On-Demand Resource Provisioning Using Virtual Machines in Cloud Computing

BATTE RAJESH
M.Tech Student,
Dept. of Computer Science and Engineering,
Sri Venkateswara College of Engineering and Technology,
Chittoor (D), Andhra Pradesh, India

DR.J.JANNET
Professor,
Dept. of Computer Science and Engineering,
Sri Venkateswara College of Engineering and Technology,
Chittoor (D), Andhra Pradesh, India

Abstract:
Cloud computing provides more flexibility to business customers to grow up and down their resource usage based on needs. Many of the cloud models come from resource multiplexing through virtualization technology. In this project, we have presented a system that uses virtualization technology to allocate data center resources dynamically based on application demands and support green computing by optimizing the number of servers in use. The concept of skewness is introduced to measure the unevenness in the resource utilization of a server. By minimizing skewness, we can combine different types of workloads nicely and improve the overall utilization of server resources. We have developed a set of heuristics that prevent overload in the system effectively while saving energy used.

Keywords: *Cloud Computing, Green Computing, Resource Management, Virtualization*

1. Introduction

Cloud computing, also referred as simply “the cloud”, which is providing the on-demand resources to applications from data centers through the Internet based on pay-for-use policy. Generally the meaning of cloud computing is that, accessing your application as a service by sitting somewhere in the world , or somewhere inside your company over the web without having all computer hardware and software on your desktop. It gives the convenient way for users to access their applications without having doubt about where accurately the software is running, hardware is located and how it is working, it’s just somewhere in the nebulous cloud that the internet represents. Most importantly, whatever the services you are using those are provided by someone and managed by you. Suppose you are using **Google Drive to store the documents, you no** need to worry of buying software license access for word-processing or maintaining them up on.

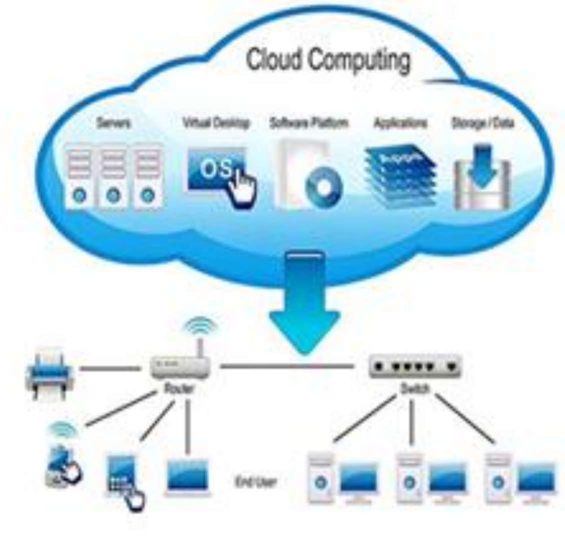

Figure 1 – Overview of cloud computing

As can be seen in Figure 2, Virtual machine monitors (VMMs) like Xen provide a mechanism for mapping virtual machines (VMs) to physical resources. This mapping is largely hidden from the cloud users. Users with the Amazon EC2 service, for example, do not know where their VM instances run. It is up to the cloud provider to make sure the underlying physical machines (PMs) have sufficient resources to meet their needs. VM live migration technology makes it possible to change the mapping between VMs and PMs while applications are running. However, a policy issue remains as how to decide the mapping adaptively so that the resource demands of VMs are met while the number of PMs used is minimized. This is challenging when the resource needs of VMs are heterogeneous due to the diverse set of applications they run and vary with time as the workloads grow and shrink.

In this paper, we present the design and implementation of an automated resource management system that achieves overload avoidance and effective utilization of resources by minimizing skewness. The remainder of the paper is structured as follows. Section 2 reviews literature. Section 3 provides details of the proposed system. Section 4 presents experimental results while section 5 concludes the paper and provides recommendations for future work.

2. Related Works

Automatic scaling of Web applications was previously studied in [1] [2] for data centre environments. In MUSE [1], each server has replicas of all web applications running in the system. The dispatch algorithm in a frontend L7-switch makes sure requests are reasonably served while minimizing the number of under-utilized servers. Work [2] uses network flow algorithms to allocate the load of an application among its running instances. For connection oriented Internet services like Windows Live Messenger, work [3] presents an integrated approach for load dispatching and server provisioning. All works above do not use virtual machines and require the applications be structured in a multi-tier architecture with load balancing provided through an front-end dispatcher.

