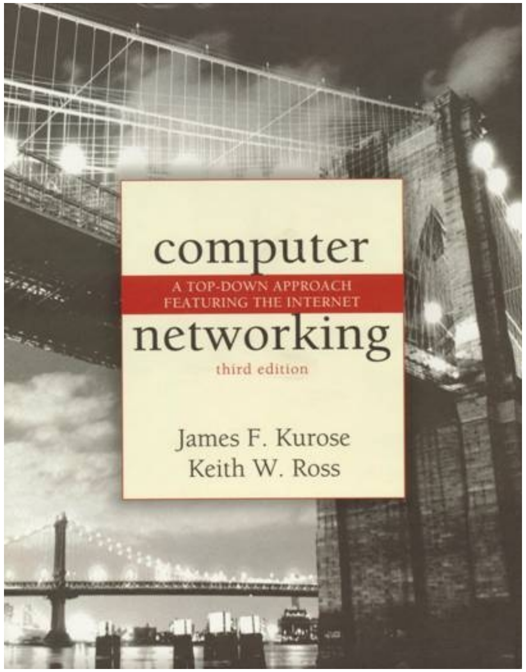
# Chapter 1
# Computer Networks and the Internet




*Computer Networking: A Top Down Approach Featuring the Internet,*

3rd edition.
Jim Kurose, Keith Ross
Addison-Wesley, July 2004.

## A note on the use of these ppt slides:

We're making these slides freely available to all (faculty, students, readers). They're in powerpoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we'd like people to use our book!)
- If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

# Chapter 1: Introduction

## Our goal:

- get context, overview, "feel" of networking
- more depth, detail *later* in course
- approach:
  - descriptive
  - use Internet as example

## Overview:

- what's the Internet
- what's a protocol?
- network edge
- network core
- access net, physical media
- Internet/ISP structure
- performance: loss, delay
- protocol layers, service models
- history

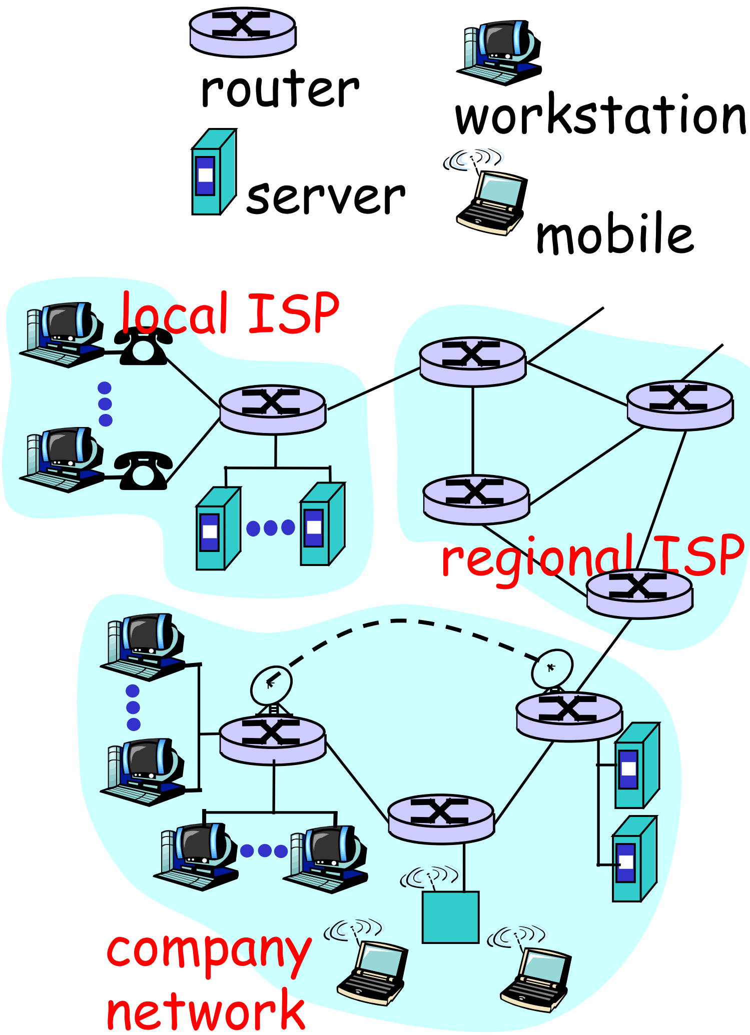
# Chapter 1: roadmap

1.1 What *is* the Internet?
1.2 Network edge
1.3 Network core
1.4 Network access and physical media
1.5 Internet structure and ISPs
1.6 Delay & loss in packet-switched networks
1.7 Protocol layers, service models
1.8 History

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

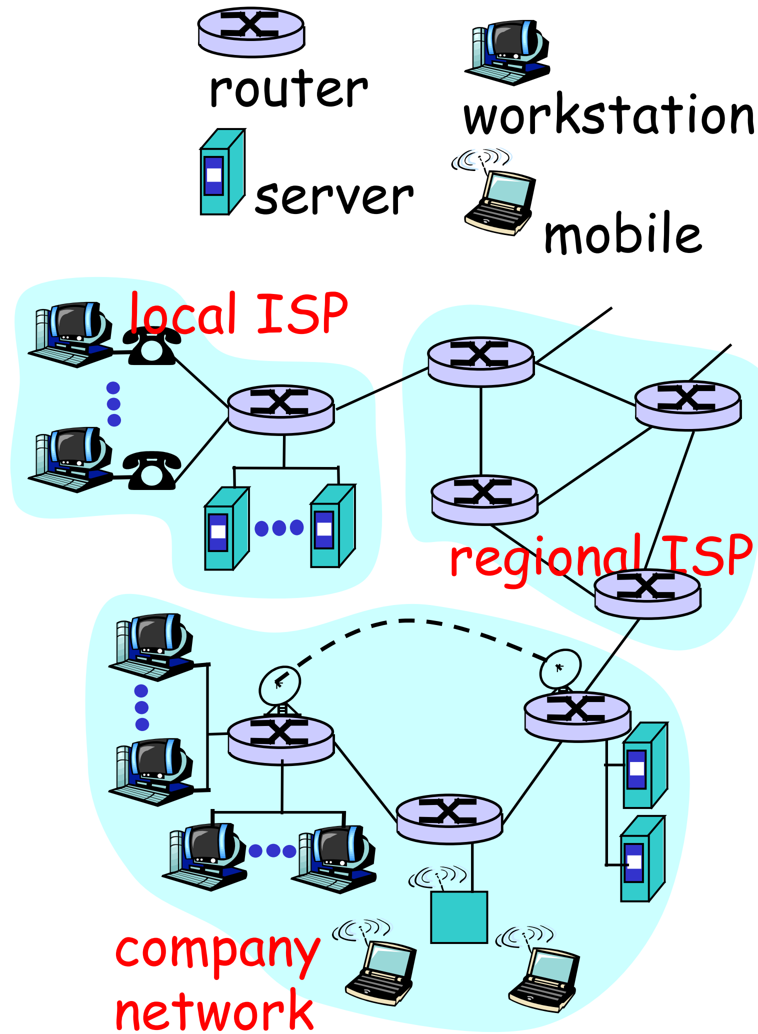
- millions of connected computing devices: *hosts = end-systems*
  - PCs workstations, servers
  - PDAs phones, toasters

  running *network apps*
- *communication links*
  - fiber, copper, radio, satellite
  - transmission rate = *bandwidth*
- *routers:* forward packets (chunks of data)



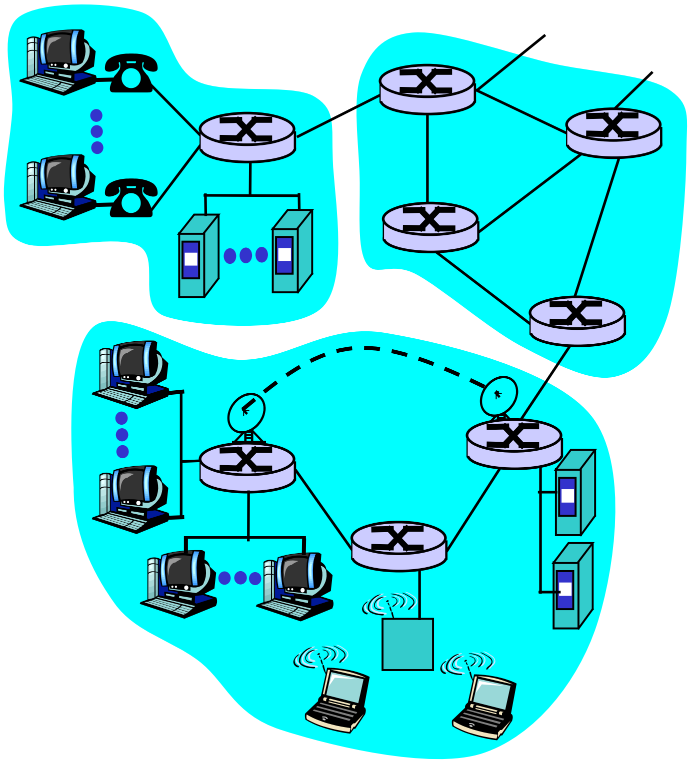
# What's the Internet: "nuts and bolts" view

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



# What's the Internet: a service view

- communication *infrastructure* enables distributed applications:
  - Web, email, games, e-commerce, database., voting, file (MP3) sharing
- communication services provided to applications:
  - connectionless
  - connection-oriented



# What's a protocol?

human protocols:

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

… specific messages sent

… specific actions taken when messages received, or other events

network protocols:

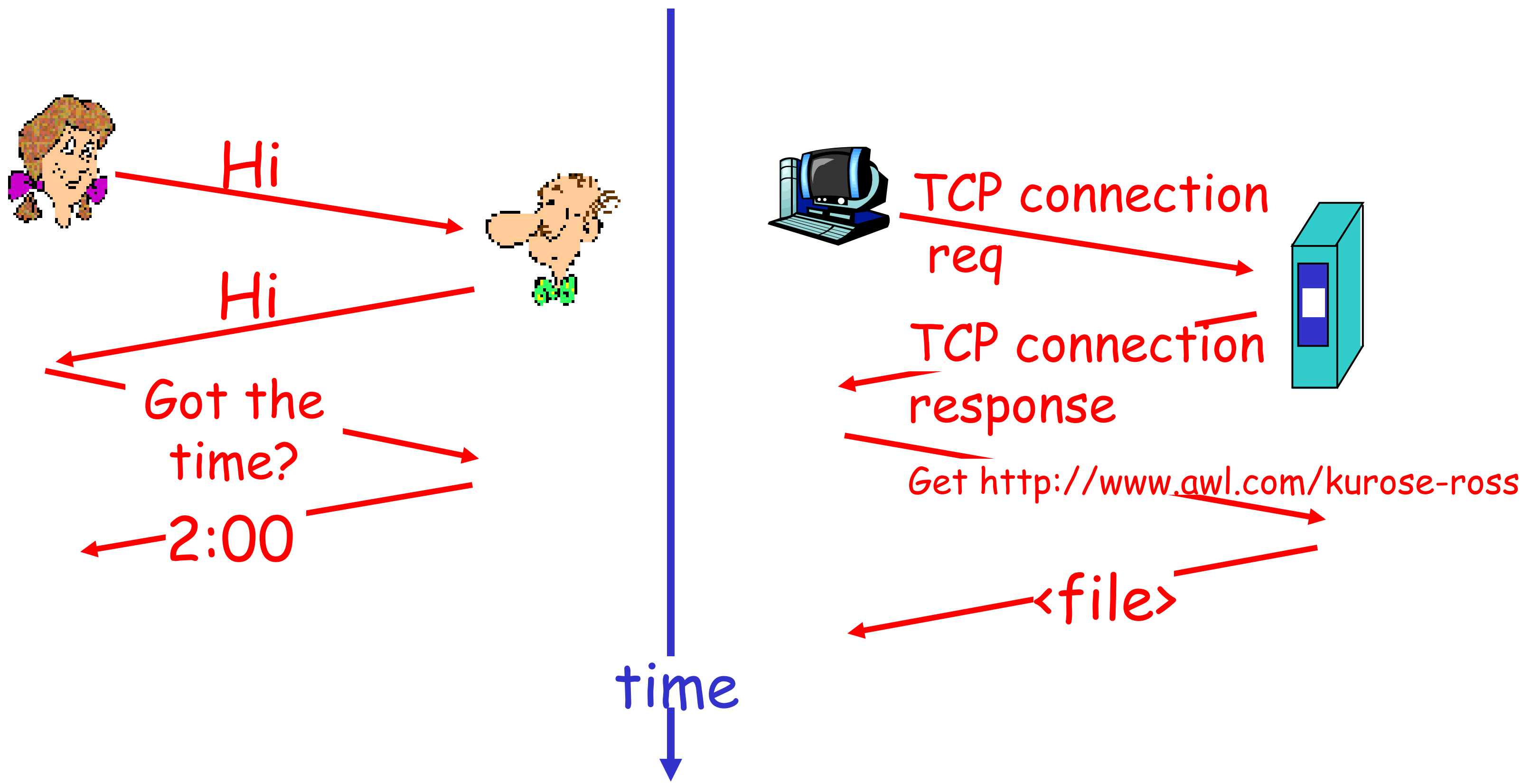
- machines rather than humans
- all communication activity in Internet governed by protocols

*protocols define*

- *format, order of messages sent and received among network entities, and*
- *actions taken on the transmission/receipt of a message*

# What's a protocol?

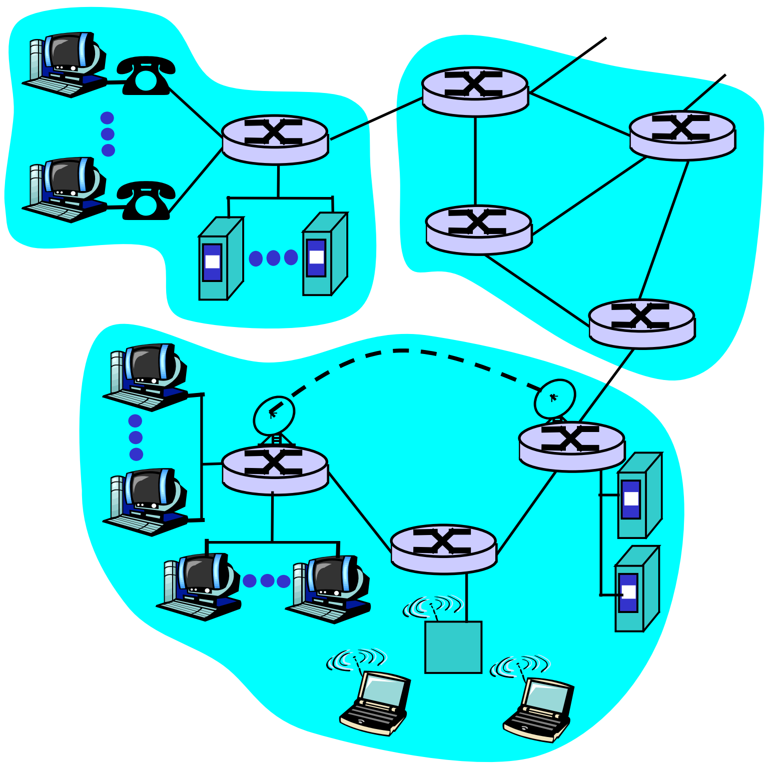
a human protocol and a computer network protocol:




Q: Other human protocols?

