VM Provisioning Policies to Improve the Profit of Cloud Infrastructure Service Providers

Komal Singh Patel
Electronics and Computer Engineering Department
Indian Institute of Technology Roorkee
Roorkee, India
Komal35iitr@gmail.com

A. K. Sarje
Electronics and Computer Engineering Department
Indian Institute of Technology Roorkee
Roorkee, India
sarjefec@iitr.ernet.in

Abstract—**Cloud computing is an emerging technology in the IT world. Some features of cloud, such as low cost, scalability, robustness and availability are attracting large-scale industries as well as small business towards cloud. A virtual machine (VM) can be defined as a software that can run its own operating systems and applications like an operating system in physical computer. As the number of users increases, allocation of resources and scheduling become a complex task. The optimization of VM provisioning policies offer improvement like increasing provider's profit, energy savings and load balancing in large datacentres. In cloud computing when resource requirement of user's requests exceed resources limits of cloud provider, to fulfil the requests the cloud provider outsources to other cloud providers resources, this concept is known as cloud federation. In this paper we propose an algorithm for VM provisioning in federated cloud environment. The approach tries to improve the cloud providers profit. We have used the CloudSim to find-out the results and result show that how Cloud federation help to Cloud providers in order to improve its profit .**

Keywords; Cloud provider, outsourcing, federation, profit, Cloud computing.

I. INTRODUCTION

In the current scenario, no one wants to buy and maintain costly computing resources permanently, everyone wants to solve his problems remotely by hiring resources from other providers on the rent basis. This approach leads to the creation of cloud computing. 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. At present some major cloud providers are Amazon Web Services [1], Microsoft Azure [2] and Google AppEngine [3]. These cloud providers offer many type of services for monitoring, managing and provisioning resources and application services.

As user's need of resources increases, sometimes a single cloud provider is not able to handle all the incoming user requests. So cloud federation comes into picture, in which multiple cloud providers make a federation to share their available resources in order to fulfill incoming user requests. By using this cloud federation technique, A cloud provider can support more number of users with same number of resources [4]. The cloud federation concept in cloud computing offers two major benefits. One, it allows providers to earn revenue from computing resources that would otherwise be idle or underutilized. Second, cloud federation enables cloud providers to expand their geographic footprints and accommodate sudden spikes in demand without having to build new points-of-presence.

Virtual machine (VM) provisioning is high level resources management of cloud provider. Web applications in present scenario shifted on cloud because number of users who access the service of particular web server change with time. During the busy hours service provider needs more resources and at other periods of time load on the service providers are very less, so there is a need of continuous scale up and scale down the service providers infrastructure of resources. These scale up and scale down operations require dynamic VM provision [6]. In this paper we propose two algorithms for VM provisioning among multiple service providers. Here we use the cloud provider, cloud service provider, service provider and resource provider interchangeably. The remainder of this paper is organized as follows. Section II presents related work. Section III describes proposed methods. Section IV describes the experimental results and Section V concludes the paper.

II. RELATED WORK

References [1]-[3] describe the fundamentals concept of cloud computing like architecture of cloud, deployment models, service models and essential characteristics of cloud, job scheduling, resource allocation, security and virtualization [4].

One of the important characteristics of cloud is unlimited resources. Users can request any number of resources at any time depending on the needs, but sometime a single cloud service provider is not able to guarantee unlimited resources. The concept of federation was first given in grid computing, with the aim to get high utilization of resources. The same idea of federation in cloud was first given in Reservoir project [5]. In this project, authors explain the difficulties that occur while merging the cloud providers with different APIs and platforms. The project did not propose any method to decide when a provider has to transfer tasks to another cloud provider. Goiri et al. [6] proposed a profit based method to make decisions related to outsourcing or selling idling resources. But, the authors did not take into account different kinds of VMs like, on-demand, reserved and spot VMs. Spot is special type of low priority VM. It is terminated when high priority request comes to provider.

Yi et al. [7] give an approach to minimize the costs of computations using Amazon EC2s spot instances for resource provisioning. This paper also considers the application of market-oriented mechanisms in federated environment [4]. These technique offers fairness and benefits for cloud providers that are the members of federation.

Till now large amount of work has been done in the area of VM provisioning in single cloud provider. But, VM provisioning among the multiple cloud provider is still open important area research. The objective of VM provisioning is to provide the guarantee that each computing resource is distributed efficiently and fairly, and in the end improves resource utilization. Most of research of VM provisioning is based on migration of VMs from one host to another.

In cloud computing environment, there are two parties which we will need to focuses on, one is client/user and other is cloud service provider. Currently, there are few algorithms available, out of which some focus on client and others on cloud service provider.

