COMP9334
Capacity Planning for Computer Systems and Networks

Week 2A: Operational Analysis (2).

Workload Characterisation

Last lecture

- Modelling a computer system as a queueing network

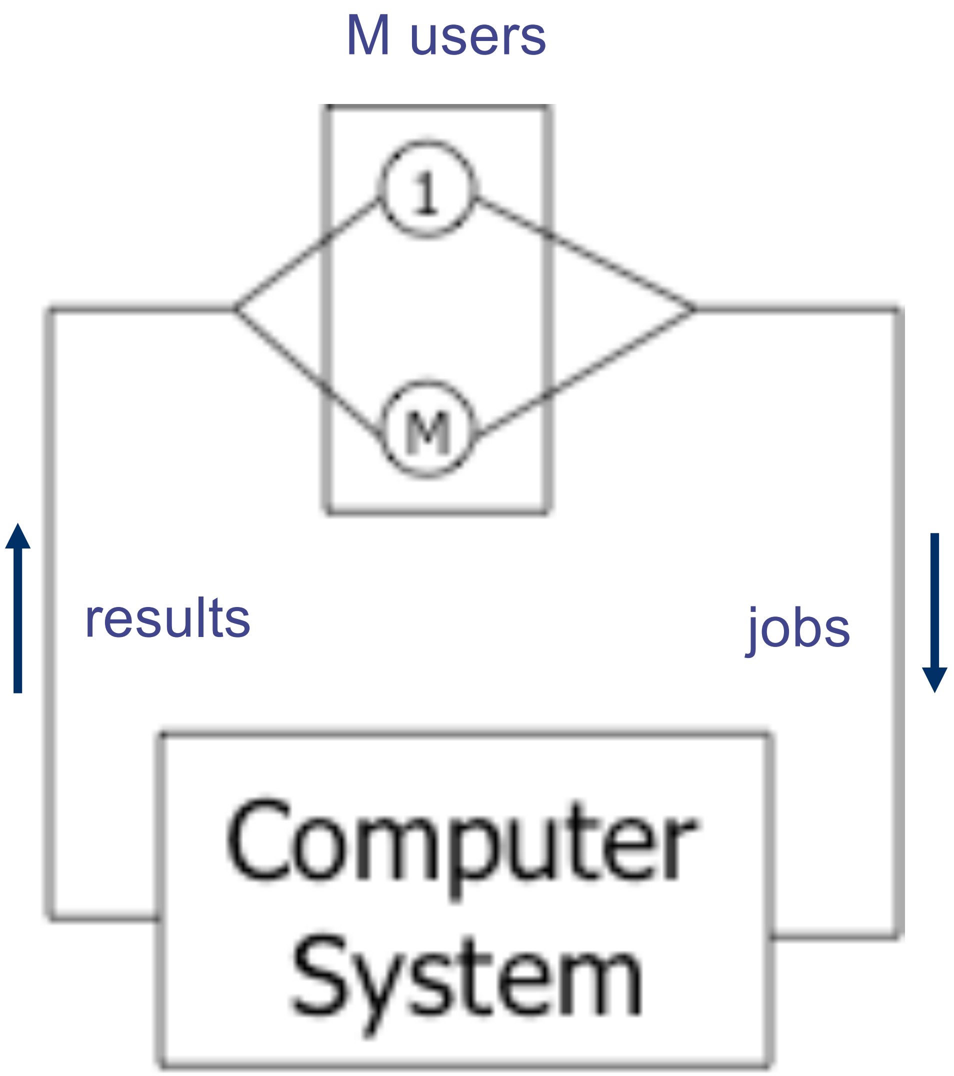
- Operational analysis on queueing networks

- We have derived these operational laws
  - Utilisation law U(j) = X(j) S(j)
  - Forced flow law X(j) = V(j) X(0)
  - Service demand law D(j) = V(j) S(j) = U(j) / X(0)
  - Little's law N = X R

This lecture

- Operational analysis (Continued)
  - Using operational law for
    - Performance analysis
    - Bottleneck analysis
- Workload characterisation
  - Poisson process and its properties

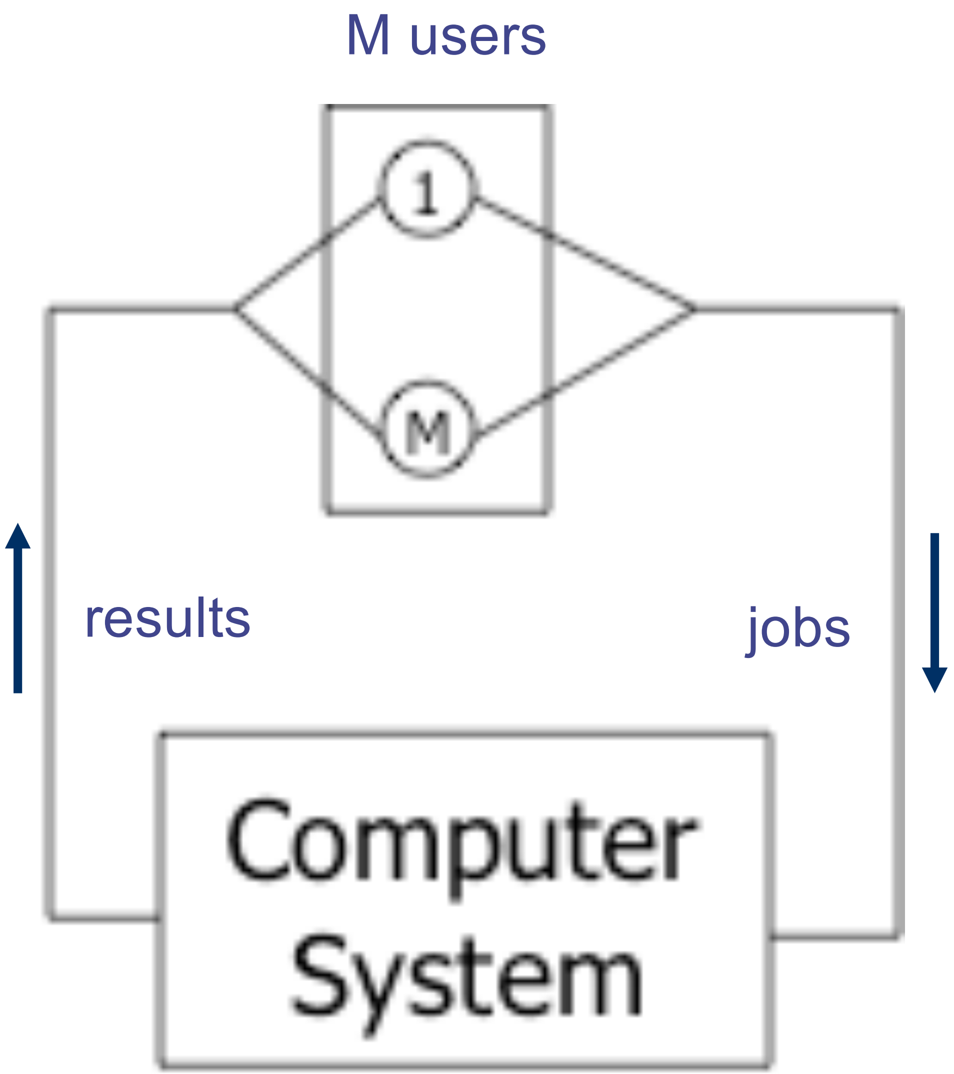
Interactive systems



- An interactive system is used to model the interaction between humans (users) and computers

- The system consists of
  - A number of users
  - A computer system

Interactive systems (Cont'd)



