Computer Networks

CS 341, CS 342
September-November 2020
Moumita Patra and
Prof. Sukumar Nandi

CS 341- Computer Networks

Welcome to CS 341 (Theory) and CS 342 (Lab) !

- Course components:
- Lectures
- Class Notes
- Assignments
- Quizes
- End sem

Quiz : 6 in total (6 x 10% weightage)
End sem: Theory + Viva (20 % + 20 % weightage)

Quiz: Approximately after every 6 lectures. Dates will be declared well ahead of time.

References:

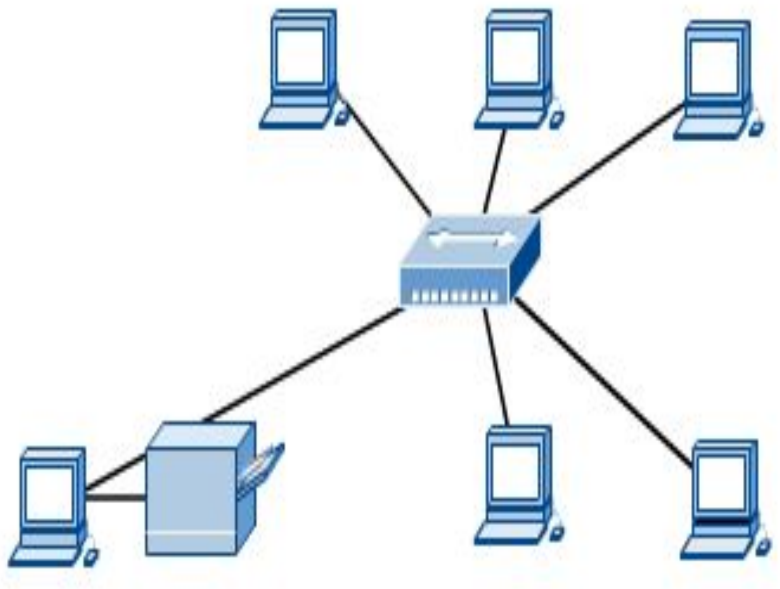
- **Computer Networking- A Top-Down Approach, Jim Kurose and Keith W. Ross**
- Computer Networks, by Andrew S Tanenbaum and David J Wetherall, 5th Ed and above, Prentice Hall.

Syllabus (Tentative , to be handled by Moumita Patra):

- Classification of Communication Networks. Standard models of communication: OSI and TCP/IP. Importance of layering and service models.
- Application layer services and protocols. Study of SMTP, HTTP, FTP, and DNS.
- Transport layer services, principles and protocols: study of TCP and UDP. Principles of reliability: sliding window protocols, selective repeat and go-back-N. Principles of congestion control: TCP case study. Details of TCP working.
- Network layer services, algorithms and protocols. Study of routing algorithms. Study of Internet router architecture. IP addressing principles: assignment and aggregation. Study of DHCP.

Computer Networks

- Computer networks connects two autonomous devices (PCs/mobile phones, etc.)
- The devices can be geographically located anywhere.



The Internet

- Millions of connected devices

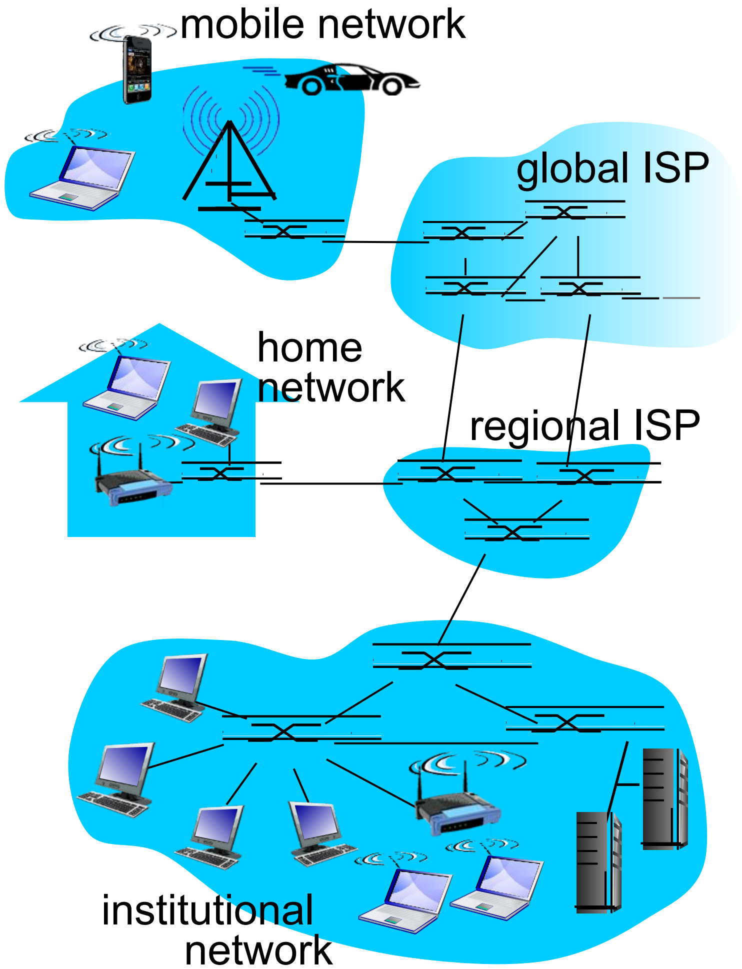
PC

server

Wireless laptop

smartphone

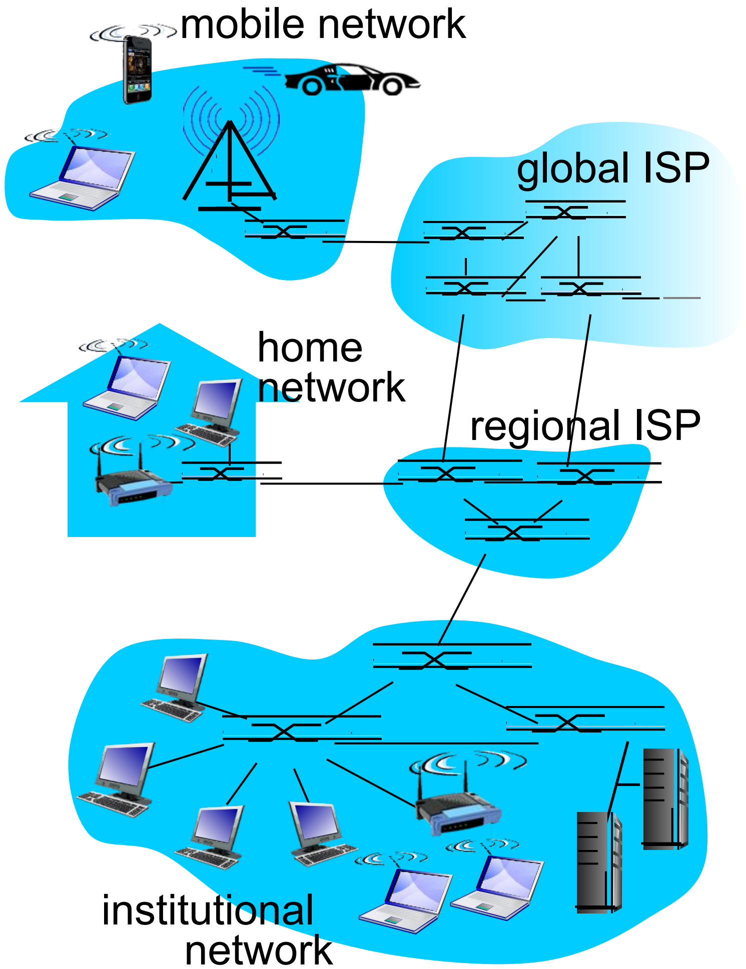
Hosts = end systems
Running network applications



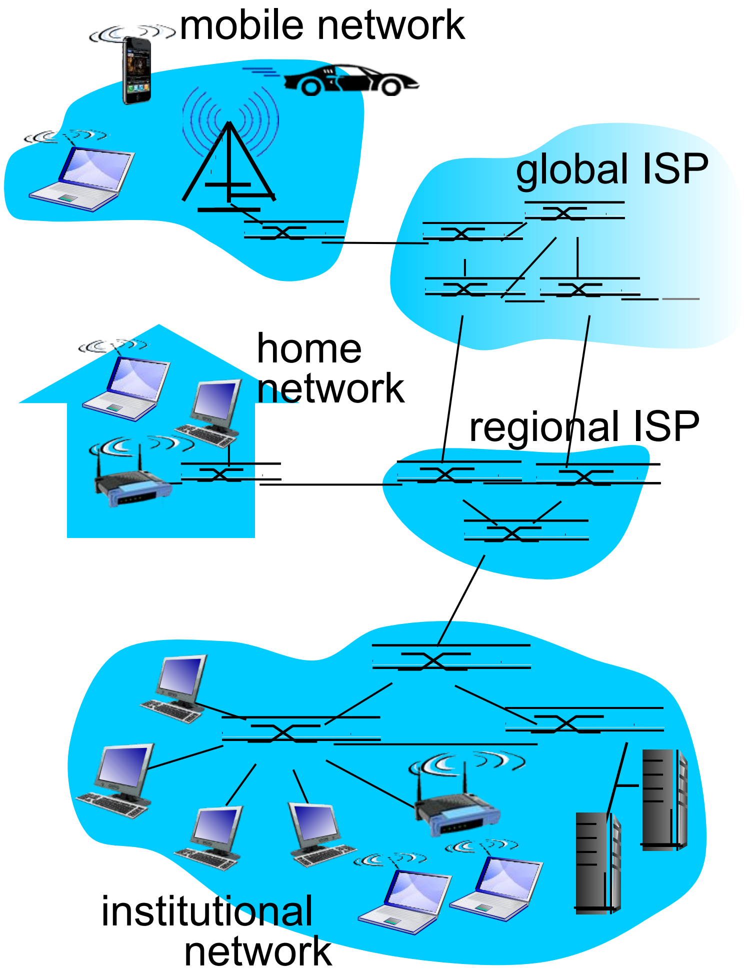
Communication Links

- Wired links
- Wireless links
- fiber, copper, radio, satellite
- transmission rate: different links can transmit data at different rates

- Routers and Switches
- Forward chunks of data (packets)

