DATA DISTRIBUTION HANDLING ON CLOUD FOR DEPLOYMENT OF BIG DATA

Samip Raut, Kamlesh Jaiswal, Vaibhav Kale, Akshay Mote, Ms. Soudamini Pawar and Mrs. Suvarna Kadam

D. Y. Patil College of Engineering, Akurdi, Savitribai Phule Pune University, Pune

ABSTRACT

Cloud computing is a new emerging model in the field of computer science. For varying workload Cloud computing presents a large scale on demand infrastructure. The primary usage of clouds in practice is to process massive amounts of data. Processing large datasets has become crucial in research and business environments. The big challenges associated with processing large datasets is the vast infrastructure required. Cloud computing provides vast infrastructure to store and process Big data. Vms can be provisioned on demand in cloud to process the data by forming cluster of Vms . Map Reduce paradigm can be used to process data wherein the mapper assign part of task to particular Vms in cluster and reducer combines individual output from each Vms to produce final result. we have proposed an algorithm to reduce the overall data distribution and processing time. We tested our solution in Cloud Analyst Simulation environment wherein, we found that our proposed algorithm significantly reduces the overall data processing time in cloud.

KEYWORDS

Cloud Computing, Big Bata, Cloud Analyst, Map Reduce, Big data distribution

1. INTRODUCTION

Management and Processing of large dataset is becoming more important in research and business environment. Big data processing engines have experienced a tremendous growth. The big challenge with large data set processing is the infrastructure required, which can demand large investment. Thus, cloud computing can significantly reduce the infrastructure capital expenditure, providing new business models in which provider offer on-demand virtualized infrastructure. For accommodates varying workloads cloud computing presents the large scale on demand infrastructure. The main data crunching technique in which data is to the computational nodes, which were shared. Big data is a set of large datasets that cannot be processed by traditional computing technique. Big data technologies are important in providing more exact analysis, which may lead to further actual decision-making resulting in a greater operational efficiency, reduced risk for business and cost reduction. In order to handle big data there are several technologies from various vendors like Amazon, IBM, Microsoft, etc. For cloud computing as well as a distributed file system Hadoop provides an open source construction.

Hadoop uses the Map Reduce model. HDFS is a file system, to done the tasks it uses the Map Reduce. In which it reads the input in huge chunks, process on that input and finally write huge chunks of output. HDFS does not handle arbitrary access well. HDFS service is provided by two processes: Name Node and Data Node. Name Node handles the file system management and provides control management and services. Data Node provides block storage and retrieval services. In HDFS file system there will be one Name Node process, and this is a single point of failure. Hadoop Core provides the Name Node automatic backup and recovery, but there is no fail over services.

DOI: 10.5121/ijccsa.2016.6302

The main objective of our proposed system is to significantly reduce the data distribution time over virtual clusters for data processing in the cloud. This will help to provide faster and secure processing and management on large datasets. Various data distribution techniques are discussed in Section 2. Section 3 presents related work. Section 4 presents our proposed system design. Section 5 presents implementation details. Section 6, presents experimental results of executing task in Cloud Sim. Section 7 concludes the paper with future work.

2. DATA DISTRIBUTION TECHNIQUE

Data distribution is a technique of distributing the partitioned data over several provisioned VMs. In this technique, the load balancing factor need to taken into concern so as to distribute equal amount of data among all provisioned VMs. The different approaches for performing data distribution among provisioned VMs are[1]:

2.1 Centralised Approach

In this approach, a central repository is used to download the required dataset by VMs. In initialization script all VMs connects to the central repository, so after boot VMs can get the required data. A central server bandwidth becomes bottleneck for the whole transfer. The limitation of centralised approach if transfers are requested in parallel is central server will drop connections and get a “flash crowd effect” that can caused by thousands of VMs requesting blocks of data.

2.2 Semi-Centralised Approach

In centralised approach, if more VMs requested in parallel to the central server then server will drop connections. So, semi-centralised approaches potentially reduce the networking infrastructure stress. Then it would be possible to share the dataset across different machines in the data centre. By this VMs do not get the same shared at the same time. The limitation of this approach is when the datasets change over time. The datasets may grow or expands its size in time then it is difficult to foresee the bias.

