Improving Processing Time of Big Data Analytic on Mobile Cloud Computing

NguWah Win
University of Computer Studies, Yangon
nguwahwin@ucsy.edu.mm

Abstract

The growth of big data is a result of the variety of data through user generated data from many devices. Today, big data analytics, extracting data knowledge from large datasets, is shifted from personal desktop computer to mobile devices because of its anywhere-anytime access facilities. Nevertheless, mobile devices still have many resources limitations. Mobile cloud computing is a best solution for these problems, because it can support infinite computing power for mobile devices. This paper proposes a new big data analytic platform on mobile cloud computing with efficient processing time. In proposed platform, RESTful web service technology and MapReduce Transformation Model are applied to produce high query processing performance with seamless connectivity. To reduce the communication cost between web service and mobile device, we use JSON format output result. According to the results, we conclude that the communication cost of proposed platform is better than traditional way.

Keywords*: Big data, Big data analytics, RESTful, MapReduce, JSON, Mobile Cloud Computing*

1. Introduction

The Internet generates the largest amount of data and it has exceeded the zetabyte levels. Processing the high volume of data is beyond the computational capabilities of traditional data warehouses, giving rise the term Big Data. Cloud computing is the powerful platform because of their well-known services. It can give many advantages to users by allowing them to use infrastructure, platforms and software by cloud providers at low cost and elastically in an on demand fashion.

Mobile cloud computing is the technology for accessing data from cloud storage by using mobile web services. It provides mobile users with data storage and processing services in clouds, providing the need to have a powerful device configuration (e.g. CPU speed, memory capacity etc), as all resource-intensive computing can be performed in the cloud. Android is the most widely using mobile Operating System and which is based on the Linux Kernel and developed by Google [1].

Because of this stream of technology requirements, many researches emphasize to integrate mobile device and big data analysis to gain the business facilities by using mobile web services. Mobile web services allow deploying, discovering and executing of web services in a mobile communication environment using standard protocol. Web service can be classified into two main categories: RESTful and SOAP-based web services.

In big data analytic, Hadoop is becoming the core technology to solve the business problem for large organizations with cloud storage. The server level architecture for Big Data consists of parallel computing platforms that can handle the associated volume and speed. Clusters or grids are types of parallel and distributed systems, where a cluster consists of a collection of interconnected stand-alone computers working together as a single integrated computing resource, and a grid enables the sharing, selection, and aggregation of geographically distributed autonomous resources dynamically at runtime [2]. A commonly used architecture for Hadoop consists of client machines and clusters of loosely coupled commodity servers that serve as the HDFS distributed data storage and MapReduce distributed data processing.

The MapReduce is the programming model for data processing. It operates via regular computer that uses built-in hard disk, not a special storage. Each computer has extremely weak correlation where expansion can be hundreds and thousands of computers. Since many computers are participating in processing, system errors and hardware errors are assumed as general circumstances, rather than exceptional. With a simplified and abstracted basic operation of Map and Reduce, many complicated problems can solve. Programmers who are not familiar with parallel programs can easily perform parallel processing for data. It supports high throughput by using many computers. As the core technology of the

Hadoop is the MapReduce parallel processing model, all of the high level query languages that run on Hadoop are the MapReduce based query languages.

A number of HLQLs have been constructed on top of Hadoop to provide more abstract query facilities than using the low-level Hadoop Java based API directly. Pig, Hive, and JAQL are all important HLQLs. Programs written in these languages are compiled into a sequence of MapReduce jobs; to be executed in the Hadoop MapReduce environment. Apache Hive [3, 4, 5] is an open-source data warehousing solution built on top of Hadoop. Hive provides an SQL dialect, called Hive Query Language (HiveQL) for querying data stored in a Hadoop cluster. Apache Pig [6, 7] provides an engine for executing data flows in parallel on Hadoop. It includes a language, PigLatin, for expressing these data flows. PigLatin includes operators for many of the traditional data operations, as well as the ability for users to develop their own functions for reading, processing, and writing data. Jaql [8] is a declarative scripting language for analyzing large semistructured datasets in parallel using Hadoop'sMapReduce framework. It consists of a scripting language and compiler, as well as a runtime component for Hadoop. It is extremely flexible and can support many semistructured data sources such as JSON [9], XML, CSV, flat files and more.

This paper presents the big data analytic platform for mobile device with different OS and it organized as follows. Section (1) briefly introduces about the research and Section (2) discuss about the related work of the research. Section (3) presents the theory and concept of the mobile cloud computing and big data analytic and Section (4) describe about our proposed platform. Finally, Section (5) is the last sections and there we conclude our paper with experimental results.

