COMP2001J Computer Networks

Lecture 1 - Introduction

Dr. Shen WANG (王燊)

shen.wang@ucd.ie

北京工业大学北京-都柏林国际学院
BEIJING-DUBLIN INTERNATIONAL COLLEGE AT BJUT

Outline

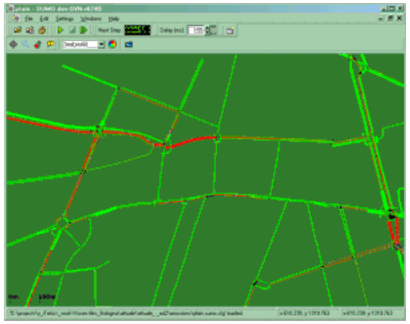
- Self-Introduction
- Logistics
  - Contents
  - Assessment
  - References
- Learning Tips
- Computer Networks
  - Examples
- Layered Architecture

Self-Introduction

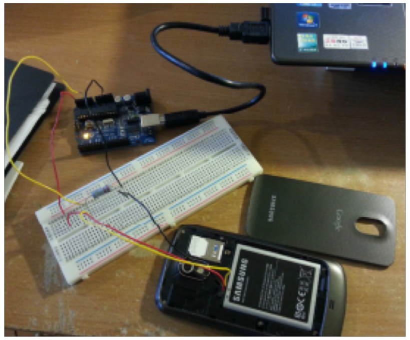
- Assistant Professor at the School of Computer Science, University College Dublin.
  - Research interests:
    - Wireless multimedia transmission
    - Vehicular communications & artificial intelligence for ITS
    - Big data streaming
    - Trajectory data mining
  - More: https://people.ucd.ie/shen.wang

Self-Introduction

- 2016. PhD:
  - Using **vehicular communications** and artificial intelligence to reduce urban road traffic



- 2012. Master of **Telecommunications Engineering:**
  - Network selection for optimal energy consumption on Android devices and Arduino testbed



Self-Introduction

- 2016-2018: CeADAR, UCD
  - EU H2020
  - ADS-B data

- 2015: IBM Research – Brazil:
  - GPS data mining

Assessment

- Continuous Assessment: **30%**
  - Midterm-Exam: **10%**
    - Week 9 (15th April)
    - Mid-tern-Exam **CANNOT** be re-scheduled!
  - Lab Work: **20%**
    - 12 Labs week 3 – 15 (**excludes week 11**)
    - 15:25-17:00 Monday TB4 room302
- Final-Exam: **70%**
  - **All contents** I will say during the lecture, present in slides, and lab sessions you will practice will be covered!
- Enrolment Key on CSMoodle: **CN19**
- **Plagiarism will be severely punished!**

Timetable (subject to change)

- Week 1:
  - Overview
  - Layered Architecture
- Week 2:
  - Physical Layer
- Week 3:
  - Link Layer - A
  - Lab 1: do quiz 1
- Week 4:
  - Link Layer - B
  - Lab 2: Packet Tracer - 1

Timetable (subject to change)

- Week 5:
  - Link Layer - C
  - Lab 3: do quiz 2
- Week 6:
  - Network Layer - A
  - Lab 4: Packet Tracer - 2
- Week 7:
  - Network Layer - B
  - Lab 5: Packet Tracer – Exam 1
- Week 8:
  - Network Layer - C
  - Lab 6: Packet Tracer - 3

Timetable (subject to change)

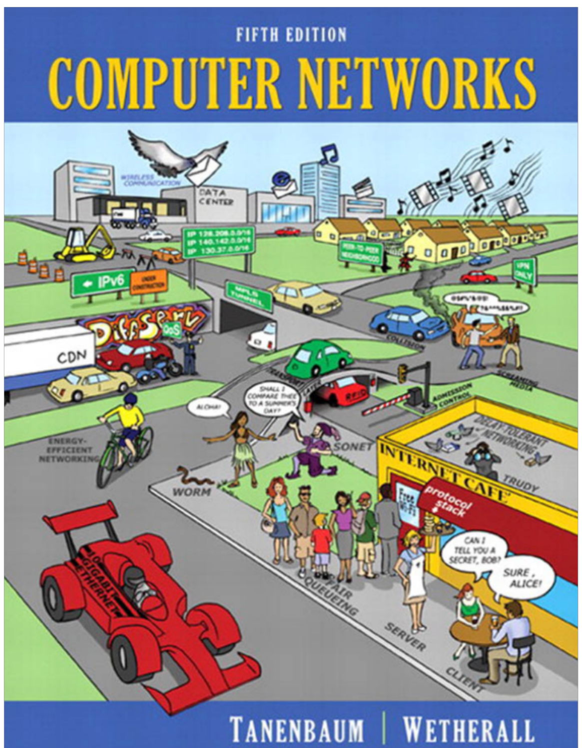
- Week 9:
  - Transport Layer - A
  - Lab 7: Mid-term exam
- Week 10:
  - Transport Layer - B
  - Lab 8: Packet Tracer – 4
- Week 11: Review week
- Week 12:
  - Application Layer
  - Lab 9: Packet Tracer – Exam 2

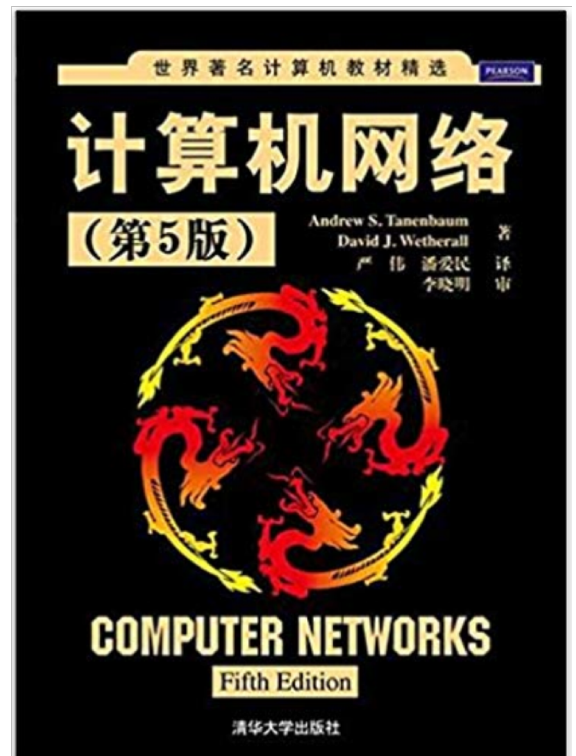
Timetable (subject to change)

- Week 13:
  - Network Security
  - Lab 10: Wireshark - Intro
- Week 14:
  - Frontiers
  - Lab 11: Wireshark - practice
- Week 15:
  - Review
  - Lab 12: Wireshark - exam

