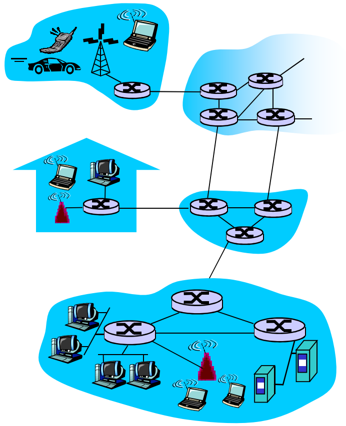
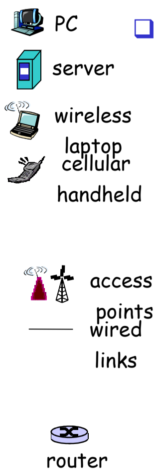
Chapter 1 Introduction

What's the Internet: a service view

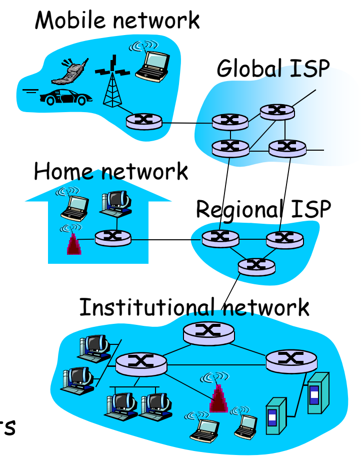
- communication *infrastructure* enables distributed applications:
  - Web, VoIP, email, games, e-commerce, file sharing
- communication services provided to apps:
  - reliable data delivery from source to destination
  - "best effort" (unreliable) data delivery



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

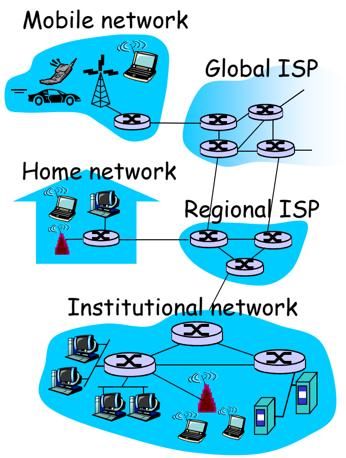


- millions of connected computing devices: *hosts = end systems*
  - running *network apps*
- *communication links*
  - fiber, copper, radio, satellite
  - transmission rate = *bandwidth*
- *routers:* forward packets (chunks of data)



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

- *protocols* control sending, receiving of msgs
  - e.g., TCP, IP, HTTP, Skype, Ethernet
- *Internet:* "network of networks"
  - loosely hierarchical
  - public Internet versus private intranet
- Internet standards
  - RFC: Request for comments
  - IETF: Internet Engineering Task Force



What's a protocol?

human protocols:
- "what's the time?"
- "I have a question"
- introductions

… specific msgs sent

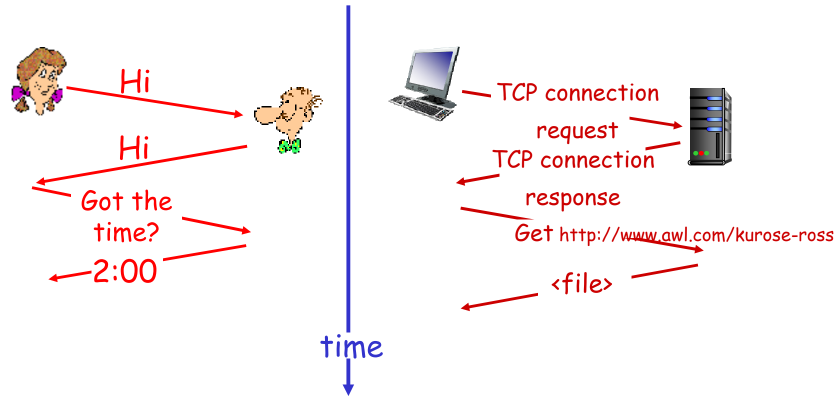
… specific actions taken when msgs received, or other events

network protocols:
- machines rather than humans
- all communication activity in Internet governed by protocols

protocols define format, order of msgs sent and received among network entities, and actions taken on msg transmission, receipt

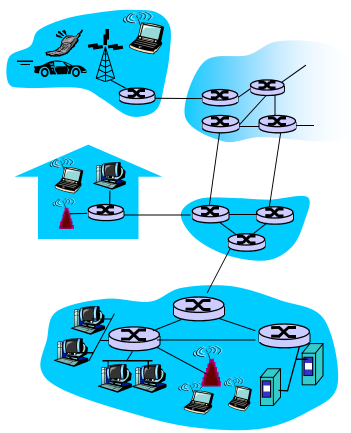
What's a protocol?

a human protocol and a computer network protocol:



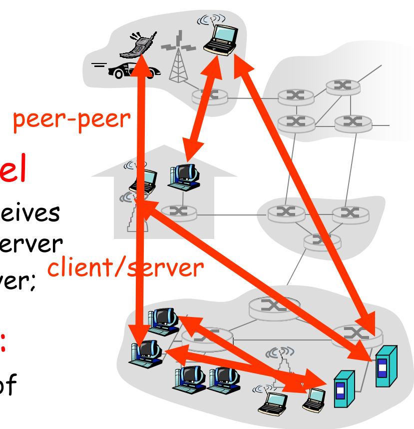
A closer look at network structure:

- network edge: applications and hosts
- access networks, physical media: wired, wireless communication links
- network core:
  - interconnected routers
  - network of networks



The network edge:

- end systems (hosts):
  - run application programs
  - e.g. Web, email
  - at "edge of network"
- client/server model
  - client host requests, receives service from always-on server
    - e.g. Web browser/server; email client/server
- peer-peer model:
  - minimal (or no) use of dedicated servers
  - e.g. Skype, BitTorrent



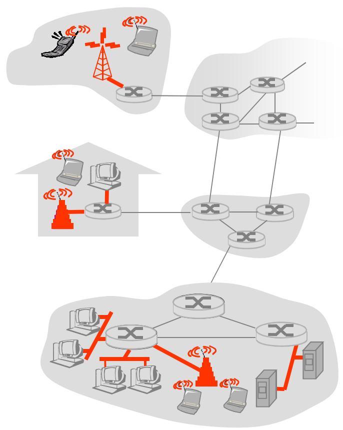
Access networks and physical media

Q: How to connect end systems to edge router?

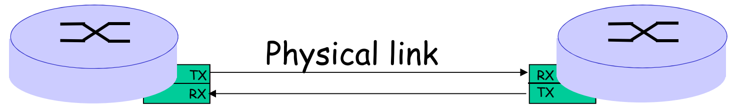
- residential access nets
- institutional access networks (school, company)
- mobile access networks

Keep in mind:

- bandwidth (bits per second) of access network?
- shared or dedicated?

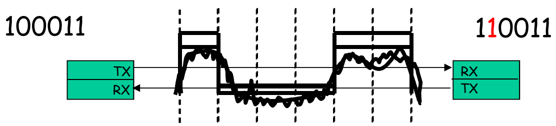


Transmission across a physical link: point-to-point networks



- Bits: propagate between transmitter and receiver
- 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

Transmission across a physical link



- Bit sequence modulates a suitable waveform which is sent across the link
- As the signal travels it experiences
  - Attenuation (absorption)
  - Distortion (limited bandwidth (frequency))
  - Noise (interference, thermal noise)
  - Influenced by medium, bit rate and distance
- Received sequence may be incorrect!!!

Maximum Channel Data Rate

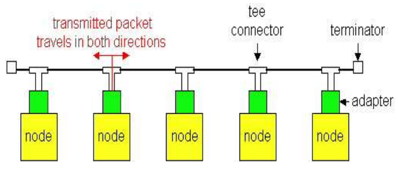
- Shannon Theorem: max data rate of a noisy channel whose bandwidth is H Hz, and whose signal to noise power ratio is S/N, is

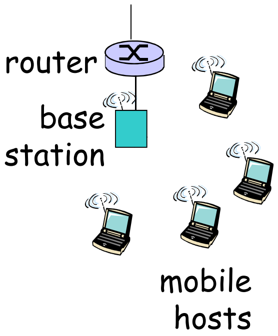
$$H \log_2 (1+S/N)$$

  - H frequency interval over which signal is transmitted
    - Depends on the physical medium

Transmission across a physical link: broadcast networks

- Wired networks
  - Legacy Ethernet
- Wireless networks
  - Wireless LAN





Physical Media: twisted pair

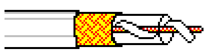
Twisted Pair (TP)

- two insulated copper wires
  - Twisted to reduce interference
- Category 3: traditional phone wires, 10 Mbps Ethernet
- Category 5 TP: 100Mbps Ethernet

Physical Media: coax, fiber

Coaxial cable:

- two concentric copper conductors
- bidirectional
- baseband:
  - single channel on cable
  - legacy Ethernet
- broadband:
  - multiple channel on cable
  - HFC



Fiber optic cable:

- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed (e.g., 5 Gps) point-to-point
- low error rate: repeaters spaced far apart; immune to electromagnetic noise

Physical Media: radio

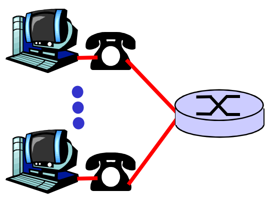
- signal carried in electromagnetic spectrum
- no physical "wire"
- bidirectional
- propagation environment effects:
  - reflection
  - obstruction by objects
  - interference

