SURVEY ON ROLE OF HADOOP IN CLOUD COMPUTING ENVIRONMENT

Vandana Vijay
Assistant Professor
Computer Science Department
S. S. Jain Subodh P.G. (Autonomous) College, Jaipur, India

Abstract: In Today's era of Internet World, Cloud Computing emerged as a new computational paradigm. It is an internet based technology that enables small business and organizations to use highly sophisticated computer applications. Basically it refers to the applications and services offered over the Internet. These services are offered from data centers all over the world, which collectively are referred to as the "cloud. Despite of its advantages, Cloud Computing has many drawbacks (low scalability, no support for stream data processing). Hadoop is an open-source software framework for storing data and running applications on clusters of commodity hardware. It provides massive storage for any kind of data, enormous processing power and the ability to handle virtually limitless concurrent tasks or jobs. Hadoop framework needs to be implemented in cloud computing to overcome its drawbacks. This paper highlights the role of Hadoop in Cloud Computing environment.

***Index Terms* -** Cloud computing, Hadoop, Hadoop Ecosystem, HDFS, MapReduce.

1. INTRODUCTION

In today's environment almost all the companies have migrated their applications and data to the cloud due to the popularity of the Internet along with a sharp increase in the network bandwidth. This has resulted in the generation of huge amounts of data on regular basis. So in current circumstances, management of big distributed data like cloud is a big challenge. For processing such gigantic amount of data, the traditional methods of database management are not appropriate since these approaches fail to handle such size of data. Hence, in order to handle such huge volume of heterogeneous data, companies are now coming up with different alternatives. One of the widely accepted solutions is Hadoop, which has attracted great attention due to its flexibility, accessible, scalable, parallel processing, ease of programming, and fault tolerance features. Hadoop is the open source implementation of MapReduce programming model. MapReduce works along with a distributed file system called Hadoop Distributed File System (HDFS) in Hadoop. Both MapReduce and HDFS in hadoop are derived from the Google's MapReduce and Google File System. The paper is structured as follows: Section 2 provides a literature review on research papers related to Cloud computing with Hadoop. Section 3 describes Hadoop architecture (HDFS and MapReduce). Section 4 defines Cloud Computing and its service models. Finally Section 5 concludes this paper with future perspectives.

2. LITERATURE SURVEY

Tripathi et al. (2018) developed a cloud enabled hadoop framework which combines cloud technology and high computing resources with the conventional hadoop framework to support the spatial big data solutions. They also compared the conventional hadoop framework and proposed cloud enabled hadoop framework. It is observed that the propose cloud enabled hadoop framework is much efficient to spatial big data processing than the current available solutions. **Malhotra et al. (2018)** proposed a model GENMR which convert any RDBMS queries to Map Reduce codes. It can effectively process data at Cloud repositories to overcome the limitations of an existing traditional RDBMS system. Proposed model consist of three modules, firstly user interface where the users can put their queries in any database language, secondly the complier that converts the queries into MapReduce Codes and lastly the enhanced optimal Cross Rack Algorithm to minimize the cross Rack Communication which consider the placement of both mapper as well as reducer. **Bashir (2017)** provides a comprehensive review and analysis of MapReduce programming model in cloud computing. This paper concludes that in the number of computing nodes, cloud MapReduce has high scalability and MapReduce simplifies the large-scale data computation. **Alam & Shakil (2016)** proposed Hadoop based workflow for handling big data. Huge amount of data as well as big data can be managing in very easy ways in less amount of time. In the research work it was found that the average processing time is very less while processing in the cloud environment. **Ikhlaq & Keswani (2016)** Review Big Data methods, Approaches and how cloud computing is implemented with challenges posed by it being addressed. **Patil et al. (2016)** proposed secured Hadoop as a service cloud service that provides need based of Hadoop service through infrastructure cloud with optimized utilization, opportunistic provisioning of cycles from idle nodes to different processes with resolving the data security issues through use of efficient encryption/decryption algorithm. Secured Hadoop as a service on the infrastructure clouds will provide the way to process big data on cloud with provided security to the data and along with the hassle free platform which keeps away users from Hadoop configurations and gives out Hadoop service as web application service. **Ansari et al. (2015)** proposed Data Cleaning mechanism in Hadoop, Push Model and caching. The Data cleaning mechanism enables to clean the already present content memory to fasten the execution process. The Push Model is also the strategy to ork which enables the job tracker to push the heart beat to the task tracker in order to work directly. An ehcache method used to search for the inputs output and save the time of computation of the mapper and reducer phase and directly generate the output. The performance of the system can increase to a good extent by these methods. **Voruganti (2014)** implements MapReduce programming model using two components: a JobTracker (masternode) and many TaskTrackers (slave nodes). Hadoop is basically designed to efficiently process very large data volumes by linking many commodity systems together to work as a parallel entity. **Gupta & Saxena (2014)** proposed big data implementation using Hadoop. It is the most required technology for Cloud Computing. As more Hadoop clusters are offered by cloud vendors in many business. They provide set up of a distributed, single node Hadoop cluster backed by HDFS running on ubuntu operating system. **Dash & Panda (2014)** propose a platform which integrates the Cloud, Big Data, NoSQL, Hadoop and analytic tools to efficiently capture, store and analyze complex datasets. **Lu et al. (2012)** describes the three most crucial parts of Hadoop, including HDFS, the distributed file system, MapReduce, the data processing model, and HBase, the distributed structured data table. Hadoop shows good performance in dealing with large data sets concurrently, there are still some shortcomings (Failure of NameNode, HDFS small files, Job Tracker overload).

