COMP2221 Networks

David Head

University of Leeds

Lecture 1

Today

- Module overview and admin.
- Materials on Minerva.
- Assessments (coursework and exam).
- A brief history of networks.
- Key concepts.

Resources

Minerva → COMP2221 Networks

Lecture notes:

- Slides (PDF).
- Also any code examples.

Worksheets

- **Formative**, *i.e.* not assessed.
- 4 worksheets, with at least one question for most lectures.
- All worksheets available now.
- Specimen answers will appear on Minerva roughly 1-2 weeks after the corresponding lecture.

Support

Support is *via* labs (see later) and Microsoft Teams.

Please use the **Microsoft Teams** for any and all queries.

- Check first your query has not already been answered.
- Use public page to ensure all students have access to the same information/guidance.

This is the best way to make sure everyone receives the same information and guidance.

- Be careful not to put any of your code for the coursework on Teams – but it is fine to post queries seeking clarification of the coursework instructions.

Labs

Labs are for practical help with coding exercises and assignments. These are all in **Bragg 2.05**:

- Mondays 3-5pm.
- Wednesdays 11am-1pm.
- Fridays 9-11am.
- Fridays 11am-1pm.
- Fridays 3-5pm

The **teaching assistant(s)** (TBA) and module staff will try to attend as many as we can, especially while the coursework is active (Weeks 6-8).

Exercises and assessments

There are two **formative exercises** on Minerva / Gradescope.

- Java refresher – **Worksheet 1 Question 1**. Can do now.
- Basic networking in Java – **Exercise**. Can do after Lecture 6.

These are **not assessed**, *i.e.* they are both optional. Both have an autograder that returns marks immediately.

One **assessed coursework** worth 30% of the module grade that requires material up to and including Lecture 11.

	Set	Due
Coursework	By 6th March	2pm, Monday 24th March

The remaining 70% is assessed in the **open book exam**:

- Format to be be announced closer to the time.

Plagiarism / Collusion

Generative AI is allowed in an assistive capacity only. Use comments in your code to declare any use of generative AI, making clear to what extent it was used.

You can use code given out in lectures as a starting point for your coursework without referencing. However, if using code from any other source, **make clear exactly which code you used and from where** by using comments.

For all submissions, **code similarity** checking tools will be used to check for potential cases of **collusion**.

Module objectives

From the module catalogue . . .

This module provides the fundamental understanding of network programming, network architecture and security consideration of network applications.

Learning outcomes: On successful completion of this module a student will have demonstrated the ability to:

- Describe the key components of a computer network.
- Design, implement and test network protocols and applications.
- Evaluate network protocols in terms of efficiency and security.
- Select appropriate network protocols for specific applications.

Syllabus

From the module catalogue . . .

This module covers the following 3 topic areas:

- Network programming : design patterns and network protocols (TCP, UDP, DNS, DHCP, SMTP and POP3).
- Network architecture : local area networks, metropolitan area networks, wide area networks, OSI model.
- Network infrastructure : Ethernet, 802.11a/b/g/n, network devices (switches, routers and access points).

The focus will be on **network programming**, in Java.

Language

For this course we will use **Java**[1]:

- **Object oriented** (OO) (inheritance, polymorphism *etc.*)
- Standard libraries for network programming, *e.g.* `java.net`, `java.nio`, `java.security` *etc.*
- We will also make use of **I/O streams** from `java.io`.

If you're worried your Java is a little rusty . . .

- Try **Worksheet 1 Question 1** now, which is in Java but requires no specific networking features.
- Also attempt the second formative **Exercise** after Lecture 6, which includes our first network–specific Java.

[1] The coursework autograder uses OpenJDK 22, as per the lab machines.

Books

This course is primarily based on two books:

- **Computer Networking: A top-down approach**.
  - *James Kurose and Keith Ross (Pearson)*.
  - Covers network architectures from applications to the hardware, but has very few coding examples.

- **Java Network Programming**.
  - *Elliotte Harold (O'Reilly, 2013)*.
  - Covers much of the coding we will need, but barely mentions architecture.
  - Chapter numbers vary greatly from earlier editions.

Copies of both in the libraries *(various editions)*, with online versions. **There is no need to buy these books!**

Course structure

Roughly speaking, the structure of this course is as follows:

1. Overview of the layered description of networks, and key concepts such as IP addresses, ports, DNS *etc.*
2. Network programming in Java, including the coursework.
3. More detailed look at the internet infrastructure to help understand performance.

This follows a similar overall structure to the Kurose and Ross book, *i.e.*, an overview followed by a 'top-down' approach.

A more detailed breakdown will be given at the end of the *next* lecture, once we've seen what this 'layered' description is.

