# Resource Management by Refining Allocation Policies Over A Cloud

**Meenu Dhingra[1], Dr. V.K. Gupta[2] , Sunil Maggu[3]**

**Professor, Rajasthan University, Jaipur[2]**

**Research Scholar, Jagannath University, Jaipur[1,3]**

## Abstract

Cloud computing applications are spreading at large scale now a days and thus the sharing of resources is major issue as increase in number of users also increase the demand for resources. Over a cloud multiple resources instead of single one are allocated simultaneously so as to avoid the congestion arises due to multiple requests for one particular resource. Thus to solve the conflict between so many requests a control mechanism is proposed so that an efficient resource allocation policy can be defined resulting in the less disruption of service of cloud and raising of reliability of cloud services. The proposed technique emphasis on normalizing the size of resource required up to some threshold limit and thus allocating the reduced optimal resource size for request here by satisfying multiple requests simultaneously.

*Keywords:* - Cloud Resource Management, normalization, capacity limit, congestion control, working ability.

## Introduction

Cloud Computing is broadly used by various IT companies for providing their customers a collaborative computing environment comprising of various end user services. Cloud is based on the abstraction of technology, resources and their locations. All these play a major role in the integration of computing environment (including networks, systems and applications). Cloud computing hierarchy (virtualization, infrastructure, platform and application) depends mainly on sharing of resources to achieve coherence over a network (typically the Internet).

The umbrella of cloud computing is a big one. As in the evolution of any technology there are various competing models evolved over time at different intervals and each model has its own significance and configuration as each is designed to uniquely satisfy the specific needs if environment. Indeed, the number of cloud permutations is nearly as diverse as the number of companies using them. Still, over time, there are consistent models that begin to emerge. Here's a look at some of the top cloud computing models in production today:

*The Internal Cloud***.** This is the most common type of cloud computing environment. Within the single organization if the virtualization is made for in house services then the internal cloud model is considered. The basis for this is to design a common internal infrastructure including server, storage, network and applications connected altogether which in turn perform the tasks or provide services to achieve maximum efficiency. This is different from a simply virtualized situation in that it allows a higher degree of automation and even a chargeback capability for the other business units.

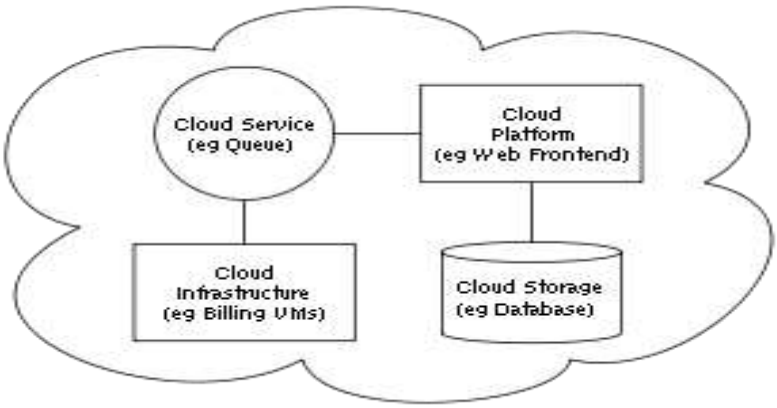


**Fig.1. Simple Cloud Architecture**

*External Cloud Hosting***.** This type of cloud model access the services required for organization from external cloud service provider and use these services through internet. It provides probably the most cost-effective way to utilize the cloud. The major issues of concern with this model are security and integrity of data along with the performance some times.

*The Hybrid Cloud.* The Hybrid cloud model is the combination of internal cloud model and external cloud model. It provides a cloud computing environment where an organization manages some services in house where as left out are accessed through external cloud service provider.

*Traditional SaaS***.** SaaS is still out there, and it's especially common among SMBs. A small business that uses 37 Signals for project management or Google for its company email is adopting the cloud on the most micro of levels.

## Need of resource allocation

Resource allocation is used to assign the available resources in an economic way. It is part of resource management. In Cloud Computing, resource allocation is the scheduling of services and the resources

required by those activities while taking into consideration both the resource availability and the processing time.

