Spring 2015

Hadoop "The Emerging Tool in the Present Scenario for Accessing the Large Sets of Data"

Chandra Kiran Movva
Governors State University

Tejaswi Sura
Governors State University

Ranjith Reddy Thipparthi
Governors State University

Recommended Citation

Movva, Chandra Kiran; Sura, Tejaswi; and Thipparthi, Ranjith Reddy, "Hadoop "The Emerging Tool in the Present Scenario for Accessing the Large Sets of Data"" (2015). *All Capstone Projects*. 99.
http://opus.govst.edu/capstones/99

ABSTRACT

Apache Hadoop is a registered trademark of the Apache Software Foundation.

Hadoop is one of the tools designed to handle big data. Hadoop and other software products work to interpret or parse the results of big data searches through specific proprietary algorithms and methods. Hadoop is an open-source program under the Apache license that is maintained by a global community of users. It includes various main components, including a MapReduce set of functions and a Hadoop distributed file system (HDFS). The idea behind MapReduce is that Hadoop can first map a large data set, and then perform a reduction on that content for specific results. A reduce function can be thought of as a kind of filter for raw data.

The HDFS system then acts to distribute data across a network or migrate it as necessary. The term "Hadoop" often refers not just to the base modules above but also to the collection of additional software packages that can be installed on top of or alongside Hadoop, such as Apache Pig, Apache Hive, Apache HBase, Apache Spark, and others. Prominent corporate users of Hadoop include Face book and Yahoo. It can be deployed in traditional onsite datacenters as well as via the cloud; e.g., it is available on Microsoft Azure, Amazon Elastic Compute Cloud (EC2) and Amazon Simple Storage Service (S3), Google App Engine and IBM Bluemix cloud services.

In this paper, we significantly identify and describe the major factors, that Hadoop approach improves accessing large sets of data say "big data" to meet the rapid changing business environments. We also provide a brief comparison Hadoop techniques with traditional systems techniques, and discuss current state of adopting Hadoop techniques. We speculate that from the need to satisfy the customer through time dependency. Hadoop is emerged as an alternative to traditional methods. The purpose of this paper is to provide an in-depth understanding, the major benefits of Hadoop approach to access, as well as provide a study report of Hadoop importance in the present scenario.

"The name my kid gave a stuffed yellow elephant. Short, relatively easy to spell and pronounce, meaningless, and not used elsewhere: those are my naming criteria. Kids are good at generating such. Googol is a kid's term."

---- Doug Cutting, (Hadoop project creator)

Contents

I. INTRODUCTION

Hadoop is a system for large scale data processing. Apache Hadoop is an open source software framework for storage and large scale processing of data-sets on clusters of commodity hardware. Hadoop is an Apache top-level project being built and used by a global community of contributors and users. It is licensed under the Apache License 2.0.Hadoop was created by Doug Cutting and Mike Cafarella in 2005. It was originally developed to support distribution for the Nutch search engine project. Doug, who was working at Yahoo!, he is now Chief Architect of Cloudera.

Hadoop Modules
Hadoop frame work composes of the following modules

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

II. WHY DO WE USE HADOOP?

We create 2.5 quintillion bytes of data, so much that 90% of the data was created in the last two years alone. The data comes from various sources via social networks such as Facebook, Instagram, purchase transaction records, pictures, videos and user data from the maps etc. .This kind of humungous data is called as “BIG DATA”. It is a buzzword that is used to describe large sets of data or massive volume of data both structured and unstructured data.

BIG DATA has become a common term in the IT industries. This term is often used when referring to petabytes and Exabyte’s of data. When dealing with larger data sets, organizations face difficulties in manipulating, creating and accessing the big data. BIG DATA is a major problem in business analytics because no standard tools and procedures are designed to search and analyze massive data sets. To access this large datasets called BIG DATA we use the tool called “HADOOP”. Hadoop is an open source implementation of a large scale batch processing system.

It is inspired by the Google’s map and reduce technique. Hadoop framework is written in java, it allows developers to deploy custom written programs coded in java or any other language to process the data in parallel time across hundreds and thousands of the servers.

