

# AUCTION BASED RESOURCEE ALLOCATION IN CLOUD COMPUTIG

A Abipriyan<sup>1</sup>, G Ramakrishnan<sup>2</sup>, E Iniyanehru<sup>3</sup>,

<sup>1</sup> Student, Information Technology, Easwari Engineering College, Tamil nadu, India  
<sup>2</sup> Professor, Information Technology, Easwari Engineering College, Tamil nadu , India  
<sup>3</sup> Technical Director, National Informatic Centre, Tamil nadu, India

## ABSTRACT

Cloud computing is one of the fastest emerging technology. As we all know, cloud is a wide pool of resource which provides resources based on the users request. Any service can be provided as a service through cloud. As the cloud contains many resources, there may be also wastage of resources. To reduce this wastage, cloud providers enter into auctioning of resources when the demand is high. Cloud computing includes distinct resources. Due to the complementary and supplementary effects between distinct assets, bidders have preferences not for just a single resource but also for a set of resources. To reduce the wastage of resources during allocation here proposed the Genetic Algorithm and executed through the web application. This method ensures both the cloud provider and the user economically benefited.

**Keyword :** - Auction , Cloud Computing, Bidding, and Genetic Algorithm

## 1. INTRODUCTION

Cloud computing is the fastest emerging technology based on demand services. It is an internet based computing where users pay-per-use Cloud users get many advantages like no need for infrastructure investment, storage capability, scalability, backup and recovery capabilities and mostly cost effective. Cloud provides many types of services that can be classified under infrastructure, platform and software.

Cloud computing is a kind of internet based computing that provides shared processing and data to cloud users based on demand. It is a model of enabling ubiquitous, convenient, on demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal effort of service provider interaction User can obtain the resources on pay-per-use model. Cloud computing provides many advantages like flexibility, disaster recovery, capital expenditure free , increased collaboration, access from anywhere.

### 1.1 The SaaS Perceptive

Software licensing is one of the key issues in data center. Provisioning adequate number of licenses required for running various applications is always a challenge. this paper aims at providing a solution to cloud

providers where by the software licenses are provisioned with almost care so that the cost involved in buying the software licenses is reduced. Also the requirements of the cloud users in terms of software licenses is met and maximum revenue is generated from the available licenses at the cloud data center. this is based using Software as a Service(SaaS) model.

## 1.2 Service Level Agreements (SLAs)

A Service Level Agreement (SLA) is a contract among customers and providers that guarantees the quality level of offered services. SLA is defined as (2): “An agreement between an IT Service Provider and a Customer. The SLA describes the IT Service, documents Service Level Targets, and specifies the responsibilities of the IT Service Provider and the Customer. A single SLA may cover multiple IT Services or multiple Customers”. (SLA negotiation paradigm in cloud can provide solution to reasonably allocate resources to the varying unpredictable workload and it ensures the elastic provisioning.[1]

The main contributions of this paper are as follows:

- Defining an algorithm for calculating the required resources for example, software licenses and effectively allocating the available licenses to the requests of cloud users. In this paper various algorithms like assignment and knapsack based algorithm are studied for efficient resource allocation.
- A formal model for resource management by cloud service provider according to need based.
- Among the algorithms, it is proved that Genetic algorithm is best suited for efficient and economic resource management by cloud service provider.

## 2. SYSTEM ARCHITECTURE

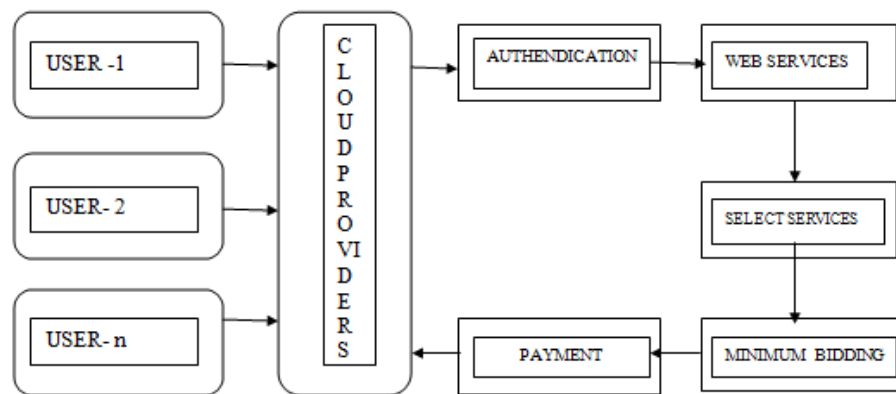


Figure 2.1 Architecture of the system

This architecture shown in Figure 2.1 aims to satisfy the cloud services based on the customer needs . there are various service providers gives their different services at different prices. here they provide the customer login and proceeds the checkout. Once the customer gets logged in they can see the various services by the different cloud providers. Customer can choose the service first and then it will show you the minimum bidding value of the slots. Then the customer must enter the required slots and enter the minimum bidding value of the storage. Next it will go the payment option, he should make sure the payment and pay threw the card. Once the payment get successful he can receive the conformation message on the screen.[5]

