Tune the Cloud Environment: Tune Provisioning & Migration

Deepa Verma  
M. Tech. Scholar (CSE)  
ITM Bhilwara, Rajasthan, India

R. K. Somani  
Associate Professor & Head (CSE)  
ITM Bhilwara, Rajasthan, India

Abstract—The Indian industry and network of data is being motivated towards cloud environment for storing their information for their future applications. Many researchers have been motivated to do research in this area for enhancing their applications from time to time. Moving with the current we analyzed the current process of resource provisioning in cloud and identify the scope for enhancing the existing mechanism. The problem identifies the need to tune the resource provisioning process in cloud environment according to user’s demands. The nearest possible solution is proposed to tune the provisioning and migration of resources in regards to virtual machine called tuning algorithm. Tuning algorithm is simulated in cloud environment using CloudSim3.0.2 framework including workflowsim for optimal results. The simulation results shows the tuned resource provisioning process and highlight that the proposal is well formed and performs the resource allocation and VM migration as desired by the user according to the qos parameter.

Keywords—Cloud Computing; resource provisioning; host; virtual machine migration; tune.

I. INTRODUCTION

A. Basic

Cloud Computing is the name specified to the recent trend in computing service provision. This trend has visualized the technological and cultural shift of computing service provision from being available locally towards being remotely provisioned and used by masses with the services being provided by third-party service providers. These third-parties offer consumers an affordable and efficient computing service that consumers would otherwise not have been possibly accessible for a single user or client [4].

The evolution of the cloud started phase by phase evolving from Grid computing, Distributed computing and so on. Cloud computing was used first in 1950s, the time during which large-scale mainframes were available in the business industry. The hardware used by the mainframe was installed in a big room and all users were accessing the mainframe through terminals. [1]

Later in the year 1970, the IBM launched OS having a number of virtual machines at a single machine. The Virtual machine OS has taken the application of 1950s, that is of sharing the access to a mainframe to a higher level by considering a number of virtual machines by providing different accessible machines at a single physical machine. Idea of cloud computing was first introduced by J.C.R Licklider and John McCarthy in 1969. The vision behind this was the need for everyone to be interconnected and be able to access data from anywhere. [1]

B. Definition

There are numerous definitions and elucidations of cloud computing furnished from multiple sources. The term “cloud computing” comes from network diagrams in which cloud shape are used to describe certain types of networks, either the Internet or internal networks. Some sources refer to cloud computing as “A set of applications delivered as services combined with the datacenter hardware and software that enables the applications”. Some others quote cloud computing as a “business model” rather than a precise technology or service. [3]
In broader terms, cloud computing consists of both technological and business components. Certain cloud-enabling technologies notably help to form the cloud, and it is implausible that cloud computing could have existed without them. But the cornerstones of cloud infrastructure such as open-source software, virtualization, distributed storage, distributed databases and monitoring systems are also worth mentioning. [3]

Cloud computing presumes that every system component or software application becomes a service or part of a service. So in order to become cloud compatible, the architecture of new or existing systems might have to be changed. Therefore in order to realize the significance of the cloud and facilitate it for an organization, businesses must typically make appreciable structural adjustments to internal IT organizations and communicate cloud philosophy to employees. Depending on the type of cloud used by an organization, this may also create competition within the company. It is typical that people refuse to accept change, so cloud evangelists often face resistance within their organizations. [3]

Due to the growing popularity there are many proposed definitions of Cloud computing defining its characteristics. Some of the definitions given by many renowned scientists and Prominent organizations are: [3]

a. Rajkumar Buyya defines Cloud computing in terms of its utility to end user as “A Cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreements established through negotiation between the service provider and consumers”.

b. National Institute of Standards and Technology (NIST) defines Cloud computing as follows: “Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This Cloud model promotes availability and is composed of essential characteristics, three service models and four deployment models.

![Fig. 2 Cloud Computing – [10]](image)

- Cloud computing is also defined as “A style of computing where IT-enabled capabilities are delivered as a service to end users using internet”.