2.3 Hierarchical Approach

If new data are continuously added then Semi centralised approach is very hard to maintain. In hierarchical approach, there is build a relay tree where data not gets from the original store by VMs, but data is get from parent node in the hierarchy. In this way all VMs will access the central server to fetch data, and again this fetch data is provide to other VMs and so on. The limitation of this approach is that it cannot provides fault tolerance during the transfer, and if one of the VM gets stuck then the VM deployments fails after the transfers have been initiated.

2.4 P2P Approach

Hierarchical approach requires more synchronization and some P2P streaming overlays like PPLive or Sopcast that are based on hierarchical multi trees (a node belongs into several trees) are used to implement this approach. In this approach, each system act as server as well as client. For accessing the VMs the data centre environment presents low-latency, no Firewall or NAT issues, and no ISP traffic shaping to deliver a P2P delivery approach for big data in the data centre.

3. RELATED WORK

Several data distribution handling technique are proposed. The most efficient data distribution technique is peer-to-peer. In [2], S. Loughran et al. presents framework that enables dynamic deployment of a MapReduce service in virtual infrastructures from either public or private cloud providers. The strategy is followed by popular MapReduce to move the computation to the location where data is stored rather than data to computational node[3]. The deployment process architecture creates a set of virtual machines (VMs) according to user-provided specifications. The framework automatically sets up a complete MapReduce architecture using a service catalog, then processes the data. In order to distribute data among provisioned VMs, the data partitioning is done. Data partitioning the data input is splits. This input splits into the multiple chunks. In partitioning service partitioning is done at a central location. Processing the data this data chunks to be distributed among the VMs.[1]. Service capacity of p2p system is modified in two regimes.

One is the transient phase and second is steady phase. . In transient phase analytical model busty demands is tries to catch up by system and trace measurement that exhibit the exponential growth of service capacity. In second regime, the steady phase the service capacity of p2p system scales with and track the offered load and rate at which the peer exit the system.[5]. In [6], Gang Chen presents BestPeer++ system in which integrating cloud computing, database, and peer-to-peer technologies to delivers elastic data sharing services. Previously to enhance the usability of P2P networks, database community have proposed a series of peer-to-peer database management system. In [7], Mohammed Radi proposed a Round Robin Service Broker policy is select the data centre to process the request. In [9], Tanveer Ahmed Yogendra Singhr presents a comparison of various policies utilized for load balancing using a tool called cloud analyst. The various policies that have being compared include Round Robin, Equally spread current execution load, Throttled Load balancing. In results the overall response time of Round Robin policy and ESCEL policy is almost same and that of Throttled policy is very low as compared to Round Robin policy and ESCEL policy. In [10], Soumya Ray and Ajanta De Sarkar present a concept of Cloud Computing along with research challenges in load balancing. It also focus on advantage and disadvantages of the cloud computing. It also focus on the study of load balancing algorithm and comparative survey of the algorithms in cloud computing with respect to resource utilization, stability, static or dynamicity and process migration. In [11], Pooja Samal, Pranati Mishra presents load distribution problem on various nodes of a distributed system which is solved by the proposed work. That improves both resource utilization and job response time by analyzing the variants of Round Robin algorithm.

