Load Balancing in Cloud Computing Environment: A Comparative Study of Service Models and Scheduling Algorithms

Navpreet Singh
M.tech Scholar, CSE & IT Deptt., BBSB Engineering College, Fatehgarh Sahib,
Punjab, India
(IKG – Punjab Technical University, Jalandhar) navpreetsaini26@gmail.com
Dr. Kanwalvir Singh Dhindsa
Professor, CSE & IT Deptt.,
BBSB Engineering College, Fatehgarh Sahib,
Punjab, India
(IKG – Punjab Technical University, Jalandhar) kanwalvir.singh@bbsbec.ac.in

ABSTRACT

Load balancing is a computer networking method to distribute workload across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resources, to achieve optimal resource utilization, maximize throughput, minimize response time, and avoid overload. Using multiple components with load balancing, instead of a single component, may increase reliability through redundancy. The load balancing service is usually provided by dedicated software or hardware, such as a multilayer switch or a Domain Name System server. In this paper, the existing static algorithms used for simple cloud load balancing have been identified and also a hybrid algorithm for developments in the future is suggested.

Keywords: **Round-Robin scheduling, Data Center, Priority based scheduling, Cloud computing, Load balancing.**

Date of Submission: May 02, 2017 Date of Acceptance: May 16, 2017

1. INTRODUCTION

Cloud computing is a technology that hosts computing services in centralized datacenters and provides access to them through the Internet. The cloud is a pool of heterogeneous resources [1]. Cloud computing is very much a utility, like electricity: sold on demand, instantly scalable to any volume, and charged by use, with the service provider managing every aspect of the service except the device used to access it [2]. Cloud load balancing refers to distributing client requests across multiple application servers that are running in a cloud environment [3]. Fig.1 represents a basic cloud balancing scenario.

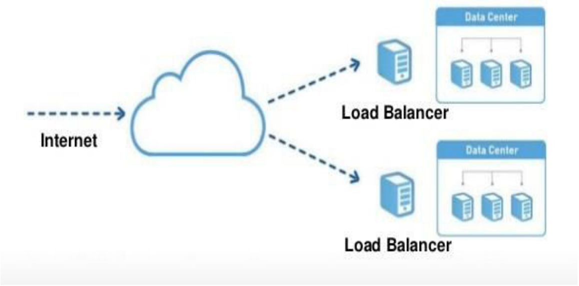


Fig 1: Cloud balancing scenario

There are different types of clouds that can be subscribed to depending on one’s needs. As a home user or small business owner, one will most likely use public cloud services.

1. Public Cloud - A public cloud can be accessed by any subscriber with an internet connection and access to the cloud space. E.g.- Google’s Gmail, dropbox.

2. Private Cloud - A private cloud is established for a specific group or organization and limits access to just that group. E.g.- Many healthcare, financial, trade and banking institutions utilize private cloud computing to maintain cloud confidentiality in highly sensitive electronic records.

3. Community Cloud - A community cloud is shared among two or more organizations that have similar cloud requirements. E.g.- several organizations may require a specific application that resides on one set of cloud servers. Instead of giving each organization their own server in the cloud for this app, the hosting company allows multiple customers connect into their environment and logically segment their sessions.

4. Hybrid Cloud- A hybrid cloud is essentially a combination of at least two clouds, where the clouds included are mixture of public, private, or community.

2. REVIEW OF LITERATURE

In this section, main focus of the discussion is on the research work related to the load balancing in cloud computing. It helps to compare the load balancing techniques available and conclude an optimized solution.

Thapar et al. [1] proposed the concept of proportion weights in order to assign the workload to data centres. On the other hand, other service broker policies based broker decisions on the basis of location. The proposed policy took into consideration the efficiency of underlying hardware, which meant greater number of hardware machines, means more virtual machines, hence large number of cloudlets could be served.

Fan [8] proposed that the development of cloud computing has received a considerable attention. For cloud service providers, packing VMs onto a small number of servers is an effective way to reduce energy costs, so as to improve the efficiency of the data centres. However allocating too many VMs on a physical machine may cause some hot spots which violate the service level agreement (SLA) of applications. Load balancing of the entire system is hence needed to guarantee the SLA.

