ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol.2, Special Issue 4, September 2014**

# Active Resource Provision in Cloud Computing Through Virtualization

K.Sobhanadri[1], A.Revathi[2]

PG Student, SVCE, JNTUA University, Tirupati, India[1]

Assistant Professor, SVCE, JNTUA University, Tirupati, India[2]

**ABSTRACT**: Cloud Computing is an emerging technology which provides effective services to the clients. It permits clients to scale up and down their resources usage depending upon their needs Because of this, under provision and over provision problems may occur. To overcome this migration of service utilization Our Paper focuses on overcoming this problem by distributing the resource to multiple clients through virtualization technology to enhance their returns. By using virtualization, it allocates datacenter resources dynamically based on applications demands and this technology also supports green technology by optimizing the number of servers in use. We present a new approach called "Skewness", to calculate the unevenness in the Multi-tier resource utilization of a server. By optimizing Skewness, we can join different types of workloads adequately and we can improve the whole consumption of server resources.

**KEYWORDS:** Cloud computing, over provision, under provision, virtualization, green computing, skewness.

## 1. INTRODUCTION

Many of the organizations show interest on cloud, because with low cost we can access resources from cloud in a flexible and secure manner. Cloud shares their resource to multiple users. Cost of resources varies significantly depending on configuration for using them. Hence efficient management of resources is of prime interest to both Cloud Providers and Cloud Users. The success of any cloud management software critically depends on the flexibility, scale and efficiency with which it can utilize the underlying hardware resources while providing necessary performance isolation. Successful resource management solution for cloud environments needs to provide a rich set of resource controls for better isolation. Here dynamic resource allocation and load balancing is the challenging task to provide effective service to clients. Due to peak demands for a resource in the server, resource is over utilized by clients through virtualization. This may degrade the performance of the server. In under utilization usage of resource is very poor when compare to over utilization, for this we are migrate client processing from VM to other VM.

Virtual machine monitors (VMMs) provide a mechanism for mapping virtual Machines (VM) to physical resources in Physical Machine (PM). But mapping is hidden from the cloud. Cloud provider should ensure that physical machine have sufficient resource to meet client need. When an application is running on VM mapping between VMs and PMs is done by migration technology. However policy issue remains in every aspect to decide the mapping adaptively so that the demands of VM were met and the number of PM used is minimized. Though it is a challenging one when the resource need of VM is heterogeneous due to the different set of applications their need might vary with time as the workloads goes ups and down. The capacity of PM can also be Heterogeneous because multiple generations of hardware coexist in a datacenter.

Here we have two main goals to provide dynamic resource allocation

1. **Optimize burdens:** PM should provide all the necessary resources required to process applications on VMs. It satisfies VM needs based on its capacity.
2. **Green Computing:** optimize unnecessary usage of PMs to save the energy

The work discussed below in our Paper makes discussions of how to overcome these two problems in cloud.

First we have to share the work to servers in a balanced way depending upon their capacity. By sharing server we can perform their task effectively to optimize load on it. Next, we have to optimize the usage of resource then only we can give flexible and effective service to clients, for this usage of resource Monitor is necessary. By monitoring, we came to know underutilization and overutilization of resources in PM through VMs. So to calculate the usage of resource we introduce a new approach called "Skewness". With the help of previously used resource logs, we have to forecast periodically for future resource needs. A client can demand for highly resource provision. At the time there may be a

