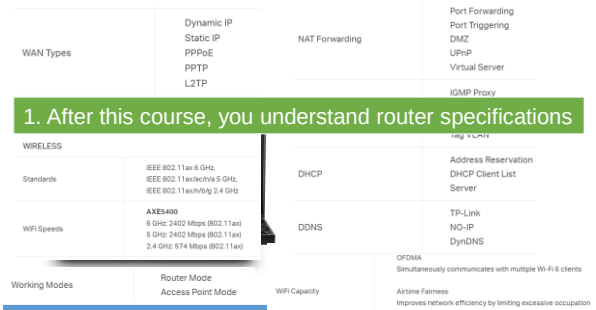
Computer Networks X_400487

Lecture 1: Introduction to Computer Networks

Welcome! Lecture starts at 15:30

Lecturer: Jesse Donkervliet

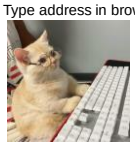
VU Vrije Universiteit Amsterdam



1. After this course, you understand router specifications

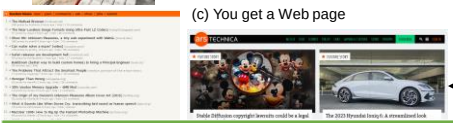
https://www.tp-link.com/en/home-networking/wifi-router/archer-axe75/#overview

(a) Type address in browser



(b) Browser does magic

(c) You get a Web page



2. After this course, you understand browser magic

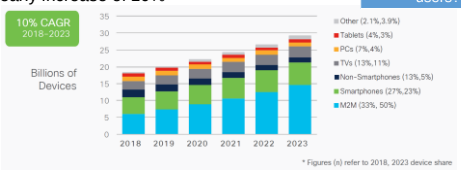
3. After this course, you understand how networks enable new applications

Image source: Nvida, NASA, https://www.pbs.org/newshour/science/katie-bouman-hardly-knew-what-a-black-hole-was-her-algorithm-helped-us-see-one

Number of devices connected to the internet

- Likely exceeds 20 billion connected devices
- Yearly increase of 10%

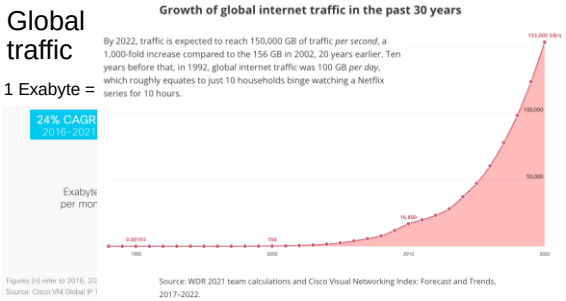
Q: How much traffic is generated by these users?

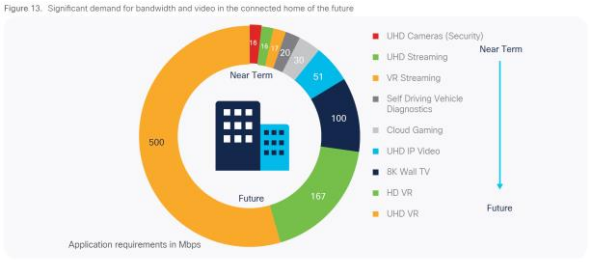


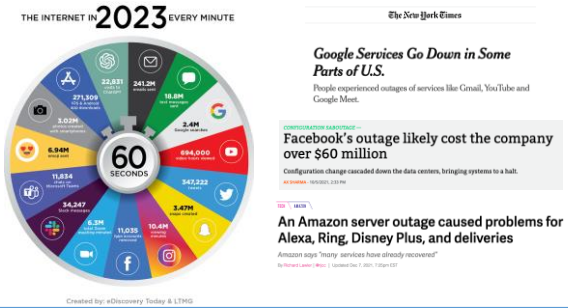
Download Cisco report: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html

Global traffic

1 Exabyte =







The New York Times

Google Services Go Down in Some Parts of U.S.

People experienced outages of services like Gmail, YouTube and Google Meet.