![Fig. 3 Service available for a cloud consumer [9]](image)
d. Barkley RAD defines Cloud Computing as: “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 pay-as-you-go manner to the general public, we call it a Public Cloud; the service being sold is Utility Computing. We use the term Private Cloud to refer to internal datacenters 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.” [11]

C. Differences in grid, cluster, cloud

There has been Computational grids are traditionally used for large computational jobs and built with many servers up-front whereas the advantage of cloud is that it can be scaled on-demand. The cloud tenders more elasticity supporting an environment which starts from only a few servers, grow quickly to hundreds of servers, and then scale back down to the initial size if required. [3]

<p>| Table 1 – Grid and Cloud Computing Technically Compared [11] |</p>
<table>
<thead>
<tr>
<th>Grid Computing</th>
<th>Cloud Computing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Means of utilization (e.g. Harris 2008)</td>
<td>Allocation of multiple servers onto a single task or job</td>
</tr>
<tr>
<td>Typical usage pattern (e.g. EGEE 2008)</td>
<td>Typically used for job execution, i.e. the execution of a program for a limited time</td>
</tr>
<tr>
<td>Level of abstraction (e.g. Jha et al. 2008)</td>
<td>Expose high level of detail</td>
</tr>
<tr>
<td>Virtualization of servers; one server to compute several tasks concurrently</td>
<td>More frequently used to support long-running services</td>
</tr>
<tr>
<td>Provide higher-level abstractions</td>
<td></td>
</tr>
</tbody>
</table>

Foster et al. (2008) for example describe the relationship between Grid and Cloud Computing as follows: [11]

“We argue that Cloud Computing not only overlaps with Grid Computing, it is indeed evolved out of Grid Computing and relies on Grid Computing as its backbone and infrastructure support. The evolution has been a result of a shift in focus from an infrastructure that delivers storage and compute resources (such is the case in Grids) to one that is economy based aiming to deliver more abstract resources and services (such is the case in Clouds).”

D. Cloud Computing characteristics

The characteristics of cloud computing are [1]:

a. Reduction of Cost

There are a number of reasons to attributing to the lowered cost of Cloud technology namely the billing model being pay as per usage, the infrastructure acquisition requirements are not there thus lowering maintenance. Initial expenses and recurring expenses are very much lower than traditional computing.

b. Elasticity

Due to the rapid provisioning and scaling capability of services offered by cloud, it offers a highly elastic and scalable environment. To consumer, the services available for provisioning appears to be endless and could be purchased whenever required.

c. Security and Availability

The cloud authorizes the data access by the end users. However security threat is always there with the end users. The requests have to be fulfilled in every case and the data and infrastructure should always available.

d. Flexibility

By using the most appropriate building blocks necessary for deployment, cloud computing mainly stresses on deployment of applications in market as quickly possible.

e. Geographical independence

The data shared and services offered by cloud can be accessed from any location across the globe

f. Multi-tenancy

The resources being offered by the service providers can be shared with the consumers at the network, host and application level.

g. Pay-as-you-go

Consumers of data are only required to pay for the resources they consume such as processing cycles and disk space.

h. Self-Provisioning of Resources

Consumers have the ability for the self-provisioning of resources without requiring any intermediate interference.
In this article, we give the tuned steps of cloud environments. Step first give the introduction of cloud computing and historical perspective have been discussed in next section. Section third is focused on resources management and allocation. The types of provisioning are described in same section. Fourth section explains the virtual machine and migration fundamental. Problem is identifying and defined in the next section and proposal has been given to solve the problem. Research methodology has been discussed to control the complexity into problem. CloudSim tools are used for simulation and result is explained. Comparative captured the results.

II. RESOURCE PROVISIONING

A. Introduction

Resource management refers to the proficient allocation of workload to the shared computing resources. This involves process of maximizing resource utility or workload throughput under a given set of constraints often technological and economical for achieving a desired goal. Resource management encompasses [6]:

a. Characterization: This involves knowledge gaining and understanding of the system workload and its resource requirements.

b. Allocation: The allotment of workload to resources across contending tasks or services.

c. Adaptation: The adjustment and accommodation of system and environmental changes such as failures and changes in workload.

