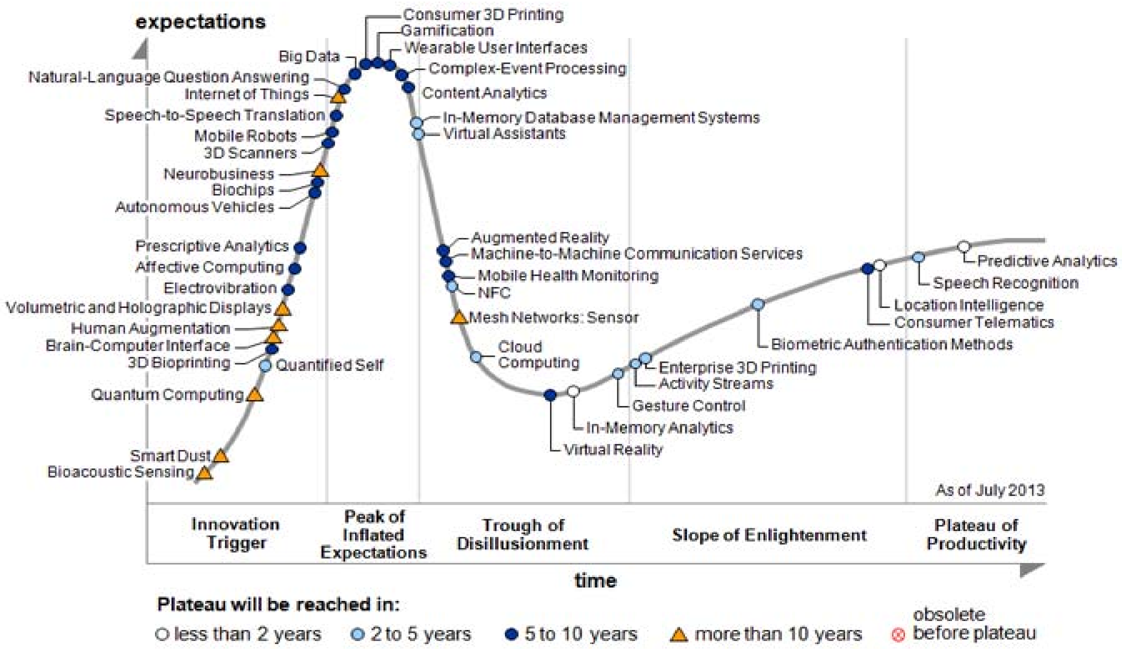
III. THE RISE OF BIG DATA AND HYPE CYCLE

Big data is a term which portrays a situation where the volume, velocity and variety of data exceed an organization's storage or compute capacity for accurate and timely decision making. Some of this data is detained in transactional data stores, the byproduct of fast-growing online activity. Machine-to-machine communications, such as metering, call detail records, environmental sensing and RFID systems, generate their own tidal waves of data. Data is pouring in from every feasible direction from operational and transactional systems, from scanning and facilities management systems, from inbound and outbound customer contact points, from mobile media and the Web. All these forms of data are expanding, and that is together with fast-growing streams of unstructured and semi structured data from social media. Organizations are swamped with data – terabytes and petabytes of it. To put it in perception, 1 TB contains 2,000 hours of CD-quality music and 10 TB could store the entire collection of US Library of Congress print. Exabytes, zettabytes and yottabytes definitely are on the prospect. According to IDC, in 2011, the amount of information produced and replicated will go beyond 1.8 zettabytes (1.8 trillion GB), growing by a factor of nine in just five years [2]. That is nearly as many bits of information in the digital universe as stars in the physical universe.

The Hype Cycle for Emerging Technologies statement is the longest running annual Hype Cycle, providing a cross-industry viewpoint on the technologies and trends that senior executives, strategists, innovators, CIOs, business developers and technology planners should regard as in developing emerging technology portfolios. "It is the broadest collective featuring technologies that are the focus of attention because of particularly high levels of hype, or those that Gartner believes have the potential for significant impact", says Jackie Fenn, vice president, Gartner group and Gartner fellow.

