CMPT 371

Data Communications and Networking

Summer 2011

Course Information

- **CMPT371**

  Classes : Wed, 17:30-20:20, HC 1700

- **Instructor**

  Marjan Marzban

  - Email : mmarzban@cs.sfu.ca
  - Office hours : Wed, 16:00-17:00 at HC 2134

- **TA**

  Haiyang Wang

  - Email : hwa17@sfu.ca
  - Office hours : Tue, 14:00-15:00 at TASC1 9002 (Burnaby)

- Text Book:
  - Computer Networking: A Top Down Approach , 5th edition. Jim Kurose, Keith Ross
- Reference Books:
  - Data and Computer Communications, Eighth edition. William Stallings
  - Computer Networking, Forth edition. *Andrew S. Tanenbaum*

Course Goals

- Understanding the principles of networking
  - Top-Down approach
    - Application layer
    - Transport layer
    - Network layer
    - Link layer
  - Use Internet as an example

Grading

- Assignment-Projects
  - 4 assignments : 13% (4%- 3%-3%-3%)
  - 2 projects : 12 %

- Midterm : 25%
- Final : 50%

Chapter1 : Introduction

Roadmap

- What's the Internet?
- What's a protocol?
- Network edge
- Network core
- Performance: loss, delay, throughput
- Protocol layers, service models