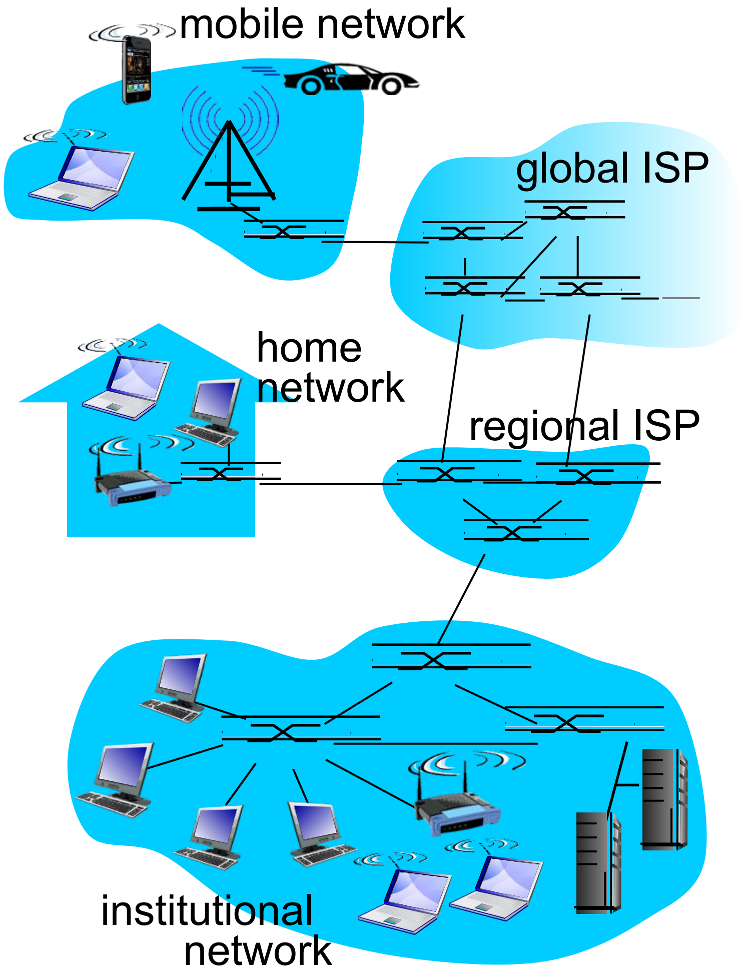


- **Internet:** "network of networks"
  - Interconnected ISPs
- **protocols** control sending and receiving of data
  - e.g., TCP, IP, HTTP, Skype, 802.11
- **Internet standards**
  - RFC: Request for comments
  - IETF: Internet Engineering Task Force
  - Internet standards are developed by the IETF.
  - The IETF standards documents are called RFCs.



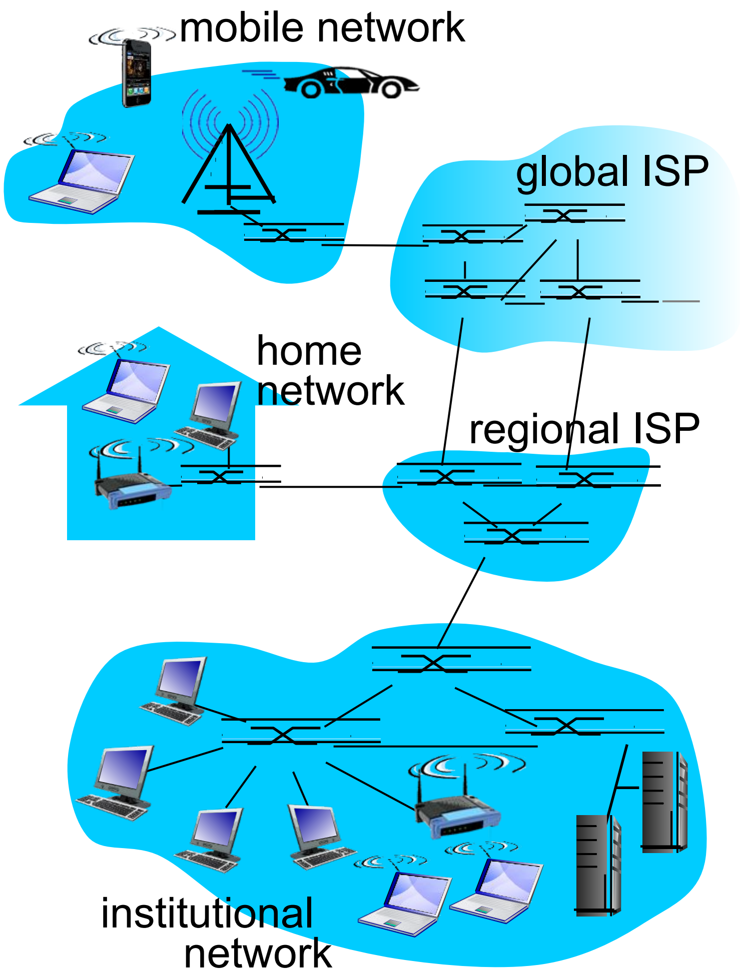
A “service view” of the Internet

- *Infrastructure that provides services to applications:*
  - E.g- Web, VoIP, email, games, e-commerce, social networking, etc.
  - Applications are said to be distributed
  - Applications involve multiple end systems that exchange data with each other.

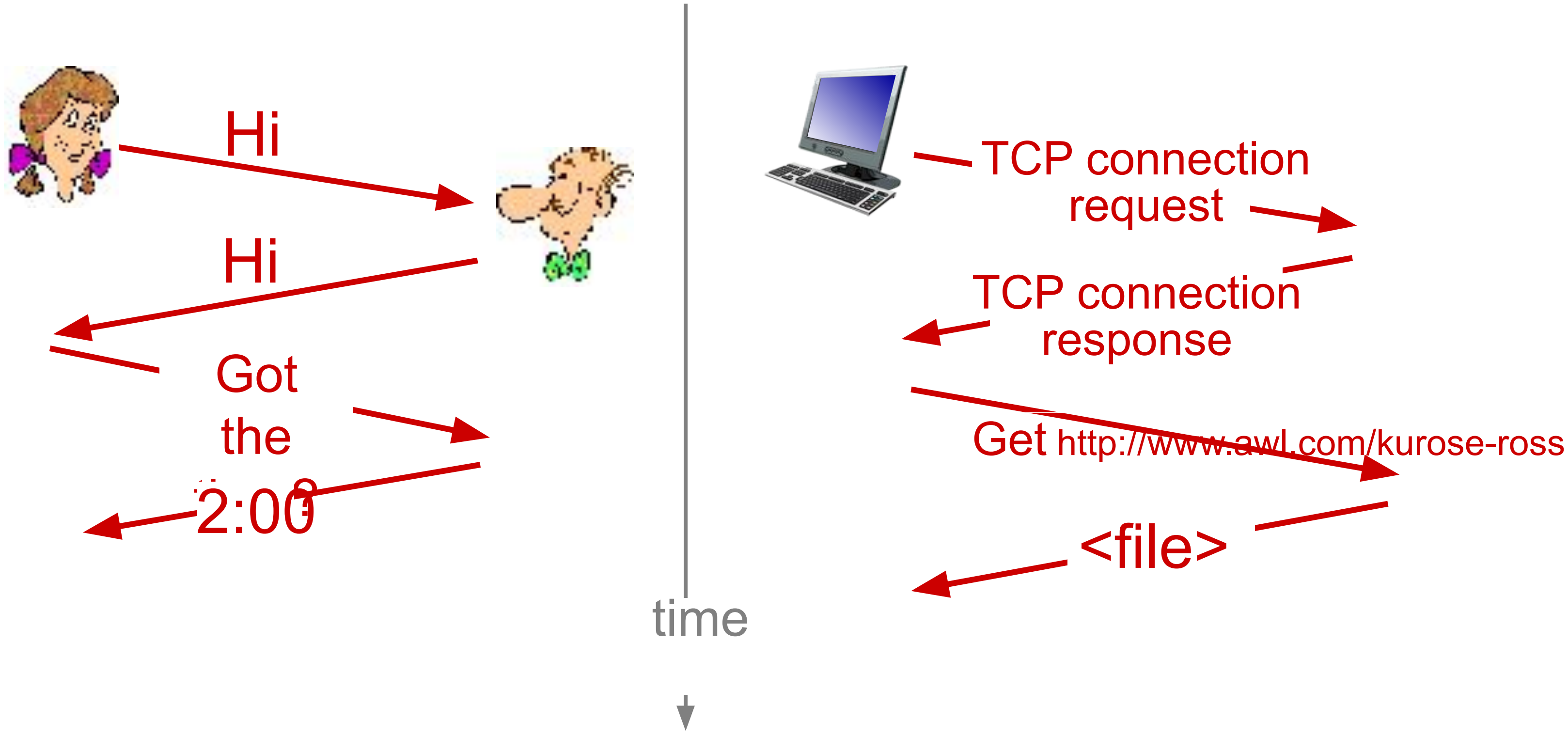


- Internet applications run on end systems- not in the network core.

- *provides programming interface to applications*
  - allows sending and receiving application programs to “connect” to Internet



Protocols


Hi
Hi
Got
the
2:00
TCP connection request
TCP connection response
Get http://www.awl.com/kurose-ross
<file>
time

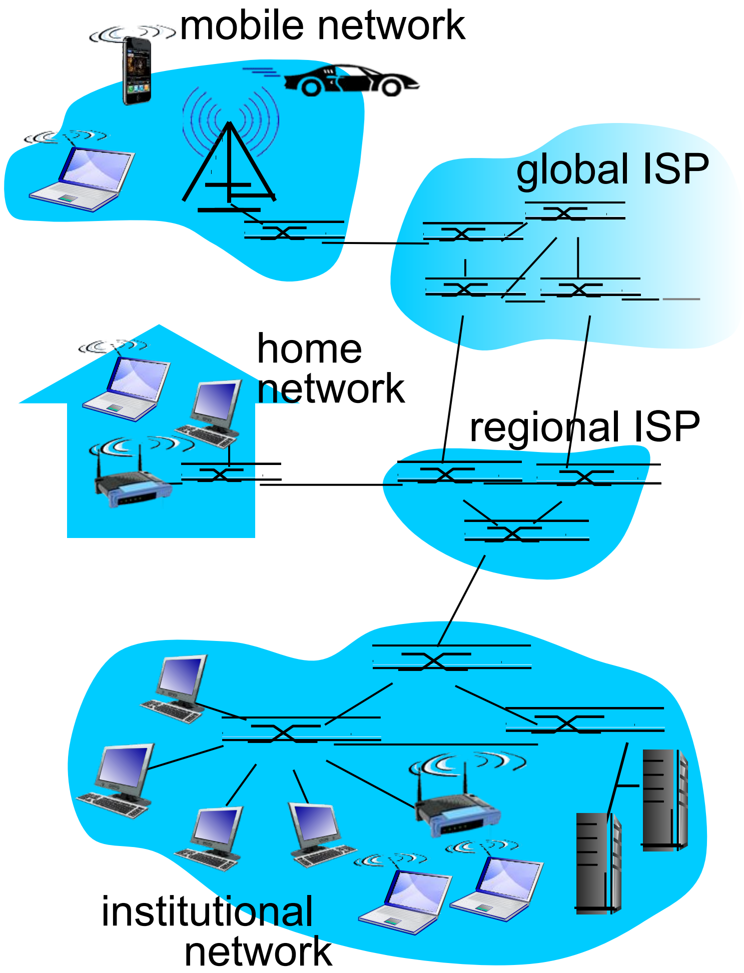
A protocol defines the format and order of messages exchanged between two or more communicating entities, as well as the actions taken on the transmission and/or receipt of a message or other event.