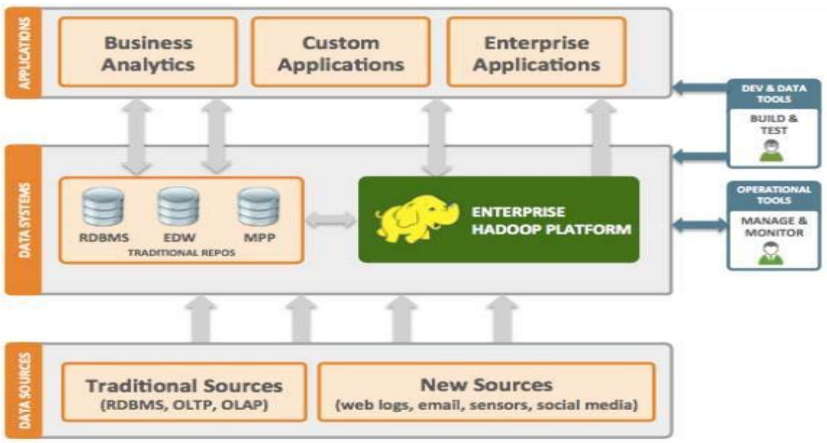
III. TRADITIONAL SYSTEM & HADOOP

The IT industries are implementing hadoop for the following reasons:

- *Reliable:* the software is fault tolerant, it expects and handles hardware and software failures
- *Scalable:* It is designed for massive scale of processors, memory and local attached storage
- *Distributed:* Handles replication .offers parallel programming model, Map Reduce.

Hadoop is currently being used for index web searches, email spam, prediction in financial services and for analysis of unstructured data such as log, text and many other services. While many of these applications could be implemented in relational database (RDBMS). Hadoop framework is functionally different from RDBMS.

`



Source: http://timoelliott.com/blog/2014/04/no-hadoop-isnt-going-to-replace-your-data-warehouse.html

The following discuss some of the differences where hadoop is particularly useful when:

- Complex information processing is needed.
- Unstructured data must be changed to structured
- Queries can't be reasonably expressed using SQL
- Heavily recursive algorithms
- Data sets are too large to fit into database RAM, discs, or require too many cores(up to PB)
- Fault tolerance is critical.
- Significant custom coding would be required to handle job scheduling

.

IV. HDFS

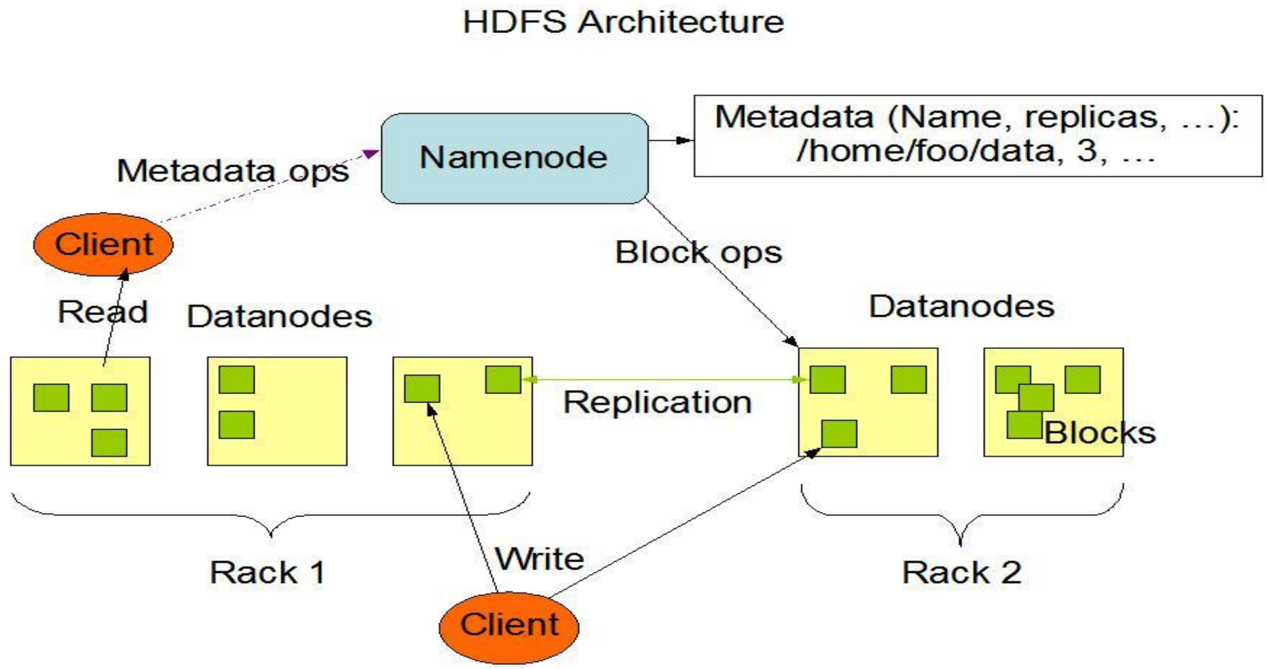
The Hadoop Distributed File System (HDFS) is the file system component of the Hadoop framework. HDFS is designed and optimized to store data over a large amount of low-cost hardware in a distributed fashion.

