MOBILE CLOUD : COMPARATIVESTUDYOF BIG DATA ACCESSING TECHNOLOGIES

[1]Dr. Archana Sharma, [2]Ms. Shweta Singh
[1]Associate Professor, [2]Assistant Professor
[1]SOIT IMS, Noida, India [2]SOIT IMS, Noida, India

Abstract : *With the advancement of mobile network technologies (4G/3G) and with an efficient mobile network infrastructure, huge volume of mobile data may be accessed. Hence, due to such a high growth in the mobile industry, research is being pushed to study how big data processing could be achieved among mobile phones since the number of users accessing data is growing with time. Large data storage are required to serve high volume transactions of data each day when users request data. Hence intelligent methods are needed to solve the insufficient data storage experienced by certain providers. This research elaborates and compares big data technologies which would be deployed in mobile devices and have been shifting towards mobile cloud computing. With new technologies and tools coming to sort big data problems, this paper would help in identifying better options for big data processing on mobile cloud.*

Index Terms-: Hadoop, MapReduce, MCC, HDFS, Spark, Hyrax, Big Data,

I. INTRODUCTION

Today the world is of mobile devices where every person carries a mobile device with him either its tablet pc, Smartphone, etc. Mobile devices has become an integral part of human life. With technology advancement everyone need information quickly. Recent trends have shown us growth in mobile traffic up to 63% [1].Hence there is severe need of technology in accessing data anywhere as everyone is busy in job and moving around concerning work. So, its solution is in Mobile Cloud Computing (MCC). As an inheritance and emergence of Cloud Computing and Mobile computing, Mobile Cloud Computing has been devised as new phenomenon since 2009. Mobile cloud genuinely addresses the problem of managing data while on moving. Mobile cloud assists in processing data outside the device which provide a great assistance to manage data securely. From simpler perspective, MCC could be thought as, an infrastructure where data storage and processing would be moved from the mobile device to powerful & centralized computing platforms located in the cloud. Examples of Mobile Cloud Computing are email, twitter, Google drive, face book, etc. By integrating cloud computing into the mobile environment, mobile cloud computing overcomes the obstacles related to the performance (e.g., battery life, storage and bandwidth), environment (e.g., heterogeneity, scalability and availability) and security [2].

II. MOBILE CLOUD COMPUTING

''Cloud computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data centers that provide those services'' [3]. A cluster of computer software and hardware that offer the services to the general public, probably for a price, forms a 'public cloud'. Therefore Computing is offered as a utility much as electricity, water, gas etc. where only you pay per use. As an example, Amazon's Elastic cloud, Google's App Engine, Microsoft's Azure platform, and Salesforce are a few public clouds which are available today. Cloud computing, however, does not include 'private clouds' that refer to data centers internal to an organization. Henceforth, cloud computing could be defined as the aggregation of computing as a utility & software as a service. Virtualization of resources forms a key requirement for a cloud provider as it is needed by statistical multiplexing that is needed for scalability of the cloud, and also for creating the illusion of infinite resources to the cloud us.

Mobile cloud computing is also based upon the basic cloud computing concepts. As discussed by Mei et al. in [4] there are certain requirements that need to be met in a cloud such as an adaptability, scalability, availability and self-awareness. Also unlike conventional mobile computing technologies, the resources in mobile cloud computing are virtualized and assigned in a group of numerous computers. Mobile Cloud Computing is a combination of Cloud Computing and Mobile technology with the help of internet, which has developed over time from reading emails to learning center on phones using applications to entertainment. With increase in the number of mobile phones and usage of internet by each user small amounts to large datasets which could help companies to make products better and give benefits onto different users as per their needs. Features as GPS, Gyroscope, Accelerometer, and Digital Compass are to be considered when Mobile Cloud Computing are talked about as these features could bring many applications to the table. However there exist a few challenges that exist in Mobile Cloud Computing which brings down the efficiency when deployed to it. Unlike conventional mobile computing technologies, the resources in mobile cloud computing are virtualized and assigned to a group of many distributed computers rather than servers or local computers. Many applications based on Mobile Cloud Computing, such as Google's Gmail, Maps and Navigation systems for mobile, Voice

Search, and some applications on Android platform, MobileMe from Apple, LiveMesh from Microsoft and Motoblur from Motorola have been developed and served to users [5].

