International Journal of Recent Development in Engineering and Technology
Website: www.ijrdet.com (ISSN 2347-6435(Online) Volume 11, Issue 04, April 2022)

Modify Load Balancing in Cloud Database Environment

Loveleen kaur[1], Aarohi Khaira[2], Nayan Tara Ganguli[3], Aaditya Sijariya[4], Taaniya Saini[5]

[1]Assistant professor, [2,3,4,5]UG Student, Department of Information Technology, JEC Jabalpur M. P, India

[1]pabla@jecjabalpur.ac.in, [2]khaira.aarohi@gmail.com, [3]nayantaraganguli786@gmail.com, [4]mohit7sijariya@gmail.com, [5]taaniyasaini18@gmail.com

Abstract-- **Cloud Computing is an important component of cloud system. The term, clouds, is mainly evaluated by ubiquity of information comparing with resource scaling. In clouds, minimizing use of computer hardware, software failures and mitigating recourse limitations. This work discusses the load balancing in cloud computing and then demonstrates a case study of system availability based on a typical Hospital Database Management solution. The main aim of this paper is to develop and implement an Optimized Load balancing algorithm in IaaS virtual cloud environment that aims to utilize the virtual cloud resources efficiently.It minimizes the cost of the applications by effectively using cloud resources and identifies the virtual cloud resources that mustbe suitable for all the applications. The web application is created with many modules.**

***Keywords:*—Cloud Computing, Resource allocation, Priority based Scheduling, Loadbalancing.**

I. Introduction

Fundamentally the cloud computing role is to determine the time that the system is up and running correctly; the length of time between failures and the length of time needed to resume operation after a failure. Availability needs to be analyzed through the use of presence information, forecasting usage patterns and dynamic resource scaling[1]. The evolution of cloud computing can handle such massive data as per on demand service. Nowadays the computational world is opting for pay-for-use models and Hype and discussion aside, there remains no concretedefinition of cloud computing. In this paper, we first develop a comprehensive taxonomy for describing cloud computing architecture. It is true that this may look like a very specific situation, and it can be argued that the hosting provider should not care about application performance and utilization, but it is up to the hosting provider to provide additional precautions through the implementations of advanced load balancing methods in order to avoid possible shortcomings in application design.[2] Load balancing models and algorithms proposed in the literature or applied in open source or commercial load balancers rely either on session switching at the application layer, packet-switching mode at the network layer or processor load balancing mode.

The analysis of detected issues for those load balancing algorithms is presented in this paper, as a preparation phase for a new load balancing model (algorithm) proposition.

This article discusses possible ways to improve the performance of cloud networks by the introduction of resource load balancing technique that uses the message-oriented middleware within the web service oriented model of software architecture.

Cloud computing[3] is the use of computing resources (hardware and software) that are delivered as a service over a network (typically the Internet). The name comes fromthe use of a cloud-shaped symbol as an abstraction for the complex infrastructure.[4] Cloud computing entrusts remote services with a user's data, software and computation. Users access cloud based applications through a web browser or a light-weight desktop or mobile application while the business software and user's data are stored on servers at a remote location. [5]

Load Balancing

Load balancing is described in [6] as follows “In a distributed network of computing hosts, the performance of the system can depend crucially on dividing up work effectively across the participating nodes”. It can also generally be described as anything from distributing computation and communication evenly among processors, or a system thatdivides many client requests among several servers.

Problem Definition

A local cluster managed by virtual machine technology to supply its user with resource required by their application is considered. The cloud computing environment is established by making the virtual resource ofa machine and sharing the virtual resource as per the user specification. If the number of user to the particular virtual machine exceeds the load balancing server will redirect the newincoming user’s request to the other virtual machines in node controller. But, this is common mechanism which does notconcentrate on time, throughput andefficiency. [7-9]

II. Architectural Framework Model

The framework model of our proposed load balancing system for multiple workflows is shown in Figure 1. It describes the working mechanism of load balancing technique in IaaS cloud environment [10].

