COMPUTER NETWORKS AND THE INTERNET

MODULE - I

Module 1: Computer Networks & the Internet

our goal:

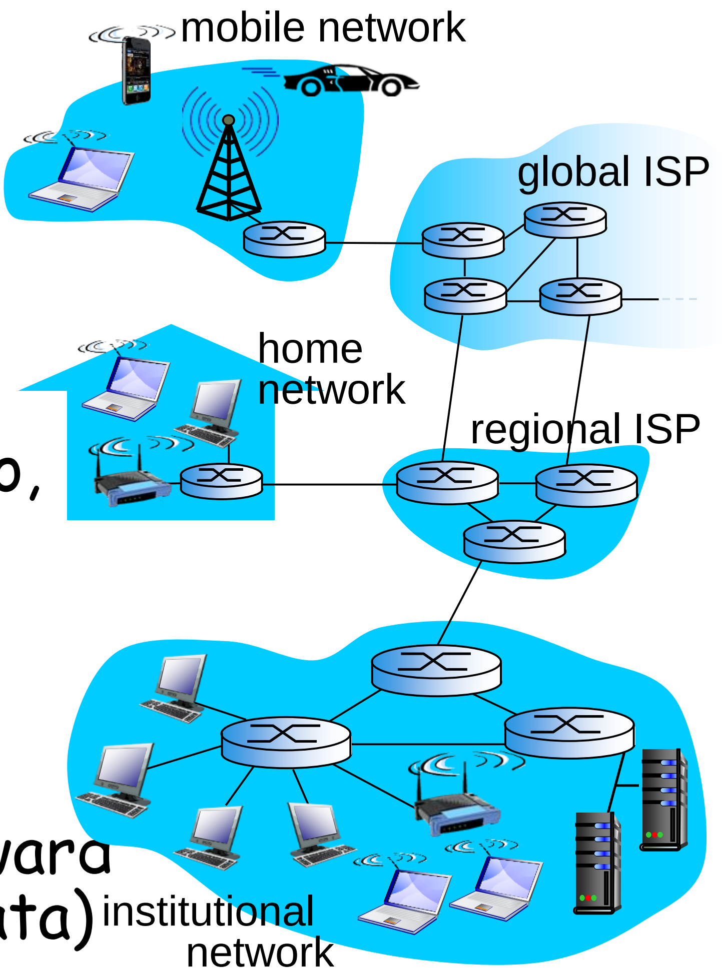
- get "feel" and terminology
- approach:
  - use Internet as example

overview:

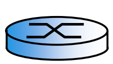
- what's the Internet?
- what's a protocol?
- Network edge; Hosts, Access net, Physical media
- Network core: Packet/Circuit switching, Internet structure
- Performance: Loss, Delay, Throughput
- Protocol layers, Service Models
- History

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

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



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



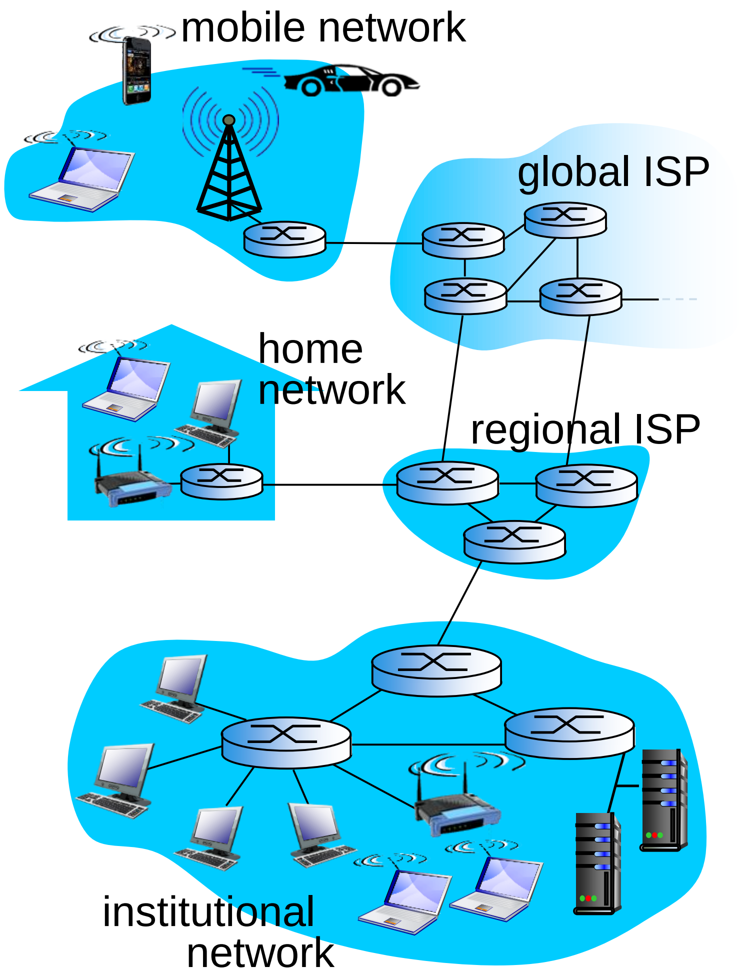
❖ *Packet switches:* forward packets (chunks of data)
  - *routers* and *switches*

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

Internet: "network of networks"
- Interconnected ISPs

protocols control sending, receiving of msgs
- e.g., TCP, IP, HTTP, Skype, 802.11

Internet standards
- RFC: Request for comments
- IETF: Internet Engineering Task Force
- http://www.ietf.org/rfc.html



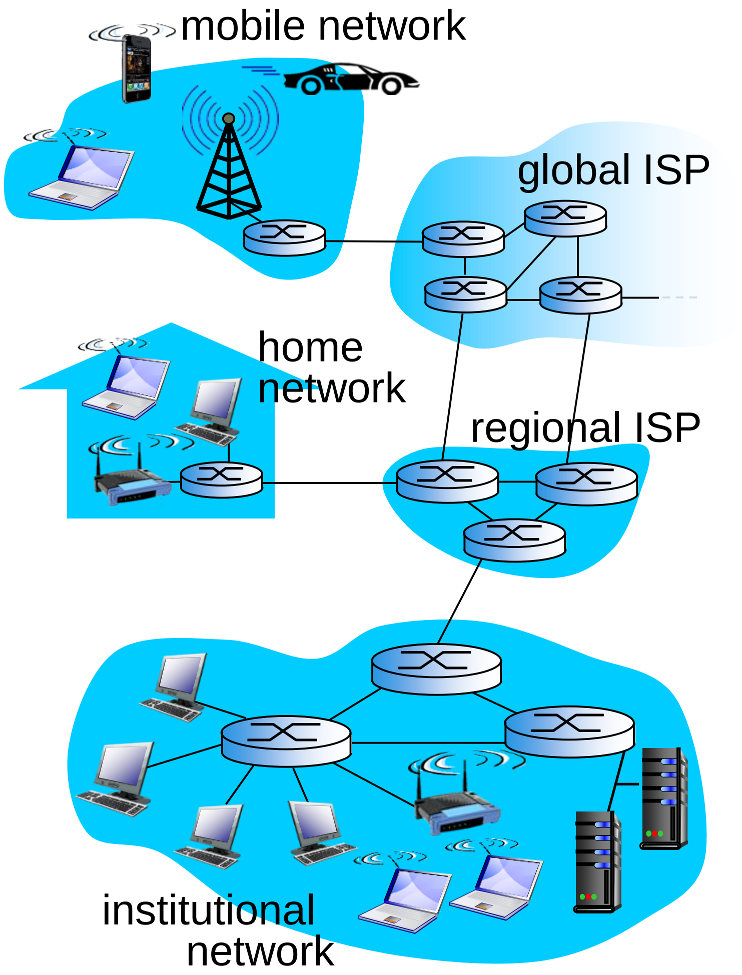
What’s the Internet: a service view

Infrastructure that provides services to applications:

- Web, VoIP, email, games, e-commerce, social nets, …

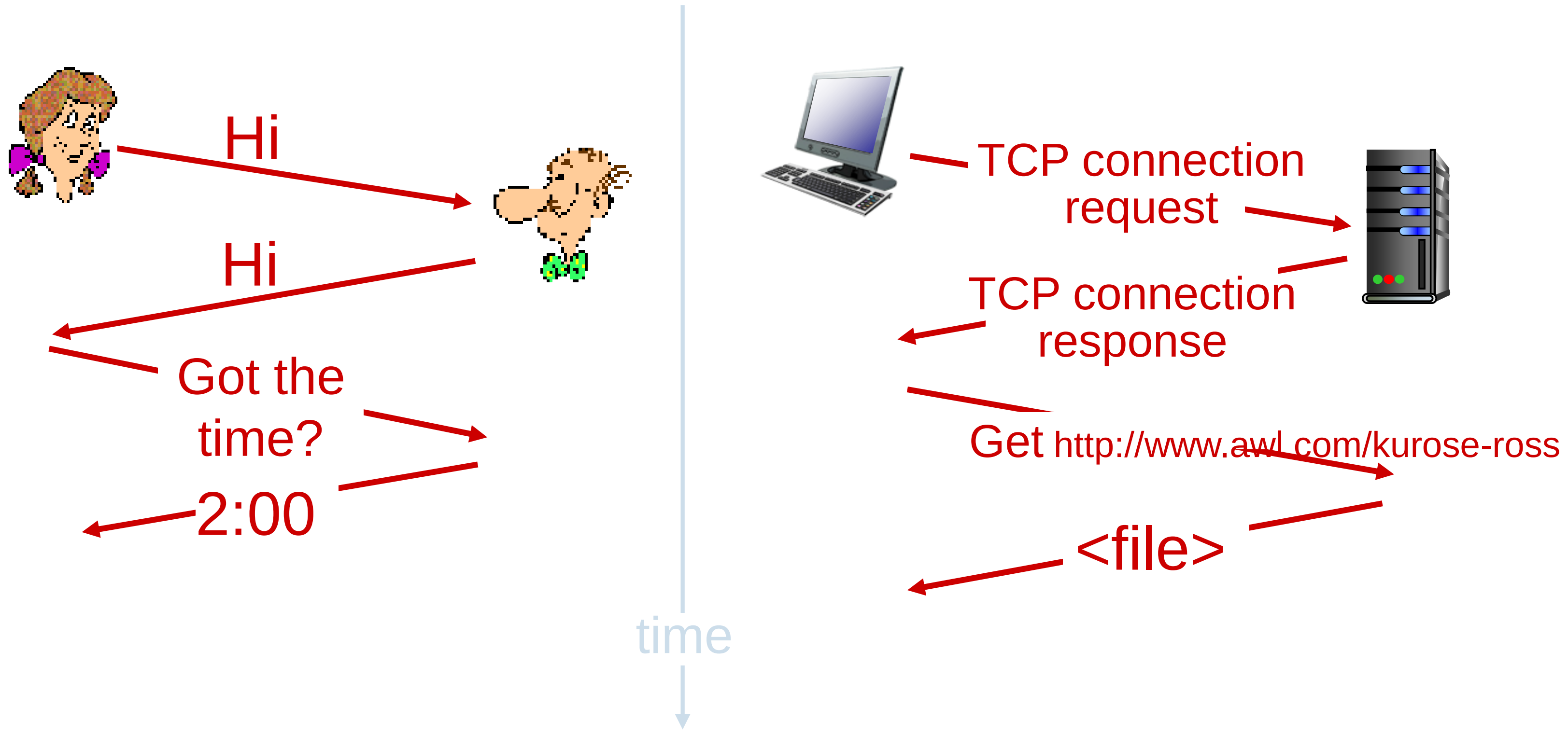
provides programming interface to apps

- hooks that allow sending and receiving app programs to “connect” to Internet
- provides service options, analogous to postal service



What's a protocol?

a human protocol and a computer network protocol:



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

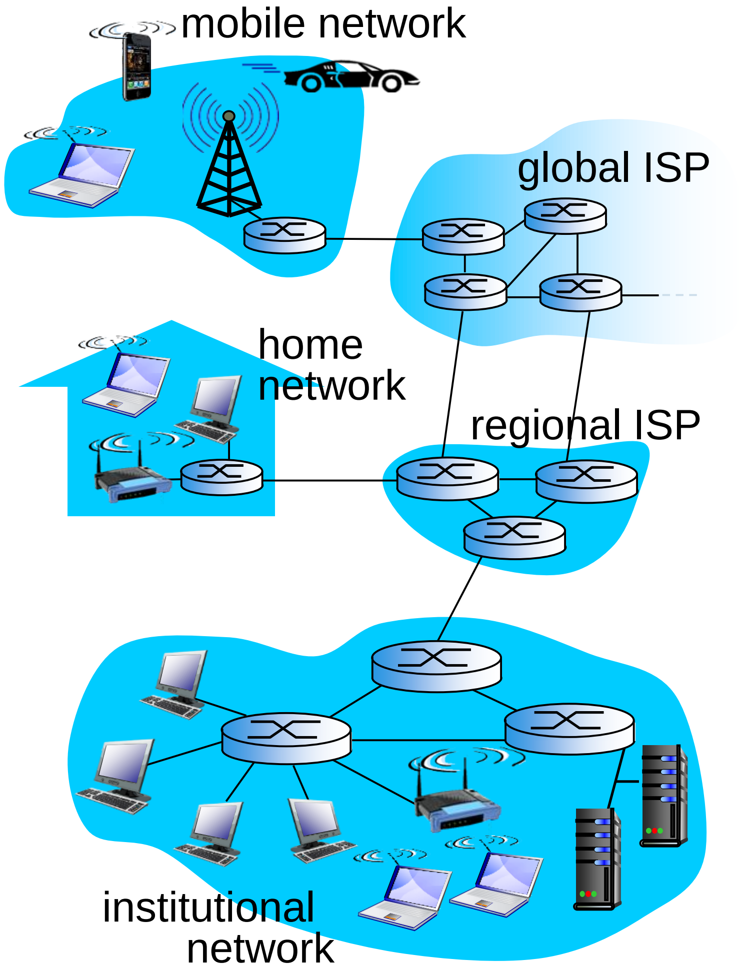
protocols define *format*, *order* of *msgs sent and received* among network entities, and *actions taken* on msg transmission, receipt

A closer look at network structure:

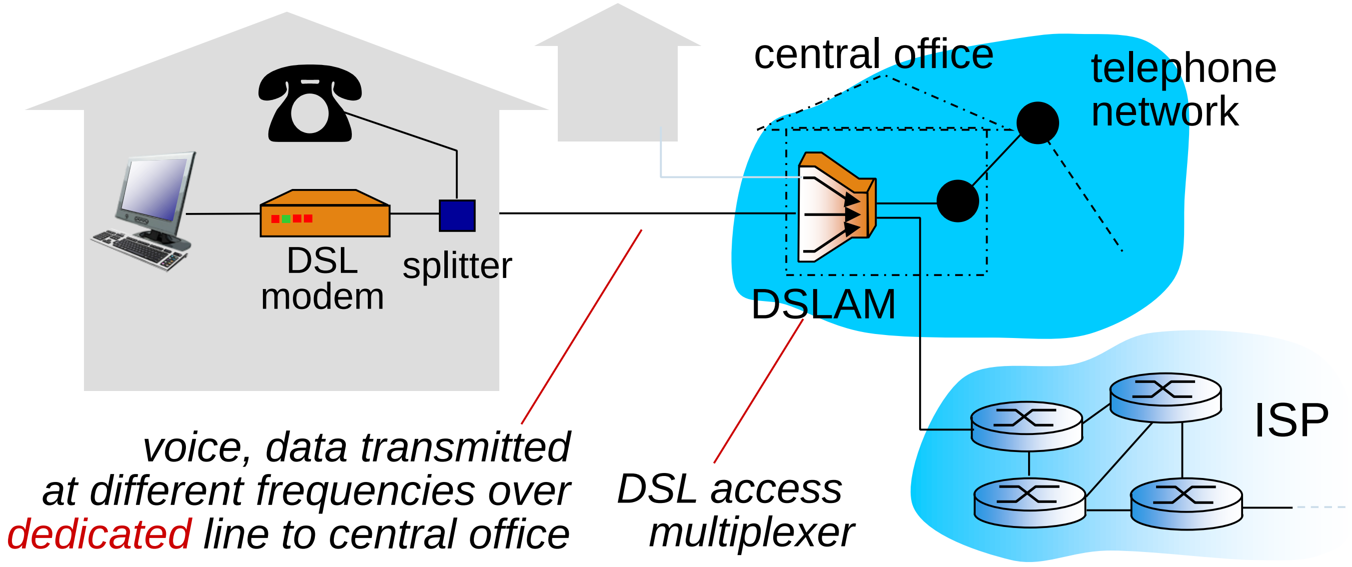
network edge:
- hosts: clients and servers
- servers often in data centers

