© 2014 IJEDR | Volume 2, Issue 3 | ISSN: 2321-9939

Scheduling Algorithm Based Virtual Machine in Cloud Computing

[1]Deepti Gupta, [2]Amit Chaudhary
[1]Research Scholar, [2]Assistant Professor
MIET, Computer Science & Engineering, Kurukshetra University

***Abstract* - Cloud computing has come out to be an interesting and beneficial way of changing the whole computing world. The traditional way for task scheduling cannot meet the cloud market well enough. This paper introduces an optimized algorithm for cost based scheduling in cloud computing and its implementation. This scheduling algorithm measures both resource cost and calculation performance, it also improves the calculation/communication ratio by grouping the user tasks according to a particular cloud resource's processing capability and sends the grouped jobs to the resource.**

***Keywords* - Cloud Computing, Virtual machine Scheduling, Cloud SIM**

I. INTRODUCTION

Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the data canters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS). The data canter hardware and software is what we will call a Cloud. The term Private Cloud is used to refer to internal data canters of a business or other organization not made available to the general public. Thus, Cloud Computing is the sum of SaaS and Utility Computing, but does not include Private Clouds. People can be users or providers of SaaS, or users or providers of Utility Computing. We focus on SaaS which have received less attention than SaaS Users. Fig.1 shows the roles of the people as users or providers of these layers of Cloud Computing. Cloud computing is an inclusive one: cloud computing can describe services being provided at any of the traditional layers from hardware to applications (Fig 1). In practice, cloud service providers tend to offer services that can be grouped into three categories: software as a service, platform as a service, and infrastructure as a service. These categories group together the various layers.

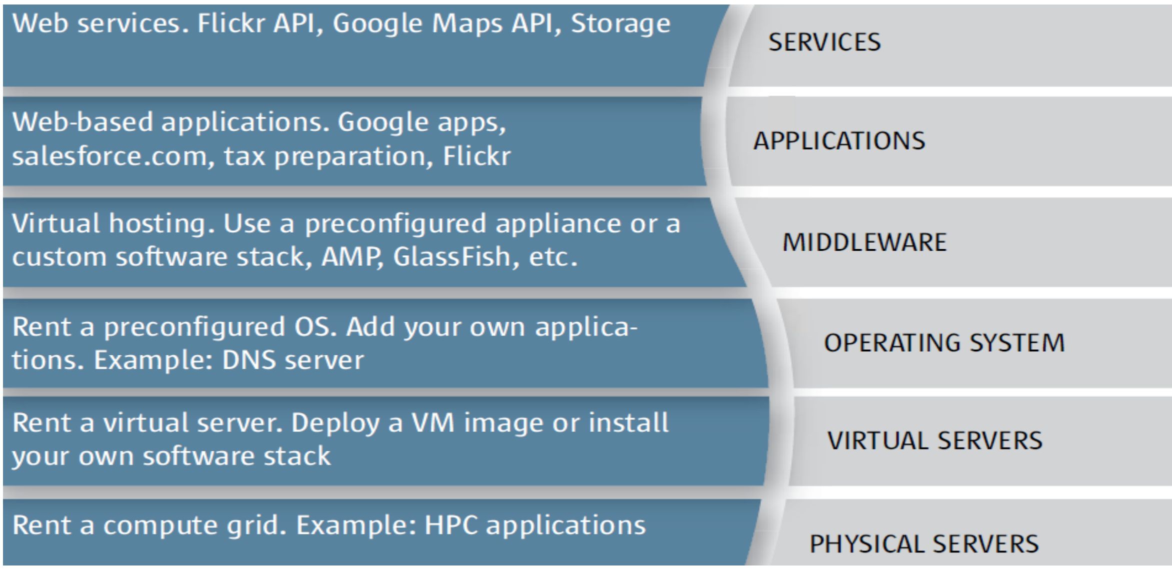


Fig 1 Cloud architecture

Cloud computing is a new technique in the Information Technology circle. It is an extension of parallel computing, distributed computing and grid computing. It provides secure, quick, convenient data storage and net computing service cantered by internet. Cloud computing delivers three kinds of services : Infrastructure as a service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). These services are available to users in a Payper- Use-On-Demand model, Virtualization, distribution and dynamic extendibility are the basic characteristics of cloud computing. Virtualization is the main character .

