ISSN 2277-3061
Volume 16 Number 1
INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY

A REVIEW ON DYNAMIC RESOURCE ALLOCATION BASED ON LEASE TYPES IN CLOUD ENVIRONMENT

Sumanpreet Kaur [(1)], Mr. Navtej Singh Ghumman [(2)]
[(1)] Research Scholar, Department of Computer Science & Engineering, SBSSTC, Ferozepur, Punjab.
sumanmanes07@gmail.com
[(2)] Assistant Professor, Department of Computer Science & Engineering, SBSSTC, Ferozepur, Punjab.
navtejghumman@yahoo.com

ABSTRACT

Load balancing is one of the essential factors to enhance the working performance of the cloud service provider. Cloud Computing is an emerging computing paradigm. It aims to share data, calculations, and service transparently over a scalable network of nodes. Since Cloud computing stores the data and disseminated resources in the open environment. Since, cloud has inherited characteristic of distributed computing and virtualization there is a possibility of machines getting unused. Hence, in this paper, different load balancing algorithms has been studied. Different kinds of job types have been discussed and their problems have been reviewed. In the cloud storage, load balancing is a key issue. It would consume a lot of cost to maintain load information, since the system is too huge to timely disperse load. Load balancing is one of the main challenges in cloud computing which is required to distribute the dynamic workload across multiple nodes to ensure that no single node is overwhelmed.

Keywords

Cloud Computing, Datacenter Broker, Virtual Machine, Host, Load balancing.

INTRODUCTION

Data and programs are being swept up from desktop PCs and corporate server rooms and installed in “the compute cloud”. With its new way to deliver services while reducing ownership, improving responsiveness and agility, and especially by allowing the decision makers to focus their attention on the business rather than their IT infrastructure [1], there is no organization that has not though about moving to the Cloud. Cloud Computing has become one of the most talked about technologies in recent times and has got lots of attention from media as well as analysts because it is offering lots of opportunities. Enterprises have been determined to reduce computing costs and for that reason most of them started using it in IT technology then adapted virtualization technology. For the good of the enterprises it is futuristic to help them in this i.e. Cloud Computing. Cloud Computing has taken the enterprise to new level and allows them to further reduce costs through improved utilization, reduced administration and infrastructure cost and faster deployment cycles.

Cloud Computing is a term used to describe both a platform and type of application. As a platform it supplies, configures and reconfigures servers, while the servers can be virtual machine or physical machine. The cloud is a representation for the Internet and is an abstraction for the complex infrastructure it conceals. There are some important points in the definition to be discussed regarding Cloud Computing. Cloud Computing differs from traditional computing paradigms as it is scalable, can be encapsulated as an abstract entity which provides different level of services to the clients, driven by economies of scale and the services are dynamically configurable. Different researchers have stated various benefits of cloud computing due to this reason they have been adopted by enterprises more preferbale. Cloud Computing infrastructure allows enterprises to achieve more efficient use of their IT hardware and software investments. This is achieved by breaking down the physical barrier inherent in isolated systems, automating the management of the group of the systems as a single entity. Cloud Computing can also be termed as virtualized system and a natural evolution for data centers which offer automated systems management.

CHARACTERISTICS OF CLOUD COMPUTING

NIST specify five characteristics of cloud computing[9]:

On-demand self-service involves customers using a web site or similar control panel interface to provision computing resources such as additional computers, network bandwidth or user email accounts, without requiring human interaction between customers and the vendor.

Broad network access enables customers to access computing resources over networks such as the Internet from a broad range of computing devices such as laptops and smartphones.

Resource pooling involves vendors using shared computing resources to provide cloud services to multiple customers. Virtualisation and multi-tenancy mechanisms are typically used to both segregate and protect each customer and their data from other customers, and to make it appear to customers that they are the only user of a shared computer or software application.

Rapid elasticity enables the fast and automatic increase and decrease to the amount of available computer processing, storage and network bandwidth as required by customer demand.