Network Structure

- *network edge:*
  - hosts: clients and servers

- *access networks, physical media:*
- wired, wireless communication links
- *network core:*
  - interconnected routers
  - network of networks

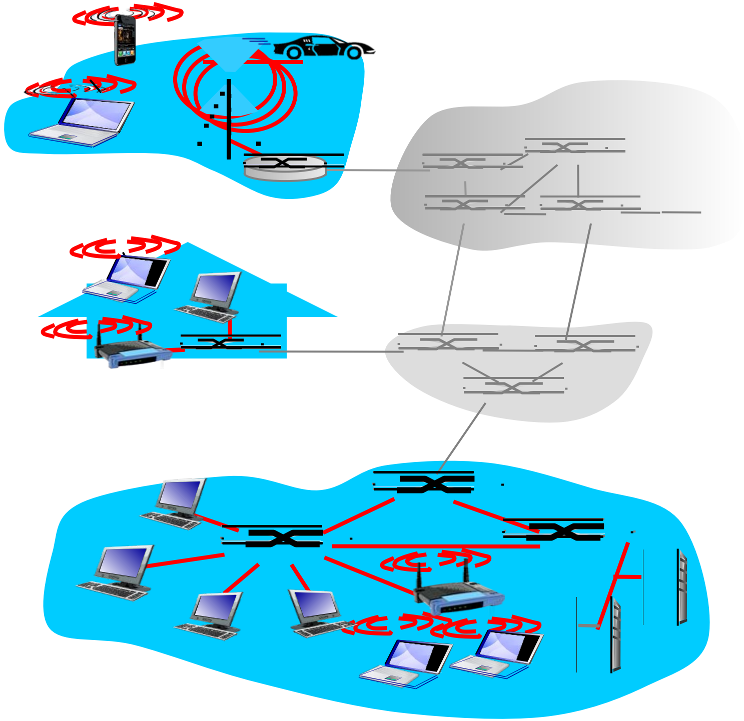
-



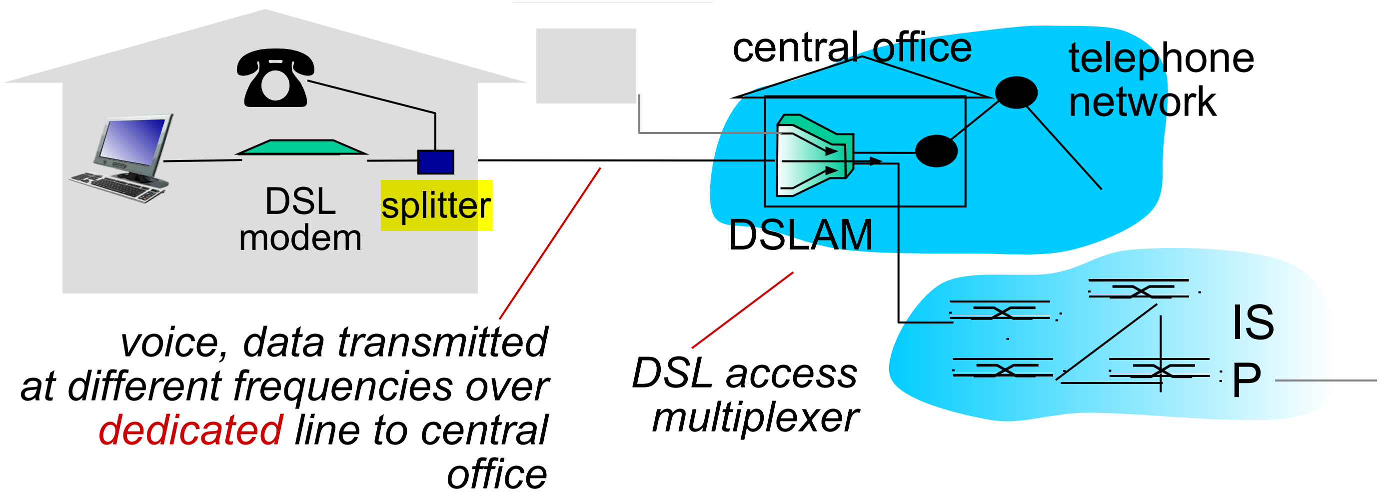
Access networks and physical media

Q: How to connect end systems to edge router?

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

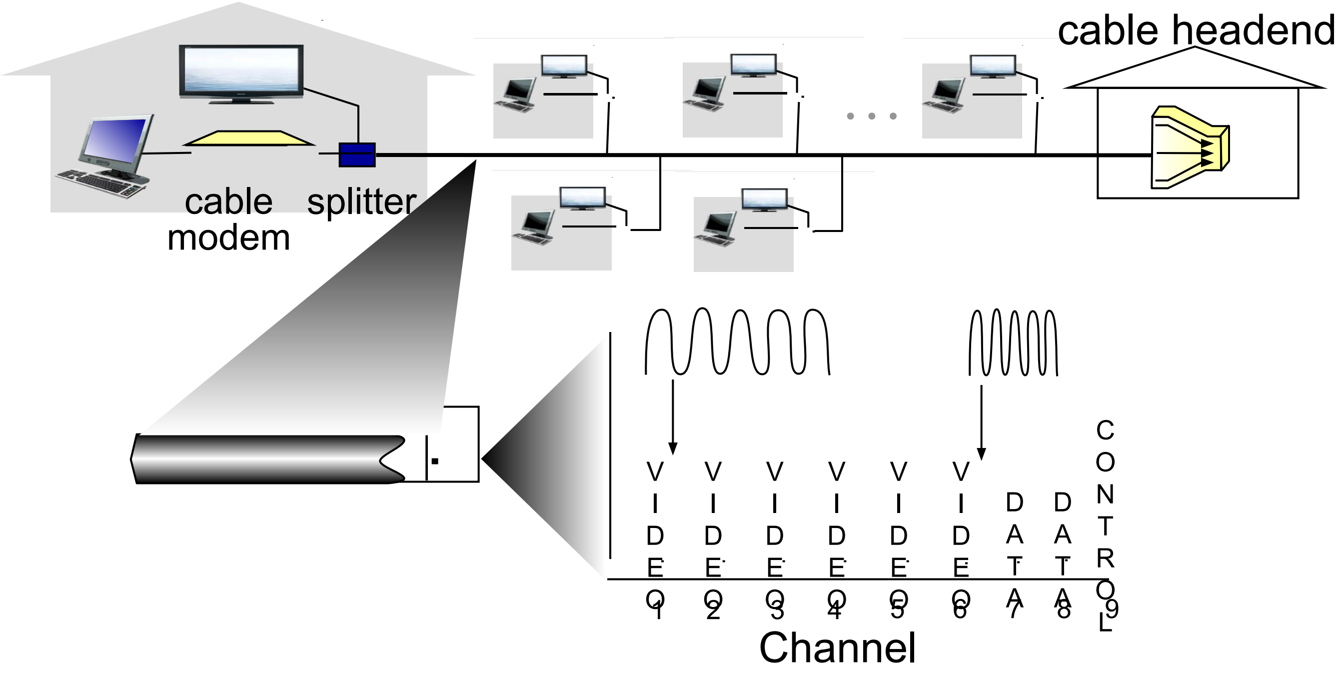


Access net: Digital Subscriber Line (DSL)



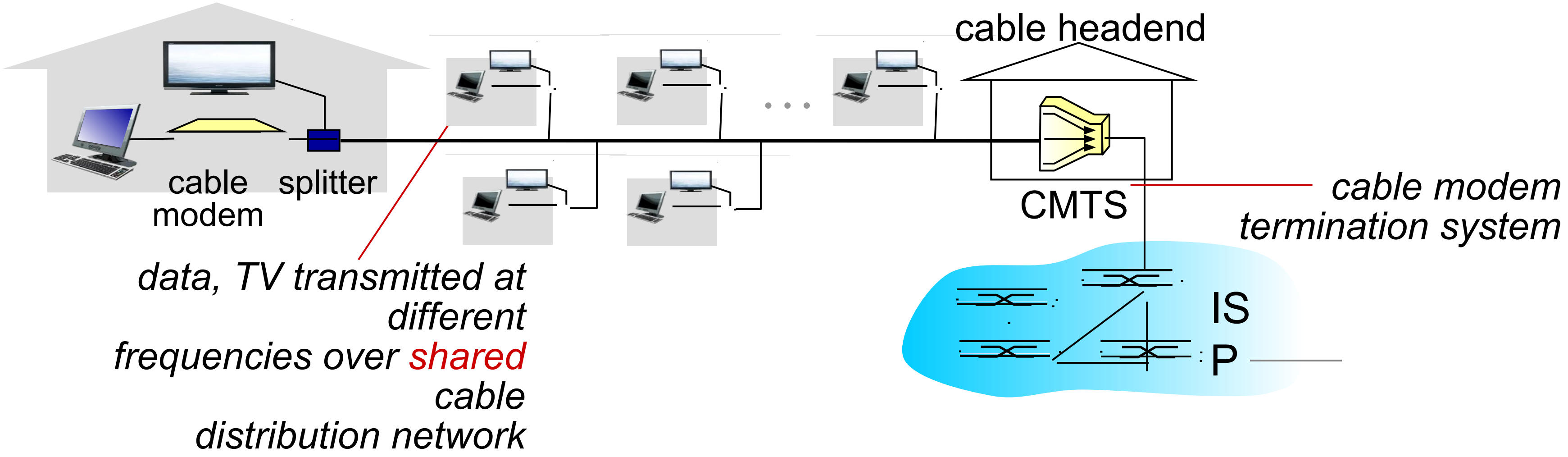
- use *existing* telephone line to central office DSLAM
  - data over DSL phone line goes to Internet
  - voice over DSL phone line goes to telephone net
- < 2.5 Mbps upstream transmission rate (typically < 1 Mbps)
- < 24 Mbps downstream transmission rate (typically < 10 Mbps)