Most software and hardware have provided support to virtualization. We can virtualize many factors such as IT resource, software, hardware, operating system and net storage, and manage them in the cloud computing platform; every environment has nothing to do with the physical platform.

II. VIRTUAL MACHINE SCHEDULING

Cloud computing technologies could never exist without the use of the underlying technology known as Virtualization. It allows abstraction and isolation of lower level functionalities and underlying hardware. This enables portability of higher level functions and sharing and/or aggregation of the physical resources .Cloud computing heavily relies on virtualization as it virtualizes many aspects of the computer including software, memory, storage, data and networks. Virtualization is known to enables you to consolidate your servers and do more with less hardware. It also lets you support more users per piece of hardware, deliver applications, and run applications faster. These attributes that virtualization hold are the core of cloud computing technologies and is what makes it possible for cloud computing's key characteristics of multitenancy, massive scalability, rapid elasticity and measured service to exist.

The standard deployment unit is a virtual machine, which by its very nature is designed to run on an abstract hardware platform. It's easy to over focus on building virtual machine images and forget about the model that was used to create them. In cloud computing, it's important to maintain the model, not the image itself.

Virtual machine images will always change because the layers of software within them will always need to be patched, upgraded, or reconfigured. What doesn't change is the process of creating the virtual machine image, and this is what developers should focus on. A developer might build a virtual machine image by layering a Web server, application server, and MySQL database server onto an operating system image, applying patches, configuration changes, and interconnecting components at each layer. Focusing on the model, rather than the virtual machine image, allows the images themselves to be updated as needed by re-applying the model to a new set of components. With this standard deployment unit, cloud architects can use appliances that help to speed deployment with lower costs. A developer might use an appliance that is preconfigured to run Hadoop on the OpenSolaris OS by interacting with the appliance's API. Architects can use content switches that are deployed not as physical devices, but as virtual appliances. All that needs to be done to deploy it is interact with its API or GUI.

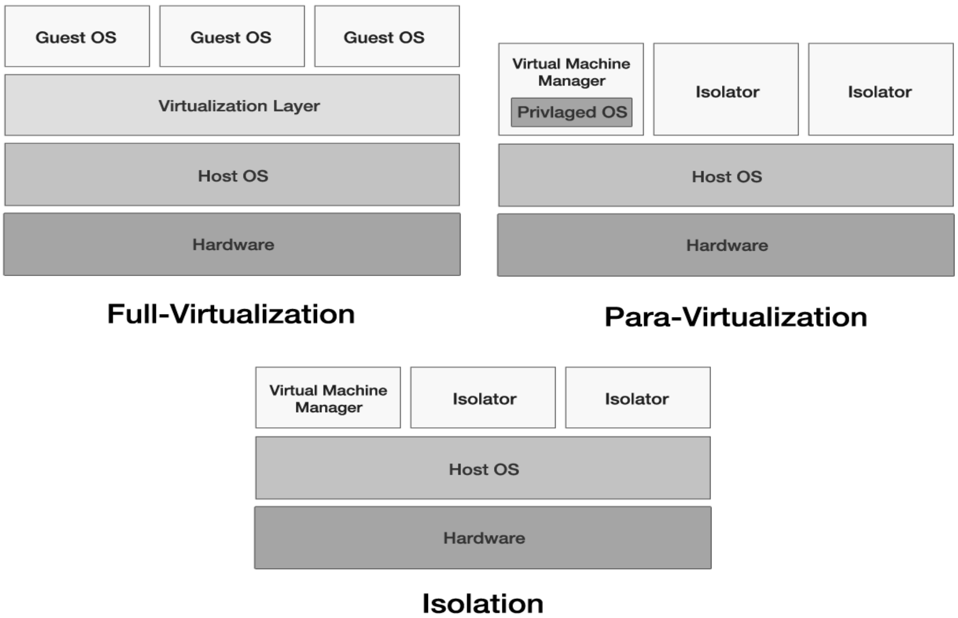


Fig 2 Three forms of virtualization

Full Virtualization

This type of virtualization operates at the processor level, which supports unmodified guest operating systems that simulate the hardware and software of the host machine.

