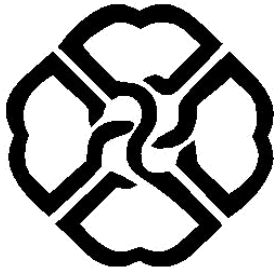


Department of Computer and IT Engineering
University of Kurdistan

Advanced Computer Networks (Graduate level)

Introduction

By: Dr. Alireza Abdollahpouri

Course Info

Instructor

Dr. Alireza Abdollahpouri

Email: abdollahpouri@gmail.com

Office: Room 219

Course Web Page

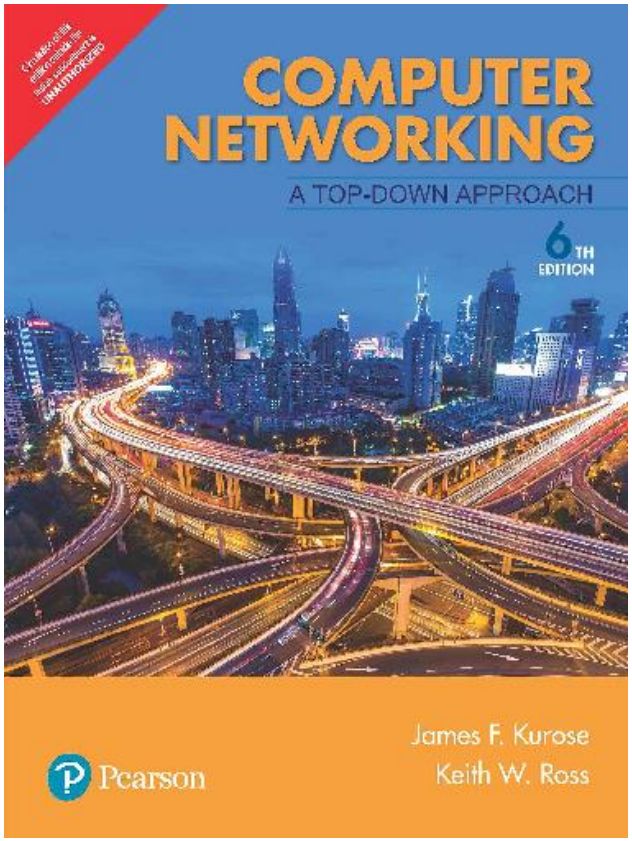
http://eng.uok.ac.ir/abdollahpouri/Advanced_Net.html

Grading Policy

Homework and Projects	**40%**
Final	**50%**
Class participation	**10%**

Course Info. (Textbooks)

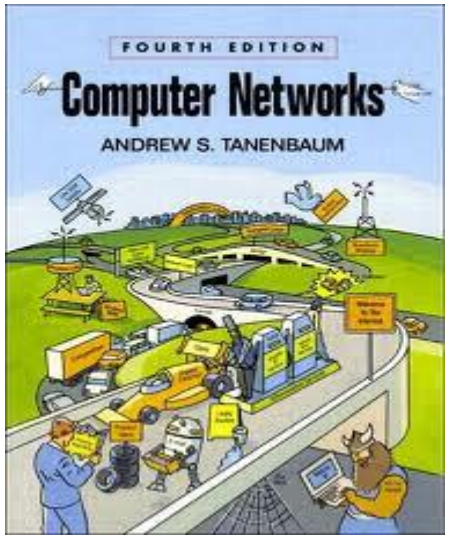
- Computer Networking: A Top-Down Approach, 6th Edition, James F. Kurose and Keith W. Ross, Pearson, 2013,

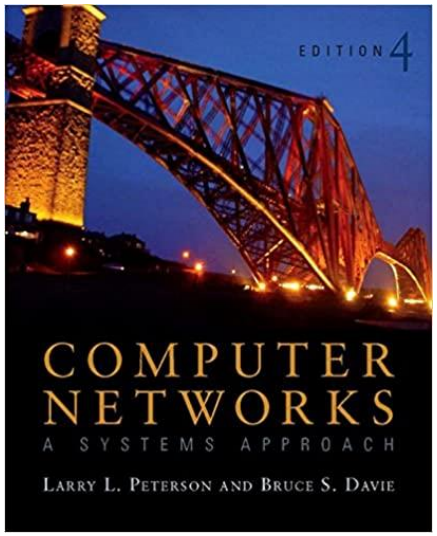


Course Info

Other useful books

- A. Tanenbaum, Computer Networks
- S. Keshav, An Engineering Approach to Computer Networking
- Peterson and Davie, Computer Networks: A Systems Approach, 4th Edition
- Abdollahpouri and Naseri, “Principles of Computer Networks” (in Persian)





Some Ground Rules

- Let's make this educational and enjoyable.
- It's a big size class, I enjoy questions and ideas from the class.
  - Ask questions and raise points.
  - Listen to other people's questions.
  - Be here.
  - Be here on time.

Course Info

Topics covered

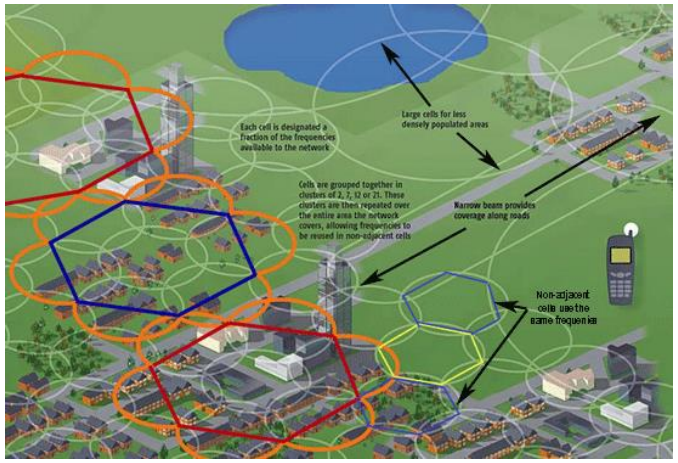
- Introduction: Overview, Network types, Protocol Layers
- Network Applications: Network applications and protocols, HTTP, DNS, Socket programming
- Transport Layer: Transport layer services and protocols, UDP, TCP, Flow and congestion control
- Network Layer: Routing algorithms, Forwarding and addressing in the Internet, IP, Router design
- Link Layer and Local Area Networks: Multiple access protocols, Error detection, Ethernet, Bridges

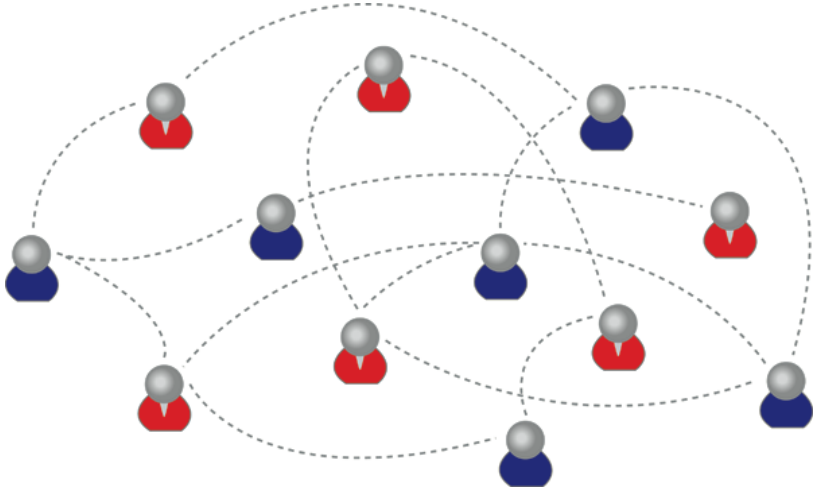
What is a Network?

There are many types of networks!

- **Transportation Networks**
  - Transport goods using trucks, ships, airplanes, …
- **Postal Services**
  - Delivering letters, parcels, etc.
- **Broadcast and cable TV networks**
- **Telephone networks**
- **Internet**
- **“Social/Human networks”**

Many types of networks





What is a Comm. Network?

A communications network is a network of links and nodes arranged so that messages may be passed from one part of the network to another

- What are nodes and links?
  - Wired or wireless
  - Computers and network devices
- What is a message?
  - **Information**

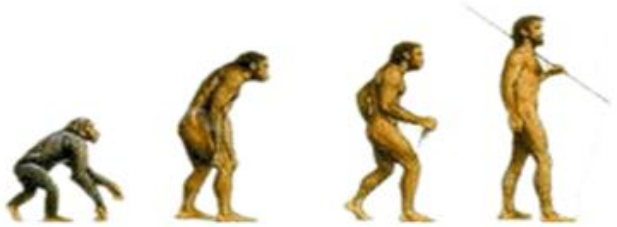
Another definition

A network can be defined as two or more computers (or other devices such as cell phones) connected together (via a communication media) in such a way that they can share resources.

Communication media
- **Wired (e.g., copper, fiber optic))**
- **Wireless (e.g., Microwave, Infrared)**

Resource
- **Software (e.g., file)**
- **Hardware (e.g., printer)**

History of communication



History of communication

- 2400 BC: Courier networks in Egypt
- 550 BC: Postal service invented in Persia

Problems:

- **Speed**
- **Reliability**
- **Security**

Towards Electric Communication

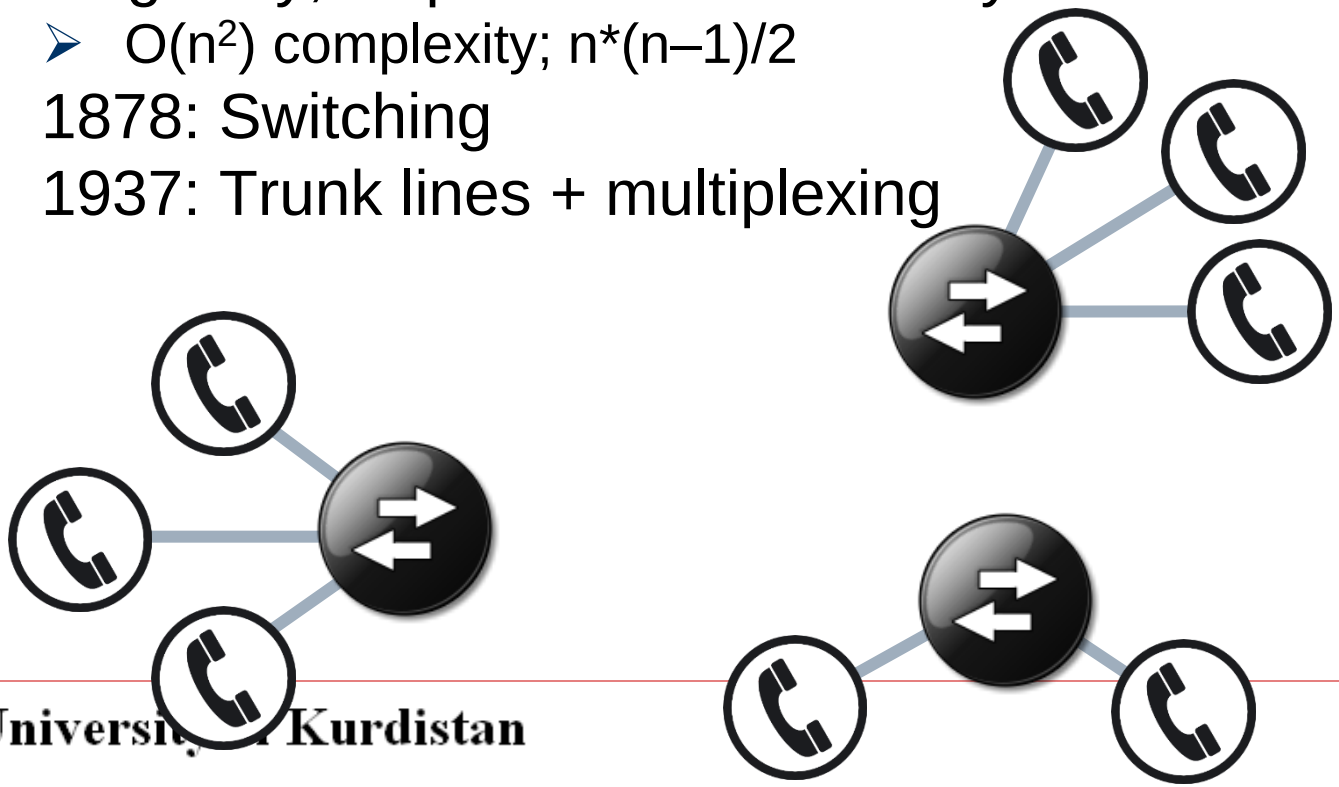
- 1837: Telegraph invented by Samuel Morse
  - Distance: 10 miles
  - Speed: 10 words per minute
  - In use until 1985!
- Key challenge: how to encode information?
  - Originally used unary encoding

    A • B •• C ••• D •••• E •••••
  - Next generation: binary encoding

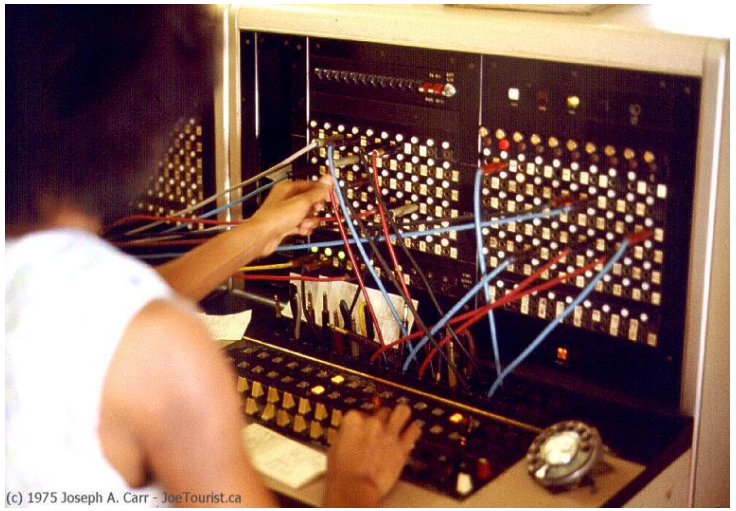
    A •– B –••• C –•–• D –•• E •

Telephony

- 1876 – Alexander Graham Bell invents the telephone
- Key challenge: how to scale the network?
  - Originally, all phones were directly connected
    - $O(n^2)$ complexity; $n*(n–1)/2$
  - 1878: Switching
  - 1937: Trunk lines + multiplexing



Telephony



Advantages

- Easy to use
- Switching mitigates complexity
- Makes cable management tractable

Problems

- Manual switching
- 1918: cross country call took 15 minutes to set up

Growth of the Telephone Network

1881: Twisted pair for local loops

1885: AT&T formed

1892: Automatic telephone switches

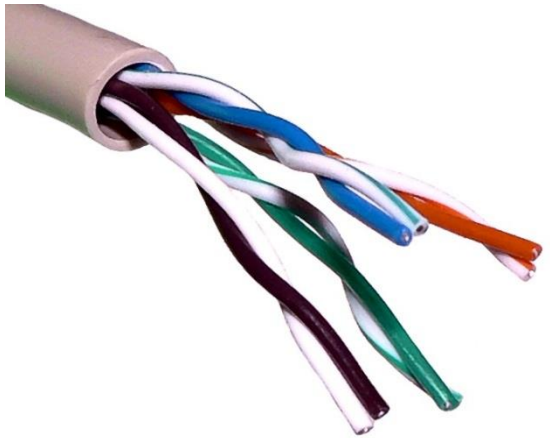
1903: 3 million telephones in the US

1915: First transcontinental cable

1927: First transatlantic cable

1937: first round-the-world call

1946: National numbering plan



at&t

From Humans to Computers

- 1958: First use of a modem
  - Machine to machine communication
  - Analog vs. digital signals
- Many different computer networks
  - Local vs. global
    - LAN, WAN
  - Private vs. public
    - Internet2, NIPRNet
  - General purpose vs. special purpose
    - E.g. credit cards, banks, defense

- **Technology**
  - **Satellite, Copper, Fiber**
  - **Circuit switched, packet switched**

History of the Internet

1961: Kleinrock @ MIT: packet-switched network
1962: Licklider's vision of Galactic Network
1965: Roberts connects computers over phone line
1967: Roberts publishes vision of ARPANET
1969: BBN installs first InterfaceMsgProcessor at UCLA
1970: Network Control Protocol (NCP)
1972: Public demonstration of ARPANET
1972: Kahn @ DARPA advocates Open Architecture
1972: Vint Cerf @ Stanford writes TCP

Bob Kahn and Vint Cerf

More Internet History

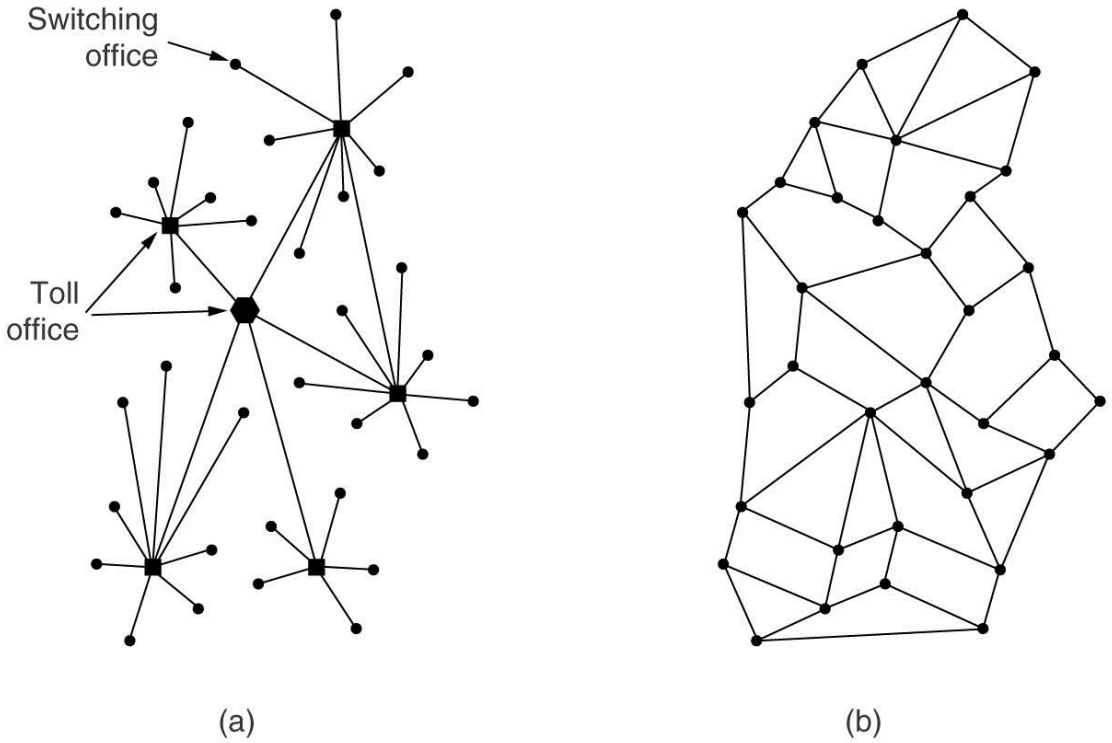
1974: Cerf and Kahn paper on TCP (IP kept separate)
1980: TCP/IP adopted as defense standard
1983: ARPANET and MILNET split
1983: Global NCP to TCP/IP flag day
198x: Internet melts down due to congestion
1986: Van Jacobson saves the Internet (BSD TCP)
1987: NSFNET merges with other networks
1988: Deering and Cheriton propose multicast
199x: QoS rises and falls, ATM rises and falls
1994: NSF backbone dismantled, private backbone
200x: The Internet boom and bust

Internet history

2005-present

- ~750 million hosts
  - Smartphones and tablets
- Aggressive deployment of broadband access
- Increasing ubiquity of high-speed wireless access
- Emergence of online social networks:
  - Facebook: soon one billion users
- Service providers (Google, Microsoft) create their own networks
  - Bypass Internet, providing “instantaneous” access to search, emai, etc.
- E-commerce, universities, enterprises running their services in “cloud” (eg, Amazon EC2)

