EAI Endorsed Transactions
on Cloud Systems

Research Article EAI.EU

# Advanced Mechanism to Achieve QoS and Profit Maximization of Brokers in Cloud Computing

Akanksha Sathish[1,*], Diana Dsouza[1], K. Ramitha Ballal[1], Archana M[1], Tarun Raj Singh[2], Glen Joshua Monteiro[3]

[1]Computer Science and Engineering, Sahyadri College of Engineering and Management, Mangalore, Karnataka, India.
[2]Computer and Communication Engineering, Manipal University of Jaipur, Rajasthan, India.
[3]Electronics and Telecommunication Engineering, St. Francis Institute of Technology, Mumbai, Maharashtra, India.

## Abstract

Cloud computing is a widely used technology today and thus, large number of applications are being stored in the cloud. To cater to the needs of the consumers an intermediate role known as cloud broker has been introduced. It helps to cut down overall expenses of the cloud users. The proposed system concentrates on configuration of the cloud broker to improve profit. This maximization scheme depends upon various factors like customer request, selling price of the resource, purchasing price of the resource, intensity of the request and so on. The system developed, aims at maximizing the profit of the cloud broker who services consumer requirements by providing cloud infrastructure at a lower cost from an infrastructure vendor. Availing pay-as-you-go schemes or on-demand payment has proven to be useful.The complex cloud landscape along with various billing schemes paves a way for a profit maximization model that incorporates temporal multiplexing by a middleman entity for a better economic upfront. The resource multiplexing is further enhanced by incorporating M/D/c queueing model and short-term and long-term renting schemes to optimize the resource allocation. The entire application has been implemented using Java especially the NetBeans IDE 8.2. Different algorithms were compared to further enhance the pricing scheme selection. Finally, Hill-Climbing algorithm is used to allocate the resources dynamically. The system further notifies the cloud broker on resource shortage through dynamic disk utilization graph. The proposed system considers quality of service and price of service as the determining factors in maximizing the net profit of the cloud broker.

**Keywords:** Cloud Broker, Cloud Computing, Multiplexing, Profit Maximization, Quality of Service.

Received on 04 June 2021, accepted on 18 June 2021, published on 23 June 2021

doi: 10.4108/eai.23-6-2021.170244

## 1. Introduction

The advancement in the field of cloud computing has seen a huge progress, this offers increased opportunities to cloud service providers. A considerate number of large-scale and small-scale cloud service providers have emerged such as AWS, GoGrid, Rackspace and so on. Cloud computing model has three divisions: Infrastructure as a Service (IaaS), Software as a Service (SaaS), and Platform as a Service (PaaS) to name them. In IaaS, an infrastructure vendor rents out storage, compute power and other resources available in the cloud to meet the customer demands through various billing schemes. In PaaS, the cloud service provider enables the customers to build, configure and deploy their applications by providing a platform where the clients do not have to model and maintain infrastructure. In SaaS, the software provider allows users to deliver their applications to multiple regions globally over the internet without having to be involved in complex management of hardware and software.

Cloud computing has become popular, as it eliminates the cost of infrastructure and maintenance. Also, the customers can dynamically scale their resources to meet

[*]Corresponding author. Email: akankshasathish315@gmail.com

the requirements in business. Cloud computing is rapidly turning into a powerful and effective method for computing assets. Cloud computing can give the most practical method for resource management. Cloud computing transforms the available data into common utilities for client usage.

One appealing cloud computing framework is the three-level construction, which comprises of cloud service providers, infrastructure vendors, and customers. The basic software and hardware functionalities are managed by the infrastructure vendor. A cloud provider rents out the required resources from an infrastructure vendor, constructs multi-server systems and delivers different kinds of services as per customer's requirement. A consumer presents an assistance solicitation to the cloud service provider, gets the ideal outcome from the provider by negotiating with the terms of the service level agreement. Based on the charge and quality of service the consumer pays for the assistance received.

Cloud computing services huge pool of consumers with guaranteed performance. Thus, maintaining the quality of service. The cloud performance is mainly contingent on task-scheduling. In order to achieve high optimization with regards to performance and resource utilization, there is a defined set of instructions and policies that are taken into consideration so as to designate tasks/jobs to copacetic resources (such as CPU, memory, and bandwidth). This is where scheduling algorithms come into picture. These algorithms have numerous advantages that take care of computing performance, resource management, improved load balance, etc. The task-scheduling can be categorized into different types. The prominent ones being static and dynamic. The characteristics of both scheduling algorithms differ in terms of load distribution. Dynamic scheduling schedules the tasks based on the system's capacity where the present state of virtual machines is considered. On the other hand, static scheduling distributes the load equally among the virtual machines. The state of the virtual machines is not a deterministic factor in the vicinity of static scheduling.

Two main aspects such as cost and income effects the pricing system of the cloud service provider. The income can be defined as the charge for the service provided to the customer and cost is the sum of the amount spent on rent and utility paid to the infrastructure vendor. The price depends on factors like number of customers over a given period, selling price of the resources, purchasing price of the resources, volume and speed of the multi-server system, the intensity of the request, the penalty in case of quality dissatisfaction, profit, task waiting time, task response time, task execution time. Thus, understanding the trend between cost and revenue is an important factor in deriving profit of the service provider. Hence, to derive profit the revenue of the service provider should increase.

### 1.1. Cloud Broker

Owing to the cloud's incredible capabilities, knowing the economics of cloud infrastructure is important. The most efficient set-up is the one that consists of basic elements such as the broker, the cloud service provider and the customer. The cloud broker rents out infrastructure from the cloud service provider and constructs an environment that is useful for serving the customer's needs. A customer makes a request for resources after going through various choices presented in the SLA catalogue. Once a request is made, the broker rents out resources to the customer at an affordable price and thus enhances the quality of service (QOS).