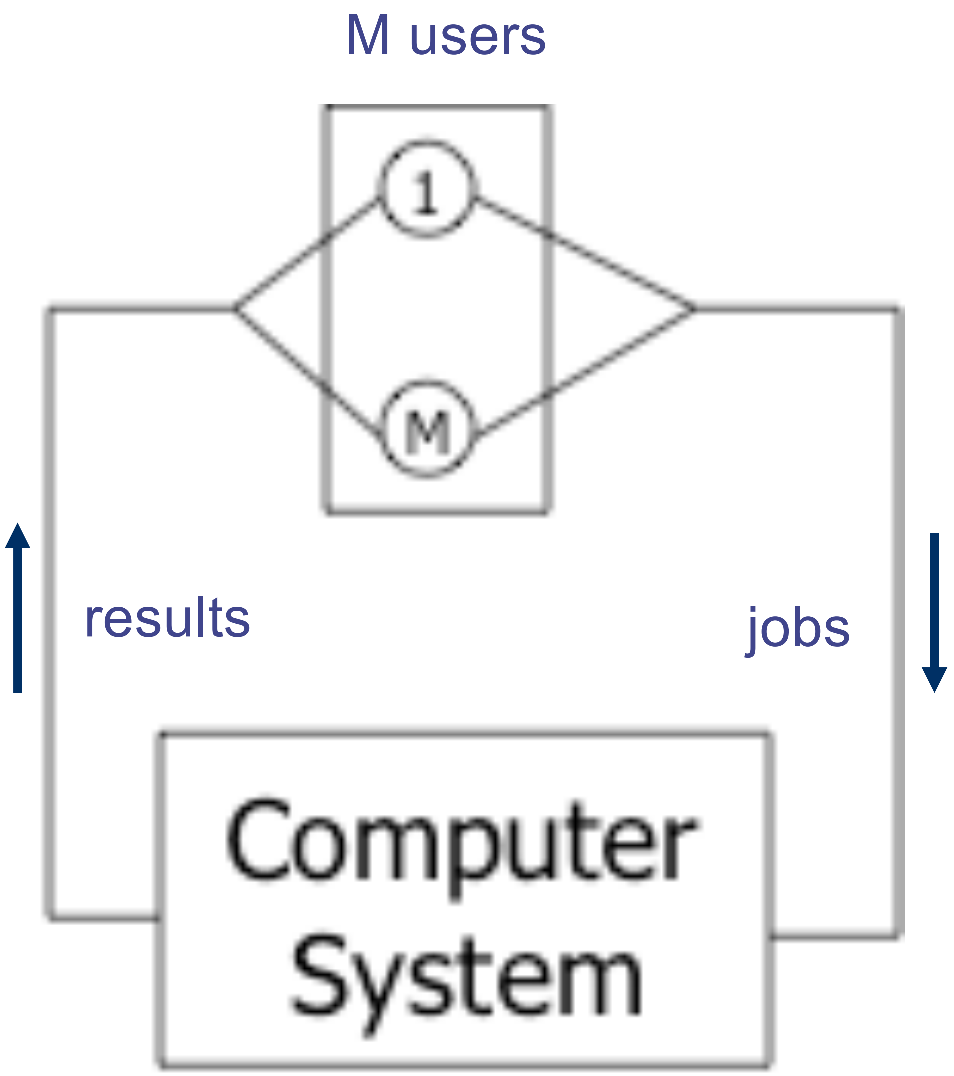
M users

results

jobs

- Interactions
  - Users send jobs to computer systems
  - After finishing processing a job, the computer system returns the result to the user
  - A user, after inspecting the results from the computer system, will send another job to the system

Interactive systems: Modelling assumptions

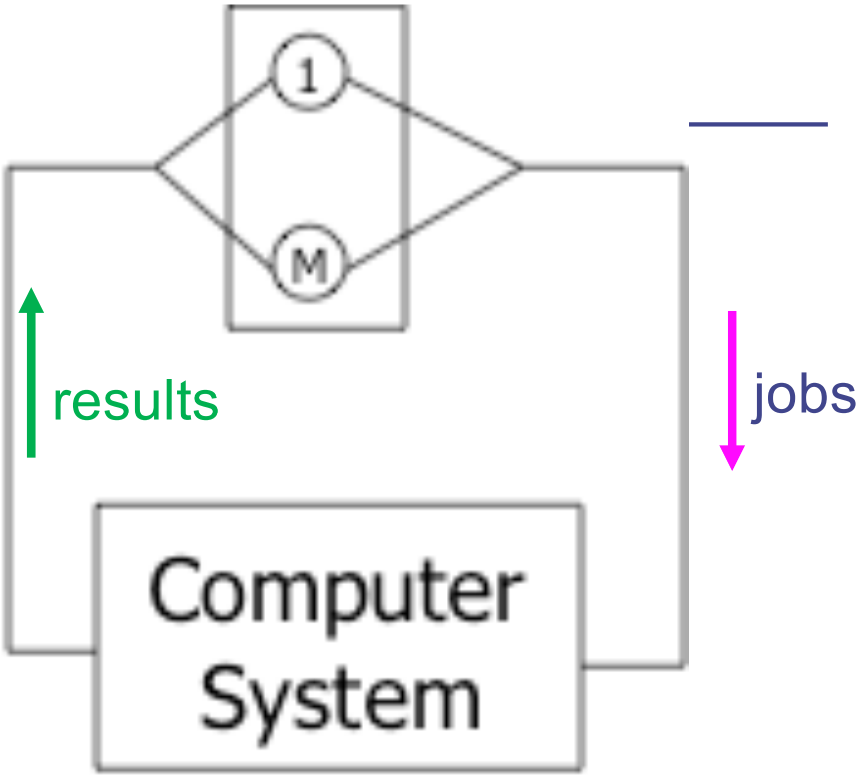

M users

results

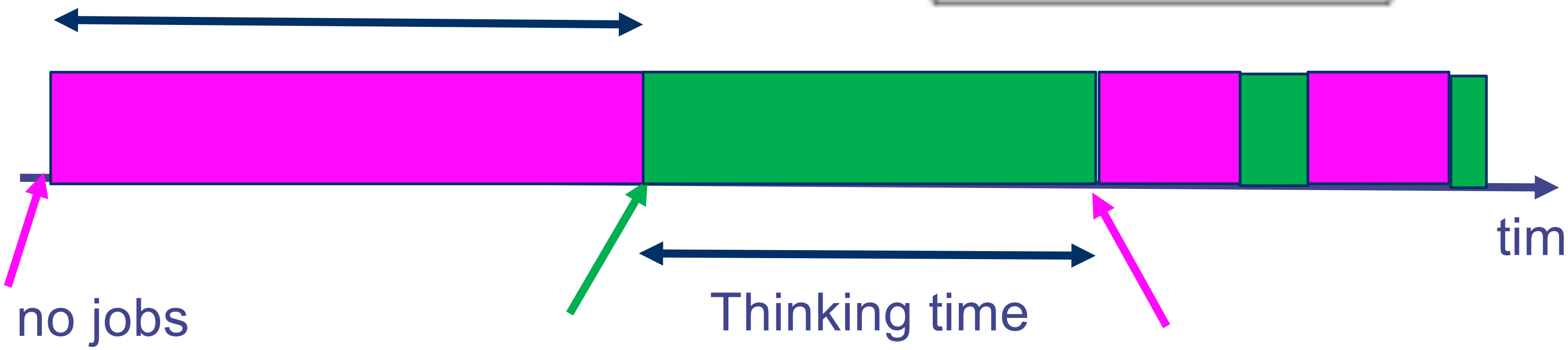
jobs

- Analyze interactive systems with specific assumptions
  - Fixed number of users denoted by M
  - Each user can have at most 1 job at the computer system
  - Each user goes through a cycle consisting of
    - Thinking time
    - Waiting for result time

Interactive cycle



- The time the job from User 1 spends in the computer system
- User 1 waiting for the results



Thinking time

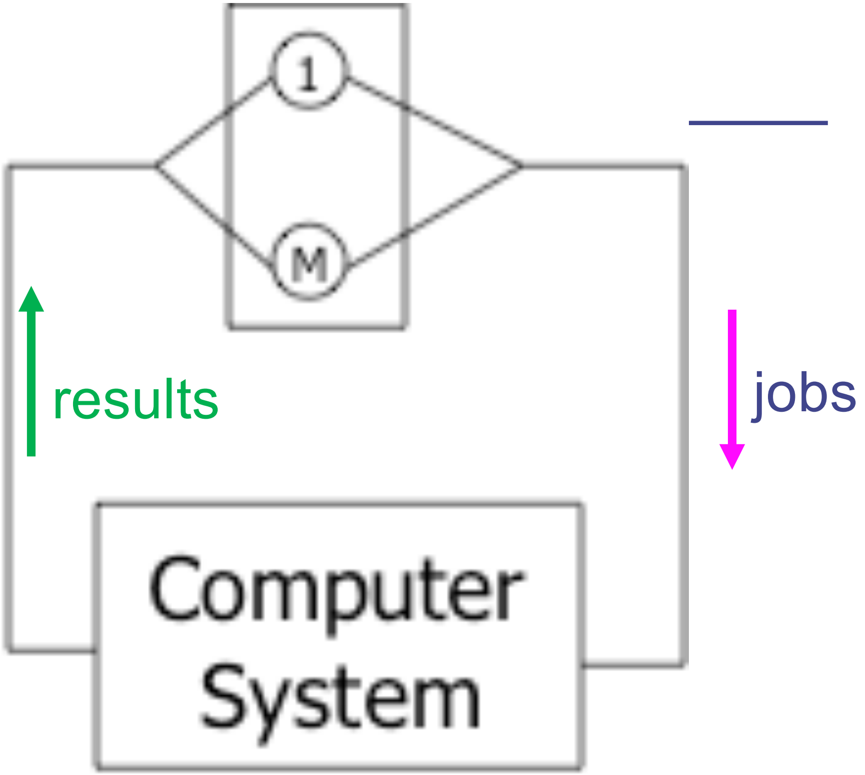
User 1 has no jobs at the system. They send a job to the computer system at this time.

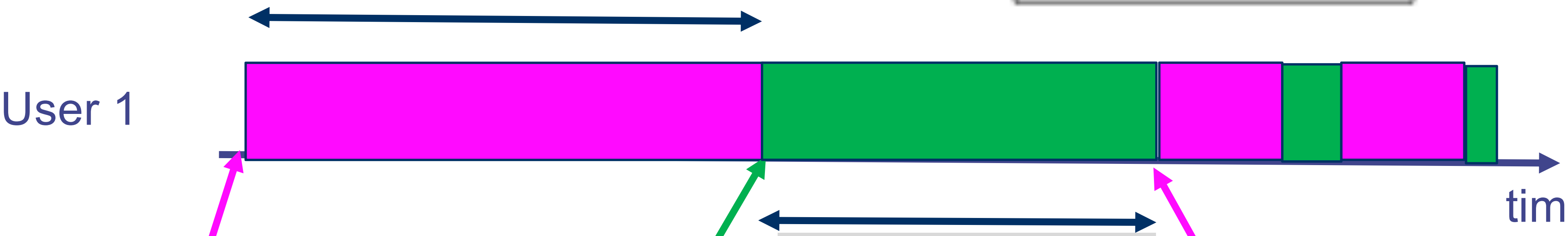
Results are returned to the user

User 1 sends a job to the computer system

Interactive cycle

- User 1’s perspective: waiting for the result of their job
- Computer system’s perspective: Response time of the job from User 1





User 1 has no jobs at the system. They send a job to the computer system at this time.