2. Related Work

There are many types of existing big data analytic platforms for large scale data. Most of them based on MapReduce, distributed file system, and no-SQL indexing. Tableau is known for its strong visualization features, which can support exploratory or discovery analytics [10]. Whether data sources are structured or unstructured, batch or streaming, large or small, SAP invariably puts forward its Hana in-memory platform as the answer to all needs -- whether analytic or transactional. When data is truly big or unstructured, SAP supports various Hadoop distributions, withHana accessing data through Hive. When data needs to be archived for long-term historical analysis, SAP IQ (formerly Sybase IQ) offers a compressed, columnar DBMS adapted to support MapReduce processing as a SQL-based alternative to Hadoop.Hana has a built-in predictive analytics library, R language support, spatial processing, natural language processing, and text analytics libraries. If need be, text and unstructured data analyses can be pushed down into Hadoop using SAP Data Services. Result sets can be returned to Hana for fast, in-memory analysis.

The Vertica Analytics platform has a high-speed, relational SQL DBMS purpose-built for analytics and business intelligence. Vertica has helped over 300 customers monetize their data in unique ways, including Zynga, JP Morgan, Verizon, Comcast, Vonage, Blue Cross Blue Shield, and others. The Vertica Analytics platform offers a shared-nothing, MPP column-oriented architecture, and has been benchmarked by many customers as being on average 10x to 200x faster than other solutions. It also uses compression very aggressively, both of data on disk and on data "in motion" during queries, which further enhances query speed while enabling cost-effective storage management. The Vertica Analytics Platform runs on clusters of inexpensive, industry-standard Linux servers and requires limited resources up front for setup and performance configuration. Unlike most solutions in this space, Vertica was purposely built from the ground up for today's most demanding analytics challenges.

The main challenges of big data analytics are performance, scalability and fault tolerance. To address these challenges, a big data platform for large scale data analysis by using HadoopMapReduce framework and GlusterFS over distributed scale-out storage system is proposed [11]. There are four layers in this big data platform. The first layer is application layer which consists of GlusterFS clients, Apache Hive, Apache Pig, and Jaql. The second layer is processing layer and MapReduce framework plays the main role in this layer. The third layer is interface layer where enhanced HadoopGluster connector connects the HadoopMapReduce with GlusterFS. The last layer is storage layer which consists of Gluster storage pool to store big data. This platform tackles the major issues. They are large number of file migrations in data rebalancing process GlusterFS, large amount of file migration time, and inefficient storage utilization.The first and second issues are solved by using consistent hashing algorithm with virtual nodes and that "virtual node concept" is borrowed from Dynamo. The last issue is also solved by migrating virtual node between

storage servers to rebalance data among these storage servers.

Our mobile cloud platform for big data analytic provides a solution to reduce the query processing time of traditional query languages by using MapReduce transformation process. To achieve the seamless connectivity between mobile and cloud storage, we used RESTful web service technology. By using this platform, users send a request from their mobile device and get back the results without noticeable amount of time.

3. Mobile Cloud Computing and Big Data Analytics Concepts

Big data and analytics require large amounts of data storage, processing, and interchange. The traditional platforms for data analysis, such as data warehouses, cannot easily or inexpensively scale to meet big data demands. Furthermore, most of the data is unstructured and unsuitable for traditional relational databases and data warehouses. Platforms to process big data require significant up-front investment. The methods for processing big data rely on parallel-processing models, such as MapReduce, in which the processing workload is spread across many CPUs on commodity compute nodes. The data is partitioned between the compute nodes at run time, and the management framework handles inter-machine communication and machine failures. The most famous embodiment of a MapReduce cluster, Hadoop, was designed to run on many machines that don't share memory or disks (the *shared-nothing* model). Alternatively, cloud computing is the perfect vehicle to scale to accommodate such large volumes of data. Cloud computing can provide cost efficiencies by using commodity compute nodes and network infrastructure, and requiring fewer administrators and programmers. So it seems that a cloud computing environment is well-suited for big data, provided the shared-nothing model can be honored [17].

3.1. Mobile Cloud Computing

Mobile cloud computing (MCC) at its simplest, refers to an infrastructure where both the data storage and data processing happen outside of the mobile device. Mobile cloud applications move the computing power and data storage away from the mobile devices and into powerful and centralized computing platforms located in the clouds, which are then accessed over the wireless connection based on a thin native client. Improving data storage capacity and processing power: it enables mobile users to store/access large data in the cloud and helps to reduce the running cost for compute intensive applications [12].

