Optimal Resource Provosioning in Cloud Computing

Shelja Jose M
Department of Computer Science and Applications
St Marys College
Thrissur, Kerala, India.

***Abstract* — Cloud computing is a technology that provides a variety of dynamically scalable and virtualized computing resources ranging from servers and storage to enterprise applications, in pay-as-you-use model via internet. Companies are able to rent resources from cloud on demand, so that their infrastructure cost can be reduced significantly. Multiple cloud users can request number of cloud services/resources simultaneously. So there must be provision that all resources are made available to requesting user in an efficient manner. However one of the major pitfalls related to this is the optimization of resources being allocated. It must be performed with the objectives of minimizing the costs associated with it, meeting customer demand and application requirements. In this paper a review of various policies for dynamic resource allocation in cloud computing is shown based on Topology Aware Resource Allocation (TARA), Linear Scheduling Strategy for Resource Allocation and Dynamic Resource Allocation for Parallel Data Processing. Moreover, significance, advantages and limitations of using Resource Allocation in Cloud computing systems is also discussed. It is believed that this paper would benefit both cloud users and researchers in overcoming the challenges faced.**

Keywords: Dynamic Resource Allocation, Cloud Computing, Resource Management, Resource Scheduling

I. INTRODUCTION

Cloud computing is a type of internet based computing that is relies on sharing of computer resource from anywhere and anytime. Possibly people can have everything they need on the cloud. It emerges as a new computing paradigm which aims to provide reliable, customized and QoS (Quality of Service) guaranteed computing dynamic environments for end-users. The basic principle of cloud computing is that user data is not stored locally but is stored in the data centre of internet. The companies which provide cloud computing service could manage and maintain the operation of these data centres. The users can access the stored data at any time by using Application Programming Interface (API) provided by cloud providers through any terminal equipment connected to the internet.

Not only are storage services provided but also hardware and software services are available to the general public and business markets. The services provided by service providers can be everything, from the infrastructure, platform or software resources. Each such service is respectively called Infrastructure as a Service (IaaS), Platform as a Service (PaaS) or Software as a Service (SaaS).

Cloud computing nowadays becomes quite popular among a community of cloud users by offering a variety of resources. Cloud computing platforms, such as those provided by Microsoft, Amazon, Google, IBM, and Hewlett-Packard, let developers deploy applications across computers hosted by a central organization. These applications can access a large network of computing resources that are deployed and managed by a cloud computing provider. Developers obtain the advantages of a managed computing platform, without having to commit resources to design, build and maintain the network. Yet, an important problem that must be addressed effectively in the cloud is how to manage QoS and maintain SLA for cloud users that share cloud resources.

The cloud computing technology makes the resource as a single point of access to the client and is implemented as pay per usage. Though there are various advantages in cloud computing such as prescribed and abstracted infrastructure, completely virtualized environment, equipped with dynamic infrastructure, pay per consumption, free of software and hardware installations, the major concern is the order in which the requests are satisfied. This evolves the scheduling of the resources. This allocation of resources must be made efficiently that maximizes the system utilization and overall performance. Cloud computing is sold on demand on the basis of time constrains basically specified in minutes or hours. Thus scheduling should be made in such a way that the resource should be utilized efficiently.

In cloud platforms, resource allocation (or load balancing) takes place at two levels. First, when an application is uploaded to the cloud, the load balancer assigns the requested instances to physical computers, attempting to balance the computational load of multiple applications across physical computers. Second, when an application receives multiple incoming requests, these requests should be each assigned to a specific application instance to balance the computational load across a set of instances of the same application. For example, Amazon EC2 uses elastic load balancing (ELB) to control how incoming requests are handled. Application designers can direct requests to instances in specific availability zones, to specific instances, or to instances demonstrating the shortest response times. In the following sections a review

of existing resource allocation techniques like Topology Aware Resource Allocation, Linear Scheduling and Resource Allocation for parallel data processing is described briefly.

A. Significance of resource allocation

In cloud computing, Resource Allocation (RA) is the process of assigning available resources to the needed cloud applications over the internet. Resource allocation starves services if the allocation is not managed precisely. Resource provisioning solves that problem by allowing the service providers to manage the resources for each individual module.

