UGC JOURNAL NUMBER 44557

International Journal of Allied Practice, Research and Review
Website: www.ijaprr.com (ISSN 2350-1294)

Hadoop: Scalable Technology for Small Scale Business

Jayati Vijaywargiya, Manya Srivastava and Dr. Vinod Maan
College of Engineering and Technology
Mody University of Science and Technology, Lakshmangarh, Sikar, Rajasthan

Abstract. In this era of bulk information the optimization of an organizations' data becomes a necessity. Efficiency in data exploration is required for an organization big or small, to achieve competitive advantage. Every organization wants to have an efficient on demand and on command digital platform for profiling the data from its various sectors. This research paper is focused on analytics of stock information of an organization. The analysis and management of raw data from organization is done using MapReduce technique on Hadoop framework. Also, analysis of execution time with size of data set is inculcated.

Keywords: ***MapRedure, Hadoop, data set, scalability***

I. Introduction

The objective of this paper is to keep all the Records of an entire website namely books, mobile, laptops etc. under a single observatory & manipulative system. By using this System operator can keep record of the all collection and other concerned activities of the organization. This system provides a better approach to keep operator user friendly to this application in the maximum possible way.

1.1 Computerization Exigency

The key tools involved are JAVA, BIGDATA, HADOOP and MAP-REDUCE, they are much scalable, flexible and user friendly than many other software language and database management system respectively.

II. Literature Review

Too many small scale industries take a back seat when it comes to digitizing the organization's information. They till date rely on traditional method of single computer storage in files and in hard copies. Although this method leads to single point of failure and thus they are on the path use the technological tools.

The approach used in our research project is MapReduce. From the reviews and knowledge of MapReduce it has the below mentioned advantages. Computing aggregation is simple using MapReduce. Although MapReduce model is simple but it is quite economically expressive. Map and Reduce functions are sufficient to define a programmer's job and there exist no need for the physical distribution of the job across nodes in hadoop. Thus this review adds on an advantage for our approach for the research topic. MapReduce is independent of data model and schema. MapReduce approach helps the programmer to deal with variety of data more easily than DBMS. Independent of the storage MapReduce is basically independent from underlying storage layers. It can work with different storage layers such as BigTable[1]. MapReduce is highly fault-tolerant. For an instance and as a fact, it is reported that MapReduce can continue to work in spite of an average of 1.2 failures per analysis job at Google [2, 3]. The best advantage of using MapReduce is high scalability. "Yahoo! reported that their Hadoop gear could scale out more than 4,000 nodes in 2008". [4].

III. Research Gap

In recent years there has been an increased focus on digitization on firm's stock information and firm's sales performance. There are studies showing that this digitization is not yet achieved by small scale industries or new start-up organizations. The major problem concerning to a software to be built for these group of business community is scalability. Developing of stock management and analysis software in an environment with limited scalability leads to future expansion problems and at the same time developing a very large scale application that is suitable for a big organization will not be feasible to manage for small industries.

IV. Model Portrayal

The implemented schema is an integration of various MapReduce Java Programs. Implementation was done using waterfall model as the objectives were clear and rigid. Steps followed in software development were:

4.1 Requirement analysis

It is the first step of software development where all the necessary requirements for software development are stated.

As our proposed and implemented model is an application of hadoop, we require single node environment set-up. This leads to requirement of a platform where we can work on standalone mode of Hadoop. Installation of Cloudera for Hadoop implementation on VMware was compassed to set-up development environment. The requirement was fulfilled by Cloudera Hadoop VMWare Single Node

Environment Setup. VMware allows Virtual Machine VMs to run on a single physical machine. MapReduce2.0: The coding phase of the software development involved coding of MapReduce programs in Java Programming Language.

4.2 Application Layout

It consists of data flow diagram and flowcharts that can be easily correlated to the working model of software. It also gives clear understanding of internal software processing and the user interaction in front end of software. The proposed data flow diagram is shown in fig. 1.Two types of designing were implemented during the development.

