**IJCTA,** 9(23), 2016, pp. 191-199


# Analysis of Hadoop and Map Reduce Tectonics through Hive in Big Data

**Shipra Jain* Aayush Gupta* and Ankur Saxena***

***Abstract :*** Every day,we see the article related to BigData which ables to handle and process the data more efficiently,profitably and provides a better solution to the problem. It all based on the technology associated with the BigData called Hadoop and MapReduce.In this paper we expect to give a thorough survey of extensive variety of proposition and systems focussing in a general sense on the backing of SAS/ACCESS Interface to Hadoop that gives us a chance to work with our information utilizing SQL builds through Hive. Then by further using the mapreduce framework which makes it more proficient and more supporting database driven operations.Hadoop is intended to store substantial information sets and progressively turning into the go-to framework for substantial scale, data intensive employments. It is useful to convert large data sets (from tetrabytes to petabytes) while MapReduce embraces an adaptable calculation model with a straightforward interface comprising of map and reduce capacities whose executions can be modified by application designers. IBM, Linkedln, Amazon, Twitter, Facebook are utilizing this innovation and many other goliaths are also moving towards the same.

***Keywords :*** Big Data, Framework, HADOOP, HDFS, Map Reduce, SAS, Hive.

## 1. INTRODUCTION

With the advancement of technology, utilization of web is developing step by step which prompts handle an excess of information by Internet administration suppliers, the topic of big data flashes on the screen.

**BIG DATA**-a gathering of substantial measure of information that are grouped together. It incorporates structured, unstructured and semi-structured information[1] .The associations-for example, huge transactional data(hospital, shopping bills etc) Amazon, Google, Facebook,data from sensors(readings done through meters ,other devices) and from biometrics(includes fingerprinting, genetics ,hand writing data) are utilizing Big Data so transactions can be managed furthermore focussing on the clients.

Big data consists four organizations: volume describes collection of huge data sets, think of petabytes instead of terabytes. Variety describes heterogeneous, complex and variable data[2],[3] such as sensor data and shadow data which include access journals and web search histories. Velocity signifies information which is produced as a consistent stream with constant queries for important data to be served on interest. Value defines the meaningful insights that convey future trends and patterns[4]. Many real world applications are based on Big data Techniques and also used in Google's self-driving car.

There are many new instruments and procedures to manage the Big data some are- Hadoop,Map reduce, Hive, Pig, Hbase, HDFS, Zookeeper, Avro, Riak, MongoDB. Many of these are java dependent which is one of the advantage to work with this environment.

Framework- a collaboration of many integrated components which produce a architecture which can be used for many related application[5]. Design pattern give us solutions whenever problem is arise because of some

* Amity University Uttar Pradesh Noida shiprajain.shipi@gmail.com, aayushgupta2295@gmail.com asaxena1@amity.edu

developing software in some particular context. In today's time many companies are developing many applications, so for handling these applications they are using the concept of framework including Hadoop and MapReduce Framework.

## 2. HADOOP

Apache Hadoop is an open source structure for creating dispersed applications that can prepare extensive measures of information. It is a stage that gives both distributed storage(HDFS) and computational capabilities(Map reduce). Because of the Hadoop popularity, it is natural to ask why we need Hadoop? Working with Hadoop provides the information regarding Core Java and some related ideas of Data Warehousing. Hadoop's library made in a way that it normally perceives and handles defaults which makes it more capable in the way it doesn't need to depend upon any hardware related equipment to distinguish the defaults. Hadoop emphasizes on moving code to data instead of data to code because code is smaller than data so it is easy to move around. It is also used for computationally intensive work in which distributed system move the data for computation and then final data moved back for storage[6].

Amazon uses Hadoop to handle their enormous number of sessions. Adobe uses it inner information stockpiling and handling. Cloudspace utilizes Hadoop for their client ventures. Hadoop has been used by eBay for their chase upgrade with investigation. Facebook uses Hadoop for the machine learning setup and to keep the records of their copies of internal log. Twitter is additionally utilizing it to deal with the information that is been produced every day in their site. IBM, Rackspace, The New York Times, Linkedln, are also utilizing Hadoop. There are some components and techniques related to hadoop.all components are accessible by means of the Apache open source permit.