4. DESIGN OF OUR PROPOSED SYSTEM

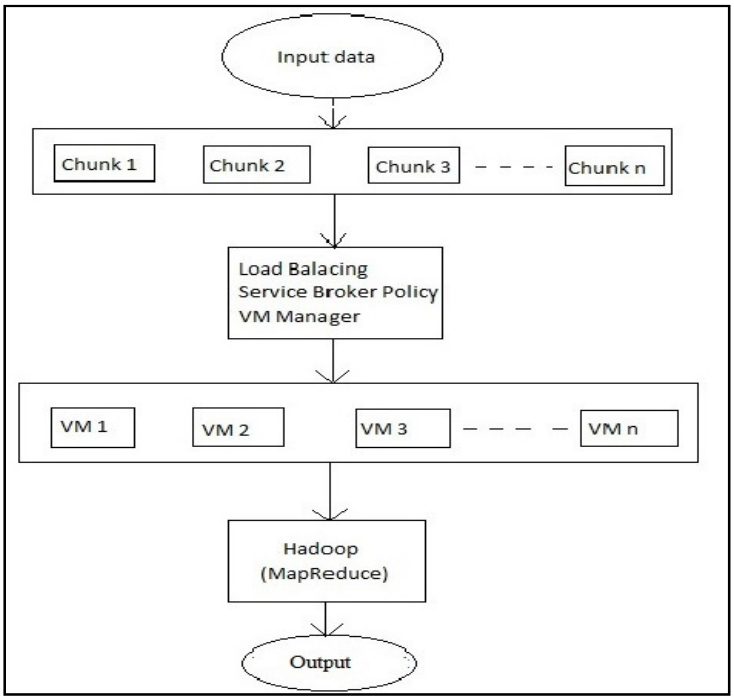

Figure 1: System Architecture

4.1 User Input Processing

The user can provide the some parameters to virtual infrastructure such as number and size of slave nodes. In addition to this user also specify the input files and the output folder location. This is the only done by end user and user interaction with the system, proposed framework done the rest of the process automatically.

4.2 Centralised Partitioning

The user can upload huge file on cloud. Centralise data partitioning split the input bulk data into multiple chunks. User can upload Large data file that file are partitioned and stored in cloud severs, in partitioning break the large files into a smaller chunks and it also reduces the storage server burden. Partition takes automatically when file is uploaded.

4.3 Data Distribution

The data chunks from the data repository are distributed to the VMs. These VMs will process the data. Distribute the data on each VM is an NP-hard problem. Our proposed system uses Peer-to-Peer data distribution technique in which there is point to point connection between the VMs. Service capacity of p2p system is modified in two regimes. One is the transient phase and second is steady phase. In transient phase analytical model busty demands is tries to catch up by system and trace measurement that exhibit the exponential growth of service capacity. In second regime, the steady phase the service capacity of p2p system scales with and track the offered load and rate at which the peer exit the system.

5. IMPLEMENTATION DETAILS

Installation of Ubuntu 14.04, Hadoop 2.7.1, Eclipse. Additional component that are require by the Hadoop are installed using apt-get-install necessary Linux packages such as java JDK 1.7.

5.1 Cloud Analyst:

Cloud Analyst is a GUI based tool that has been developed on CloudSim architecture. CloudSim is a toolkit that is allowed to do modeling, simulation and other experimentation. The main problem in CloudSim is that every work has to be done programmatically. It allow user to do repeated simulation with small change in parameter very simply and rapidly. The cloud analyst allows setting location of the users that generating application and also location of data centers. In Cloud Analyst various configuration parameter can be set like number of users, number of VMs, number of processors, network bandwidth, amount of storage and other parameters. Based on parameters tool computes the simulation result and show them in a graphical form. The result consists of response time, processing time, and cost.[13]

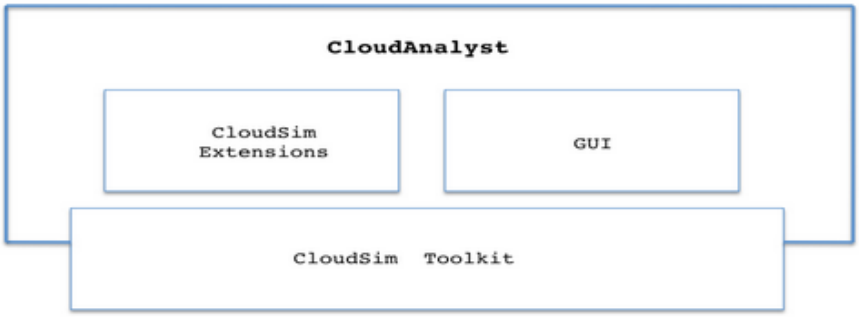

Figure 2: Cloud Analyst Architecture [13]

5.2 Methodologies of Problem solving and Efficiency issues

The two main methodology used to balance the load among the VMs are Throttled and Round Robin VM load balancing policy.

a) Throttled: In throttled algorithm to perform the required operation the client requests the load balancer to find a suitable VM. Firstly the process started by maintaining the entire VMs list, each row is indexed individually to speed up the lookup process. If a match of the machine is found on the basis of size and availability, then the load balancer accepts the request of the client and allocates that VM to the client. If there is no VM available that matches the criteria then the load balancer returns -1 and the request is queued.