The idea of packet switching

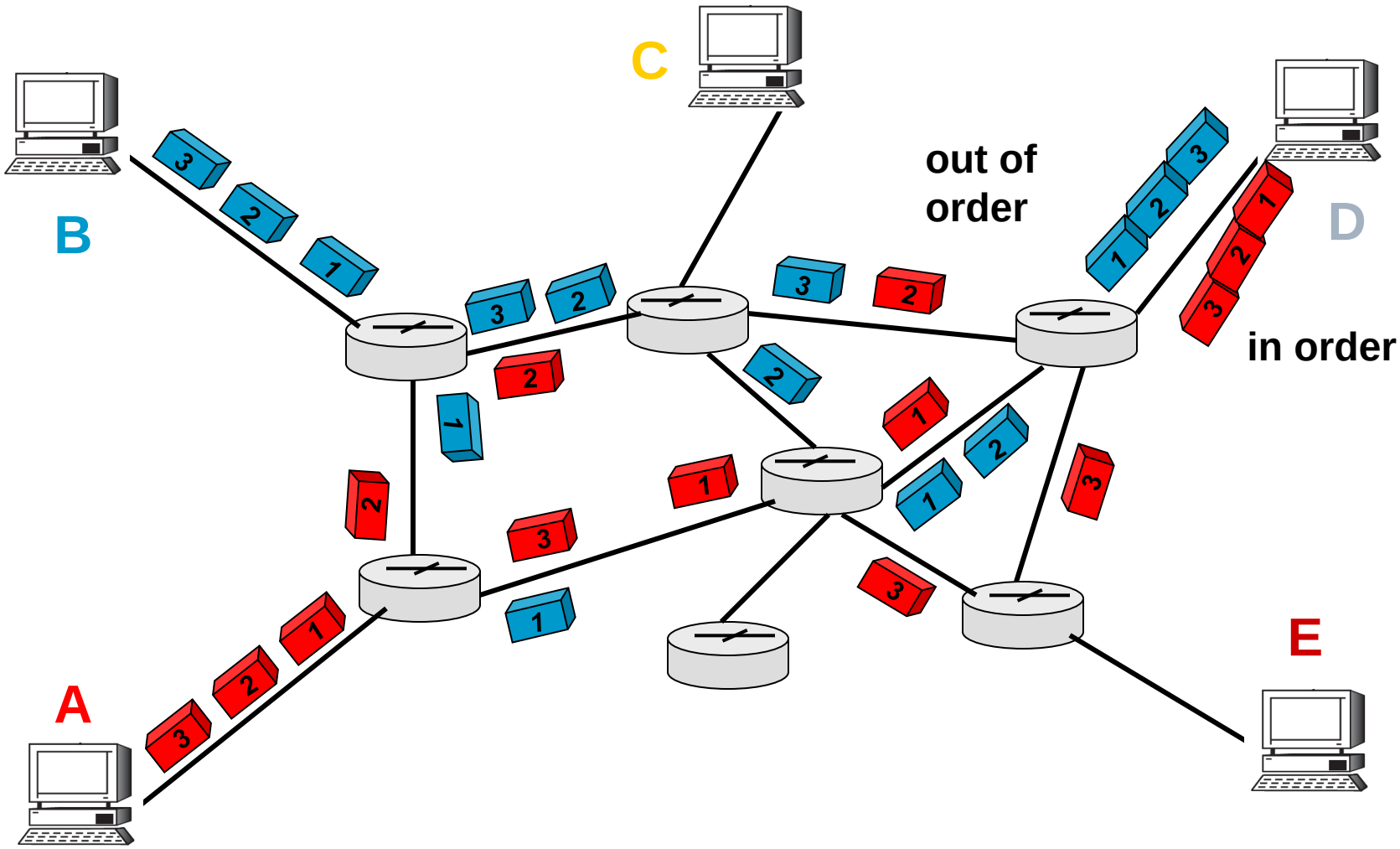


Paul Baran

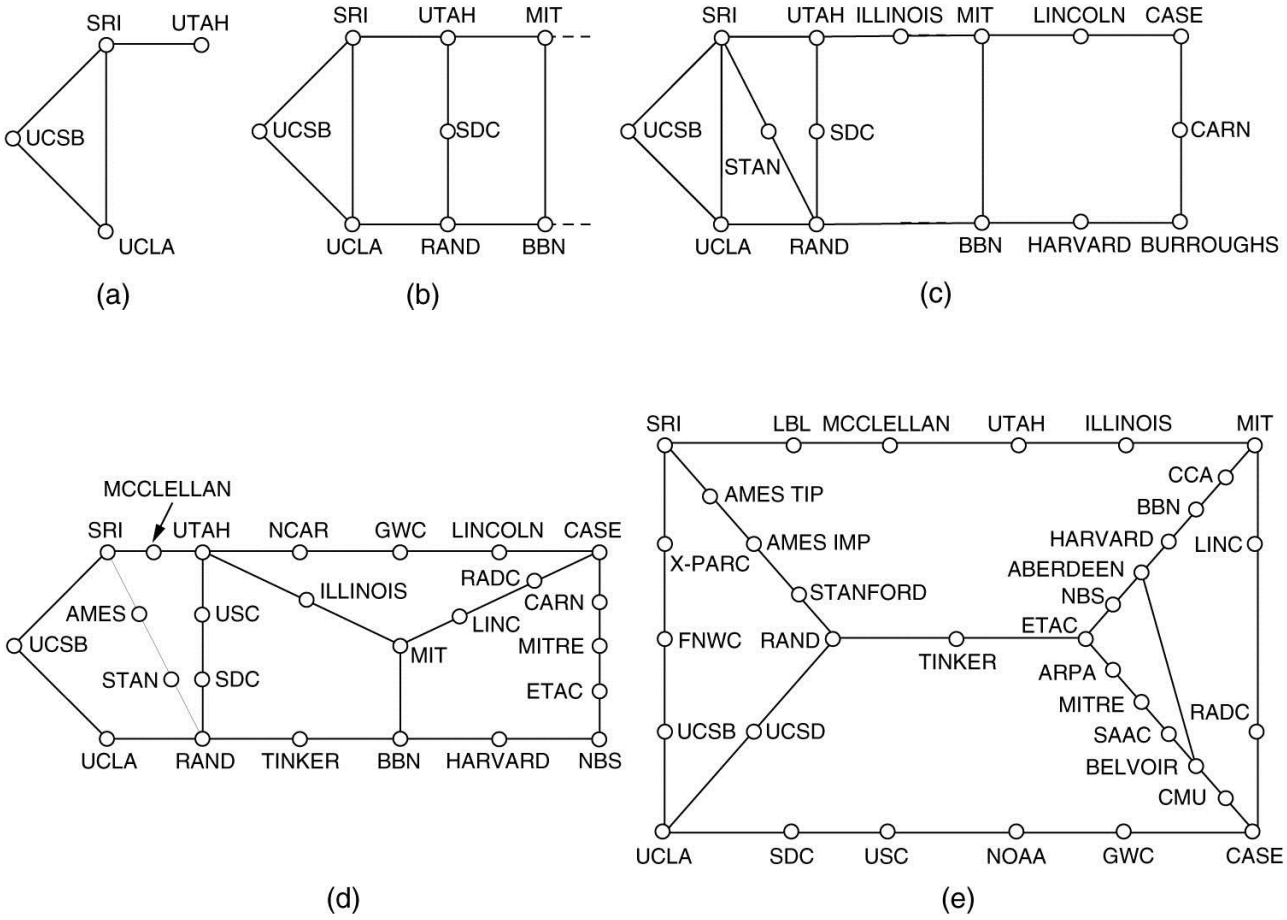
Leonard Kleinrock

From circuit switching to packet switching

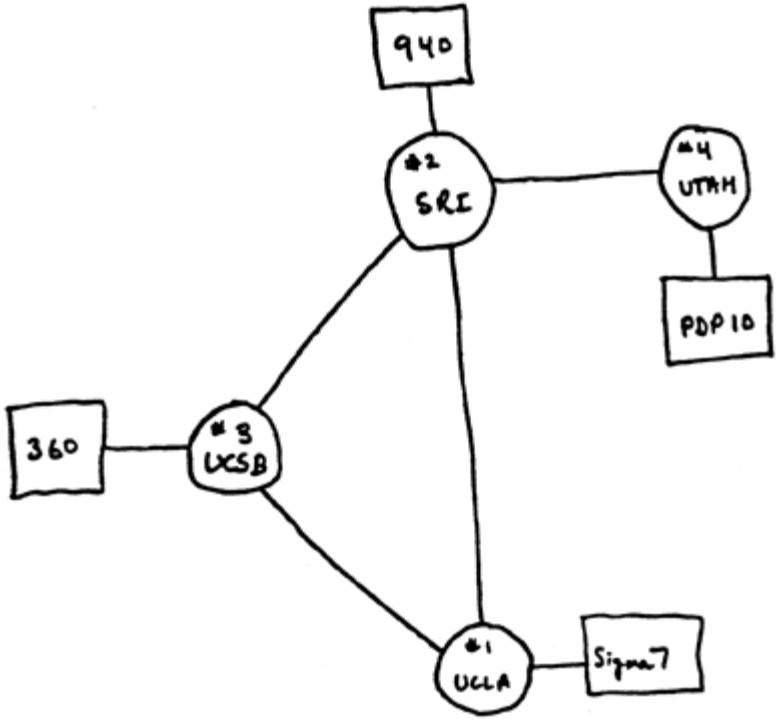
Datagram Packet Switching



Growth of the ARPANET (1969-1972)



4-node ARPANET diagram



IMP

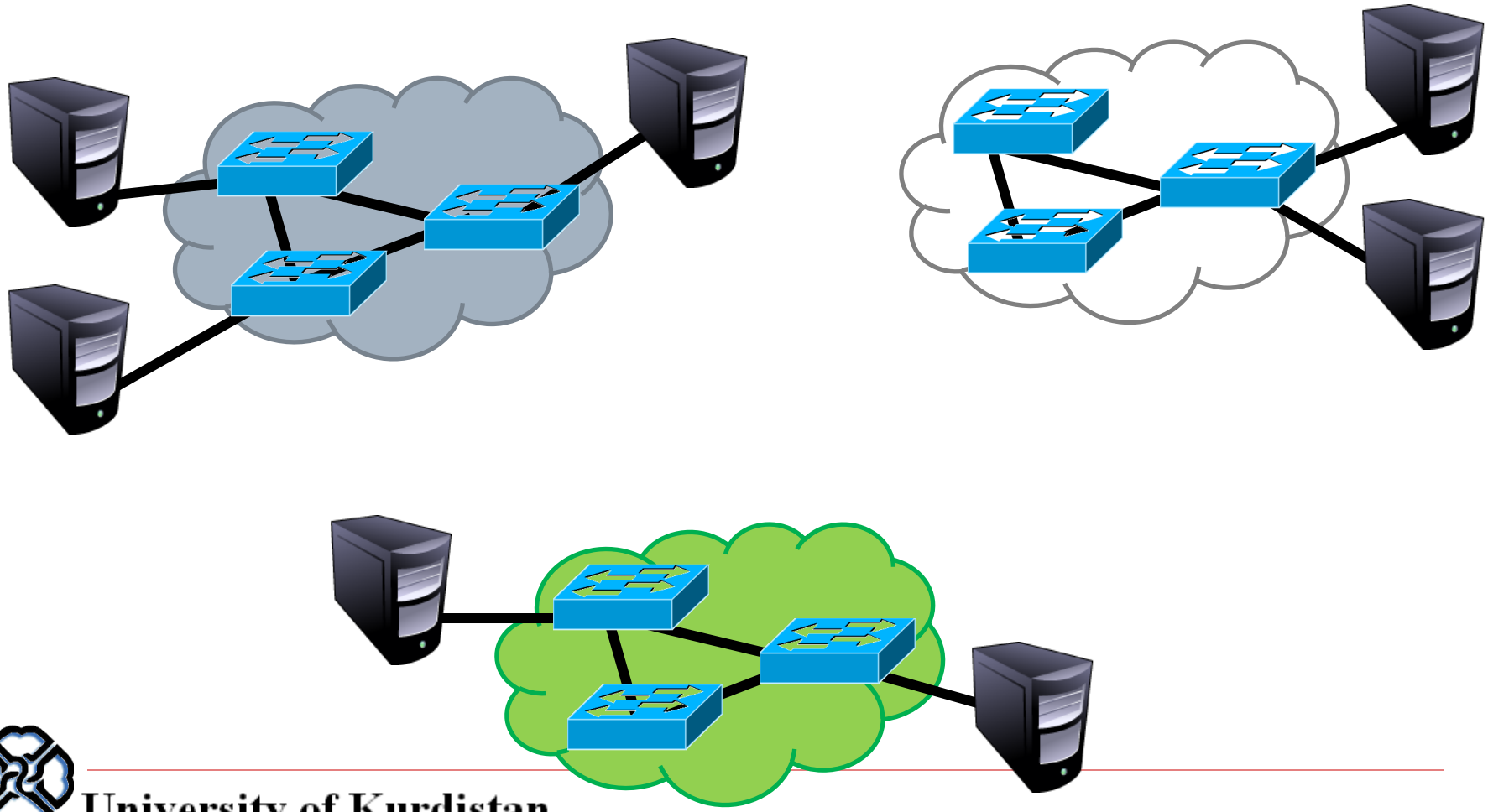
Leonard Kleinrock and the
first **I**nterface **M**essage **P**rocessor

Why Packet Switching?

- Telephone networks are circuit switched
  - Each call reserves resources end-to-end
  - Provides excellent quality of service
- Problems
  - Resource intense (what if the circuit is idle?)
  - Complex network components (per circuit state, security)
- Packet switching
  - No connection state, network is store-and-forward
  - Minimal network assumptions
  - Statistical multiplexing gives high overall utilization

Growing Pains

Problem: early networks used incompatible protocols

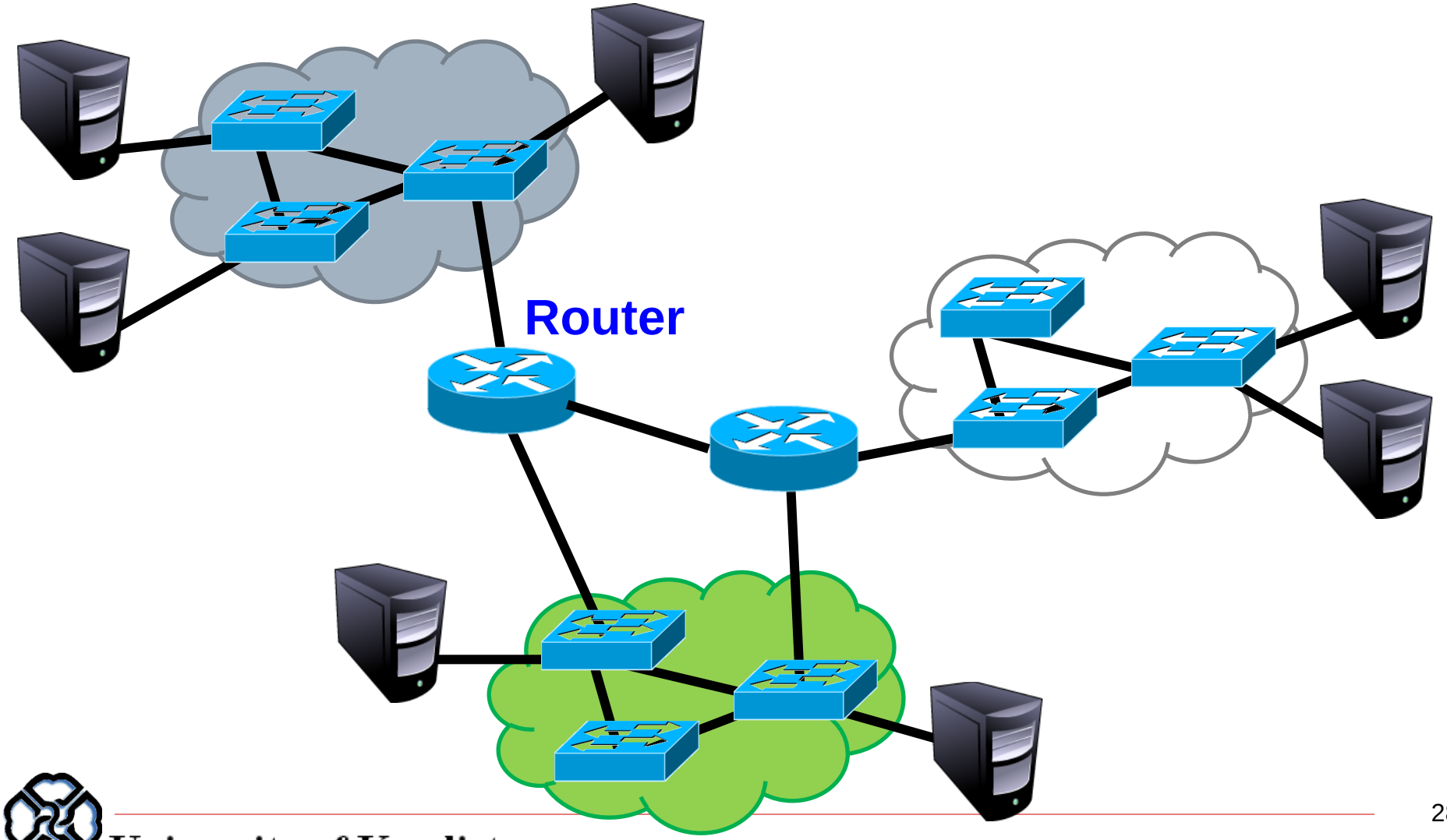


Kahn's Ground Rules

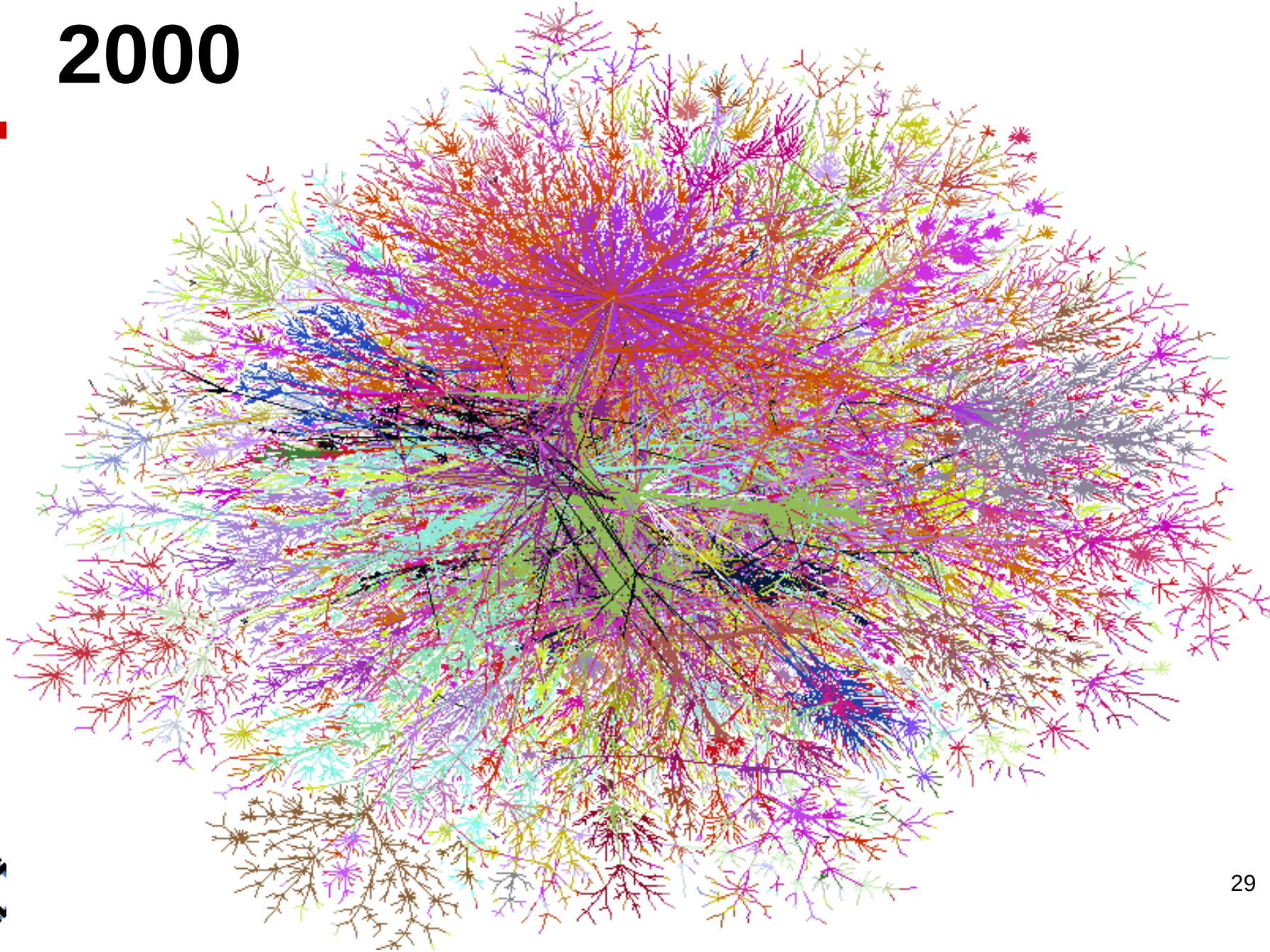
1. Each network is independent, cannot be forced to change
2. Best-effort communication (i.e. no guarantees)
3. Routers connect networks
4. No global control

- Principals behind the development of IP
- Led to the Internet as we know it
- Internet is still structured as independent networks

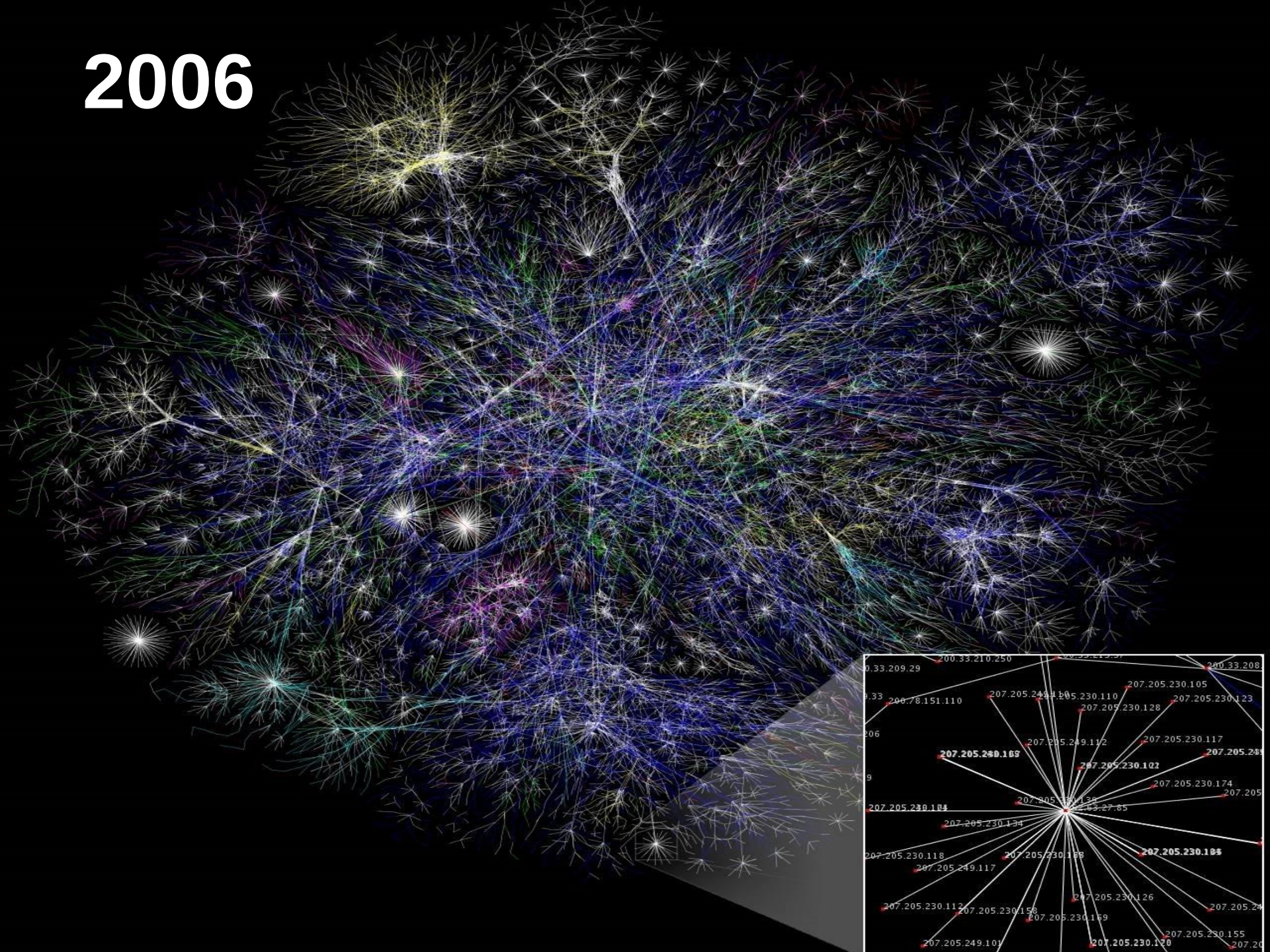
The Birth of Routing

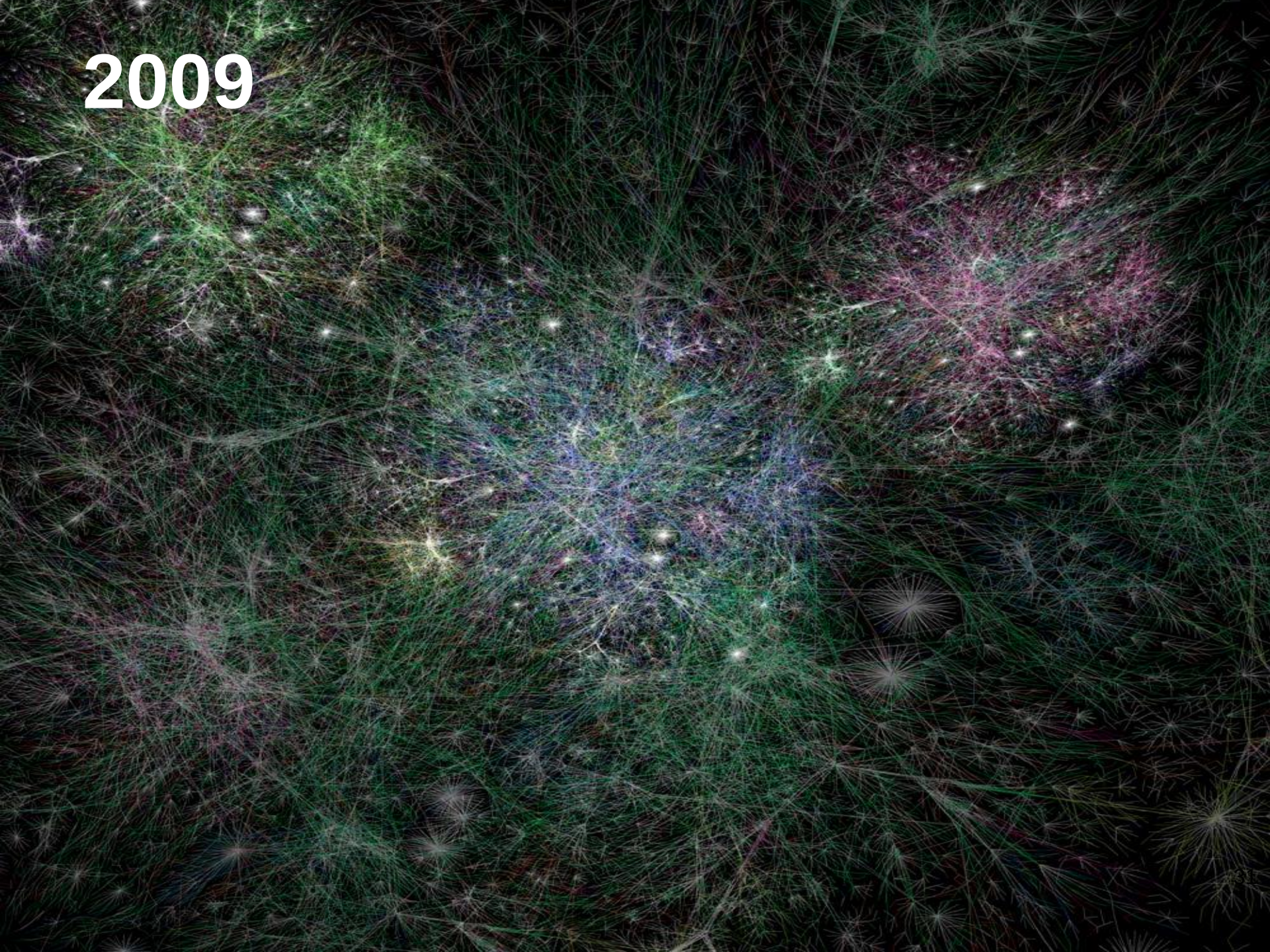


2000



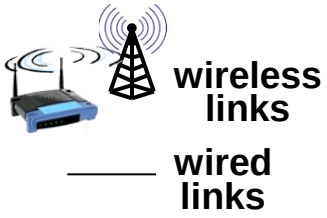
2006


200.33.210.250
200.33.208
207.205.230.105
207.205.230.110
207.205.230.123
207.205.230.128
207.205.249.112
207.205.230.117
207.205.230.174
207.205.230.134
207.205.230.118
207.205.249.117
207.205.230.126
207.205.230.155
207.205.249.101
207.205.230.120