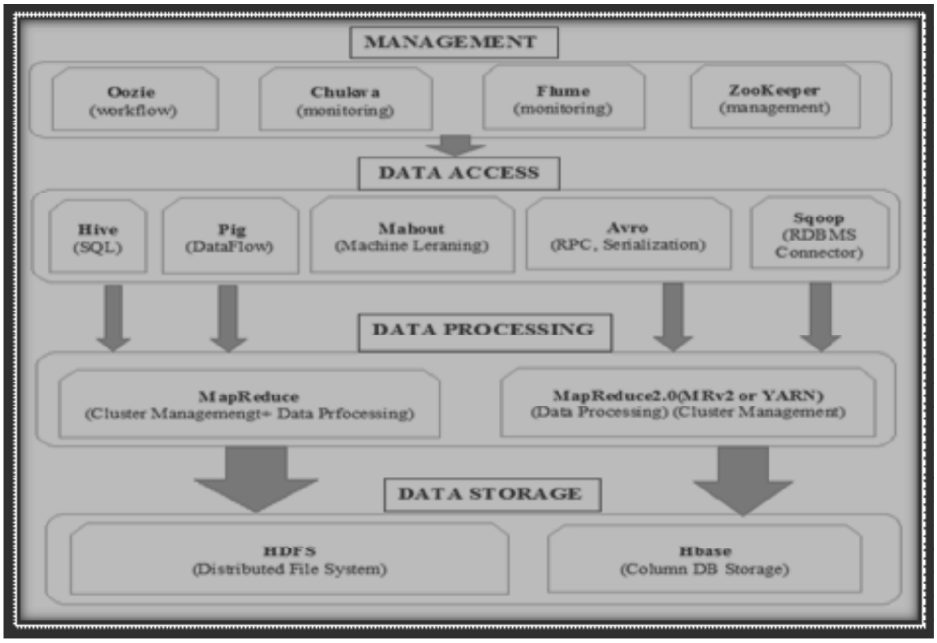



**Fig. 1. Hadoop Ecosystem architecture.**

### 2.1. HDFS

HDFS popularly known as Hadoop Distributed File System which is a block oriented documented framework. HDFS work with any platform as it is based on Java programming language[7] and used to store the large datasets. HDFS cluster contains two kind of nodes-NameNode and DataNodes.

HDFS split the information into numerous parts and appropriate every part over various hubs. Example-making of a record that contains the enrolment no. which serve as personality for everybody in the amity university; the general population with the name initiating with letter A directly handover to server1,initiating with B on server 2, etc. In this technology, bits of this personality will be kept away over agroup, and to recreate the overall characters, the program will need the blocks from each and every server associated in the bunch. To fulfill openness

as parts miss the mark, HDFS reproduces these little pieces onto two additional servers as is normally done.[This abundance can be lessened on or extended for every data premise or for a whole area; for example, a progression Hadoop group generally don't trouble with any data redundancy]. This emphasis offers different preferences,The most clear is higher openness.

**Key features and Advantages**

- Have high bandwidth to support map reduce workloads- At an immense information rate, HDFS can convey information into the process foundation. HDFS can without much of a stretch surpass 2 gigabits for every second per PC into the guide diminish layer, on a minimal effort shared system.
- Low cost per byte-HDFS utilizes item coordinate joined stockpiling and shares the expense of the system and PCs, it keeps running on with the component of the Hadoop called MapReduce[8].
- Process and capacity can be scaled autonomously as opposed to inside the altered limit of a hub.Both information and capacity can be shared between various Hadoop occurrences and an expanded level of assurance around the HDFS metadata can be given.
- Accessible-Hadoop keeps running on expansive bunches of ware machines or on distributed computing administrations, for example, Amazon's Elastic Compute Cloud (EC2) [6].
- Hadoop specifically handles greater data with the addition of many hubs to the group.
- Robust- hadoop is expected to keep running on commodity hardware, has the presumption of regular equipment breakdowns and most such disappointments are effortlessly handle.
- Hadoop allowed to concentrate on what is most essential to us and our information and what we need to do with it.

