Unit-3: Queuing System

Introduction of Queuing System:

Queuing system are the waiting lines in which the system attribute are waiting for a service. The queue may be of the customer waiting for the server or server waiting for customer. The waiting line situation arises either there is too much demand on the service facility so that customer have to wait for getting service or there is too less demand in which service facility have to wait for the customer.

- The line where the entities or customers wait is generally known as queue.
- The combination of all entities in system being served and being waiting for services will be called a queuing system.
- The general diagram of queuing system can be shown as a queuing system involves customers arriving at a constant or variable time rate for service at a service station.
- Customers can be students waiting for registration in college, airplane queuing for landing at airfield, or jobs waiting in machines shop.
- If the customer after arriving can enter the service center, it is good, otherwise they have to wait for the service and form a queue i.e. waiting line. They remain in queue till they are provided the service.
- Sometimes queue being too long, they will leave the queue and go, it results a loss of customer.
- Customers are to be serviced at a constant or variable rate before they leave the service station.

The basic concept of queuing theory is the optimization of **wait time**, **queue length**, and the **service available** to those standing in a queue.

Cost is one of the important factors in the queuing problem.

- Waiting in queues incur cost, whether human are waiting for services or machines waiting in a machine shop. On the other hand if service counter is waiting for customers that also involves cost.
- In order to reduce queue length, extra service centers are to be provided but for extra service centers, cost of service becomes higher.

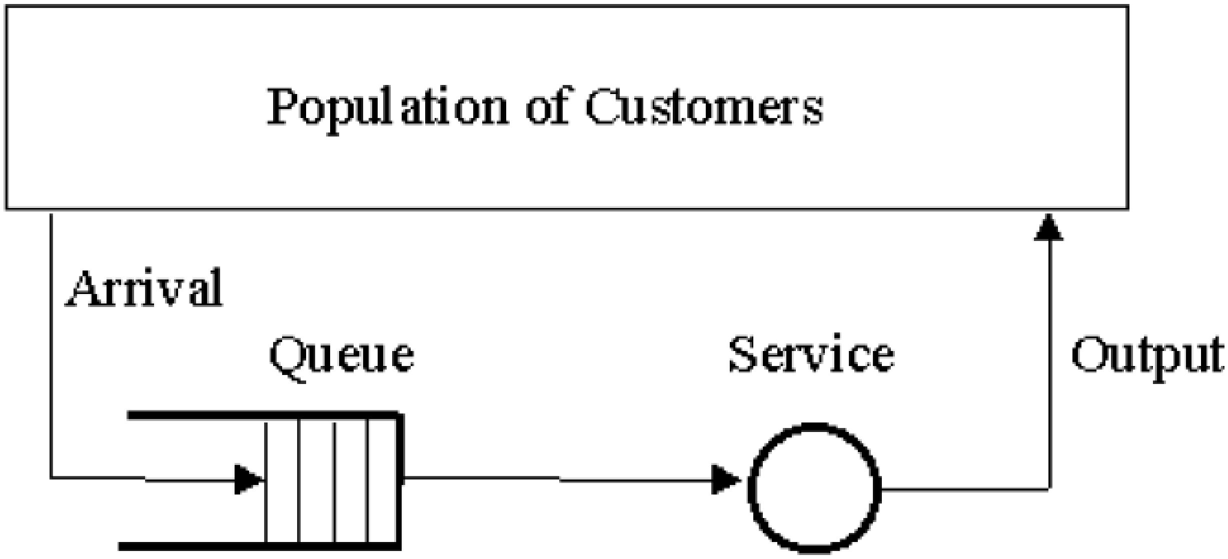


Fig: Queuing system in service center

Fig: Queuing system in the traffic

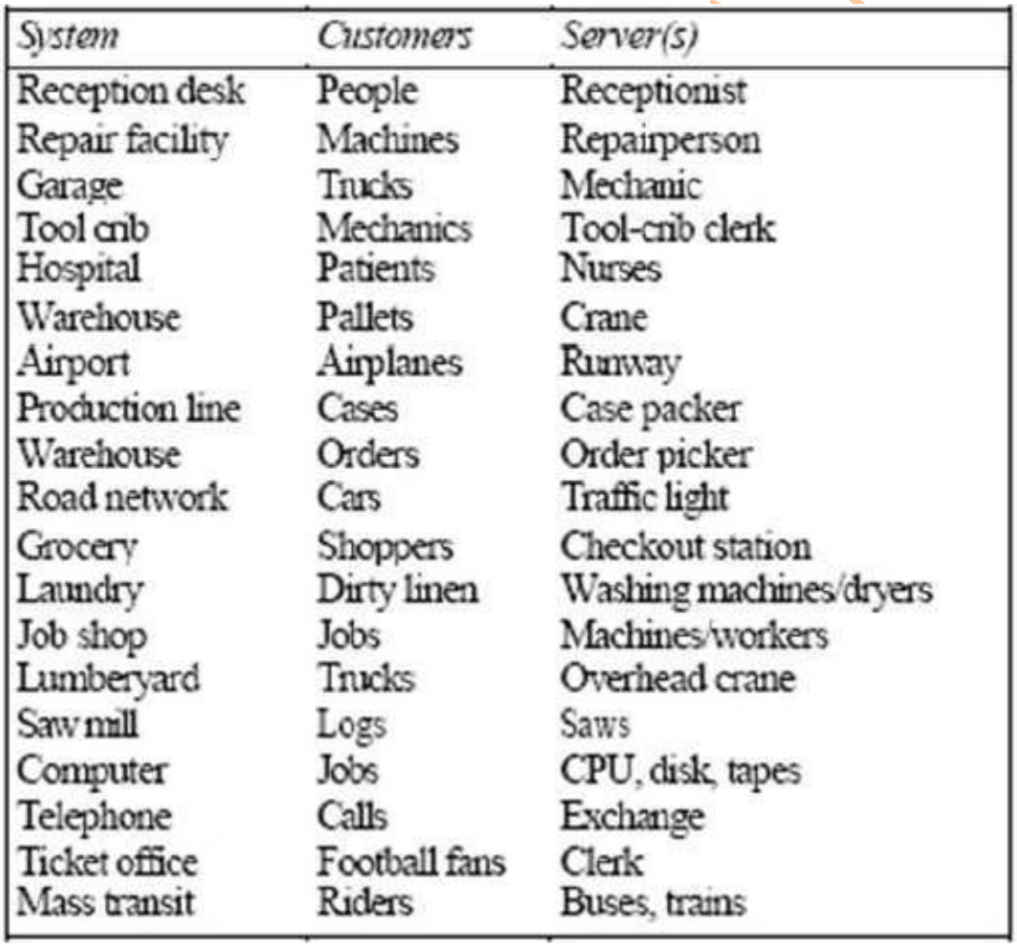
Characteristic or Elements of Queuing System:

The key elements, of a queuing system are the ***customers*** and ***servers***.

