**I J C T A**, 9(17) 2016, pp. 8527-8532
© International Science Press

# SLA Mechanism for Tenant Submission in Multiple Queues

**Pankaj Deep Kaur*** **and Jashanjot Singh***

**ABSTRACT**

Cloud computing is used to share the services and applications that are hosted on internet to reduce the computational power and processing time Virtualization is the key concept behind cloud computing using which applications and services can be used by end users without any extra overhead of processing. This will also reduce the cost of deployment and installation of hardware and software. In this paper, we are proposing a flexible mechanism of resource access without starving the other tenants and allow the fulfilment of SLA, ensure higher CSL and hence well organised utilization of the resources as the service providers allow for a control accessed.

***Keywords:*** SLA, Tenant submission, multiple queues.

## I. INTRODUCTION

Cloud computing is related to the services and applications which run on a shared network making use of virtualized resources, are accessed with the help of internet protocols and networking standards. It is noted by the fact that the resources are virtual and unlimited and also the details of the real system on which the software usually runs are withdrawn from the user. In Cloud computing all the particulars of the system implementation are abstracted from the users and the developers. Applications run on the physical systems that are not specified and also the data is stored in locations which are unknown. In Cloud computing the systems are virtualized by pooling and sharing resources. Costs are assessed on a measured basis. Basically Cloud computing is a representation in which a pool of systems is connected in the public or private networks in order to provide scalable infrastructure dynamically for application, file storage and data and with this technology the cost of computation, hosting application, storing content and delivery is reduced.

Cloud computing is a practical approach to observe direct cost benefits and it has the potential to change a data centre from a capital-investment set up to a varying priced environment. Cloud resources are shared by multiple users and are also dynamically reallocated per demand. There are various types of services offered by the cloud providers that can be listed into three categories i.e. SaaS stands for Software as a Service based on internet and the applications service provider which are hosted on the cloud. PaaS stands for Platform as a Service is used to provide the platform for designing, developing, building and testing the software applications and are hosted on cloud infrastructure. Infrastructure as a Service (IaaS): This pay per use model includes the services like storage, database management and compute capabilities are offered on demand.

Cloud computing is a developed computing terminology based on functioning and utility of computing resources. Cloud computing includes deploying several groups of servers that are remote and software networks that allow the centralized data storage and also online access to the computer services or resources. Cloud computing also focuses on improving the efficiency of the shared resources. The cloud resources are

* Department of Computer Science and Engineering, Guru Nanak Dev University, Regional Campus Jalandhar, Punjab, India, *E-mail-pankajdeepkaur@gmail.com; jashan.goraya@live.com*

shared by multiple users and are also dynamically reallocated per demand. Cloud Computing refers to the transfer of computing resources over internet. Instead of keeping the data on personal hard drive or updating applications as per need, a service is used over the internet at any other location giving rise to privacy involvement. With cloud computing, multiple users can access a single server to obtain and update the data without buying licenses for the various applications. It virtualizes the systems by pooling and sharing the resources, systems and storage can be provisioned as required from a centralized infrastructure.

The rest of the paper is organized as follows. All the existing techniques carried out by various authors are explained in section II. Proposed Technique and Experimental results are presented in section III. Conclusion and Future work are given in section IV.