### 2.1.1. Architecture

HDFS has an expert/slave engineering.(HDFS) part huge information records into block which are overseen by various nodes in the cluster and write- once-read many semantics on records[9] are also supported by it but at the time of write process no read operation can perform to a file and the document close is the exchange that permits users to see the information, At the point when a disappointment happens, unclosed records are erased. HDFS group includes a single NameNode, a specialist server that works with the file system namespace and oversees access to reports by clients[10].Namespace is a chain of command of documents and directories and has traits like modification, transfer a file from one directory to another. NameNode helps to map the data blocks to DataNodes.A data get part into one or more lumps and set of pieces are secured in DataNodes[11]. The DataNodes are accountable for serving read and write requests from the record framework clients. The DataNodes moreover perform eradication, block creation and replication in light of rule from the NameNode. Notwithstanding this every block is imitated over a few machines, so that a solitary machine disappointment does not bring about any information being inaccessible. In initial phase, the NameNode remains in a state called Safemode,blends all the editlog documents and composes the metadata into the fsimage file[12]. After then flush out the editlog document and will remain in the Safemode until Replication of data blocks occur. The NameNode receives the blockreport and heartbeat messages from the DataNodes[13]. A Blockreport carries the rundown of data blocks that a DataNode is encouraging. Every chunk has predetermined minimum number of imitations, considered safely repeated when the NameNode checked the base number of imitations of that data block. once the imitated information pieces checks in with the NameNode safely(notwithstanding an additional 30 seconds), NameNode leaves the Safemode state. It then chooses the summary of data blocks (expecting any) that regardless have not exactly the predefined number of impersonations, NameNode then rehashes these pieces to various DataNodes, considered old DataNode to be dead also, does not forward any new I0 request to them.A dynamic checking framework then re-recreates the information in light of system failure which can bring about partial storage.

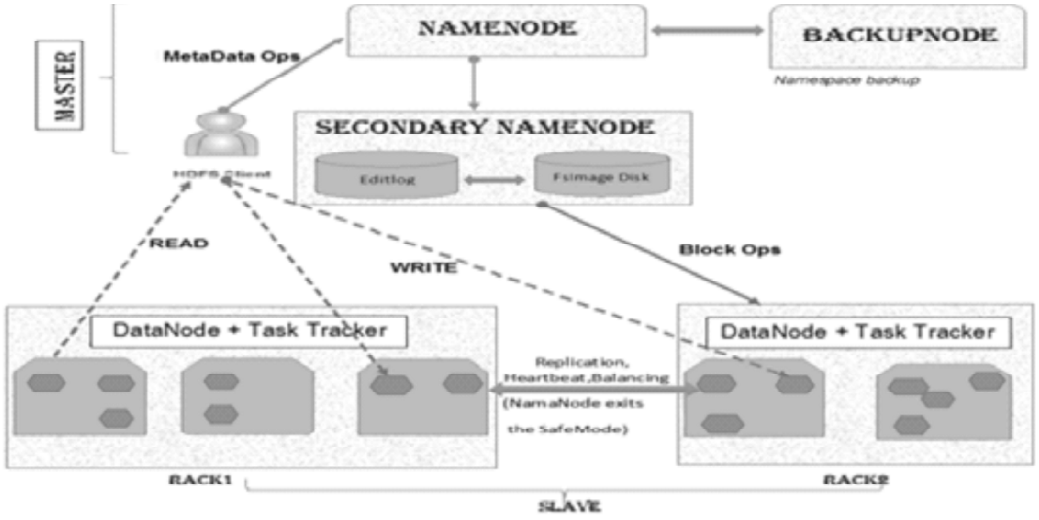



**Fig. 2. Tectonics of HDFS.**

## 2.2. Map Reduce

MapReduce was originally proposed by Google to handle web search applications[10] on large scale.it act as a parallel preparing framework on huge information and the analysis part of Hadoop system so as to improve the challenges experienced while preparing and breaking down substantial information sets. MapReduce forms the huge measure of organized and unorganized information by utilizing map and reduce capacities. This methodology has been ended up being a successful programming approach for creating machine learning, information mining and inquiry applications in server farms[14]

The Map function play out the sorting and isolating operation by creating an arrangement of intermediate key/ esteem sets[15]that are stored on the HDFS while the Reduce function will then play out a framework operation on the sorted information, joins all halfway values connected with the same intermediate key to deliver the output. It is OS independent.