Resource Allocation Strategy (RAS) is all about integrating cloud provider activities for utilizing and allocating scarce resources within the limit of cloud environment so as to meet the needs of the cloud application. It requires the type and amount of resources needed by each application in order to complete a user job. The order and time of allocation of resources are also an input for an optimal RAS.

An optimal RAS should avoid the following criteria as follows:

*a) **Resource contention** situation arises when two applications try to access the same resource at the same time.*

*b) **Scarcity of resources** arises when there are limited resources.*

*c) **Resource fragmentation** situation arises when the resources are isolated. [There will be enough resources but not able to allocate to the needed application.]*

*d) **Over-provisioning** of resources arises when the application gets surplus resources than the demanded one.*

***e) Under-provisioning** of resources occurs when the application is assigned with fewer numbers of resources than the demand.*

Resource users' (cloud users) estimates of resource demands to complete a job before the estimated time may lead to an over-provisioning of resources. Resource providers' allocation of resources may lead to an under-provisioning of resources. To overcome the above mentioned discrepancies, inputs needed from both cloud providers and users for a RAS. From the cloud user's angle, the application requirement and Service Level Agreement (SLA) are major inputs to RAS. The offerings, resource status and available resources are the inputs required from the other side to manage and allocate resources to host applications by RAS. The outcome of any optimal RAS must satisfy the parameters such as throughput, latency and response time. Even though cloud provides reliable resources, it also poses a crucial problem in allocating and managing resources dynamically across the applications.

From the perspective of a cloud provider, predicting the dynamic nature of users, user demands, and application demands are impractical. For the cloud users, the job should be completed on time with minimal cost. Hence due to limited resources, resource heterogeneity, locality restrictions, environmental necessities and dynamic nature of resource demand, we need an efficient resource allocation system that suits cloud environments.

Cloud resources consist of physical and virtual resources. The physical resources are shared across multiple compute requests through virtualization and provisioning [23]. The request for virtualized resources is described through a set of parameters detailing the processing, memory and disk needs which is depicted in Fig.1. Provisioning satisfies the request by mapping virtualized resources to physical ones. The hardware and software resources are allocated to the cloud applications on-demand basis. For scalable computing, Virtual Machines are rented.

The complexity of finding an optimum resource allocation is exponential in huge systems like big clusters, data centers or Grids. Since resource demand and supply can be dynamic and uncertain, various strategies for resource allocation are proposed. This paper puts forth various resource allocation strategies deployed in cloud environments.

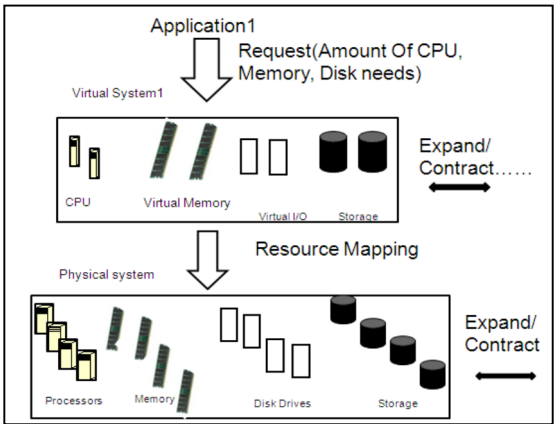

Figure1. Mapping of virtual to physical resources

II. RELATED WORK

Dynamic resource allocation problem is one of the most challenging problems in the resource management problems. The dynamic resource allocation in cloud computing has attracted attention of the research community in the last few years. Many researchers around the world have come up with new ways of facing this challenge.

In [2] authors have explained the algorithm for negotiation protocol for resource provisioning in detail. In [1], authors have made a comparison of many resource allocation strategies. In [3] authors propose a model and a utility function for location-aware dynamic resource allocation. A comprehensive comparison of resource allocation policies is covered in [4]. In [5] author has used a Genetic Algorithm for scheduling of tasks in cloud computing systems. This paper is not intended to address any specific resource allocation strategy, but to provide a review of some of the existing resource allocation techniques.

Not many papers which analyses various resource allocation strategies are variable as cloud computing being a recent technology. The literature survey focuses on resource allocation strategies and its impacts on cloud users and cloud providers. It is believed that this survey would greatly benefit the cloud users and researchers.