Para-virtualization

Utilizes the use of a virtual machine monitor, which is software that allows a single physical machine to support multiple virtual machines. It allows multiple virtual machines to run on one host and each instance of a guest program is executed independently on their own virtual machine.

Isolation

Is similar to Para virtualization although it only allows virtualization of the same operating system as the host and only supports Linux systems but it is considered to perform the best and operate the most efficiently. VM image management, and advanced data center design. Within the framework, there are two major areas which can lead to improvements. First, we can expand upon the baseline functioning of virtual machines in a cloud environment. This is first done with deriving a more efficient scheduling system for VMs. The Scheduling section addresses the placement of VMs within the Cloud infrastructure while minimizing the operating costs of the Cloud itself. This is typically achieved by optimising either power of the server equipment itself or the overall temperature within the data centres. Due to the inherent disposability and mobility of VMs within a semi homogeneous data canter, we can leverage the ability to move and manage the VMs to further improve efficiency. The image management section attempts to control and manipulate the size and placement of VM images in various ways to conserve power and remove unnecessary bloat. Furthermore, the design of the virtual machine images can also lead to a drastic power savings

III. DESIGN AND IMPLEMENTATION OF CLOUDSIM

The Class design diagram for the simulator is depicted in Fig 3. In this section, we provide finer details related to the fundamental classes of CloudSim, which are building blocks of the simulator. BWProvisioner. This is an abstract class that models the provisioning policy of bandwidth to VMs that are deployed on a Host component. The function of this component is to undertake the allocation of network bandwidths to set of competing VMs deployed across the data center. Cloud system developers and researchers can extend this class with their own policies(priority, QoS) to reflect the needs of their applications. The SimpleBWProvisioning allows a VM to reserve as much bandwidth as required, if the host has enough bandwidth not allocated to other VMs.