MapReduce comprise of numerous parts: JobTracker - the expert hub,deals with all jobs and resources in a cluster. TaskTracker-pros passed on to each machine in the group to run the map and reduce task. JobHistoryServer - a portion that keeps record of complete jobs, and is commonly passed on as an alternate capacity or with JobTracker.

**Example :** As teacher has to check the whole class papers in one day containing 75 students as a single server application, but that is too tedious for a single teacher. By contrast, teacher can split the task among 3 different teachers, so each will take some sets of papers, after finishing continue with other set. This is the map aspect of MapReduce. And if a teacher leaves, another teacher will takes his place. This is the condition of MapReduce's fault-tolerant element. After completion of checking,each teacher sorts the papers in the stack roll number wise.The number of students with same grades is an example of reduce aspect of MapReduce. The processing of map and reduce is illustrated in the given figure below.

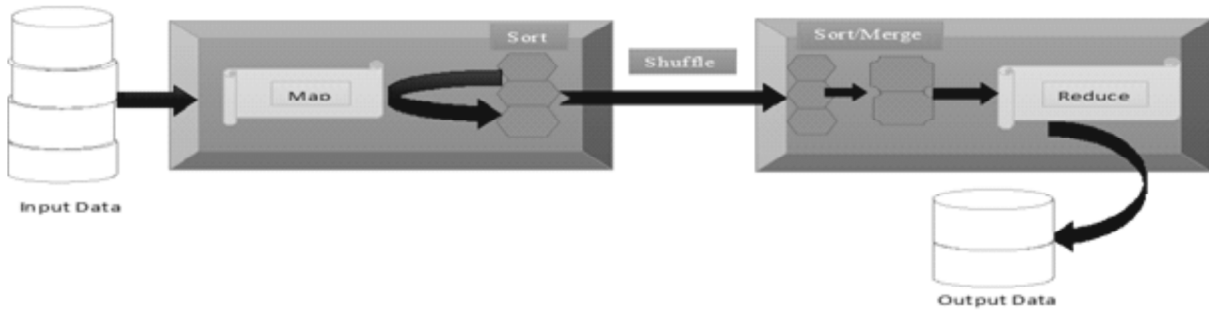



**Fig. 3. Tectonics of Map Reduce.**

## 3. METHODOLOGY

In this part of paper we have talked about accessing hive data using SAS interface to Hadoop

**Example :** consider Student_master and Student_details as a large DBMS tables

**TABLE 1.** Create table Student_master

(RN char(10) primary key, Name varchar(20) not null, Address varchar(50) default ('Amity,Noida'), Phone number(8,2) unique, DOB date);

//Add some details to the Table- Student_master//

Insert into Student_master

Values ('A-105','NISHA','DELHI',23546644,'1-MARCH-1993');
('A-103','VINAY','NOIDA',45375868,'24-AUGUST-1990');
('A-206','AKASH','NOIDA',25477658,'29-JULY-1994');
('A-515','VINI','GHAZIABAD',25337445,'5-DECEMBER-1992');
('A-333','SARAN','JAIPUR',42658753,'25-NOVEMBER-1993');
('A-612','SHIPRA','DELHI',65389509,'1-DECEMBER-1994');