III. RESOURCE ALLOCATION STRATEGIES & ALGORITHMS

Recently many resource allocation schemes have come up in the literature of cloud computing as this technology has started maturing. Researchers around the world have proposed and / or implemented several types of resource allocation. Few of the strategies for resource allocation in cloud computing are covered here briefly.

A. Topology Aware Resource Allocation (TARA)

Different kinds of resource allocation mechanisms are proposed in cloud. The one mentioned in proposes architecture for optimized resource allocation in Infrastructure-as-a-Service (IaaS) based cloud systems. Current IaaS systems are usually unaware of the hosted application's requirements and therefore allocate resources independently of its needs, which can significantly impact performance for distributed data-intensive applications.

To address this resource allocation problem, an architecture that adopts a "what if" methodology to guide allocation decisions taken by the IaaS is proposed. The architecture uses a prediction engine with a lightweight simulator to estimate the performance of a given resource allocation and a genetic algorithm to find an optimized solution in the large search space. Results showed that TARA reduced the job completion time of these applications by up to 59% when compared to application-independent allocation policies.

1) *Architecture of TARA (Figure 2)*: TARA is composed of two major components: a prediction engine and a fast genetic algorithm-based search technique. The prediction engine is the entity responsible for optimizing resource allocation. When it receives a resource request, the prediction engine iterates through the possible subsets of available resources (each distinct subset is known as a candidate) and identifies an allocation that optimizes estimated job completion time. However, even with a lightweight prediction engine, exhaustively iterating through all possible candidates is infeasible due to the scale of IaaS systems. Therefore a genetic algorithm-based search technique that allows TARA to guide the prediction engine through the search space intelligently is used.

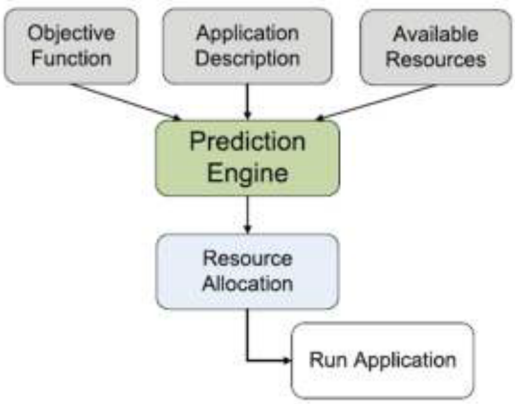

Fig. 2 Basic Architecture of TARA

2) *Prediction Engine:* The prediction engine maps resource allocation candidates to scores that measures their "fitness" with respect to a given objective function, so that TARA can compare and rank different candidates. The inputs used in the scoring process can be seen in Figure2, Architecture of TARA.

3) *Objective Function:* The objective function defines the metric that TARA should optimize. For example, given the increasing cost and scarcity of power in the data centre, an objective function might measure the increase in power usage due to a particular allocation.

4) *Application Description:* The application description consists of three parts: i) the framework type that identifies the framework model to use, ii) workload specific parameters that describe the particular application's resource usage and iii) a request for resources including the number of VMs, storage, etc.

5) *Available Resources:* The final input required by the prediction engine is a resource snapshot of the IaaS data centre. This includes information derived from both the virtualization layer and the IaaS monitoring service. The information gathered ranges from a list of available servers, current load and available capacity on individual servers to data centre topology and a recent measurement of available bandwidth on each network link.

B. Linear Scheduling Strategy for Resource Allocation

Considering the processing time, resource utilization based on CPU usage, memory usage and throughput, the cloud environment with the service node to control all clients request, could provide maximum service to all clients. Scheduling the resource and tasks separately involves more waiting time and response time. A scheduling algorithm named as Linear Scheduling for Tasks and Resources (LSTR) is designed, which performs tasks and resources scheduling respectively. Here, a server node is used to establish the IaaS cloud environment and KVM/Xen virtualization along with LSTR scheduling to allocate resources which maximize the system throughput and resource utilization.

Resource consumption and resource allocation have to be integrated so as to improve the resource utilization. The scheduling algorithms mainly focus on the distribution of the resources among the requestors that will maximize the selected QoS parameters. The QoS parameter selected in our evaluation is the cost function. The scheduling algorithm is designed considering the tasks and the available virtual machines together and named LSTR scheduling strategy. This is designed to maximize the resource utilization.

