Energy – Optimized Virtual Machine Scheduling Schemes In Cloud Environment

[1]Ghanshyam Parmar, [2]Vimal Pandya

[1]N. V. Patel College of Pure and Applied Sciences, Vallabh Vidyanagar, Gujarat, India.

[2]Shri H. K. Arts College, Ahmedabad, Gujarat, India.

Abstract **—** The Cloud Computing services such as IaaS, SaaS, and PaaS are provided to the consumers are based on the subscription given by the providers. Consumer can acquire their instances as per their requirements like reservation, on-demand or spot or any other methods given by the service providers. Consumers can use networked storage space or computer resources through high-speed internet at their own place. In Cloud computing Virtual Machine is allowing to use large amounts of computing power in para-virtualized, full-virtualized mode where combined resources are used. Cloud computing signifies energy optimization of Virtual Machine resources such as hardware, software, LAN, networked devices. For achieving energy optimization the main objective is reduce energy consumption and improvement in utilization of VMs. In this paper we proposed Priority – Based Energy Optimized Scheduling Scheme called PEOSS, where an auto – scheduler software manages all VMs as per the request of the consumers in reservation or on-demand manner. The auto VM on or off, pause or resume is taken place as per the date and the time duration of the consumer request. So, by making pause or off the VM, when they not required the amount of energy consumption is reduce. For consumer satisfaction before the time VM is started. To manage and test this scheme the cloud environment is developed in VMware Workstation.

Index Terms **–** Virtual Machine, auto-scheduler, energy, power, PBEOSS, VM request.

I. INTRODUCTION

Cloud computing is a computing term which based on utility, consumption and sharing of computer resources. Cloud computing can also be categorized as a new paradigm for the online provisioning of computing software, hardware, data as a service through different pricing models like, reserved or leased, on-demand, spot, pay-as-you-go. Services provided by the Cloud computing is Infrastructure as a Service, Software as a Service, and Platform as a Service. The cloud can be basically deployed as private, public or hybrid. Cloud computing is open source to developers of cloud environment. To develop open source or commercial cloud environment the Virtual Machine Monitor or hypervisors are used.

The main technology that supports cloud computing is virtualization. Virtualization software splits a physical computing device or physical computer into one or more virtual computers, each of which can be work separately. Cloud computing is available as open source and commercial base for cloud developers. Virtualization can be deployed by different hypervisors namely KVM, VMware [11], Xen, Virtual box, Eucalyptus etc. The services can be offered by on the bases of energy, power, pay-per use, on-demand models. [3] The hardware of computer system consumes highest energy. Energy consumption is increases when usage for no of computer resources increases.

For managing Virtual Machines, a scheduler is required as per the requirement selected by consumer provided in the form of reserved or on-demand instances. The job of the scheduler to schedule (ON/OFF) VMs as per the date and time or duration of the request in such a way to achieve greater consumer satisfaction, proper resource utilization, load balancing and saving energy.

In this paper we discuss review of energy optimization schemes, background of our Priority Bases Energy Optimized Scheduling Scheme (PEOSS) which use Auto-Scheduler, detail of algorithmic design with step-by-step process of scheduler and conclusion of this paper. We refer virtual machine as VM and virtual machine request as VM request in this paper.

II. REVIEW OF ENERGY OPTIMIZATION SCHEMES

In this paper we refer many algorithms, methods, approaches, paradigms, techniques, schemes for how to scheduling virtual machines running on physical hardware and also focus on less energy consumption, resource optimization, load balancing, and no misuse of physical machine's resources.

A. Beloglazov et al., [2] proposed a scheduling scheme where the problem of VM allocation can be divided in two: the first part is the admission of new requests for VM provisioning and placing the VMs on hosts, whereas the second part is the optimization of the current VM allocation. The first part can be seen as a bin packing problem with variable bin sizes and prices. For solve that they use to solve it we apply a modification of the Best Fit Decreasing (MBFD) algorithm. The optimization of the current VM allocation is carried out in two steps: at the first step we select VMs that need to be migrated, at the second step the chosen VMs are placed on the hosts using the MBFD algorithm.This algorithm basically minimize the Migration of VMs.