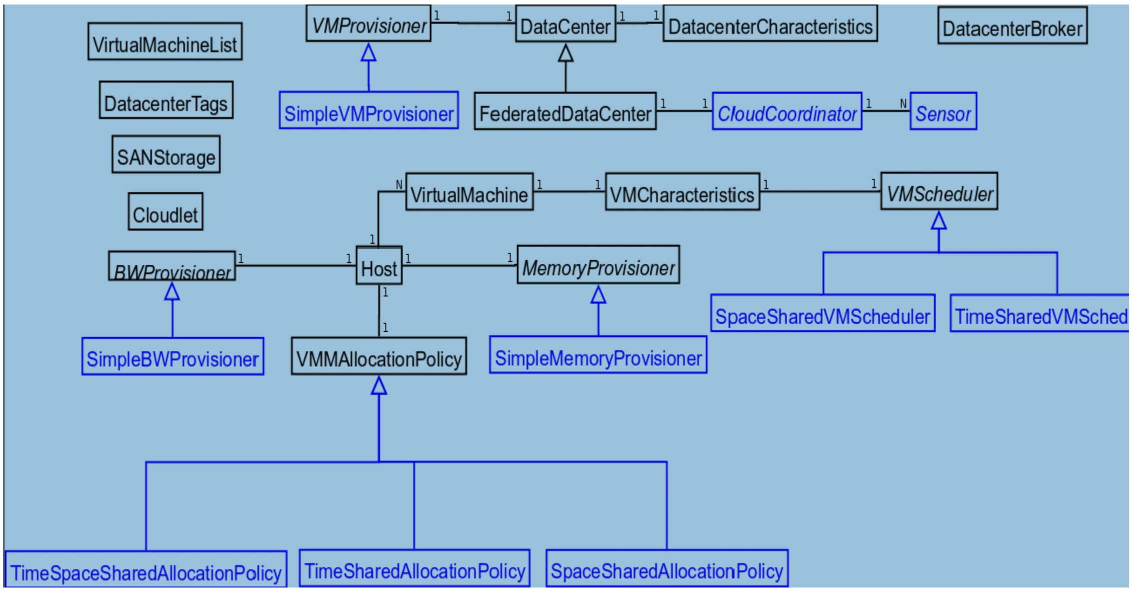


Fig 3 Cloud Sim

CloudCoordinator

This abstract class provides federation capacity to a data center. It is able to periodically monitor the internal data center status, and based on that it is able to decide whether part of the data center load can be migrated or not.Concrete implementation of this object contains the specific sensors and the policy to apply during load transferring.Monitoring is performed periodically by the updateDatacenter() method that queries to different Sensors, one for each monitored feature. Discovery is implemented in the setDatacenter()abstract method, which allows for CloudSim users to determine the way discovery should performed in the specific implemented scenario. This component can also be used to implement services such as Amazon's Load-Balancer. Developer aiming for deploying their application services across multiple clouds can extend this class for implementing their custom resource provisioning policies.

Cloudlet

This class models the Cloud-based application services (content delivery, social networking, business workflow),which are commonly deployed in the data centers. CloudSim represents the complexity of an application in terms of its computational requirements. Every application component has a pre-assigned instrction length (inherited from GridSim's Gridlet component) and amount of data transfer (both pre and post fetches) that needs to be undertaken for successfully hosting the application. This class can also be extended to support modeling of other metrics for application execution, such transactions in database-oriented applications

DataCenter

This class models the core infrastructure level services (hardware, software) offered by service provider(Amazon, Azure, App Engine) in a Cloud computing environment. It encapsulates a set of compute hosts that can be either homogeneous or heterogeneous as regards to their resource configurations (memory, cores, capacity, and storage). Furthermore, every DataCenter component instantiates a generalized application provisioning component that implements a set of policies for allocating bandwidth, memory, and storage devices.

DatacenterBroker

This class models a broker, which is responsible for mediating between users and service providers depending on users' QoS requirements and deploys service tasks across Clouds. The broker acting on behalf of users. It identifies suitable Cloud service providers through the Cloud Information Service (CIS) and negotiates with them for an allocation of resources that meets QoS needs of users. The researchers and system developers must extend this class for conducting experiments with their custom developed application placement policies. The difference between this component and the CloudCoordinator is that the former represents the customer (i.e, decisions of this components are made in order to increase user-related performance metrics), while the former acts on behalf of the data center, i.e, it tries to maximize performance from the data center point of view, without considering needs of specific computer.

IV. RELATED WORK

Carlin, Curran et al[1] It is clear that cloud computing is here to stay for the foreseeable future, as the topic has had buzz around it for years now and it is finally being adopted by many with more to follow. The key concepts, terminologies and underlying technologies of cloud computing that were outlined should clarify and aid in the understanding of this complex topic. he cloud computing stack containing the three essential services (SaaS, PaaS and IaaS) that have come to define the technology and its delivery model. The underlying virtualization technologies that make cloud computing possible are also identified and explained. The various challenges that face cloud computing technologies today are investigated and discussed. The future of cloud computing technologies along with its various applications and trends are also explored, giving a brief outlook of where and how the technology will progress into the future. The future of cloud computing is not definite but by analyzing the trends it seems that cloud technology will play a large part in our day to day lives. In the future business and consumers will benefit from higher interoperability between clouds and maybe even a cloud network which will improve sharing of resources and information. There are many uses for this technology and it is surely going to change the way in which we handle our data, services and access/store our digital content but for its full potential to be unlocked a broader understanding, appreciation and investment in cloud computing technologies is required.