Radio link types:

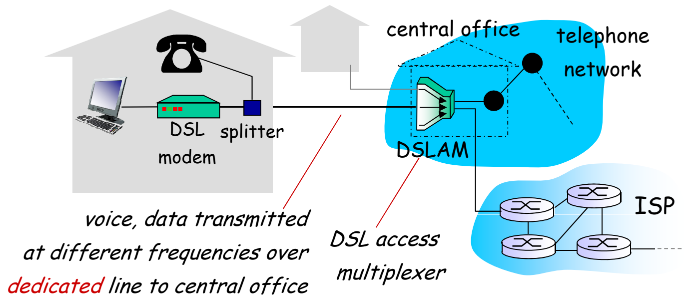
- terrestrial microwave
  - e.g. up to 45 Mbps channels
- LAN (e.g., WiFi)
  - 11Mbps, 54Mbps
- wide-area (e.g., cellular)
  - e.g. 3G: hundreds of kbps
- satellite
  - up to 50Mbps channel (or multiple smaller channels)
  - 270 msec end-end delay
  - geosynchronous versus LEOS

Residential access: point to point access

- Dialup via modem
  - up to 56Kbps direct access to router (often less)
  - Can't surf and phone at same time: can't be "always on"

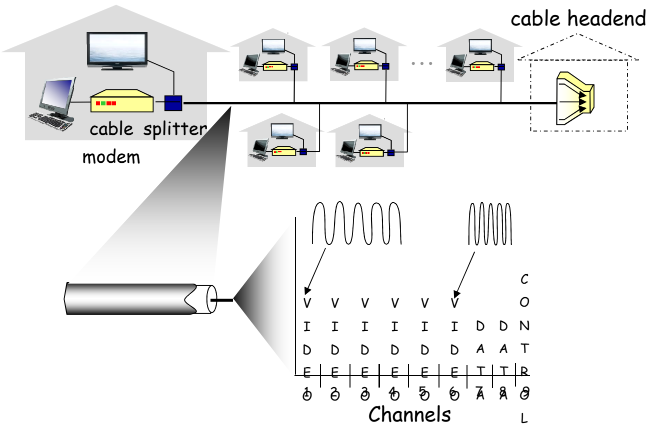


Access net: digital subscriber line (DSL)



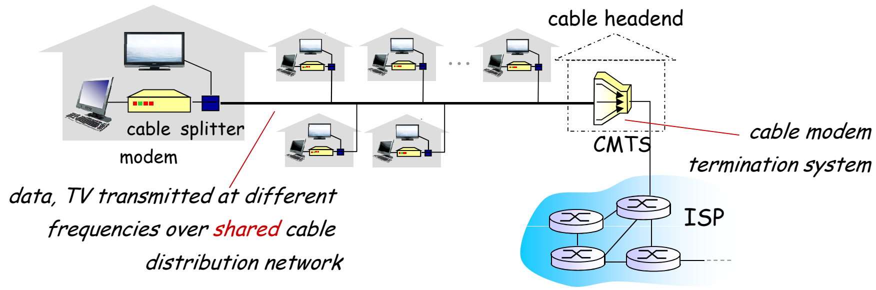
- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- < 2.5 Mbps upstream transmission rate (typically < 1 Mbps)
- < 24 Mbps downstream transmission rate (typically < 10 Mbps)

Access net: cable network



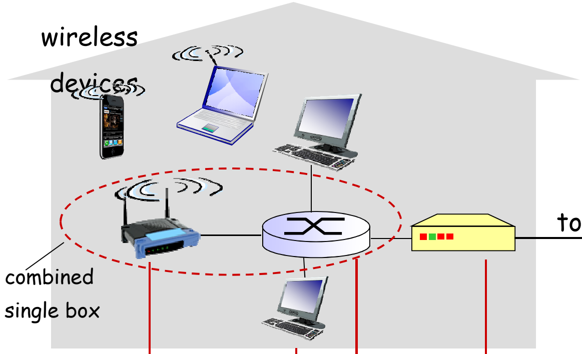
frequency division multiplexing: different channels transmitted in different frequency bands

Access net: cable network

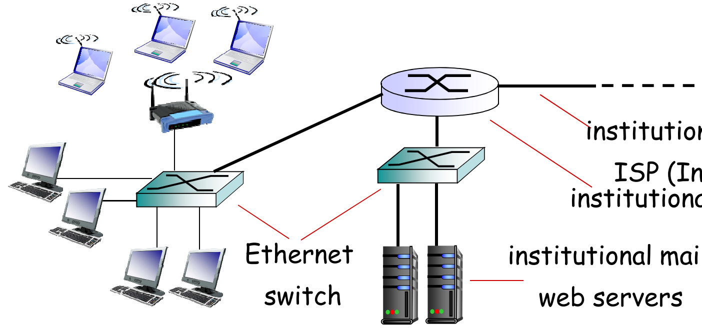


- HFC: hybrid fiber coax
  - asymmetric: up to 30Mbps downstream transmission rate, 2 Mbps upstream transmission rate
- network of cable, fiber attaches homes to ISP router
  - homes *share access network* to cable headend
  - unlike DSL, which has dedicated access to central office

Access net: home network



Enterprise access networks (Ethernet)



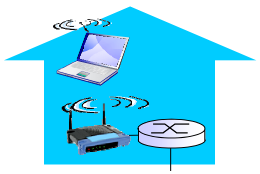
- typically used in companies, universities, etc
- 10 Mbps, 100Mbps, 1Gbps, 10Gbps transmission rates
- today, end systems typically connect into Ethernet switch

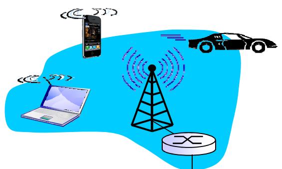
Wireless access networks

- shared *wireless* access network connects end system to router
  - via base station aka "access point"

wireless LANs:
- within building (100 ft)
- 802.11b/g (WiFi): 11, 54 Mbps transmission rate

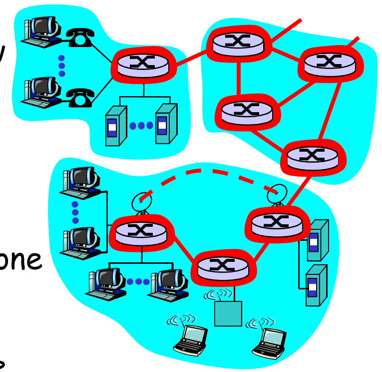
wide-area wireless access
- provided by telco (cellular) operator, 10's km
- between 1 and 10 Mbps
- 3G, 4G: LTE





The Network Core

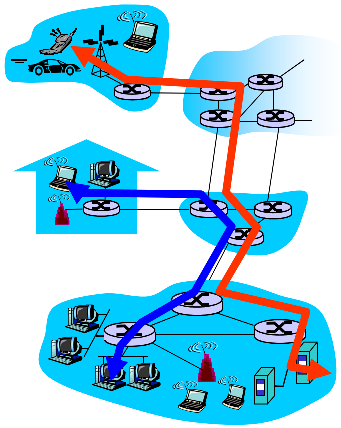
- mesh of interconnected routers
- fundamental questions: how data transferred through net? How are network resources shared?
  - circuit switching: dedicated resources (circuit) per call: telephone net
  - packet-switching: data sent thru net in discrete "chunks". Resources allocated on demand



Network Core: Circuit Switching

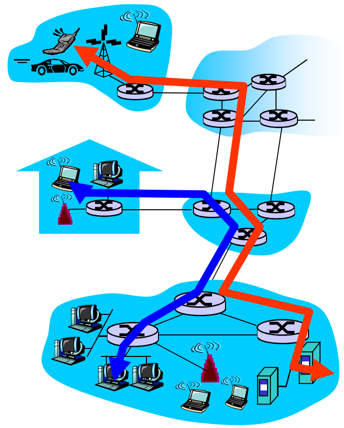
End-end resources reserved for "call"

- link bandwidth, switch capacity
- dedicated resources: no sharing
- circuit-like (guaranteed) performance
- call setup required



Network Core: Circuit Switching

- 3 phases
  1. Call setup
     - Resources Allocation
  2. Data transfer
     - Resources Usage
  3. Call Teardown
     - Resources Release
  - required for all connection-based services

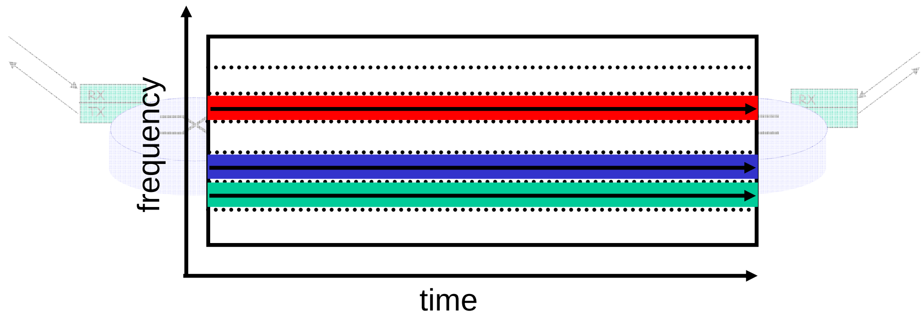


Network Core: Circuit Switching

- network resources (e.g., bandwidth) divided into "pieces"
  - pieces allocated to calls