QOS can be maintained by catering to the requests within the given time frame and by choosing the best plan available. The profit of the broker can be maximized by adopting various schemes that could decrease the prices of the resources. Multiplexing of resources is also a way to maximize the revenue. The failure in providing the services within the time frame can lead to penalties. To overcome this, various queueing models can be implemented to maintain the QOS.

## 2. Related Work

In this section, we present a snapshot of the existing research relevant to the study stated in this paper.

There are numerous studies on various factors affecting the profit of a cloud service provider. [1] states that resource allocation is one such factor. The study focusses on the construction of a dynamic model for resource allocation by considering different heterogenous resource requirements [19]. The goal here is to maximize the profit of the cloud service provider.

The jobs submitted by the users are categorized into 3 different time frames. They are jobs submitted in fixed time intervals, jobs submitted in time window intervals and jobs submitted in time window slices. Here polynomial time can be seen as a solution for the given ideal allocation problem [28] . Furthermore, it proposes that bounded performance can allocate all the jobs at the same time. The study presents an algorithmic approach for effective resource allocation of multiple VM's (Virtual machines) [24].

The study in [20] focusses on adopting a strategic pricing model to maximize the profit of the cloud broker. Cloud service provider can make use of pay as you go and pay less per unit schemes to allocate resources from the infrastructure vendors. The infrastructure vendor offers volume discounts on bulk resources rented for a long period of time.

Devising an efficient billing scheme in the IaaS environment is difficult because of the complex landscape [29]. The resources allocated to the customers is billed on per hour basis, irrespective of the intensity of usage. This results in resource wastage for which the customers pay heavily. This is where the cloud broker comes into picture. The cloud broker can reduce the overall expenses of the users by making the right economic decisions [30].

The study proposes two multiplexing techniques [27]. They are temporal multiplexing and spatial multiplexing. In temporal multiplexing many customers are accommodated for resource allocation, this not only reduces the cost individually but also guarantees complete usage of the unused resources.

A packed set of customers are allocated huge bulk of resources from the infrastructure provider who offers volume discounts [23]. Thus, reducing both the cost spent by the broker in buying these resources and the individual cost of the customers. The advantages of the temporal multiplexing technique have been studied whereas,the spatial multiplexing technique is still being explored [6].

The quality dissatisfaction can have a negative impact on the profit maximization of the cloud service provider. In [15] we study how customer satisfaction can have a huge impact in deriving an optimal configuration for profit maximization. More and more people are drawn towards the concept of cloud computing because of its numerous advantages. Cloud computing can be a go to option for many startups, small businesses and people working in the public sectors because of no upfront capital investment on infrastructure and their maintenance [8]. Rather, these cloud users pay only for the resources that they use. As a result, many cloud service providers have emerged.

In this study, QoS (Quality of service) and PoS (Price of a service) is considered to develop a formula for customer satisfaction [32]. Then the correlation between the customer satisfaction and profit is observed to develop a profit maximization model. The QoS can be a measure of how well a service is moving ahead. If the service is attractive to a customer the number of customers will increase and thus, the rate of request arrival also increases which affects the customer satisfaction [3]. PoS is decided by the service provider. If the exact price of the service is lesser than the actual price of the service, it effects the customer satisfaction positively.

The profit of the cloud broker can be maximized by increasing the service capacity of the cloud service platform [7]. Therefore, an algorithmic approach is considered to find an optimal platform configuration to maximize the profit using customer satisfaction as a deriving factor.

The research [31] studies the pros and cons of considering the QoS (Quality of Service) requirements as a factor in determining the quality of the cloud service. SLA (Service Level Agreement) between the customers and providers contains the billing prices and penalties if the QoS is not delivered. Here QoS can be seen as a trait that guarantees certain level of performance. Different research theories are studied based on QOS.

A ranking methodology is proposed in a scenario where the cloud service provider is not able to deliver a service guaranteed to the consumer due to large number of pending requests [33].

Concepts like intercloud is still being explored. Hence, it is difficult for a cloud broker to design a dynamic methodology for resource management and profit maximization. [23] provides a framework for designing a dynamic model which considers the relinquish probabilities of the customers based on their prior data usage records and customer traits.

The study shows, the trust between the consumer and provider can be a contributing factor . The provider needs to multiplex the resources so that it addresses the resource underutilization issue [12]. And the user must carry out effective utilization as per the contract between them. Furthermore, factors like profit, service value, SLA, rate of utilization, quality degradation were also considered [14].

## 3. Proposed System

The aim of this project is to establish a system which helps in achieving maximum profit for a cloud broker and guarantees quality of service to its users.

### 3.1 Overview

The system allows a cloud broker to reserve required number of optimal resources from various cloud service providers. We use the Hill Climbing algorithm to select the best plan from all the existing plans for billing the resources during the negotiation between the BSP (Business Service Provider) and the user. The cloud broker will then assign these resources to various users based on their requests. The system also allows the users to make use of the obtained cloud resources. The M|D|c queuing model is adopted to queue the customer requests. The quality of service has been achieved by adopting these above methods in the system. This system allows the users to upload their files, images, and videos to the cloud.