Access net: cable network



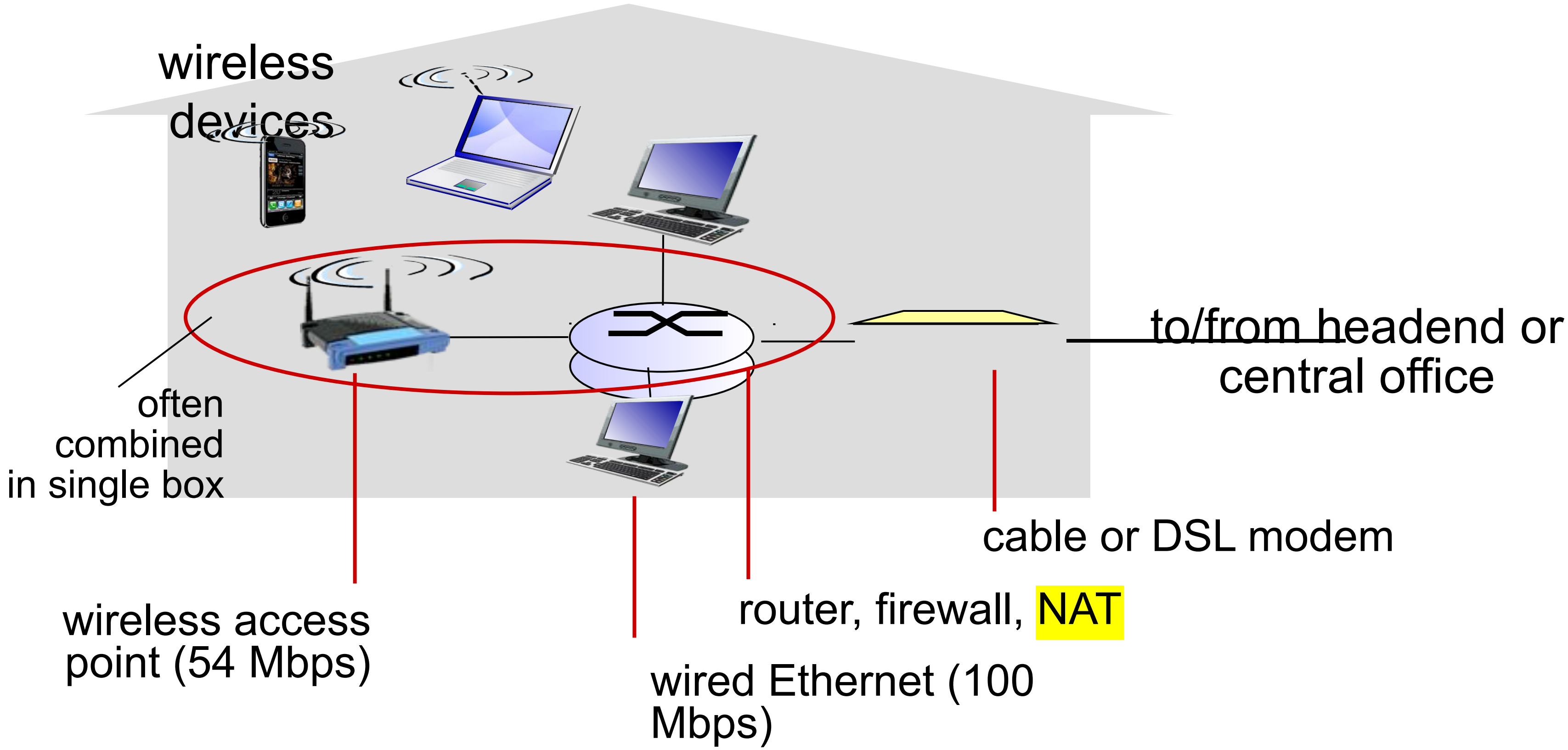
frequency division multiplexing: different channels transmitted in different frequency bands

Access net: cable network

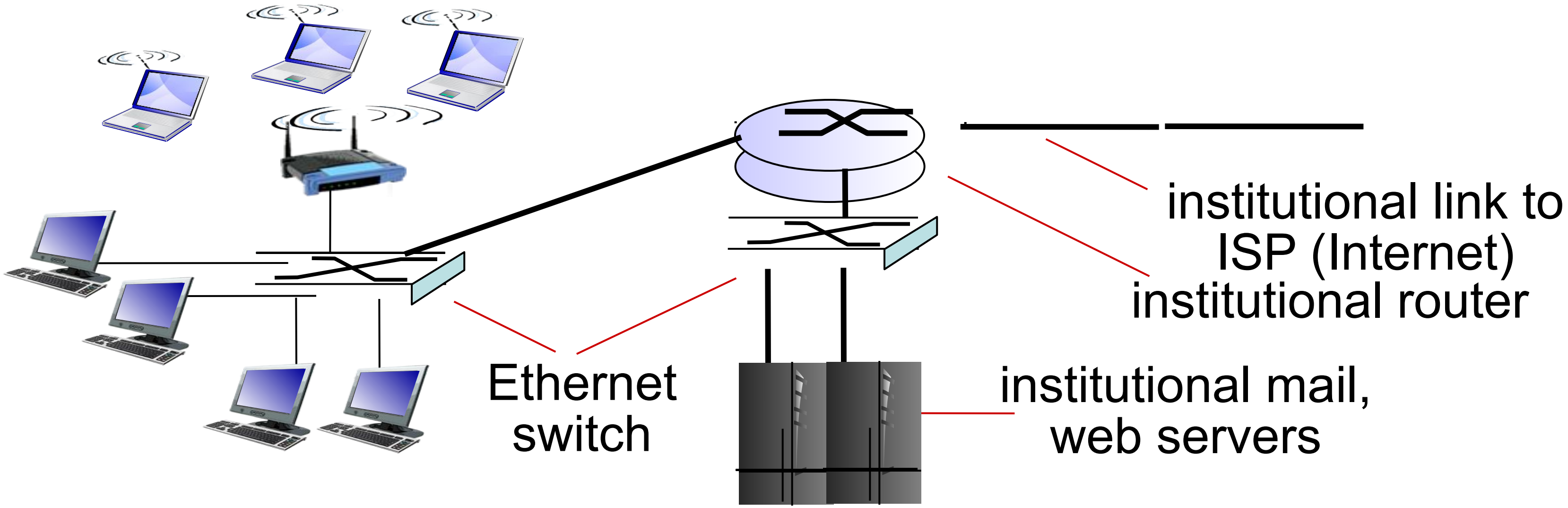


- HFC: hybrid fiber coax
  - asymmetric: upto 30 Mbps downstream transmission rate, 2 Mbps upstream transmission rate
- network of cable, fiber attaches homes to ISP router
  - homes *share access network* to cable head end
  - unlike DSL, which has dedicated access to central office

Access net: home network



Enterprise access networks (Ethernet)



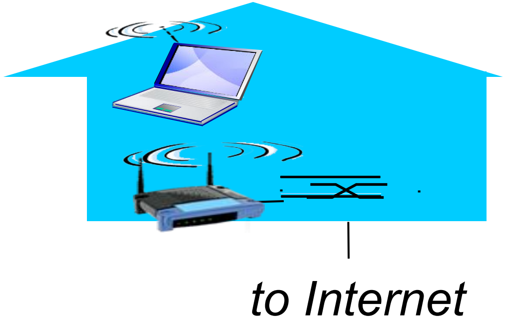
- typically used in companies, universities, etc
- 10 Mbps, 100Mbps, 1Gbps, 10Gbps transmission rates
- today, end systems typically connect into Ethernet switch

Wireless access networks

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

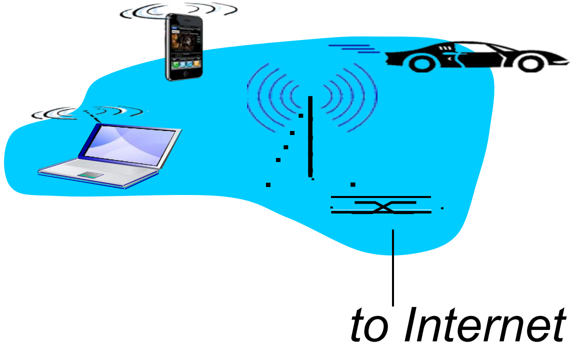
wireless LANs:

- 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



Physical Media

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

twisted pair (TP)

- two insulated copper wires
  - Category 5: 100 Mbps, 1 Gpbs Ethernet
  - Category 6: 10Gbps

Physical Media: coax and fiber

coaxial cable:

- two concentric copper conductors
- bidirectional
- broadband:
  - multiple channels on cable
  - HFC(High Fiber Coax)- Both fiber and coaxial cable

fiber optic cable:

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

Physical media: Radio

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

radio link types:

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

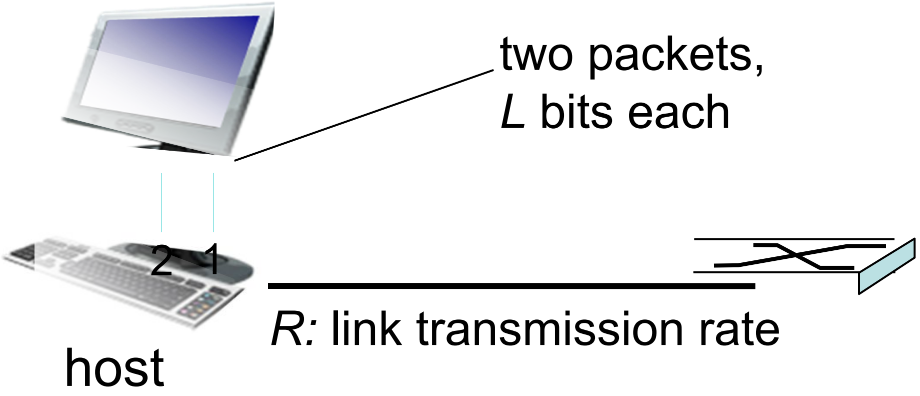
Network core

- packet switching, circuit switching, network structure

Host: sends *packets* of data

host sending function:

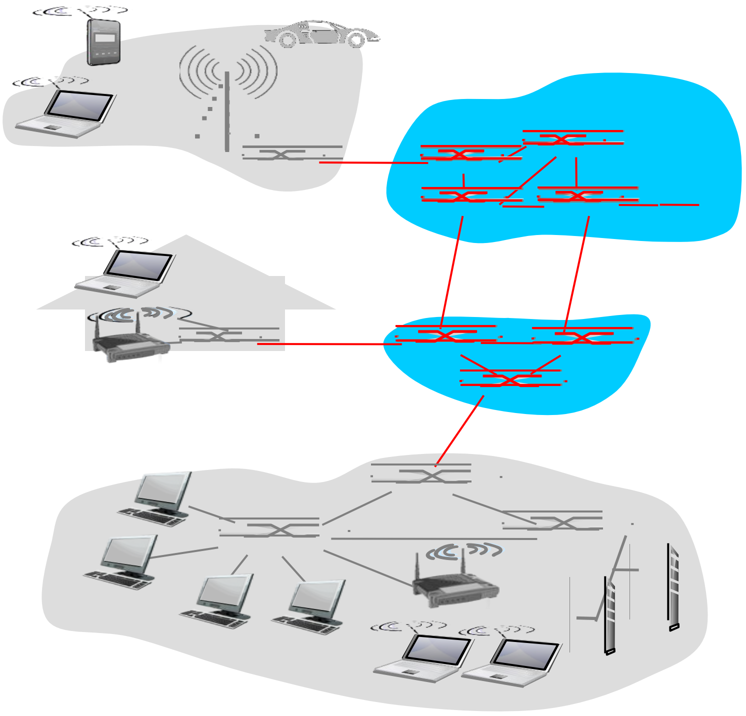
- takes application message
- breaks into smaller chunks, known as *packets*, of length *L* bits
- transmits packet into access network at *transmission rate R*
  - link transmission rate, aka link *capacity, aka link bandwidth*



