International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 3 Issue: 5 3266 – 3268

Memory Management in Cloud Storage

Akshatha.A [1] Monisha.V [2] Nischitha.S [3] Shravya .K.N [4] Prof. Shabana Sultana [5]
[1,2,3,4,5] Department of Computer Science & Engineering
The National Institute of Engineering ,Mysore,India
akshathaashok2@gmail.com [1] monisha0609@gmail.com [2] nischitha.n.shankar@gmail.com [3]
shravyakn28@gmail.com [4] shabnamkbn2k@yahoo.co.in [5]

Abstract- Cloud computing allows customers to expand resource based on the data used ,many are multiplexed by virtualization technology .Here the system uses virtualization technique to allocate data used by the users depending on their usage ,also assymetric is also checked of a server[1] .Overall performance of the system is improvised and energy is also saved[2] .Hence by using this algorithm good performance is obtained[3].

Keywords*: VM Scheduler,Physical machine,predictor,hotspot,coldspot*

*****

I. INTRODUCTION

The paper is based on the cloud storage facility, Storage as a service (Saas) and also How we store the data of the user dynamically in cloud.

The user time is saved such that user need not send a request for extra block of memory. The user automatically gets the extended part of memory dynamically depending on the usage of the user. If the user is rarely using the cloud storage, that extra block of memory won't be allocated. The storing of data dynamically is implemented in a efficient way.

Replacement of data is not done which is quite time consuming. Instead of that just extending the existing memory[4]. Only if the user uploads more files at a time in that case the user needs to wait for few moments, as in this case re-allocation of data might take place. This is usually a rare case.

Hence this would help in storing the data efficiently which is done dynamically here.

II. REVIEW

A EXISTING METHOD

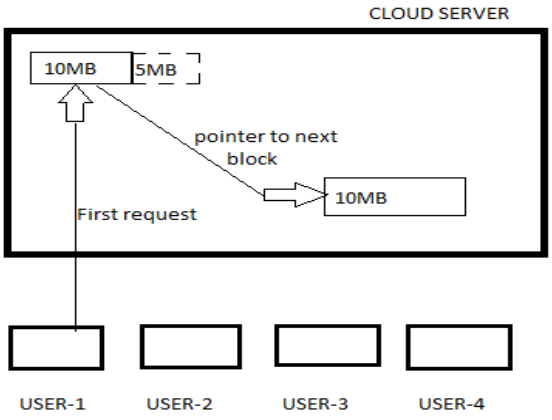


Figure1:Existing System

The user has to be idle till the server allocates the new block of memory. Every time the user has to request for the new block of memory when the existing memory is not sufficient Replacement of data from one block of memory to another is time consuming and also complicated. In this case the data of the user might get lost in the process of replacing.

Disadvantages:

- User has to be idle.
- Every time user has to request.
- Process is time consuming
- Chances of loss of data in the process

B. PROPOSED METHOD

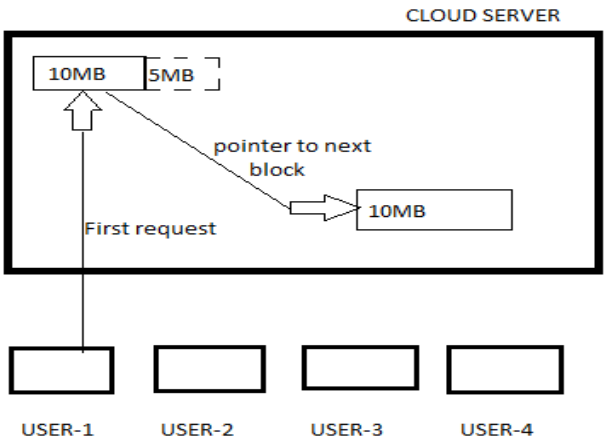


Figure2:Proposed System

The user need not have to send the request to the server whenever the allocated memory is not sufficient. The server automatically allocates the memory when the users usage of memory reaches a threshold.

The user need not wait till the server allocates the memory because dynamically the storage of the data is done[5].

The allocation of new block of memory is not time consuming in our proposed system because the old block of memory is only extended to certain value. Replacement of data into new block of memory at a new location is avoided.

Advantages:

- The user need not have to send the request.
- User time is saved.
- Memory is extended dynamically.

