DYNAMIC RESOURCE ALLOCATION USING VIRTUAL MACHINES FOR CLOUD COMPUTING ENVIRONMENT

[1]SHAIK KARIMUNNI, [2]SHAIK KARIMULLA, [3]P.BABU

PG SCHOLAR, CSE, QCET, NELLORE

ASSOCIATE PROFESSOR, CSE, QCET, NELLORE

ASSOCIATE PROFESSOR, CSE, QUBA ENGINEERING COLLEGE, NELLORE

ABSTRACT: *Cloud computing allows business customers to scale up and down their resource usage based on needs. Many of the touted gains in the cloud model come from resource multiplexing through virtualization technology. In this paper, we present 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. We introduce the concept of "skewness" to measure the unevenness in the multidimensional 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 develop a set of heuristics that prevent overload in the system effectively while saving energy used. Trace driven simulation and experiment results demonstrate that our algorithm achieves good performance.*

1. INTRODUCTION

The elasticity and the lack of upfront capital investment offered by cloud computing is appealing to many businesses. There is a lot of discussion on the benefits and costs of the cloud model and on how to move legacy applications onto the cloud platform. Here we study a different problem: how can a cloud service provider best multiplex its virtual resources onto the physical hardware? This is important because much of the touted gains in the cloud model come from such multiplexing. Studies have found that servers in many existing data centers are often severely under-utilized due to over-provisioning for the peak demand. The cloud model is expected to make such practice unnecessary by offering automatic scale up and down in response to load variation. Besides reducing the hardware cost, it also saves on electricity which contributes to a significant portion of the operational expenses in large data centers.

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. The capacity of PMs can also be heterogeneous because multiple generations of hardware co-exist in a data center.

We aim to achieve two goals in our algorithm:

1. overload avoidance: the capacity of a PM should be sufficient to satisfy the resource needs of all VMs running on it. Otherwise, the PM is overloaded and can lead to degraded performance of its VMs.

*2.*green computing: the number of PMs used should be minimized as long as they can still satisfy the needs of all VMs. Idle PMs can be turned off to save energy.

There is an inherent trade-off between the two goals in the face of changing resource needs of VMs. For overload avoidance, we should keep the utilization of PMs low to reduce the possibility of overload in case the resource needs of VMs increase later. For green computing, we should keep the utilization of PMs reasonably high to make efficient use of their energy.

In this paper, we present the design and implementation of an automated resource management system that achieves a good balance between the two goals. We make the following

2.EXISTING SYSTEM:

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. The capacity of PMs can also be heterogeneous because multiple generations of hardware coexist in a data center.

2.1 DISADVANTAGES OF EXISTING SYSTEM:

- 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. The two main disadvantages are overload avoidance and green computing.

3.PROPOSED SYSTEM:

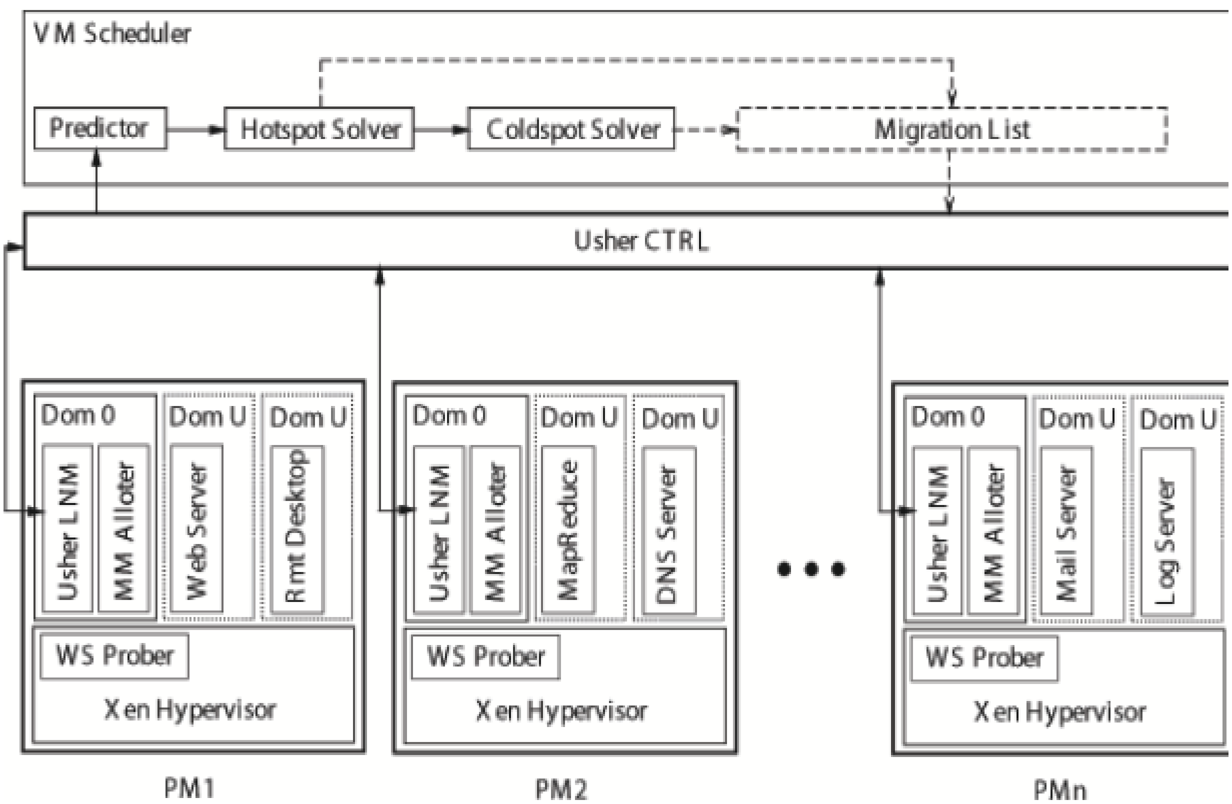
In this paper, we present the design and implementation of an automated resource management system that achieves a good balance between the two goals. Two goals are **overload avoidance** and **green computing**.