Pinheiro et al., [8]. In this work the authors have proposed a technique for minimization of power consumption in a heterogeneous cluster of computing nodes serving multiple web-applications. The main technique applied to minimize power consumption is concentrating the workload to the minimum of physical nodes and switching idle nodes off. This approach requires dealing with the power or performance trade-off, as performance of applications can be degraded due to the workload consolidation. Requirements to the throughput and execution time of applications are defined in SLAs to ensure reliable QoS.

T. Thanavanich and P. Uthayopas et al., (2013) proposed an energy aware scheduling for the cloud called EHEFT is proposed. This algorithm trying to achieve more energy reduction while maintaining the same performance as much as possible. The approach of EHEFT is to use performance metric called RE to help identify inefficient processor in the system. Then, shutdown these processors and reschedule the task to some other processors. The simulation results show that the proposed method can help reduce the energy consumption without increasing schedule length substantially for many classes of parallel applications. The result of this work can lead to a more energy efficient cloud. In the future, the reduction of time complexity of this algorithm can be explored along with how to make more efficient use of DVS support that is built into the processor. Some extension to maintain the same make span for the task is one of the areas that will be investigated further. [9]

Devare et. al., (2010) implemented the Desktop Cloud system, at the University of Calabria. This system uses the idle resources of the desktops with permission of the owner. The system works on the ―utilization factor and mutual agreement between ―the scheduler strategies, owner and consumer. The various new cloud lease schemes and strategies are under development in Desktop Cloud System. [5]

Li et al., (2011) proposed Hybrid energy efficient scheduling algorithm which is use for private cloud computing. The algorithm which use dynamic migration. The experiment results shows reduce response time, conserve more energy and achieve higher level of load balancing. [7]

A. Beloglazov et al., (2010) proposed the underlying infrastructure is represented by a large-scale Cloud data center comprising n heterogeneous physical nodes. Each node has a CPU, which can be multicore, with performance defined in Millions Instructions Per Second (MIPS). The software system architecture is tiered comprising a dispatcher, global and local managers. The local managers reside on each physical node as a part of a Virtual Machine Monitor (VMM). They are responsible for observing current utilization of the node's resources and its thermal state. The local managers choose VMs that have to be migrated to another node. The local managers send to the global managers the information about the utilization of resources and VMs chosen to migrate. The system operation consists of New requests for VM provisioning, Dispatching requests for VM provisioning, VM migration, VM resizing, VM scheduling. [1]

R. Vijindra et al., (2012) proposed a Ranking Algorithm for virtual machines. By using the ranking algorithm, virtual machines are ranked based on the resources. The algorithm accepts the user request based on the job characteristics, it will goes to the resource broker, and after that it goes to policy prioritizer. The scheduler is responsible for priority policies, match making services of the user request. Ranking algorithm is used to rank the virtual machines (VMs). [10]

III. BACKGROUNDS

To compare and implement Energy Optimization schemes are complex tasks, various virtualization supported tools required, tools capability with para-virtualization, full-virtualization, VM scheduler, load balancing on virtual machines which are provided to consumers.

Basic Architecture of Cloud Computing:

In basic architecture of cloud computing, the virtualization is done on the actual or physical machine is called host machine, and the guest machine is called the virtual machine. For differentiate the physical and virtual machine the terms host and guest is used. To create virtual machine on host or physical machine the Virtual Machine Monitor is used which is a software or firmware. [3]