References

- Computer Networks 5th
  - By Andrew S. Tanenbaum. 2010



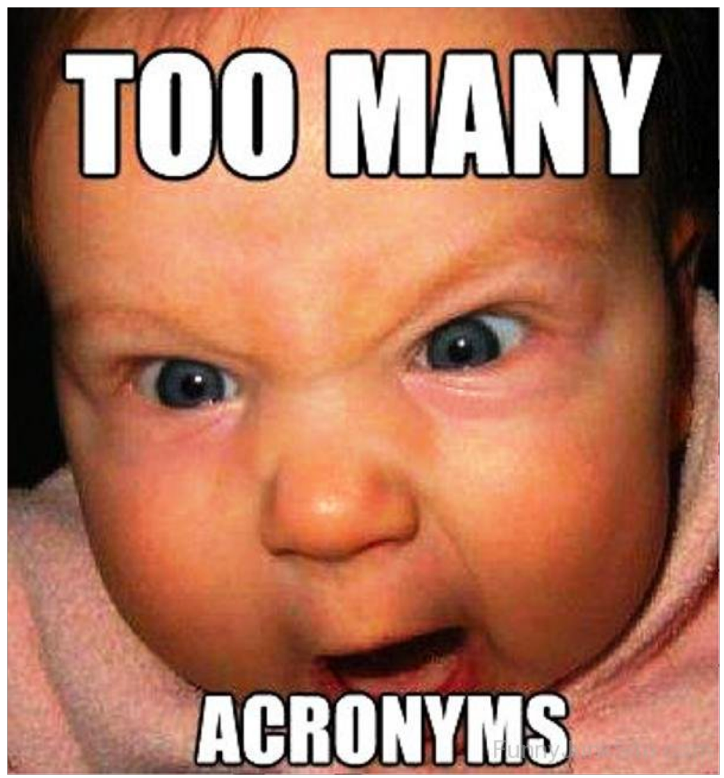


References

- TCP/IP Illustrated, Volume 1: The Protocols (2nd Edition)
  - by Kevin R. Fall and W. Richard Stevens, 2011
- Computer Networking: A Top-Down Approach (7th Edition)
  - by James Kurose and Keith Ross, 2016

Difficult to Learn

- IP TCP UDP ICMP DNS HTTP IGMP ARP MAC …



Tips to Learn - 1

WHAT

WHY

HOW

Tips to Learn - 2

- Always start from the simplest case!

Example: Computer Networks

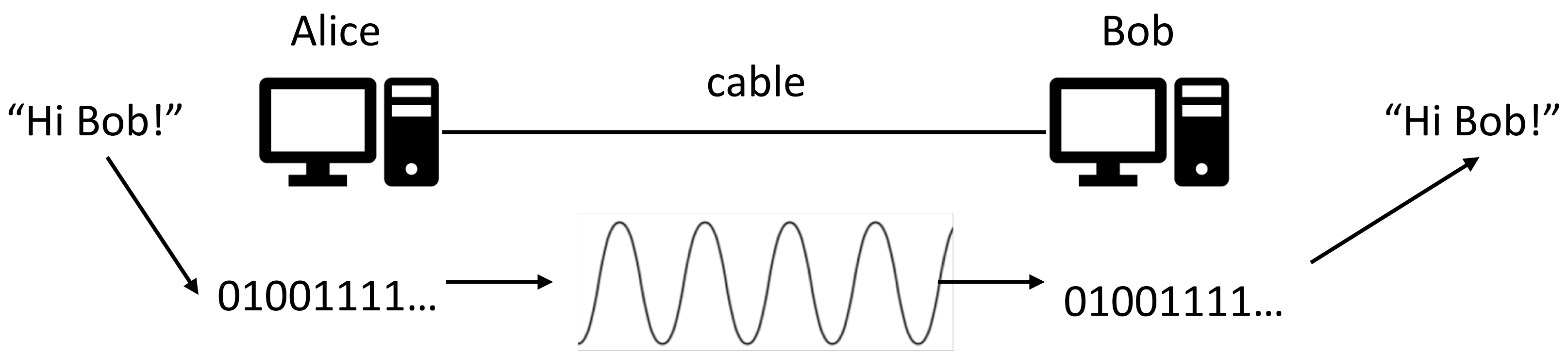
- What are computer networks?
  - Definition
- Why do we need computer networks?
  - Motivation
- How do computer networks work?
  - Technical Details

The simplest case

- Send “Hi Bob!” from Alice to Bob

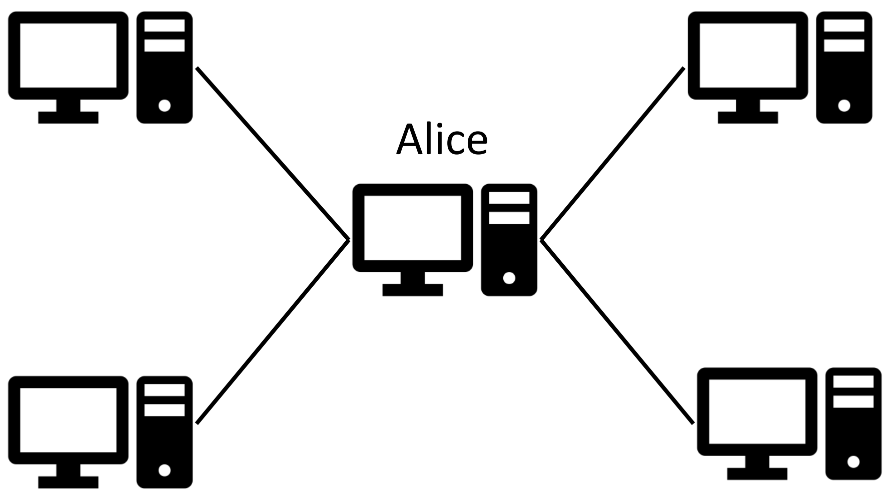
The simplest case

- Modulation/Demodulation
- Network Protocol



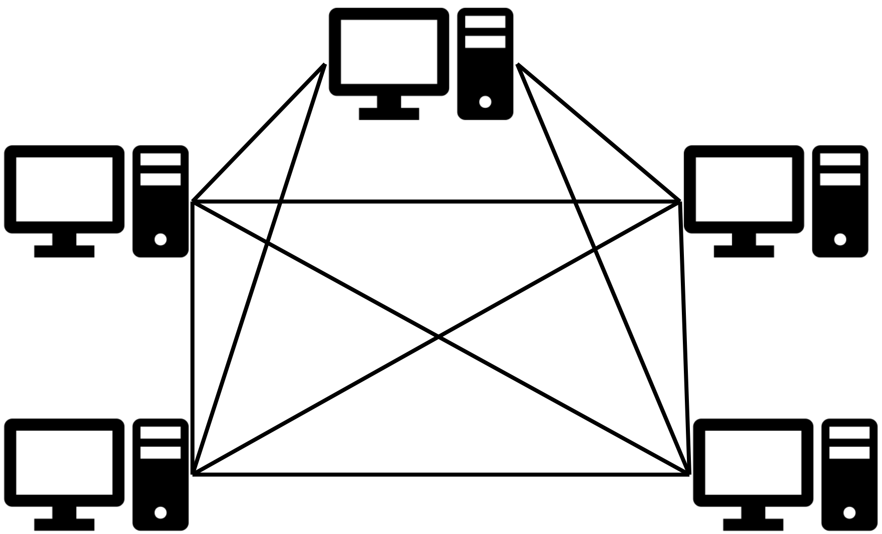
A more complicated case

- Alice wants to talk to more people
- But the rest people can talk to Alice only