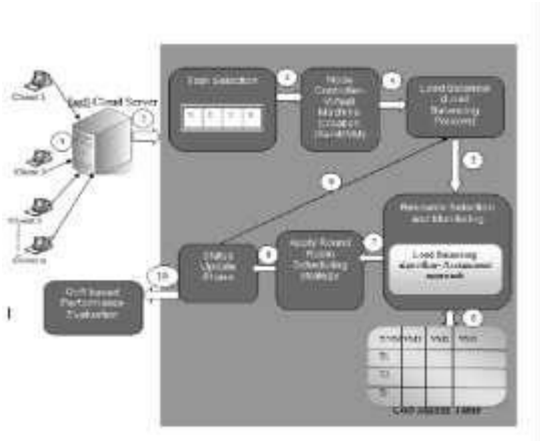

Figure 1: Architectural Framework Model

Deployment Model of Cloud:

A. Private Cloud The cloud infrastructure is wrought exclusively for all organization build his over cloud. Private cloud permits organization to retain total control over their infrastructure, application and data.

B. Public Cloud It is the most shared and prevalent form in which cloud infrastructure is made available to the universal public or a huge organization to universal is maintained by an industry trade cloud service which means that pay as per use.[11]

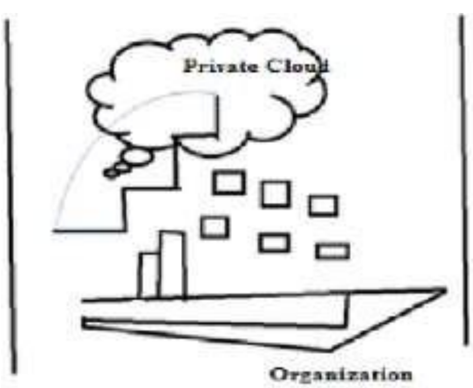

Figure 2 : Model of Cloud

In cloud computing the scheduling of virtual machine requests is an important issue. The requested tasks can be completed in a minimum time according to the user defined time. In [12] to evaluate a scheduling algorithm that is an efficient technique for scheduling virtual machines between servers. In this the scheduling technique results are compared. Comparing these techniques the priority based scheduling algorithm improves the resource utilization and reduce the waiting time.

There are many nodes in a public cloud which are at different locations. The cloud has a main controller (MC) which chooses the suitable partitions for arriving jobs. The appropriate partition is selected by using best load balancing strategy[13]. All the status information is gathered and analyzed by main controller and balancers. They also perform the load balancing operations. The system status then provides a basis for choosing the right load balancing strategy. In this paper we will use approximately 4 different servers, which are partitioned into small clouds called balancers (each balancer will have some servers) figure 3. Cloud Service Provider (CSP) is used to handle a Main cloud (which is made up of small Clouds) called Main Controller or Controller main. Client interacts with cloud using a web application called client Site. When client uploads file it will be stored in the server. The cloud will take care that it will be loaded into the server which has minimum load [14].

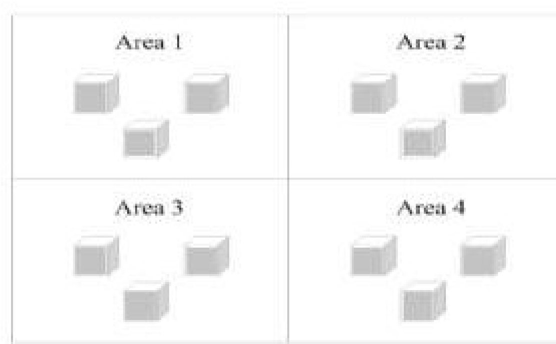

Figure 3: Four Quadrate Model of Cloud Area

Algorithms: The status of every server is updated by the balancers and depending on the status the partition is selected. The cloud partition status can be divided into three types: (1) *Idle:* When the load exceeds alpha (2) *Normal:* When the load exceeds beta (3) *Overload:* When the load exceeds gamma The parameters alpha, beta, and gamma are set by the cloud partition balancers.