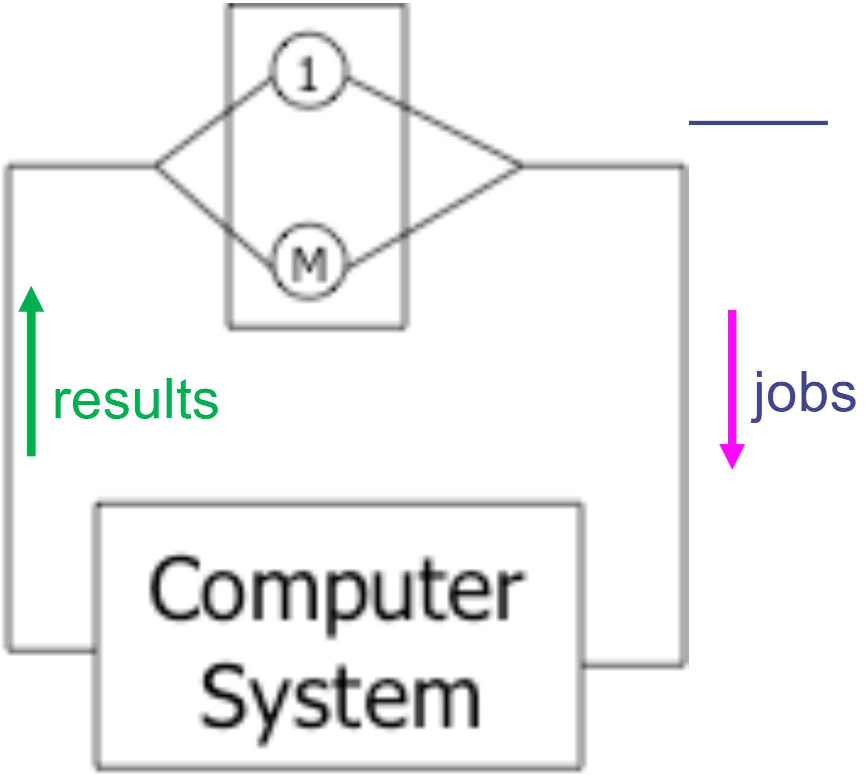
Results are returned to the user

Thinking time = Processing time of the user. That’s why a user is depicted as a circle

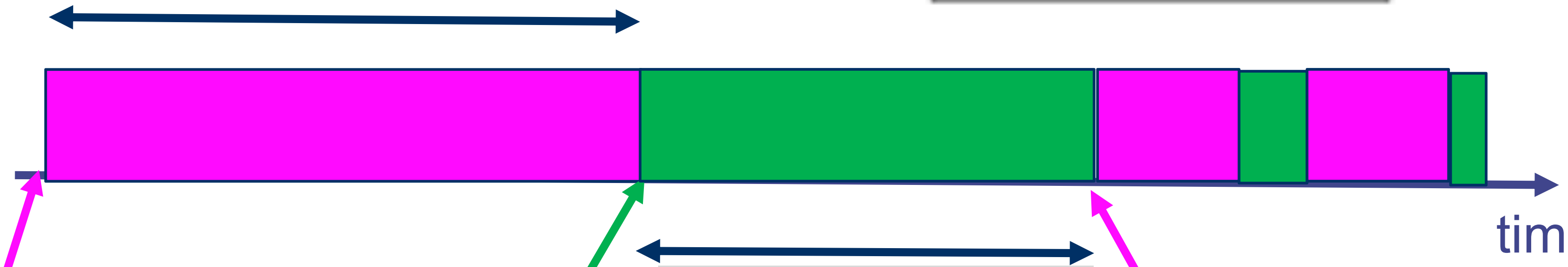
User 1 sends a job to the computer system

Quiz

- Question: At any time, what is the sum of the number of busy users and the number of jobs at the computer system? M



- User idling



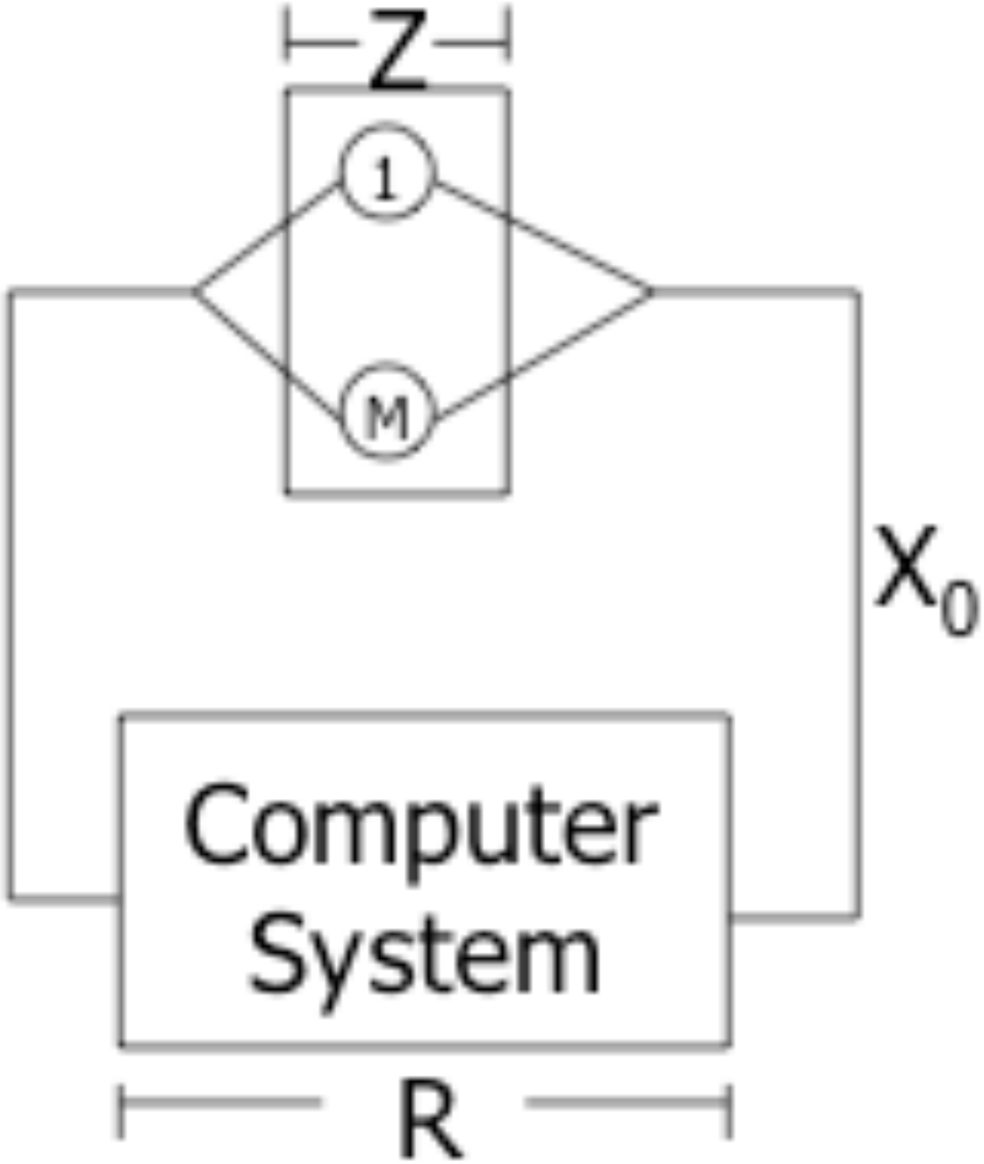
User 1 has no jobs at the system. They send a job to the computer system at this time.

Results are returned to the user

Thinking time = User busy

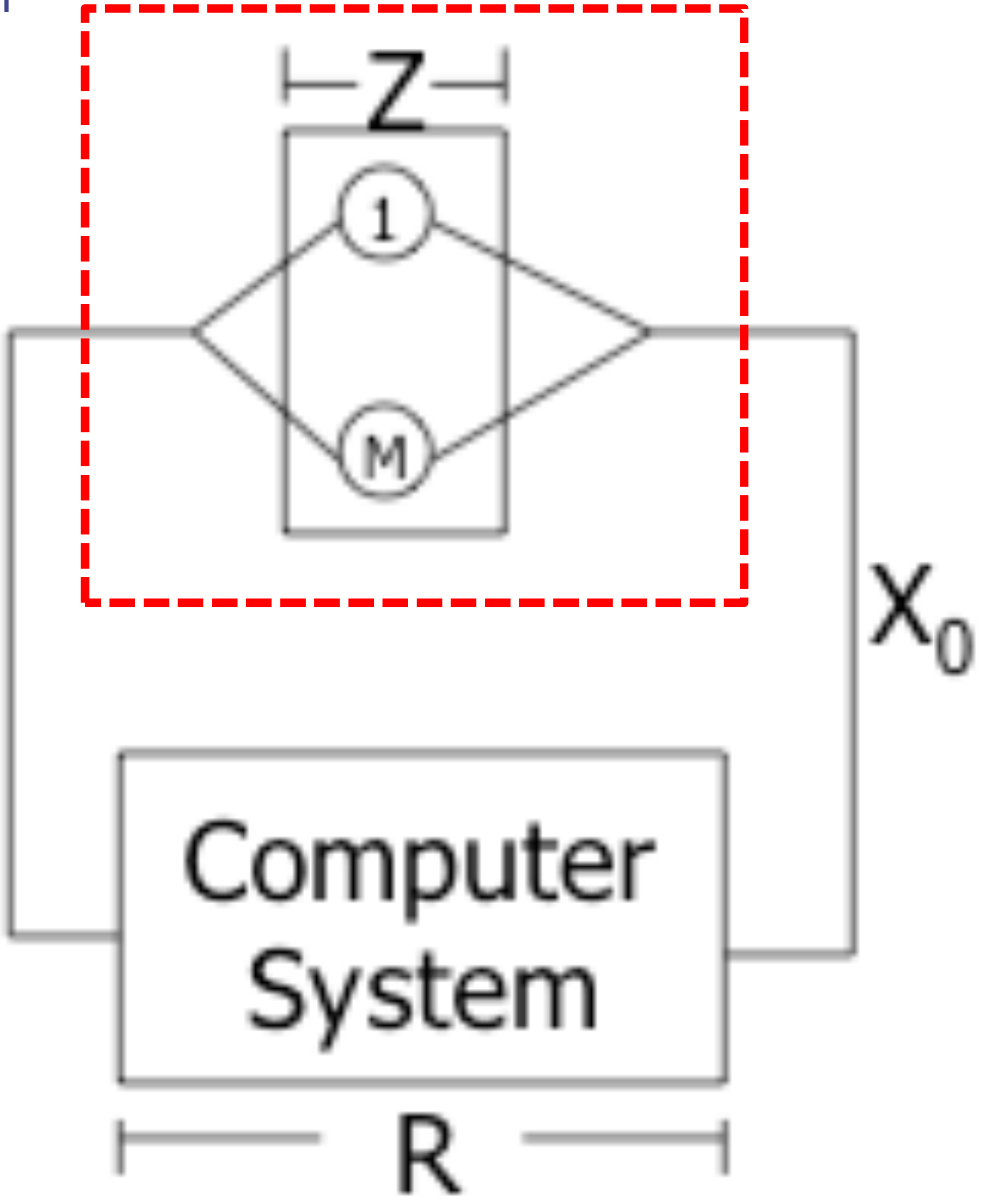
User 1 sends a job to the computer system