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

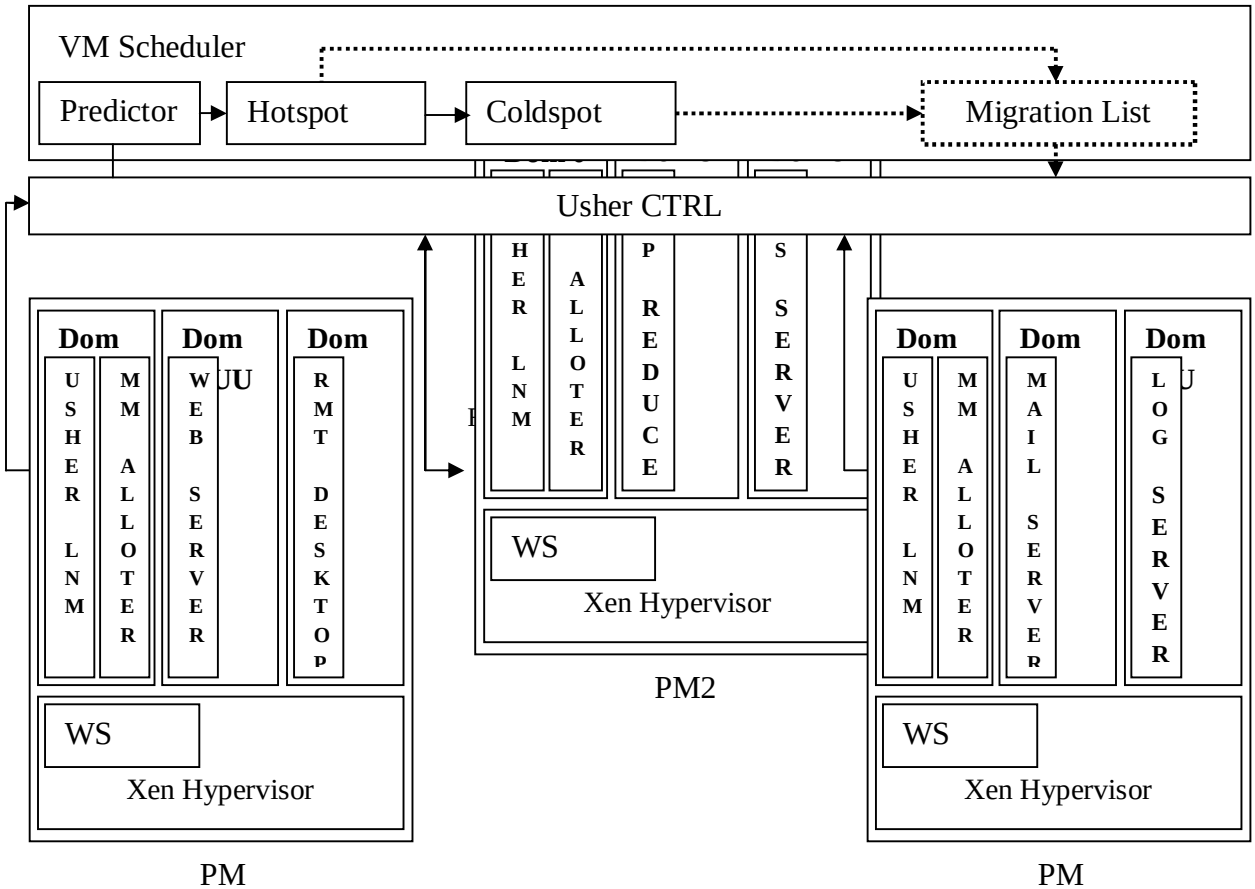
(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 4, September 2014

chance for insufficient resource, while providing that service to the intended client, resource as well as memory forecasting is necessary. For this we design "resource forecasting algorithm".

## II. SYSTEM OVERVIEW

The Architecture represented in Fig. 1, Each PM runs with xen hypervisor Processor with domain 0 and one or more domain U in this system. In each domain U 'n' no. of resource are there to provide service to clients. All PM's are connected to backend databases for storing information's. Sharing of PM's resources to VM's is




maintained by Usher center controller (Usher CTRL). Each processing on usher local node Manager (LNM) in domain 0 which gathers the resource usage information like CPU and network in each VM monitoring the action performing in xen. Memory usage in VM is largely hidden to hypervisor. Shortage of memory is indicated by swap activities.

VM scheduler gathers the usage of resources information frequently from all PM's; send it to Usher Center Control to schedule the VM. Predictor in VM scheduler predicts the feature resource needs by resource forecasting algorithm. Memory and CPU allocations are changing in xen, when a new demands came for processing LMN has to satisfy those requests by adjusting the old one.