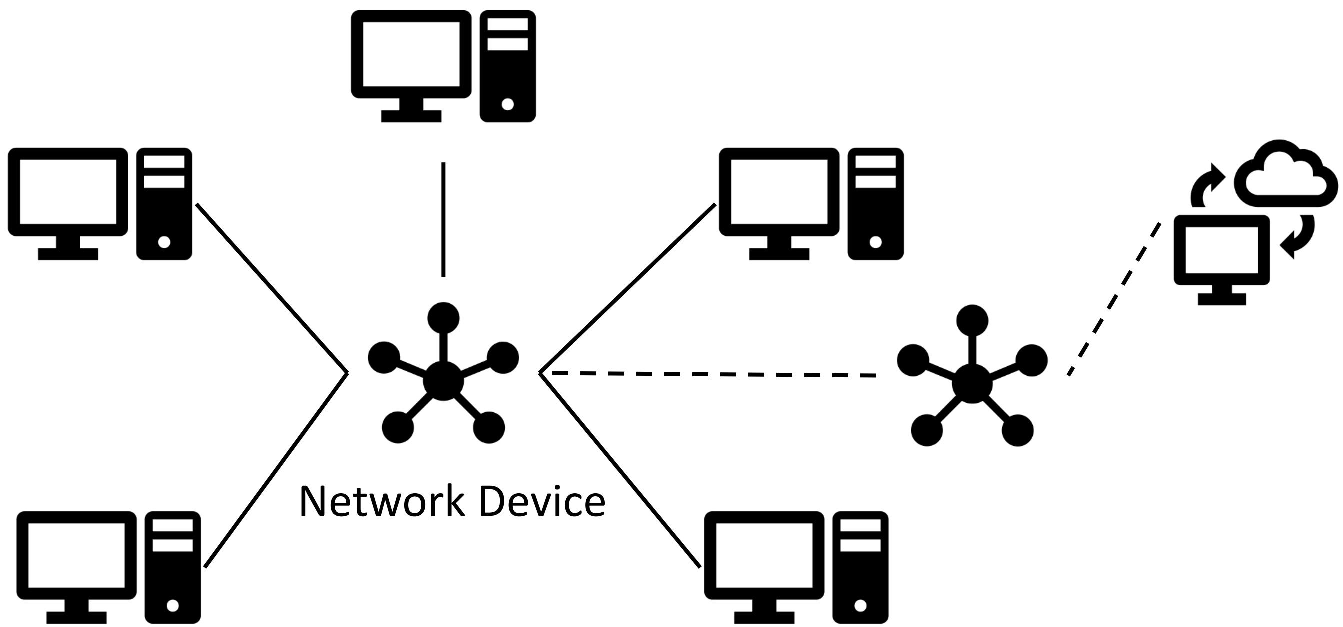
A more complicated case

- Now everyone can talk each other, but difficult to maintain
- If we have n computers want to talk to each other, we need n*(n-1)/2 cables



A more complicated case

- Now the number of cables reduced to n
- But how to coordinate if everyone wants to talk at the same time? And across multiple networks?...



Layered architecture

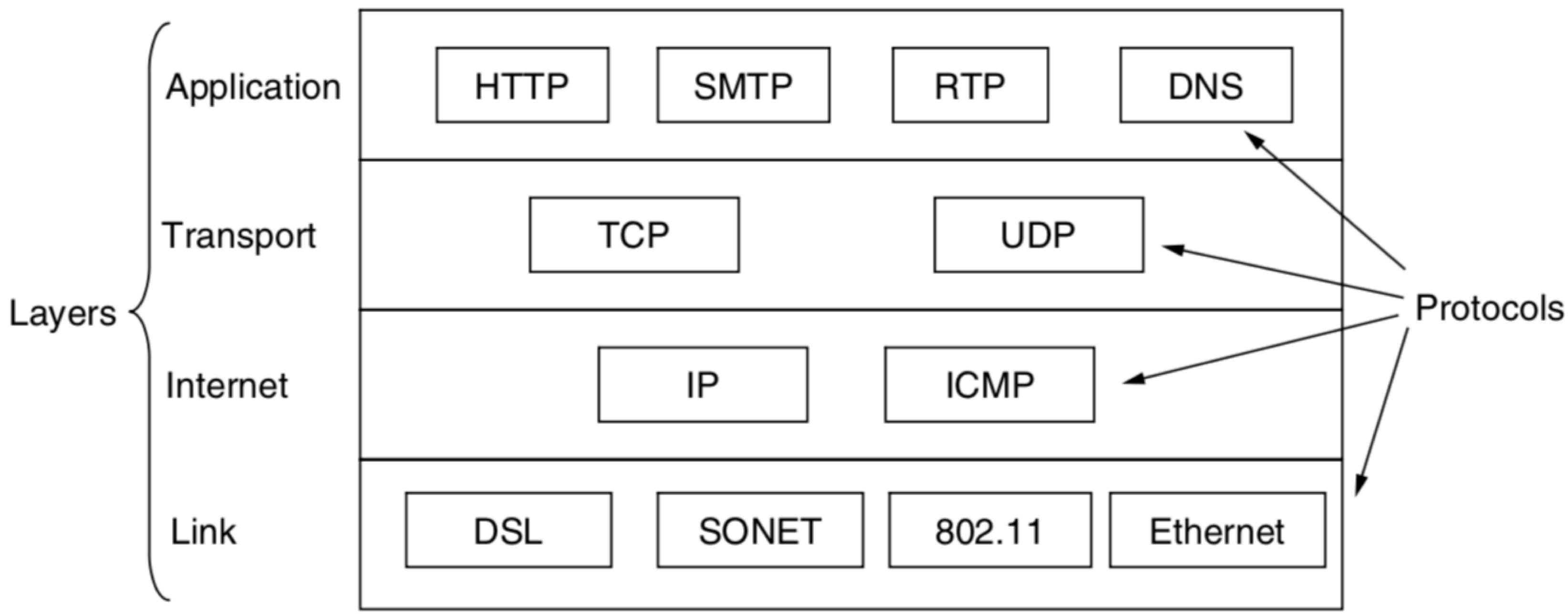


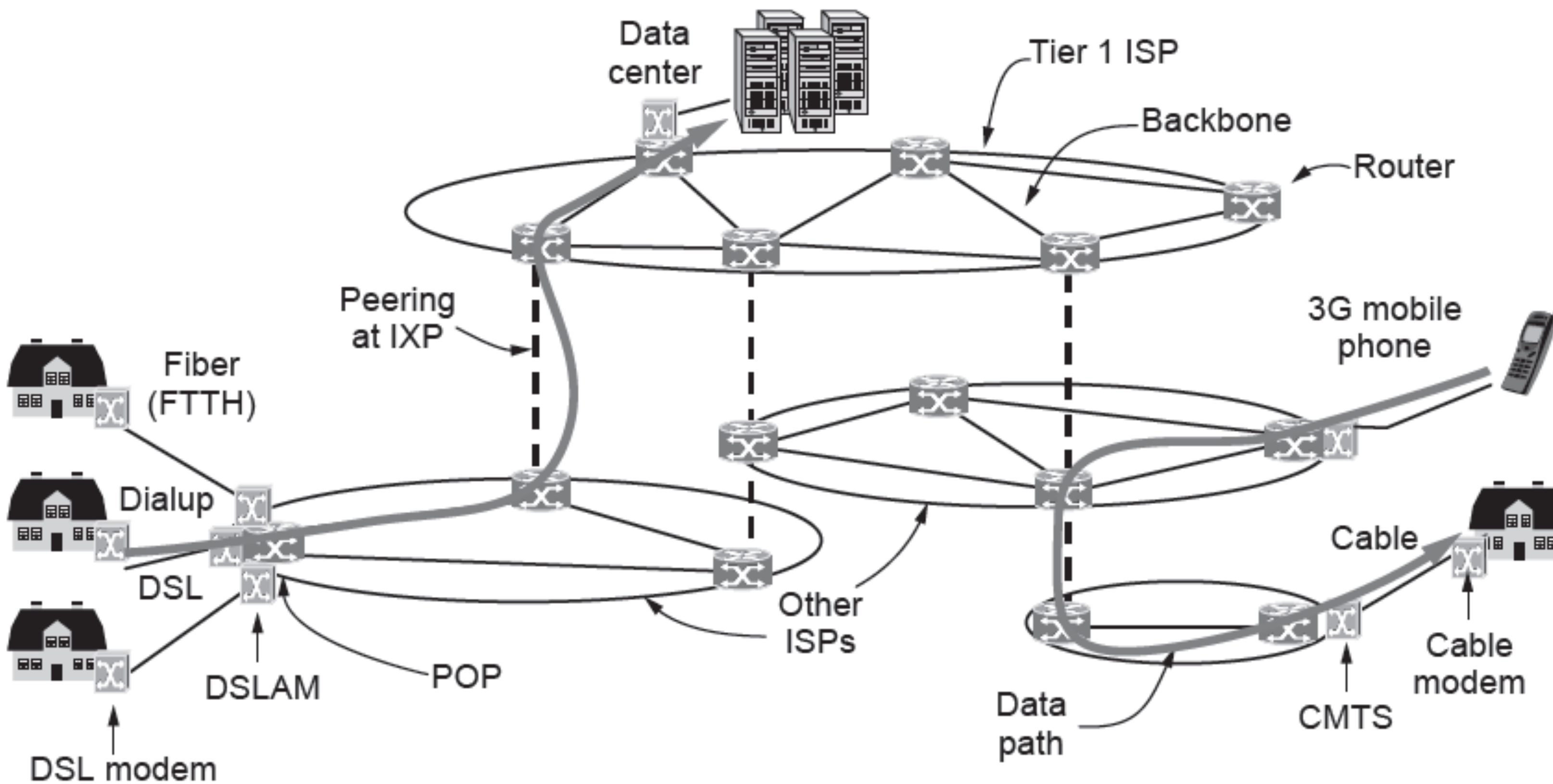
Figure 1-22. The TCP/IP model with some protocols we will study.