# A closer look at network structure:

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

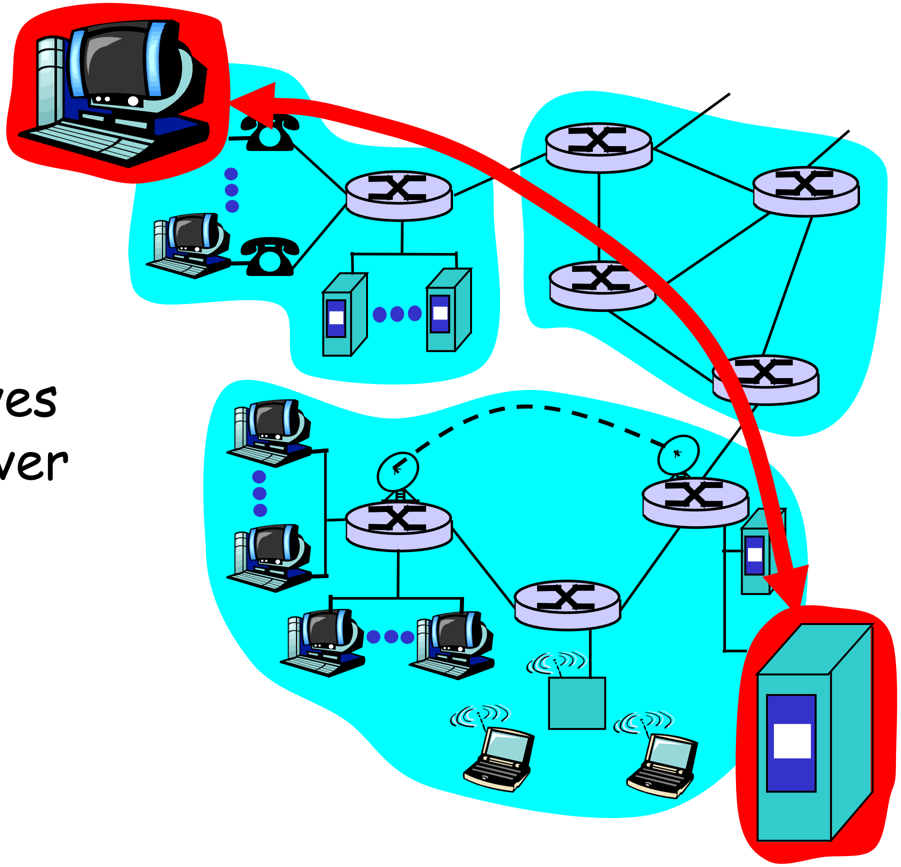


# Chapter 1: roadmap

1.1 What *is* the Internet?
1.2 Network edge
1.3 Network core
1.4 Network access and physical media
1.5 Internet structure and ISPs
1.6 Delay & loss in packet-switched networks
1.7 Protocol layers, service models
1.8 History

# 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. Gnutella, KaZaA



# Network edge: connection-oriented service

*Goal:* data transfer between end systems

- *handshaking:* setup (prepare for) data transfer ahead of time
  - Hello, hello back human protocol
  - *set up "state"* in two communicating hosts
- TCP - Transmission Control Protocol
  - Internet's connection-oriented service

## TCP service [RFC 793]

- *reliable, in-order* byte-stream data transfer
  - loss: acknowledgements and retransmissions
- *flow control:*
  - sender won't overwhelm receiver
- *congestion control:*
  - senders "slow down sending rate" when network congested

# Network edge: connectionless service

**Goal:** data transfer between end systems

- same as before!

- **UDP** - User Datagram Protocol [RFC 768]: Internet's connectionless service
  - unreliable data transfer
  - no flow control
  - no congestion control

**App's using TCP:**

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

**App's using UDP:**

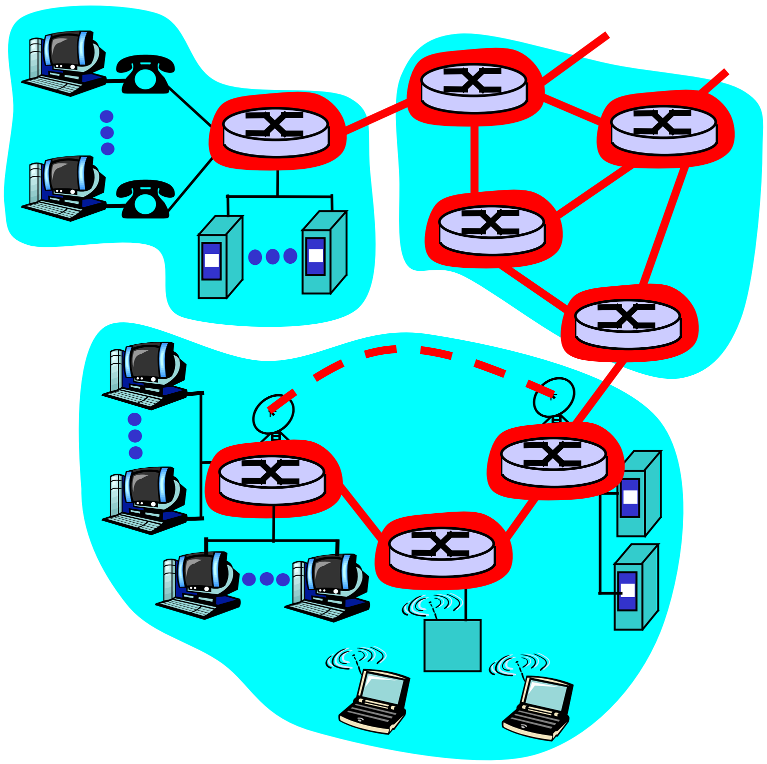
- streaming media, teleconferencing, DNS, Internet telephony

# Chapter 1: roadmap

1.1 What *is* the Internet?
1.2 Network edge
1.3 Network core
1.4 Network access and physical media
1.5 Internet structure and ISPs
1.6 Delay & loss in packet-switched networks
1.7 Protocol layers, service models
1.8 History

# The Network Core

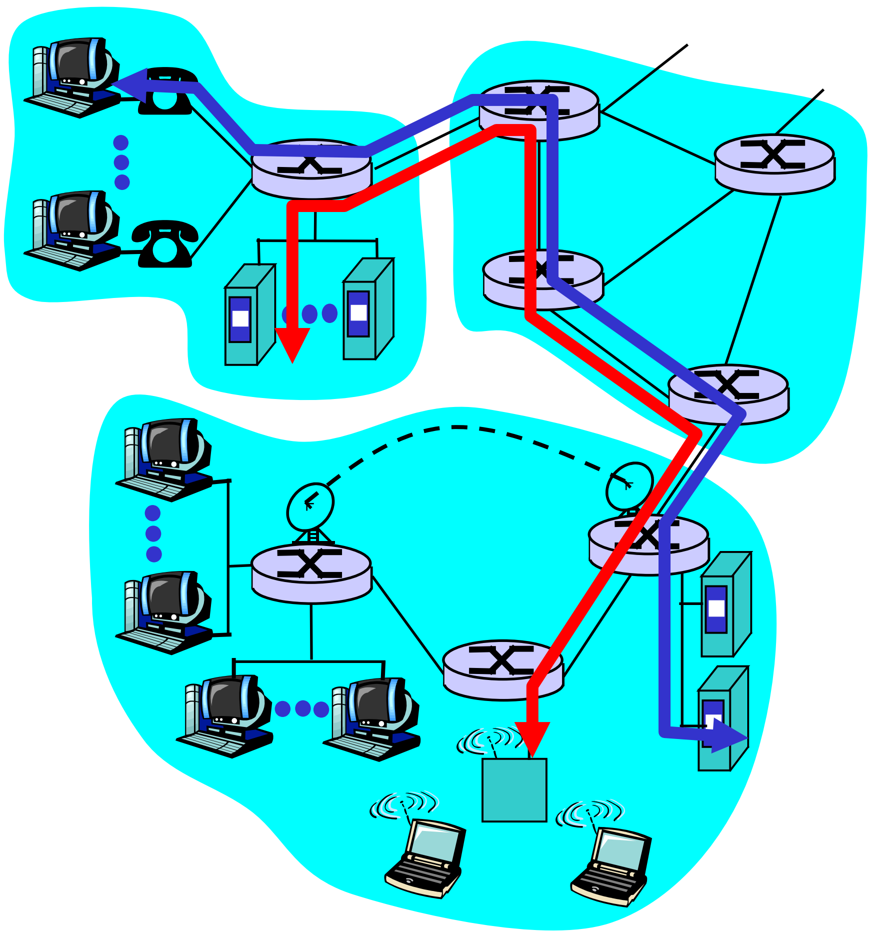
- mesh of interconnected routers
- *the* fundamental question: how is data transferred through net?
  - circuit switching: dedicated circuit per call: telephone net
  - packet-switching: data sent thru net in discrete "chunks"



# 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

network resources (e.g., bandwidth) divided into "pieces"

- pieces allocated to calls
- resource piece *idle* if not used by owning call *(no sharing)*

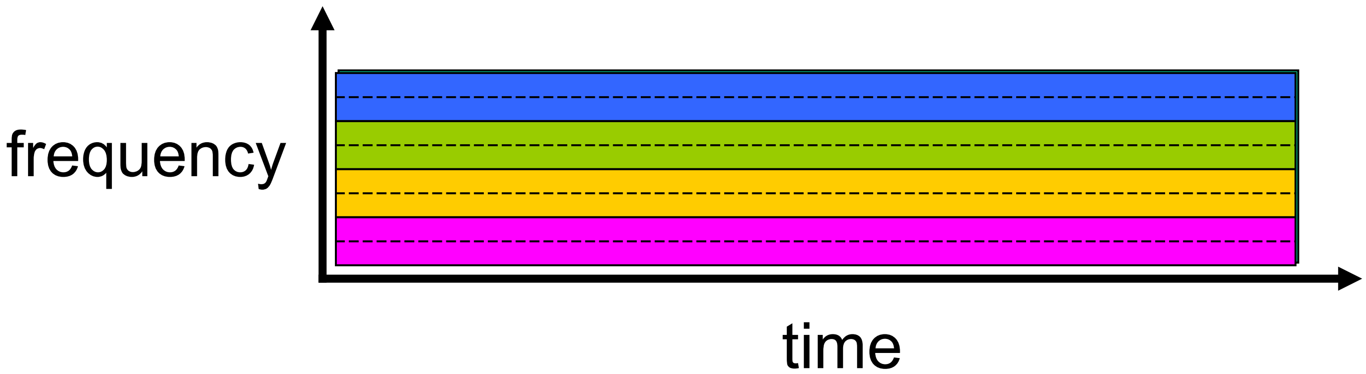
- dividing link bandwidth into "pieces"
  - frequency division
  - time division

# Circuit Switching: FDMA and TDMA

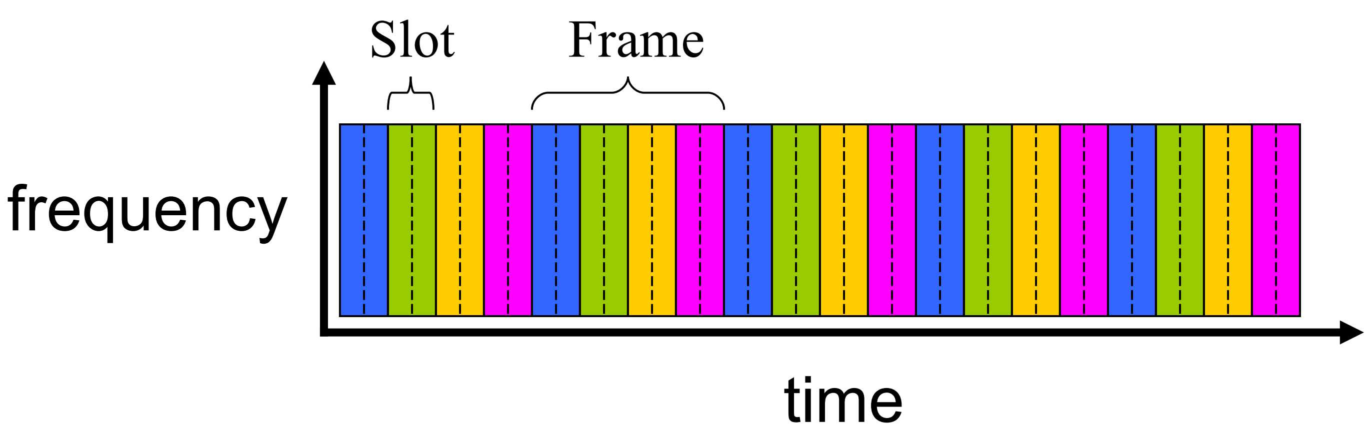
Example:

4 users

FDMA




TDMA



# Network Core: Packet Switching

each end-end data stream divided into *packets*

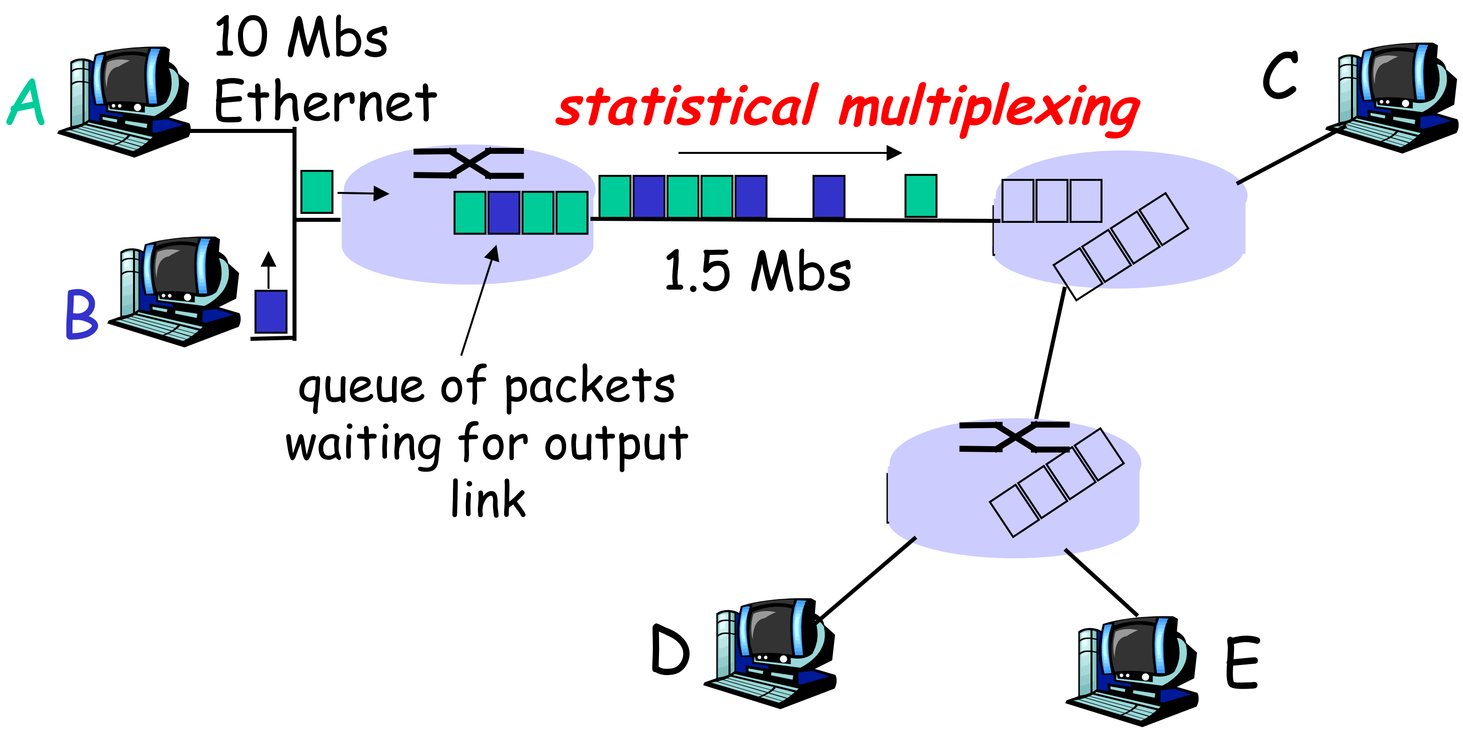
- user A, B packets *share* network resources
- each packet uses full link bandwidth
- resources used *as needed*