Best PartitionSearching Algorithm:

Begin

While User_reques doBest _ partition_searching_strategy(User_request);

If partition_status == idle OR partition_status == normal then Assign user_request to Partition;

Else

Search for another Part;

End if

End while

End

Following is the process of flow work figure 4[15,16]

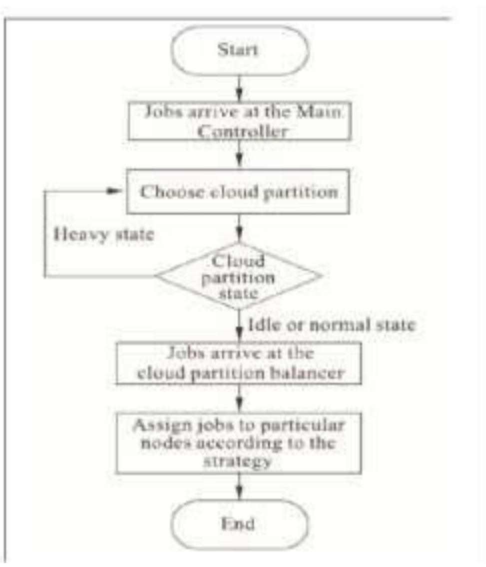

Figure 4 : Flowchart of Partition Searching Algorithm

The establishment of an effective load balancing algorithm and how to use Cloud computing resources efficiently for effective and efficient cloud computing is one of the Cloud computing service provider's ultimate goals. The basic idea of cloud computing is to offer resources such as VMs as services on demand.

Allocating efficient VM on demand is being carried out with the help of the load balancing algorithms in the cloud computing. As the load balancing algorithm plays a vital role while deciding which VM is to be allocated on demand of the user. While providing services it is likely to have a number of requests at a time and due to that some requestors need to stay in queue though they have possibility to send request to other service provider.

A. Round Robin Algorithm figure 5 It is a static load balancing algorithm, which does not consider the previous load state of a node at the time of assigning jobs. It makes use of the round robin scheduling algorithm for allocating jobs. It selects the first node arbitrarily and then, allocates jobs to all other nodes in a round robin manner [17]. This algorithm works on random selection of the virtual machines. The datacenter controller allocates the requests to a list of VMs on a rotating basis. The first request is allocated to a VM chosen randomly from the group and then the Data Center controller assigns the requests in a circular order. Once the VM is allotted the request, the VM is shifted to the end of the list [18,19].

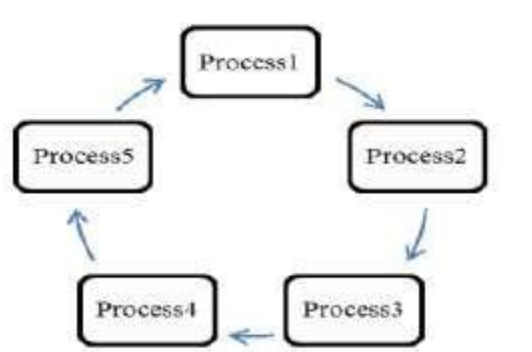

Figure 5 : Static Load Balancing Algorithm Process

III. ANALYSIS

Figure 6 shows an interface designedfacilitating the user to create bucket on Cloud and send request to the instance monitor.

Client provides an option for creating bucket, deleting bucket, refreshing AWS amazon web account and downloading the contents resides in the bucket. When user sends a request tothe Cloud, IaaS converts it into an instanceand sends it to job manager.

Figure 6: Client Interphase

Figure 7 shows an Job manager, receives the user request from client. Received instance is to be divided into number of sub tasks (instances). After dividing an instance sends itto task manager.

Figure 7: Job Manager

IV. Experimental Setup

In this section we describe the experimentalsetup that was used to run workflows. Java language is used for implementing VM loadbalancing algorithm.

