

COMP9334

# Capacity Planning of Computer Systems and Networks

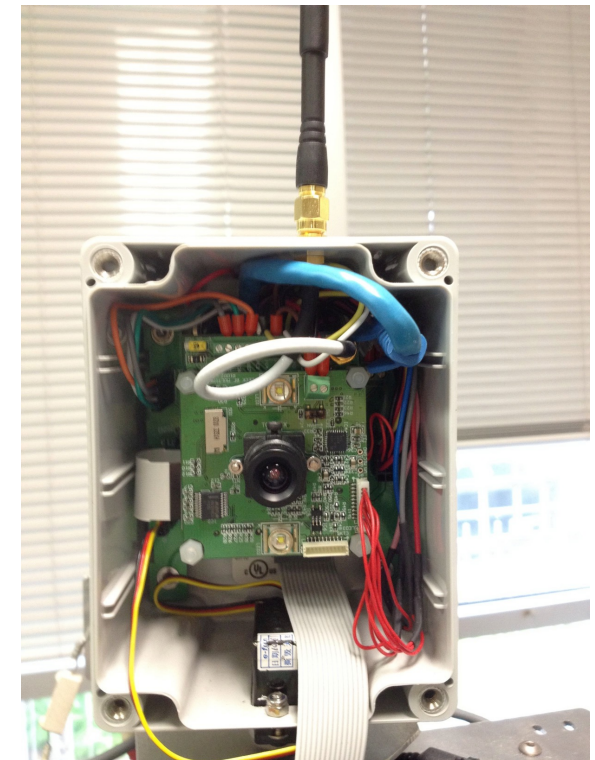
---

Week 1A: Introduction to Capacity  
Planning

Chun Tung Chou

# About your lecturer

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



<https://research.unsw.edu.au/people/associate-professor-chun-tung-chou>

# Course organisation

---

- Course web site: [www.cse.unsw.edu.au/~cs9334](http://www.cse.unsw.edu.au/~cs9334)
- Email: [cs9334@cse.unsw.edu.au](mailto:cs9334@cse.unsw.edu.au)
- Read the course outline
- Lectures: 12-2 on Tue & 10-12 on Fri
  - Simultaneous in-person (Colombo B/O'Shane104) and online (Zoom)
  - You can ask questions live or via chat
  - I will ask questions. You can answer them live or via chat.
  - Recordings will be made available via Moodle
- Consultation:
  - In-person consultation: Thur 12:45-1:15pm in 608/K17
  - Online on Zoom (different link from lectures): Wed 3-3:30pm
  - Time may change. Please check “Timetable” in the course website to confirm.

## 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 course forum to ask questions
  - Don't think your question is silly, other may have the same problem
  - We will use the Ed platform. The forum link at the course website has instructions on how you can enrol yourselves in the forum.
- 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 (20%)
  - Project (30%)
  - Final exam (open book) (50%)
- Assignment: Extended tutorial questions
- Project: Simulation (coding + statistics)
- Overall mark:
  - Overall mark = Weighted mean of all assessment components
  - Pass if overall mark  $\geq 50$

# Exam

---

- Information will be provided as soon as it is available
- Likely to be an in-person invigilated exam in the exam period



# 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 thought that gave me a mathematical edge."

# Lecture outline

---

- Capacity planning
  - Why?
  - What?
- Quality of service metrics
- Quantitative performance analysis  $\leftrightarrow$  Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Single server queues