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

resource contention:

- aggregate resource demand can exceed amount available
- congestion: packets queue, wait for link use
- store and forward: packets move one hop at a time
  - transmit over link
  - wait turn at next link

# Packet Switching: Statistical Multiplexing



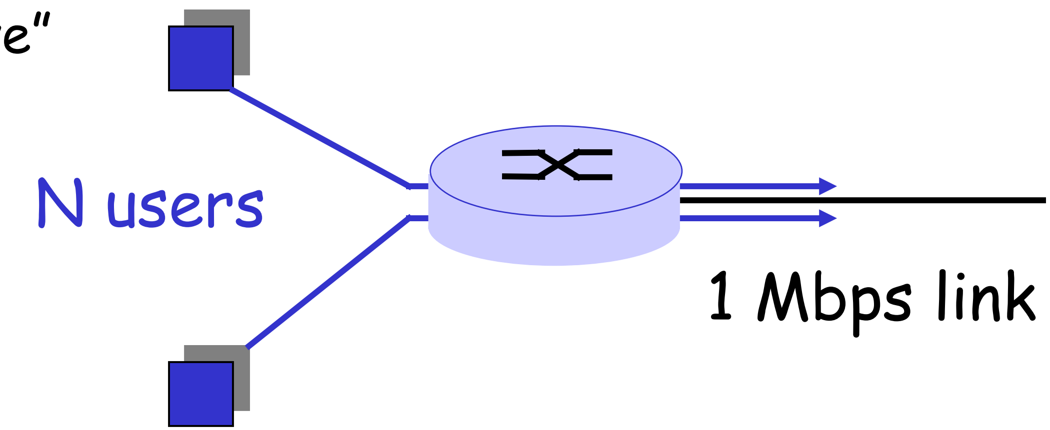

Sequence of A & B packets does not have fixed pattern  *statistical multiplexing*.

In TDM each host gets same slot in revolving TDM frame.

# 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

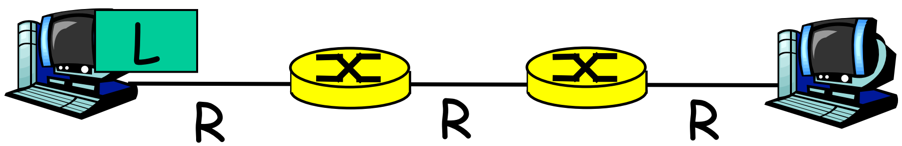


# Packet switching versus circuit switching

Is packet switching a "slam dunk winner?"

- 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: store-and-forward

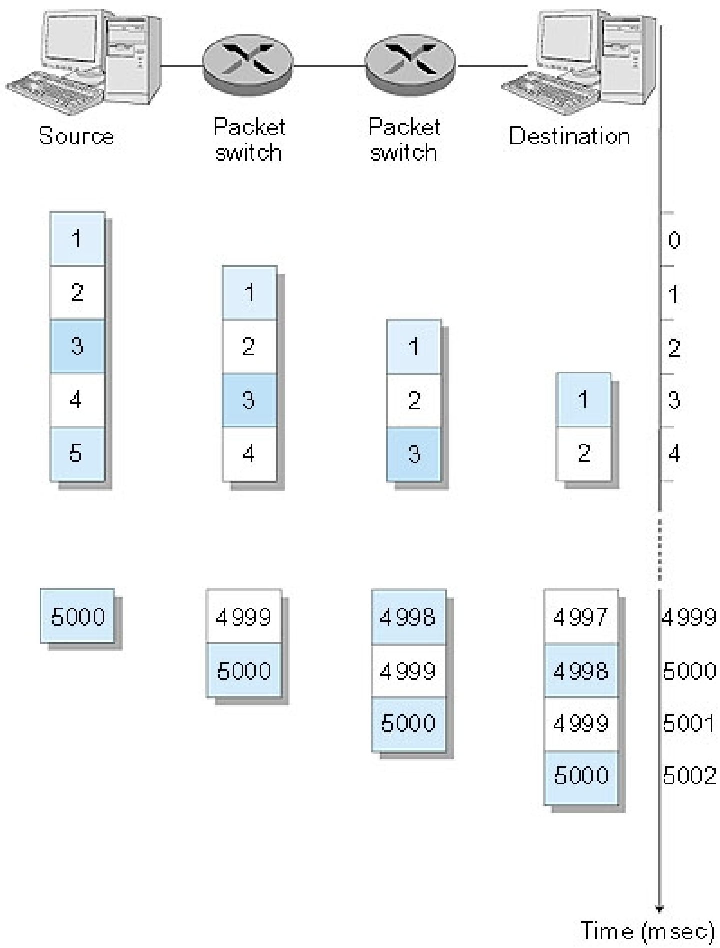



- Takes L/R seconds to transmit (push out) packet of L bits onto a link of R bps
- Entire packet must arrive at router before it can be transmitted on next link: *store and forward*
- delay = 3L/R

Example:

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

# Packet Switching: Message Segmenting



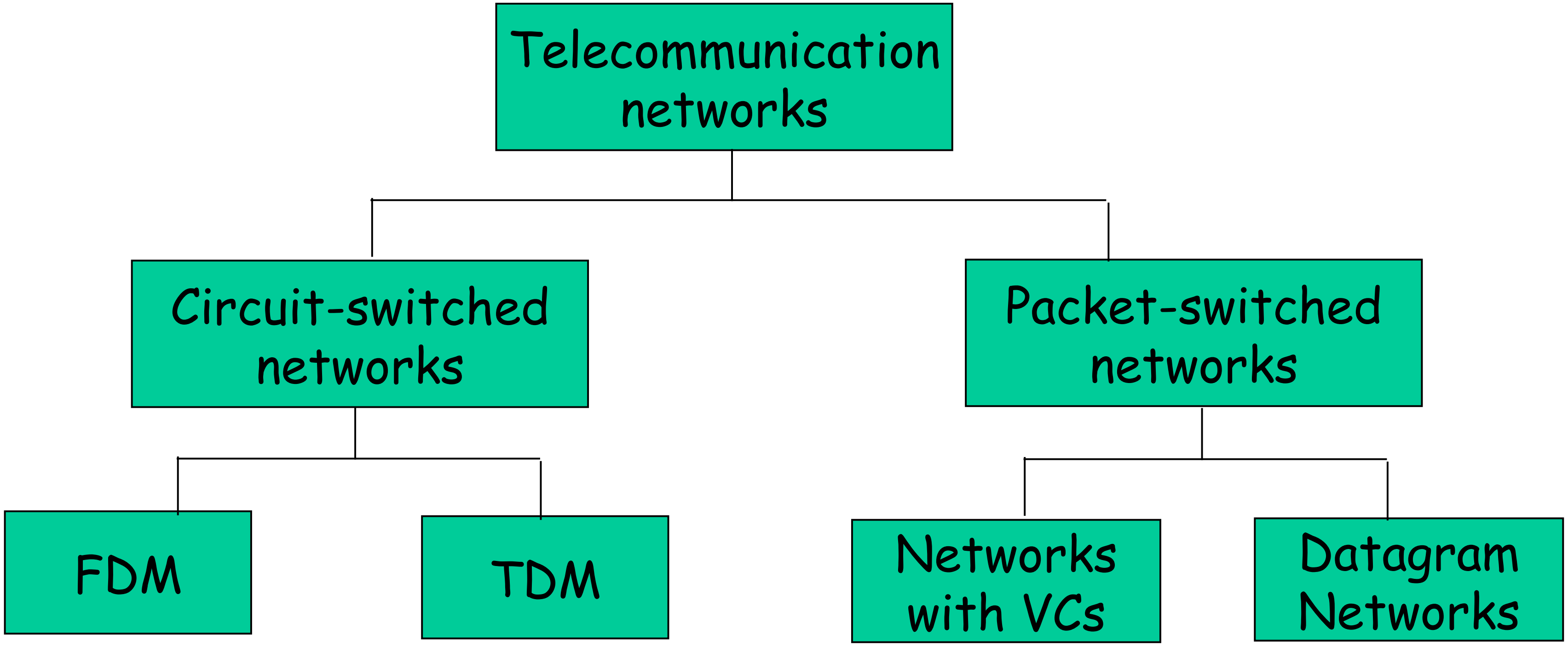

Now break up the message into 5000 packets

- 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-switched networks: forwarding

- *Goal:* move packets through routers from source to destination
  - we'll study several path selection (i.e. routing) algorithms (chapter 4)
- datagram network:
  - *destination address* in packet determines next hop
  - routes may change during session
  - analogy: driving, asking directions
- virtual circuit network:
  - each packet carries tag (virtual circuit ID), tag determines next hop
  - fixed path determined at *call setup time*, remains fixed thru call
  - *routers maintain per-call state*

# Network Taxonomy




- Datagram network is *not* either connection-oriented or connectionless.
- Internet provides both connection-oriented (TCP) and connectionless services (UDP) to apps.

# Chapter 1: roadmap

1.1 What *is* the Internet?
1.2 Network edge
1.3 Network core
1.4 Network access and physical media
1.5 Internet structure and ISPs
1.6 Delay & loss in packet-switched networks
1.7 Protocol layers, service models
1.8 History

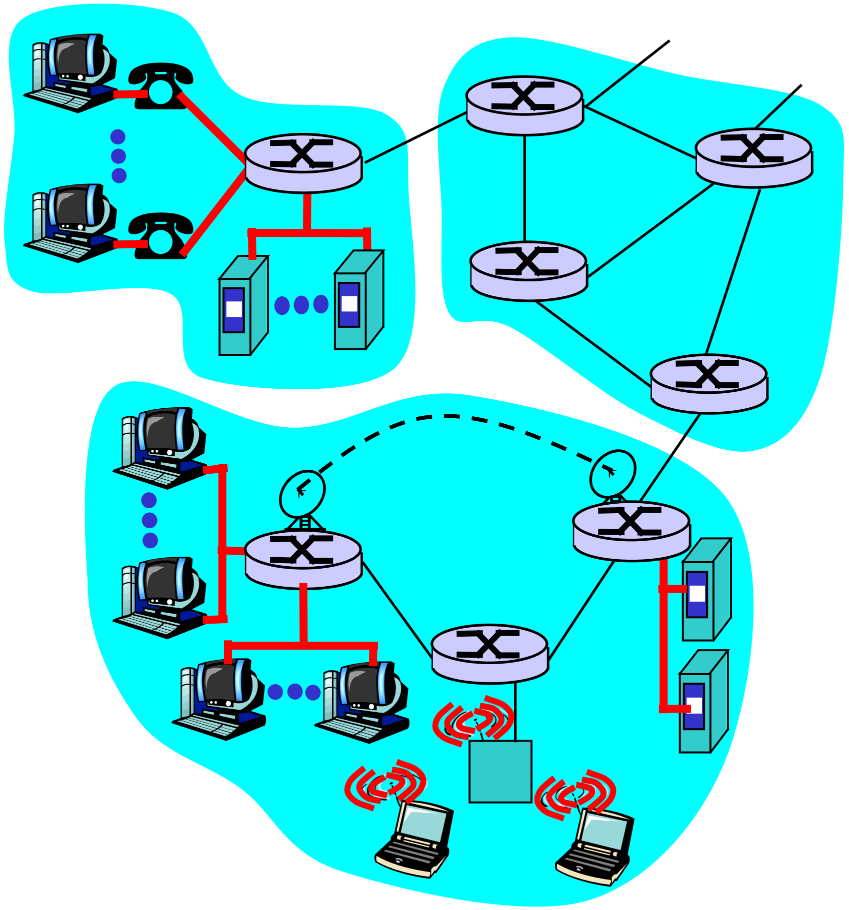
# Access networks and physical media

*Q: How to connect end systems to edge router?*

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

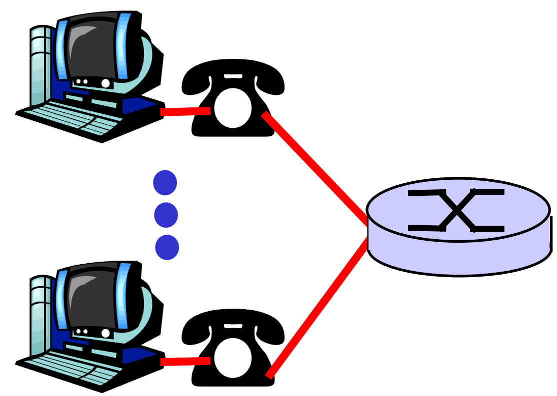
*Keep in mind:*

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



# 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"



- ADSL: asymmetric digital subscriber line
  - up to 1 Mbps upstream (today typically < 256 kbps)
  - up to 8 Mbps downstream (today typically < 1 Mbps)
  - FDM: 50 kHz - 1 MHz for downstream

    4 kHz - 50 kHz for upstream

    0 kHz - 4 kHz for ordinary telephone

# Residential access: cable modems

- HFC: hybrid fiber coaxial cable
  - asymmetric: up to 30Mbps downstream, 2 Mbps upstream
- network of cable and fiber attaches homes to ISP router
  - shared access to router among home
  - issues: congestion, dimensioning
- deployment: available via cable companies, e.g., MediaOne