The phenomenon of resource provisioning implies the selection, deployment, and run-time management of software (e.g. load balancers, database management servers) and hardware resources (e.g., CPU, network & storage) for guaranteeing assured performance for applications. The resource provisioning also utilizes Service Level Agreement into consideration for providing service to the cloud users. SLA is an initial agreement between the cloud users and cloud service providers which ensures Quality of Service parameters like availability, reliability, performance, response time etc. Based on the application requirements Static/Dynamic Provisioning of resources needs to be made in order to efficiently make use of the resources without violating SLA and meting these QoS parameters. Over provisioning and under provisioning of resources must be circumvented. [12]

B. Types of provision

For Basically are three types [12]

a) Static Provisioning

This type of provisioning technique may be utilized for allocation of resources for applications having consistent and predictable with unchanging demands or workloads. The facility of advance provisioning allows the customer to have a contract with the provider for services and the provider in advance reserves the appropriate resources for service. The customer can be charged a flat fee or can be billed on a monthly basis.

b) Dynamic Provisioning

When the applications’ demand by applications varies or may change with time the techniques of dynamic provisioning techniques have been “dynamic provisioning” techniques have been recommended. In this technique, the VMs may be migrated to new nodes within the cloud and the provider allocates more resources as they are needed and removes them when they are not. The customer is billed on a pay-per-use basis. When a hybrid cloud is created using dynamic provisioning then it is referred to as cloud bursting.

c) User Self-provisioning

Using the facility of cloud self- service or user self- provisioning, the customer can purchase resources from the cloud provider through a web form by creating a customer account and then paying for resources with a credit card. The provider's resources are available for customer to use within the specified time frame.

C. Parameters for Resource Provisioning

Response Six parameter are considered for resource provisioning [12]

(i) Response time: The resource provisioning algorithm used should be in order to take minimal time to respond while executing the task.

(ii) Minimize Cost: From the users’ point of view, the cost of services charged should be minimized.

(iii) Revenue Maximization: From the Cloud Service Provider’s view, the revenue obtained by providing services should be maximized.

(iv) Fault tolerant: The algorithm should be robust enough and continue to provide services in spite of failure of nodes.

(v) Reduced SLA Violation: The algorithm must be designed to be so efficient so as to be able to reduce SLA violation.

Reduced Power Consumption: Power consumption is an important constraint in cloud environment and must be minimized in VM placement & migration techniques.

III. VIRTUAL MACHINE MIGRATION

Virtualization means something which isn’t real”, but gives all the facilities of a real. Virtualization refers the creation of a virtual version of an object, as opposed to tangible variant. In Cloud Computing context, Resource Virtualization refers
to the separation or combination of part or all of the computing device’s hardware resource for different or shared purposes respectively. Resource Virtualization can be viewed in three aspects namely [6]:

a. Network Virtualization: This aspect deals with the method of multiplexing network devices although separating traffic abstracting the complexity of the underlying network topology.

b. Storage Virtualization: The physical storage devices from multiple networked devices are pooled at a central place presenting the appearance of a unified storage device manageable from a central location.

c. Server Virtualization: Virtualization of servers provides independent access to server resources whilst hiding the underlying implementation details of the hardware away from the end-user for the purpose of increasing utilization and improving ease of management.

A. Definition

There are many definitions, with almost the same meaning, but it is necessary to put it in context to give a more exact one. Virtualization of operating systems also known as server virtualization has been defined by as “a way of making a physical computer function as if it were two or more computers where each non-physical or virtualized computer is provided with the same basic architecture as that of a generic physical computer. Virtualization technology therefore allows the installation of an operating system on hardware that does not really exist.” [10]

B. Virtualization Techniques

The basic virtualization techniques are [6]:

a. Full Virtualization

b. Hardware Assisted Virtualization (HVM)

c. Partial Virtualization:

   (i) Paravirtualization
   (ii) Hybrid Virtualization
   (iii) Operating System-Level Virtualization

a) Full Virtualization suggests creating enough hardware to allow a guest operating system from a potentially different architecture to run in isolation and having performance at a level achievable to combat the penalty due to the overhead associated with emulating hardware at the transistor level. The dedicated remote datacenter delivers the services in a fully virtualized manner. The advantages offered by full virtualization include [8]:

   ➢ allocating a computer system among multiple users
   ➢ separating users from each other and from the control program
   ➢ emulating the desired hardware on another machine

Fig. 4 Virtualization architectures [10]
b) Hardware Assisted Virtualization exploits the additional hardware capabilities in the form of Virtual Machine Extensions (VMX) inside the host processor instruction set to speed up and separate context switching between processes running in different virtual machines. This in turn increases the computational performance of a virtual machine since the instructions can be directly passed to the host processor without having to be inferred and isolated, at the expense of limiting guest operating systems to use the same instruction set as the host machine. Complete Hardware Assisted Virtualization of all computer subsystems such as I/O and memory management is yet to be implemented completely in any VMM.