# Why capacity planning?

---

## **Hot eBusiness News**

---

# **Poor Web Site Performance Is Costing Retailers Millions**

# Why capacity planning?

---

## Hot eBusiness News

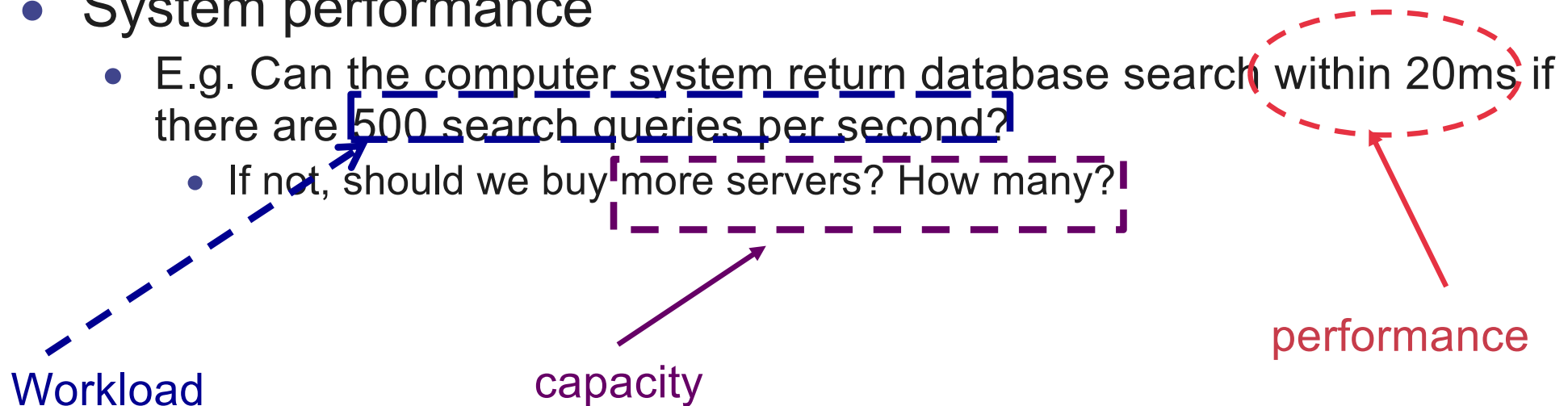
---

### 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
  - 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?
- performance
- capacity