The term **"customer"** can refer to people, machines, trucks, mechanics, patients etc. anything that arrives at a facility and requires service.

The term **"server"** might refer to receptionists, repairpersons, CPUs in a computer, or washing machines etc. any resource (person, machine, etc.) which provides the requested service.

Following table shows the Customers and Servers of different Queuing System



System	Customers	Server(s)
Reception desk	People	Receptionist
Repair facility	Machines	Repairperson
Garage	Trucks	Mechanic
Tool crib	Mechanics	Tool-crib clerk
Hospital	Patients	Nurses
Warehouse	Pallets	Crane
Airport	Airplanes	Runway
Production line	Cases	Case packer
Warehouse	Orders	Order picker
Road network	Cars	Traffic light
Grocery	Shoppers	Checkout station
Laundry	Dirty linen	Washing machines/dryers
Job shop	Jobs	Machines/workers
Lumberyard	Trucks	Overhead crane
Saw mill	Logs	Saws
Computer	Jobs	CPU, disk, tapes
Telephone	Calls	Exchange
Ticket office	Football fans	Clerk
Mass transit	Riders	Buses, trains

In order to model queuing systems, we first need to be a bit more precise about what constitutes a queuing system.

The some basic elements common to all queuing systems are:

1. The calling population
2. System capacity
3. Arrival Process or patterns
4. Service process or patterns
5. Service time
6. Queuing discipline

The Calling Population:

The population of potential customers, referred to as the calling population, which may be finite or infinite.

Examples of infinite populations include the potential customers of a restaurant, bank, etc.

The main difference between finite and infinite population models is how the arrival rate is defined. In an infinite-population model, the arrival rate is not affected by the number of customers who have left the calling population and joined the queuing system. On the other hand, for finite calling population models, the arrival rate to the queuing system does depend on the number of customers being served and waiting.

System Capacity:

In many queuing systems there is a limit to the number of customers that may be in the waiting line or system.

For example, an automatic car wash may have room for only 10 cars to wait in line to enter the mechanism.

An arriving customer who finds the system full does not enter but returns immediately to the calling population. Some systems, such as concert ticket sales for students, may be considered as having unlimited capacity. There are no limits on the number of students allowed to wait to purchase tickets.

When a system has limited capacity, a distinction is made between the arrival rate (i.e., the number of arrivals per time unit) and the effective arrival rate (i.e., the number who arrive and enter the system per time unit).

Arrival Process or patterns:

Before entities can be processed or subjected to waiting, they must first enter the system. Depending on the environment, entities can arrive smoothly or in an unpredictable fashion. They can arrive one at a time or in clumps (e.g., bus loads or batches). They can arrive independently or according to some kind of correlation.

Typically the arrival is described by random distribution of intervals also called arrival pattern. Arrival process for infinite-population models is usually characterized in terms of inter-arrival times of successive customers. Arrivals may occur at scheduled times or at random times. When at random times, the inter arrival times are usually characterized by a probability distribution.

The most important model for random arrivals is the ***Poisson arrival process*** where the arrival pattern is random that means arrival time of next event does not depend on the previous event, these phenomena is called memory less.

Example: Phone calls arriving at an exchange, customers arriving at a fast food restaurant, hits on a web site, and many others.

Service process or pattern:

Once entities have entered the system they must be served. The physical meaning of “service” depends on the system. Customers may go through the checkout process. Parts may go through machining. Patients may go through medical treatment. Orders may be filled. And so on. From a modeling standpoint, the operational characteristics of service matter more than the physical characteristics.

- Specifically, we care about whether service times are long or short, and whether they are regular or highly variable.
- We care about whether entities are processed in first-come-first-serve (FCFS) order or according to some kind of priority rule.
- We care about whether entities are serviced by a single server or by multiple servers working in parallel etc.

Markov Service Process:

A special service process is the Markov service process, in which entities are processed one at a time in FCFS order and service times are independent and exponential.

For example, in a Markov service process would imply that the additional time required resolving a caller's problem is 15 minutes, no matter how long the technician has already spent talking to the customer. It is explained in a case where the average service time is 15 minutes, but many customers require calls much shorter than 15 minutes (e.g., to be reminded of a password or basic procedures) while a few customers require significantly more than 15 minutes (e.g., to perform complex diagnostics or problem resolution).

Simply knowing how long a customer has been in service doesn't tell us enough about what kind of problem the customer has, to predict how much more time will be required.

Service time:

The service times of successive arrivals are denoted by S1, S2, S3...They may be constant or of random duration. The exponential, Weibull, gamma, lognormal, and truncated normal distributions have all been used successfully as models of service times in different situations.

Sometimes services may be identically distributed for all customers of a given type or class or priority, while customers of different types may have completely different service-time distributions. In addition, in some systems, service times depend upon the time of day or the length of the waiting line.

For example, servers may work faster than usual when the waiting line is long, thus effectively reducing the service times.

A queuing system consists of a number of service centers and interconnecting queues. Each service center consists of some number of servers denoted as c, working in parallel. Parallel service mechanisms are either single server ($c = 1$), multiple server ($1 < c < \infty$), or unlimited servers ($c= \infty$). (A self-service facility is usually characterized as having an unlimited number of servers.)