Allocation policy should be such that it should be easily available when it is demanded. As we know the resources are scattered at various location in the cloud so when a client's consumer who want to access the services what a client is providing will pass a request to use them. The request will be reached to cloud service provider via client. In this way the networking of the process occurs.

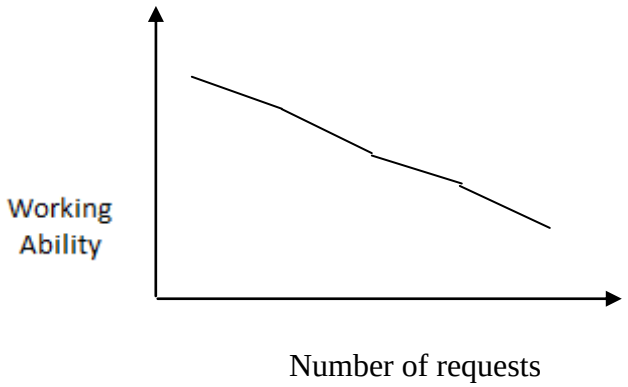


**Fig. 2. Trend between no. of requests and working ability**

As shown in Fig 2 trend between working ability and number of requests seems to be inversely proportional. As the number of requests is increasing working ability of the cloud system is decreasing. In the figure it is on the decreasing side with the increase in number of request. So the consequence that arises here is decrement in working ability.

This is the actual problem area which is to be handled very carefully. So that working ability and the capacity limit can be maintained

Various points which show the need for a resource allocation policy:-

- Due to a well planned allocation policy accessing will be fast and efficiency will be increased.
- With the use of planned allocation policy the problem of congestion while allocation of resources can be avoided.
- A planned allocation policy will help the cloud to handle problem of attacks from both sides i.e. insides attack and outside attack.

**Importance and Relevance of Resource Allocation**

As we know that cloud is providing services for the clients by running so many application for client favor and giving client's consumer a platform to access those facilities by giving him a virtual image. This will produce like; consumer is using the facilities on client end and not on some other place.

As per the functionality of the cloud services now days, we know that a cloud is capable of handling a million of user at one time and numbers of services and application that are running on the cloud are to be managed so that problems that we discussed above can be minimized and can be handled properly.

Our proposed technique here will lead the CSP to add a new edge to the allocation policy among a million of users at one time of the number of resources. A refined policy always a less irritating fact for clients. By the help of good allocation policy clients will be able to access the facilities that they are authorized easily and efficiently. Waiting time for a particular resource can be reduced by our proposal. The main emphasis is on the allocation policy here so that average waiting time for a resource of a client can be reduced to minimum.