Tziritas [9] discussed the problem of virtual machine (VM) placement onto physical servers to jointly optimize two objective functions. The first objective is to minimize the total energy spent within a cloud due to the servers that are commissioned to satisfy the computational demands of VMs. The second objective is to minimize the total network overhead incurred due to: (a) communicational dependencies between VMs, and (b) the VM migrations performed for the transition from an old assignment scheme to a new one.

Guo [10] suggested that the instances which are used to process big data need higher CPUs and disks than other kinds of instances. If the instances of disk resource consuming are placed in the same physical node, clearly, the disk I/O bandwidth would be used up quickly that would affect the performance of the entire node seriously. Guo hence proposed an instance placement algorithm FFDL, which based on disk I/O for private cloud environment dealt with big data that would adopt the disk I/O load balancing strategy and reduce competition for the disk I/O load between instances.

Garcia [11] discussed that load management in cloud data centres must take into account: (a) hardware diversity of hosts (b) heterogeneous user requirements (c) volatile resource usage profiles of virtual machines (VMs) (d) fluctuating load patterns, and (e) energy consumption. Garcia hence proposed distributed problem solving techniques for load management in data centres supported by VM live migration. Collaborative agents were endowed with a load balancing protocol and an energy- aware consolidation protocol to balance and consolidate heterogeneous loads in a distributed manner while reducing energy consumption costs.

Hsieh [12] suggested that as a cloud data centre may be located over many regions and the network environment within a cloud data center may differ from traditional ones, how Virtual Machines (VMs) are deployed will influence service performance. Author, based on the Eucalyptus cloud computing and Software-Defined Networking platform, proposed a load balancing scheduling mechanism that works on the current network status between users and associated VMs to improve the cloud services. Author also set up a node controller on the same subnet and different subnet.

Dinita [13] described an optimized and novel approach to an Autonomous Virtual Server Management System in a `Cloud Computing' environment and it presented a set of preliminary test results. One key advantage of this system is its ability to improve hardware power consumption through autonomously moving virtual servers around a network to balance out hardware loads. This has a potentially important impact on issues of sustainability with respect to both energy efficiency and economic viability.

3. ROLE OF SERVICE BROKER IN CLOUD COMPUTING

A cloud broker is an intermediary between the provider and the purchaser of the cloud computing service. It is generally a third party individual or sometimes a business entity.

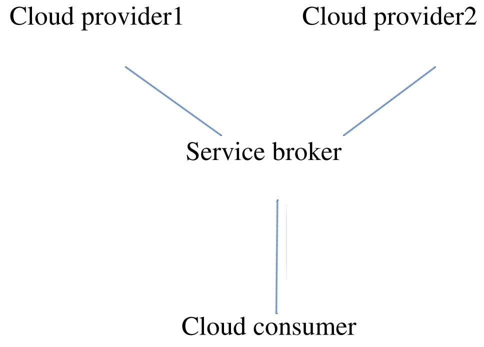


Fig.2: Role of service broker

Fig.2 above represents the role of a service broker which facilitates the distribution of work between different cloud service providers.

The entire process of serving a client is a part of any one of the services defined in the service model. It begins with a request for a particular resource or application, be it for development, or just accessing the storage of the service provider. The request is serviced by the cloud service provider through a series of steps, the first one passing through a cloud service broker, which acts as the intermediary between a cloud consumer and the cloud service providers. The service broker makes use of any one of the available service broker policies in order to send the request to the most appropriate data center. The role of a service broker is shown in fig.2 above.

After choosing the data center that is going to perform computation, the load balancer at the data center comes into action. It makes use of the implemented load balancing algorithms to select the appropriate virtual machine to which the request has to be sent for execution. The innermost abstraction layer comprises virtual machine management. The virtual machine manager is responsible for the management and migration of virtual machines in the cloud data centers. Out of the above tasks, the use of an efficient service broker policy is quite necessary to ensure that the later tasks are carried out with efficiency and least response time.

4. CLOUD SERVICE MODELS

Cloud computing is a convenient on-demand network access model enabling the access to a shared pool of configurable computing resources. Cloud model is composed of three service models.