**Hotspot** solver in VM Scheduler identifies the usage of resources from PM. If its usage is high then we can call it as "**Hot threshold**". VM scheduler migrate those VM to reduce the burden on PM. If the VM uses resources averagely from PM's it is below the "**Green Computing Threshold or Cold Threshold**". We are shut downing PM's to save the energy those are in under utilization list. Migrate all the listed VM to Usher CTRL for execution.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 4, September 2014

## III. FORECASTING FUTURE RESOURCE BY CALCULATION

Smart service maintenance has some parameter like efficiency, security and flexibility. We have to take care by monitoring and analyze the service periodically then only we can provide effective services to clients. Depending upon demands we require some changes in VM to maintain it as effective service and demand for that service increases. Based on previous usage history we can adapt some services and make changes in hardware also. Here our main aim to utilize present resource in VM effectively by monitoring usage. This is possible by necessary calculation. For this we proposed exponentially weighted moving average (EWMA) in TCP.

$$E(t) = \alpha * E(t - 1) + (1 - \alpha) * O(t), 0 \leq \alpha \leq 1.$$

Where E (t): Estimate load at time t.
O (t): Observed load at time t.

Forecast the CPU weights on DNS server by using the EWMA formula. We compute the weights periodically to reduce burden and estimate next weight on it. We show the CPU utilization in a graphical representation to adjust the weights on it. In Fig. 2a show the results for $\alpha = 0.7$. Dot represents the observed value and curve represents the Predicted value in the Graph. Curve cuts the middle of the dots which indicates a quite precise prediction. This is also verified by table 1. Median errors are computed based on observed value: $| E(t) - O(t)| / O(t)$.

Even though it seems that it is suitable, but it was not getting resource usage fluctuation details. Unfortunately when $\alpha$ is between 0 and 1, the predicted value is always between historic value and observed one. So we required small changes to speed up we set $\alpha$ value as negative ($-1 \leq \alpha \leq 0$).

We have to balance the observed O (t) and estimated E (t) loads on time being. For example if observed resource usage is going down. Here we can to decrease the estimated load in some situations. Based on the situation we have to take precaution to maintain the server. For this we have two parameters $\uparrow\alpha$ and $\downarrow\alpha$ to balance the server. We can call this as Fast Up and Slow down (FUSD) algorithm. We can see in fig 2b show the effectiveness of the FUSD algorithm for $\uparrow\alpha = -0.2$ and $\downarrow\alpha = 0.7$ these values are obtained by doing experiment on online application. We keep a window W recently observed value and take O(t) as hight. We take W=8 and increase the percentage of resource demand to peak usage that is 90%$^{th}$ show in the Fig. 2c. Below

Fig.2, CPU load prediction for the DNS server at our university. W is the measurement window

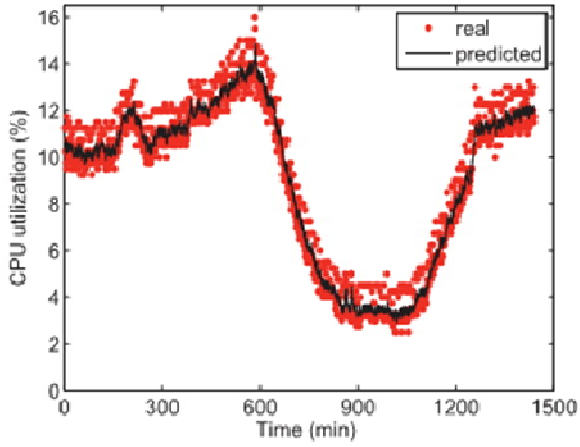



Fig. 2 (a) EWMA $\alpha = 0.7$, W = 1

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol.2, Special Issue 4, September 2014**