Gartner just released the new Hype Cycle for Emerging Technologies of 2013 (see Fig.1). Gartner's Hype Cycle Special statement offers strategists and planners with an evaluation of the maturity, business benefit and future direction of more than 2,000 technologies, grouped into 98 areas. Hype Cycles guesstimates how long technologies and trends will acquire to reach maturity and helps organizations make a decision when to implement. It characterizes the five stages of new technology adoption and starts with a Technology Trigger: a new invention or innovation. It goes next all the way up to a "peak of inflated expectations" and after that down to a "trough of disillusionment". Successful innovations possibly will climb the "slope of enlightenment" and, finally attain "plateau of productivity".

One of the significant items on this year's Hype Cycle for Emerging Technologies is that Gartner tones down the expectations of big data [3]. In the 2012 version of the Hype Cycle for Emerging Technologies, Gartner predicted that it would take 2-5 years before big data would reach the plateau of productivity [4]. In this year's edition Gartner adjusted that prediction to 5-10 years. The intimately connected trend of the Internet of Things still necessitate over 10 years to reach the plateau of productivity, just like it was predicted last year.



Source: Gartner August 2013

Figure 1. Gartner Hype Cycle for Emerging Technologies 2013

IV. HADOOP DISTRIBUTED FILE SYSTEM (HDFS) ARCHITECTURE

To really understand how it is possible to scale a Hadoop cluster to hundreds and thousands of nodes, we should start with HDFS. Hadoop consist of two basic components: a distributed file system (inspired by Google File System [5]) and the computational framework (Google MapReduce [6]). In the first component of above two, data is stored in Hadoop Distributed File System (HDFS). Hadoop Distributed File System (HDFS) uses a write-once, read-many model that breaks data into blocks that it spreads across many nodes for fault tolerance and high performance.

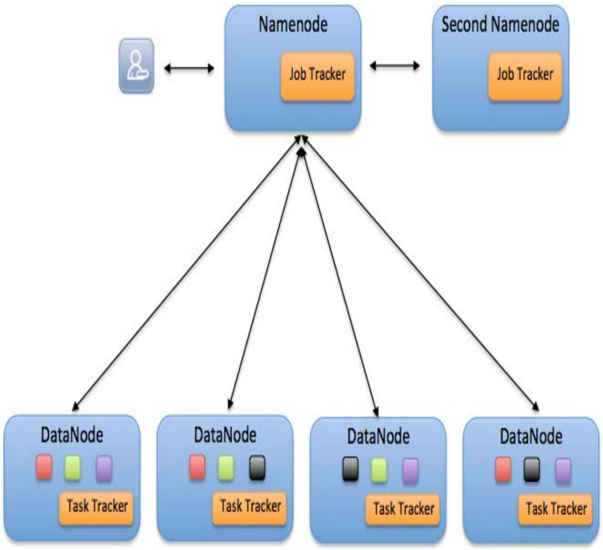


Figure 2. HDFS Daemons

Hadoop and HDFS make use of master-slave architecture. HDFS is written in Java language, with an HDFS cluster consisting of a primary NameNode – a master server that manages the file system namespace and also controls access to data by clients. There is also a Secondary NameNode (see Fig. 2) which maintains a copy of the NameNode data to be used to restart the NameNode when failure occurs, although this copy may not be current and so some data loss is still likely to occur. In addition to it, there are a number of DataNodes; generally, there is a one-to-one relationship between a DataNode and a physical machine. Each DataNode manages the storage attached to the boxes that it runs on. HDFS makes use of a file system namespace that enables data to be stored in files. Each file is divided into one or more blocks, which are then shared across a set of DataNodes. The NameNode is accountable for tasks such as opening, renaming, and closing files and data directories. It also deals with the job of mapping blocks to DataNodes, which are then responsible for managing incoming I/O requests from clients. The DataNode looks after block replication, creation, and removal of data when instructed to do so by the NameNode. A typical Hadoop deployment with HDFS looks like in Fig. 2.