In addition to several great benefits in using the mobile cloud computing, there are still a few limitations like the delays encountered when the mobile devices access the cloud services from bigger distances which are mainly from/due to the mobile devices.

III. MOBILE CLOUD ARCHITECTURE

In mobile cloud computing or MCC the mobile network and cloud computing are combined thereby providing optimal services for mobile clients. Cloud computing exists when data and task are kept on individual devices. Applications are run on a remote server and are then sent to the client. The mobile devices, here, are connected to the mobile networks through the base stations; they will establish and control the connections (air interface) and functional interfaces between the mobile devices and mobile networks. Mobile users send service requests to the cloud through desktop application or a web browser. The information's are transmitted to the central processors that are connected to the servers providing mobile network services like Authentication, Authorization and Accounting(AAA) which could be provided to the users based upon Home Agent (HA) and subscriber's data stored in databases. In a broader sense a general architecture is presented in Figure 1.

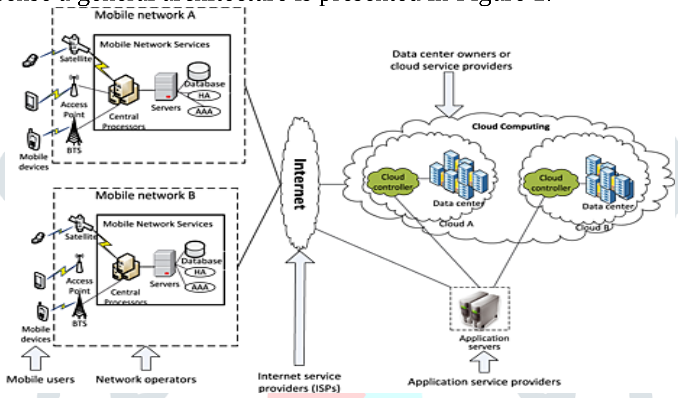

Figure 1: Mobile Cloud Architecture

The subscribers' requests are thereafter delivered to a cloud via the internet. Whereas cloud controllers present in the cloud, process the requests in order to provide the mobile users with the corresponding cloud services. These services are developed based upon the concepts of virtualization, utility computing, and service-oriented architecture. The basic function of a cloud computing system is storing data on the cloud and by using technology, the client access that data. Many business models have rapidly evolved to harness this technology by providing programming platforms, software applications, data-storage, computing infrastructure and hardware as services.

3.1 Advantages of Mobile Cloud Computing

Cloud computing is a promising solution for Mobile Computing because of many reasons as mobility, communication, and portability. It is described below as to how the cloud could be used to overcome obstacles in Mobile Computing, thereby bringing out the advantages of Mobile Cloud Computing

1. ***It extends battery lifetime***. Battery is one of the prime concerns for mobile devices. Computation offloading technique is proposed with the goal to migrate the large computations and complex processing from resource-limited devices like mobile devices to resourceful machines as servers in clouds. It avoids taking a long application execution time on the mobile devices that results in large amount of power consumption.

2. ***It improves data storage capacity and processing power***. Storage capacity is as well a constraint in case of mobile devices. Mobile Cloud Computing has been developed to enable mobile users to access/store the large data on the cloud through wireless networks. First example is the Amazon Simple Storage Service [7] which supports file storage service. Another example is Image Exchange which utilizes the large storage space in clouds for mobile users [8]. The mobile photo sharing service enables mobile users to upload images to the clouds instantly after capturing. Mobile users could access all images from any devices. With the cloud, the mobile users can save considerable amount of energy and storage space on their mobile devices since all images are sent and processed on the clouds. Flicker [9] and ShoZu [10] are also the successful mobile photo sharing applications based on MCC. Facebook [11] is the most successful social network application today, and it is also a typical example of using cloud in sharing images.
3. ***It improves reliability***. Storing data or running applications on clouds is an effective means to improve the reliability since the data & application are stored and are backed up on a no. of computers. It reduces the chance of data and application lost on the mobile devices. In addition, Mobile Cloud Computing could be designed as a comprehensive data security model for both service providers and mobile users. Besides, the cloud may be remotely provided to users with security services like malicious code detection, virus scanning, and authentication .