### 3.2 System Architecture

Architecture diagrams are used to depict the relationship between different components of a system. It is essential to understand the complete concept of the system. The Fig. 1 shows the architectural diagram of the proposed system. In the system, the participant at every checkpoint uses the user interface (i.e., a website) to enter the related legitimate data to that checkpoint. This data is then stored into a database via a server.
The Components and sub-systems involved in the System Architecture are:

1. CSP (Cloud Service Provider)
2. Profit Maximization System
    i) BSP (Business Service Provider/Cloud Broker)
    ii) M/D/c Queuing Model
3. Users

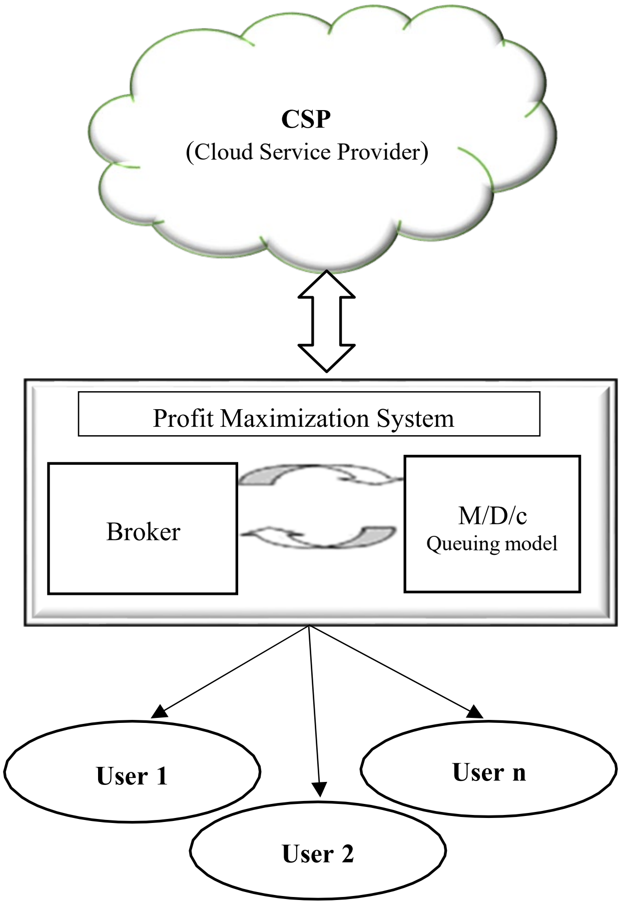


**Figure 1.** System Architecture Diagram

1. The CSP (Cloud Service Provider) provides resources to the broker based on the request from the broker. The request can be either long-term or short-term.

2. The Profit Maximization System consists of a BSP and an M/D/c queuing model. The BSP acts as a middleman between the users and the CSP. The BSP allocates long-term resources to the users based on their demand and maximizes his profit by using short-term plans. M|D|c queuing model is used to assign user requests into a queue whenever the BSP suspects a shortage of resources based on the service time and arrival time.

3. There is no upper limit on the number of users in this system. The user can request the resources from the BSP. User can also upload files into the cloud based on the size of the requested resources.

### 3.3 Data Flow Design

A DFD (Data Flow Diagram) is a graphical representation of the flow of data through an information system. DFD's can also be used for visualization of data processing. The Fig.2 shows the data flow diagram of the proposed system. A DFD comprises of an entity, process, data store, dataflow.

The entities involved in the DFD are:

1) User
2) BSP

The processes involved in the DFD are:

1) Process for the user:

a) The user signs in and the user data is processed for verification.
b) User uses the SLA catalogue to buy plans according to his/her needs.
c) User then uploads the files by making use of the allocated storage.

2) Process for the BSP:

a) BSP selects a suitable long-term plan from the cloud provider.
b) BSP purchases the long-term plan from the cloud provider pool.
c) BSP executes the requests from the customer. This involves permitting access to upload files.
d) When resources are used to full edge by the users the BSP allocates the extra resources using the short-term plans from the cloud provider.

## 4 Implementation

### 4.1 Components

Following components are required:

1. **Cloud service:** This provides the needed cloud resources to the broker. The plans can be either long-term or short-term. It presents the cloud broker with various options to choose from.

2. **Database:** It is used to store information about the requests received and resources utilized between the user and the broker and the broker and the cloud service provider.

3. **Broker Homepage:** This homepage presents the broker view. It contains resource request information, user information, different resource plans, resource utilization statistics, resource pricing information.