Pay-per-use measured service involves customers only paying for the computing resources that they actually use, and being able to monitor their usage. This is analogous to household use of utilities such as electricity.

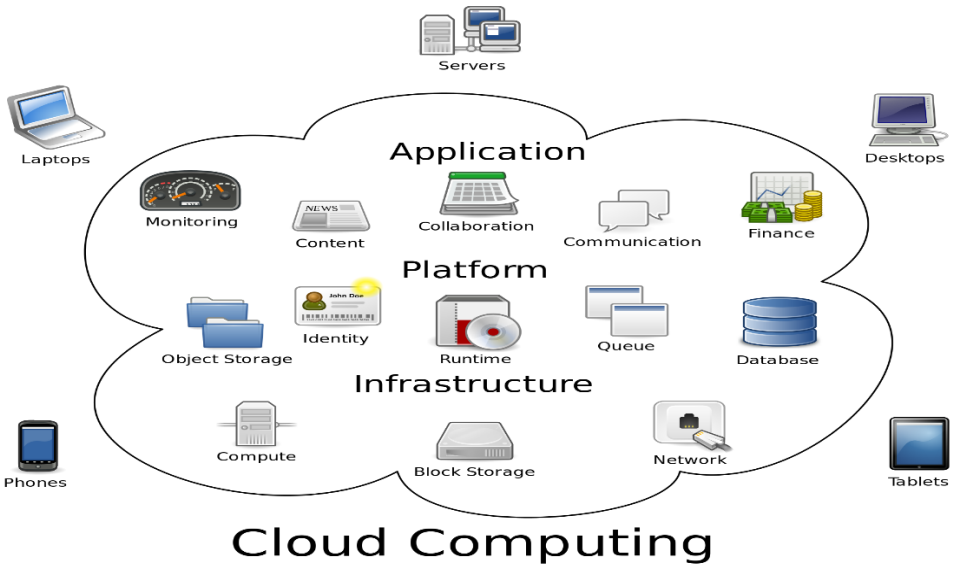

Figure 1.1 Cloud Computing Model

As cloud computing is in its evolving stage, so there are many problems prevalent in cloud computing [11],[16]. Such as:

- Ensuring proper access control (authentication, authorization, and auditing).
- Network level migration, so that it requires minimum cost and time to move a job.
- To provide proper security to the data in transit and to the data at rest.
- Data availability issues in cloud.
- Legal quagmire and transitive trust issues.
- Data lineage, data provenance and inadvertent disclosure of sensitive information is possible .

And the most prevalent problem in Cloud computing is the problem of load balancing.

Load balancing is the pre requirements for increasing the cloud performance and for completely utilizing the resources. Load balancing is centralized or decentralized. Load Balancing algorithms are used for implementing. Several load balancing algorithm are introduced like round robin algorithm a mining improvement in the performance. The only differences with this algorithm are in their complicity. The effect of the algorithm depends on the architectural designs of the clouds [4]. Today cloud computing is a set of several data centres which are sliced into virtual servers and located at different geographical location for providing services to clients. The objective of paper is to suggest load balancing for such virtual servers for higher performance rate.

In general, load balancing algorithms follow two major classifications:

- Depending on how the charge is distributed and how processes are allocated to nodes (the system load);
- Depending on the information status of the nodes (System Topology).

In the first case it is designed as centralized approach, distributed approach or hybrid approach and in the second case as static approach, dynamic or adaptive approach.

a) Classification According to the System Load

- **Centralized approach**: In this approach, a single node is responsible for managing the distribution within the whole system.
- **Distributed approach**: In this approach, each node independently builds its own load vector by collecting the load information of other nodes. Decisions are made locally using local load vectors. This approach is more suitable for widely distributed systems such as cloud computing.
- Mixed approach: A combination between the two approaches to take advantage of each approach.

b) Classification According to the System Topology

- **Static approach:** This approach is generally defined in the design or implementation of the system.
- **Dynamic approach**: This approach takes into account the current state of the system during load balancing decisions. This approach is more suitable for widely distributed systems such as cloud computing.