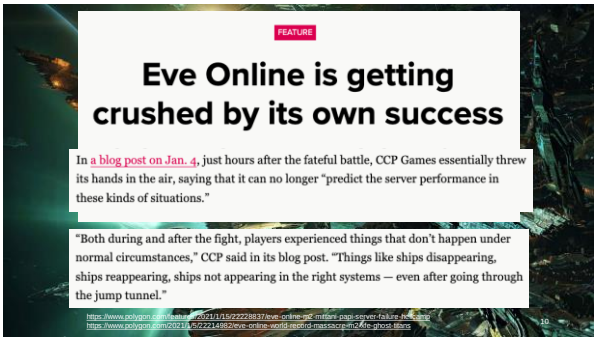
Facebook's outage likely cost the company over $60 million

Configuration change cascaded down the data centers, bringing systems to a halt.

An Amazon server outage caused problems for Alexa, Ring, Disney Plus, and deliveries

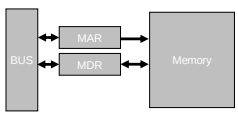
Amazon says "many services have already recovered"

Further reading on Facebook outage:
1. https://blog.cloudflare.com/october-2021-facebook-outage/
2. https://engineering.fb.com/2021/10/05/networking-traffic/outage-details/



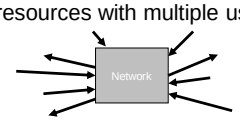
https://www.polygon.com/features/2021/1/5/22228837/eve-online-pc-online-pod-servers-failure-battle-bug
https://www.polygon.com/2021/1/5/22214982/eve-online-world-record-massacre-m2-fix-ghost-titans

Why are existing systems not good enough?

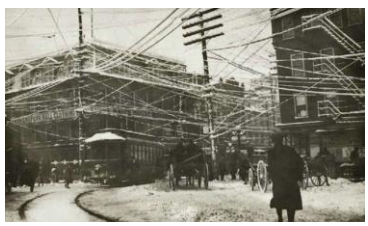


Examples of challenges:

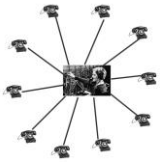
1. Latency is unknown and/or unbounded
2. Data channels are unreliable
3. Sharing resources with multiple users
4. …



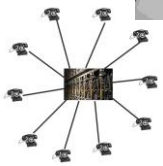
Early telephone system



Telephone switching



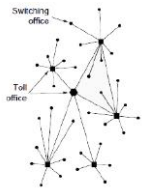
Human operator



Strowger gear

Hierarchical topology

Hierarchical topology

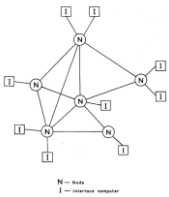


Military is a big fan of resilient systems

- US military asked RAND Corporation to design a better system (in 1960).
- Paul Baran (RAND employee) designed a fault tolerant network.
- Military asked AT&T to build it.
- They refused…
- Baran's design was forgotten…
- But design improved upon by NPL, built by ARPA.

Network designed by the National Physical Laboratory

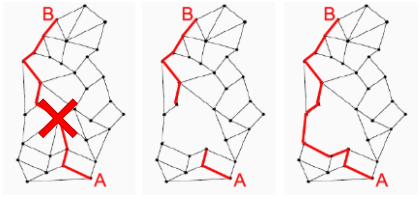
- NPL paper cited Baran but went further
- **Divided files into chunks called packets**
- **Store-and-forward packet switching network**



They did not build a prototype, but described its design.