III. DESIGN PROCEDURES AND METHODOLOGY

A. USE- CASE DIAGRAM

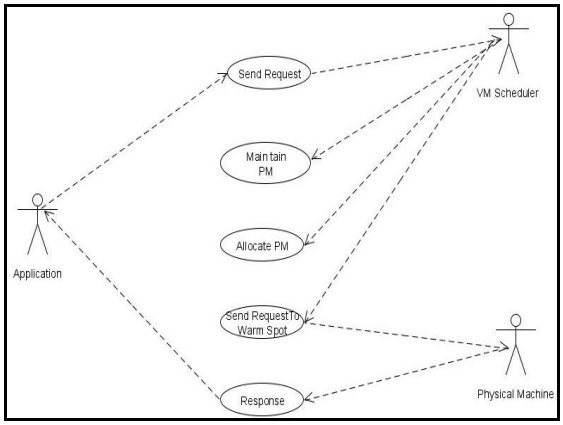


Figure3:Use-case diagrams

A use case diagram is a type of behavioral diagram defined and created from a Use-case analysis in UML. The main aim is to present a graphical overview of the functionality provided by a system in terms of actors, their goals, and all dependencies between those use cases. We use use-case diagram to show what system functions are performed for which actor. Each actors roles in the system present can be seen in use case diagrams.

B. ACTIVITY DIAGRAM

Activity diagrams are representing of workflows of stepwise activities and actions , repetition and concurrency graphically. In Unified Modeling Language(UML), activity diagrams are made to explain the business and working step-by-step flow of components in a system. An activity diagram shows the general flow of control.

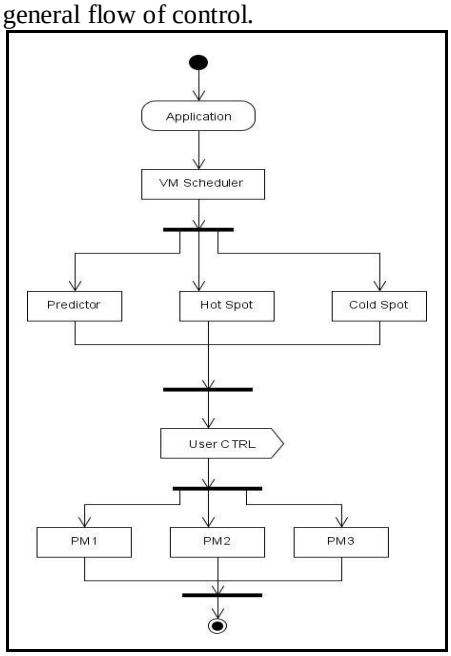


Figure4:Activity Diagram

IV. FUNCTIONAL MODULES

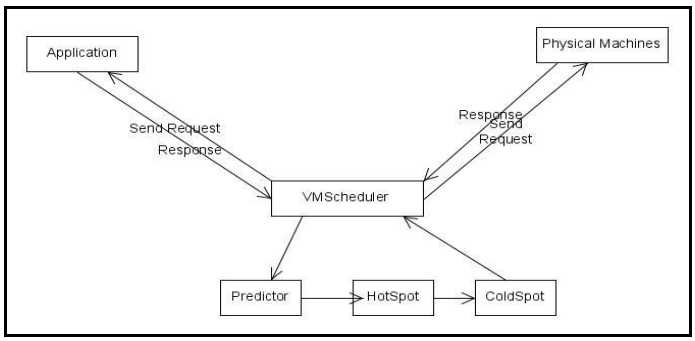


Figure5:Functional modules

A. VM Scheduler :

VM Scheduler run and invoked periodically receives from the user LNM (Local Node Manager)the resource demand history of VMs(virtual machines), the capacity and the load history of PMs(personal machine), and the current layout of VMs on PMs[6]. Then it can forward the request to predictor.

B. Predictor

Predictor predicts the future resource demands of virtual machines and the future requirements of physical machines based on past stats. We estimate the load of a PM by aggregating the resource usage of its VMs. The feature of the load prediction algorithm will be described in the next section. The LNM at each node initially attempts to satisfy the recent demands locally by adjusting the resource allocation of VMs sharing the VMM. Xen can change the CPU allocation between the VMs by adjusting their density in CPU scheduler[7]. The MM Allotter domain 0 of each node is responsible for adjusting the local memory allocation[8].

C . Hotspot Solver :

The hot spot solver in our VM Scheduler detects if the resource utilization of any PM is above the hot threshold (i.e., a hot spot). If so, some VMs running on them will be migrated away to reduce their load. Then it can give the request to coldspot solver.

D. Cold Spot Solver:

The cold spot solver checks if the average utilization of actively used PMs (APMs) is below the green computing threshold. If so, some of those PMs could potentially be turned off to save energy. It identifies the set of PMs whose utilization is below the cold threshold (i.e., cold spots) and then attempts to migrate away all their VMs then it forward request to migration list.

E. Migration List:

When migration list can receive the request from coldspot solver and it can compiles list of VMs and migration list can passes it response to the Usher CTRL (user controller) for execution.