In contrast, our work targets Amazon EC2-style environment where it places no restriction on what and how applications are constructed inside the VMs. A VM is treated like a black box. Resource management is done only at the granularity of whole VMs. Map Reduce [4] is another type of popular Cloud service where data locality is the key to its performance. Qunicy adopts min-cost flow model in task scheduling to maximize data locality while keeping fairness among different jobs [5]. The “Delay Scheduling” algorithm trades execution time for data locality [6]. Work [7] assign dynamic priorities to jobs and users to facilitate resource allocation.VM live migration is a widely used technique for dynamic resource allocation in a virtualized environment [8] [9] [10]. Our work also belongs to this category.

Sandpiper combines multi-dimensional load information into a single Volume metric [8]. It sorts the list of PMs based on their volumes and the VMs in each PM in their volume-to-size ratio (VSR). This unfortunately abstracts away critical information needed when making the migration decision. It then considers the PMs and the VMs in the pre-sorted order. We also compare our algorithm and theirs in real experiment. The harmony system applies virtualization technology across multiple resource layers [10]. It uses VM and data migration to mitigate hot spots not just on the servers, but also on network devices and the storage nodes as well. It introduces the Extended Vector Product (EVP) as an indicator of imbalance in resource utilization. Their load balancing algorithm is a variant of the Toyoda method [11] for multi-dimensional knapsack problem. Unlike our system, their system does not support green computing and load prediction is left as future work.

3. Proposed System

Our proposed architecture is shown in Figure 2. There are four modules in the proposed system. They are known as Cloud Computing Module, Resource Management Module, Virtualization Module and Green Computing Module.

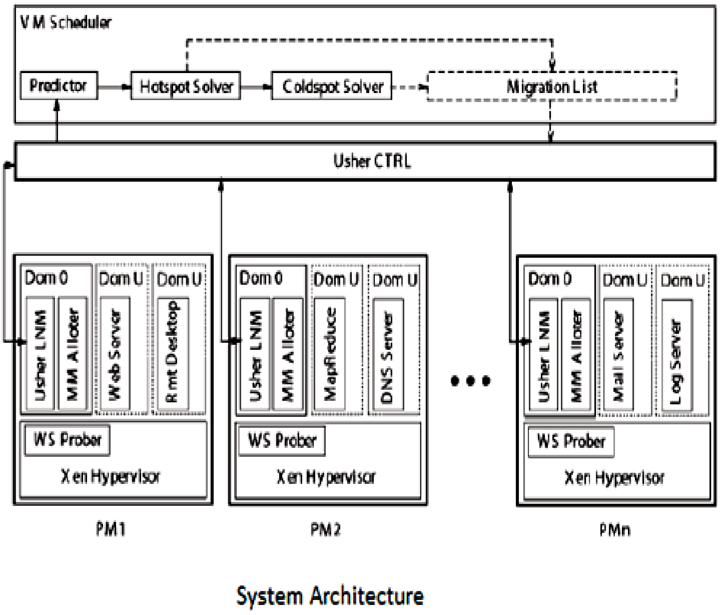

Figure 2 : Proposed architecture for on-demand resource provisioning using virtual machines

Cloud computing refers to applications and services offered over the Internet. These services are offered from data centers all over the world, which collectively are referred to as the "cloud." Cloud computing is a movement away from applications needing to be installed on an individual's computer towards the applications being hosted online. Dynamic resource management has become an active area of research in the Cloud Computing paradigm. 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. Virtualization, in computing, is the creation of a virtual (rather than actual) Version of something, such as a hardware platform, operating system, and a storage device or network resources.VM live migration is a widely used technique for dynamic resource allocation in a virtualized environment.

```
Algorithm: Skewness
IF ( Cloud Server is RUNNING ) THEN
{IF ( VM is CREATED ) THEN
{IF ( USER is logged in)
{Browse and Upload the FILE
While ( File is uploading the FILE){Check Threshold;
IF ( Threshold exceeds){“SERVER is OVERLOADED”;
return HOTSPOT;}ELSE{return COLDSPOT;}
IF (HOTSPOT){Migrate the VMs;}
IF (COLDSPOT){Migrate the VMs and turn OFF the PMs;}}}}
ELSE{Ask ADMIN to allocate VM to USER}}
```

To calculate the unevenness in the utilization of multiple resources on a server we have used the concept of skewness. Let n be the number of resources we consider and ri be the utilization of the i-th resource. We define the resource skewness of a server p as