# 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. Can you 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

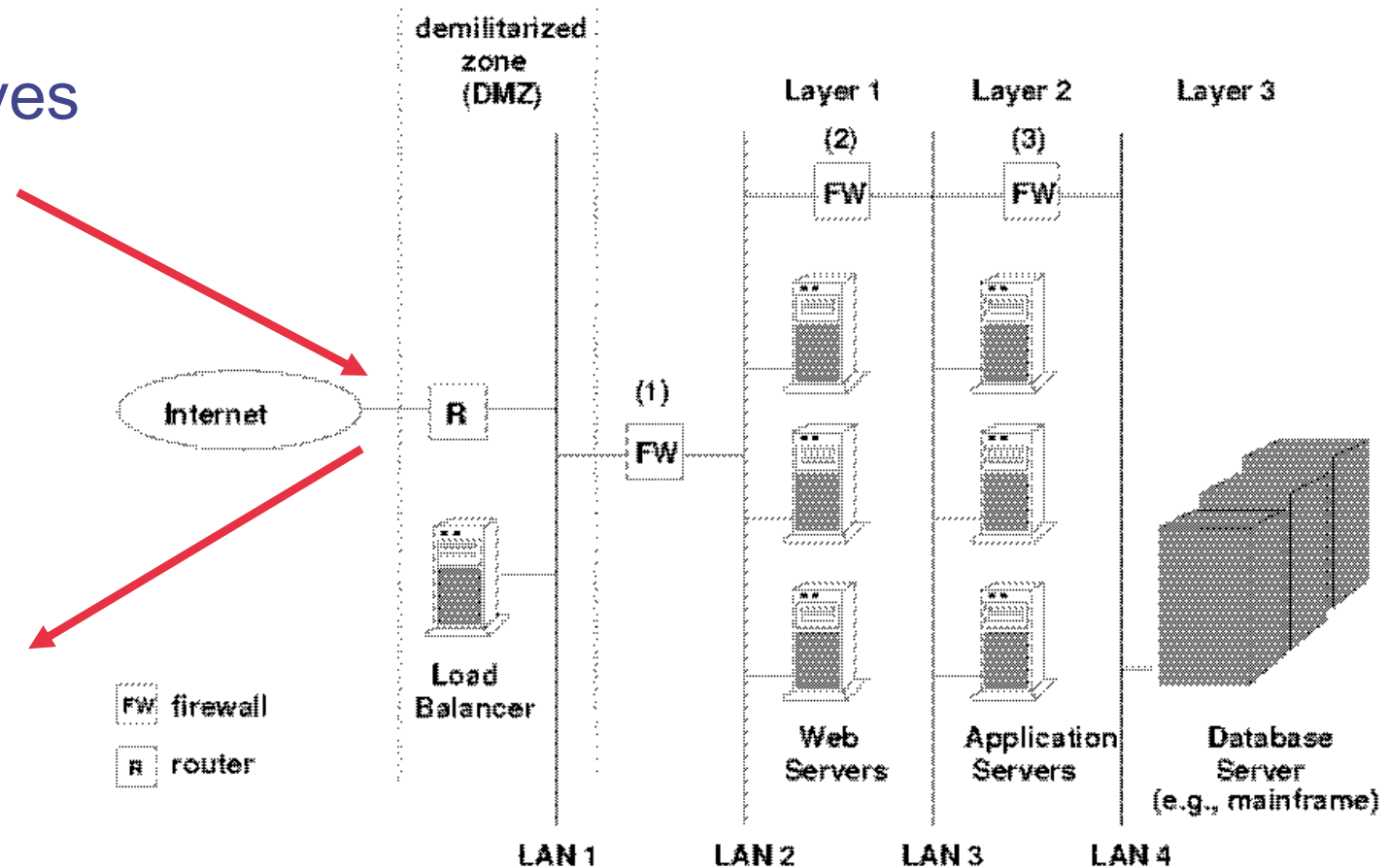
---

- 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  $t_1$

Request  
completes  
and leaves  
at time  $t_2$



Response time =  $t_2 - t_1$ .