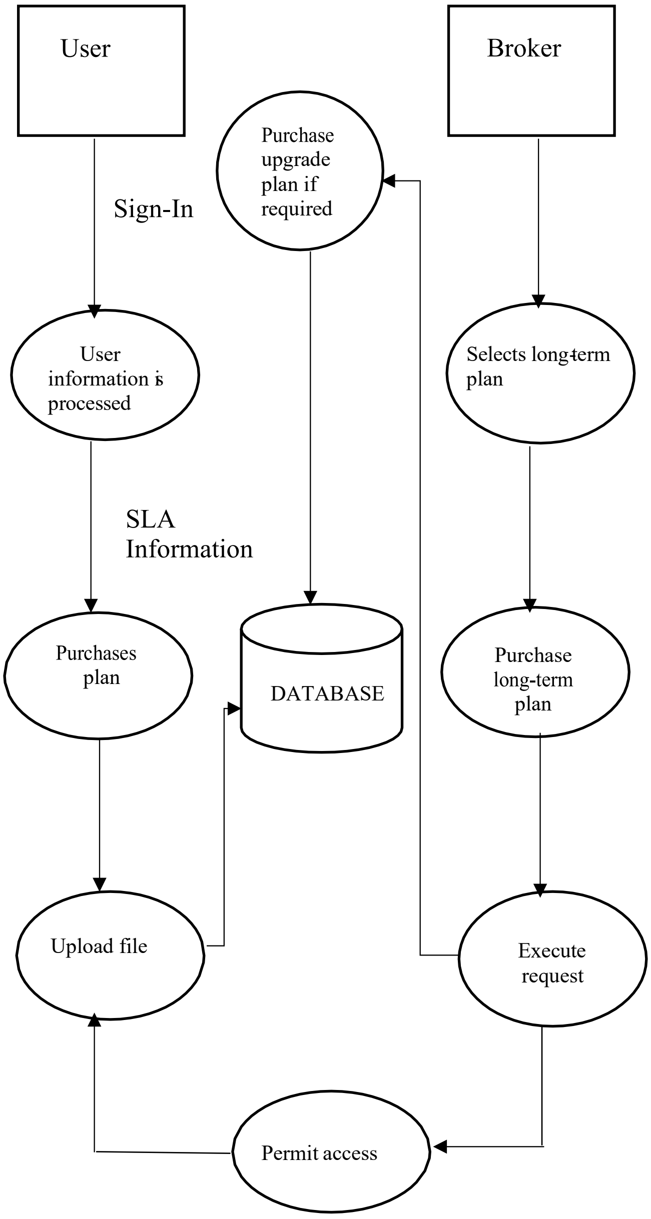


**Figure 2.** Data Flow Design

4. **Client Homepage:** This is the homepage of the users who register themselves with the broker. It contains information about the statistics of client resource usage. Clients use the resources either for storing files in the cloud or to deploy their applications.

## 4.2 General Constraints

- Internet connection is required
- Saving files in the cloud

## 4.3 Functional Requirements

1. **Login Module**: This module allows the users, the broker, and the CSP to login to the system. Table.1 gives the testcases for the login module.
2. **Registration Module:** This module allows the users to register themselves before using the system. Table.2 gives the testcases for the registration module.
3. **Upload Module**: This module allows the users to upload their data to the cloud.
4. **Decrypt Module**: This module allows the users to decrypt the files when required.
5. **Encrypt Module**: This module permits the users to encrypt the files that are to be uploaded to the public cloud.
6. **Dynamic Resource Calculation Module**: This module is used to keep track of the resource consumption by the BSP and the users dynamically.
7. **Notification Module:** This module is used to notify the broker whenever the resource utilization reaches a threshold.
8. **Operating Environment:** The developed algorithm is platform independent. The encoding is done in Java which is a platform independent language. The Graphical User Interface is created using JavaFX, a framework in Java Run-time environment (JRE) version 6 or above should be installed in the system to successfully run the simulation application and the algorithm.

Table 1. Testcases for Login Module

| SL. No. | Test case name | Valid Data | Remarks |
|---|---|---|---|
| 1 | Tst_mobile_textbox | 10-digit number | Pass |
| 2 | Tst_passwrd_textbox | Valid password | Pass |

Table 2. Testcases for Registration Module

| SL. No. | Test case Name | Description | Valid Data | Remarks |
|---|---|---|---|---|
| 1 | Tst_name_testbox | Test for valid name | Characters from 0-250 | Pass |
| 2 | Tst_mobileno_textbox | Test for valid mobile number | 10-digit number | Pass |
| 3 | Tst_email_textbox | Test for valid email address | Valid email address | Pass |
| 4 | Tst_address_textbox | Test for proper address | Proper address | Pass |
| 5 | Tst_pincode_textbox | Test for proper pin code | Proper pin code | Pass |

## 5. Result and Discussion

Below are the snapshots of the GUI for the proposed system. The system is programmed in Java where Spy log Enterprise is used as the database tool in NetBeans IDE 8.2.

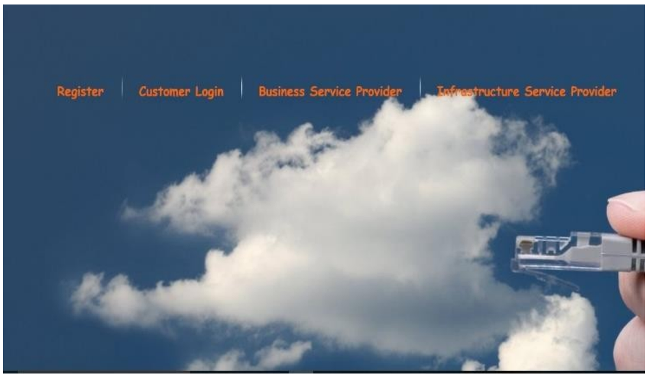
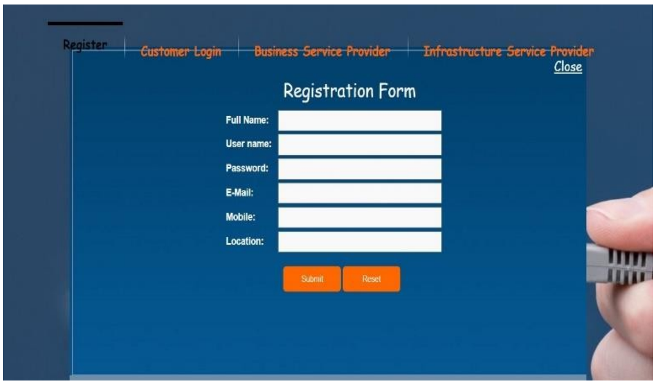


**Figure 3.** Home Page

This is the project home page. It has been divided into 3 modules. They are as follows:

- Infrastructure Service Provider (ISP)
- Business Service Provide or Cloud Broker (BSP)
- User or customer



Registration Form

Full Name:

User name:

Password:

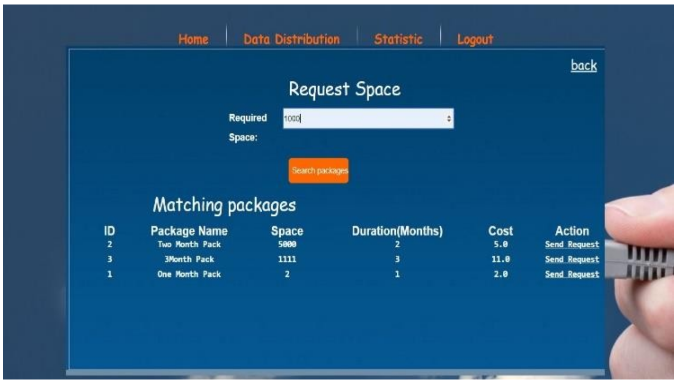
E-Mail:

Mobile:

Location:

**Figure 4.** Customer Registration Page

Customers who wish to utilize the resources must initially register themselves with the BSP. Registration details include username, password, email, mobile number, and address. There are similar registration views for the cloud service provider and the cloud broker. Once registered, the customers can make use of the services available in the system portal. There are similar login page views for all the three entities such as the BSP, the CSP and the user.



Request Space

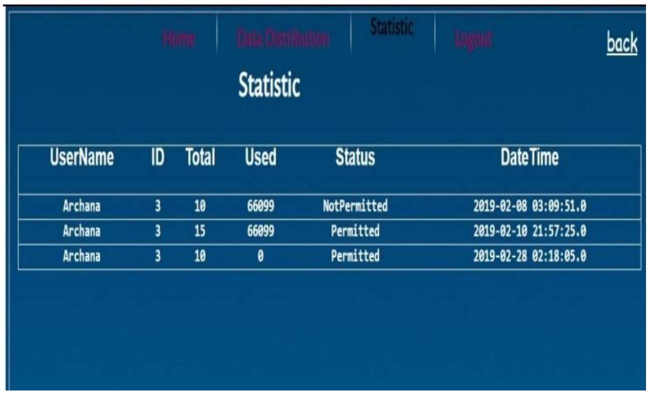
Required Space: 1000

Matching packages

| ID | Package Name | Space | Duration(Months) | Cost | Action |
|---|---|---|---|---|---|
| 2 | Two Month Pack | 5000 | 2 | 5.0 | Send Request |
| 3 | 3Month Pack | 1111 | 3 | 11.0 | Send Request |
| 1 | One Month Pack | 2 | 1 | 2.0 | Send Request |

**Figure 5.** Customer Resource Request Space

This is the customer resource request page. By clicking on the 'search package' button depending on the volume of the requested resource, the Hill-Climbing algorithm will present a list of matching packages from the set of available packages. This not only reduces the time taken by the consumer to go through the entire catalogue but also enhances the quality of service. Thus, the algorithm provides the best local solution available to the users. The customer can also view the access permissions assigned to him by the cloud broker. The other entities presented to the users are as follows: package id, package name, space requested by the user for allocation, duration in months, cost of the available packages, and the permission list. The user can forward his request by clicking on the ' send request' button.

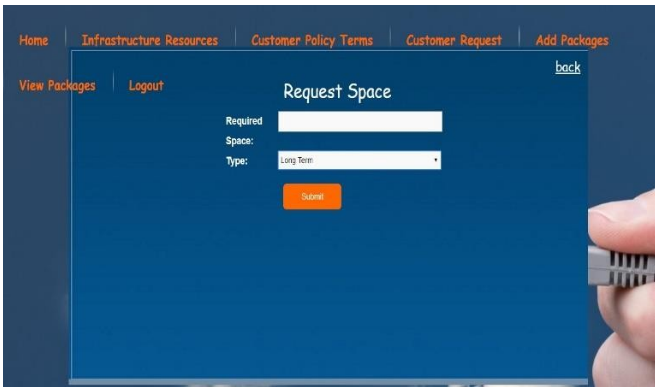


Statistic

| UserName | ID | Total | Used | Status | DateTime |
|---|---|---|---|---|---|
| Archana | 3 | 10 | 66099 | NotPermitted | 2019-02-08 03:09:51.0 |
| Archana | 3 | 15 | 66099 | Permitted | 2019-02-10 21:57:25.0 |
| Archana | 3 | 10 | 0 | Permitted | 2019-02-28 02:18:05.0 |

**Figure 6.** Customer Statistic Page

From the customer statistic page, the users can obtain details about the requested resources and the utilization index. Customers can find out if they are permitted by the BSP to consume the requested resources. Each registered customer has been allocated with a unique ID. Furthermore, the page provides information on date and time of the requested resources, total resource allocated to the user and the amount of it utilized by the user.



Request Space

Required Space:

Type: Long Term

**Figure 7.** BSP Resource Request Space

This is the cloud broker's resource request space. Whenever the broker gets notified about the resource shortage, he purchases additional resources from the infrastructure vendor. By clicking on the 'send' button the cloud broker places a request for resource allocation where he is presented with either short-term schemes or long-term schemes. Initially, the broker invests in a huge bulk of long-term schemes at a greater volume discount from the vendor. After a certain period, he faces lack of resources due to fulfilling the necessary business needs of the customers, he then purchases the short-term schemes to scale the resources accordingly. This is then multiplexed among 'n' customers to prevent underutilization of resources thus, improving both the efficiency and net profit of the cloud broker.