The main goals of the algorithms which focus on the clients are –

- Reduction of the cost of user's job execution at the cloud provider side.
- Allocation of resources in such a manner that jobs complete within a given time span.
- Minimization of job completion time .
- Allocation of VM such that QoS (quality of services) for a user should be met.

The goals of the algorithms which focus on cloud Providers are-

- To improve the profit of cloud provider
- To minimize the number of physical nodes (servers) and power .
- To provide load balancing among the Datacenters.

All the above mentioned algorithms work within a single cloud provider. But in the current scenario, where the number of users and its requirements are increasing much rapidly, it is not possible to handle all the clients' requirements by a single cloud provider. So there is a need of research on resource allocation in cloud federation, where several cloud providers share their resources to fulfill user's requirements.

III. SYSTEM ARCHITECTURE

Figure 2 shows the model for federated clouds [10]. Cloud providers share the resources with each other on the basis of pre-defined rules of trust. In this way, in cloud federation provider get opportunity of outsourcing the resource of other provider on the basis of its client requirements. It also allows to rent its free resources to other providers in order to improve resource utilization.

In cloud federation, we need an entity that will have the information about the resources available, cost of each resource and other necessary field of different cloud providers. This entity handles incoming user requests and allocates the resources to users depending on their needs. This proposed model is an extension of the cloudsim simulator for federated scenarios.

Classes used in the Model-

Cloud Brokers- Broker is an entity that work as a inter-mediatory between users and cloud service provider. It keeps information about available resources, cost of each resources, on the basis of these values and user quality of service requirement, broker sends the requests to appropriate cloud provider.

Cloud collaborator- This entity performs the responsibility as a central component that keeps track of the available resource within the each cloud service provider, what are the specification of resources. All cloud collaborator provide the information about the resource utilization, free resources, etc. On the basis of all knowledge of federation it decides the resource price and other strategies. This entity executes in a process with the aim to maximize the performance of the resource allocation.

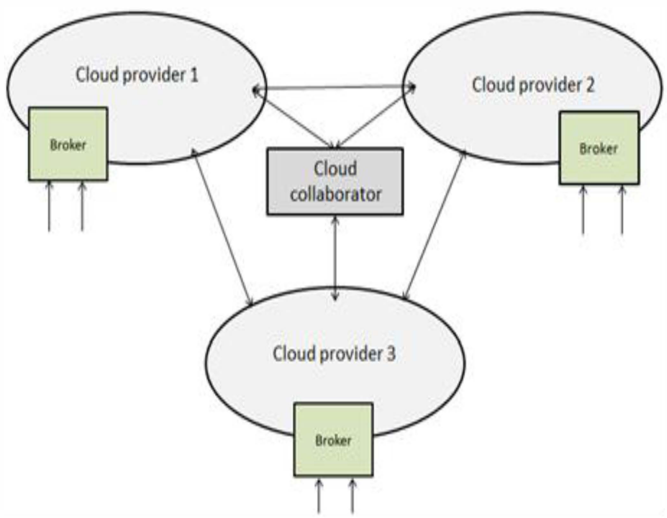

Figure 2: System Model

Cloud provider- Class (entity) keeps all the resources of cloud provider and characteristics of these resources. Our algorithm mainly focuses on pricing models, available resources, utilization of data center. Cloud provider periodically watch the datacenter utilization, transfer it to next datacenter so that load balancing can be achieved. this component controls the private cloud provider, and identifies the user's requirements to negotiate the provision of resources according to the demand. This entity helps the cloud broker in services allocations.

IV. PROPOSED METHODS

Currently, algorithms for improving the profit of the cloud providers are not available. In this paper, we propose algorithms which focus on increasing the availability of resources to customers and increasing the profit of cloud service providers. In the first algorithm, when a user request comes to a cloud provider, it first searches resources in its own datacenter. If it doesn't find free resources, then it contacts other cloud providers by using the shared component between them and allocate the resource from other cloud provides to virtual machine.

Let us consider the following cases.

A. Without Federation

In case without federation when a request comes to particular cloud provider. If cloud provider has sufficient resources, it accepts the requests and allocates the required number of resources. And if service provider does not have required resources it simply rejects the request. So cloud provider earns revenue only by selling its own resources. Cost of managing datacenters includes a fixed cost needed at beginning to establish the datacenters and a variable cost, needed to keep resources (servers) up in datacenters.

B. Outsourcing

Consider the case when requests from user come to particular cloud provider, provider does not have sufficient resources to fulfill the requests. We can say cloud provider (datacenter) is fully loaded. Cloud provider has two options, either to reject the requests or outsource the resource from other cloud provider on the basis of agreements. If cloud provider rejects the requests, it not only loses the money but market value of that service provider also decreases. So in the case of outsourcing total revenue becomes the revenue of provider itself and revenue earned by first outsourcing and, then selling the resources.

