Introduction to Data Communication

Tor Skeie

Email: tskeie@ifi.uio.no

(based on slides from Kjell Åge Bringsrud and Carsten Griwodz)

Introduction

Goal

- Give an overview of the topic
- Approach
  - Descriptive
  - Use Internet as example

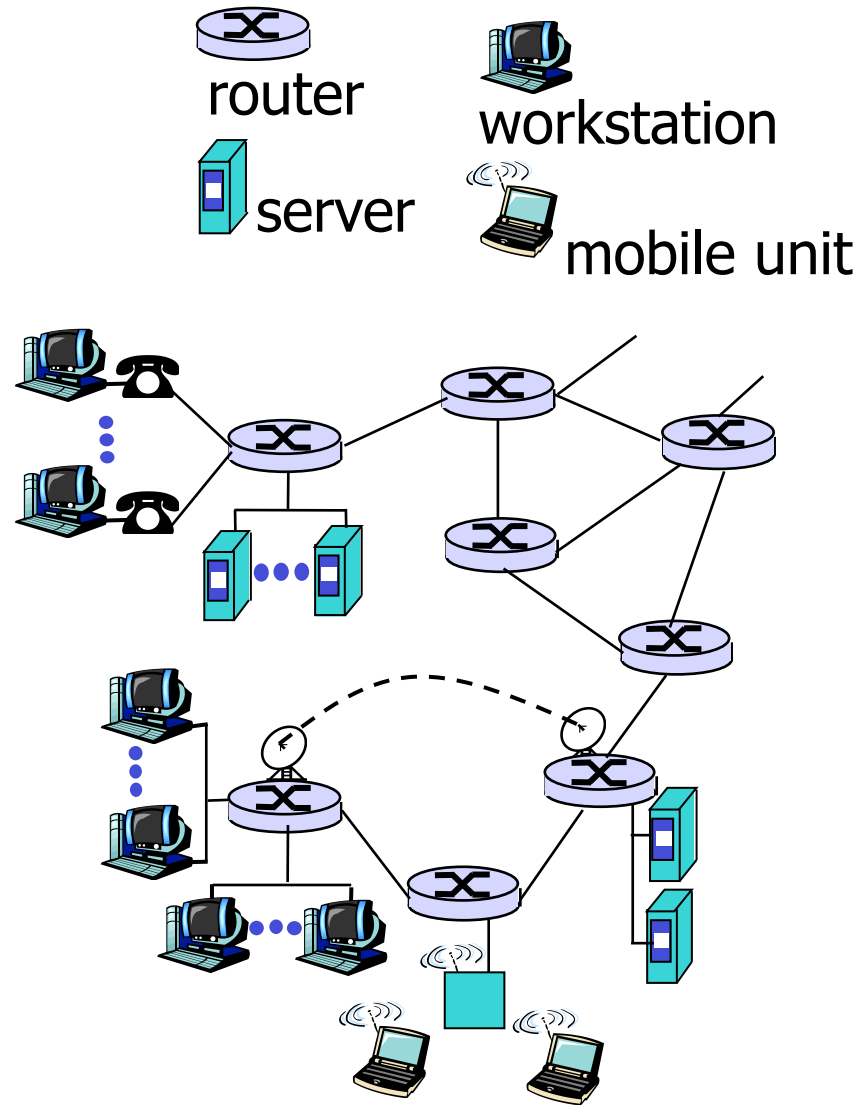
Content

- What is the Internet?
- What is a protocol?
- End systems
- Access network and physical media
- Core networks
- Throughput, delay, and loss
- Protocol layers, service models
- Backbones, NAP'er, ISP'er
- History

What is the Internet?

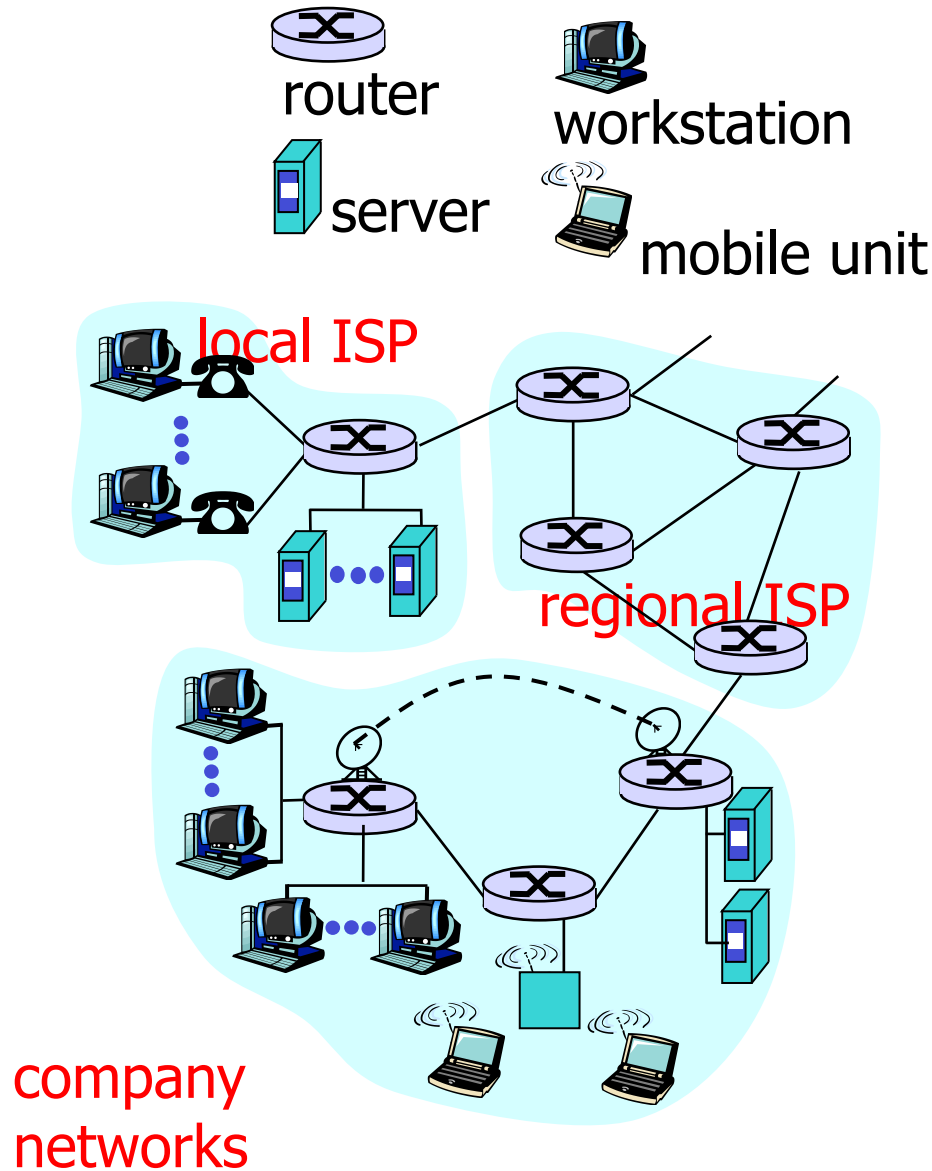
- Millions of interconnected devices: host computers, end systems
  - PCs, workstations, servers
  - PDAs, telephones, fridges … which run

    network applications
- *Communication links*
  - Fiber, copper, radio, satellite
- Routers
  - passing on packets of data through the network



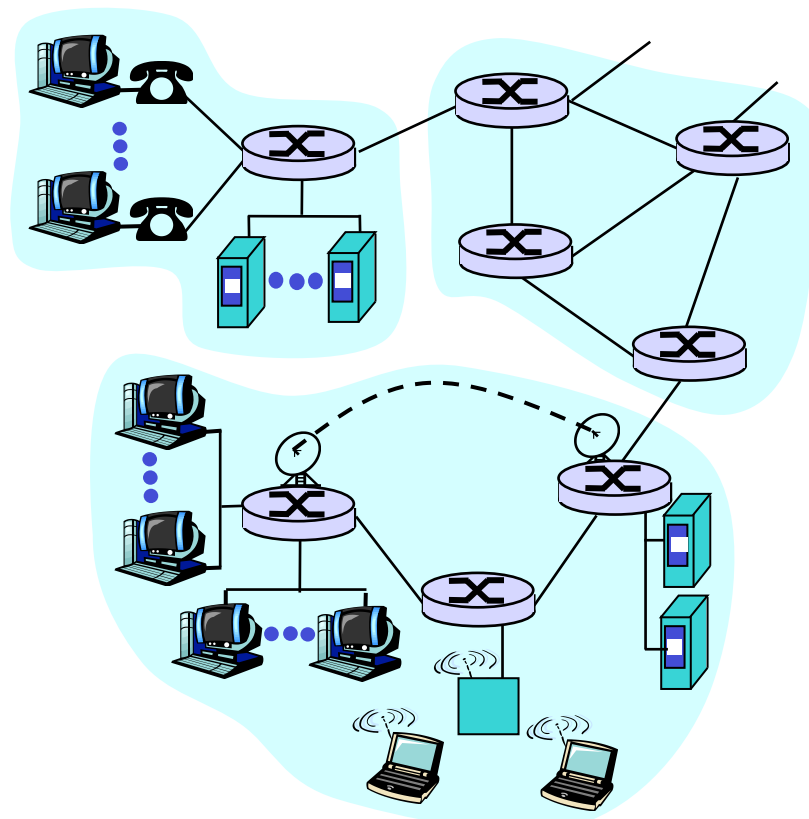
What is the Internet?

- *Internet: "network of networks"*
  - Partly hierarchical
  - ISPs: Internet Service Providers
  - Public Internet versus private intranet
- *Protocols*
  - Control sending, receiving of messages
  - E.g., TCP, IP, HTTP, FTP, PPP



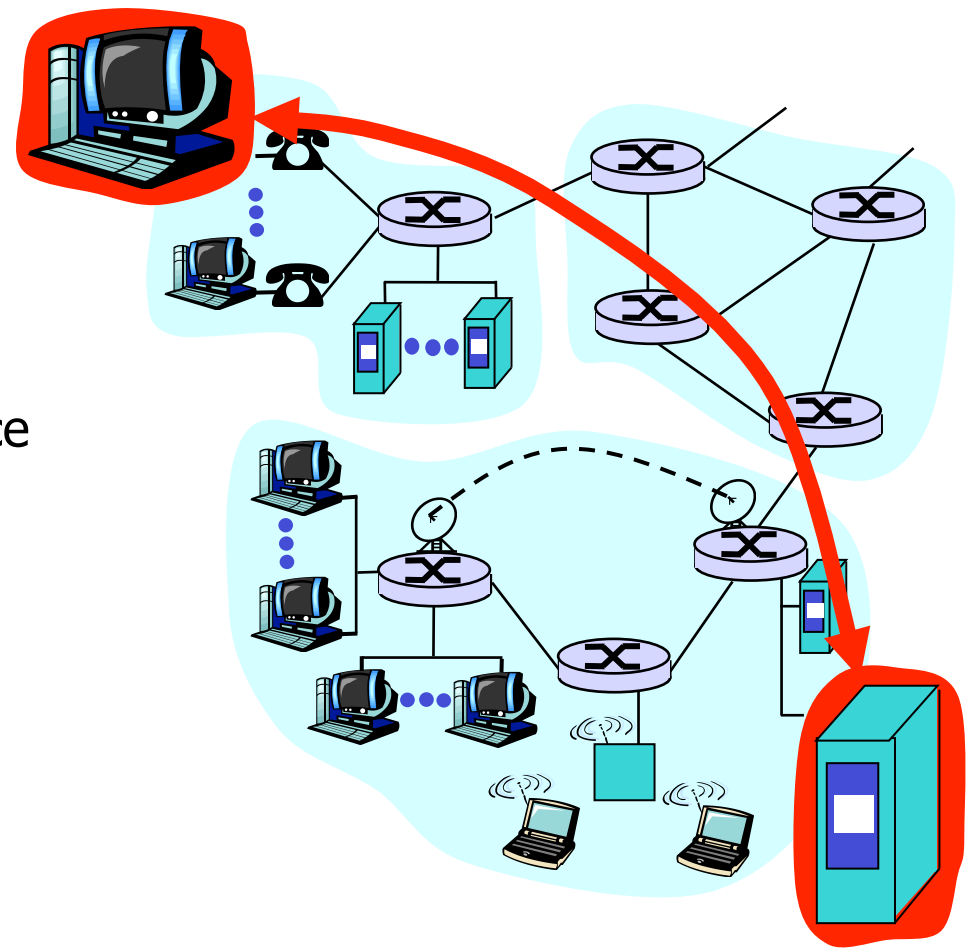
What is the Internet from a service view?