*Name no*de:

Name node is a type of master node, which is having the information that means Meta data about the all data node there is address (use to talk), free space, data they store, active data node, passive data node, task tracker, job tracker and many other configuration such as replication of data.

The NameNode records all of the metadata, attributes, and locations of files and data blocks in to the DataNodes. The attributes it records are the things like file permissions, file modification and access times, and namespace, which is a hierarchy of files and directories. The NameNode maps the namespace tree to file blocks in DataNodes. When a client node wants to read a file in the HDFS it first contacts the Name node to get the location of data blocks associated with that file.

A NameNode stores information about the overall system because it is the master of the HDFS with the DataNodes being the slaves. It stores the image and journal logs of the system. The image of the system is a list of blocks and data for each file stored in the HDFS. The journal is just a modification log of the image. The NameNode must always store the most up to date image and journal. Basically, the NameNode always knows where the data blocks and replicates are for each file and it also knows where the free blocks are in the system so it keeps track of where future files can be written



Source: http://hadoop.apache.org/docs/r1.2.1/hdfs_design.html

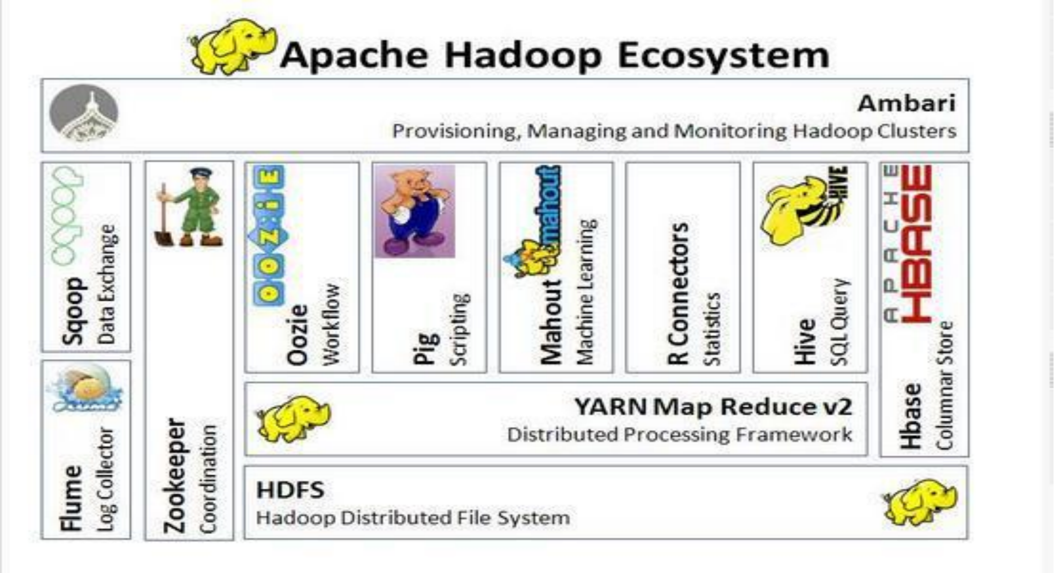
*DataNod*e:

Data node is a type of slave node in the hadoop, which is used to save the data and there is task tracker in data node which is used to track on the ongoing job on the data node and the jobs which coming from name node.

The DataNodes store the blocks and block replicas of the file system. During startup each DataNode connects and performs a handshake with the NameNode. The DataNode checks for the accurate namespaceID, and if not found then the DataNode automatically shuts down. New DataNodes can join the cluster by simply registering with the NameNode and receiving the namespace ID. DataNode keeps track of a block report for the blocks in its node. Each DataNode sends its block report to the NameNode every hour so that the NameNode always has an up to date view of where block replicas are located in the cluster. During the normal operation of the HDFS, each DataNode also sends a heartbeat to the NameNode every ten minutes so that the NameNode knows which DataNodes are operating correctly and are available. If after ten minutes the NameNode doesn’t receive a heartbeat from a DataNode then the NameNode assumes that the DataNode is lost and begins creating replicas of that DataNode’s lost blocks on other DataNodes. The nice thing about the HDFS architecture is that the NameNode doesn’t have to reach out to the DataNodes, it instead waits for the DataNodes to send their block reports a n d heartbeats to it. The NameNode can receive thousands of DataNode’s heartbeats every second and not adversely affect other NameNode operations. The following figure depicts the hadoop ecosystem.