# Residential access: cable modems

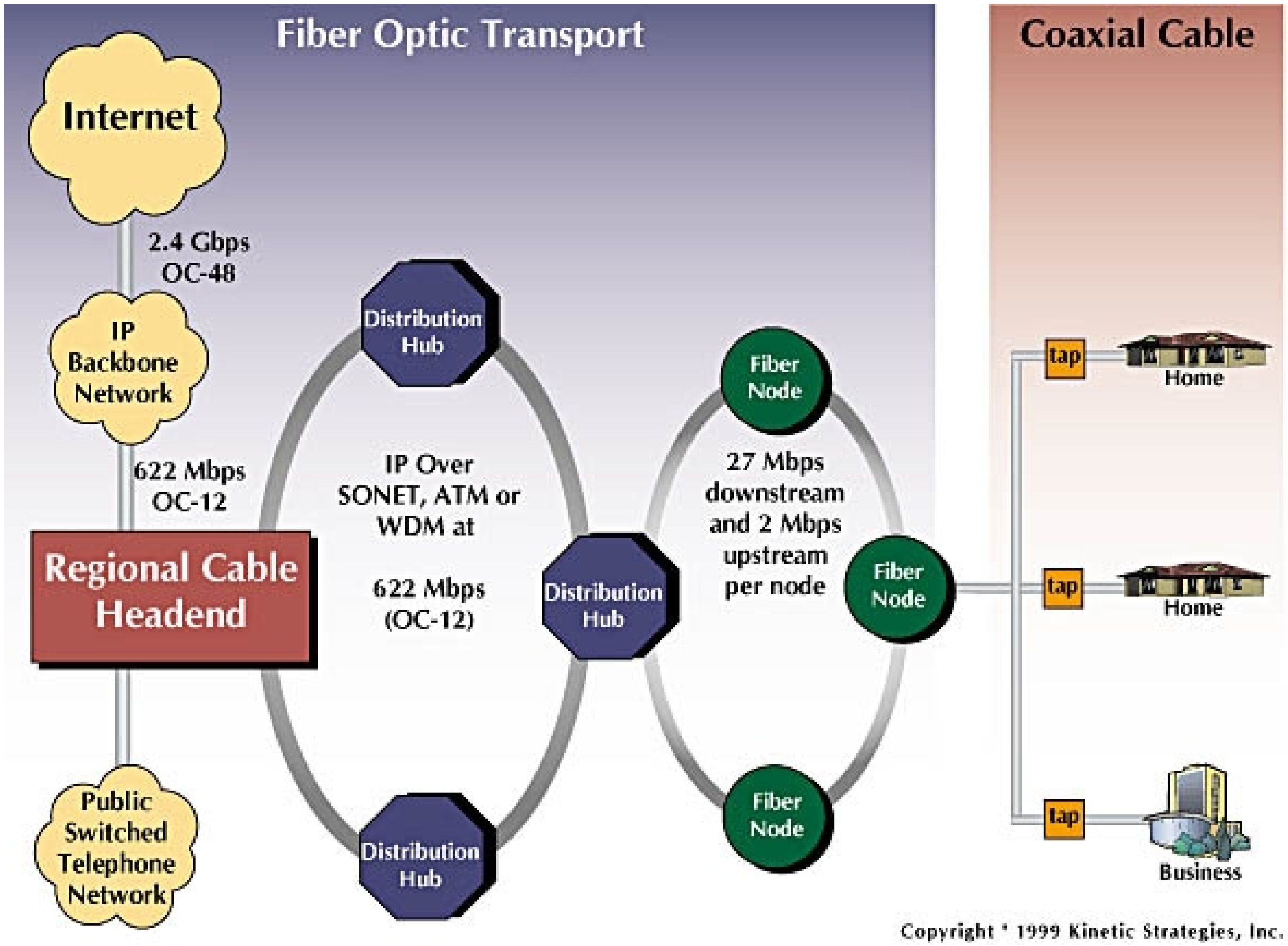
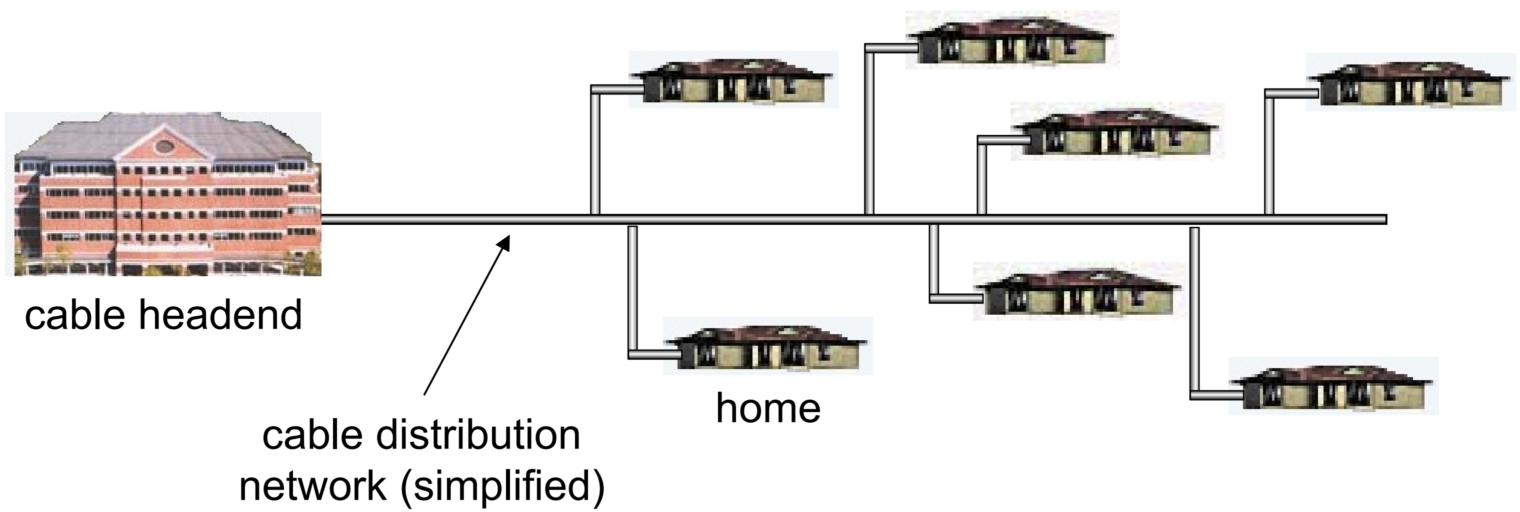