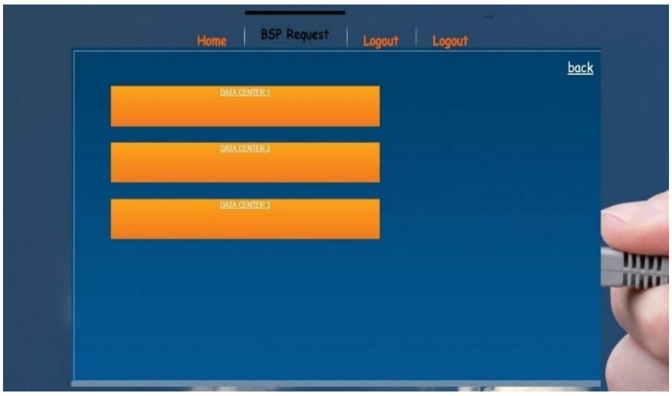


**Figure 8.** ISP Multi-Server System

This shows the data centers available to hold the requests from BSP. Here there are 3 data centers. This shows the usage of the multi-server queueing model in our system. M/D/c is the queuing model used in the given system. Each data center has a particular queue size. For example, if data center 1 has a queue size of 3 then 3 allocation requests can be serviced there. All the requests from BSP will be placed in either of the 3 data centers based on the empty slots available.

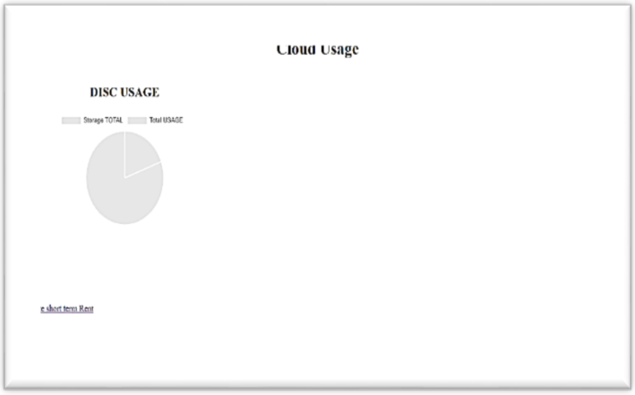
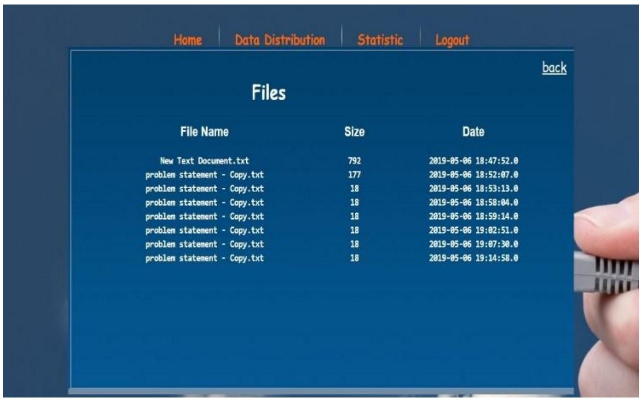


**Figure 9.** Disk Utilization Graph

This is the BSP disk utilization graph. This represents the analysis part of the cloud broker. The graph is mainly observed by the broker to use the multiplexing scheme in consumer resource allocation. Broker can both manually analyse the graph and, receive automated notifications upon reaching a threshold utilization index. The pie graph has two divisions:

**Storage Total:** This represents the total storage that ISP has allocated to BSP which has not been used by the BSP or the consumer.

**Total Usage:** This represents the portion of the total resource utilized by the consumers.



Files

| File Name | Size | Date |
|---|---|---|
| New Text Document.txt | 792 | 2019-05-06 18:47:52.0 |
| problem statement - Copy.txt | 177 | 2019-05-06 18:52:07.0 |
| problem statement - Copy.txt | 18 | 2019-05-06 18:53:13.0 |
| problem statement - Copy.txt | 18 | 2019-05-06 18:58:04.0 |
| problem statement - Copy.txt | 18 | 2019-05-06 18:59:14.0 |
| problem statement - Copy.txt | 18 | 2019-05-06 19:02:51.0 |
| problem statement - Copy.txt | 18 | 2019-05-06 19:07:30.0 |
| problem statement - Copy.txt | 18 | 2019-05-06 19:14:58.0 |

**Figure 10.** Uploaded File View Page

On approval from the BSP, a customer can upload his files onto the cloud. Customer must first select the file by clicking on 'choose file' option, once the file is selected, then he can upload it by clicking on the 'upload' option. This page provides a list of files uploaded by the user. There is separate page for each individual user, so that it only contains a list of files uploaded by that specific user. It also gives few other information about the uploaded file such as file size, date, and time stamp.

## 5.1. Technologies Used

The quality of service is achieved in this system using the following technologies:

**M/D/c Queuing Model:** In M/D/c queueing model the requests are accepted based on:

- Arrival time
- Service time

Quality of service is achieved by increasing the number of servers present in the queue. Here 'c' stands for the number of servers in the queue that is served in first-in , first-out manner.Thus, these servers help in serving the jobs faster. The request arrival is based on Poisson Distribution and exponentially distributed service time. The requests can be serviced by multiple servers at the same time. Thus, many requests can be queued and responded to by the servers unlike the M/m/1 model.

Erlang presented that when $\rho = (\lambda\ D)/c < 1$, the waiting time distribution has distribution F($y$) given by [17]

$$F(y) = \int_0^\infty F(x+y-D)\frac{\lambda^c x^{c-1}}{(c-1)!}e^{-\lambda x}dx, y \geq 0\ c \in N$$

Crommelin presented that, writing $P_n$ for the stationary probability of a system with $n$ or fewer customers given by [18]

$$P(W \leq x) = \sum_{n=0}^{c-1} P_n \sum_{k=1}^{m} \frac{(-\lambda(x-kD))^{(k+1)c-1-n}}{((K+1)c-1-n)!} e^{\lambda(x-kD)}, mD \leq x < (m+1)D$$

**Hill Climbing Algorithm**: Hill Climbing algorithm is a method that is used to solve the optimization problems. This method starts with a sub optimal solution and this solution is optimized repeatedly until a best solution is found. This final optimized solution is compared with all the previously obtained results. Hence, it helps in obtaining the next best solution for solving the problem. In this project, we use the algorithm to select the best plan among the available plans. We have implemented hill climbing algorithm in our project to find out the best resource billing package available among all other available packages. The goal here is to make it easier for the users to obtain the packages without having to work through the entire catalogue. This not only reduces the time taken by the customer for negotiation, but also enhances QOS.