Interactive system: Parameters



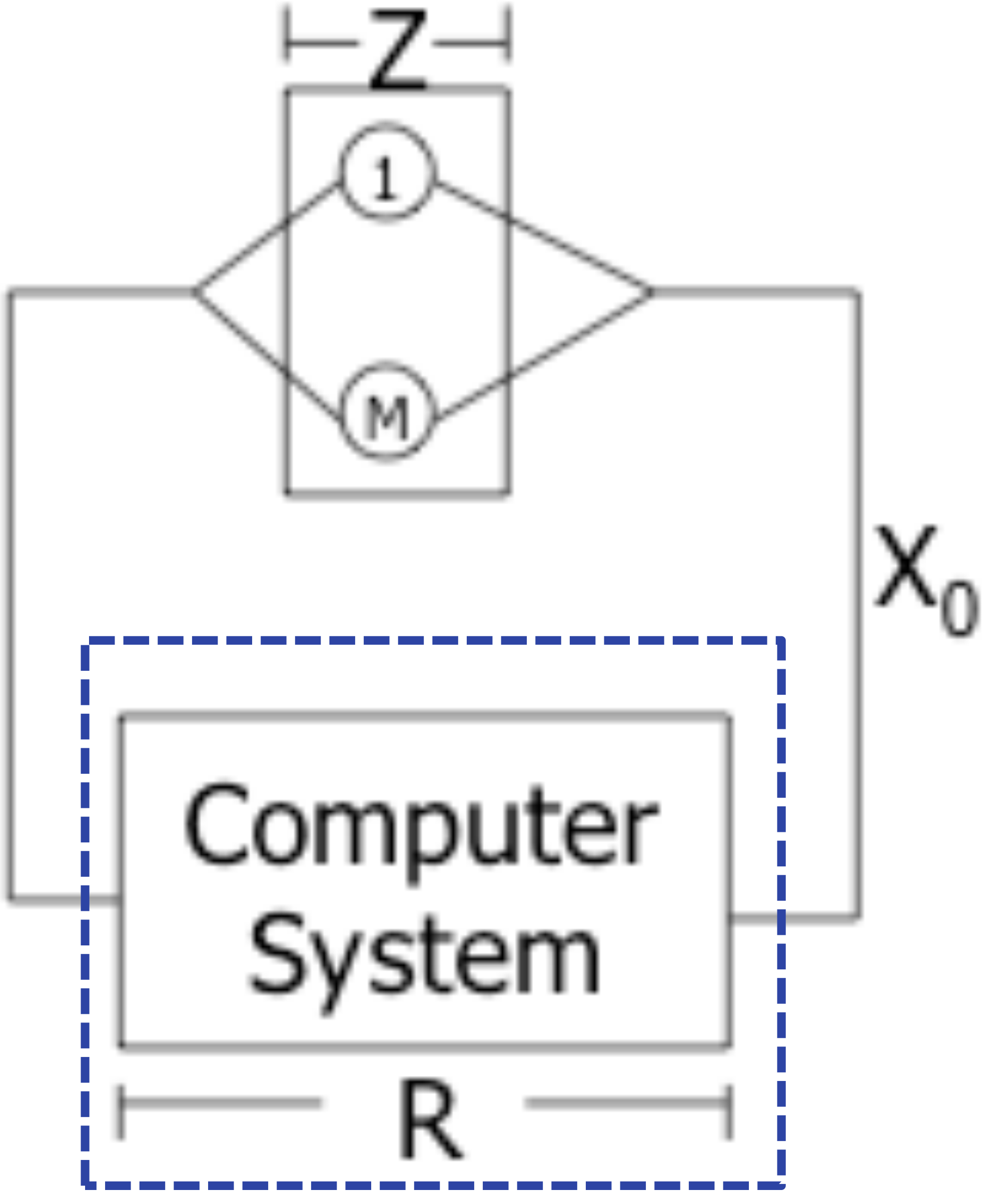
- M interactive users
- Z = mean thinking time
- R = mean response time of the computer system
- X0 = throughput

Analyzing interactive system: Quiz 1



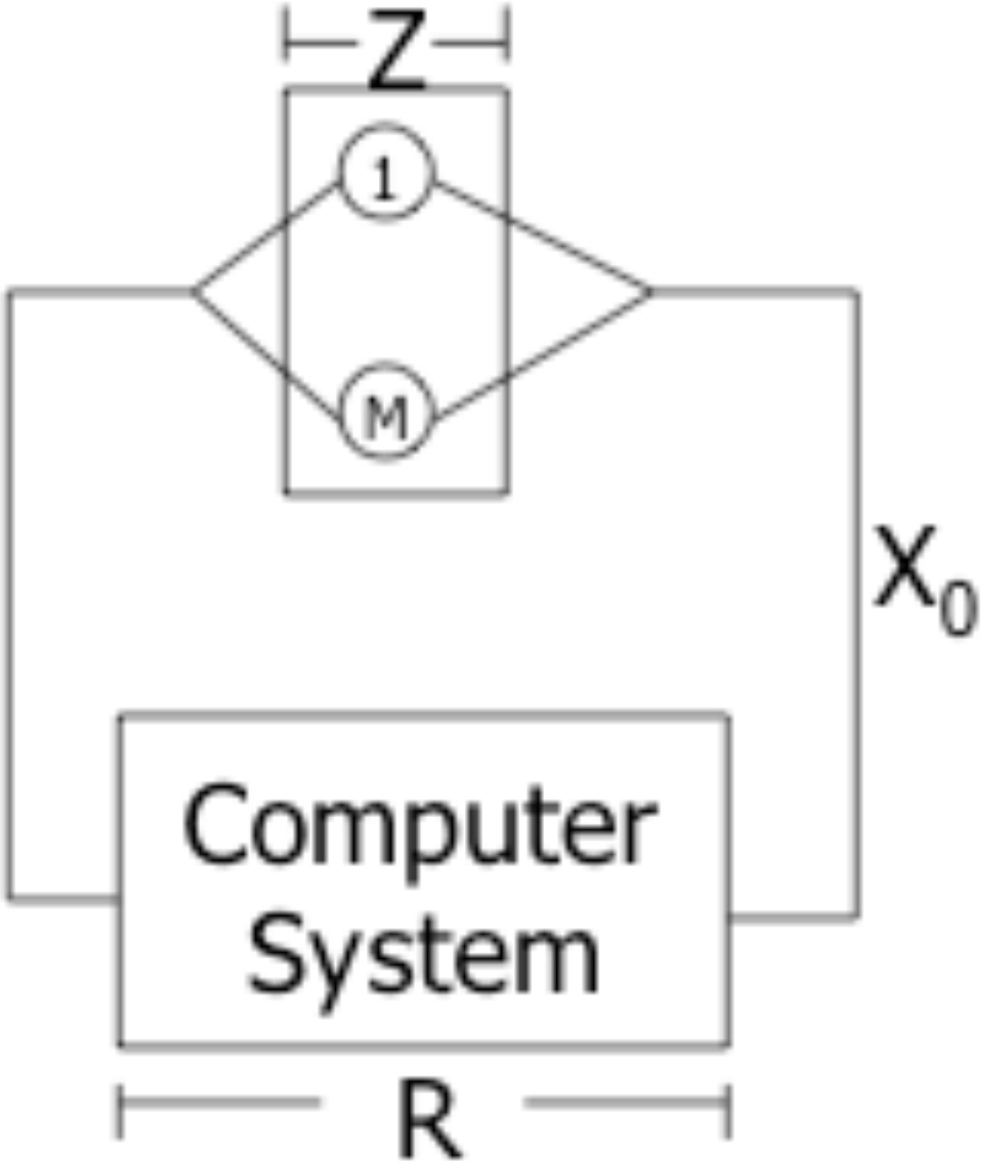
- Mavg = mean # busy users
- Z = mean thinking time
- X0 = throughput
- Apply Little's Law to the red box. What do you get?
  - Mavg = Z * X0