For example: Consider a warehouse where customers may either serve themselves; or Wait of three clerks, and finally leave after paying a single cashier. The system is represented by the flow diagram in figure below:

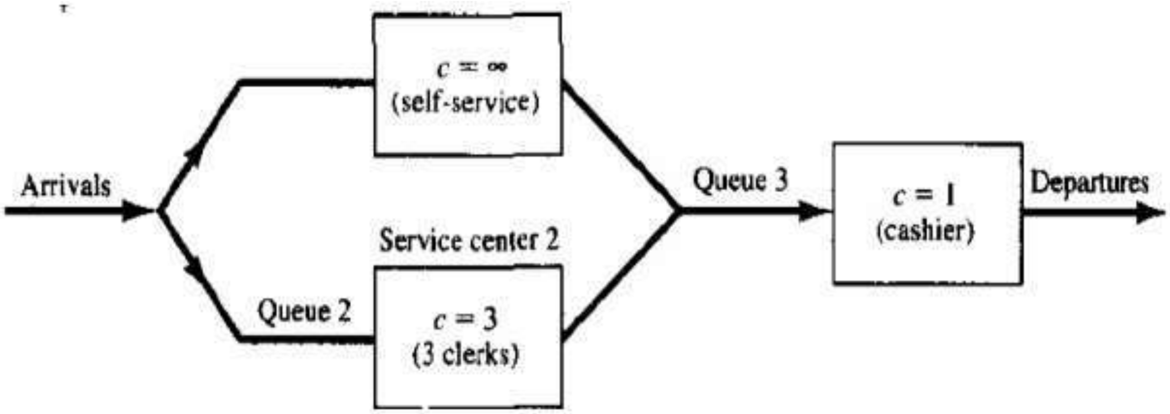


Fig: Warehouse with three service centers

Queuing Discipline:

The third required component of a queuing system is a ***queue***, in which entities wait for service.

- The number of customer can wait in a line is called system capacity.
- The simplest case is an unlimited queue which can accommodate any number of customers. It is called system with unlimited capacity.
- But many systems (e.g., phone exchanges, web servers, call centers), have limits on the number of entities that can be in queue at any given time.

Arrivals that come when the queue is full are rejected (e.g., customers get a busy signal when trying to dial into a call center). Even if the system doesn't have a strict limit on the queue size, the logical ordering of customer in a waiting line is called Queuing discipline and it determines which customer will be chosen for service.

We may say that queuing discipline is a rule to choose the customer for service from the waiting line.

The queuing discipline includes:

a) **FIFO (First in First out):** According to this rule, Service is offered on the basis of arrival time of customer. The customer who comes first will get the service first. So in other word the customer who get the service next will be determine on the basis of longest waiting time.

b) **Last in First out (LIFO):** It is usually abbreviated as LIFO, occurs when service is next offered to the customer that arrived recently or which have waiting time least. In the crowded train the passenger getting in or out from the train is an example of LIFO.

c) **Service in Random order (SIRO):** it means that a random choice is made between all waiting Customers at the time service is offered i.e. a customer is picked up randomly forms the waiting queue for the service.

d) **Shortest processing time First (SPT):** it means that the customer with shortest service time will be chosen first for the service i.e. the shortest service time customer will get the priority in the selection process.

e) **Priority:** a special number is assigned to each customer in the waiting line and it is called priority. Then according to this number, the customer is chosen for service.

<u>Queuing Behavior:</u>

There are some behavior that customer can show during the queuing system to get faster service or to give-up from waiting in queue.

- Customers may balk at joining the queue when it is too long (e.g., cars pass up a drive through restaurant if there are too many cars already waiting). ***It is called balking.***
- Customer may also exit the system due to impatience (e.g., customers kept waiting too long at a bank decide to leave without service) or perishable (e.g., samples waiting for testing at a lab spoil after some time period). ***It is called reneging.***
- When there is more than one line forming for the same service or server, the action of moving customer from one line to another line because they think that they have chosen slow line. ***It is called Jockeying.***

Queuing Notation (Kendall's Notation)

Recognizing the diversity of queuing systems, Kendall [1953] proposed a notational system for parallel server systems which has been widely adopted.

An abridged version of this convention is based on the format **A / B / c / N / K.** These letters represent the following system characteristics:

- **A** represents the inter-arrival time distribution.
- **B** represents the service-time distribution.
- Common symbols for A and B include
  - **M** (exponential or Markov)
  - **D** (constant or deterministic)
  - **Ek** (Erlang of order k)
  - **PH** (phase-type)
  - **H** (hyperexponential)
  - **G** (arbitrary or general)
  - **GI** (General independent)
- **c** represents the number of parallel servers.
- **N** represents the system capacity.
- **K** represents the size of the calling population

If the capacity is not specified, it is taken as infinity, and if calling population is not specified, it is assumed unlimited or infinite

Example 1: M / M / 1 / ∞ / ∞ indicates a single-server system that has unlimited queue capacity and an infinite population of potential arrivals. The inter-arrival times and service times are exponentially distributed.

When N and K are infinite, they may be dropped from the notation.

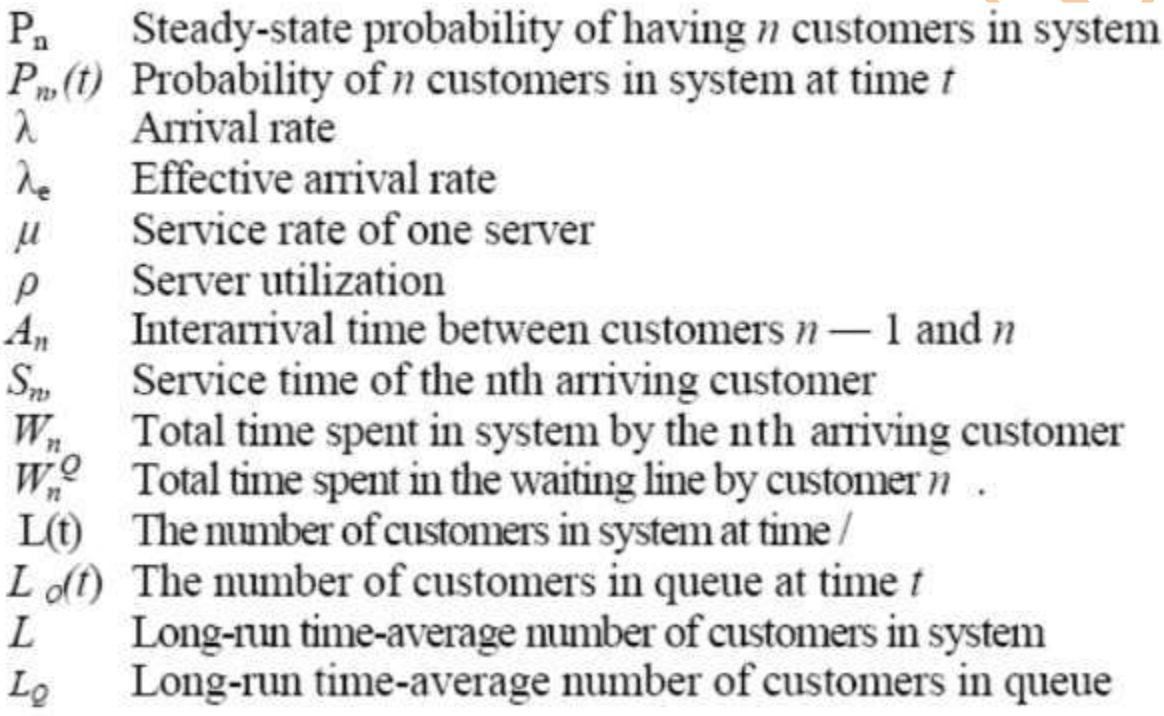
For example, M / M / 1 / ∞ / ∞ is often shortened to M/M/l.