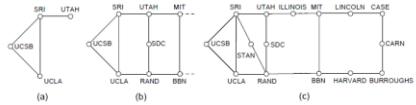
The ARPANET A mesh-structured network

The ARPANET
Fault tolerance

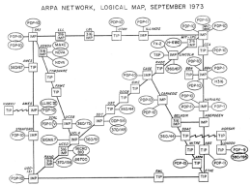


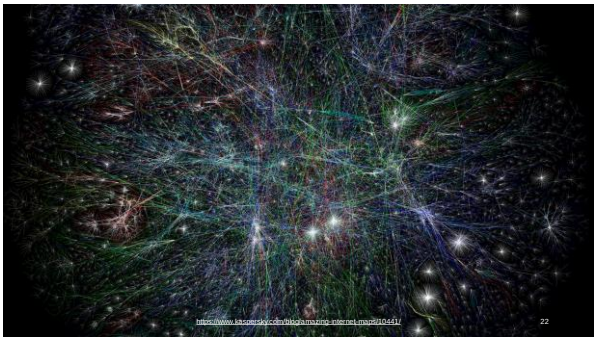
The ARPANET
Growth over time

Growth of the ARPANET.
(a) December 1969.
(b) July 1970.
(c) March 1971.



The ARPANET
Network state in 1973





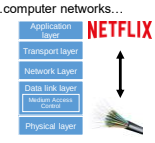
Layered architecture

Can be found in…

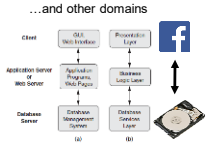
Q: Why use a layered architecture?

Q: Can you think of another domain that uses layered architectures?

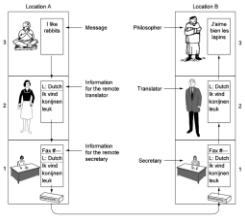
…computer networks…



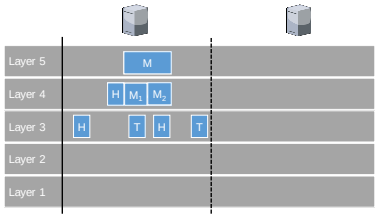
…and other domains



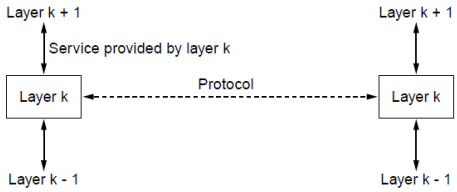
Layered architecture in computer networks: an analogy



Layered architecture in computer networks: an overview



Illusion of direct communication



How scale affects networks design

Personal Area Network (PAN)
- Example: Bluetooth

Local Area Network (LAN)
- Examples: WiFi (802.11)

Metropolitan Area Network (MAN)

Wide Area Network (WAN)

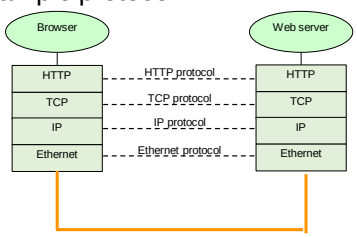
The Internet

Interprocessor distance	Processors located in same	Example
1 m	Square meter	Personal area network
10 m	Room	Local area network
100 m	Building	
1 km	Campus	
10 km	City	Metropolitan area network
100 km	Country	Wide area network
1000 km	Continent	
10,000 km	Planet	The Internet

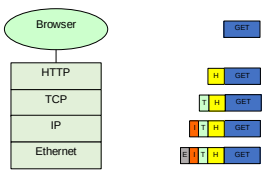
How the medium affects network design

Different frequencies have different physical properties!

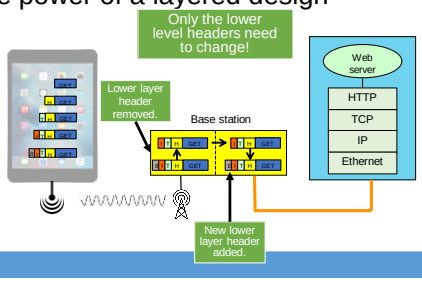
An example protocol



Encapsulation in a protocol stack



The power of a layered design



Multiple reference models for computer networks

Each models has both advantages and disadvantages.