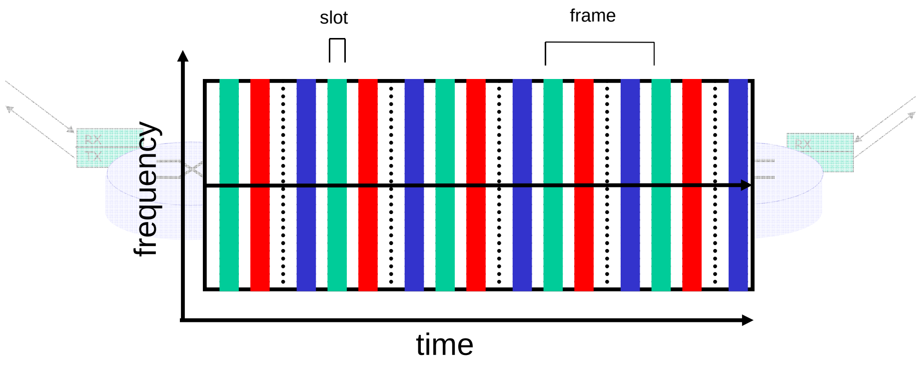
Network Core: Circuit Switching

- FDM: Frequency Division Multiplexing
  - Different frequency intervals allocated to different calls

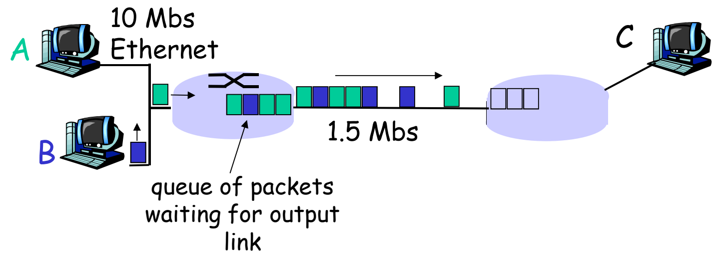


Network Core: Circuit Switching

- TDM: Time Division Multiplexing
  - Different time intervals allocated to different calls
  - For each call: one slot per frame



Network Core: Packet Switching



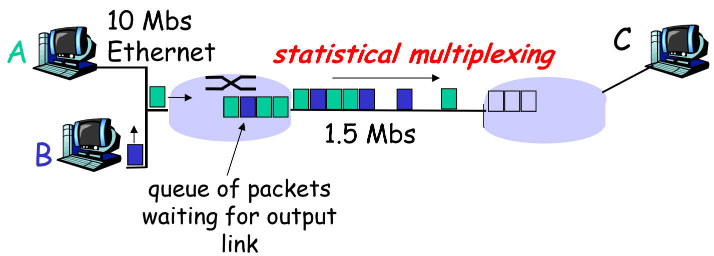
each end-end data stream divided into *packets*

- user A, B packets *share* network resources
- each packet uses full link bandwidth

store and forward

- Entire packet must arrive at router before it can be transmitted on next link
- packets move one hop at a time
  - transmit over link
  - wait turn at next link

Network Core: Packet Switching



Statistical Multiplexing: Sequence of A & B packets does not have fixed pattern

- resources used *as needed*

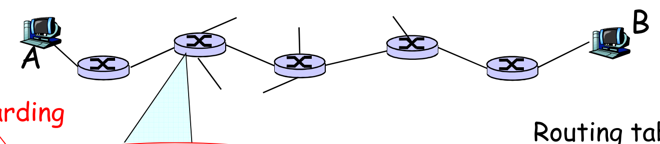
resource contention:

- aggregate resource demand can exceed amount available
- congestion: packets queue, wait for link use
  - May get dropped if buffer gets full

~~Bandwidth division into "pieces"~~
~~Dedicated allocation~~
~~Resource reservation~~

Router

- Forward a chunk of information (called *packet*) arriving on one of its communication links to one of its outgoing communications link (the *next hop* on the source-to-destination path)



forwarding

-Receives the packet
-Based on a routing table and the destination address, computes the 'next hop' to the destination
-Forwards the packet to the next hop
-The process of computing and maintaining the routing table is called Routing

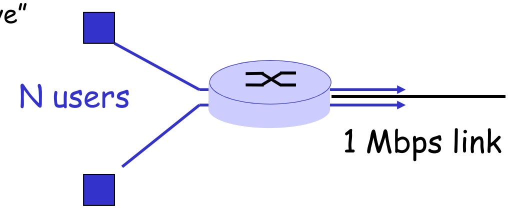
Routing table

Dest. Address	Next Hop

Packet switching versus circuit switching

Packet switching allows more users to use network!

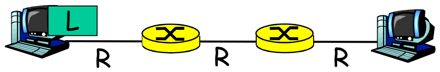
- 1 Mbit link
- each user:
  - 100 kbps when "active"
  - active 10% of time
- circuit-switching:
  - 10 users
- packet switching:
  - with 35 users, probability > 10 active less than .0004



Packet switching versus circuit switching

- Packet switching: Great for bursty data
  - resource sharing
  - simpler, no call setup
- Excessive congestion: packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- Q: How to provide circuit-like behavior?
  - bandwidth guarantees needed for audio/video apps
  - still an unsolved problem (chapter 6)

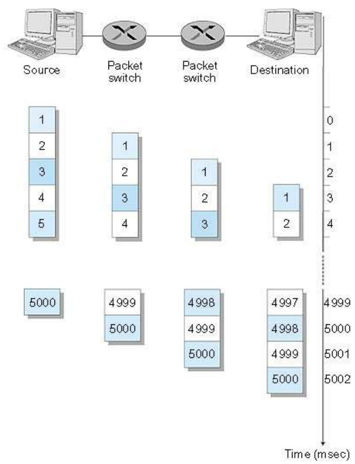
Packet-switching vs Message Switching



- Takes L/R seconds to transmit (push out) packet of L bits on to link or R bps
- Entire packet must arrive at router before it can be transmitted on next link: *store and forward*
- delay = 3L/R

Example:
- L = 7.5 Mbits
- R = 1.5 Mbps
- delay = 15 sec

Packet Switching: Message Segmenting



Now break up the message into 5000 packets

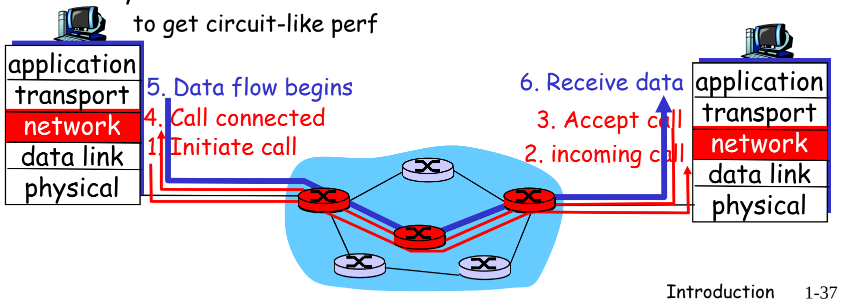
- Each packet 1,500 bits
- 1 msec to transmit packet on one link
- *pipelining:* each link works in parallel
- Delay reduced from 15 sec to 5.002 sec

Packet Switching: Virtual Circuits Networks

- Connection Based

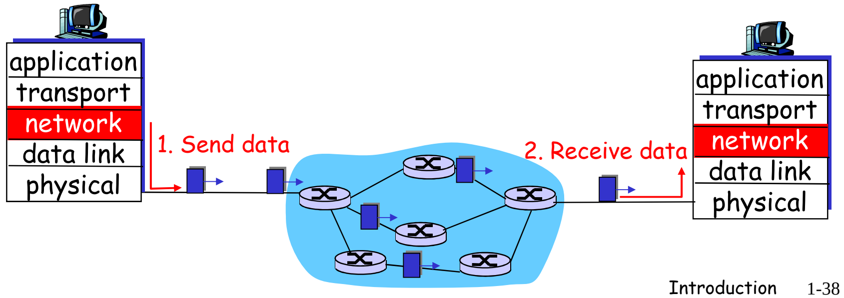
1. Call setup
   - fixed path determined at *call setup time*, remains fixed
   - link, bandwidth, buffers may be *allocated* to VC

     to get circuit-like perf
2. Data transfer
   - each packet carries VC identifier
3. Teardown

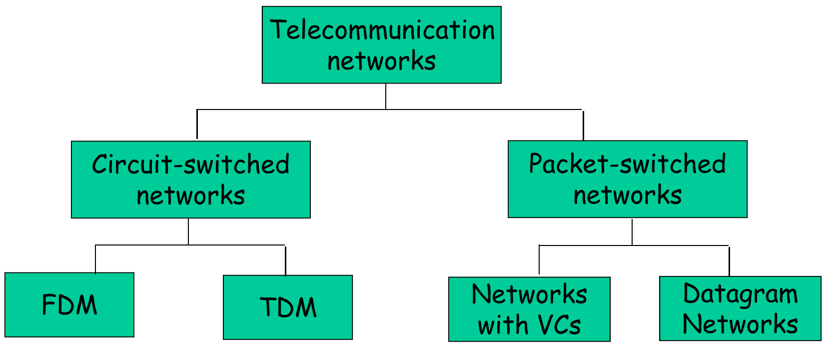


Packet Switching: Datagram Networks(The Internet Model)