**Table 1. Student_master.**

| *RN* | *NAME* | *ADDRESS* | *PHONE* | *DOB* |
|---|---|---|---|---|
| A-105 | NISHA | DELHI | 23546644 | 1-MARCH-1993 |
| A-103 | VINAY | NOIDA | 45375868 | 25-AUGUST-1990 |
| A-206 | AKASH | NOIDA | 25477658 | 29-JULY-1994 |
| A-515 | VINI | GHAZIABAD | 25337445 | 5-DECEMBER-1992 |
| A-333 | SARAN | JAIPUR | 42658753 | 25-NOVEMBER-1993 |
| A-612 | SHIPRA | DELHI | 65389509 | 1-DEEMBER-1994 |

**TABLE2.** Create table Student_details

(RN char(10) primary key, programme varchar(20),department varchar(20), SEM number check (SEM between 1 and10), fees number(8,2) check fees>25000);

//Add some details to the Table- Student_details//

Insert into Student_details

Values ('A-105','BTB','AIB',6,60000);
('A-103','MBA','AIM',2,250000);
('A-206','BTC','ASET',5,70000);
('A-515','MSC','ASET',3,150000);
('A-333','DETT','AIE',3,30000);
('A-612','BTB','AIB',7,65000);

**Table 2. Student_details**

| *RN* | *PROGRAMME* | *DEPARTMENT* | *SEM* | *FEES* |
|---|---|---|---|---|
| A-105 | BTB | AIB | 6 | 60000 |
| A-103 | MBA | AIM | 2 | 225000 |
| A-206 | BTC | ASET | 5 | 70000 |
| A-515 | MSC | ASET | 3 | 150000 |
| A-333 | DETT | AIE | 3 | 30000 |
| A-612 | BTB | AIB | 7 | 65000 |

**Passing through facility specifics for the Hadoop interface using SQL are :**

- The dbms-name should be HADOOP.
- The CONNECT proclamation is required.
- PROC SQL bolsters different associations with Hadoop.
- The CONNECT proclamation database-association contentions are indistinguishable to its LIBNAME.

```
Proc sql ;
Connect to Hadoop (server = duped user = shipra password = amity);
Execute (create table shipra_1
                row format delimited fields terminated by '\001'
                 stored as textfile
as
select department, Min(fees), Max(fees)
 from Student_details
group by department) by Hadoop
  disconnect from Hadoop;
quit;
//SAS interface to Hadoop gives a LIBNAME engine to access a Hadoop Hive data.
/* simple libname statement */
Libname mylib Hadoop server = duped user = shipra password = amity;
```

If we need to recover the lines from an inward join of above two tables. PROC SQL distinguishes the join between two tables in the DBLIB library (which references an Oracle database), and SAS/ACCESS passes the join straightforwardly to the DBMS. The DBMS forms the internal join between the two tables and returns just the result to SAS.

```
/* create a SAS data set from Hadoop data */
Proc sql;
   Create table info.join_test as  (
        Select  sm.name  "Student Name", sd.department
          From mylib.student_master sm, mylib.Student_details sd
         0n sm.RN = sd.RN);
Quit;
//using custom map/reduce to hive queries//
From (
      From Student_mater , Student_details
      MAP Student_master.sm  Student_details.sd
       Using 'map_script'
        As mp1,mp2
       Cluster by mp1) map_output
  Insert Overwrite Student_master Student_details SomeOtherTable
      Reduce map_output.mp1, map_output.mp2
       Using 'reduce_script'
        As reduces, reducesd;
```