Analyzing interactive system: Quiz 2



- Navg = average # jobs in the computer system
- R = mean response time at the computer system
- X0 = throughput
- Apply Little's Law to the computer system (i.e. the blue box), what do you get?
  - Navg = R * X0

Analyzing interactive system: Quiz 3

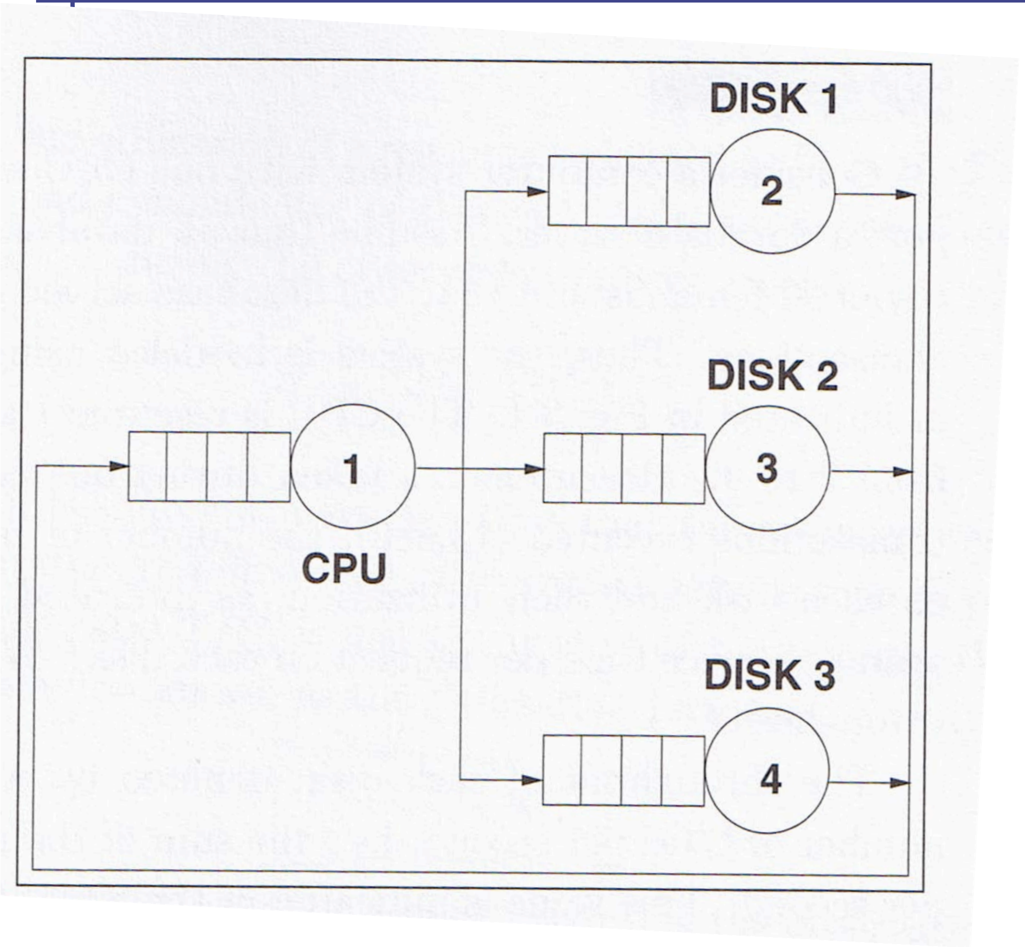


- Quiz 1: Mavg = Z * X0
- Quiz 2: Navg = R * X0
- What is Mavg + Navg?
  - M
- Interactive response time law
  - M = X0*(Z+R)

The operational laws

- These are the operational laws
  - Utilisation law U(j) = X(j) S(j)
  - Forced flow law X(j) = V(j) X(0)
  - Service demand law D(j) = V(j) S(j) = U(j) / X(0)
  - Little's law N = X R
  - Interactive response time M = X(0) (R+Z)
- Applications
  - Mean value analysis (later in the course)
  - Bottleneck analysis
  - Modification analysis

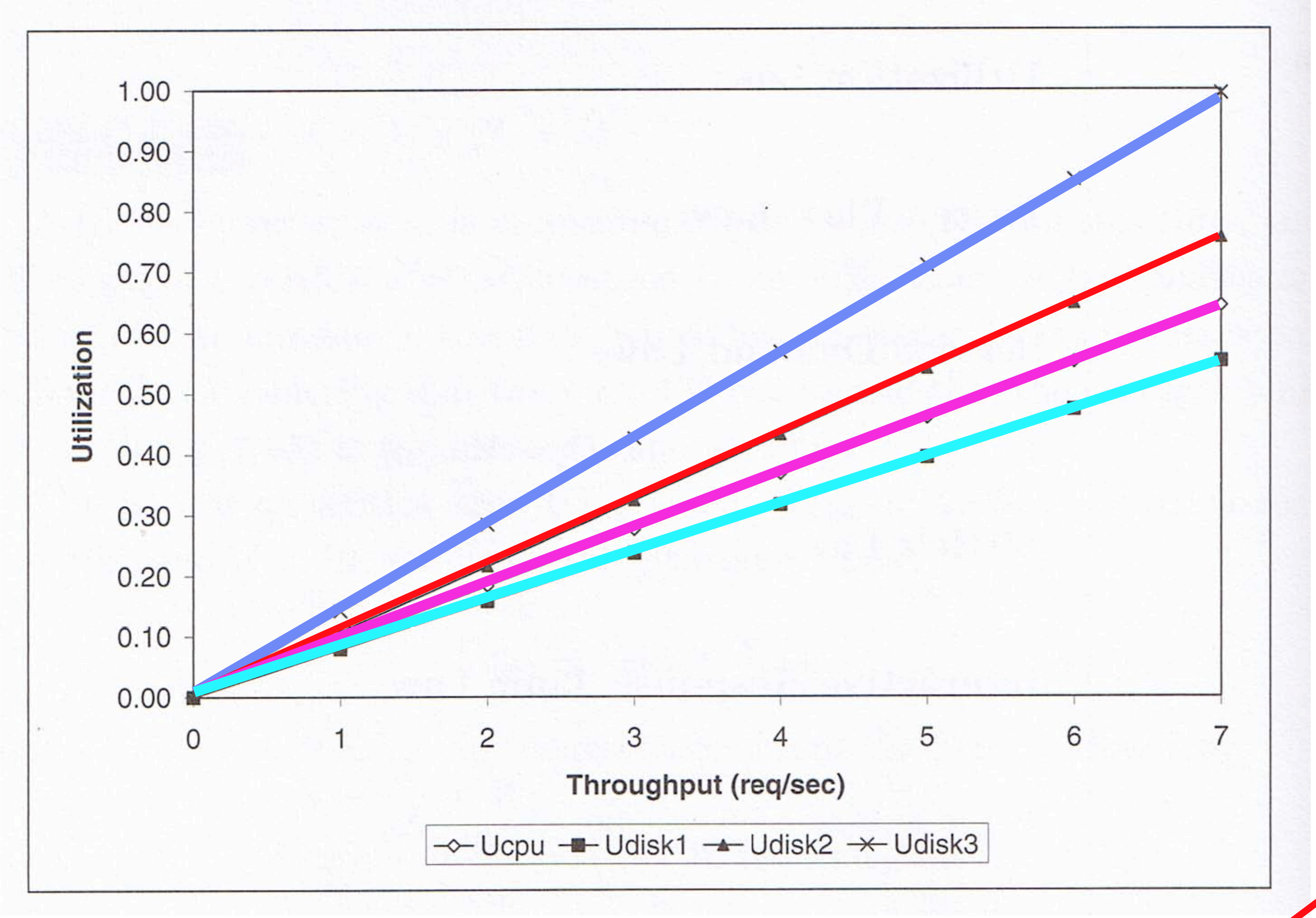
Bottleneck analysis - motivation



	D(j)	Utilisation
Disk 1	79ms	0.30
Disk 2	108ms	0.41
Disk 3	142ms	0.54
CPU	92ms	0.35

Service demand law: D(j) = U(j) / X(0)
==> U(j) = D(j) X(0)
Utilisation increases with increasing throughput and service demand

Utilisation vs. throughput plot U(j) = D(j) X(0)



Disk 3

Disk 2

CPU

Disk 1

What determines this order?

service demand

Observation: For all system throughput:
Utilisation of Disk 3 > Utilisation of Disk 2 >
Utilisation of CPU > Utilisation of Disk 1

Bottleneck analysis

- Recall that utilisation is the busy time of a device divided by measurement time
    - What is the maximum value of utilisation? 1
- Based on the example on the previous slide, which device will reach the maximum utilisation first?