3.2. Big Data Analytics

Big data analytics is the process of examining large data sets containing a variety of data types to uncover hidden patterns, unknown correlations, market trends, customer preferences and other useful business information.

There are two main techniques for analyzing big data: the store and analyze, and analyze and store [13]. The store and analyze integrates source data into a consolidated data store before it is analyzed. The advantages of this are improved by data integration and data quality management, plus the ability to maintain historical information. The disadvantages are additional data storage requirements and the latency introduced by the data integration task.

Analyze and store technique analyzes data as it flows through business processes, across networks, and between systems. The analytical results can then be published to interactive dashboards and published into data store for user access, historical reporting and additional analysis. This can also be used to filter and aggregate big data before it is brought into a data warehouse.

3.3. Hadoop Distributed File System and MapReduce

The Hadoop distributed file system (HDFS) is designed to store very large datasets reliably and all servers are fully connected and communicate with each other using transmission control protocol (TCP) based protocols. Hadoop MapReduce is a software framework for easily writing applications which process vast amounts of data in parallel on large clusters of commodity hardware in a reliable, fault-tolerant manner. The framework sorts the outputs of the map, which are then input to the reduce tasks. Typically, both the input and the output of the jobs are stored in a file system. The framework takes care of scheduling tasks, monitoring them and re-executing the failed tasks [14].

3.4. RESTful Based Web Service

REST stands for Representational State Transfer: it is a resource oriented technology and it is defined by Fielding in [15] as an architectural style that consists of a set of design criteria that define the proper way for using web standards such as HTTP and URIs. Although REST is originally defined in the context of the web, it is becoming a common implementation

technology for developing web services. RESTful web services are implemented with web standards (HTTP, XML and URI) and REST principles. REST principles include addressability, uniformity, connectivity and stateless. RESTful web services are based on uniform interface used to define specific operations that operate on URL resources.

3.4. JavaScript Object Notation

JavaScript Object Notation (JSON) is a lightweight data-interchange format. It is easy for machines to parse and generate. It is based on a subset of the JavaScript Programming Language, Standard ECMA-262 [18]. JSON is a text format that is completely language independent but uses conventions that are familiar to programmers of the C-family of languages, including C, C++, C#, Java, JavaScript, Perl, Python, and many others. These properties make JSON an ideal data-interchange language.JSON is built on two structures:

- A collection of name/value pairs. In various languages, this is realized as an *object*, record, struct, dictionary, hash table, keyed list, or associative array.
- An ordered list of values. In most languages, this is realized as an *array*, vector, list, or sequence.

These are universal data structures. Virtually all modern programming languages support them in one form or another. It makes sense that a data format that is interchangeable with programming languages also be based on these structures.

4. Proposed Big Data Analytic Platform on Mobile Cloud Computing

Big data can be analyzed as part of advanced analytics disciplines such as predictive analytics, data mining, text analytics and statistical analysis. Statistical analysis is a component of data analytics involves collecting and scrutinizing every single data record in a set of dataset. In our proposed platform, we used US Census statistical dataset and we extract the summary information of user request from mobile device.

Our mobile cloud platform consists of application layer, processing layer and storage layer. The web service contains RESTful web service and which are identified by logical URLs. After that it sends the HTTP request to the cloud storage. The application layer organizes with MRT (MapReduce Transformation Model) and it receives the user request query from web service layer and works the analytical process with processing layer and storage layer. The processing layer contains JobTracker and TaskTracker nodes and which works as MapReduce framework. The Storage layer consists of NameNode and DataNode.The DataNode clusters are used to store the big data.