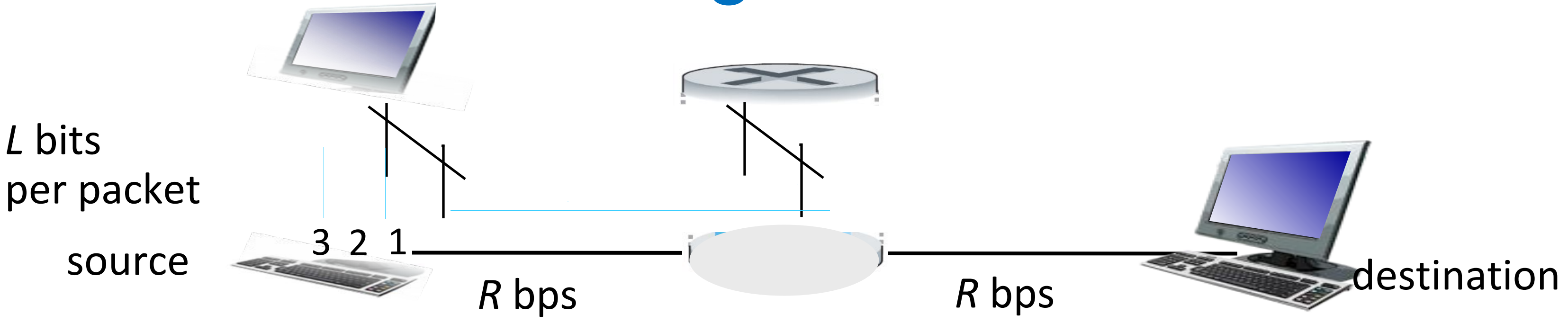
$$\text{packet transmission delay} = \text{time needed to transmit } L\text{-bit packet into link} = \frac{L\ \text{(bits)}}{R\ \text{(bits/sec)}}$$

The Network Core

- mesh of interconnected routers
- packet-switching: hosts break application-layer messages into *packets*
  - forward packets from one router to the next, across links on path from source to destination
  - each packet transmitted at full link capacity



Packet Switching: Store-and-forward

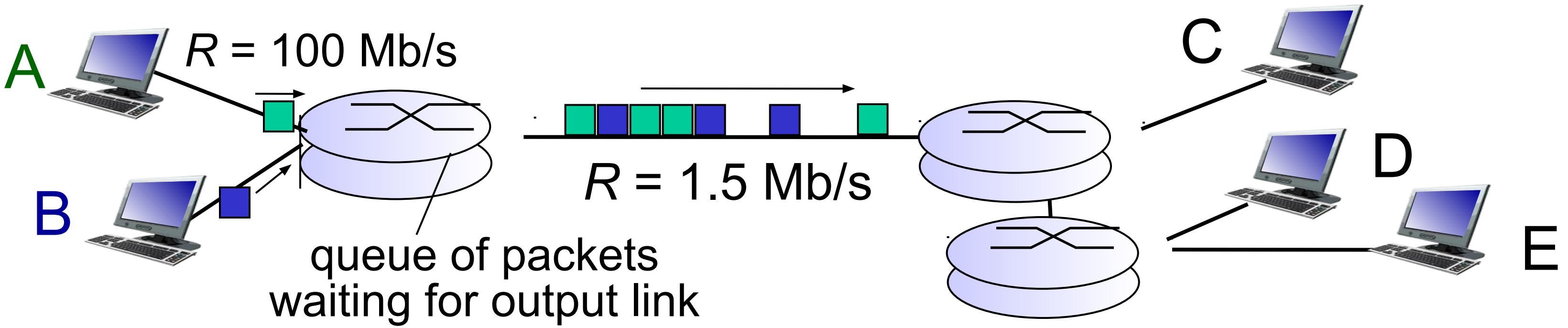


- takes L/R seconds to transmit (push out) L-bit packet into link at R bps
- *store and forward:* entire packet must arrive at router before it can be transmitted on next link
- end-end delay = $2L/R$ (assuming zero propagation delay)

one-hop numerical example:

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

Packet Switching: queueing delay, loss



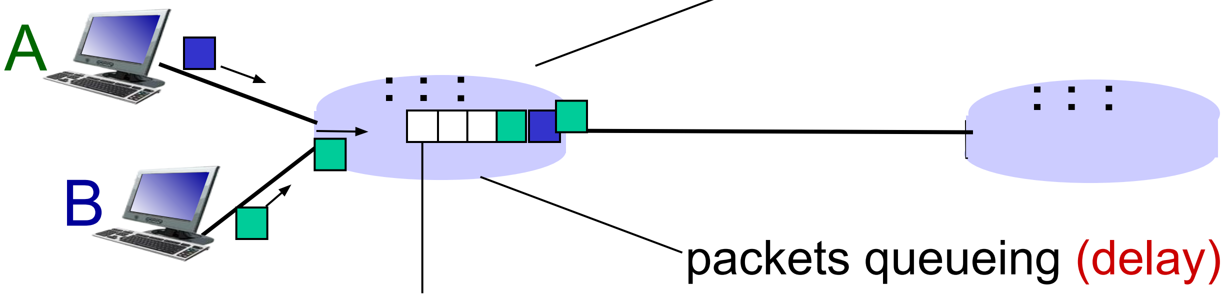
queuing and loss:

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

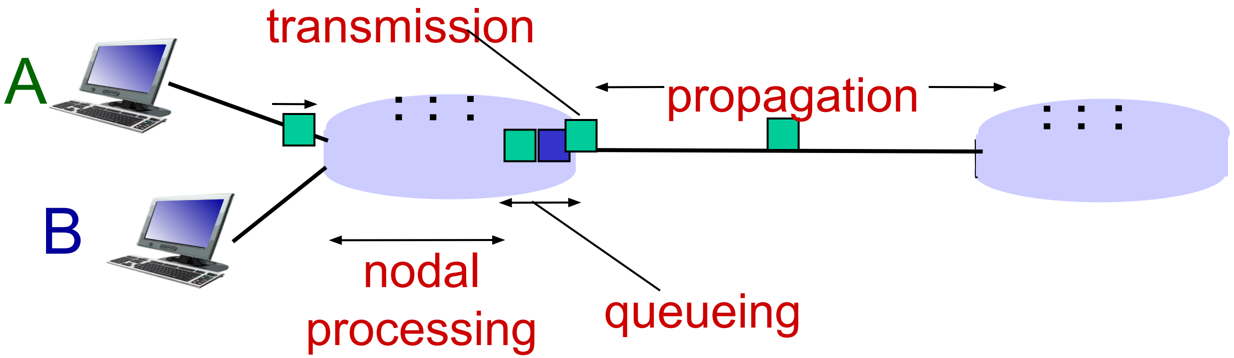
How do loss and delay occur?

packets *queue* in router buffers

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



Sources of packet delay

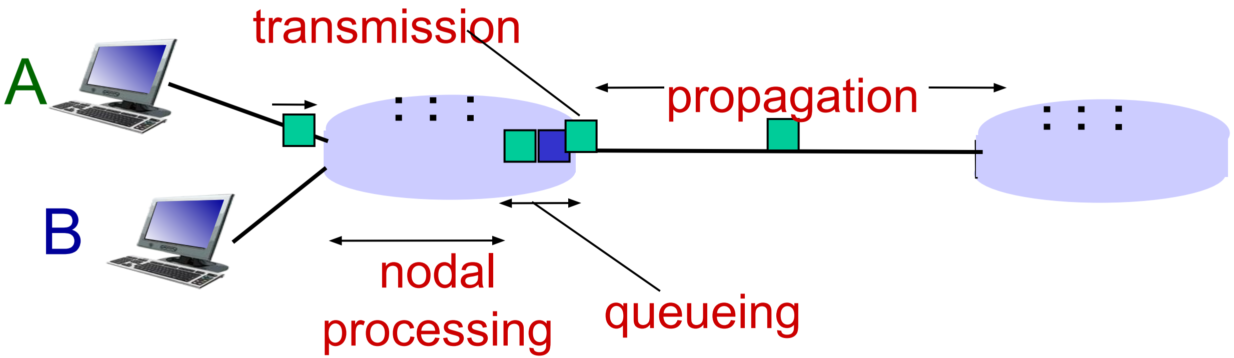


$$d_{nodal} = d_{proc} + d_{queue} + d_{trans} + d_{prop}$$

d_{proc}: nodal processing
- check bit errors
- determine output link
- typically < msec

d_{queue}: queueing delay
- time waiting at output link for transmission
- depends on congestion level of router

Sources of packet delay



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

d_{trans}: transmission delay:

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

d_{trans} and d_{prop} *very different*

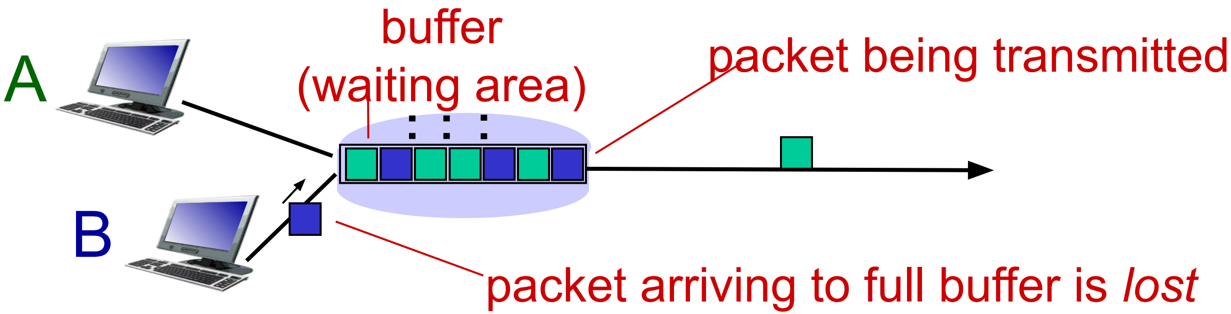
d_{prop}: propagation delay:

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

* Check out the Java applet for an interactive animation on trans vs. prop delay

Packet loss

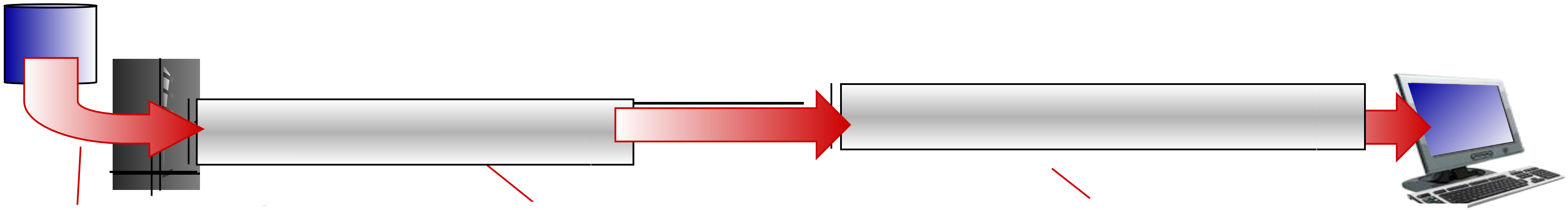
- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



* Check out the Java applet for an interactive animation on queuing and loss

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



server sends bits (fluid) into pipe

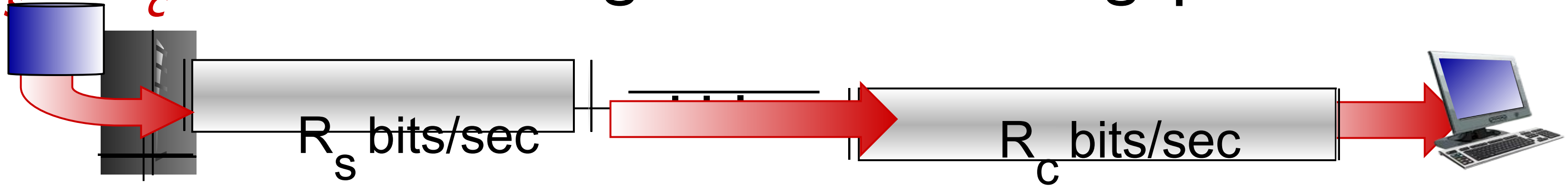
pipe that can carry fluid at rate R_s bits/sec)

pipe that can carry fluid at rate R_c bits/sec)