- *access networks, physical media:* wired, wireless communication links

- *network core:*
  - interconnected routers
  - network of networks

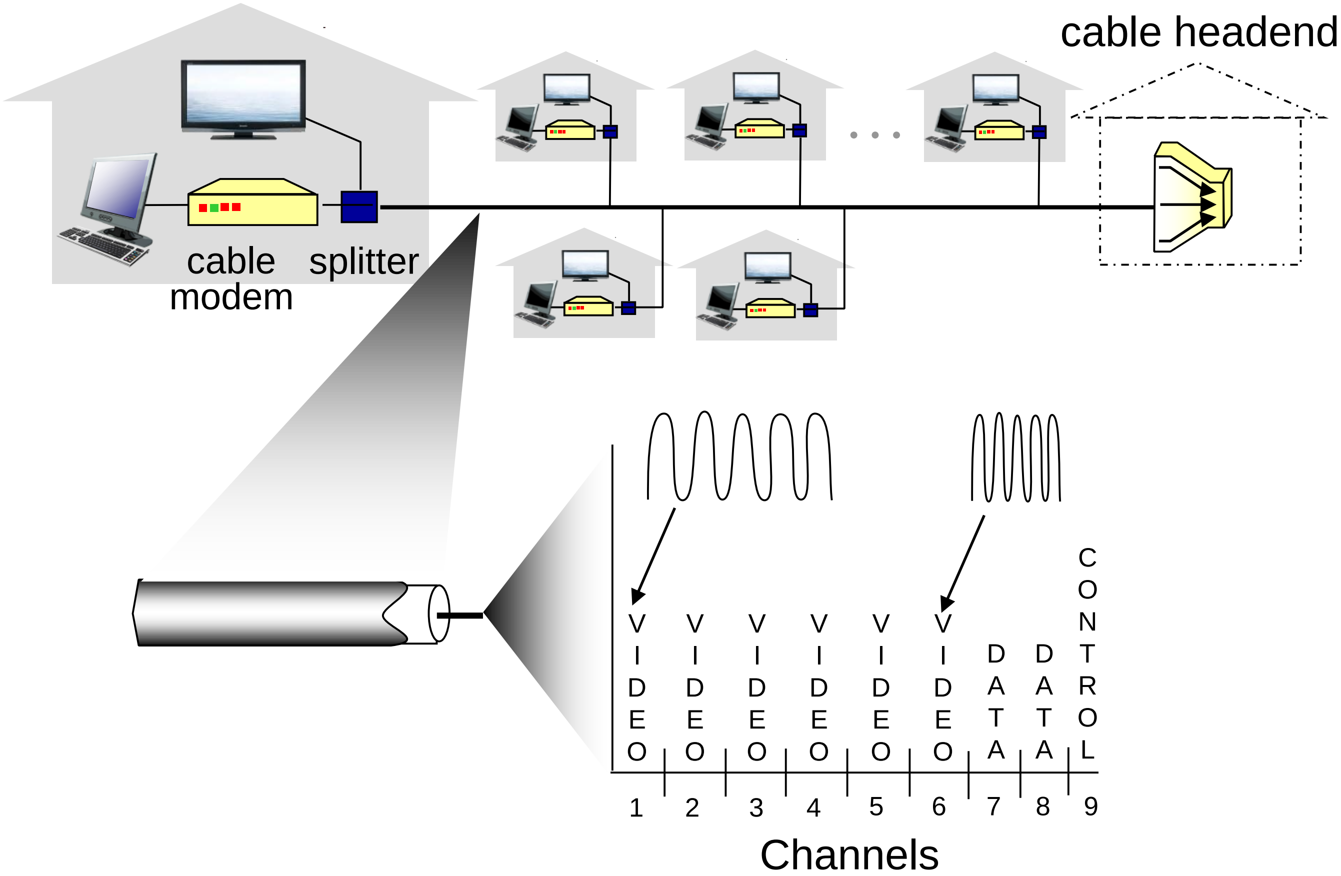


Access net: digital subscriber line (DSL)



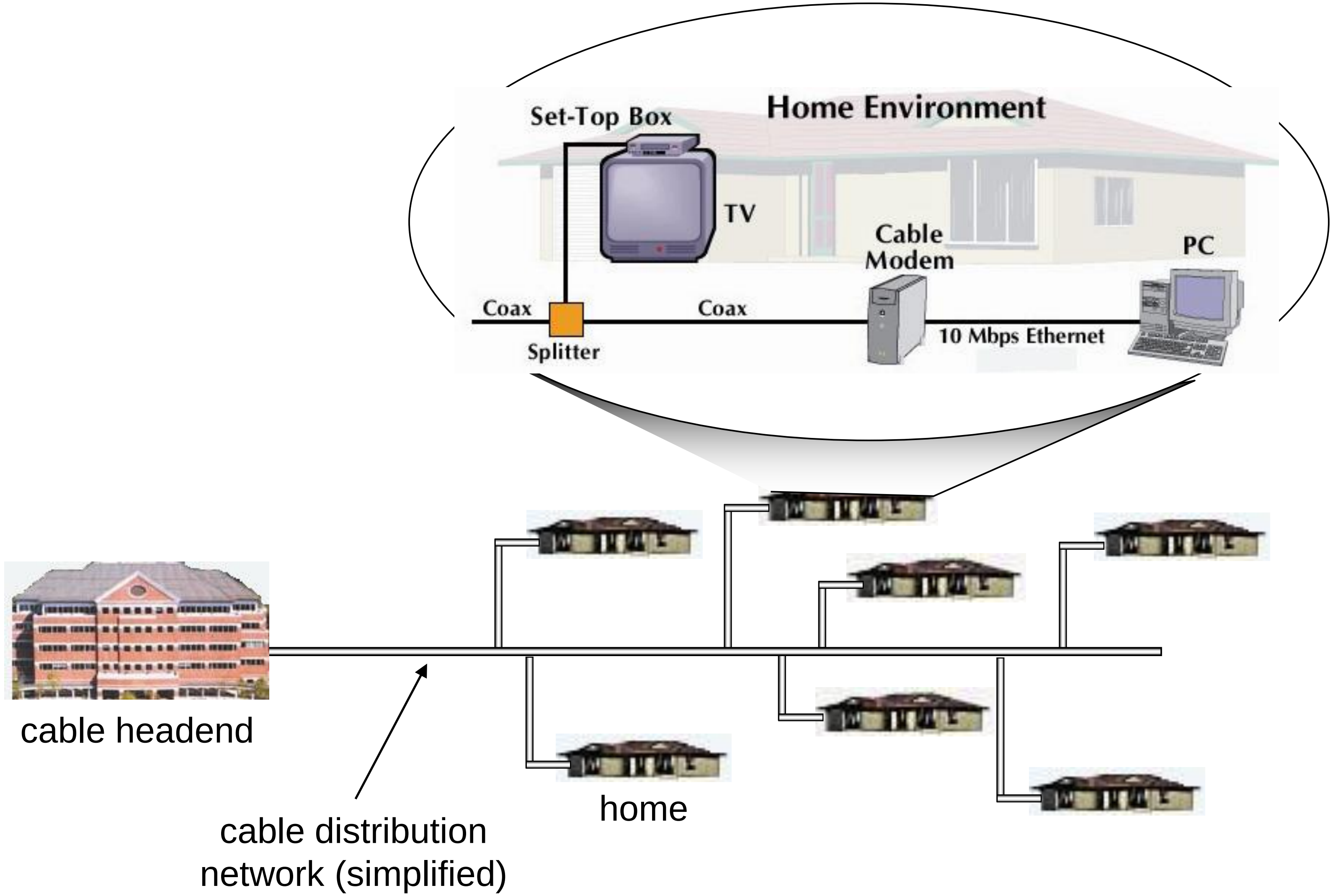
- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net

Access net: cable network

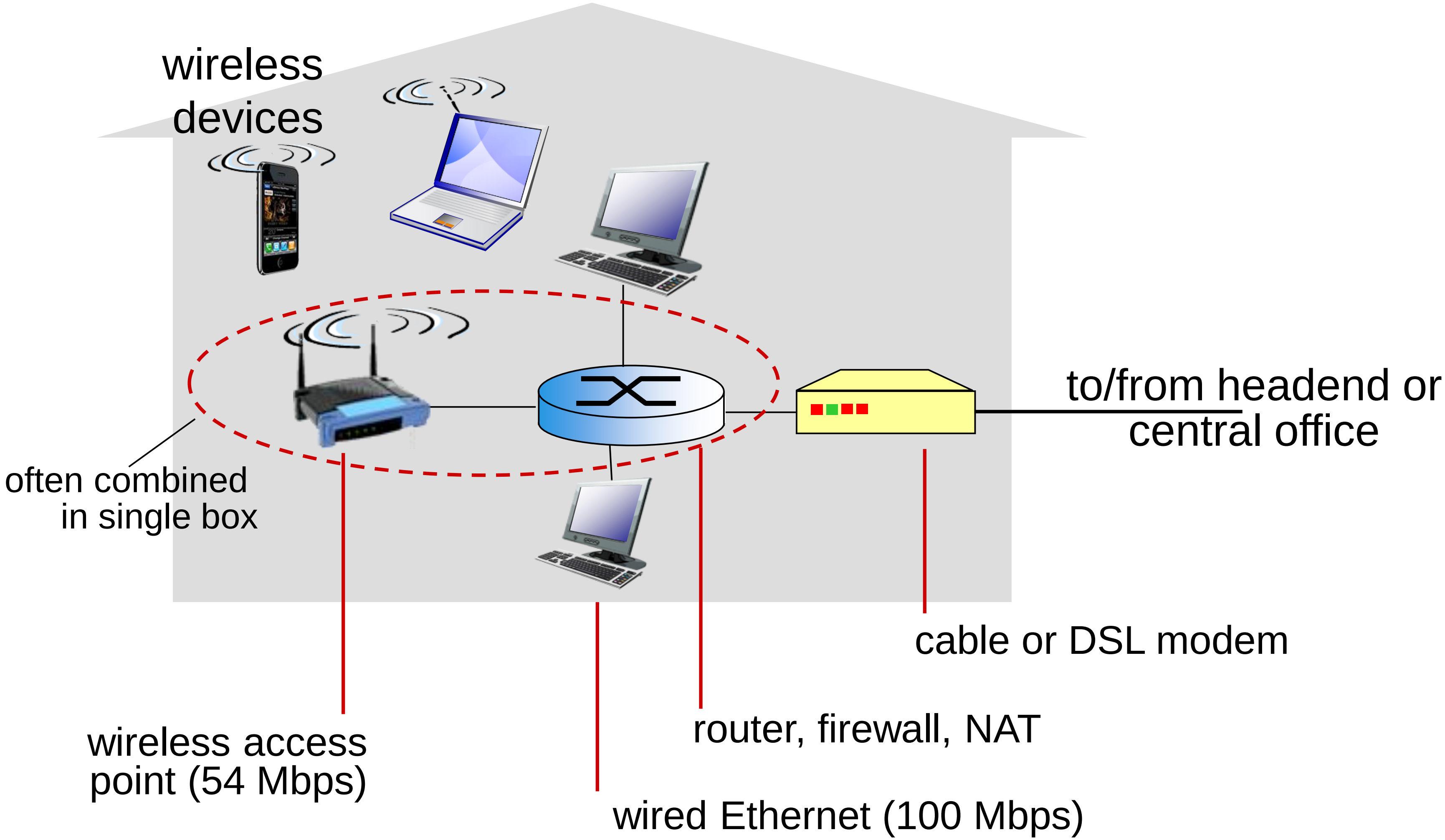


frequency division multiplexing: different channels transmitted in different frequency bands

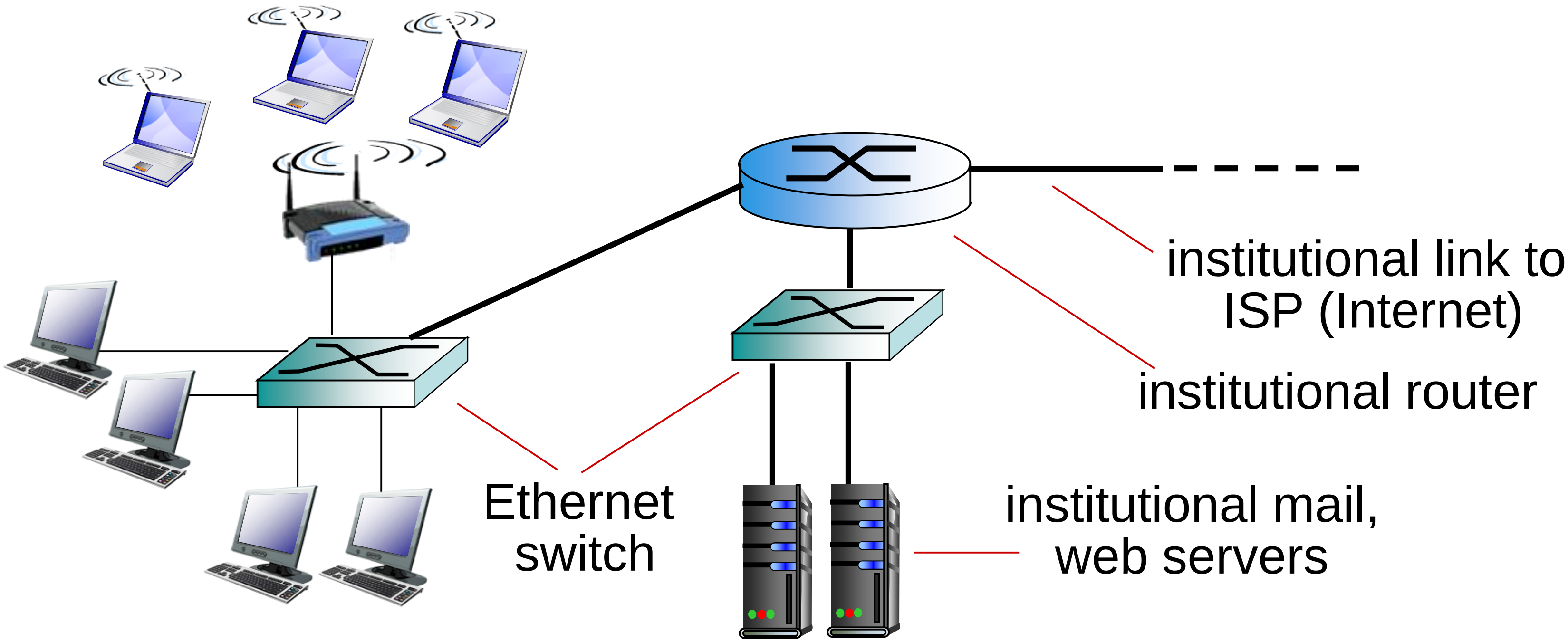
Cable Network Architecture: Overview



Access net: home network



Enterprise access networks (Ethernet)