2009

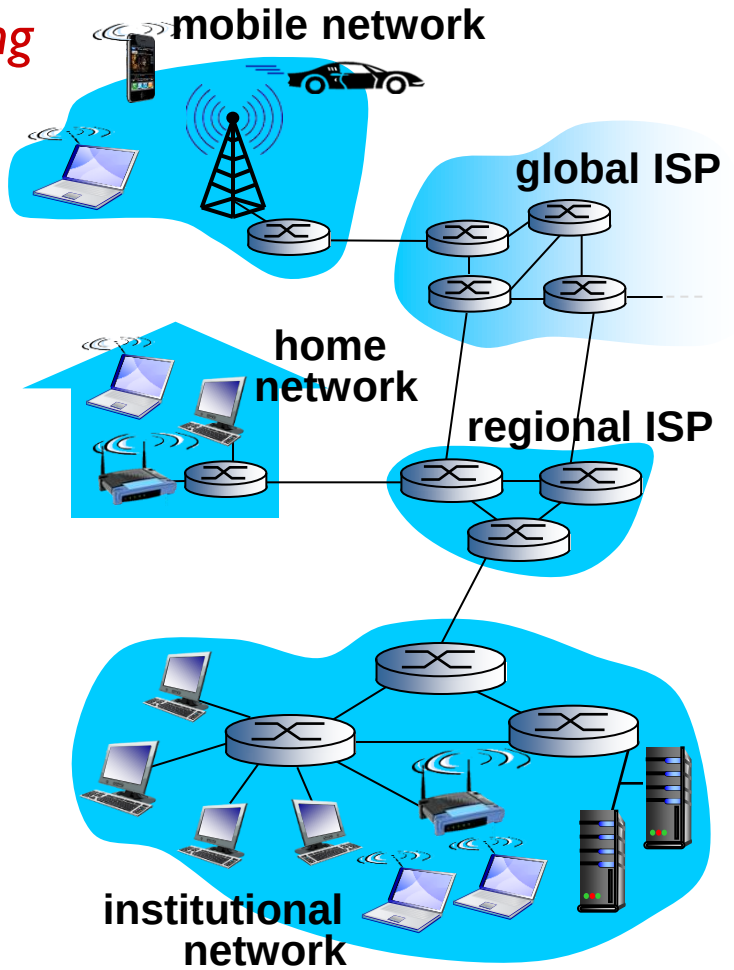
What's the Internet: "nuts and bolts" view

- *millions of connected computing devices:*
  - hosts = end systems
  - running network apps



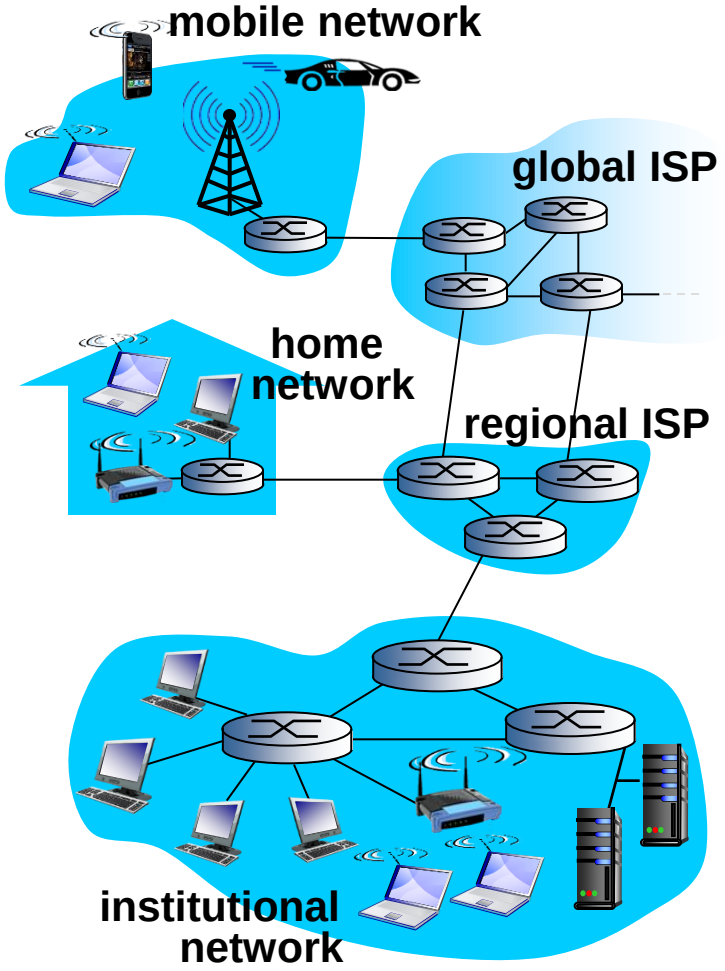
- *communication links*
  - fiber, copper, radio, satellite
  - transmission rate: *bandwidth*

- *Packet switches:* forward packets (chunks of data)
  - *routers* and *switches*



What's the Internet: a service view

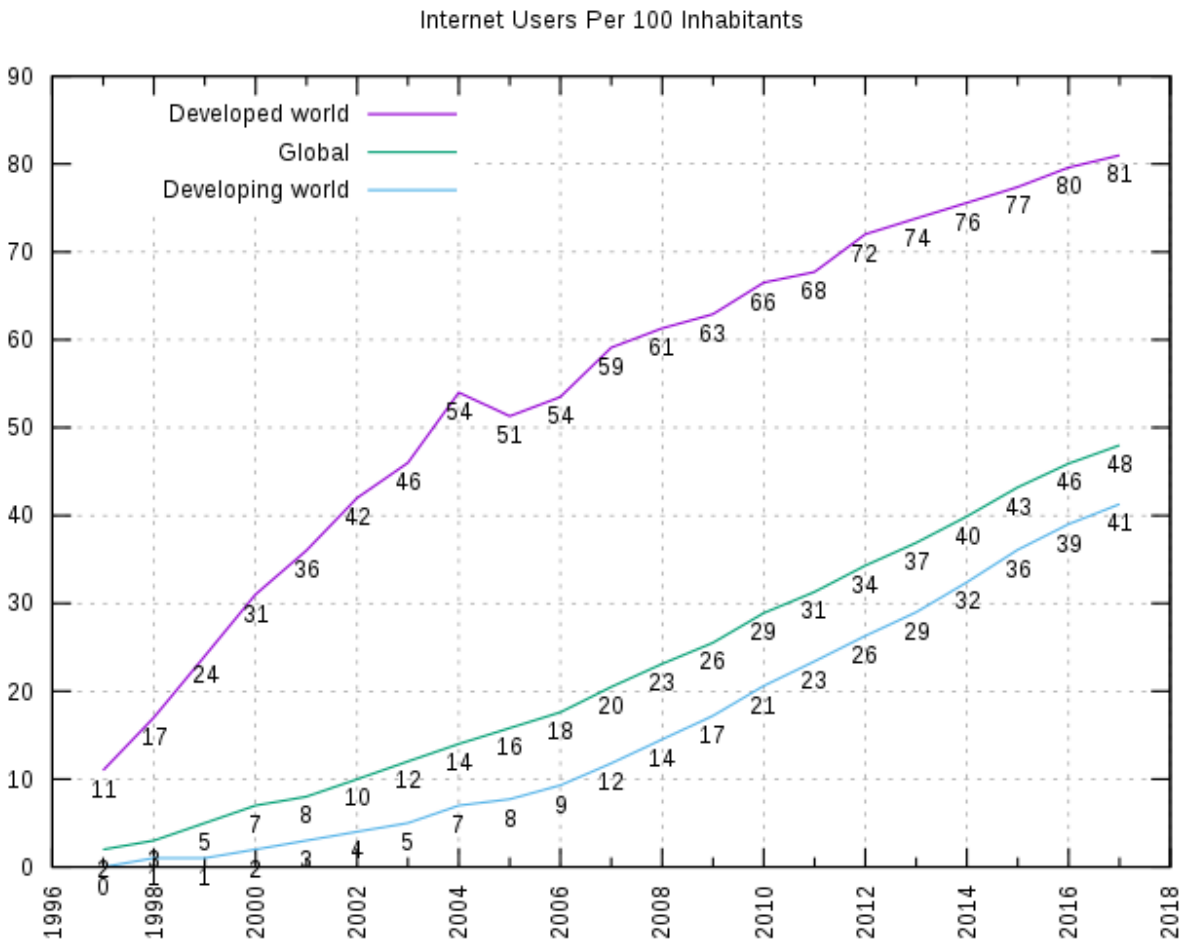
- *Infrastructure that provides services to applications:*
  - Web, VoIP, email, games, e-commerce, social nets, …
- *provides programming interface to apps*
  - hooks that allow sending and receiving app programs to "connect" to Internet
  - provides service options, analogous to postal service



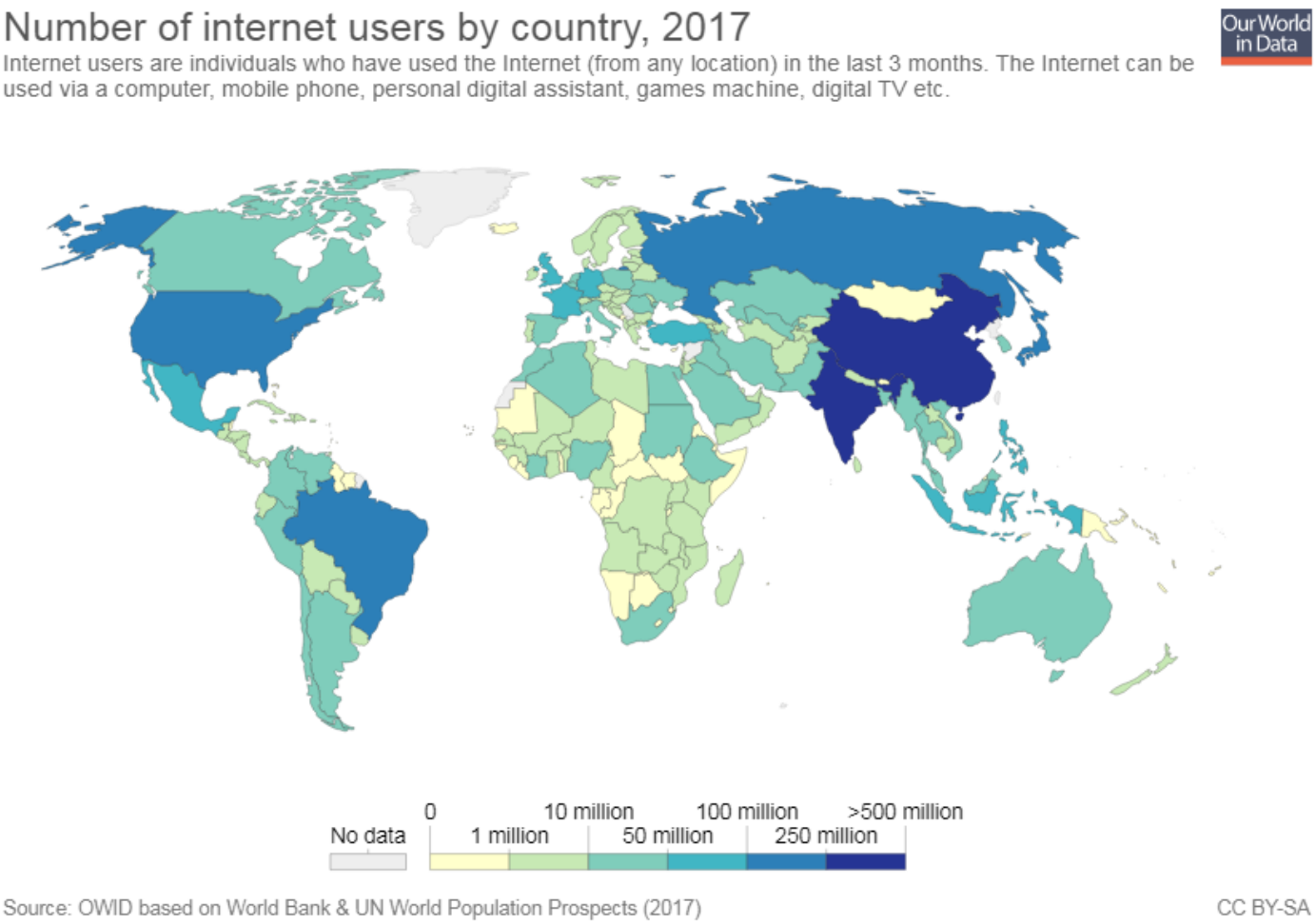
Internet Applications Over Time

- 1972: Email
- 1973: Telnet – remote access to computing
- 1982: DNS – "phonebook" of the Internet
- 1985: FTP – remote file access
- 1989: NFS – remote file systems
- 1991: The World Wide Web (WWW) goes public
- 1995: SSH – secure remote shell access
- 1995-1997: Instant messaging (ICQ, AIM)
- 1998: Google
- 1999: Napster, birth of P2P
- 2001: Bittorrent
- 2004: Facebook
- 2005: YouTube
- 2007: The iPhone

Global Usage of Internet



Number of Internet users in 2017



Applications of Computer Networks

Business Applications

- Resource sharing (Programs, equipment, information)
- Communication medium (E-mail, Video-conference)
- E-commerce (Business to Business, Business to Customer)

Home Applications

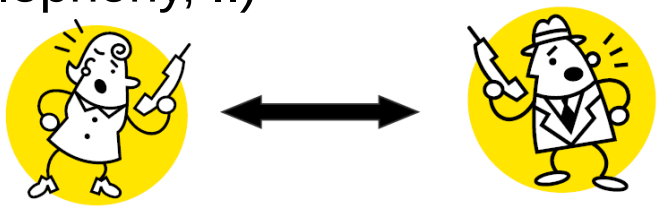
- Access to remote information
- Person to person communication (E-mail, chat, peer to peer..)
- Interactive Entertainment
- E-commerce

Mobile Applications

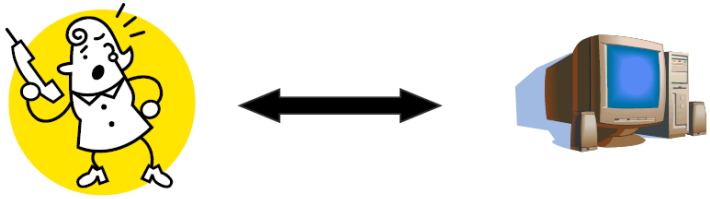
- Portable office (access to internet, E-mail, information, etc)
- Access to information on the move
- Navigation and maps

Evolution of communication

Step 1:
Person to person (Direct communication, telephony, ..)

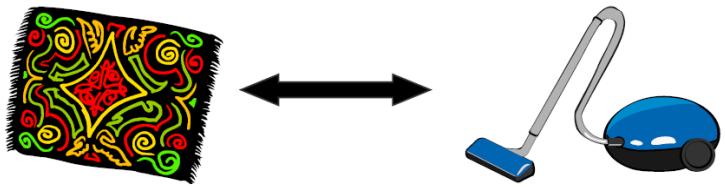


Step 2:
Person to machine (Fax, PC usage, ...)



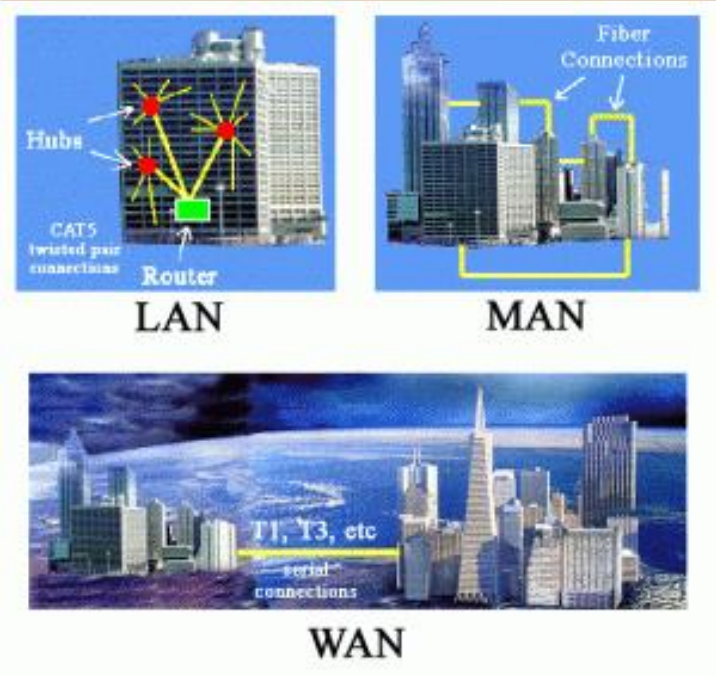
Step 3:
Machine to machine (Grid Computing)

Step 4:
Things to things (Internet of Things)



Network categorization (based on scale)

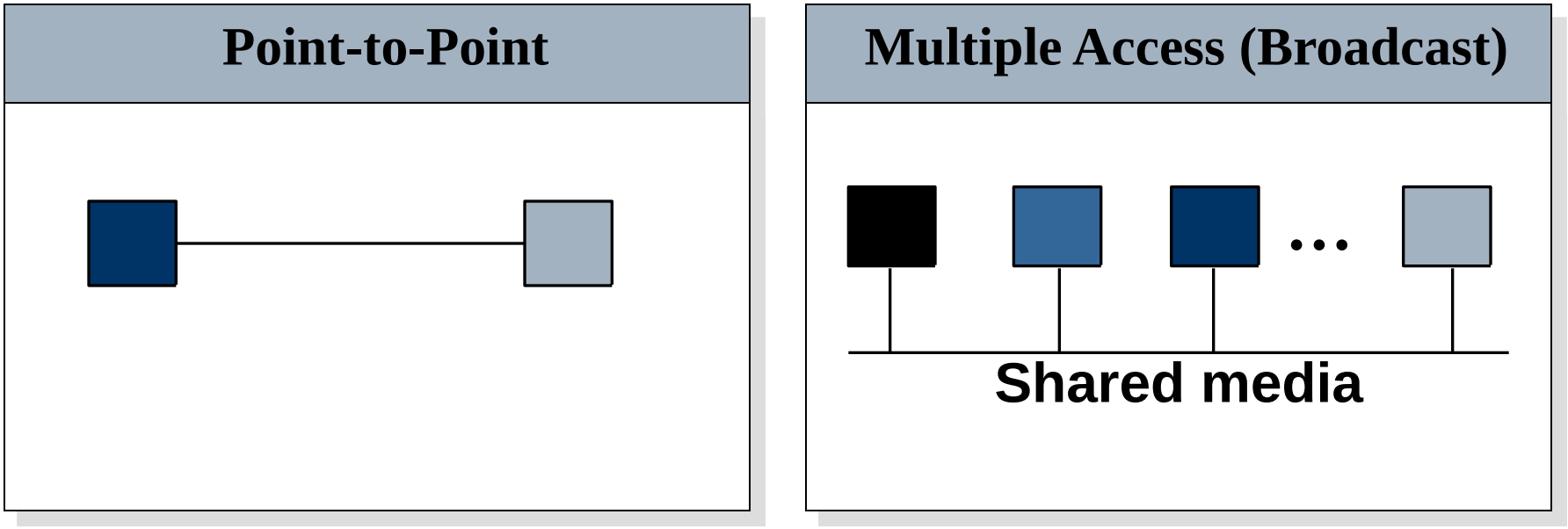
- PAN (Personal Area Network)
- LAN (Local Area Network)
- MAN (Metropolitan Area Network)
- WAN (Wide Area Network)
- Internet

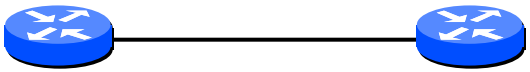


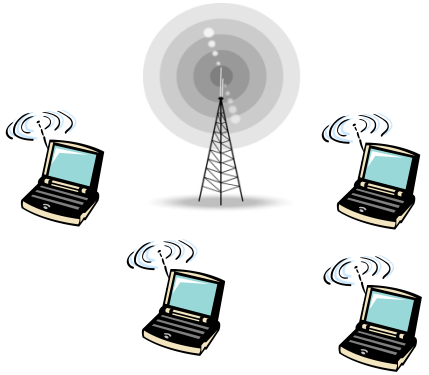
Network categorization (based on scale)

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

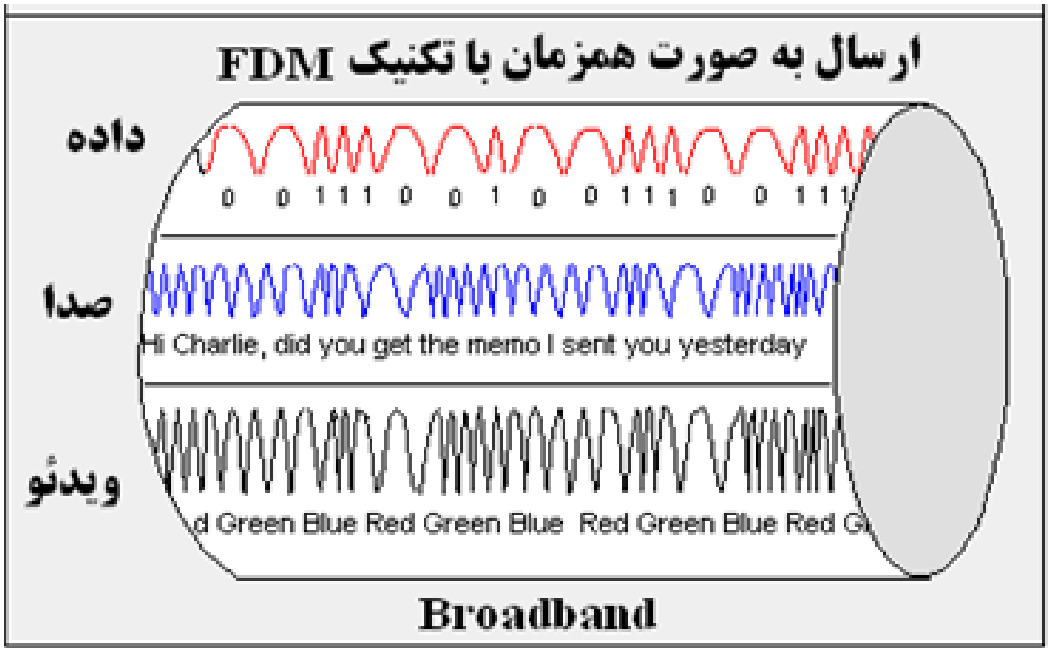
Types of Links

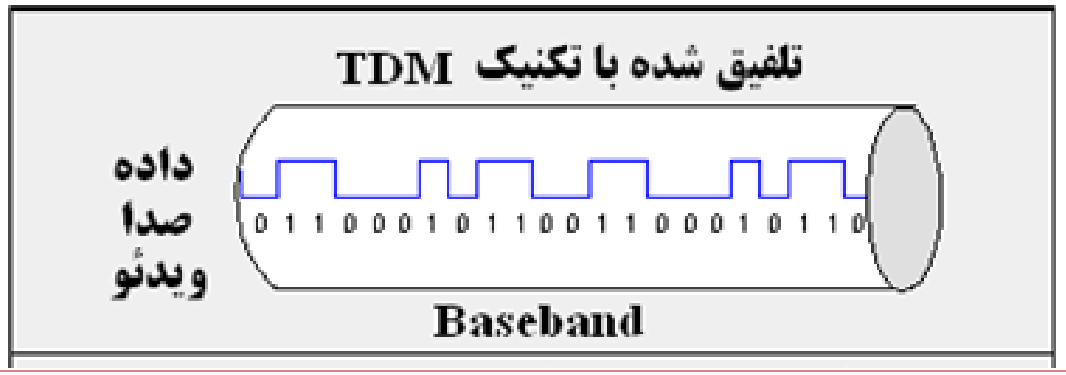




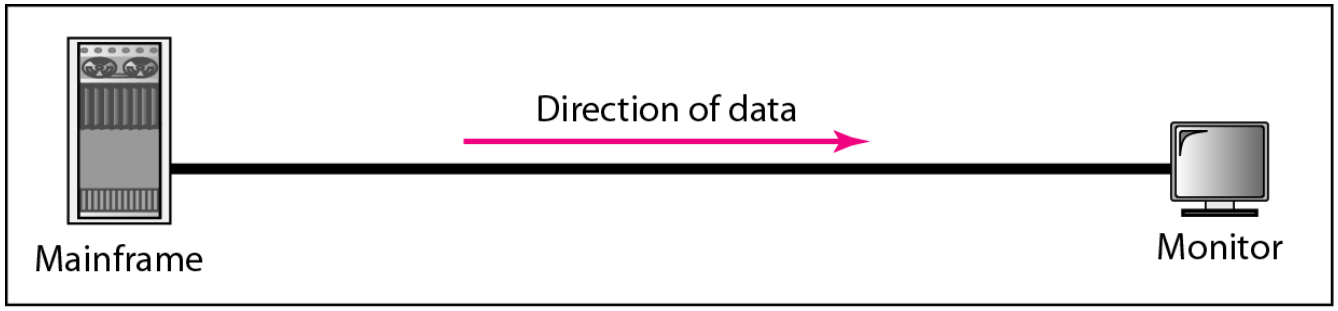


Baseband vs. Broadband

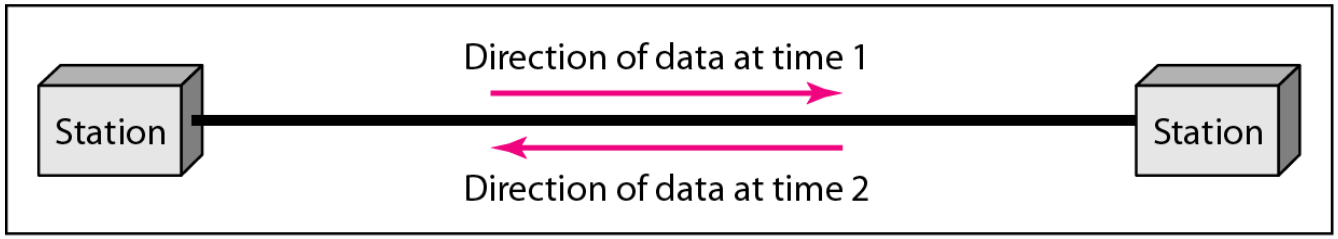




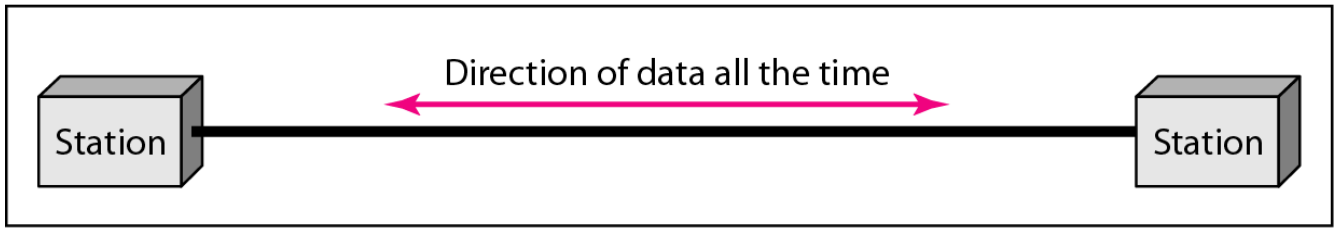
Data flow (simplex, half-duplex, and full-duplex)