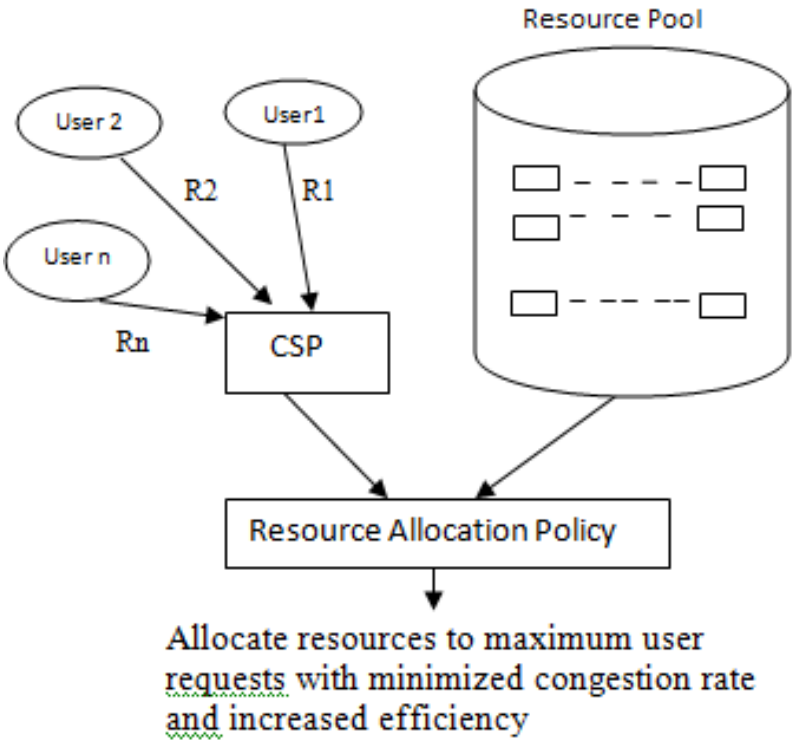


**Fig 3. : Resource Allocation Strategy over cloud**

As shown in figure 3 the resource pool contains all the resources available for the users associated with cloud. These resources are allocated to users on their demand basis. R1, R2….Rn are the requests

generated by the n number of users over cloud and by application of efficient resource allocation policy CSP allocate resources in such a way that maximum number of user requests can be handled efficiently. But problem arises when same resource is demanded by more than a limit user. So here allocation policy selection becomes critical. If resource allocation is done without any strategy then problem of congestion can be raised.

In our approach we can limit the user by assigning him a limit of using resources. In this way the topic that we selected here is relevant for now a days and is of maximum meaning. Here we are allocating every user a bandwidth within which he can perform his task even all the resources that he want to avail. When the limit will be reached then user will automatic exited from the services that he was using. So by doing this congestion will be solved up to a limit. And efficiency of the cloud resource allocation and working will be raised.

**Proposed Working**

As cloud computing services rapidly expand their customer base, it has become important to provide them economically. To do so, it is essential to optimize resource allocation under the assumption that the required amount of resource can be taken from a common resource pool and rented out to the user on an hourly basis. The amount of resource required and the period in which it is used are not fixed. They can vary greatly from user to user and from service to service.

Problem arises when there is surge of requests for a particular service and competition occurs between these requests for the use of the resources. It originates unavoidable conditions resulting in the restriction on use of resources and hence brings down the efficiency of the cloud.

*Types Of Services Available on Cloud*

The cloud services can be classified into two categories.

*Immediate processed services:-* These are the services that allocates a spare resource immediately to user upon arrival of request and reject if there is no spare capacity.

*Waiting System:* - These are the system that allocates a spare capacity to user in the sequence in which their request has arrived, instead of allocating resources immediately upon arrival of request.

In our technique we have worked upon the immediate processing system with static resource allocation.

*Proposed Solution of the Above Problem*

Here we assume that the physical facilities for providing the cloud computing services are distributed over multiple centers. So that it is easy to meet the increasing demands and to enhance the reliability. Various centers are defined which provide resources such as **working ability** and **capacity limit to** the clients. The maximum size of working ability and capacity limit at center i (i=1,2,….n) is assumed to be Wmaxi and Cmaxj. When a request for resource is made one feasible center is selected among total no. of centers i.e. n and both the resources that is working ability and capacity limit in that center are allocated to that request for a quantum of time. In case any of the centers is not able to handle the request or to allocate resources then the request is rejected.