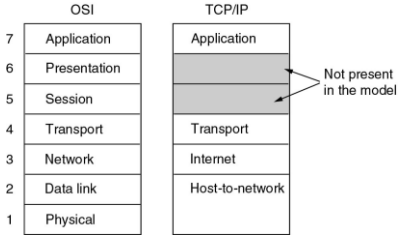
OSI model

1. Design by committee.
2. Strictly separated layers.

TCP/IP model

1. Widely used in practice.
2. Low generality.
3. Poor separation of concerns and interface design.

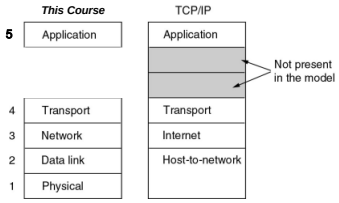
OSI versus TCP/IP



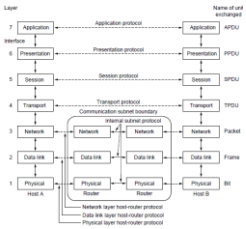
The model used in this course

The OSI model is well-designed, but layer 5 and 6 are almost empty

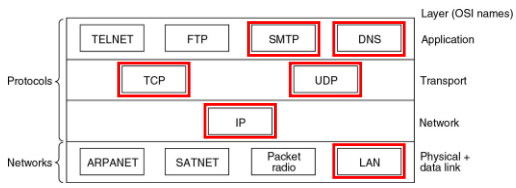
So we skip them!