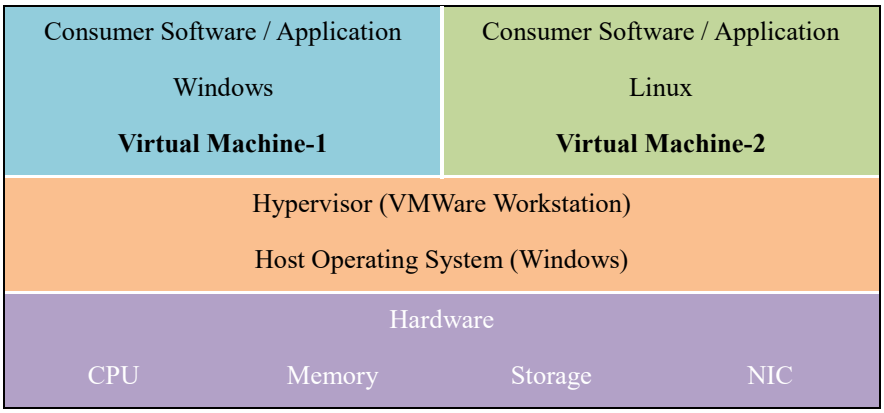

Fig. 1. Basic Architecture of Cloud Computing

In our domain VMware Workstation is a hypervisor which schedules number of virtual machines. An auto-scheduler application is also design which schedules virtual machines. The hypervisor has capability migrate, clone, pause, resume virtual machines on physical machine through auto-scheduler using Application Programming Interface (API). The virtual machine request of consumer is in the form of reserved and on-demand instances which provides by clouds provider on web. The consumer request contains the type of instance (reserved / on-demand), configuration of virtual machines, for reserved instance subscription type (in months / years) and time duration to utilize VM, for on-demand instance date and time duration.

VMware Workstation:

VMware Workstation is a hypervisor that runs on Intel 32- or 64-bit computers. It enables users to set up one or more virtual machines (VMs) on a single physical machine, and use them concurrently along with the physical machine. Each virtual machine can execute its own operating system, including versions of Microsoft Windows, Linux, BSD, and MS-DOS. [13]

VMware Workstation supports bridging existing host network adapters and share physical disk drives and USB devices with a virtual machine. In addition, it can simulate disk drives. It can mount an existing ISO image file into a virtual optical disc drive so that the virtual machine sees it as a real one. Likewise, virtual hard disk drives are made via .vmdk files. [13]

The VIX API of VMware gives you the rich set of functions to manage VM operations like power on / off, clone, pause / resume on VMware Workstation. In VIX API the set of functions are given into a library which contains scripts and programs to automate virtual machines. These scripts and programs are in high-level languages so it is, easy to use, easy to programming for script developers and application programmers. The VIX API is designed for three kinds of users: Technically Adventurous Users, Partners, and VMware Engineering. [12]

The VIX API supported operating systems are manly Microsoft Windows and Linux platforms. In Windows and Linux the version supported are Windows-95 or later and 2.4.x kernel or later. The support of programming languages are C, Perl, and COM based langaues like VBscript, and C#. [4] For VMware Workstation 10 VIX API version 1.13 is supported.

IV. PRIORITY BASED ENERGY OPTIMIZED SCHEDULING SCHEME (PEOSS):

In this significance we introduce Priority based Energy Optimized Scheduling Scheme PEOSS with Virtual Machine (VM) power ON / OFF, clone, pause, resume based on the consumer request coming as First Come First Serve (FCFS). Here, there are two modes are given to the consumer for instance purchasing reserved and on-demand. In the reserved instance the VM may occupied not for one day and some hours of time, but for long period of days with specific time. Other than in on-demand instance the VM is occupied at specific date and stipulated time. So, here priority-based means each request of consumer is given priority, so no one must wait for VM. The consumer request is coming in FCFS manner but assigning VM to the request is based on the priority and therefore if no VM is free available than clone process of new VM is proceed. Here, working of auto-scheduler application taken place, which manages the VM request. Consumer provide request in the form of advanced reservation or on-demand using Request Management System (RMS) which collect these requests in queue, VMs are also in queue, PEOSS schedules request to VM, implementation is as shown in below Fig. 2. working procedure of PEOSS.