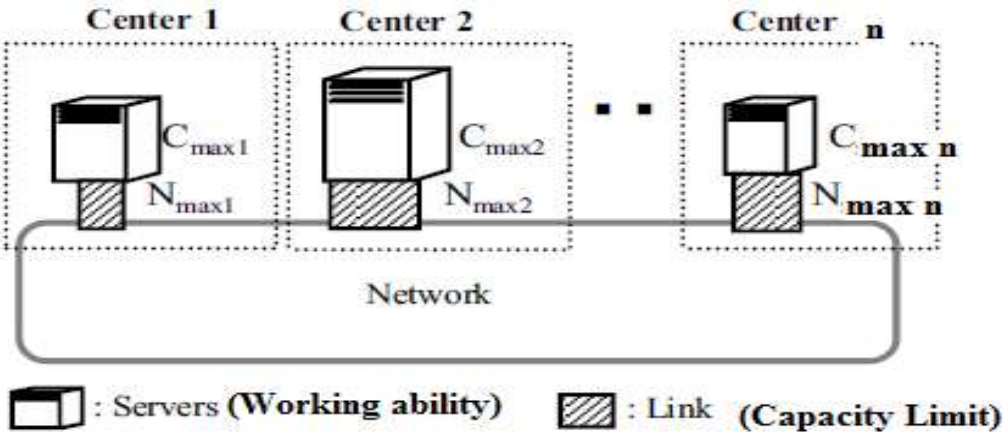


**Fig.4. Role of Working ability and Capacity Limit**

In the proposed technique we assume to reduce the size of resource requested if it is greater than threshold value. So that proper utilization of resources can be made and traffic over a cloud can be reduced. Whenever the request for resource is generated then for optimal resource allocation we cannot emphasis on single resource but we have to consider the status of other resources also related to a service. For example as by analyzing figure 5 we come across two cases. In first case only working ability was chosen as selection parameter for center and we have two centers and if the center selected by the use of best fit approach , then a congestion state is raised as the demanded resource is allocated , but the other resource required for completion of work is not available. As only working ability is considered then the requests 1 and 2 for resources is considered and center1 is allocated to them. Similarly requests 3 and 4 is fulfilled through center2. But as the request 5 arrives we don't have working ability available in center1 but it is in center 2 and we don't have capacity limit in center 2 but in center1. This arises in state of congestion. So to avoid congestion state like this we have to consider

a sequence of resources (2 or more) collectively in order to achieve the job. As in case2 suppose if request 1 is made than it can be assigned to any of two centers 1 and 2 as both are available. Then for request 2 center 2 is best suited if center 1 is allocated to request1. Again for request 3 it is assigned to center 1 based on best fitted resources availability. Request 4 is fulfilled through center2 and request 5 can be completed by any of center1 and center5.

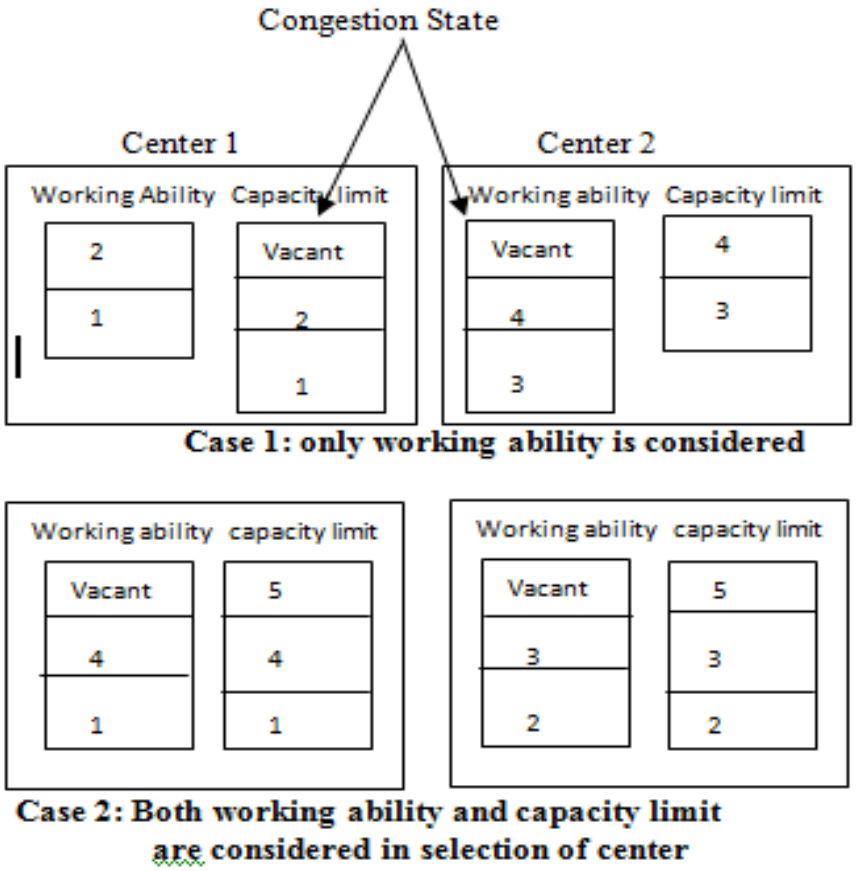


**Fig.5: Congestion while resource allocation**

*Step-Wise Refinement of the Proposal*

**Step1:-** When the client make a request for a resource it specifies a minimizing ratio p (0>p>1) for the resource. Along with request of resource, parameters such as size of resource require and time interval of service is also to be mentioned.

