ISSN : 2231-0711

The Queueing Theory in Cloud Computing to Reduce the Waiting Time

T. Sai Sowjanya*, D.Praveen,K.Satish, A.Rahiman
Department of Computer Science and Engineering, Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada, INDIA

***Abstract*-- Cloud computing is an emerging technology of business computing and it is becoming a development trend[r].. The process of entering into the cloud is generally in the form of queue, so that each user need to wait until the current user is being served. In the system, each Cloud Computing User (CCU) requests Cloud Computing Service Provider (CCSP) to use the resources, if CCU(cloud computing user) finds that the server is busy then the user has to wait till the current user completes the job which leads to more queue length and increased waiting time. So to solve this problem, it is the work of CCSP's to provide service to users with less waiting time otherwise there is a chance that the user might be leaving from queue. CCSP's can use multiple servers for reducing queue length and waiting time. In this paper, we have shown how the queuing model, M/M/S model is used for multiple servers to reduce the mean queue length and waiting time**.

Keywords: cloud computing, queuing theory, waiting time, queuing length

I. INTRODUCTION

Multiple tenants enter into the Cloud requesting for various resources or services. In traditional approach of Cloud computing, only a single server serves all the entire Cloud Computing Users (CCU's) and so the overload on that single server increases which affects the system performance.

So for solving this problem a new approach of having multiple servers in the cloud is presented in this paper. Cloud computing can be used for Software as a service (SaaS).By using multiple servers, the cloud computing users can be divided into groups based on the request of service or resource. Among the multiple servers available, each server considers the jobs of each group and so the software as a service(SaaS) is being shared.

The main advantage of having multiple servers in Cloud computing is, the system performance increases effectively by reducing the mean queue length and waiting time than compared to the traditional approach of having only single server so that the CCU's need not wait for a long period of time and also queue length need not be large.

The approach for the implementation of this paper is as follows. Initially the difference between and also the pros and cons of Single-user architecture and Multi-user architecture are considered briefly. In the next session the concept of Queuing theory is been used for understanding the input process. Queuing models are used for the representation of the entire process of queuing in mathematical manner. Next by using Kendall's notation the queuing models are represented for single server and for multiple server. Numerical results and graphs are used finally for showing the better results with multiple servers than having a single server. These numerical results clearly shows that mean queue length and waiting time are been reduced by using multiple servers.

II. CLOUD COMPUTING

Cloud computing refers to the provision of computational resources on demand via a computer network. There are three kinds of cloud services model, namely, Software as a Service (SaaS), Platform as a Service (PaaS) and Cloud Infrastructure as a Service (laaS).

In Single-User architecture, as the name suggests, the software can be tailored for each CCU separately. The system houses the data for each company on a separate server. Each user runs its own copy of the software. The separation can be either virtual (virtual machines on a same server) or physical (running on separate machines).

When a single server is used for multiple organizations or users, the code cannot be modified and customized as changes made for one user could affect the other tenants using the same server. Hence, use of meta-data to customize by configuration is done by each organization to suit their specific needs.

In a Multi-User architecture, the data from multiple companies are placed on the same server generally separated from each other via a simple partition that prevents data to migrate from one company to another. As

The data is housed on the same server, each company using the software is running the same basic application, with the same basic functionality and with the same limited

The configuration capabilities of Multi-user-efficient architecture maximizes sharing of hardware, memory and other resources amongst multiple users by still being able to be secure and exclusive. .

A Multi-User Architecture also provides Improved Scalability,Operational Efficiency, Software Development Life Cycle Management, etc.

III. QUEUEING THEORY

Queuing theory is a collection of mathematical models of various queuing systems. Queues or waiting lines arise when demand for a service facility exceeds the capacity of that facility i.e. the customers do not get service immediately upon request but must wait or the service facilities stand idle and waiting for customers.

The basic queuing process consists of customers arriving at a queuing system to receive some service. If the servers are busy, they join the queue in a waiting room (i.e., wait in line). They are then served according to a prescribed

queuing discipline, after which they leave the system.

Queueing model is characterized by[1]:

The arrival process of customers

In many practical situations customers arrive according to poisson stream so here we considered the arrival rate as Poisson distribution.

The behaviour of customers

The customers may be patient be patient or impatient to wait

The service discipline

The customers can be served one by one or in batches

The service capacity

There may be a single server or group of servers helping the customer

In brief a queueing system is a place where customers arrive according to an 'arrival process to obtain service from a service facility. The service facility may contain more than one server (or more generally resources) and it is assumed that a server can serve one customer at a time.