C. Insourcing

When the provider has underutilized resources, it has two options either to shut down or sell (insource) the extra resources (servers). It is not desirable for cloud provider to shut down its resources because providers already has spent a lot of money to established the datacenters. It is a market rule, that when any organization comes into market, it never thinks about going back, even if it sell its services at lower prices. Here total revenue becomes sum of the revenue earned from the requests that come to provider and revenue earned by renting its resources to other providers.

D. Insourcing and Outsourcing in Federation

In this case cloud providers are allowed for insourcing and outsourcing of services to/from other service providers. So in this case revenue generated becomes the sum of revenue generated by provider's own resources, revenue generated from outsourcing and revenue generated from insourcing as discussed above. And cost is same as in case of insourcing. Flow chart given in fig 1 shows how profit is calculated in different scenarios. Here the following notation is used.

C= cost of managing datacenter
F= fixed cost required to purchase the resources
H= time in hours
P = price per VM
N= total number of required VMs to complete user's requests
R1= revenue earned from its own resources
R2= revenue earned by insourcing its resources
R= total revenue earned in particular case
Co= cost that need to be paid to other providers when outsourcing other's resources,

value of a and b depend on the contract matrix among the service providers. Value of C is same for the case without-federation and insourcing. But in outsourcing and fully-federated case it is equal to sum of without federated and cost needed to pay for outsourcing.

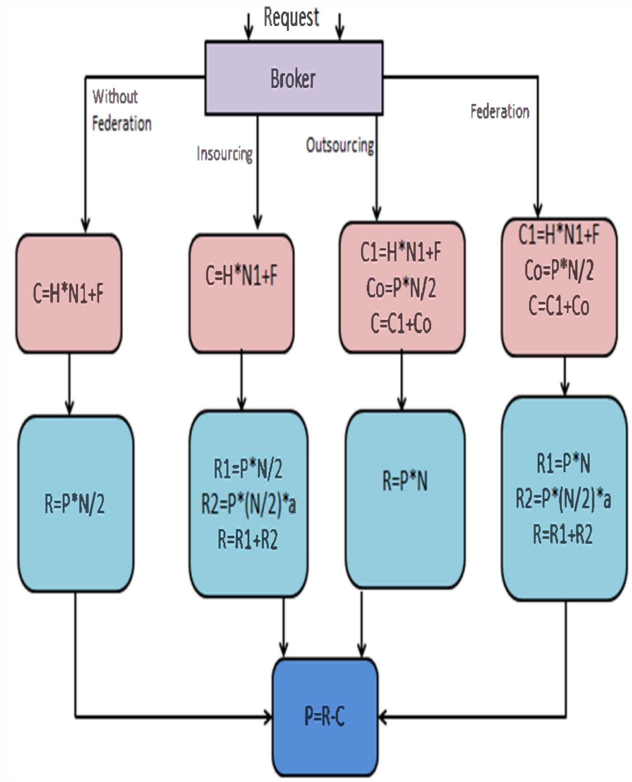

Figure 1: Profit Computation in cases

Revenue R is the amount that cloud provider earns by selling its resources. Without federation revenue is just the amount that provider gets by selling own resources, but in insourcing provider earns money by selling its resources to other cloud providers and in outsourcing provider earns money by taking the resources from one provider and selling that resources to user. In fully-federated case provider earns both from insourcing and outsourcing. Profit equals total cost subtracted from total revenue.

V. PROFIT EVALUATION

The first experiment aims at proving that federated infrastructure of clouds has potential to deliver better performance and service quality as compared to existing non federated approaches. A simulation environment that models federation of Cloud providers and a user is modeled.

A. Experiment Setup

The experiments presented in this section were developed using CloudSim [11] discrete-event Cloud simulator. The simulated Cloud scenario is composed of a federation containing Cloud providers. The number of providers is one of the simulation parameters, and we evaluated the effect of the policies considering different number of federation members. These midrange servers support in mean a maximum amount of 6 VMs per node, assuming small EC2 instances, which have a cost of 0.085 $/hour. For the sake of accuracy, each experiment is carried out 20 times by using different workloads and the average of the results is reported.

VI. RESULTS

We performed the experiments with different values of contract matrix [Cij]. The information contained in the matrix helps in calculating the amount of revenue that provider will get for the resources. The value in C_{ij} means that revenue multiplied by C_{ij} get to provider 'i', if the i^{th} provider uses the resources of j^{th} provider. For first experiment, values of contract matrix given below.