Bottleneck (1)

- Disk 3 has the highest service demand
- It is the bottleneck of the whole system

Operational law: $X(0) = \dfrac{U(j)}{D(j)}$

Utilisation limit: $U(j) \leq 1$

$$\left.\begin{array}{l} X(0) = \dfrac{U(j)}{D(j)} \\ U(j) \leq 1 \end{array}\right\} \quad X(0) \leq \frac{1}{D(j)}$$

Bottleneck (2)

$$X(0) \leq \frac{1}{D(j)}$$

Should hold for all K devices in the system

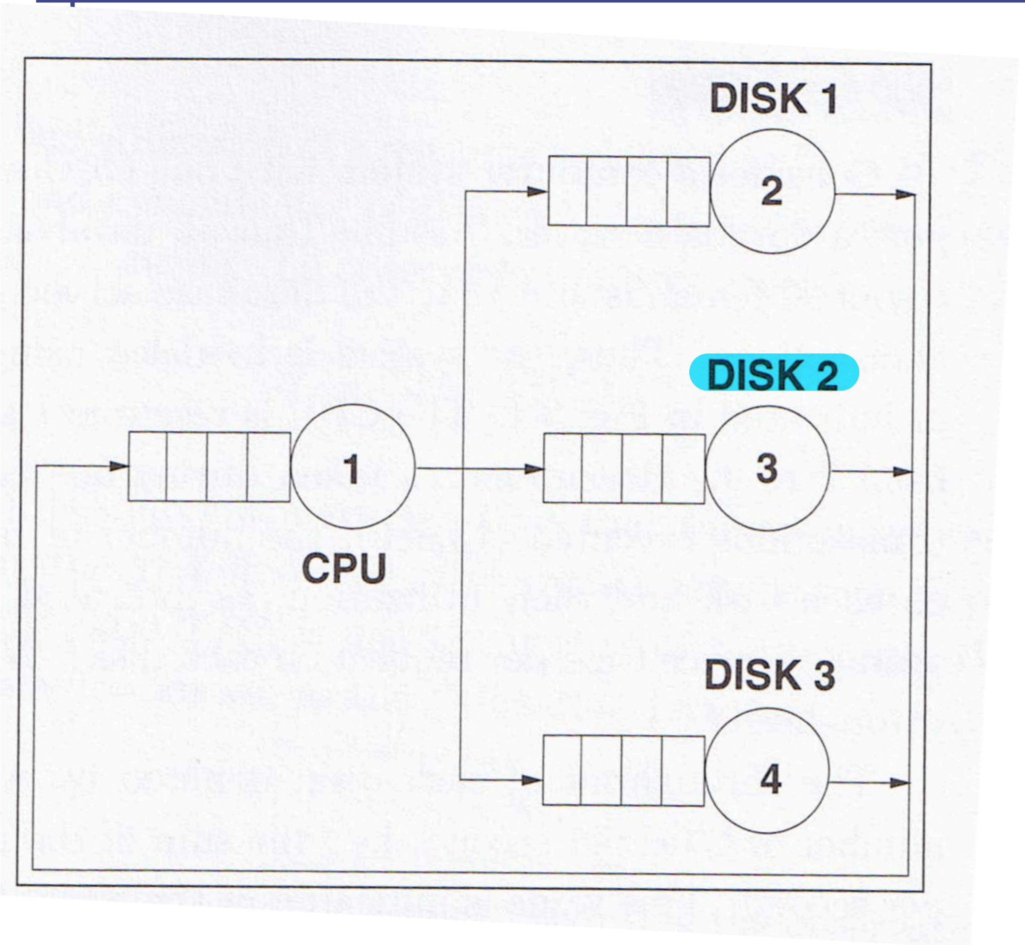
$$i.e. X(0) \leq \frac{1}{D(1)}, \ldots, X(0) \leq \frac{1}{D(K)}$$

$$\Rightarrow X(0) \leq \min \frac{1}{D(j)}$$

$$\Rightarrow X(0) \leq \frac{1}{\max D(j)}$$

Bottleneck throughput is limited by the maximum service demand

Bottleneck exercise



	D(j)	Utilisation
Disk 1	79ms	0.30
Disk 2	108ms	0.41
Disk 3	142ms	0.54
CPU	92ms	0.35

The maximum system throughput is 1 / 0.142 = 7.04 jobs/s. What if we upgrade Disk 3 by a new disk that is 2 times faster, which device will be the bottleneck after the upgrade? You can assume that service time is inversely proportional to disk speed.

Another throughput bound

- Little's law

$$N = R \times X(0) \geq (\sum_{i=1}^{K} D_i) \times X(0)$$

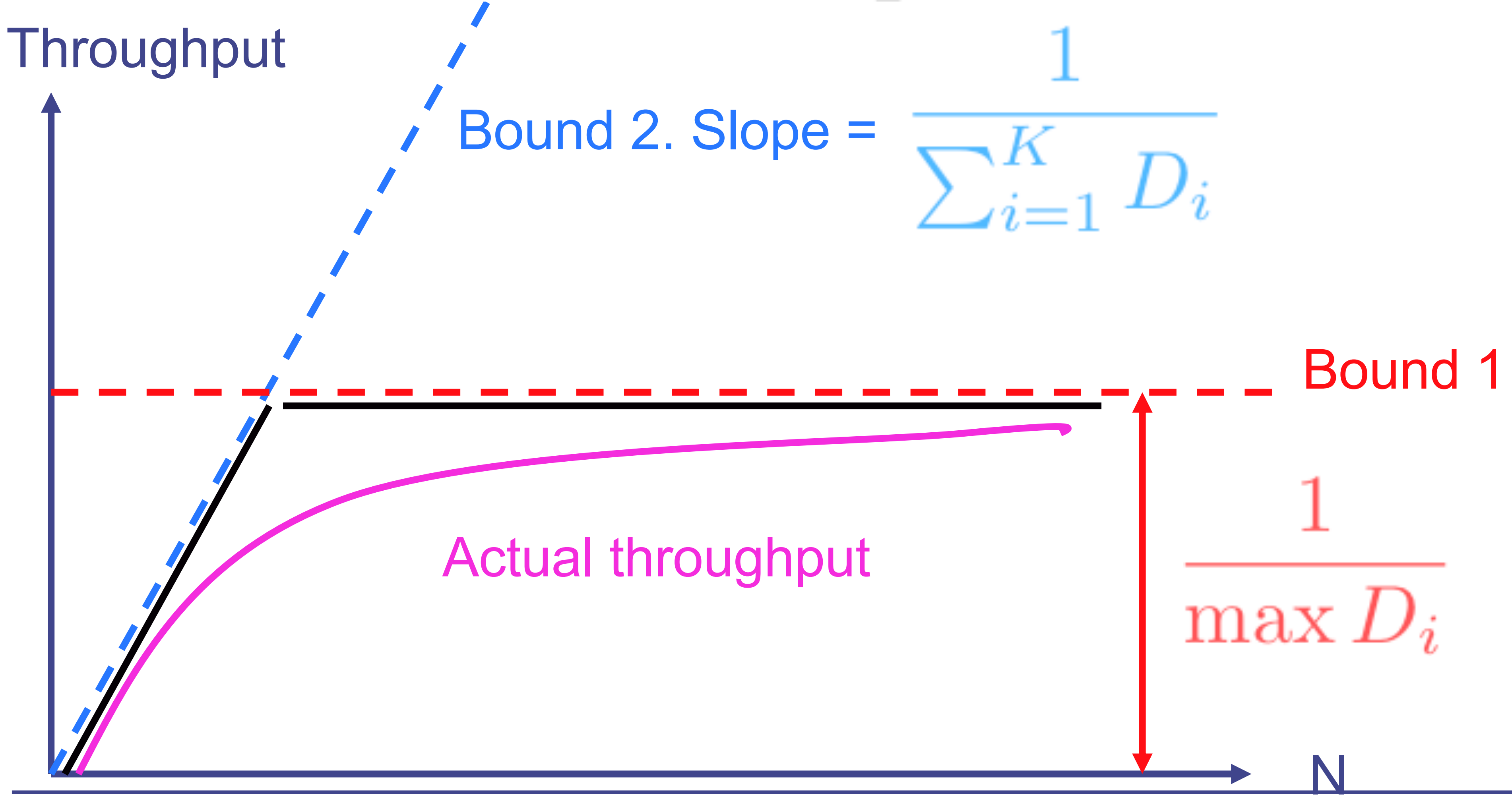
$$\Rightarrow X(0) \leq \frac{N}{\sum_{i=1}^{K} D_i}$$

Previously, we have $X(0) \leq \frac{1}{\max D(j)}$

Therefore: $X(0) \leq \min \left[\frac{1}{\max D_i}, \frac{N}{\sum_{i=1}^{K} D_i} \right]$

Throughput bounds

$$X(0) \leq \min \left[\frac{1}{\max D_i}, \frac{N}{\sum_{i=1}^{K} D_i} \right]$$



Bottleneck analysis

- Simple to use
  - Needs only utilisation of various components
- Assumes service demand is load independent

Modification analysis (1)

- (Reference: Lazowska Section 5.3.1)
- A company currently has a system (3790) and is considering switching to a new system (8130). The service demands for these two systems are given below:

System	Service demand (seconds)	
	CPU	Disk
3790	4.6	4.0
8130	5.1	1.9

- The company uses the system for interactive application with a think time of 60s.
- Given the same workload, should the company switch to the new system?
- Exercise: Answer this question by using bottleneck analysis. For each system, plot the upper bound of throughput as a function of the number of interactive users.

