COMP9334
Capacity Planning of Computer Systems and Networks

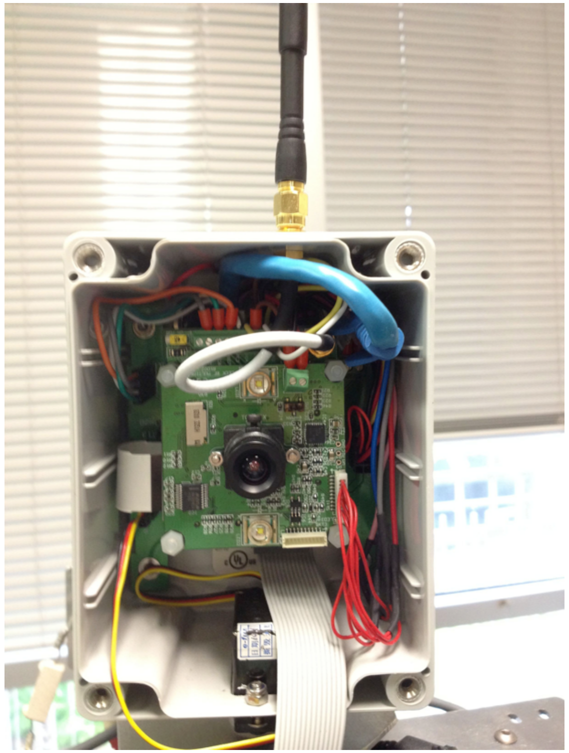
Week 1: Introduction to Capacity Planning

Chun Tung Chou

About your lecturer

http://www.cse.unsw.edu.au/~ctchou/

- Research in Computer Networks and Embedded Systems
- Example research projects
  - Derive efficient algorithms for embedded devices
  - Enabling biological computers to talk to each other
  - Enabling nano-scale devices to talk to each other
- Tools I use in my research
  - Measurements
  - Mathematical analysis
  - Simulation
  - Program and test



Course organisation

- Course web site: www.cse.unsw.edu.au/~cs9334
- Email: cs9334@cse.unsw.edu.au
- Read the course outline
- Lectures: Tue 12-3, CLB6
- Sometimes we may discuss revision problems during the lectures

Course objective:

- Aim: The *design* of computer systems and networks to meet performance specifications
- Example problem: You want to design a computer system that can deal with 400,000 HTTP hits per minute. How can you make sure that your system will meet this demand?
- You will learn how to solve capacity planning problems using *mathematical modelling*.

How to learn?

- Lectures
  - Key concepts, illustration by small examples
  - Don’t just depend the lecture notes, you must
- Read the reference materials too
- Revision problems
  - Try if you can solve the problem
- Try also the exercises in the book
- Use discussion board
  - Don’t think your question is silly, other may have the same problem
- The key is understanding, not memorisation
- Mathematics is something that you can get used to

Resources

- Books and reference materials
  - We will use materials from a number of books
  - Available in library as hard copy or electronically
- Two key books:
    - Menasce et al. Performance by Design. PH. 2004 (Hard copy)
    - Harchol-Balter. Performance Modelling and Design of Computer Systems. CUP, 2013. (Electronic)
- On-line resources
  - Journal and conference articles
  - IEEE and ACM
- Two general books
  - Solving mathematical problems
    - Polya, “How to solve it?” (Highly recommended)
  - How to do well in maths and science
    - Barbara Oakley, “A mind for numbers”.

Assessment

- Three assessment components
  - Assignment (15%)
  - Project (25%)
  - Final exam (open book, no laptop/tablet) (60%)
- Assignment: Extended tutorial questions
- Project: Simulation (coding + statistics)
- Overall mark:
  - C = Assignment + Project -> Rescale C to be out of 100
  - E = Exam mark -> Rescale E to be out of 100
  - Overall mark = weighted harmonic mean of C and E
  - 1 / (0.60/E + 0.40/C)
  - Implication of harmonic mean

Assumed knowledge

- Mathematics
  - Probability
    - Probability density function, independence, conditional probability
  - Statistics
  - Vectors and matrices, linear equations
  - Differentiation and integration

- A good review of probability is in Chapter 3 of Harcol-Balter, “Performance Modeling and Design of Computer Systems”

A quick test on probability

- Probability is fun and very useful, but is sometimes tricky
- Prof. Sheldon Cooper (Big Bang Theory) made a wrong argument in the following clip. Can you use the language of probability to explain his error?
- https://www.youtube.com/watch?v=bjUwSHGsG9o
- Sheldon's reply on why he thought the person's name should be Mohammed Li. "Mohammed is the most common first name in the world. Li the most common surname. As I didn't know the answer, I though that gave me a mathematical edge."

Lecture outline

- Capacity planning
  - Why?
  - What?
- Quality of service metrics
- Quantitative performance analysis ←→ Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Queueing models
  - Queues → Waiting time

Why capacity planning?

Hot eBusiness News

Poor Web Site Performance Is Costing Retailers Millions

Why capacity planning?

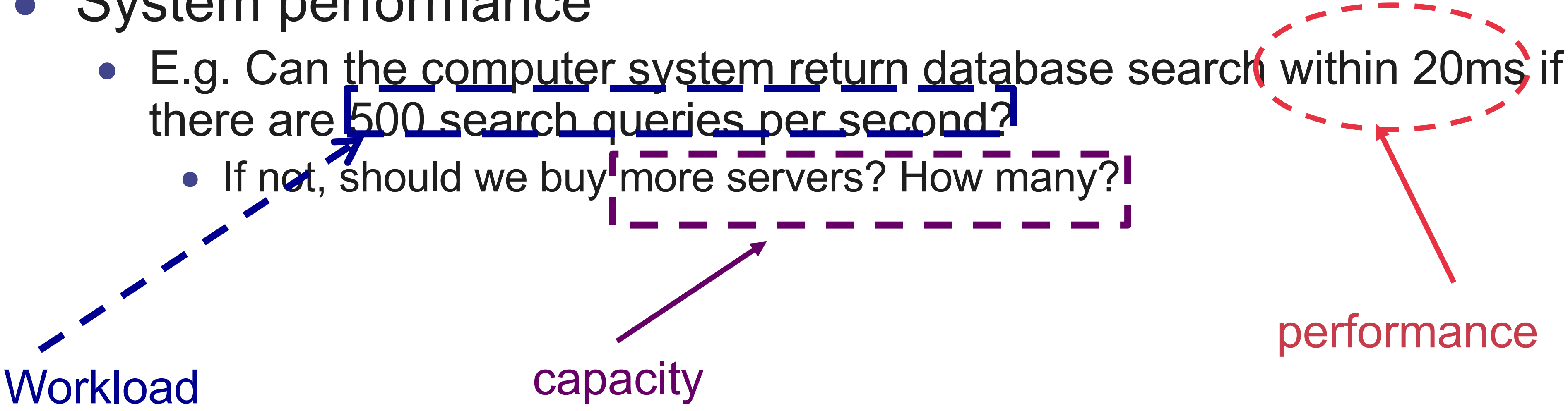
Hot eBusiness News

Poor Web Site Performance Is Costing Retailers Millions

- The aim of capacity planning is to improve *performance* of computer systems by adding “*capacity*”.
- What is performance?
- What is capacity?

Design of an e-Commerce systems

- Functional requirements
  - Product search, database management functions etc
    - Search correctness, algorithmic efficiency
- Computer and network security
- System performance
  - E.g. Can the computer system return database search within 20ms if there are 500 search queries per second?
    - If not, should we buy more servers? How many?



- Can you think of other system performance requirements?

Web search engine

- Say you are planning a computer system which will host a search engine that rivals Google
- Current expected workload
  - 1000 searches per second
- Performance specification
  - Return results within 10ms

performance

- What hardware and network should you use?
  - How many servers? How much disk space? Etc.
- What if workload is expected to increase by 50% in one year, can the system still maintain its performance?

capacity

- Question: Can you think of other capacity parameters?

Capacity planning problems

- Focused on capacity planning of computer systems and networks
- Elements of a capacity planning problems
  - Given:
    - Workload specifications
    - Performance specifications
  - Find:
    - Capacity e.g. hardware or network requirements, personnel requirements etc.

- Capacity planning problems are everywhere in life. Brainstorm with your neighbours to come out with some capacity planning problems in real life. For each problem, you must identify the workload, performance and capacity parameters.