Contract [Cij]={{1,0.3,0.4,0.8},{0.7,1,0.6,0.2},{0.6,0.4,1,0.5},{0.2,0.8,0.5,1}}

Table 1: Simulation Results for Profit Revenue

	Revenue	Profit
Non-federated	0	0
Insourcing	31	26
Outsourcing	35	26
Fully-federated	49	44

The provider has many hosts. Hosts have same or different configuration. As we have discussed, users demands change with time. Using a traditional resource management methods, the provider becomes incapable to handle the request and rejects the requests which have exceed its current maximum capacity. As a result, cloud provider loses many user requests during busy hours.

When a user request is rejected, it results in loss to cloud provider in term of revenue and trust of the consumers. To avoid such situation outsourcing is performed. These results either in the table 1, show provider profit is zero when we perform for non-federated scenario. Its shows, when provider does not have resources and not able to use the resources of other provider, it reject the request.

Second experiment we did with the special value of contract matrix that is 0.5. it indicates both cloud provider one is requesting for resource of next provider and next provider are getting same amount of revenue.

Contract[Cij] ={{1,0.5,0.5,0.5},{0.5,1,0.5,0.5},{0.5,0.5,1,0.5},{0.5,0.5,0.5,1}}

Table 2: Simulation Results for Profit

	Revenue	Profit
Non-federated	0	0
Insourcing	31	26
Outsourcing	35	26
Fully-federated	49	44

VII. Conclusion and Future Work

In this paper, we have presented methods for VM allocation among multiple cloud providers, we have presented two algorithms. First one, allocates the resources to VM, so that it improve the cloud service providers profit. Second one, allocates VM in order to balance the load among the multiple datacenters in federated cloud environment.

We have shown in the results, how the providers can enhance their profit in cloud federation. Our first algorithm includes equations that helps cloud providers in making decisions (i.e. when to outsource and insource resources to and from other providers) for VM scheduling in federated cloud. Our simulation experiments have evaluated these equations to find the effect of some parameters in the provider's profit. These parameters include workload, the amount of free resources to be sold, the cost of outsourcing additional resources, the amount of outsourced resources, and the cost of maintaining servers.

There are a number of challenging issues that still remain open for investigation. In first algorithm to be profitable when all the nodes are working, our first proposed method gives us what should be minimum utilization and minimum price per VM. Based on these results, we plan to develop a scheduler for a real cloud system that will use the presented method of VM allocation for taking resource management decisions.

For our second algorithm it is describe to evaluate the results in real cloud computing environments, and propose VM provisioning policy based of QoS requirement mention by user in federated cloud environment

References

[1] Amazon Elastic Compute Cloud," http://aws.amazon.com/ec2 .

[2] Windows Azure Platform, http://www.microsoft.com/azure/ (March 17, 2010)

[3] Google App Engine, http://appengine.google.com (March 17, 2010)

[4] R. Buyya, R. Ranjan, and R. N. Calheiros, "InterCloud: Utility-oriented federation of Cloud computing environments for scaling of application services," in Proceedings of the 10th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'10), ser. Lecture Notes in Computer Science, vol. 6081. Busan: Springer, May 2010, pp. 13–31.

[5] B. Rochwerger, D. Breitgand, E. Levy, A. Galis, K. Nagin, I. M. Llorente, R. Montero, Y. Wolfsthal, E. Elmroth, J. Caceres, M. Ben-Yehuda, W. Emmerich, and F. Galan, "The Reservoir model and architecture for open federated Cloud computing," IBM Journal of Research and Development, vol. 53, no. 4, pp. 1–11, Jul. 2009.

[6] http://www.vmware.com/virtualization

[7] Íñigo Goiri, Jordi Guitart, Jordi Torres, "Economic model of a Cloud provider operating in a federated Cloud", Springer Science+Business Media, LLC 2011 Inf Syst Front DOI 10.1007/s10796-011-9325-x

[8] M. Mihailescu and Y. M. Teo, "Dynamic resource pricing on federated Clouds," in IEEE International Symposium on Cluster Computing and the Grid. Los Alamitos, USA: IEEE Computer Society, 2010, pp. 513–517.

[9] S. Yi, D. Kondo, and A. Andrzejak, "Reducing costs of Spot instances via checkpointing in the Amazon Elastic Compute Cloud," in In Proceedings of the 2010 IEEE 3rd International Conference on Cloud Computing (Cloud '10). Washington, DC, USA: IEEE, 2010, pp. 236– 243.

[10] B. Song, M. M. Hassan, and E.-N. Huh, "A novel Cloud market infrastructure for trading service," in Proceedings of the 9th International Conference on Computational Science and Its Applications (ICCSA). Suwon: IEEE Computer Society, Jun. 2009, pp. 44–50.

[11] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. D. Rose, and R. Buyya, "CloudSim: A toolkit for modeling and simulation of Cloud computing environments and evaluation of resource provisioning algorithms,"Software: Practice and Experience, vol. 41, no. 1, pp. 23–50, Jan. 2011.