V. HADOOP ARCHITECHUTRE

"Hadoop" often refers not to just the base Hadoop package but rather to the **Hadoop Ecosystem** , which includes all of the additional software packages that can be installed on top of or alongside Hadoop, such as Apache Hive, Apache Pig and Apache Spark

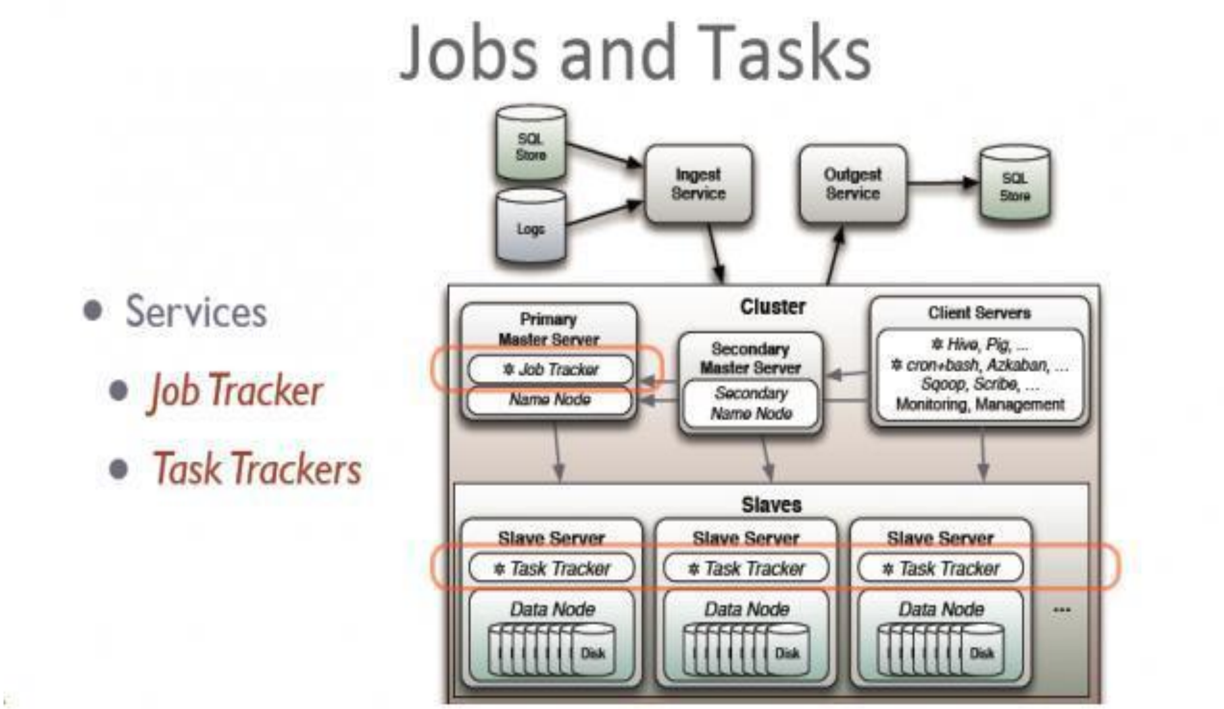


Source: http://opensource.com/life/14/8/intro-apache-hadoop-big-data

VI. MAP REDUCE FRAMEWORK

Map Reduce is a software framework for distributed processing of large data sets on computer clusters. It is first developed by Google .Map Reduce is intended to facilitate and simplify the processing of vast amounts of data in parallel on large clusters of commodity hardware in a reliable, fault-tolerant manner.

Map Reduce is the key algorithm that the Hadoop Map Reduce engine uses to distribute work around a cluster. Typical Hadoop cluster integrates Map Reduce and HFDS layer. In Map Reduce layer job tracker assigns tasks to the task tracker .Master node job tracker also assigns tasks to the slave node task tracker



Jobs and Tasks

- Services
  - Job Tracker
  - Task Trackers

Source: http://opensource.com/life/14/8/intro-apache-hadoop-big-data

Above the file systems comes the Map Reduce engine, which consists of one Job Tracker, to which client applications submit Map Reduce jobs. The Job Tracker pushes work out to available Task Tracker nodes in the cluster, striving to keep the work as close to the data as possible.

With a rack-aware file system, the Job Tracker knows which node contains the data, and which other machines are nearby. If the work cannot be hosted on the actual node where the data resides, priority is given to nodes in the same rack. This reduces network traffic on the main backbone network.

If a Task Tracker fails or times out, that part of the job is rescheduled. The Task Tracker on each node spawns off a separate Java Virtual Machine process to prevent the Task Tracker itself from failing if the running job crashes the JVM. A heartbeat is sent from the Task Tracker to the Job Tracker every few minutes to check its status. The Job Tracker and Task Tracker status and information is exposed by Jetty and can be viewed from a web browser.