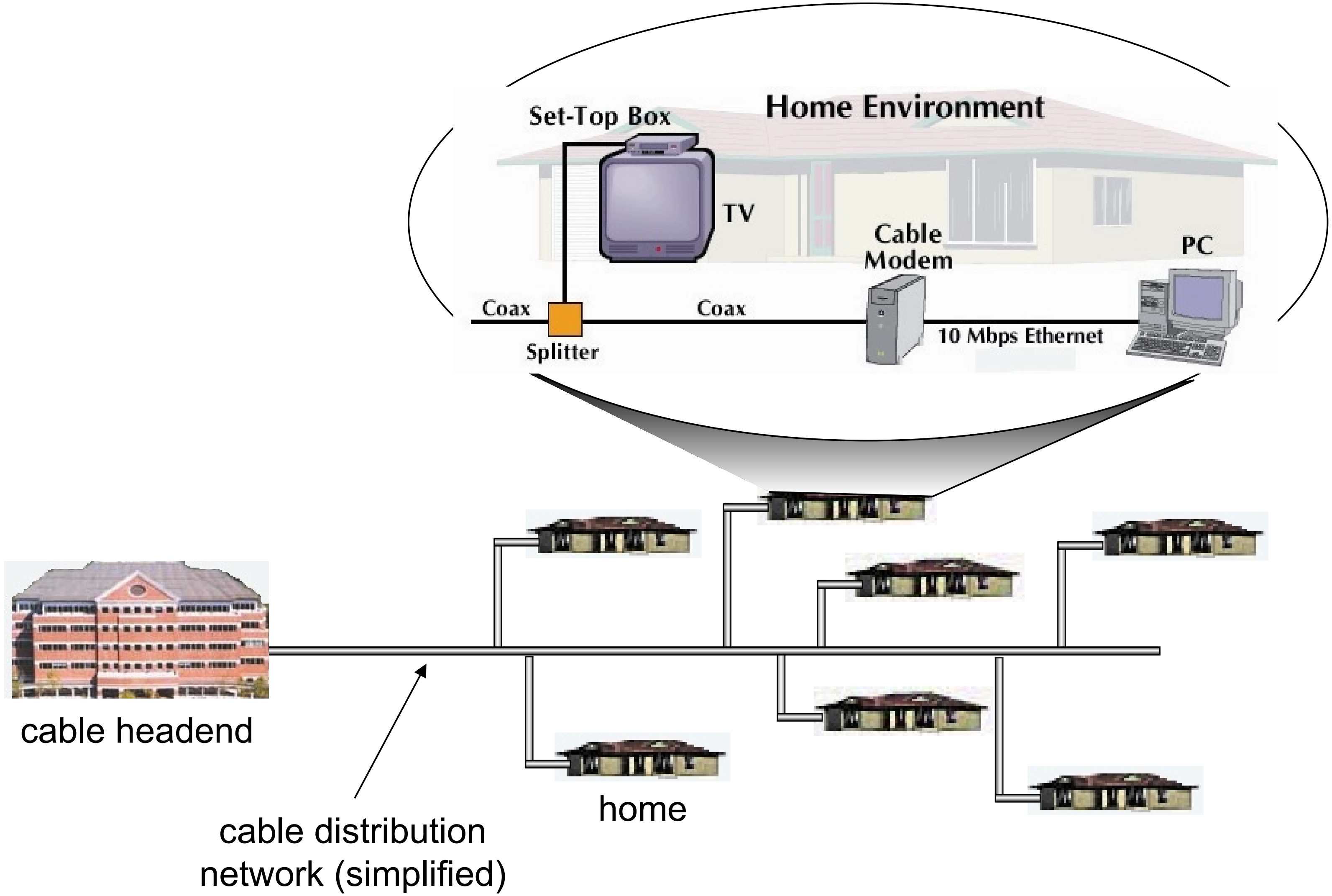

Diagram: http://www.cabledatacomnews.com/cmic/diagram.html

# Cabel Network Architecture: Overview

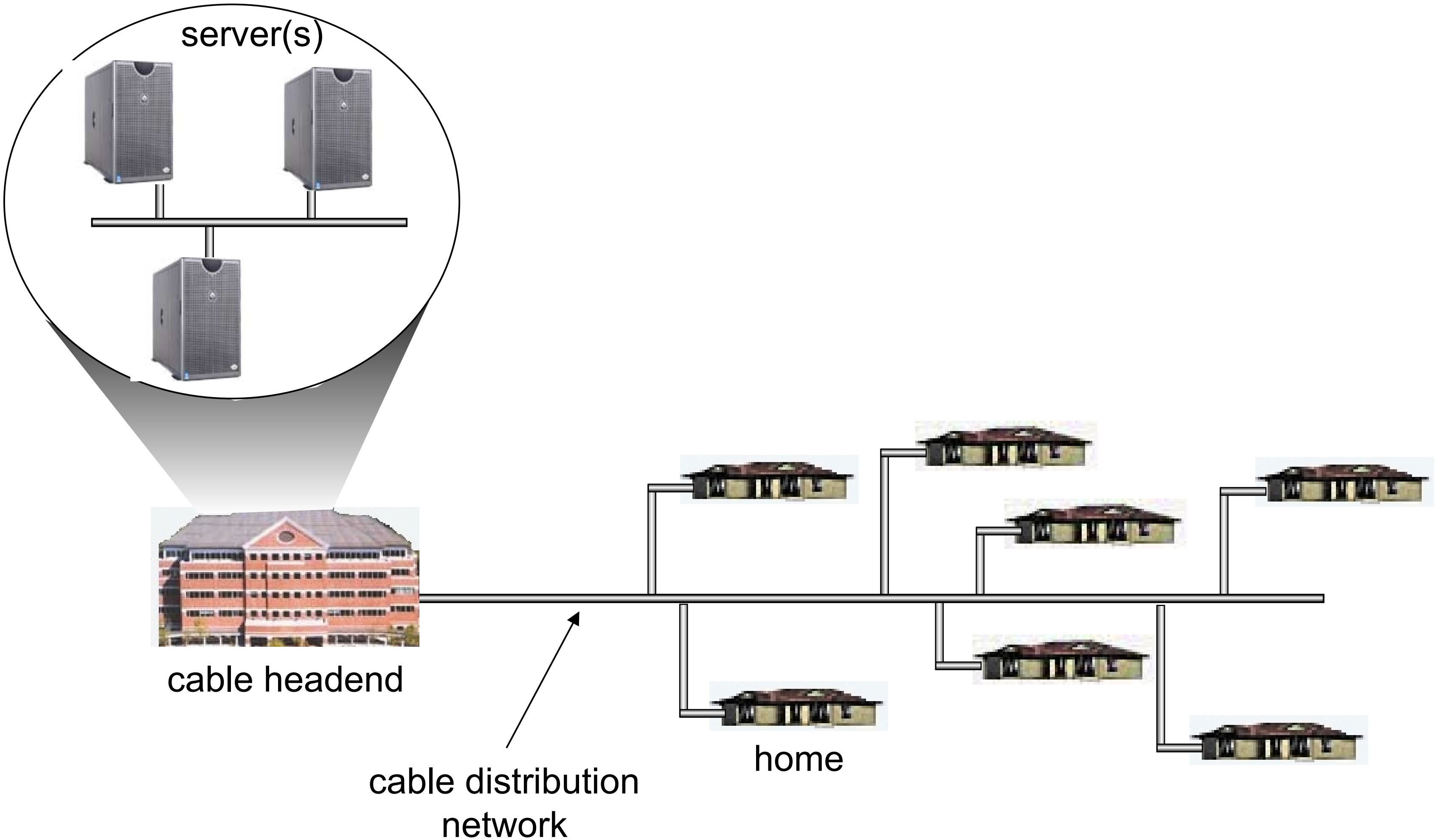
Typically 500 to 5,000 homes



# Cable Network Architecture: Overview

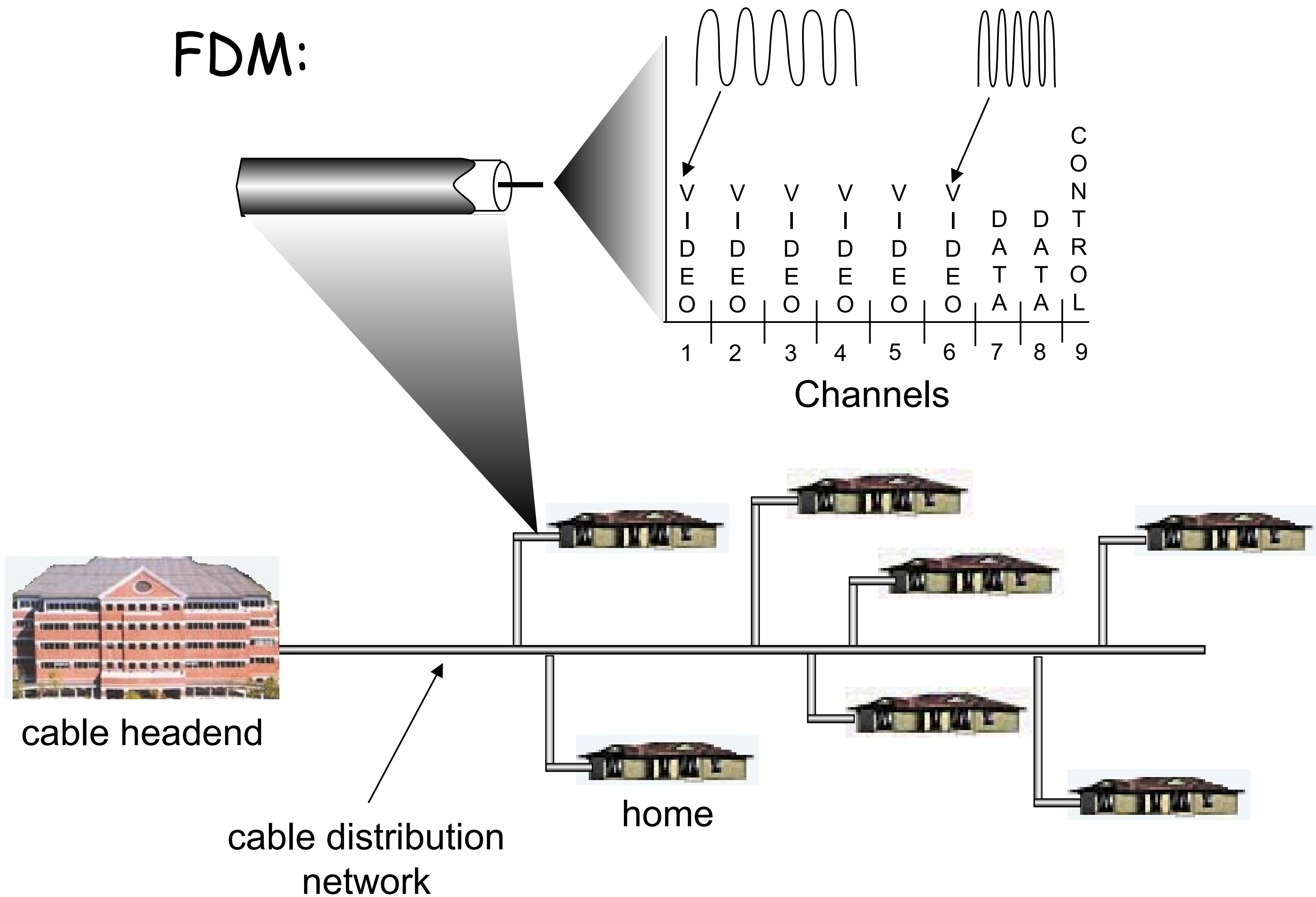


# Cable Network Architecture: Overview



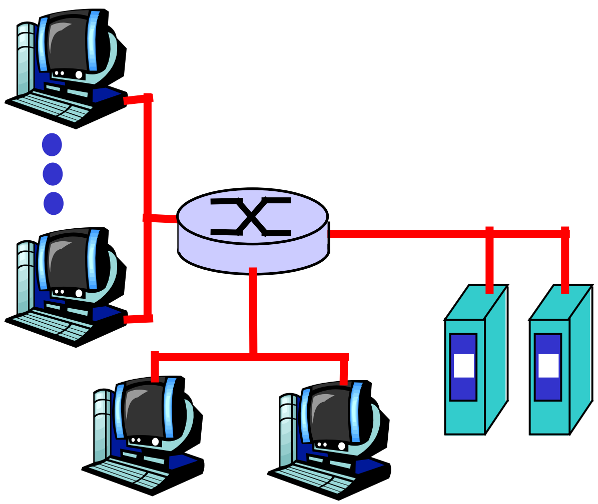
# Cable Network Architecture: Overview

FDM:



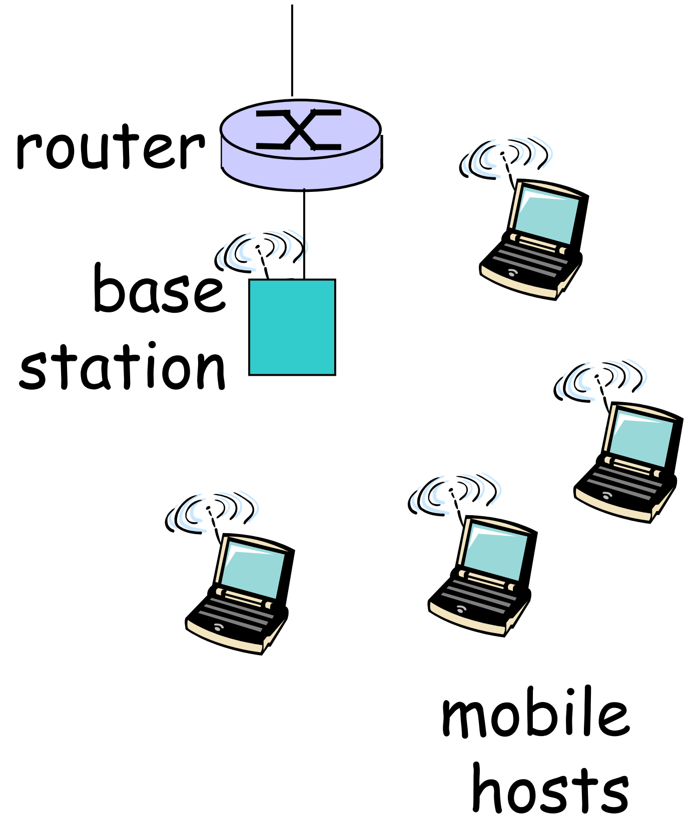
# Company access: local area networks

- company/university local area network (LAN) connects end system to edge router
- Ethernet:
  - shared or dedicated link connects end system and router
  - 10 Mbs, 100Mbps, Gigabit Ethernet
- deployment: institutions, home LANs happening now
- LANs: chapter 5



# Wireless access networks

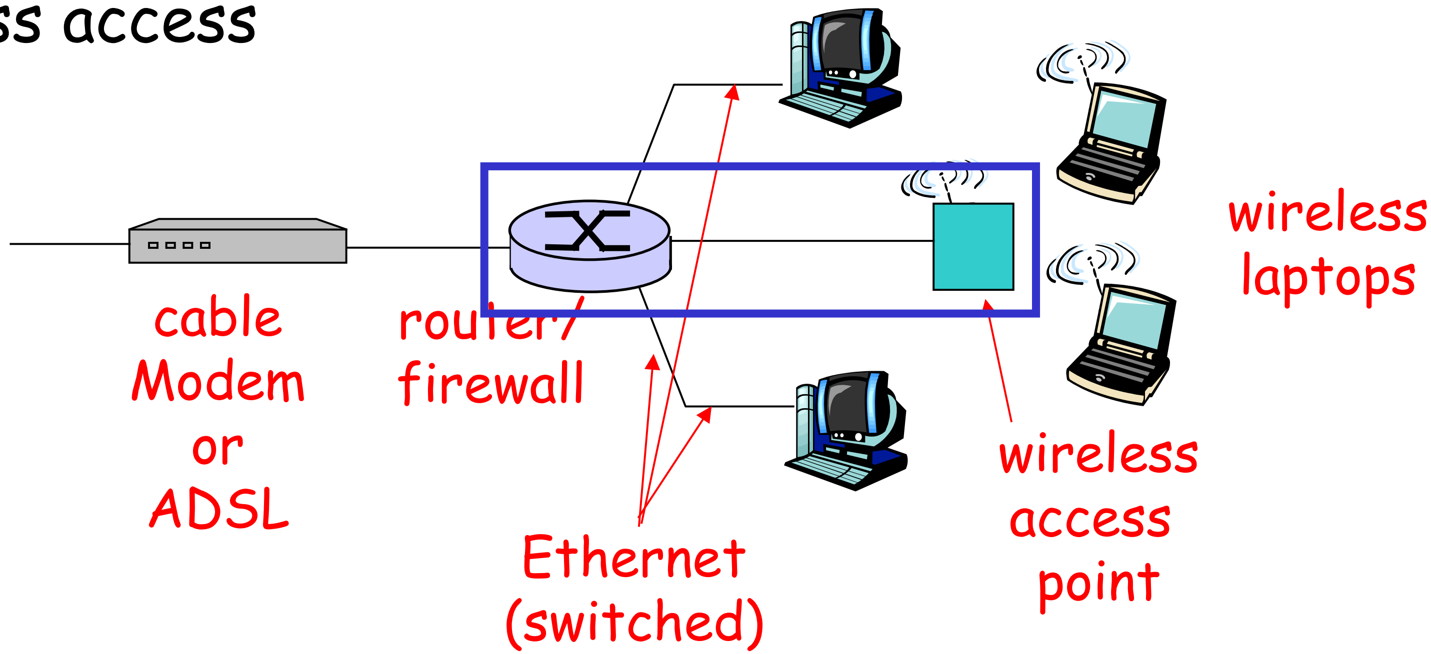
- shared *wireless* access network connects end system to router
  - via base station also known as "access point"
- wireless LANs:
  - 802.11b (WiFi): 11 Mbps
  - 802.11g: 54 Mbps
  - 802.11n: 108 Mbps
- wider-area wireless access
  - provided by telcom operator
  - 3G ~ 384 kbps
    - Will it happen??
  - 4G
  - WAP/GPRS in Europe
  - 802.16a/e (WiMax):



# Home networks

Typical home network components:

- ADSL or cable modem
- router/firewall/NAT
- Ethernet
- wireless access point



# 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

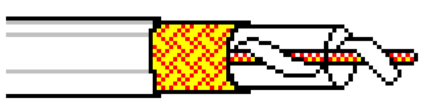
## Twisted Pair (TP)

- two insulated copper wires
  - Category 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 channels on cable
  - HFC



**Fiber optic cable:**

- glass fiber carrying light pulses, each pulse a bit
- high-speed operation:
  - high-speed point-to-point transmission (e.g., 5 Gps)
- 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)
  - 2Mbps, 11Mbps, 54Mbps, 108 Mbps
- 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 low-earth-orbit satellites (LEOS)

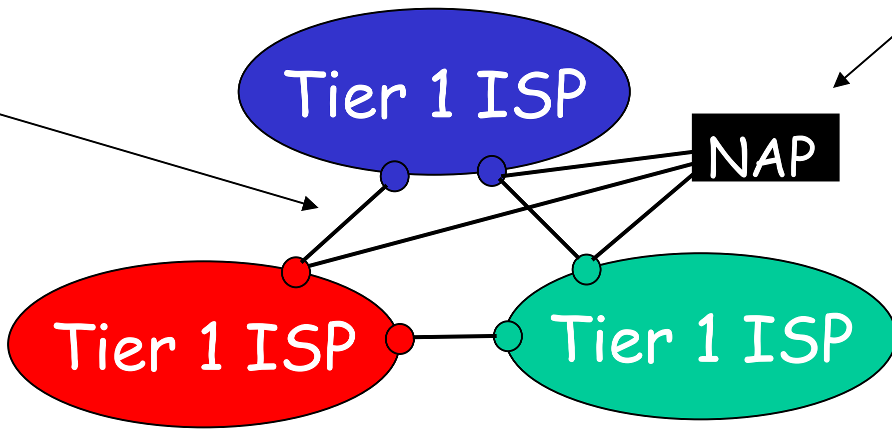
# Chapter 1: roadmap

1.1 What *is* the Internet?
1.2 Network edge
1.3 Network core
1.4 Network access and physical media
1.5 Internet structure and ISPs
1.6 Delay & loss in packet-switched networks
1.7 Protocol layers, service models
1.8 History

# 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

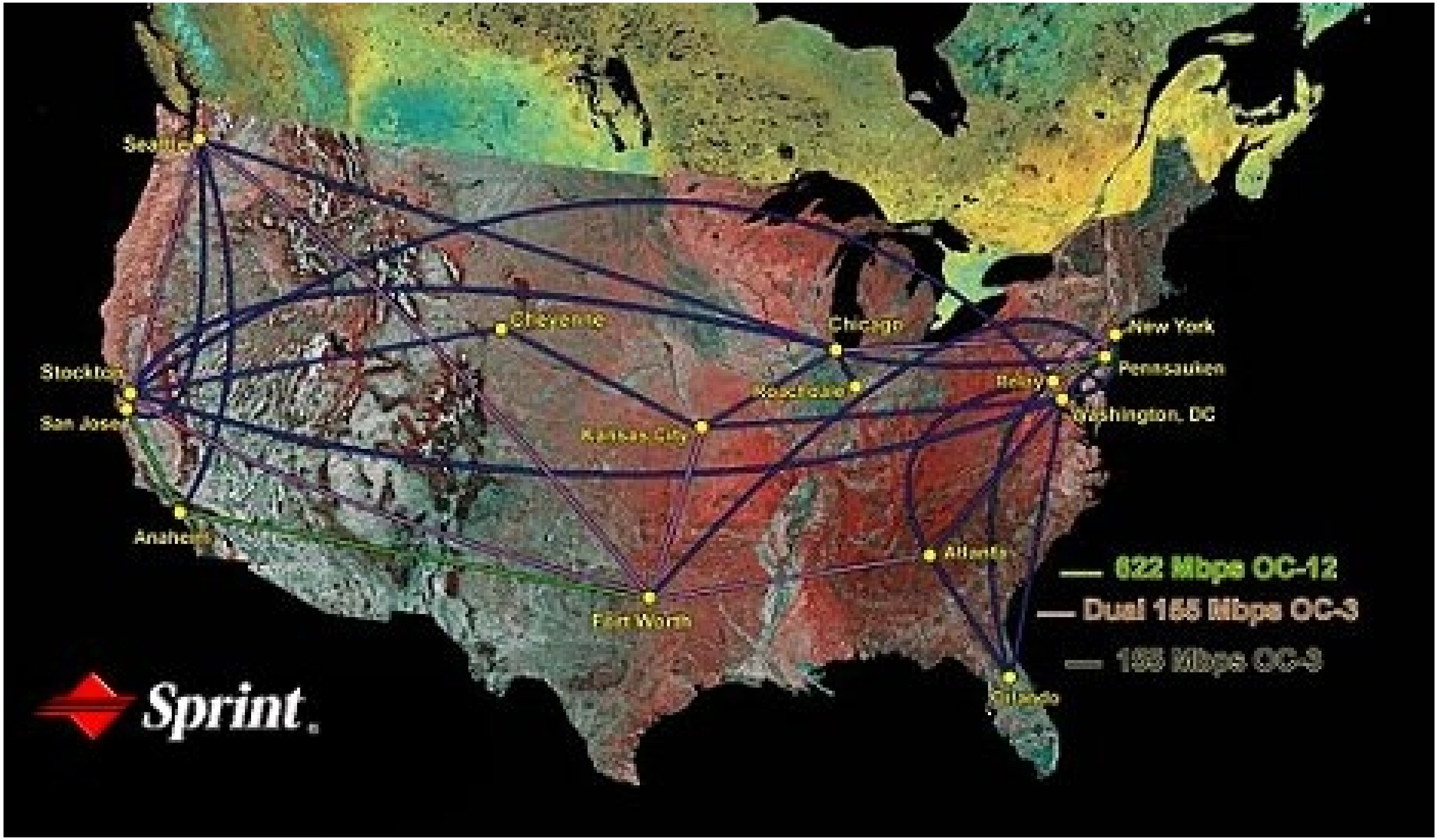
Tier-1 providers interconnect (peer) privately




Tier-1 providers also interconnect at public network access points (NAPs)