a. Simplex



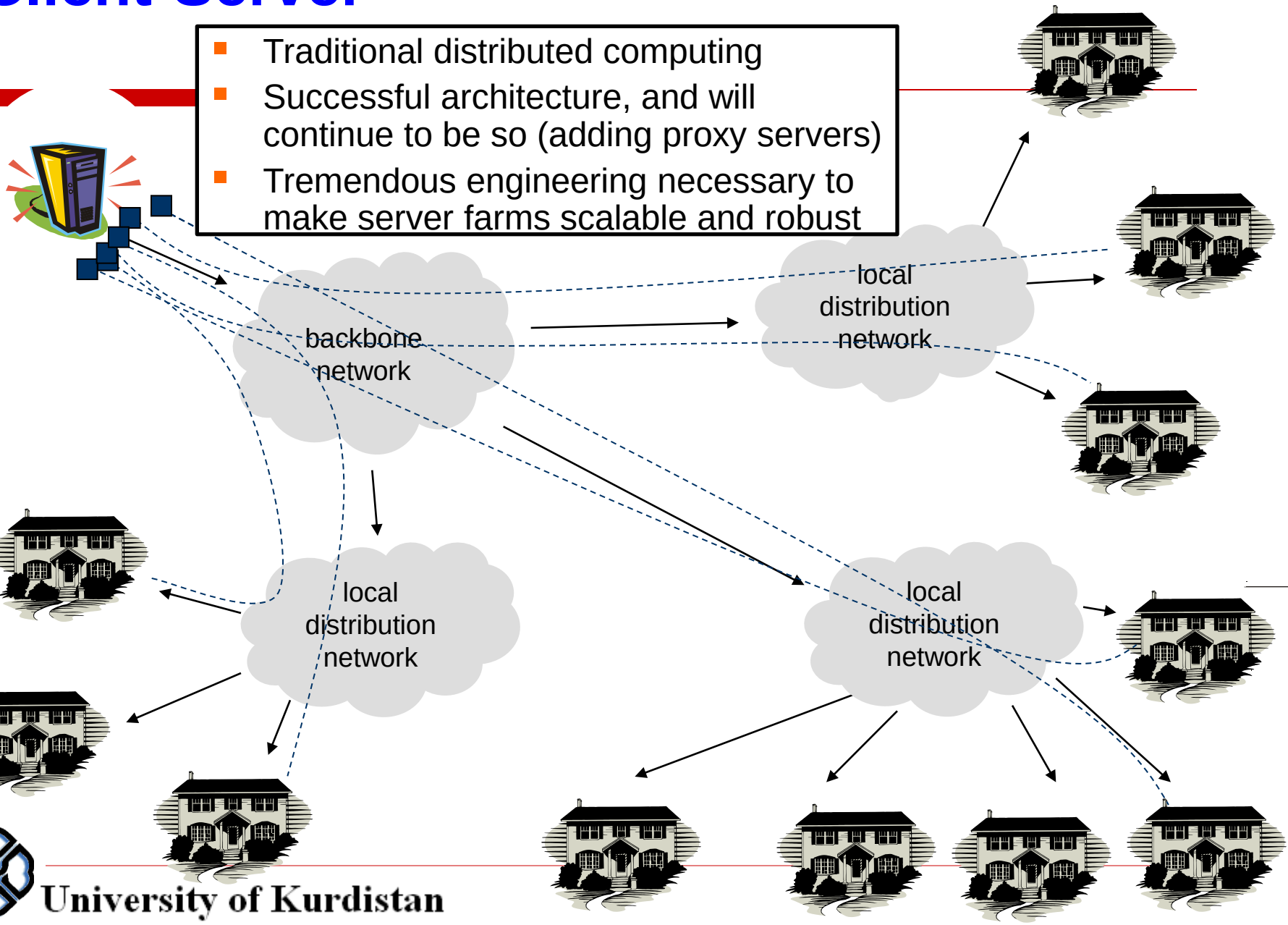
b. Half-duplex



c. Full-duplex

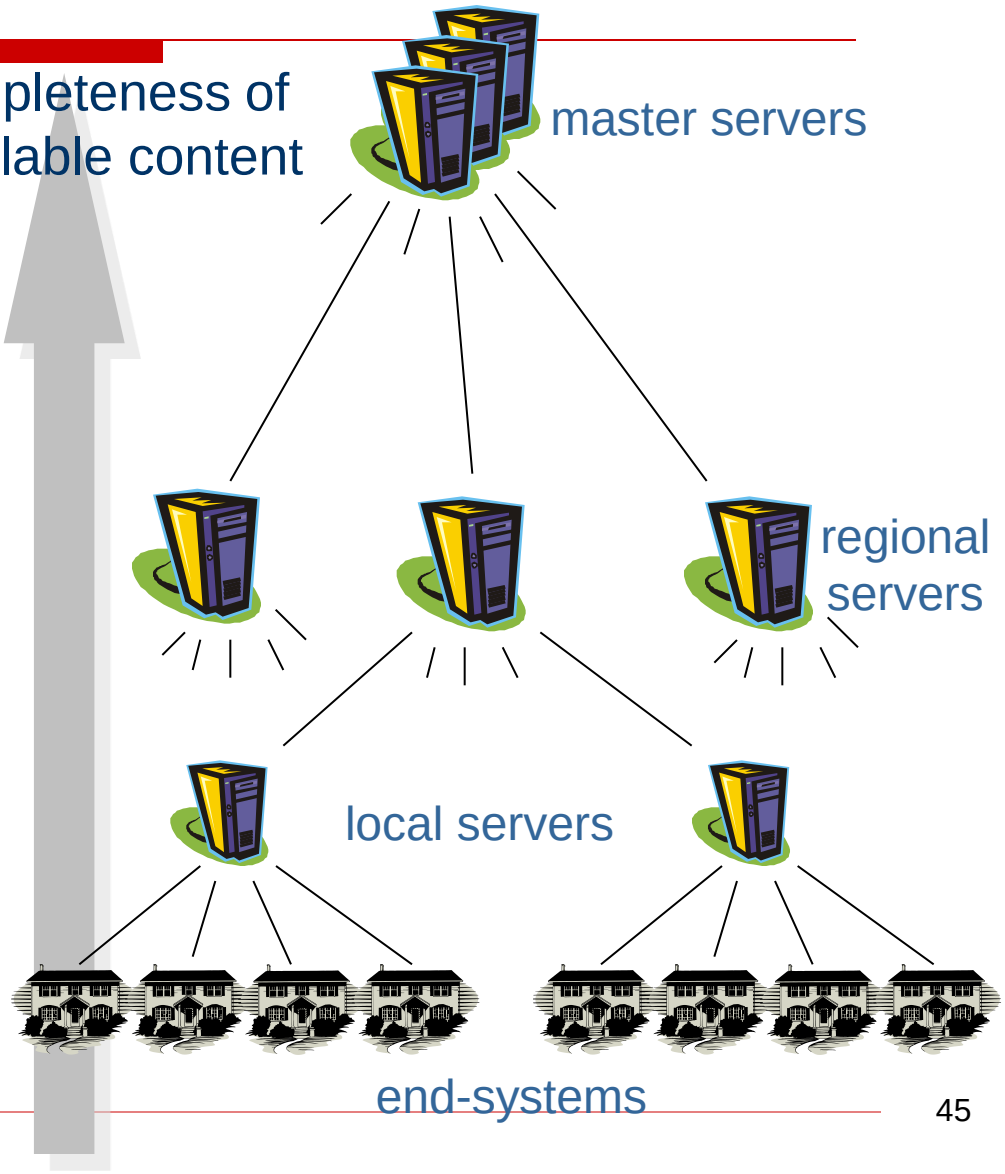
Client-Server

- Traditional distributed computing
- Successful architecture, and will continue to be so (adding proxy servers)
- Tremendous engineering necessary to make server farms scalable and robust



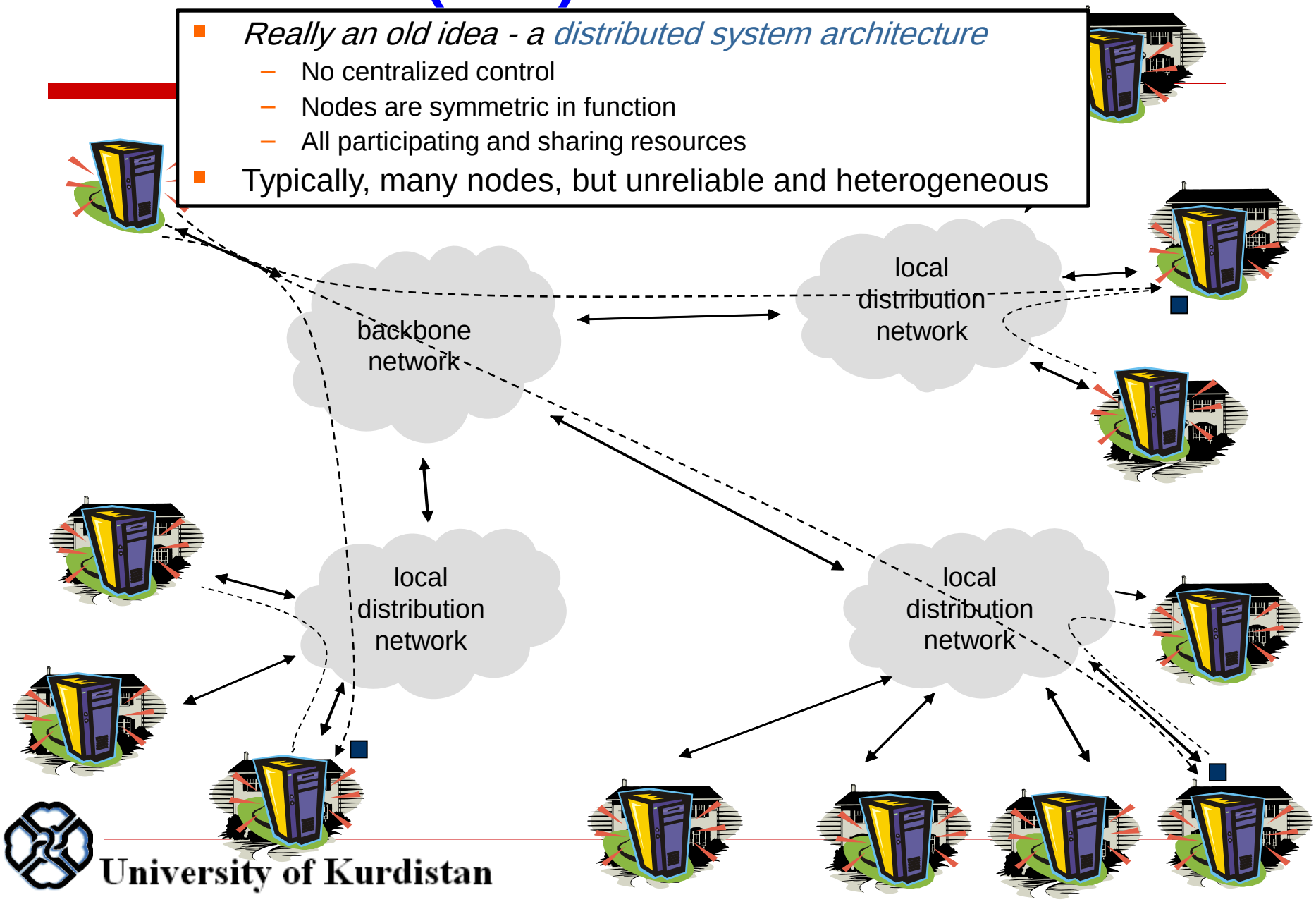
Server Hierarchy

- Intermediate nodes or proxy servers may offload the main master server
- Popularity of data:
  not all are equally popular – most request directed to only a few (Zipf distribution)
- Straight forward hierarchy:
  - popular data replicated and kept close to clients
  - locality vs. communication vs. node costs

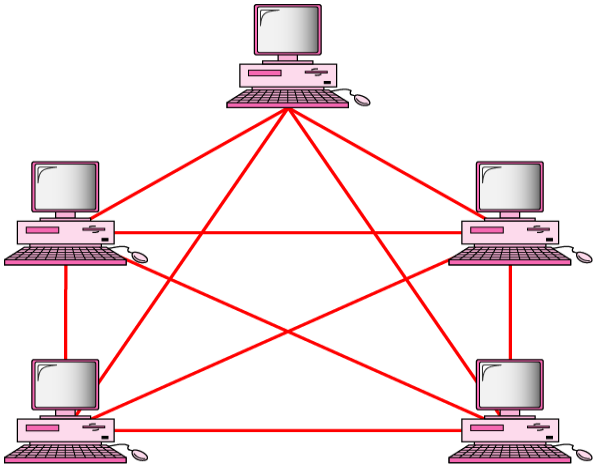


Peer-to-Peer (P2P)

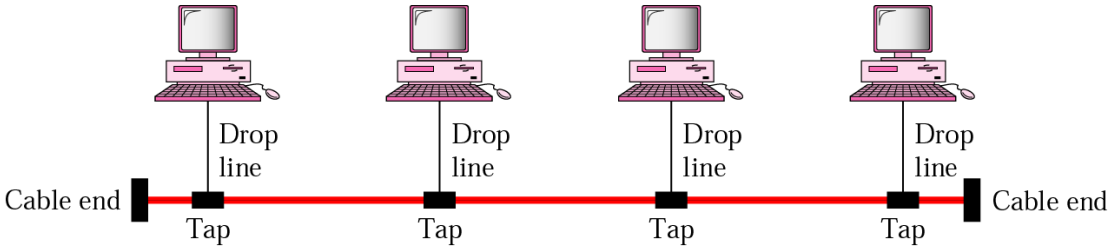
- *Really an old idea - a distributed system architecture*
  - No centralized control
  - Nodes are symmetric in function
  - All participating and sharing resources
- Typically, many nodes, but unreliable and heterogeneous



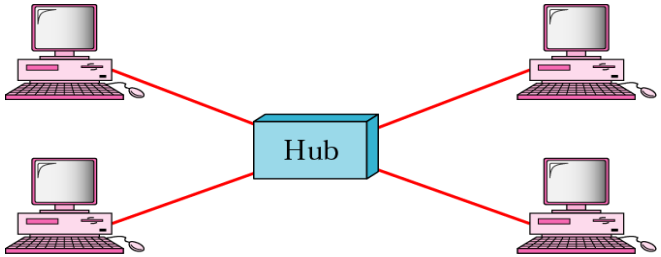
Mostly used network topologies



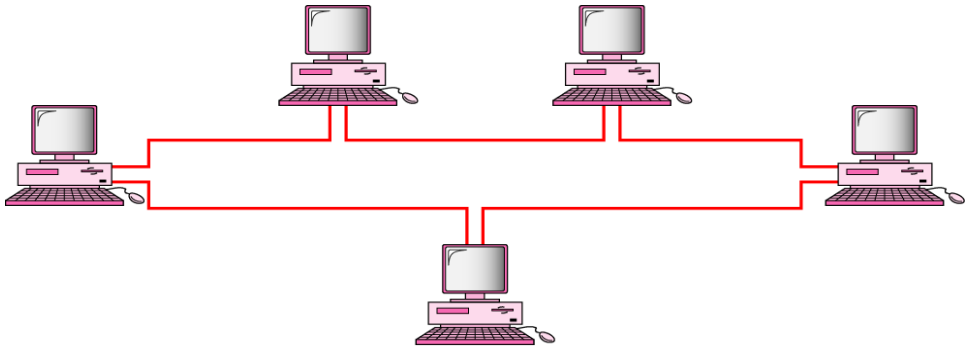
mesh



bus

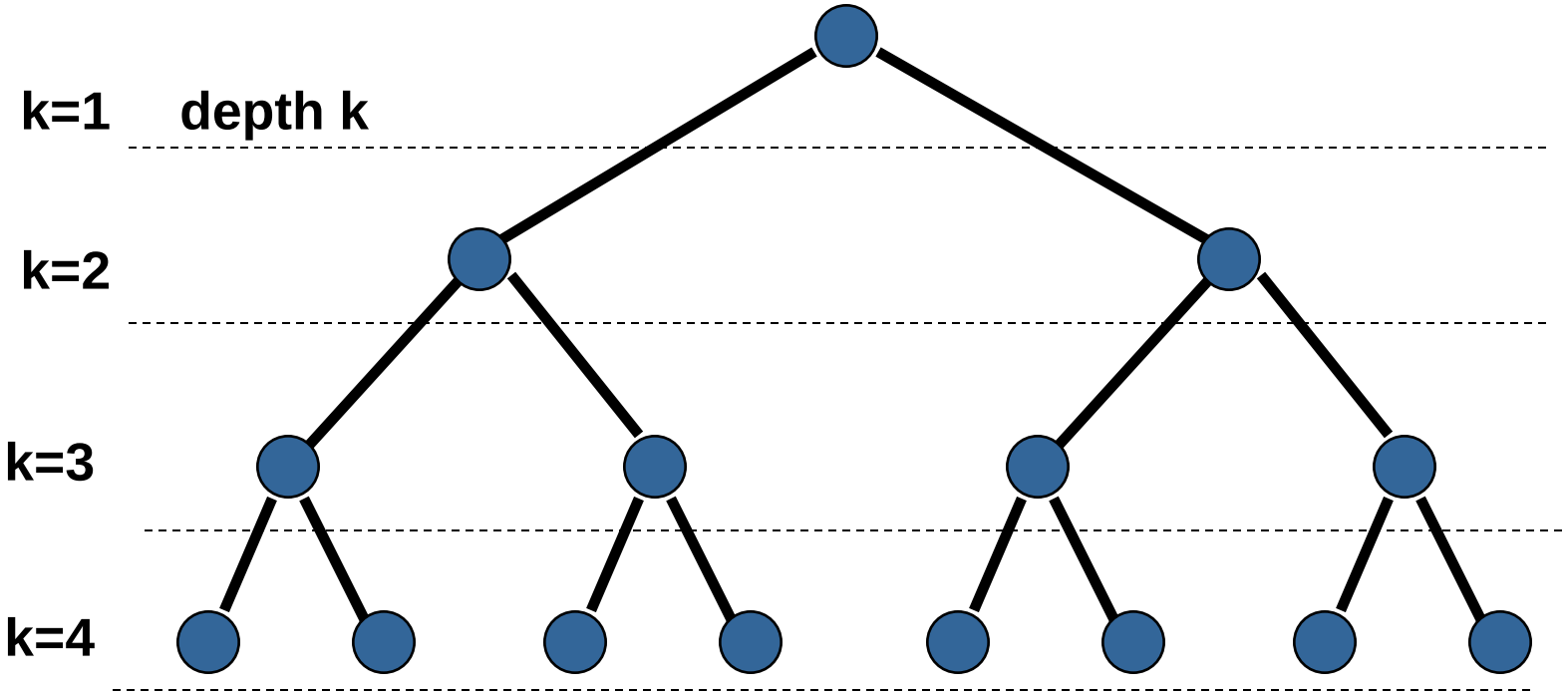


star

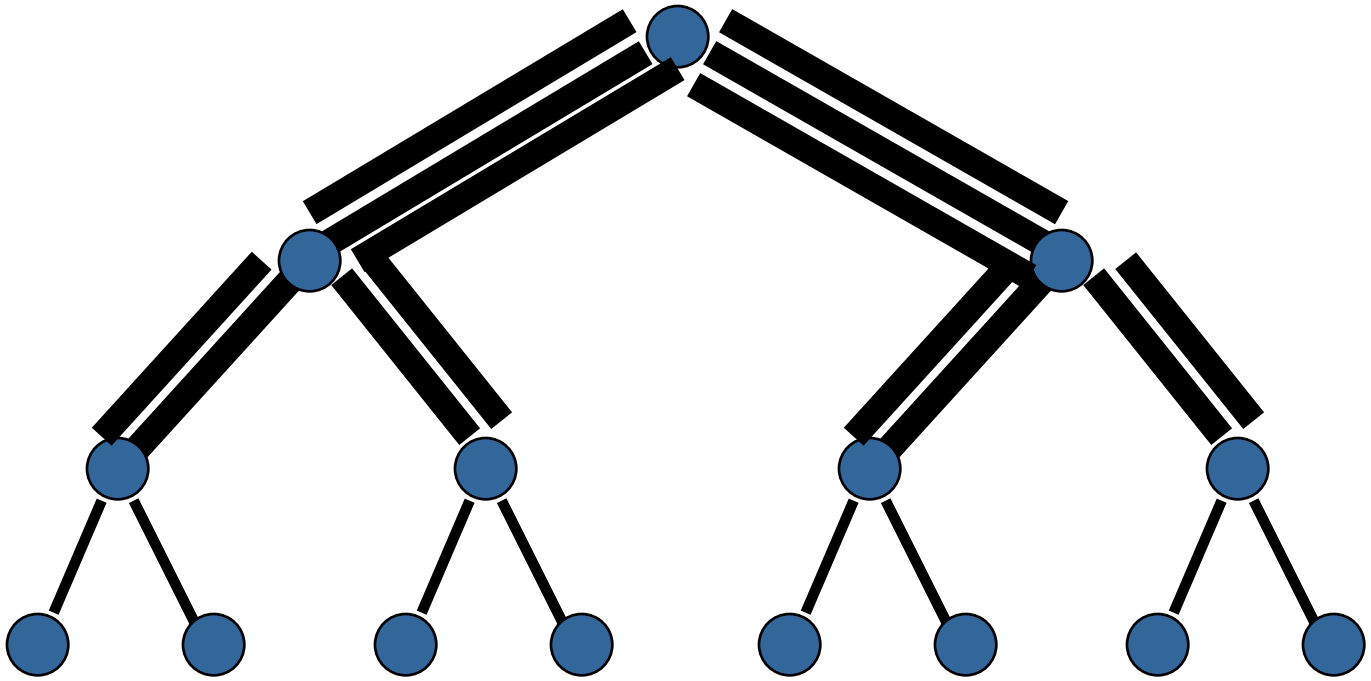


ring

Tree topology

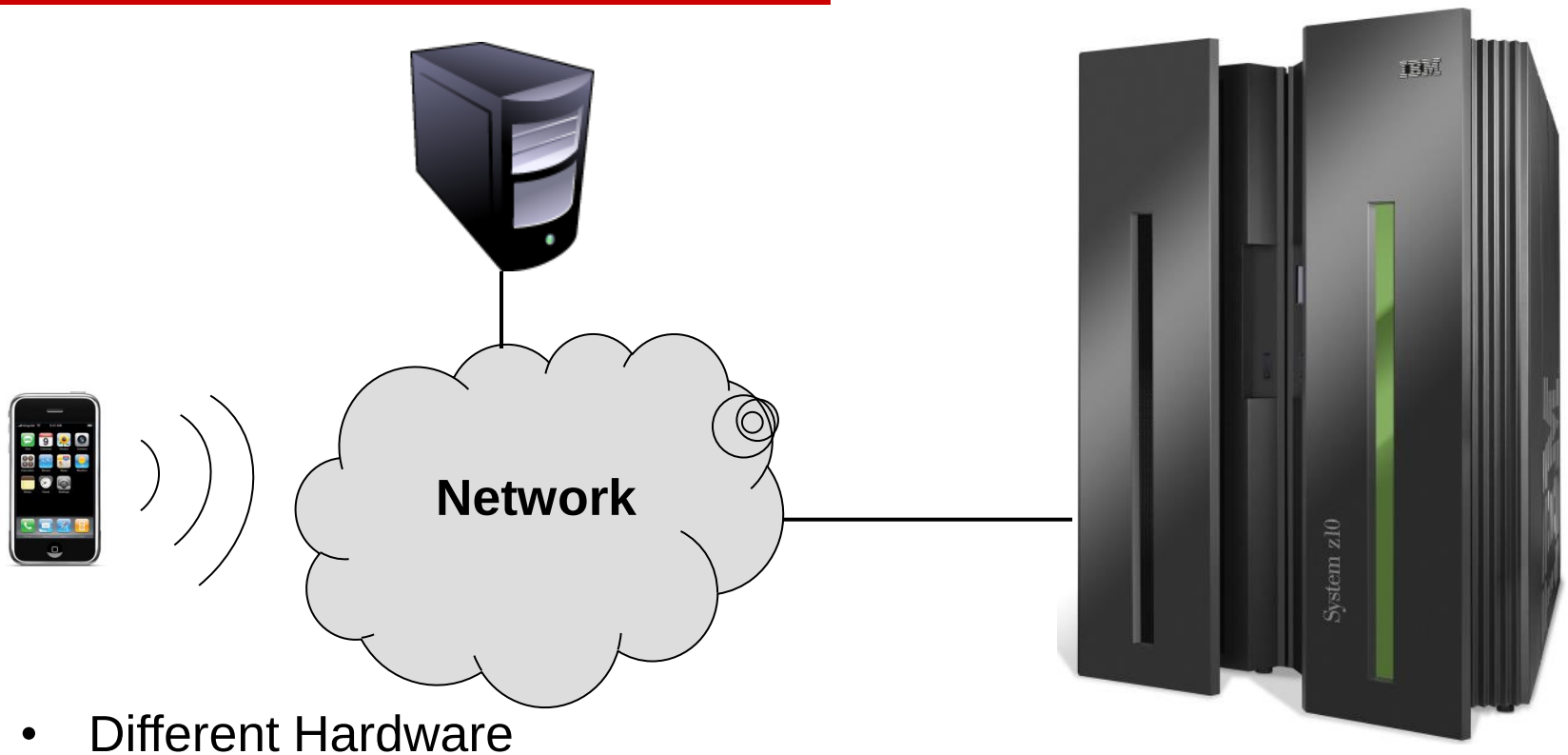


Fat tree topology



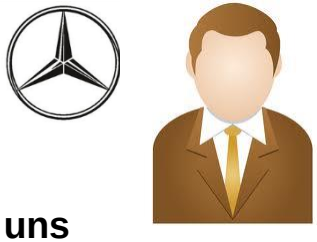
scale bandwidth per level

Complex problem of networking



- Different Hardware
- Different Operating system
- Different Coding
- Different Applications
- …

Layering Concept



Wollen Sie mit uns zusammenarbeiten?