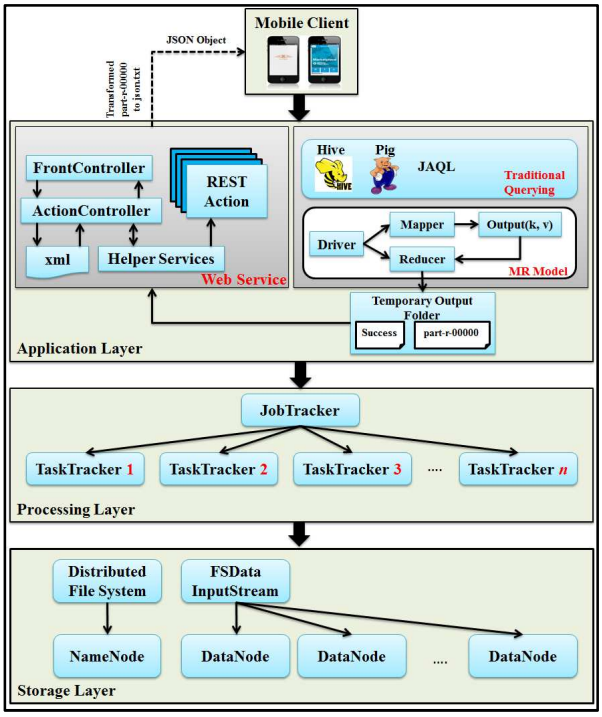

Figure 1. Mobile cloud platform architecture

The processing flow of our proposed system is as follows:

1. The mobile client access the web service with its URL and send the request as a parameter.
2. The web service receives this request and invokes the Hadoop program according to this request parameter.
3. The output result from Hadoop program is accessed by web service and transforms the output into JSON format object.
4. And then extracting the value from JSON object on mobile device.
5. Finally, the output result is display on mobile device with graphical representation.

The traditional analytic method takes a large amount of time to receive the output result. To improve the query processing performance, reducing query execution time focuses on this platform. The query processing time of this proposed platform is reduced by using MapReduce Transformation Model. To decrease he communication cost between mobile device and cloud storage, we used JSON output format.

4.1. Experiment Environment

We implement the mobile platform for big data analytics and evaluate on different Operating Systems and different high level query languages. To build a storage cluster, we created 12 VMs for NameNode, Secondary NameNode, DataNode, JobTracker and TaskTracker. The NameNode in Hadoop is the node where Hadoop stores all the location information of the files in HDFS. In other words, it holds the metadata for HDFS. This information is required when retrieving data from the cluster as the data is spread across multiple machines. The secondary name node is responsible for performing periodic housekeeping functions for the NameNode. It only creates checkpoints of the filesystem present in the NameNode. The DataNode is responsible for storing the files in HDFS. It manages the file blocks within the node. It sends information to the NameNode about the files and blocks stored in that node and responds to the NameNode for all filesystem operations. JobTracker is responsible for taking in requests from a client and assigningTaskTrackers with tasks to be performed. The JobTracker tries to assign tasks to the TaskTracker on the DataNode where the data is locally present (Data Locality). TaskTracker is a daemon that accepts tasks (MapReduce and Shuffle) from the JobTracker. The TaskTracker keeps sending a heartbeat message to theJobTracker to notify that it is alive.

The specifications of devices and necessary software component used in mobile cloud infrastructure, and data set used in MapReduce processing are described in table 1.

Table 1. Experiment Parameters

Parameters	Specification
OS	- Ubuntu 12.04 Linux, - Red Hat Enterprise Linux 6.4
Host Specification	Intel ® Core i7-2600 CPU @ 3.40GHz, Intel ® Core i7-3770 CPU @ 3.40GHz, 8GB Memory, 1TB Hard Disk
VMs Specification	1GB RAM, 50 GB Hard Disk
Mobile Device Specification	Huawei G730-U00, Android OS version 4.2.2 (Jelly Bean), Quad-core 1.3 GHz Cortex-A7, 4GB internal memory
Software Component	- Hadoop 1.1.2 - Hive 0.9.0, Pig 0.12.1, Jaql 0.5.1
Data Set	US census dataset [16] - 114 GB 47 population tables, 14 housing tables, 10 population tables

4.2. Evaluation and Results Discussion

The query processing time of our MapReduce Model is compared with query processing time of other query languages.Sample queries of HiveQL, PigLatin, and Jaql for query processing are shown in Figure 2.

The **HiveQL** (Hive Query Language) is

```
hive> create table population (ID int, FILEID string, STUSAB string, CHARITER string, CIFSN string, LOGRECNO string, POPCOUNT int) row format delimited fields terminated by '\,' stored as textfile;hive> load data inpath '/user/root/Rec250000.csv' overwrite into table population;hive> select STUSAB, sum(POPCOUNT) from population group by STUSAB;
```

The **PigLatin** is

```
grunt> population = load '/user/root/families.csv' using PigStorage(',') as (ID: int, FILEID:
chararray, STUSAB: chararray, CHARITER:
chararray, CIFSN: chararray, LOGRECNO:
chararray, POPCOUNT: int);
grunt>grouped = group population by STUSAB;
grunt> result = foreach grouped generate group, SUM(population.POPCOUNT);
grunt> dump result;
```

The **Jaql** is

```
jaql>$ population = read(del("/user/root/families.csv", { schema: schema { ID: long, FILEID: string, STUSAB: string, CHARITER: string, CIFSN: string, LOGRECNO: string, POPCOUNT: long} }));
jaql> $population -> group by $STUSAB={$.STUSAB} into {$STUSAB, total:sum($[*].POPCOUNT)};
```

Figure 2. Sample queries of high level query languages on families table.