Capacity planning motivations

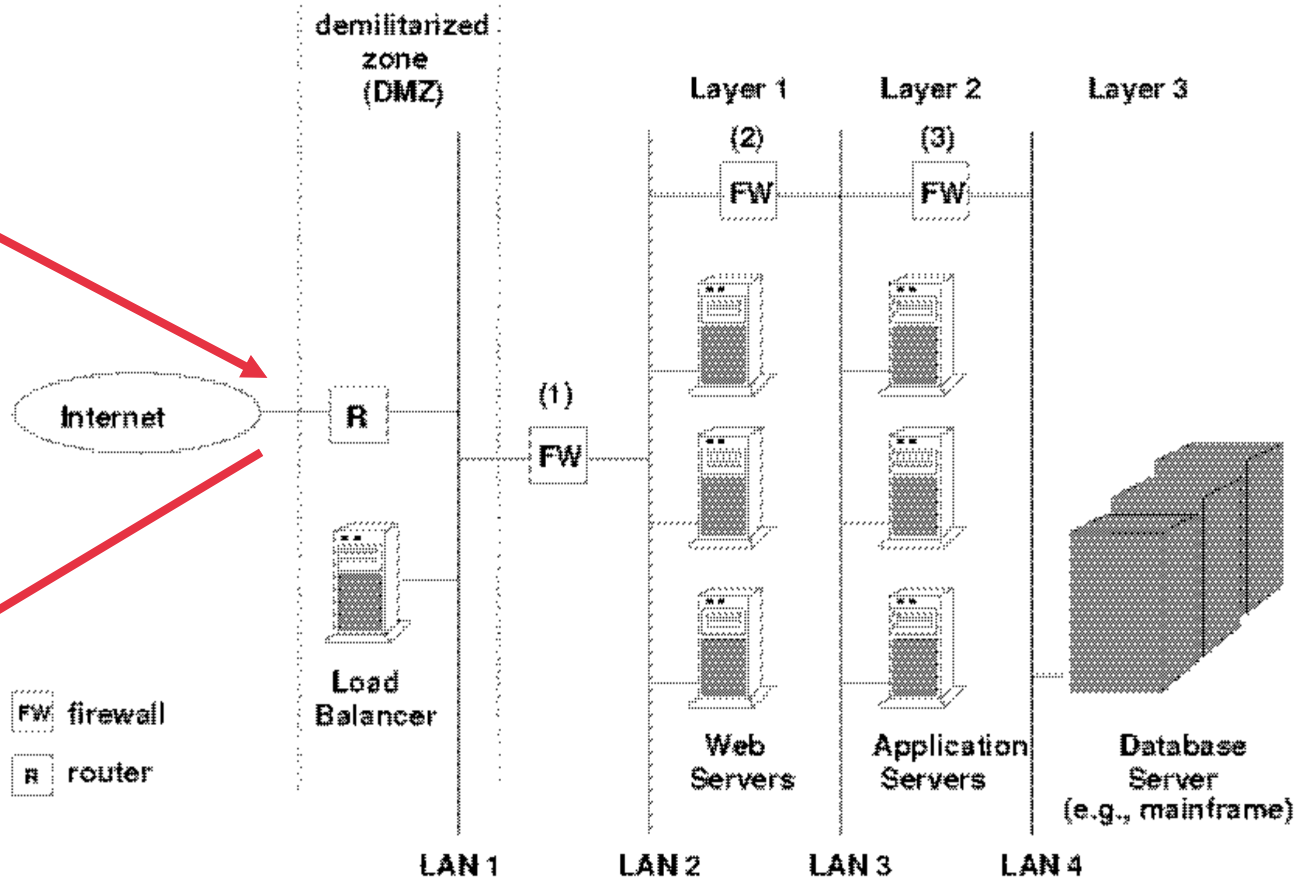
- Importance of performance
  - Can be life and death
    - *Availability* of critical infrastructure e.g. emergency services
  - Customer satisfaction
    - *Availability*
    - *Response time*
- The italicised terms are examples of computer system related performance metrics
  - Also known as Quality of service (QoS) metrics

Response time

- Response time
  - What is it? (Next slide)
  - Possible performance specifications
    - Mean response time is less than 1 s when no more than 5000 requests arrive per second
    - 95% of the requests are completed within 1s when no more than 5000 requests arrive / s
      - Note: Workload characteristics are also part of the performance specification

Response time of a system

Request arrives at time t1



Request completes and leaves at time t2

Response time = t2 - t1.
Measured in seconds. Can be expressed as mean, standard deviation, probability distribution etc.

Availability

- Fraction of time the system is up and useable by users
  - Ex: It is common for Internet Service Providers (ISP) to sign Service Level Agreement (SLA) with their commercial customers. One ISP guarantees that its network outage is less than 6 hours per 30 days. The network availability is 1 - 6/(30*24) = 99.17%

Lecture outline

- Capacity planning
  - Why?
  - What?
- Quality of service metrics
- Quantitative performance analysis ←→ Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Queueing models
  - Queues → Waiting time

Capacity Planning ➔ Performance analysis

- Capacity planning question:
  - A web server needs to complete an HTTP request within 20ms when there are 500 HTTP requests per second, what CPU speed do you need?
- Let us turn the capacity planning question into a performance analysis question
- Performance analysis question:
  - If the web server has a CPU with *x* MIPS, what is the response time when there are 500 HTTP requests per second?
- If you can solve the performance analysis question for any value of *x*, you can also solve the capacity planning question

Exercise:

- As a capacity planner, your task is to choose the CPU speed (in MIPS) of a web server so that the mean response time to a specific workload is no more than 25ms.
- You talk to a performance analyst about your problem. The analyst knows an algorithm that predicts the mean response time for any CPU speed.
- You take the algorithm and plug in a number of different CPU speeds. The results are recorded below.
- Can you solve your capacity planning problem?

CPU Speed (MIPS)	Predicted mean response time (ms)
2000	40
2500	32
3000	26
3500	22
4000	18

Three performance analysis strategies

- Build the system and perform measurement
- Simulation
- Mathematical modelling

- This course will look at
  - Quantitative methods to determine the QoS metrics of computer systems using
    - Queueing networks
    - Markov chains
  - Using simulation to study performance
  - Optimisation methods such as linear and integer programming

Ex. 1: Database server

- A database server has a CPU and 2 disks (Disk1 and Disk2)
- The response time is 10s for each query. How can we improve it?
  - Change the CPU? To what speed?
  - Add a CPU? What speed?
  - Add a new disk? What to move there?
- Technique: Queueing networks

Ex 2: Composite web services

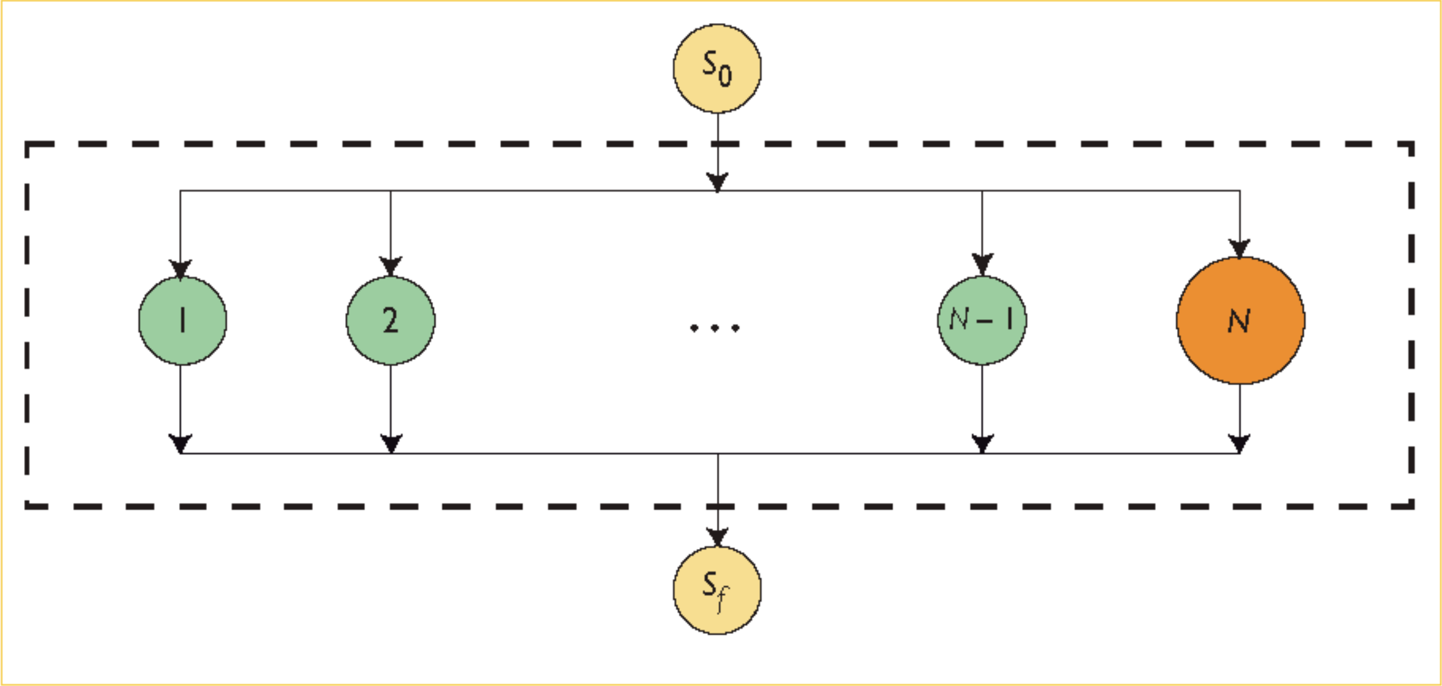


Figure 1. A composite Web service. After an initialization step S_0, N Web services are invoked in parallel. Service N takes longer than the others, and the final step S_f can only be carried out after all N services have completed.

- Aim: Determine response time
- Queueing networks with fork-join

Picture: IEEE Internet Computing Feb 2004

Ex. 3: Server farm power allocation

- A server farm consists of multiple servers
- The servers can run at
  - Higher clock speed with higher power
  - Lower clock speed with lower power
- Ex: Given
  - Higher power = 250W, lower power = 150W
  - Power budget = 3000W
  - You can have
    - 12 servers at highest clock speed
    - 20 servers at lowest clock speed
    - Other combinations
  - Which combination is best?