Cloud computing is able to provide a variety of services at the moment but main three services are Infrastructure As-A-Service, Platform-As-A-Service and Software-As-A Service, also called as service model of Cloud computing described in fig.4.

a) Software-As-A-Service (SaaS)

It describes any cloud service where consumers are able to access software applications over the internet. The applications are hosted in "the cloud" and can be used for a wide range of tasks for both individuals and organizations. There are a number of reasons why SaaS is beneficial to organizations and personal users alike. e.g.- Google Apps, Salesforce, Twitter, Facebook, etc.

- No additional hardware costs: the processing power is supplied by the cloud provider.
- No initial setup costs: applications are ready to use once the user subscribes.
- Pay for what you use: if a piece of software is only needed for a limited period then it is only paid for over that period and subscriptions can usually be halted at any time.
- Usage is scalable: if a user decides they need more storage or additional services, it can be subscribed to at any time.
- Updates are automated: update is available online
  to existing customers, often free of charge.
- Cross device compatibility: SaaS applications can be accessed via any internet enabled device,
- Accessible from any location with an internet enabled device.
- Applications can be customized: with some software, customization is available.

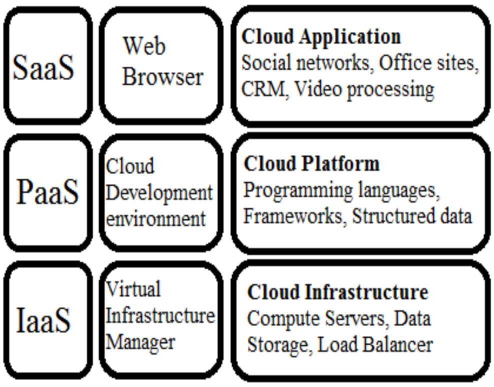

SaaS
Web Browser
Cloud Application
Social networks, Office sites, CRM, Video processing
PaaS
Cloud Development environment
Cloud Platform
Programming languages, Frameworks, Structured data
IaaS
Virtual Infrastructure Manager
Cloud Infrastructure
Compute Servers, Data Storage, Load Balancer

Fig.3: Cloud Computing Service model applications

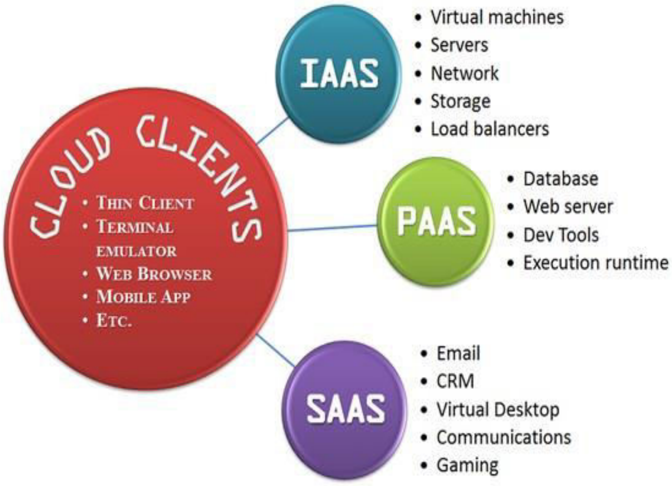


Fig.4: Basic cloud service models description

b) Platform-As-A-Service (PaaS)

It provides a development platform to its users so that the user can develop and maintain respective applications and cloud specific utilities. It is different from SaaS because SaaS is a developed and deployed application whereas PaaS provides a platform or ground to develop those applications. PaaS provides development environment and platform, so all supporting material i.e. programming environment, development tools and infrastructure etc. must be provided by cloud provider. e.g.-Google App Engine, WordPress, etc.

- The users don't have to invest in physical infrastructure. This leaves them free to focus on the development of applications.
- Makes development possible for 'non-experts':
  With some PaaS offerings anyone can develop an application through their web browser utilizing one-click functionality.
- Flexibility: Customers can 'pick and choose' the features they feel are necessary.
- Adaptability: Features can be changed if circumstances dictate that they should.
- Teams in various locations can work together: As an internet connection and web browser are

all that is required, developers spread across several locations can work together on the same application build.