Algorithm:

1) The requests are collected between every predetermined interval of time
2) Resources Ri => {R1,R2, R3,...,Rn}
3) Requests RQi => {RQ1, RQ2, RQ3,..., RQn}

4) Calculate Threshold (static at initial)
5) Th = Σ Ri
6) for every unsorted array A and B
7) sort A and B
8) for every RQi
9) if RQ*i* < Th then
10) add RQ*i* in low array, A[RQ*i*]
11) else if RQ*i* > Th then
12) add RQ*i* in high array B[RQ*i*]
13) for every B[RQ*i*]
14) allocate resource for RQ*i* of B
15) R*i* = R*i* - RQ*i*; Th = Σ R*i*
16) satisfy the resource of A[RQ*i*]
17) for every A[RQ*i*]
18) allocate resource for RQ*i* of A
19) R*i* = R*i* - RQ*i*; Th = Σ R*i*
20) satisfy the resource of B[RQ*i*]

The dynamic allocation could be carried out by the scheduler dynamically on request for additional resources. This is made by the continuous evaluation of the threshold value. The resource requests are collected and are sorted in different queues based on the threshold value. The requests are satisfied by the VM's. Evaluation is made by creating VM in which the virtual memory is allocated to the longer and shorter queues based on the best fit strategy. This scheduling approach and the calculation of dynamic threshold value in the scheduler are carried out by considering both task and the resource. This improves the system throughput and the resource utilization regardless of the starvation and the dead lock conditions.

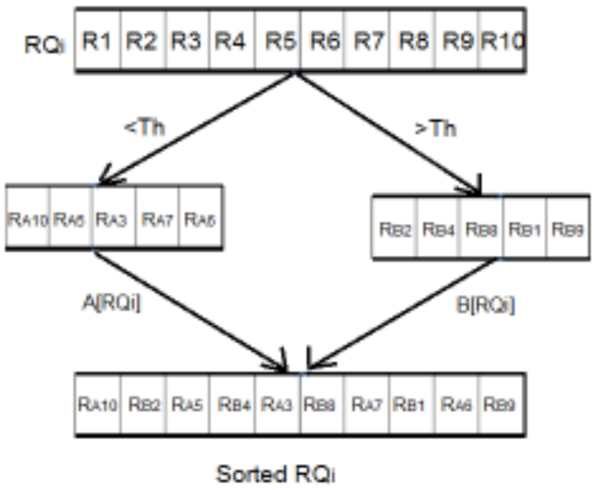

Fig. 3 Example of LSTR algorithm [3]

C. Dynamic Resource Allocation for Parallel Data Processing

Dynamic Resource Allocation for Efficient Parallel data processing introduces a new processing framework explicitly designed for cloud environments called Nephele. Most notably, Nephele is the first data processing framework to include the possibility of dynamically allocating/deallocating different compute resources from a cloud in its scheduling and during job execution. Particular tasks of a processing job can be assigned to different types of virtual machines which are automatically instantiated and terminated during the job execution.

1) *Architecture*: Nephele's architecture follows a classic master-worker pattern as illustrated in Figure. Before submitting a Nephele compute job, a user must start a VM in the cloud which runs the so called Job Manager (JM). The Job Manager receives the client's jobs, is responsible for scheduling them, and coordinates their execution. It is capable of communicating with the interface the cloud operator provides to control the instantiation of VMs. We call this interface the Cloud Controller. By means of the Cloud Controller the Job Manager can allocate or de-allocate VMs according to the current job execution phase.