**Step2:-** Ratio of resource allocation time i.e. T is calculated which is based on total time for the processing of request i.e. from generation of request to the completion of service.

**Step3:-** A threshold value is calculated which will act as a decision parameter in the reduction of resource size. The allocated resource size is reduced based on value of p and T. So threshold will also depend upon ratio of resource allocation time and minimizing ratio for resource.

**Step4:-** if the requested resource size is equal to or greater than threshold the allocated resource size is reduced and if less than no action is will take

place as resource can be provided to the user without any reduction.

The proposed technique is supposed to contribute toward proper resource utilization by balancing the load over a cloud and also reduce the traffic made through unprocessed request. Here the use of multiple resources simultaneously is performed as independent resources if considered contribute towards a more congestive state. Also by the use of proposed technique there is less chances of request loss over the cloud.

**References**

1. G.Reese: "Cloud Application Architecture," O'Reilly&Associates, Inc., Apr. 2009.

2. J.W.Rittinghouse and J.F.Ransone: "Cloud Computing:Impremention, Management, and Security," CRC Press LLC, Aug. 2009.

3. Amazon Elastic Compute Cloud (Amazon EC2) http://aws.amazon.com/ec2/

4. Google App Engine http://code.google.com/intl/ja/appengine/

5. S.Kuribayashi, "Optimal Joint Multiple Resource Allocation Method for Cloud Computing Environments," International Journal of Research and Reviews in Computer Science (IJRRCS), Vol. 2, No.1, pp.1-8, Feb. 2011.

6. S.Tsumura and S.Kuribayashi: "Simultaneous allocationof multiple resources for computer communications networks," In Proceeding of 12th Asia-Pacific Conferenceon Communications (APCC2006), 2F-4, Aug. 2006.

7. M.Gusat, R.Birke and C.Minkenberg, "Delay-based cloudcongestion control," In Proceeding of GLOBECOM'2009,Nov. 2009.

8. B.Raghavan, K.Vishwanath, S.Ramabhadran, K.Yocum and A.C.Snoeren, "Cloud control with distributed rate limiting," In Proceeding of SIGCOMM'07, Aug. 2007.

9. P. Li, K. Agrawal, J. Buhler, and R. D. Chamberlain. Deadlock avoidance for streaming computations with filtering. In ACM Symp. on Parallelism in Algorithms and Architectures, 2010.

10. P. Li, K. Agrawal, J. Buhler, R. D. Chamberlain, and J. M. Lancaster. Deadlock-avoidance for streaming applications with split-join structure: Two case studies. In IEEE Int'l Conf. on Application-specific Systems, Architectures and Processors, pages 333–336, July 2010.

11. W. Thies and S. Amarasinghe. An empirical characterization of stream programs and its implications for language and compiler design. In Int'l Conf. on Parallel Architectures and Compilation Techniques, pages 365–376, 2010.

12. MELL P, GRANCE T. The NIST Definition of Cloud Computmg[EB/OL]. [2010-05-10]. http://csrc.nist.gov/groups/SNS/cloud-computing/.

13. E. Knorr and G. Gruman, "What cloud computing really means," InfoWorld, April 2008. Electronic Magazine, Available at http://www.infoworld.com/d/cloud-computing/what-cloudcomputing- really-means-031 [Last accessed October 6, 2010] .

14. A. Khajeh-Hosseini, I. Sommerville, and I. Sriram, "Research challenges for enterprise cloud computing," unpublished, http://arxiv.org/abs/1001.3257, 2010.

15. Fesehaye, D., Malik, R., and Nahrstedt, K. A Scalable Distributed File System for Cloud Computing. Technical report, University of Illinois at Urbana-Champaign (UIUC), 03 2010.

16. Fesehaye, D., Nahrstedt, K., and Caesar, M. A Network Congestion control Protocol (NCP). In *CS/UIUC Technical Report* (Urbana, IL, USA, 2010).