Three key points are considered when using Hill Climbing algorithm as an approach to select the best available package.

1) It is a heuristic search. The Hill climbing algorithm does not provide an optimal global solution but instead provides the best available or local solution to maximize or minimize the given problem at hand within a good time frame.

2) It is a variant of the Generate-and-Test algorithm.Step 1: It generates possible solution.
Step 2: It analyses the generated solution and checks whether it is the expected solution.
Step 3: If it is the required solution it stops. If not, it proceeds with a new solution.
Thus, the feedback is taken from the test procedure and utilized by the generator to come up with next solution in the search space.

3) It uses the Greedy approach to arrive at the solution.
Step 1: Select solution 1
Step 2: Select solution 2
Step 3: Compare both the solutions.
Step 4: If solution 1 is better, select solution 1. Step 5: If solution 2 is better, select solution 2. It selects

the best possible way to reach the required solution.

The proposed pseudo code for the Hill-Climbing algorithm:

```
Step 1: Select initial package as S.
Step 2: Evaluate S, if it the best available price then stop, else set the initial package as the current package.
Step 3: Select the next available package as R from the neighboring packages.
Step 4: Compare the prices of S and R
Step 5: If R is a better option, then set R as the current package S and return to Step 3.
Step 6: If R is not a better option, then set S as the current package and return to Step 3.
Step 7: Loop until all packages are explored.
```

The profit in this system is earned by using these techniques:

**Double Renting Scheme**: In the proposed system we make use of a double renting scheme. The schemes that we employ here are long-term and short-term. Use of long-term schemes alone in cloud service leads to wastage of resources due to underutilization and thus, effects the quality of service. To address this issue, we employ double renting scheme. Double renting scheme is a combination of both long-term and short-term schemes which improve both poor quality of service and the mass wastage of resources.

**Resource Multiplexing:** The concept defines a possibility of serving the customer needs through virtualization to reap the profit benefits. The main idea behind resource multiplexing is that the cloud broker allocates the resources requested by the customers even though he lacks infrastructure to do so. When the resource requests reach a peak point based on the usage graph the broker purchases the required infrastructure for short-term and thus, addresses the issue of resource wastage. By allocating resources to customers irrespective of the acquired infrastructure the broker profits from this set-up

## Conclusion

A business service provider is an intermediate unit between a cloud provider and a consumer, who purchases reserved instances from the cloud providers for an extensive stretch of time and reevaluates them as on-demand at a lower cost with respect to what the cloud service provider charges. Due to the lower service cost and enhanced quality of service when contrasted with public cloud, the cloud broker can diminish the expense for the customers. The virtual asset platform is accommodated as an M/D/c queuing model, and a profit amplification issue is implicit in which the influencing factors are investigated. The proposed system alerts the broker about resource shortage through allocated storage graph, this project can be further enhanced by implementing graphs for profit and time taken for handling the service requests.

## Acknowledgements.

This project has been carried out in CoE Digital Forensics Intelligence and Cyber Security, supported by VGST, Govt of Karnataka.

## References

[1] Laghari, A. A., He, H., Khan, A., Kumar, N., & Kharel, R. (2018). Quality of experience framework for cloud computing (QoC). IEEE Access, 6, 64876-64890.

[2] Laghari, Asif Ali, Awais Khan Jumani, and Rashid Ali Laghari. Review and State of Art of Fog Computing. Archives of Computational Methods in Engineering (2021): 1-13.

[3] Yadav, Rahul, Weizhe Zhang, Keqin Li, Chuanyi Liu, and Asif Ali Laghari. Managing overloaded hosts for energy-efficiency in cloud data centers. Cluster Computing (2021): 1-15.

[4] Karim, Sajida, Hui He, Arif Hussain Magsi, and Rashid Ali Laghari. Quality of service (QoS): measurements of image formats in social cloud computing. Multimedia Tools and Applications (2020): 1-26.

[5] Kumar, V., Laghari, A. A., Karim, S., Shakir, M., & Brohi, A. A. (2019). Comparison of fog computing & cloud computing. Int. J. Math. Sci. Comput, 1, 31-41.

[6] Halepoto, I. A., Memon, M. S., & Parveen, S. (2017). Analysis of quality of experience frameworks for cloud computing. IJCSNS, 17(12), 228.

[7] Shafiq, M., & Khan, A. (2018). Assessment of quality of experience (QoE) of image compression in social cloud computing. Multiagent and Grid Systems, 14(2), 125-143.

[8] Laghari, A. A., He, H., Karim, S., Shah, H. A., & Karn, N. K. (2017). Quality of experience assessment of video quality in social clouds. Wireless Communications and Mobile Computing, 2017.

[9] V. Shetty, S. Vishwakarma, Harisha and A. Agrawal, Design and implementation of video synopsis using online video inpainting, 2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, pp. 1208-1212.

[10] Rai Bellipady, S., Shetty, S.M. & Airbail, H. Quality of Information Analysis in WSN: An Application in BASN. J Ambient Intell Human Comput (2019).

[11] Karkera, T., Singh, C. Autonomous Bot Using Machine Learning and Computer Vision. SN COMPUT. SCI. 2, 251 (2021).