Level 3 modules

Some level 3 modules have COMP2221 Networks as a pre-requisite:

Module	**Relationship**
COMP3911 Secure Computing[1]	Extends what we cover in Lecture 13
COMP3211 Distributed Systems	Builds on the material we cover in this module to something much closer to modern networking applications

There is also some overlap with COMP3221 Parallel Computation *(some multi-threading in Lectures 9-11 and the coursework)*.

[1] Compulsory for most programmes; not available for CS Maths *(please check)*.

What are networks?

Networking is pervasive in the modern world:

- Local area networks.
- The internet.
- Cloud computing.
- Peer-to-peer distribution.
- *Many* more.

The first network: ARPAnet

(Following Kurose and Ross, 7th ed., §1.7)

The first computer network was **ARPAnet** at MIT, when in 1969 two machines were connected together.

- Used a **packet switch**, similar to phone networks.
- Built on technologies developed by MIT, the Rand Institute, and the National Physical Laboratory, UK.
- By 1972 had 15 nodes.

This was followed by many others: **ALOHANet** linking Hawaiian universities *via* microwaves; **DARPA** networks for defense; the commercial **Telenet**; **Cyclades** in France, and more.

The Internet

Interconnecting networks, or a **network of networks**, was pioneered by Cerf and Kahn in 1974.

- Was termed **internetting** . . .
- . . . from which we get the word **internet**.
- Development of **protocols**, including an early version of TCP = Transmission Control Protocol.

Rapid growth of networks in the 1980's.

- BITNET (email, file transfers, . . .)
- CSNET (university computer science network)
- Minitel (France; phone directory, banking, . . .)
- . . .

The 1990's

There was an explosion in internet usage in the 1990's, not least of which because of the development of the **World Wide Web**.

- Invented by Tim Berners-Lee at CERN (the European Nuclear Research Centre) to help particle physicists disseminate data, research papers *etc.*
- Involved HTML, HTTP, a Web server and a browser.
- Early browsers were Mosaic, and especially **Netscape**.
- Netscape eventually lost out to Microsoft Explorer (which has now lost out to Chrome, Edge *etc.*)

Also rapid uptake of e-mail, using clients such as `pine` (now replaced by `alpine`).

Netscape

http://www.andrewturnbull.net/mozilla/history.html

The Leeds connection

The University of Leeds was the first institution in the UK to have an officially sanctioned WWW presence[1].

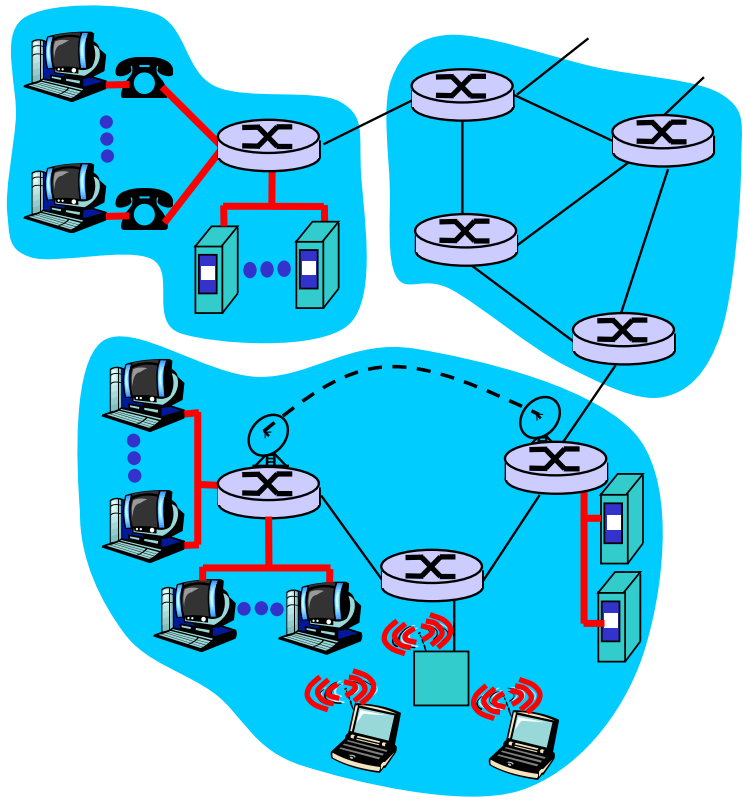
- Possibly helped by a visit from Robert Cailliau, a colleague of Tim Berners-Lee, who was visiting his partner's relatives in Yorkshire at the same time Leeds was developing a Campus Wide Information Service (CWIS).
- Music department was the first to have a web presence (April 1993).
- First UK university to put its computing service newsletter on the web (Nov. 1993).