VII. PIG

Pig is a high level scripting language that is used with Apache Hadoop. Pig excels at describing data analysis problems as data flows. Pig is complete in that you can do all the required data manipulations in Apache Hadoop with Pig. In addition through the User Defined Functions (UDF) facility in Pig you can have Pig invoke code in many languages like JRuby, Python and Java. Conversely you can execute Pig scripts in other languages. The result is that you can use Pig as a component to build larger and more complex applications that tackle real business problems.

Pig scripts are translated into a series of Map Reduce jobs that are run on the Apache Hadoop cluster. As part of the translation the Pig interpreter does perform optimizations to speed execution on Apache Hadoop. We are going to write a Pig script that will do our data analysis task. Pig is made up of two components: the first is the language itself, which is called Pig Latin (people naming various Hadoop projects do tend to have a sense of humor associated with their naming conventions), and the second is a runtime environment where Pig Latin programs are executed. Think of the relationship between a Java Virtual Machine (JVM) and a Java application. In this section, we'll just refer to the whole entity as Pig. Let's first look at the programming language itself so that you can see how it's significantly easier than having to write mapper and reducer programs.

1. The first step in a Pig program is to LOAD the data you want to manipulate from HDFS.

2. Then you run the data through a set of transformations (which, under the covers, are translated into a set of mapper and reducer tasks).
3. Finally, you DUMP the data to the screen or you STORE the results in a file somewhere

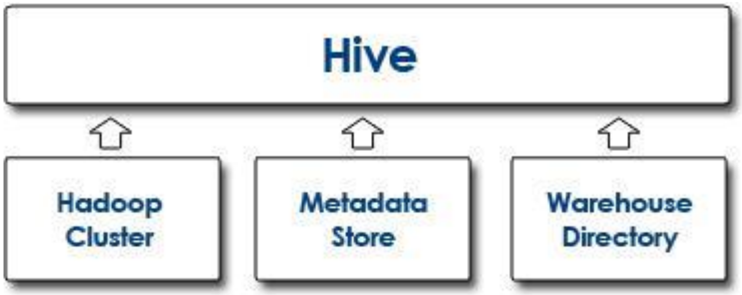
VIII. HIVE

Although Pig can be quite a powerful and simple language to use, the downside is that it's something new to learn and master. Some folks at Facebook developed a runtime Hadoop® support structure that allows anyone who is already fluent with SQL (which is commonplace for relational data-base developers) to leverage the Hadoop platform right out of the gate.

Their creation, called **Hive**™, allows SQL developers to write Hive Query Language (HQL) statements that are similar to standard SQL statements; now you should be aware that HQL is limited in the commands it understands, but it is still pretty useful. HQL statements are broken down by the Hive service into Map Reduce jobs and executed across a Hadoop cluster.

Hive looks very much like traditional database code with SQL access. However, because Hive is based on Hadoop and Map Reduce operations, there are several key differences. The first is that Hadoop is intended for long sequential scans, and because Hive is based on Hadoop, you can expect queries to have a very high latency (many minutes). This means that Hive would not be appropriate for applications that need very fast response times, as you would expect with a database such as DB2. Finally, Hive is read-based and therefore not appropriate for transaction processing that typically involves a high percentage of write operations.

Hive has three main functions: data summarization, query and analysis. It supports queries expressed in a language called HiveQL, which automatically translates SQL-like queries into Map Reduce jobs executed on Hadoop. In addition, HiveQL supports custom Map Reduce scripts to be plugged into queries. Hive also enables data serialization/deserialization and increases flexibility in schema design by including a system catalog called Hive-Metastore.



Source: http://www.j2eebrain.com/java-J2ee-hadoop-hive.html\

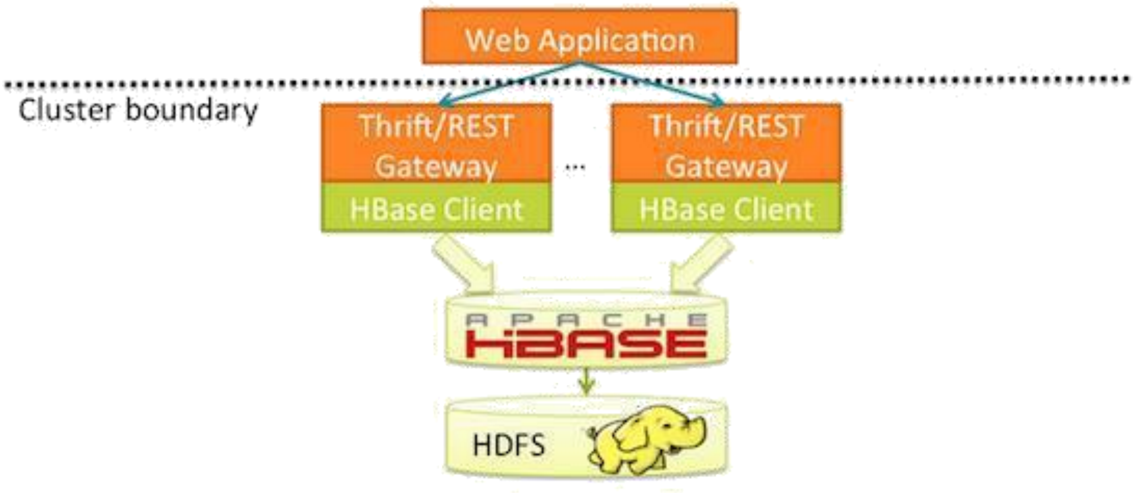
Hive supports text files (also called flat files), Sequence Files (flat files consisting of binary key/value pairs) and RCFiles (Record Columnar Files which store columns of a table in a columnar database way.)

