

## A Cloud Based Framework for Cloud Computing

GowthamKrishnaKumar S

M. Tech (CSE),  
ASCET, Gudur.

Ramesh Ragala

Associate Professor (CSE),  
ASCET, Gudur.

### ABSTRACT

*In recent days, cloud computing and big-data turns into backbone of internet. Big-data and cloud computing are powerful keywords in present IT. Internet service providers are providing services with cloud computing and mobile computing. Various mobile devices in cloud are producing trillions, even in Quintillion bytes of data. So we need huge amount of computational resources to manipulate these data. So cloud computing becomes more costly and taking more time to manipulate. So it is not capable of handling large amount data as of user flexibility. We need simulation tool in cloud, which support data processing with map reduce functionality. This tool can be useful for processing large amount of datasets. It will helpful to both cloud computing and data centers.*

### Keywords

**Cloud computing, Hadoop, Map-reduce, Performance Evaluation.**

### 1. INTRODUCTION

A cloud is a type of parallel and distributed system consisting of inter-connected and virtualized computers [4]. A cloud computing system consists of a collection of interconnected and virtualized computers. Producers and Consumers involved in negotiation of service level agreements (SLA's) to access unified computing resources. In agent-based cloud computing, cooperation, negotiation, and coordination protocols of agents are adopted to automate the activities of *resource pooling* and *sharing* in clouds [5]. In last two decades, internet access and speed has increased across the world causing a blast amount data being produced and collected. In 2012, the amount of data created per day is about 2.5 quintillion bytes. In that 90% of the data in the globe today has been occurred in last two years only. This data is called big data [1].

The four dimensional measures of Big data are Volume, Velocity, Variety and Veracity. Volume of big data, determines growing of data in huge size i.e., terabytes and even petabytes of information. Big data's second dimensional measure, determines the execution speed of the task. This is very important measure for big data, because time is crucial factor in data processing. The big data

include multiple types of data. It may conations structured, semi-structured, and unstructured data. This characteristic can be specified by variety measure of big data. In most of the data processing application, we need high level of accuracy. This can be specified by Veracity of big data. This measure supports correct decision making.

Hadoop is a free open source frame work for distributed applications. It provides reliability, scalability and also provides new methodology to storing and processing data in distributed environments.

On the other side, software agents [7] can be used for implementing intelligence in Cloud computing systems as more adaptive and flexible in resource management, service provisioning and also in running large-scale applications. Software gent is an active permanent intelligent computer entity. It has its own ideas to accomplish their own tasks. Software agents are: perceived, reasonable intelligent acted, communicated with others agents. Agents may have represented as equipments as the valves, pumps, boilers, heaters, sensors, units of the equipments, or the whole unit and others. Agents may have also represented the operations like control and diagnostic procedures.

In this paper, we focus on Mapreduce functionality [8]. This model is frequently used in cluster of servers. Mapreduce is a platform for data processing mechanism. We construct an efficient mapreduce model in cloud environment, which provides easier and cheaper way to verify the mapreduce operations. The main aim of this work is to provide frame work which provides manipulation of huge amount of data with lower cost, which in turn reduces the data storage in cloud computing environments.

The rest of the paper is classified as we explained mapreduce functionality model and features of social data in section 2. In section 3, we focused on conceptual model of simulation tool for cloud environment. In section 4, introduce how we implement the system. In section 5, we evaluated the performance of the proposed system with two applications. Finally in section 6 we conclude the paper.

### 2. BACKGROUND

Mapreduce is a programming model as well as a framework. The main idea of the Mapreduce model is to hide the details of

parallel execution and allow users to focus only on data processing strategies. The Mapreduce model consists of two primitive tasks are map and reduce tasks. Each map task in Hadoop is a list of (key1, value1) pairs broken into the following phases: *record reader*, *mapper*, *combiner*, and *partitioner*. The output of the map tasks, called the intermediate keys (key2, value2). The intermediate key-value pairs are then grouped together on the key equality basis, i.e. (key2, list (value2)) are sent to the reducers. For each key2, the reduce tasks are broken into the following phases: *shuffle*, *sort*, *reducer*, and *output format*.

The nodes in which the map tasks run are optimally on the nodes in which the data rests. However the Users can redefine the Map and Reduce, if they want to use mapreduce model in their data processing applications accordingly. Each job in Hadoop is broken down into many Map tasks as input data blocks and one or more Reduce tasks during processing. Figure 1 map reduce functionality model. A map reduce model is performed in two stages. Stages are mapper and reducer. Mapper function takes care of mapping the data, and then shuffling and sorting the data. Reduce function take care of reduce the data and finally displays the data.