Do you want to cooperate with us?

منشی

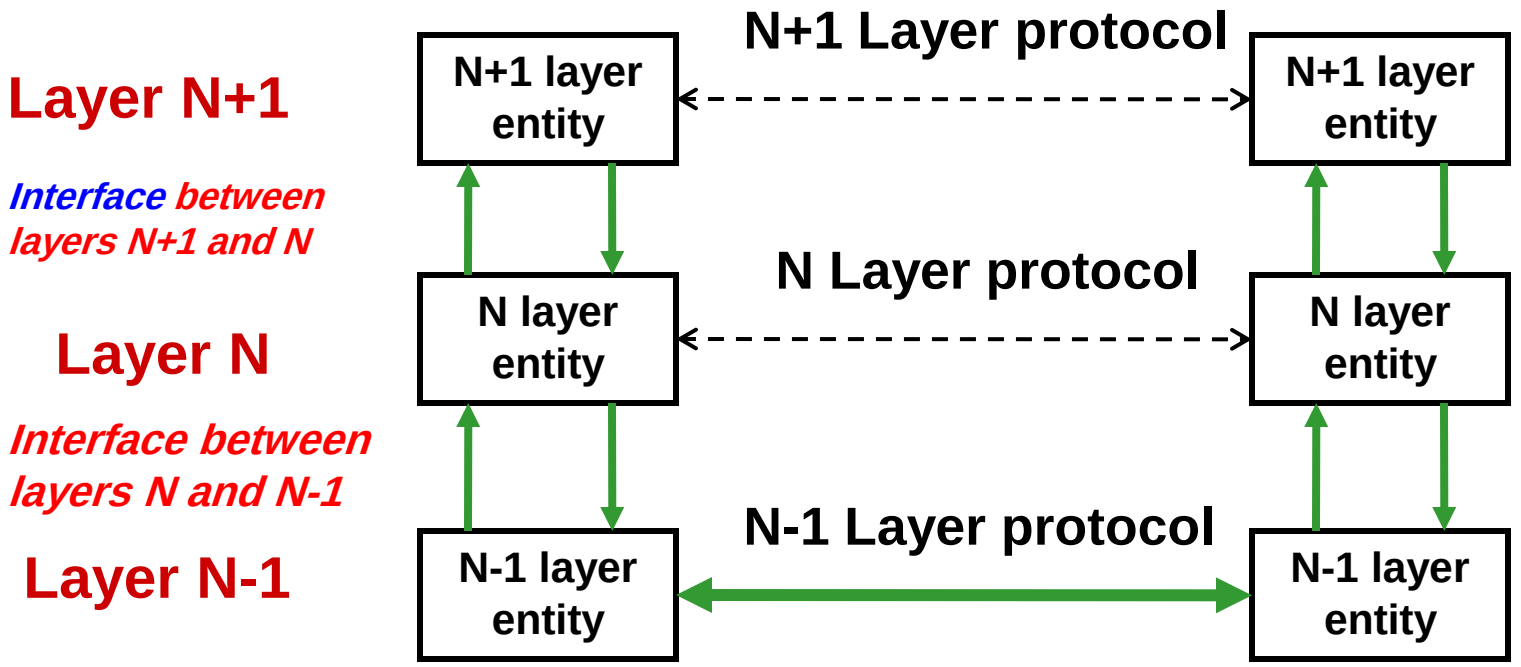
مترجم

Do you want to cooperate with us?

مدیر عامل

آیا می خواهید با ما همکاری کنید؟

Layered communication system



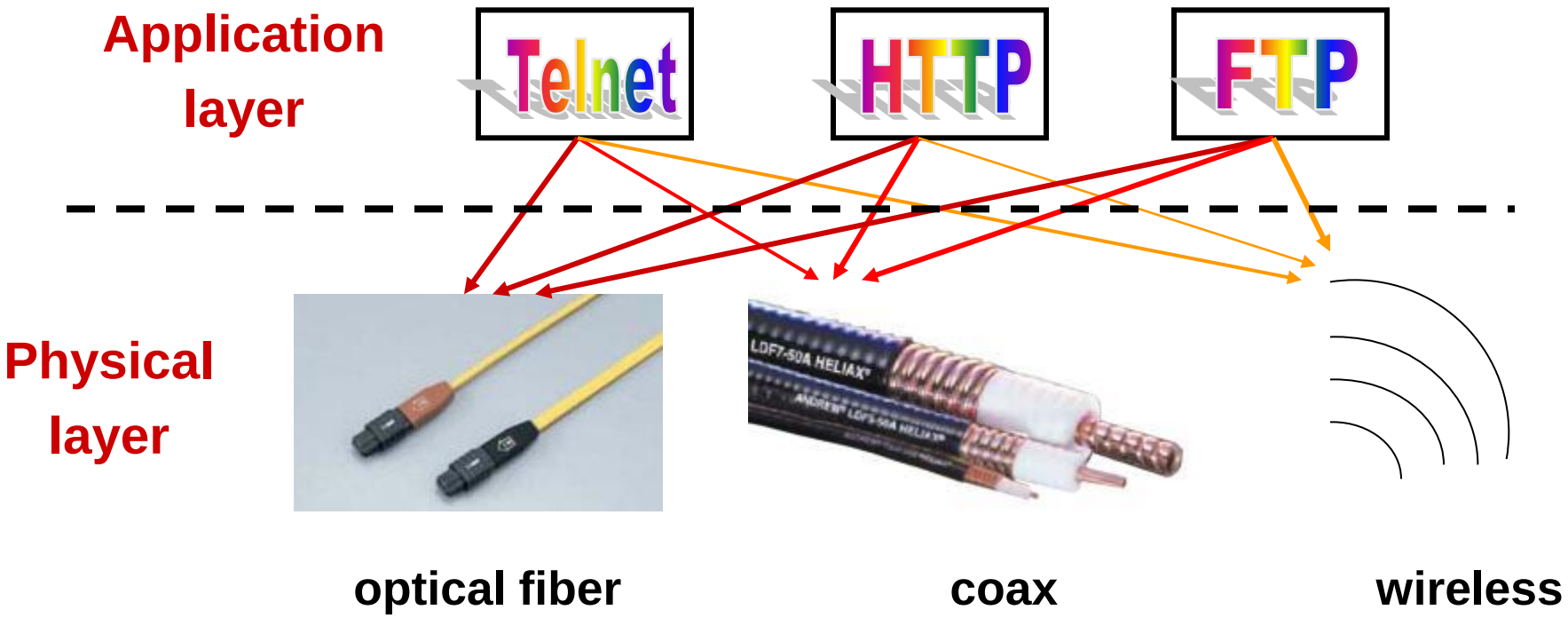
Entities from the same layers - peers

Why layered communication?

- To reduce complexity of communication task by splitting it into several layered small tasks
- Functionality of the layers can be changed as long as the service provided to the layer above stays unchanged
  - makes easier maintenance & updating

- Each layer has its own task
- Each layer has its own protocol

Why layering?

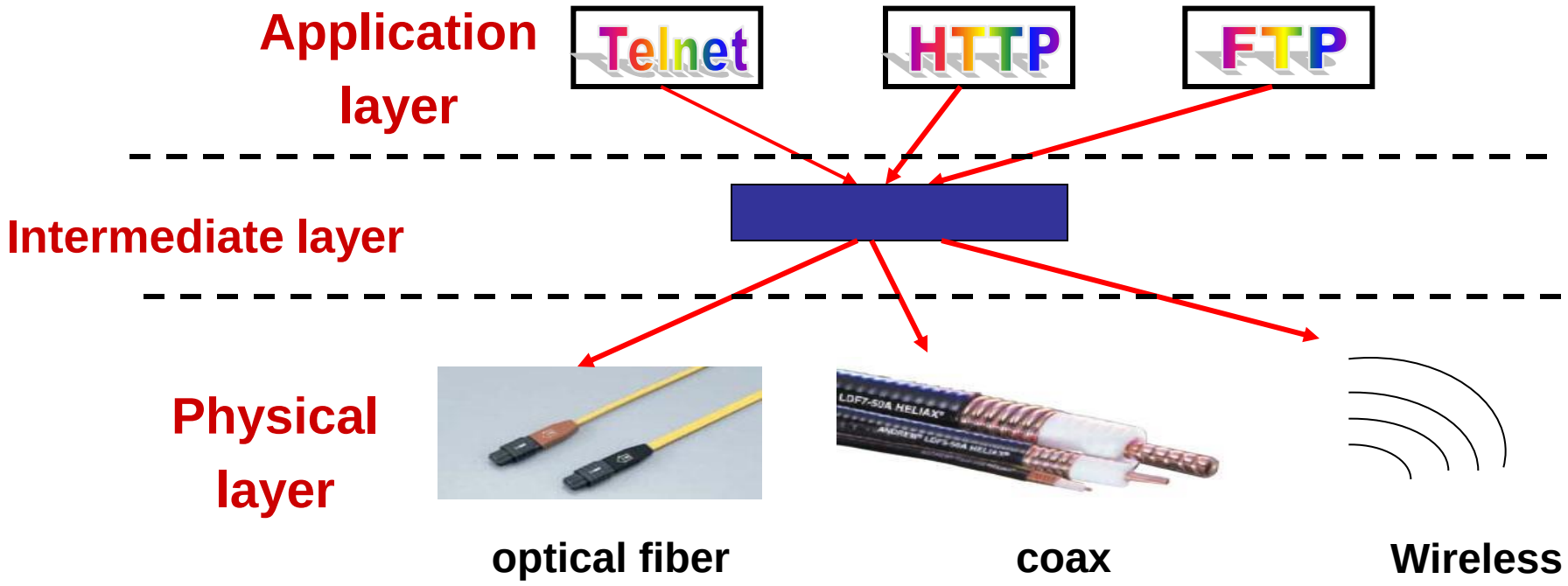


Each new application has to be re-implemented for every network technology

Benefit of layering

Solution to the problem:

introduce an *intermediate layer* that provides a common abstraction for various network technologies



Examples of Layer Design Issues

Addressing: specifying source and destination

Data transfer rules

simplex / half-duplex / duplex

logical channels per connection

priorities, e.g. one normal and one urgent channel

Error control

detection / correction / retransmission

Packet order and sequencing

Flow control

regulate traffic; avoid overflow

Message length: cannot be arbitrary long

(re)assembly needed!

Multiplexing

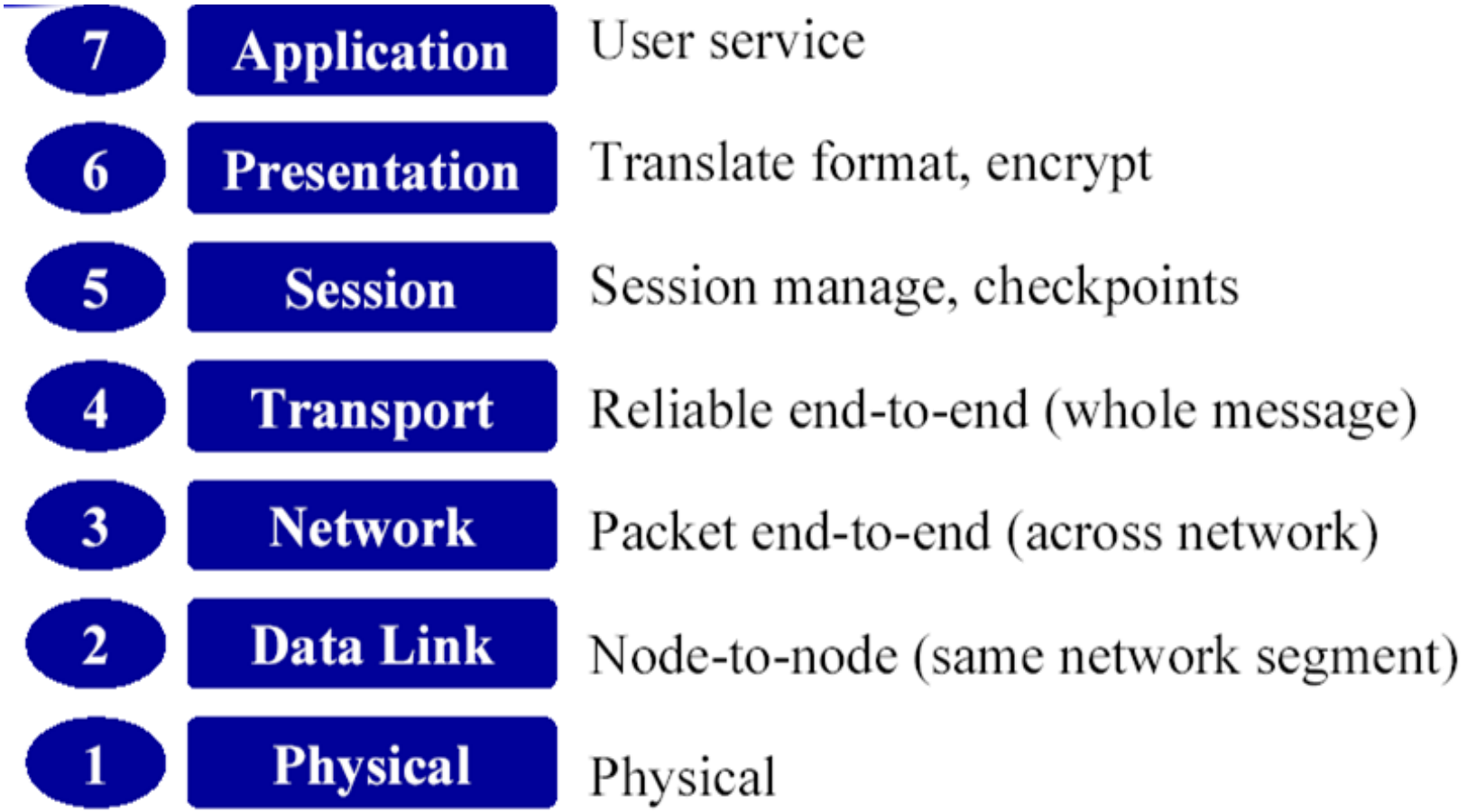
Routing

multiple paths

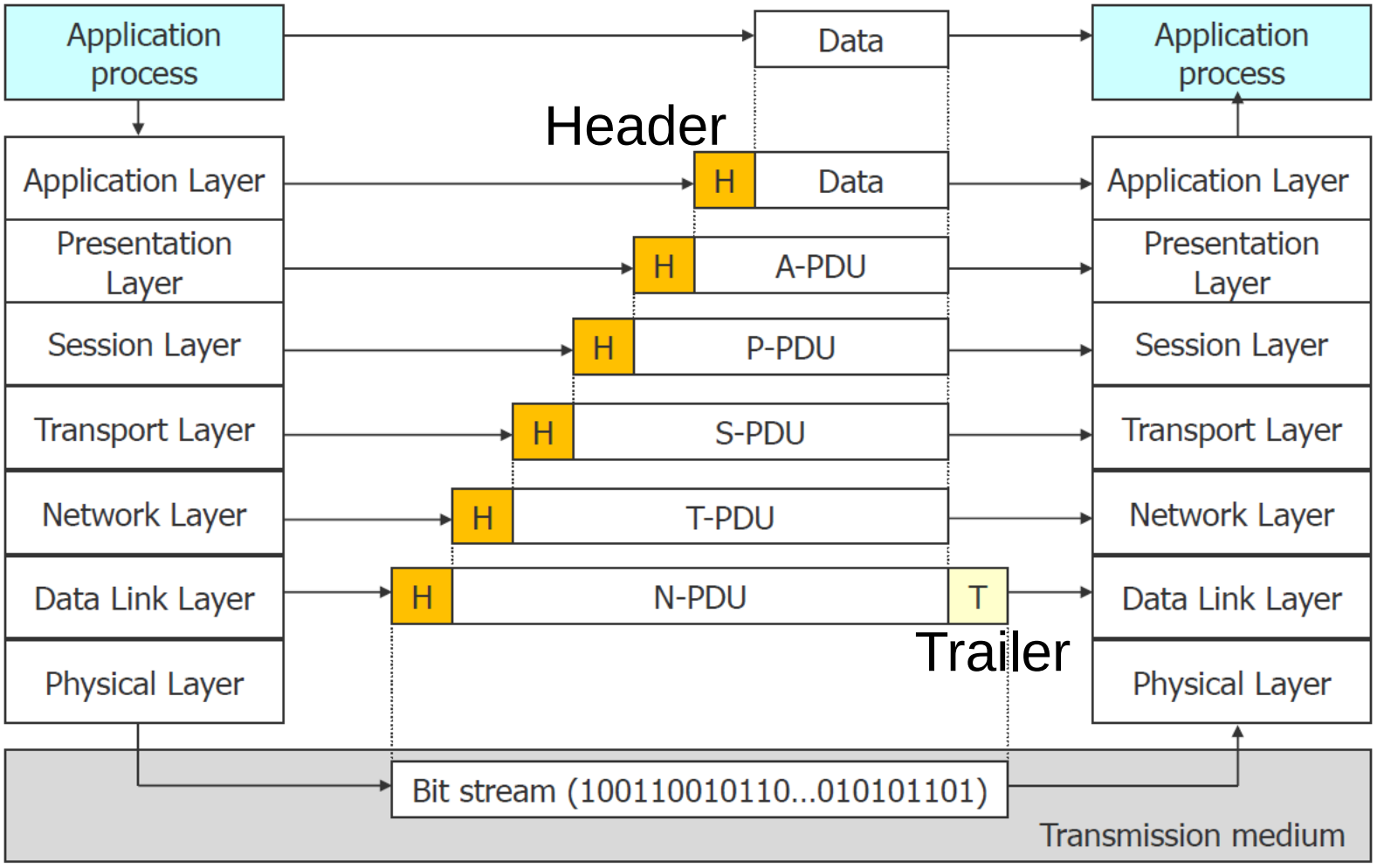
Security

Networking is more than connecting nodes!

OSI reference model



Encapsulation and decapsulation



What is a protocol?

A protocol is an agreement between the communicating parties on how the communication is to proceed

Analogy: politician meeting, defense ceremony

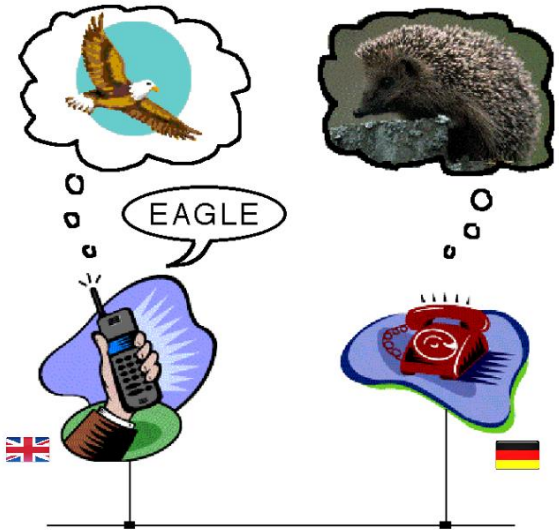
A protocol is a set of rules that specify

- the format of the messages exchanged
- a number of different protocol states and what messages are allowed to be sent in each state;
- these states determine, among others, the order of the messages, timing constraints and other non-functional properties, if any

Example: HTTP, FTP, TCP…

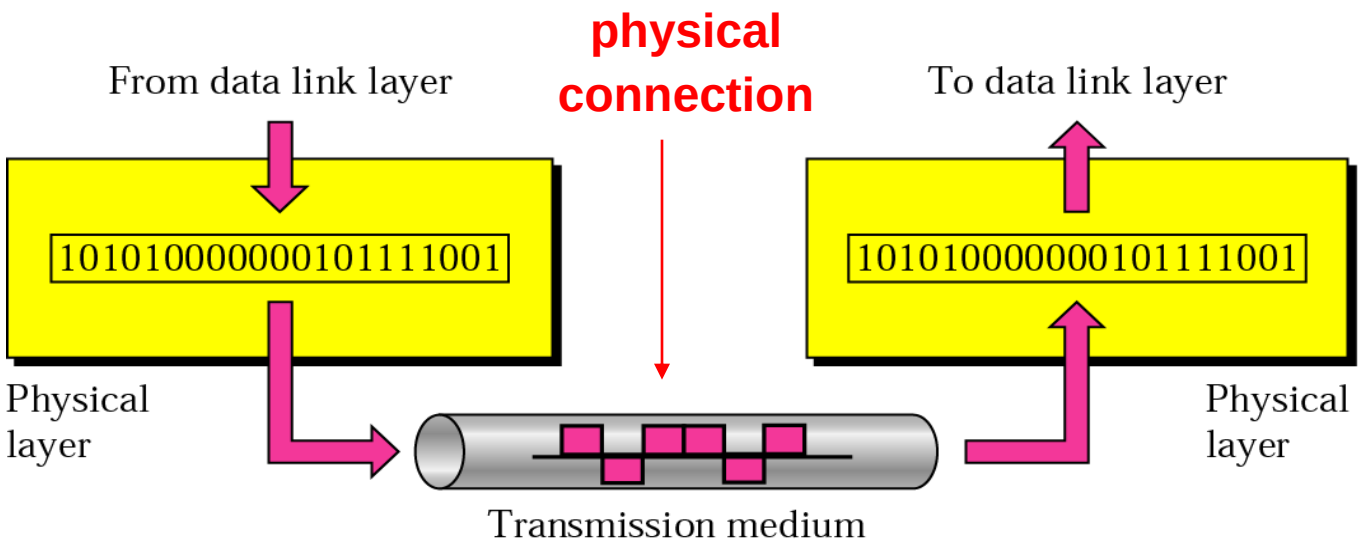
Why we need protocols?