### 2.1 EXISTING MODEL

A model has based on SaaS was used to allocate resources for the tenants to specify their dynamic SLA requirements. Here they assuming the static set of  $P$  tenants  $C = \{C_1, C_2, C_3, \dots, C_P\}$  and a SaaS vendor  $S$  in our SaaS system. All tenant has several distinct *modes* of operation, each having different values for the tenant work parameters. And it explains the formalisms of the SaaS vendor has a planning span of  $\tau$  slots. For example, the SaaS vendor may plan the tenant selection (the set of tenants to be chosen for serving / on boarding) strategy on a daily basis ( $\tau = 24$  in this case). In this work, we consider one such window of  $\tau$  slots and analyze the on boarding problem within it. The SaaS vendor has a total number of  $L$  licenses on sale for its service. [3]

## 2.2 THE TENANT ONBOARDING PROBLEM

They collecting the no of tenants analyzing the all possible onboard tenants. all tenants have their respective deadlines with  $L$  licenses. Vendor may select the multiple on boarding strategies. In several modes multiple tenants have critically restrictions to be scheduled the same slots. Sum of licenses needs exceeds  $L$ .

### Case I- All tenants can be on boarded

With  $y_i = 1; \forall 1 \leq i \leq P$ , if the family of constraints are collectively satisfiable for a given value of  $L$ , here they present the strategy for on boarding meeting their deadlines. Boolean variables satisfy the conjunction of the boolean constraints.

### Case II- All tenants cannot be on-boarded

If the family of constraints are unsatisfiable for a given value of  $L$ , it means  $\exists i$  such that  $y_i = 0$  for some  $i$ ,  $1 \leq i \leq P$ . Essentially, all tenants cannot be on-boarded in this case and the vendor has to resort to a tenant selection activity with different objectives as discussed below. We consider three objectives here, namely, (a) maximum tenant on-boarding (b) revenue maximization and (c) a combination of revenue and tenant on boarding. However, our model is generic to handle others as well. [6]

## 2.3 OPTIMIZATION APPROACHES

### A. LICENSE MINIMIZATION - A GREEDY APPROACH

Here they proposed the minimum licenses possible to all tenants. Let  $L_k$  be the total licenses required for slot  $k$ . Initially  $L_k = 0$  for all  $k$ . Once some tenant  $C_i$  is allocated at slot  $k$  in mode  $m_{ij}$ ,  $L_k$  will be updated as  $L_k = L_k + Q_i(j)$ . Therefore after considering all the tenants, final value of  $L_k$  will give the total licenses required for that slot. Finally the maximum of  $L_k \forall k$  will give the minimum license requirement of the vendor.[4]

### B. MAXIMUM TENANT SELECTION

Here they proposed the maximum license to the maximum tenants. We need to select appropriate to allot given slot for a given tenant. We scheduling their deadlines with ascending order. The procedure of Greedy Heuristic takes  $C$  and  $L$  as input and returns the number or tenants on -boarded.[9]

## 2.4 PROPOSED METHOD

## 2.5 GENETIC ALGORITHM

A genetic algorithm (GA) is a method for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution. The algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm randomly selects individuals from the current population and uses them as parents to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution.

You can apply the genetic algorithm to solve problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, non differentiable, stochastic, or highly nonlinear. Generates a population of points at each iteration. The best point in the population approaches an optimal solution. Selects the next population by computation which uses random number

### **ADVANTAGES**

- Does not require any derivative information (which may not be available for many real-world problems).
- Is faster and more efficient as compared to the traditional methods.
- Has very good parallel capabilities.
- Optimizes both continuous and discrete functions and also multi-objective problems.
- Provides a list of “good” solutions and not just a single solution.
- Always gets an answer to the problem, which gets better over the time.
- Useful when the search space is very large and there are a large number of parameters involved.

### **DISADVANTAGES**

- GAs are not suited for all problems, especially problems which are simple and for which derivative information is available.
- Fitness value is calculated repeatedly which might be computationally expensive for some problems.
- If not implemented properly, the GA may not converge to the optimal solution.

## **3. IMPLEMENTATION**

### **MODULES**

#### **3.1 UI LOGIN AND DEVELOPMENT**

The User Interface screens are designed to monitor Inventory of web services required for the application and follow up on each stage of deployment till all the web services are successfully deployed.

#### **3.2 Database design and development**

Here we use the WAMP as server database .Each and every data has been stored in the php database and we can mange the entire data can be stored in the local server. We can maintain the customer profiles and data there the WAMP server only .