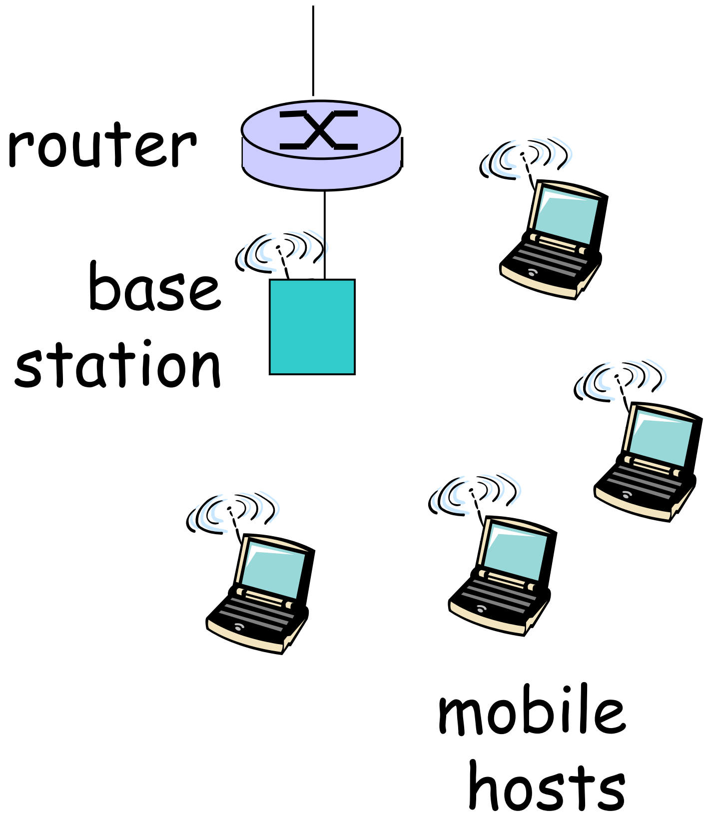


typically used in companies, universities, etc

- 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:
  - 802.11b (WiFi): 11 Mbps
- wider-area wireless access
  - provided by telco operator
  - 3G ~ 384 kbps
  - WAP/GPRS in Europe



Physical Media

- Bit: propagates between transmitter/rcvr 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

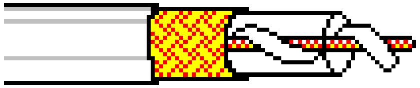
Twisted Pair (TP)

- two insulated copper wires
  - Category 3: traditional phone wires, 10 Mbps Ethernet
  - Category 5: 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 point-to-point transmission (e.g., 5 Gps)
- low error rate: repeaters spaced far apart ; immune to electromagnetic noise

Physical media: radio

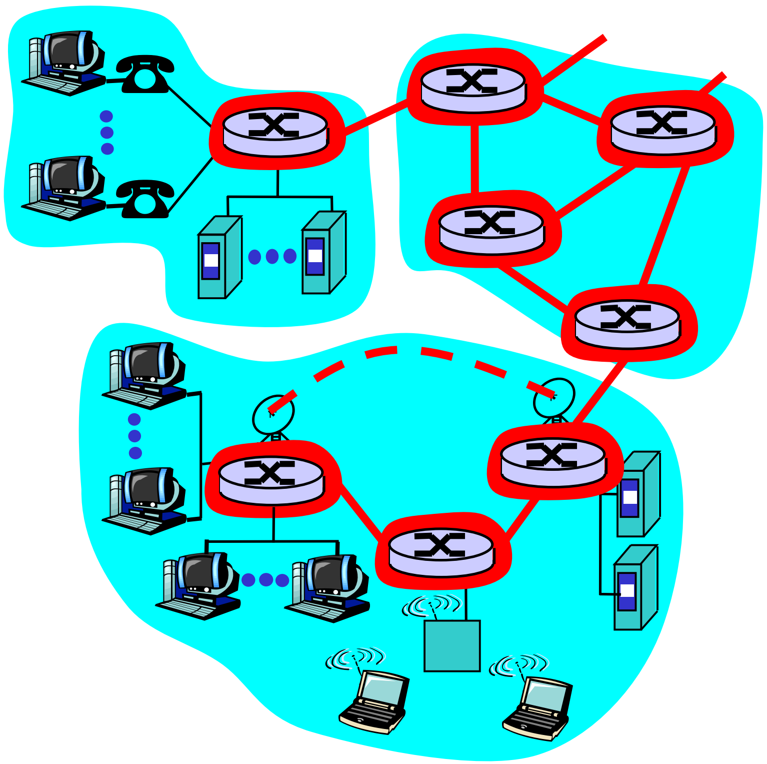
- signal carried in electromagnetic spectrum
- no physical "wire"
- bidirectional
- propagation environment effects:
  - reflection
  - obstruction by objects
  - interference

Radio link types:

- terrestrial microwave
  - e.g. up to 45 Mbps channels
- LAN (e.g., Wifi)
  - 2Mbps, 11Mbps
- 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 altitude

The Network Core

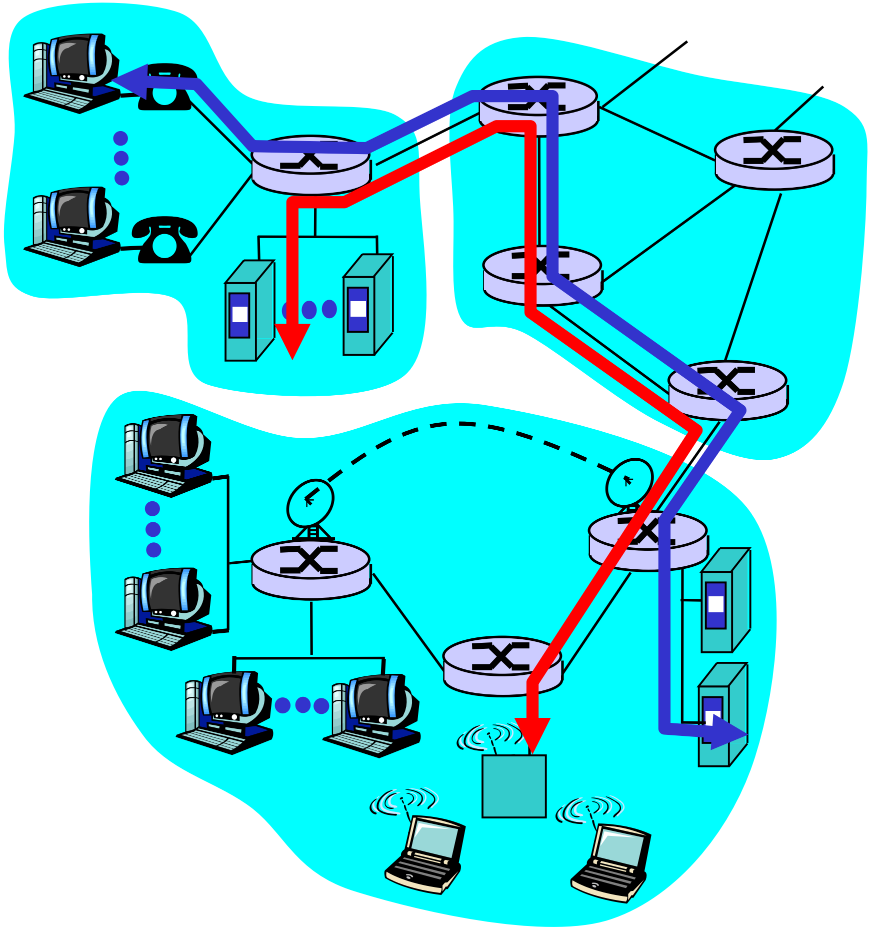
- mesh of interconnected routers
- *the* fundamental question: how is data transferred through net?
  - circuit switching: dedicated circuit per call: telephone net
  - packet-switching: data sent thru net in discrete "chunks"



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



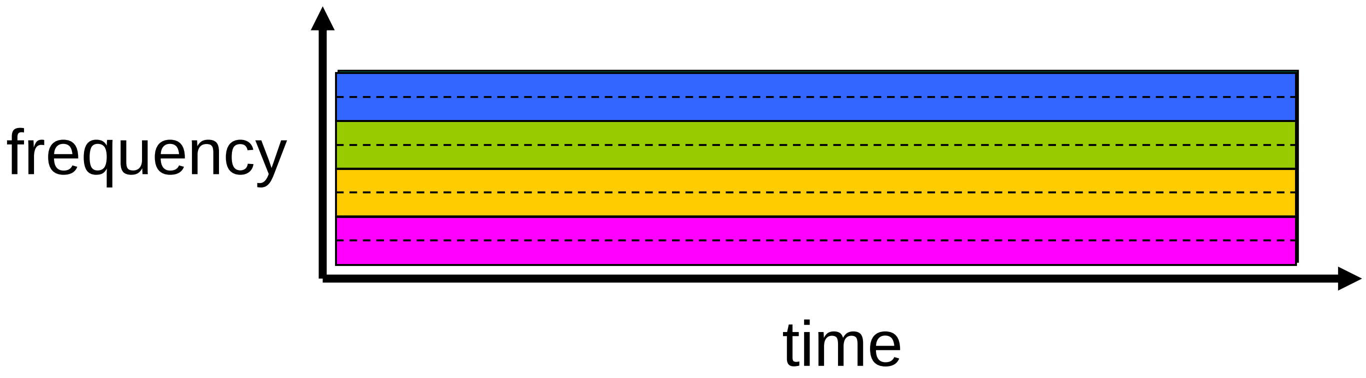
- must divide link bw into pieces...

Circuit Switching: FDM and TDM

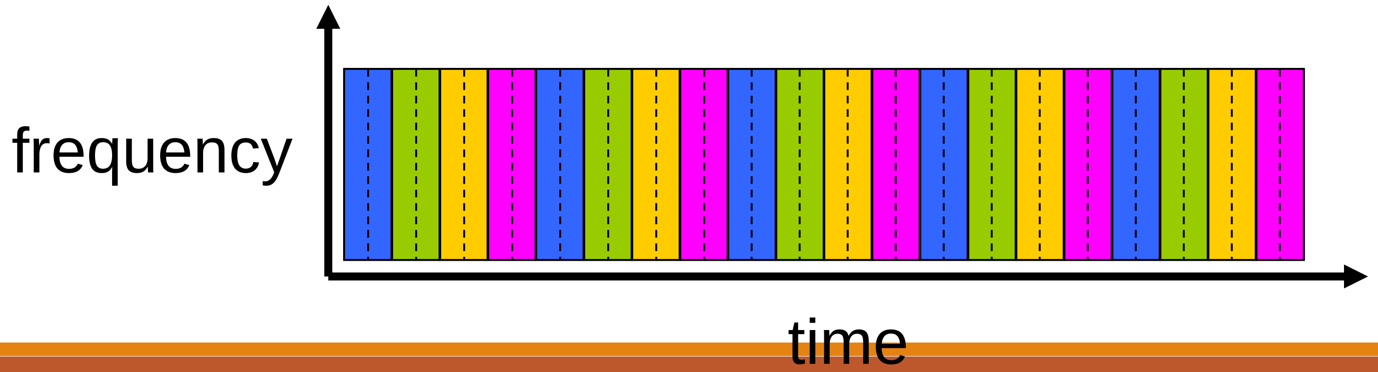
Example:

4 users

FDM



TDM



Network Core: Packet Switching

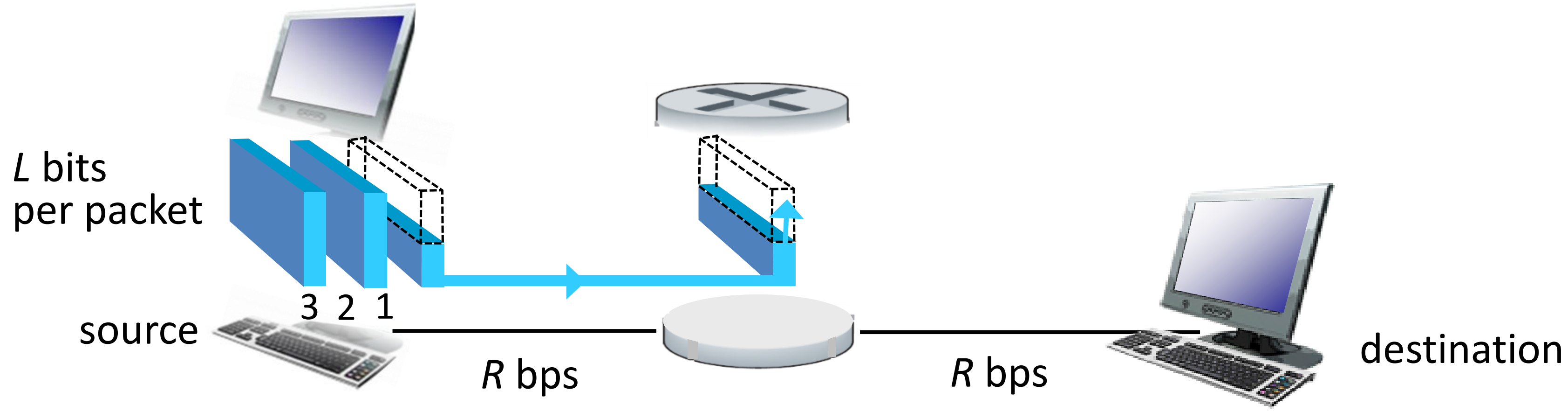
each end-end data stream divided into *packets*

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

resource contention:

- aggregate resource demand can exceed amount available
  - what happens if bandwidth is not available?
- congestion: packets queue, wait for link use
- store and forward: packets move one hop at a time
  - Node receives complete packet before forwarding

Packet-switching: store-and-forward



takes *L*/*R* seconds to transmit (push out) *L*-bit packet into link at *R* bps

store and forward: entire packet must arrive at router before it can be transmitted on next link

- end-end delay = 2*L*/*R* (assuming zero propagation delay)

more on delay shortly …

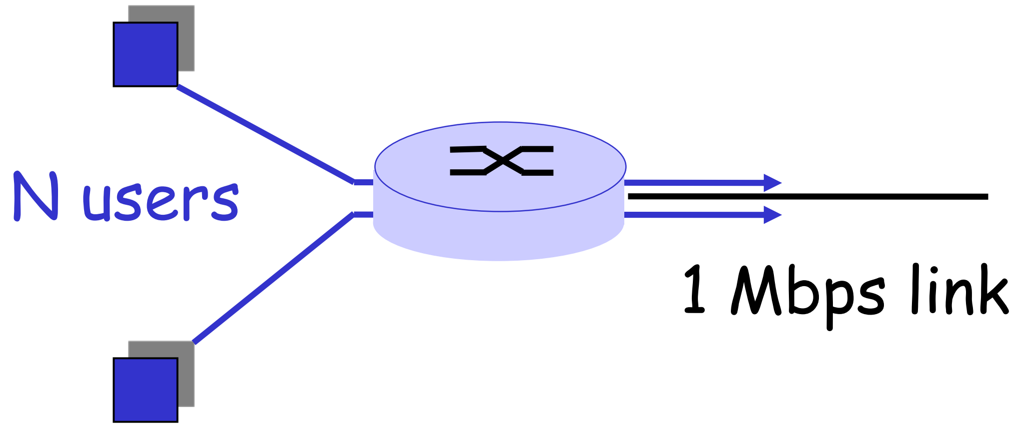
one-hop numerical example:

- *L* = 7.5 Mbits
- *R* = 1.5 Mbps
- one-hop transmission delay = 5 sec

Packet switching versus circuit switching

Packet switching allows more users to use network!