Example 2: If notation is given as M / D / 2 means exponential arrival time, deterministic service time, 2 servers, infinite service capacity, and infinite population.

Example 3: M / D / 2 / 5 / ∞ stands for a queuing system having exponential arrival times, deterministic service time, 2 servers, capacity of 5 customers, and infinite population.

Additional Notations:

Additional notation used for parallel server systems is listed in Table 1 given below. The meanings may vary slightly from system to system. All systems will be assumed to have a FIFO queue discipline.



P_n	Steady-state probability of having *n* customers in system
$P_n(t)$	Probability of *n* customers in system at time *t*
λ	Arrival rate
λ_e	Effective arrival rate
μ	Service rate of one server
ρ	Server utilization
A_n	Interarrival time between customers *n* — 1 and *n*
S_n	Service time of the nth arriving customer
W_n	Total time spent in system by the nth arriving customer
W_n^Q	Total time spent in the waiting line by customer *n* .
L(t)	The number of customers in system at time /
$L_Q(t)$	The number of customers in queue at time *t*
L	Long-run time-average number of customers in system
L_Q	Long-run time-average number of customers in queue

ώ Long-run average time spent in system per customer

$ώ_Q$ Long-run average time spent in queue per customer

Single Server Queuing System:

For the case of simplicity, we will assume for the time being, that there is single queue and only one server serving the customers. We make the following assumptions.

- **First-in First-out (FIFO):** Service is provided on the first come, first served basis.
- **Random**: Arrivals of customers is completely random but at a certain arrival rate.
- **Steady state**: The queuing system is at a steady state (does not change with time) condition.

The above conditions are very ideal conditions for any queuing system and assumptions are made to model the situation mathematically. First condition only means irrespective of customer, one who comes first is attended first and no priority is given to anyone.

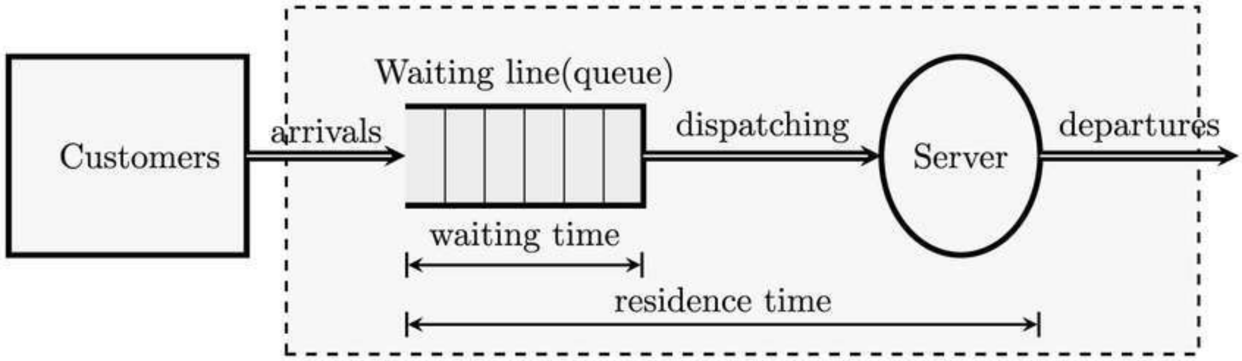


Fig: Single Server Queuing System

Poison arrival Patterns:

Second condition says that arrival of a customer is completely random. This means that an arrival can occur at any time and the time of next arrival is independent of the previous arrival. With this assumption it is possible to show that the distribution of the inter-arrival time is exponential. This is equivalent to saying that the number of arrivals per unit time is a random variable with a Poisson's distribution.

If X = number of arrivals per unit time, then, probability distribution function of arrival is given as:

$$f(x) = \Pr(X = x) = \frac{e^{-\lambda}\lambda^{x}}{x!}, \begin{cases} x = 0, 1, 2, \ldots \\ \lambda > 0 \end{cases}$$

$$E(X) = \lambda$$

Where, **λ** is the average number of arrivals per unit time, and **X** is the number of customers per unit time. This pattern of arrival is called Poisson’s arrival pattern.

Example:

In a single pump service station, vehicles arrive for fueling with an average of 5 minutes between arrivals. If an hour is taken as unit of time, cars arrive according to Poison’s process with an average of $\lambda = 12$ cars/hr.

The distribution of the number of arrivals per hour is,

$$f(x) = \Pr(X = x) = \frac{e^{-\lambda}\lambda^{x}}{x!} = \frac{e^{-12}12^{x}}{x!}, \begin{cases} x = 0, 1, 2, \ldots \\ \lambda > 0 \end{cases}$$

$$E(X) = 12 \text{ cars/hr}$$

Here x is the numbers of vehicles arrive, place the value of x to calculate probability of x numbers of vehicles arrivals.

<u>Multi-Server Queuing System:</u>

In the multi-server queuing system, more than one server is serves to all arrival entities which make faster service.

Figure bellow shows a generalization of the simple model for multiple servers, all sharing a common queue. If an entity arrives and at least one server is available, then the entity is immediately dispatched to that server. It is assumed that all servers are identical; thus, if more than one server is available, it makes no difference which server is chosen for the entity.