3. HADOOP

Hadoop is an Apache open source framework written in Java that allows distributed computing environment for processing of huge amount of data across cluster of computers in parallel manner (Kumari, 2014). Hadoop is core part of a cloud computing infrastructure and is being used by companies like Yahoo, Facebook, IBM, LinkedIn, and Twitter. Hadoop has two main components: Hadoop Distributed File System (HDFS) and MapReduce framework. HDFS supports the storage of the big data within distributed environment while the MapReduce provide the processing of the information ((Bhosale and Gadekar, 2014; Kaur and Kaur, 2015). Hadoop architecture has the following components as shown in figure 1. It works on the concept of master and slave node. Masters have the name node, secondary node and Job Tracker, while slaves have the data node and Task Tracker. JobTracker’s main duty is to initiate tasks, track and dispatch their implementation. TaskTracker manages the processing of local data and the collecting of result data according to the requests from applications and reports the states and performance to JobTracker. NameNode and DataNode are charged with fulfilling HDFS tasks, while JobTracker and TaskTracker mainly deal with MapReduce tasks.

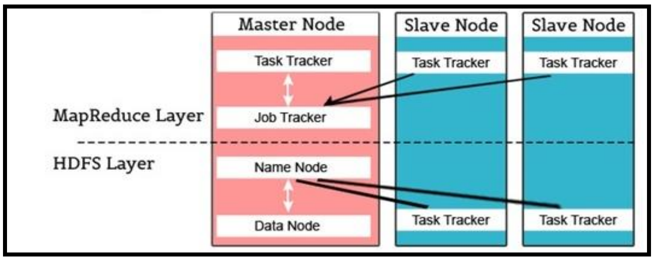

Figure 1: Hadoop architecture

3.1. HDFS

Hadoop distributed file system is composed of a NameNode and some DataNodes. NameNode is a master server, managing system metadata, maintaining fife system namespace and mapping data from database to DataNode. Generally, the number of DataNode in a cluster is one in one node. DataNode stores actual data and manages the storage of physical nodes on which they are located and deals with read/write requests from clients. Every Datanode in the cluster, during startup, makes itself available to the name node through a registration process. A part from that, each Datanode informs Namenode which blocks has in its possession by sending a block report. A block reports are sent periodically or when a change takes place. Furthermore, every datanode sends heartbeat messages to the Namenode to confirm that it remains operational and that the data is safe and available. If a Datanode stops operating, there are error mechanisms in order to overcome the failure and maintain the availability of the block. Heartbeat messages also hold extra information, which helps the Namenode run the cluster efficiently (e.g. storage capacity which enables Namenode to make load balancing).The HDFS architecture is shown in figure 2.