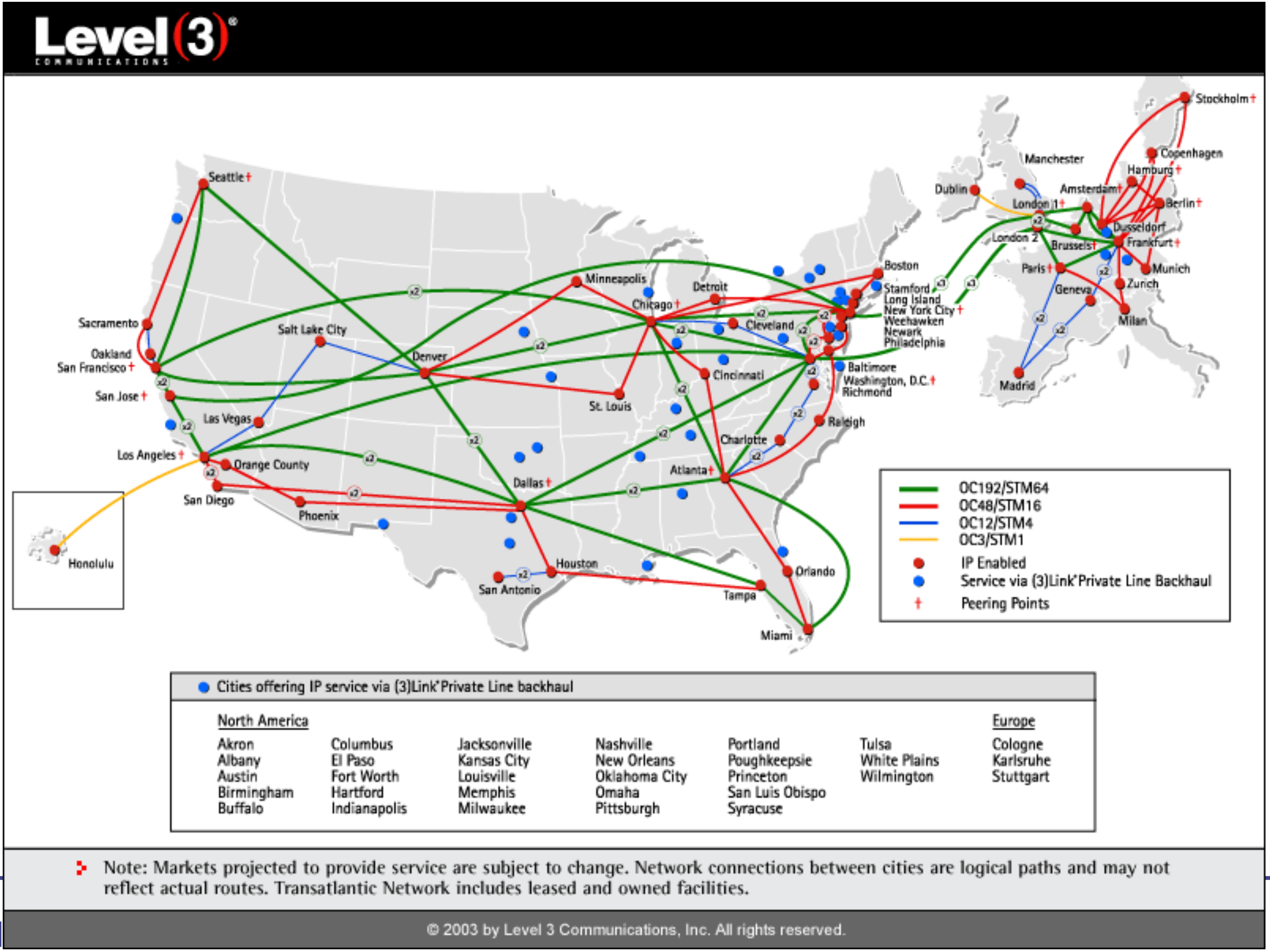
- Queueing theory

Ex 4: Internet data centre availability

- Distributed data centres
- Availability problem:
  - Each data centre may go down
    - Mean time between going down is 90 days
  - Mean repair time is 6 hours
  - Can I maintain 99.9999% availability for 3 out of 4 centres
- Technique: Markov Chain

Ex 5: Network expansion

- You would like to add communication links to a network. The design questions are: Where to add? How much capacity?
- Technique: Integer programming



● Cities offering IP service via (3)Link Private Line backhaul

North America

Akron
Albany
Austin
Birmingham
Buffalo
Columbus
El Paso
Fort Worth
Hartford
Indianapolis
Jacksonville
Kansas City
Louisville
Memphis
Milwaukee
Nashville
New Orleans
Oklahoma City
Omaha
Pittsburgh
Portland
Poughkeepsie
Princeton
San Luis Obispo
Syracuse
Tulsa
White Plains
Wilmington

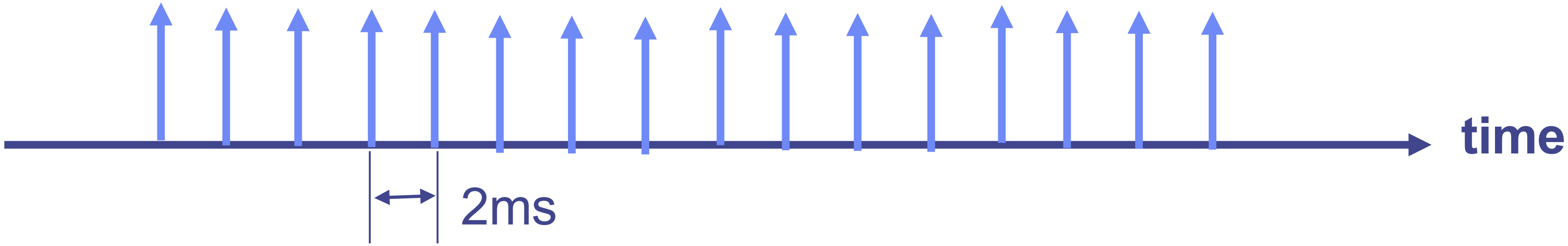
Europe

Cologne
Karlsruhe
Stuttgart

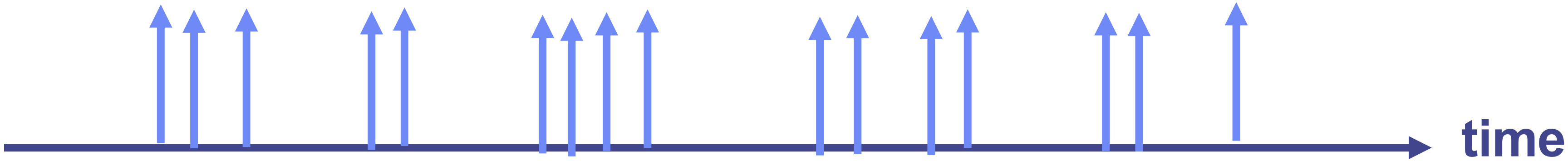
Note: Markets projected to provide service are subject to change. Network connections between cities are logical paths and may not reflect actual routes. Transatlantic Network includes leased and owned facilities.

Why probability?

- The mathematical methods that we are going to study are based on probability theory. Why probability?
- Let us say 500 HTTP requests arrive at the web server in one second
- A deterministic world will mean
  - An HTTP request arrives every 2ms



- But the arrival pattern is not deterministic, it's random



Lecture outline

- Capacity planning
  - Why?
  - What?
- Quality of service metrics
- Quantitative performance analysis ←→ Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Queueing models
  - Queues → Waiting time

QoS metrics

- We have seen 2 QoS metrics
  - Response time
  - Availability
- More QoS metrics
  - Throughput
  - Reliability
  - Scalability

Throughput (1)

- The rate at which requests are completed
- Ex: For network routers, throughput can be measured in
  - Packets per second (pps)
    - Ex: 10 Mpps for 40-byte packets
    - Note: Should specify packet size
  - Mb/s
- Other throughput measures
  - Web site: HTTP requests/s, bytes/s
  - CPU: MIPS, FLOPS

Throughput (2)

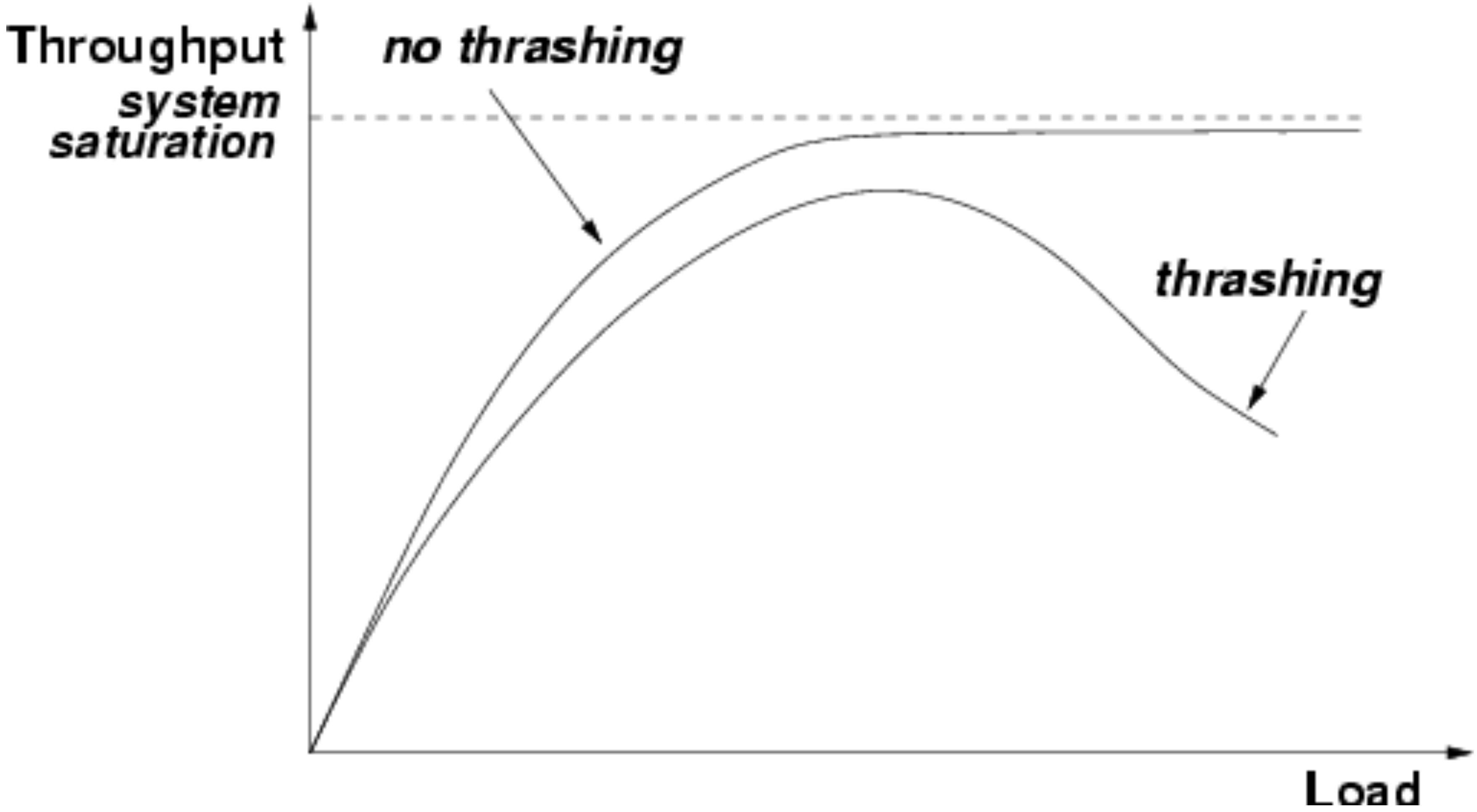
- Throughput is a function of the load
  - A disk takes 0.01s to perform an I/O operation
  - Maximum number of I/O operation per s = 100
  - If 50 I/O operations arrive per second, the throughput = 50 I/O operations/s
  - If 110 I/O operations arrive per second, the throughput = 100 I/O operations