[1] *How the web was born*, Gillies and Cailliau (Oxford University Press, 2000).

More modern networks

Some common network features:

- **Hosts** or **end systems** at start and end of the chain of communication.
- Includes **servers** *(machines with no monitor)*.
- **Routers** *(discs with an 'X')*.
- **Edge routers** or **access network** that directly connect to an end system.

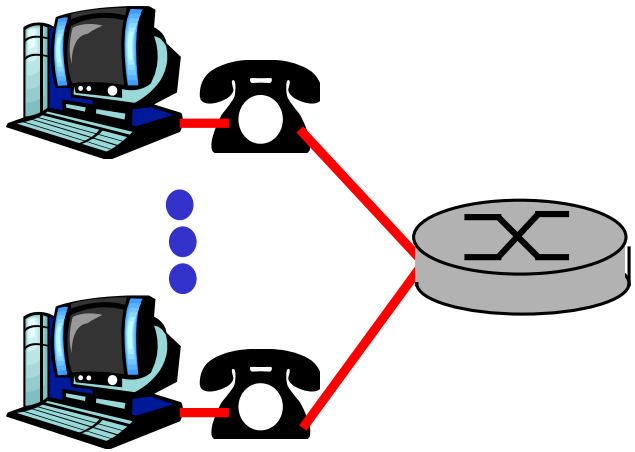


http://www-net.cs.umass.edu/kurose-ross-ppt-6e

Early residential access

Early residential access used standard **phone lines** and **modems**, and transmitted data as **sound**.

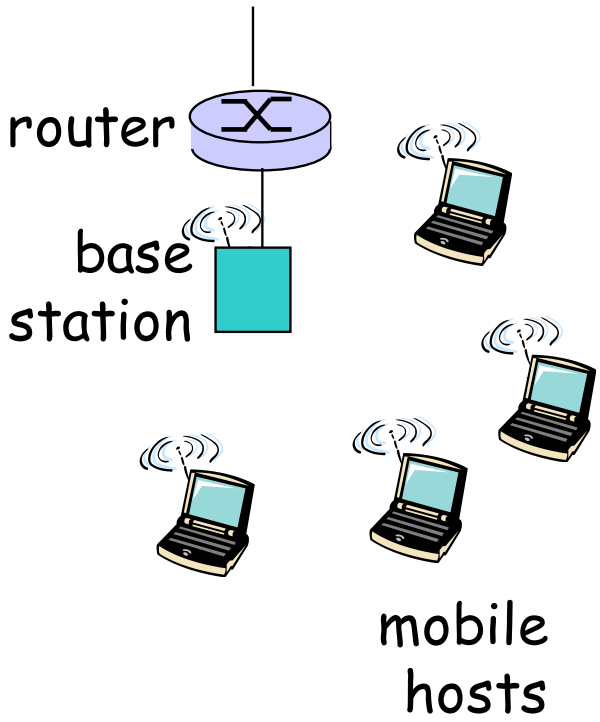
- Could not use phone and browse at the same time.
- Typically slow.
- Improved with **ADSL**, *i.e.* Asymmetric Digital Subscriber Line, which downloads faster than it uploads.



Wi-Fi

More common these days to have a **base station** that transmits to multiple **hosts** 'simultaneously.'

