A Survey on Energetic Service Allocation using Virtual Machines for Cloud Computing Domain

Santhosh Naik D[1], Dr. G. F Ali Ahammed[2]
[1] Student, M.Tech, Dept. of computer Science and engineering
VTU, CPGS Bangalore Region
[2] Associate Professor, Dept. of computer Science and engineering
VTU, CPGS Bangalore Region
Bangalore, India.

Abstract **:- The cloud user may scale up or down their resource needs from cloud provider depending on their requirements. In cloud environment preventing overloading of different virtual machines on physical machines and making efficient use of resources becomes major problem. In this paper we make use of virtualization technique for allocating resources to different business users depending on their needs i.e. for mapping different virtual machines to physical machines depending on their resource needs. And to avoid overloading we use load prediction algorithm, green computing technique for minimizing number of physical machines used as long as they can satisfy resource needs and making efficient use of PM's.**

1. INTRODUCTION

In this paper we mainly concentrates only on two main concepts overload avoidance and green computing. We also learn about how a service provider of cloud can best multiplexing its virtual resources. A cloud model should satisfy all resource needs of all business customers regardless of elasticity nature of its entire service user. Thus a cloud model is expected to have a scale up and down in order to manage the load variation. In this paper cloud model which we propose can also reduce the hardware cost and electricity can be saved. In existing VMM such as Xen the mapping of virtual machine to the physical machines a mechanism provided by the virtual machine monitors are hidden from the cloud users. So .it's the responsibility of the cloud providers to make the resources meet their needs. The live migration technology of VM makes VM and PM mapping possible even under execution running. The two main goals that we achieve here are.

1. The utilization of PM's should be done efficiently by making use of each PM's to their maximum threshold.
2. The number of PM's should be minimized this can be achieved by combining different types of workloads nicely and effectively. Thus in this case we have to maintain the utilization of Pm's high to satisfy their needs.

The three main contributions we have made in this paper are

- To avoid overloading we develop a load prediction algorithm that helps our resource allocation system to efficiently allocate resources to all virtual machines and also minimizes the total number of servers used.
- To measure unevenness in multi-dimensional resource environment we introduce a concept called "skewness" it will help in combining different types of workload.
- We also use green computing to minimize the number of PM's used to satisfy resource needs of all VM's.To save energy Idle PM's can be turned off.

Virtual Machine Monitors:
A VMM is nothing but a software program that permits the creation, management and governance of virtual machines and it handles each operation of virtualization environment on top PM.

Virtualization:
Virtualization is a process of creating a virtual version of resource such as server, storage, network or even an operating system. It results in sharing of single CPU among various operation systems. Memory can be shared using more level of indirections. Virtualization architecture gives an illusion through a hypervisor.

Virtual machine:
VM is a software implementation of a computing environment in which we can install an operating system or program and run.

Amazon's cloud uses Xen VMM for virtualization infrastructure though it provide mapping of virtual machined to physical resources the mapping is largely hidden from the user. The users of Amazon EC2 service for example they do not know where their VM instance runs. Even though Xen provides live migration technique the policy issue remains as how to decide mapping efficiently so that the resource demands of all VMs are met with the number of PMs used minimized and this becomes challenging when resource needs of VMs are heterogeneous.