Buyya, Ranjan et a[2]lCloud applications have different composition, configuration, and deployment requirements. Quantifying the performance of resource allocation policies and application scheduling algorithms at finer details in Cloud computing environments for different application and service models under varying load, energy performance (power consumption, heat dissipation), and system size is a challenging problem to tackle. The CloudSim toolkit supports modelling and creation of one or more virtual machines(VMs) on a simulated node of a Data Center, jobs, and their mapping to suitable VMs. Grids [5] have evolved as the infrastructure for delivering high-performance services for compute and data-intensive scientific applications. To research and development of new Grid, policies, and middleware; several Grid simulators, such as GridSim [9], SimGrid [7], and GangSim [4] have been proposed. SimGrid is a generic framework for simulation of distributed applications on Grid platforms. Similarly, GangSim is a Grid simulation toolkit that provides support for modeling of Grid-based virtual organisations and resources. On the other hand, GridSim is an event-driven simulation toolkit for heterogeneous Grid resources. newly developed methods and policies,researchers need tools that allow them to evaluate the11 hypothesis prior to real deployment in an environment one can reproduce tests. Simulation-based approaches in evaluating Cloud computing systems and application behaviors offer significant benefits, as they allow Cloud developers: (i) to test performance of their provisioning and service delivery policies in a repeatable and controllable environment free of cost; and (ii) to tune the performance bottlenecks before real-world deployment on commercial Clouds. To meet these requirements, we developed the CloudSim toolkit for modeling and simulation of extensible Clouds. As a completely customizable tool, it allows extension and definition of policies in all the components of the software stack, which makes it suitable as a research tool that can handle the complexities arising from simulated environments. As future work, we are planning to incorporate new pricing and provisioning policies to CloudSim, in order to offer a built-in support to simulate the currently available Clouds. Modeling and simulation of such environments that consist of providers encompassing multiple services and routing boundaries present unique challenges.

Wang et al[3] we described what is cloud computing and took Google's cloud computing techniques as an example, summed up key techniques, such as data storage technology (Google File System), data management technology (BigTable), as well as programming model and task scheduling model (Map Reduce), used in cloud computing, and then some example of cloud computing. Cloud Computing refers to both the applications delivered as services over the Internet and the hardware and systems software in the datacenters that provide those services. The services themselves have long been referred to as Software as a Service (SaaS). The datacenter hardware and software is what we will call a Cloud. When a Cloud is made available in a payas-you-go manner to the general public, which called a Public Cloud; the service being sold is Utility Computing. The term Private Cloud is used to refer to internal data centers of a business or other organization not made available to the general public. Cloud computing is a new technology widely studied in recent years. Now there are many cloud platforms both in industry and in academic circle. How to understand and use these platforms is a big issue. In this paper, we not also described the definition, styles and characters of cloud computing, but also took Google's cloud computing techniques as an example, summed up key techniques, and then some example of cloud computing vendors were illustrated and compared. Though each cloud computing platform has its own strength, one thing should be noticed is that no matter what kind of platform there is lots unsolved issues.

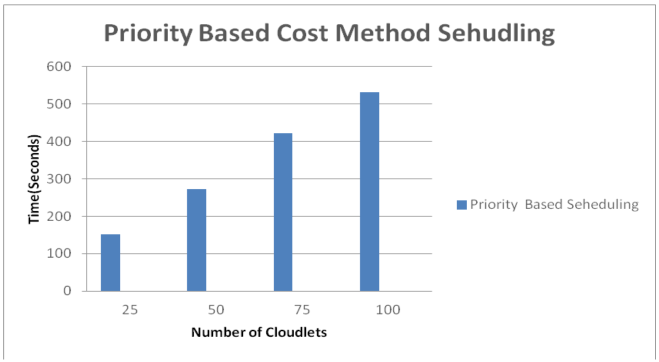
V. PURPOSED WORK

In order to measure direct costs of applications, every individual use of resources (like CPU cost, memory cost, I/O cost, etc.) must be measured. When the direct data of each individual resources cost has been measured, more accurate cost and profit analysis based on it than those of the traditional way can be got.Cost of every individual resources use is different. The priority level can be sorted by the ratio of task's cost to its profit. For easy management, three lists can be built for the sorted task, each list has a label of priority level such as HIGH, MID and LOW. Cloud systems can take someone out from the highest priority list to compute. Maps should be scanned every turn to modify the priority level of each task. Some restrictive conditions like maximum time user can wait should to be measured as extra factors.