c) Partial Virtualization revolves around the simulation of the majority but not all the underlying hardware of a host and supporting resource sharing but at the same time not completely guaranteeing isolated guest operating system instances. This fundamental approach is utilized in Paravirtualization, Hybrid Virtualization and Operating System-Level Virtualization. Paravirtualization creates all or most hardware by providing software interfaces or API’s which are similar to that of the underlying hardware of the host. These can be employed to create virtual hardware device drivers for guest operating systems that achieve near native performance to that of the host.

Paravirtualization has the following advantages as given in [8]:
- Disaster recovery
- Migration
- Capacity management

Hybrid Virtualization utilizes the principles of both Hardware Assisted Virtualization and Paravirtualization to enhance the native performance from guest operating systems but incorporating the disadvantages of both.

C. Virtual Machine Migration
Migration of VMs across clouds and datacenters has been the most challenging task of cloud computing because of the following four main reasons:
- First, large volumes of data are transferred across datacenters in migrating across VMs. The VM images are itself large (typically 1-30GB in size) and often the applications deployed on the cloud involve multiple VMs with different images (e.g. a three-tier web application, a business analytics solution or a virtual cluster). The process of application migration often involves transferring multiple large VM images.
- Second, such huge data transfers can only be accomplished over wide area networks having limited bandwidth. A sudden or unexpected application migration approach may obstruct the network links between the datacenters leading to unacceptable performance degradation.
- Third, a VM image transfer service has to restrict within the limits imposed by the existing APIs of clouds’ VM image repositories.
- Any changes to a cloud’s VM image repository is not acceptable by the cloud providers for strategic as well as for technical reasons during the migration process.

Finally, the problem becomes more complex when the transferred VM images are to be instantiated at the destination site.

IV. PROBLEM IDENTIFICATION & PROPOSAL
There have been numerous optimal proposals and algorithms addressing the problem of resource allocation in both time as well as space shared scenario in cloud environment. Their aim is to maximize throughput and minimization of cost factor. But little attention has been paid to cater user requirements in the allocation process. In order to incorporate users’
perspective and guaranteeing quality of service there is a need to fine tune the resource allocation process as per user demands. Thus in order to fine tune the resource allocation mechanism according to qos factor specified by the user, we proposed the tuning algorithm [16]. Here we integrate our previous proposal to simulate the Tuning algorithm [16]. CloudSim environment with workflow has been deployed for simulation purpose.

![Flow chart for tuning algorithm](image)

**Tuning Algorithm [16]**

1) Suppose n number of clients $C = \{C_1, C_2, C_3, ..., C_n\}$ sends request for Resources $R = \{R_1, R_2, R_3, ..., R_m\}$ to Resource Provisioning Manager (RPM) in a cloud environment.

2) RPM forwards this request to Request Tuner (RT).

3) RT analyses the quality of service parameters (Throughput, Cost, Delay, Loss) mentioned in client’s request.

4) RT generate a command and send this to all resources (VMs) connected in cloud network.

5) VMs reply their status to RT in the form of a header.

6) Header mentions the VM ID, availability status (ON, OFF), lifecycle phase (Creation, Migration, Termination), busy schedule etc.

7) RT receives all the information from the VMs and creates a Matrix for warehouse.

8) Matrix includes columns like VM ID, QoS parameters. Rows signify the corresponding attributes of each VM.

9) RT tunes the client’s request demanding particular QoS requirements with the available VM’s QoS parameters

   a) If N number of client requests match with m number of available VMs then assign the resources to clients following Priority along with Round Robin scheduling to ensure fair sharing of resources.

   b) If a single request matches with more than one VM then choose the VM following the Load Balancing Strategy.

   c) If the client request doesn’t find a suitable QoS VM then assign the client a VM which guarantees Best effort services.

10) In case of any changes in the QoS parameters of VM, the VM notify RT for the required updation.