V. MAP REDUCE FRAMEWORK

Another basic component of Hadoop is MapReduce, which affords a computational framework for data processing. MapReduce is a programming replica and an associated implementation for processing and generating large data sets [6]. MapReduce programs are inherently parallel and thus very suitable to a distributed environment. Hadoop takes a cluster of nodes to run MapReduce programs massively in parallel. A single JobTracker schedules all the jobs on the cluster, as well as individual tasks. Here, each benchmark test is a job and runs by itself on the cluster. A job is split into a set of tasks that execute on the worker nodes. A TaskTracker running on each worker node is responsible for starting tasks and reporting progress to the JobTracker. As the name implies, a MapReduce program consists of two major steps, namely, the Map step processes input data and the next step Reduce assembles intermediate results into a final result. Both use key-value pairs defined by the user as input and output. This allows the output of one job to provide directly as input for another. MapReduce programs runs on local file system and local CPU for each cluster node. Data are broken into data blocks (usually in size of 64MB blocks), stored across the local files of different nodes, and replicated for reliability and fault tolerance. The local files constitute the file system which is called as Hadoop Distributed File System being discussed above. The number of nodes in each cluster differs from hundreds to thousands of machines. Naturally, we can write a program in MapReduce to compute the output as shown in the Fig. 3. The high-level structure would look like this:

```
mapper (filename, file-contents):
  for each word in file-contents:
    emit (word, 1)

reducer (word, values):
  sum = 0
  for each value in values:
    sum = sum + value
  emit (word, sum)
```

Problem: Count the number of times that each word appears in a given paragraph.

A simple MapReduce program can be written to resolve how many times different words appear in a set of files. For example, if we had the following files, then the result would be as shown in Fig.3.

Siva has a red mobile, which has no camera. Ram has a red tablet. Prabu has no mobile or tablet.

VI. HDFS & MAP REDUCE PUT TOGETHER

A. *The Hadoop Distributed File System (HDFS)*

HDFS is a fault tolerant and self-healing distributed file system intended to turn a cluster of industry standard servers into a massively scalable pool of storage [7]. HDFS being developed exclusively for large-scale data processing workloads where scalability, flexibility and throughput are