- no call setup
- routers: no state about end-to-end connections
  - no network-level concept of "connection"
- packets forwarded using destination host address
  - packets between same source-dest pair may take different paths

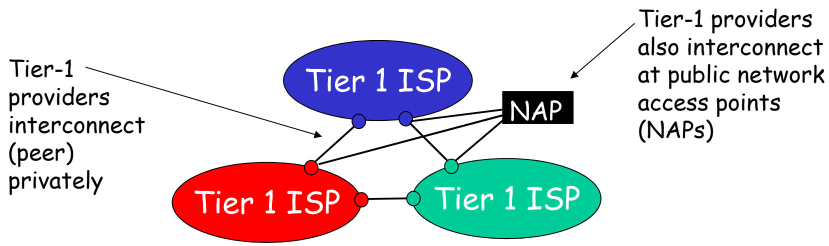


Network Taxonomy

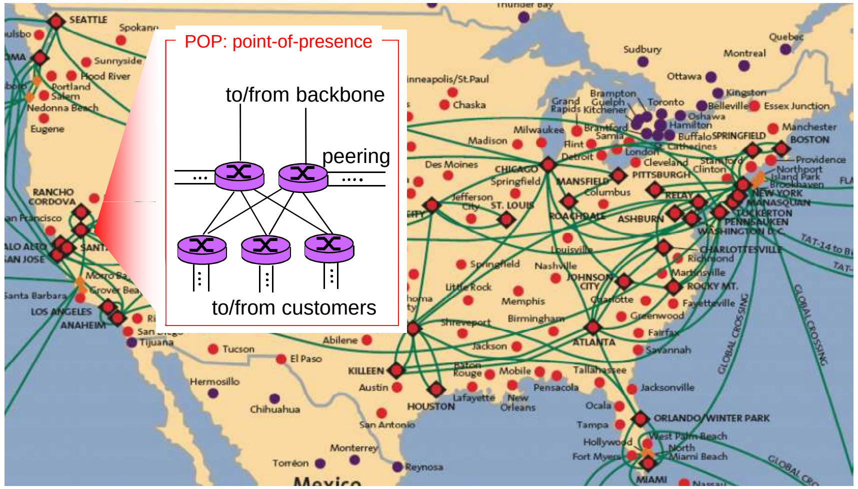


Internet structure: network of networks

- roughly hierarchical
- at center: "tier-1" ISPs (e.g., UUNet, BBN/Genuity, Sprint, AT&T), national/international coverage
  - treat each other as equals



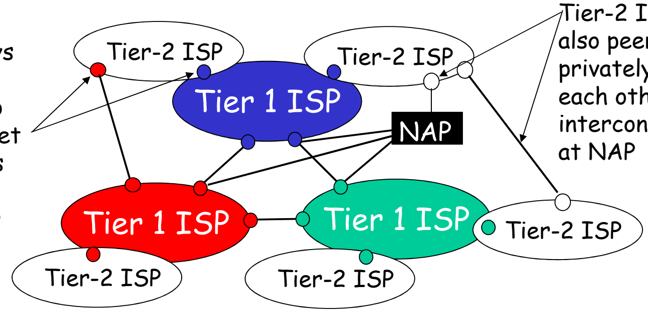
Tier-1 ISP: e.g., Sprint



Internet structure: network of networks

- "Tier-2" ISPs: smaller (often regional) ISPs
  - Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs

Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet
- tier-2 ISP is *customer* of tier-1 provider

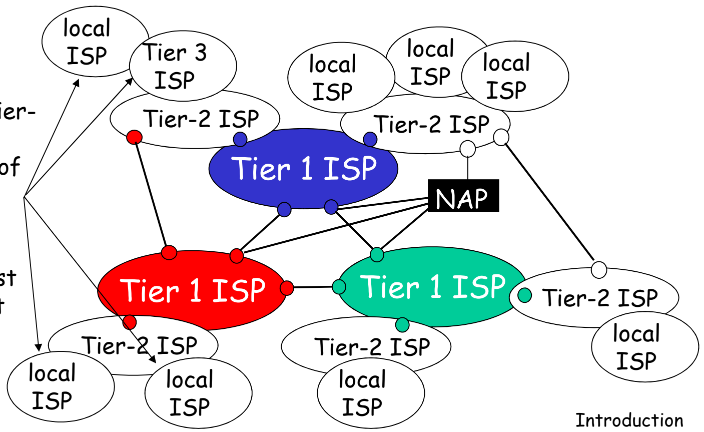


Tier-2 ISPs also peer privately with each other, interconnect at NAP

Internet structure: network of networks

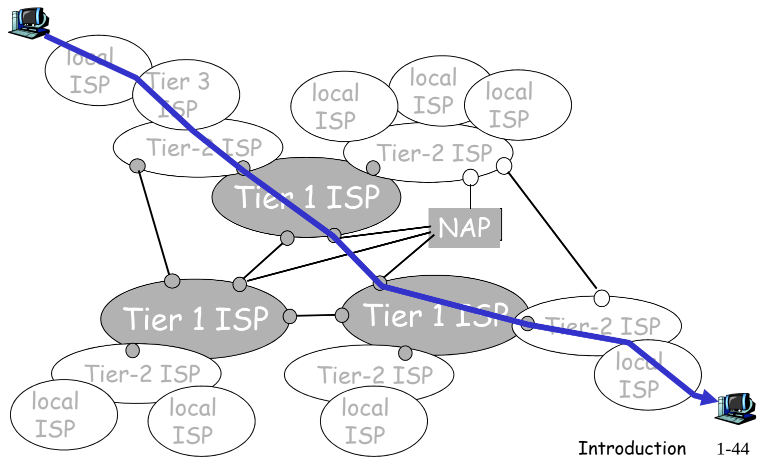
- "Tier-3" ISPs and local ISPs
  - last hop ("access") network (closest to end systems)

Local and tier-3 ISPs are *customers* of higher tier ISPs connecting them to rest of Internet



Internet structure: network of networks

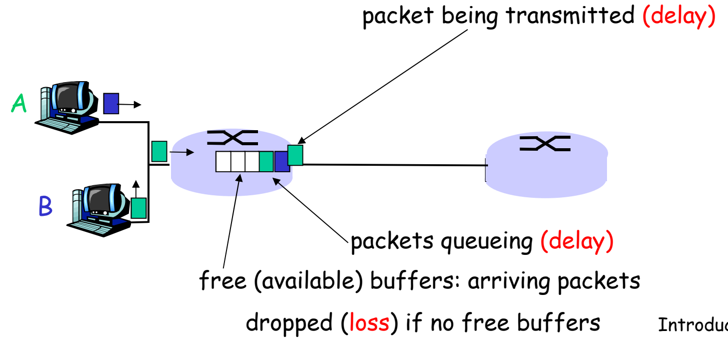
- a packet passes through many networks!



How do loss and delay occur?

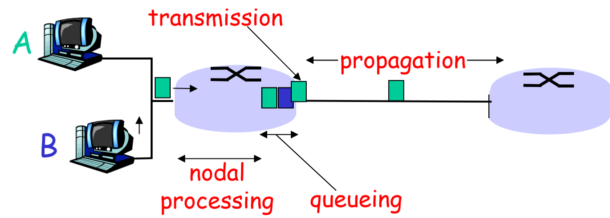
packets *queue* in router buffers

- packet arrival rate to link exceeds output link capacity
- packets queue, wait for turn



How do loss and delay occur?

- 1. nodal processing:
  - check bit errors
  - determine output link
- 2. queueing
  - time waiting at output link for transmission
  - depends on congestion level of router



How do loss and delay occur? (2)

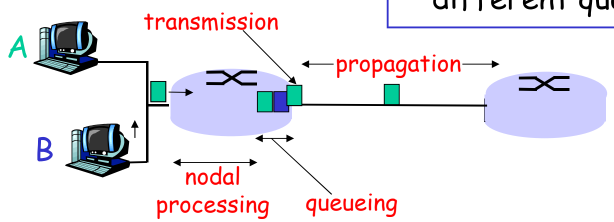
3. Transmission delay:

- R=link bandwidth (bps)
- L=packet length (bits)
- time to send bits into link = L/R

4. Propagation delay:

- d = length of physical link
- s = propagation speed in medium (~2×10^8 m/sec)
- propagation delay = d/s

Note: s and R are *very* different quantities!



Nodal delay

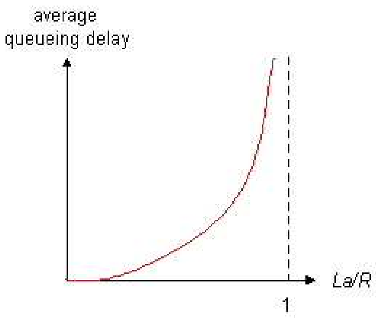
$$\boxed{d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}}$$

- d_{proc} = processing delay
  - typically a few microsecs or less
- d_{queue} = queuing delay
  - depends on congestion
- d_{trans} = transmission delay
  - = L/R, significant for low-speed links
- d_{prop} = propagation delay
  - a few microsecs to hundreds of msecs

Queueing delay (revisited)

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

traffic intensity = La/R



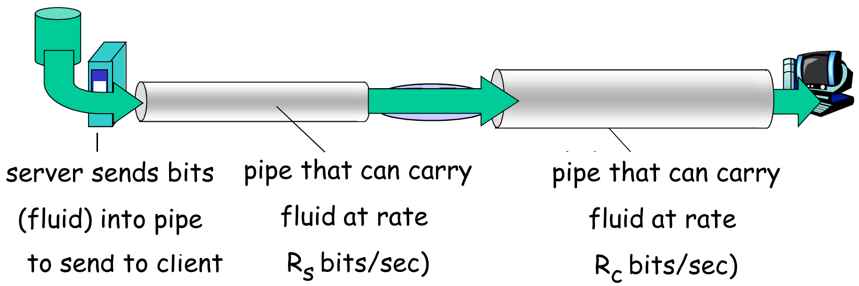
- La/R ~ 0: average queueing delay small
- La/R -> 1: delays become large
- La/R > 1: more "work" arriving than can be serviced, average delay infinite!