I. Physical design: It gives the physical relation between the program modules and their implementation.

II. Logical design: It tells about the database requirement and the overview of implantation details.

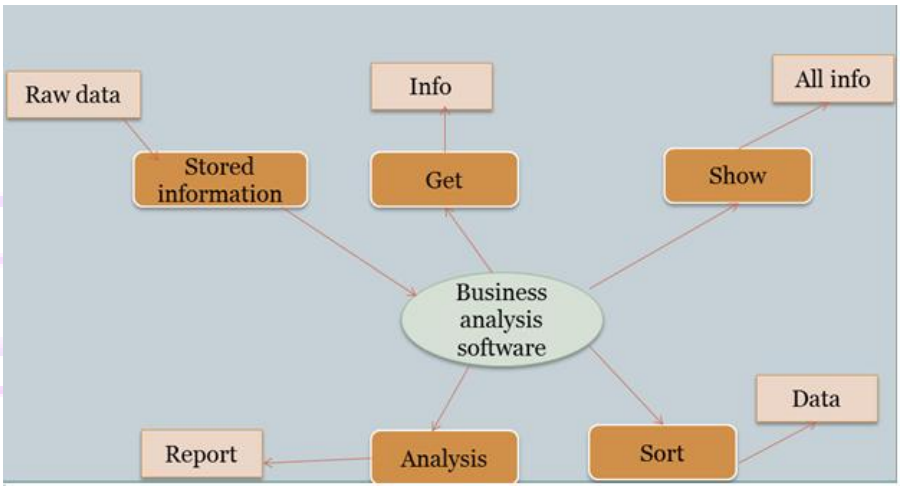

Fig 1 Data Flow diagram –Level 1

4.3 Testing

Verification and validation of the proposed and implemented model is a must step before deployment. White box testing or glassdoor testing was performed at least level of development to check the appropriate execution of various MapReduce programs. Practical needs of computer code were tested by black box testing. Performance error and function interface errors were resolved during the execution.

4.4 Deployment

MapReduce in Hadoop framework using Java was used in coding phase of software development. Map and Reduce task are sent to respective cluster by Hadoop. All other task issues such as verifying task completion, copying data are managed by framework itself. As this is a standalone application thus computing takes place on the single node with data on the local machine. After computation the reducer result is aggregated to produce final output of program.

V. Model Implementation

5.1 Pre-requisite

The execution of the proposed model begins with starting cloudera in the virtual platform VMware. In google chrome we can access the cloudera manager in order to enable all the services required for execution of the software. The following services needs to be enabled:

1. HDFS
2. HIVE
3. HBase
4. Flume

5.2 Software Stipulation

VMware is a virtual platform for execution of Hadoop based programs. It is used in complementary with cloudera and CentOS6.2 operating system on top of it. Implementation of our proposed schema needs background services. For this some services are switched to active mode. These services include HBase, Flume, HDFS, Hive, Hue and Mapreduce.

Cloudera Manager is opened in Google Chrome so as to enable the required services. In *All Services*, the *Action Tab* against the service mentioned above, needs to be clicked in order to change its status to *Active mode*.

5.3 Gait to Map and Reduce jobs

The execution starts in console window by first setting the JAVA path by using commands *echo $JAVA HOME, Export HADOOP CLASSPATH=${JAVA HOME}libtools.jar*.

Compilation of MapReduce programs is done by hadoop command:

com sun tools javacMain Login.java FrameMain.java mr1.java ...(This includes all the Java programs created).

Jar files creation and linking with the class files in bin is done using command

jar cvf prj.jar C bin.