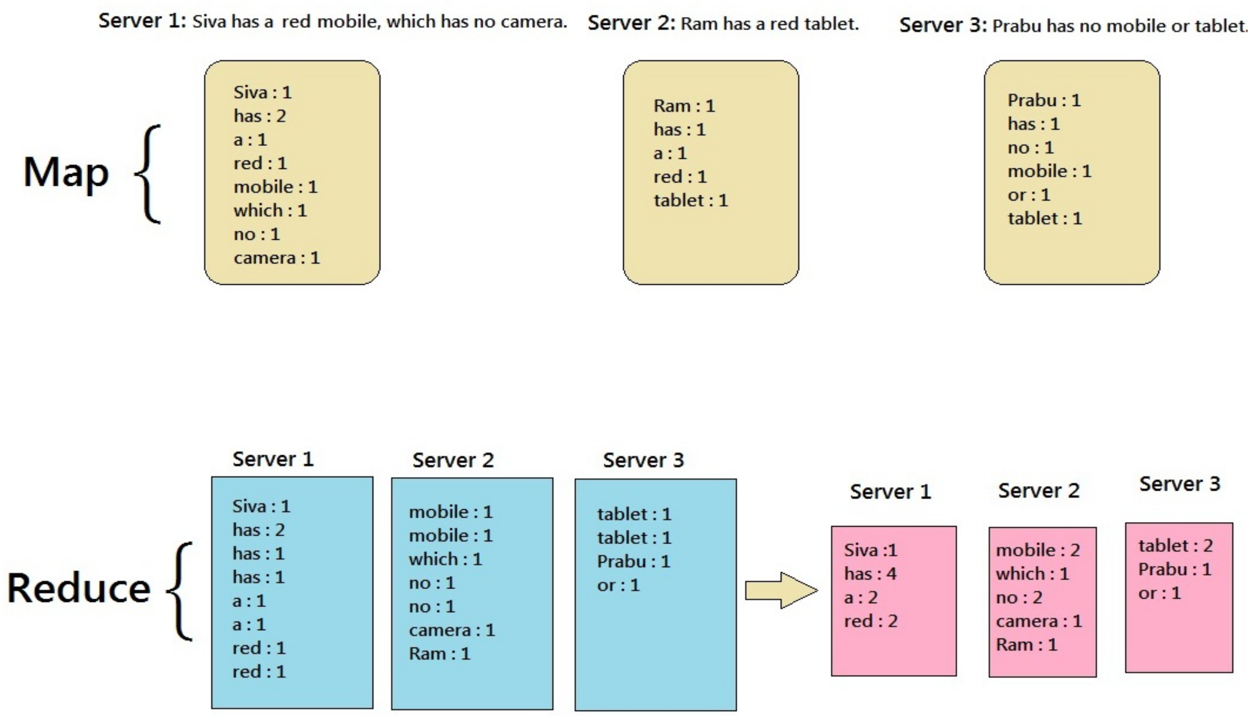


Figure 3. MapReduce Word Count Example

critical, it accepts data in any format despite of schema, optimizes for high bandwidth streaming, and scales to definite deployments of 100 Peta Bytes and beyond. In HDFS, data is replicated across multiple nodes for compute performance and data protection.

1) HDFS key attributes

a) High Availability: Provides mission-critical workflows and applications

b) Fault Tolerance: Automatically and flawlessly recover from failures

c) Scale-Out Architecture: Can add servers to increase capacity

d) Flexible Access: Multiple and open frameworks for serialization and file system mounts

e) Load Balancing: Position data intelligently for maximum efficiency and utilization

f) Tunable Replication: Multiple copies of each file provide data protection and computational performance

B. MapReduce

MapReduce is a massively scalable, parallel processing framework that works in connection with HDFS. With MapReduce and Hadoop, compute is executed at the location of the data, rather than moving data to the compute location; data storage and computation coexist on the same physical nodes in the cluster. MapReduce processes exceedingly large amounts of data without being affected by traditional bottlenecks like network bandwidth by taking advantage of this data proximity. MapReduce divides workloads up into multiple tasks that can be executed in parallel.

1) MapReduce key attributes

a) Resource Manager: Employs data locality and server resources to determine optimal computing operations

b) Optimized Scheduling: Completes jobs according to prioritization

c) Flexibility: Procedures can be written in virtually any programming language

d) Resiliency & High Availability: Multiple job and task trackers ensure that jobs fail independently and restart automatically

e) Scale-out Architecture: Can add servers to increase processing power

VII. REAL-WORLD HADOOP USE CASES

One of the goals of this section is around showcasing end-to-end use cases for Hadoop. This is a collection of some use cases of Hadoop. This is not meant to be a comprehensive list, but a sample that will give some ideas.

A. Healthcare (Storing and Processing Medeical Records)

1) Problem: A health IT company instituted a policy of saving seven years of historical claims and remit data, but its in-house database systems had trouble meeting the data retention requirement while processing millions of claims every day.

2) Solution: A Hadoop system allows archiving seven years' claims and remit data, which requires complex processing to get into a normalized format, logging terabytes of data generated from transactional systems daily, and storing them in CDH for analytical purposes

3) Hadoop vendor: Cloudera

4) Cluster/Data size: 10+ nodes pilot; 1TB of data / day

This real-time use case based on storing and processing medical records is examined from [9].

B. Nokia

1) Problem:

a) Dealing with 100TB of structured data and 500TB+ of semi-structured data

b) 10s of PB across Nokia, 1TB / day

2) Solution: HDFS data warehouse allows storing all the semi/multi structured data and offers processing data at peta byte scale.

3) Hadoop Vendor: Cloudera

4) Cluster/Data size:

a) 500TB of data

b) 10s of PB across Nokia, 1TB / day

Nokia collects and analyzes vast amounts of data from mobile phones. This use case was based on a case study [10] where Nokia needed to find a technology solution that would support the collection, storage and analysis of virtually unlimited data types and volumes.