Packet loss

- queue (aka buffer) preceding link in buffer has finite capacity
- when packet arrives to full queue, packet is dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not retransmitted at all

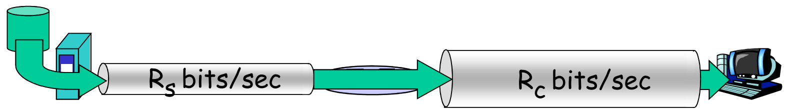
Throughput

- *throughput:* rate (bits/time unit) at which bits transferred between sender/receiver
  - *instantaneous:* rate at given point in time
  - *average:* rate over longer period of time

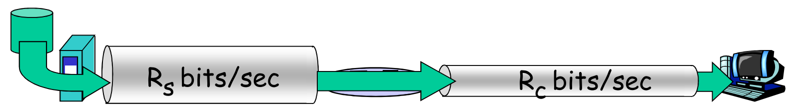


Throughput (more)

- $R_s < R_c$ What is average end-end throughput?



- $R_s > R_c$ What is average end-end throughput?

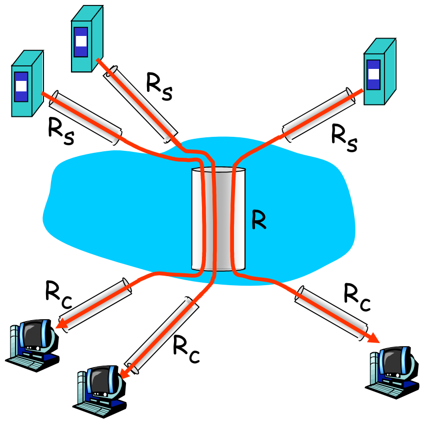


bottleneck link

link on end-end path that constrains end-end throughput

Throughput: Internet scenario

- per-connection end-end throughput: $\min(R_c, R_s, R/10)$
- in practice: R_c or R_s is often bottleneck



10 connections (fairly) share backbone bottleneck link R bits/sec

Protocol "Layers"

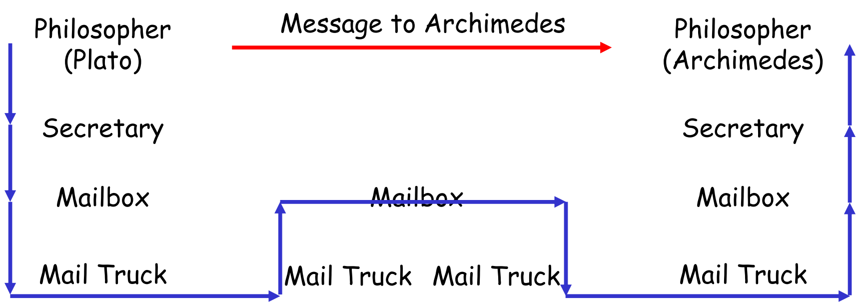
Networks are complex!

- many "pieces":
  - hosts
  - routers
  - links of various media
  - applications
  - protocols
  - hardware, software

Question:
Is there any hope of *organizing* structure of network?

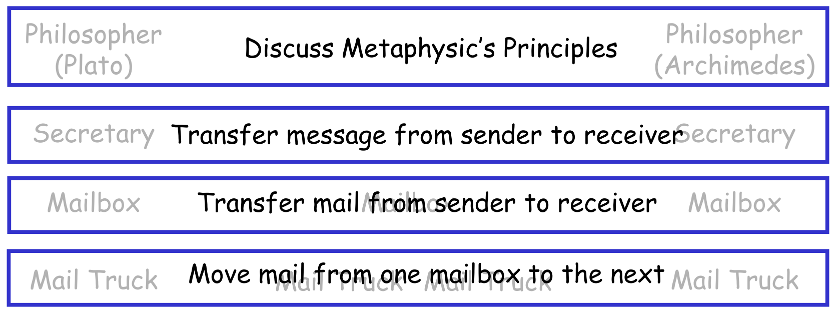
Or at least our discussion of networks?

Example: Plato and Archimedes…



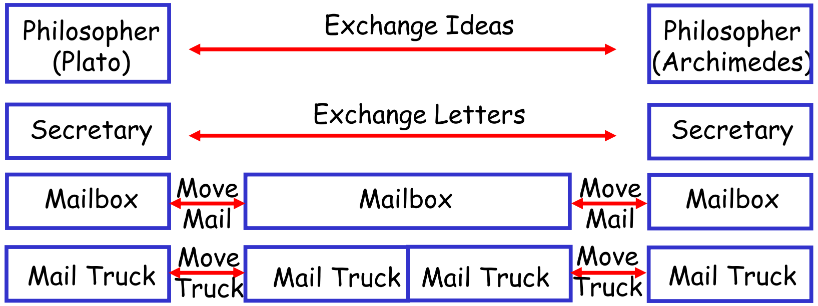
- a series of steps

Plato & Archimedes: a different view



- Layers: each layer implements a service
  - relying on services provided by layer below

Plato & Archimedes: a different view



- Layers: each layer implements a service
  - via its own internal-layer actions
    - governed by rules (protocols)

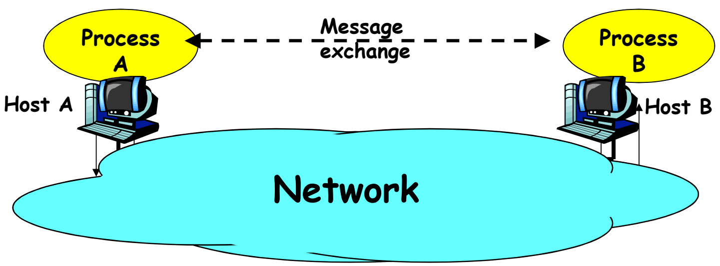
Why layering?

Dealing with complex systems:

- Decompose the system into subsystems
  - Each implementing a subset of overall functionalities
- layered subsystems as reference model for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change from mail-truck to bike (environmental concerns…)
- layering considered harmful?

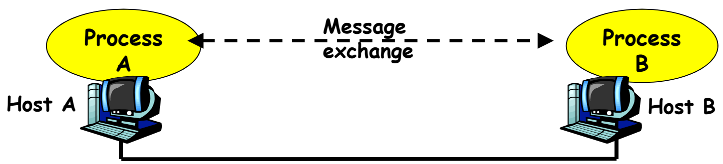
Layered Architecture

- Network provides communications services to applications



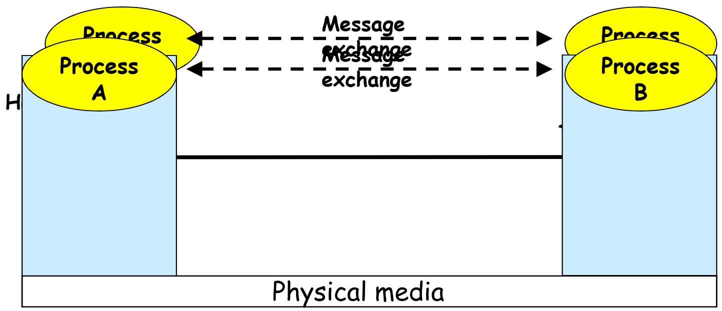
Layered Architecture

- Network provides communications services to applications



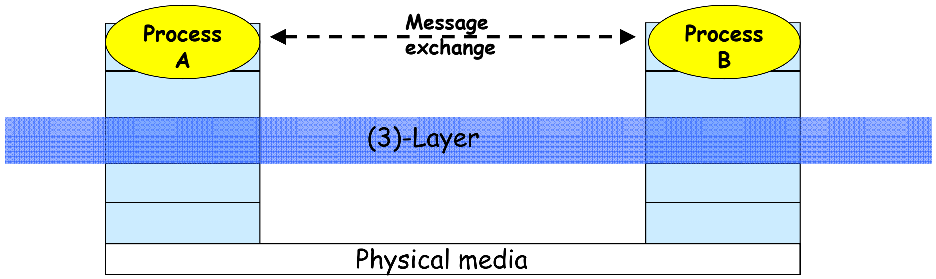
Layered Architecture

- Abstract Model of the communication environment
- Communication system as composed of ordered set of layers
  - Each implementing a subset of overall functionalities