Basically, a queueing system consists of three major components:

- The input process
- The system structure
- The output process.

A. Characteristics of the Input Process

1) The size of arriving population

The size of the arriving customer population may be infinite in the sense that the number of potential customers from external sources is very large compared to those in the system, so that the arrival rate is not affected by the size.

2) Arriving patterns

Customers may arrive at a queueing system either in some regular pattern or in a totally random fashion. When customers arrive regularly at a fixed interval, the arriving pattern can be easily described by a single number – the rate of arrival.

3) Behaviour of arriving customers

Customers arriving at a queuing system may behave differently when the system is full due to a finite waiting queue or when all servers are busy.

B. Characteristics of the System Structure

1) Physical number and layout of servers

The service facility may comprise of one or more servers. A customer at the head of the waiting queue can go to any server that is free, and leave the system after receiving his/her service from that server,

2) The system capacity

The system capacity refers to the maximum number of customers that a queueing system can accommodate, inclusive of those customers at the service facility.

C. Characteristics of the Output Process

1) Queueing discipline or serving discipline

Queueing discipline, sometimes known as serving discipline, refers to the way in which customers in the waiting queue are selected for service. In general, we have:

- First-come-first-served (FCFS)
- Last-come-first-served (LCFS)
- Priority
- Processor sharing
- Random

2) Service-time distribution

Similar to arrival patterns, if all customers require the same amount of service time then the service pattern can be easily described by a single number. But generally, different customers require different amounts of service times; hence we again use a probability distribution to describe the length of service times the server renders to those customers.

IV.KENDALL NOTATION

There are many stochastic processes and a multiplicity of parameters (random variables) involved in a queueing system, so given such a complex situation how do we categorize them and describe them succinctly in a mathematical short form.

A queue is described in shorthand notation by ***A/B/C/K/N/D*** or the more concise *A/B/C*. In this concise version, it is assumed $K = \infty$, $N = \infty$ and D = FIFO.

Terms used in Kendall notation

A: The arrival processes

B: The service time distribution

C: The number of servers

K: The number of places in the system

N: The calling population

D: The queue's discipline

A. M/M/S/∞ Model for Single server.

A single server processes the customers one at a time. An arriving customer that finds the idle server and enters the service immediately. Otherwise, it enters a buffer and joins the end of queue of customers waiting for service

Arrival process server

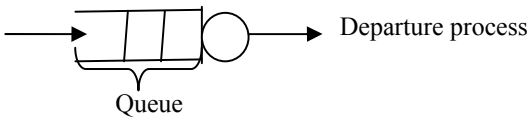

Figure1 *Arrival process into single server through queue*

In M/M/S/∞ S represents number of servers and ∞ indicates CCU

Since we are using single server here S=1

Traffic intensity (ρ) can be calculated by

$$\rho = \lambda/\mu \quad (1)$$

The probability of n customers in the queue is

$$P_n = (1-\rho)\rho^n \quad (2)$$

Mean number/expectation of participants in queue system is

$$L = \rho/(1-\rho) = \lambda/(\mu-\lambda) \quad (3)$$

Mean queue length is

$$L_q = \lambda^2/\mu(\mu-\lambda) \quad (4)$$

Mean staying time is

$$W = 1/(\mu-\lambda) \quad (5)$$

Mean waiting time is

$$W_q = \lambda/\mu(\mu-\lambda) \quad (6)$$

B.M/M/S/∞ Model for Multiple Servers

In a single server process, if CCU finds the server busy the customer needs to enter the buffer and waits for the service which leads to more waiting time and if the user who entered the server takes more service time the queue length also increases.

So in this model we proposed M/M/S/∞ model for multiple servers

The traffic intensity for n servers is

$$\rho_s = \rho/S \quad (7)$$

Mean staying time is:

$$W = W_q + 1/\mu \quad (8)$$

Mean waiting time is

$$W_q = L_q / \lambda \quad (9)$$

Mean queue length is

$$L_q = P_0 \rho^s \rho_s / s!(1-\rho_s)^2 \quad (10)$$

$$P_n = \begin{cases} (\rho^n/n!)\, P_0 & n=1,2,..s-1 \\ (\rho^n/s!s^{n-s})\, P_{0\ n=s,s+1,\ldots} & \end{cases} \quad (11)$$

The probability that no customers in the queue is given by:

$$P_0 = \left[\sum_{n=0}^{s-1} (\rho^n/n!) + \rho^s/s!(1-\rho_s) \right]^{-1} \quad (12)$$