Chapter1 : Introduction

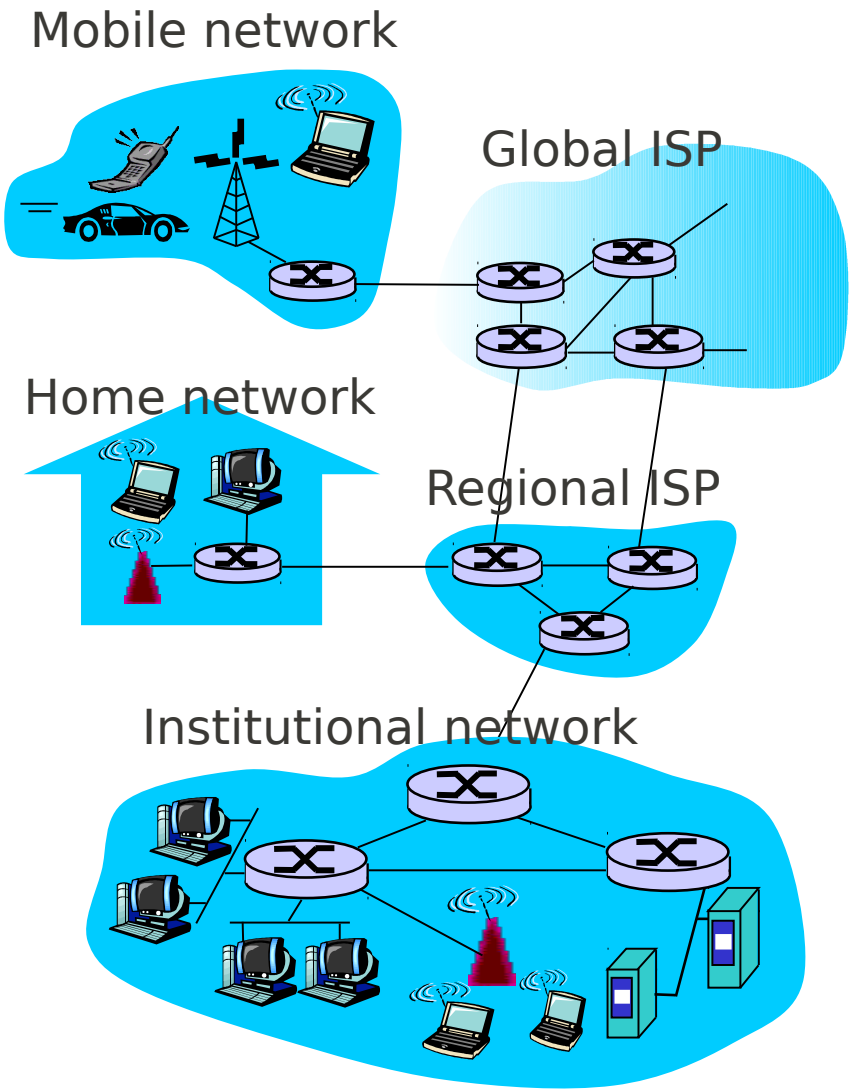
Roadmap

- What's the Internet?
- What's a protocol?
- Network edge
- Network core
- Performance: loss, delay, throughput
- Protocol layers, service models

- What’s the Internet?
  - Basic hardware and software components (nuts and bolts).
  - Infrastructures that provides services to distributed applications.

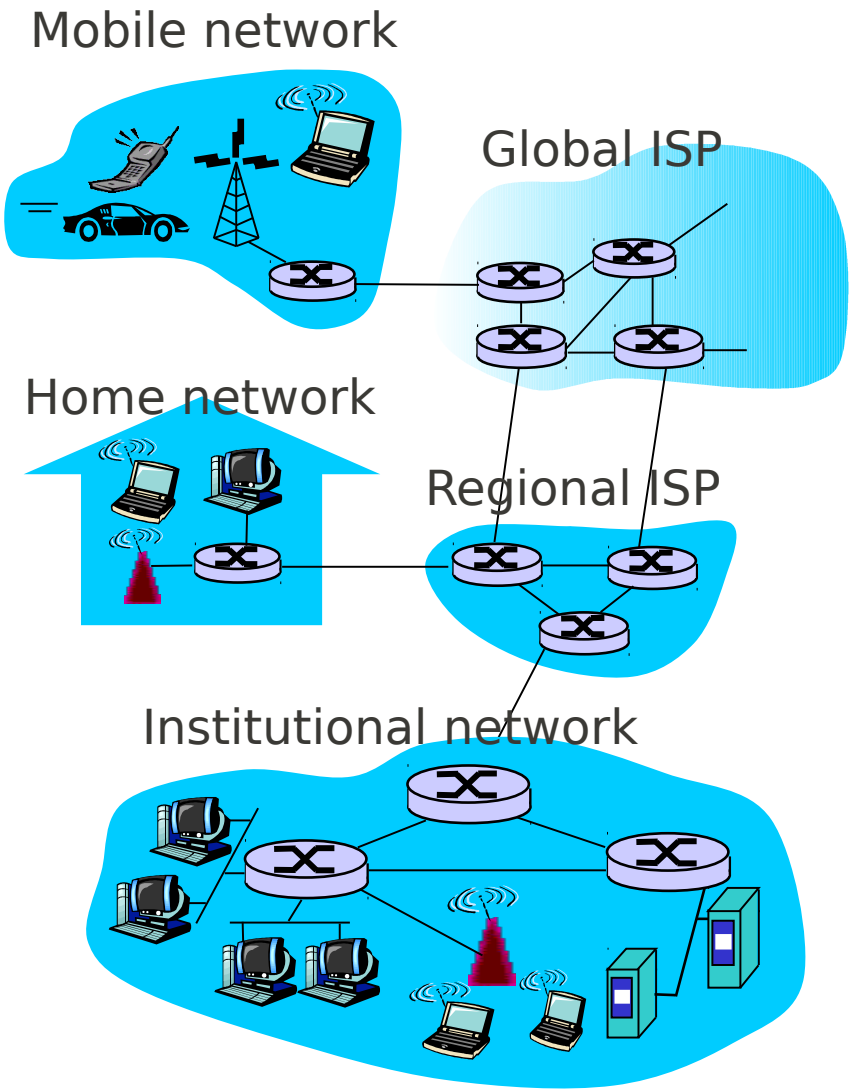
What is the Internet? (Nuts-and Bolts view)

- Hardware
  - Hosts or End systems
    - PC, Servers, Cellphones
    - Run network applications
  - Communication links
    - Coaxial cable, fiber optics, ...
    - Transmission rate
  - Routers



What is the Internet? (Nuts-and Bolts view)

- Software
  - Protocols control sending and receiving messages.
    - TCP, IP, HTTP,...



