Queuing system:

A queueing system in the context of stochastic processes is a mathematical model used to describe systems in which entities (customers, tasks, packets of data, etc.) arrive randomly over time and wait in line (or queue) to be served by one or more servers.

It helps in understanding and predicting system characteristics like the **average waiting time in the queue**, **the average number of entities in the system**, **system capacity**, and **service efficiency**.

Examples of queuing systems:

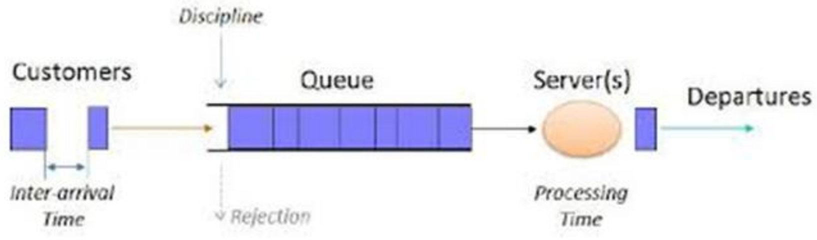
1. A personal or shared computer executing tasks sent by its users
2. An internet service provider whose customers connect to the internet, browse, and disconnect
3. A printer processing jobs sent to it from different computers
4. A customer service with one or several representatives on duty answering calls from their customers
5. A toll area on a highway, a fast-food drive-through lane, or an automated teller machine (ATM) in a bank, where cars arrive, get the required service and depart
6. Public transport- waiting for a train or a bus
7. Supermarkets - waiting for service

Main component of queuing system:

The main components of queuing system are:

1. Arrival process
2. Queuing and routing to servers
3. Server (Service mechanism)
4. Queue discipline
5. Departure

In designing queueing systems, we need to aim for a balance between service to customers (short queues implying many servers) and economic considerations (not too many servers).



Notation:

- A: Arrival process (Probability distribution for the arrival process)
- S: Service time distribution (Probability distribution for the service process)
- m: Number of servers (Number of channels)
- B: Number of buffers (system capacity)

- K: Population size (Maximum no. of customers in total)
- SD: Service discipline

The simplest queueing system has a Poisson arrival distribution, an exponential service time distribution and a single channel (one server)

Arrival Process:

It describes how entities or customers arrive at the queueing system. The times between arrivals are often assumed to be randomly distributed, with the Poisson arrival process being a common model where the time between arrivals follows an exponential distribution.

- **Population size**: The pool of potential customers or entities that may require service, can be finite or infinite.
- **Arrival pattern**: It describes how entities arrive at the system, randomly or deterministically, singly or in groups (batch or bulk arrivals). For random arrival, arrival distribution is often modelled by Poisson process.
- **Inter-arrival time distribution**: It describes how arrivals are distributed in time (e.g., what is the probability distribution of time between successive arrivals i.e. the ***interarrival time distribution***.
- **Customers Behaviour**
  - ✓ **Balking.** A customer may not like to join the queue due to long waiting line.
  - ✓ **Reneging.** A customer may leave the queue after waiting for some time due to impatience.
  - ✓ **Collusion.** Several customers may cooperate and only one of them may stand in the queue.
  - ✓ **Jockeying.** When there are a number of queues, a customer may move from one queue to another in hope of receiving the service quickly.

The arrival process can also be deterministic. A **deterministic arrival process** refers to a scenario where arrivals occur at **predetermined times** or with **fixed intervals**. Unlike stochastic processes, there's no randomness involved in when the next arrival will happen. For example, a bus arriving every 15 minutes or a factory machine producing a new item every second.

Typically, jobs arrive to a queuing system at **random times**. A counting process A(t) tells the number of arrivals that occurred by the time t. In stationary queuing systems (whose distribution characteristics do not change over time), arrivals occur at arrival rate

$$\lambda_A = \frac{EA(t)}{t}$$

for any t > 0, which is the expected number of arrivals per 1 unit of time. Then, the expected time between arrivals is

$$\mu_A = \frac{1}{\lambda_A}$$

Queuing and Routing to servers:

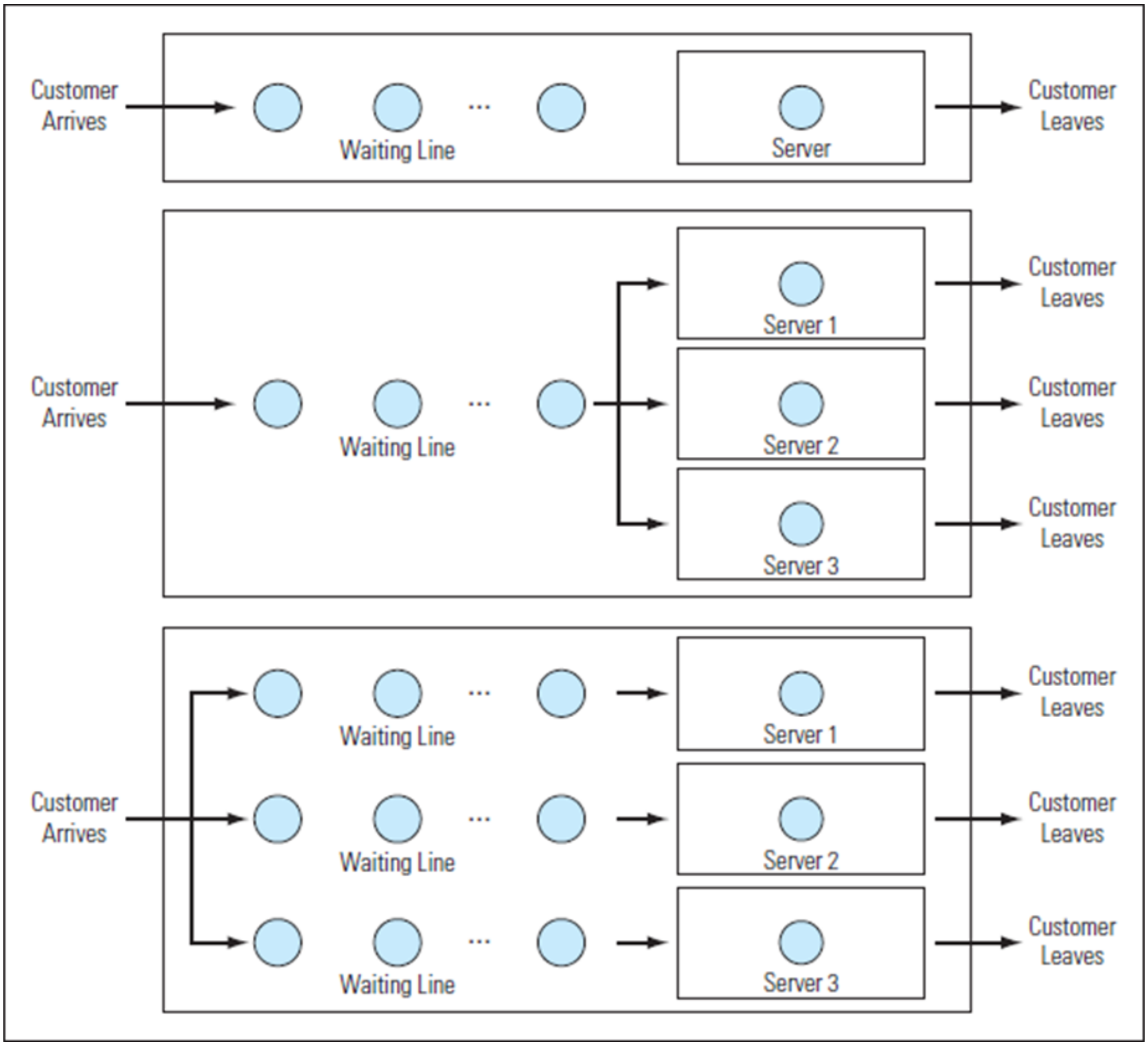
The **queue** is where customers wait patiently (or sometimes impatiently) for their turn at the service facility.

- ✓ **Queue discipline**: The rule by which entities in the queue are selected for service, for example, First-In-First-Out ((FIFO), Last-In-First-Out (LIFO) or priority. Arrived jobs are typically processed according to the order of their arrivals, on a “first come–first serve” basis.
- ✓ **Queue Capacity**: The maximum number of entities the queue can hold. It can be unlimited or have a finite limit.
- ✓ **Number of Queues**: Systems can have a single queue or multiple queues, affecting how entities are organized for service.