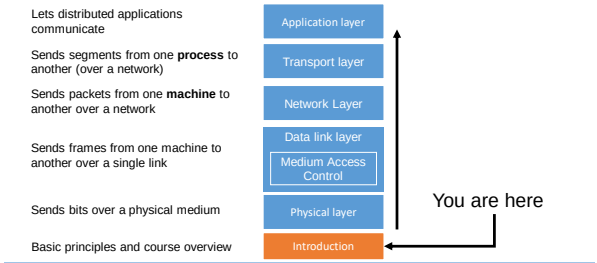
The OSI reference model

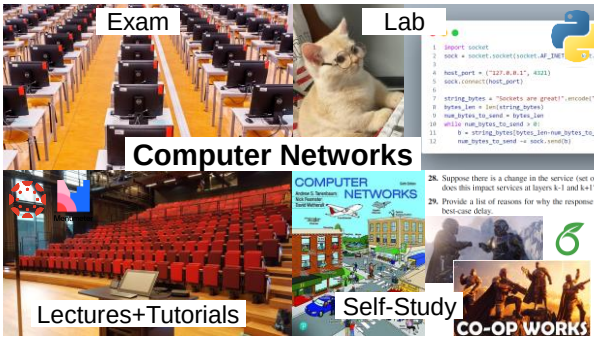


Protocols and Networks from the TCP/IP model

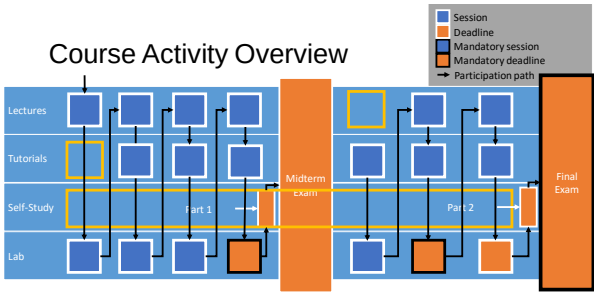


Roadmap of the Computer Networks Course

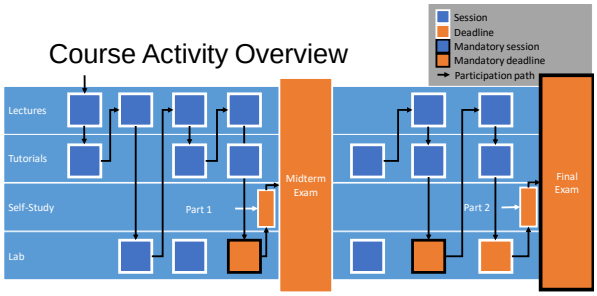




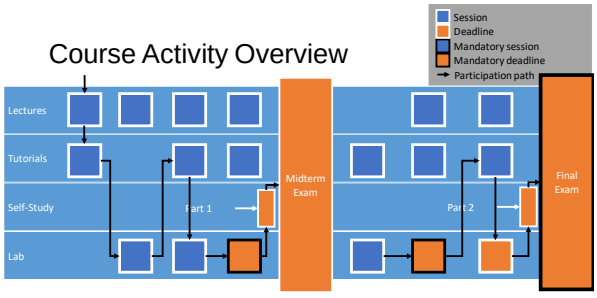
Course Activity Overview



Course Activity Overview



Course Activity Overview



How Am I Graded?

$$\text{grade} = \frac{\text{exam} + \text{lab} + \text{in class} + \text{self study}}{1000}$$

* You cannot pass the course without passing the mandatory lab assignment.

Lectures

Collect points by:
- Giving good answers to questions
- Answering correctly questions from the *in-lecture quizzes*

First quiz is today!

Tutorials: Plenary Practice Sessions

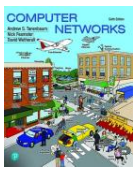
Please use:
- Pen
- Paper

Do not use:
- Calculators
- AI Chatbots
- Other external tools

Self-Study: Completing Book Exercises

Complete exercises from the book in a group.

Earn more points by completing more chapters.



Self-Study Checkpoints

Graded at two “checkpoints.”

Part 1:
- If you successfully completed 2 chapters, you receive +500 points.

Part 2:
- If you complete 2 more chapters, you receive +500 points.
- If you complete 4 more chapters, you receive +1000 points.

How to Participate in the Self-Study?

Join a Self-Study group on Canvas.
Deadline: 12 April.

Submit your Self-Study Plan.
Deadline: 12 April.

We recommend starting as soon as possible, and not wait for this deadline

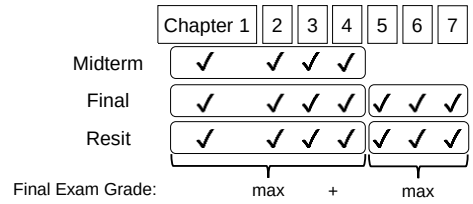
Exams

- Midterm (April 22) and Final (May 31)
- Computer-based (TestVision)
- Multiple-choice questions
- Every correctly answered question earns you 300 points

Getting 60% on the exam is not sufficient to pass the course!

Register for the exam on VUnet

Exam Content and Grades



Lab

Logistics

Lab

Labs on Wednesdays and Fridays.

Use the Canvas groups page to enroll for the one of the sessions.

Lab

Lab Guide specifies several optional assignments

Assignment 1 and 2 are mandatory

For assignment descriptions,
see the ***Lab Guide*** on Canvas.

Lab

Collect points by Completing Lab assignments.

- Small reward for the mandatory assignments
- Larger rewards for the optional assignments

How to Participate in the Lab?

1. On Canvas, create a CodeGrade group for each assignment

Resit students: please team up with another resit student or work by yourself

2. Complete the assignment(s)
3. Submit the assignment(s) on Canvas
4. Enqueue **during** the lab to discuss your solution with a TA

Show and explain your solution for the two mandatory assignments during a lab session in ***week 4*** and ***week 7*** at the latest, respectively

Show and explain your solution for other assignments during a lab session during or in ***week 8*** at the latest

Assignments uploaded to Canvas are not accepted without without TA approval obtained during the lab

Lab Logistics

Important:

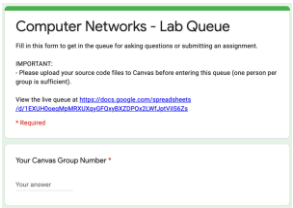
1. The assignments may take more than 4 hours to complete.
2. Getting your assignment approved takes time.

Complete the assignments before the day of their deadline!

How to Hand in Lab Assignments

Submission System:

1. Complete assignment.
2. Upload code/report to Canvas.
3. Enter Queue ➞
4. Wait for, and discuss with, TA.
5. ✅ Assignment approved.
   -or-
   ❌ Go to step 1.