IX. HBASE

HBase is a column-oriented database management system that runs on top of HDFS. It is well suited for sparse data sets, which are common in many big data use cases. Unlike relational database systems, HBase does not support a structured query language like SQL; in fact, HBase isn't a relational data store at all. HBase applications are written in Java much like a typical Map Reduce application. HBase does support writing applications in Avro, REST, and Thrift.

An HBase system comprises a set of tables. Each table contains rows and columns, much like a traditional database. Each table must have an element defined as a Primary Key, and all access attempts to HBase tables must use this Primary Key. An HBase column represents an attribute of an object; for example, if the table is storing diagnostic logs from servers in your environment, where each row might be a log record, a typical column in such a table would be the timestamp of when the log record was written, or perhaps the server name where the record originated. In fact, HBase allows for many attributes to be grouped together into what are known as column families, such that the elements of a column family are all stored together. This is different from a row-oriented relational database, where all the columns of a given row are stored together. With HBase you must predefine the table schema and specify the column families. However, it's very flexible in that new columns can be added to families at any time, making the schema flexible and therefore able to adapt to changing application requirements.

Just as HDFS has a NameNode and slave nodes, and Map Reduce has Job Tracker and Task Tracker slaves, HBase is built on similar concepts. In HBase a master node manages the cluster and region servers store portions of the tables and perform the work on the data. In the same way HDFS has some enterprise concerns due to the availability of the NameNode (among other areas that can be "hardened" for true enterprise deployments by Info Sphere Big Insights), HBase is also sensitive to the loss of its master node.



Source: http://blog.cloudera.com/blog/2013/03/how-to-use-the-apache-hbase-rest-interface-part-1/

X. HADOOP ADOPTION AND MARKET SIZE

In its research report on the growth patterns of the Hadoop Market Analysis**,** Allied Market Research forecasts that the global Hadoop market will grow at a CAGR of 58.2% between 2013 and 2020. The global market revenue, which was estimated at $2.0 billion in 2013, is rapidly expanding and may grow up to a staggering $50.2 billion by 2020.The sudden spurt in demand for big data analytics is the major driver for the expansion of the Hadoop market. In the recent years, businesses have experienced a massive explosion of raw, structured and unstructured data, which has necessitated the utilization of big data analytics. This marked trend—along with the need for agile, cost-friendly processing of business data has established “big data” as the chosen platform for data analytics over conventional, data analysis platforms like relational database management systems or data warehouses. At present, the major concerns related to Hadoop adoption are distributed computing and security issues, which may have slowed down adoption of this technology to some degree. However, industry leaders are addressing these concerns on a priority basis. Also, the lowering costs of data-management solutions may have fueled the growth of this market.

The market research report divides the market into three distinct segments: hardware, software and services.

HADOOP HARDWARE MARKET

1. The Hadoop hardware market is broadly divided into the following segments: servers, storage, and network equipment.
2. The storage market remained the leader in Hadoop hardware market in 2013. Social media sites Facebook and Twitter contributed to this growth by adding terabytes of data every day!
3. The Hadoop server market is expected to grow at a CAGR of 60.1% during 2013-2020 as the demand for more volume and more velocity are ever increasing.

4. Hadoop hardware-based, solution providers have been the highest receivers of venture capital funding. The recent times have witnessed a steep demand for real-time, operational analytics.

HADOOP SOFTWARE MARKET

1. The software segment is projected to have the highest growth rate in the overall Hadoop market.
2. The huge VC funding activities has triggered phenomenal growth in the number of Hadoop distributors, which in turn has led to the growth of Hadoop software market.
3. The Hadoop software market is further segmented as application software, management software, packaged software and performance monitoring software markets.
4. The application software segment generated the maximum revenue in the overall 2013 Hadoop software market. The major contributors to this market are the application software built for big data analytics.