- Security is provided, including data security, backup and recovery.

c) Infrastructure-As-A-Service (IaaS)

As with all cloud computing services, it provides access to computing resource in a virtualized environment "the Cloud", across a public connection, usually the internet. In case of IaaS, the computing resource provided is specifically that of virtualized hardware, in other words, computing infrastructure. The definition includes such offerings as virtual server space, network connections, bandwidth, IP addresses and load balancers. Physically, the pool of hardware resource is pulled from a multitude of servers and networks usually distributed across numerous data centers, all of which the cloud provider is responsible for maintaining. The client, on the other hand, is given access to the virtualized components in order to build their own IT platforms.

e.g.- Salesforce.

- Scalability: Resource is available as and when the client needs it and, therefore, there are no delays in expanding capacity or the wastage of unused capacity.
- No investment in hardware: The underlying physical hardware is set up and maintained by the cloud provider.
- Utility style costing: The client only pays for the resource that they actually use.
- Location independence: The service can usually be accessed from any location as long as there is an internet connection.
- No single point of failure: If one server or network switch, for example, were to fail, the broader service would be unaffected due to the remaining multitude of hardware resources and redundancy configurations.

5. TYPES OF SCHEDULING ALGORITHMS

The scheduling algorithms are aimed on improving the performance and the quality of service by reducing the execution time and costs. The various scheduling algorithms are as follows:

5.1 Round-robin load balancing is one of the simplest methods for distributing client requests across a group of servers. Going down the list of servers in the group, the round-robin load balancer forwards a client request to each server in turn. When it reaches the end of the list, the load balancer loops back and goes down the list again (sends the next request to the first listed server, the one after that to the second server, and so on).

Figure 5 below represents assigning of various jobs to servers for their execution in Round Robin fashion.

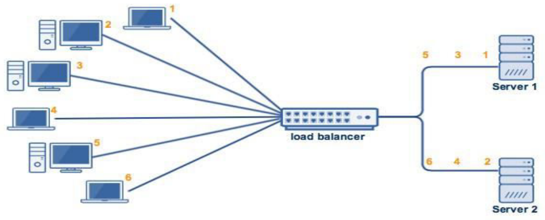


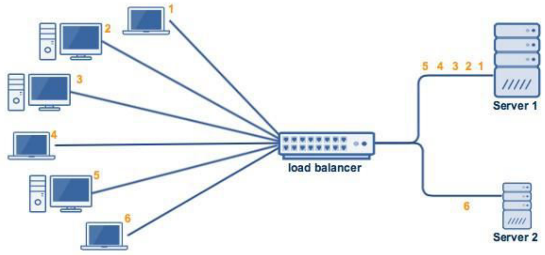
Fig.5: Round-Robin balancing technique

It does not always result in the most accurate or efficient distribution of traffic, because many round-robin load balancers assume that all servers are the same: currently up, currently handling the same load, and with the same storage and computing capacity. The following variants to the round-robin algorithm take additional factors into account and can result in better load balancing.

5.2 Weighted round robin - A weight is assigned to each server based on criteria chosen by the site administrator; the most commonly used criterion is the server's traffic-handling capacity. The higher the weight, the larger the proportion of client requests the server receives. If, for example, server 1 is assigned a weight of 3 and server 2 a weight of 1, the load balancer forwards 3 requests to server 1 and for each 1 it sends to server 2.

As shown in figure 6 below, server 1 is assigned a weight of 5 and server 2 is assigned a weight of 6. So a request 6 having weight 5 is assigned to server 2 while all others are assigned to server 1.

Fig.6: Weighted Round-Robin load balancing technique



5.3 Dynamic round robin - A weight is assigned to each server dynamically, based on real-time data about the server's current load and idle capacity.

5.4 Priority based scheduling – A priority is assigned to each request and then the request is processed depending on its priority. The requests of Equal priority are scheduled in FCFS order [2]. Priority of a request can be either defined externally or internally. Priorities defined internally for a request are computed using some measurable quantities or qualities. To each admitted queue, a priority is assigned.

To increase the performance of the overall system, all the resources shall be evenly distributed to satisfy the customer requirement by distributing the load dynamically among the nodes. Figure 7 below depicts a basic scenario in which a job request is serviced by a cloud platform to achieve the maximum performance.