To enable understanding in communication, all communication partners
Have to speak the same “language”.



- Data formats and their semantics
- Control over media access
- Priorities
- Handling of transmission errors
- Sequence control
- Flow control mechanisms
- Segmentation and composition of long messages
- Multiplexing
- Routing

Physical layer



Transporting bits from one end node to the next

- **type of the transmission media** (twisted-pair, coax, optical fiber, ether)
- **bit representation** (voltage levels of logical values)
- **data rate** (speed)
- **synchronization of bits** (time synchronization)

Physical media

- bit: propagates between transmitter/receiver pairs
- physical link: what lies between transmitter & receiver
- guided media:
  - signals propagate in solid media: copper, fiber, coax
- unguided media:
  - signals propagate freely, e.g., radio

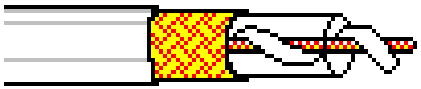
twisted pair (TP)

- two insulated copper wires
  - Category 5: 100 Mbps, 1 Gpbs Ethernet
  - Category 6: 10Gbps
  - CAT-5 is rated to 100 Mhz
  - CAT-5e is rated to 350 Mhz
  - CAT-6 and CAT6e is rated to 550 Mhz or 1000 Mhz depending on source
  - -7 is rated to 700 Mhz or 1000
  - -8 is rated to 2GHz.

Physical media: coax, fiber

coaxial cable:

- two concentric copper conductors
- bidirectional
- broadband:
  - multiple channels on



fiber optic cable:

- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed point-to-point transmission (e.g., 10's-100's Gpbs transmission rate)
- low error rate:
  - repeaters spaced far apart
  - immune to electromagnetic noise

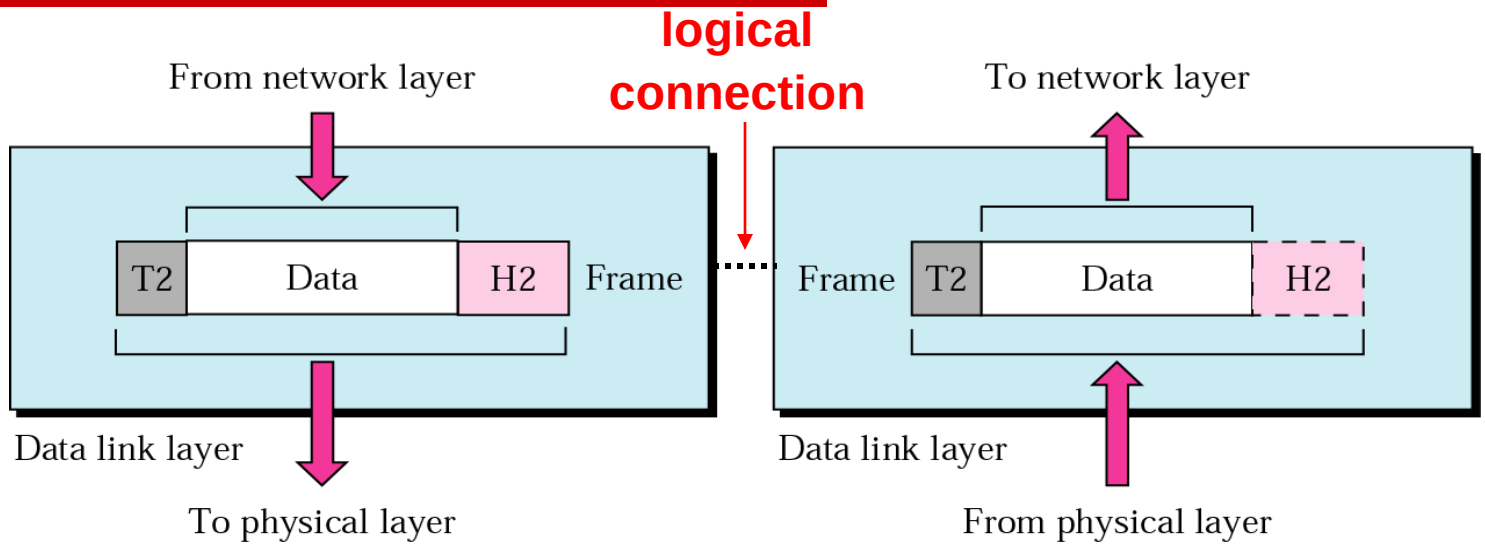
Physical media: radio

- signal carried in electromagnetic spectrum
- no physical "wire"
- bidirectional
- propagation environment effects:
  - reflection
  - obstruction by objects
  - interference

radio link types:

- terrestrial microwave
  - e.g. up to 45 Mbps channels
- LAN (e.g., WiFi)
  - 11Mbps, 54 Mbps
- wide-area (e.g., cellular)
  - 3G cellular: ~ few Mbps
- satellite
  - Kbps to 45Mbps channel (or multiple smaller channels)
  - 270 msec end-end delay
  - geosynchronous versus low altitude

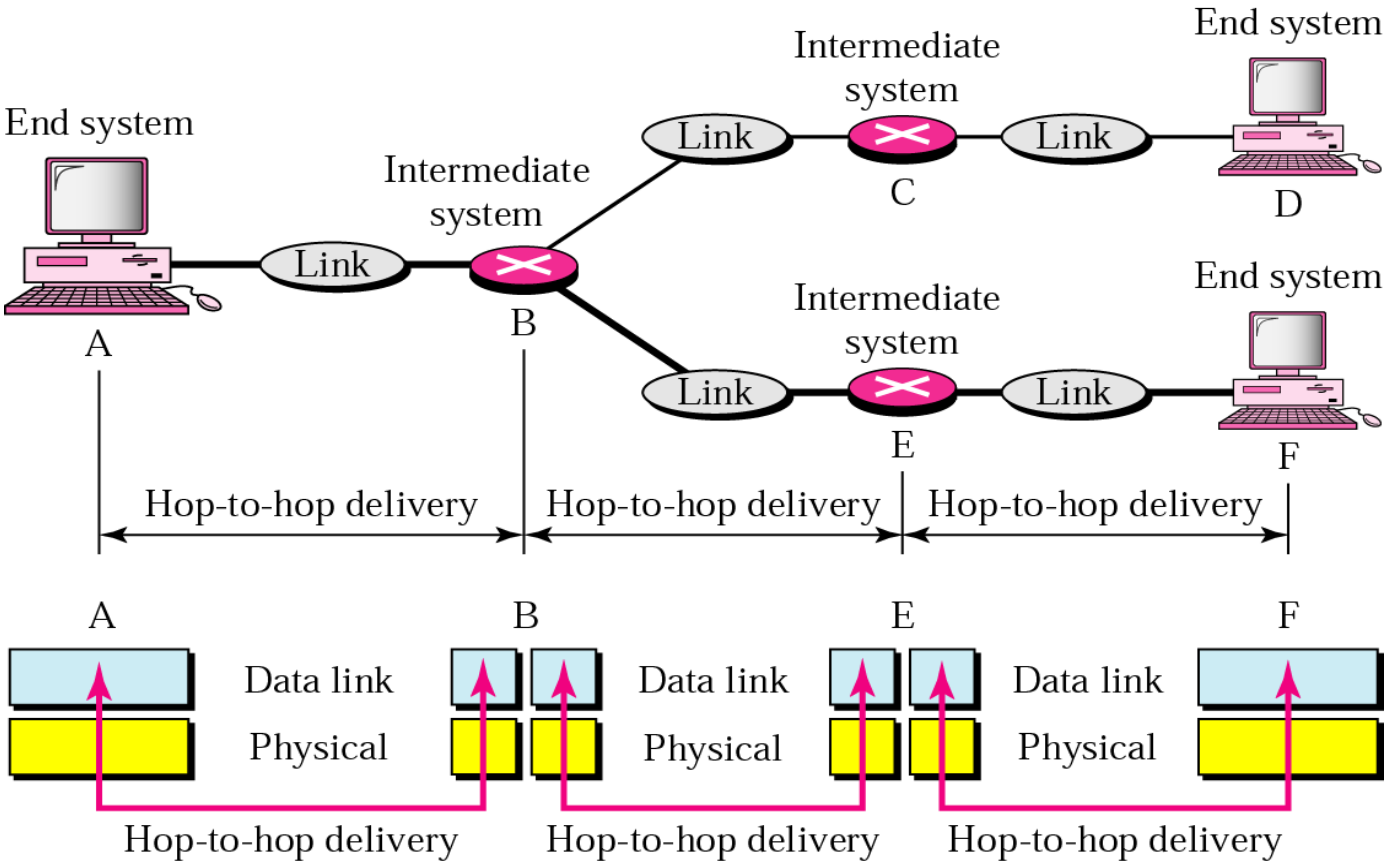
Data Link layer



Transporting frames from one end node to the next one

- framing
- physical addressing
- flow control
- error control
- access control (broadcast networks)

Data Link layer

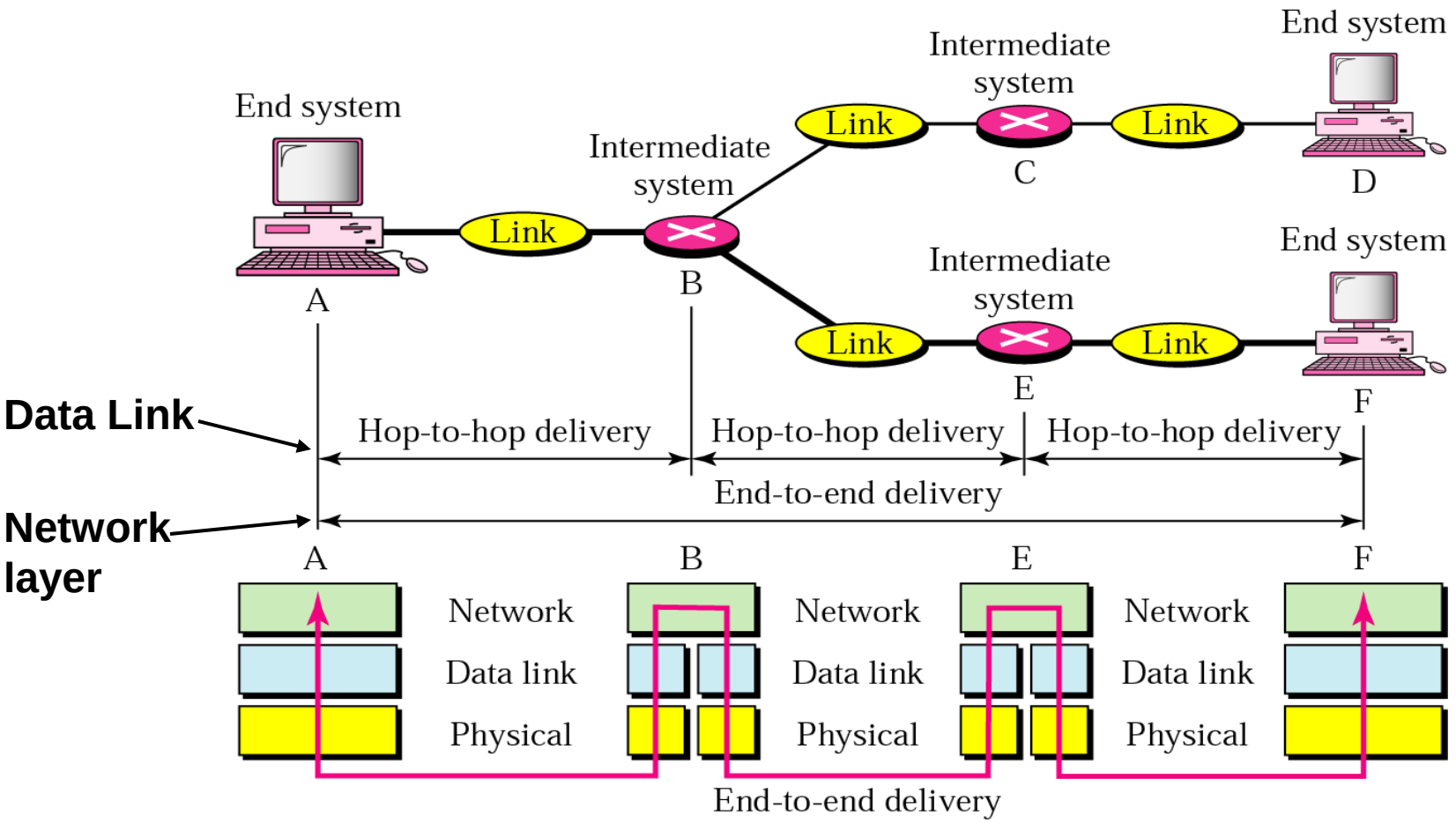


hop-to-hop delivery

Network layer

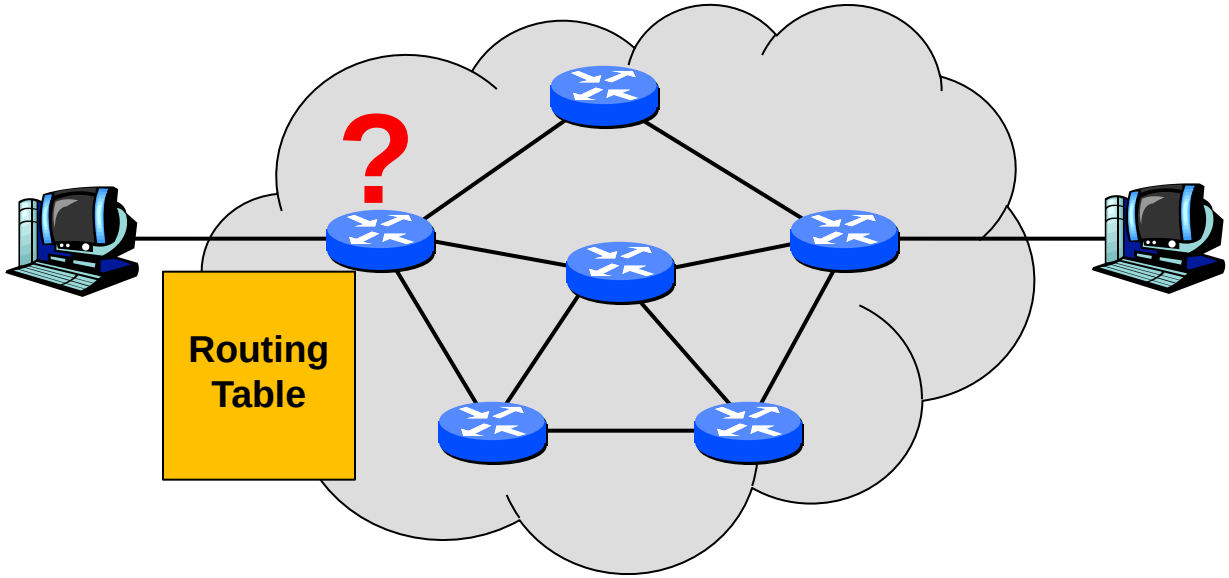
- End-to-End **packet** delivery
  - From the original source to a destination
- Needed when 2 devices are attached to different networks
  - What is the network definition here?
- Main duties:
  1. Logical addressing
  2. Routing
  3. Congestion control and QoS

Network layer



Source to destination delivery

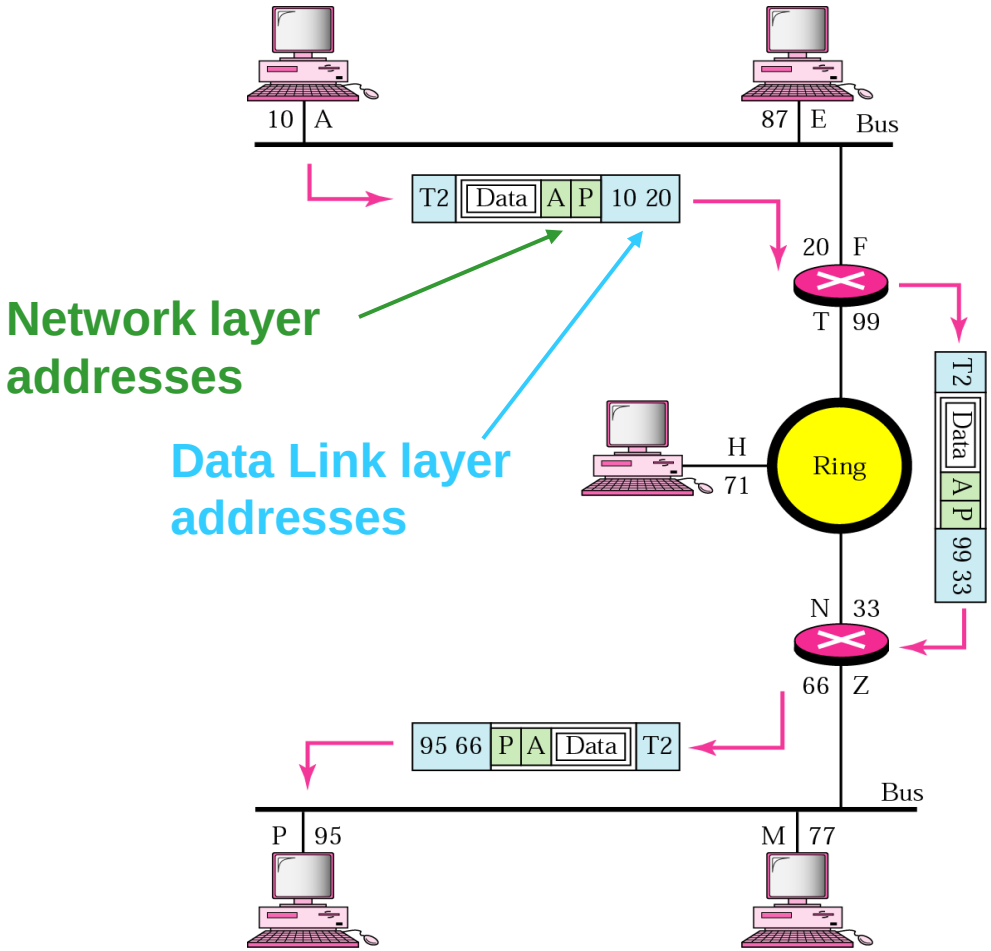
Network layer



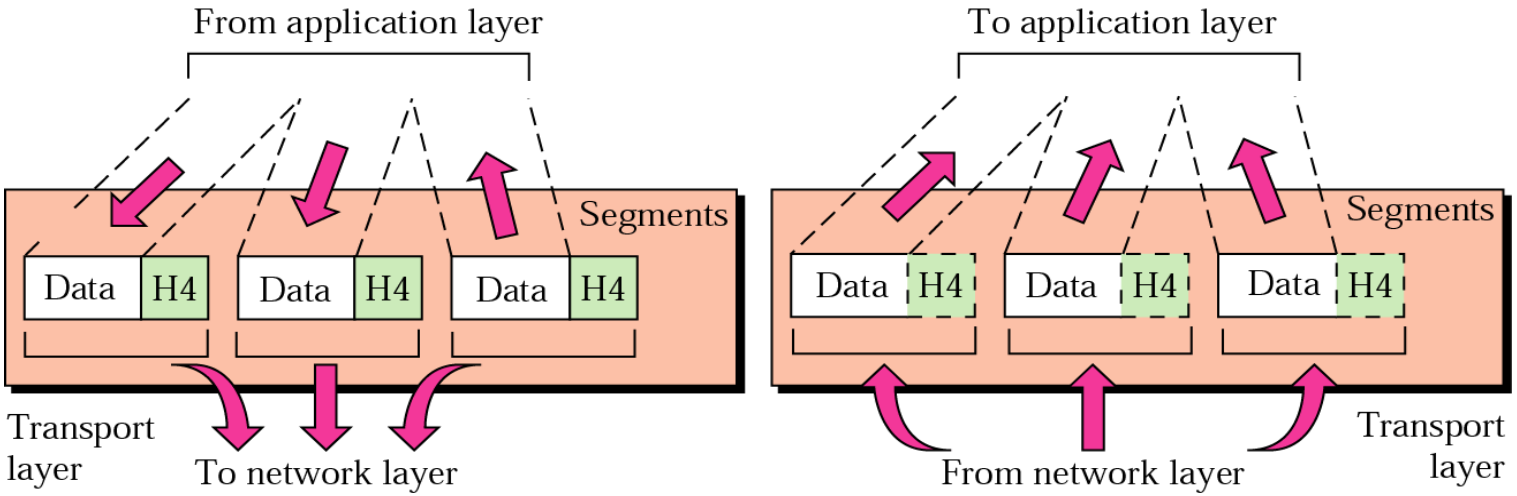
Routing: determining the path from the source of a message to its destination

Congestion Control: handling traffic jams

Network layer (example)



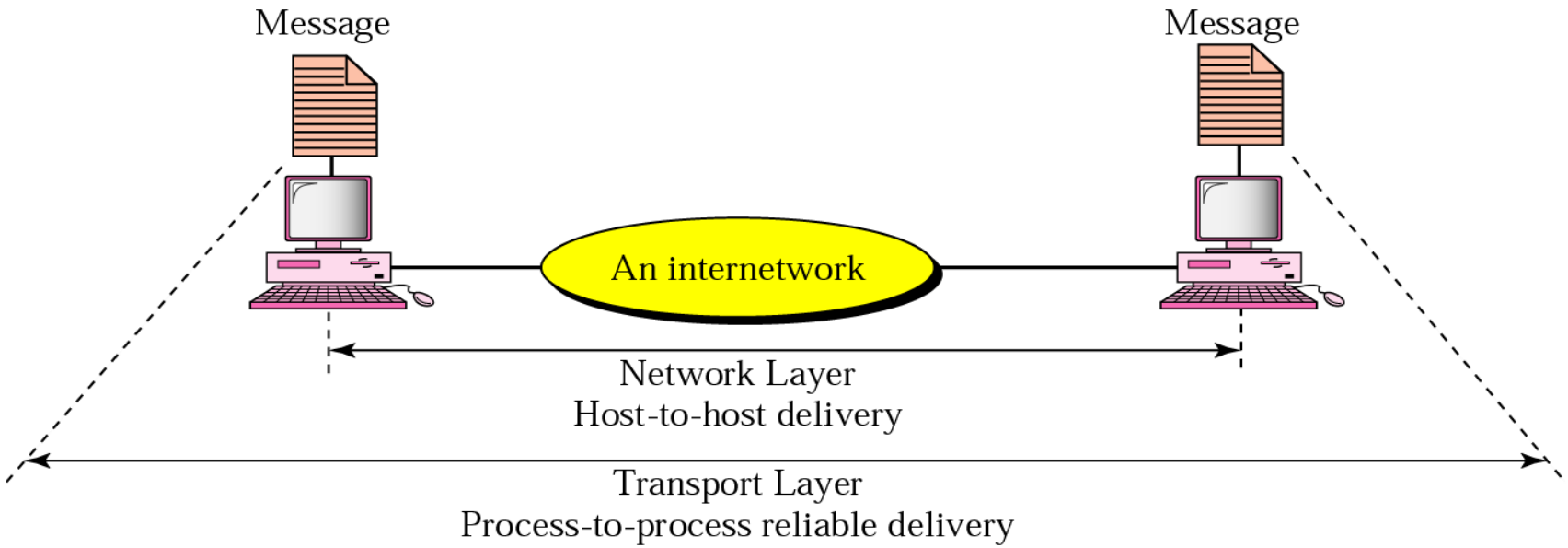
Transport layer



Transport layer

- Process-to-Process delivery of the entire **message**
  - From the original source to destination
- Needed when several processes (running programs) active at the same time
- **Main tasks:**
  - Port addressing
  - Segmentation and reassembly
  - Congestion control
  - Flow control
  - Error control

Transport layer



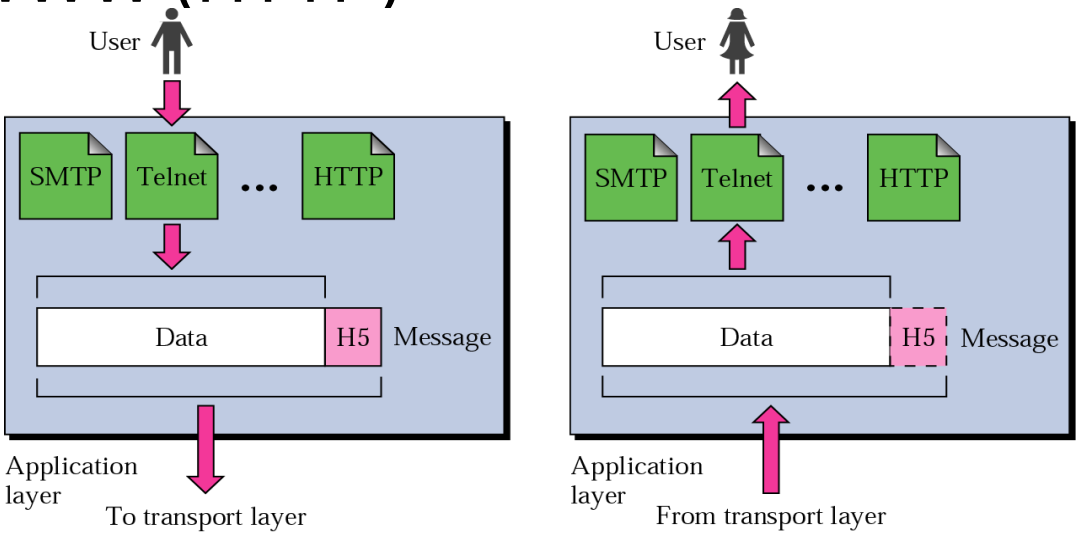
Process-to-Process delivery

Upper Layers

- **Session Layer**
  - user–to–user connection
  - synchronization, checkpoint, and error recovery
- **Presentation Layer**
  - data representation/compression
  - cryptography and authentication