Throughput

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

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



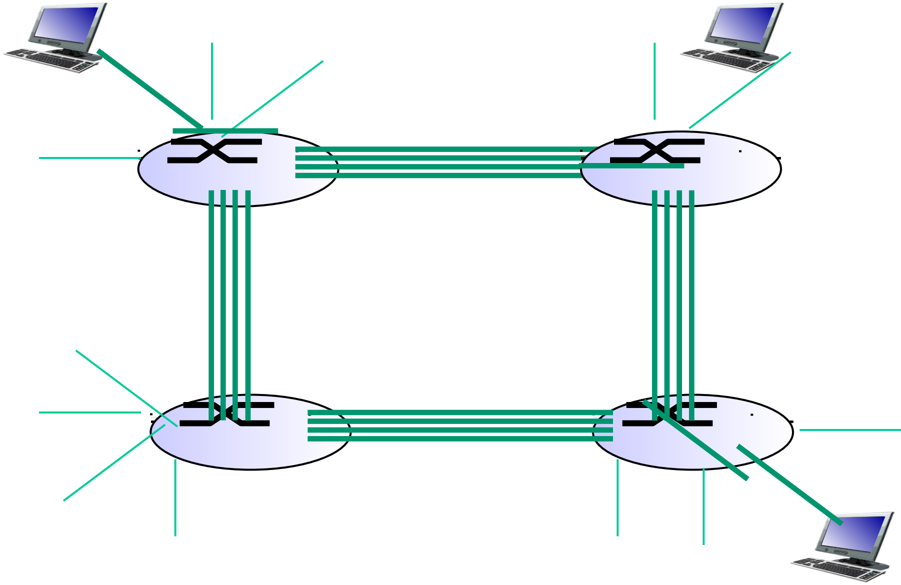
bottleneck link

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

Alternative core: circuit switching

end-end resources allocated to, reserved for "call" between source & dest:

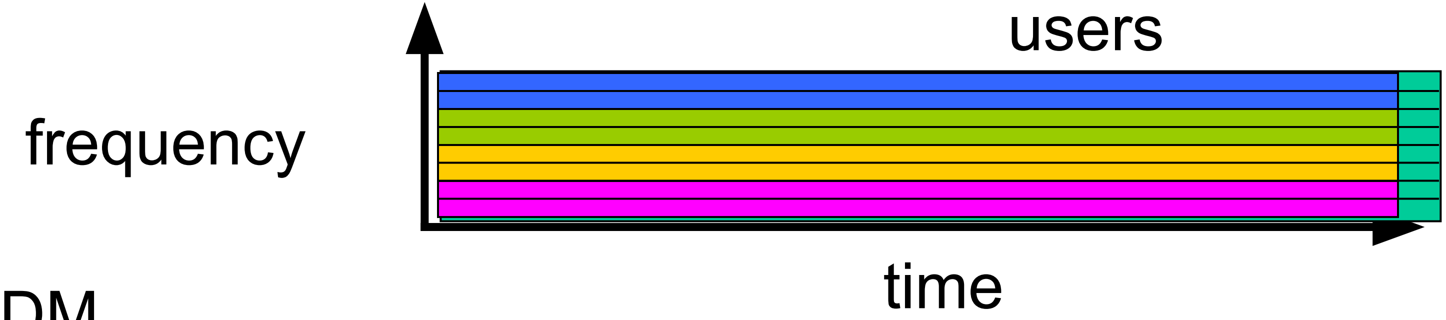
- In diagram, each link has four circuits.
  - call gets 2nd circuit in top link and 1st circuit in right link.
- dedicated resources: no sharing
  - circuit-like (guaranteed) performance
- circuit segment idle if not used by call *(no sharing)*
- Commonly used in traditional telephone networks



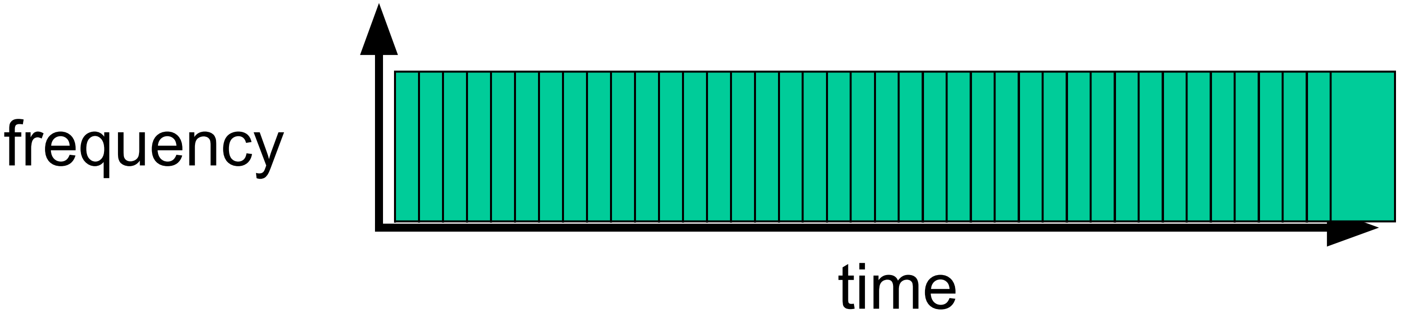
Circuit switching: FDM versus TDM

Example:
4 users

FDM



TDM



Packet Switching

- ❖Resources are not reserved.
- ❖A session’s message use the resources on demand.
- ❖Message may have to wait for access to a communication link.

Advantages over circuit-switching

- ❖Better sharing of bandwidth
- ❖Simple
- ❖Efficient
- ❖Less costly

Packet switching versus circuit switching

is packet switching a "slam dunk winner?"

- great for bursty data
  - resource sharing
  - simpler, no call setup
- excessive congestion possible: 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

Internet structure: network of networks

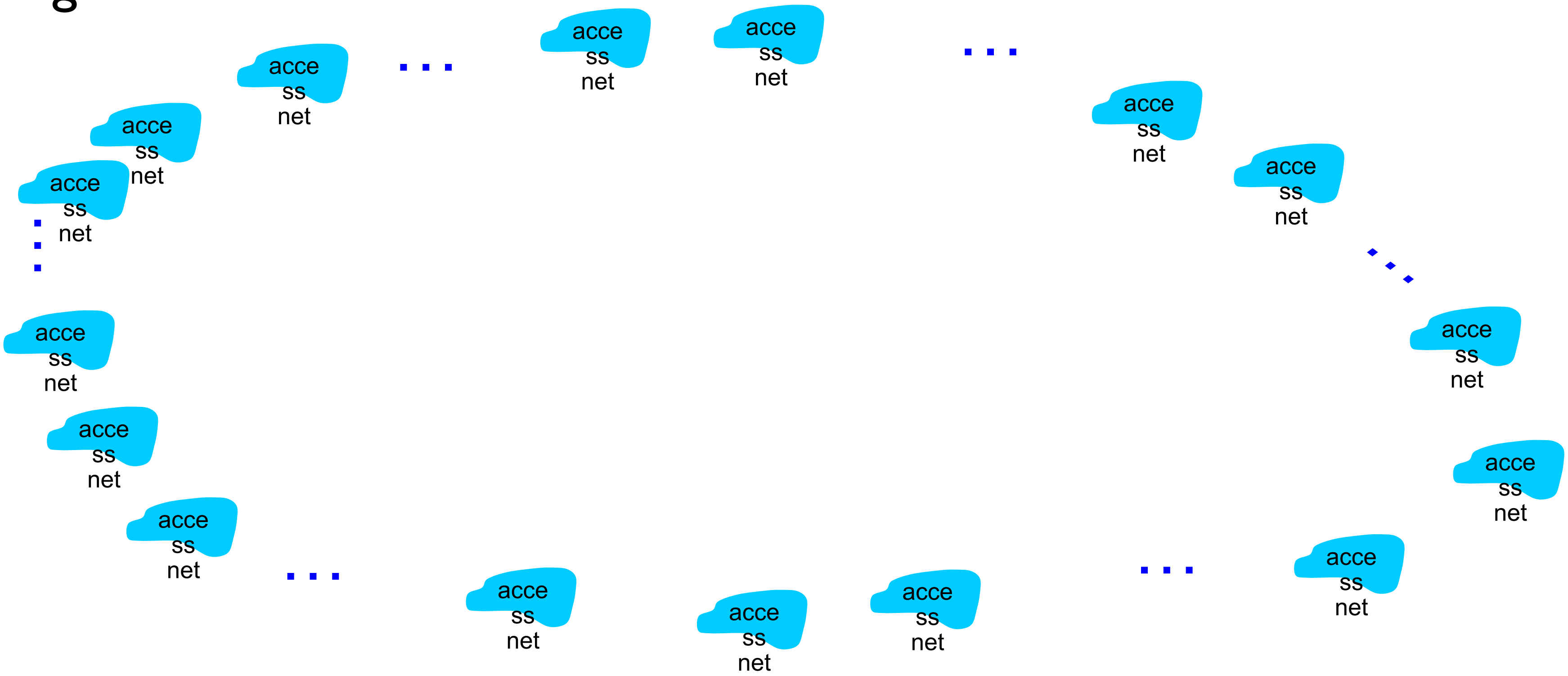
- End systems connect to Internet via access ISPs (Internet Service Providers)
  - Residential, company and university ISPs