- 1 Mb/s link
- each user:
  - 100 kb/s 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

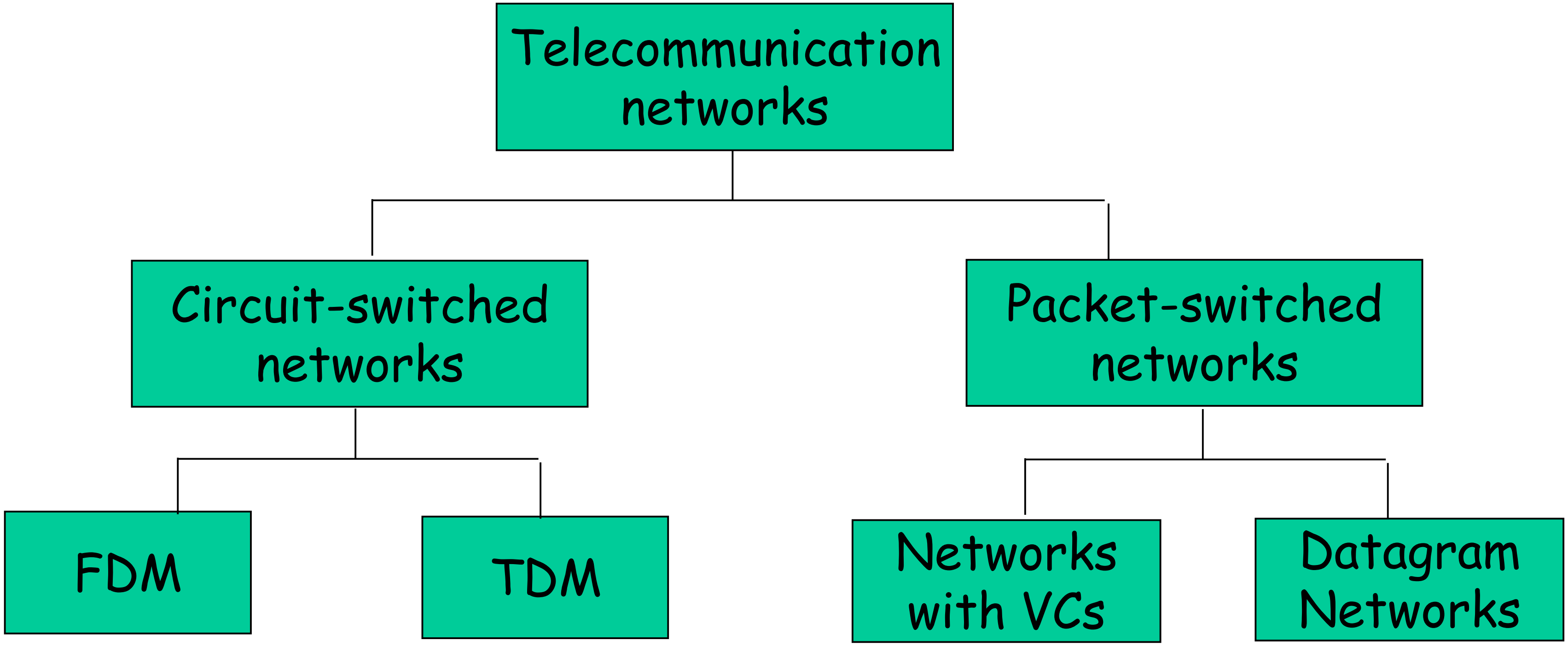
Is packet switching a "slam dunk winner?"

- Great for bursty data
  - resource sharing
  - simpler, no call setup
- Excessive congestion: packet delay and loss
  - protocols needed for reliable data transfer, congestion control
- Circuit Switching = Guaranteed behavior

Packet-switched networks: forwarding

- *Goal:* move packets through routers from source to destination
- 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



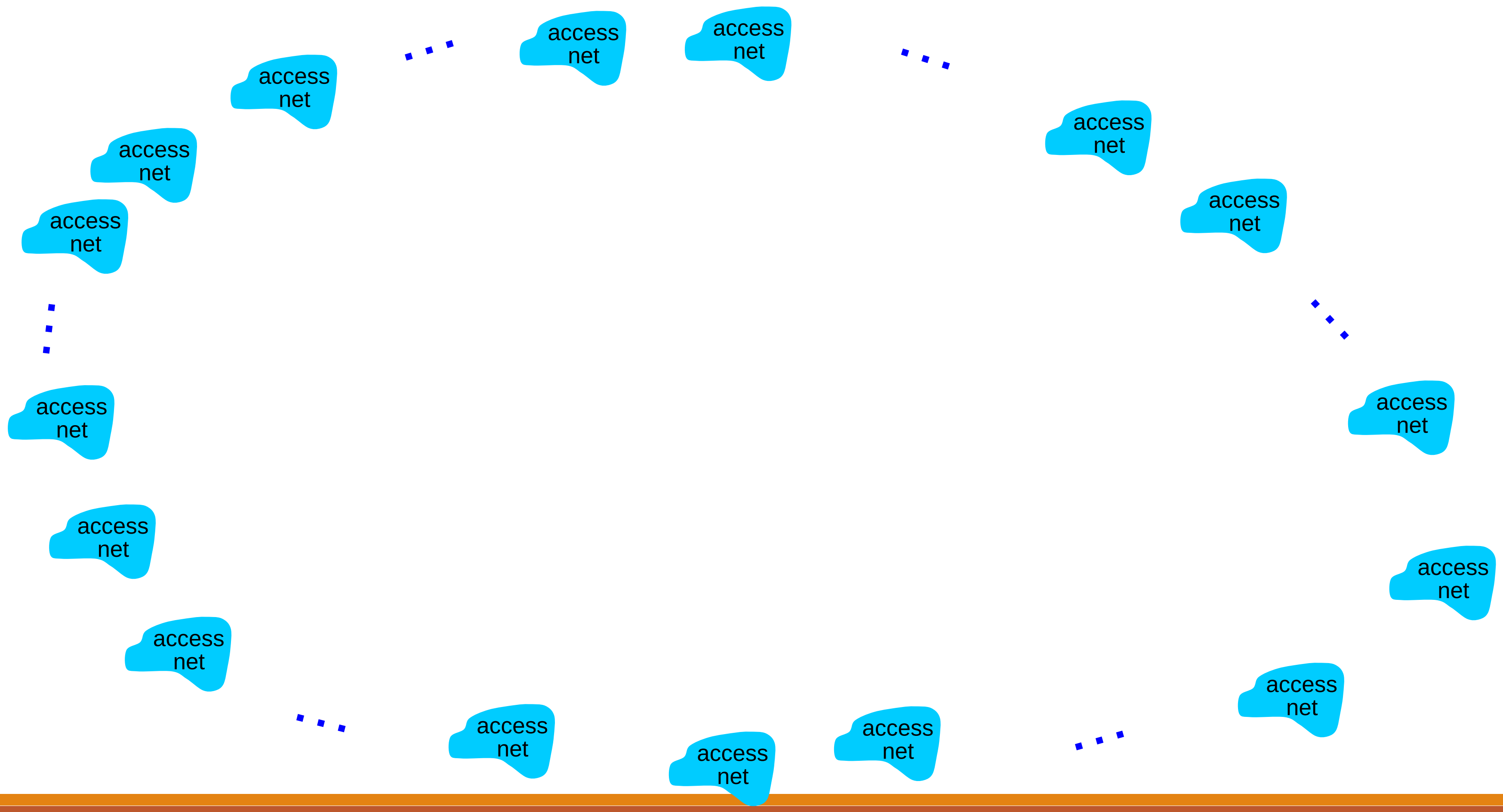
- Internet provides both connection-oriented (TCP) and connectionless services (UDP) to apps.

Internet structure: network of networks

- End systems connect to Internet via access ISPs (Internet Service Providers)
  - Residential, company and university ISPs
- Access ISPs in turn must be interconnected.
  - So that any two hosts can send packets to each other
- Resulting network of networks is very complex
  - Evolution was driven by economics and national policies
- Let's take a stepwise approach to describe current Internet structure

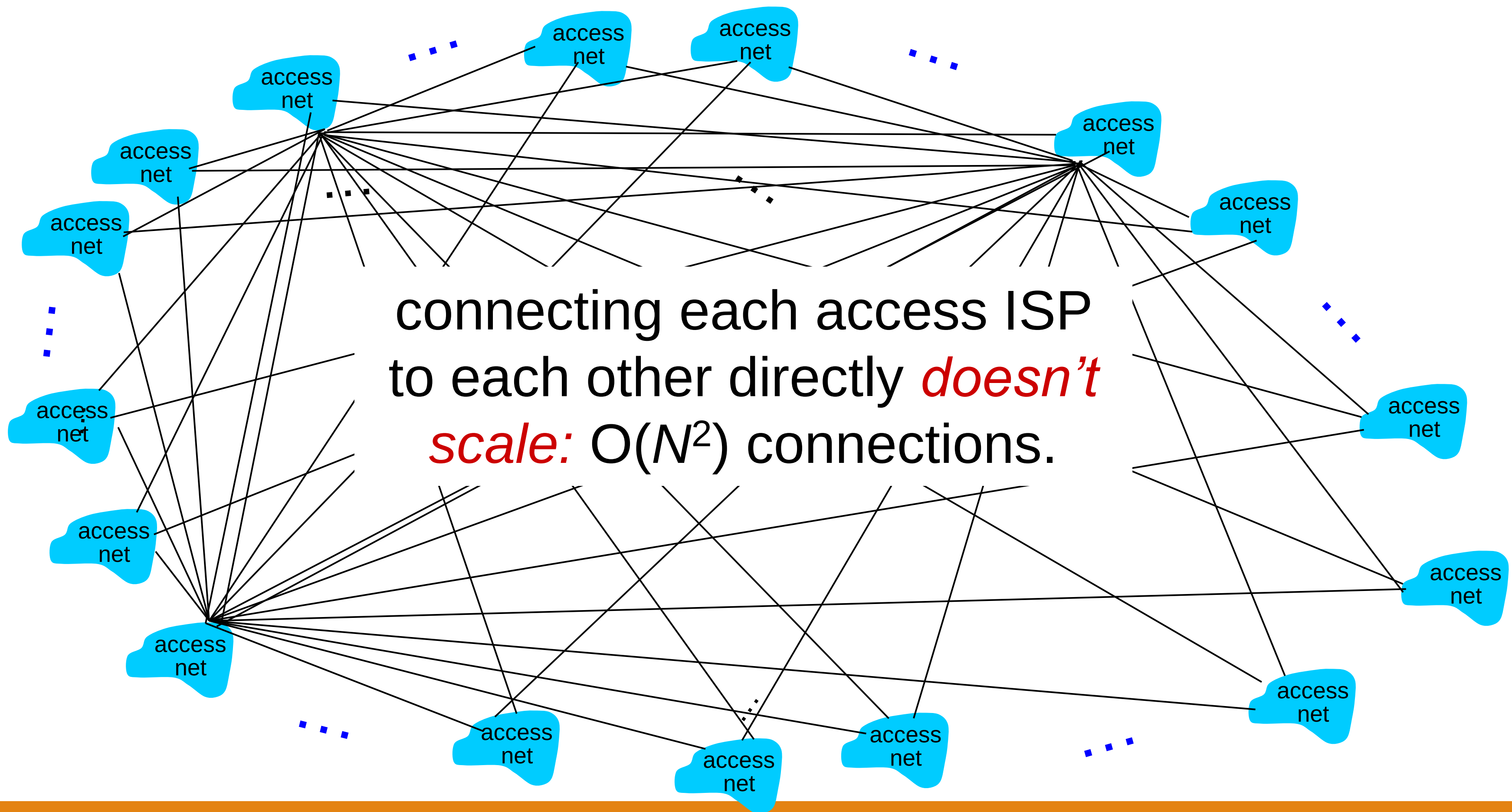
Internet structure: network of networks

Question: given *millions* of access ISPs, how to connect them together?



Internet structure: network of networks

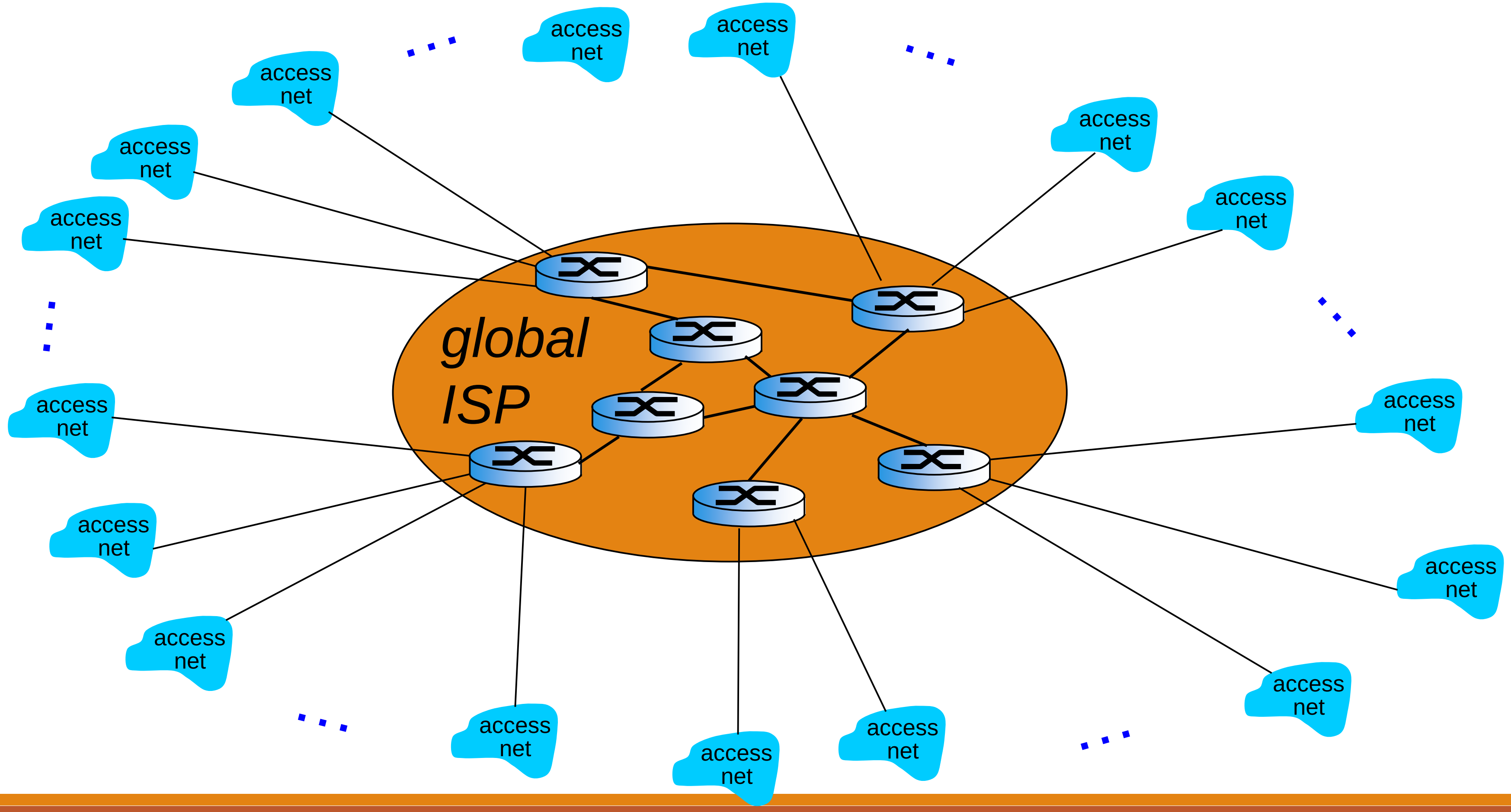
Option: connect each access ISP to every other access ISP?



connecting each access ISP to each other directly *doesn't scale:* O(N^2) connections.