## II. LITERATURE SURVEY

Kurmus et al. [1] proposed to maintain the increasing level of multi-tenancy such as hardware level, hypervisor level, OS level and application level. Perceptively, the more the level of multi-tenancy the more it becomes easy to have a resource efficient implementation and design but at the same time it also becomes harder to isolate the tenants securely from each other. Vashishta et al. [2] proposed a novel system to serves the Multi-tenancy as one of the key characteristic of SaaS applications since the SaaS providers can easily carry out the functionality and the delivery cost is also reduced for the large number of tenants. Liu et al. [3] proposed the procedure of requesting and accepting cloud services by the users must be done via the Internet. The flows of user requests for cloud services are first received by the regulator and the shaped flows are then multiplexed through the FIFO multiplexing module and finally the output from the multiplexing module goes through the admission control phase producing the accepted flows. Elnikety et al. [4] proposed admission control and manages the overload that used to inhibit systems from being overloaded in the presence of determined or momentary overload. Two approaches to do this are reducing the amount of work and confrontation. Chidambaram et al. [5] proposed the Schedule based Fair Queue Weight is initiated in order to schedule the software services based on weighted approximated processor sharing. The weight is then allocated by the tenant to cloud structure based on the time period the request is made. After that the weight is considered to determine the tenant’s positional point to ensure the delivery of software service using cloud. The dynamic admission control helps to obtain the positions of several dynamic set of tenants. Kounev et al. [6] proposed advanced solutions to provide every tenant with a different application instance installed in separate virtual machines with a defined set of resources if the performance impact between the tenant’s needs be low. Most of the existing works concentrate on the SLA aware placement of tenants onto the existing application nodes. Chen et al. [7] rather than admitting the requests until the resources are exhausted as a condition for admission control the resources are dynamically reserved to requests based on the cost-based scheme so that the system is able to maximize the total reward achieved by the system in response to the workload changes in the environment. He et al. [8] proposed the admission control strategy is based essentially on Effective Bandwidth which termed as EB-Based Admission Control or EBBAC. This means only those cloud service requests that meet the admission control will be accepted otherwise they will be denied. For such accepted requests the required buffer size are calculated which ensures the delay constraint of all the cloud services is less than D (the delay constraint) with the same amount of cloud services that are accepted. A set of maximum values which satisfies the proposed equation represents the maximum amount of cloud services that a node can serve. Wu et al. [9] proposed the prevalent objective of the SaaS providers to minimize the cost and maximize the customer satisfaction level (CSL). The cost incorporates the infrastructure cost, penalty cost caused by SLA violations and the administration operation cost. Customer satisfaction level depends on to what level the SLA is satisfied. The platform layer uses the admission control to explain and analyze the QoS parameters of the user and decides whether to accept or deny the request based on the availability, capability and price of VMs.

## III. EXPERIMENT AND RESULT

Result is calculated by executing the proposed algorithm using Matlab tool. The requests from multiple tenants are submitted to the SLA admission controller which checks for each random job that arrives and verifies its order of execution depending upon the SLA type. The scenarios for our proposed system include the single queue and multiple queue structure for performing the required operations to analyze the results.

### 3.1. Single Queue

The processing starts with a trigger to the data centre, the VM followed by the application attachment. Having triggered the above three buttons, the actual processing starts by clicking on start process which gives a dummy preview of an actual cloud server and the corresponding resource usage and load along with the data being uploaded on the cloud environment.

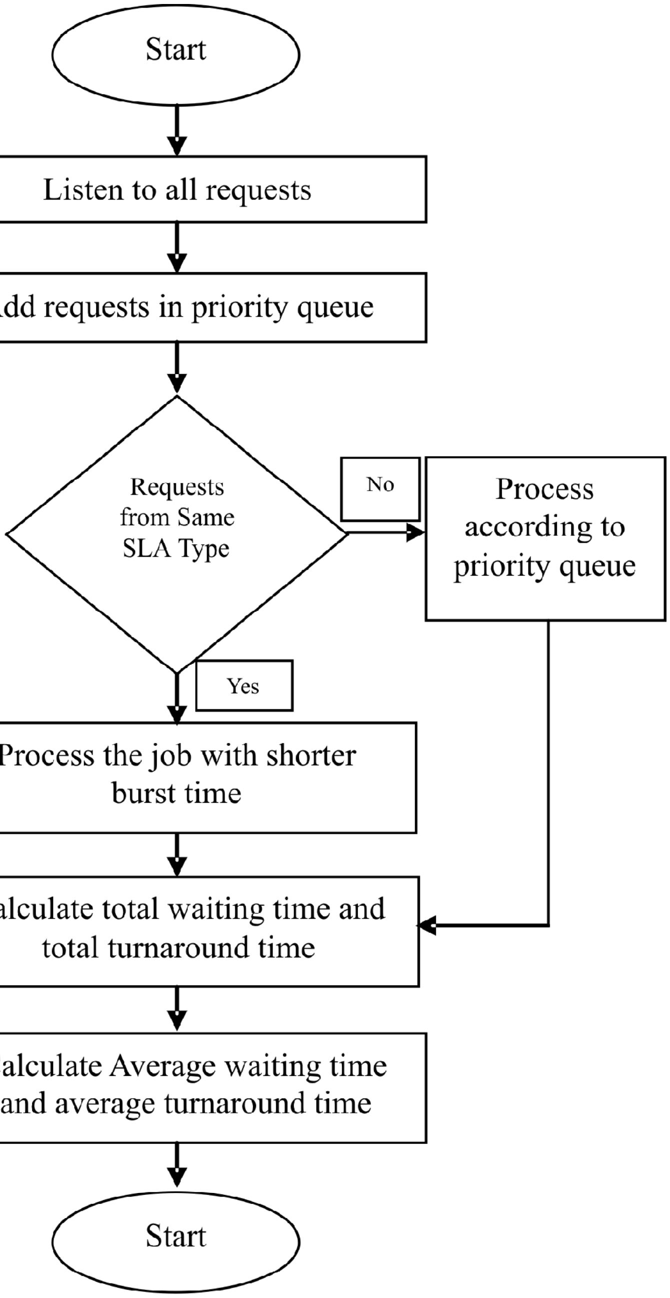
Start
Listen to all requests
Add requests in priority queue
Requests from Same SLA Type
No
Process according to priority queue
Yes
Process the job with shorter burst time
Calculate total waiting time and total turnaround time
Calculate Average waiting time and average turnaround time
Start