- 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

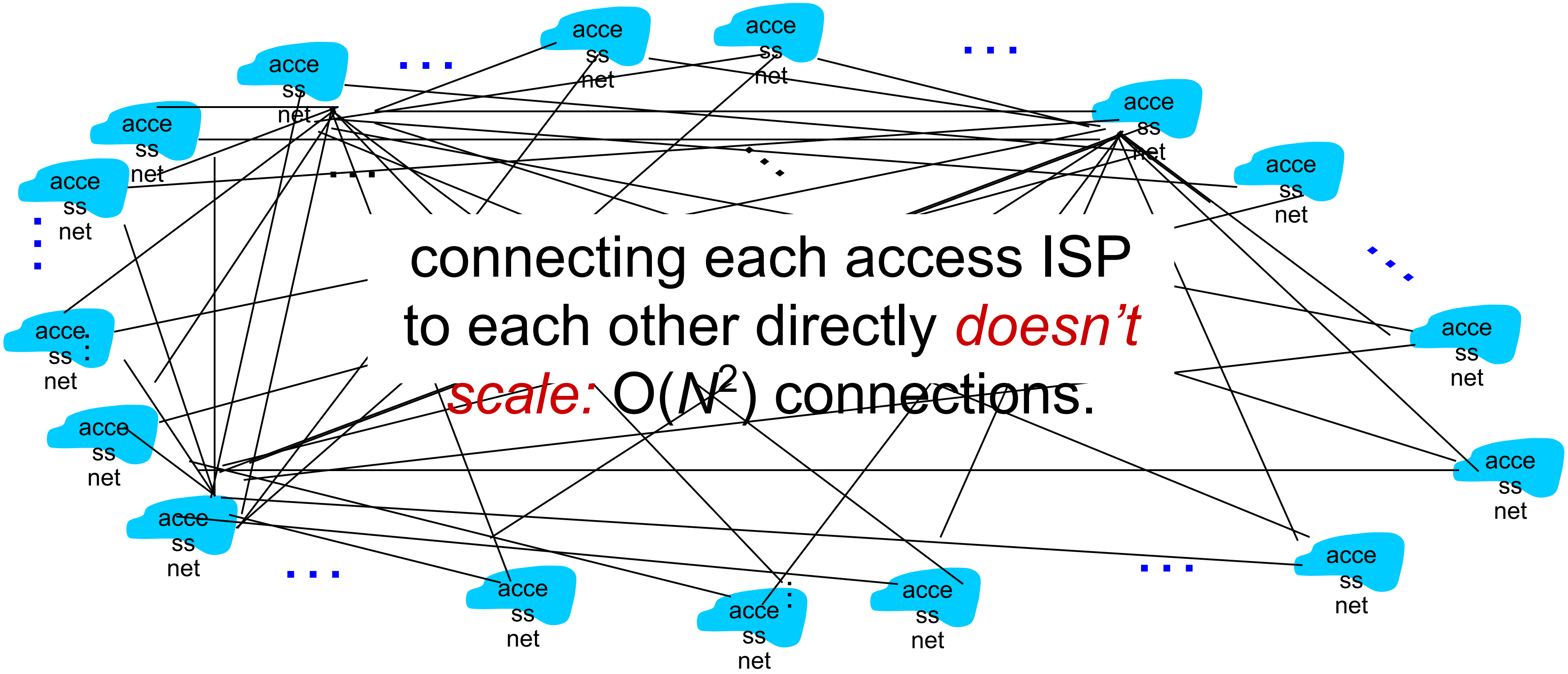
Internet structure: network of networks

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



Internet structure: network of networks

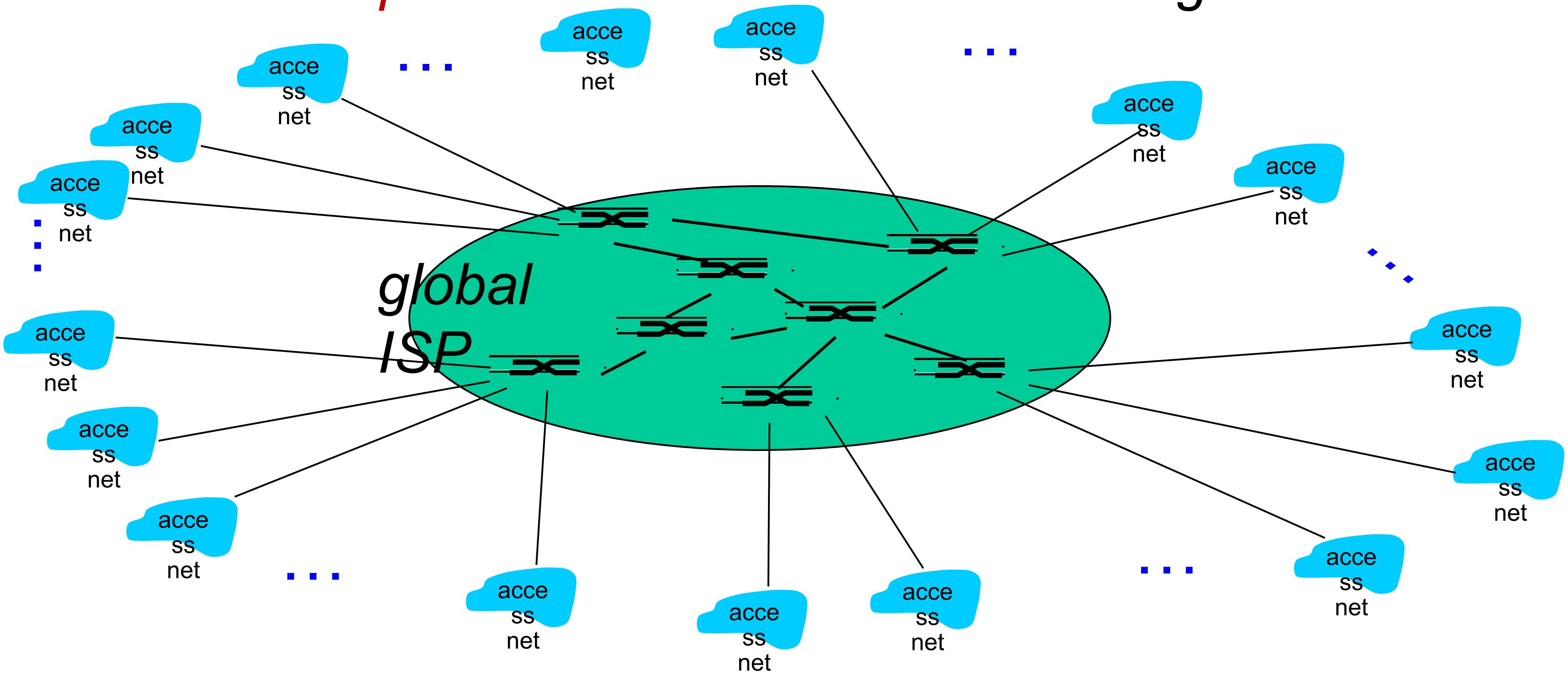
Option: connect each access ISP to every other access ISP?