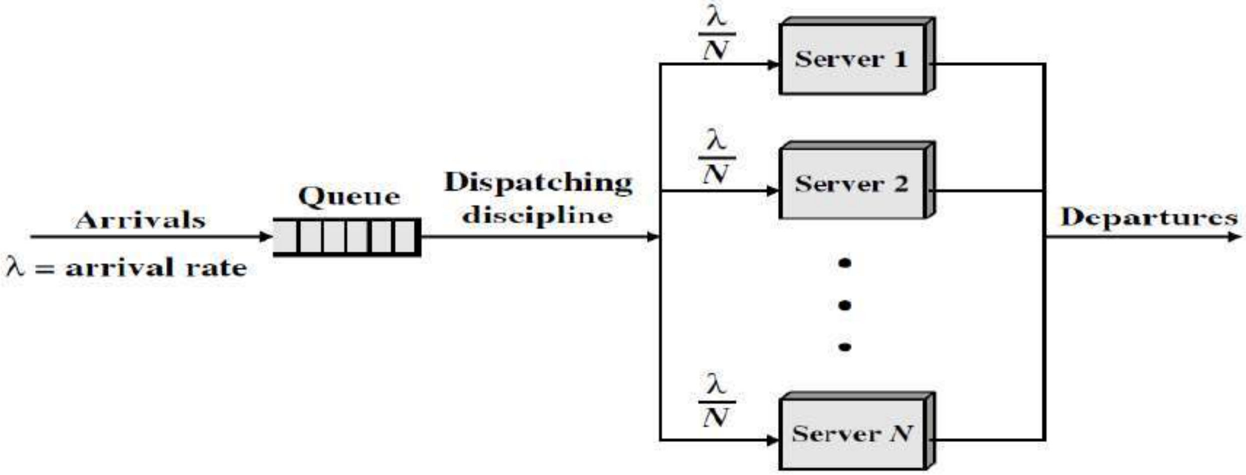


Fig: Multi-server queuing system

If all servers are busy, a queue begins to form. As soon as one server becomes free, an item is dispatched from the queue using the dispatching discipline in force. The key characteristics typically chosen for the multi-server queue correspond to those for the single-server queue. That is, we assume an infinite population and an infinite queue size, with a single infinite queue shared among all servers. Unless otherwise stated, the dispatching discipline is **FIFO**. For the multi-server case, if all servers are assumed identical, the selection of a particular server for a waiting entity has no effect on service time.

The total server utilization in case of Multi-server queue for N server system is:

$$\rho = \lambda / c\mu$$

Where **μ** is the service rate and **λ** is the arrival rate and **c** is the constant which denotes the number of servers used.

There is another concept which is called multiple single server queue system as shown below:

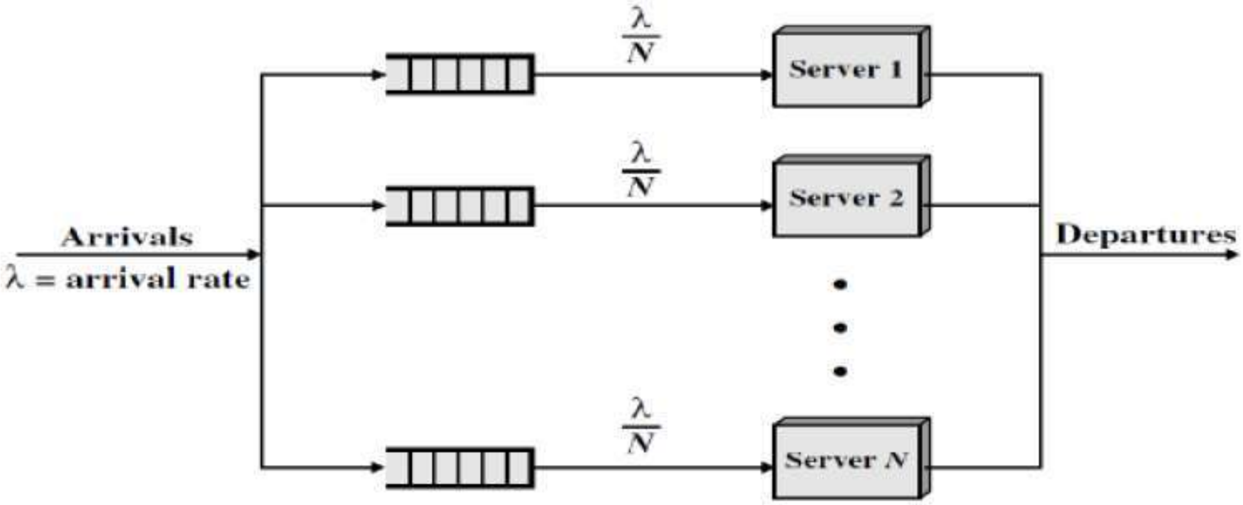


Fig: Multiple single server queuing system

Measurements of Queuing System Performance:

The performance of a queuing system can be evaluated in terms of a number of response parameters, however the following four are generally employed.

- Average number of customer in the queue or in the system
- Average waiting time of the customer in the queue or in the system
- System utilization (Server utilization)
- The cost of waiting time and idle time

Each of these measures has its own importance. The knowledge of average number of customers in the queue or in the system helps to determine the space requirements of the waiting entities. Also too long a waiting line may discourage the prospectus customers, while no queue may suggest that service offered is not good quality to attract customers.

The following measures are used in the analysis of queue system:

Traffic Intensity:

As we know, Ta be mean arrival time, Ts be service time with corresponding rates:
Average arrival rate (λ) = 1/Ta (Ta is denoted by tou (τ))

Average service rate (μ) = 1/Ts

Then, the ratio of mean service rate and mean inter-arrival rate is called the **traffic intensity** (***u***).

u = λ" Ts

λ = 1/Ta

So, Traffic Intensity (u) can also represented by equations:

u =Ts / Ta

If there is any balking or reneging, not all arriving entities get served. It is necessary therefore to distinguish between actual arrival rate and the arrival rate of entities that get served.

Here λ" denoted the all arrivals including balking or reneging.

- The probability that an entity have to wait more than a given time is known as **delay distribution**.
- The knowledge of average waiting time in the queue is necessary for determining the cost of waiting in the queue.

Server or System Utilization:

System utilization, that is, the percentage capacity utilized reflects the extent to which the facility is busy rather than idle. It consists of only the arrival entity that gets served. It is denoted by ρ = λTs.

System utilization factor (ρ) is the ratio of average arrival rate (λ) to the average service rate (μ).