Modification analysis (2)

Operational analysis

- Operational analysis allows you to bound the system performance but it does NOT allow you to find the throughput and response time of a system
- To order to find the throughput and response time, we need to use queueing analysis
- To order to use queueing analysis, we need to specify the workload

Workload analysis

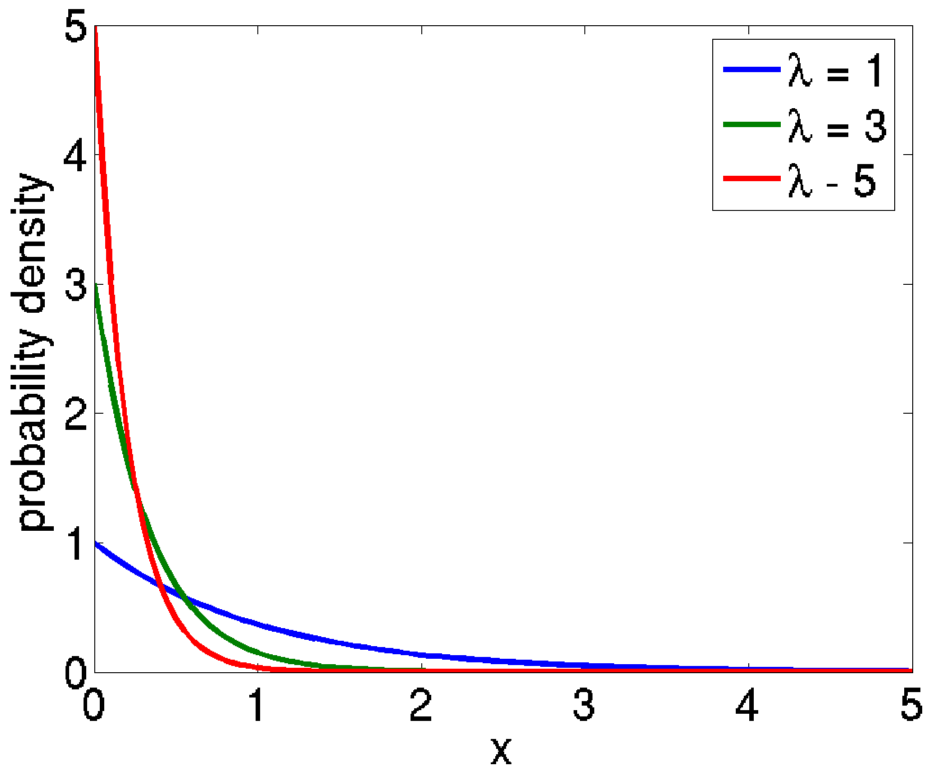
- Performance depends on workload
  - When we look at the performance bound earlier, the bounds depend on number of users and service demand
  - Queue response time depends on the job arrival probability distribution and job service time distribution
    - Recall from Lecture 1A:
      - Uniform arrival times and uniform processing times result in zero waiting time
      - But non-uniform distributions give non-zero waiting time

- Need to specify workload by using probability distribution.
- We will look at a well-known arrival process called Poisson process today.
- Poisson process has a close relationship with exponential distribution and this is our starting point

Exponential distribution (1)

- A continuous random variable is exponentially distributed with rate λ if it has probability density function

$$f(x) = \begin{cases} \lambda e^{-\lambda x} & x \geq 0 \\ 0 & x < 0 \end{cases}$$



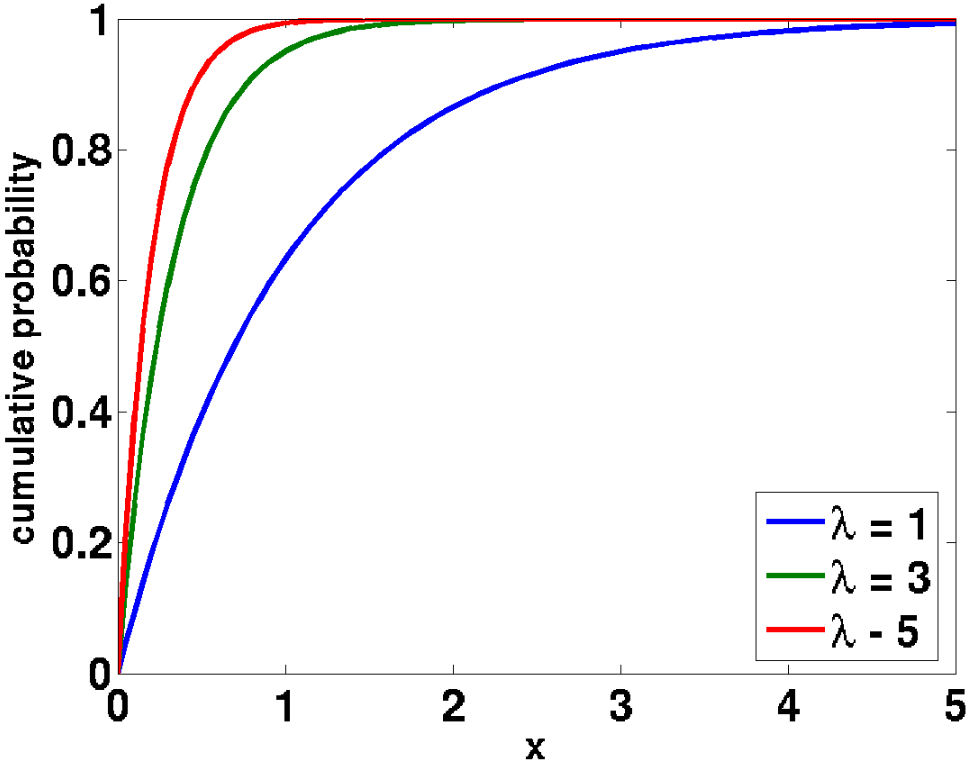
Probability that $x \leq X \leq x + \delta x$ is

$$f(x)\,\delta x = \lambda \exp(-\lambda x)\,\delta x$$

Exponential distribution - cumulative distribution

- The cumulative distribution function F(x) = Prob(X ≤ x) is:

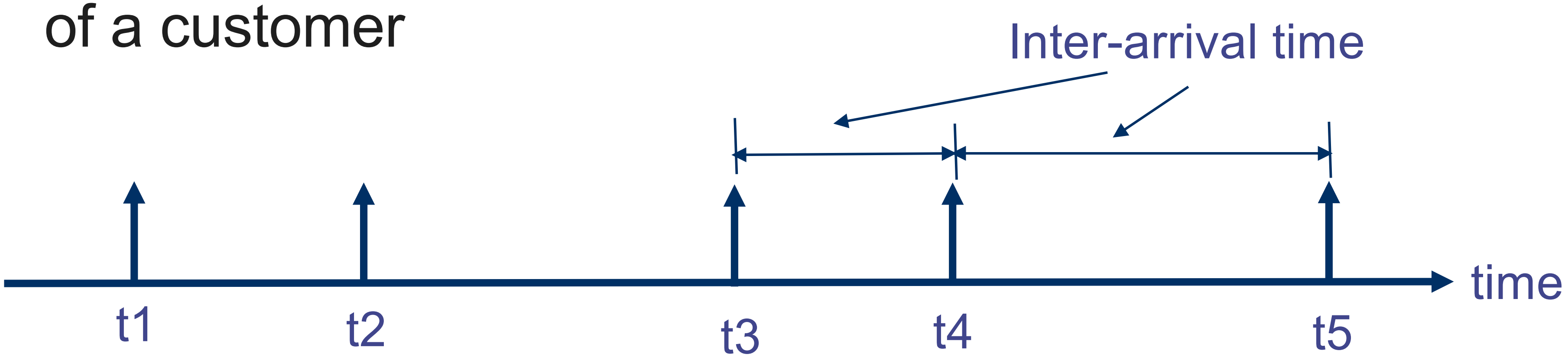
$$F(x) = \int_0^x \lambda e^{-\lambda z} dz = 1 - e^{-\lambda x} \text{ for } x \geq 0$$



What is Prob(X ≥ x)?

Arrival process

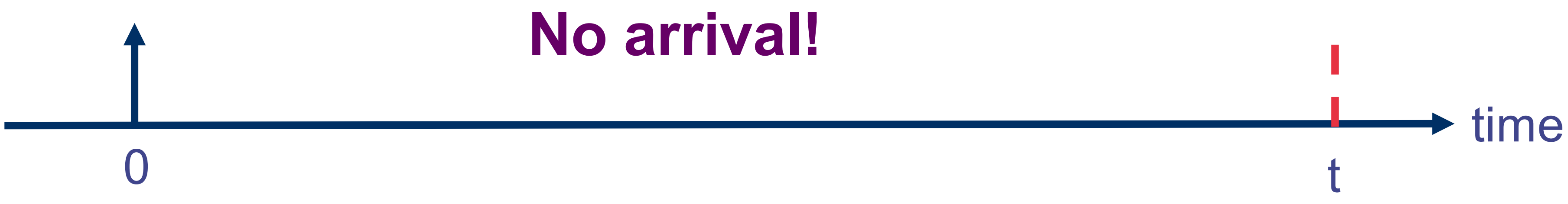
- Each vertical arrow in the time line below depicts the arrival of a customer



- An arrival can mean
  - A telephone call arriving at a call centre
  - A transaction arriving at a computer system
  - A customer arriving at a checkout counter
  - An HTTP request arriving at a web server
- The inter-arrival time distribution will impact on the response time.
- We will study an inter-arrival distribution that results from a large number of **independent** customers.