The traditional big data analytic platform performs analytics directly over Hadoop MapReduce framework. All the queries for analytics are executed as Map and Reduce jobs over big data placed into HDFS file system. Hadoop MapReduce processes these

data with user friendly query languages such as HiveQL, PigLatin and Jaql to get analytical results.

In our big data analytic platform, we use MRT model and processes the data that are stored in HDFS on commodity servers to extract information. At this time, we record the total processing time of our proposed platform.

Figure 3 and 4 show the total processing time of traditional big data analytic paltform with three querying methods on Ubuntu OS and Red Hat OS with varied workloads. According to this result, we can show that different numbers of records are used and the Hive query language provides better execution time for querying data than other query languages, Pig and Jaql, on both OS.

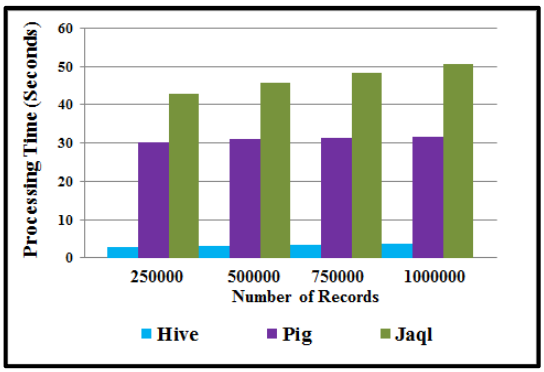

Figure 3.Comparison of processing time of Traditional Methods on Ubuntu OS

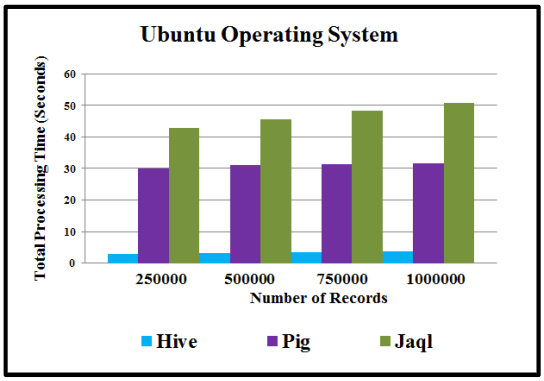

Figure 4.Comparison of processing time of Traditional Methods on Red Hat OS

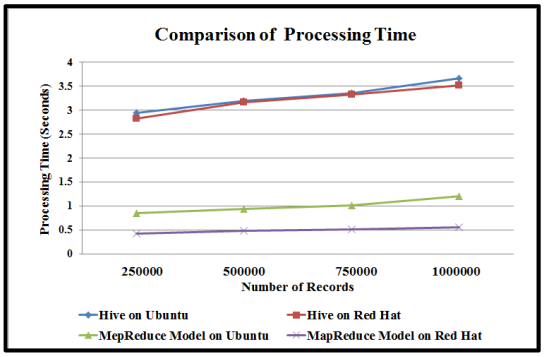

Figure 5.Comparison of processing time of MapReduce Model and Hive on Ubuntu and Red Hat OS

Figure 5 shows the total processing time of our proposed platform with MapReduce Transformation Model on Red Hat and Ubuntu operating system. As a result, we can conclude that the proposed MapReduce Transformation based big data analytic platform can give better performance of processing time on both OS.

Because, other query languages take a large amount of time to transform into the MapReduce model at runtime. By using JSON format output, we can easily extract the value and can show user friendly graphical view on mobile device. For a querying point of view, we can conclude that MapReduce Transformation Model of proposed platform is better than other query languages on both OS. From the operating system point of view, we can also conclude that the Red Hat OS is more convenient than Ubuntu OS for this proposed platform. This proposed platform developed the predefined process that transforms the query into MapReduce programming model. So, it reduces the extra time to transform query language into the MapReduce model and it makes better performance in querying data.