- **Adaptive approach**: This approach adapts the load distribution to system status changes, by changing their parameters dynamically and even their algorithms. This approach is able to offer better performance when the system state changes frequently. This approach is more suitable for widely distributed systems such as cloud computing.

LOAD BALANCING ALGORITHMS

In order to balance the requests of the resources it is important to recognize a few major goals of load balancing algorithms:

a) **Cost effectiveness**: primary aim is to achieve an overall improvement in system performance at a reasonable cost.

b) **Scalability and flexibility**: the distributed system in which the algorithm is implemented may change in size or topology. So the algorithm must be scalable and flexible enough to allow such changes to be handled easily.

c) **Priority**: prioritization of the resources or jobs need to be done on before hand through the algorithm itself for better service to the important or high prioritized jobs in spite of equal service provision for all the jobs regardless of their origin.

Following load balancing algorithms are currently prevalent in clouds:-

- ***Round Robin:*** In this algorithm [7], the processes are divided between all processors. Each process is assigned to the processor in a round robin order. The process allocation order is maintained locally independent of the allocations from remote processors. Though the work load distributions between processors are equal but the job processing time for different processes are not same. So at any point of time some nodes may be heavily loaded and others remain idle. Thi s algorithm is mostly used in web servers where http requests are of similar nature and distributed equally.

- ***Connection Mechanism:*** Load balancing algorithm [8] can also be based on least connection mechanism which is a part of dynamic scheduling algorithm. It needs to count the number of connections for each server dynamically to estimate the load. The load balancer records the connection number of each server. The number of connection increases when a new connection is dispatched to it, and decreases the number when connection finishes or timeout happens.

- **Randomized**: Randomized algorithm is of type static in nature. In this algorithm [7] a process can be handled by a particular node n with a probability p. The process allocation order is maintained for each processor independent of allocation from remote processor. This algorithm works well in case of processes are of equal loaded. However, problem arises when loads are of different computational complexities. Randomized algorithm does not maintain determini stic approach. It works well when Round Robin algorithm generates overhead for process queue.

- **Equally Spread Current Execution Algorithm**: Equally spread current execution algorithm [9] process handle with priorities. it distribute the load randomly by checking the size and transfer the load to that virtual machine which is light ly loaded or handle that task easy and take less time , and give maximize throughput. It is spread spectrum technique in which the load balancer spread the load of the job in hand into multiple virtual machines.

- **Throttled Load Balancing Algorithm:** Throttled algorithm [9] is completely based on virtual machine. In this client first requesting the load balancer to check the right virtual machine which access that load easily and perform the operations which is give by the client or user. In this algorithm the client first requests the load balancer to find a suitable Virtual Machine to perform the required operation.

- **A Task Scheduling Algorithm Based on Load Balancing:** Y. Fang et al. [10] discussed a two-level task scheduling mechanism based on load balancing to meet dynamic requirements of users and obtain high resource utilization. It achieves load balancing by first mapping tasks to virtual machines and then virtual machines to host resources thereby improving the task response time, resource utilization and overall performance of the cloud computing environment.

- **Min-Min Algorithm**: It begins with a set of all unassigned tasks. First of all, minimum completion time for all tasks is found. Then among these minimum times the minimum value is selected which is the minimum time among all the tasks on any resources. Then according to that minimum time, the task is scheduled on the corresponding machine. Then the execution time for all other tasks is updated on that machine by adding the execution time of the assigned task to the execution times of other tasks on that machine and assigned task is removed from the list of the tasks that are to be assigned to the machines. Then again the same procedure is followed until all the tasks are assigned on the resources. But this approach has a major drawback that it can lead to starvation [12].