IV. BIG DATA IN MOBILE CLOUD COMPUTING

The quantity of data generated annually over the Internet has surpassed the zetabyte levels. Processing data with such large volume far exceeds the computational capabilities of present datacenters and computers, giving rise to the term Big Data. Although the top supercomputers are able to handle Big Data analysis, their highly-specialized designs are not affordable for commercial use while instead, large commodity computer clusters are used, where interconnect speeds are limited and faults are common. Also the storage and management of Big Data poses different challenges: Whereas the storage has to be performed by high availability and high-performance distributed file systems, also it need be done in a way so as to permit application of efficient ata analytics later. Also being able to perform analytics on this data is crucial.

Big Data in Mobile Cloud Computing describes a collection of data from mobile users that is stored in the cloud, mobile users can request these data at anytime from anywhere and hence a good data management system is required owing to high data transfer. While billions of data transactions and streams come from devices; one of the major challenges for current data management is to provide a service without any loss and with low throughput latency. However, after enabling and integrating a cloud management system, there are still difficulties serving all data streams and transactions [12].

The storage of Big Data is performed by file systems which are drastically different than traditional file systems such as NT File System, in short, NTFS. One such user-level distributed file system – GFS or Google File System permits not only the distributed storage of Big Data, but also its access with high availability and fault tolerance owing to the built-in redundancy in Google File System. This file system also states how the processing should be performed: Standardized methods like MapReduce ease the handling of Big Data and provide for a tool for cloud operators to make their platform more accessible.

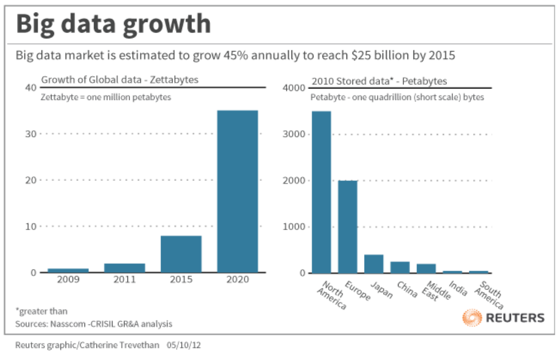

Figure 2: Big Data growth from 2009 to 2020 [13]

V. ACCESSING BIG DATA THROUGH MOBILE DEVICES

While accessing Big Data in the cloud through mobile devices, mobile-cloud computing is the key enabling technology in this process and with the expansion of mobile applications and the support of cloud computing for different services for mobile users, mobile-cloud computing has been introduced and considerably investigated as an integration of cloud computing into the mobile environment Mobile-cloud computing(MCC) facilities for mobile users to take full advantage of cloud computing and enables access to Big Data anywhere at any time. In the last decade, mobile devices have become increasingly more powerful to handle most of the daily operations but are not powerful enough for data-intensive computations as querying and analyzing the Big Data. However, considering huge amount of mobile devices and rapid development of wireless networks, a loosely organized cluster of mobile devices could be powerful enough so as to collectively handle quite heavy computations together with the cloud. By forming an integrated computing system, while simultaneously maintaining the energy efficiency, some of the technologies involved in big data processing are Hadoop MapReduce, Spark, Hyrax, Hive etc.

VI. COMPARISON AMONG HADOOP MAPREDUCE, HYREX AND SPARK

6.1. MapReduce

MapReduce has been a popular framework for processing and generating large datasets in parallel over a cluster as introduced by Google whereas Hadoop as an open source implementation of MapReduce has been successfully applied in many applications as web indexing, report generating, log file analysis, data mining. The MapReduce system runs on the top of Hadoop Distributed File System, in short, HDFS, in which data is loaded and partitioned into chunks while each chunk replicated across multiple machines.