ρ = $\lambda/$μ , in case of single server model

ρ = $\lambda/$nμ , in case of 'n' server model

Thus probability of finding service counter (server) free is (1 – ρ)

Where, ρ is the service counter utilization.

If the value of ρ is zero, then there is 100% service counter is free.

The system utilization can be increased by increasing the arrival rate which amounts to increasing the average queue length as well as the average waiting time, as shown is figure bellow. Under normal circumstances 100% system utilization is not a realistic goal.

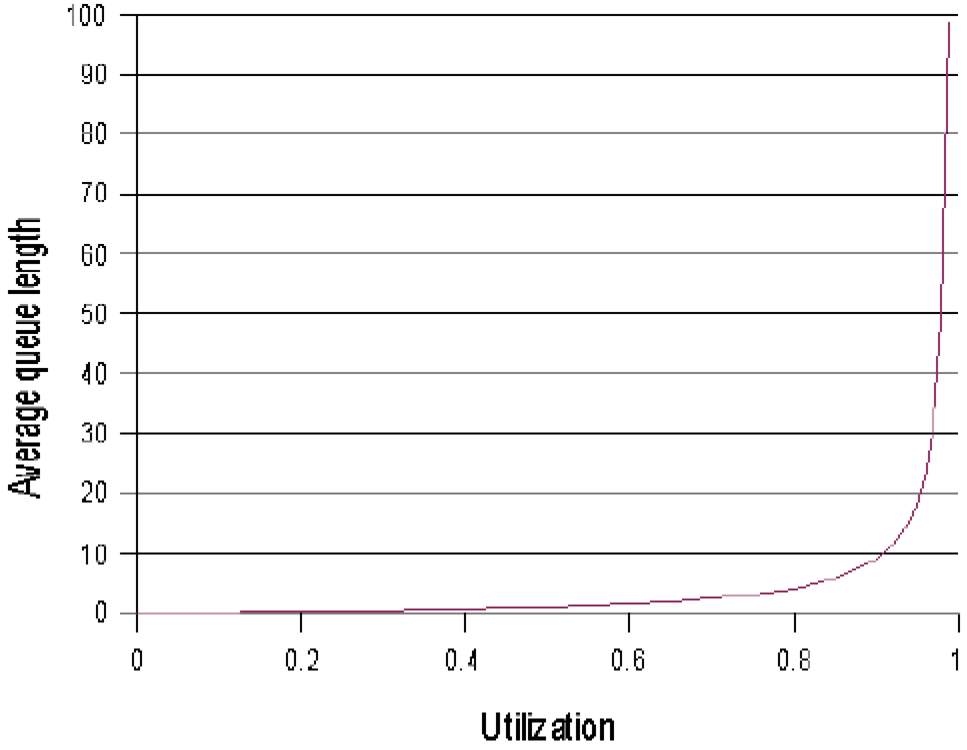


Fig: Average queue length as a function of utilization

<u>Some notation or Formula used to Measure the different parameter of queue:</u>

Two principal measures of queuing system are;

1. The mean number of customers waiting
2. The mean time the customer spend waiting

Bothe these quantities may refer to the total number of entities in the system, those waiting and those being served or they may refer only to customer in the waiting line.

Average number of customers in the System $\overline{L}_S = \frac{\rho}{1-\rho} = \frac{\frac{\lambda}{\mu}}{1-\frac{\lambda}{\mu}} = \frac{\lambda}{\mu-\lambda}$

Average number of customers in the Queue $\overline{L}_Q$

= Average number of customers in the System − Server Utilization

$$= \overline{L}_S - \frac{\lambda}{\mu} = \frac{\lambda}{\mu-\lambda} - \frac{\lambda}{\mu} = \frac{\lambda^2}{\mu(\mu-\lambda)}$$

Average waiting time in the System $\overline{W}_S = \frac{\textit{Average number of customer in the system}}{\textit{Mean arrival rate}}$

$$= \frac{\overline{L}_S}{\lambda} = \frac{\frac{\lambda}{\mu-\lambda}}{\lambda} = \frac{1}{\mu-\lambda}$$

Average waiting time in the Queue $\overline{W}_Q = \frac{\textit{Average number of customer in the Queue}}{\textit{Mean arrival rate}}$

$$= \frac{\overline{L}_Q}{\lambda} = \frac{\frac{\lambda^2}{\mu(\mu-\lambda)}}{\lambda} = \frac{\lambda}{\mu(\mu-\lambda)}$$

Where, λ is average arrival rate and μ is average service rate.

Example 1:

At the ticket counter of football stadium, people come in queue and purchase tickets. Arrival rate of customers is 1/min. It takes at the average 20 seconds to purchase the ticket.

If a sport fan arrives 2 minutes before the game starts and if he takes exactly 1.5 minutes to reach the correct seat after he purchases a ticket, can the sport fan expects to be seated for the kick-off?

Solution:

Let minute is used as unit of time. Since ticket is disbursed in 20 seconds, this means, three customers enter the stadium per minute, that is service rate is 3 per minute.

Therefore,

Average arrival rate (λ) = 1 arrival/min

Average service rate (μ) = 3 arrivals/min

Time taken to reach the seat =1.5 minutes

Remaining time for kick-off = 2 minutes

Waiting time in the system ($\bar{\mathbf{W}}_S$) = 1/ (μ- λ) = 1/ (3-1) = 1/2 = 0.5 minutes

Total time required for fan to process and reach to seat =

Waiting time in system + Time taken to reach the seat

= 0.5 +1.5 = 2 minutes

The average time to get the ticket plus the time to reach the correct seat is 2 minutes exactly, so the sports fan can expect to be seated for the kick-off.

Example2:

Customers arrive in a bank according to a Poisson's process with mean inter arrival time of 10 minutes. Customers spend an average of 5 minutes on the single available counter, and leave.

1. What is the probability that a customer will not have to wait at the counter?
2. What is the expected number of customers in the bank?
3. How much time can a customer expect to spend in the bank?

Solution:

We will take an hour as the unit of time. Inter-arrival time is 10 minutes means 6 customers can arrive in 1 hour. Thus,

Average arrival rate (λ) = 6 customers/hour,

Single counter takes 5 minutes to process 1 customer that means 12 customers in 1 hour. Thus,

Average service rate (μ) = 12 customers/hour.