b) Round Robin: Round Robin is the simplest scheduling techniques that utilize the principle of time slices. In Round Robin the time is divided into multiple slices and a particular time slice is given to each node i.e. it utilizes the time scheduling principle. A quantum is given to each node and the node will perform its operations in this quantum. On the basis of this time slice the resources are provided to the requesting client by the service provider. Therefore, Round Robin algorithm is very simple but in this algorithm on the scheduler there is an additional load that decide the quantum size.[8]

6. RESULT ANALYSIS

Round Robin Load Balancer Algorithm and Throttled load balancer algorithm are implemented for a simulation. Java language has been used for implementing VM load balancing algorithm. In cloud analyst tool configuration of the various components need to be set to analyze load balancing policies. We have to set the parameters for the data center configuration, user base configuration, and application deployment configuration. We have taken two data centers. The duration of simulation is 60hrs.

Parameter	**Value Used**
VM Image Size	10000
VM Memory	1024 MB
VM Bandwidth	1000
Architecture(Data Center)	X86
Operating System(Data Center)	Linux
VMM(Data Center)	Xen
No. of Machines(Data Center)	50
No. of processors per machine(Data Center)	4
Processor Speed(Data Center)	100 MIPS
VM Policy(Data Center)	Time Shared
User Grouping Factor	1000
Request Grouping Factor	100
Executable Instruction Length	250

Table 1: Parameters value

Following table show a overall response time of VM load balancing algorithm.

Number of VM's	Overall average response time (milliseconds)	
	Round Robin	Throttled
50	220.9	150.93
100	226.18	151.12
200	228.81	152.85

Table 2: Comparison of average response time of VM Load balancing Algorithm

In Cloud Analyst result are computed after performing the simulation is as shown in the following figures.

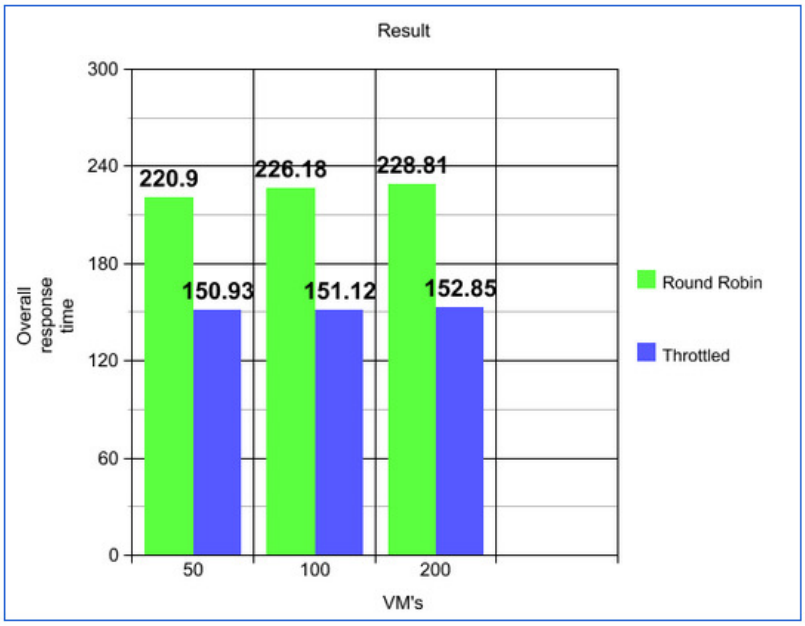

Figure 3: Comparison of average response time of VM Load balancing Algorithm

The above shown figure 3 and table 2 clearly indicates that the overall response time in Round Robin are much greater whereas overall response time are improved in Throttled algorithm. Therefore, we can easily identify among all the algorithm Throttled algorithm is best. In this live migration of load is done in virtual machine.

7. CONCLUSION

This paper presents a concept of cloud computing and focus on the challenges of the load balancing. It also focus on the time requires to process the big data. We have also gone through the comparative study of big data, hadoop and cloud computing. We have also seen the connectivity of cloud with the hadoop. Major amount of time is given for the study of different load balancing algorithm, followed by hadoop's map-reduce for data partitioning. This paper aims to achieve qualitative analysis on previous VM load balancing algorithm and then implemented in CloudSim and java language along with the hadoop. Load balancing on the cloud will develop the performance of cloud service substantial. It will prevent overloading of the server which degrades the performance and response time will also be improved. We have simulated two different scheduling algorithms for executing user request in a cloud environment. Every algorithm is observed and their scheduling criterion likes average response time, data centre derive. We efficiently used the hadoop in order analysis the data which is enhanced in cloud.