Internet Standards:

- IETF : Internet Engineering Task Force
- RFC : Request For Comments

What is the Internet? (Service view)

- Internet is and infrastructure that provides services to distributed applications such as e-mail, Internet radio, Web,...

What is a protocol?

human protocols:

- “what’s the time?”
- “I have a question”

In our human protocol there are specific msgs sent, and specific actions taken when we received, reply msgs or other events

What is a protocol?

human protocols:

- “what’s the time?”
- “I have a question”

In our human protocol there are specific msgs sent, and specific actions taken when we received, reply msgs or other events

Network protocols:

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

What is a protocol?

human protocols:

- “what’s the time?”
- “I have a question”

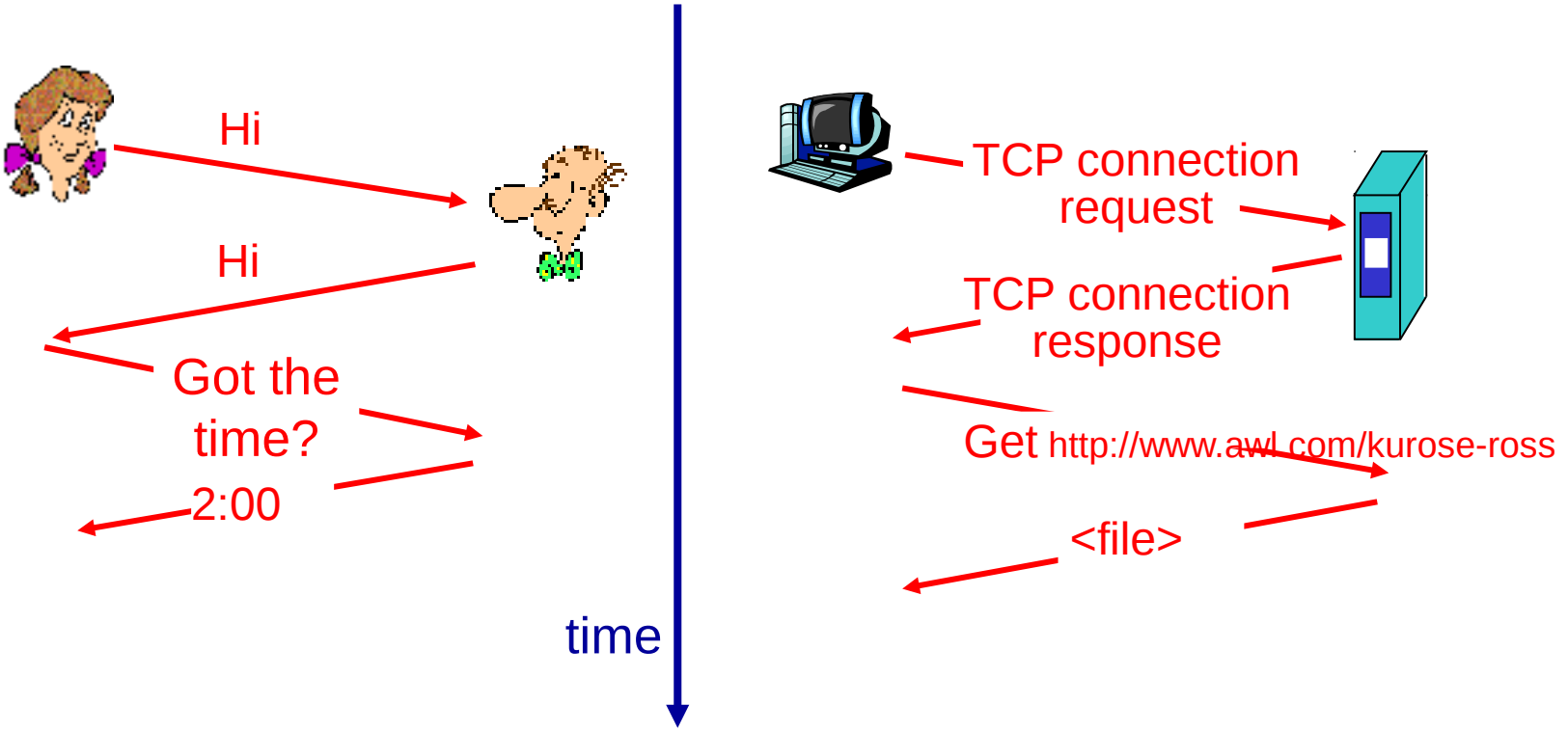
In our human protocol there are specific msgs sent, and specific actions taken when we received, reply msgs or other events