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  $\leftrightarrow$  Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Single-server queues

# 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: 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

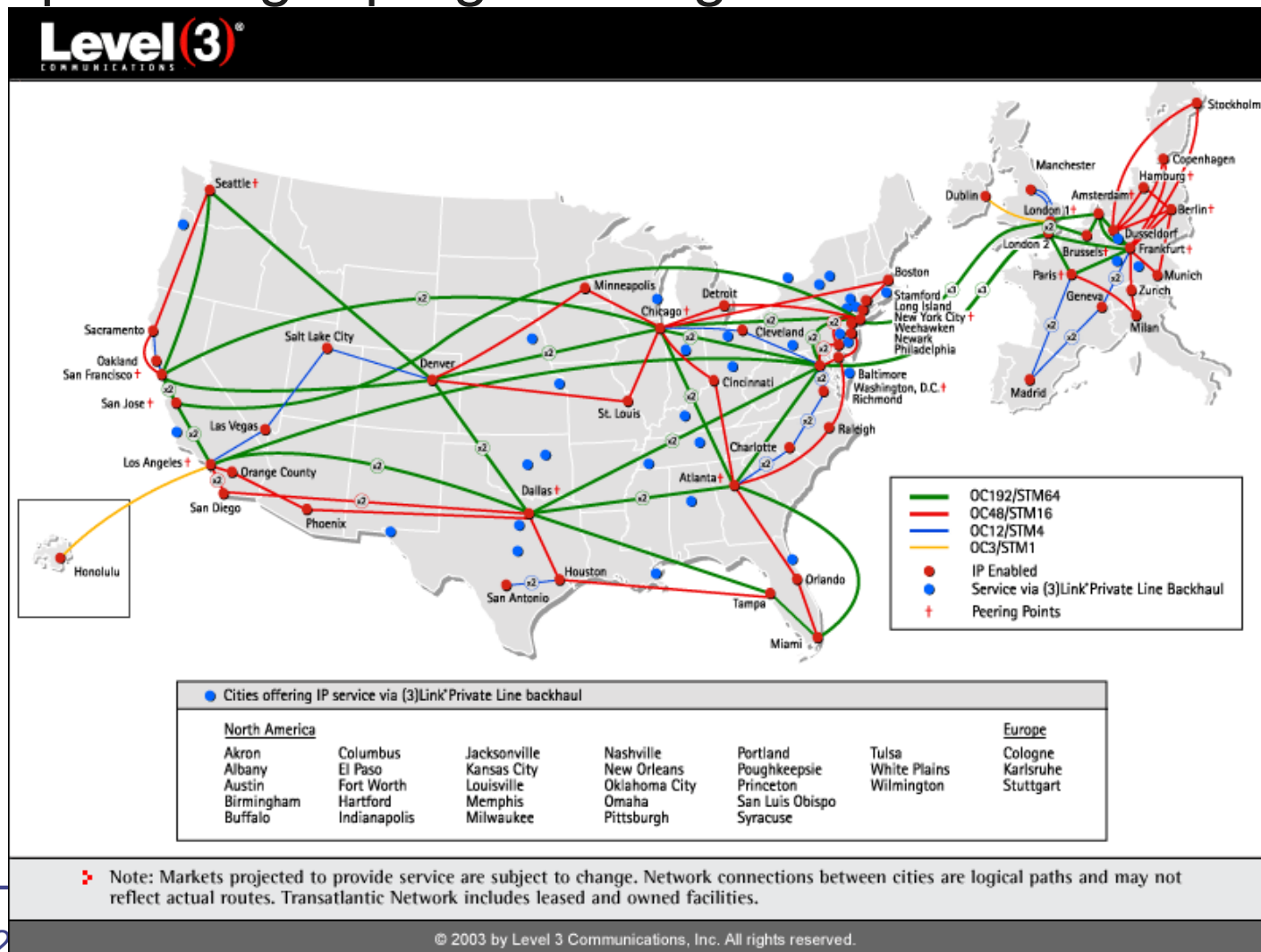
## Ex 2: Internet data centre availability