HADOOP SERVICES MARKET

1. At present, the services market is dominating with a market share of about 50% of the global Hadoop market due to growing necessity of big data analytics in worldwide organizations.
2. The Hadoop services market comprises the following segments: consulting, training and outsourcing, integration and deployment, and middleware & support services.
3. The Consulting and training segment, along with the outsourcing segment, accounted for highest revenue generation in 2013.
4. The integration and deployment segment is projected to be the fastest growing services market largely because of huge investments in big data analytics and due to the need for operational analytics.

ADOPTION OF HADOOP

Hadoop has been most rapidly adopted by the government, banking, finance, IT and ITES, and insurance sectors. The government sector happens to be the largest revenue generator in Hadoop application market. The report claims that by 2020, BFSI would supersede the government sector and become the highest revenue-generating segment in the overall Hadoop application market.

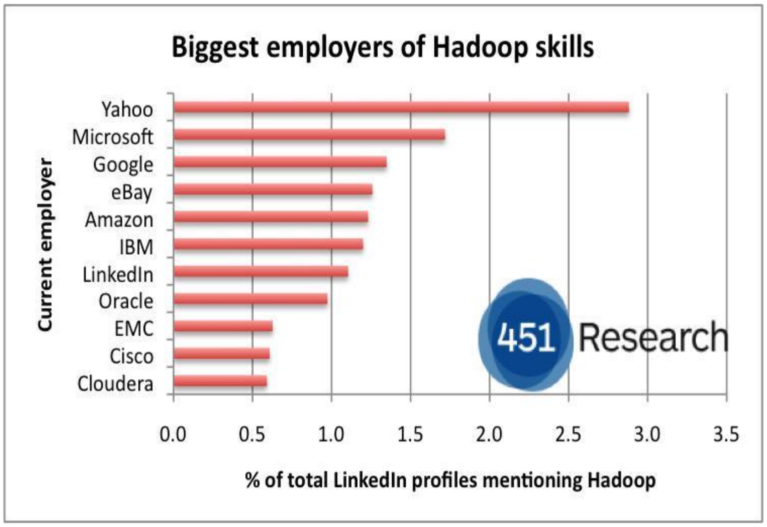
Geographical analysis of the market seems to suggest that North America is the leading revenue generating market and will continue to remain so till 2020. This geographical control of the Hadoop market growth is specifically due to the presence of many prominent Hadoop distribution companies in the N American region.

Next, Asia Pacific is projected to be the fastest growing regional market at a CAGR at 59.2% during 2013-2020.The report has profiled some of the key players of the market such as Amazon Web Services, Cloudera, Datameer, Hortonworks, Cisco Systems, etc.

XI. EMPLOYER STATISTICS

The Indian Big Data industry is predicted to grow five-fold from the current level of $200 million to $1 billion by 2015 which is 4% of the expected global share. At the same time Gartner has predicted that there is going to be significant gap in job openings and candidates with Big Data skills. This is the right time to take advantage of this opportunity. This skill gap in Big Data can be bridged through comprehensive learning of Apache Hadoop that enables professionals and fresher's alike, to add the valuable Big Data skills to their profile.

LinkedIn is the best place to get information on the number of existing Hadoop professional. The above info graph talks about the top companies employing Hadoop professionals and who is leading of them all. Yahoo! happens to be leading in this race.