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

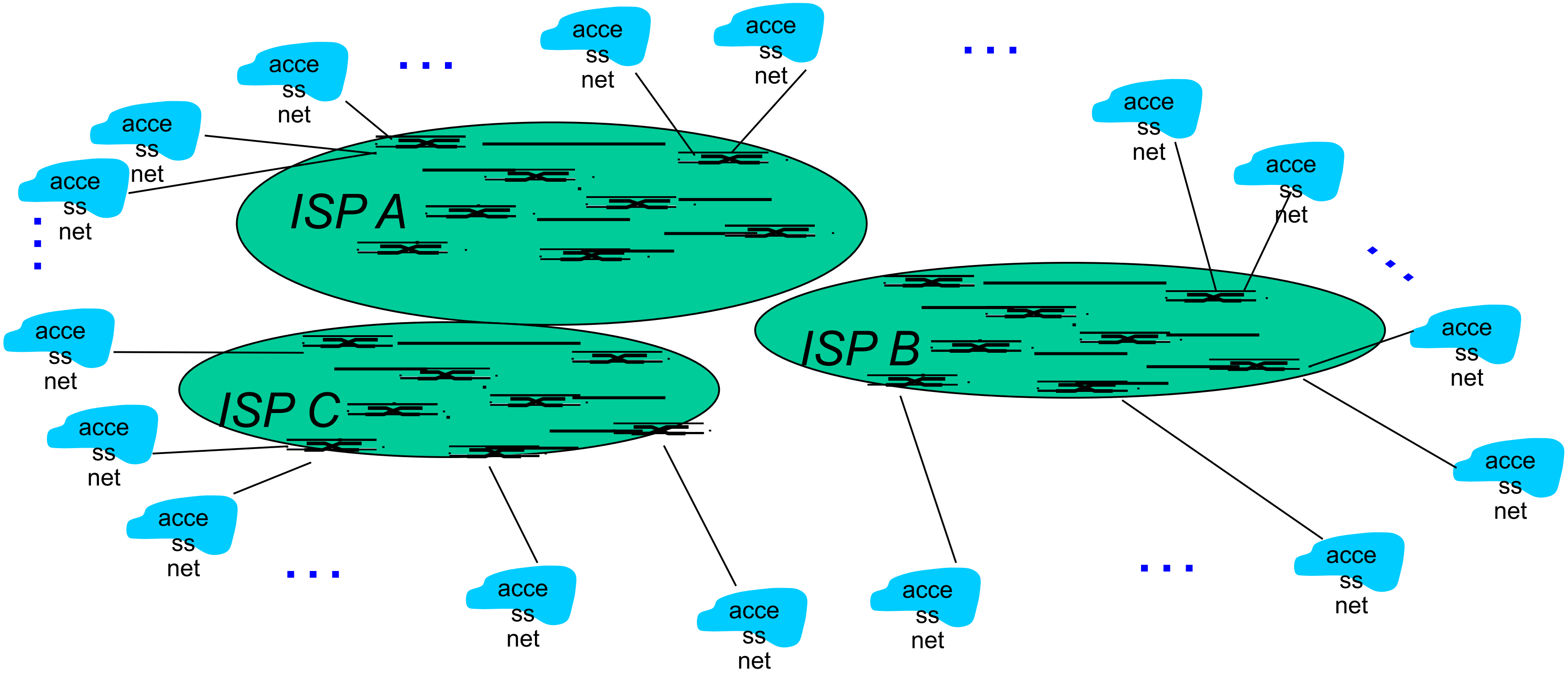
Internet structure: network of networks

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



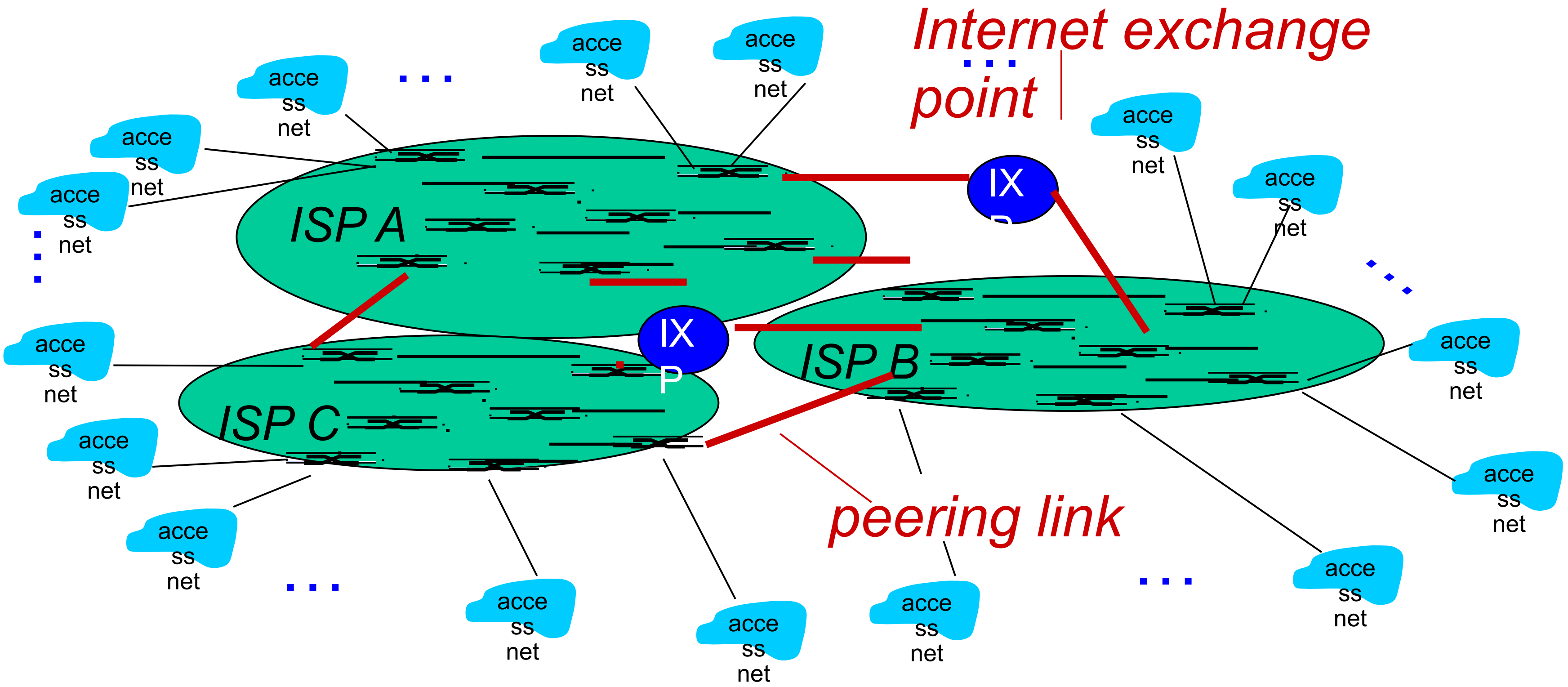
Internet structure: network of networks

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



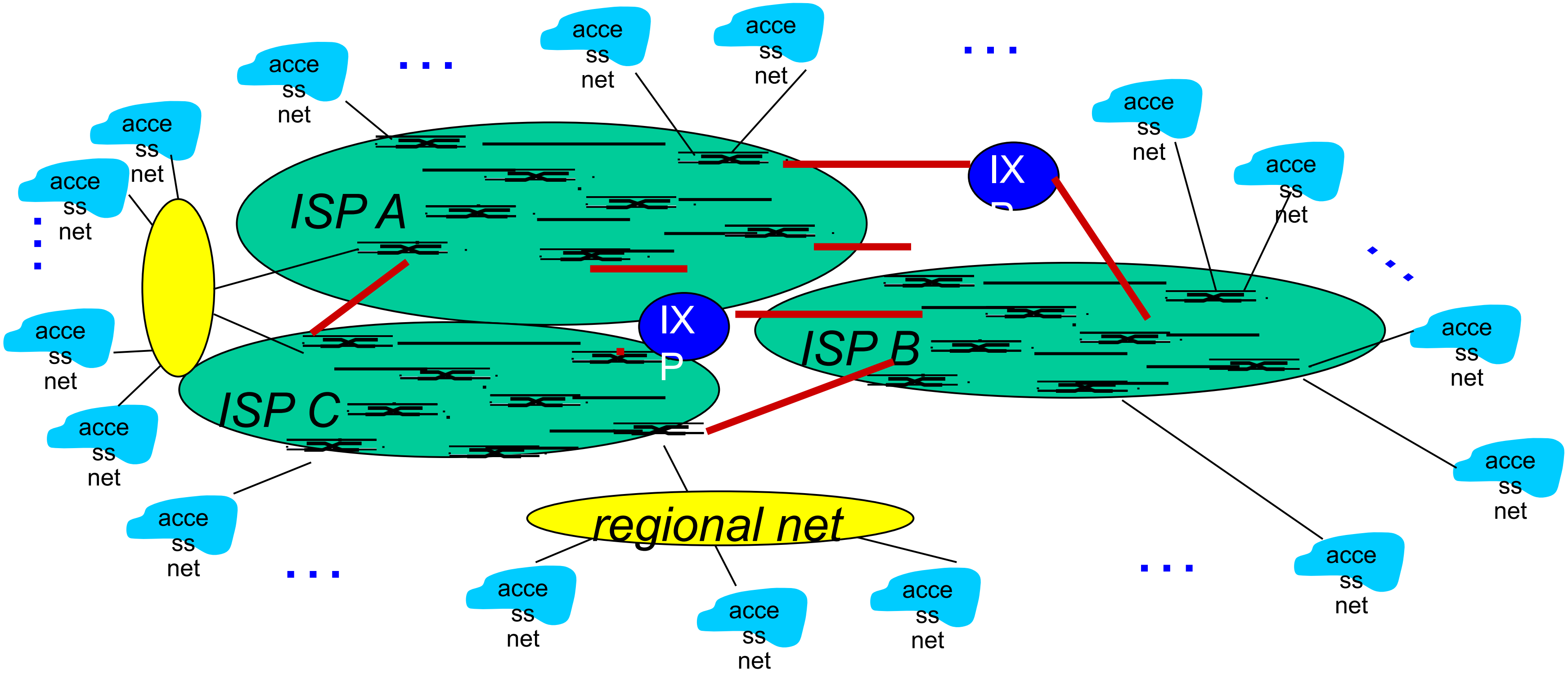
Internet structure: network of networks

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



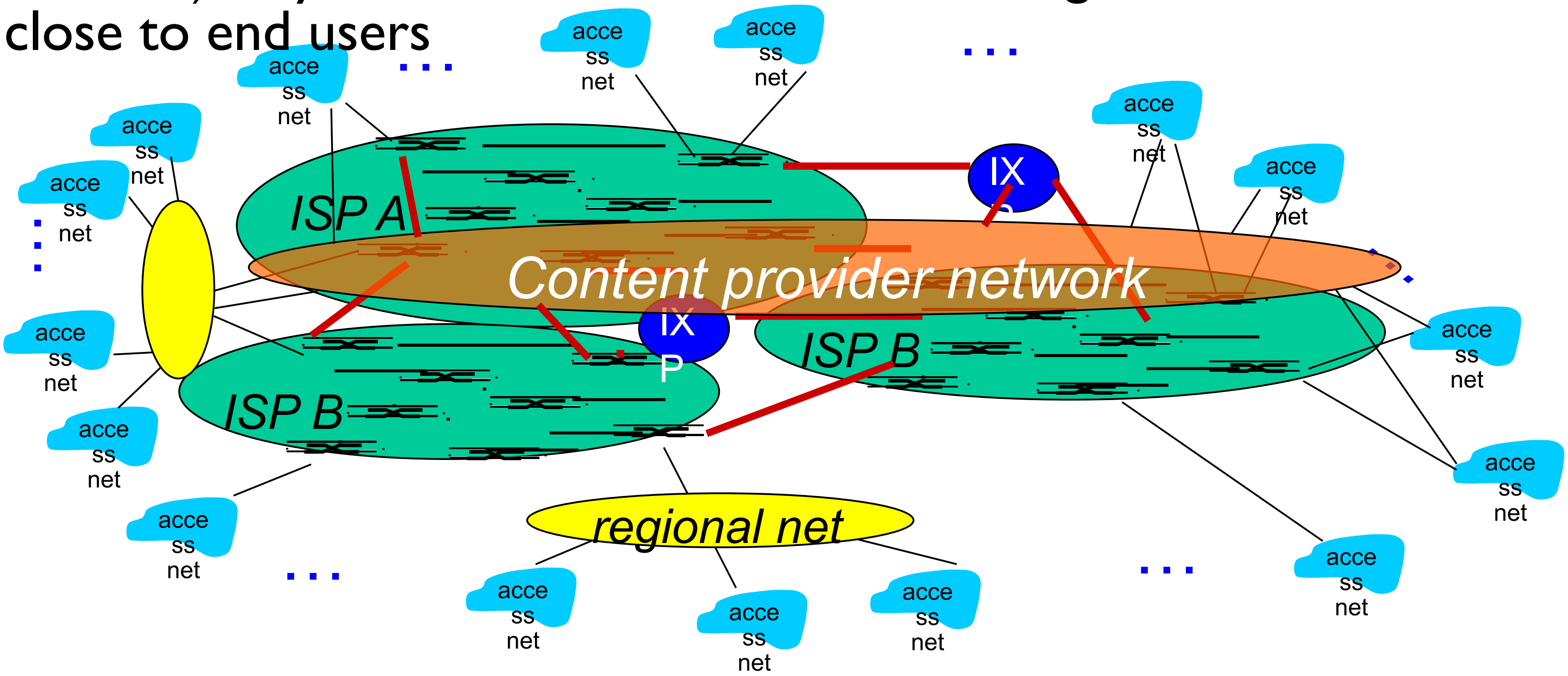
Internet structure: network of networks

… 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



Protocol "layers"

Networks are complex, with 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?

- **Layering** – Structured discussion of system components.

- A layer is a structured component which tells us what are the protocols that a packet has to follow in order to get transmitted over a medium.

Why layering?

- To understand how a message is transmitted from source to destination.
- A lot of steps are involved.
- A lot of different protocols.
- In order to have better understanding, layering can be used.
- Maintained by ISO.

Advantages:

- Provides a structured way to discuss system components.
- Provides modularity.

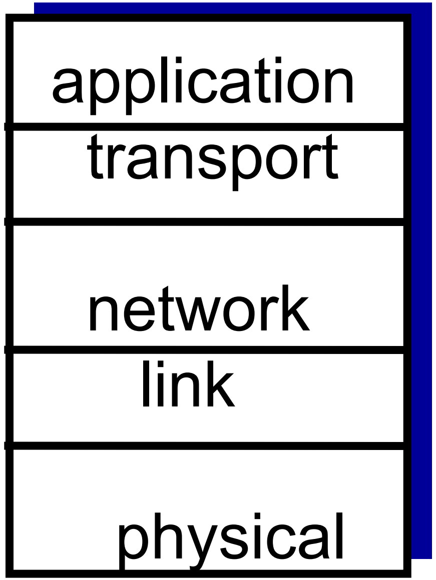
Drawbacks:

- One layer may duplicate lower layer functionality.
- One layer may need information from other layer, which violates the goal of separation of layers.

- **Internet Protocol (IP) stack**
- **OSI reference model**

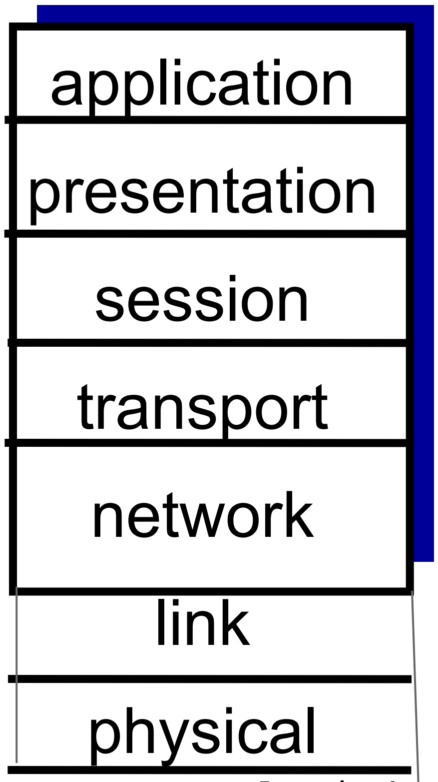
Internet protocol stack

- *application:* supporting network applications
  - FTP, SMTP, HTTP
- *transport:* process-process data transfer
  - TCP, UDP
- *network:* routing of datagrams from source to destination
  - IP, routing protocols
- *link:* data transfer between neighboring network elements
  - Ethernet, 802.11 (WiFi)
- *physical:* bits "on the wire"



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
  - needed?



Encapsulation