#### **3.3 Web Services design and development**

In this project we provide the various web services based on the customer needs. Customer can choose their own plan based on their requirement. So they have options to choose the slots and pay for their purchased slots only.

#### **3.4 Payment**

After purchased the slots by bidding process the payment options has shown to the customer portal. Then they should make the payment threw the card in the payment gateway. After make the payment they can receive the conformation message on the screen.

### 3.5 SERVER

Php MyAdmin is a web-based interface to a MySQL server. The interface is written in the PHP language, which is frequently used for web-based interactive programs. phpMyAdmin translates what you enter into the web browser, sends queries to the MySQL database, and translates the returned information back to web format. As we will see, there are other ways to access the MySQL database, including directly from Mathematical.



Figure 3.1 Wamp server

### 4. WORKING PROCESS

Here customer should sign up the webpage. After login process gets over there are various services available from different cloud providers. Then the customer can choose his required slots and make the payment process. Once the payment gets over the conformation message shown on the screen.

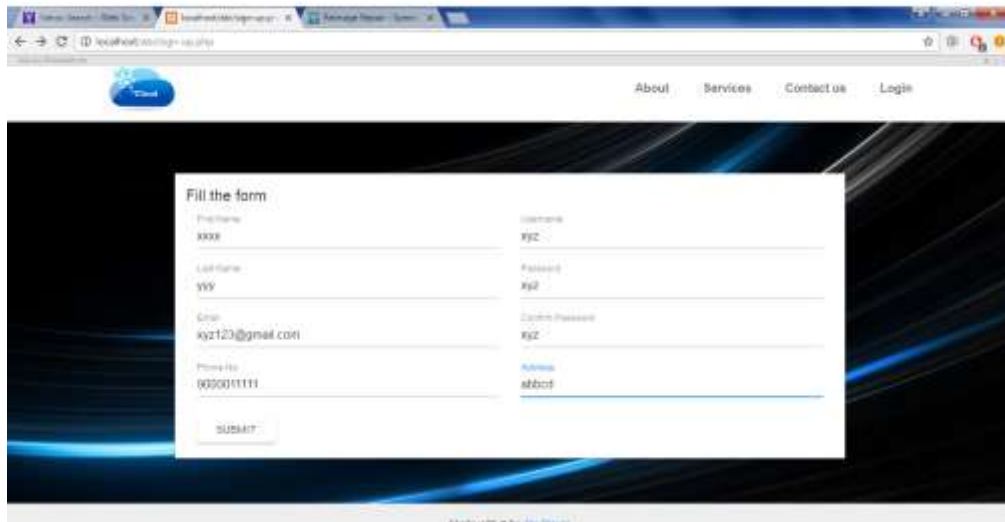


Figure 4.1 sign up page

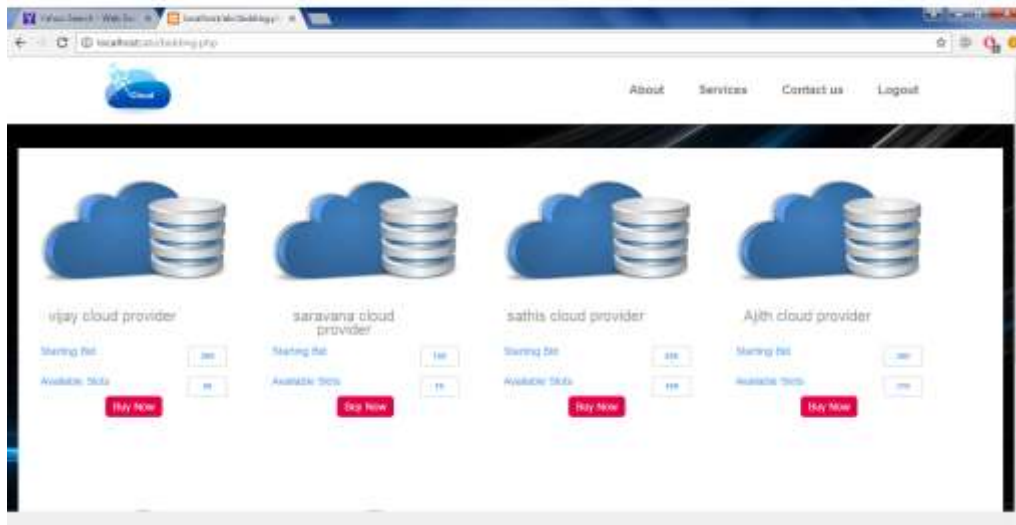


Figure 4.2 Cloud providers

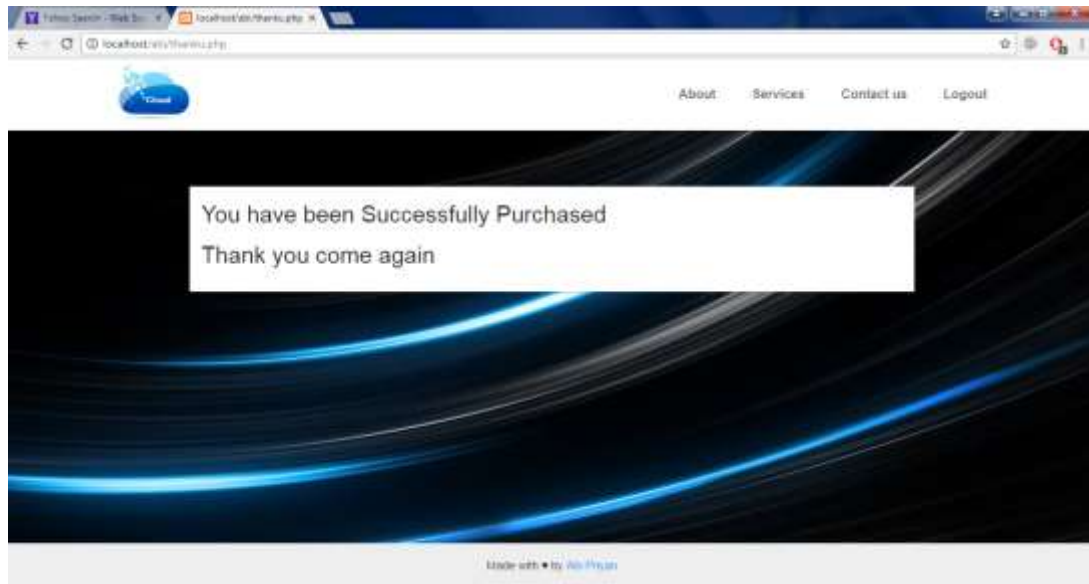


Figure 4.3 Output

## CONCLUSION

Thus the phase II report gives the clear idea about the various cloud providers gives the different services based on the customer needs. Customer can choose their services and select the required slots, then enter the minimum bidding amount. Then make the payment by the payment gateway. Once he makes the payment he gets the conformation on the screen.

So here the customer can gets his required slots by pay the correct amount . So the resource allocation should not be wasted in this method . In this both the cloud providers and the end users can economically benefited.