When a new job arrives, it may find the system in different states. If one server is available at that time, it will certainly take the new job. If several servers are available, the job may be randomized to one of them, or the server may be chosen according to some rules. For example, the fastest server or the least loaded server may be assigned to process the new job. Finally, if all servers are busy working on other jobs, the new job will join the queue, wait until all the previously arrived jobs are completed, and get routed to the next available server.

Various additional constraints may take place. For example, a queue may have a buffer that limits the number of waiting jobs. Such a queuing system will have limited capacity; the total number of jobs in it at any time is bounded by some constant C. If the capacity is full (for example, in a parking garage), a new job cannot enter the system until another job departs.

Also, jobs may leave the queue prematurely, say, after an excessively long waiting time. Servers may also open and close during the day as people need rest and servers need maintenance.



Service Mechanism:

The service mechanism in a queuing system refers to how customers are served after they join a queue. It's essentially the engine that keeps the queue moving and determines how long customers wait.

It refers to the configuration of servers (single server, multiple servers) and their service rates, often modeled by statistical distributions (e.g., exponential distribution for service times).

- **No. of servers available**: The number of servers available such as service counters, call centre agents, or processing machines. There might be single server or multiple servers for the service.
    - ✓ Single Server: This is the most basic setup, with one server handling customers one at a time. Think of a bank with a single teller window.
    - ✓ Multiple Servers: Here, you have multiple servers (like cashiers in a supermarket) working in parallel, reducing waiting times.
    - ✓ Unlimited Servers: This is a theoretical concept where an infinite number of servers are available, ensuring immediate service (rarely seen in practice).

- **Service time distribution**: Service time is the time it takes for a server to complete the service for a customer (how long the service will take). So, survive time distribution is the variability in how long it takes to serve each customer. It can be random, constant, or follow a specific pattern. Service time can be modelled by some probability distribution such as exponential distribution. The service rate determines how quickly customers are served
- **Arrangement of servers**: Servers are in series (each server has a separate queue) or in parallel (one queue for all servers)
- **Service Discipline**: Similar to queue discipline, it's the rule applied within the service facility for handling the entities, particularly in systems with multiple servers or phases of service.
- **Batching:** Sometimes, customers are grouped (batched) before service, which can improve efficiency but increase waiting time within batches.
- **Capacity**: The number of entities that can be served simultaneously, determined by the number of service channels or servers.
- **Server Behavior:** In rare cases, the service time might depend on factors like queue length, with servers speeding up or slowing down based on workload.

Once a server becomes available, it immediately starts processing the next assigned job. In practice, service times are random because they depend on the amount of work required by each task. The average service time is μ_S. The average service time may vary from one server to another as some computers or customer service representatives work faster than others. The service rate λ_S is defined as the average number of jobs processed by a continuously working server during one unit of time. It is given by,

$$\lambda_S = \frac{1}{\mu_S}$$

Departure

Once the service is complete, entities or customers exit the system. This component can also include post-service feedback or actions.

Main parameters of performance of queueing system

The following parameters and random variables describe the performance of a queuing system.

- ✓ Arrival Rate (λ_A)
- ✓ Service Rate (λ_S)
- ✓ Average (mean) intervariable time $\mu_A = \frac{1}{\lambda_A}$
- ✓ Average (mean) service time $\mu_S = \frac{1}{\lambda_S}$
- ✓ Utilization or arrival-to-service ratio (r) = $\frac{\lambda_A}{\lambda_S} = \frac{\mu_S}{\mu_A}$

The ‘r’ is the important parameter. It shows whether or not a system can function under the current or even higher rate of arrivals and how much the system is over-loaded or under-loaded.

Random variables of a queuing system

$Xs(t)$ = number of jobs receiving service at time t

$Xw(t)$ = number of jobs waiting in a queue at time t

$X(t) = Xs(t) + Xw(t)$ = the total number of jobs in the system at time t

S_k = service time of the k-th job

W_k = waiting time of the k-th job

$R_k = S_k + W_k$ = response time, the total time a job spends in the system from its arrival until the departure

A queuing system is stationary if the distributions of S_k, W_k and R_k are independent of k.