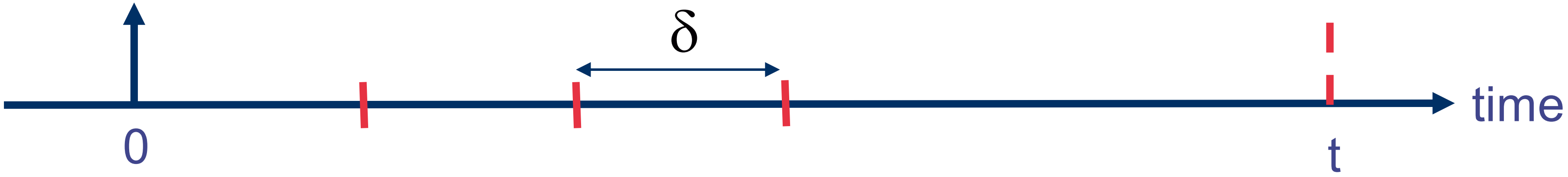
Many independent arrivals (1)

- Assume there is a large pool of N independent customers
- Behaviour of each customer: Within a small time interval of δ, a customer makes a request (or arrives) with a probability of $p\delta$
  - p is a constant
- Quiz: If there are 2 (= N) customers, what is the probability that both of them do not send any request in the time interval δ
  - Answer:
- If a customer arrives at time 0, we want to calculate the probability that the next customer does not arrive before time t
  - Why is this interesting? No arrival before time t = Inter-arrival time is at least t



Many independent arrivals (2)

- Aim: Want to find the probability of no arrivals in [0,t]
- Divide the time t into intervals of width δ



- No arrival in [0,t] = no arrival in each interval δ from N users
- Probability of no arrival in δ =
- There are t / δ intervals
- Probability of no arrival in [0,t] is

$$(1 - Np\delta)^{\frac{t}{\delta}} \rightarrow e^{-Npt} \text{ as } \delta \rightarrow 0$$

Exponential inter-arrival time

- We have showed

  Probability(no arrival in [0,t]) = exp(- N p t)

- Since we assume that there is an arrival at time 0, this means

  Probability(inter-arrival time > t) = exp(- N p t)

- This means

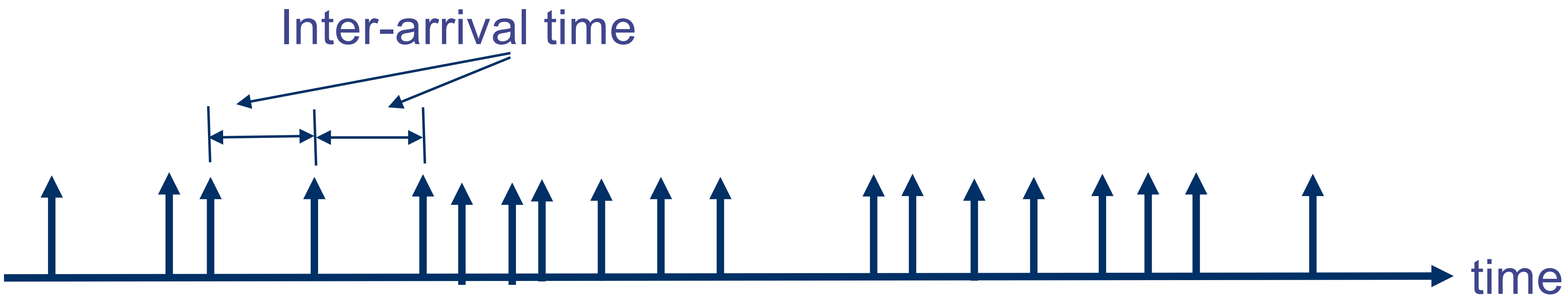
  Probability(inter-arrival time $\leq$ t) = 1 - exp(- N p t)

- What this shows is the inter-arrival time distribution for independent arrival is exponentially distributed

- Define: λ = Np
  - λ is the mean arrival rate of customers

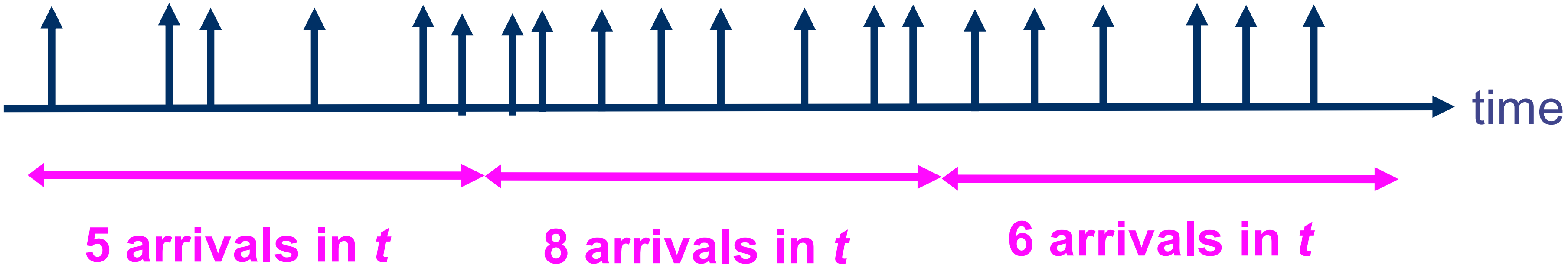
Two different methods to describe arrivals

Method 1: Continuous probability distribution of inter-arrival time



Two different methods to describe arrivals

Method 2: Use a fixed time interval (say *t*), and count the number of arrivals within *t*.

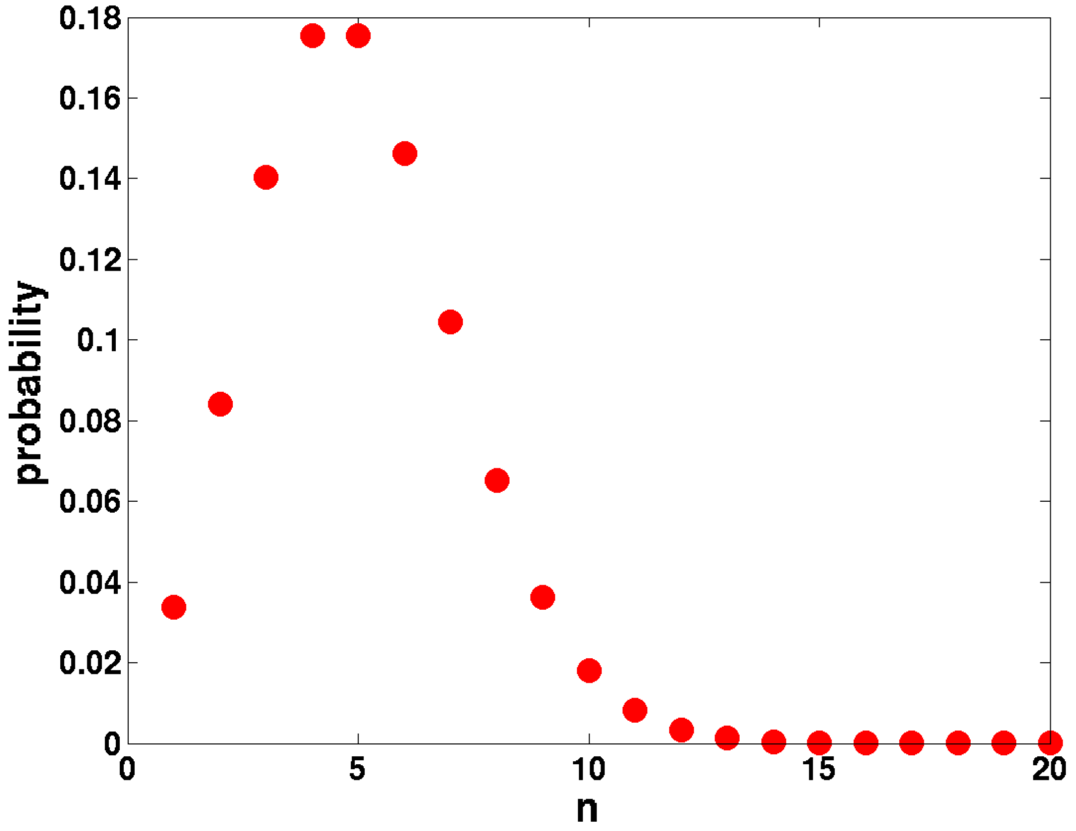


- The number of arrivals in *t* is random
- The number of arrivals must be an non-negative integer
- We need a discrete probability distribution:
  - Prob[#arrivals in t = 0]
  - Prob[#arrivals in t = 1]
  - etc.

Poisson process (1)

- Definition: An arrival process is Poisson with parameter λ if the probability that *n* customer arrive in any time interval *t* is

$$\frac{(\lambda t)^n e^{-\lambda t}}{n!}$$



Example:

Example:
λ= 5 and t = 1

Note: Poisson is a discrete probability distribution.

Poisson process (2)

- Theorem: An exponential inter-arrival time distribution with parameter λ gives rise to a Poisson arrival process with parameter λ

- How can you prove this theorem?
  - A possible method is to divide an interval t into small time intervals of width δ. A finite δ will give a binomial distribution and with $\delta \rightarrow 0$, we get a Poisson distribution.

Customer arriving rate

- Given a Poisson process with parameter λ, we know that the probability of n customers arriving in a time interval of t is given by:

$$\frac{(\lambda t)^n e^{-\lambda t}}{n!}$$

- What is the mean number of customers arriving in a time interval of t?

$$\sum_{n=0}^{\infty} n \frac{(\lambda t)^n e^{-\lambda t}}{n!} = \lambda t$$

- That's why λ is called the arrival rate.

Customer inter-arrival time

- You can also show that if the inter-arrival time distribution is exponential with parameter λ, then the mean inter-arrival time is $1/\lambda$
- Quite nicely, we have

  Mean arrival rate = 1 / mean inter-arrival time

Application of Poisson process

- Poisson process has been used to model the arrival of telephone calls to a telephone exchange successfully
- Queueing networks with Poisson arrival is tractable
    - We will see that in the next few weeks.
- Beware that not all arrival processes are Poisson! Many arrival processes we see in the Internet today are not Poisson. We will see that later.

References

- Operational analysis
  - Lazowska et al, Quantitative System Performance, Prentice Hall, 1984. (Classic text on performance analysis. Now out of print but can be download from http://www.cs.washington.edu/homes/lazowska/qsp/
    - Chapters 3 and 5 (For Chapter 5, up to Section 5.3 only)
  - Alternative 1: You can read Menasce et al, "Performance by design", Chapter 3. Note that Menasce doesn't cover certain aspects of performance bounds. So, you will also need to read Sections 5.1-5.3 of Lazowska.
  - Alternative 2: You can read Harcol-Balter, Chapters 6 and 7. The treatment is more rigorous. You can gross over the discussion mentioning ergodicity.

- Poisson process: Harcol-Balter Chapter 11