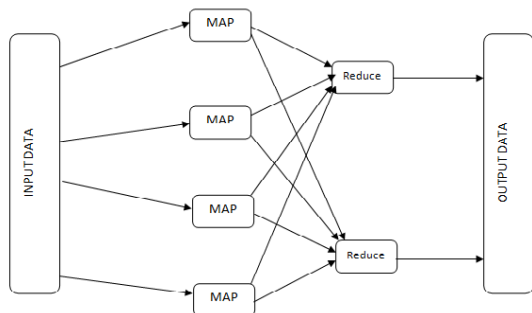


Figure 1: MapReduce Model

There are several algorithms are available for mapreduce functionality on their requirement to handle the large datasets. Some of the algorithms of mapreduce model are sorting, searching, page rank, BFS, TF-TDF, etc [11].

### 3. DESIGN OF MAPREDUCE MODEL IN CLOUD

We explain in this section how mapreduce functionality is designed and then it is used in cloud environment. In this section, we are explaining the mapreduce functionality and its design in cloud environment. The cloud environment is comprises of physical resources, virtual resources, datacenters, and servers. In cloud, the clients and virtual machine's are able to send data to data centers. The data centers [10] are processing the data collectively. The map reduce framework is able to optimize the cloud resources and cloud management directives based on collected data. This process can be done before and after the data collection to provide the system with low unpredicted errors like allocating more resources. So, we implemented the mapreduce

functionality on cloud. To provide more compatibility with data centers, we uses agent to interact data centers.

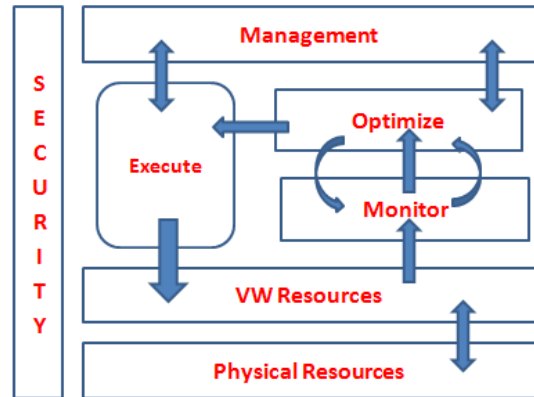


Figure 2: Design model of cloud

### 4. IMPLEMENTATION

The implementation of mapreduce model is prepared with two parts based on design decisions. One is addition and second one is modification. The mapreduce is added. Modifications are made on cloud including data center and data center agent. Figure 3 explains an execution workflow of mapreduce on cloud.

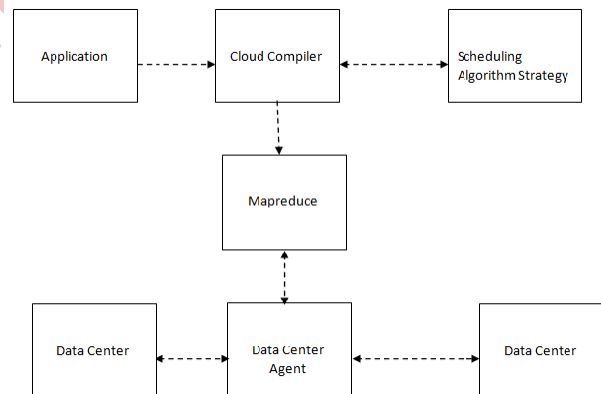


Figure 3 Workflow model of framework

- a. Application: All users. They can access their requirements using cloud.
- b. Cloud Compiler: Cloud Compiler is used to compile the cloud user programs.
- c. Scheduling Strategy: Scheduler strategy can takes care of user jobs.
- d. Cloud computing model: it is used to call the mapper and reducer functions.
- e. Data Center Agent: it is an agent between computing model and data center. It is mainly used for sending and

- gathering all the information from all the data centers across the cloud.
- f. Mapreduce model: it is a computational model of hadoop. It is used to compute the large volume of datasets.
- g. Data Center: it is used to handle the data storage operations.

The structure of implemented model is as follows

- 1) Generate a workload and run it with mapper class
- 2) Mapper class creates mapreduce files.
- 3) Created files are marked as map files and allocate keys manually.
- 4) Each mapper file generates a reducer file and they are combined.
- 5) Data center receives the mapreduce files then it performs the map operations.
- 6) Based on the map operations, the data center agent sends the map files to the data centers for mapping
- 7) Whenever mapping completed the reducer tasks are sends to the data centers.

The list of classes constructed in this model is as follows:  
aggregator, mapper, mapreduce file, reducer classes.

- a) Aggregator: It is a class. It denotes a list of intermediate files. The intermediate files are sent to the master node to another VM. So scheduler externally defined. So an aggregator required.
- b) Mapper: it is also a class. It is used to separate the input files into small subtasks based on mapper functions.
- c) Mapreduce Pair: This class is used to bind the map and reduce. So it reduces a lot of time.
- d) Reducer: It is a reducer class is used to combine the subtasks of input files based on sorting and shuffling functions and gives an output of input files.

## 5. PERFORMANCE EVALUATIONS

In this section we report the performance results of running data intensive applications on virtual clusters in cloud. We build two virtual clusters using two VM types, c1.xlarge and m1.xlarge. Each cluster configurations is summarized in table 1.