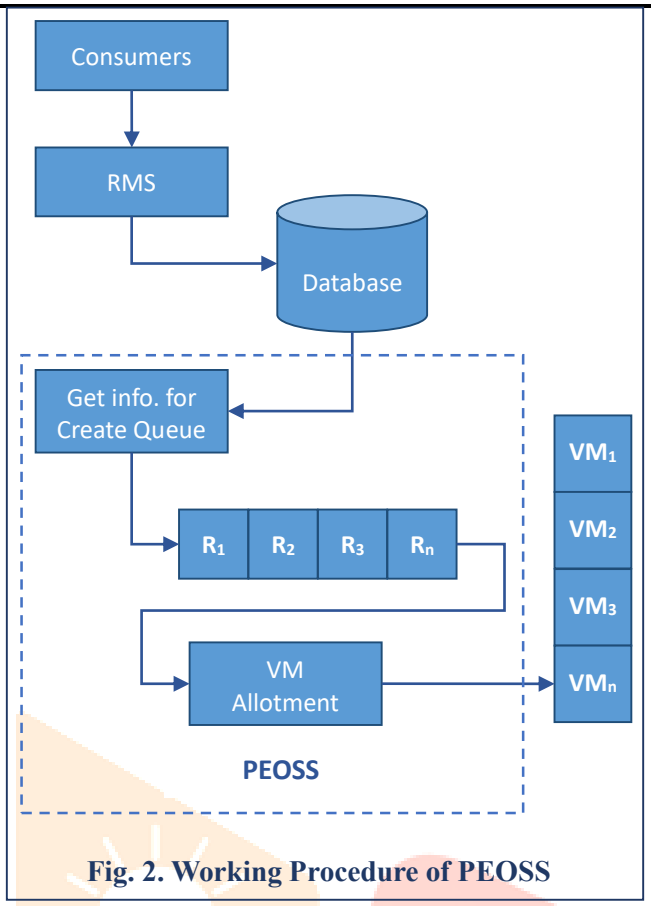

Fig. 2. Working Procedure of PEOSS

Features of PEOSS

- Use Power ON / OFF, pause, resume, and clone of VMs as per the consumer requests.
- Scheduling the VMs as per the time and duration given into the request.
- Utilization of less VMs by power ON / OFF.
- Auto check for availability of VM for use than consider pause or power OFF.
- The VM is not used by consumer and that is no longer to be used than VM is power OFF otherwise pause.
- All VMs are allocated than clone of that type of VM is created and assigned.

States of Virtual machine:

Virtual machine is running on different modes, we say state of virtual machine categories in different states and use these states in our PEOSS see fallowing states.

Running: virtual machine is ON and waits for Job request from consumer.

Active: Virtual machine is running and executes job.

PowerON: Virtual machine is just powering on state.

PowerOFF: Virtual machine is shutting down state.

Pause: Virtual machine is running state to pause state.

Resume: Virtual machine is pause state to resume (running) state.

Parameters used in PEOSS:

VM_I – Set of total Virtual Machine (Guest) in a Host and I = 1, 2, 3, 4,…, n.
R – Incoming requests from consumer for VM as PaaS, R_1, R_2, R_3, R_4, … R_n.
R_R – Request for advanced reservation.
R_O – Request for On-demand.
C_R – Request of Virtual Machine requires Clone.
L_R – List of request of current date.
R_{ST} – Time for PowerOn VM for current request
R_{ST} + 1 – Time for PowerOn VM for next request

R_{ET} – Time for PowerOff VM

C_T – Current time of Host

D_T – Difference of R_{ET} and R_{ST} + 1

Functions used in PEOSS:

PowerOn( ): Virtual Machine starts as per the consumer request and time.

PowerOff( ): Virtual Machine is shutdown when in no longer use for saving energy.

Pause( ): Pause the running VM for saving energy.

IsPause( ): Check whether the VM is in pause state and return True or False.

Unpause( ): Resume the pause VM for saving energy.

Clone( ): Cloning of one VM to another VM on same Host.

Pseudo code for PEOSS:

Steps of Algorithm:

// Set default values of C_R, L_R, R_{ST}, R_{ET}, C_T, and D_T.