  - Can you find a formula relating throughout, offered load and max capacity?
  - Throughput = min(offered load, max capacity)

Throughput (2*)

- If you find it difficult to do the previous page, you can try this real life analogy.
- Throughput is a function of the load
  - A barister can make a cup of coffee every 30 seconds
  - Maximum number of cups of coffee the barister can make in an hour = 120
  - If 50 customers arrive in an hour and each customer orders a coffee, the barister's throughput = 50 coffees / hour
  - If 150 customers arrive in an hour and each customer orders a coffee, the barister's throughput = 120 coffees / hour

Throughput (3)



Thrasing = congestion collapse

Throughput (4)

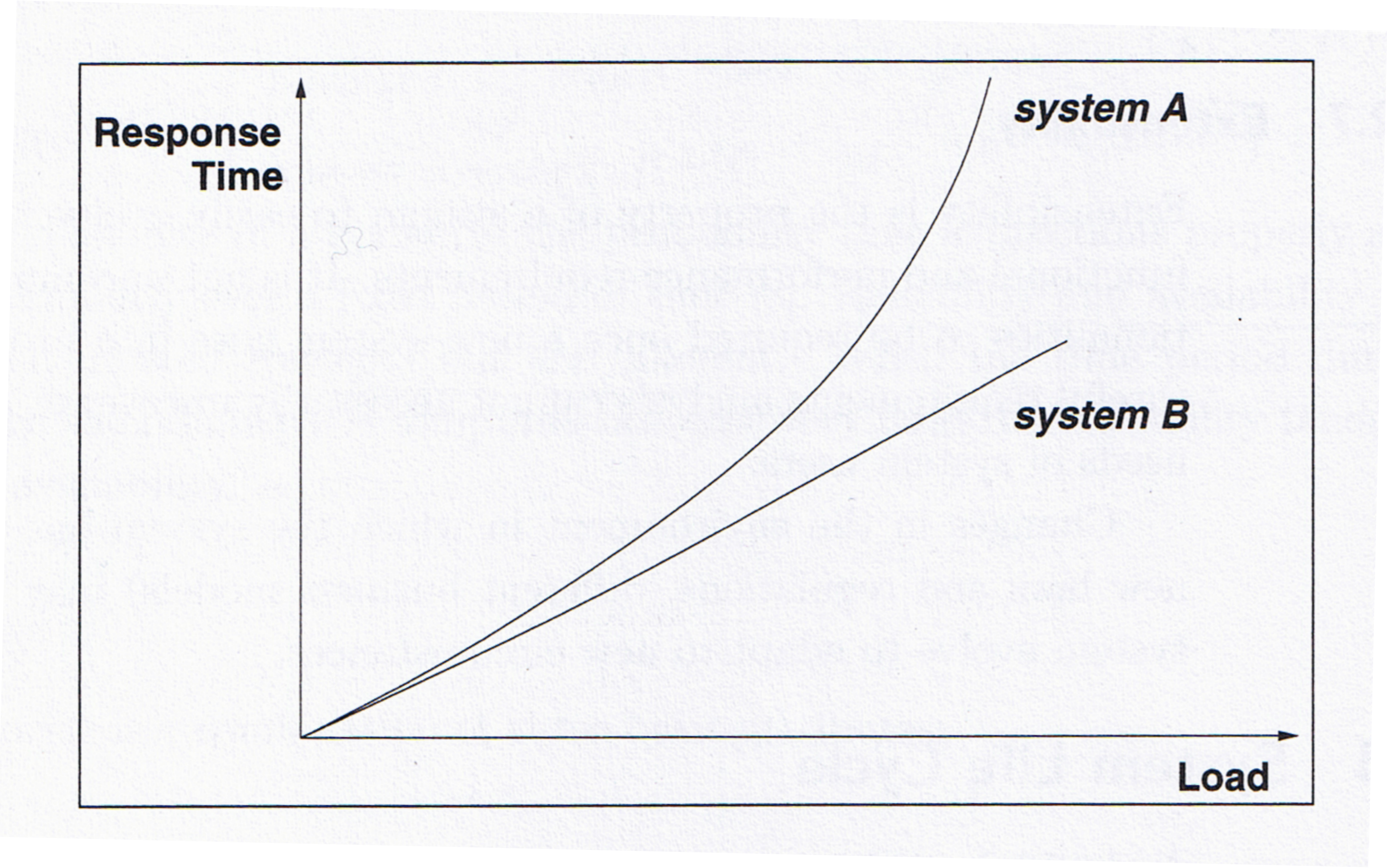
- Performance evaluation can be used to determine the maximum throughput of computer systems
  - Example: bottleneck analysis
    - Topic for next week

Reliability

- The probability that a system will function
- Possible metrics are
  - Mean-time-to-failure (MTTF)
    - The mean time between two system failures
  - Probability of system failure at any time
- Related metric
  - Mean-time-to-repair (MTTR)

Scalability

- How fast does performance degrade with increasing load or users?



Which system is more scalable?

Lecture outline

- Capacity planning
    - Why?
    - What?
- Quality of service metrics
- Quantitative performance analysis ←→ Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Queueing models
    - Queues → Waiting time

Quantitative performance analysis (3)

- Sample performance analysis question:
  - If the web server has a CPU with *x* MIPS, what is the response time when there are 500 HTTP requests per second?

- Performance analysis question:
  - Given:
    - A computer system with a certain capacity
    - The workload
  - Find
    - The performance (response time, throughput etc) of the system

- Our method is:
  - Build analytical models of computer systems

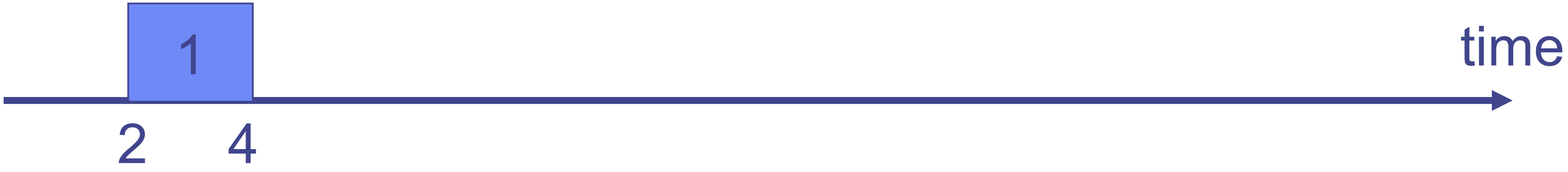
- An important part of the analytical model is “queue”
  - You can surely relate “queues” to “waiting time”

Single server FIFO queue

- Queueing Theory terminologies
  - Server: Processing unit
  - FIFO: First-in first-out
  - Work conserving server
    - The server cannot be idle when there are jobs waiting to be processed in the queue
- Ex: Shop with only one checkout counter
- The server is a resource
  - Queues result from resource contention
- Main concern: response time

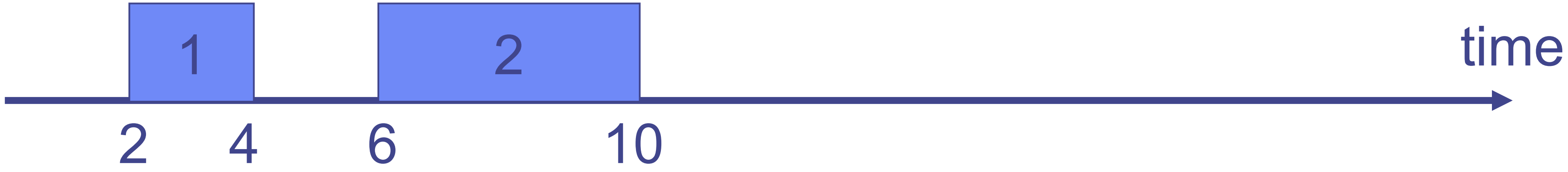
Job index	Arrival time	Processing time required
1	2	2
2	6	4
3	8	4
4	9	3

Assumption: server is idle when job #1 arrives



Job #1 is admitted into the server immediately since the server is idle.
Job #1 is completed and leaves the system at time 4.