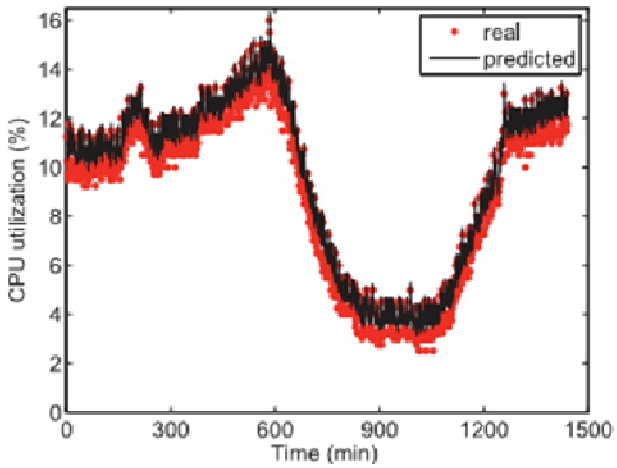



Fig. 2 (b) FUSD: ↑ α = - 0.2, ↓ α = 0.7, W = 1

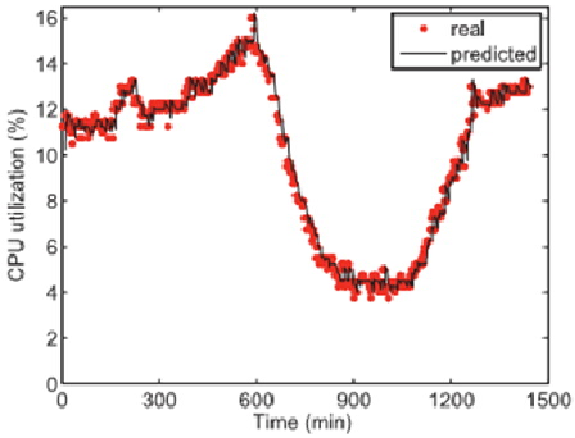



Fig. 2 (c) FUSD: ↑ α = - 0.2, ↓ α = 0.7, W = 8

| | **ewma (0.7) W = 1** | **fusd (-0.2, 0.7) W = 1** | **fusd(-0.2, 0.7) W = 8** |
|---|---|---|---|
| Median error | 5.6% | 9.4% | 3.3% |
| High error | 56% | 77% | 58% |
| Low error | 44% | 23% | 41% |

**Table 1:** Load Prediction Algorithms

## *IV.* SKEWNESS ALGORITHM

Skewness can count underutilization and overutilization of multiple resources on a server. Let *n* be the no of resources, $r_i$ be the $i^{th}$ resource usage and *p* be the server.

$$Skewness\ (p) = \sqrt{\sum_{i=1}^{n} \left(\frac{ri}{\bar{r}} - 1\right)^2}$$

Where $\bar{r}$ is the average usage of all resource for server *p.* we apply this calculation specifically on inconsistent servers to know the resource usage whether the resource is overutilization or underutilization. Migrate the workloads which is overly utilizes the resource from one server to another server. If the resources are underutilized from a server then we can joins some more workloads to improve the overall utilization of server resources smartly.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 4, September 2014

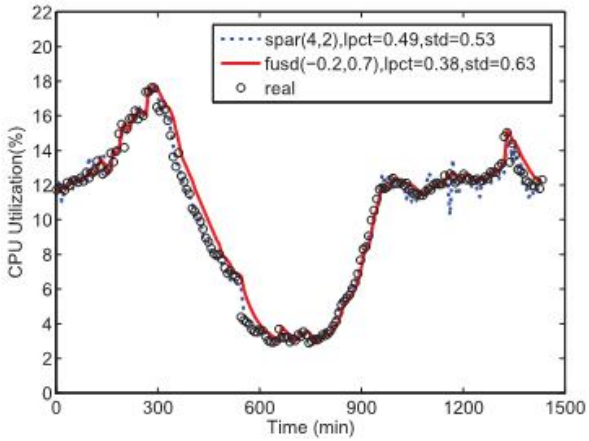


Fig. 3, Comparison of SPAR and FUSD

**4.1 Hot and Cold Spots**

We have explained previously about Hotspot and cold spot in section 2. Depending on the Load we are defining temperature of a server. If the utilization of resources in server is high then it is in hot threshold. This shows the server is highly busy with overload on it. By reducing burden on server, it performs its tasks nicely. We are diverting some VM's to another PM's which is suitable to fulfill their requirements. *p* is the temperature of hotspot.

$$Temperature\ (p) = \sum_{r \in R} (r - rt)^2$$

Where R is the set of overloaded resources in server p and $r_t$ is the hot threshold for resource r. if the temperature is zero then the server is in cold spot. if temperature is one server is in hot spot.

