'National Conference on Recent Advances of Computational Intelligence Techniques in Science, Engineering and Technology'
International Journal of Scientific Research in Computer Science, Engineering and Information Technology | ISSN : 2456-3307 (www.ijsrcseit.com)

Framework for Efficient Resource Allocation in Cloud with Green Computing

Mrs. K. Revathi
Assistant Professor
Department of Computer Science and Engineering, East Point College of Engineering, Bengaluru, Karnataka, India

ABSTRACT

Green cloud depict the potential environmental benefits that information technology services delivered over the Internet can offer society. Green computing is the environmentally responsible and eco-friendly use of computers and their resources. This practices include the development of environmentally sustainable production practices, energy efficient computers and improved disposal and recycling procedures. In this paper we present a system where virtualization is used to allocate resources based on application demands. Here the main aim is to minimize the skewness. "Skewness" is uneven resource utilization of a server. It prevent overload in the system effectively by migrating the request from server to server. Because when the PM is overloaded it can lead to degraded performance of its VMs. Here a set a algorithms were used to predict the load and avoid overloading of Physical Machines. Idle Physical machines are turned off to save energy. Overload avoidance is maintained in this paper which leads to achieve good performance.

Keywords— Cloud Computing, Virtualization, Green Cloud

I. INTRODUCTION

The primary environmental problem associated with the cloud is energy use. According to market research conducted by Pike Research, the wide-spread adoption of cloud computing could lead to a potential 38% reduction in 2020. virtualization is software that separates physical infrastructures to create various dedicated resources. It is the fundamental technology that powers cloud computing. Virtualization software makes it possible to run multiple operating systems and multiple applications on the same server at the same time. Virtualization software makes it possible to run multiple operating systems and multiple applications on the same server at the same time. Virtual machines (VMs) provide flexibility and mobility through easy migration, which enables dynamic mapping of VMs to available resources.

Cloud computing has several appealing implications where under-provisioning of resources compromise service quality and over-provisioning wastes investment as well as electricity.Some times in data mining applications, a user may require a large number of servers for a short period of time.

Such requirement is possible only because of cloud computing.Main uses of virtualization concepts are:
Maximize resources — Virtualization can reduce the number of physical systems you need to acquire, and you can get more value out of the servers. Most traditionally built systems are underutilized. Virtualization allows maximum use of the hardware investment.
Multiple systems — With virtualization, you can also run multiple types of applications and even run different operating systems for those applications on the same physical hardware.
IT budget integration — When you use virtualization, management, administration and all the attendant requirements of managing your own infrastructure remain a direct cost of your IT operation.

II. RELATED WORK

The Green computing concept of closing the idle server is discussed in [6][7]. In *LiteGreen*[6], the desktop system saves energy by migration between user PM and VM server. Thus, the user's desktop environment is "always on", maintaining its network presence fully even when the user's physical desktop machine is switched off and thereby saving energy. This seamless operation allows LiteGreen to save energy during short idle periods as well the desktop VM is migrated back to the physical desktop machine. Thus, even when it has been migrated to the VM server, the user's desktop environment remains alive (i.e., it is "always on"), so ongoing network connections and other activity are *not* disturbed, regardless of the application involved.
According to dynamic server migration and consolidation algorithm is introduced. The algorithm is shown to provide substantial improvement over static server consolidation in reducing the amount of required capacity and the rate of service level agreement violations. Benefits accrue for workloads that are variable and can be forecast over intervals shorter than the time scale of demand variability. The management algorithm reduces the amount of physical capacity required to support a specified rate of SLA violations for a given workload by as much as 50% as compared to static consolidation approach. Another result is that the rate of SLA violations at fixed capacity may be reduced by up to 20%.
In Energy aware server provisioning and load dispatching for connection intensive internet services. Two techniques are used 1. dynamic provisioning that dynamically turns on a minimum number of servers required to satisfy application specific quality of service. 2. Load dispatching that distributes current load among the running machines.
According to sleep server concept SleepServer, a system that enables hosts to transition to such low-power sleep states while still maintaining their application's expected network presence using an on demand proxy server. Our approach is particularly informed by our focus on practical deployment and thus SleepServer is designed to be compatible with existing networking infrastructure, host hardware and operating systems. Using SleepServer does not require any hardware additions to the end hosts themselves, and can be supported purely by additional software running on the systems under management.