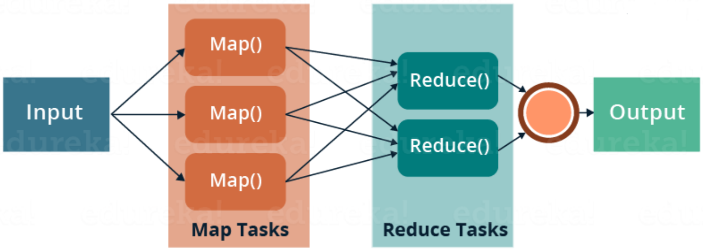

Figure 3 : MapReduce Framework

Also the MapReduce runtime system handles splitting the input data, scheduling map and reduce tasks, besides transferring input and output data to the machines running the tasks. Jobs are managed by a master which assigns tasks to slave machines and also, provides the locations of intermediate values to reduce tasks. Computation on the machine where upon the input data is already stored is preferred in order to reduce network transfers. Large data transfers are performed directly between the machine where the data is stored and the machine which needs the data. Data transfers between machines on the same rack are preferred to transfers between machines that are more far from each other in the network.

6.2 Hadoop

Hadoop is an open source software, which permits distributed processing of large data sets across multiple clusters of computers with simple programming model. Hadoop mainly comprises Hadoop Distributed File System and MapReduce (distributed computing system). Hadoop is at present extensively used in the Internet, and has attracted universal attention from research society [14].

6.2.1 Hadoop Distributed System

HDFS[15] is a very huge distributed file system which provides us fault tolerance and also has high outputs. Hadoop Distributed File System stores and saves the files as a sequence of blocks and then it replicates these data blocks for fault tolerance. These gigantic data sets give the global access to files in clusters computers. Hadoop Distributed File saves metadata on a particular and dedicated server, named as Name Node. Application data is stretched on another network named as Data Node. All these servers are completely connected and they communicate with each other by using TCP-IP based protocols.

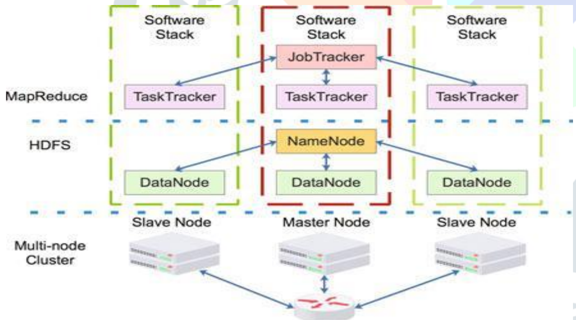

Figure 4: HDFS Architecture

Hadoop Distributed File architecture consists of four parts:

1. **Name Node:** This is in charge of managing all the metadata and file system activities. This manages the file system namespace like open, close and rename both on file and directory. This, also, creates decisions concerning replication of the blocks and has to maintain the tree of namespace besides maps the file blocks to DataNodes i.e. physical location of file's data. A single NameNode is considered a bottleneck for managing requests in scientific environments.

2. **Data Node**: This saves the data in the Hadoop Distributed File System. All DataNode saves the data blocks on account of local or the remote patrons and each block is stored as a separate file in the local file system of the node. While on starting-up, DataNode is connected to the NameNode and then creates a handshake. The purpose of handshake is to authenticate the namespace ID and the version of software used by DataNode. If NameNode gets matched with DataNode, the DataNode shuts down automatically. Whereas when the handshake is victorious, the DataNode gets registered with NameNode. DataNodes continuously stores their unique storage IDs which is used as an internal identifier for the DataNode and which makes it as identifiable even if this is started with an altered IP address or a port. The storage ID is attached to the DataNode when this is registered with the NameNode at the very first time and then never changes later. This DataNode then provides response to the requirements that are approaching from the NameNode, for operations of file system. Thereafter this DataNodes provide service of read, write and file replication requests that are based on the way coming from NameNode.