Final execution is done using *hadoop jar prj.jar hadoop.Login.*

The results of the execution of Map and Reduce programs will be shown in the backend, command prompt while the front end serves as an interactive window for data extraction. The below table mentions few of the results in commands prompt:

Table 1. Execution inference of MapReduce implementation

Total input path to process	**1**
Running job	**Job_20171217515_0001**
Map	**0%**
Reduce	**0%**
Map	**100%**
Reduce	**0%**
Number of bytes read	**301**
Number of bytes written	**70**
Number of read operation	**2**

Table 2. Map-Reduce Framework

Map input Records	**5**
Map Output Records	**5**
Input split bytes	**122**
Spilled Records	**0**
CPU time spent(ms)	**870**
Physical memory (bytes) snapshots	**109637632**
Virtual memory (bytes) snapshots	**657166336**
Total committed heap usage	**60751872**

5.4 Scalability

Survey of execution time with respect to maximum and minimum data set is plotted below. Mappers corresponding to 2-itemsets of highest and lowest data contents are taken for analysis of impact of execution time on scalability. Fig.2 shows the plot of execution time of 12 Mappers for two values of data set with extreme large size difference.

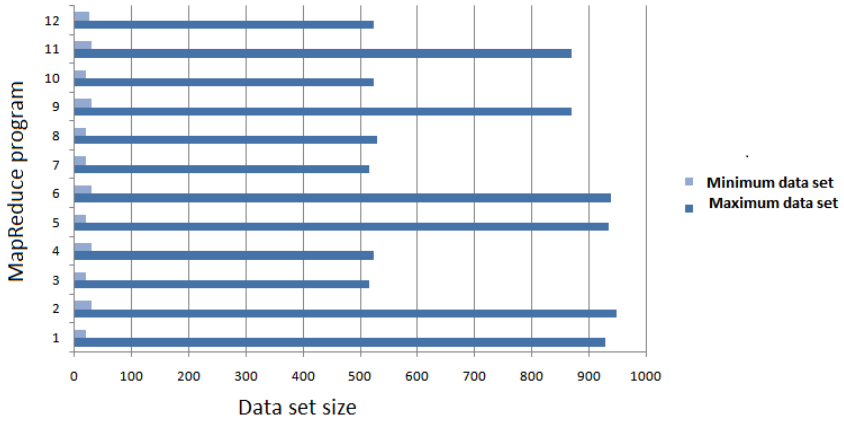

Fig. 2 Plot of execution time with respect to extreme of data set

VI. Constraints

Developers desire to design their product as much user friendly as they can, but unless and until product is used by the user, developers can't pre examine every flaw that user may later encounter. The development of the product is on "observation basis" while the user judges the product on "use basis". The time frame in which product is developed is earlier to the time frame in which user uses the product. This difference may result in user dissatisfaction, as by the time user receives final product user needs may have changed. So the result of the communication gap and time difference in product development and product deployment may lead to shortcomings.

The limitation faced during the development is that the project was developed and implemented on 4GB RAM laptops. Thus, inculcation of voluminous variety data was not achieved in the implementation of proposed design.

The biggest limitation of using MapReduce is that only aggregations are possible in big data.

VII. Conclusion

The proposed model was able to be implemented using Single node Hadoop Architecture. The approach used for the fulfilment of desired goal of a scalable user friendly management and analysis application was MapReduce. An organization's data could be managed using the software schema proposed and implemented. Development of user friendly and interactive interface was partially completed. This proposed model is highly scalable and it can further be enhanced by inculcation of variety of data, Better UI (User Interface) and more MapReduce programs.

VIII. References

1. F Chang et al, Bigtable A distributed storage system for structured data ACM Transactions on Computer Systems, 2008

2. Jeffrey Dean et al, Mapreduce Simplified data processing on large clusters In In Proceedings of the 6th USENIX, 2004

3. J. Dean et al, MapReduce Simplified data processing on large clusters Communications of the ACM , 2008

4. A Anand, Scaling Hadoop to 4000 nodes at Yahoo, 2008

5. J. Han and M. Kamber, Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 2006.

6. **J. S. Ward and A. Barker, "Undefined by data: a survey of big data definitions," http://arxiv.org/abs/1309.5821v1"**