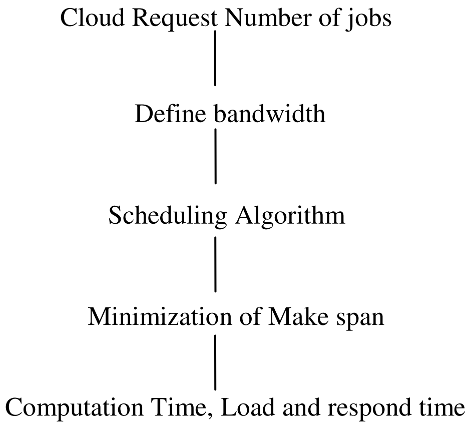

Cloud Request Number of jobs
Define bandwidth
Scheduling Algorithm
Minimization of Make span
Computation Time, Load and respond time

Fig.7: A basic cloud load balancing scenario using any approach.

6. COMPARISON OF SERVICE MODELS AND SCHEDULING ALGORITHMS

There are different types of service models in cloud computing atmosphere as follows:

Table 1: Service models in cloud computing

S. No.	FEATURES	SaaS	PaaS	IaaS
1	**Feature**	Software delivered over web.	Platform delivered over web, for creation of software.	Infrastructure (software or hardware) delivered on web as an on demand service.
2	**Offerings**	User has nothing to worry about. A pre configured package as per requirement is given and payed accordingly.	User gets what is demanded. Hardware, Software, Web environment, OS. Payment is made accordingly and user gets the platform to use.	User gets the infrastructure and pays accordingly. Can install any OS, composition or software.
3	**Level**	Complete pack of all services.	Top of IaaS	Basic layer of computing.
4	**Feasibility**	Used by a variety of users. Used over web on various locations (home, road, office).	All technical stack requirements met by the platform offerings.	For people or companies not willing to invest too much on hardware. For those trying to do something temporarily.
5	**Technical skill requirement**	No need of any technical knowledge.	Knowledge of the subject is required. Only the basic setup is provided.	Technical knowledge is required.
6	**Deals with**	Only applications (like Gmail,Yahoo, etc). Social Networking sites (like Facebook)	Runtimes, Database and web servers.	Virtual machine storage, load balancers, network, servers.
7	**Consumption graph**	Most widely used among a common man or companies which that don't have to worry about technicalities.	Popular among developers as they don't need to worry about traffic load or server management.	High popularity among skilled developers or researchers who have need of custom configuration.
8	**Disadvantage**	1. Security concern. 2. Certain organizations have regulation related to where data is stored.	1. Limited flexibility. 2. Integration problem with the in-house systems and the application as it could trigger an increase in complexity.	Dependence on a specific provider. Also to mitigate any security relates risk, it is important to consider what data is to be sent to the cloud.

From the Table 1 above, it can be concluded that a cloud can be scaled dynamically as per the needs of the users, and also, there is no need for any company to deploy its IT staff to manage this service, since the cloud service provider is responsible for providing the software and hardware necessary for the service.

The comparative study of the scheduling algorithms (as shown in Table 2 below) has its advantages as it gives its user a better picture of the appropriate class of scheduling algorithms available for different types of required services as per the requirements of consumers and service providers.

Table 2: Comparison of various Scheduling algorithms

S. No.	**SCHEDULING ALGO**	**Round-Robin**	**Weighted Round Robin**	**Priority based**
1	**Concept**	Designed specifically for time sharing systems.	Designed to handle servers better depending upon their processing capabilities.	Designed to schedule the serving of requests based on their priority.
2	**Implementation**	Similar to FCFS but each request is served for a fixed interval of time. All requests are kept in a circular queue known as ready queue.	Each server is assigned a weight, integer value that describes the processing capacity. Higher the weight, higher the number of connections received by the server.	It involves assignment of priority to every request. Requests with high priority are served first, while ones with same priority are served in FCFS order. In case of low priority job, remedy to starvation is aging, in which priority gradually increases for jobs that are in queue for long period of time.
3	**Advantages**	Main advantage - it is Starvation free.	Weights assigned to servers create a longer time slice, hence making it starvation free.	Important jobs are served first.