# Tier-1 ISP: e.g., Sprint

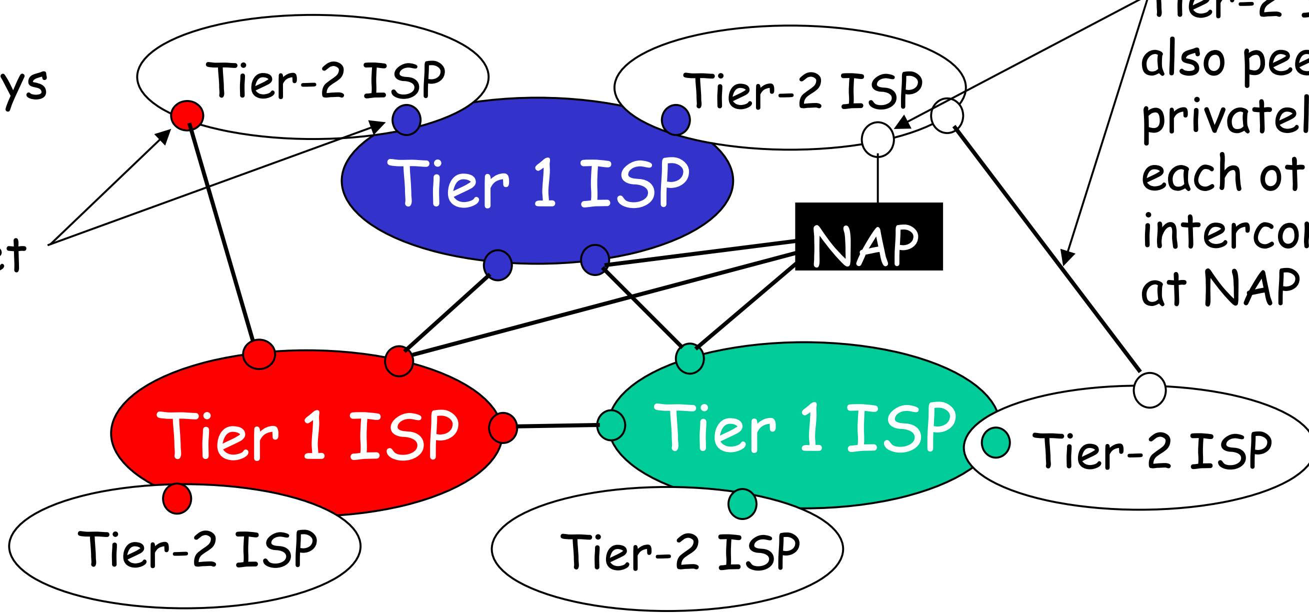
Sprint US backbone network



# 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

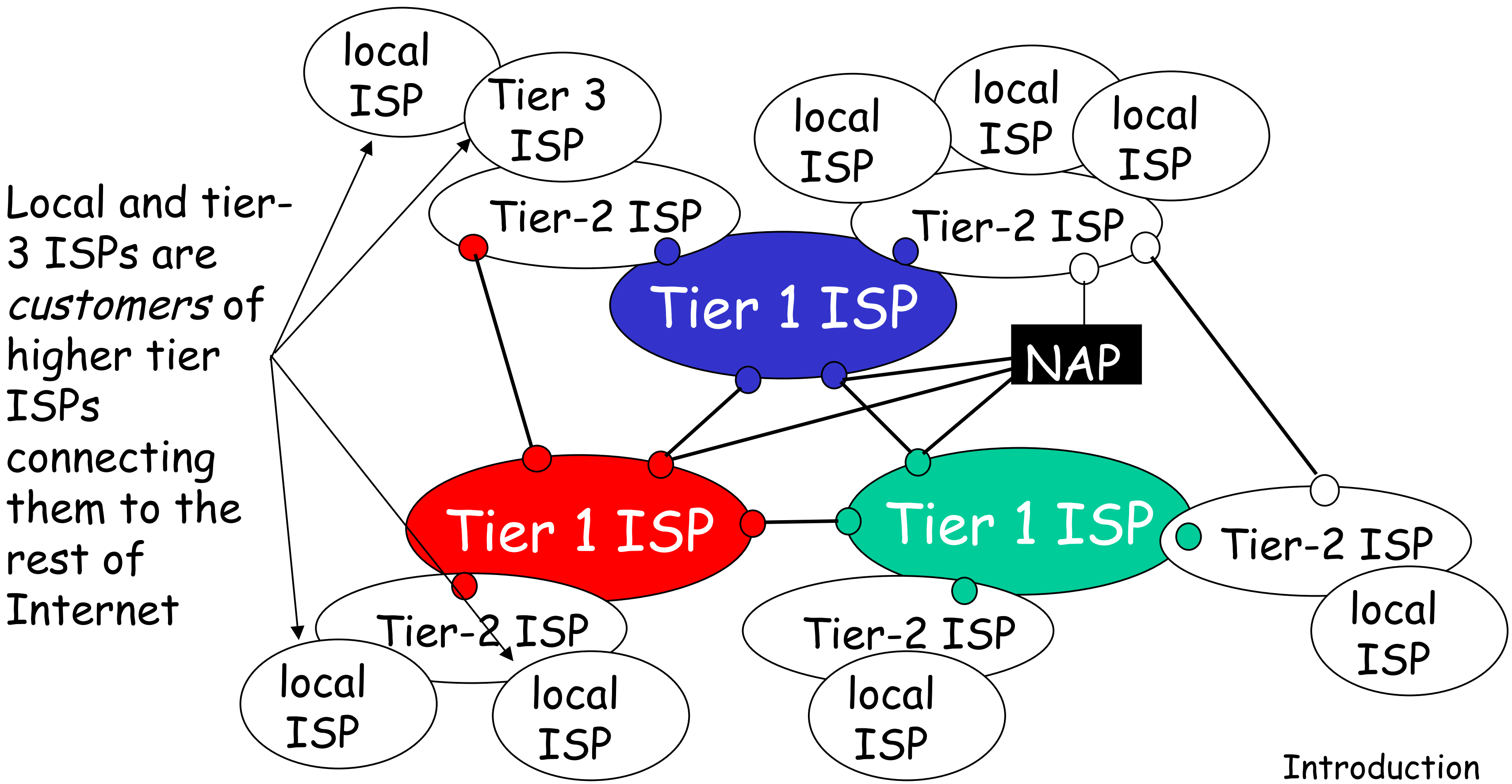
- Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet
- tier-2 ISP is *customer* of tier-1 *provider*




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

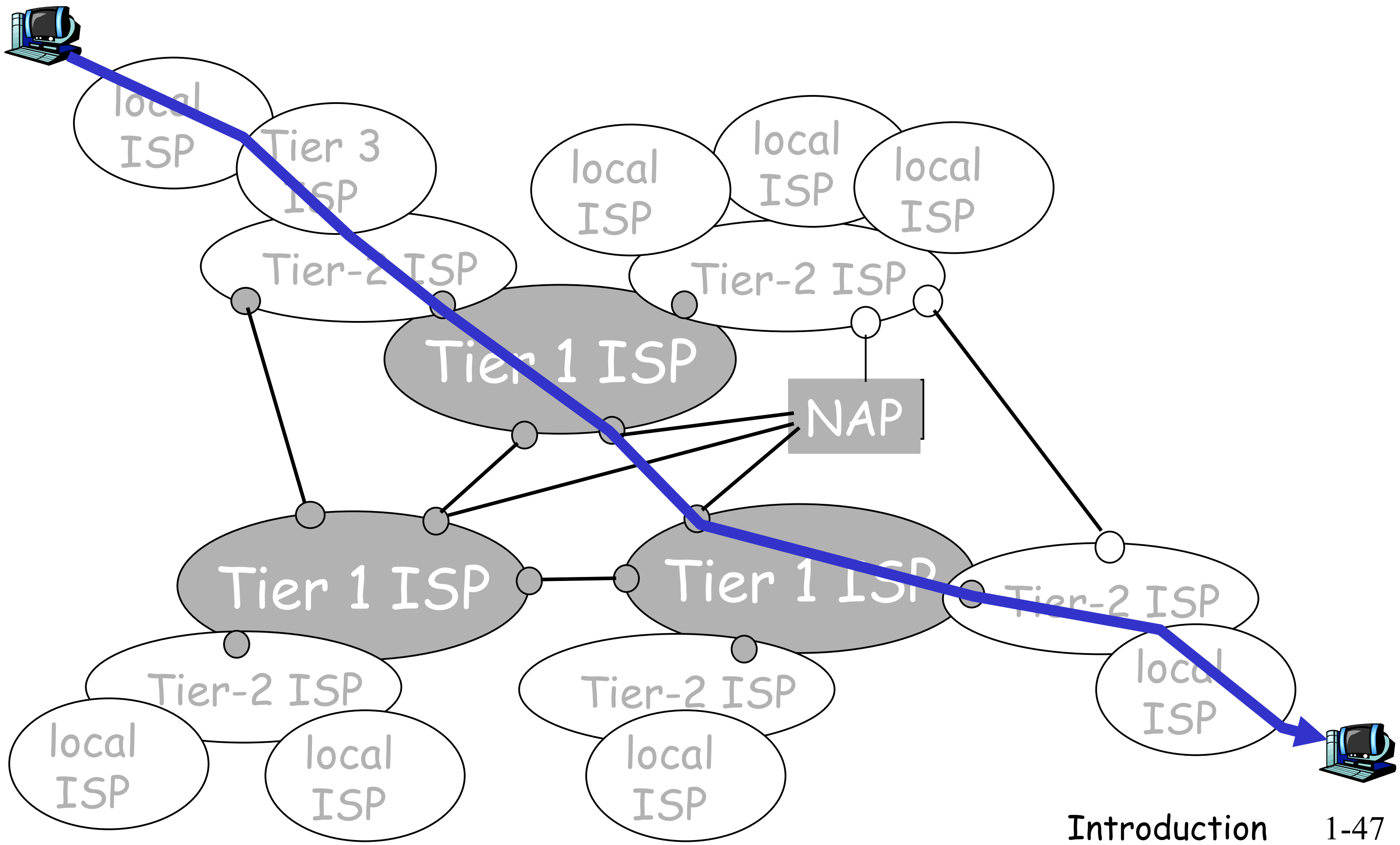
# Internet structure: network of networks

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



# Internet structure: network of networks

- a packet passes through many networks!



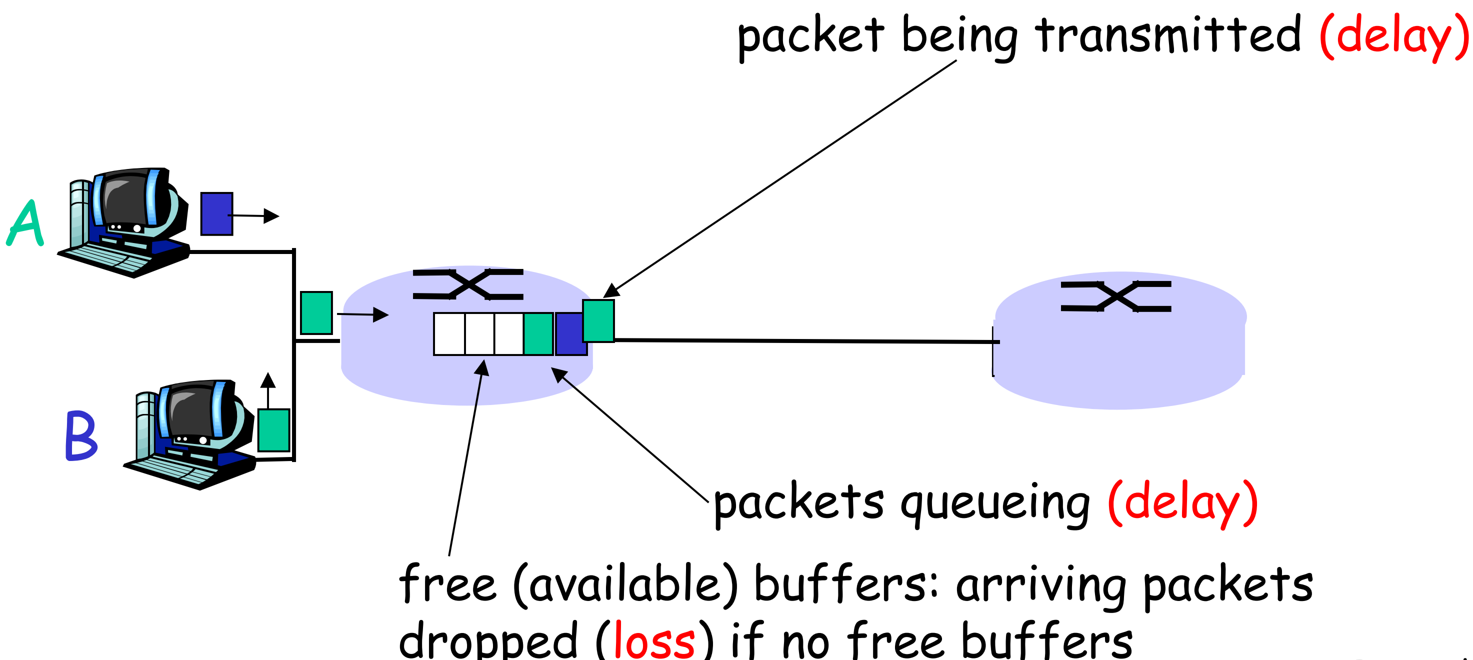
# Chapter 1: roadmap

1.1 What *is* the Internet?
1.2 Network edge
1.3 Network core
1.4 Network access and physical media
1.5 Internet structure and ISPs
1.6 Delay & loss in packet-switched networks
1.7 Protocol layers, service models
1.8 History

# How do loss and delay occur?

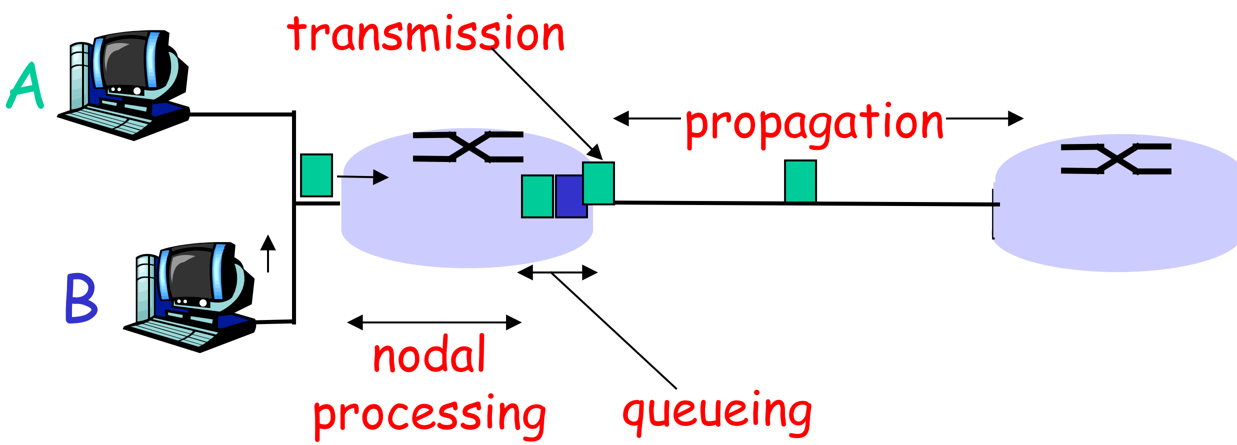
packets *queue* in router buffers

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



# Four sources of packet delay

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



# Delay in packet-switched networks

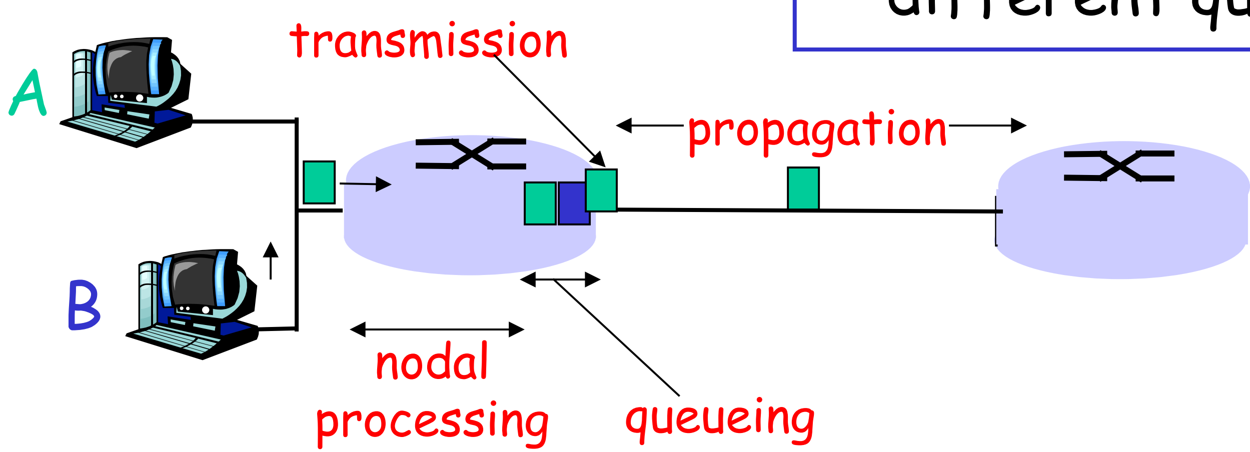
3. Transmission delay:

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

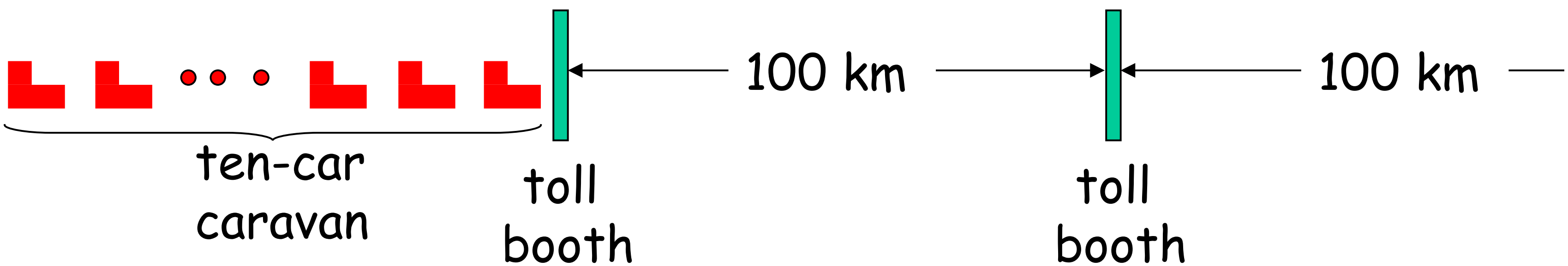
4. Propagation delay:

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

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

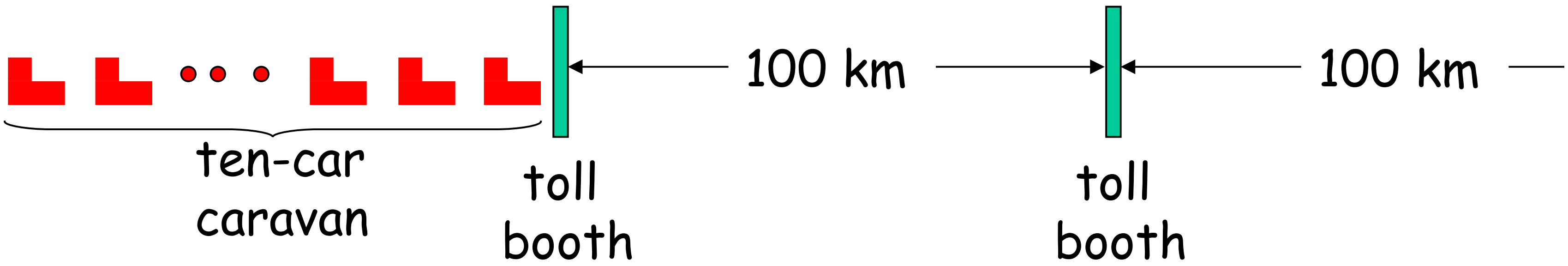


# Caravan analogy




- Cars "propagate" at 100 km/hr
- Toll booth takes 12 sec to service a car (transmission time)
- car~bit; caravan ~ packet
- Q: How long until caravan is lined up before 2nd toll booth?
- Time to "push" entire caravan through toll booth onto highway = 12*10 = 120 sec
- Time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr)= 1 hr
- A: 62 minutes

# Caravan analogy (more)




- Cars now "propagate" at 1000 km/hr
- Toll booth now takes 1 min to service a car
- Q: Will cars arrive to 2nd booth before all cars serviced at 1st booth?

- Yes! After 7 min, 1st car at 2nd booth and 3 cars still at 1st booth.
- 1st bit of packet can arrive at 2nd router before packet is fully transmitted at 1st router!
  - See Ethernet applet at AWL Web site

# Nodal delay

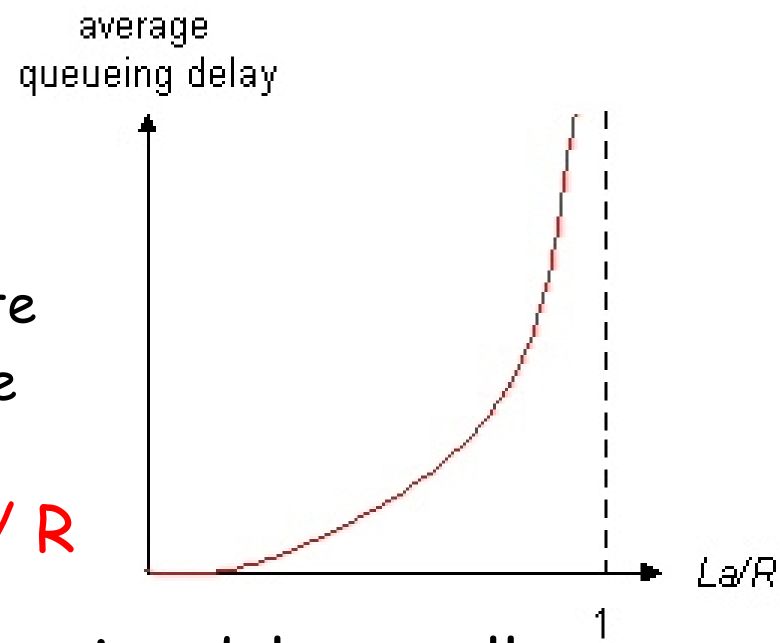
$$\boxed{d_{\text{nodal}} = d_{\text{proc}} + d_{\text{queue}} + d_{\text{trans}} + d_{\text{prop}}}$$

- $d_{proc}$ = 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
- L · a = average bit arrival rate

traffic intensity = (L · a) / R



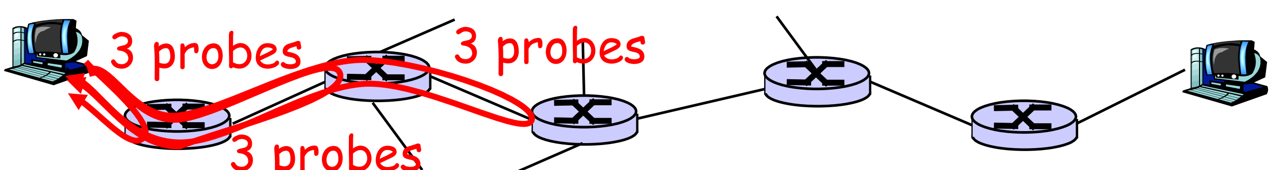

- (L · a)/R ~ 0: average queueing delay small
- (L · a)/R -> 1: delays become large
- (L · a)/R > 1: more "work" arriving than can be serviced, average delay infinite! (infinite queue length) -- or packet loss! (finite queue length)

# "Real" Internet delays and routes

- What do "real" Internet delay & loss look like?
- Traceroute program: provides delay measurement from source to router along end-end Internet path towards destination.

For *i = 1,2,3,…*

- sends three packets that will reach router *i* on path towards destination
- router *i* will return packets to sender
- sender times interval between transmission and reply.



# “Real” Internet delays and routes

traceroute: gaia.cs.umass.edu to www.eurecom.fr

Three delay measements from gaia.cs.umass.edu to cs-gw.cs.umass.edu

1 cs-gw (128.119.240.254) 1 ms 1 ms 2 ms
2 border1-rt-fa5-1-0.gw.umass.edu (128.119.3.145) 1 ms 1 ms 2 ms
3 cht-vbns.gw.umass.edu (128.119.3.130) 6 ms 5 ms 5 ms
4 jn1-at1-0-0-19.wor.vbns.net (204.147.132.129) 16 ms 11 ms 13 ms
5 jn1-so7-0-0-0.wae.vbns.net (204.147.136.136) 21 ms 18 ms 18 ms
6 abilene-vbns.abilene.ucaid.edu (198.32.11.9) 22 ms 18 ms 22 ms
7 nycm-wash.abilene.ucaid.edu (198.32.8.46) 22 ms 22 ms 22 ms
8 62.40.103.253 (62.40.103.253) 104 ms 109 ms 106 ms
9 de2-1.de1.de.geant.net (62.40.96.129) 109 ms 102 ms 104 ms
10 de.fr1.fr.geant.net (62.40.96.50) 113 ms 121 ms 114 ms
11 renater-gw.fr1.fr.geant.net (62.40.103.54) 112 ms 114 ms 112 ms
12 nio-n2.cssi.renater.fr (193.51.206.13) 111 ms 114 ms 116 ms
13 nice.cssi.renater.fr (195.220.98.102) 123 ms 125 ms 124 ms
14 r3t2-nice.cssi.renater.fr (195.220.98.110) 126 ms 126 ms 124 ms
15 eurecom-valbonne.r3t2.ft.net (193.48.50.54) 135 ms 128 ms 133 ms
16 194.214.211.25 (194.214.211.25) 126 ms 128 ms 126 ms
17 * * *
18 * * *
19 fantasia.eurecom.fr (193.55.113.142) 132 ms 128 ms 136 ms

trans-oceanic link

* means no reponse (probe lost, router not replying)

# Packet loss

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

# Chapter 1: roadmap

# 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?

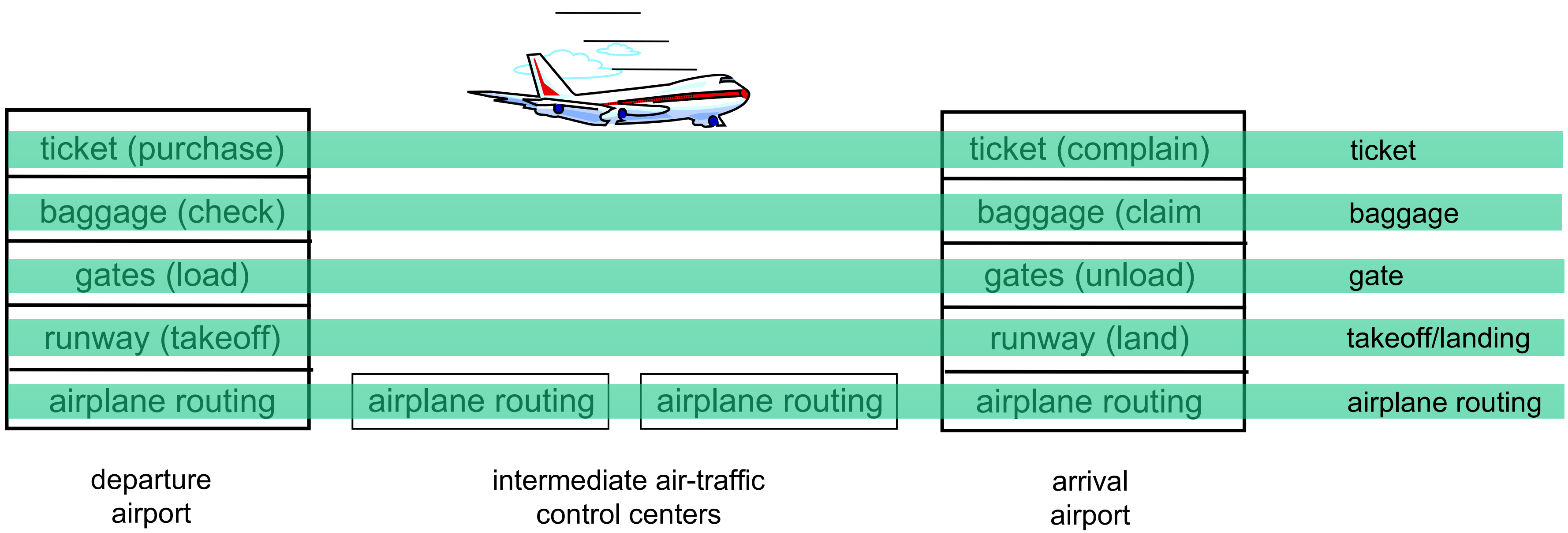
# Organization of air travel

| | |
|---|---|
| ticket (purchase) | ticket (complain) |
| baggage (check) | baggage (claim) |
| gates (load) | gates (unload) |
| runway takeoff | runway landing |
| airplane routing | airplane routing |

airplane routing

- a series of steps

# Layering of airline functionality




Layers: each layer implements a service

- via its own internal-layer actions
- relying on services provided by layer below

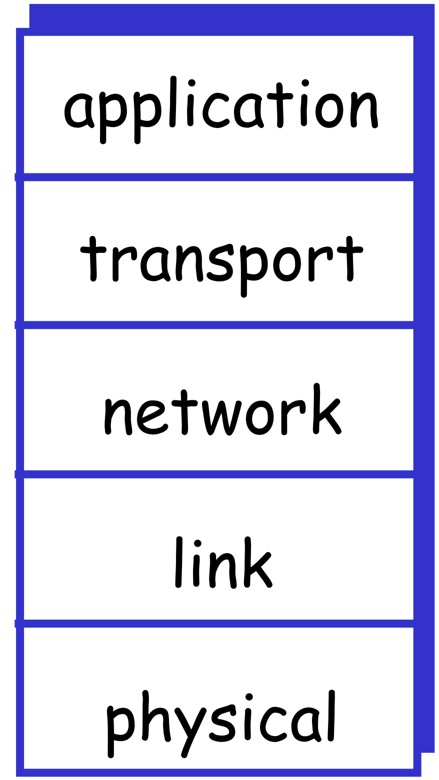
# Why layering?

