What is a queue?

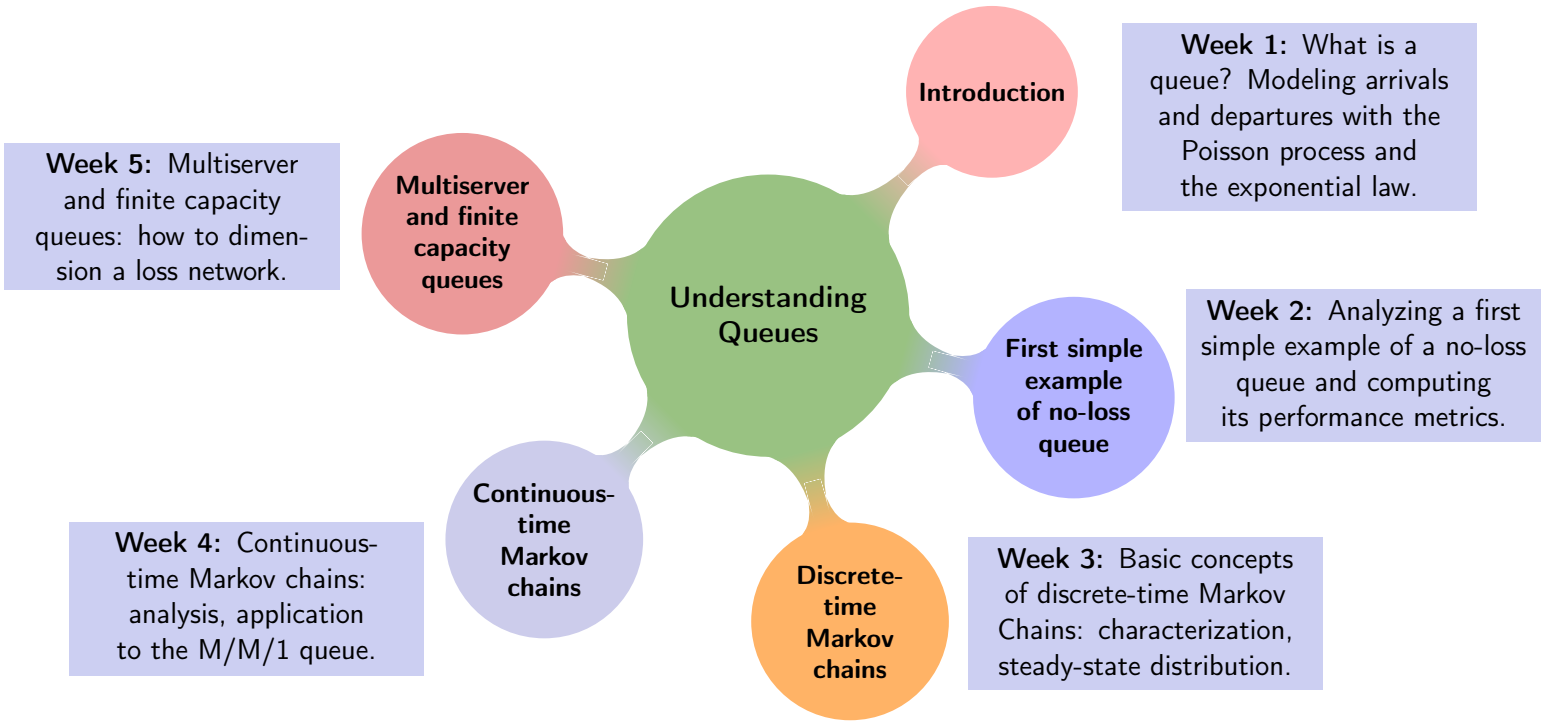
Course objectives

Hello, my name is Sandrine. I am going to introduce this MOOC “Understanding Queues”.