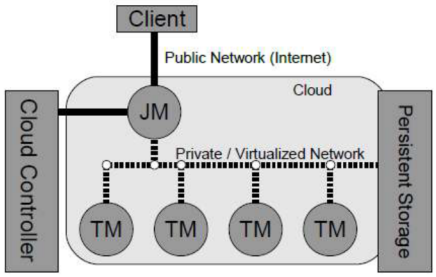

Fig. 4 Design Architecture of Nephele Framework

The actual execution of tasks which a Nephele job consists of is carried out by a set of instances. Each instance runs a so-called Task Manager (TM). A Task Manager receives one or more tasks from the Job Manager at a time, executes them, and after that informs the Job Manager about their completion or possible errors

2) *Job Description:* Jobs in Nephele are expressed as a directed acyclic graph (DAG). Each vertex in the graph represents a task of the overall processing job; the graph's edges define the communication flow between these tasks Job description parameters are based on the following criteria:

- Number of subtasks
- Data sharing between instances of task
- Instance type
- Number of subtasks per instance

3) *Job Graph:* Once the Job Graph is specified, the user submits it to the Job Manager, together with the credentials he has obtained from his cloud operator. The credentials are required since the Job Manager must allocate/deallocate instances during the job execution on behalf of the user.

IV. ADVANTAGES AND LIMITATIONS

There are many benefits in resource allocation while using cloud computing irrespective of size of the organization and business markets. But there are some limitations as well, since it is an evolving technology. Let's have a comparative look at the advantages and limitations of resource allocation in cloud.

A. Advantages:

1) The biggest benefit of resource allocation is that user neither has to install software nor hardware to access the applications, to develop the application and to host the application over the internet.

2) The next major benefit is that there is no limitation of place and medium. We can reach our applications and data anywhere in the world, on any system.

3) The user does not need to expend on hardware and software systems.

4) Cloud providers can share their resources over the internet during resource scarcity.

B. Limitations

1) Since users rent resources from remote servers for their purpose, they don't have control over their resources.

2) Migration problem occurs, when the users wants to switch to some other provider for the better storage of their data. It's not easy to transfer huge data from one provider to the other.

3) In public cloud, the clients' data can be susceptible to hacking or phishing attacks. Since the servers on cloud are interconnected, it is easy for malware to spread.

4) Peripheral devices like printers or scanners might not work with cloud. Many of them require software to be installed locally. Networked peripherals have lesser problems.

5) More and deeper knowledge is required for allocating and managing resources in cloud, since all knowledge about the working of the cloud mainly depends upon the cloud service provider.

V. CONCLUSION

Cloud computing technology is increasingly being used in enterprises and business markets. In cloud paradigm, an effective resource allocation strategy is required for achieving user satisfaction and maximizing the profit for cloud service providers. This paper summarizes the classification of RAS and its impacts in cloud system. Some of the strategies discussed above mainly focus on CPU, memory resources but are lacking in some factors. Hence this paper will hopefully motivated me to come up with smarter and secured optimal resource allocation algorithms and framework to strengthen the cloud computing paradigm.

REFERENCES

[1] V. Vinothina, Dr. R. Shridaran, and Dr. Padmavathi Ganpathi, *A survey on resource allocation strategies in cloud computing*, International Journal of Advanced Computer Science and Applications, 3(6):97--104, 2012.

[2] Bo An, Victor Lesser, David Irwin and Michael Zink, *Automated Negotiation with Decommitment for Dynamic Resource Allocation in Cloud Computing*, Conference at University of Massachusetts, Amherst, USA.

[3] Gihun Jung and Kwang Mong Sim, *Location-Aware Dynamic Resource Allocation Model for Cloud Computing Environment*, International Conference on Information and Computer Applications (ICICA), IACSIT Press, Singapore, 2012.

[4] Chandrashekhar S. Pawar and R.B. Wagh, *A review of resource allocation policies in cloud computing*, World Journal of Science and Technology, 2(3):165-167, 2012.

[5] Sandeep Tayal, *Tasks Scheduling Optimization for the Cloud Computing systems*, International Journal of Advanced Engineering Sciences and Technologies (IJAEST), 5(2): 111 - 115, 2011.

[6] Jung G, Sim KM; Location-Aware Dynamic Resource Allocation Model for Cloud Computing Environment, International Conference on Information and Computer Applications (ICICA), IACSIT Press, Singapore, 2012.

[7] Chandrashekhar PS, Wagh RB; A review of resource allocation policies in cloud computing, World Journal of Science and Technology, 2012; 2(3):165-167.

[8] Tayal S; Tasks Scheduling Optimization for the Cloud Computing systems, International Journal of Advanced Engineering Sciences and Technologies (IJAEST), 2011; 5(2): 111 – 115.