---

- 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

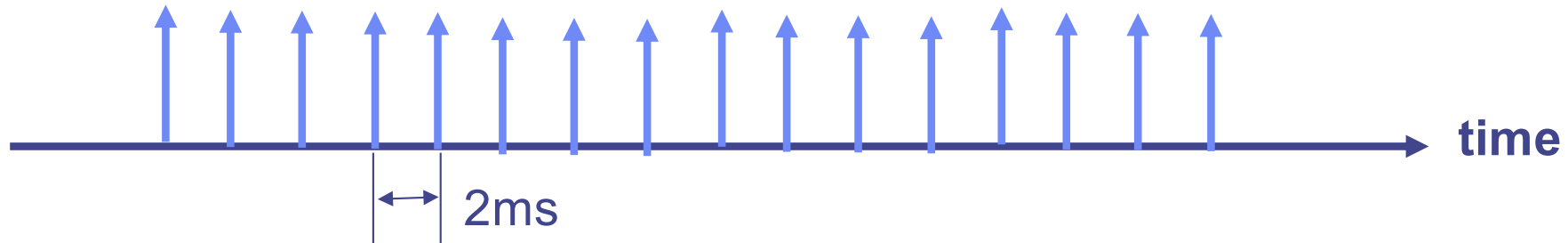
## Ex 3: Network expansion

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

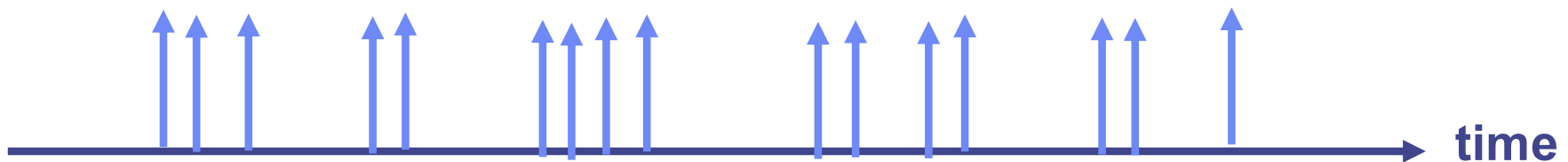


# Why probability?

- 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  $\leftrightarrow$  Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Single-server queues