Course objectives

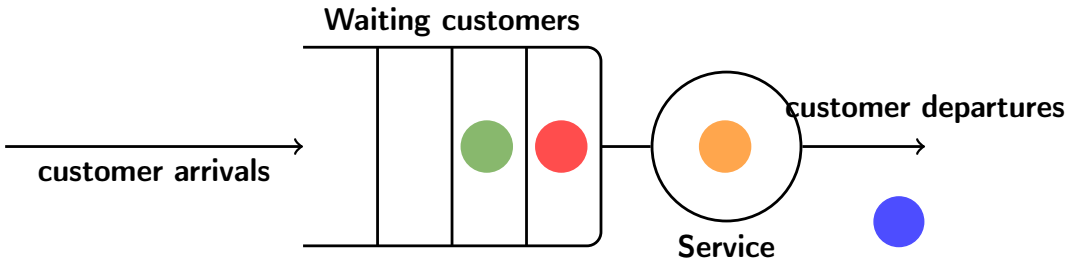
- modeling of **waiting phenomena**
- characterization of those phenomena through **mathematical models**
- evaluation of **average system performance**
- **system dimensioning**

First of all, what are the course objectives ?
Queuing theory aims to model waiting or blocking phenomena. To be more precise, in queuing theory, these phenomena are characterized by mathematical models.
This makes it possible to compute average system performance such as an average delay or a blocking probability.
Reciprocally it is possible to dimension system resources in order to reach a given performance goal.



This course will be divided into 5 weeks.
Week 1 is an introduction to queuing theory. We will introduce basic notions such as arrivals and departures. Particular attention will be paid to the Poisson process and to exponential distribution, two important particular cases of arrivals and service times.
During week 2 we will analyze a first simple example of a no-loss queue, the so-called M/M/1 queue, and we will compute its average performance metrics.
Week 3 will be dedicated to a basic course in discrete time Markov chains. We will learn how they are characterized and how to compute their steady-state distribution.
Then in week 4 we will move on to continuous time Markov chains. Again we will learn how to characterize them and how to analyze their steady-state distribution. Equipped with these tools we will then analyze the M/M/1 queue.
In week 5 we will study multi-server and finite-capacity queues and study how to dimension a loss network.

Queuing systems: a first simple example



Let us start with a very simple example. Consider a queue with one server and an infinite waiting buffer. Clients need a service that can be provided by the server, and if the server is empty they can wait in the buffer until the server is free.
Here, when the first client arrives – the blue one – the server is free so the blue client starts to be served immediately.
A second orange colored client arrives before the service of the blue client has finished, so that the orange client is stored in first position in the buffer.
A third red colored client arrives and the server is still busy serving the blue client. So the red client waits in second position in the buffer. Again a new green colored client arrives and is stored in the next position in the buffer.
Then in this example, when the blue client is no longer being served it leaves the system and releases the server. The first client in the waiting buffer – the orange one – then starts being served and all the other clients move forward one position in the buffer.

Queuing theory

Queuing theory is used to model problems of concurrent access to a shared resource:

- highway toll station, line at a restaurant, airport security check, etc...
- bandwidth, server, disk, memory, etc...

Queuing theory is used to model any problem of concurrent access to a shared resource. The modeled situation can be quite varied : a highway toll station, a line at a restaurant, an airport security check, for example... It is also very useful to model situations that occur in computer science such as concurrent access to bandwidth, to a server, or to disk or memory.

Queuing systems problems

- **performance evaluation:** evaluate performance parameters
- **dimensioning:** how to dimension the system to guarantee a target Quality of Service (QoS) level?

Queuing theory aims to solve problems of two types : performance evaluation, and system dimensioning. In performance evaluation we seek to evaluate performance parameters such as a blocking probability, an average delay or a server's utilization rate. In system dimensioning a target Quality of Service level (average delay, or blocking probability for example) is given and the aim is to dimension resources (such as a number of servers or the size of a waiting buffer) in order to reach this Quality of Service.