III. PROPOSED SYSTEM OVERVIEW

The User request for the VM and the VM scheduler checks for the authentication of request.

After that the skewness algorithm is performed and skewness is calculated. The result of skewness is forwarded to the VM scheduler. It checks the server status and also it verifies the status. If the server is not HOTSPOT, then the request is forwarded to server otherwise migration Process takes place.
Here the request is migrated from Hotspot server to warm spot server. So that Load balancing can be achieved. Then to achieve the Green computing the server which is in cold spot for long time a timer is triggered. The VM scheduler request for the server details and also its active connections. The server monitors the details and verify whether to shut down the server. The VM scheduler request for the prediction algorithm and its predicts the future resource needs. Then the processes running in that servers are migrated and the particular cold spot server is shutdown to achieve Green computing.

IV. OVERLOAD AVOIDANCE

In favour of performance and stability, we designed a pragmatic algorithm, Skewness. It is inspired from the fact that if a PM runs too many memory intensive VMs with light CPU load, much CPU resources will be wasted because it does not have a enough memory for extra VM. Here skewness is used to qualify the unevenness in utilization of multiple resources on a server.
The resource skewness of server p is given as

$$\text{skewness}(p)=\sqrt{\sum_{i=1}^{n}\left(\frac{v_i}{\bar{v}}-1\right)^2}$$

Here n - number of resources
Vi - utilization of i th resource
By minimizing the skewness, we can combine different types of workloads nicely and improve the overall utilization of server resources.

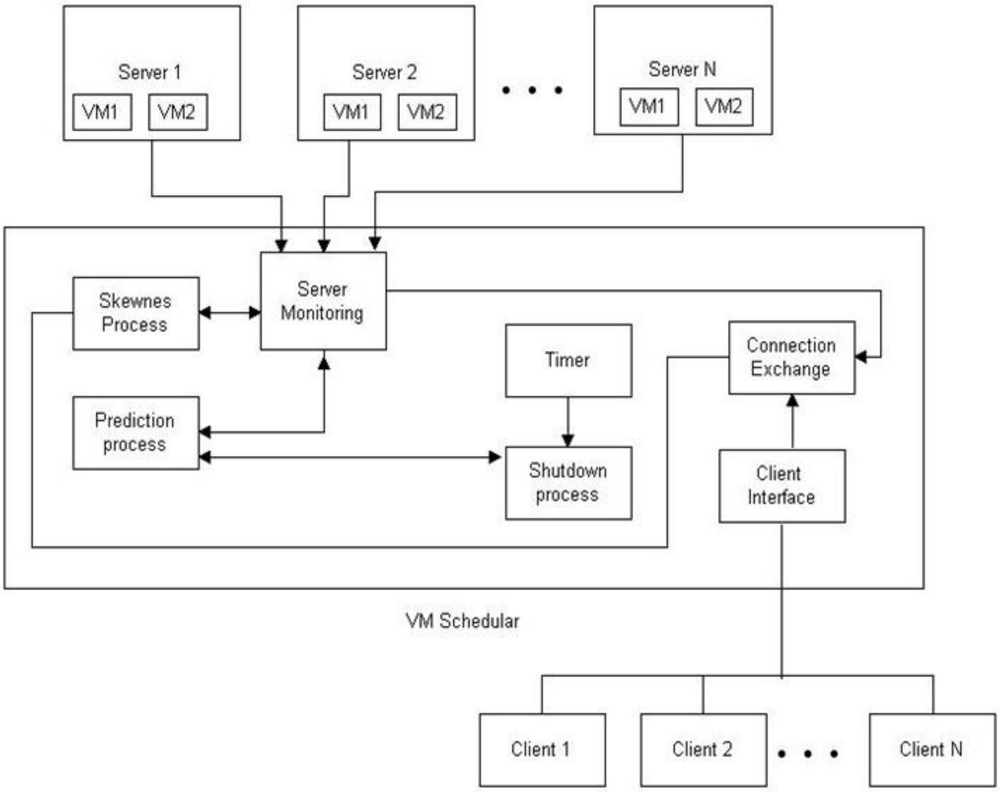
To perform skewness algorithm, we are categorizing the server status spot as:

- Hot spot-: We define a server as a hot spot if the utilization of any of its resources is above a hot threshold. This indicates that the server is overloaded and hence some VMs running on it should be migrated away.
- Cold spot-: We define a server as a cold spot if the utilizations of all its resources are below a cold threshold. This indicates that the server is mostly idle and a potential candidate to turn off to save energy.
- Warm spot-: We define a server as a warm spot if the utilizations of all its resources are below a hot threshold and below cold threshold. This indicates that the server is ready to run VMs.

For each scheduling round, the skewness takes two steps 1) Hot Spot Mitigation 2) Greencomputing to calculate a migration list.
In hot spot mitigation, we first migrate the process from the server whose temperature is hottest or which resource utilization is more. for each hot spot we try to migrate the away VM that can reduce the servers

temperature the most.We choose the server with most skewness reduction. Hot spot mitigation is finished after all hot spots are resolved.

Next the green computing step is invoked here we will try to solve the cold spots in ascending order of memory utilization. To resolve cold spot all its VMs have to be migrated such that the resource utilization should be below the warm threshold after accepting the VM. We also restrict the number of cold spots that can eliminated in each run of the algorithm



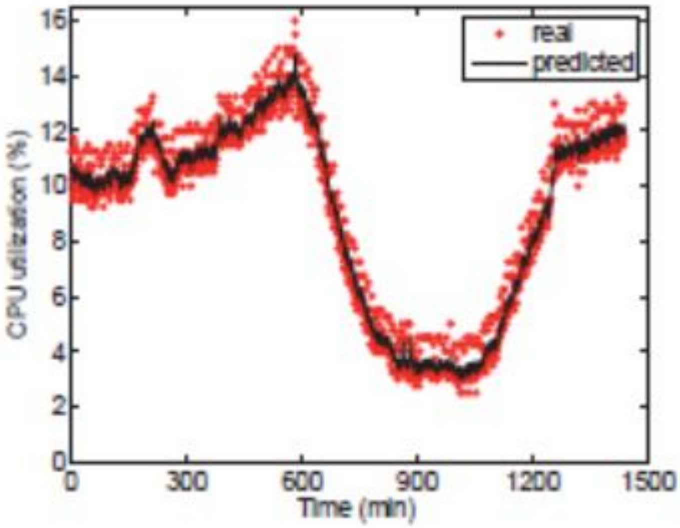
V. LOAD PREDICTION

The prediction algorithm is used to predict the future needs of VMs since we can't shut down or migrate a process to other server. One solution is to look inside the VM for application level statistics. Here we are making our prediction based on the past external behaviours of VMs. In this project, the exponentially weighted moving average (EWMA) prediction algorithm plays an important role in improving the stability and performance of our resource allocation decisions. Based on Physical machines (PMs) Usage we will select server using EWMA algorithm.

The Estimated load and Observed load values are founded by measuring the load every minute and predict the load in the next minute. The load prediction is based on three errors. Median error is calculated on percentage of observed values. Higher and Lower errors are calculated based on predicted values.

TABLE 2
Parameters in our simulation

symbol	meaning	value
h	hot threshold	0.9
c	cold threshold	0.25
w	warm threshold	0.65
g	green computing threshold	0.4
l	consolidation limit	0.05



(a) EWMA: $\alpha = 0.7$, $W = 1$

Each dot in figure is an observed value and the curve represents the predicted values.The Curve cuts through the middle of the dots which indicates the accurate prediction.

VI. CONCLUSION

We have employed the resource provisioning in cloud computing in which we have reached the goal of achieving the sustainability and green computing in our system.We have also used the skewness concept to combine the VM's so that all the servers are utilized. And Prediction algorithm to predict the future demands.

VII. REFERENCES

[1]. 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.

[2]. 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.

[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]. 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.

[5]. 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.

[6]. 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.

[7]. 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.

[8]. 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.

[9]. 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.