1. The customer will not have to wait if there are no customers in the bank. Thus,

   $\mathbf{P_0} = 1 - \rho = 1 - \lambda/\mu = 1 - 6/12 = 0.5$

2. Expected numbers of customers in the bank are given by

   $\bar{\mathbf{L}}_S = \lambda / (\mu - \lambda) = 6/(12-6) = 6/6 = 1$

3. Expected time to be spent in the bank is given by

   $\bar{\mathbf{W}}_S = 1/(\mu - \lambda) = 1/(12-6) = 1/6$ hour = 10 minutes.

Example 3:

At the college, student comes in queue for take admit card. Arrival rate of student is 1/min. It takes at the average 22 seconds to take admit card from counter.

If college will close after 3 minutes and if a student is coming to take admit card and he takes exactly 1.5 minutes to reach the counter, can the student take admit card and leave college in time?

Solution:

Let minute is used as unit of time. Given that,

Average arrival rate of student (λ) = 1 arrival/min

Average time taken by counter to process is 22 sec that means only 2 students can take service in 1 minute.

Average service rate (μ) = 2 arrivals/min

Waiting time in the system ($\bar{W}_S$) = $1/(\mu - \lambda) = 1/(2-1) = 1$ minutes

Now total time will be 1+1.5 = 2.5 minutes.

College will close after 3 minutes and student required 2.5 minutes to completer his service. Hence he can take the admit card and leave the college in time.

Introduction to Network of Queues:

Many communication systems must model as a set of interconnected queues which is called a queuing network.

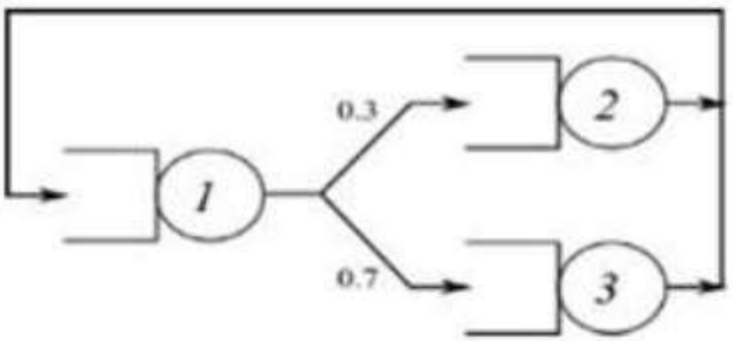


Fig: Network of queues

Systems modeled by queuing networks can roughly be grouped in to four categories:

- Closed Network
- Open Network
- Mixed Network
- Networks with population constraints (Loss Networks)

Closed Network:

Fixed number of entities (k) are trapped in the system and circulate among the queues is called closed network.

Example: CPU job scheduling problem

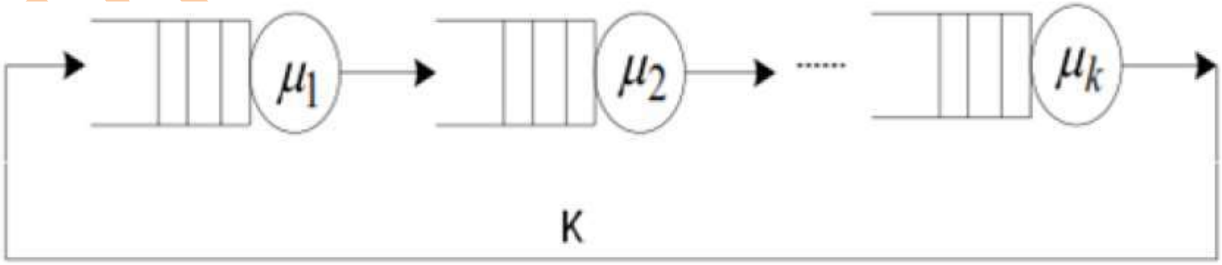


Fig: Close Network

Open Network:

Entities arrive from outside the system are served and then depart, this kind of network is called open network.

Example: Packet switching data network

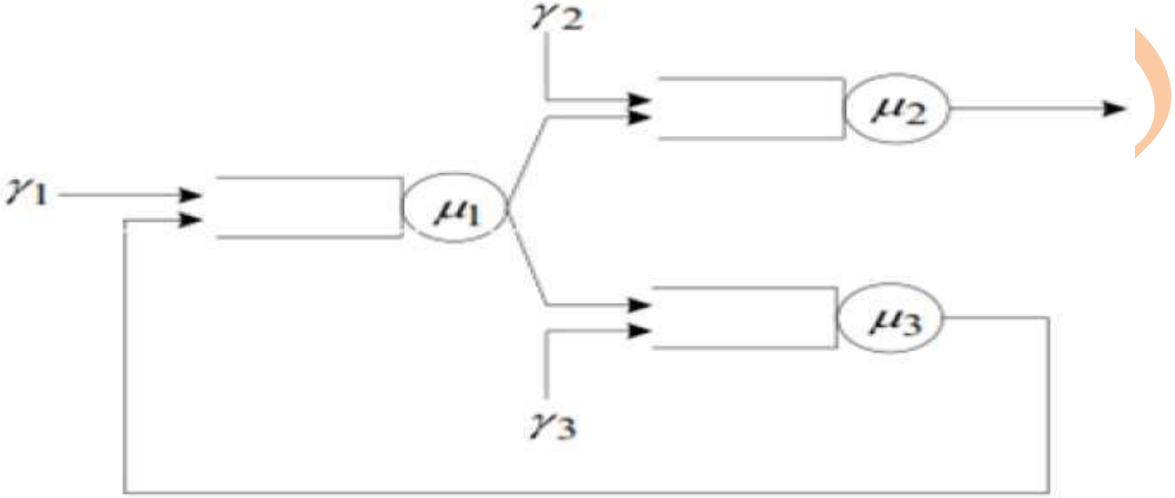


Fig: Open Network

Mixed Network:

Any combination of open and closed network is called mixed network.