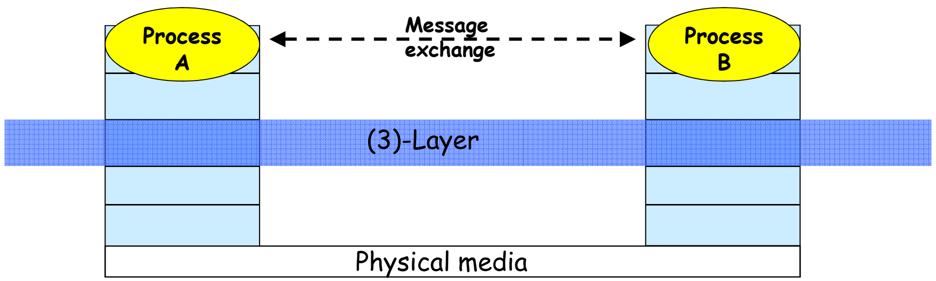
Layered Architecture

- Systems logically decomposed in subsystems
- Layer: collection of subsystems at the same level
  - (N) Layer: Layer at level N



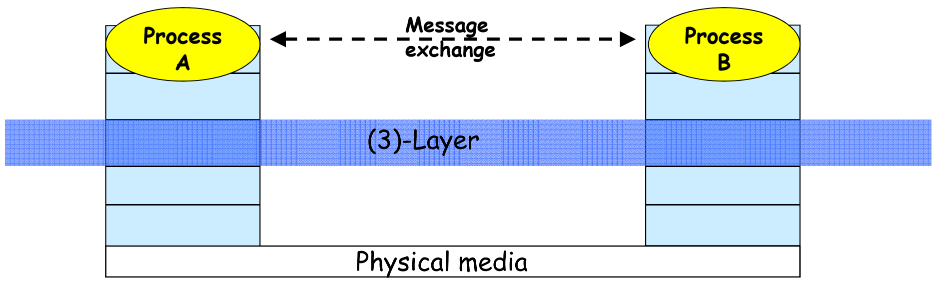
Layered Architecture

- Each layer provides a service (to the layer above)
  - Service: set of functions offered to the layer above
    - Error Control, Flow Control
  - relying on services provided by layer below
  - via its own internal-layer actions



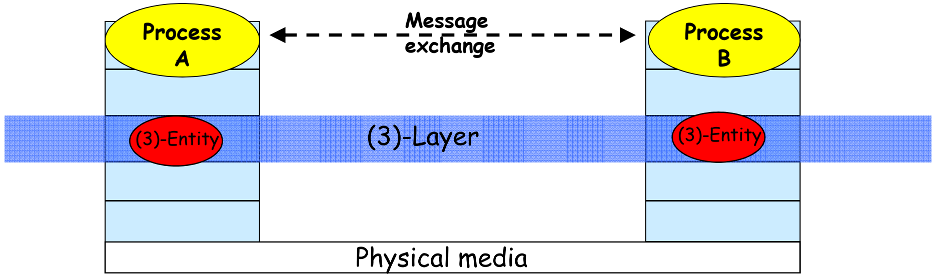
Layered Architecture

- Each layer provides a “value added” (communication) service with respect to the service provided by the layer below



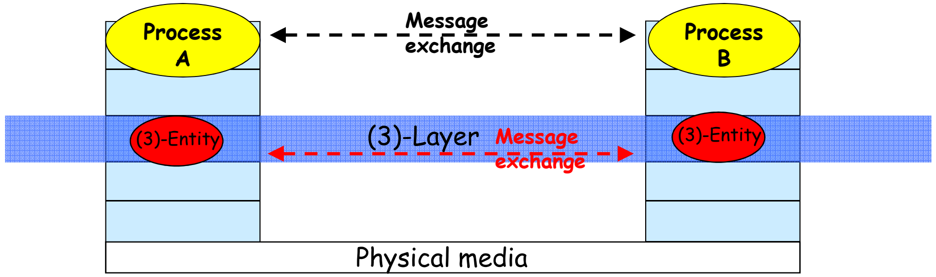
Entities

- Active layers' elements
  - (N)-Layer entity: (N)-Entity



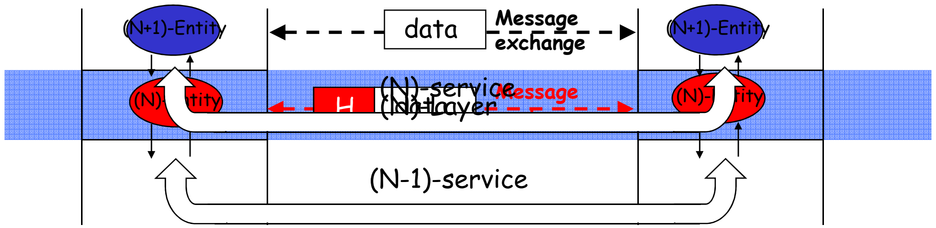
Entities

- Implement layer functions at nodes
  - Require entities cooperation, exchanging messages with peers
  - Using the (communication) service provided by the layer immediately below



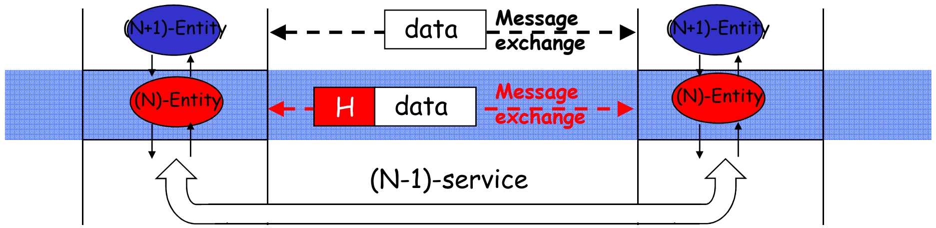
Layer Service

- (N)-layer provides the (N)-service to the (N+1)-entities ((N)-service users)
  - (N) Service: a collection of (N) functions
- (N)-function is carried out by the (N)-entities
  - Requires entities cooperation
  - Using the (N-1)-service

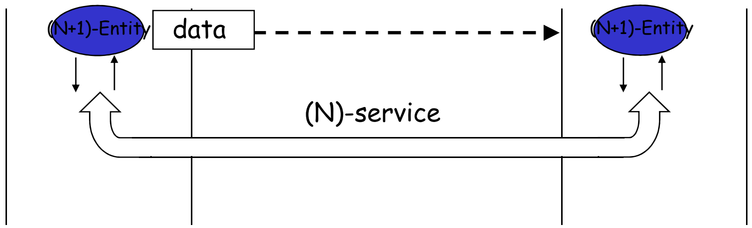


(N)-Protocol

- Cooperation among (N)-entities is governed by (N)-protocols
  - Syntax of message types: what fields in messages & how fields are delineated
  - Semantics of the fields, ie, meaning of information in fields
  - Rules for when and how processes send & respond to messages

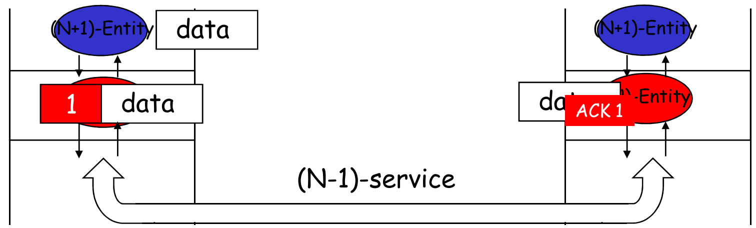


Logical Communication



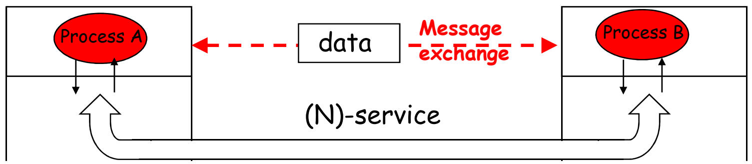
Logical Communication

- Ex: Reliable (N)-Service over unreliable (N-1) service



Layered Architecture

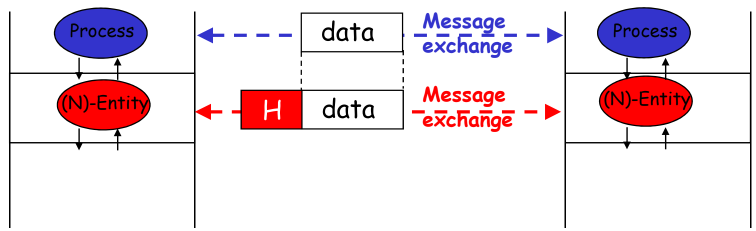
- Highest level: application level



- Lowest level directly interface with the physical media

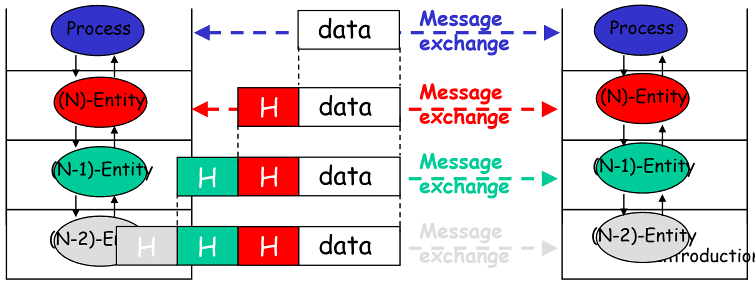
Protocol Data Unit (PDU)

- (N)-PDU
  - Messages exchanged between (N)-Entities
  - Comprises
    - (N)-Service Users data
    - Header (to carry out (N)-functions)