- **Max-Min Algorithm:** Max-Min is almost same as the min-min algorithm except the following: after finding out minimum execution times, the maximum value is selected which is the maximum time among all the tasks on any resources. Then according to that maximum time, the task is scheduled on the corresponding machine. Then the execution time for all other tasks is updated on that machine by adding the execution time of the assigned task to the execution times of other tasks on that machine and assigned task is removed from the list of the tasks that are to be assigned to the machines[12].

SIMULATION IN CLOUD: CLOUDSIM

In CloudSim, cloud computing infrastructures and application services allowing its users to focus on specific system design issues that they want to investigate [8]. Simulation in a CloudSim means implementation of actual environment towards benefit of research. The users or researcher actually analyse the proposed design or existing algorithms through simulation. Resources and software are shared on the basis of client's demand in cloud environment. Essentially, dynamic utilization of resources is achieved under different conditions with various previous established policies. Sometime it is very much difficult and time consuming to measure performance of the applications in real cloud environment. In this consequence, simulation is very much helpful to allow users or developers with practical feedback in spite of having real environment. In this research work, simulation is carried out with a specific cloud simulator, CloudSim [7]. Figure 1 shows Layered CloudSim architecture.

A brief description of these vital components and the working relationship between them is presented in the following [7].

Data centre: Data centre encompasses a number of hosts in homogeneous or heterogeneous configurations (memory, cores, capacity, and storage). It also creates the bandwidth, memory, and storage devices allocation.

Virtual Machine (VM): VM characteristics comprise of memory, processor, storage, and VM scheduling policy. Multiple VM can run on single hosts simultaneously and maintain processor sharing policies.

Host: This experiment considers VM need to handle a number of cores to be processed and host should have resource allocation policy to distribute them in these VMs. So host can arrange sufficient memory and bandwidth to the process elements to execute them inside VM. Host is also responsible for creation and destruction of VMs.

Cloudlet: Cloudlet is an application component which is responsible to deliver the data in the cloud service model. So the length, and output file sizes parameter of Cloudlet should be greater than or equal to 1. It also contains various ids for data transfer and application hosting policy.

RESEARCH MOTIVATION

Load balancing is one of the central issues in cloud computing. It is a mechanism that distributes the dynamic local workload evenly across all the nodes in the whole cloud to avoid a situation where some nodes are heavily loaded while others are idle or doing little work. It helps to attain a high customer satisfaction and resource utilization ratio, consequently improving the overall performance and resource utility of the system. It also makes sure that every computing resource is distributed efficiently and fairly. It further prevents bottlenecks of the system which may occur due to load imbalance. When all existing resources (VMs) are allocated to low priority jobs and a high priority job comes in, the low priority job (deadline is high) has to be preempted its resources allowing a high priority job (deadline is low) to run in its resource. When a job arrives, availability of the VM is checked. If the VM is available then job is allowed to run on the VM. If the VM is not available then the algorithm find a low priority job taking into account the job's lease type. The low priority job is paused its execution by preempting its resource. The high priority job is allowed to run on the resources preempted from the low priority. When any other job running on VMs are completed, the job which was paused early can be resumed if the lease type of the job is suspendable. If not the suspended job has to wait for the completion of high priority job running in its resources, so that it can be resumed. The lease types associated with the jobs are

Cancellable: These requests can be scheduled at any time after their arrival time. It need not be resumed later. Cancellable leases do not guarantee the deadline.

Suspendable: Leases of this type can be suspended at any time but should be resumed later. This type of lease guarantees the execution but not in a specific deadline. Suspendable leases are flexible in start time and can be scheduled at any time after their ready time. In the case of preemption, these leases should be rescheduled to find another free time-slot for the remainder of their execution.

Non-Preemptable: The leases associated with such requests cannot be preempted at all.

- In the existing work, there has been no criteria explained for categorizing the jobs into 3 lease types.
- The existing work is applicable only in the homogeneous environment where all the Vm's are of same capacity.
- No cost has been computed for the jobs of different lease types.