EC2 was chosen because it is currently the most popular, feature-rich, and stable commercial Cloud Workflows areloosely- coupled parallel applications that consist of a set of computational tasks linkedvia data and control-flow dependencies. Unlike tightly-coupled applications in whichtasks communicate directly via the network.

[20] In order to have an unbiased comparison of the performance of workflows on EC2 the experiments presented in this paper attempt to account for these differences by (a) runningall experiments on single nodes and (b) running experiments using the local disk on EC2. Although single-node experiments do not enable us to measure the scalability of Cloud services they do provide an application -oriented understanding of the capabilities of the underlying resources that can help in making provisioning decisions. Testing the scalability of Cloud services when running workflows on multiple nodes is left forfuture work. [21]

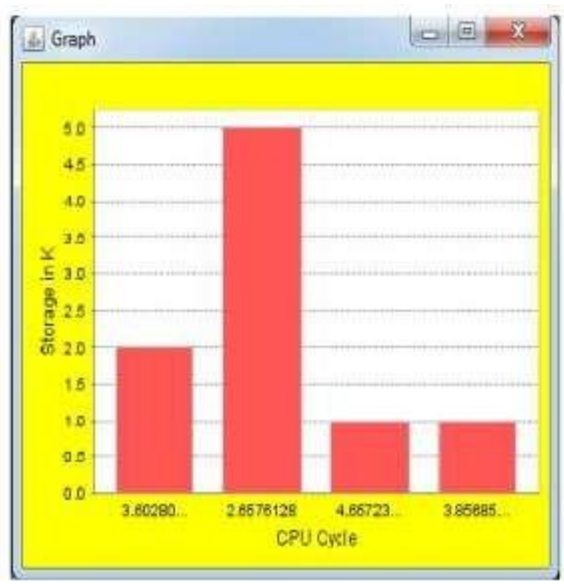

Figure 8: Shows the graphical representation of theTable 2 with respect to Cpu cycle and Storage in Kb.

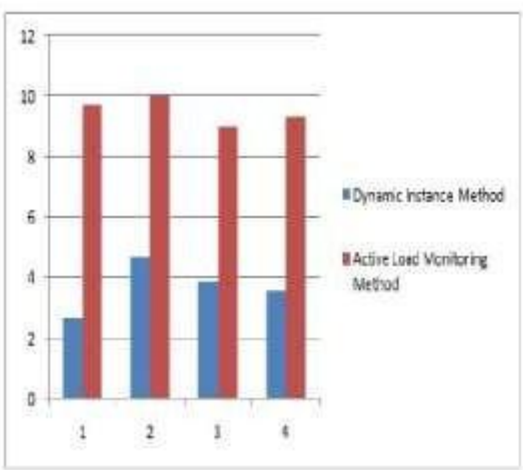

Figure 9 Shows the comparison graphical representation of Table 1 and Table 2 with respect to Cpu cycle. X – axis describes number of instances, Y –axis describes CPU cycle.

Determine whether to trigger the live migration or not**.** After the control node collects the load values from all computational nodes, the total and average utilizations of all the machines involved in load balancing are computed.[22] We adopt a threshold based strategy for deciding when virtual machines should be migrated between the nodes. The VM live migration will be triggered if Cm, the mean value of the sum of the maximum and minimum distances (respectively Cdif fmax and Cdif fmin) from the average utilization (Cavg), is greater than a threshold T. Ci is the total CPU utilization of the i th computational node, i ∈ 1,.., n; n is the current number of nodes in the VM environment. Cavg = 1 n Pn i=1 Ci Cdif fi = Ci − Cavg Cdif fmax = maxi∈{1,..,n} Cdif fi Cdif fmin = mini∈{1,..,n} Cdif fi Cm = (Cdif fmax + Cdif fmin)/2. [23]

Schedule the live migration by checking the load balancing history record**.** Whenever a migration is triggered, the control node checks the history record for a similar CPUs utilization scenario, i.e. a similar load distribution on the computational nodes. Note that the same Cavg doesn't ensure the same CPU utilizations scenario; the mapping is the key. If a previous record exists, we can schedule the VM live migration by choosing the same source and destination nodes. If several similar records exist, we just follow the latest record and schedule the migration.