2. SYSTEM ARCHITECTURE

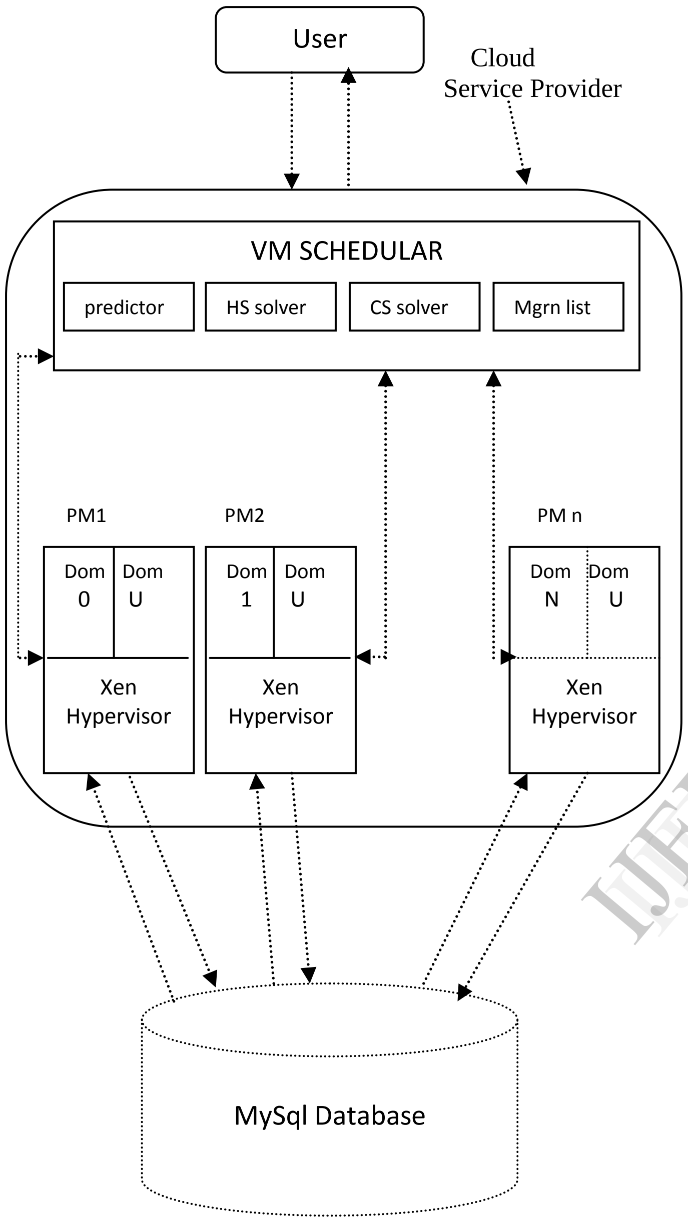


Fig 1.System Architecture.

Figure 1 represents the architecture of the system each PM executes Virtual Machine Monitor (Xen hypervisor) which will supports a entitled domain 0 and one or more domain U.VM machine of each domain U supports one or more applications such as remote desktop, mail etc.The Usher framework facilitates multiplexing of VMs to PMs. The system which we present can be viewed as a set of plug-ins to Usher.

VMM it act as primary software behind virtualization environment and implementation.VMM provides facilities such as creation of VM's, each with a separate applications and operating system.

Usher central controller gathers history of resources from each PM, VMM (Virtual Machine Monitor) runs in Usher central controller. The VM scheduler receives load history of PMs and demand history of VMs.

The Predictor consists of several components such as predictor, hot spot solver, cold spot solver, migration list. Where predictor used to predict future load, hot spot and cold spot used for skewness.

3 LOAD PREDICTION ALGORITHM

To predict the load on PMs we need to estimate the future resource needs of VMs.We make our load prediction depending on the past resource load on each VMs.One solution is to examine the VM for application level statistics e.g. by inspecting logs of pending request. We use TCP-like scheme to find exponentially weighted moving average (EWMA).

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

Where E(t) is estimated load and $O(t)$ is observed load at time t, α is tradeoff between stability and responsiveness.

We make use of following two parameter for combining multiple requests.

Skewness: It measures the unevenness utilization of PMs or servers which are used to satisfy the resource needs of VMs. When the value of skewness is high then it means multiple servers are used to satisfy the resource needs. By minimizing this value we can combine multiple workloads nicely and hence we can improve utilization servers to satisfy resource needs.

Threshold: It is the intensiveness of PMs that must be exceeded for certain condition or particular results.

We use two types of threshold values here.

1. Hot threshold: A threshold value that indicates maximum number of resource request a server can handle without overloading.

2. Cold threshold: A threshold value that indicates minimum number of resource request that a server must handle.

In this paper a server can be defined as **hot spot** if utilization of resources is above the **hot threshold.** This indicates that the server is under over utilization and hence some resource request should be moved to other servers or PMs.In similar way a server can be defined as **cold spot** if utilization of resources is below the **Cold threshold.** So if the system is in cold spot it means the server is not being used efficiently so resource request should be moved to other server and this system must be turned off for energy saving purpose.