7. CONCLUSION

Scheduling is a major issue in the management of service requests in cloud environment. Various phases have to be used for the development of the load balancing system in the cloud computing environment. These different phases have to be implemented for the completion of the proposed work. Load balancing has been done by dividing different tasks into a number of jobs so that they can be allocated to different resources for processing to complete in less computation time. In cloud computing scenario, number of tasks has to be assigned on various processes to handle load on the cloud. These tasks have been divided into sets and the dependency checking is done for prevention of dead lock state or to prevent demand of various extra resources for allocation. Hybrid algorithm is better than others because unlike PB scheduling, it automatically increases the priority for the old processes having low initial priority, hence executing them eventually.

REFERENCES

[1] K. Kishor, V. Thapar, "An efficient service broker policy for Cloud computing environment", International Journal of Computer Science Trends and Technology (IJCST), Vol. 2, Issue 4, July-Aug 2014.

[2] Pinal Salot, "A Survey of Various scheduling algorithms in cloud computing environment", ISSN: 2319 - 1163, Vol.2, Issue 2, pp. 131-135, June 2014.

[3] Z. Xiao, W. Song, and Q. Chen, "Dynamic resource allocation using virtual machines for cloud computing environment," IEEE Transactions on Parallel and Distributed Systems, Vol. 24, No. 6, pp. 1107–1117, 2013.

[4] L. D. Babu and P. V. Krishna, "Honey bee behavior inspired load balancing of tasks in cloud computing environments," Applied Soft Computing Journal, Vol. 13, No. 5, pp. 2292–2303, 2013.

[5] Y. Zhang, "Dynamic load-balanced multicast based on the Eucalyptus open-source cloud-computing system", pp. 456 – 460, IEEE, 2011.

[6] R. Basker, V. R. Uthariaraj, and D. C. Devi, "An enhanced scheduling in weighted round robin for the cloud infrastructure services," International Journal of Recent Advance in Engineering & Technology, Vol. 2, No. 3, pp. 81–86, 2014.

[7] Y. Wen, "Load balancing job assignment for cluster-based cloud computing", pp. 199 – 204, IEEE, 2014.

[8] Z. Fan, “Simulated-Annealing Load Balancing for Resource Allocation in Cloud Environments”, IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 1-6, Taipei, 2013.

[9] N. Tziritas, “Application-Aware Workload Consolidation to Minimize Both Energy Consumption and Network Load in Cloud Environments”, IEEE International Conference on Parallel Processing, pp.-449-457, Washington D.C., USA, October 2013.

[10] J. Guo, “An instances placement algorithm based on disk I/O load for big data in private cloud”, IEEE International Conference on Wavelet Active Media Technology and Information Processing, pp. 287-290, 2012.

[11] J. O. Garcia, “Collaborative Agents for Distributed Load Management in Cloud Data Centres Using Live Migration of Virtual Machines”, IEEE International Conference on Services Computing, pp. 916-929, 2015.

[12] W. K. Hseih, “Load balancing virtual machines deployment mechanism in SDN open cloud platform”, IEEE International Conference on International Conference on Advanced Communication Technology, pp. 329-335, 2015.

[13] R. I. Dinita, “Hardware loads and power consumption in cloud computing environments”, IEEE International Conference on International Conference on industrial Technology, pp. 1291-1296, 2013.

[14] A. Goyal Bharti, “A Study of Load Balancing in Cloud Computing using Soft Computing Techniques”, International Journal of Computer Applications (0975 – 8887) Vol. 92, No.9, April 2014.

[15] N. Kaur, T.S. Aulakh, R.S. Cheema, “Comparison of Workflow Scheduling Algorithms in Cloud Computing”, International Journal of Advanced Computer Science and Applications, Vol. 2, No. 10, 2011.

[16] M.S. Rana, S. Kumar, N. Jaisankar, “Comparison of Probabilistic Optimization Algorithms for Resource Scheduling in Cloud Computing Environment” International Journal of Engineering and Technology, pp. 153-163, Vol. 3, No.6, July 2016.

[17] C. Kalpana, U. Karthick Kumar, R. Gogulan, “Max - Min Particle Swarm Optimization Algorithm with Load Balancing for Distributed Task Scheduling on the Grid Environment”, IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 3, No. 1, May 2012.