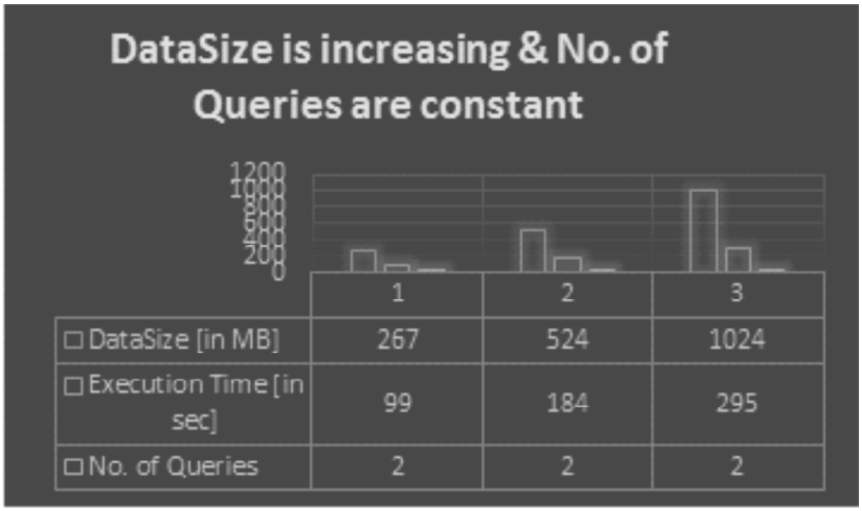
In above query map_script collect the data from two tables- Student_master and Student_details and will map it to mp1, mp2 fields then further transfer to reduce_script which will play out a framework operation on the sorted information, joins all halfway values connected with the same intermediate key to deliver the output *i.e*. specified in the cluster by mp1 and the result of the reduce_script output will shown to SomeOtherTable.

## 4. RESULT AND DISCUSSION

As Hadoop is broadly useful information for stockpiling and processing stage that incorporates database-like instruments, for example, Hive and HiveServer 2.

SAS/ACCESS Interface to Hadoop gives us a chance to work with our information utilizing SQL builds through Hive and HiveServer2[16],[17]. It additionally gives us a chance to get to information straightforwardly from the fundamental information stockpiling layer, the Hadoop Distributed File System (HDFS), with this interface we can perform read and write operations to and from Hadoop and provides quick, proficient access to information put away in Hadoop through HiveQL. To increase in Execution Time We reduce the information by using custom map/reduce to hive queries to work well and productively and the outcomes thoroughly rely on upon the span of Hadoop group. The execution of above application has been appeared concerning execution time, dataset size and number of queries.

**Case 1.** When DataSize is increasing and No. of Queries are constant




| | 1 | 2 | 3 |
|---|---|---|---|
| DataSize [in MB] | 267 | 524 | 1024 |
| Execution Time [in sec] | 99 | 184 | 295 |
| No. of Queries | 2 | 2 | 2 |

**Fig. 4. Chart of Applied Queries.**

**Case 2.** When DataSize is constant and Number of Queries are increasing.

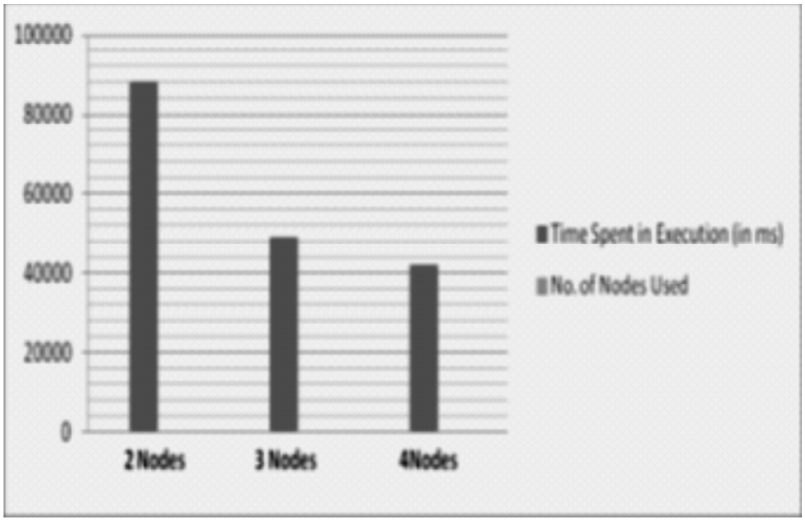



**Fig. 5. Comparison chart of Data and Queries.**

**Case 3.** When DataSize and Number of Queries both are increasing.


**Fig. 6. Chart of execution Tim.**

In this experiment, we have seen that in case1 the Execution time increases with increase in DataSize while in case3 we conclude that-As DataSize and No. of Queries are increasing the Execution time varies means it first decrease and then increase linearly this is due to the load exerting on the Queries with increase in communication between those Queries.