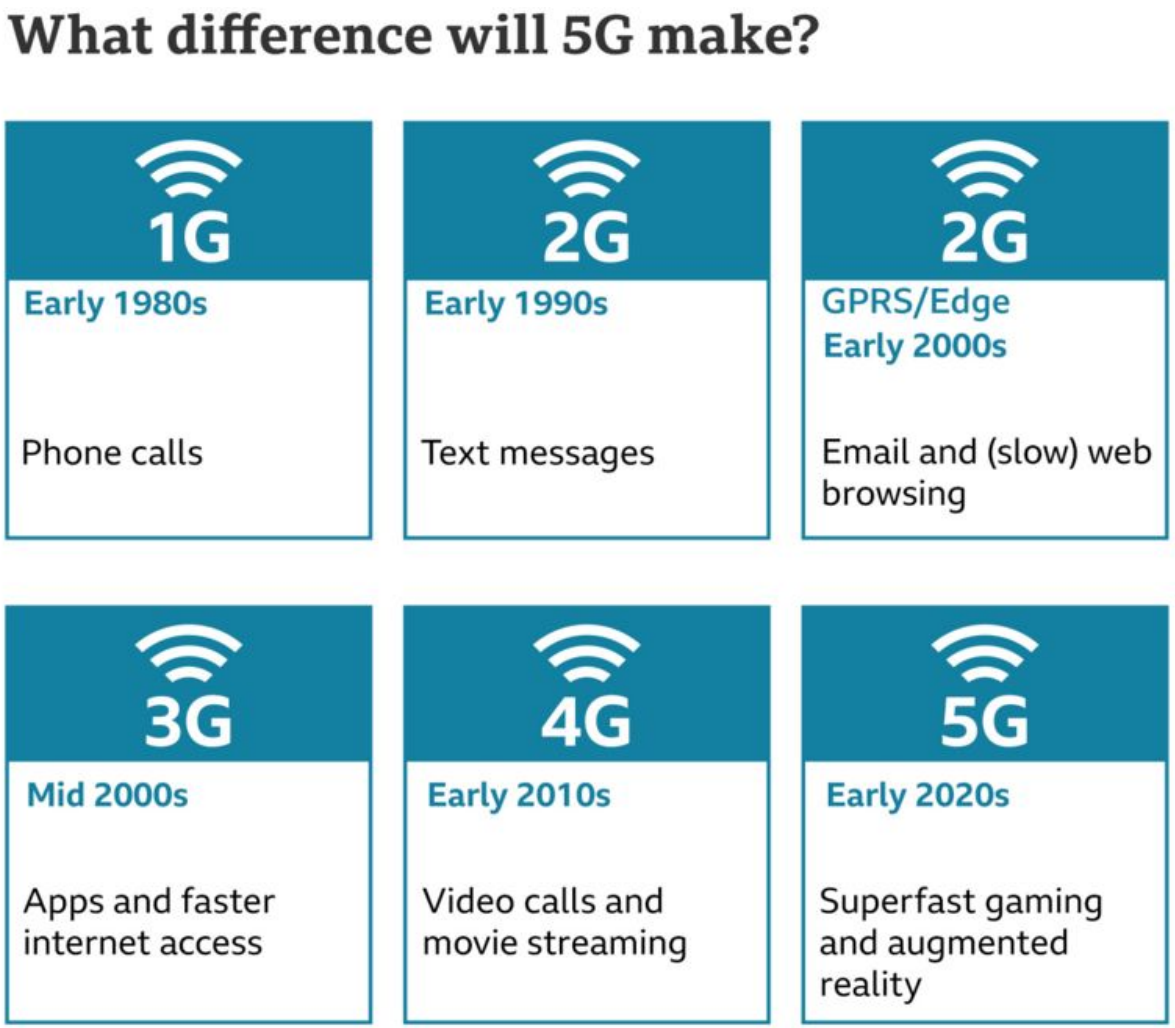
- 802.11b/g, up to newer versions, *e.g.* 802.11be.
- Up to gigahertz frequency bands.



Wider-area standard used for phones.

- 3G $\sim$ 7 Mbps[1].
- 4G $\sim$ 100 Mbps.
- 5G $\sim$ 10 Gbps.

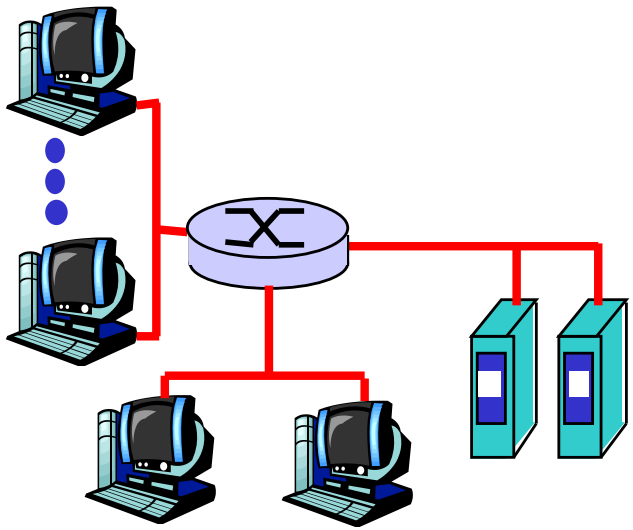
[1] Mbps: Megabits per second; sim. for kbps, Gbps *etc.*



https://www.bbc.co.uk/news/technology-54510361

Local area networks

Companies or universities connect **end systems** (*i.e.* computers, devices) to the **edge router** (*i.e.* the first/last router that data passes through from or to the device).



Typically connected *via* **ethernet**:

- Shared or dedicated link.
- Fast, *e.g.* '40G Ethernet' = 40 Gbps.

Summary

Today we have briefly covered the history of networks and some key concepts in modern networks:

- **Hosts** or **end systems** (laptops, devices, servers *etc.*)
- **Routers**.
- **Edge routers** or the **access network** (the last nodes before the end systems).

.

Over the next $3\frac{1}{2}$ lectures, we will cover basic network architectures and overview the **protocol stack**.

- We will then be able to start on application development.
- At the end of next lecture the whole course will be outlined.