- Communication *infrastructure*
  - Allows distributed applications:
  - WWW, email, games, e-commerce, database, elections,
  - More?
- Internet standards:
  - RFC: Request for comments, e.g. TCP is RFC 793
  - IETF: Internet Engineering Task Force



End systems

- End systems
  - Run application programs
  - E.g., web browser, web server, email
  - At “the edge” of the net
- Client/server model
  - Clients ask for, and get a service from the servers
  - E.g. WWW client (browser)/ server; email client/server
- Peer-to-peer model
  - Interactions are symmetrical
  - E.g. telephone conferences



What is a protocol?

Human protocols:

- "What time is it?"
- "I have a question"
- Formal phrases…

… are special "messages" that are sent, which lead to …

… defined events or actions when the message is received

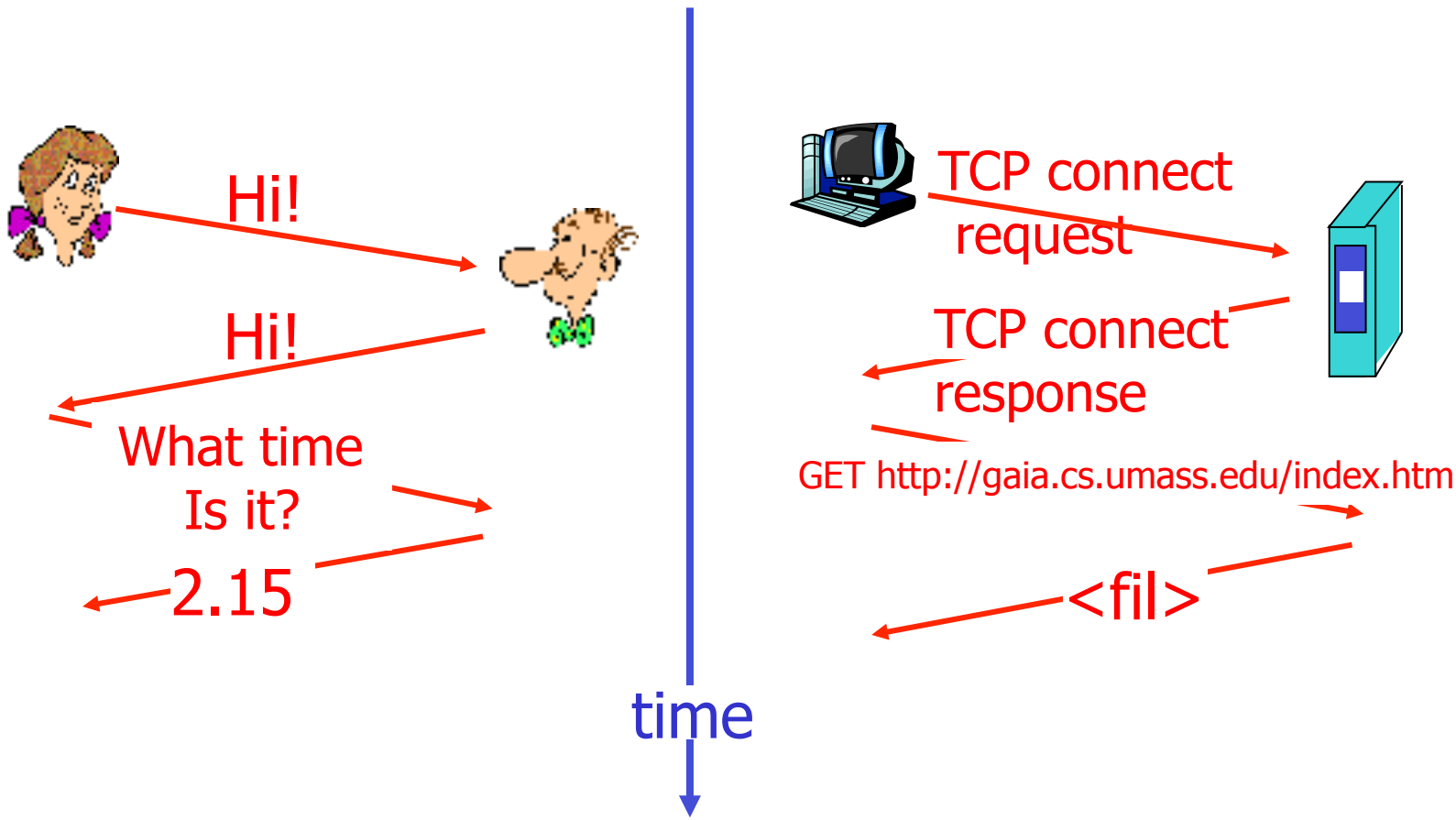
Network protocols:

- Machine instead of people
- All communication activity in the Internet is controlled by protocols

Protocols define formats, order of sending and receiving of messages, and the actions that the reception initiates.

What is a protocol?

A human protocol and a computer protocol:



What are protocol layers?

Several layers of communication

Snakker du norsk?

Sprechen Sie Deutsch?

Do you speak English?

Yes!

Use the language for all further messages!

What's your name?

Peter

Use name in further messages now!

Peter, have you met Paul?

...

time

What are protocol layers?

Networks are complex

- Many parts:
  - Hardware, software
  - End systems, routers
  - Links of different kinds
  - Protocols
  - Applications

Question:

Is it possible to *organize* the structure of a network?

Or at least our discussion of networks?

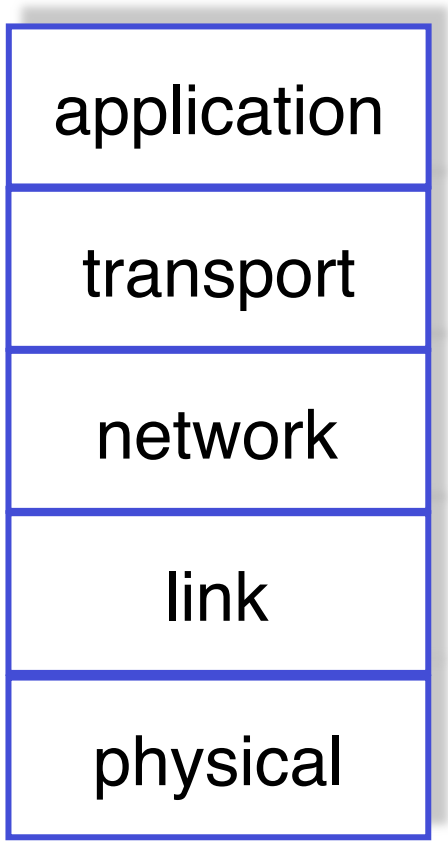
Why layering?

Management of complex systems:

- Modularisation simplifies
  - Design
  - Maintenance
  - Updating of a system
- Explicit structure allows
  - Identification of the individual parts
  - Relations among them
- Clear structure: *layering*
  - Layered reference model
  - Goal: different implementation of one layer fit with all implementations of other layers

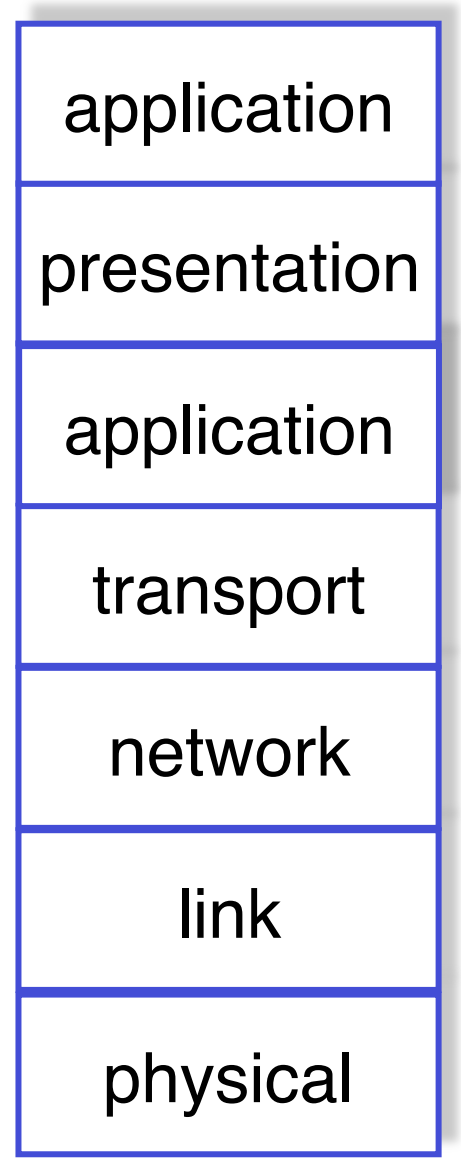
TCP/IP - protocol stack

- application: supports network applications
  - ftp, smtp, http
  - Your applications
- transport: data transfer from end system to end system
  - TCP, UDP
- network: finding the way through the network from machine to machine
  - IP
- (data) link: data transfer between two neighbors in the network
  - ppp (point-to-point protocol), Ethernet
- physical: bits "on the wire"



OSI - model

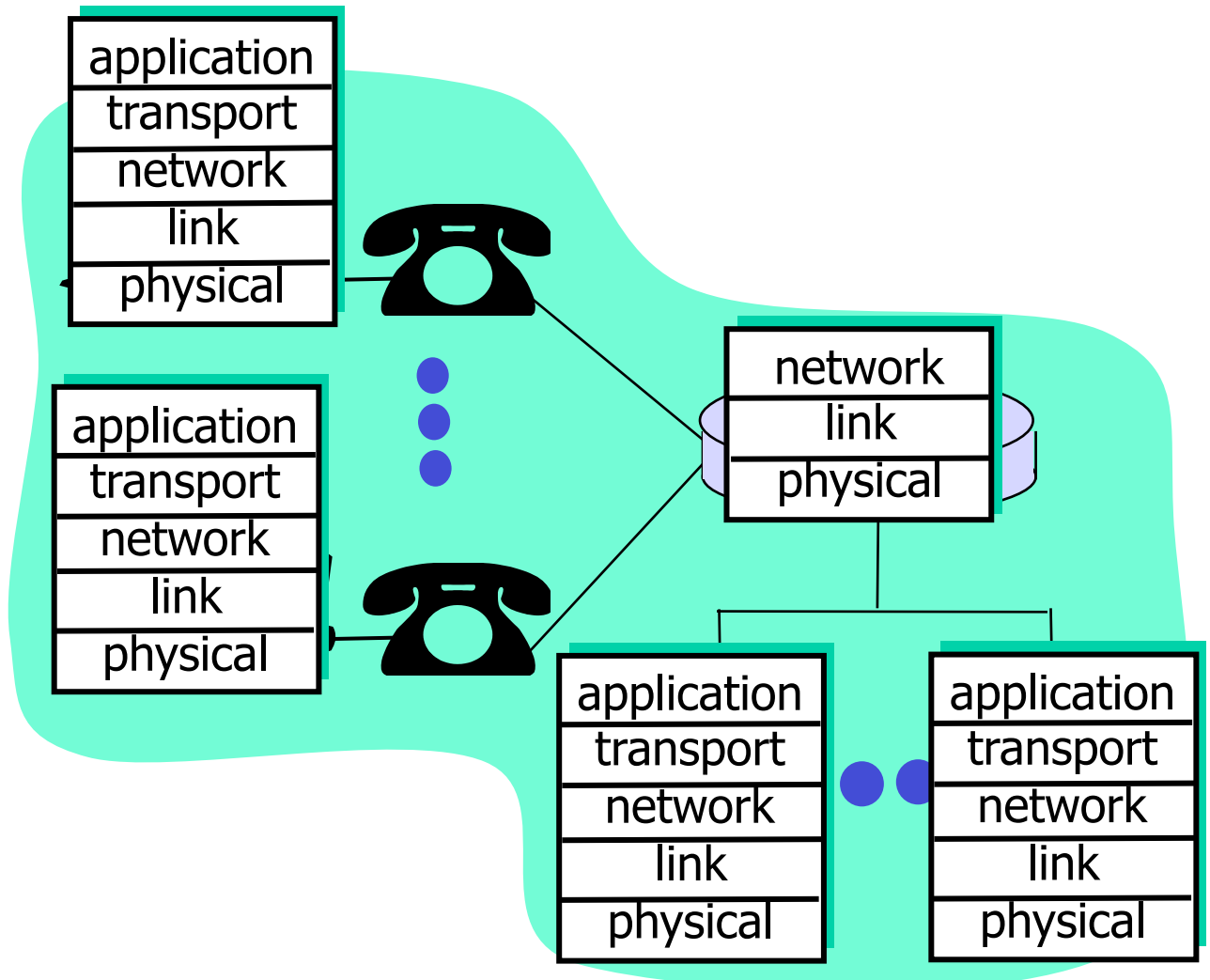
- A standard for layering of communication protocols
  - Open Systems Interconnection
  - by the ISO – International Standardization Institute
- Two additional layers to those of the Internet stack
- presentation: translates between different formats
  - XML, XDR
  - provides platform independence
- session: manages connection, control and disconnection of communication sessions
  - RTP