C. Telecoms

1) Problem: Storing billions of mobile call records and providing real time access to the call records and billing information to customers. Traditional storage/database systems couldn't scale to the loads and provide a cost effective solution

2) Solution: HBase is used to store billions of rows of call record details. 30TB of data is added monthly

3) Hadoop vendor: Intel

4) Hadoop cluster size: 100+ nodes

Intel hardware and software solutions help China Mobile Guangdong build a new high-performance, reliable and cost-effective detailed billing statement inquiry system as it gears up for continuous customer growth. This use case is based on the Intel study report [11].

D. Data Storage

NetApp collects diagnostic data from its storage systems deployed at customer sites. This data is used to analyze the health of NetApp systems.

1) Problem: NetApp collects over 600,000 data transactions weekly, consisting of unstructured logs and system diagnostic information. Traditional data storage systems proved inadequate to capture and process this data.

2) Solution: A Cloudera Hadoop system captures the data and allows parallel processing of data.

3) Hadoop Vendor: Cloudera

4) Cluster/Data size: 30+ nodes; 7TB of data / month

Cloudera [12] offer organizations a solution that is highly scalable with enterprise storage features that improve reliability and performance and reduce costs.

E. Financial Services (Dodd-Frank Compliance at a bank)

A leading retail bank is using Cloudera and Datameer [13] to validate data accuracy and quality to comply with regulations like Dodd-Frank

1) Problem: The previous solution using Teradata and IBM Netezza was time consuming and complex, and the data mart approach didn't provide the data completeness required for determining overall data quality.

2) Solution: A Cloudera + Datameer platform allows analyzing trillions of records which currently result in approximately one terabyte per month of reports. The results are reported through a data quality dashboard.

3) Hadoop Vendor: Cloudera + Datameer

4) Cluster/Data size: 20+ nodes; 1TB of data / month

VIII. CONCLUSION

We are in the era of Big Data. Every day, we generate 2.5 quintillion bytes of data showing that the data in the world today has been created in the last two years alone. In this paper we have highlighted the evolution and rise of big data using Gartner's Hype Cycle for emerging technologies. We have discussed how HDFS produces multiple replicas of data blocks and distributes them on compute nodes throughout a cluster to enable reliable, exceptionally fast computations. We have implemented MapReduce concept that scales to large clusters of machines comprising thousands of machines. Finally the paper ends with the discussion of Real-World Hadoop use cases which helps in Business Analytics.

IX. REFERENCES

[1] Neil Raden, "Big Data Analytics Architecture - Putting All Your Eggs in Three Baskets", 2012

[2] IDC study 2011, EMC. http://www.emc.com/collateral/analyst-reports/idc-extracting-value-from-chaos-ar.pdf

[3] Gartner's Report, "Gartner's 2013 Hype Cycle for Emerging Technologies Maps Out Evolving Relationship Between Humans and Machines", 2013 http://www.gartner.com/newsroom/id/2575515

[4] Gartner's Report, "Hype Cycle for Big Data, 2012", 2012

[5] Ghemawat, Sanjay, Howard Gobioff, and Shun-Tak Leung, "The Google File System" SOSP'03, Oct 19-23, ACM 2003

[6] Dean, Jeffrey and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters", OSDI 2004

[7] HDFS Architecture Guide, http://hadoop.apache.org/docs/r1.0.4/hdfs_design.html

[8] What is Apache Hadoop? http://hadoop.apache.org/

[9] Cloudera Customer Case Study, "Streamlining Healthcare Connectivity with Big Data", 2012.

[10] Cloudera Customer Case Study, "Nokia: Using Big Data to Bridge the Virtual & Physical Worlds", 2012.

[11] Intel Case Study, "China Mobile Guangdong Gives Subscribers Real-Time Access to Billing and Call Data Records", 2012

[12] Cloudera Customer Case Study, "NetApp Improves Customer Support by Deploying Cloudera Enterprise", 2012.

[13] Cloudera Customer Case Study, "Joint Success Story: Major Retail Bank", 2012