Re-emphasize!

WHAT? WHY? HOW?

ALWAYS START FROM THE SIMPLEST CASE!

Applications – Internet



- ISP: Internet Service Provider

Internet

- ISP networks serve as the Internet backbone
- ISPs connect or peer to exchange traffic at IXPs
- Within each network routers switch packets
- Between networks, traffic exchange is set by business agreements
- Customers connect at the edge by many means
  - Cable, DSL, Fiber-to-the-Home, 3G/4G wireless, dialup
- Data centers concentrate many servers ("the cloud")
- Most traffic is content from data centers (esp. video)
- The architecture continues to evolve

ARPA(Advanced Research Projects Agency)NET

- In the late 1950s, the U.S. DoD wanted a command-and-control network that could survive a nuclear war
  - the destruction of a few key toll offices could fragment it into many isolated islands
  - each IMP would be connected to at least two other IMPs. If some lines and IMPs were destroyed, messages could be automatically rerouted along alternative paths

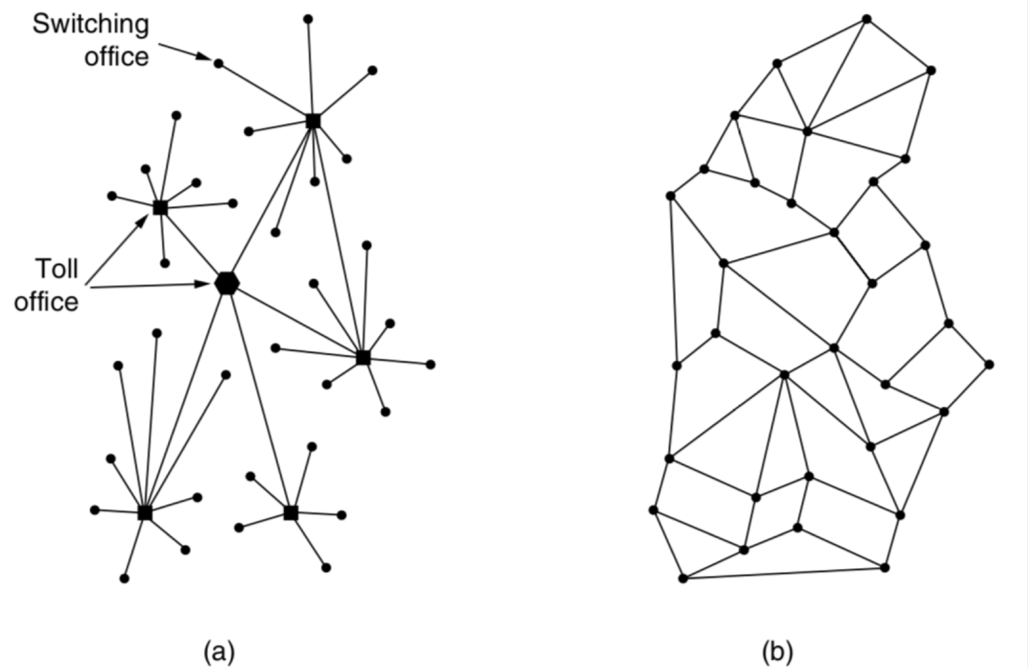


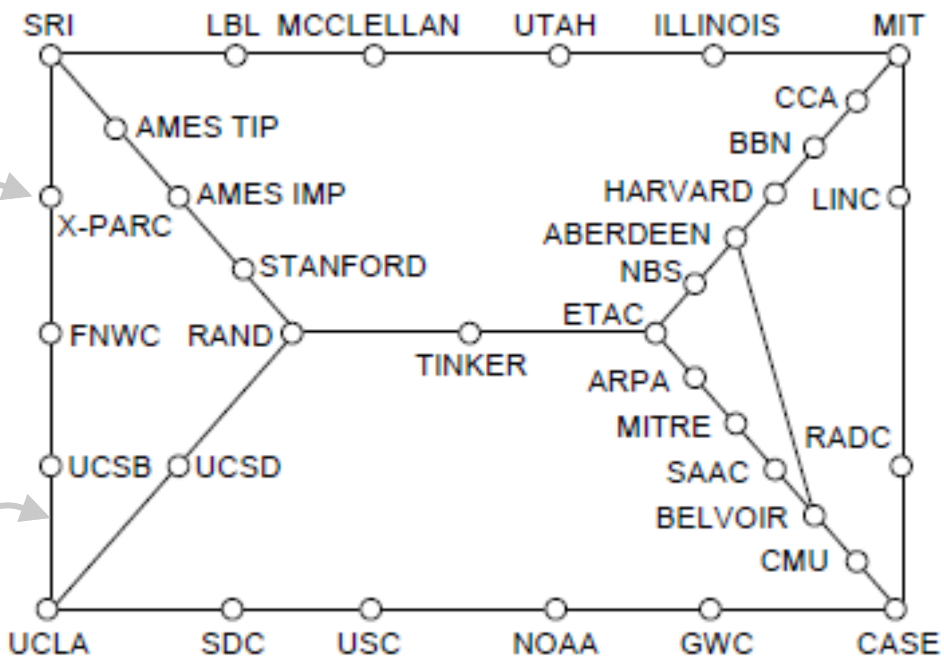
Figure 1-25. (a) Structure of the telephone system. (b) Baran's proposed distributed switching system.