Layering: logical communication

Each layer:

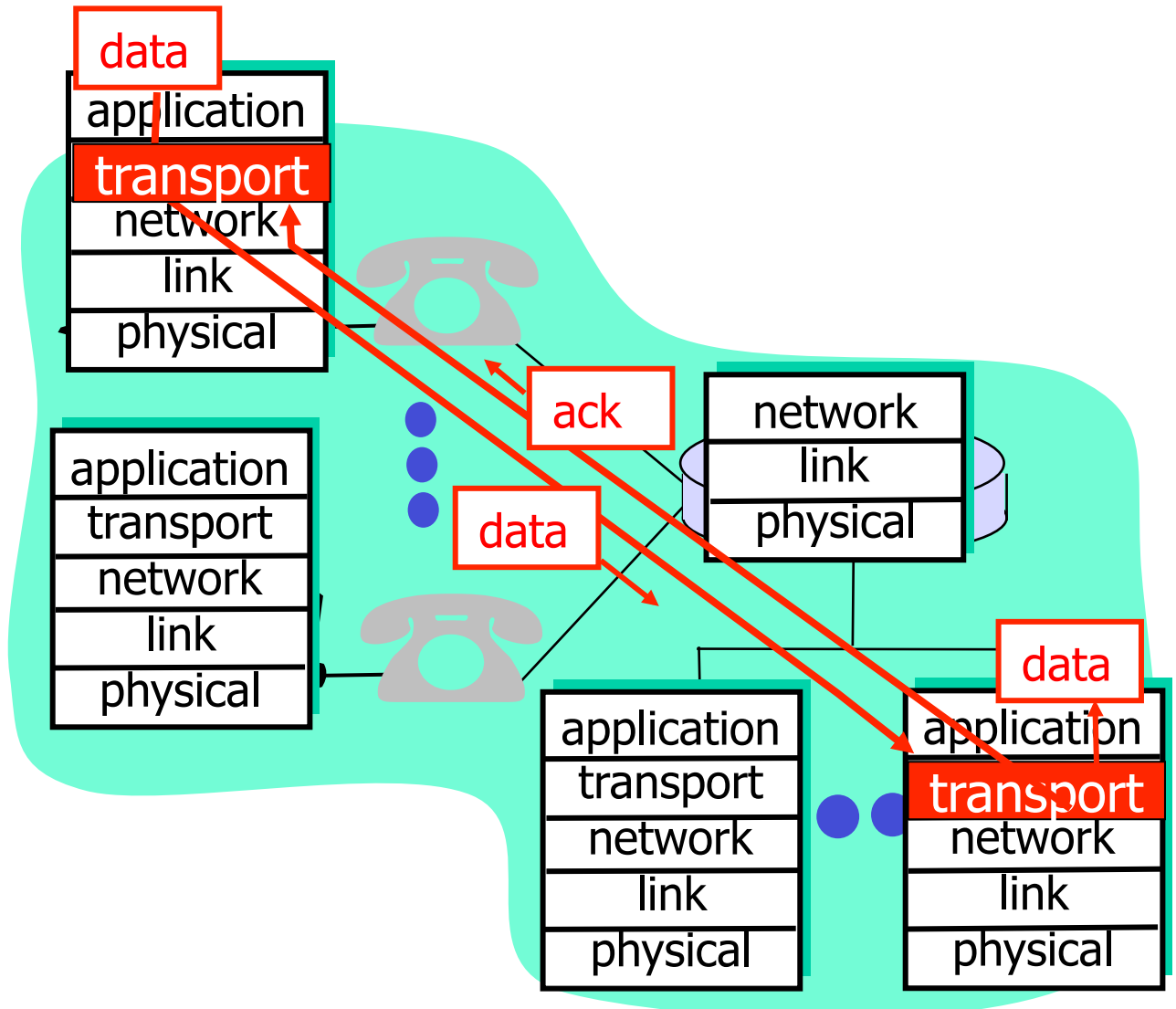
- distributed
- "units" implement functionality of each layer in each node
- Units execute operations, and exchange messages with other units of the same layer



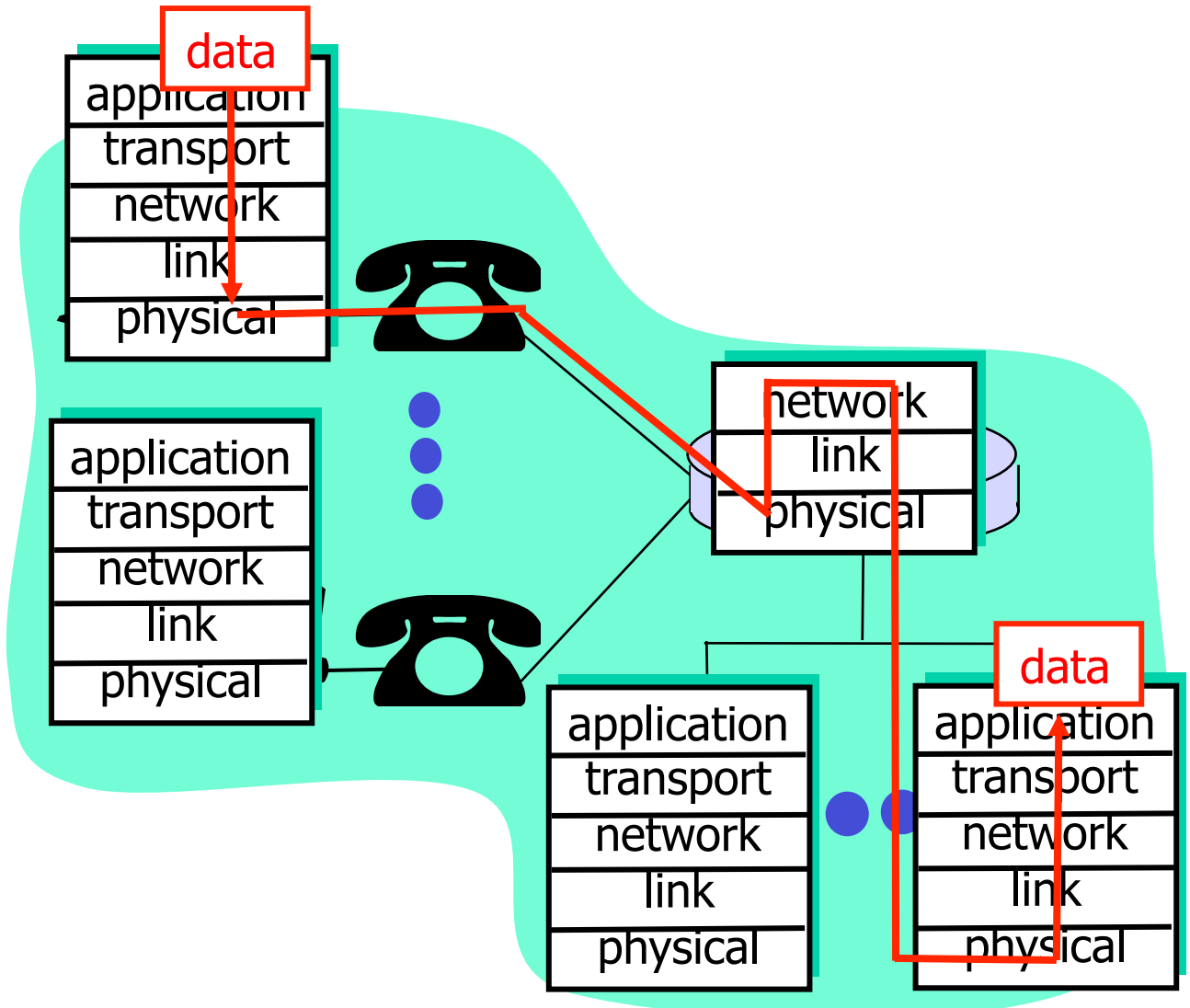
Layering: *logical* communication

E.g. transport

- Receive data from the application
- Add receiver address, reliability check, information to create a "datagram"
- Send datagram to the transport layer in the receiver node
- Wait for "ack" from the transport layer in the receiver node
- Analogy: post office



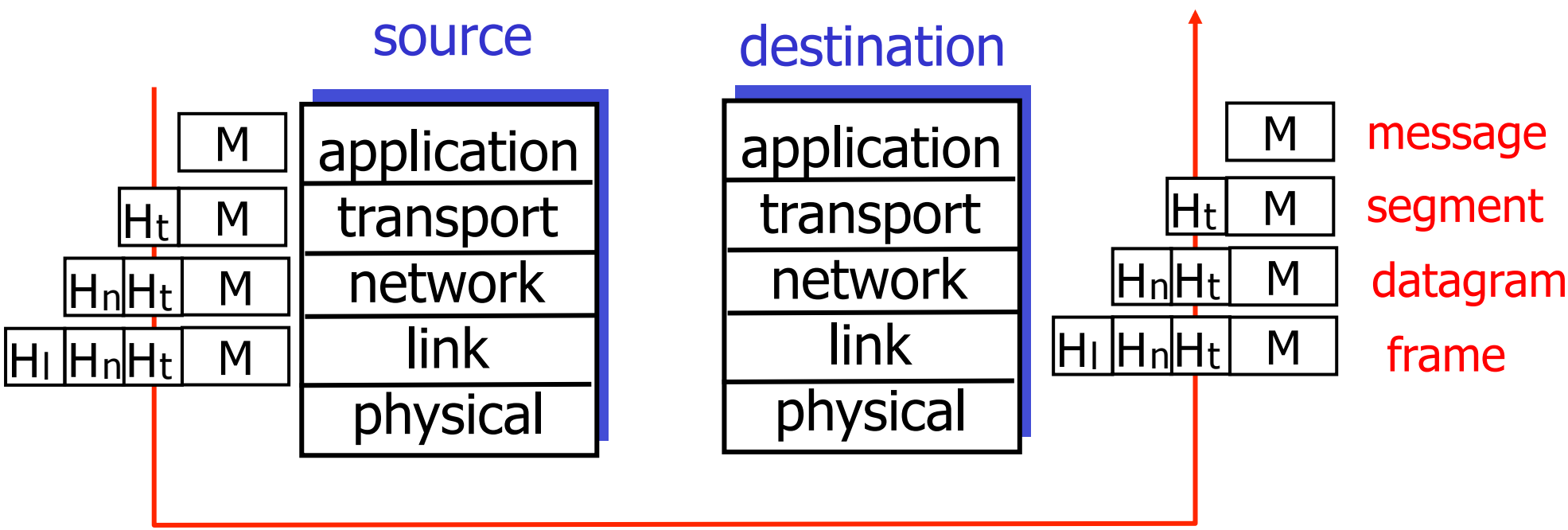
Layering: physical communication



Protocol layer and data

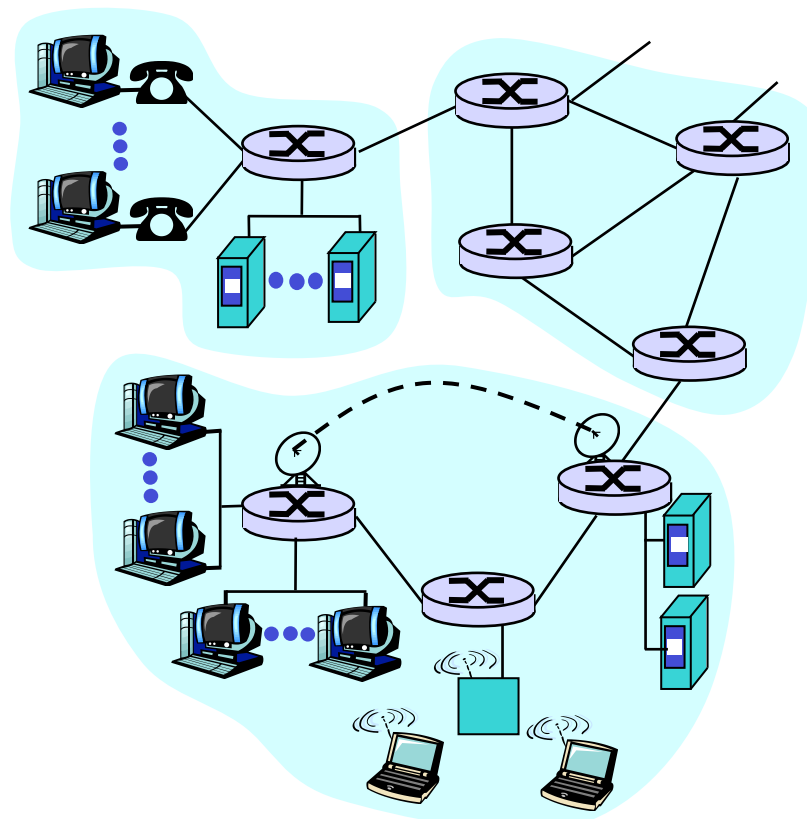
Each layer takes data from next higher layer