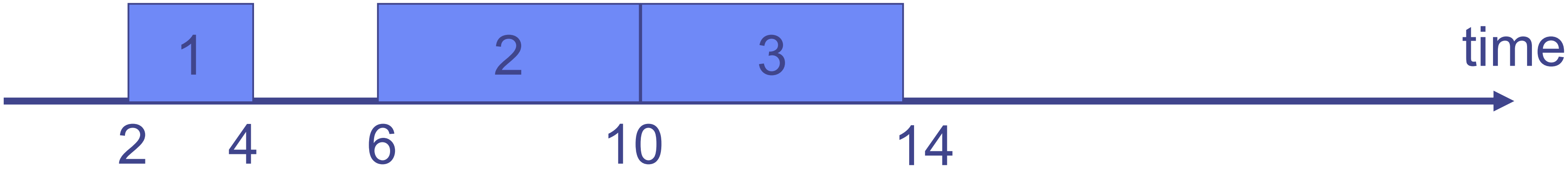
Job index	Arrival time	Processing time required
1	2	2
2	6	4
3	8	4
4	9	3



Job #2 arrives when the server is idle. It gets admitted immediately.

Job #2 will be completed at time 10.

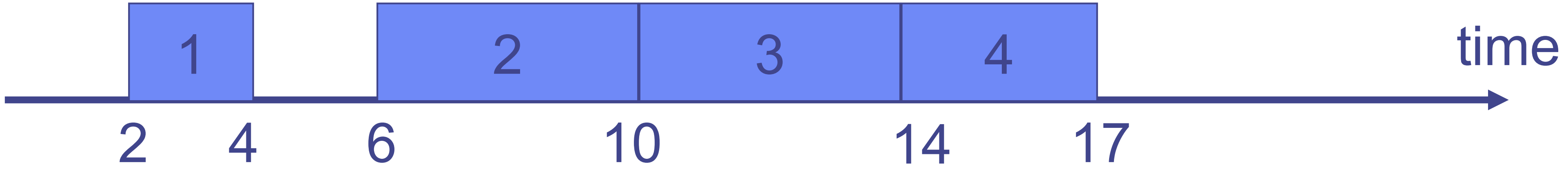
Job index	Arrival time	Processing time required
1	2	2
2	6	4
3	8	4
4	9	3



Job #3 arrives when Job #2 is being served i.e. the server is busy. Job #3 has to wait in the queue.

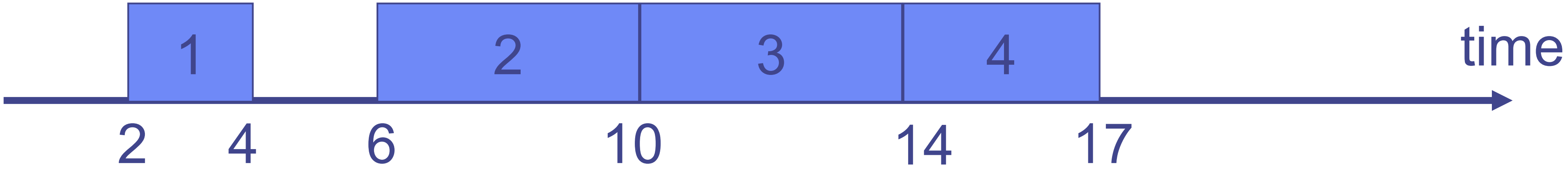
Server starts processing Job #3 immediately after finishing Job #2.

Job index	Arrival time	Processing time required
1	2	2
2	6	4
3	8	4
4	9	3



Job #4 arrives when the server is processing Job#2 and Job#3 is in the queue. Job #4 joins the queue. It gets served at time 14, immediately after Job#3 is completed.

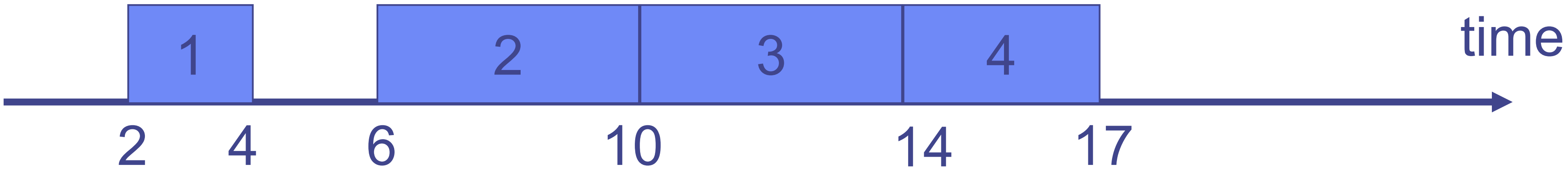
Job index	Arrival time	Processing time required
1	2	2
2	6	4
3	8	4
4	9	3



- Definition: Response time = Departure time - arrival time

  Ex: Response time for Job#4 = 8
- Response time = Waiting time + Processing time

Job index	Arrival time	Processing time required
1	2	2
2	6	4
3	8	4
4	9	3



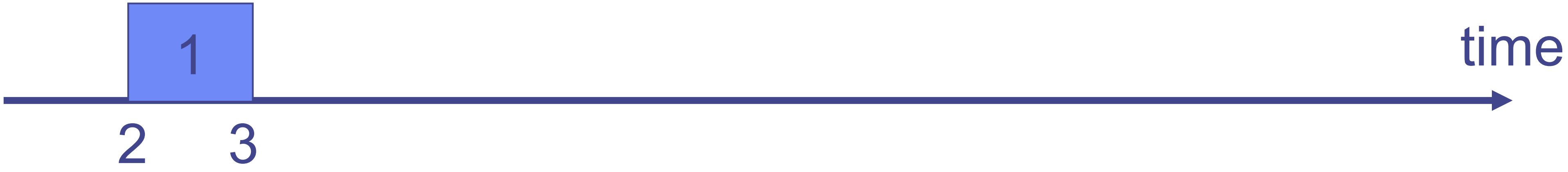
- Definition: Utilisation = Percentage of time over which the server is busy
- What is the utilisation of the server over the first 12s?
    - 8/12 = 66.7%

Single server FIFO queues

- Can be used to model
  - Shop with only one checkout counter
  - A single processor processing jobs in FIFO order
  - A disk processing job in FIFO order
- Model
  - An abstraction of the real system
  - Need to capture enough details to meet our analysis requirements

What if both inter-arrival time and processing time are determinisitic?

Job index	Arrival time	Processing time required
1	2	1
2	4	1
3	6	1
4	8	1



What is the waiting time for each job?
What is the response time for each job?

Determining response time

- Generally we need to know
  - The arrival pattern
    - Ex: The arrival rate
    - Ex: The inter-arrival time statistical distribution
  - The service time distribution
    - The time required to process the job
- Since we are interested in response time, our models capture the time related aspects of the real systems e.g. queueing, processing units
- We will learn different methods to determine response time in this course

Service time

- Time require to process a request at a resource
  - Ex: The service time to send a 1000 byte packet over a 10 kbps link is 0.8s. In this case,
    - Service time = packet size / transmission rate
  - Ex: The service time for to get a X byte large file from a disk is
    - Seek time + X / transfer rate
  - For a class of resources, we have
    - Service time = Overhead + Job size / Processing rate

Response time of M/M/1 queue (1)

- M/M/1 queue
  - A type of single server queue characterised by
    - Average arrival rate of jobs is λ
    - Average service demand per job is $1/\mu$
      - μ is the processing rate
    - Inter-arrival time and service demand are drawn from exponential distribution
  - Queueing theory shows that the mean response time for M/M/1 queue is $1 / (\mu - \lambda)$ if $\mu > \lambda$

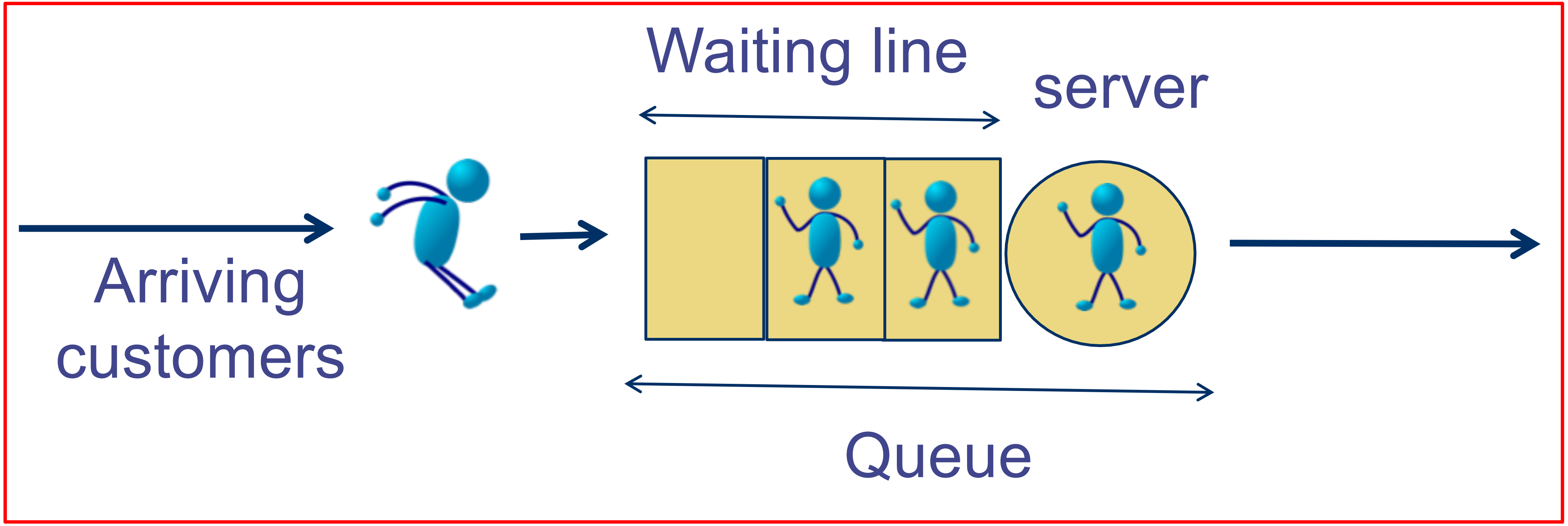
Response time of M/M/1 queue (2)