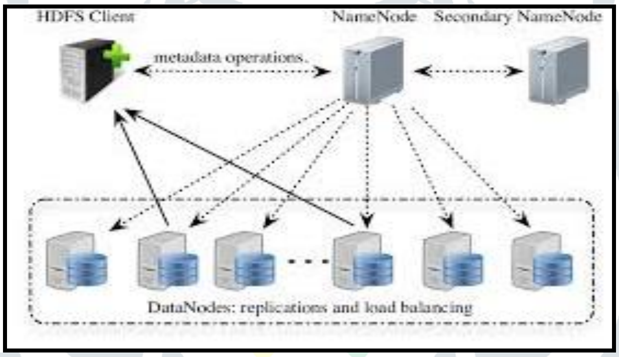

Figure 2: HDFS architecture

In HDFS, files are broken into block-sized chunks, which are independently distributed in nodes. Each block is saved as a separate file in the node's local file system. The size of the block is large and a typical value would be 128MB, but it is a value chosen per client and per file. HDFS is also designed to be fault tolerant, which means that each block should remain accessible in case of system failures. The fault-tolerance feature is implemented through a replication mechanism. Every block is stored in more than one node making highly unlikely that it can be lost. By default, two copies of the block are saved on two different nodes in the same rank and a third copy in a node located in a different rank. The HDFS software continually monitors the data stored on the cluster and in case of a failure (node becomes unavailable, a disk drive fails, etc.) automatically restores the data from one of the known good replicas stored elsewhere on the cluster.

3.2. MapReduce

MapReduce is a programming framework which allows performing parallel and distributed processing on large data sets in a distributed environment using a large number of nodes. MapReduce programming framework consists of two components: a Job Tracker (masternode) and many Task Trackers (slave nodes). The Job Tracker is responsible for accepting job requests, for splitting the data input, for defining the tasks required for the job, for assigning those tasks to be executed in parallel across the slaves, for monitoring the progress and finally for handling occurring failures. The Task Tracker executes tasks as ordered by the master node. The task can be executed either using a map function or reduce function. The MapReduce architecture is shown in figure 3.

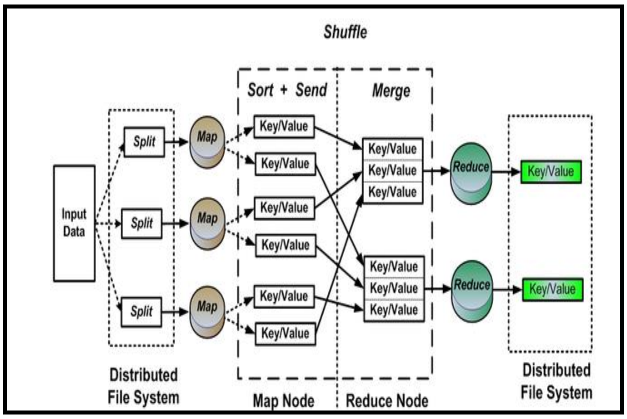

Figure 3: MapReduce architecture

The Map function takes one pair of data with a type in one data domain, and returns a list of pairs in a different domain: Map (k1, v1) → list (k2, v2). The Reduce function is then applied in parallel to each group, which in turn produces a collection of values in the same domain: Reduce (k2, list (v2)) → list (v3).