Protocol Data Unit (PDU)

- (N)-PDU
  - Messages exchanged between (N)-Entities
  - Comprises
    - (N)-Service Users data
    - Header (to carry out (N)-functions)



Connection oriented and Connectionless Service

- Connection Oriented (N)-Service
  - E.g. telephone call
  - To communicate, (N)-users
    - Setup Connection
    - Transfer Data
    - Tear-down Connection
- Connectionless (N)-service
  - E,g,Postal service
  - Trasfer Data

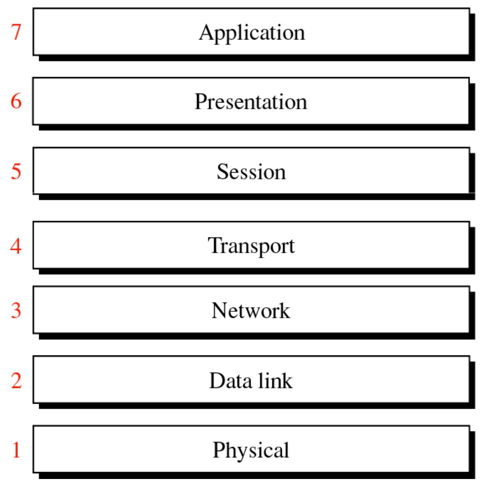
Layered Architecture: Models

- OSI (Open System Interconnection)
  - '70 Standard de iure
  - 7 levels
  - Architecture first
  - Protocols later…if any
- TCP/IP Model
  - '80… Standard de facto
  - 5 levels
  - Protocols first

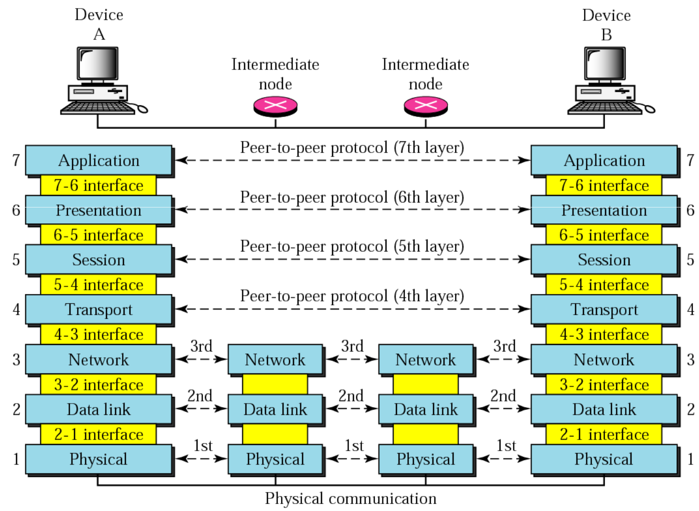
OSI (Open System Interconnection): Principles

- Minimize the number of layers
- Establish boundaries at points where interaction is minimum
- Layers should include different functions
- Layers could be redesigned without affecting interfaced with adjacent layers
- Each layer should add value

OSI (Open System Interconnection)



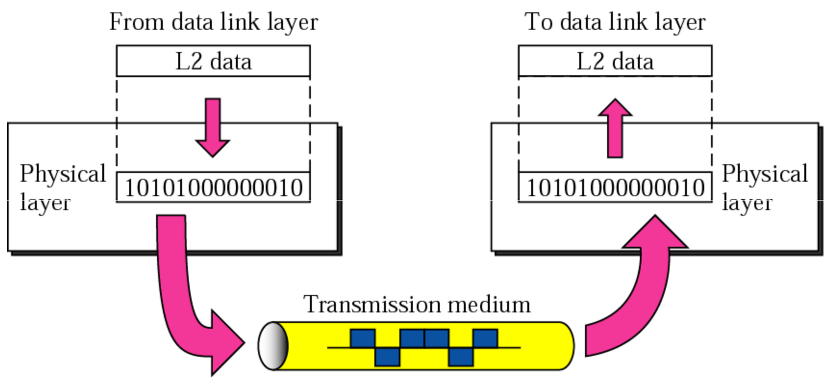
OSI Layers



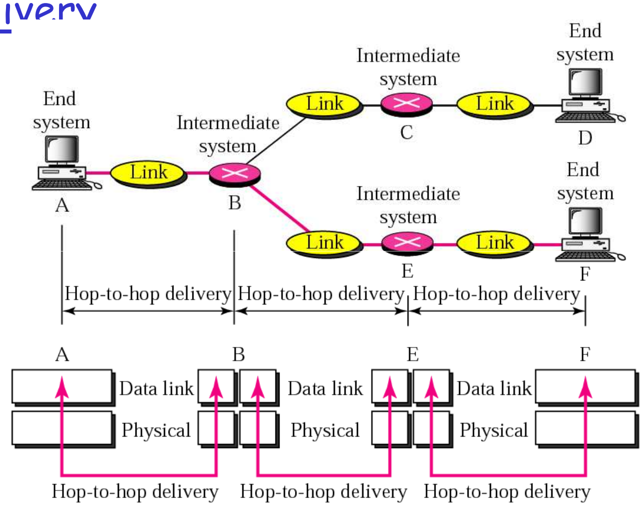
Physical Layer

- Coordinates the functions required to transmit a bit stream over a physical medium
- Defines
  - the characteristics of interfaces and media
  - Bit representation
  - Transmission rate
  - Bit Synchronization
  - Line configuration, transmission mode, etc.

Physical Layer



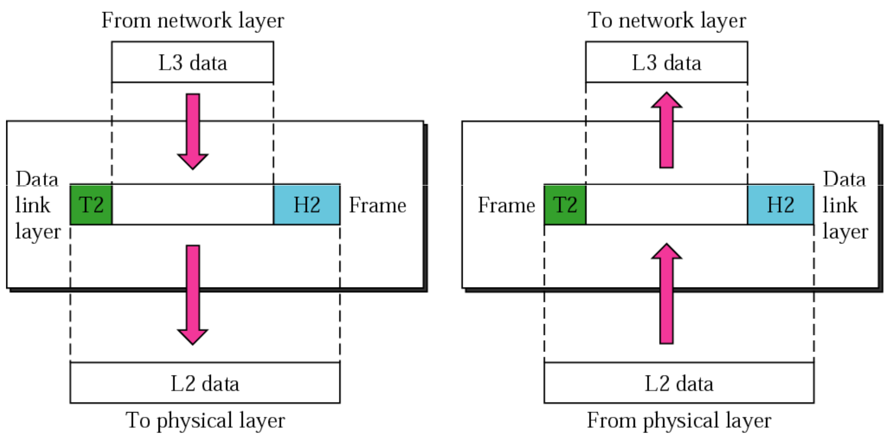
Data Link Layer: Node-to-Node Delivery



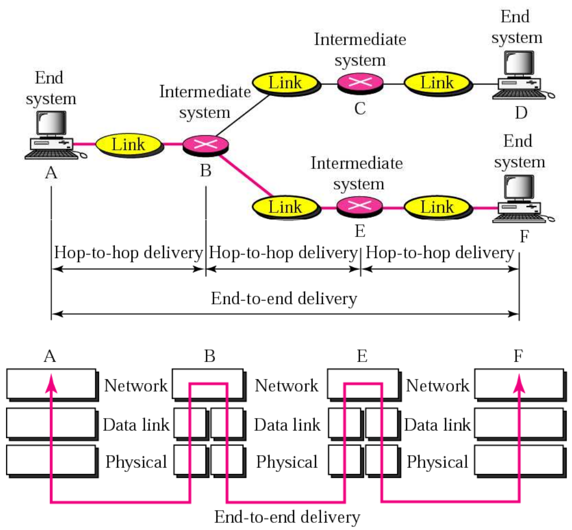
Data Link Layer

- Transforms the physical layer – which is error prone - to a reliable link
- Responsibilities
  - Framing
  - Physical Addressing
  - Flow control
  - Error Control
  - Access Control
    - Important in broadcast network

Data Link Layer



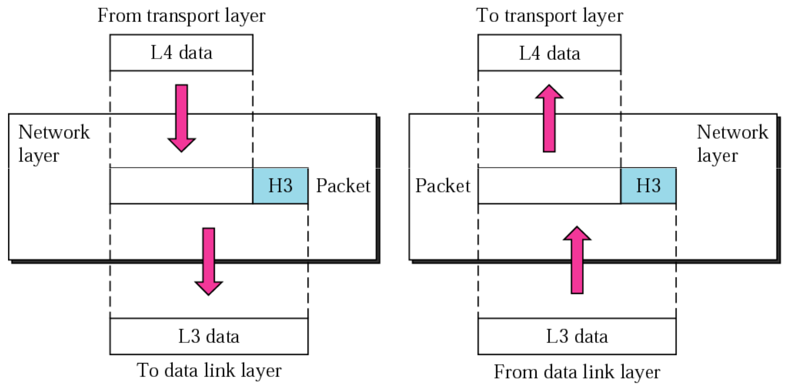
Network Layer: End-to-end delivery



Network Layer

- Source to destination delivery of packets
- Responsibilities
  - Logical Addressing
  - Routing