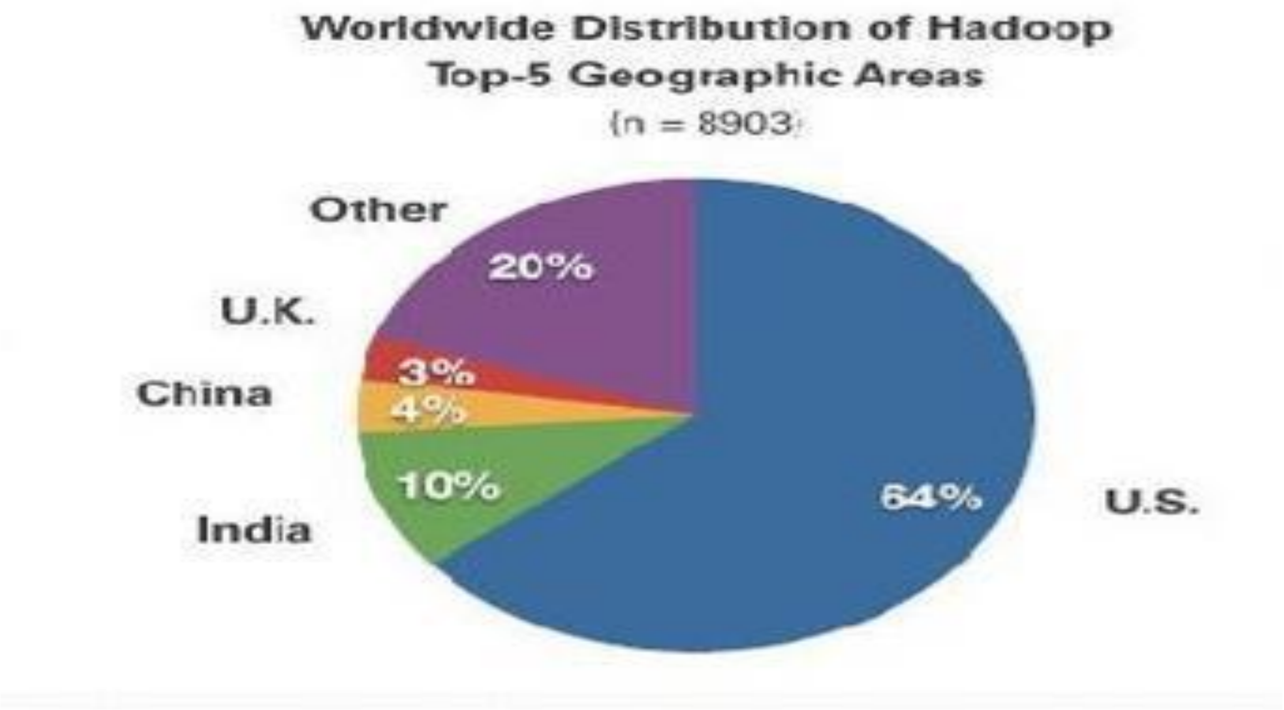
Source: http://blogs.the451group.com/information_management/2012/08/10/the-distribution-of-hadoop-and-mapreduce-skills-according-to-linkedin/

- Microsoft is the second largest employer of both Hadoop and MapReduce skills, according to LinkedIn member profiles.
- Redmond employs 1.7% of all LinkedIn members with Hadoop in their profiles, and 3.0% of members with MapReduce in their profiles.
- Yahoo is the largest employer of Hadoop skills (2.9%), with other 'non-vendors' also well represented: Google (1.3%), eBay (also 1.3%), Amazon (1.2%), LinkedIn (1.1%).
- Google is by far the largest employer of MapReduce skills, as you might expect, with 7.1%. Also well represented are Yahoo (2.7%), Amazon (1.8%) and LinkedIn (1.4%).

XII.WORLD WIDE DISTRIBUTION OF HADOOP

One wellspring of information that could be utilized to inspect this inquiry is Linkedin: individuals who use Hadoop may be slanted to say it in their profiles. In no way, shape or form a flawless measure, it can maybe offer knowledge into the geographic and industry conveyances of Hadoop clients

While it would be perfect to work with Linkedin's information researchers to perform the investigation, a snappy and simple methodology is to utilize the propelled hunt work on the site. "Hadoop" is sensibly unambiguous to the extent magic words go. A pursuit of Linkedin individuals yields the accompanying recurrence circulations.



Source: http://analytics.ncsu.edu/?page_id=3626

XIII.CONCLUSION

Exponential increase in the amount of data generated (also called big data) is majorly contributing to the growth of Hadoop solution. The application such as Banking, Financial Services and Insurance (BFSI), retail, healthcare and life sciences, media and entertainment, government, and telecommunication among others are generating a massive amount of data. So, there is a need of a tool to handle and analyze big data. Hadoop is a cost effective solution and can manage structured as well as unstructured data unlike traditional solutions such as RDBMS. The need to track and analyze consumer behavior, maintain inventory and space, target marketing offers on the basis of consumer preferences and attract and retain consumers, are some of the factors pushing the demand for Hadoop architecture solutions.

XIV. REFERENCES

https://hadoop.apache.org/ http://opensource.com/life/14/8/intro-apache-hadoop-big-data

http://www.ijcsit.com/docs/Volume%205/vol5issue06/ijcsit20140506229.pdf

http://www.cloudera.com/content/cloudera/en/about/hadoop-and-big-data.html

http://www-01.ibm.com/software/data/infosphere/hadoop/hive/

http://www.informit.com/articles/article.aspx?p=2253412/

http://hortonworks.com/hadoop/yarn/

http://www.tomsitpro.com/articles/hadoop-2-vs-1,2-718.html

http://searchcloudcomputing.techtarget.com/definition/Hadoop

http://searchdatamanagement.techtarget.com/definition/Apache-Hive

http://www.experfy.com/blog/hadoop-market-size-adoption-growth-2020/

http://www.alliedmarketresearch.com/hadoop-market

http://www.transparencymarketresearch.com/hadoop-market.html http://www-01.ibm.com/software/data/infosphere/hadoop/hbase/ http://www-01.ibm.com/software/data/infosphere/hadoop/hive/