ARPANET

- IMPs (Interface Message Processors)

Nodes are IMPs, or early routers, linked to hosts

56 kbps links

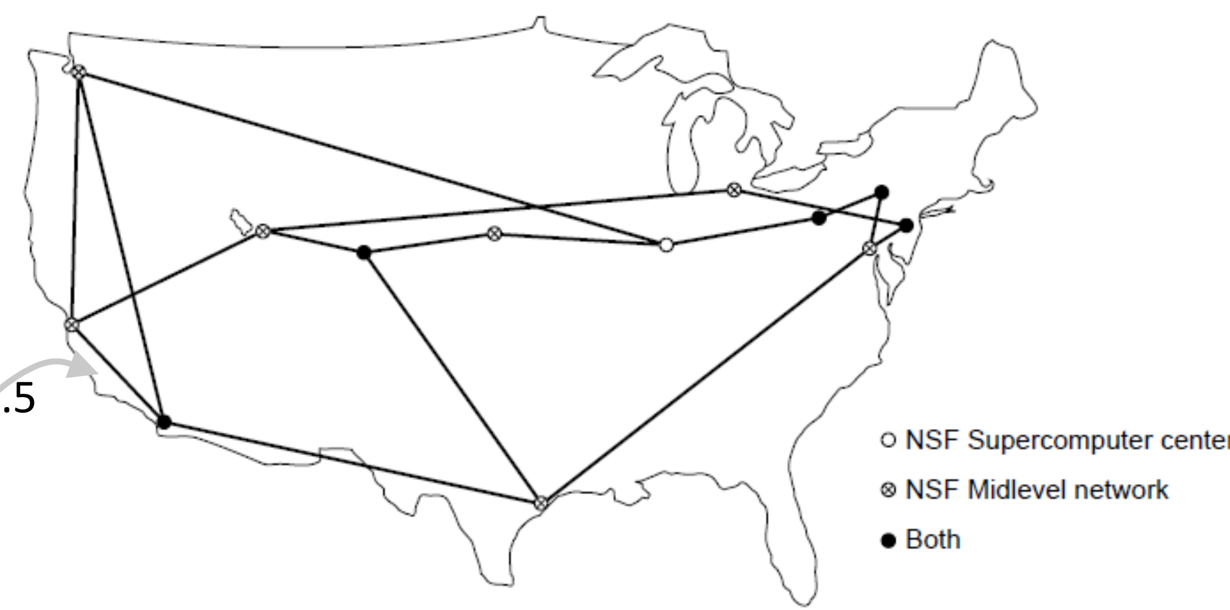


ARPANET topology in Sept 1972.

NSFNET

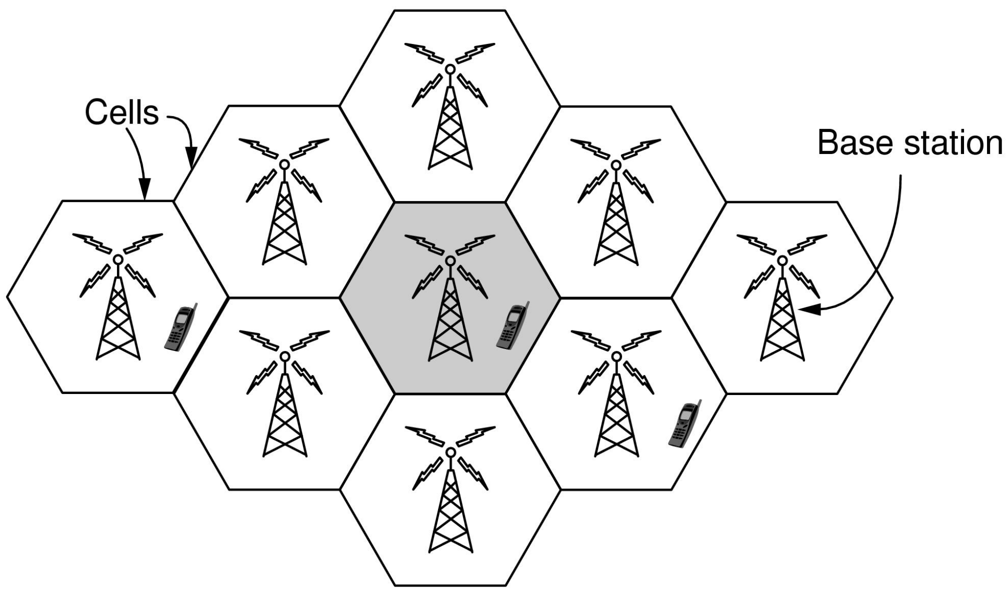
- The early Internet used NSFNET (1985-1995) as its backbone; universities connected to get on the Internet

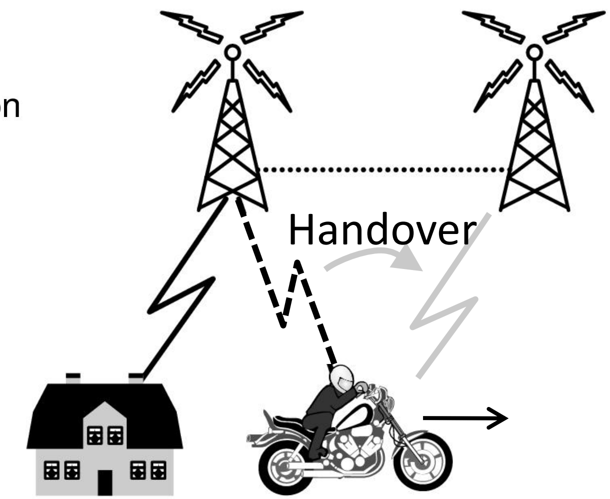
T1 links (1.5 Mbps)