Hadoop helps us to process huge data sets by distributing the replicated forms of same data into several datanodes whose information is stored in a namenode server. There is a job tracker that splits the job into several tasks each of which is handled by a task tracker. The split files are fed into mappers where the mapping function works and keys and values are generated as (k,v) sets. These are shuffled and put to reducers who cumulate or combine the count or value of similar data sets there by reducing redundancy of data. Also several parallel processing can be obtained by such a framework. The bottom line is that we divide the job, load it in HDFS, employ MapReduce on them, solve them in parallel, and write the cumulative results back to the HDFS. It ensures a powerful, robust and fault tolerant system that can be used to deploy huge data set processing.

Table 1: Hadoop Ecosystem

Hadoop Ecosystem Components	**Information**
HDFS	Hadoop Distributed File System
MapReduce	Distributed computation framework
YARN	Yet Another Resource Negotiator
Spark	In-memory Data Processing
PIG	Dataflow language and parallel execution
HIVE	Data warehouse infrastructure
HBase	Column-oriented table service
Mahout, Spark MLlib	Machine Learning
Apache Drill	SQL on Hadoop
Zookeeper	Managing Cluster (Cooridinator)
Hcatalog	Table and storage management service
Sqoop	Bulk data transfer
Avron	Data serialization system

Ambari	Provision, Monitor and Maintain cluster
Solr & Lucene	Searching & Indexing
Flume, Sqoop	Data Ingesting Services (data collection)

Table 2: Uses of MapReduce

Google	**Yahoo!**	**Facebook**
Index building for Google Search	Index building for Yahoo! Search	Ad optimization
Statistical machine translation	Spam detection for Yahoo! Mail	Spam detection
Article clustering for Google News		

4. CLOUD COMPUTING

Cloud computing (Mell et al. 2015) is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Cloud is a network of servers that pools different resources. With the advent of this technology, the cost of computation, application hosting, content storage and delivery is reduced significantly. Cloud computing is a practical approach to experience direct cost benefits and it has the potential to transform a data center from a capital-intensive set up to a variable priced environment.

Cloud Providers offer services that can be grouped into categories depending upon either the type of service being provided or on the basis of location as shown in figure 5. According to Almorsy et al. (2016), the three basic service models are described as follows:

1. Infrastructure-as-a-service (IaaS): where cloud providers deliver computation resources, storage and network as an internet-based services. This service model is based on the virtualization technology. Amazon EC2 is the most familiar IaaS provider.

2. Platform-as-a-service (PaaS): where cloud providers deliver platforms, tools and other business services that enable customers to develop, deploy, and manage their own applications, without installing any of these platforms or support tools on their local machines. The PaaS model may be hosted on top of IaaS model or on top of the cloud infrastructures directly. Google Apps and Microsoft Windows Azure are the most known PaaS.

3. Software-as-a-service (SaaS): where cloud providers deliver applications hosted on the cloud infrastructure as internet-based service for end users, without requiring installing the applications on the customers' computers. This model may be hosted on top of PaaS, IaaS or directly hosted on cloud infrastructure. SalesForce, CRM is an example of the SaaS provider.

As per Zhang, (2010) enterprises can choose to deploy applications on Public, Private or Hybrid clouds. These are described as follows:

1. Public clouds: A cloud in which service providers offer their resources as services to the general public. Public clouds offer several key benefits to service providers, including no initial capital investment on infrastructure and shifting of risks to infrastructure providers.

2. Private clouds: Also known as internal cloud, and they are designed for exclusive use by a single organization. A private cloud may be built and managed by the organization or by external providers. A private cloud offers the highest degree of control over performance, reliability and security.

3. Hybrid clouds: A hybrid cloud is a combination of public and private cloud models that tries to address the limitations of each approach. In a hybrid cloud, part of the service infrastructure runs in private clouds while the remaining part runs in public clouds. Hybrid clouds offer more flexibility than both public and private clouds.