How to Hand in Lab Assignments

We use a ***queue***, which means First-Come, First-Serve (FCFS)

Important:

1. Queue closes ***before*** the end of the lab session.
2. Closed queue not a valid excuse for not completing assignments.

Enqueue on time

Do not wait until last session before the deadline

Lab Assignments

Getting Started

Network layer services

Sends segments from one **process** to another (over a network) — Application layer / Transport layer

Sends packets from one **machine** to another over a network — Network Layer

Sends frames from one machine to another over a single link — Data link layer / Medium Access Control

Sends bits over a physical medium — Physical layer

Network layer services

Sends segments from one **process** to another (over a network) — Application layer / Transport layer

This is the service used by your application

What kind of interface does it use?

Socket Primitives in TCP

Socket – create a new communication *endpoint*.

Connect – connect to a remote *listening* socket.

Q: Are we missing something?

Send – send data to the other application.
Receive – receive data from the other application.
Close – close the connection.

Used to allow incoming connections

Bind – assign a *local address* to the socket.
Listen – wait for a connection.
Accept – passively accept an incoming *connection request*.

TCP Sockets in Python

```
# Import the socket library.
import socket
# Create a new socket.
s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
# Connect to another application.
s.connect(("hostname", port_number))
# Send bytes.
num_bytes_sent = s.send(buffer)
s.sendall(buffer)
# Receive bytes.
buffer = s.recv(2048)
# Close connection.
s.close()
```

Network layer protocol

Transport layer protocol

Max number of bytes to receive

Application layer
Transport layer
Network Layer
Data link layer
Medium Access Control
Physical layer

TCP provides a reliable byte-stream

Q: What does this mean for your application?

1. The program waits until data is available
2. It may return an arbitrary number of bytes

s.recv(2048)

H	E	L	L	O	-	F	R	O	M		J	E	S

s.recv(2048)

S	E	\n	W	H	O	\n

s.recv(2048)

S	E	N	D		E	C	H	O	B	O	T		H

s.recv(2048)

E	L	L	O		W	O	R	L	D	\n

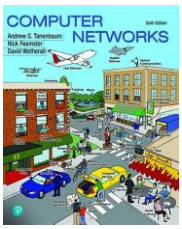
Threading Python

```
# Import threading library.
import threading
# A regular call to print.
print("Hello", "World")
# A threaded call to print.
t = threading.Thread(target=print, args=("Hello", "World"))
# Run target in new thread.
t.start()
# Wait 100ms for thread to finish.
t.join(0.1)
```

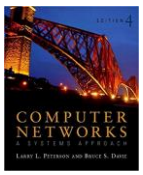

Course Material

Course Material:

1. Course Slides
2. Book:
   Computer Networks,
   6th edition,
   Andrew S. Tanenbaum, Nick Feamster, and David J. Wetherall



Other Computer Networks Books



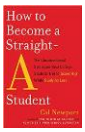
Peterson and Davie

Available for free at
https://book.systemsapproach.org

Kurose and Ross

How to Study (for this Course)

Or: how we view our own course



Teaching (us) and learning (you)

Our recommendations:

- Producing is better than consuming (Do exercises, write programs, quiz yourself, etc.)
- Learn how to study effectively
  - Example books: Make It Stick, How to Become a Straight-A Student

Meet the Team!

How to Contact the Team?

Talk to us at the lab/lecture/tutorial

- Expected response time: *minutes*

Canvas discussion board

- Expected response time: *hours*

Mail us at compnet2024.beta@vu.nl

- Expected response time: *days*

Next steps

1. Participate in the Entry Quiz! Earn your first points!
2. Read the course syllabus (10 pages)
3. Obtain a copy of the book!
4. Find a lab partner.
   The lab is done in teams of **2** students.
   1. Register your group on Canvas
   2. Can't find a partner? Look for one on the Canvas discussion board
   3. Contact the Computer Networks team
5. Start looking for a self-study team

Next stop: Physical Layer