If we can't find such a record, the current situation is totally new and the nodes with Ci value close to Cdif fmax are used as sources, while the nodes with Ci value close to Cdif fmin are used as destinations of the live migration. After the migration, we add such situation (or mapping) as a new entry in the history record. [24,25]

Migration Mechanisms Comparison

We evaluate the performance of migrating VMs running two types of workloads: "generic", by runing the Apache [4] webserver and "memory-intensive", by running Sysbench [3] (e.g., 25% read operations and 75% write operations). We let each node run 4 VMs with the same assigned memory size. We conduct the evaluation while varying the guest memory size from128MB to 512MB, in order to investigate the impact of memory size on the downtime and other performance characteristics[26]. To verify the effectiveness of our adaptive migration mechanism we again add another competitor, DCbalance-i. We refer to our migration design for generic applications as DCbalance, whereas we refer to its improved version which better handles memory- intensive applications as DCbalance-i. [27]

REFERENCES

[1] M. Armbrust, et al., A view of cloud computing. Commun. ACM. vol. 53 (2010), pp. 50-58

[2] Zh o n g X u , R ong H u a n g , "Performance Study of Load Balancing Algorithms in Distributed Web Server Systems", CS213 Parallel and Distributed Processing Project Report.

[3] Cary Landis and Dan Blacharski, "Cloud Computing Made Easy", Version 0.3.

[4] Bhaskar Prasad Rimal, Eummi Choi and Ian Lumb, "A Taxonomy and Survey of Cloud Computing System", Proceedings of the 5th International Joint Conference on INC, IMS and IDC, pp. 44-51, 2009.

[5] Tharam Dillon, Chen Wu and Elizabeth Chang, "Cloud Computing: Issues and Challenges", Proceedings of 24th International Conference on Advanced Information Networking and Applications, pp. 22-33, 2010.

[6] E. Anshelevich, D. Kepme, J. Kleinberg, "Stability of Load Balancing Algorithms in Dynamic Adversarial Systems", Proceeding of the 34th annual ACM symposium on Theory of Computing, 2002.

[7] Sandeep Sharma, Sarabjit Singh and Meenakshi Sharma "Performance Analysis of Load Balancing Algorithms", World Academy of Science, Engineering and Technology 38, pp. 269- 272, 2008.

[8] B. Sotomayor, R.S. Montero, I.M. Llorente and I. Foster, "Virtual Infrastructure Management in Private and Hybrid Clouds", IEEE Internet Computing, Vol. 13, No. 5, pp. 14- 22, 2009.

[9] Martin Randles, David Lamb and A. Taleb-Bendiab, "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing", Proceedings of IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, pp. 551-556, 2010.

[10] Aobing Sun, Tongkai Ji, Qiang Yue and Feiya Xiang, "IaaS Public Cloud Computing Platform Scheduling Model and Optimization Analysis", International Journal of Communications, Network and System Sciences, Vol. 4, No. 12, pp. 803-811, 2011.

[11] Prabavathy.B, Priya.K, Chitra Babu "A Load Balancing Algorithm For Private Cloud Storage" IEEE - 316614th ICCCNT 2013 July 4-6, 2013, Tiruchengode, India

[12] Amin Mehranzadeh and Seyyed Mohsen Hashemi (2013), A Novel-Scheduling Algorithm for Cloud Computing based on Fuzzy Logic", International Journal of Applied Information Systems (IJAIS), vol.5, pp 28-31.

[13] Kumar Nishant, Pratik Sharma,Vishal Krishna "Load Balancing of Nodes in Cloud Using Ant Colon y O p t i m i z a t i o n " 2 0 1 2 1 4 t h International Conference on Modelling and Simulation.