Application layer

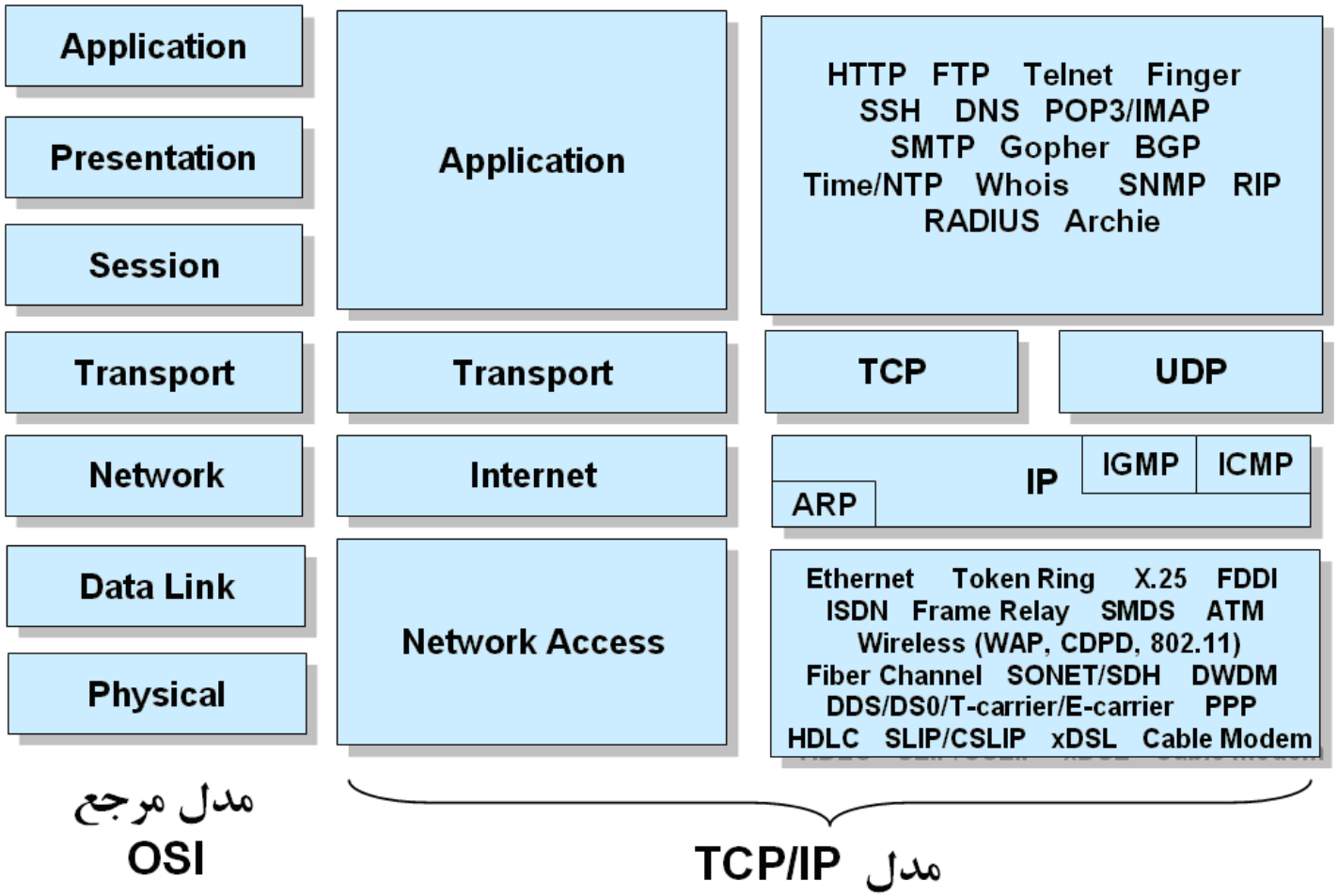
- Enables user to access the network
- Provides services to a user
  - E-mail
  - Remote file access and transfer (Telnet, FTP)
  - Access to WWW (HTTP)



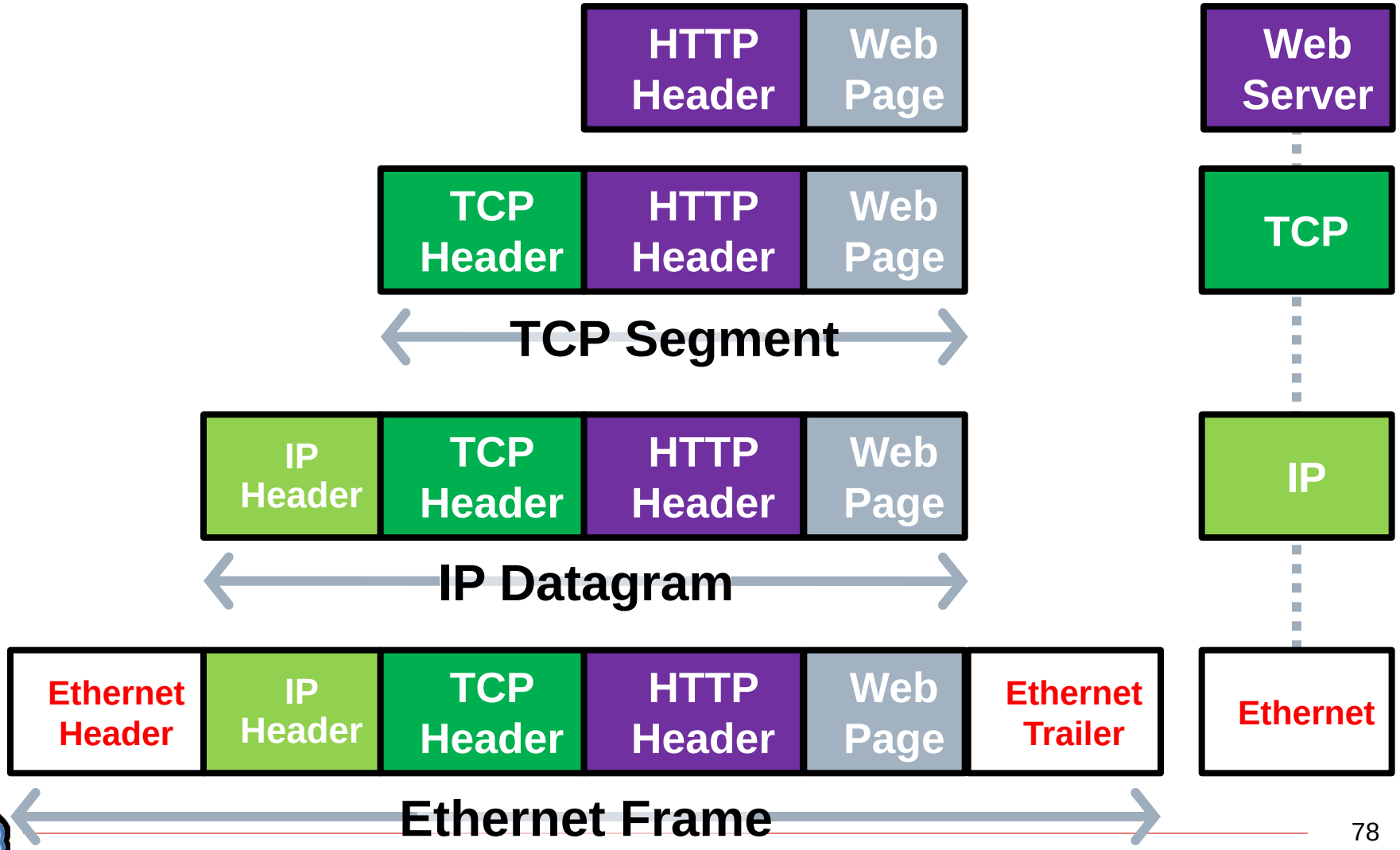
Internet model (TCP/IP)

- There are no presentation and session layers in the Internet model.
- The internet layer is the equivalent of the network layer in the OSI model.
- The physical and data link layers in the OSI model are merged to the “Host to Network” layer.

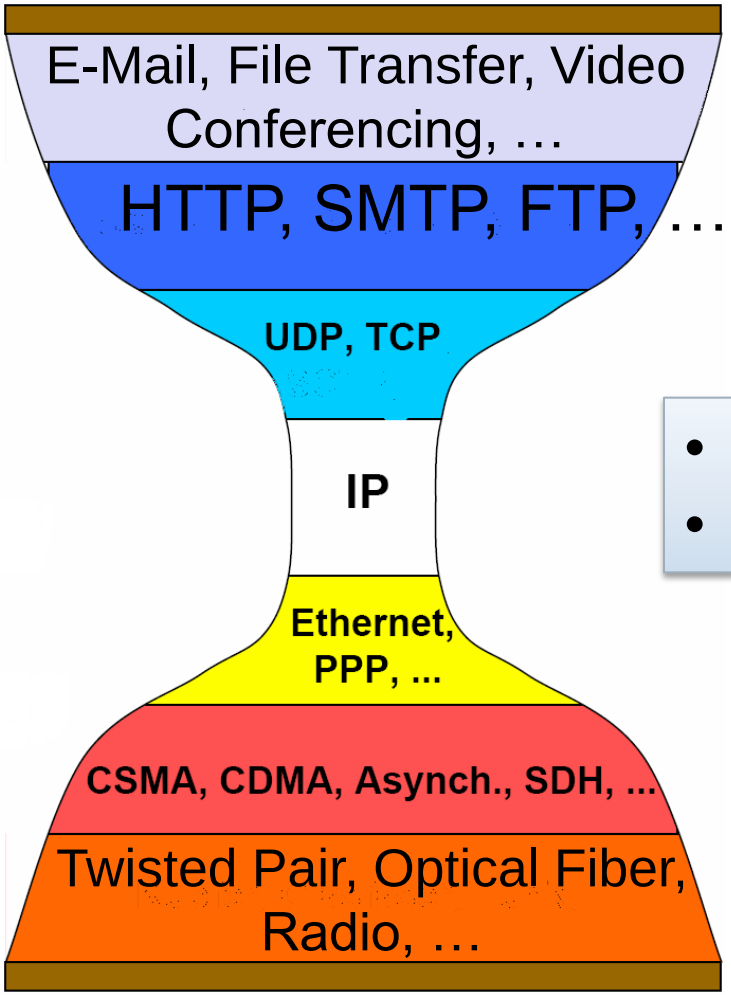
OSI VS. TCP/IP


Application
Presentation
Session
Transport
Network
Data Link
Physical
Application
Transport
Internet
Network Access
HTTP FTP Telnet Finger
SSH DNS POP3/IMAP
SMTP Gopher BGP
Time/NTP Whois SNMP RIP
RADIUS Archie
TCP
UDP
IP
IGMP
ICMP
ARP
Ethernet Token Ring X.25 FDDI
ISDN Frame Relay SMDS ATM
Wireless (WAP, CDPD, 802.11)
Fiber Channel SONET/SDH DWDM
DDS/DS0/T-carrier/E-carrier PPP
HDLC SLIP/CSLIP xDSL Cable Modem
مدل مرجع
OSI
مدل TCP/IP

Encapsulation



The hourglass architecture of the Internet



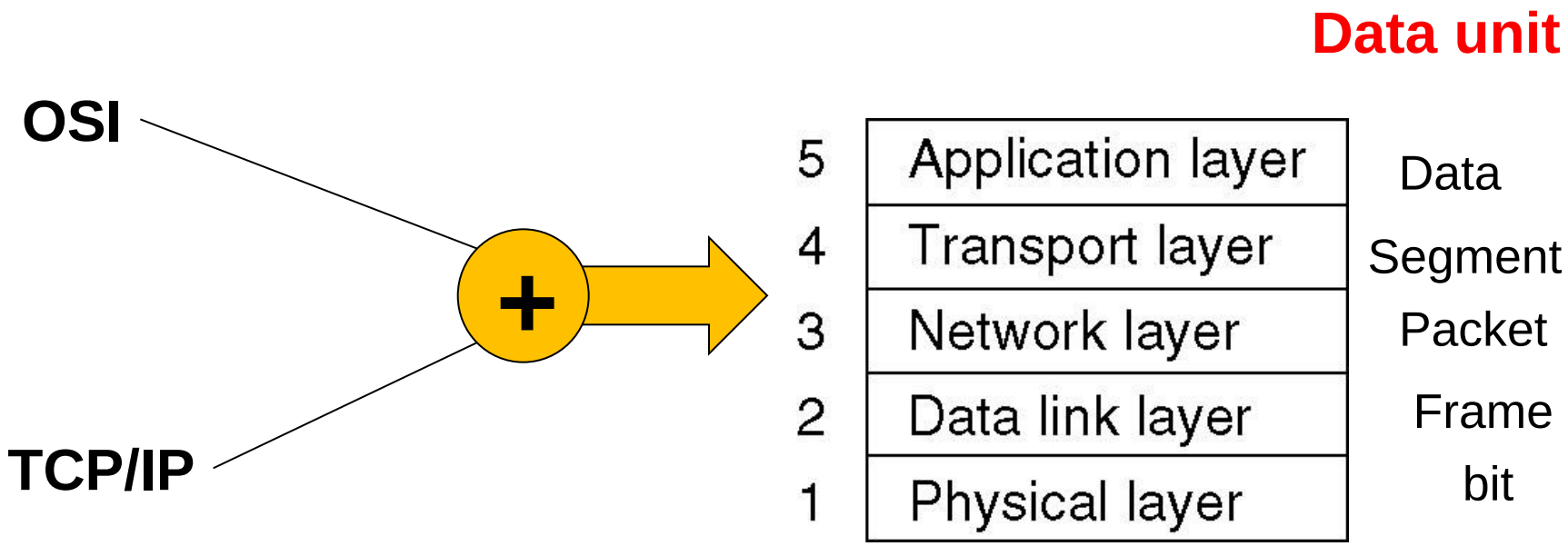
- **Everything over IP**
- **IP over everything**

Implications of Hourglass

A single Internet layer module:

- Allows all networks to interoperate
  - all networks technologies that support IP can exchange packets
- Allows all applications to function on all networks
  - all applications that can run on IP can use any network
- Simultaneous developments above and below IP

Model in this class



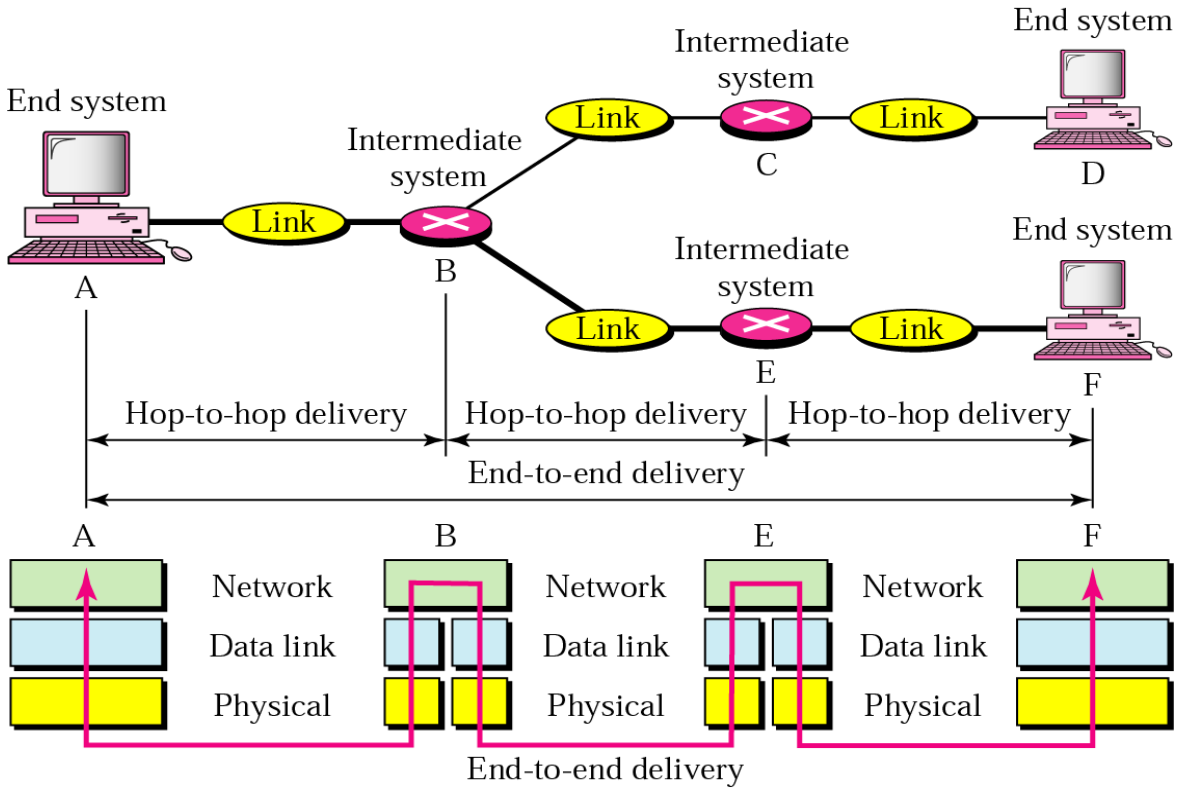
Key design issue:

How to divide the functionality among the layers?

End-to-End argument

- The application knows the requirements best, place functionality as high in the layer as possible
- If the application can implement a functionality correctly, implement it at a lower layer only as a performance enhancement
- Think twice before implementing a functionality that you believe that is useful to an application at a lower layer

Example: Reliability



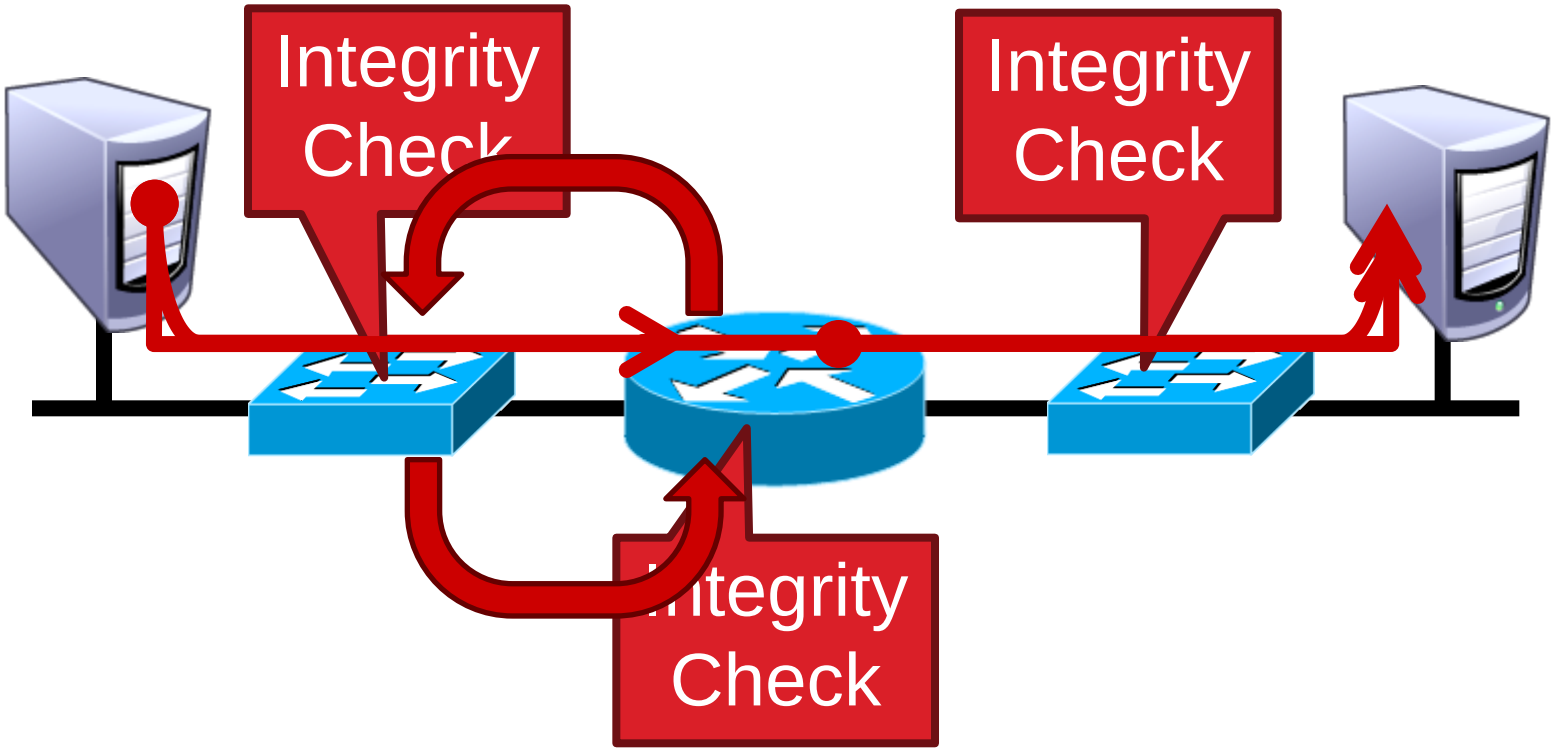
Solution 1: **Make hop-to-hop delivery reliable and concatenate them**

Solution 2: **End-to-End control and retransmission**

Example: Reliability (cnt'd)

- The receiver has to do the check anyway!
- Thus, full functionality can be entirely implemented at the upper layer; no need for reliability from lower layers
  - Is this always correct?

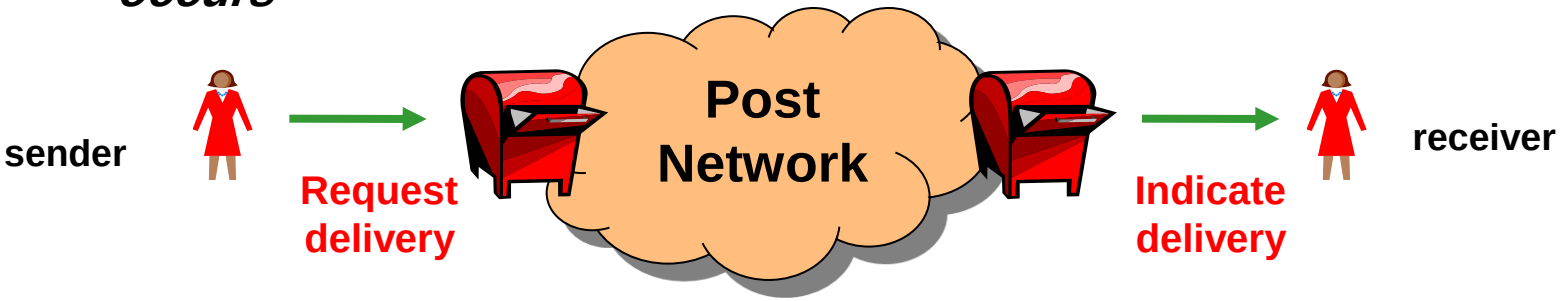
Example: Reliable File Transfer



Network services

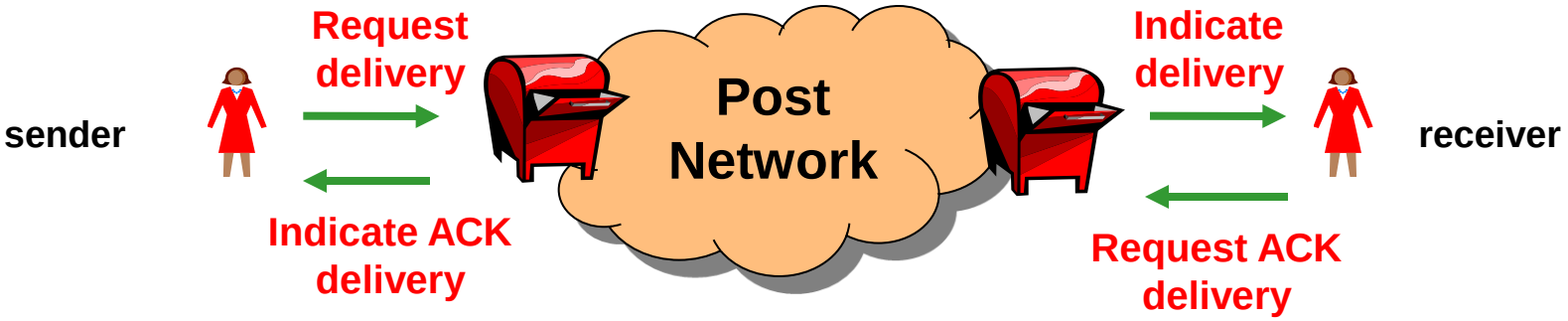
Services provided by different layers

- **Unconfirmed** (Best effort) service: *no feedback if delivery occurs*



- **Confirmed** (Acknowledged) service:

sender gets a confirmation (acknowledgment) of delivery



Network services (cnt'd)

- Unreliable services
  - No guaranteed delivery (no acknowledgments)
  - An example: a basic service of datagram networks
- Reliable services
  - Guaranteed delivery
  - Implementation of this service through combination of *timers, acknowledgment and retransmission*
  - An example: FTP, E-mail

Why would anyone use an unreliable service?