Table 1: Configuration Of The Vms Used For The Virtual Clusters

VM type	Configuration element	Description
c1.xlarge	Number of virtual cores	4
	Memory Size	4GB
m1.xlarge	Number of virtual cores	2
	Memory Size	2GB
Hard Drive	20GB	
Network	1 GigE Ethernet	
OS	Ubuntu 10.04 Lucid, 2.6.27.21-0.1-xen	
JVM	1.7.0_03	
Hadoop	Hadoop 0.20.2-cdh3u3	

For this comparative study we employed 2 classes of MapReduce benchmarks: Wordcount and Sorting. The sorting benchmark is changed version of the popular Terasort [6]. The wordcount aims at counting the occurrence of individual words in a set of files stored in the HDFS storage. These two applications show the some level of speedup. We are interested in the performance consequences of different VM types. We build two homogeneous clusters using either type of VMs, c1.xlarge or m1.xlarge. We report the performance result for each cluster configuration. The two clusters are different in terms of virtual core count per VM(2 vs 1 cores) and available virtual RAM Memory (2GB vs 1GB). Each virtual cluster is composed of 10 VMs interconnected with 1 virtual Gigabit Ethernet network. To control the performance calculation environment, we used the storage size of each VM to 20GB.

Figure 4 evaluates the performance of the sorting. By using the equal number of VMs, the expected speedup is increased for 2 times for the c1.xlarge cluster. The benefits of multi-core processing (c1.xlarge) decreases as the problem size increases until 15 GB. Since the computing complexity  $O(n \lg n)$  is higher than the communication complexity  $O(n)$  [2] [3], the speed differences stabilize.

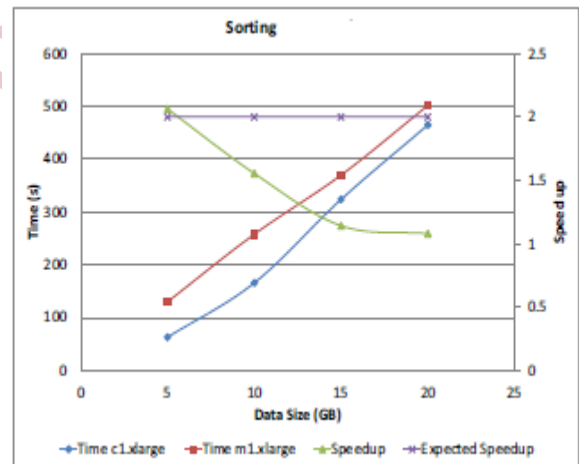


Figure 4 Terasort Performance

Figure 5 reports the virtual cluster performance for the Wordcount application with increasing sizes of text corpus. In this application the execution speedup is increased to 1.5 times c1.xlarge cluster. This is because of computing and communication complexities of application is the same  $O(n)$  [2] [3]. The multi-core benefits reduce after the network is flooded.

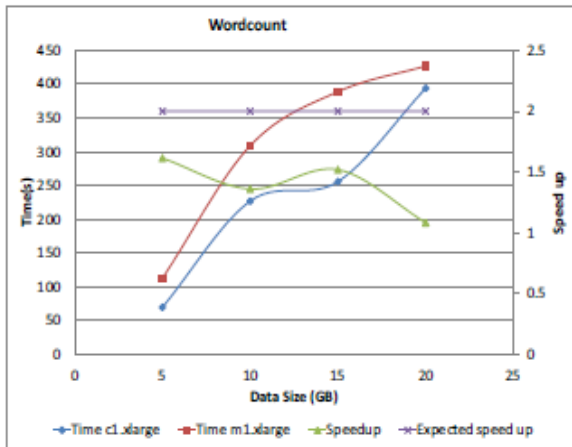


Figure 5 Wordcount Performance

## 6. CONCLUSION

In this paper, we implemented the cloud based framework based on mapreduce model. It is helpful tool in cloud computing for handling large datasets in cloud. Also evaluates the performance of this framework. We tested the framework with two applications in virtual clusters in cloud. These applications gave a better performance results than existing. So it is used as a simulator for cloud computing and data centers.

## 7. REFERENCES

- [1] IBM- Big Data
- [2] T. Cormen, *Introduction to algorithms*. The MIT press, 2001.
- [3] J. Shi, "Program scalability analysis," in *International Conference on Distributed and Parallel Processing*, Georgetown University, Washington D.C., October 1997.
- [4] NIST definition of cloud P. Mell, 2011
- [5] Agent based cloud computing K.M.sim, 2011
- [6] O. Malley and A. Murthy, "Winning a 60 second dash with a yellow elephant," *Proceedings of sort benchmark*, 2009.
- [7] An introduction to multi agent systems, M. Wooldridge.
- [8] Mapreduce by J.Dean, 2004
- [9] Introduction to data centers, cisco learning network.
- [10] Best practices for data centers, S. Greenberg.
- [11] Cloudera mapreduce algorithms- [blog.cloudera.com](http://blog.cloudera.com)