- Adds header information to create a new data unit (message, segment, frame, packet …)
- Send the new data unit to next lower layer



A closer look at network structures

- End systems
  - applications and host computers
- Access network, physical medium
  - Communication links
- Core networks
  - Routers
  - Network of networks



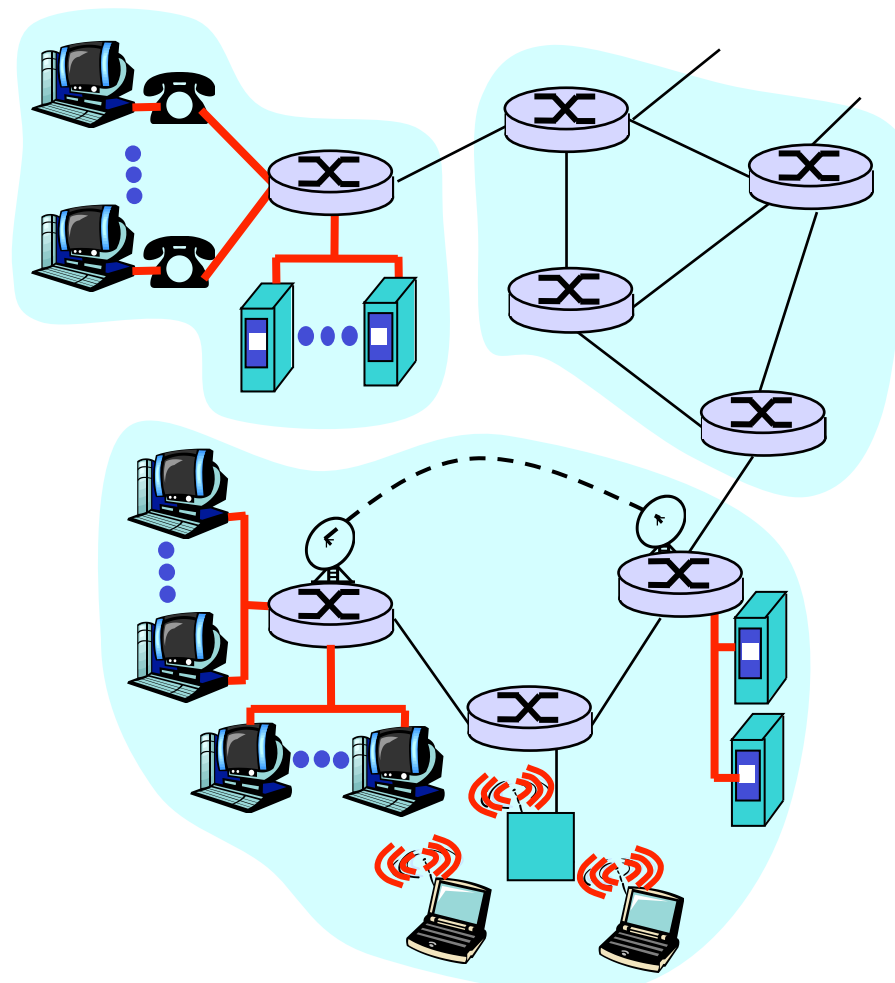
Access network and physical media

How to connect end systems to edge routers?

- Home network
- Company network (schools, companies)
- Mobile access network

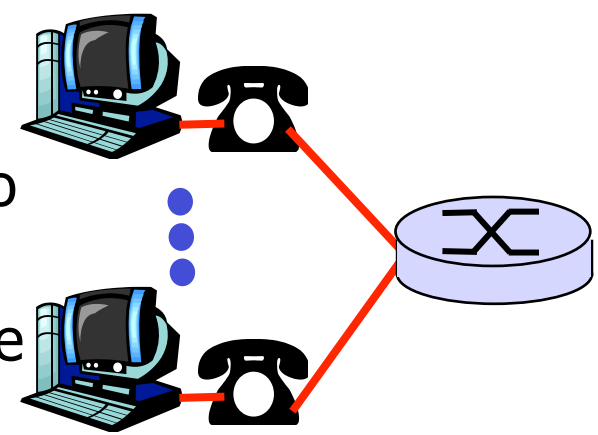
Keep in mind when choosing a technology:

- Bandwidth?
- Shared or dedicated medium?



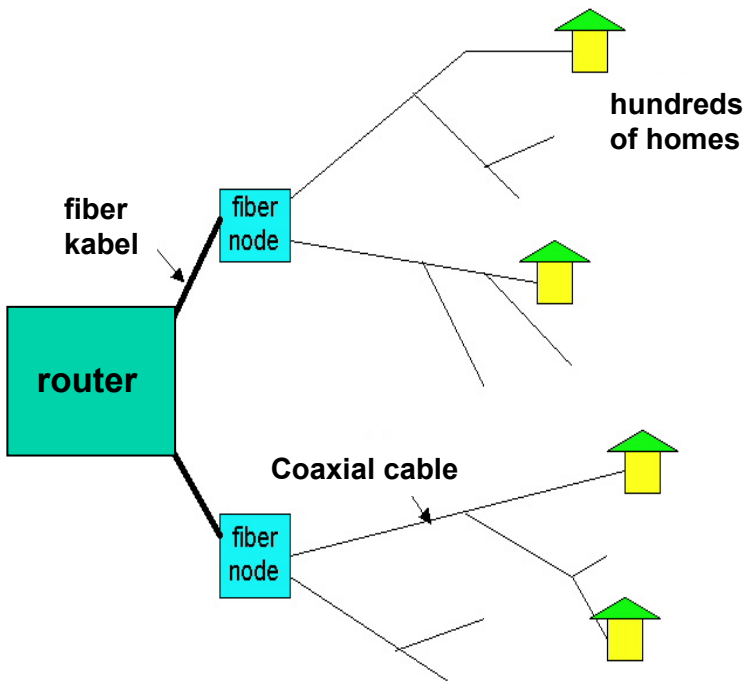
Home network: point to point

- Dial-up via modem
  - Up to 56Kbps direct access to the router (at least in theory)
- ISDN: integrated services digital network
  - 128Kbps purely digital connection to the router
- ADSL: asymmetric digital subscriber line
  - Up to 5 Mbps *uplink* (home-to-router, ver. ADSL2++, in development)
  - Up to 52 Mbps *downlink* (router-to-home)



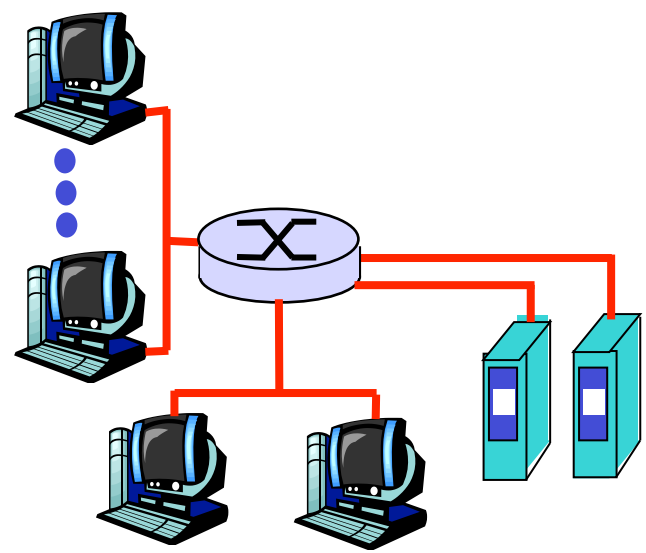
Home network: Broadband An example

- HFC: hybrid fiber coax
  - Asymmetrical: e.g. 25 Mbps downlink, 5 Mbps uplink
- Network of copper cable and optical fiber connects homes to ISP routers
  - Shared access to router for several homes
  - Problems: congestion, dimensioning



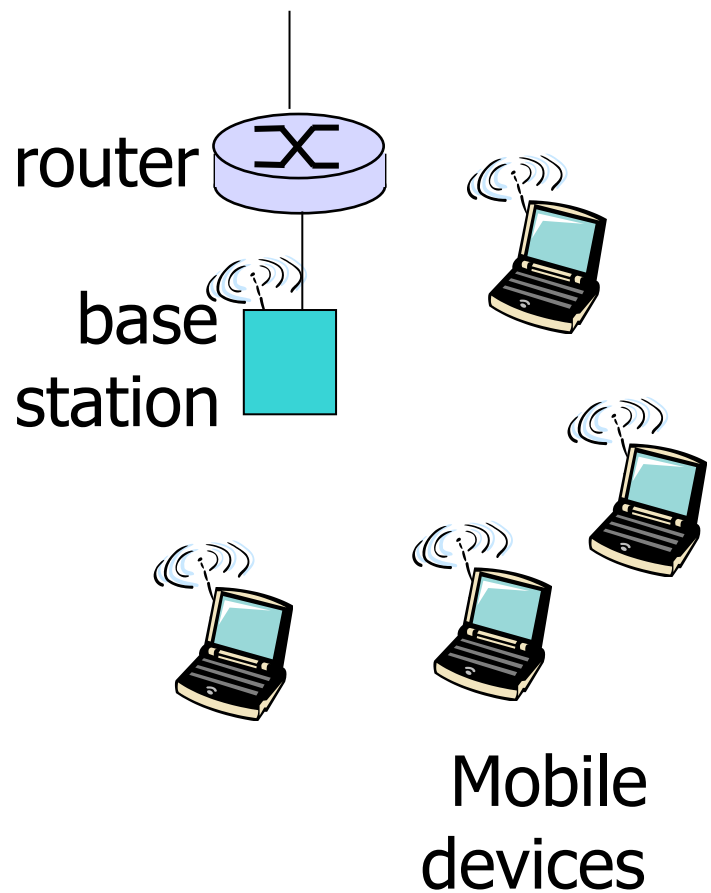
Institutional access networks (LAN)

- Company/university local area network (LAN) connects end systems to the rest of the net
- Ethernet:
  - Shared or dedicated cable connects end systems and routers
  - 10 Mbps, 100Mbps, Gigabit Ethernet



Wireless access networks

- Shared wireless access networks connect end systems to routers
- Wireless LANs:
  - radio spectrum replaces cable
  - E.g.
    - IEEE 802.11g - 54 Mbps
    - IEEE 802.11h – 100 Mbps
    - IEEE 802.11n – 600Mbps
    - IEEE 802.11ac – 1,3Gbps
- Wireless access over long distances
  - 3G/4G for example…

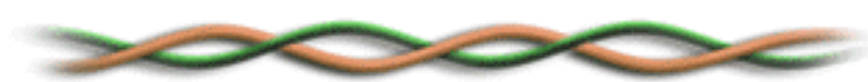


Physical medium

- Physical link: a sent bit propagates through the link
- Closed media:
  - Signals propagate in cable media (copper, fiber)
- Open media:
  - Signals propagate freely, e.g. radio.

Twisted Pair (TP)

- Two isolated copper cables
  - Category 3: traditional telephone cables, 10 Mbps Ethernet
  - Category 5 TP: 100Mbps Ethernet
  - Category 6 TP: 1Gpbs Ethernet



Physical medium: coax, fiber

Coaxial cable

- Wire (signal carrier) in a wire (shielding)
  - baseband: a single channel on a cable
  - broadband: multiple channels on a cable
- Bi-directional
- Typically used for 100Mbs Ethernet.