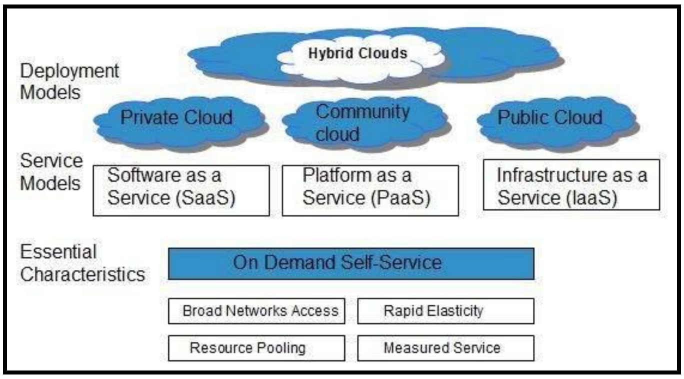

Figure 5: Cloud Computing

5. CONCLUSION

This paper described a systematic flow of survey on Hadoop role in context of cloud computing. The key issues, including Hadoop architecture, HDFS, MapReduce, Hadoop Ecosystem and Cloud Computing environment is described. Hadoop is widely used for large scale data processing in cloud platforms. It is an open-source implementation of framework which hides the complexity of parallel execution across hundreds of servers in a cloud environment. It allows developers to process terabytes of data. After conducting a comprehensive review, this paper concludes that, on cloud platform, Hadoop has been proven to be a useful tool for distributing the processing over as many processors as possible. In Future, Hadoop will become the first choice for cloud computations.

REFERENCES

[1] Alam, M., & Shakil, K. A. (2016). Big Data Analytics in Cloud environment using Hadoop. *arXiv preprint arXiv:1610.04572*.

[2] Gupta, N., & Saxena, K. (2014). Cloud computing techniques for big data and hadoopimplementation. *International journal of Engineering Research & Technology*, *3*(4), 722-726.

[3] Ikhlaq, S., & Keswani, B. (2016). Computation of Big Data in Hadoop and Cloud Environment. *IOSR Journal of Engineering*, *6*(1), 31-39.

[4] Patil, A. U., Patil, R. U., Pande, A. P., & Patil, B. S. Secured Hadoop as A Service Based on Infrastructure Cloud Computing Environment.

[5] Tripathi, A. K., Agrawal, S., & Gupta, R. D. (2018). a Comparative Analysis of Conventional Hadoop with Proposed Cloud Enabled Hadoop Framework for Spatial Big Data Processing. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, *45*, 425-430.

[6] Bashir, B. (2017). An Approach of MapReduce Programming Model For Cloud Computing. *International Journal of Advanced Research in Computer Science*, *8*(2).

[7] Ansari, S. M., Chepuri, S., & Wadhai, V. (2015). Efficient Map Reduce Model with Hadoop Framework for Data Processing.

[8] Lu, H., Hai-Shan, C., & Ting-Ting, H. (2012, October). Research on Hadoop cloud computing model and its applications. In *2012 third international conference on networking and distributed computing* (pp. 59-63). IEEE.

[9] Bashir, B. (2017). An Approach of MapReduce Programming Model For Cloud Computing. *International Journal of Advanced Research in Computer Science*, *8*(2).

[10] Malhotra, S., Doja, M. N., Alam, B., & Alam, M. (2018). Generalized Query Processing Mechanism in Cloud Database Management System. In *Big Data Analytics* (pp. 641-648). Springer, Singapore.

[11] Kumari, S., 2014. A Review Paper on Big Data and Hadoop. International Journal of Scientific and Research Publications 4, 2250–3153

[12] Kaur, G., Kaur, M., 2015. Review Paper On Big Data Using Hadoop. International Journal of Computer Engineering & Technology (IJCET) 6, 65–71.

[13] Peter Mell and Timothy Grance, “The NIST Definition of Cloud Computing” (draft), http://www.nist.gov/customcf/get_pdf.cfm?pub_id=909616, accessed November 12, 2015.

[14] Zhang, Q., Cheng, L., & Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications*, *1*(1), 7-18.

[15] Almorsy, M., Grundy, J., & Müller, I. (2016). An analysis of the cloud computing security problem. *arXiv preprint arXiv:1609.01107*.