NSFNET topology in 1988

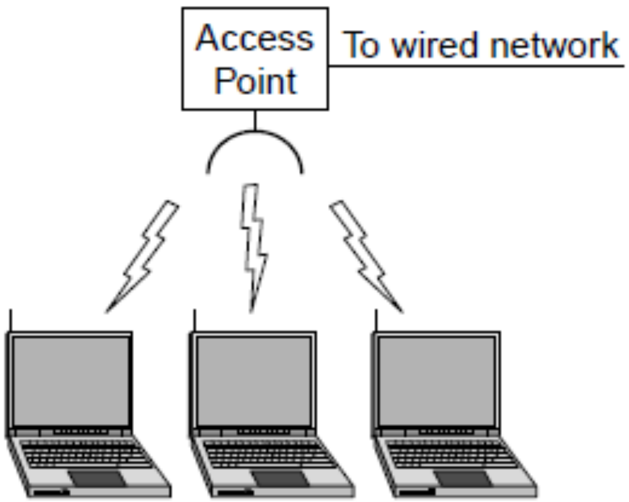
Applications – Mobile



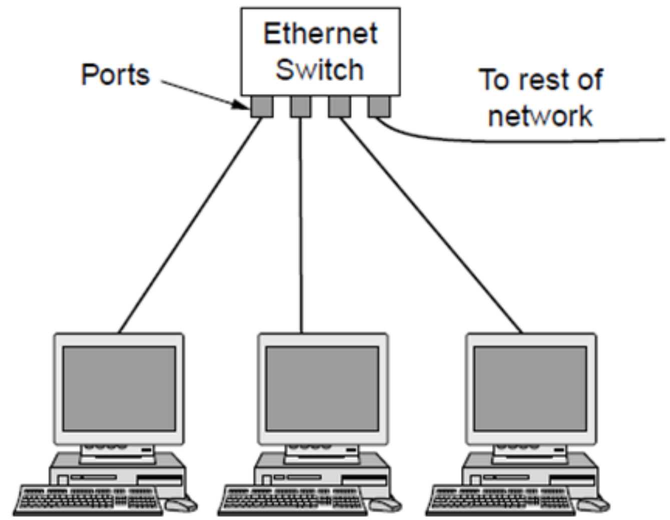


Applications – LAN

- Local Access Network - Connect devices in a home or office building



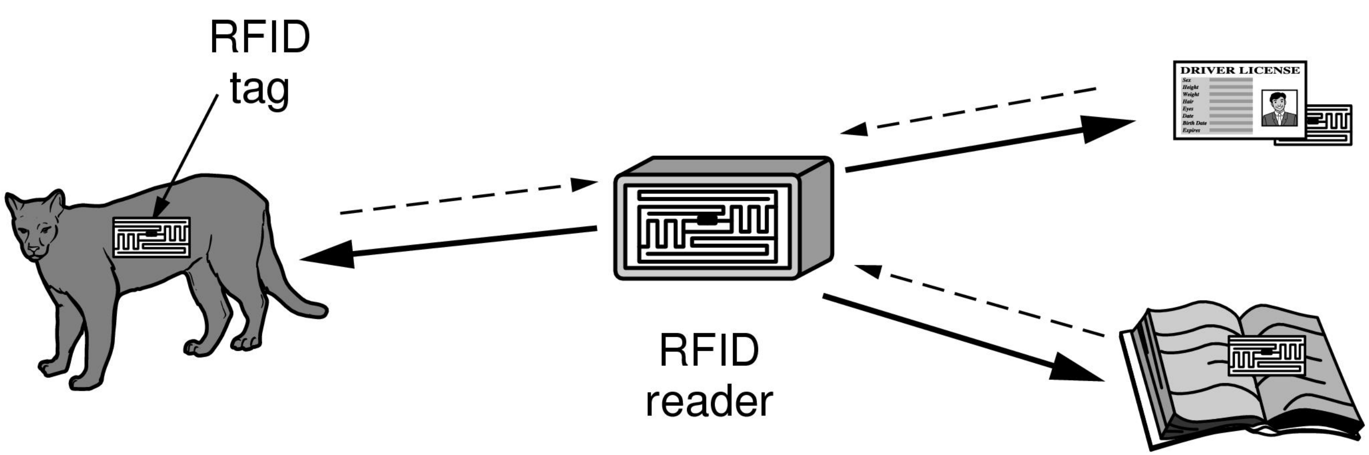
Wireless LAN with 802.11



Wired LAN with switched Ethernet

Applications - RFID

- Radio-Frequency Identification
  - Tags (stickers with not even a battery) are placed on objects
  - Readers send signals that the tags reflect to communicate



Design Issues for the Layers

- Each layer solves a particular problem but must include mechanisms to address a set of recurring design issues

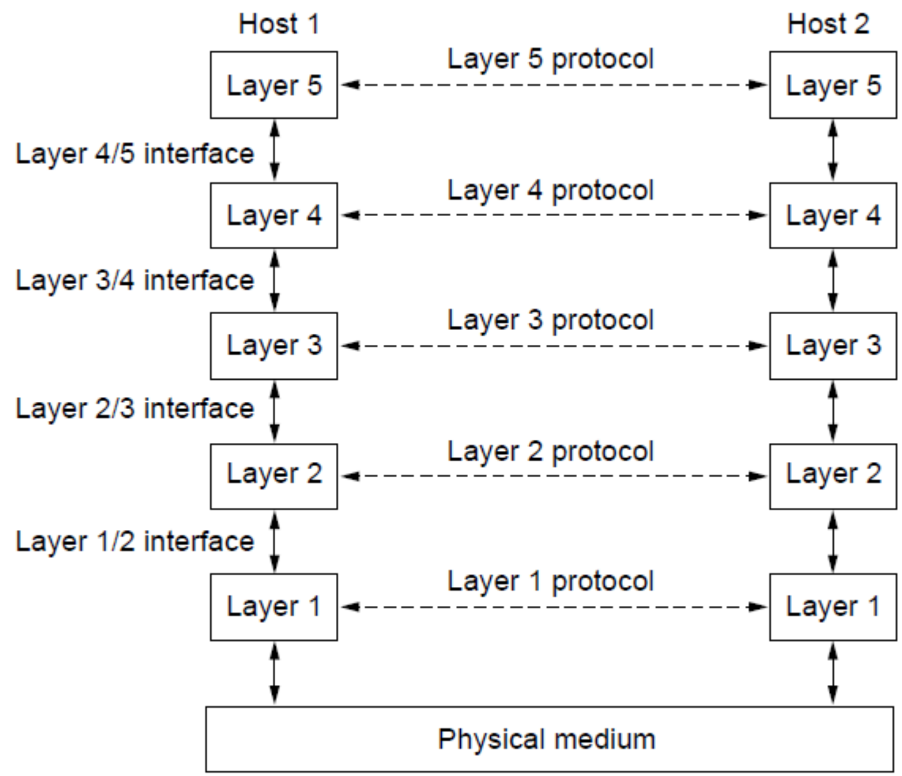
Issue	Example mechanisms at different layers
Reliability despite failures	Codes for error detection/correction (§3.2, 3.3) Routing around failures (§5.2)
Network growth and evolution	Addressing (§5.6) and naming (§7.1) Protocol layering (§1.3)
Allocation of resources like bandwidth	Multiple access (§4.2) Congestion control (§5.3, 6.3)
Security against various threats	Confidentiality of messages (§8.2, 8.6) Authentication of communicating parties (§8.7)

Protocol Layers

- Protocol layering is the main structuring method used to divide up network functionality.
- Basically, **protocol** is an agreement between the communicating parties on how communications is proceed.
- The entities comprising the corresponding layers on different machines are called **peers**.
- Between each pair of adjacent layers is an **interface**.

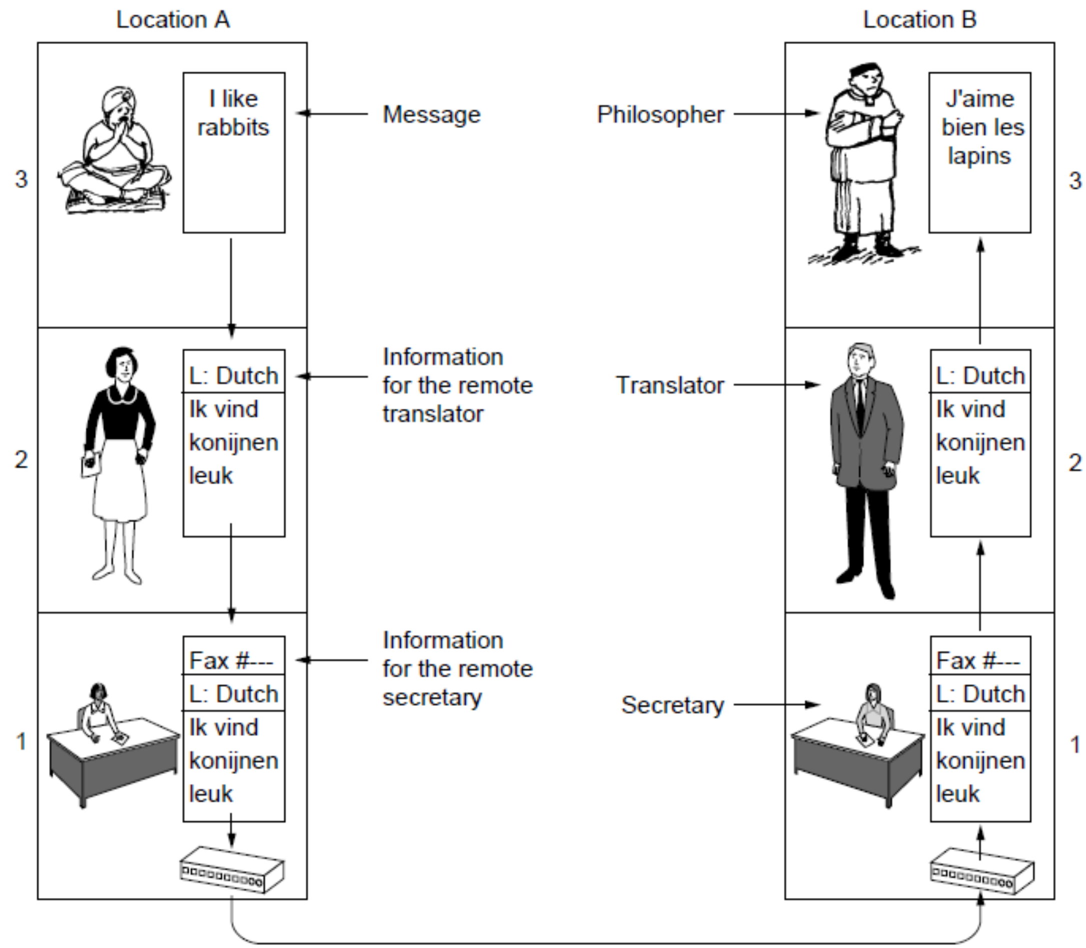
Protocol Layers

- Each protocol instance talks virtually to its <u>peer</u>
- Each layer communicates only by using the one below
- Lower layer <u>services</u> are accessed by an <u>interface</u>
- At bottom, messages are carried by the medium