**Figure 1: Proposed systems flowchart**

**Table 1**
**Single Queue**

| | *Single Queue* | | |
|---|---|---|---|
| *Process* | *Burst Time* | *Waiting Time* | *Turn Around Time* |
| P1 | 200 | 0 | 200 |
| P2 | 1100 | 200 | 1300 |
| P3 | 1200 | 1300 | 2500 |
| **P4** | **300** | **2500** | **2800** |

After all the requests have been submitted and processed the overall waiting time of the single queue system is estimated on the basis of burst time of each job initiated from its respective tenant. The average waiting time and the average turnaround time is calculated and returned as output and is then compared with the output as that of the multiple queuing system.

**Table 2**
**Gold Queue**

| *Process* | *Burst Time* | *Waiting Time* | *Turn Around Time* |
|---|---|---|---|
| P1 | 350 | 0 | 350 |
| P2 | 400 | 0 | 400 |
| P3 | 200 | 0 | 200 |
| **P4** | **300** | **200** | **500** |

**Table 3**
**Silver Queue**

| *Process* | *Burst Time* | *Waiting Time* | *Turn Around Time* |
|---|---|---|---|
| P1 | 1400 | 0 | 1400 |
| P2 | 1700 | 0 | 1700 |
| P3 | 1500 | 0 | 1500 |
| **P4** | **1100** | **1400** | **2500** |

**Table 4**
**Bronze Queue**

| *Process* | *Burst Time* | *Waiting Time* | *Turn Around Time* |
|---|---|---|---|
| P1 | 1200 | 0 | 1200 |
| P2 | 1100 | 0 | 1100 |
| P3 | 1500 | 0 | 1500 |
| **P4** | **600** | **1100** | **1700** |

After simulation of the single queuing system has been executed which serves the requests in the order of their arrival the jobs are then submitted in the similar order to the multiple queue which prioritizes the requests according to the type of SLA done with the tenant to which it belongs.

This clearly shows the difference in values that are obtained for both the queues and hence verifies that the average waiting time and average turnaround time value in case of single queue is more as compared to the multiple queue which serves each job on a priority basis as defined by their respective SLAs.

**Table 5**
**Average waiting time**

| Average Waiting Time | |
|---|---|
| *Single Queue* | *Multiple Queue* |
| **1000** | **720** |

**Table 6**
**Average turn around time**

| Average Turn Around Time | |
|---|---|
| *Single Queue* | *Multiple Queue* |
| **2100** | **1100** |