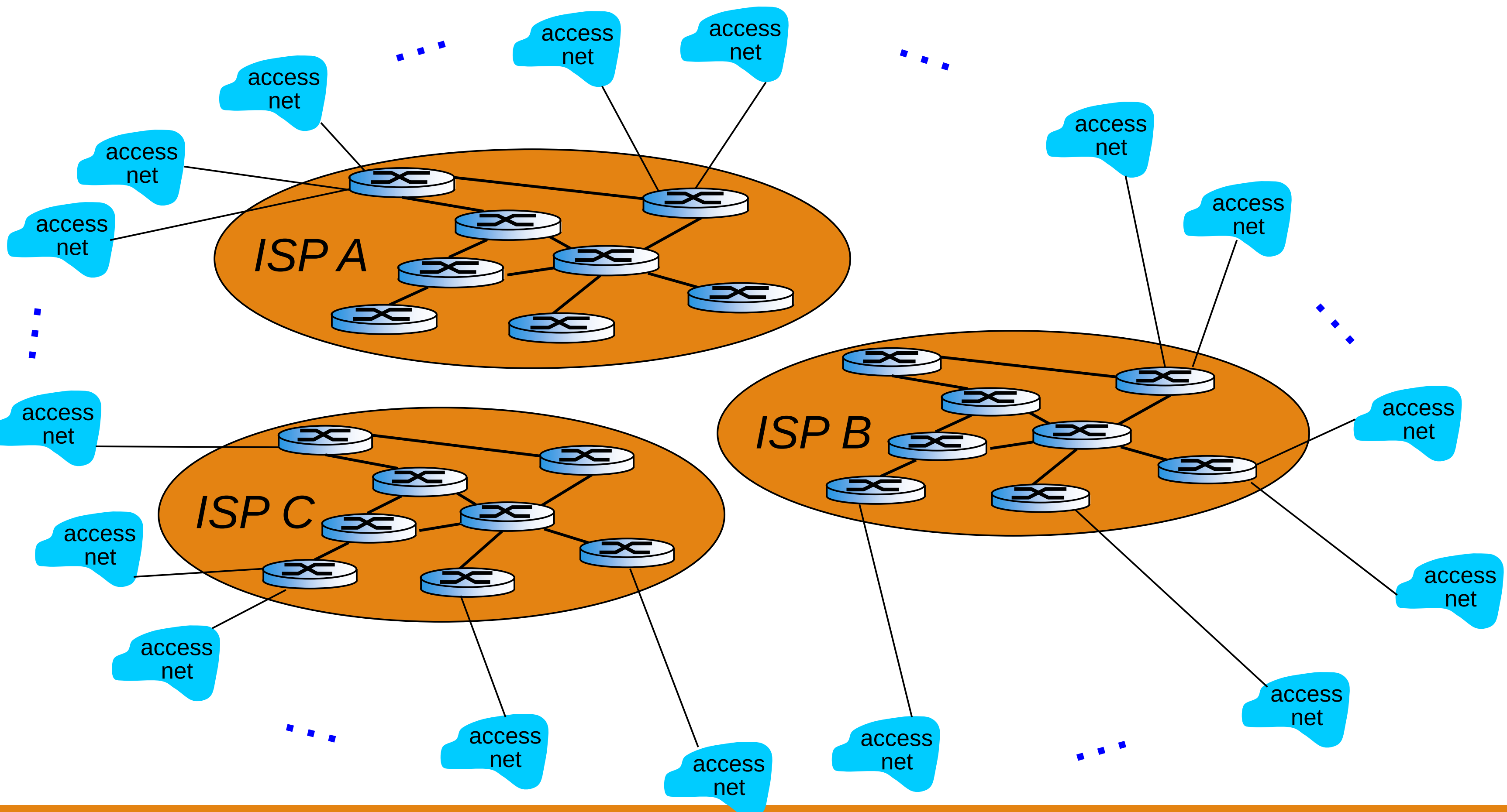
Internet structure: network of networks

Option: connect each access ISP to a global transit ISP?
Customer and provider ISPs have economic agreement.



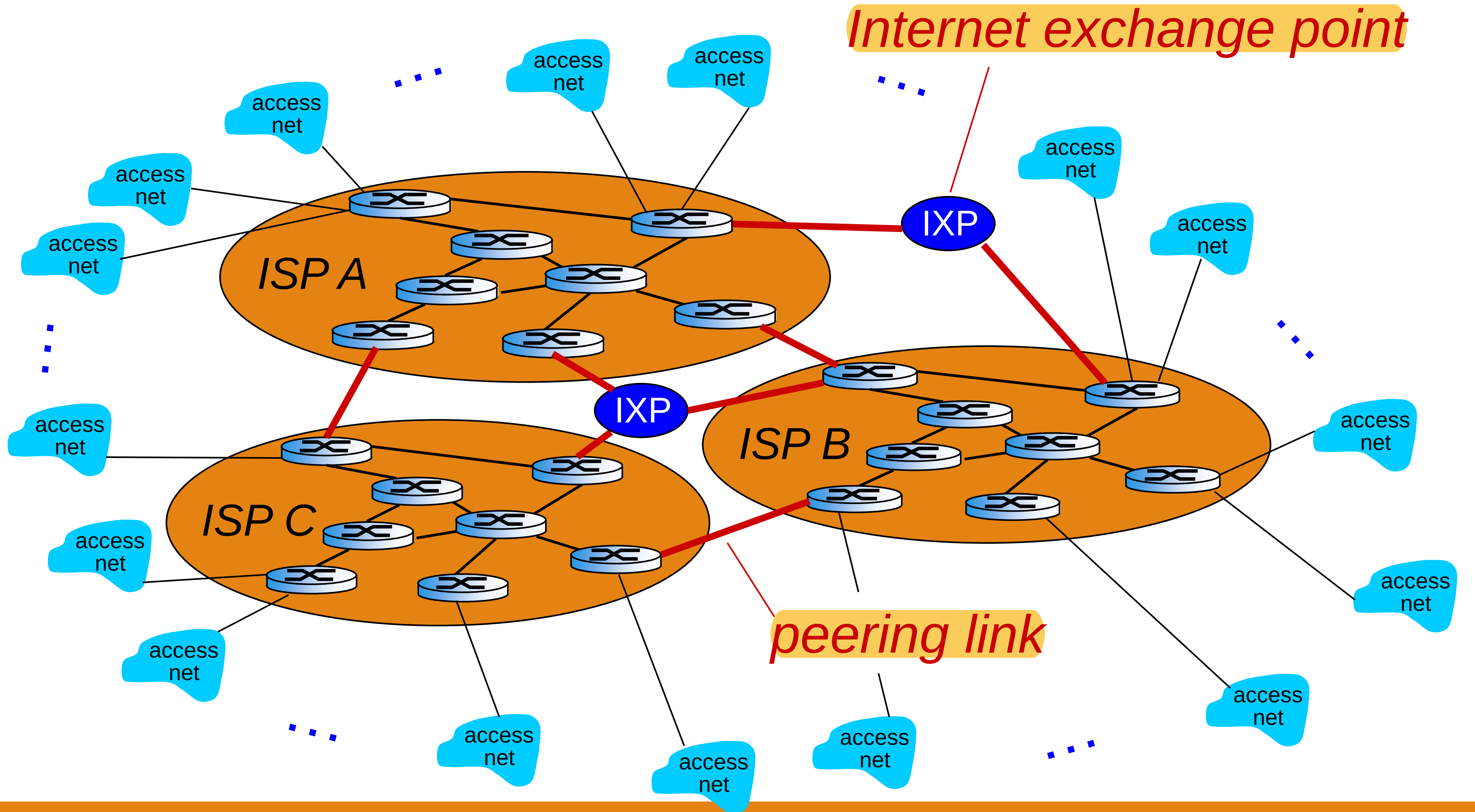
Internet structure: network of networks

But if one global ISP is viable business, there will be competitors ….



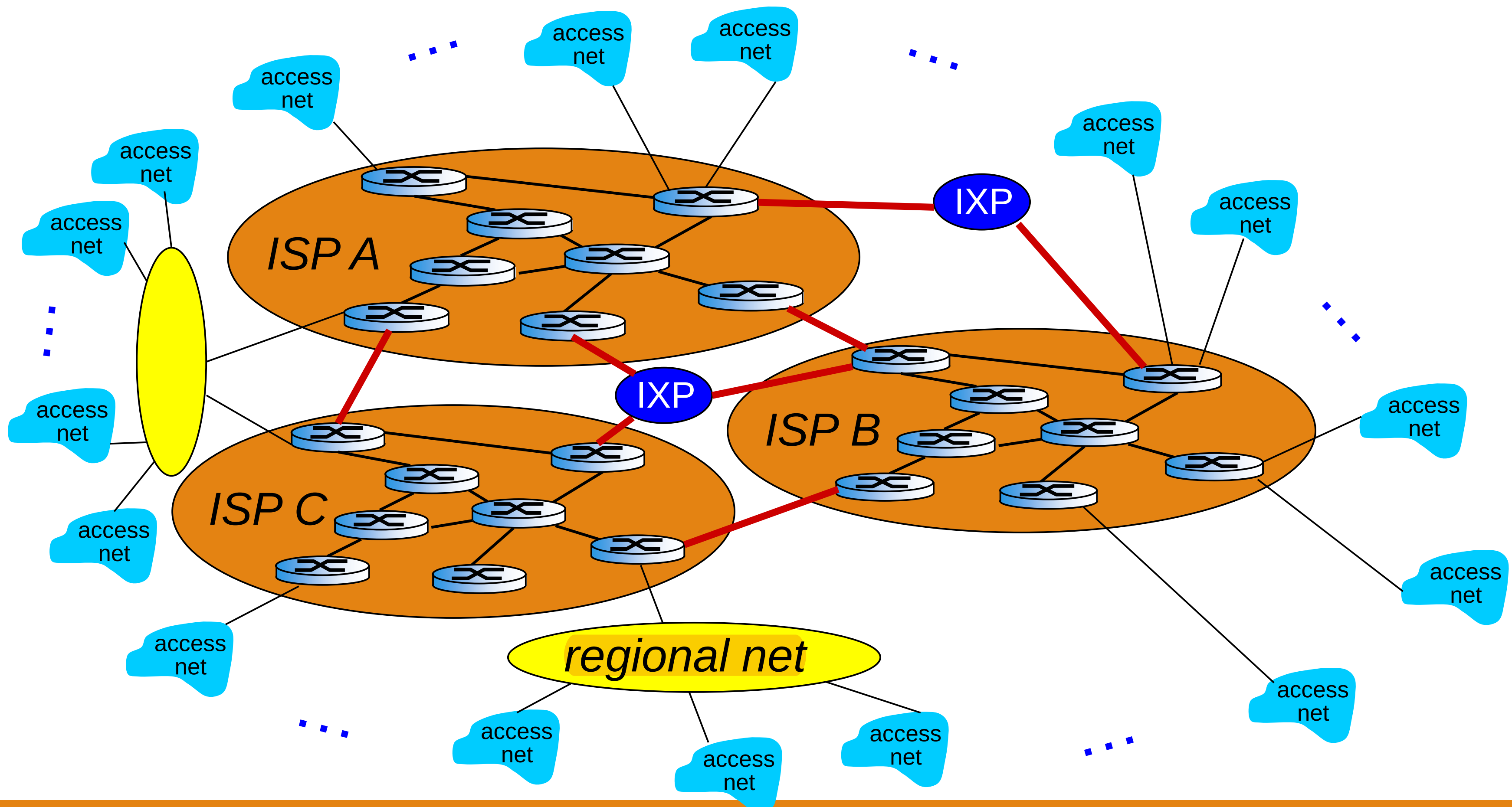
Internet structure: network of networks

But if one global ISP is viable business, there will be competitors …. which must be interconnected