**V. RESEARCH METHODOLOGY**

To address such a significant scenario of resource provisioning as per user demands, the qualitative research methodology has been followed. The resource tuner has been introduced between the resource provision manager and virtual machine for verifying the qos parameters from clients request to VM’s. The matching VM’s can be allocated to the appropriate client as per the qos desired by the user. The tuning allocation policy utilizes the following strategies:

**A. Round Robin**

This algorithm attempts to optimize the resource allocation process by granting each user a fair share of resources for a quantum of time in a round robin fashion. [15]
B. Priority Based
Priority based resources provisioning algorithms uses some form of priority i.e. load balancing factor, job relevance, or some random priority feature and allocates the resources to higher priority jobs first and then subsequent priority jobs are serviced. [15]

C. Load Balancing Strategy
The phenomenon of load balancing makes sure that all the processors in the system or nodes in the network does approximately the equal amount of work at any instant of time. The load can be memory capacity, CPU load, delay or network load. [15]

VI. SIMULATIONS & RESULT DISCUSSION
CloudSim has been used for simulation of the tuning algorithm since it provides an efficient framework for modeling and simulation of Cloud Computing services and infrastructure.
Simulations have been carried out in two steps:-
1) Existing resource allocation process.
2) Involving qos parameter for resource allocation mechanism.
   1) In this simulation process we start by increasing the host and datacenter one by one along with received cloudlets. At initial level the single datacenter with single host has been employed using the simple allocation policy.

We increase the datacenter to two and host is constant, then allocation is performed.

Now we increase the datacenter and host one by one in similar way of increments and observe the allocation.
2 Involving qos parameter for resource allocation mechanism

Resource allocation has been carried out using single datacenter with single host.

ii) Next we increase the datacenter to two and host is constant, then allocation is performed.

Now both the datacenter and host are increased one by one.
Then this variance has been performed according to the proposal and the resource allocation process is monitored.

Finally we analyze the output of resource allocation in all three cases as per the proposal. The tuning algorithm has been introduced in the existing variances and appends the qos parameter into the VM characteristics and the same is also provided by the client during request phase. We introduce the boolean qos parameter in VM along with long size (image size), int ram (vm memory), int mips, long bw, int pesNumber (number of cpus), and String vmm (VMM name). The new VMs is created along with userId, mips, pesNumber, ram, bw, size, vmm, qos, new CloudletSchedulerTimeShared. The datacenter broker follows the Priority, Round Robin and Load Balancing Strategy in the process of resource allocation as suggested by the tuning algorithm. The resource allocation process has been accomplished considering the qos factor desired by the client i.e. the qos factor mentioned in the request is matched with the qos guaranteed by the VM. In case a match is found with more than one VM then allocation is performed according to Load Balancing Strategy. When N requests match with M VMs then the resources are allocated on the basis of priority and Round Robin Algorithm. In case of no qos demanded by the user a VM guaranteeing best effort service is allotted.
Limitation:
1. The proposed algorithm will be applicable when there are more than two VMs (which is generally the case in cloud environment) and the allocation policy conflicts for more than two similar resources with matching qos factor.
2. For applications without quality of service requirements, the tuning process is futile.

VII. FUTURE SCOPE
1. The researchers are invited to deploy this algorithm at broad level and check the accuracy for different combinational cases. The case studies can trigger more improvements for the enhancement of algorithm.
2. We considered the application of qos parameter for tuning purpose in case when the qos demanded by user matches the qos provided by the host but this may also be improved for involving false scenarios when no qos is desired by the user or no qos is guaranteed by the host.

VIII. CONCLUSIONS
Resource provisioning has been a major area of concern in cloud environment since its advent. The research focus has been to optimize the resource allocation in terms of throughput and cost. But a little attention has been paid on users demands and the quality expected by the user during allocation phase. In order to cater this requirement, tuning of resource provisioning and VM migration has been proposed as per clients’ demands. To involve users’ need in cloud environment for allocating the resources we analyzed the previous allocation process and introduced a new parameter qos to tune the existing mechanism. Basically tuning is designed for both sides: in client side, as desired qos parameter in request and host side qos parameter in host VM for providing optimal results for resource provisioning and migrating the virtual machines. The proposed scenario has been simulated in CloudSim environment. The simulation results have been analyzed and the tuned performance is evaluated. To conclude, the limitations of the work and the future scope have been identified.

ACKNOWLEDGMENT
Special thanks to Mr. Rajkumar Buyya for contributing CloudSim as a modeling and simulation tool for simulating cloud environment. We would also acknowledge the research scholars for contributing their master’s and doctoral thesis in this domain.

REFERENCES