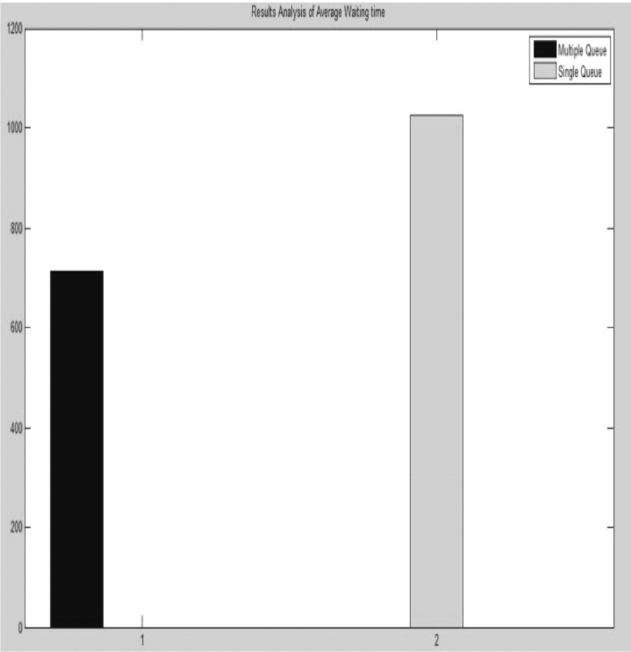

**Figure 2: Average waiting time for single and multiple queues**

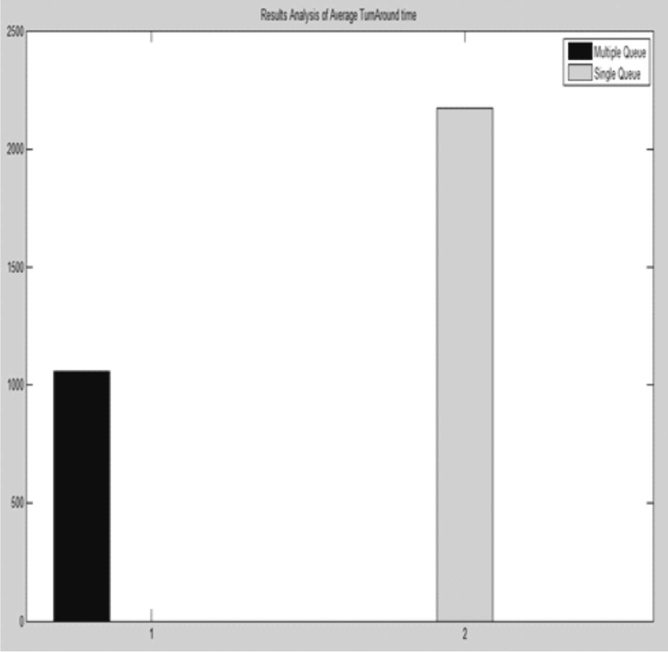

**Figure 3: Average turnaround time for single and multiple queues**

## IV. CONCLUSION AND FUTURE WORK

In a multi tenant environment involving requests coming to the cloud from various tenants at the same time raise an issue of compromise with the performance and the SLA associated with the intended user. Even with the prioritization of requests on the basis of cost-effective admission control and scheduling algorithms to maximize the SaaS provider's profit; still being unable to satisfy the efficient resource utilization and SLA in parallel. A solution is proposed which allows for the service providers to manage the data- centers and their resources efficiently by providing a control access to the tenants requesting for their services. The requests are received and prioritized on the basis of SLA which the intended tenant holds.

The future scope of this study lies in proving the method which will provide a better solution to the multi-tenant cloud environments with the help of which it will be easier for the service providers to reduce the waiting time. The SLA violations will be lesser and there will be an improved performance to some extent which will ensure that the multiple tenants do not impose an overhead on the network due to the delays and waiting time. The queuing method which is followed in the proposed method ensures an optimized way of handling multiple jobs by simultaneously abiding by the SLA.

## REFERENCES

[1] Anil Kurmus, Moitrayee Gupta, Roman Pletka, Christian Cachin, and Robert Haas, “A Comparison of Secure Multi-tenancy Architectures for Filesystem Storage Clouds”, IBM Research – Zurich.

[2] Avneesh Vashistha, Pervez Ahmed, “SaaS Multi-Tenancy Isolation Testing-Challenges and Issues”, International Journal of Soft Computing and a Engineering.

[3] Yanbing Liu, Yunlong He and Jun Huang, “ Admission Control for Aggregate Flow in Cloud Computing”, School of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China.

[4] Sameh Elnikety, Erich Nahum, John Tracey, Willy Zwaenepoel, “A Method for Transparent Admission Control and Request Scheduling in E-Commerce Web Sites”, School of Computer and Communication Sciences EPFL CH 1015 a Lausanne, Switzerland.

[5] Chidambaram, chandrasekar, “a software service model using schedule based fair queue weight for dynamic admission control on cloud infrastructure”, karpagam university, computer science department.

[6] Prof. Dr. Samuel Kounev, Prof. Dr. Ralf Reussner, “Performance Isolation in a Multi-Tenant Applications”, Rouven Krebs aus Speyer, Deutschland.

[7] Ing-Ray Chen and Naresh Verma, “A Cost-Based Admission Control Algorithm for Digital Library Multimedia Systems Storing Heterogeneous Objects”, Department of Computer Science, Virginia Polytechnic Institute and State University.

[8] Yunlong He, Jun Huang, “A Novel Admission Control Model in Cloud a Computing”, The Pennsylvania State University, Abington.

[9] Linlin Wu, Saurabh Kumar Garg, Rajkumar Buyya, “SLA-based admission control for a Software-as-a-Service provider in Cloud computing environments”, (CLOUDS) Laboratory, Department of Computer Science and Software Engineering, The University of Melbourne, Australia.