CONCLUSION

In present days cloud computing is one of the greatest platform which provides storage of data in very lower cost and available for all time over the internet. But it has more critical issue like security, load management and fault tolerance. In this paper we are discussing load balancing approaches. Resource scheduling management design on Cloud computing is an important problem. Scheduling model, cost, quality of service, time, and conditions of the request for access to services are factors to be focused. A good task scheduler should adapt its scheduling strategy to the changing

environment and load balancing Cloud task scheduling policy. Cloud Computing is high utility software having the ability to change the IT software industry and making the software even more attractive.

REFERENCES

[1] G. Patela, R. Mehtab and U. Bhoic, "Enhanced Load Balanced Min Min algorithm for Static Meta Task Scheduling in Cloud Computing," Elsevier, p. 545–553, 2015.

[2] Chun-Chieh Chen and Ming-Syan Chen, "HiClus: Highly Scalable Density-based Clustering with Heterogeneous Cloud," Elsevier , p. 149–157, 2015.

[3] Atul Vikas Lakra and Dharmendra Kumar Yadav, "Multi-Objective Tasks Scheduling Algorithm for Cloud Computing Throughput Optimization," Elsevier, p. 107 – 113, 2015.

[4] Geethu Gopinath P P and Shriram K Vasudevan, "An in-depth analysis and study of Load balancing techniques in the cloud computing environment.," Elsevier, p. 427 – 432, 2015.

[5] Ruitao Xie, Yonggang Wen, Xiaohua Jia and Haiyong Xie, "Supporting Seamless Virtual Machine Migration via Named Data Networking in Cloud Data Center," IEEE, pp. 1-14, 2013.

[6] Byung Chul Tak, Youngjin Kwon and Bhuvan Urgaonkar, "Resource Accounting of Shared IT Resources in Multi-Tenant Clouds," IEEE, pp. 1-14, 2015.

[7] Danuta Sorina Chisca, Ignacio Casti, Deepak Mehta and Barry O'Sullivan, "On Energy- and Cooling-Aware Data Centre Workload Management," IEEE, pp. 1111-1114, 2015.

[8] V. Tyagia and T. Kumar, "ORT Broker Policy: Reduce Cost and Response Time Using Throttled Load Balancing Algorithm," Elsevier, p. 217 – 221, 2015.

[9] Song Wu, Yaqiong Peng and Hai Jin, "Time Donating Barrier for efficient task scheduling in competitive multicore systems," Elsevier, pp. 469-477, 2016.

[10] Jia Zhao, Kun Yang, Xiaohui Wei, Yan Ding, Liang Hu and Gaochao Xu, "A Heuristic Clustering-based Task Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment," IEEE, pp. 1-14, 2015.

[11] Xiang Deng, , Di Wu, Junfeng Shen and Jian He, "Eco-Aware Online Power Management and Load Scheduling for Green Cloud Datacenters," IEEE, pp. 1-10, 2014.

[12] Zhi Zhou, Fangming Liu, Ruolan Zou, Jiangchuan Liu, Hong Xu, and Hai Jin, "Carbon-aware Online Control of Geo-distributed Cloud Services," IEEE, pp. 1-14, 2015.

[13] Gongzhuang Peng, Hongwei Wang, Jietao Dong and Heming Zhang, "Knowledge-Based Resource Allocation for Collaborative Simulation Development in a Multi-tenant Cloud Computing Environment," IEEE, pp. 1-13, 2015.

[14] Weiwei Kong, Yang Lei and Jing Ma, "Virtual machine resource scheduling algorithm for cloud computing based on auction mechanism," Elsevier , pp. 1-6, 2016.

[15] Amir Nahir, Ariel Orda and anny Raz, "Replication-based Load Balancing," IEEE, pp. 1-14, 2015.

[16] Jin Yang, Jianmin Pang, Ning Qi and Tao Qi , "On-Demand Self-Adaptivity of Service Availability for Cloud Multi-Tier Applications," IEEE, pp. 1237-1240, 2015.