1. Start
2. Create L_R for current date
3. if (R_I == R_R) then
   Check for availability of VM for given time duration
   If available then update C_R = Yes
   else C_R = No
   end if
   end if
4. if (R_I == R_O) then
   Check availability of VM in L_R
   If available then update C_R = Yes
   else call Clone( ) function
   end if
   end if
5. if (R_{ST} == C_T) then
   Call function IsPause( )
   if True then UnPause( ) VM
   else PowerOn( ) VM
   end if
   end if
6. if (D_T < 15 minutes) then Pause( ) VM
   else PowerOff( ) VM
   end if
7. Exit( )

V. CONCLUSION

In Platform as a Service (PaaS), resource (VMs) allocation and scheduling is a key issue in the cloud computing environment. In this paper we propose an Energy Optimized Scheduling Scheme for the cloud environment, where consumer requires a service, PaaS, where each consumer had its own VM. The challenging part is how to manage VMs so, less energy can be consumed. For less energy consumption, means less power consumption. Our Energy Optimization scheme PEOSS manages the VMs as per the date and time or time duration. PEOSS provides auto-scheduler, which automatically manage VMs by PowerOn, PowerOff, Pause, Unpause, Clone as per the consumer request. It will reduce energy consumption by using Pause, Unpause where PowerOff then On function take more energy. When use of VMs is completed for stipulated time and not used for longer time then it will PowerOff otherwise it will be Pause. PEOSS is always beneficial for service providers who provides Platform as a Service.

References

[1] Beloglazov, A. and R. Buyya. "Energy Efficient Resource Management in Virtualized Cloud Data Centers." *International Conference on Cluster, Cloud and Grid Computing*. 2010.

[2] Beloglazov, Anton, Jemal Abawajy and Rajkumar Buyya. "Energy-aware resource allocation heuristics for efficient management of data centers for Cloud computing." *Future Generation Computer Systems* (2012): 755-768.

[3] Buyya, Rajkumar, James Broberg and Andrzej Goscinski. *CLOUD COMPUTING: Principles and Paradigms*. A Jhon Wiley & Sons, Inc. Publication, 2011.

[4] Deshwal , A. and A. Dobhal. "Virtualization: A Tool to a Sustainable Education System." *International Journal of Advanced Research in Computer Science and Software Engineering* III.3 (2013): 539-544.

[5] Devare, M., M. Sheikhalishahi and Grandinetti Lucio. "A Prototype Implementation of Desktop Clouds." *International workshop on HPC, Grids and Clouds*. Cetraro, Netherland, 2010.

[6] Kumar, P., D. Singh and A. Kaushik. "Power and Data Aware Best Fit Algorithm for Energy Saving in Cloud Computing." *International Journal of Computer Science and Information Technologies* V.5 (2014): 6712-6715.

[7] Li, Jiandun, Junjie Peng and Wu Zhang. "A Scheduling Algorithm for Private Clouds." *Journal of Convergence Information Technology* VI.7 (2011): 1-9.

[8] Pinheiro, E., et al. "Load balancing and unbalancing for power and performancee in cluster-based systems." *Workshop on Compilers and Operating Systems for Low Power*. 2001. 182–195.

[9] Thanavanich , T. and P. Uthayopas. "Enhancing Energy-Aware Scheduling for Parallel Applications on a Cloud." *Journal of Communications and Information Sciences* III.4 (2013).

[10] Vijindra, R. "Energy Efficient Scheduling Framework for Cloud Computing using Ranking Algorithm." *International Journal of Scientific & Engineering Research* III.10 (2012).

[11] VMware. "Virtualization for Desktop & Server, Application." n.d. 15 May 2015. <http://www.vmware.com/in>.

[12] VMware. "VIX API." n.d. 16 May 2015. <https://www.vmware.com/support/developer/vix-api/vix16_reference/intro.html>.

[13] VMware. "VMware Workstation." n.d. 16 May 2015. <http://en.wikipedia.org/wiki/VMware_Workstation>.