The total number of customers in the queue (excluding number of customers being served)

$$N_q = \rho P_0/1-\rho \quad (13)$$

The average number of busy servers for a system in steady state

$$N_S = 2^* \rho \quad (14)$$

The total number of customers

$$N = N_q + N_S \quad (15)$$

Mean number of participants in the queue/mean queue length

$$L = L_q + \rho \quad (16)$$

V.NUMERICAL EXAMPLE RESULTS AND ANALYSIS

We validate our models by using different stream of arrival rates, λ and service rates, μ.Here we validated by assuming n=5 (with 5 customers)

A.INITIAL PARAMETERS

λ	μ
20	40
60	70
120	122

Table 1: Initial parameters (arrival rate and service rate[3])

B. M/M/1 Model

λ	μ	L	Lq	W	Wq
20	40	1	0.5	0.05	0.025
60	70	5.6	5.14	0.1	0.08
120	122	49	5.9	0.5	0.49

Table 2: Results of queue length and waiting time for single server

C. M/M/2 model

λ	μ	L	Lq	W	Wq
20	40	0.506	0.0006	0.02	0.0003
60	70	0.9	0.1	0.01	0.001
120	122	13.46	12.5	0.108	0.1

Table 3: Results of queue length and waiting time for two servers

D.*GRAPHS*

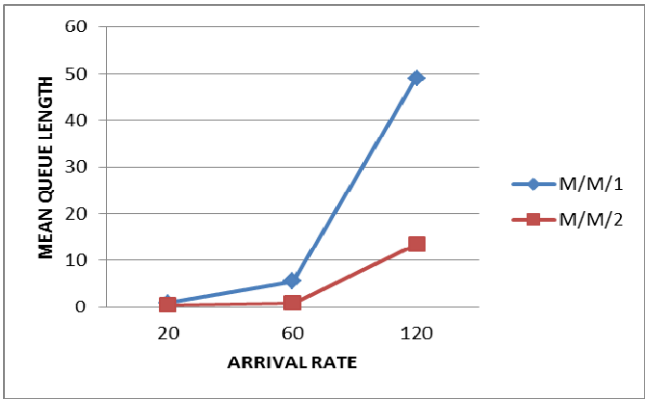

Fig:2 A graph analysing the mean queue length for single server and two servers.

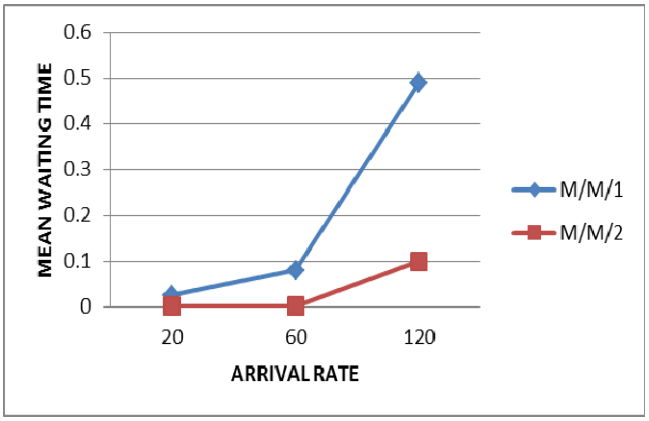

Fig:3 A graph analyzing the mean waiting time for single server and two servers.

VI CONCLUSION

In this paper, we have shown M/M/S model for two servers which increases the performance over using one server by reducing the queue length and waiting time. Analysis and numerical results clearly shows that our approach for M/M/2 reduces queue length and waiting time when compared to M/M/1 and also guarantee the QoS requirements of the CCU's jobs, and also can make the maximum profits for the CCSP.

REFERENCES

1. Lotfi Tadj "Waiting in line", IEEE POTENTIALS 1996
2.Huimin Xiao, Guozheng Zhang ,"The Queuing Theory Application in Bank Service Optimization", IEEE 2010.
3.Luqun Li, "An Optimistic Differentiated Service Job Scheduling System for Cloud Computing Service Users and Providers", Third International Conference on Multimedia and Ubiquitous Engineering,2009
4.Hock, N.C., "*Queueing Modelling Fundamentals".* JOHN WILEY&SONS, 1997
5. T.K.Y.Iyengar, S.Ranganatham,"Probability and statistics ", S.Chand,2008
6.https://www.xing.com/img/forums/2/3/f/cc02b3b5c.32464_1_2.png?priv=pri35d3edx%2f
7 .http://www.articlesfactory.com/articles/internet/experience-limited-downtime-with-cloud-server-hosting.html