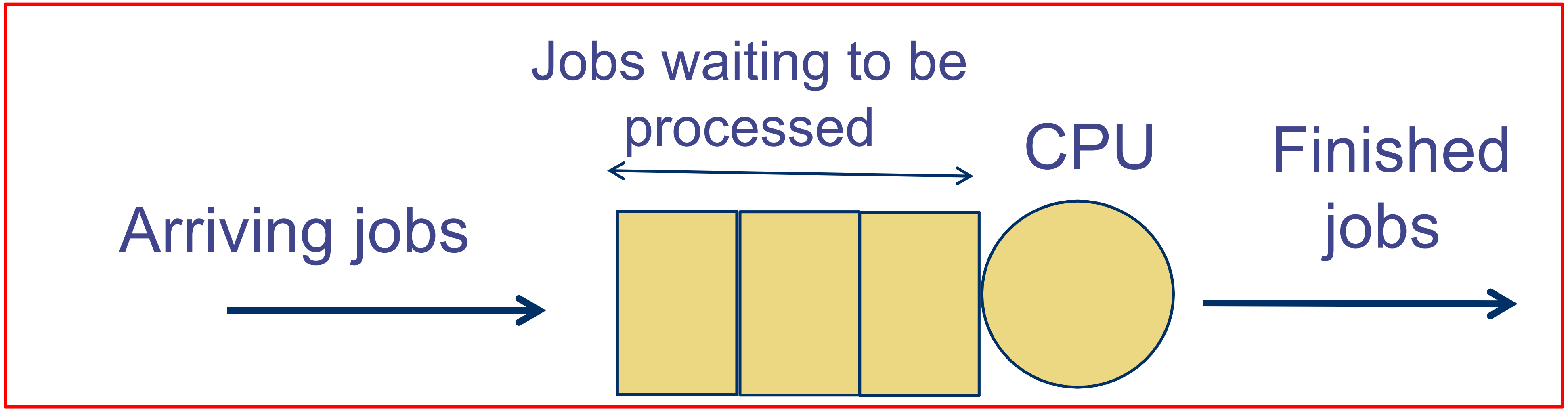
- Example:
  - Current system:
    - Mean arrival rate λ is 2 requests/s
    - Mean service time $1/\mu = 0.2$s => $\mu = 5$
    - The response time = 1 / (5 - 2) = 0.33s
  - What if arrival rate λ is doubled?
    - The new response time = 1 / (5 - 4) = 1s
      - Nonlinear increase!
  - If the new response time is too big, what are your options assuming you still want the new customers?

Modelling computer systems

- Single server queue considers only a component within a computer system
- A request may require multiple resources
  - E.g. CPU, disk, network transmission
- We model a computer systems with multiple resources by a Queueing Networks (QNs)

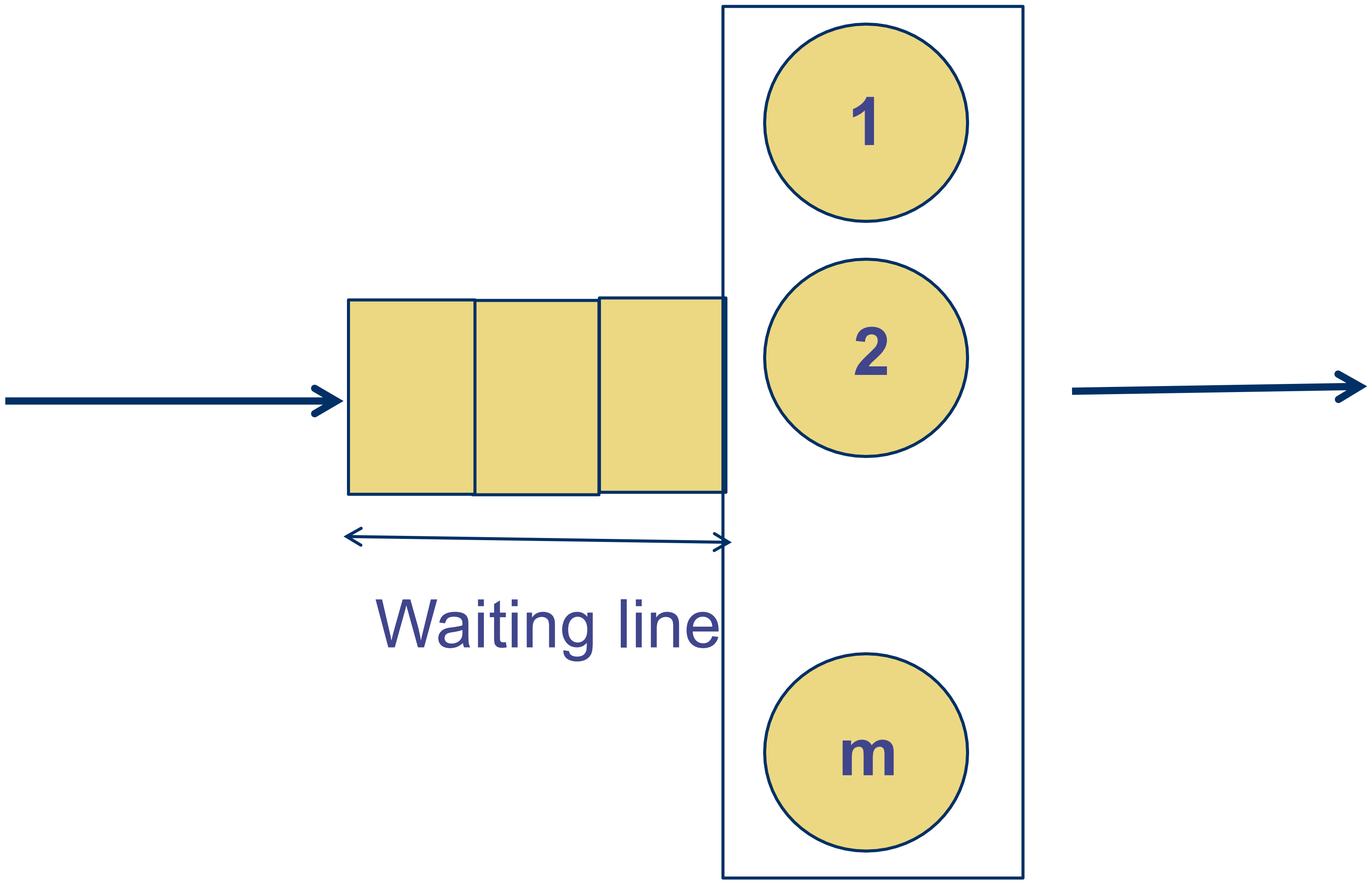
Pictorial representation of single server queues