Fiber optic cable

- Optical fiber that carries light impulses
- High-speed transfer:
  - High-speed point-to-point transmission
- Low error rate
- Longer distances
  - 100Mbps, 1-100Gbps Ethernet

Physical media: radio

Radio

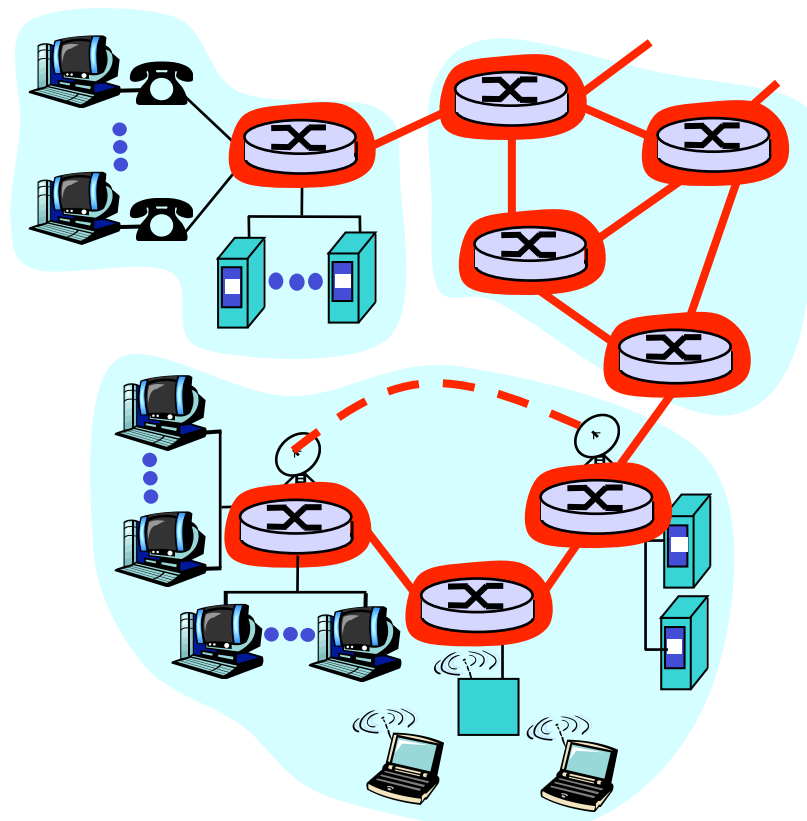
- Signal in electromagnetic spectrum
- No physical ”cable”
- Bi-directional
- Effects of environment on the distribution:
  - Reflection
  - Obstruction by blocking objects
  - Interferences

Types of radio links

- microwaves
  - E.g. up to 45 Mbps
- WLAN
  - 54Mbps, 600Mbps, 1,3Gbps
- wide-area
  - 3G, 14,4Mbps (in theory)
- satellite
  - Up to 50Mbps per channel (or several thinner channels)
  - 270 ms end-to-end delay (limited by speed of light).

Core networks

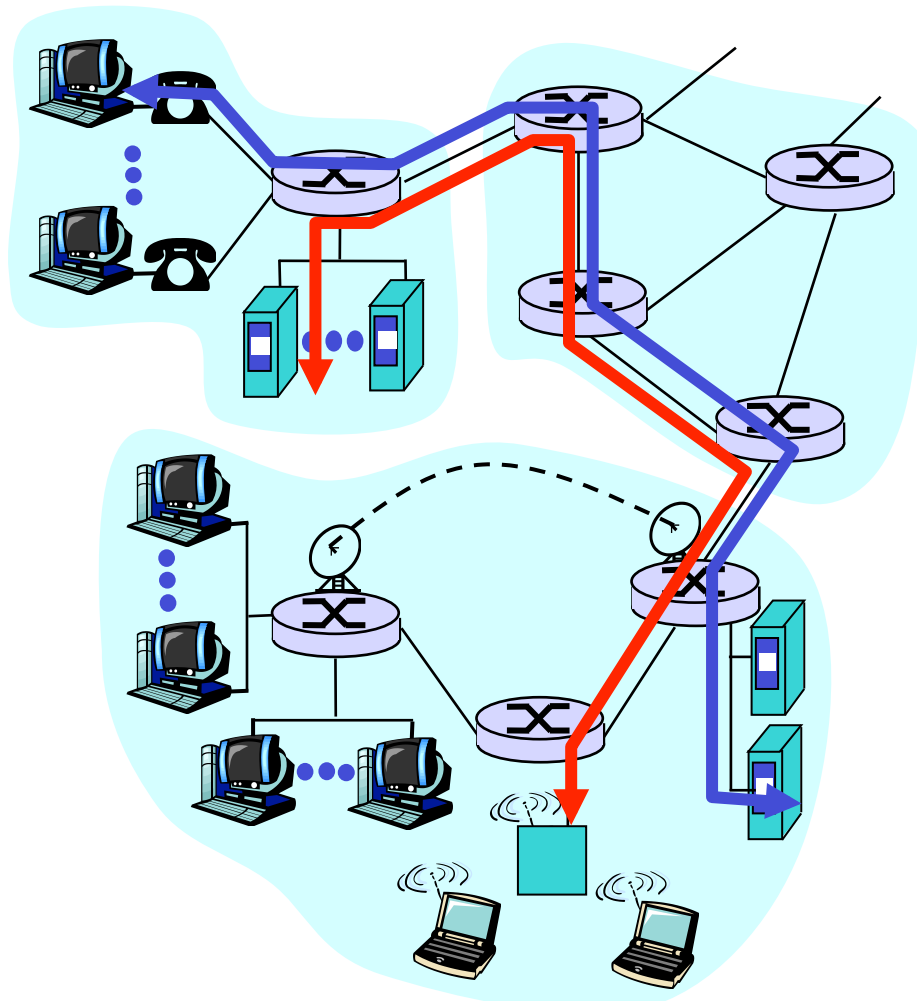
- Graph of interconnected routers
- *One* fundamental question: how is data passed through the net?
  - Circuit switching
  - Packet switching
- Circuit switching
  - Dedicated line through the network
- Packet switching
  - Discrete *data units* are sent through the network



Core networks: Circuit Switching

End-to-end resource reservation for a "session"

- Setup phase is required
- Dedicated resources (no sharing)
- Link bandwidth, router capacity
- Guaranteed throughput



Core networks: Circuit Switching

Historical:

- Analog telephone networks
- Network consists of resources
  - Cables
  - Switches with relays
- Establish a physical connection
  - Relays switch to connect cables physically
  - Create a circuit
  - Guaranteed resources
  - No difference between talking and silence

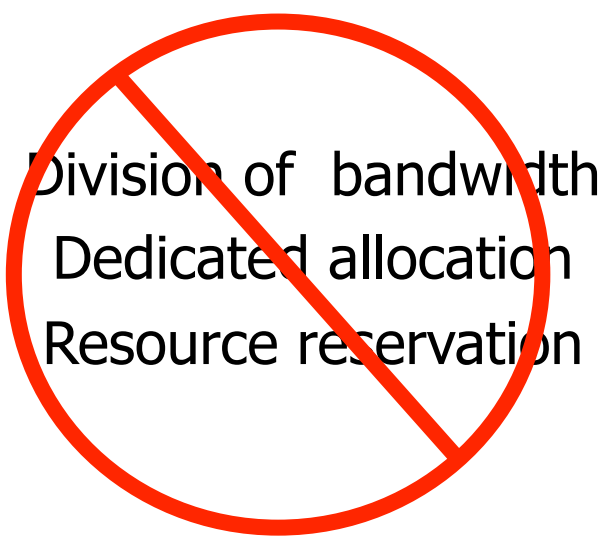
Modern:

- Networks consist of resources
  - Cables
  - Routers or switches
  - Network resources can be shared
- Establish a connection
  - Switches reserve part of available resource
- Division of link bandwidth into parts
  - Frequency division
  - Time division

Core networks: Packet Switching

Each end-to-end data stream is divided into packets

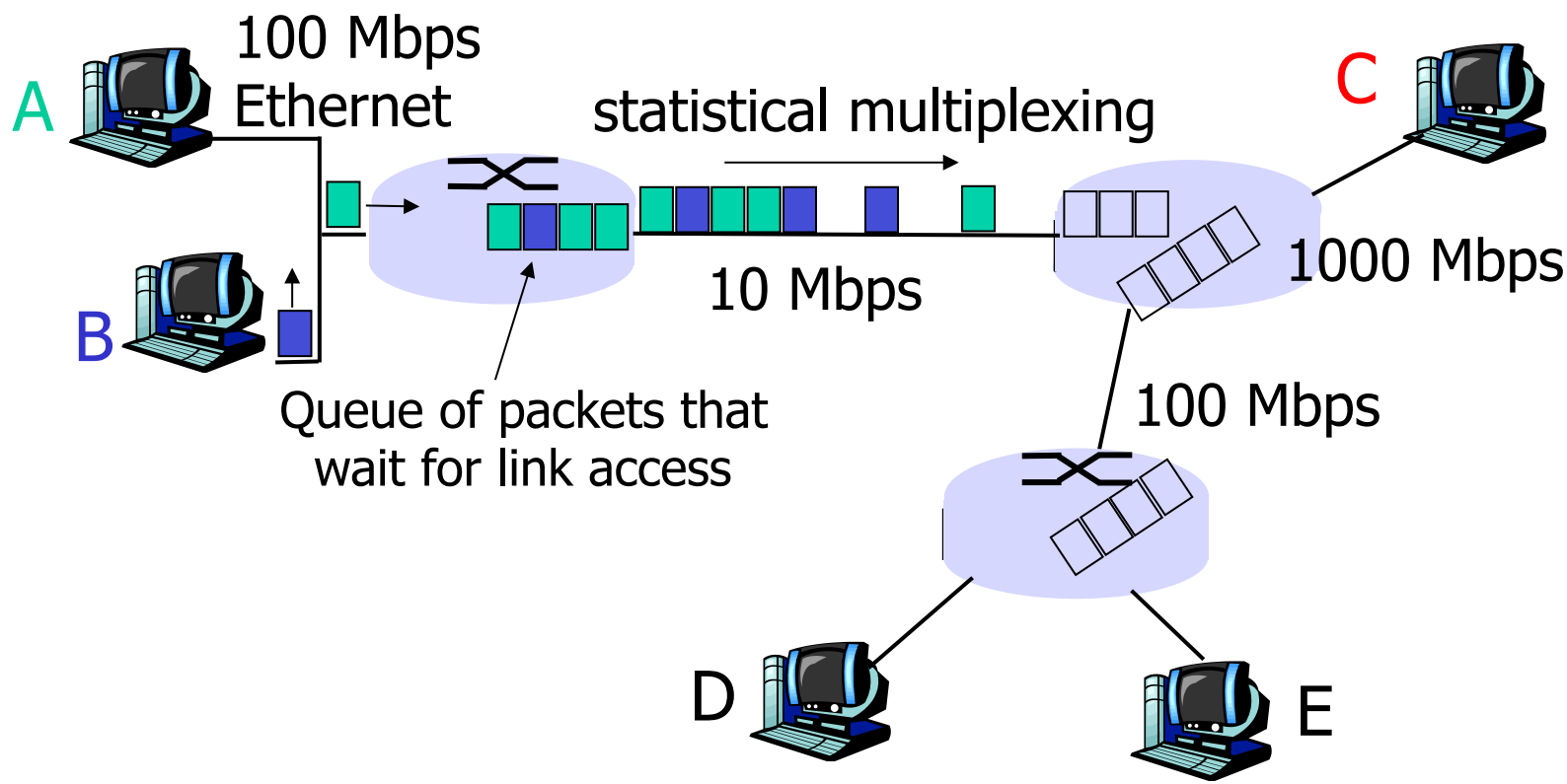
- Data streams *share* network resources
- Each packet uses the entire bandwidth of a link
- Resources are used as needed



Competition for resources:

- Combined resource need can exceed the available resources
- Congestion: packets are queued in front of "thin" links
- Store and forward: packets move one link at a time
  - Send over a link
  - Wait for your turn at the next link

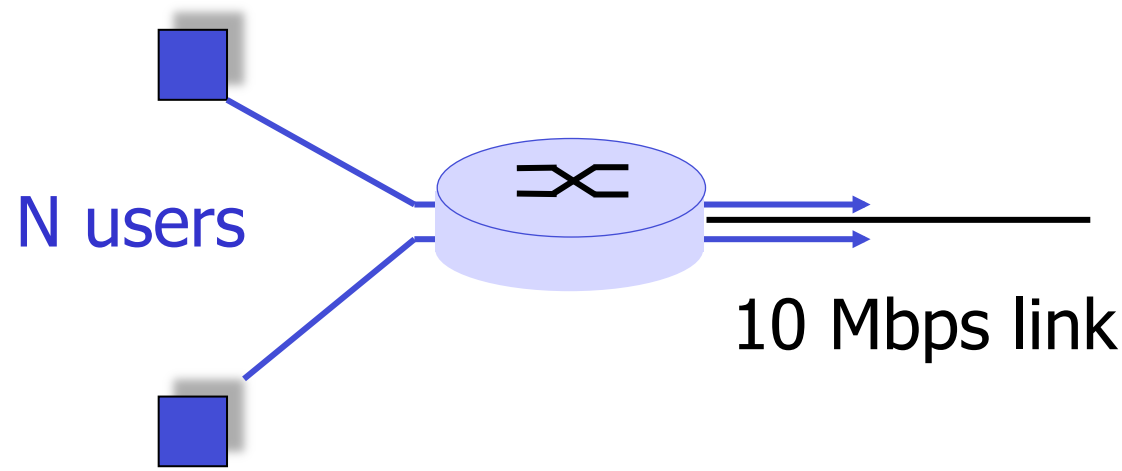
Core networks: Packet switching



Packet switching versus circuit switching

Packet switching allows more users in the net!

- 10 Mbps link
- Each user
  - 1Mps when “active”
  - Active 10% of the time, at random times
- Circuit switching
  - max 10 users
  - Loss probability: 0%
  - Waste: ~90% capacity
- Packet switching
  - >10 may be active concurrently!
  - Loss probability >0%
  - Waste: < 90% capacity



Packet switching versus circuit switching

Is packet switching always the best approach?