Connection-oriented vs. services

- Establish connection
- Use connection
- Release connection

- Protocol Data Units (PDUs) are delivered in-sequence of transmission without duplication
- Implementation of this service:
  - Virtual-circuit packet switched network
  - In datagram networks, a connection-oriented service can be accomplished by end systems with sequence numbers, retransmission, and other mechanisms

Example: Service of **TCP** protocol,

Frame relay – for connecting LANs

X.25 – Typically across telephone lines

Connectionless service

Example: Postal system

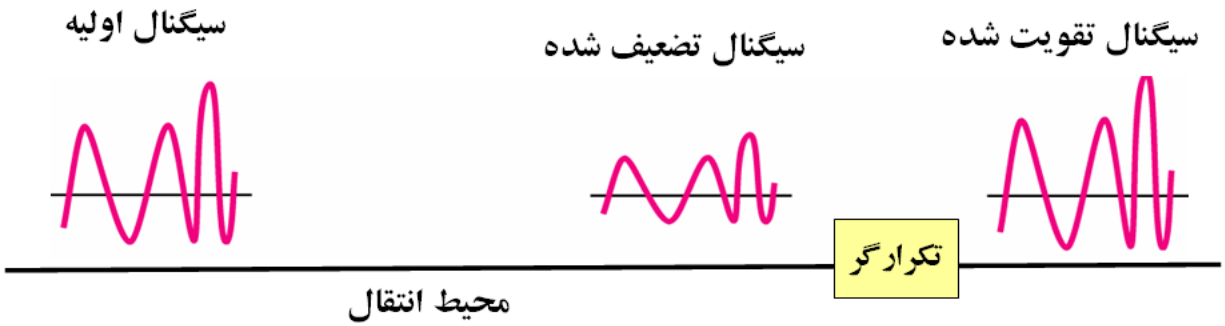
- No guarantee of in-sequence delivery. Losses are possible.

- Implementation of the service
  - default service for datagram communication
  - inefficient to implement in circuit-switching networks and virtual circuit packet switching networks

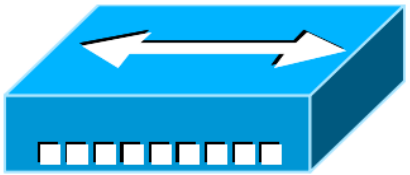
Example: services of both the IP and UDP protocol

Network Devices- Repeater

- **Works at the Physical layer**
- **Regenerates received bits before it sends them out**



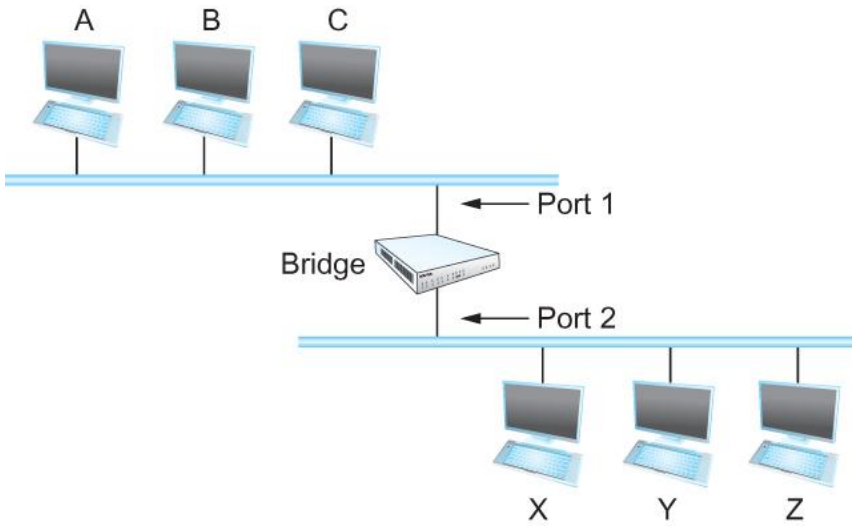
Network Devices- Hub



- **Multiport repeater (layer 1 device)**
- **Just knows bits**

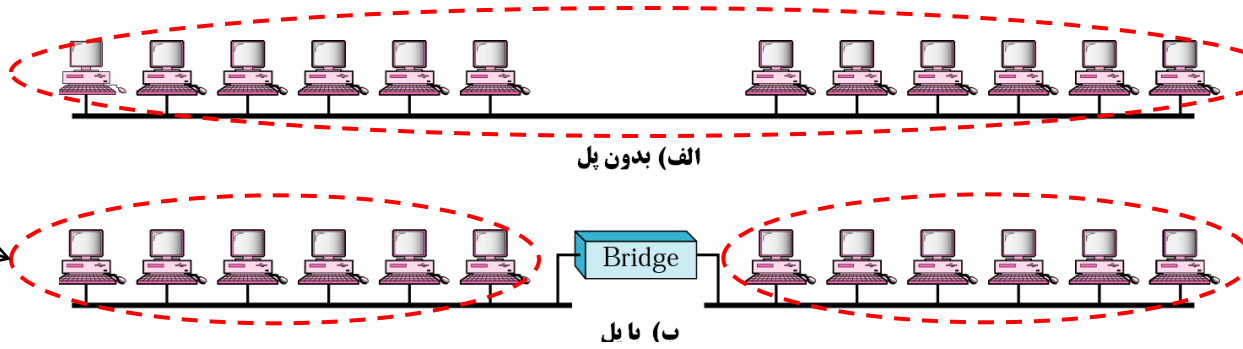
Network Devices- Bridge

Host	Port
A	1
B	1
C	1
X	2
Y	2
Z	2



layer 2 device

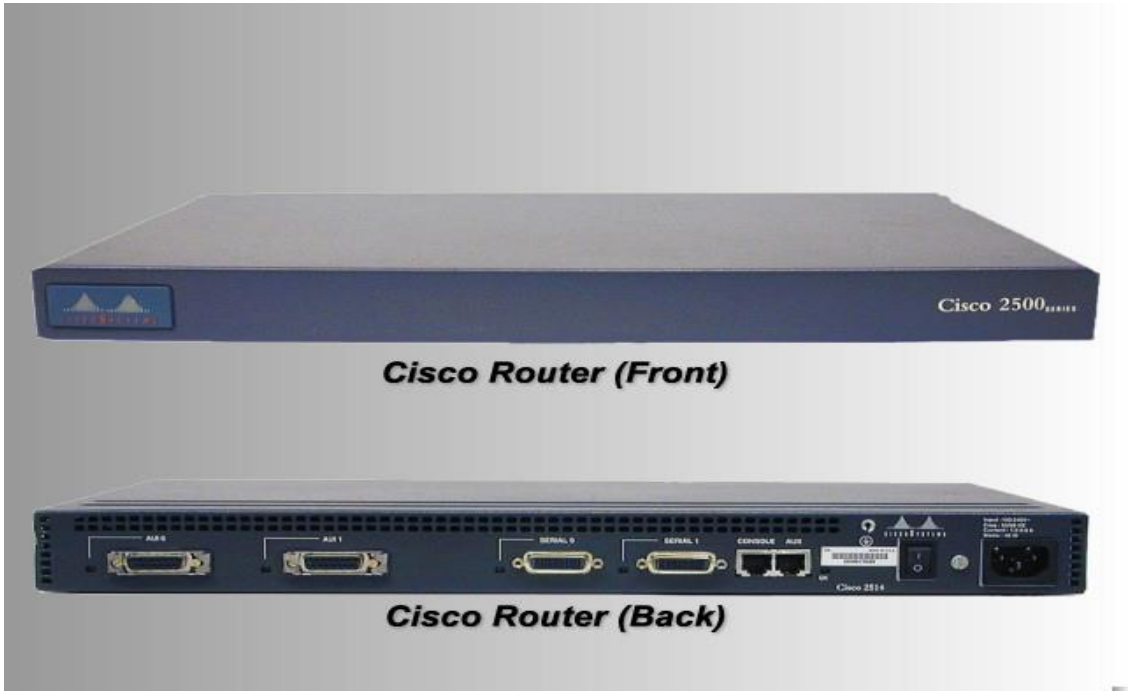
Collision domain



Network Devices- switch

- (layer 2 device)
- Knows MAC addresses

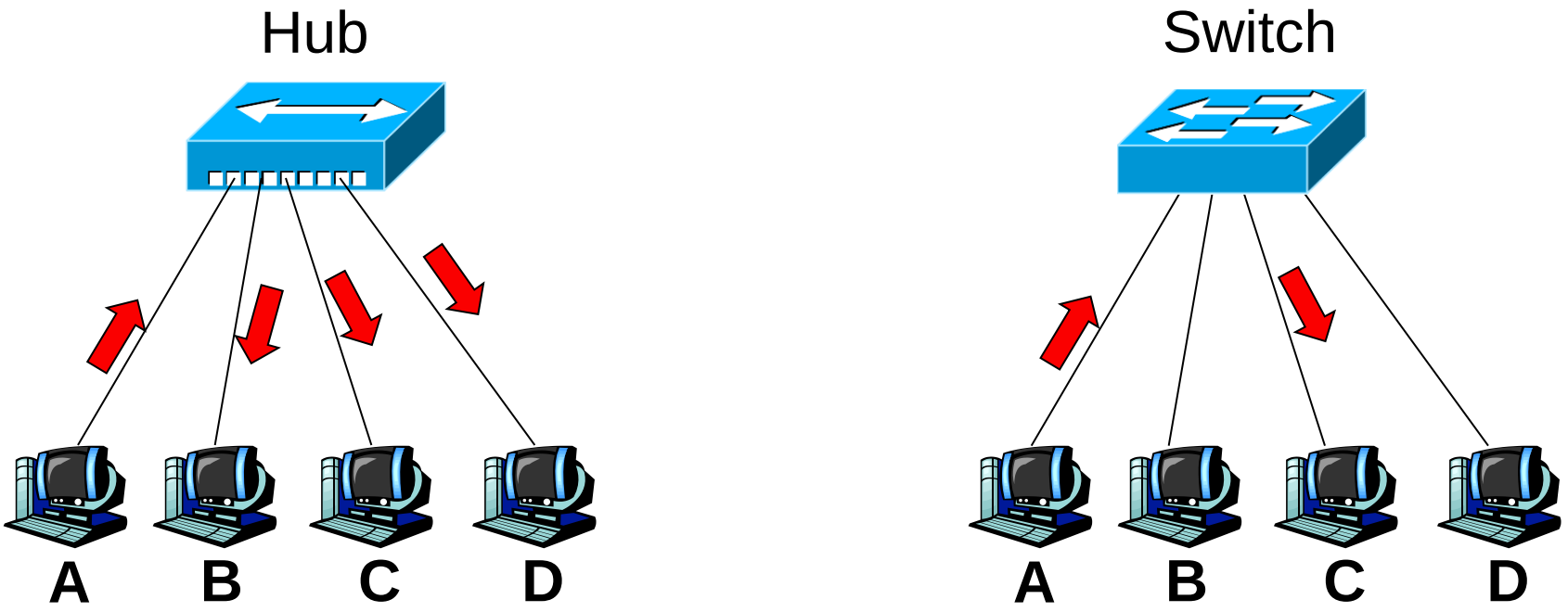
Network Devices- Router



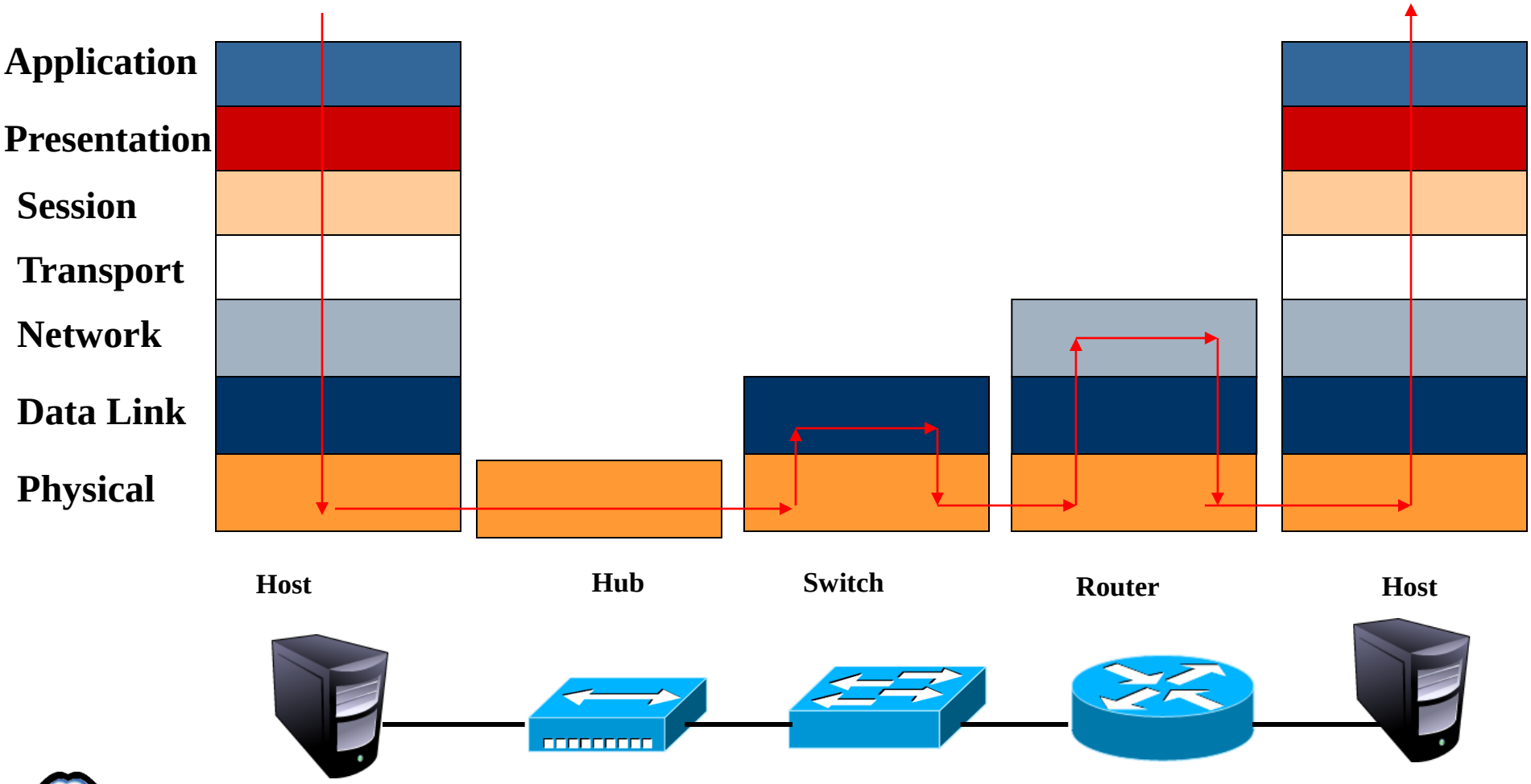
- (layer 3 device)
- Knows Logical addresses (e.g., IP)

Hub vs. Switch

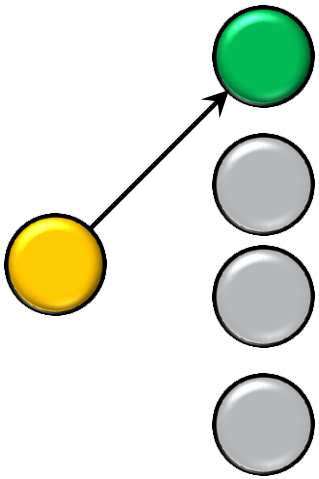
A sends to C



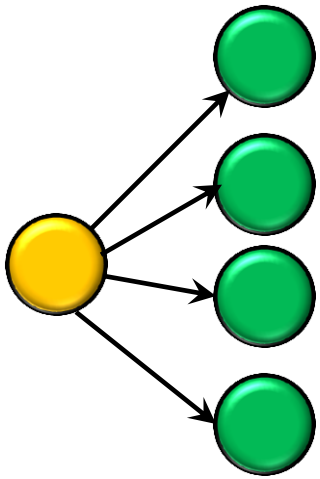
OSI Layers and Net. devices



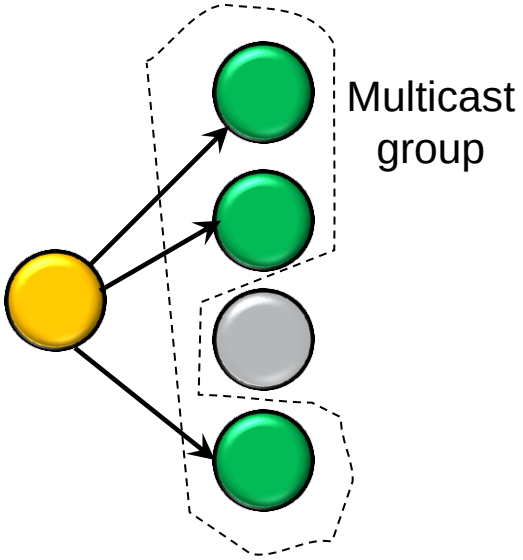
Unicast, Multicast and Broadcast



Unicast



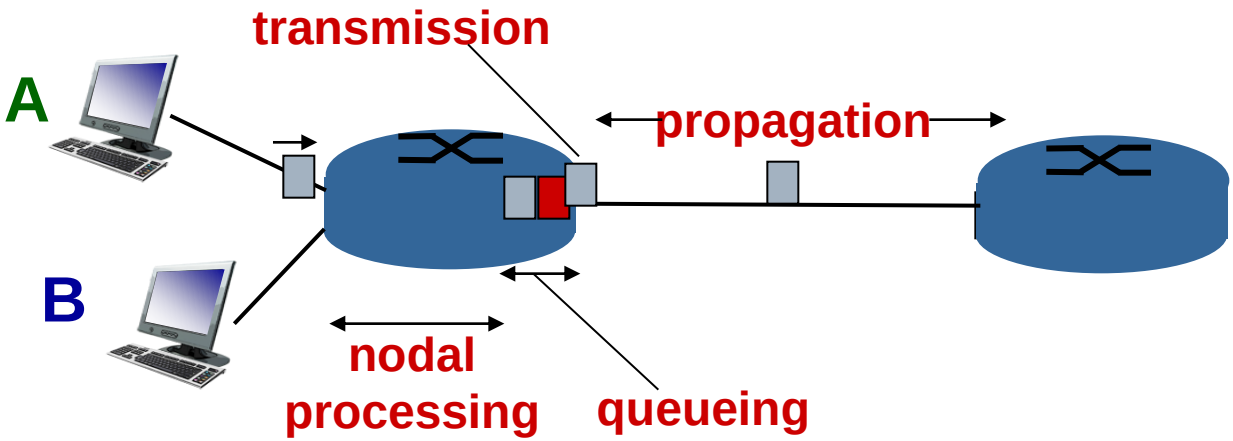
Broadcast



Multicast

Delay analysis

Four sources of packet delay



$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

Delay analysis

d_{proc}: processing delay

- check bit errors
- determine output link
- typically < msec

d_{queue}: queueing delay

- time waiting at output link for transmission
- depends on congestion level of router

d_{trans}: transmission delay:

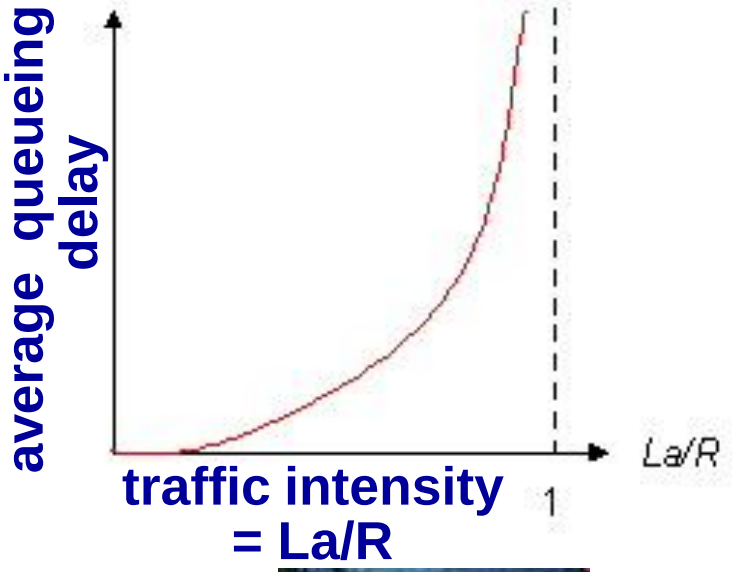
- ***L*: packet length (bits)**
- ***R*: link *bandwidth (bps)***
- $d_{trans} = L/R$

d_{prop}: propagation delay:

- ***d*: length of physical link**
- ***s*: propagation speed in medium (~2×10^8 m/sec)**
- $d_{prop} = d/s$

Queueing delay

- R: link bandwidth (bps)
- L: packet length (bits)
- a: average packet arrival rate

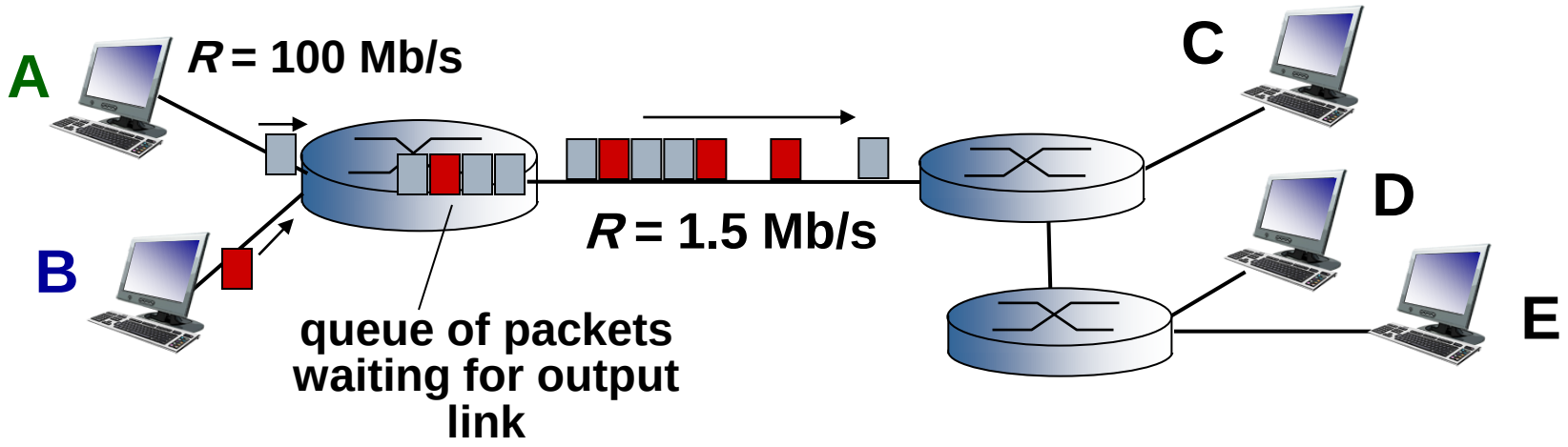


- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving than can be serviced, average delay infinite!

La/R ~ 0

La/R -> 1

Packet Switching: queueing delay, loss



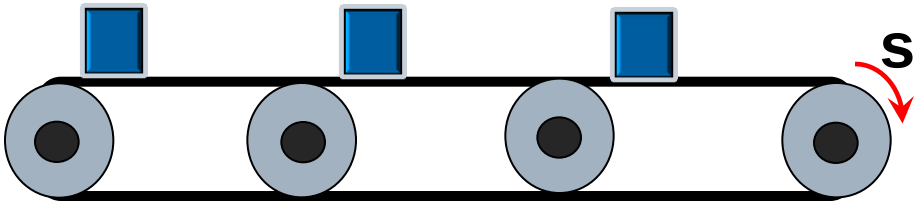
queuing and loss:

- If arrival rate (in bits) to link exceeds transmission rate of link for a period of time:
  - packets will queue, wait to be transmitted on link
  - packets can be dropped (lost) if memory (buffer) fills up

Transmission and Propagation Delays (analogy)

L

R



S

d

Transmission delay

Propagation delay


Questions