1. **Overload avoidance**: The capacity of a PM should be sufficient to satisfy the resource needs of all VMs running on it. Otherwise, the PM is overloaded and can lead to degraded performance of its VMs.
2. **Green computing**: The number of PMs used should be minimized as long as they can still satisfy the needs of all VMs. Idle PMs can be turned off to save energy.

3.1ADVANTAGES OF PROPOSED SYSTEM:

I develop a resource allocation system that can avoid overload in the system effectively while minimizing the number of servers used.I introduce the concept of "skewness" to measure the uneven utilization of a server. By minimizing skewness, we can improve the overall utilization of servers in the face of multidimensional resource constraints. I design a load prediction algorithm that can capture the future resource usages of applications accurately without looking inside the VMs. The algorithm can capture the rising trend of resource usage patterns and help reduce the placement churn significantly.

4.SYSTEM ARCHITECTURE:



Hardware Requirements

Pentium IV processes architecture

256 MB RAM.   40 MB Hard Disk Space

Software Requirements

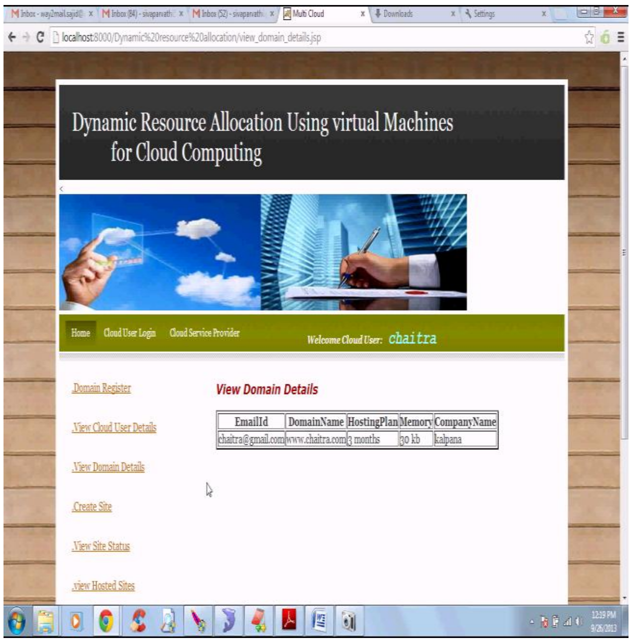
Operating System : Windows

Technology : Java and J2EE

Web Technologies : Html, JavaScript, CSS

Database : MySql

Java Version : JDK 1.7



5.CONCLUSION:

We have presented the design, implementation, and evaluation of a resource management system for cloud computing services. Our system multiplexes virtual to physical resources adaptively based on the changing demand. We use the skewness metric to combine VMs with different resource characteristics appropriately so that the capacities of servers are well utilized. Our algorithm achieves both overload avoidance and green computing for systems with multi resource constraints..

REFERENCES

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

[2] L. Siegele, "Let it rise: A special report on corporate IT," in The Economist, 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," in Proc. of the ACM Symposium on Operating Systems

Principles (SOSP'03), Oct. 2003.

[4] "Amazon elastic compute cloud (Amazon EC2), http://aws.amazon.com/ec2/."

[5] C. Clark, K. Fraser, S. Hand, J. G. Hansen, E. Jul, C. Limpach, I. Pratt, and A. Warfield, "Live migration of virtual machines," in Proc. of the Symposium on Networked Systems Design and Implementation (NSDI'05), May 2005.

[6] M. Nelson, B.-H. Lim, and G. Hutchins, "Fast transparent migration for virtual machines," in Proc. of the USENIX Annual Technical Conference,

2005.

[7] M. McNett, D. Gupta, A. Vahdat, and G. M. Voelker, "Usher: An extensible framework for managing clusters of virtual machines," in

Proc. of the Large Installation System Administration Conference (LISA'07), Nov. 2007.

[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] C. A. Waldspurger, "Memory resource management in VMware ESX

server," in Proc. of the symposium on Operating systems design and implementation (OSDI'02), Aug. 2002.

[10] 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.

[11] 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.

[12] 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.

[13] "TPC-W: Transaction processing performance council, http://www.tpc.org/tpcw/."

[14] 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.

[15] 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.

[16] 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.

[17] 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.

[18] 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.

[19] T. Sandholm and K. Lai, "Mapreduce optimization using regulated dynamic prioritization," in Proc. of the international joint conference