[14] J. P. Mell and T. Grance, "The NIST definition of cloud computing", http://csrc.nist.gov/ publications/ nistpubs/800- 145/SP800-145.pdf, 2012.

[15] N. G. Shivaratri, P. Krueger, and M. Singhal, "Load distributing for locally distributed systems", Computer, vol. 25, no. 12, pp. 33-44, Dec. 1992.

[16] S. Aote and M. U. Kharat, "A game- theoretic model for dynamic load balancing in distributed systems", in Proc. The International Conference on Advances in Computing, Communication and Control (ICAC3 '09), New York, USA, 2009, pp. 235-238.

[17] S h a n t i S w a r o o p M o h a r a n , Rajadeepan D. Ramesh, Digamber Powar, Analysis of load balancers in cloudcomputing, International Journal of Computer Science and Engineering,Vol. 2, Issue 2, May 2013, pp.101-108.

[18] Yatendra Sahu, R.K. Pateriya, Cloud Computing Overview with Load Balancing Techniques, International Journal of Computer Applications, Volume 65– No.24, March 2013.

[19] Meenakshi, Comparative Study of Load Balancing Algorithms in Cloud Computing Environment, International Journal of Engineering Research & Technology, Vol. 2 Issue 10, October 2013.

[20] G. Soundararajan et at al, "Resource Allocation for Database Servers Running on Virtual Storage", In Proc. of 6th USENIX Conference on File and Storage Technologies, 2009.

[21] Meenakshi Sharma, Pankaj Sharma, Dr. Sandeep Sharma, Efficient Load Balancing Algorithm in VM Cloud Environment, IJCST Vol. 3, Issue 1, Jan. - March 2012.

[22] Z. Li, W. Luo, X. Lu, and J. Yin. A live migration strategy for virtual machine based on performance predicting. In Computer Science Service System (CSSS), 2012 International Conference on, pages 72–76, Aug. 2012.

[23] C. Liu, B. T. Loo, and Y. Mao. Declarative automated cloud resource orchestration. pages 26:1–26:8, 2011.

[24] R. Subrata, A. Y. Zomaya, and B. Landfeldt. Cooperative power-aware scheduling in grid computing environments. volume 70, pages 84– 91, Orlando, FL, USA, Feb. 2010. Academic Press, Inc.

[25] C. A. Waldspurger. Memory resource management in vmware esx server. SIGOPS Oper. Syst. Rev., 36:181– 194, December 2002.

[26] C. Wang, F. Mueller, C. Engelmann, and S. L. Scott. Proactive process- level live migration in hpc environments. In Proceedings of the 2008 ACM/IEEE conference on Supercomputing, SC '08, pages 43:1– 43:12, Piscataway, NJ, USA, 2008.

[27] J. Xu, M. Zhao, J. Fortes, R. Carpenter, and M. Yousif. Autonomic resource management in virtualized data centers using fuzzy logic-based approaches. Cluster Computing, 11 (3):213–227, Sept. 2008.

[28] R. Subrata, A. Y. Zomaya, and B. Landfeldt. Cooperative power-aware scheduling in grid computing environments. volume 70, pages 84– 91, Orlando, FL, USA, Feb. 2010. Academic Press, Inc.

[29] C. A. Waldspurger. Memory resource management in vmware esx server. SIGOPS Oper. Syst. Rev., 36:181– 194, December 2002.

[30] C. Wang, F. Mueller, C. Engelmann, and S. L. Scott. Proactive process- level live migration in hpc environments. In Proceedings of the 2008 ACM/IEEE conference on Supercomputing, SC '08, pages 43:1– 43:12, Piscataway, NJ, USA, 2008.

[31] J. Xu, M. Zhao, J. Fortes, R. Carpenter, and M. Yousif. Autonomic resource management in virtualized data centers using fuzzy logic-based approaches. Cluster Computing, 11 (3):213–227, Sept. 2008.