- Good for data with “bursty” behavior
  - Resource sharing
  - No ”setup phase” required
- In a congested network: delay and packet loss
  - Protocols/mechanisms required for reliable traffic and congestion control
- How to achieve a behavior like that of circuit switching?
  - Bandwidth guarantees are required for audio/video applications

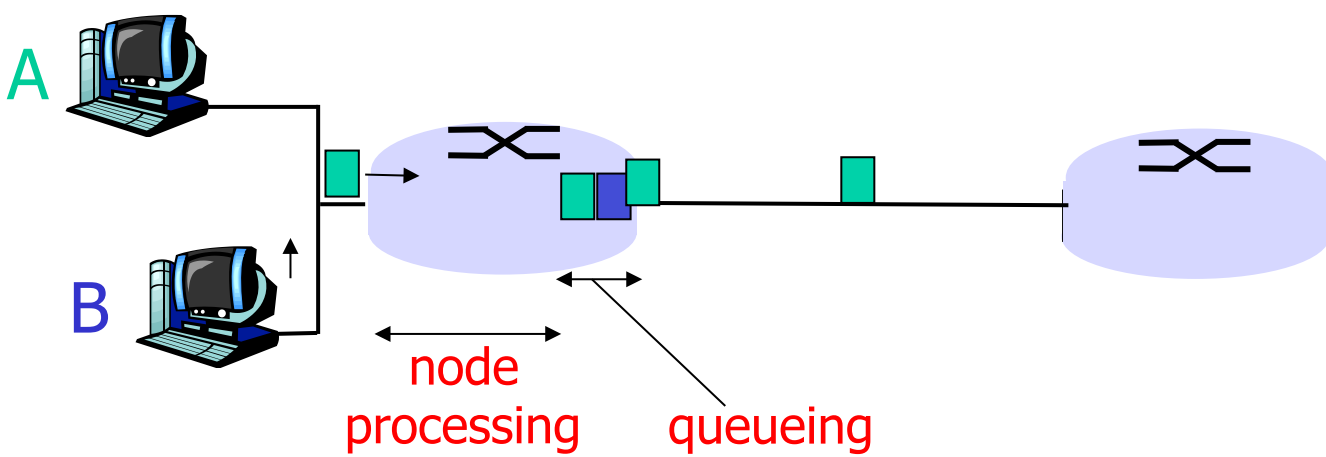
QoS concepts have to be used for that purpose!

Delay in packet switching networks

Packet experiences delay on the way from sender to receiver

- four sources of delay in each hop.

- Node processing:
  - Determining the output link – address lookup
  - Checking for bit errors
- Queuing
  - Waiting for access to the output link
  - Depends on the congestion level of the router



Delay in packet switching networks

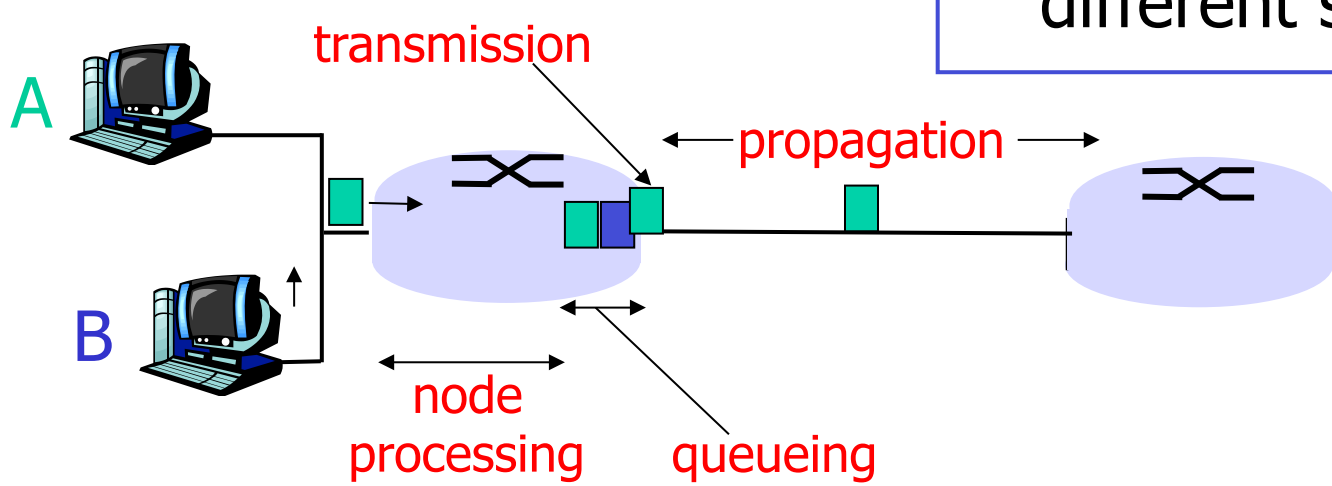
Transmission delay:

- R = link bandwidth (bps)
- L = packet size (bits)
- Time required to send a packet onto the link = L/R

Propagation delay:

- d = physical link length (m)
- s = propagation speed in the medium (~$2x10^8$ m/sec)
- Propagation delay = d/s

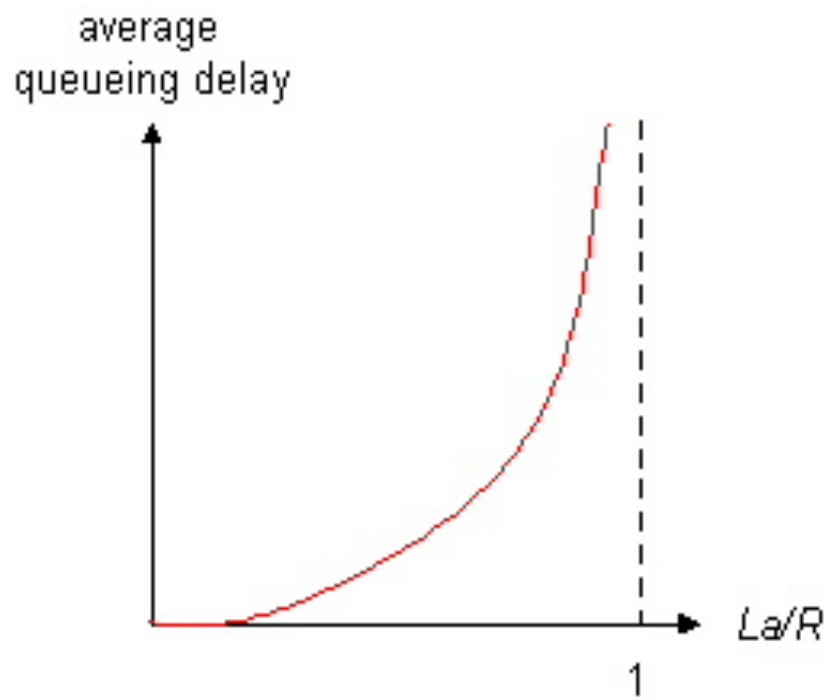
Note: s and R are of very different size!



More about queueing delays

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

traffic intensity = La/R

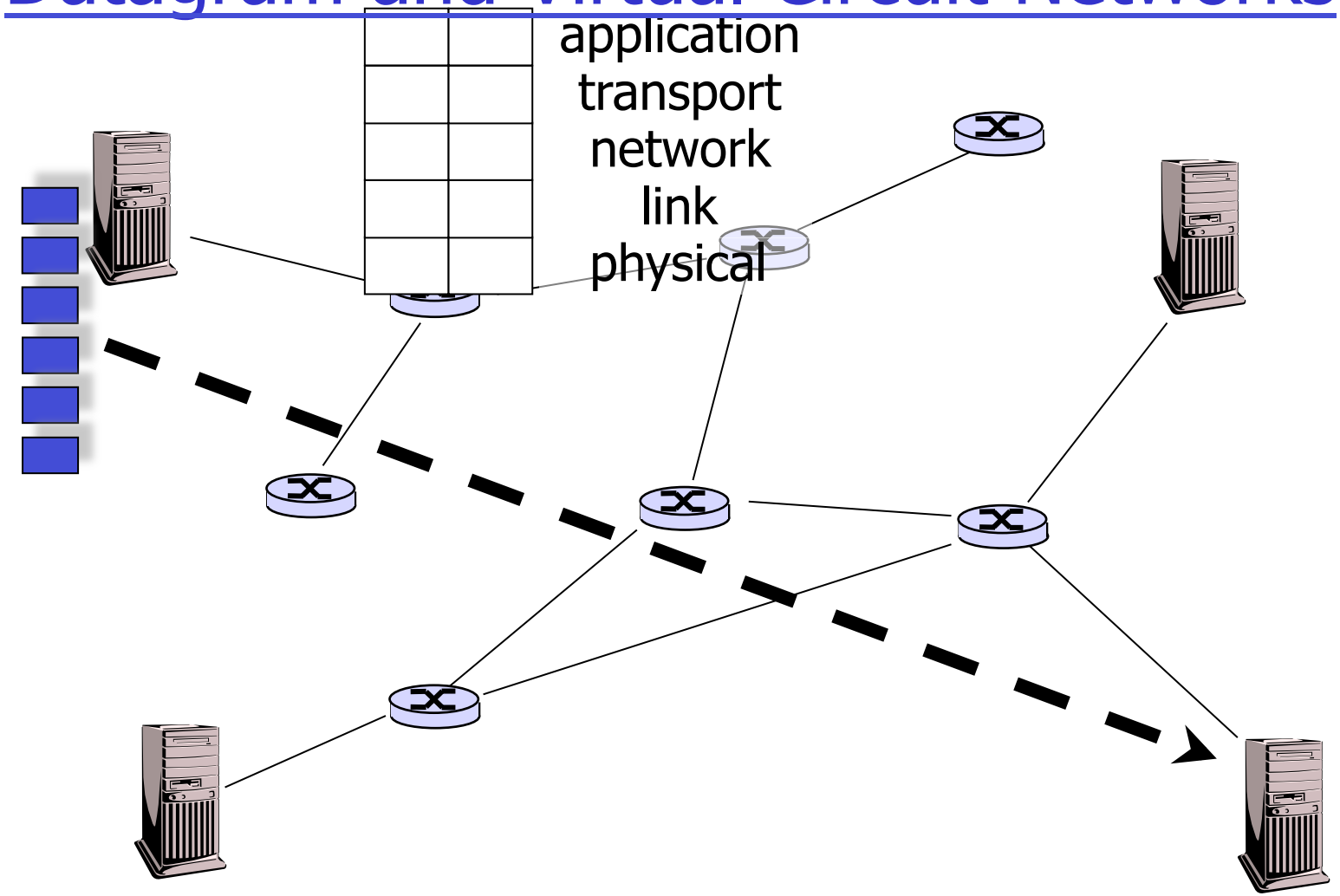


- La/R ~ 0: average queuing delay is small
- La/R -> 1: queuing delay grows
- La/R > 1: more data is arriving at the link than it can handle → link goes into congestion (Average delay is infinite!)