3. **Job Tracker:** It speaks to the NameNode so as to establish the location of the data. While JobTracker schedules every single map reduces or intermediate reconciling operations to the specific machines and checks the success and losses of these individual (single) tasks. This works to finish the entire batch job. In case the task fails, the JobTracker then automatically relaunchs the task, possibly on a changed node, up to a certain amount of retries.

4. **TaskTracker:** JobTracker is considered the prime master for supervision of overall implementation of a MapReduce job whereas the TaskTrackers administers the execution of individual i.e. single task on every slave node. Although there is a single TaskTracker/slave node. Each TaskTracker can produce or have multiple Java Virtual Machines, in short, JVMs so as to handle multiple map or decrease tasks in parallel. The TaskTracker transmits major messages to the JobTracker, typically after every few minutes, to restore confidence that JobTracker is still a live and working [16] .

6.2.2 Hadoop Drawbacks

Despite several advantages, Hadoop is less ideal for some mobile aspects of mobile-cloud computing. It implements much of the functionality which most platform require, but it does not cover all of the requirements. It is just because Hadoop was designed and implemented with commodity server hardware than resource constrained hardware. The major drawbacks are as follows:

a. **Not conservative in CPU and memory usage:** One of the problems is that Hadoop is not conservative in Central Processing Unit (CPU) and memory usage. While Hadoop was designed for I/O bound jobs i.e. for those in which reading, writing, and transferring data are the most time-consuming operations. Also Hadoop's liberal use of CPU and memory is exemplified by numerous aspects of its codebase.

b. **Not well suited for mobile devices:** Hadoop as well uses technologies which are not well-suited for mobile devices. For example, it uses XML extensively that is notoriously expensive to parse. Besides it also uses servlets to serve intermediate results, even though a light-weight custom HTTP server would need less overhead. JSPs, which need dynamic compilation, are used for providing a monitoring interface to DataNodes and TaskTrackers. This particular inefficiency is magnified on a mobile device.

Hadoop has also some other drawbacks as:

i. It requires a very high memory and large storage to perform replication technique.
ii. Single master (Name Node) which requires care.
iii. It is designed for batch processing.
iv. It chains allotment of tasks only and does not have any plan to support the scheduling of tasks.

6.3 Spark

Hadoop MapReduce framework is mainly designed for batch processing which makes it less suitable for ad-hoc data exploration, machine learning processes. Whereas Apache Spark is an increasingly popular alternative to replace MapReduce with a better performant execution engine but still use Hadoop HDFS as storage engine in case of large data sets. Apache Spark is an open source cluster computing framework in case of real-time data processing. The prime feature of Apache Spark is its in-memory cluster computing which increases the processing speed of an application. Spark provides an interface for programming, entire clusters with implicit data parallelism and fault tolerance and is designed to cover a wide range of workloads as batch applications, iterative algorithms, streaming and interactive queries.

Spark has numerous advantages as compared to other big data and MapReduce technologies such as Hadoop and Storm and it gives us a comprehensive, unified framework to manage big data processing requirements with a variety of data sets which are diverse in nature like text data, graph data etc as well as the source of data i.e. batch v. real-time streaming data. It enables applications in Hadoop clusters to run up to 100 times faster in memory while 10 times faster even when running on disk. Besides to Map and Reduce operations, it supports SQL queries, machine learning, streaming data, and graph data processing. Developers could use these capabilities as stand-alone or combine them to run in a single data pipeline use case.
Hadoop as a big data processing technology has been for over 10 years and has proven to be the able solution of choice for processing large data sets whereas MapReduce is a great solution for one-pass computations, but is not very efficient for use cases which require multi-pass computations and algorithms. Each of the steps in the data processing workflow has one Map phase and one Reduce phase and any use case need to be converted into MapReduce pattern to leverage this solution.