The temperature of a hotspot can be defined as square sum of its resource utilization above the hot threshold.

$$\text{temperature } (p) = \sum_{r \in R} (r - r_t)^2$$

Where p and r_t is the hot threshold for resource r and R is the set of overloaded resources. The degree of the overload will be indicated by the temperature of hot spot. By using this temperature or overload of a server we can predict the future workload on a particular server or PM.

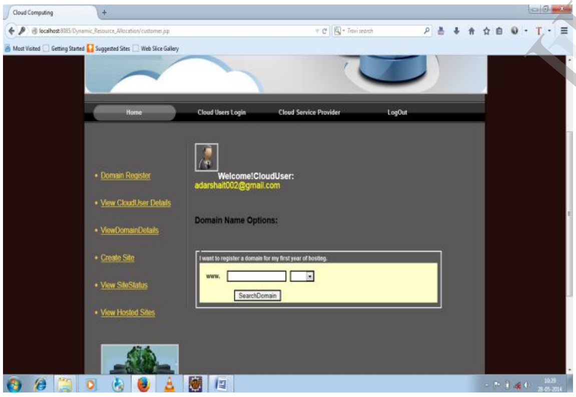
4. GREEN COMPUTING

In this paper we support Green Computing by minimizing the number of servers in use. To save energy some of the active server can be turned off only when if the resource utilization of active server is low. Here the challenge is by giving importance on reducing number of active servers in use we should not degrade performance now or in future and we should avoid variation in the system.

Our Green Computing Algorithm will be invoked only when the average resource utilization of all resources on active servers are less than some green computing threshold. Before going to shutdown the active server which are in use we should migrate all its VMs,this can be done by sorting the list of all cold spots in the system in ascending order based on their memory size. The cold spot with lowest cost will be eliminated first.

For each VM on cold spot *p* we try to find a destination server to migrate it and finally all VMs of cold spot *p* will be moved to that destination. After migrating all VMs to destination server, the resource utilization of accepting server should be below hot threshold. Over-doing migration process may result in hot spots in future the *hot threshold* is designed to prevent this. However we accept *cold spot* as destination server while migration. By keeping the record of sequence of migration will help in predicting load on a server in advance.

Screen shots

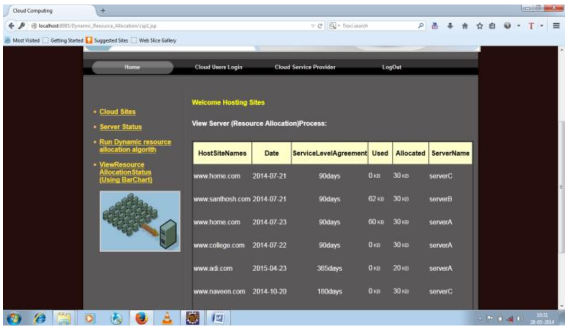


Welcome!CloudUser:
adarshait002@gmail.com

Domain Name Options:

I want to register a domain for my first year of hosting.

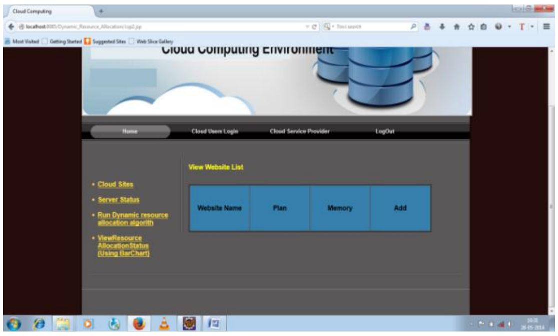
www.



Welcome Hosting Sites

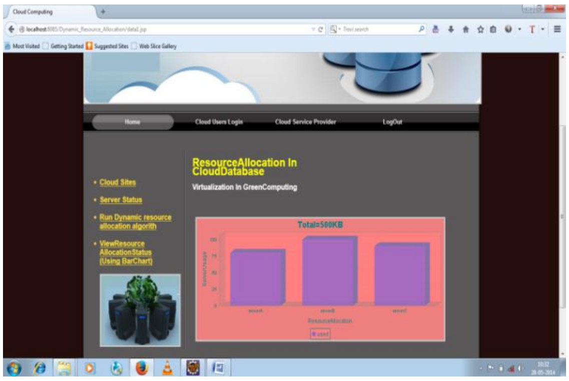
View Server (Resource Allocation)Process:

HostSiteNames	Date	ServiceLevelAgreement	Used	Allocated	ServerName
www.home.com	2014-07-21	90days	0 KB	30 KB	serverC
www.santhosh.com	2014-07-21	90days	62 KB	30 KB	serverB
www.home.com	2014-07-23	90days	60 KB	30 KB	serverA
www.college.com	2014-07-22	90days	0 KB	30 KB	serverA
www.adi.com	2015-04-23	365days	0 KB	20 KB	serverA
www.naveen.com	2014-10-20	180days	0 KB	30 KB	serverC


Cloud Computing
localhost:8085/Dynamic_Resource_Allocation/csp2.jsp
Most Visited
Getting Started
Suggested Sites
Web Slice Gallery
Cloud Computing Environment
Home
Cloud Users Login
Cloud Service Provider
LogOut
Cloud Sites
Server Status
Run Dynamic resource allocation algorith
ViewResource AllocationStatus (Using BarChart)
10:31
28-05-2014

View Website List

Website Name	Plan	Memory	Add



ResourceAllocation In CloudDatabase

Virtualization In GreenComputing

5. CONCLUSION

In this paper the resource management concept in cloud computing is implemented successfully and we have accomplished the overload avoidance using load prediction algorithm and the green computing concept achieved successfully. To make server utilization efficient we used the concept of skewness that will combine variousVMs.

REFERENCESS

[1] Armbrust .M et al., “Above the clouds: A Berkeley view of cloud computing,” University of California, Berkeley, Tech. Rep., Feb 2009.

[2] Siegele .L “Let it rise: A special report porate IT,” in The Economist, Oct. 2008.

[3] A. Singh, M. Korupolu, and D. Mohapatra, “Server-storage virtualization: integration and load balancing in data centers,” in Proc. of the ACM/IEEE conference on Supercomputing, 2008.

[4] T. Wood, P. Shenoy, A. Venkataramani, and M. Yousif, “Black-box and gray-box strategies for virtual machine migration,” in Proc. of the Symposium on Networked Systems Design and Implementation (NSDI’07), Apr. 2007.

[5] G. Chen, H. Wenbo, J. Liu, S. Nath, L. Rigas, L. Xiao, and F. Zhao, “Energy-aware server provisioning and load dispatching for connection- intensive internet services,” in Proc. of the USENIX Symposium on Networked Systems Design and Implementation (NSDI’08), Apr. 2008.

[6] P. Padala, K.-Y. Hou, K. G. Shin, X. Zhu, M. Uysal, Z. Wang, S. Singhal, and A. Merchant, “Automated control of multiple virtualized resources,” in Proc. of the ACM European conference on Computer systems (EuroSys’09), 2009.

[7] M. Zaharia, D. Borthakur, J. Sen Sarma, K. Elmeleegy, S. Shenker, and I. Stoica, “Delay scheduling: a simple technique for achieving locality and fairness in cluster scheduling,” in Proc. of the European conference on Computer systems (EuroSys’10), 2010.

[8] T. Sandholm and K. Lai, "Mapreduce optimization using regulated dynamic prioritization," in Proc. of the international joint conference on Measurement and modeling of computer systems (SIGMETRICS'09), 2009.

[9] Y. Agarwal, S. Hodges, R. Chandra, J. Scott, P. Bahl, and R. Gupta, "Somniloquy: augmenting network interfaces to reduce pc energy usage," in Proc. Of the USENIX symposium on Networked systems design and implementation (NSDI'09), 2009. [10] T. Das, P. Padala, V. N. Padmanabhan, R. Ramjee, and K. G. Shin, "Litegreen: saving energy in networked desktops using virtualization," in Proc. of the USENIX Annual Technical Conference, 2010.

[11] Y. Agarwal, S. Savage and R. Gupta, "Sleepserver: a software-only approach for reducing the energy consumption of pcs within enterprise environments," in Proc. of the USENIX Annual Technical Conference, 2010.

[12] N. Bila, E. d. Lara, K. Joshi, H. A. Lagar- Cavilla, M. Hiltunen and M. Satyanarayanan, "Jettison: Efficient idle desktop consolidation with partial vm migration," in Proc. of the ACM European conference on Computer systems (EuroSys'12), 2012.