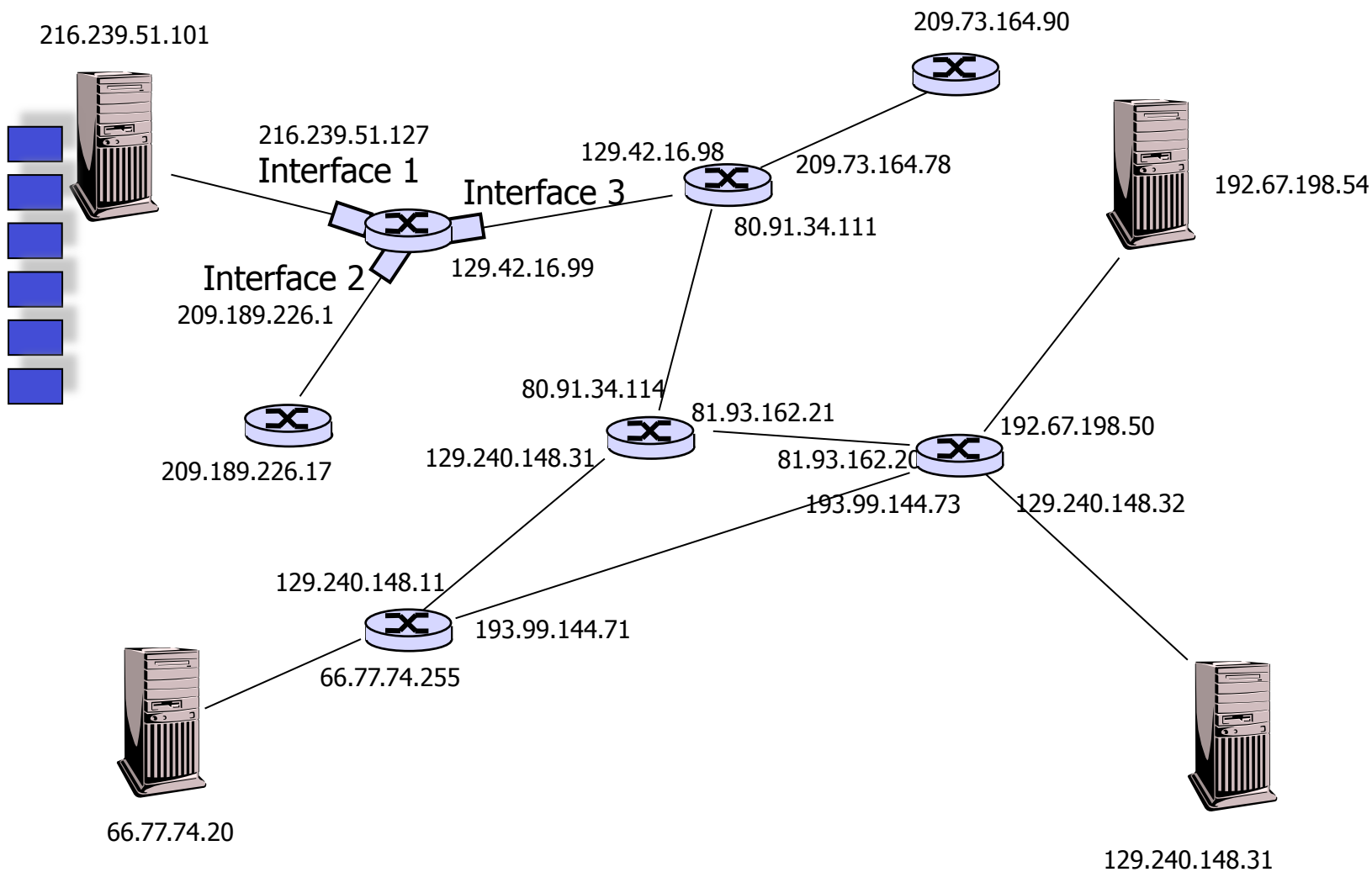
Packet switched network: Routing

- *Goal:* move packets from router to router between source and destination
  - There are two methods to find the path of packets.
- Datagram network:
  - *Destination address* determines the next hop.
  - Path can change during the sessions.
  - Routers need no information about sessions.
  - Analogy: ask for the way while you drive.
- Virtual circuit network:
  - Each packet has a "tag" (virtual circuit ID), which determines the next hop.
  - Path is determined when connection is set up, and remains the same for the entire session.
  - Routers need state information for each virtual circuit.

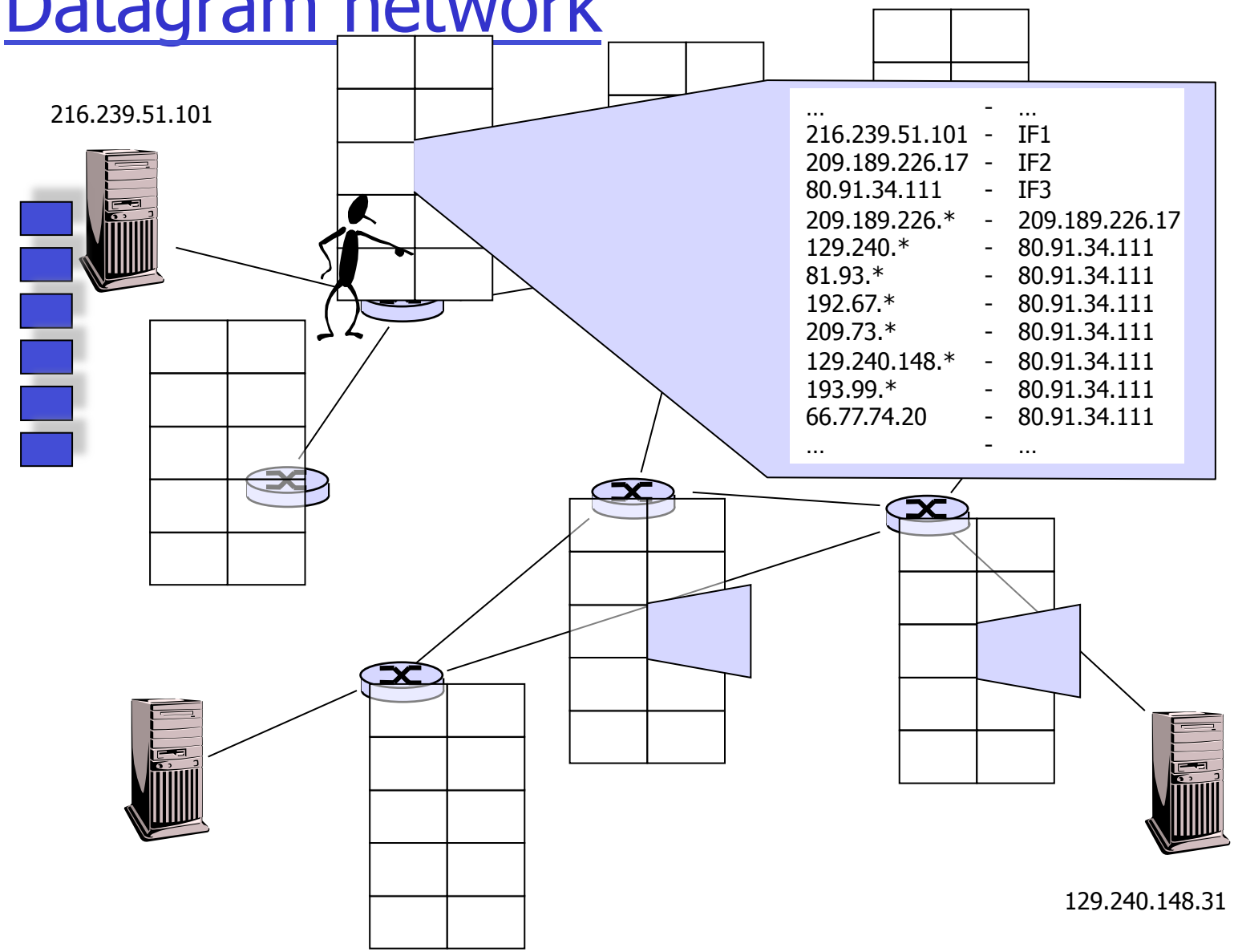
Datagram and Virtual Circuit Networks



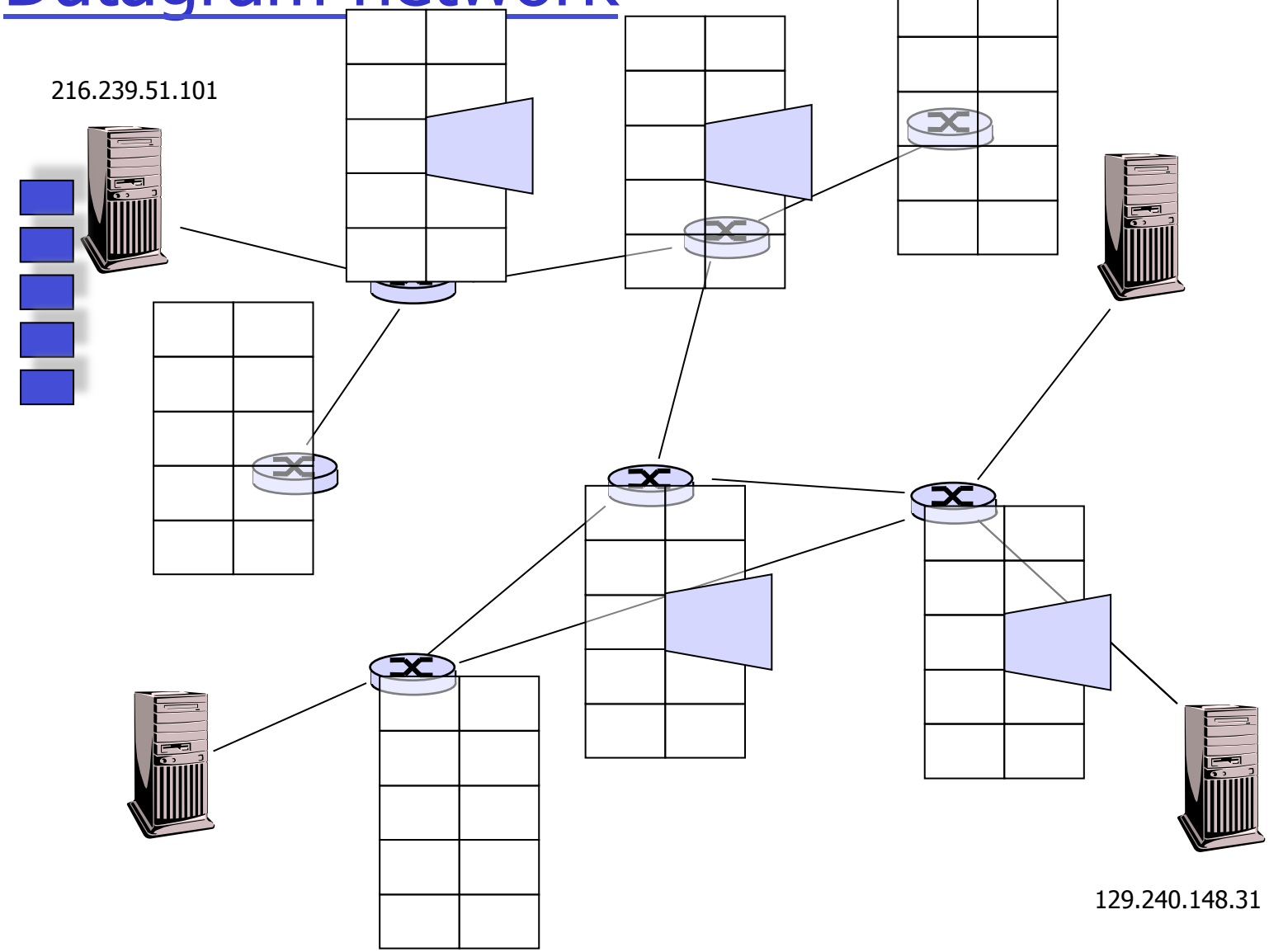
Datagram and Virtual Circuit Networks



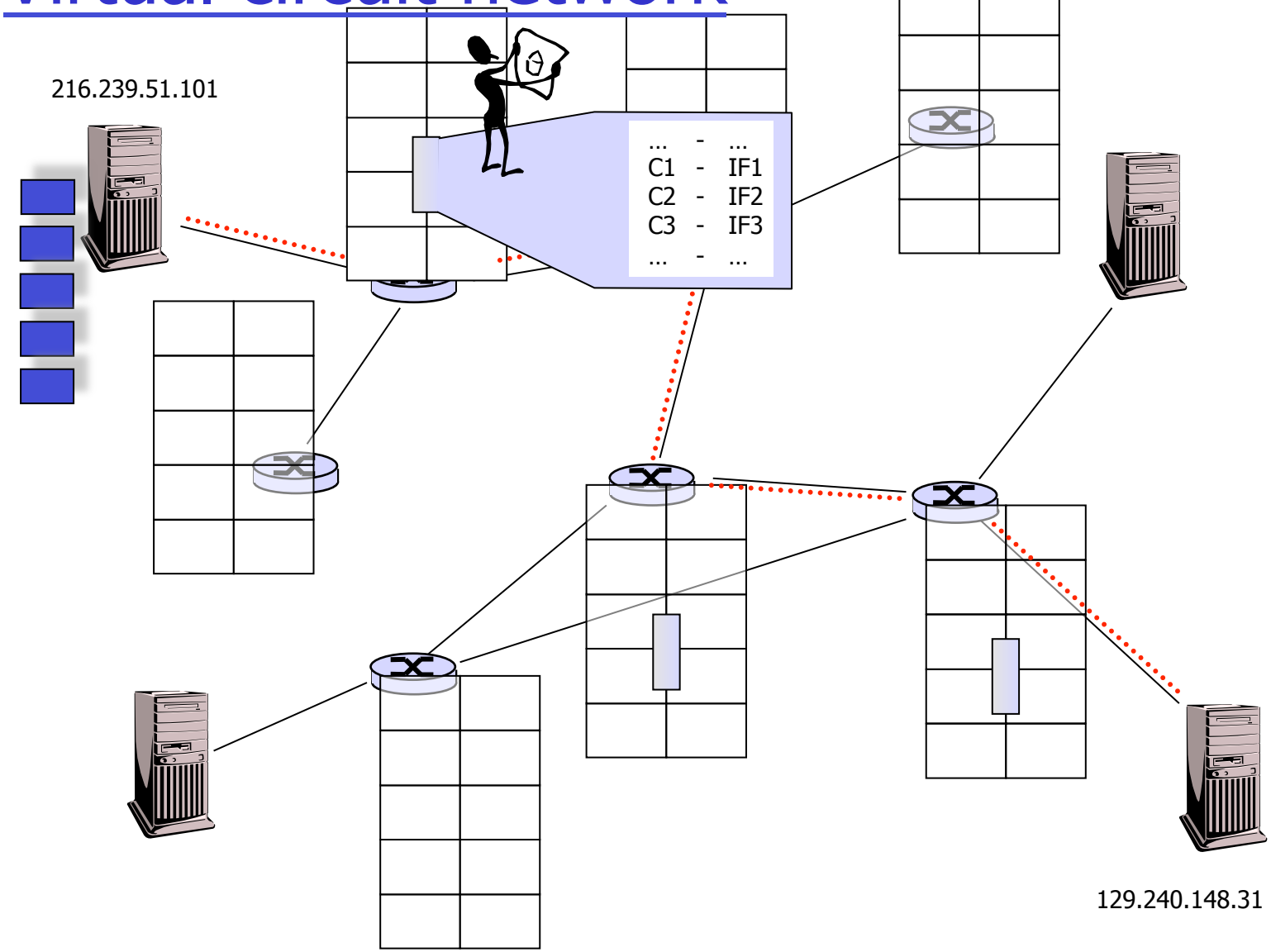
Datagram network



Datagram network



Virtual circuit network



Network layer: IP

Datagram switching

- IP
  - Internet Protocol
  - Datagram service of the Internet
  - RFC 791

IP offers:

- Addressing
- Routing
- Datagram service
  - Unreliable
  - Unordered

IP networks can use virtual circuits

- IPv4: circuit is one hop
- IPv6: can have a tag

Connection-oriented service

Goal: data transfer between end systems

- Start of communication
  - *Handshaking*
  - Initial preparation of data transfer
  - Hi!, hi! Is a human handshaking protocol
  - Creates a *"state"* in the two machines that communicate.
  - End systems know their communication partners
- During communication
  - *Connection*
  - End system expects messages from connected end system
  - End system knows when messages belong to the connection
- End of communication
  - *Teardown*
  - Bye! Bye! Is a human teardown protocol
  - New handshake required for re-establishing connection

Connectionless service

Goal: data transfer between end systems

- As before!
- Start of communication
  - No connection setup
  - No preparation for data transfer
  - Programs must expect messages at all times
- During communication
  - *No connection*
  - No state in the machines
  - Senders don't know whether messages are expected
  - Sender must identify itself in each message
- End of communication
  - *No teardown*
  - Just stop sending

Services over Switching Approaches

- Services requested
  - Between end systems
  - Connection-oriented service
  - Connectionless service
- Switching approaches
  - From host to host
  - Circuit switching
  - Packet switching

	Connection-oriented service	Connection-less service
Circuit switching	Fits well	Setup wasted
Packet switching	Additional work needed	Fits well

Transport layer: TCP

Connection-oriented service

- TCP
  - Transmission Control Protocol
  - Connection-oriented service of the Internet
  - RFC 793

TCP offers:

- *Connections*
  - *Handshake, end-system state, teardown*
- *Reliable, ordered, stream-oriented* data transfer
  - Loss: acknowledgements and retransmissions
- *Flow control:*
  - Send not faster than receiver can receive
- *Congestion control:*
  - Send slower when the network is congested.

Transport layer: UDP

Connectionless service

- UDP
  - User Datagram Protocol
  - Connectionless service of the Internet
  - RFC 768

UDP offers:

- *No connections*
  - Send immediately
- *Unreliable, unordered, packet-oriented* data transfer
  - Loss: messages are simply lost
  - Messages arrive exactly as send
- *No flow control*
  - Send as fast as programs want to
- *No congestion control*
  - Ignore network problems

Transport layer: applications

Applications that use TCP:

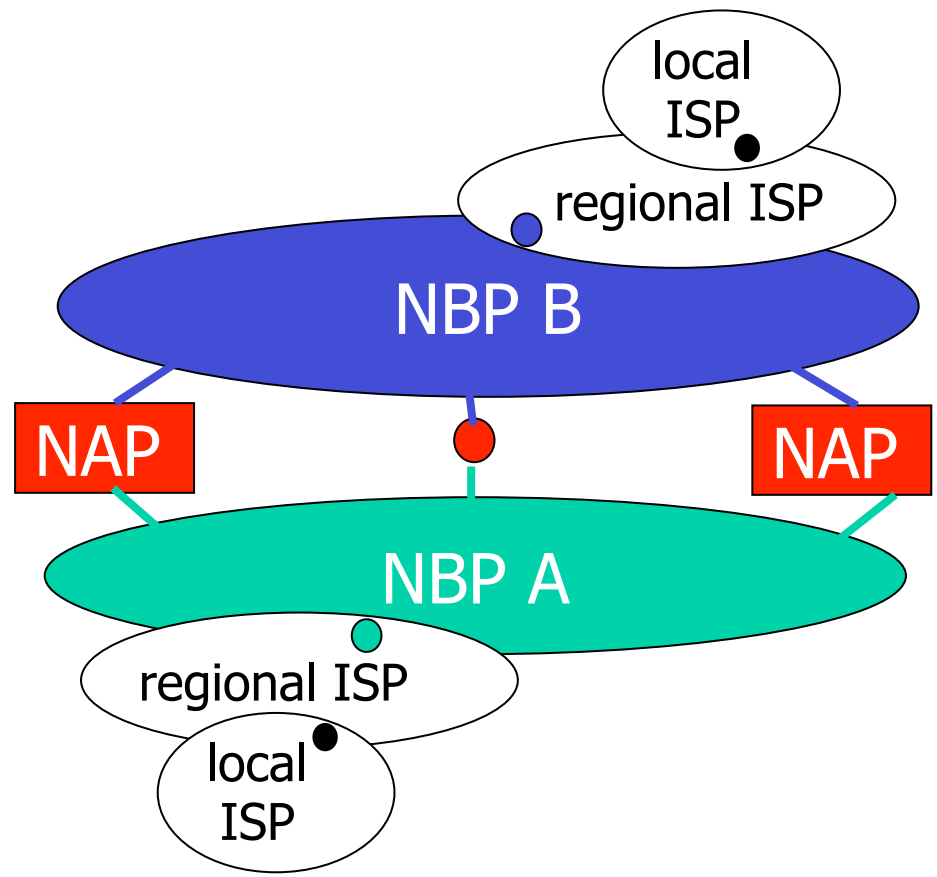
- HTTP (WWW)
- FTP (file transfer)
- SMTP (email)
- Telnet (remote login)

Applications that use UDP:

- Streaming media
- Video conferencing
- Internet telephony
- NTP (network time protocol)

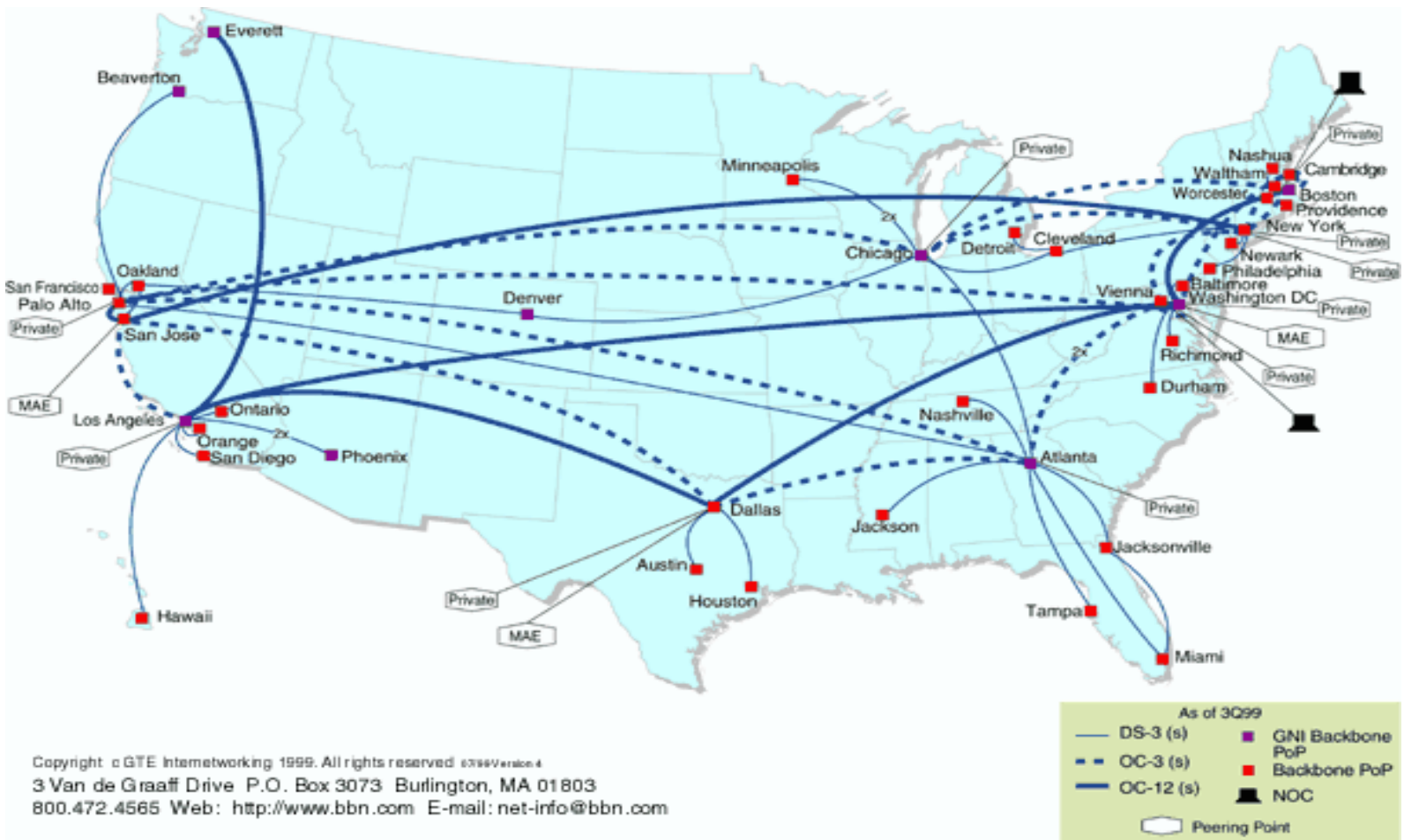
Internet structure: network of networks

- More or less hierarchical
- National/international "backbone providers" (NBPs)
  - These interconnect either privately, or at so-called Network Access Point (NAPs)
- regional ISPs
  - Connect to NBPs
- local ISPs, companies
  - Connect to regional ISPs



National Backbone Provider

example BBN/GTE US backbone network



History of the Internet

1961-1972: Early packet-switching concepts

- 1961: Kleinrock – queueing theory proves that packet switching is effective
- 1964: Baran – packet switching in military networks
- 1967: ARPAnet starts Advanced Reearch Projects Agency
- 1969: first ARPAnet node operational
- 1972:
  - ARPAnet publically demonstrated
  - NCP (Network Control Protocol) first machine-machine protocol
  - first e-mail program
  - ARPAnet has 15 nodes

History of the Internet

1972-1980: Internetworking – new, proprietary networks

- 1970: ALOHAnet satellite network on Hawaii
- 1973: Metcalfe's doctor thesis proposes Ethernet
- 1974: Cerf and Kahn – architecture to the interconnection of many networks
- End of the 70s: proprietary architectures: DECnet, SNA, XNA
- 1979: ARPAnet has 200 nodes

Cerf og Kahn's internetworking principles:

- Minimalism, autonomy – no internal network changes necessary to interconnect networks
- best effort service model
- Statekess routers
- Decentralized control

This defines mostly today's Internet architecture

History of the Internet

1980-1990: new protocols – the Net grows

- 1983: first use of TCP/IP
- 1982: e-mail protocol SMTP defined
- 1983: DNS defined to translate a name into an IP address
- 1985: ftp protocol defined
- 1988: TCP congestion control
- New national networks: Csnet, BITnet, NSFnet, Minitel
- 100,000 machine connected to the Net.

History of the Internet

1990's: commercialization, WWW

- Early 1990s: WWW
  - hypertext [Bush 1945, Nelson 1960's]
  - HTML, http: Berners-Lee
  - 1994: Mosaic, later Netscape
  - late 1990s: commercialization of the www

Late 1990s:

- ca. 50 million machines on the Internet
- ca. 100 million+ users
- backbone links operate at 1 Gbps

Summary

Covering a large area!

- Overview over the Internet
- What is a protocol?
- Network components
- Throughput, loss, delay
- Layering and service models
- backbone, NAPer, ISPer

Hopefully you have now:

- An impression and overview of the area
- More depth and details in the following lessons, and in later courses