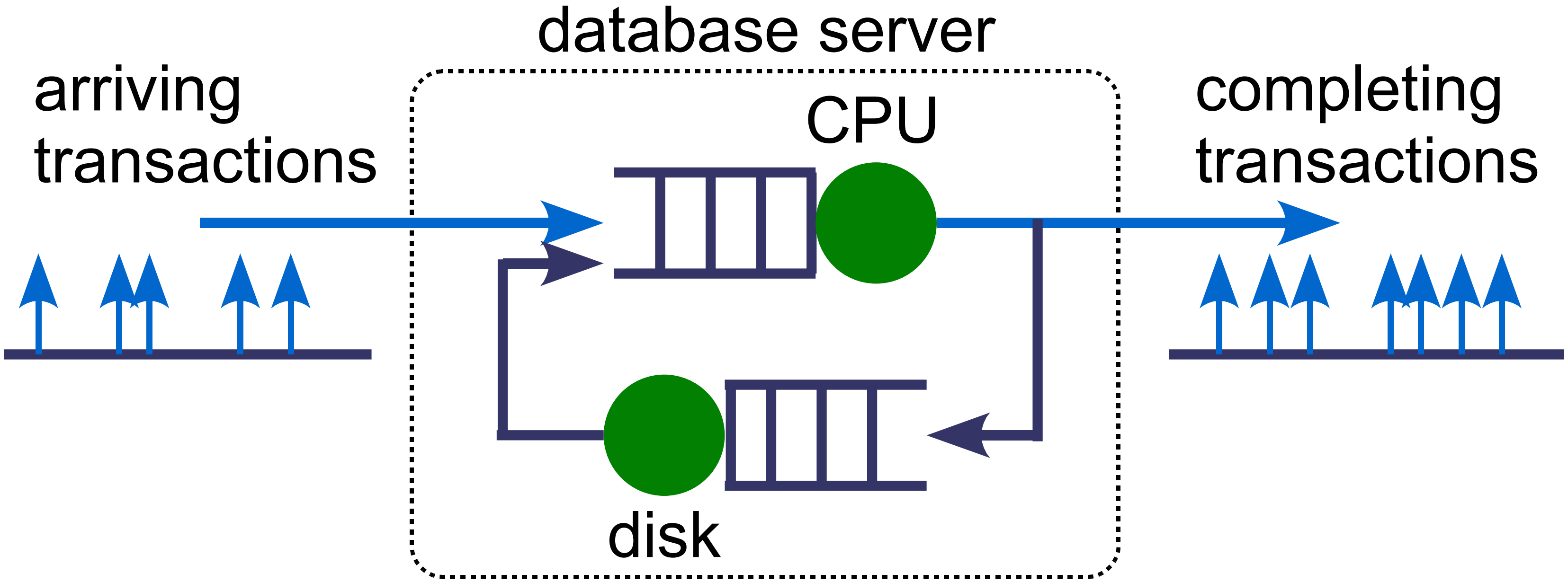
Pictorial representation of queues

Systems with *m* servers



A simple database server

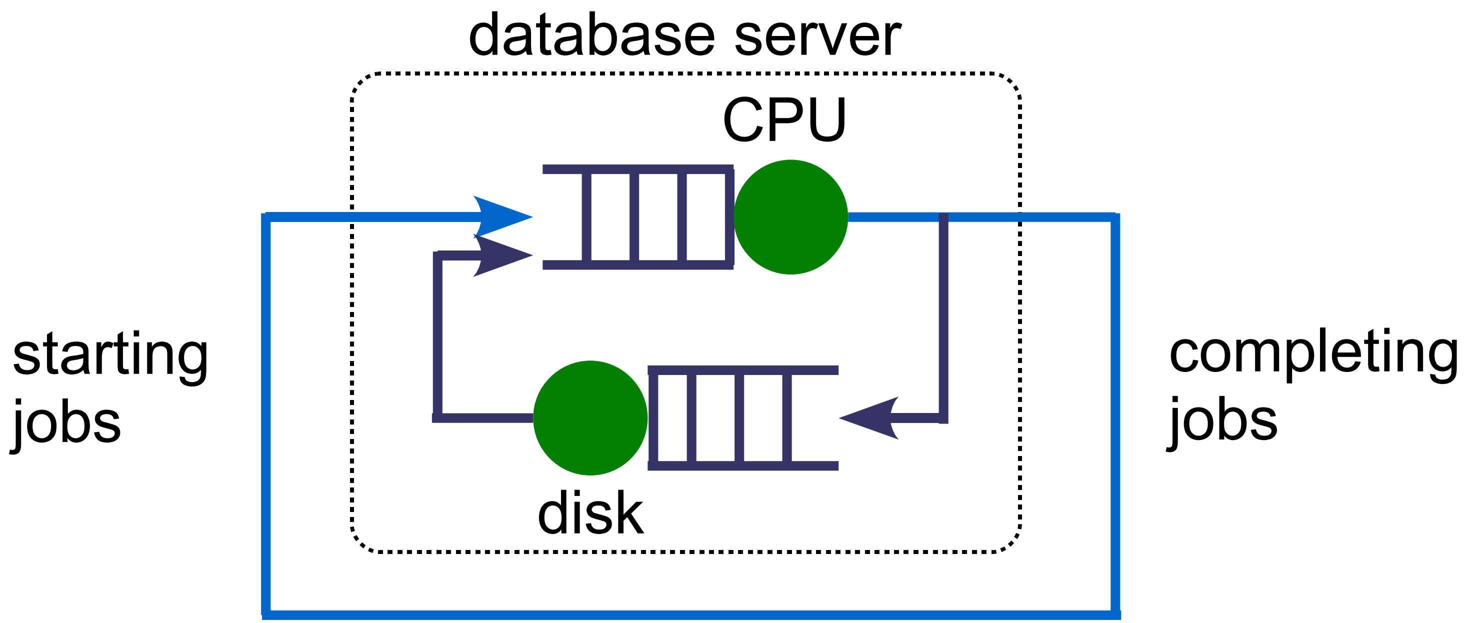
The server has a CPU and a disk.



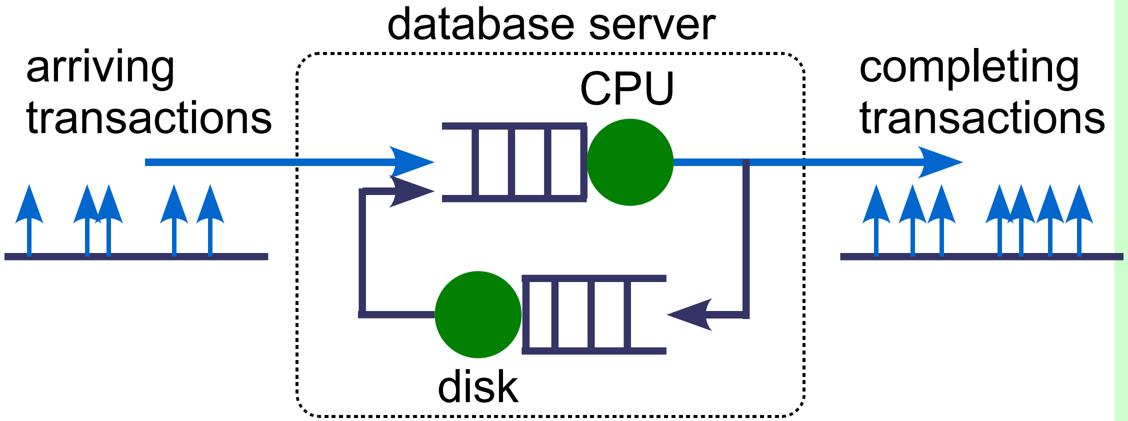
A transaction may visit the CPU and disk multiple times.

DB servers for batch jobs

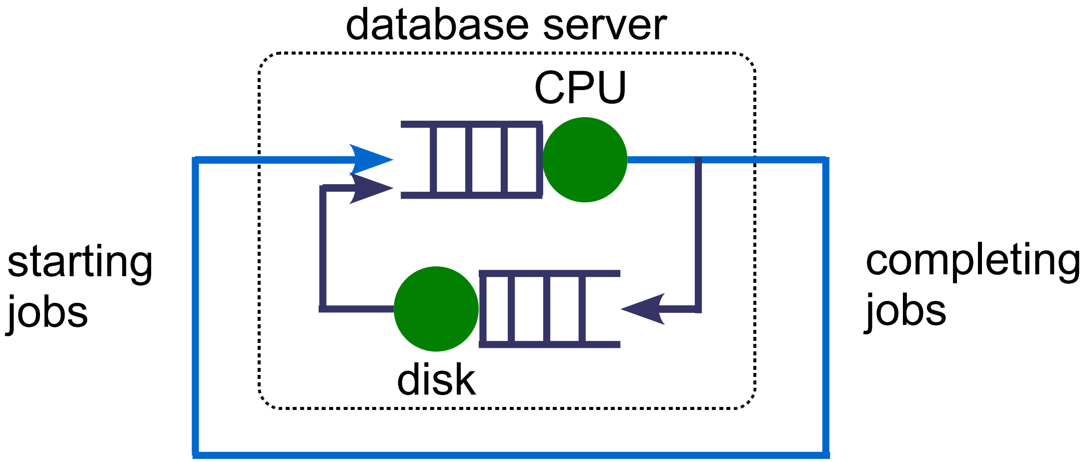
- Example: Batch processing system
  - For summarising transactions only
  - No on-line transactions



Open vs. closed queueing networks (1)

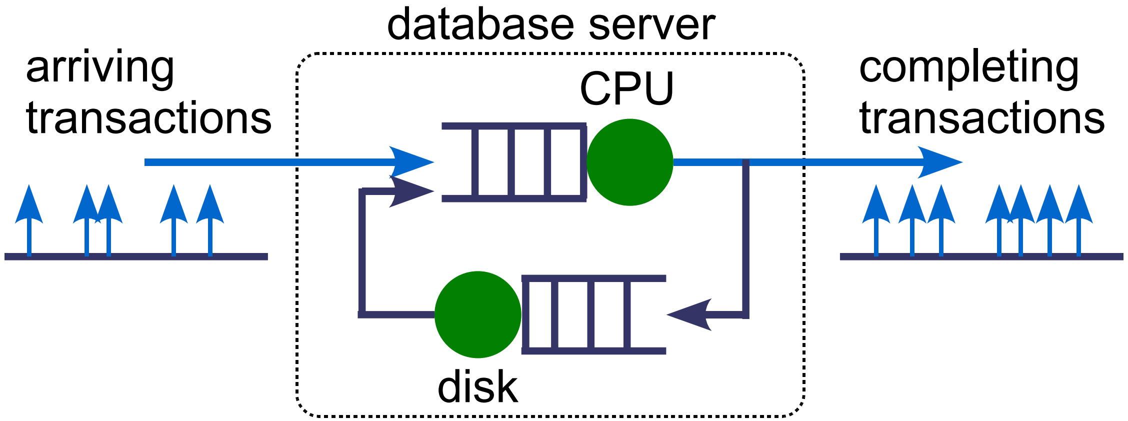


Open queueing network
- External arrivals
- Workload intensity specified by arrival rate



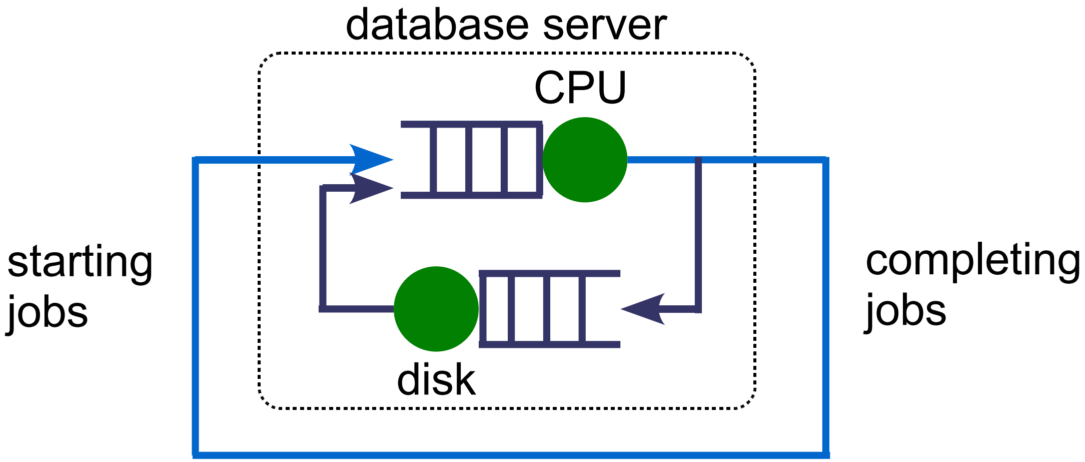
Closed queueing network
- No external arrivals
- Workload intensity specified by customer population