F. Modules Functioning Sequence:

In this system ,the VM scheduler plays the main role ,it assigns the job .As soon as the user reaches the threshold value VM Scheduler informs to predictor. Predictor's is to check the users usage , It will check the memory used and informs hotspot to expand the memory space. The hotspot expands the

users memory space as per usage only when adjacent memory block is not free for expansion of memory ,the coldspot reallocates the memory space. The old pointer which was pointing the old memory space is made to point to new location. Hence dynamically the memory is allocated, all the uploaded files of user are stored in physical Machine which can be viewed by admin.

V. CONCLUSION

The method multiplexes virtual to physical resources adaptively based on the changing demand.

The skewness metric to combine VMs with different resource characteristics appropriately so that the capacities of servers are well utilized. This algorithm achieves both overload avoidance and green computing for systems with multi resource constraints.A management algorithm for dynamic allocation of virtual machines to physical servers is presented. The algorithm pro-actively adapts to demand changes and migrates virtual machines between physical hosts thus providing probabilistic SLA guarantees. Time series forecasting techniques and binpacking heuristic are combined to minimize the number of physical machines required to support a workload. A method for characterizing the gain that a given virtual machine can achieve from dynamic migration is also presented. Experimental studies of the algorithm and its applicability using traces from production data centers are shown.

The algorithm achieves significant reduction in resource consumption (up to50% as compared to the static allocation). This can be extended for research plans that include refining the model to capture the relationships between multiple resources, such as CPU and I/O.

REFERENCES

[1] A. Singh, M. Korupolu, and D. Mohapatra, "Server-Storage Virtualization: Integration and Load Balancing in Data Centers,"Proc. ACM/IEEE Conf. Supercomputing, 2008.

[2] J.S. Chase, D.C. Anderson, P.N. Thakar, A.M. Vahdat, and R.P.Doyle, "Managing Energy and Server Resources in HostingCenters," Proc. ACM Symp. Operating System Principles (SOSP '01),Oct. 2001.

[3] Mr. M.Armbrust "Above the Clouds: A Berkeley Cloud Computing,"technical report, Univ. of California, in Feb.2009.

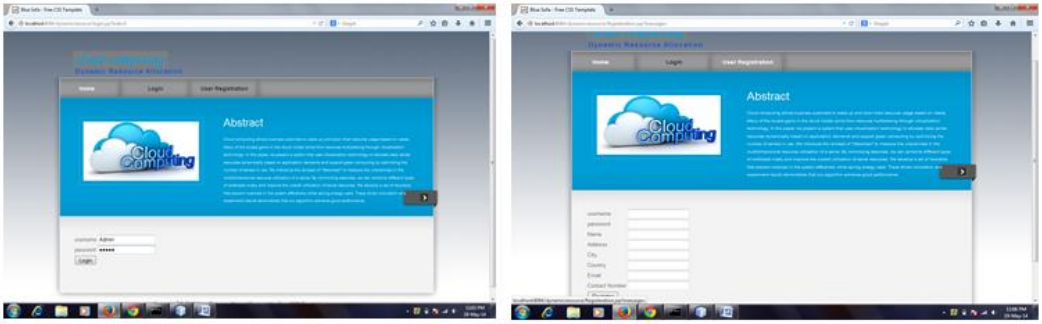
[4] Mr. M. McNett, Mr. D. Gupta, A. Vahdat, and G.M. Voelker, "Usher: An Extensible Framework for Managing Clusters of Virtual Machines,"Proc. Large Installation System Administration Conf.(LISA '07), Nov. 2007.

[5] N. Bobroff, A. Kochut, and Beaty,"Dynamic Placement ofVirtual Machines for Managing SLAViolations," Proc. IFIP/IEEEInt'lSymp. Integrated Network Management (IM '07), 2007.

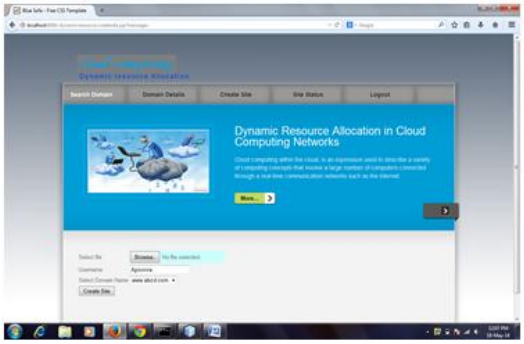
[6] Mr. C.A. Waldspurger, "Memory Resource Management in VMwareESX Server," Proc. Symp. Operating Systems Design and Implementation in Aug. 2002.

[7] 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.

[8] Y. Toyoda, "A Simplified Algorithm for Obtaining ApproximateSolutions to Zero-One Programming Problems," ManagementScience, vol. 21, pp. 1417-1427, Aug. 1975.



Web page(Admin side) Client side details



Domain search

Screenshots