The CloudSim simulation layer provides support for modeling and simulation of virtualized Cloud-based data center environments including dedicated management interfaces for virtual machines (VMs), memory, storage, and bandwidth. The fundamental issues such as provisioning of hosts to VMs, managing application execution, and monitoring dynamic system state are handled by this layer. A Cloud provider, who wants to study the efficiency of different policies in allocating its hosts to VMs (VM provisioning), would need to implement their strategies at this layer. Such implementation can be done by programmatically extending the core VM provisioning functionality. There is a clear distinction at this layer related to provisioning of hosts to VMs. A Cloud host can be concurrently allocated to a set of VMs that execute applications based on SaaS provider's defined QoS levels. This layer also exposes functionalities that a Cloud application developer can extend to perform complex workload profiling and application performance study. The top-most layer in the CloudSim stack is the User Code that exposes basic entities for hosts (number of machines, their specification and so on), applications (number of tasks and their requirements), VMs, number of users and their application types, and broker scheduling policies. By extending the basic entities given at this layer, a Cloud application developer can perform following activities: (i) generate a mix of workload request distributions, application configurations; (ii) model Cloud availability 7 scenarios and perform robust tests based on the custom configurations; and (iii) implement custom application provisioning techniques for clouds and their federation CloudSim has been used to create the simulation environment. The inputs to the simulations are total number of tasks, average MI of tasks, MI deviation percentage, granularity size and task overhead time. The MIPS of each resource is specified in Table 1.

Table 1 Mips of cloud resources

Resource	MIPS
R1	120
R2	131
R3	153
R4	296
R5	126
R6	210



VI. CONCLUSION

Profit Based Task Scheduling is a way of measuring both the cost of the objects and the performances of activities and it can measure the cost more accurate than traditional ones in cloud computing. This pa introduces an optimized algorithm for task scheduling based on Priority Based Scheduling in cloud computing and the implementation of it. Compared with the traditional way of task scheduling, Profit based Task Scheduling method has its own advantages.

REFERENCES

[1] Xu Wang, Beizhan Wang, Jing Huang," Cloud computing and its key techniques", ©2011 IEEE,pp 404-410

[2] Rajkumar Buyya1, Rajiv Ranjan2 and Rodrigo N. Calheiros," Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities", @csse.unimelb.edu.au, rajiv@unsw.edu.au, pp 1-11

[3] Sean Carlin, Kevin Curran," Cloud Computing Technologies",@ June 2012, International Journal of Cloud Computing and Services Science (IJ-CLOSER), Vol.1, No.2, ISSN: 2089-3337,http://iaesjournal.com/online/index.php/ IJ-CLOSER,pp 59-65

[4] Paul Martinaitis, Craig Patten and Andrew Wendelborn "Remote Interaction and Scheduling Aspects of Cloud Based Streams"2009 IEEE.

[5] Enda Barrett, Enda Howley, Jim Duggan" A Learning Architecture for Scheduling Workflow Applications in the Cloud" 2011 Ninth IEEE European Conference on Web Services978-0-7695-4536-3/11 $26.00 © 2011 IEEE.

[6] Boonyarith Saovapakhiran, George Michailidis†, Michael Devetsikiotis" Aggregated-DAG Scheduling for Job Flow Maximization in Heterogeneous Cloud Computing"2011 IEEE 978-1-4244-9268-8/11.

[7] Hsu Mon Kyi, Thinn Thu Naing" AN EFFICIENT APPROACH FOR VIRTUAL MACHINES SCHEDULING ON A PRIVATE CLOUD ENVIRONMENT"2011 IEEE 978-1-61284-159-5/11.

[8] V. Nelson, V. Uma" Semantic based Resource Provisioning and Scheduling in Inter-cloud Environment"2012 IEEE978-1-4673-1601-9/12 .

[9] Jinhua Hu, Jianhua Gu, Guofei Sun Tianhai Zhao "A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment"2010 IEEE 978-0-7695-4312-3/10.

[10] Shu-Ching, Wang, Kuo-Qin, Yan, Shun-Sheng, Wang, Ching-Wei, Chen" A Three-Phases Scheduling in a Hierarchical Cloud Computing Network" 2011 Third International Conference on Communications and Mobile Computing 978-0-7695-4357-4/11