Open vs. closed queueing networks (2)



Open queueing network

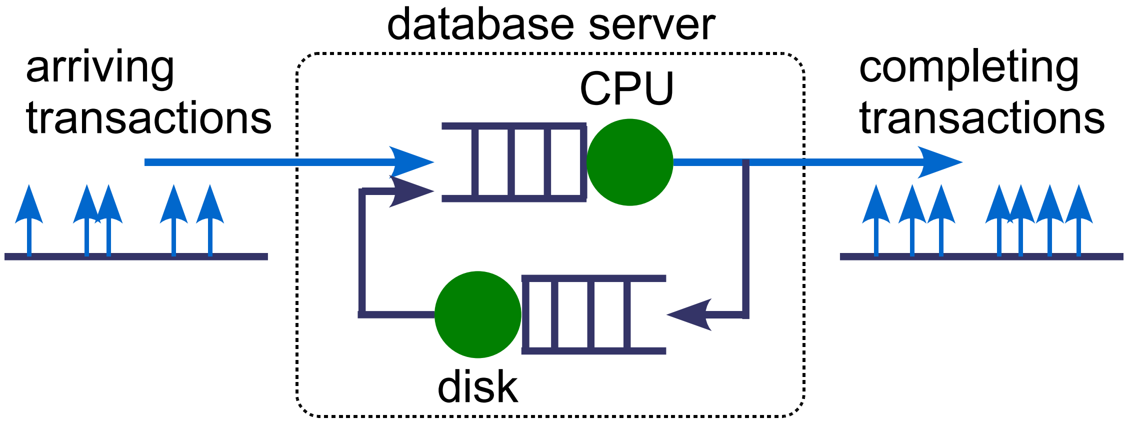
- Unbouned #customers
- For stable equilibrium Throughput = arrival rate



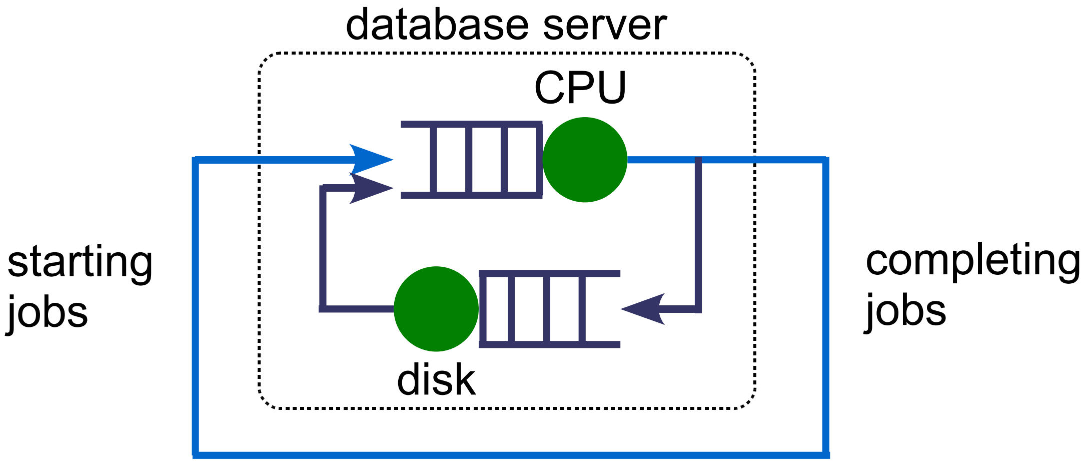
Closed queueing network

- Known #customers
- Throughput depends on # customers etc.

Open vs. closed queueing networks - Terminology



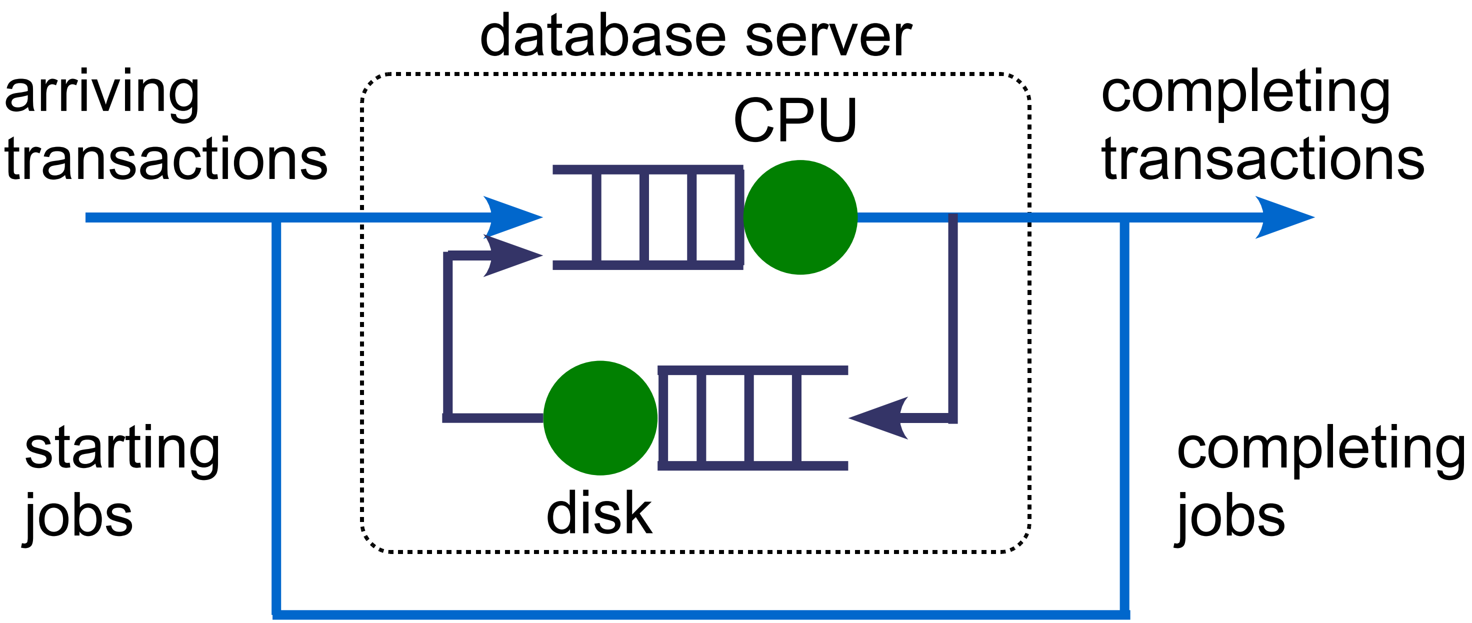
Work in an open queueing network is called transaction



Work in a closed Queueing network is called jobs

DB server - mixed model

- The server has both
  - External transactions
  - Batch jobs



Different techniques are needed to analyse open and closed queueing networks

DB server – Multi-programming level

- Some database server management systems (DBMS) set an upper limit on the number of active transactions within the system
- This upper limit is called multi-programming level (MPL)

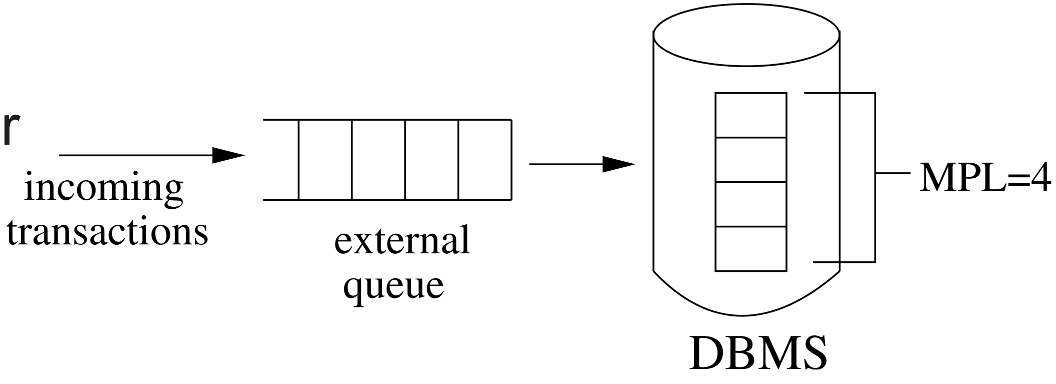


Figure 1. *Simplified view of the mechanism used in external scheduling. A fixed limited number of transactions (MPL=4) are allowed into the DBMS simultaneously. The remaining transactions are held back in an external queue. Response time is the time from when a transaction arrives until it completes, including time spent queueing externally to the DBMS.*

- A help page from SAP explaining MPL
- http://dcx.sap.com/1200/en/dbadmin_en12/running-s-3713576.html
- Picture from Schroder et al. “How to determine a good multi-programming level for external scheduling”

DB Server - Interactive systems

- Modelling client interaction
  - A client sends a job to the server
  - Upon receiving results from the server, the client goes into thinking mode and send a next job
- Model the client as a delay source with no waiting line.

Capacity planning in action

- Modelling
  - Computer Systems ---> Queueing Networks
- You will learn different techniques to analyse a number of different classes of queueing networks:
  - Open/closed single/multiple class
  - Operational Analysis & Bottleneck Analysis
    - The last two will be the topics for next week
- The QN model will allow you to do what-if analysis?
  - What if the arrival rate increases by 20%
  - The increase in arrival rate has increased response time by 10%. What if I change the disk to one that is 20% faster, will I have restored the original performance?

References

- Reading:
  - Menasce et al, Chapters 1 & 2
  - OR
  - Harcol-Balter. Chapters 1 & 2.
- Exercises:
  - Revision problems:
    - See course web site
  - You are expected to try these exercises. Solutions will be available on the web.