Example: Simple model of virtual circuit that is window flow controlled

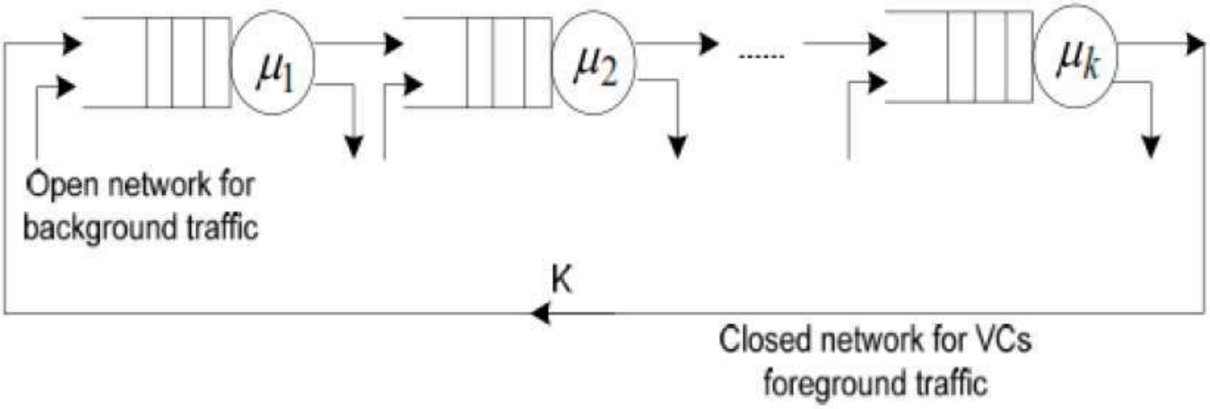

μ1
μ2
......
μk
Open network for background traffic
K
Closed network for VCs foreground traffic

Fig: Mixed network

Network with population constraints (Loss Network):

Entities arrive from outside the system if there is room in the system. They enter, served and then depart. In this network entities are lost when arriving to room of system is full.

Example: Queue sharing a common buffer pool

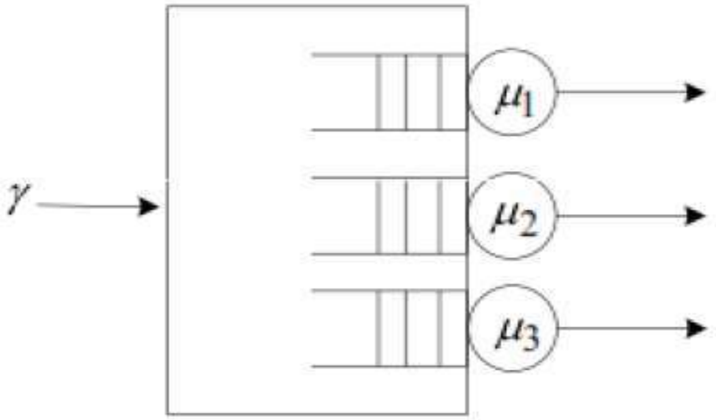


Fig: Loss Network

Application of Network Queuing:

Network queuing is a very important application of queuing theory. The term 'network of queues' describes a situation where the input from one queue is the output from one or more others. This is true in many situations from telecommunications to a PC.

Below is a description of some of the broad applications of network queuing.

Computer Networks:

A simple example of network queuing is the central server network. This consists of a CPU (Central Processing Unit), storage units it can access and input devices to access it. The tasks the CPU performs are placed on queues on different criteria. Also, the storage units could have their own individual queues.

Network communication:

There are several broad methods connected to network communication:

- **Circuit Switching:** When a call is made from a source to a destination it must traverse several nodes along the way. Which nodes it traverses is determined by the availability of free channels along the way. Each node has a queue for calls requesting a channel. Once a channel has been opened the call can progress to the next node and wait for a channel there. The channel remains open until the source or destination (once reached) closes the call.
- **Packet Switching:** Messages are transmitted through intermediate stages and the route a message takes depends entirely upon the current load on the system. The route allocation is dynamic. Each stage requires a random amount of time reflecting the length of the queue at that stage.

Broadcasting:

There are several broad methods connected to Broadcasting:

- **Radio Communication:** Considering the nodes as transmitters/receivers you can treat each as having a queue for their channels. Without going into great detail of the various systems used: it is always necessary to consider the fact that to open a channel you must check to see if the two adjacent channels are also free as interference blocks transmissions. When the channels are not free it may be necessary to re-allocate communications that already have channels to make room.
- **Digital Communication:** This is done on the basis of time slots. For a given communication link it could have several or all slots filled and no interference would take place making allocation far simpler. The aspect of nodes with queues still applies however.

Other application of queuing system:

- Telecommunications
- Transportation
- Logistics
- Finance
- Emergency services
- Computing
- Industrial engineering
- Project management
- Operation research

Little's Law

Little's Law connects the capacity of a queuing system, the average time spent in the system, and the average arrival rate into the system without knowing any other features of the queue.

The formula is quite simple and is written as follows:

$$L = \lambda W$$

Or,

$$\lambda = \frac{L}{W}$$

Or,

$$W = \frac{L}{\lambda}$$

Where:

- ***L*** is the average number of customers in the system
- ***λ*** (lambda) is the average arrival rate into the system
- ***W*** is the average amount of time spent in the system

Project management processes like Lean and Kanban wouldn't exist without the Little's Law queuing models. They're critical for business applications, in which Little's Law can be written in plain English as:

$$Work\ in\ Progress = Throughput * Lead\ Time$$

$$Throughput = \frac{Work\ in\ Progress}{Lead\ Time}$$

$$Lead\ Time = \frac{Work\ in\ Progress}{Throughput}$$

Example 1:

If you're waiting in line at a Coffee shop, there are 15 people in line, one server, and 2 people are served per minute. Find how much time you have to wait for coffee?

Solution:

Here given,

No of people in line (L) = 15

Rate of service (λ) = 2/min

According Little's Law:

$$\frac{L}{\lambda} = W$$

$$\frac{15\ people\ in\ line}{2\ people\ served\ per\ minute} = 7.5\ minutes\ of\ waiting$$

Hence, I have to wait 7.5 minutes for a coffee.

Assignment:

Long questions

1. What do you mean by Queuing system? Explain the characteristics of Queuing system with example.
2. Define the queuing system. Explain the elements of queuing system with example.
3. Explain about the Poison arrival process and Service process with example.

Short Questions:

1. Explain about the server utilization and Traffic intensity.
2. What do you mean by multi server queues?
3. What are the kendall notations of queuing system?
4. What do you mean by Queuing notation? Explain with example
5. Explain about the Queuing Discipline and behaviors.

End of Unit-3