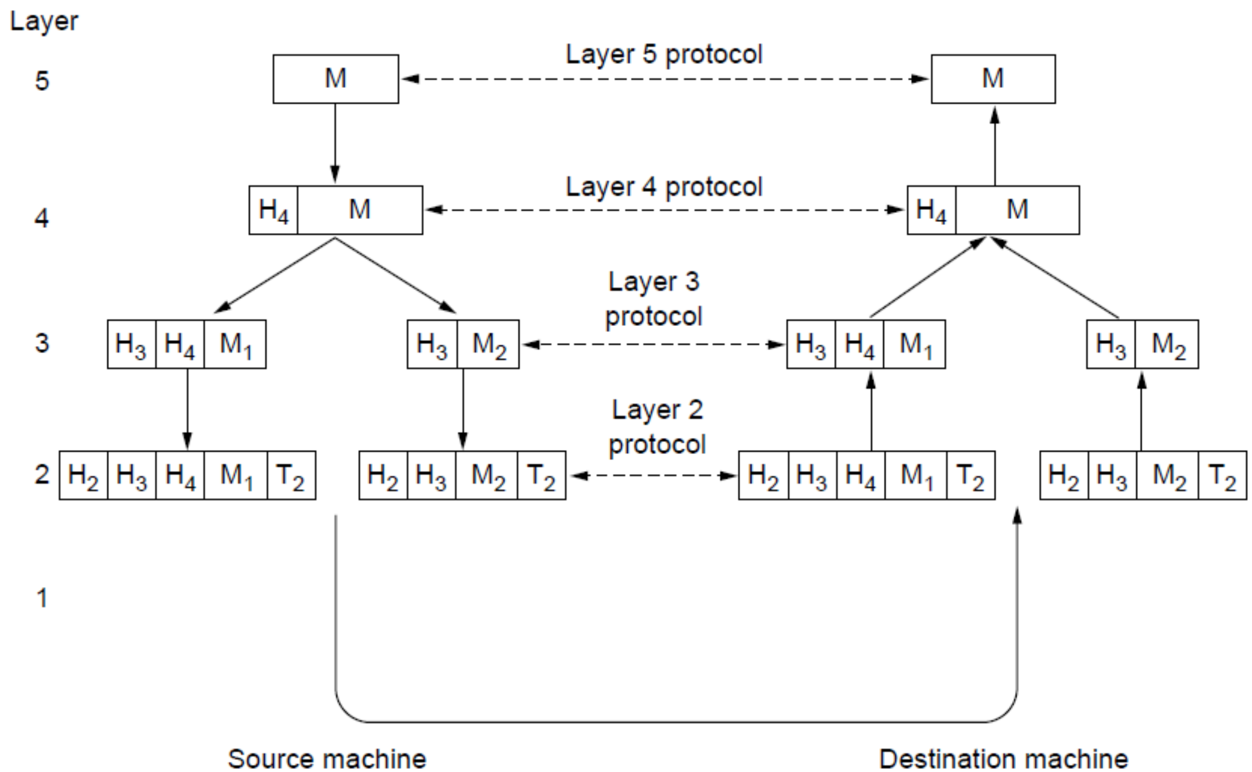
Protocol Layers

- Example: the philosopher-translator-secretary architecture
- Each protocol at different layers serves a different purpose



Protocol Layers

- Each lower layer adds its own <u>header</u> (with control information) to the message to transmit and removes it on receive
- Layers may also split and join messages, etc.



Connection-Oriented vs. Connectionless

- Service provided by a layer may be kinds of either:
  - Connection-oriented, must be set up for ongoing use (and torn down after use), e.g., phone call
  - Connectionless, messages are handled separately, e.g., postal delivery

	Service	Example
Connection-oriented	Reliable message stream	Sequence of pages
	Reliable byte stream	Movie download
	Unreliable connection	Voice over IP
Connection-less	Unreliable datagram	Electronic junk mail□
	Acknowledged datagram	Text messaging
	Request-reply	Database query

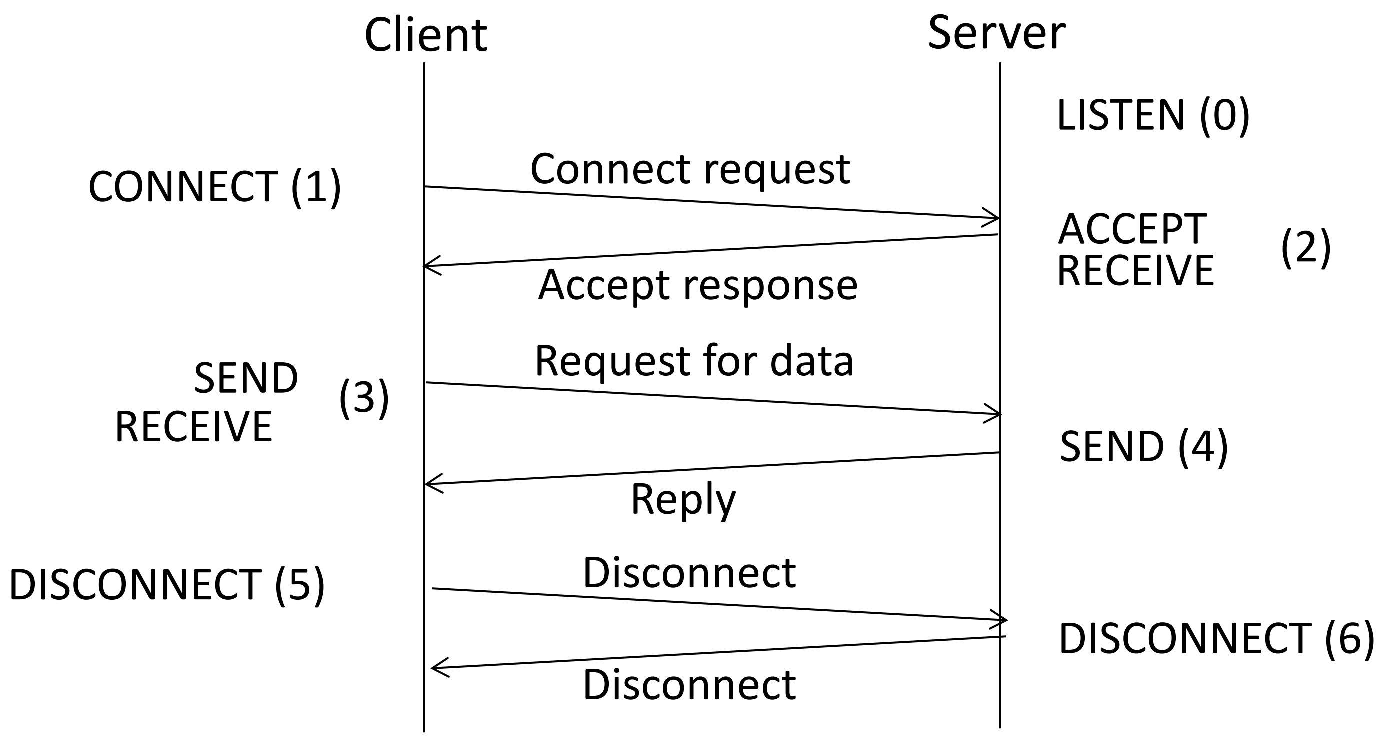
Service Primitives

- A service is provided to the layer above as primitives
- Hypothetical example of service primitives that may provide a reliable byte stream (connection-oriented) service:

Primitive	Meaning
LISTEN	Block waiting for an incoming connection
CONNECT	Establish a connection with a waiting peer
ACCEPT	Accept an incoming connection from a peer
RECEIVE	Block waiting for an incoming message
SEND	Send a message to the peer
DISCONNECT	Terminate a connection

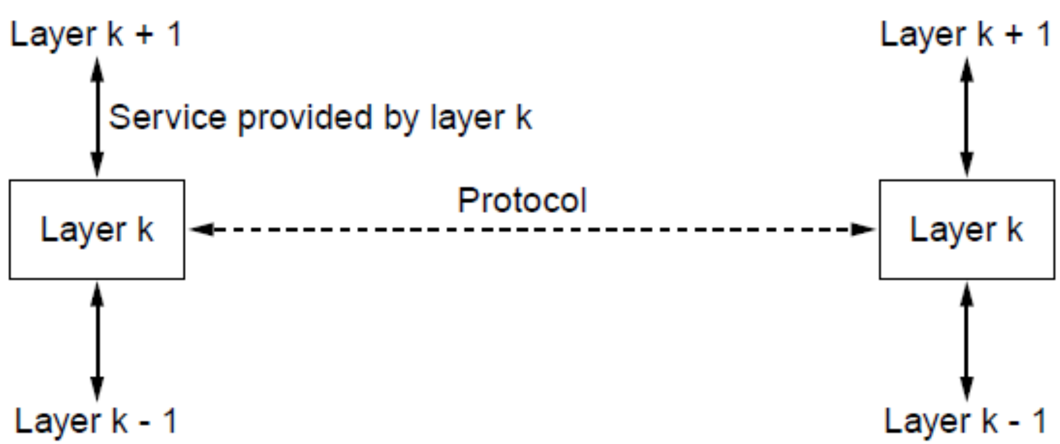
Service Primitives

- Hypothetical example of how these primitives may be used for a client-server interaction



Services and Protocols

- A **service** is a set of primitives (operations) that a layer provides to the layer above it.
- A **protocol** is a set of rules governing the format and meaning of the packets, or messages, that are exchanged by the peer entities within a layer.
- A layer provides a <u>service</u> to the one above [vertical]
- A layer talks to its peer using a <u>protocol</u> [horizontal]



Design principles of OSI layer model

- ISO-**OSI** (The Open Systems Interconnection model)
- A layer should be created where a **different** abstraction is needed.
- Each layer should perform a **well-defined** function
- The function of each layer should be chosen with an eye toward defining internationally standardized protocols
- The layer boundaries should be chosen to **minimize the information flow across the interfaces**
- The number of layers should be **large enough** that distinct functions need not be thrown together in the same layer out of necessity and **small enough** that a architecture does not become unwieldly.

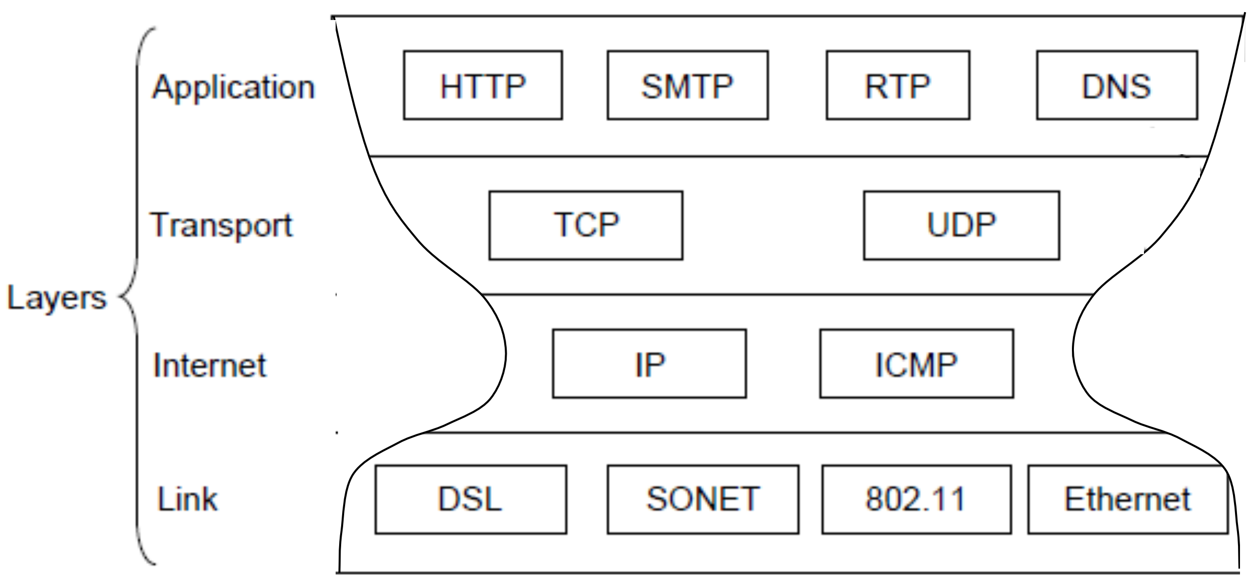
OSI Reference Model

- A principled, international standard, seven layer model to connect different systems

7	Application	– Provides functions needed by users
6	Presentation	– Converts different representations
5	Session	– Manages task dialogs
4	Transport	– Provides end-to-end delivery
3	Network	– Sends packets over multiple links
2	Data link	– Sends frames of information
1	Physical	– Sends bits as signals

TCP/IP Reference Model

- A four layer model derived from experimentation; omits some OSI layers and uses the IP as the network layer.



Critique of OSI & TCP/IP

- OSI:
  - Very influential model with clear concepts
  - Models, protocols and adoption all bogged down by politics and complexity
- TCP/IP:
  - Very successful protocols that worked well and thrived
  - Weak model derived after the fact from protocols

Model Used in this Book

- It is based on the TCP/IP model but we call out the physical layer and look beyond Internet protocols.

5	Application
4	Transport
3	Network
2	Link
1	Physical

Five Layers

- Physical Layer
  - how to transmit bits across different kinds of media as electrical (or other analogue) signals
- Link Layer
  - how to send finite-length messages between directly connected computers with specified levels of reliability.
  - Protocol examples: Ethernet and 802.11.

Five Layers

- Network Layer
  - how to combine multiple links into networks, and networks of networks, into internetworks so that we can send packets between distant computers.
  - E.g.: finding the path along which to send the packets
  - Main protocol: IP(Internet Protocol)
- Transport Layer
  - strengthens the delivery guarantees of the Network layer, usually with increased reliability, and provide delivery abstractions, such as a reliable byte stream, that match the needs of different applications. E.g. TCP, UDP

Five Layers

- Application Layer
  - contains programs that make use of the network.
  - Many, but not all, networked applications have user interfaces, such as a Web browser (HTTP protocol is an example in this case).
  - There are also important support programs in the application layer, such as the DNS, that are used by many applications.

Network Standardization

- Some standard bodies

Body	Area	Examples
ITU	Telecommunications	G.992, ADSL H.264, MPEG4
IEEE	Communications	802.3, Ethernet 802.11, WiFi
IETF	Internet	RFC 2616, HTTP/1.1 RFC 1034/1035, DNS
W3C	Web	HTML5 standard CSS standard