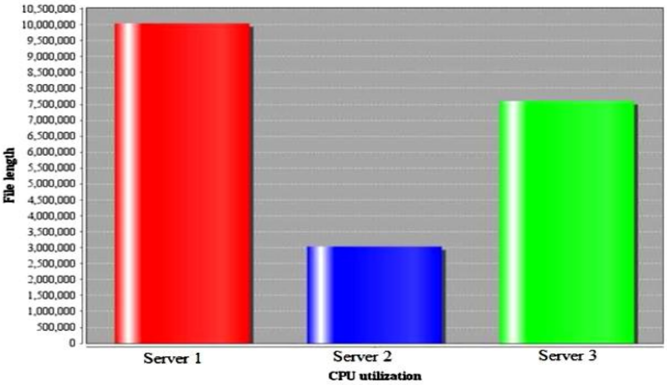

Figure 3: File Length vs CPU utilization

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

4. Experimental Results

Experiments are made in term of file length and the cpu utilization. The experiments are made with custom simulator building using Java platform. The results are compared with different algorithms.

As can be seen in Figure 3, it is evident that the cpu utilization is represented by horizontal axis while the vertical axis represents the file length. The results reveal that the proposed system shows better performance.

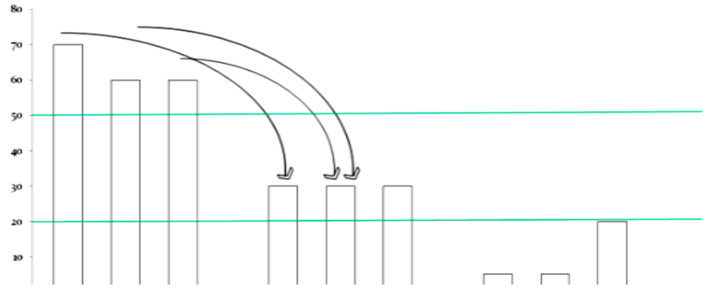

Figure 4 : Migration of load from hot spot to warm spot

As can be seen in Figure 4, it represents that the load can be migrated from hot spots to warm spots for balancing the load among cpu's. The results reveal that the proposed system shows better performance.

5. Conclusion and Future Work

The design presented in this paper is used for implementation and evaluation of a resource management system for cloud computing services. This concept produces a system which multiplexes the virtual resources to physical resources, adaptively based on the changing demand. Also the skewness algorithm is used to combine Virtual Machines with different resource characteristics appropriately so that the capacities of servers are well utilized. Skewness algorithm achieves both overload avoidance and green computing for systems with multi-resource constraints. Thus resource allocation on Server under high load is avoided. Failure or crashes of server is limited. The overall utilization of resources is improved without performance degradation. In future the proposed work can be extended by the development of software platform that supports energy efficient management and allocation of data center resources. Automatic scaling of Web applications was studied in for data center environments. The dispatch algorithm in a frontend L7-switch makes sure requests are reasonably served while minimizing the number of under-utilized servers. Work uses network flow algorithms to allocate the load of an application among its running instances.

References

1. J. S. Chase, D. C. Anderson, P. N. Thakar, A. M. Vahdat, and R. P. Doyle, "Managing energy and server resources in hosting centers," in Proc. Of the ACM Symposium on Operating System Principles (SOSP'01), Oct. 2001.
2. C. Tang, M. Steinder, M. Spreitzer, and G. Pacifici, "A scalable application placement controller for enterprise data centers," in Proc. Of the International World Wide Web Conference (WWW'07), May 2007.
3. 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.
4. M. Zaharia, A. Konwinski, A. D. Joseph, R. H. Katz, and I. Stoica, "Improving MapReduce performance in heterogeneous environments," in Proc. of the Symposium on Operating Systems Design and Implementation (OSDI'08), 2008.
5. M. Isard, V. Prabhakaran, J. Currey, U. Wieder, K. Talwar, and A. Goldberg, "Quincy: Fair scheduling for distributed computing clusters," in Proc. of the ACM Symposium on Operating System Principles (SOSP'09), Oct. 2009.
6. 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.
7. 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.
8. 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.

9. N. Bobroff, A. Kochut, and K. Beaty, “Dynamic placement of virtual machines for managing sla violations,” in Proc. of the IFIP/IEEE International Symposium on Integrated Network Management (IM’07), 2007.
10. 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.
11. Y. Toyoda, “A simplified algorithm for obtaining approximate solutions to zero-one programming problems,” Management Science, vol. 21, pp. 1417–1427, august 1975.

Biography

Mr. Batte Rajesh is a PG Scholar in Computer Science and Engineering at Sri Venkateswara College of Engineering and Technology, Chittoor. My research area is Cloud Computing.

Dr. J. Jannet M.E., Ph. D. is working as Head of the Department, Computer Science Department at Sri Venkateswara College of Engineering & Technology, Chittoor. Her Research Area is Cloud Computing