## REFERENCES

1. AbinandanPrasad and ShrishaRao (2014), 'A Mechanism Design Approach to Resource Procurement in Cloud Computing', Proc. IEEE Transactions on Computers, pp. 17-30.
2. AhuvaMualem and Noam Nisan (2002), 'Truthful approximation mechanisms for restricted combinatorial auctions,'Proc. 18th Nat'l conf. Artificial Intelligence, pp. 379-384.
3. IniyaNehuru E, Infant Smile ShyniJ, RanjithBalakrishnan (2016)'Auction Based Dynamic Resource Allocation in Cloud', ICCPT.
4. IniyaNehuruE,Veilumuthu M, Shobana Y,(2016) 'Auction Based Resource Management in Cloud', ICCPT.
5. Lehmann D, O'Callaghan L.I, andShoham Y,(2002) 'Truth Revelation in Approximately Efficient Combinatorial Auctions', J. the ACM, vol. 49, no. 5, pp. 577-602.
6. Quiroz A , Kim H, Parashar M, Gnanasambandam N , and Sharma N (2009), 'Towards Autonomic Workload Provisioing for Enterprise Grids and Clouds' Proc. IEEE/ACM 10th Int'l Conf. Grid Computing, pp. 50-57.

7. Rodrigo Calheiros, Rajiv Ranjan, Anton Beloglazov, Cesar A. F. De Rose and Rajkumar Buyya (2010), 'CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms', Proc. Wiley Online Library.
8. Vinothina V, Dr. Sridaran R and Dr. Padmavathi Ganapathi (2012), 'A Survey on Resource Allocation Strategies in Cloud Computing', Proc. Int'l Journal of Advanced Computer Science and Application, pp. 97-104.
9. Youwen Lan, Weiqin Tong, Zongheng Liu, Yan Hou (2012), 'Multi-Unit Continuous Double Auction Based Resource Allocation Method', Int'l Conf. Intelligent Control and Information Processing, pp. 773-777.
10. Zaman S and Grosu D (2010), 'Combinatorial Auction-Based Allocation of Virtual Machine Instances in Clouds', Proc. IEEE Second Int'l Conf. Cloud Comp. Technology and Science, pp. 127-134.
11. Zaman S and Grosu D (2013), 'A Combinatorial Auction-Based Mechanism for Dynamic VM Provisioning and Allocation in Clouds', Proc. IEEE Transactions on Cloud Comp., pp. 129-141.