## 5. CONCLUSION AND FUTURE WORK

It is imperative to note that in these experiment, we entirely utilized Hive inside the Hadoop environment,reenacted the information distribution center type workload in which information is stacked in clump, and afterward queries are executed to answer vital business questions.Every queries executed through SAS/ ACCESS to Hadoop were submitted by means of the Hive environment and were interpreted into MapReduce framework so that we can understand how to structure the tables in hadoop with large amount of data and performance get increases with the increase in execution time by decrease the amount of data handled..

The future examination incorporates new applications for Hadoop nature of administration on the distinctive size of the datasets utilizing MapReduce. Execution assessment of MapReduce with various form of hardware and software configurations. We can extend this work with the integration of Spring, Struts and Hibernate of java to improve more security of web applications.

## 6. REFERENCES

1. Dhakite A , Thakur S.” A Survey on Hadoop Technology and its Role in Information Technology”, International Journal of Advanced Research in Computer Science and Software Engineering 5(4), April- 2015, pp. 375-379.
2. Deepika P , Anantha Raman G R.” International Journal of Advanced Research in Computer Science and Software Engineering”, International Journal of Advanced Research in Computer Science and Software Engineering 5(9), September- 2015, pp. 160-164.
3. R. Waters (2013). “Google search proves to be new word in stock market prediction”. Financial Times Retrieved 2013.
4. A.Saxena, N.Kaushik, N. Kaushik”Implementing and Analyzing Big Data Techniques with Spring Framework in Java & J2EE “Second International Conference on Information and Communication Technology for Competitive Strategies (ICTCS) ACM Digital Library 2016.
5. Saxena A “Web Based Custom Validation Using Framework in Java” International Journal of Computer Science Trends and Technology (IJCST) Volume 3 issue 1, 90-96 2015..
6. Shafer J, Rixner S, Cox AL. The Hadoop Distributed Filesystem: Balancing Portability and Performance, in Proceedings of IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS 2010), White Plains, NY, 2010.

7. Shvachko K, Kuang H, Radia S and Chansler R. The Hadoop Distributed File System in Proceedings of the 26th IEEE Symposium on Massive Storage Systems and Technologies, 2010.
8. Satyanarayanan M, Kistler J.I, Kumar, P et ai, "Coda: A highly available file system for a distributed workstation environment", IEEE Transactions on Computers, 1990, 39(4):447-459.
9. Sage A. Weil, Kristal T. Pollack, Scott A. Brandt, Ethan L.Miller ,"Dynamic Metadata Management for Petabyte-scale File Systems ", In Proceedings of IEEE University of California, 2004.
10. Jeffrey Dean and Sanjay Ghemawat, "MapReduce: Simplified Data Processing on Large Clusters, "In Communications of the ACM, Volume 51, Issue 1, pp 107-113, 2008.J.Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon,1892, pp.68–73.
11. Shvachko K, Kuang H, Radia S, Chansler R., "The Hadoop Distributed File System," Mass Storage Systems and Technologies (MSST), IEEE 26th Symposium on IEEE, 2010,
12. S. Chandra Mouliswaran, S. Sathyan. et al., Journal of Science ,Vol 2 , Issue 2 (2012 ) 65-70.
13. Weil S, Brandt S, Miller E, Long, ,Maltzahn C, "Ceph: A Scalable, High-Performance Distributed File System," In Proc. of the 7th Symposium on Operating Systems Design and Implementation, Seattle, WA, November 2006www.folkstalk.com/2013/10/namenode-secondary-safe-mode-hadoop.html.
14. Thusoo A,,Sarma, J.S, Jain N, Shao N,Chakka P,Anthony S, H. Liu, P. Wyckoff, R. Murthy, "Hive – A Warehousing Solution Over a MapReduce Framework," In Proc. of Very Large Data Bases, vol. 2 no. 2, August 2009, pp. 1626-1629International Journal of Computer and Electrical Engineering, Vol. 6, No. 1, February 2014.
15. Techniques in Processing Data on Hadoop Donna De Capite, SAS Institute Inc., Cary, NC/ paper SAS033-2014.