# QoS metrics

---

- We have seen 2 QoS metrics
  - Response time
  - Availability
- More QoS metrics
  - Throughput
  - Reliability (Will discuss later in the course)
  - Scalability (Not discussed)

# 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 second =
  - If 50 I/O operations arrive per second, the throughput =  I/O operations/s
  - If 110 I/O operations arrive per second, the throughput =  I/O operations
- Can you find a formula relating throughput, offered load and max capacity?
-

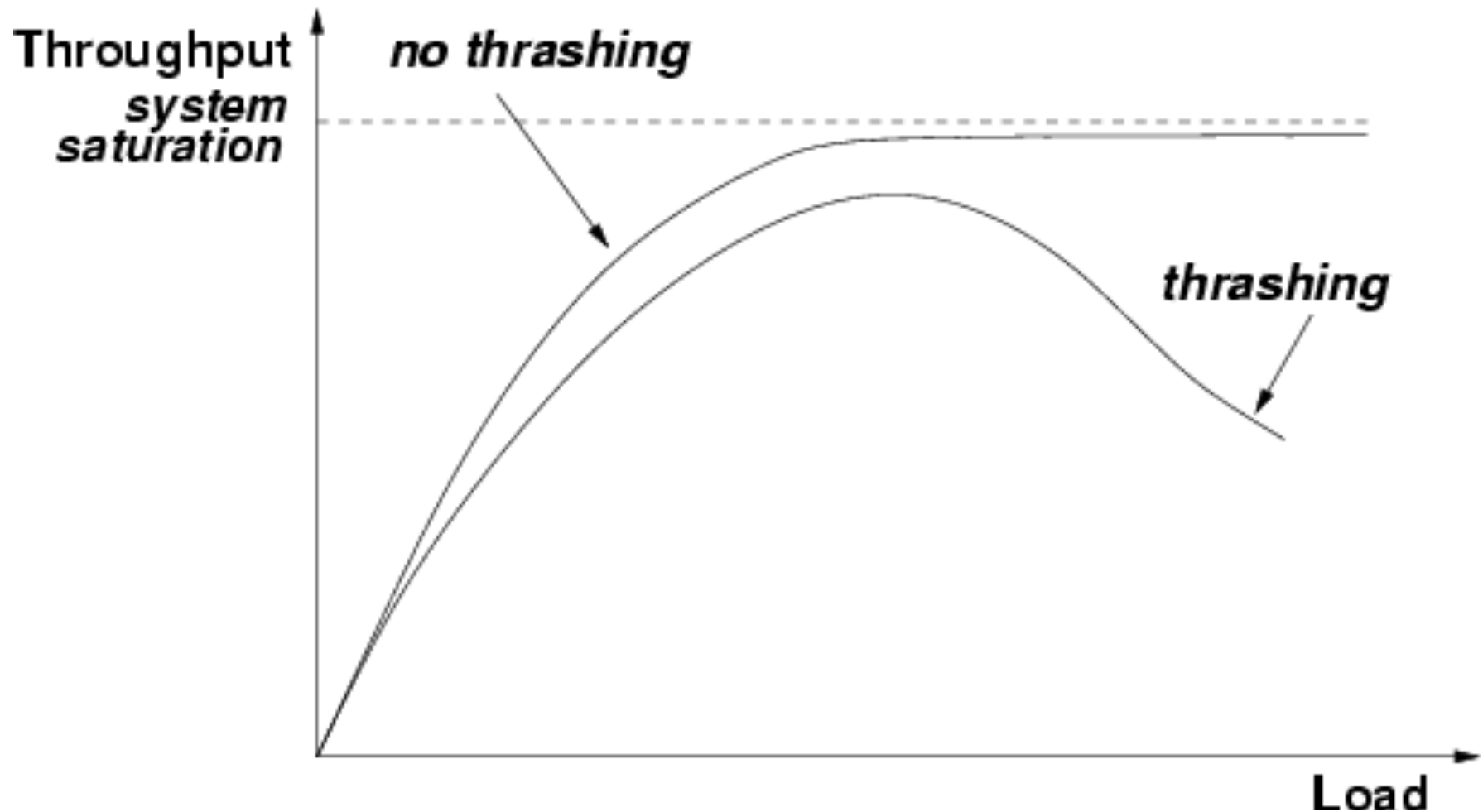


## Throughput (2\*)

---

- 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 barista can make a cup of coffee every 30 seconds
  - Maximum number of cups of coffee the barista can make in an hour =
  - If 50 customers arrive in an hour and each customer orders a coffee, the barista's throughput =  coffees / hour
  - If 150 customers arrive in an hour and each customer orders a coffee, the barista's throughput =  coffees / hour

## Throughput (3)



Thrashing = congestion collapse

## Throughput (4)

---

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

# Lecture outline

---

- Capacity planning
  - Why?
  - What?
- Quality of service metrics
- Quantitative performance analysis  $\leftrightarrow$  Capacity Planning
- What techniques you will learn
- More quality of service metrics
- Single-server queues

# 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.

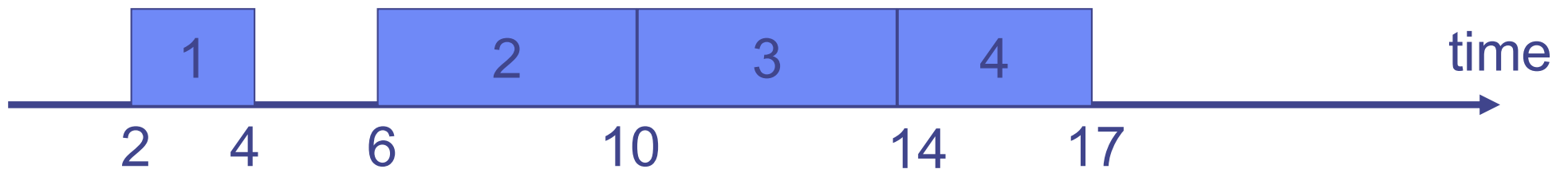


Job index	Arrival time	Processing time required
1	2	2
2	6	4
3	8	4
4	9	3



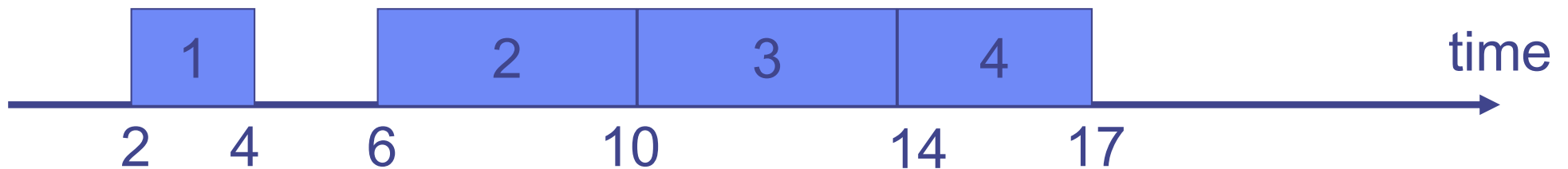
Job #3 arrives when Job #2 is being served i.e. the server is busy. Job #3 has to wait in the queue.  
 Server starts processing Job #3 immediately after finishing Job #2.

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 =  $17 - 9 = 8$  (= **5** + 3)
- 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 deterministic?

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 probability distribution
  - The service time probability 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,
    - $\text{Service time} = \text{packet size} / \text{transmission rate}$
  - Ex: The service time to fetch a X byte large file from a disk is
    - $\text{Seek time} + X / \text{transfer rate}$
  - For a class of resources, we have
    - $\text{Service time} = \text{Overhead} + \text{Job size} / \text{Processing rate}$



# Summary

---

- What capacity planning is
- Very important: A capacity planning problem can be solved by solving a series of performance analysis problems
- Performance metrics
  - Response time, waiting time, throughput
- Modelling of single server queues

# 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.