As the Job output data between each step has to be stored in the distributed file system before the next step begins. Therefore, this approach tends to be slow due to replication & disk storage. Besides, Hadoop solutions typically include clusters which are hard to set up and manage. This also requires the integration of several tools for different big data use cases as Mahout for Machine Learning and Storm for streaming data processing.

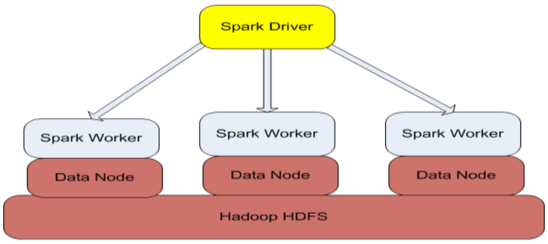

Figure 5: Spark Architecture

Spark permits programmers to develop complex, multi-step data pipelines by using directed acyclic graph pattern which also support in-memory data sharing across DAGs, so that different jobs could work with the same data. The DAG engine assists to eliminate the MapReduce multi-stage execution model and offers significant performance improvements. This runs on top of existing Hadoop Distributed File System infrastructure so as to provide enhanced and additional functionality. Spark should be looked as an alternative to Hadoop MapReduce than a replacement to Hadoop. This is not intended to replace Hadoop but to provide a comprehensive and unified solution to manage different big data use cases and needs. Spark takes MapReduce onto the next level with lesser expensive shuffles in the data processing. With capabilities as in-memory data storage and near real-time processing, the performance could be several times faster in comparison to other big data technologies. Beisdes Spark also supports lazy evaluation of big data queries that helps with optimization of the steps in data processing workflows and provides for a higher level API to improve developer productivity besides a consistent architect model for big data solutions.

Spark holds intermediate results in memory than writing them to disk that is very useful especially when it is needed to work on the same dataset multiple times. This is designed to be an execution engine which works both in-memory and on-disk. Spark operators perform external operations when data does not gets fit in the memory. Also Spark can be used for processing datasets that are larger than the aggregate memory in a cluster and Spark would attempt to store as much as data in memory and then shall spill to disk. It could store part of a data set in memory and the remaining data on the disk. Data need be looked at for use cases to assess the memory requirements. Spark comes with performance advantage with this in-memory data storage.

6.3.1 Spark Limitations

a. Latency: Apache Spark has higher latency.
b. Expensive: In-memory capability could become a bottleneck when cost-efficient processing of big data is needed as keeping data in memory is quite expensive, the memory consumption is very high, and which is not handled in a user-friendly manner. Apache Spark needs lots of RAM to run in-memory, thus the cost of Spark is very high.
c. Manual Optimization: The Spark job needs to be manually optimized and is adequate for specific datasets. If partition is required and cache in Spark to be correct, it need be controlled manually.
d. Problem with Small Files: If Spark with Hadoop is used, a problem of a small file is there. HDFS provides a limited no. of large files than a large no. of small files.
e. No File Management : Apache Spark does not have its own file management system. Therefore it relies upon some other platform as Hadoop or another cloud-based platform which is one of the Spark known issues.

6.4 Hyrax

The goal of Hyrax is to develop a mobile cloud infrastructure which enables smartphone applications with distributed data and computation. Hyrax permits applications to conveniently use data & execute computing jobs upon smartphone networks and heterogeneous networks of phones and servers. Research has been focused primarily on implementation and evaluation of mobile cloud computing infrastructure based upon MapReduce.

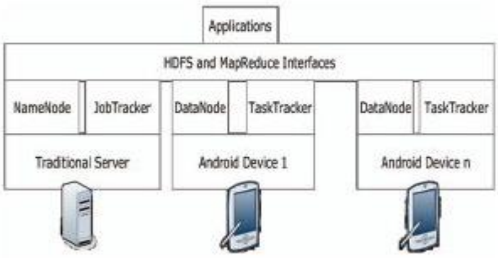

Figure 6: Hyrax Infrastructure

In Hyrax, every machine runs one instance of NameNode & one instance of JobTracker while DataNode and TaskTracker are run on each phone in separate Android service processes within the same application. Android applications consist of multiple processes, some of which run as background services. Since DataNode and TaskTracker are run as Android services, they could run in the background of other applications. A thread is also spawned to record information concerning the systemload i.e. which includes power level and CPU, network, memory, and disk Input/Output statistics — into the local file system. Within the application, a server is run to permit external scripts so as to control data uploading, kill the program, and then check the program status.