If the utilization of resource is below average to its actual capacity then it is in cold threshold. Potentiality of server is wastage due to the server running idly, so it was not satisfying the green computing threshold, to save the energy we are show downing it. The utilization of all resources from server which satisfies the green computing threshold but not in risk stage then we can call it as warm threshold.

**4.2 Hot Spot Improvement**

Here our aim to remove hot spots in a server to provide smooth accesses flexibility to client. A client has authority to customize their service within his long run. While providing these services, server has to allocate high level of resources to satisfy their needs based on their demands. At this time server is busy (server temperature is high or Hot spot) while supplying those resources and give high priority for new demands, while providing these service a problem may arises for already using the same resource going down. Overcome this problem by migrating VM's to another server's. With the help of Skewness we are collecting the list of VM's which are over utilizing the resources from server, similarly for remaining servers. A VM migration which depends on highest priority in hot spot list that is in descending order. Responsibility of server has to check before migrate the VM from one server (source server) to another server (Destination server). Source server has to check whether the destination server is suitable, if it provides all his resource or not and after accepting this VM it does not go to hotspot. While seeing all these a aspect then we are migrating the VM to another server. In this way we are optimizing the server temperature as low as possible.

**4.3 Green Computing**

The main intension of "Green computing algorithm" is to utilize the maximum level of resources on server and also it optimize the number of active servers running with less weight it does not sacrifice its performance now or in future. Our aim is to optimize the server resources by migrating VM's to another suitable server which are underutilized its resources. By doing this we can save the server's energy. To avoid this green computing algorithm request's periodically to get the information of underutilized resources which satisfies green computing threshold from all servers in cloud also known as cold spot. By collect this information we are migrating the VM's in ascending order based on memory size.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol.2, Special Issue 4, September 2014**

## V. CONCLUSION

We have presented an approach for implementation and evaluation of a resource management system for cloud computing services. We have also shown in our paper of how we can multiplex virtual resource allocation to physical resource allocation effectively based on the fluctuating demand. We also make use the skewness metric to determine different resource characteristics appropriately so that the capacities of servers are well utilized. We can apply our algorithm to achieve both overload avoidance and green computing for systems which support multi-resource constraints.

## REFERENCES

[1] Zhen Xiao, Senior Member, IEEE, Weijia Song, and Qi Chen, "Dynamic Resource Allocation Using Virtual Machines for Cloud Computing Environment", VOL. 24, NO. 6, JUNE 2013
[2] L. Siegele, "Let It Rise: A Special Report on Corporate IT," The Economist, vol. 389, pp. 3-16, Oct. 2008.
[3] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the Art of Virtualization," Proc. ACM Symp. Operating Systems Principles (SOSP '03), Oct. 2003.
[4] "Amazon elastic compute cloud (Amazon EC2)," http://aws. amazon.com/ec2/, 2012.
[5] C. Clark, K. Fraser, S. Hand, J.G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live Migration of Virtual Machines," Proc. Symp. Networked Systems Design and Implementation (NSDI '05), May 2005.
[6] M. Nelson, B.-H. Lim, and G. Hutchins, "Fast Transparent Migration for Virtual Machines," Proc. USENIX Ann. Technical Conf., 2005.
[7] M. McNett, D. Gupta, A. Vahdat, and G.M. Voelker, "Usher: An Extensible Framework for Managing Clusters of Virtual Machines," Proc. Large Installation System Administration Conf. (LISA '07), Nov. 2007.
[8] T. Wood, P. Shenoy, A. Venkataramani, and M. Yousif, "Black-Box and Gray-Box Strategies for Virtual Machine Migration," Proc. Symp. Networked Systems Design and Implementation (NSDI '07), Apr. 2007.
[9] C.A. Waldspurger, "Memory Resource Management in VMware ESX Server," Proc. Symp. Operating Systems Design and Implementation (OSDI '02), Aug. 2002.