[12] Laghari, A. A., He, H., Shafiq, M., & Khan, A. (2017, May). Impact of storage of mobile on quality of experience (QoE) at user level accessing cloud. In 2017 IEEE 9th international conference on communication software and networks (ICCSN) (pp. 1402-1409). IEEE.

[13] Memon, K. A.., Halepoto, I. A., & Khan, A. (2019). Quality of experience (QoE) in cloud gaming models: A review. Multiagent and Grid Systems, 15(3), 289-304.

[14] Laghari, Asif Ali, Hui He, Muhammad Shafiq, and Asiya Khan. Assessing effect of Cloud distance on end user's Quality

of Experience (QoE). In 2016 2nd IEEE international conference on computer and communications (ICCC), pp. 500-505. IEEE, 2016.

[15] Laghari, A., He, H., Laghari, R., Khan, A., & Yadav, R. (2019). Cache Performance Optimization of QoC Framework. EAI Endorsed Transactions on Scalable Information Systems, 6(20).

[16] Wagan, A., & Umrani, A. (2020). Effect of Packet Loss and Reorder on Quality of Audio Streaming. EAI Endorsed Transactions on Scalable Information Systems, 7(24).

[17] Schmiedeke, P. Kelm, and T. Sikora, DCT-based features for categorisation of social media in compressed domain, in Proc. IEEE 15th Int. Workshop Multimedia Signal Process. (MMSP), Pula, Croatia, Sep. 2013, pp. 295–300.

[18] Junwei Cao, Senior Member, IEEE Kai Hwang,Fellow, IEEE Keqin Li, Senior Member, IEEE Albert Y. Zomaya, Fellow, IEEE. Optimal Multi server Configuration for Profit Maximization in Cloud Computing. IEEE transactions on parallel and dis- tributed systems, 2012.

[19] Meixuan Li, Yu-E Sun, He Huang, Jing Yuan, YangDu, Yu Bao, Yong long Luo. Profit Maximization Resource Allocation in Cloud Computing with Performance Guarantee. BPB Publications, 2015.

[20] Rui Zhang, Kui Wu†, Minming Li, Jianping Wang. Online Resource Scheduling under Concave Pricing for Cloud Computing. Department of Computer Science,University of Texas at Dallas, McGraw Hill, 2017

[21] Sergio Nesmachnow and Yong Yan. Competitive Cloud Resource Procurements via Cloud Brokerage. IEEE transactions, 2013.

[22] XuanjiaQiu, Chuan Wu, Hongxing Li, Zongpeng Li and Francis C.M. Lau. Federated Private Clouds via Broker's Marketplace: A Stackelberg-Game Perspective. IEEE transactions on parallel and distributed systems, 2014.

[23] Mohammad Aazam, Eui-Nam Huh, Members, IEEE,Marc St-Hilaire, Senior Member, IEEE, Chung-Horng Lung, IoannisLambadaris, Members, IEEE. Cloud Customer's Historical Record Based Resource Pricing. IEEE transactionson parallel and distributed systems, 2015.

[24] Javier Alsina, Santiago Iturriaga, Sergio Nesmachnow. Virtual Machine Planning for Cloud Brokering Considering Geolocation and Data Transfer. IEEE transactions, 2016.

[25] Alexei A. Gaivo ronski, DarijusStrasunskas, Per J. Nesse, and Stein Svae. Beyond Best Effort: Choosing Connectivity Portfolio for Cloud Brokering Platform by Risk/Profit Tradeoff. IEEE transactions, 2013.

[26] Shujin Ye, Hai Liu, Yiu-Wing Leung, and Xiaowen Chu. Reinsurance-Emulated Collaboration Mechanism in Cloud Federation. IEEE transactions, 2017.

[27] Xin Jin, Yu-Kwong Kwok, and Yong Yan. Competitive Cloud Resource Procurements via Cloud Brokerage. IEEE transactions, 2013.

[28] Mukesh Kumar Gupta and Annnappa B in. Virtual Machine Planning for Trusted Partner Selection in Broker Based Cloud Federation. IEEE transactions, 2016.

[29] Gourav Saha and Ramkrishna Pasumarthy, DarijusStrasunskas, Per J. Nesse, and Stein Svae. Maximizing Profit of Cloud Brokers under Quantized Billing Cycles: a Dynamic Pricing Strategy based on Ski-Rental Problem. IEEE transactions, 2015.

[30] Teerawat Kumrai, Kaoru Ota, Mianxiong Dong, JayKishigami, and Dan Keun Sung. Multi objective Optimization in Cloud Brokering Systems for Connected Internet of Things. IEEE transactions, 2016.

[31] N. Ani Brown Mary and K. Jayapriya. An Extensive Survey on QoS in Cloud computing. IEEE transactions, 2017.

[32] Jing Mei, Kenli Li and Keqin L. Customer- Satisfaction-Aware Optimal Multi server Configuration for Profit Maximization in Cloud Computing. IEEE transactions,2015.

[33] Mr. K. Saravanam and M. Lakshmi Kantham. An enhanced QoS Architecture based Framework for Ranking of Cloud Services. Published in the Proceedings International Journal of Engineering rends and Technology (IJETT) – Volume 4 Issue 4 – April 2013.

[34] Franx, G. J. (2001). A simple solution for the M/D/cwaiting time distribution. Operations Research Letters. 29 (5): 221–229.

[35] Crommelin, C.D. (1932). Delay probability formulas when the holding times are constant. P.O. Electr. Engr. *J.* 25:41–50.

[36] Singh, Chandra, K.V.S.S.S.S. Sairam, and M.B. Harish. Global Fairness Model Estimation Implementation in Logical Layer by Using Optical Network Survivability Techniques. International Conference on Intelligent Data Communication Technologies and Internet of Things. Springer, Cham, 2018.