Dealing with complex systems:

- explicit structure allows identification, relationship of complex system's pieces
  - layered 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 in gate procedure doesn't affect rest of system

# Internet protocol stack

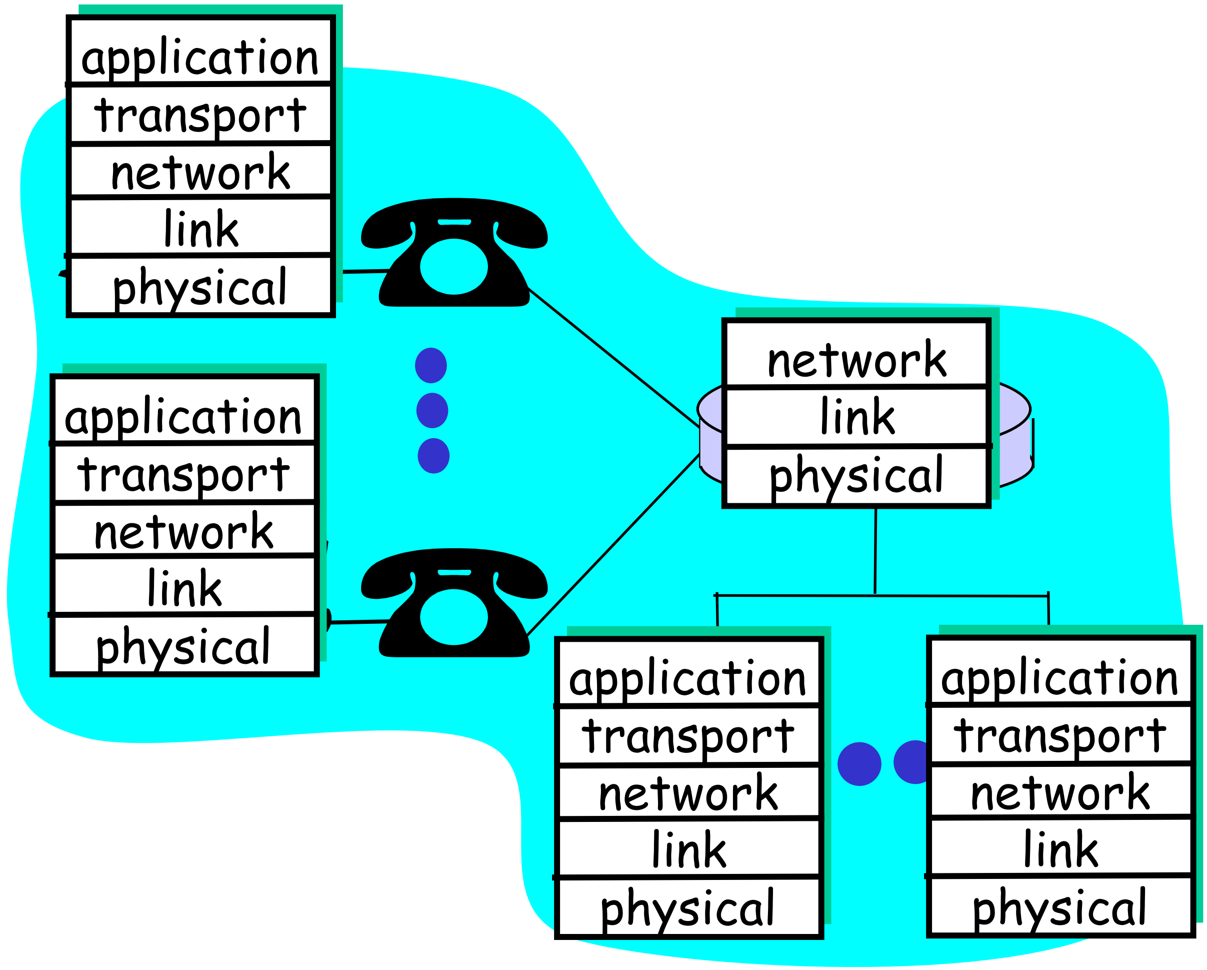
- application: supporting network applications
  - FTP, SMTP, HTTP
- transport: host-host data transfer
  - TCP, UDP
- network: routing of datagrams from source to destination
  - IP, routing protocols
- link: data transfer between neighboring network elements
  - PPP, Ethernet
- physical: bits "on the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



# Layering: logical communication

Each layer:

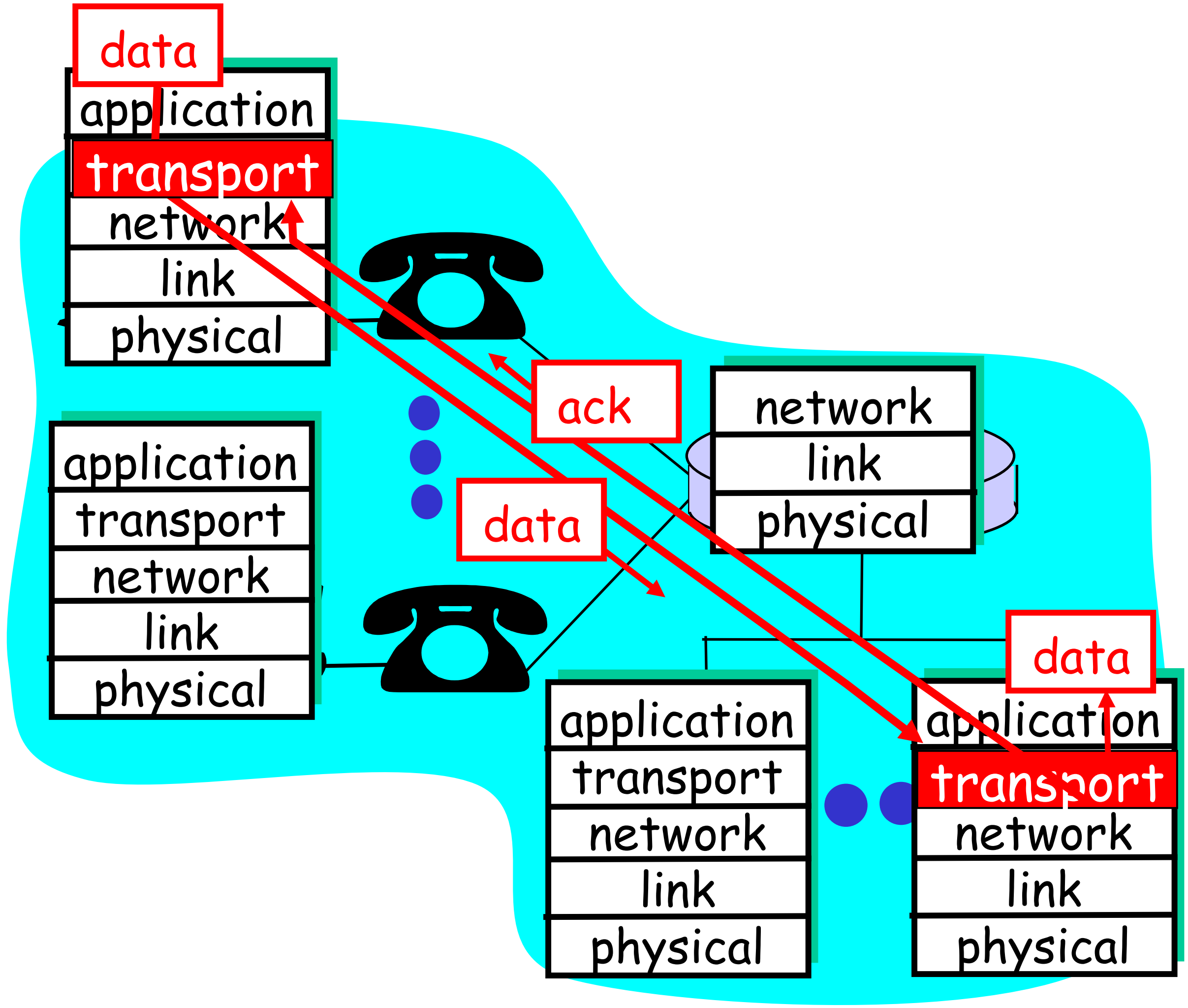
- distributed "entities" implement layer functions at each node
- entities perform actions, exchange messages with peers



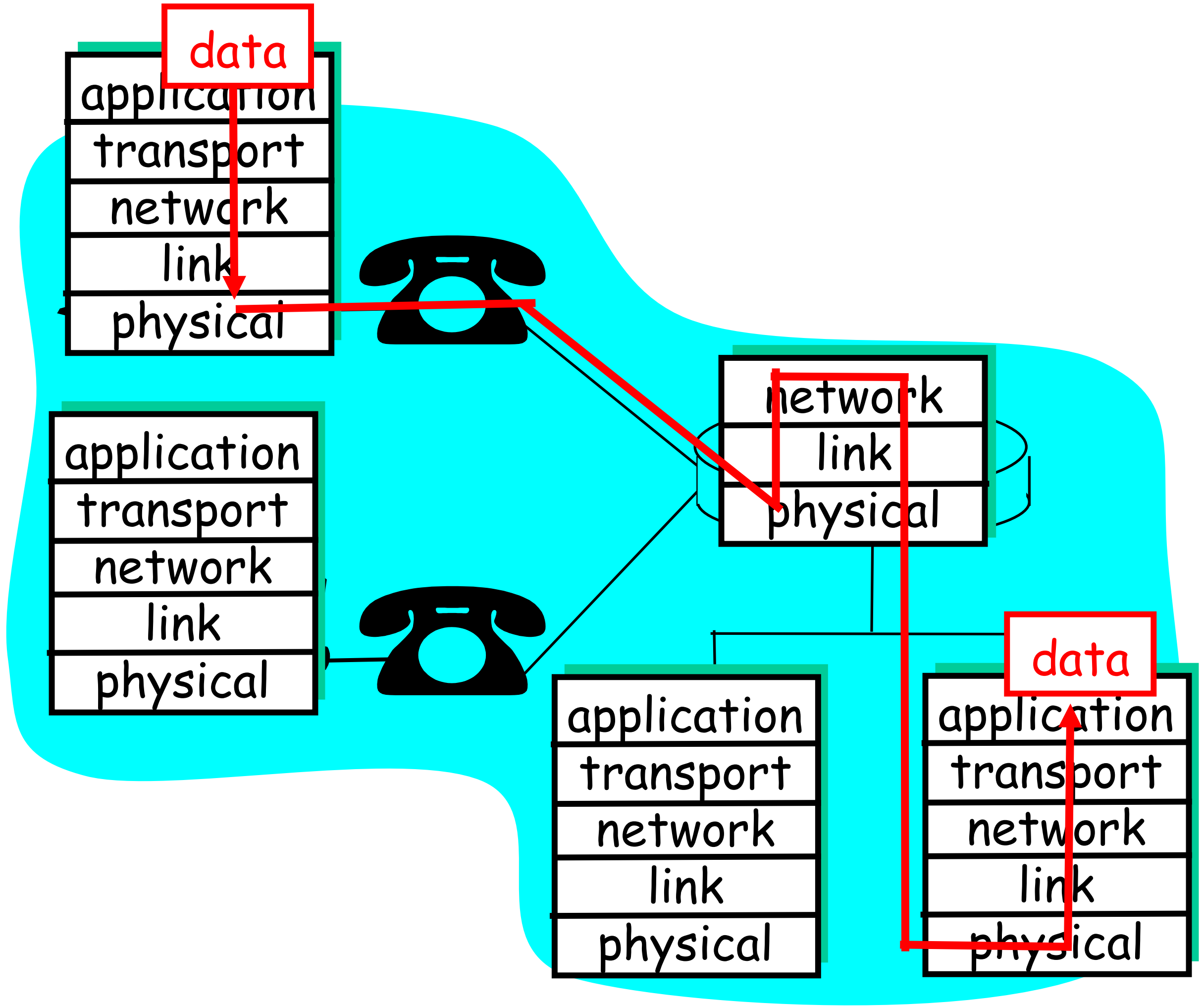
# Layering: *logical* communication

E.g.: transport

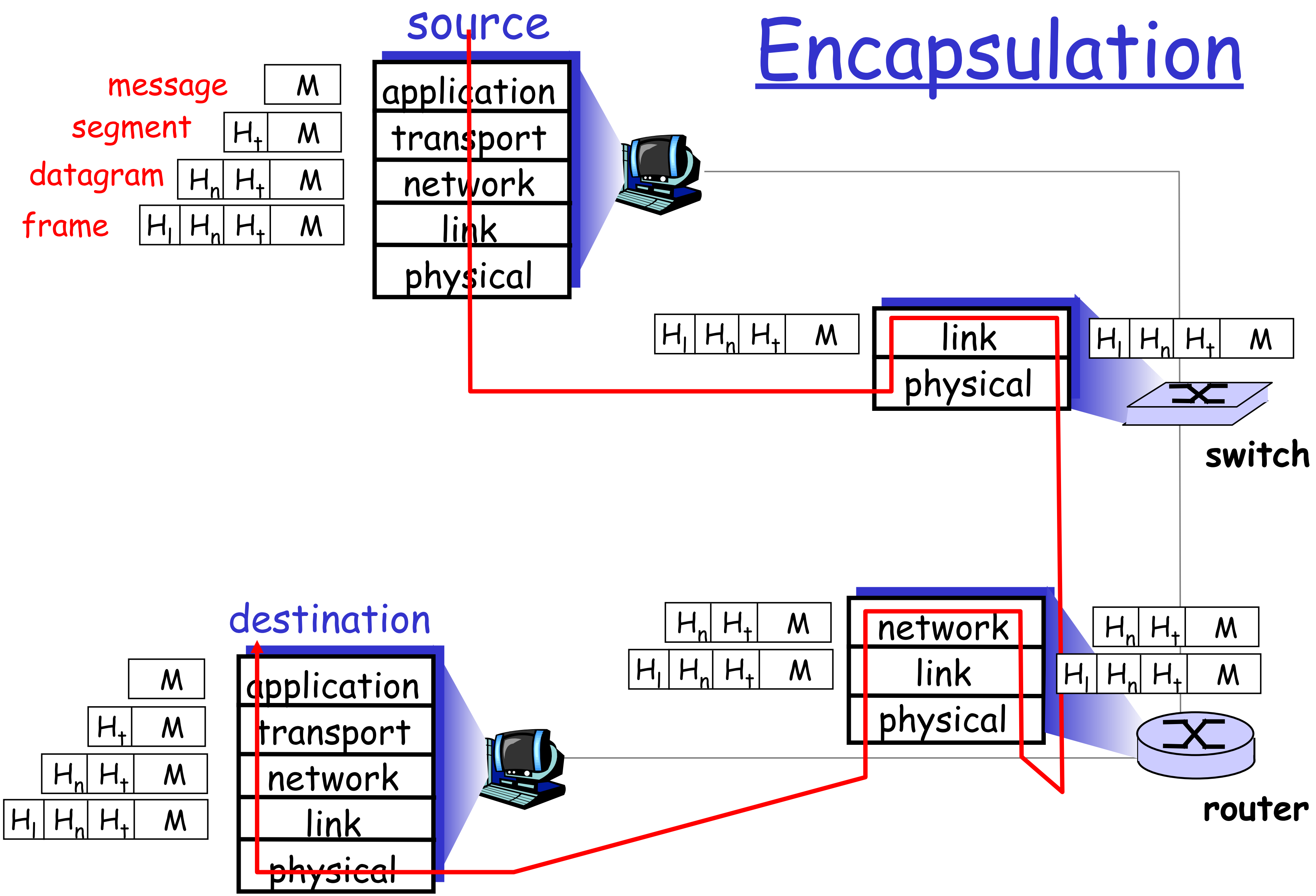
- take data from app
- add addressing, reliability check info to form "datagram"
- send datagram to peer
- wait for peer to ack receipt
- analogy: post office



# Layering: physical communication



# Encapsulation



# Chapter 1: roadmap

# 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

# Introduction: Summary

Covered a "ton" of material!

- Internet overview
- what's a protocol?
- network edge, core, access network
  - packet-switching versus circuit-switching
- Internet/ISP structure
- performance: loss, delay
- layering and service models
- history

You now have:

- context, overview, "feel" of networking
- more depth, detail *to follow!*