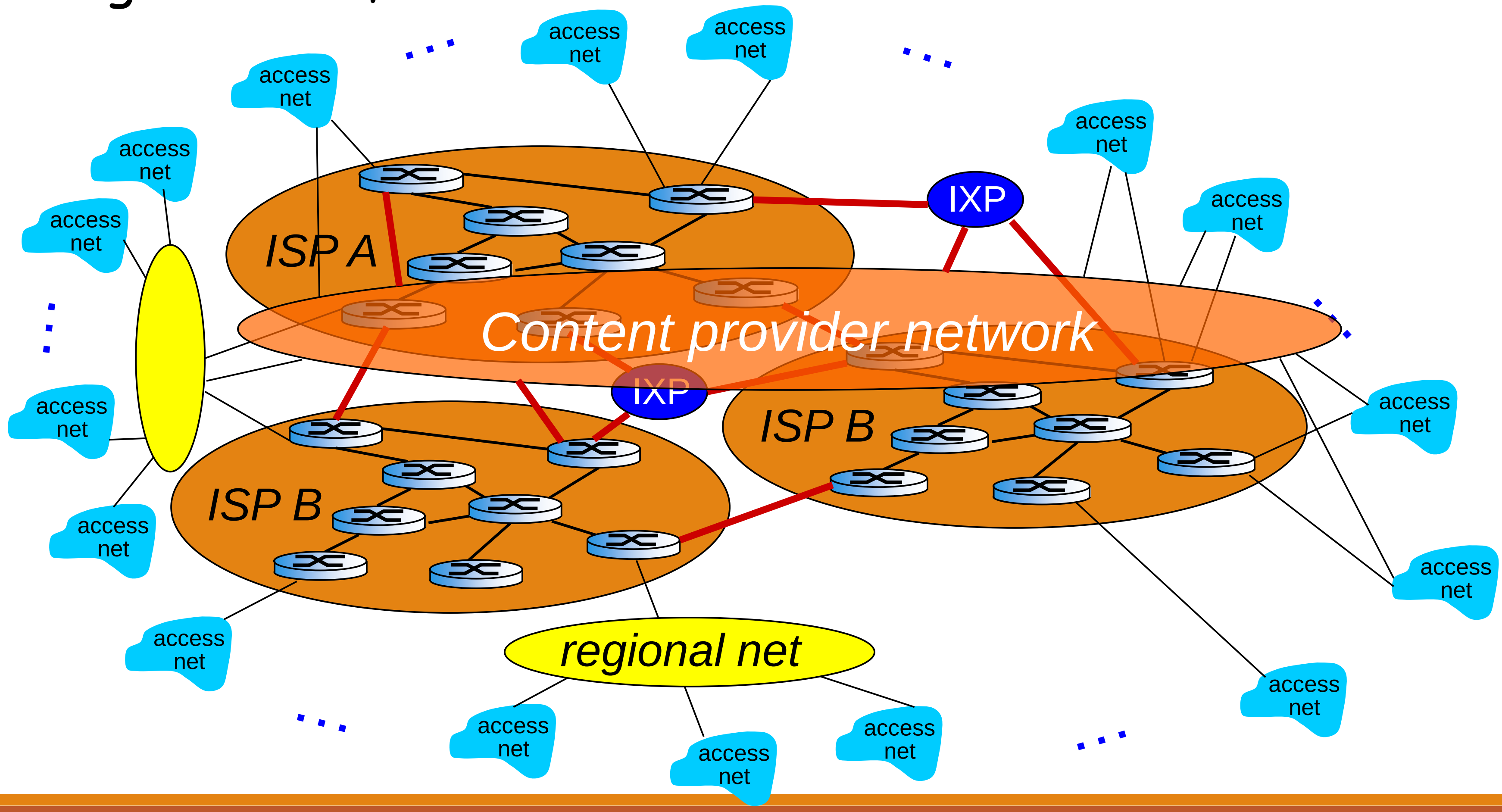
Internet structure: network of networks

… and regional networks may arise to connect access nets to ISPS

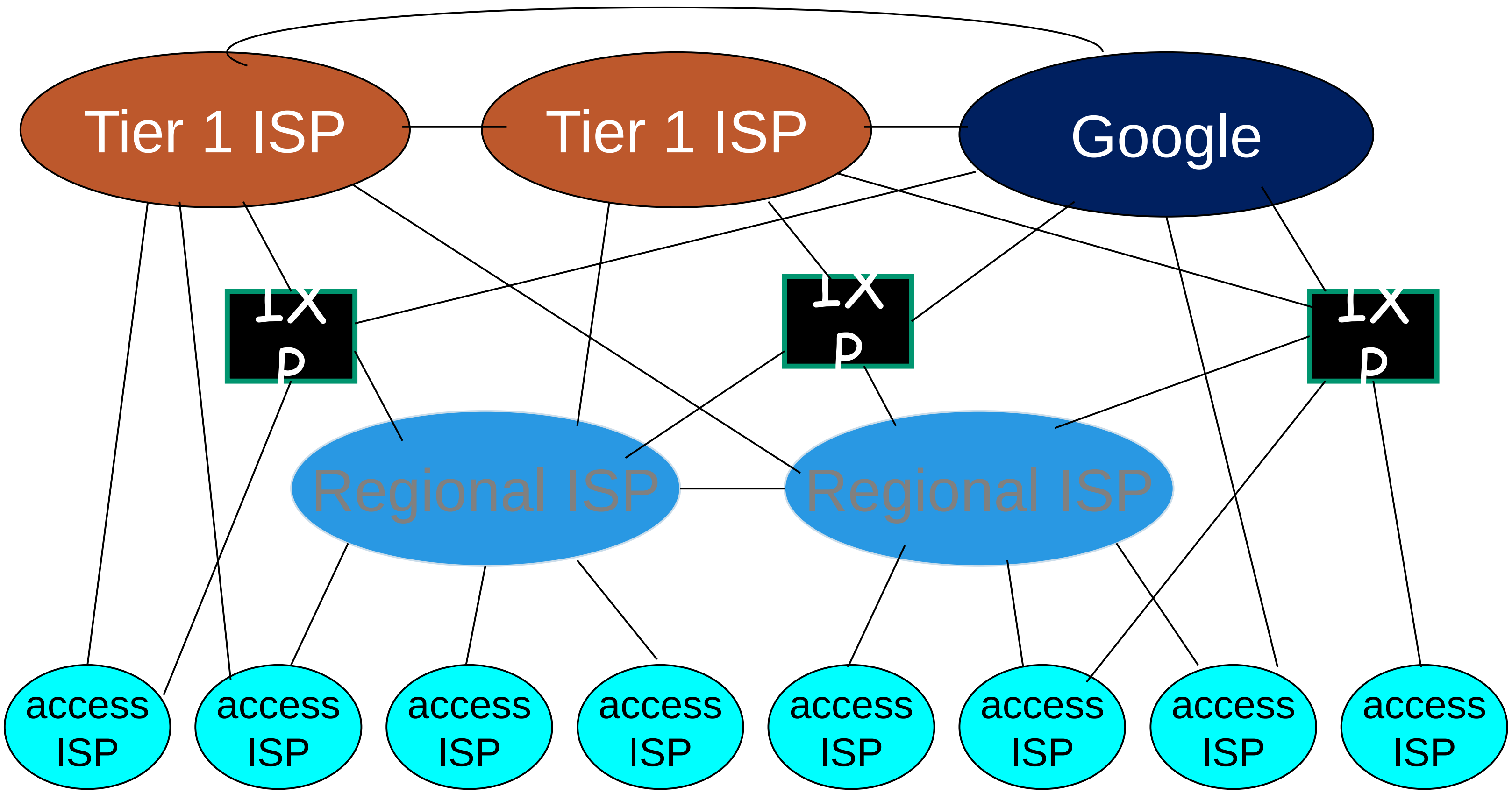


Internet structure: network of networks

… and content provider networks (e.g., Google, Microsoft, Akamai) may run their own network, to bring services, content close to end users



Internet structure: network of networks



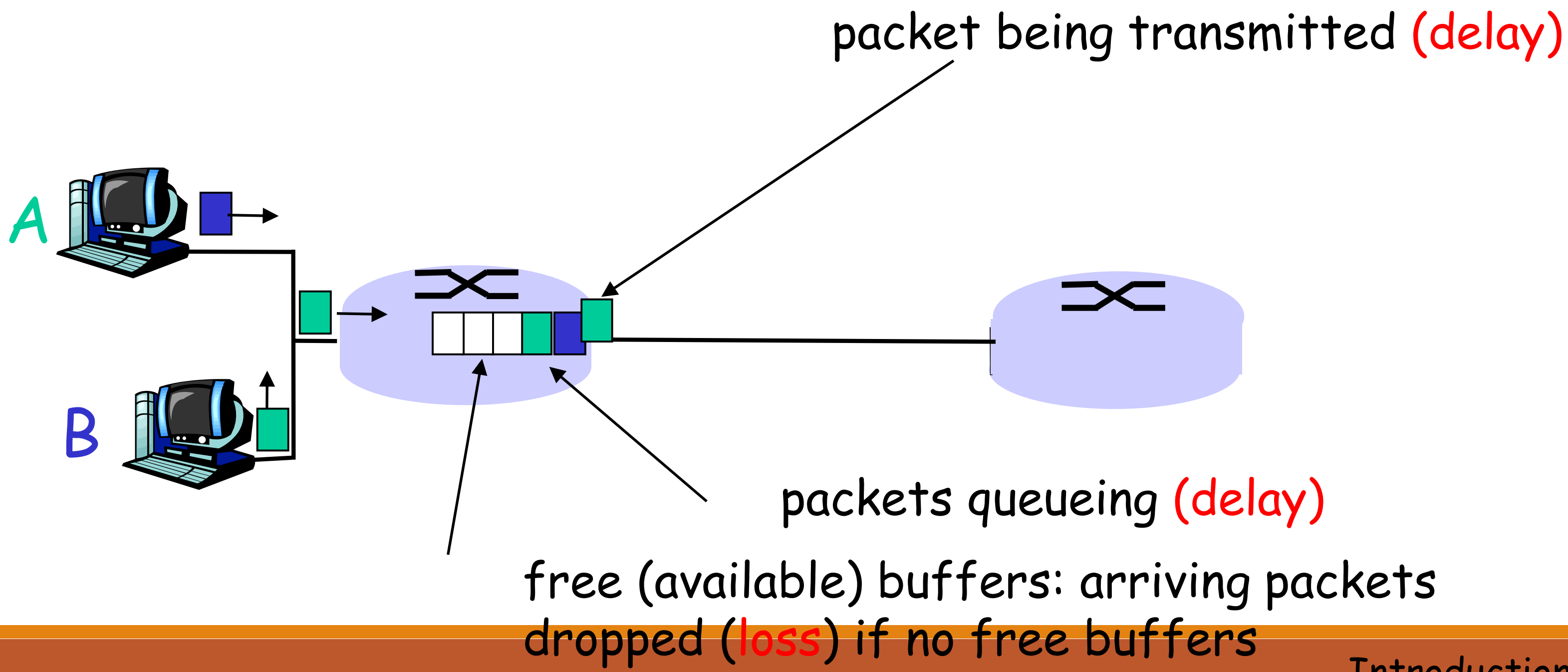
at center: small # of well-connected large networks

- "tier-1" commercial ISPs (e.g., Level 3, Sprint, AT&T, NTT), national & international coverage
- content provider network (e.g, Google): private network that connects it data centers to Internet, often bypassing tier-1, regional ISPs

How do loss and delay occur?

packets *queue* in router buffers

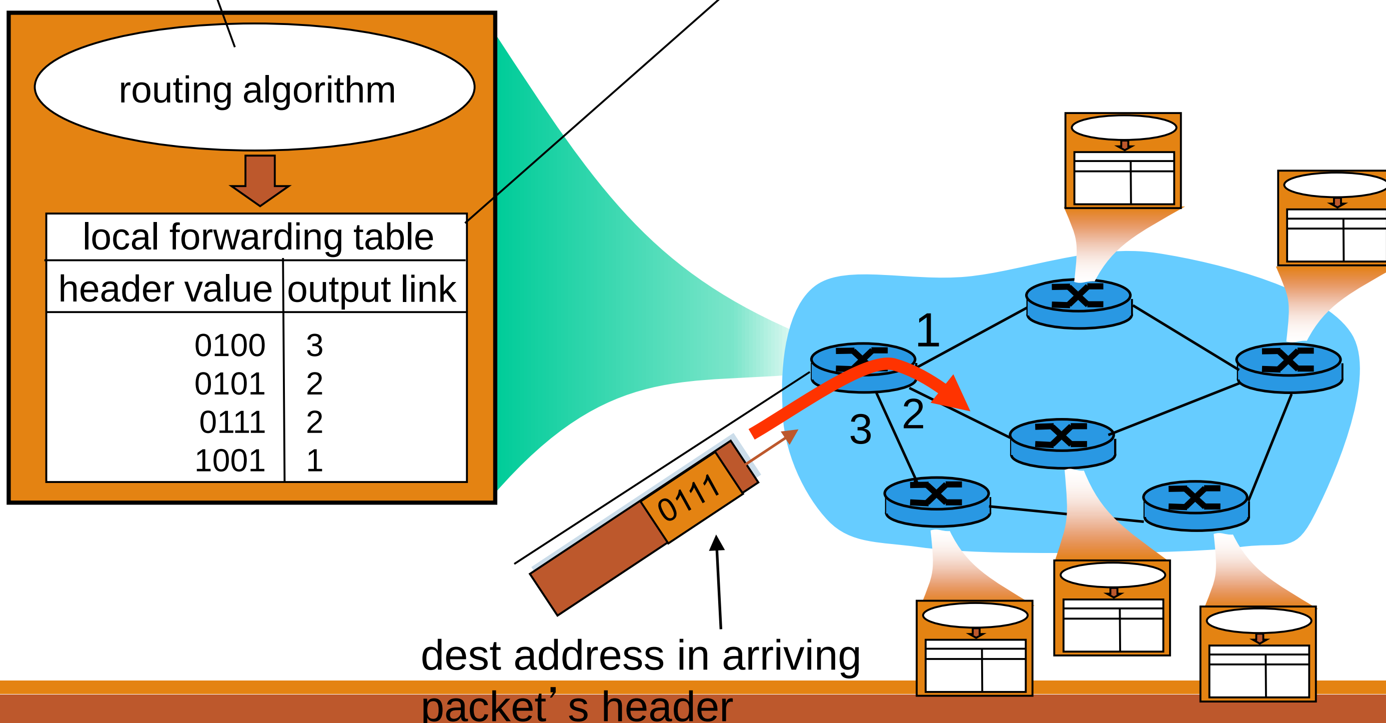
- packet arrival rate to link exceeds output link capacity
- packets queue, wait for turn



Two key network-core functions

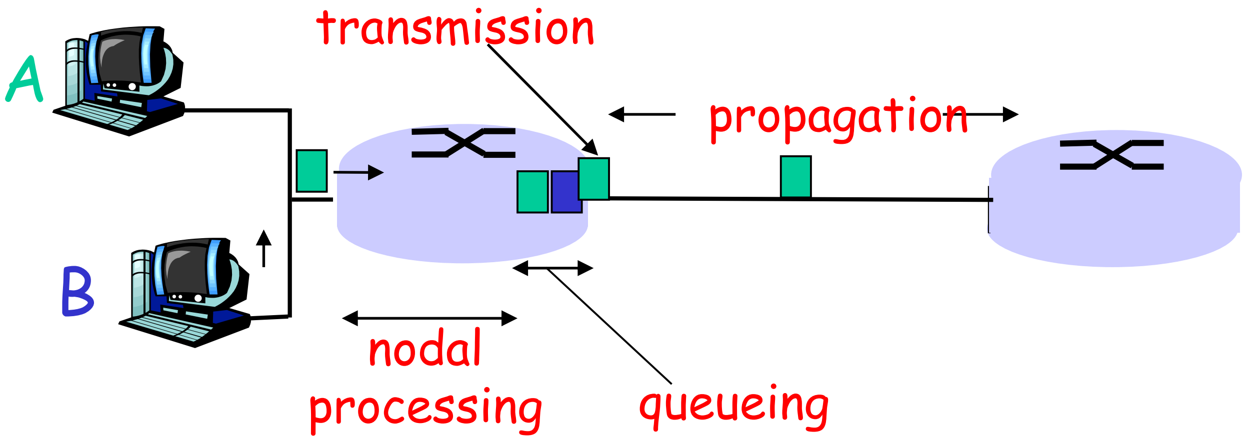
routing: determines source-destination route taken by packets
- *routing algorithms*

forwarding: move packets from router's input to appropriate router output



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

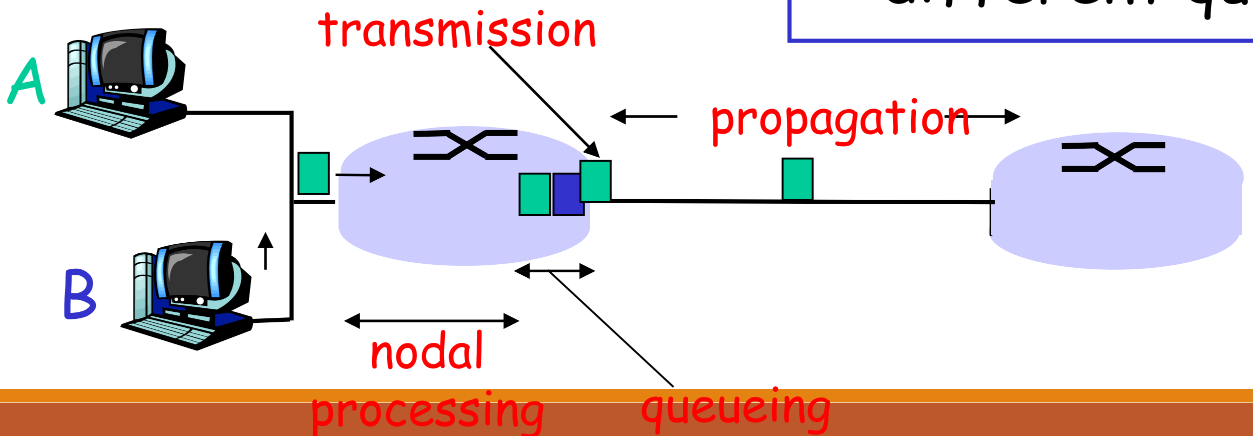
3. Transmission delay:

- 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 (~2×10^8 m/sec)
- propagation delay = d/s

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



Total Nodal delay

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

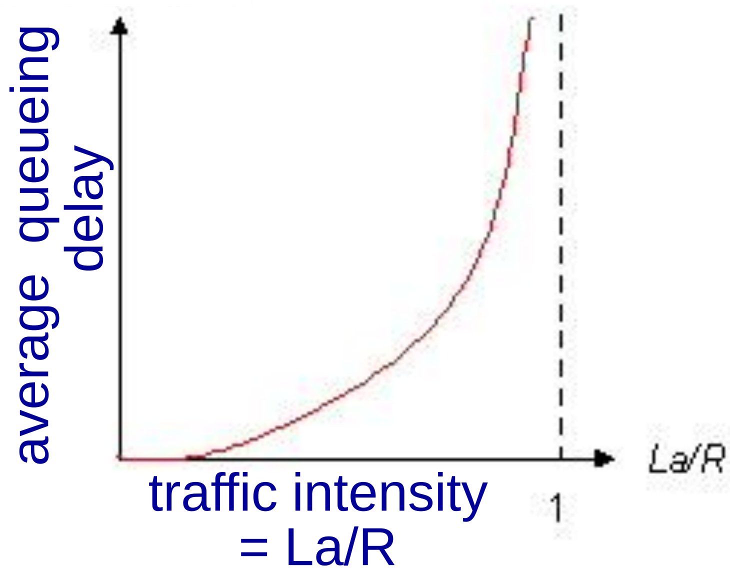
- d_{proc} = processing delay
  - typically a few microsecs or less
- d_{queue} = queuing delay
  - depends on congestion
- d_{trans} = transmission delay
  - = L/R, significant for low-speed links
- d_{prop} = propagation delay
  - a few microsecs to hundreds of msecs

Queueing delay (revisited)

R: link bandwidth (bps)

L: packet length (bits)

a: average packet arrival rate



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

La/R ~ 0

La/R -> 1

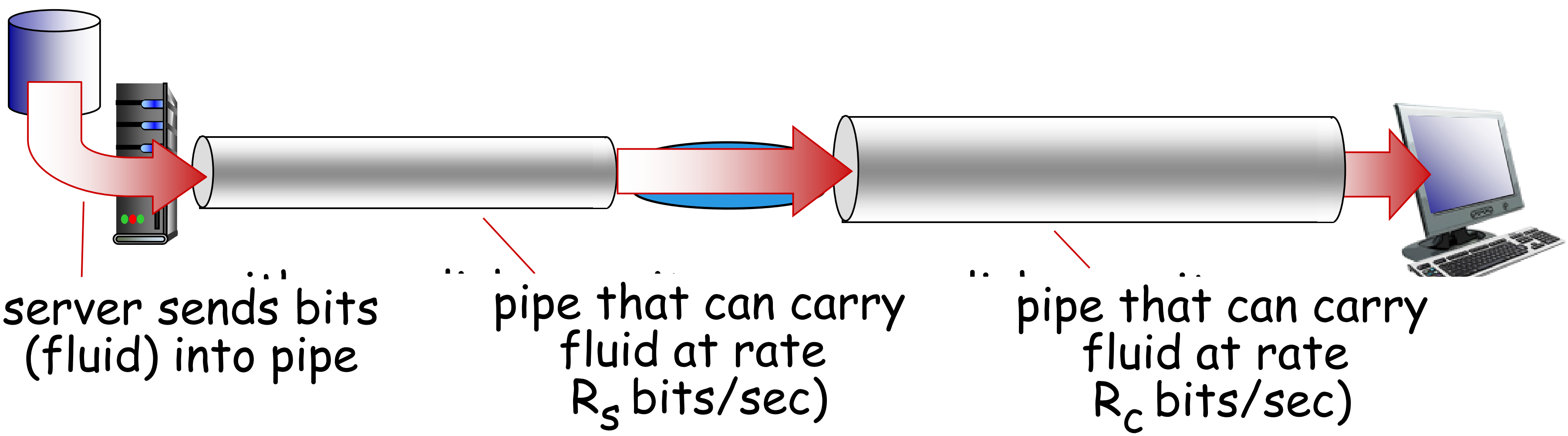
Packet loss

- 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

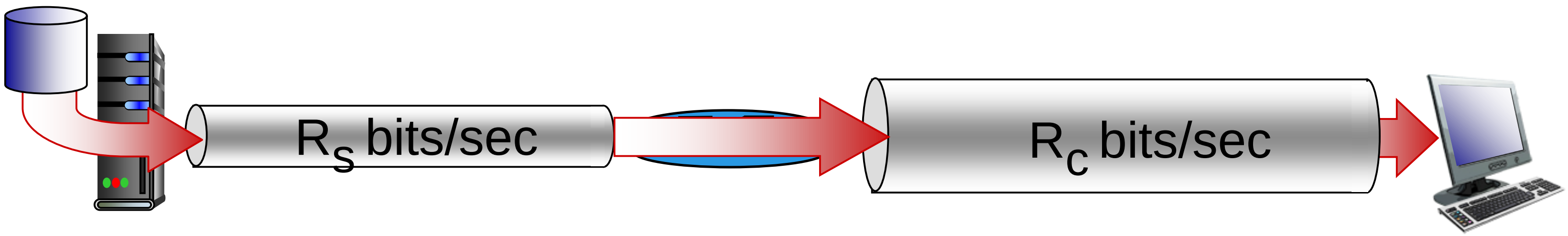
throughput: rate (bits/time unit) at which bits transferred between sender/receiver

- *instantaneous:* rate at given point in time
- *average:* rate over longer period of time

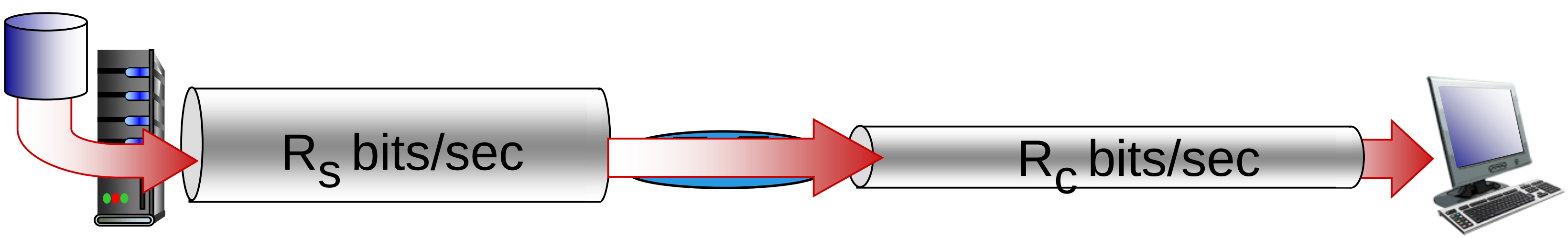


Throughput (more)

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



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

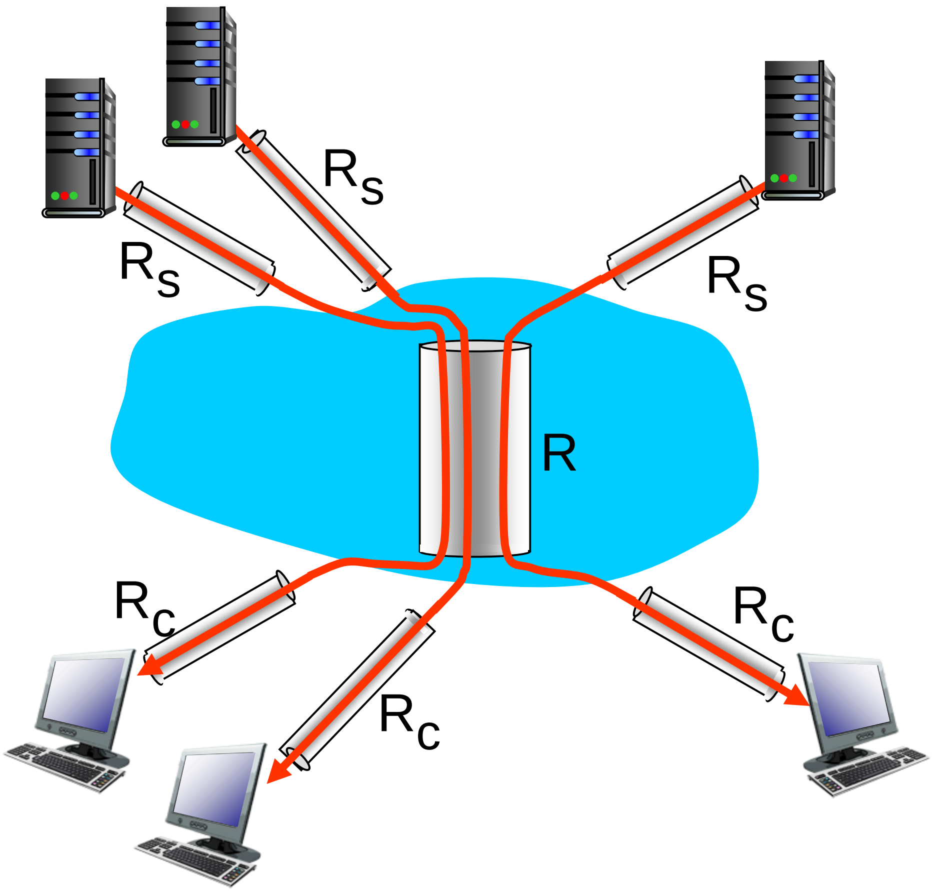


bottleneck link
link on end-end path that constrains end-end throughput

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

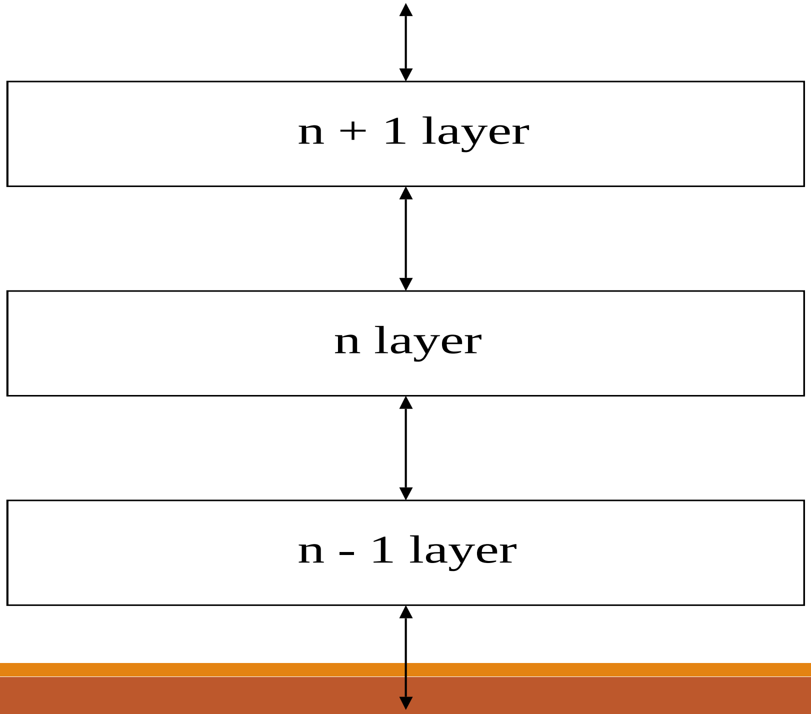
Network System Modularity

Like software modularity, but:

- Implementation is distributed across many machines (routers and hosts)
- Must decide:
  - – How to break system into modules
    - **Layering**
  - – Where modules are implemented
    - **End-to-End Principle**
  - – Where state is stored

Layering Concept

- A restricted form of abstraction: system functions are divided into layers, one built upon another
- Often called a *stack*; but **not** a data structure!



Layers and Communications

- Interaction only between adjacent layers
- *layer n* uses services provided by *layer n-1*
- *layer n* provides service to *layer n+1*
- Bottom layer is physical media
- Top layer is application

Entities and Peers

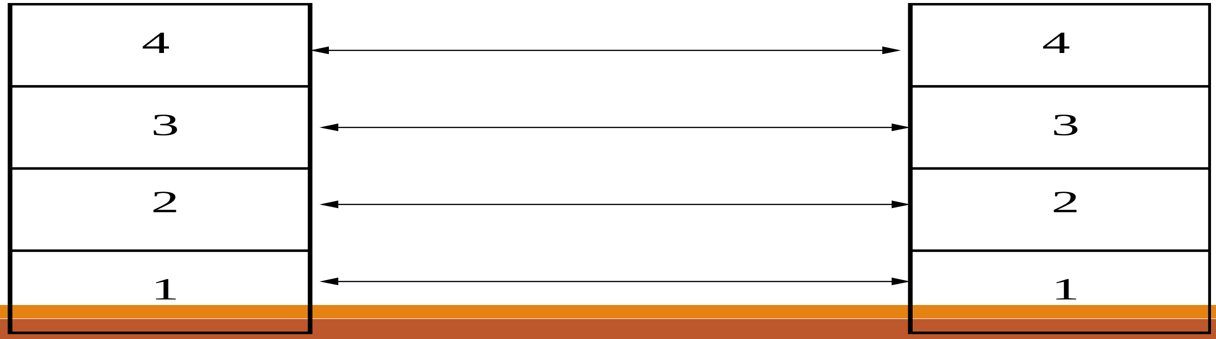
Entity – a *thing* (an independent existence)

Entities *interact* with the layers above and below

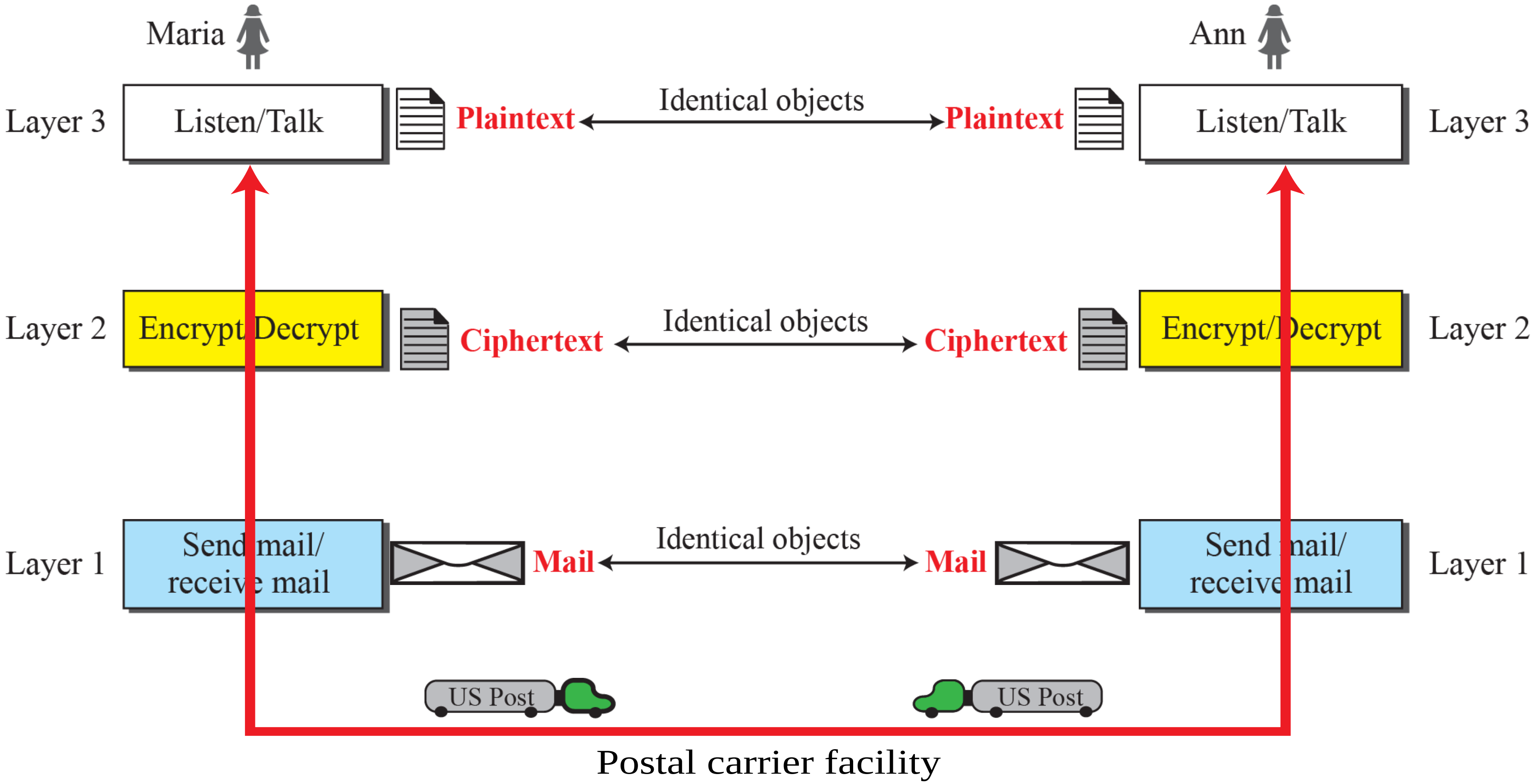
Entities *communicate* with *peer* entities