8. REFERENCES

[1] Luis M. Vaquero, Member, Antonio Celorio, Felix Cuadrado, Member, IEEE, and Ruben Cuevas, (2015) "Deploying Large-Scale Datasets on-Demand in the Cloud: Treats and Tricks on Data Distribution" IEEE Trans. vol. 3, no. 2.

[2] S.Loughran, J.Alcaraz, Caleroand J.Guijarro, (2012) "Dynamic cloud deployment of a mapreduce architecture," IEEEInternetComput.,vol.16,no.6,pp.40–50.

[3] Jeffrey Dean and Sanjay Ghemawat,(2008) **"**MapReduce: Simplified Data Processing on Large Clusters".

[4] L. Garcés-EriceAffiliated withInstitut EURECOM, E. W. Biersack, P. A. Felbe, K. W. Ross, G. Urvoy-Keller,(2003) "Hierarchical Peer-to-Peer Systems" Volume 2790 of the series Computer Science pp 1230-1239.
[5] Xiangying Yang and Gustavo de Veciana, (2004) "Service Capacity of Peer to Peer Networks" IEEE
[6] Gang Chen, Tianlei Hu, Dawei Jiang, Peng Lu, Kian-Lee Tan, Hoang Tam Vo, and Sai Wu, (2014) "BestPeer++:A Peer-to-Peer Based Large-Scale Data Processing Platform" IEEE Trans. vol. 26, no.
[7] Mohammed Radi, (2014)"Efficient Service Broker Policy For Large-Scale Cloud Environments" IJCCSA, Vol.2, No.
[8] Tejinder Sharma, Vijay Kumar Banga, (2013)"Efficient and Enhanced Algorithm in Cloud Computing" IJSCE ISSN: 2231-2307, Volume-3, Issue-1
[9] Tanveer Ahmed , Yogendra Singh, (2012)"Analytic Study Of Load Balancing Techniques Using Tool Cloud Analyst." IJERA, Vol. 2, Issue 2, pp.1027-1030
[10] Soumya Ray and Ajanta De Sarkar (2012)"Execution Analysis of Load Balancing Algorithm in Cloud Computing Environment" IJCCSA, Vol.2, No.5
[11] Pooja Samal, Pranati Mishra, (2013)"Analysis of variants in Round Robin Algorithms for load balancing in Cloud Computing" IJCSIT, Vol. 4 (3) , 416-419
[12] Samip Raut, Kamlesh Jaiswal, Vaibhav Kale, Akshay Mote, Soudamini Pawar, Hema Kolla (2016)"Survey on Data Distribution Handling Techniques on Cloud" IJRAET, Volume-4, Issue -7
[13] Bhathiya Wickremasinghe "CloudAnalyst: A CloudSim-based Tool for Modelling and Analysis of Large Scale Cloud Computing Environment".

AUTHORS

Samip Raut is pursuing the Bachelor's degree in Computer Engineering from SPPU, Pune. His area of interest includes Cloud Computing and Networking.

Kamlesh Jaiswal is pursuing the Bachelor's degree in Computer Engineering from SPPU, Pune. His area of interest includes Cloud Computing and Big Data.

Vaibhav Kale is pursuing the Bachelor's degree in Computer Engineering from SPPU, Pune. Area of interest includes Cloud Computing and Big Data.

Akshay Mote is pursuing the Bachelor's degree in Computer Engineering from SPPU, Pune. His area of interest includes Cloud Computing and Big Data.

Ms. Soudamini Pawar has received her BE(CSE) from Gulbarga University, Gulbarga and her ME from SPPU, Pune. She has teaching experience of about 12 years. She is currently working as Assistant Professor in D. Y. Patil college of Engineering, Akurdi, Pune.

Mrs. Suvarna Kadam has completed her PG in Computer Engineering from SPPU, Pune. She has 15 years of experience in computing with variety of roles including developer, entrepreneur and researcher. She is currently working at Department of Computer Engineering as Asst. Professor and enjoys guiding UG students in the state of the art areas of research including machine learning and high performance computing.