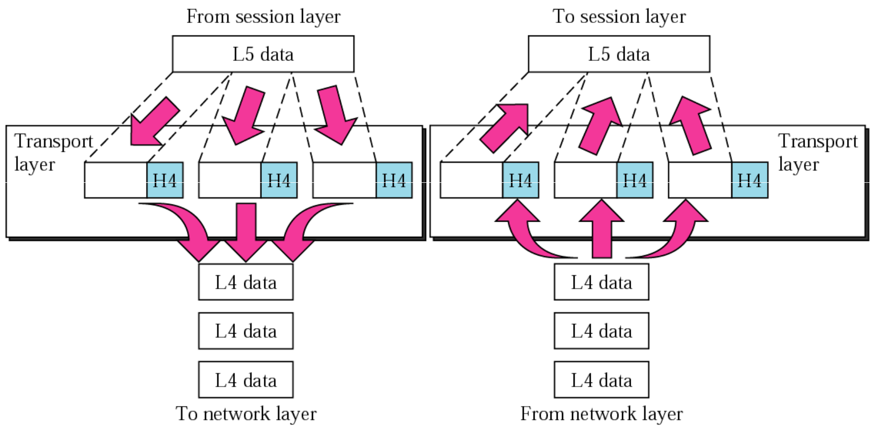
Network Layer



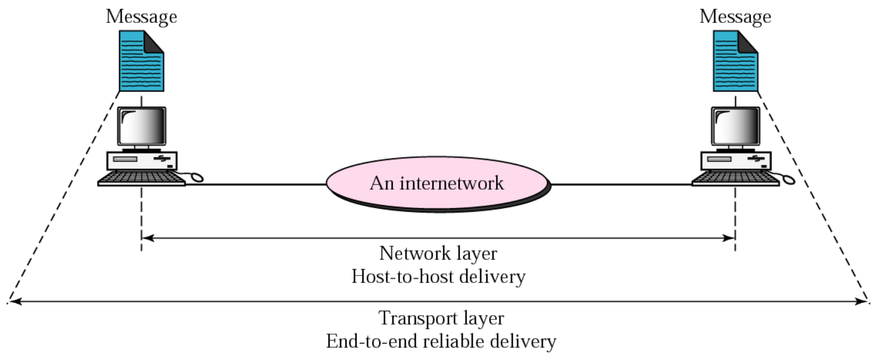
Transport Layer

- Process to process delivery of messages
- Responsibilities
  - Service-point addressing
  - Segmentation and reassembly
  - Connection control
  - Flow control
  - Error control

Transport Layer



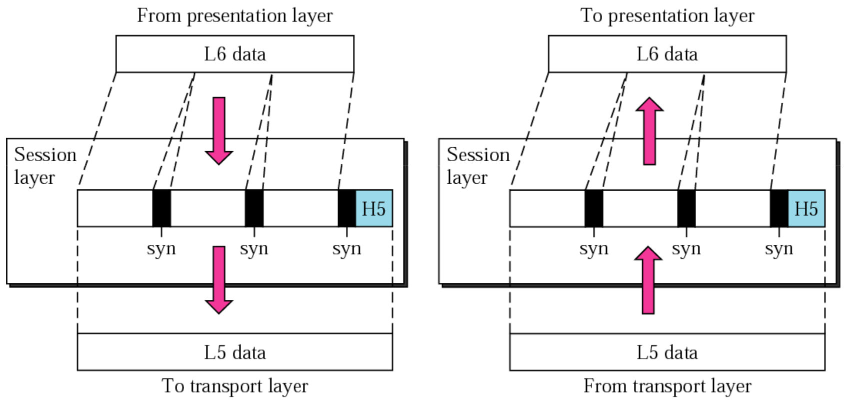
Process-to-Process Communication



Session Layer

- Session Management
- Responsibilities
  - Dialog Control
  - Synchronization

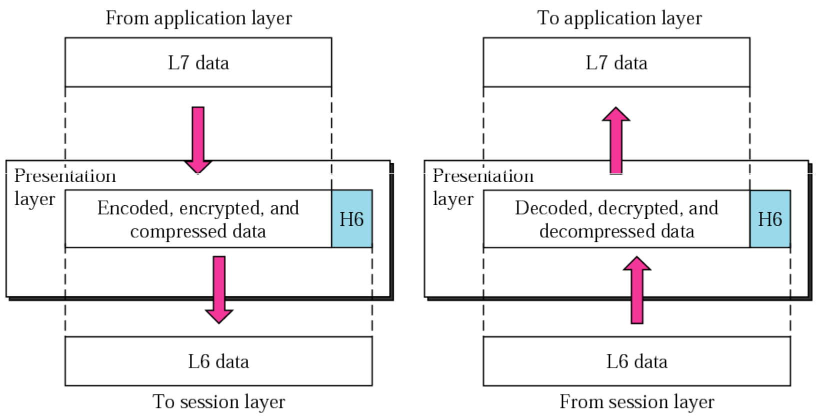
Session Layer



Presentation Layer

- Deals with syntax and semantics of exchanged information
  - Make communication not dependent from data encoding
- Responsibilities
  - Translation
  - Encryption
  - Compression

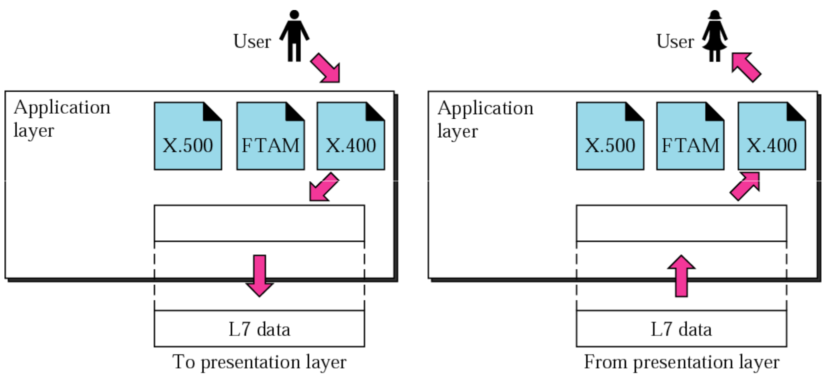
Presentation Layer



Application Layer

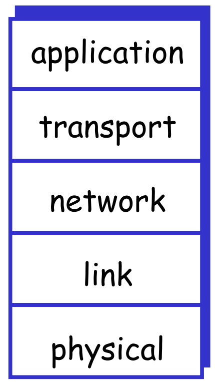
- Enables users to access the network
  - Providing user interfaces and support for services
- Specific services:
  - Network virtual terminal
  - File transfer, access and management
  - Mail services
  - Directory services

Application Layer



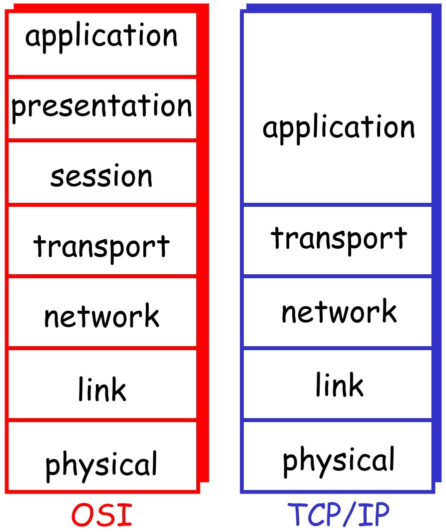
Internet protocol stack

- application: supporting network applications
  - FTP, SMTP, HTTP
- transport: app-to-app data transfer
  - TCP, UDP
- network: routing of datagrams from source to destination
  - IP, routing protocols
- link: data transfer between neighboring network elements
  - PPP, Ethernet
- physical: bits "on the wire"



OSI vs. TCP/IP

- No presentation/ session layers in TCP/IP
- OSI provide connection and connectionless service at all layers
  - TCP/IP network layer is connectionless (IP)



Internet History

1961-1972: Early packet-switching principles

- 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
- 1964: Baran - packet-switching in military nets
- 1967: ARPAnet conceived by Advanced Research Projects Agency
- 1969: first ARPAnet node operational
- 1972:
  - ARPAnet demonstrated publicly
  - NCP (Network Control Protocol) first host-host protocol
  - first e-mail program
  - ARPAnet has 15 nodes

Internet History

1972-1980: Internetworking, new and proprietary nets

- 1970: ALOHAnet satellite network in Hawaii
- 1973: Metcalfe's PhD thesis proposes Ethernet
- 1974: Cerf and Kahn - architecture for interconnecting networks
- late70's: proprietary architectures: DECnet, SNA, XNA
- late 70's: switching fixed length packets (ATM precursor)
- 1979: ARPAnet has 200 nodes

Cerf and Kahn's internetworking principles:
- minimalism, autonomy - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

define today's Internet architecture

Internet History

1980-1990: new protocols, a proliferation of networks

- 1983: deployment of TCP/IP
- 1982: SMTP e-mail protocol defined
- 1983: DNS defined for name-to-IP-address translation
- 1985: FTP protocol defined
- 1988: TCP congestion control
- new national networks: Csnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks

Internet History

1990, 2000's: commercialization, the Web, new apps

- Early 1990's: ARPAnet decommissioned
- 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- early 1990s: Web
  - hypertext [Bush 1945, Nelson 1960's]
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, later Netscape
  - late 1990's: commercialization of the Web

Late 1990's – 2000's:

- more killer apps: instant messaging, peer2peer file sharing (e.g., Naptser)
- network security to forefront
- est. 50 million host, 100 million+ users
- backbone links running at Gbps