– same level but different place (eg different person, different box, different host)

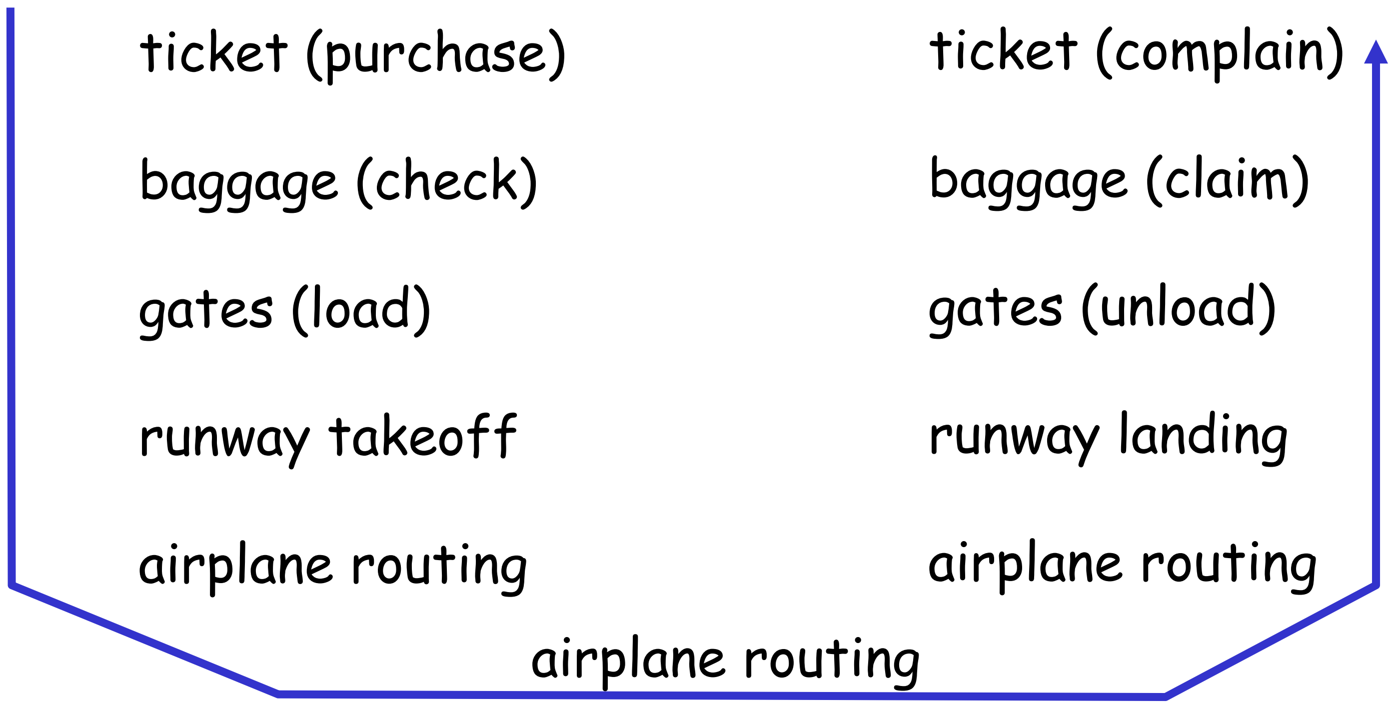
Communications between peers is supported by entities at the lower layers



Example: A three-layer protocol

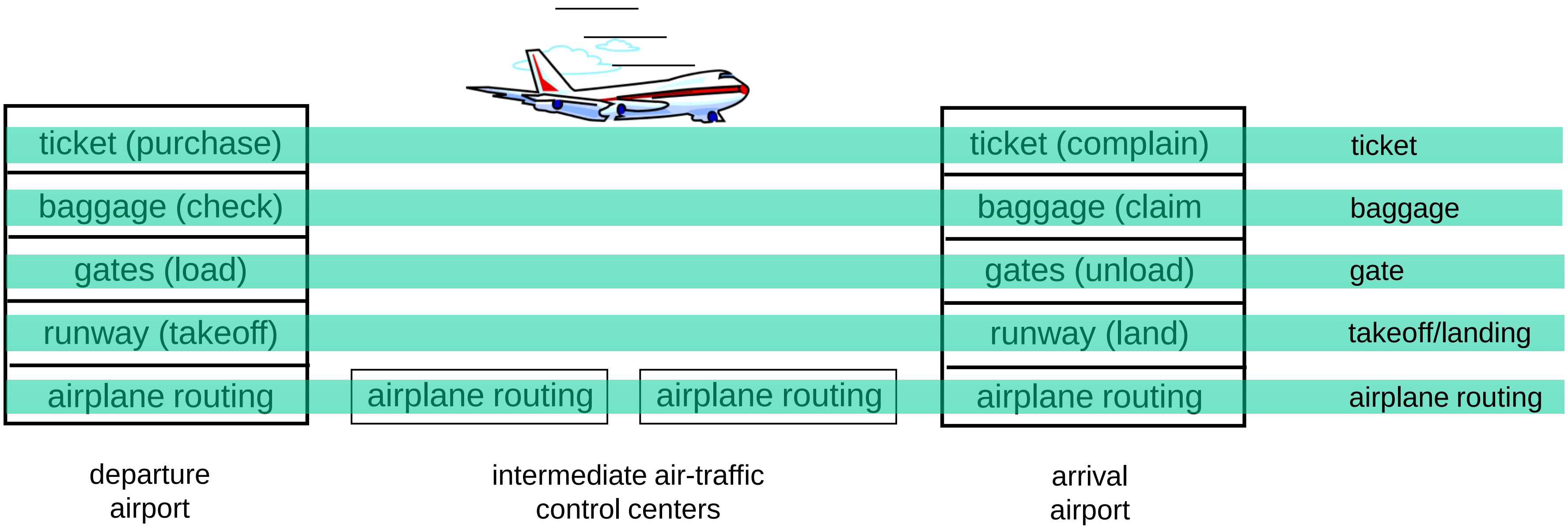


Organization of air travel



- 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

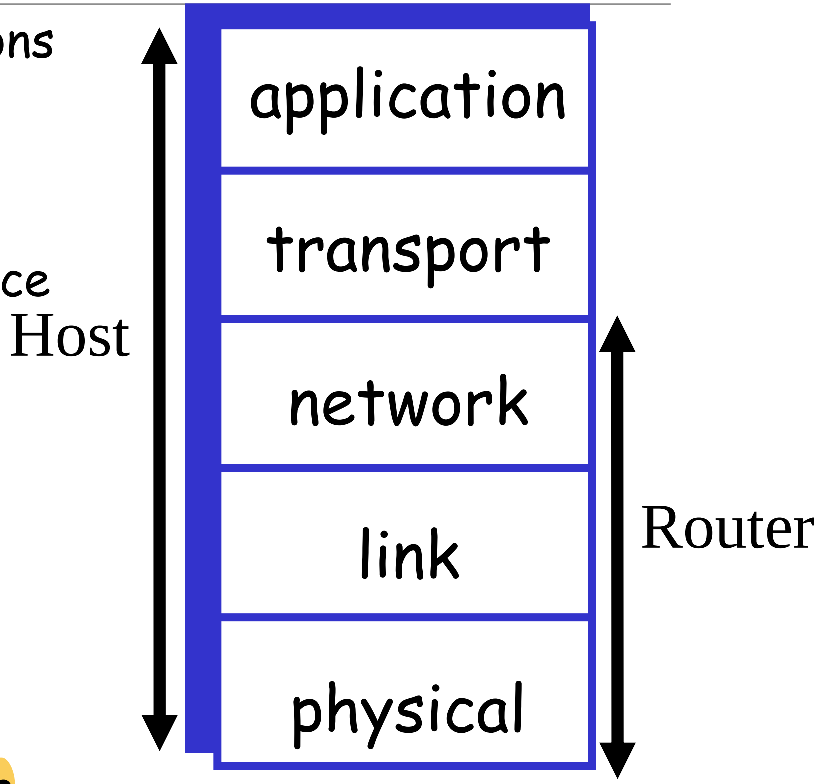
Why layering?

Dealing with complex systems:

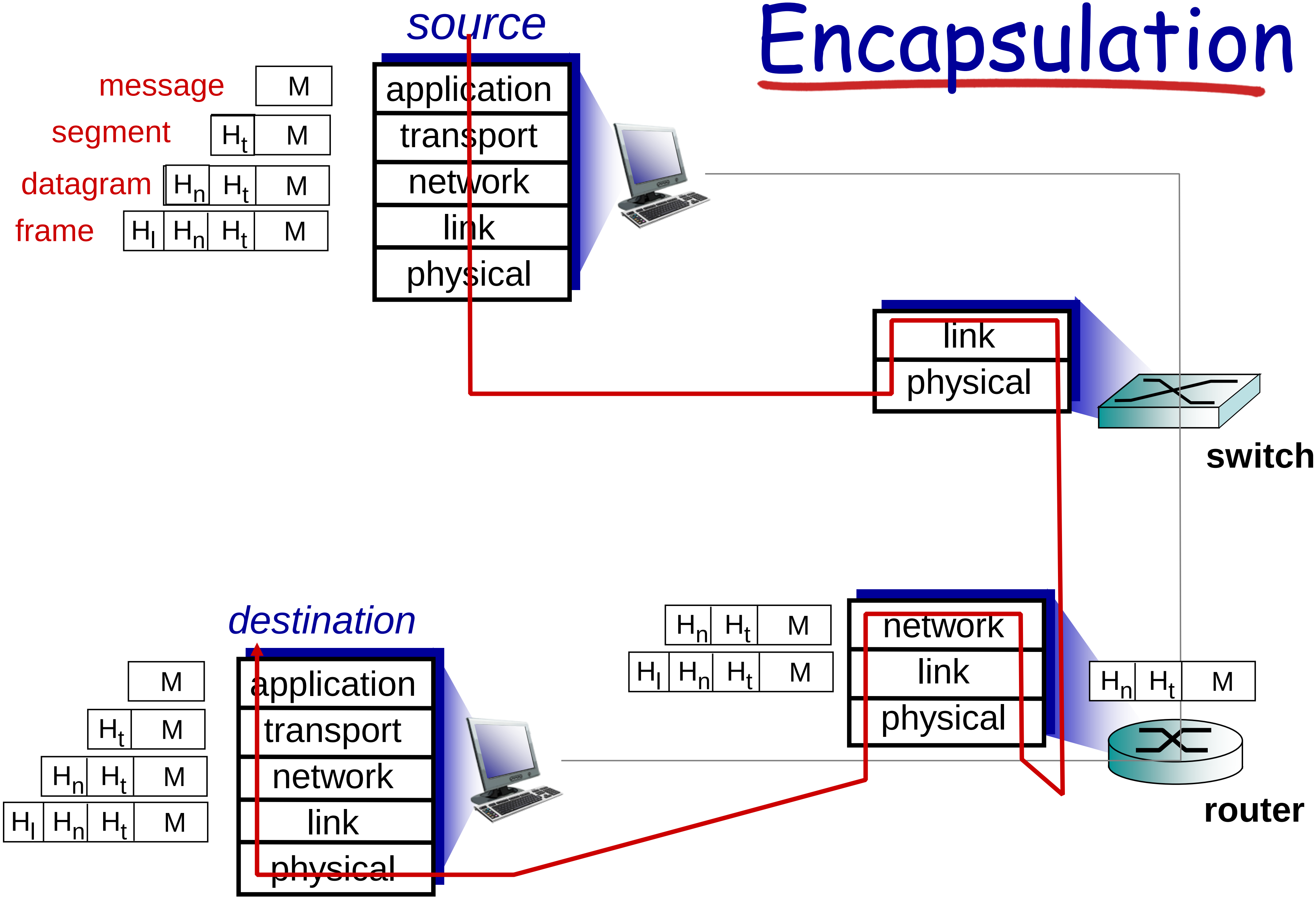
- explicit structure allows identification, relationship of complex system's pieces
  - layered reference model for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't 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", modulation scheme, line-coding format, electrical & physical specifications, etc.
- Routers in the network operate only up to the Network Layer



Encapsulation



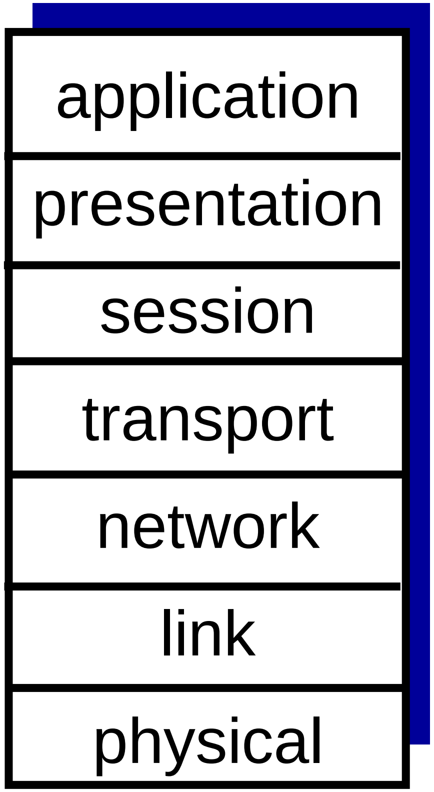
ISO/OSI reference model

presentation: allow applications to interpret meaning of data, e.g., encryption, compression, machine-specific conventions

session: synchronization, checkpointing, recovery of data exchange

Internet stack "missing" these layers!

- these services, *if needed*, must be implemented in application



1.9 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

Intern
et History

1990's: commercialization, the WWW

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

Late 1990's & 2000's:

- est. 50 million computers on Internet
- est. 100 million+ users
- backbone links running at 1 Gbps