6.4.1 Hyrax Limitations

Currently, Hyrax only works on the Android platform. Whereas this enables a substantial proportion of the mobile users population to make use of Hyrax, this still leaves out a significant proportion of mobile users of other platforms such as the iPhone or BlackBerry. This would be beneficial to develop Hyrax also for these other platforms so that mobile clouds can be created that are based not only on different mobile devices, but also on different mobile operating systems and platforms as well. By this way, Hyrax would genuinely be working on a heterogeneous cloud.

CONCLUSION

Mobile Cloud Computing or MCC is a combination of Cloud Computing and Mobile technology with the assistance of the internet and which has progressed over time from reading emails onto playing games to still further learning center on phones with the help of applications,screen size, the accessibility to a graphical user interface like viewing an image and working upon multiple screens. The large amount of data generated by the devices and internet sources on a regular basis constitutes big data. In order to find the difference, understand and compare big data technologies that could be deployed onto mobile devices, this research has made a comparative study on technologies and recent trends which have taken place in last few years, which have been shifting towards mobile computing. All of the above technologies discussed above have some advantages over others but all of them work on Mobile Cloud computation that makes work portable as compared to the conventional manner of working on the desktop.

REFERENCES

[1] Mobile Traffic: Date accessed: 03/02/2016: Available from http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobilewhite-paper-c11-520862.html

[2] Pragya Gupta, sudha Gupta “Mobile Cloud Computing-The Future of Cloud”, International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. Vol. 1, Issue 3, September 2012.

[3] L. Mei, W. Chan, T. Tse, A tale of clouds: paradigm comparisons and some thoughts on research issues, in Proceedings of the Asia-Pacific Services Computing Conference, APSCC’08, IEEE, 2008.

[4] M. Satyanarayanan, “Mobile computing: the next decade,” in ACM Workshop on Mobile Cloud Computing Services: Social Networks and Beyond, 2010.

[5] Hoang T. Dinh, Chonho Lee, DusitNiyato, and Ping Wang, “A survey of Mobile Cloud Computing: Architecture, Applications and Approaches”, Wireless Communications and Mobile Computing, 2011.

[6]. Xia Tang, et al. A Reduce Task Scheduler for MapReduce with Minimum Transmission Cost Based on Sampling Evaluation International Journal of Database Theory and Application Vol.8, No.1 (2015), pp.1-10.

[7] http://aws.amazon.com/s3/

[8] Vartiainen E, Mattila KV-V.User experience of mobile photo sharing in the cloud. In *Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia (MUM)*,2010.

[9] http://www.flickr.com/

[10] http://www.shozu.com/portal/index.do.

[11] http://www.facebook.com/

[12] Abdul Kadir, E., Shamsuddin, S.M., Abdul Rahman, T., and Ismail, A.S. 2015. Big Data Network Architecture and Monitoring Use Wireless 5G Technology, Int. J. Advance Soft Compu. Appl, 7(1), 1-14.

[13]. Reuters , Stastitics Graph-Big Data growth 2099 to 2020

[14] “Hadoop Online Training”, 2015, http://es.slideshare.net/smconsultantdaniel/hadoop-online-training-at-s-m-consultant.

[15] Chandarana, P., & Vijayalakshmi, M., 2014, “Big Data Analytics Framework”, Proceedings of International Conference on Circuits, System, Communication and Information Technology Applications IEEE.

[16] “Hadoop Tutorials”, 2014, http://www.authorstream .com/Presentation/smconsultantdaniel-2799775-hadoop-tutorial/