Hadoop was originally a widely adopted implementation of the MapReduce paradigm and HDFS. In our proposed platform, we bring the whole Hadoop infrastructure for into the private cloud deployment model on local network system. The future work of our research is to develop this platform on public cloud deployment model.

5. Conclusion

Today, it is very important to think of big data and analytics together. *Big data* is the term used to describe the recent explosion of different types of data from disparate sources. *Analytics* is about examining data to derive interesting and relevant trends and patterns, which can be used to inform decisions, optimize processes, and even drive new business models. Cloud computing seems to be a perfect vehicle for hosting big data workloads. However, working on big data in the cloud brings its own challenge of reconciling two contradictory design principles. Moreover, the convergence of mobile computing and cloud computing into a single platform has become an efficient platform for big data analysis. In this paper, we implement a new mobile cloud platform for big data analysis. This platform operates with RESTful web service technology to provide seamless connectivity between mobile device and cloud storage. To improve the query performance, we developed a

MapReduce Transformation Model to transform users' requests into MapReduce form. The analytical JSON output result is transferred to the mobile by using RESTful web service technology. As a result, performance evaluations are conducted to prove that the proposed platform provides three times faster than other high query languages in both operating systems.

References

[1] Phandroid.com, 3 September, 2013, " Android device activation numbers reach 1 billion worldwide".

[2] Buyya, R., Yeo, C. S., Venugopal, S., Broberg, J., & Brandic, I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. Future Generation Computer Systems, 25(6), 599-616. doi: 10.1016/j.future.2008. 12.001

[3] J.Rutherglen, D.Wampler and E.Capriolo, "Programming Hive", O'Reilly Media, Inc., October 2012.

[4] A.Thusoo, J.S.Sarma, N.Jain,Z.Shao, "Hive-A Petabyte Scale Data Warehouse Using Hadoop", In Proceedings of the 26th International Conference on Data Engineering, Long Beach, CA, USA, March 1-6, 2010, pp.996-1005.

[5] A.Thusoo, J.S.Sarma, N.Jain,Z.Shao, "Hive-A Warehousing Solution Over a Map-Reduce Framework", In Proceedings of VLDB Endowment, Vol.2, Issue. 2, August 2009, pp. 1626-1629.

[6] A. Gates, "Programming Pig", O'Reilly Media, Inc., October 2011.

[7] C.Olston, B.Reed, U.Srivastava, R.Kumar, "Pig Latin: A Not-So-Foreign Language for Data Processing", In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data (SIGMOD 2008), Vancouver, BC, Canada, June 9-12, 2008, pp. 1099-1110.

[8] K.S. Beyer, V.Ercegovac, R.Gemulla, A.Balmin, "Jaql: A Scripting Language for Large Scale Semistructured Data Analysis", In Proceedings of the VLDB Endowment, Vol.4, No.12, 2011, pp. 1272-1283.

[9] M.Droettboom, "Understanding JSON Schema", December 30,2013.

[10] A. Nandeshwar, "Tableau Data Visualization Cookbook", Packt Publishing Ltd., August 26, 2013.

[11] Kyar Nyo Aye, and Thandar Thein," A Comparison of Big Data Analytics Approaches Based on Hadoop MapReduce",In Proceedings of the 12th International Conference on Computer Application, Yangon, February 2014.

[12] V.Kottari, V. Kamath, L.P. Saldanha, C. Mohan, "A Survey on Mobile Cloud Computing: Concept, Applications and Cahallenges" , In Proceedings of the International Journal of Advanced and Innovative Research, March 30, 2013, pp. 487-492.

[13] C.White, "Using Big Data For Smarter Decision Making", BI Research, July 2011.

[14] K.Shvachko, H.Kuang, S. Radia and R.Chansler, "The Hadoop Distributed File System", In Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies, Incline Village, NV, USA, May3-7, 2010, pp.1-10.

[15] C.Ahn and Y.Nah, "Design of Location-based Web Service Framework for Context-Aware Applications in Ubiquitous Environments," in Proceeding of International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), Newport Beach, CA, USA, 7-9 June 2010, pp.426-433.

[16] http://www2.census.gov/census_2010/04-Summary_File_1

[17] Wei-Dong Zhu, M.Gupta, V.Kumar, S.Perepa, A. Sathi, C.Statchuk, "Building Big Data and Analytics Solutions in the Cloud", International Technical Support Organization, December , 2014.

[18] Rue du Rhone,"Standard ECMA-262, ECMAScript Language Specification, 5.1 Edition", June 2011.