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 is a protocol?



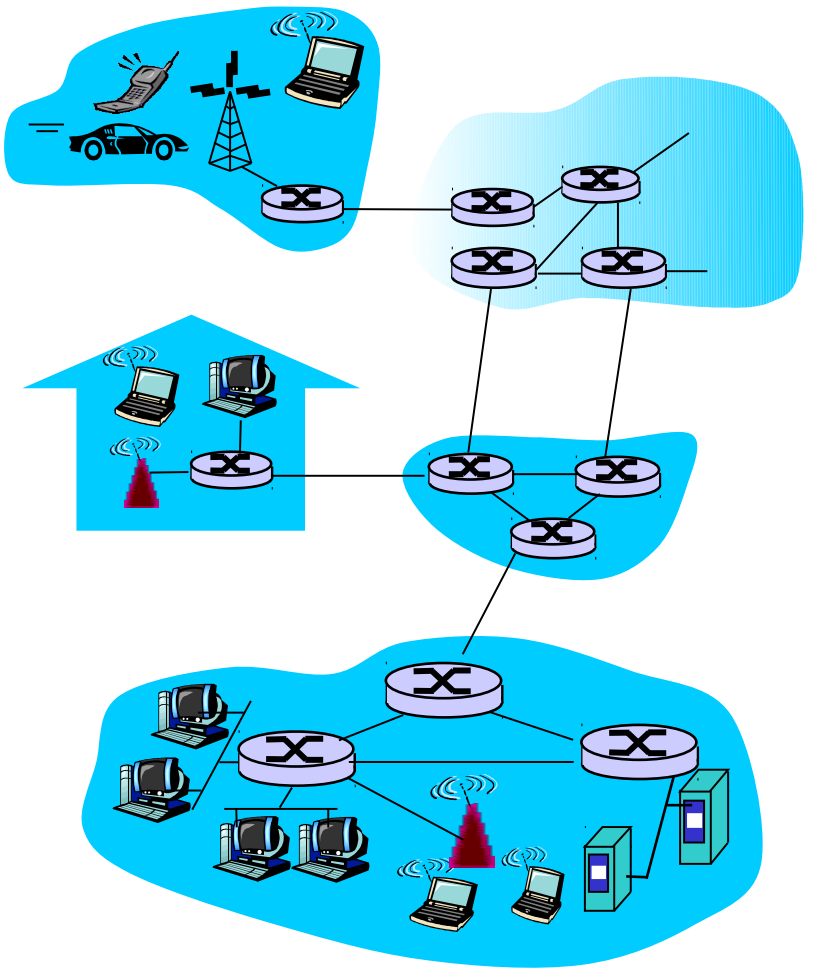
Chapter1 : Introduction

Overview:

- What's the Internet?
- What's a protocol?
- Network edge:
  - Hosts
  - Access net
  - Physical media
- Network core
- Performance: loss, delay, throughput
- Protocol layers, service models

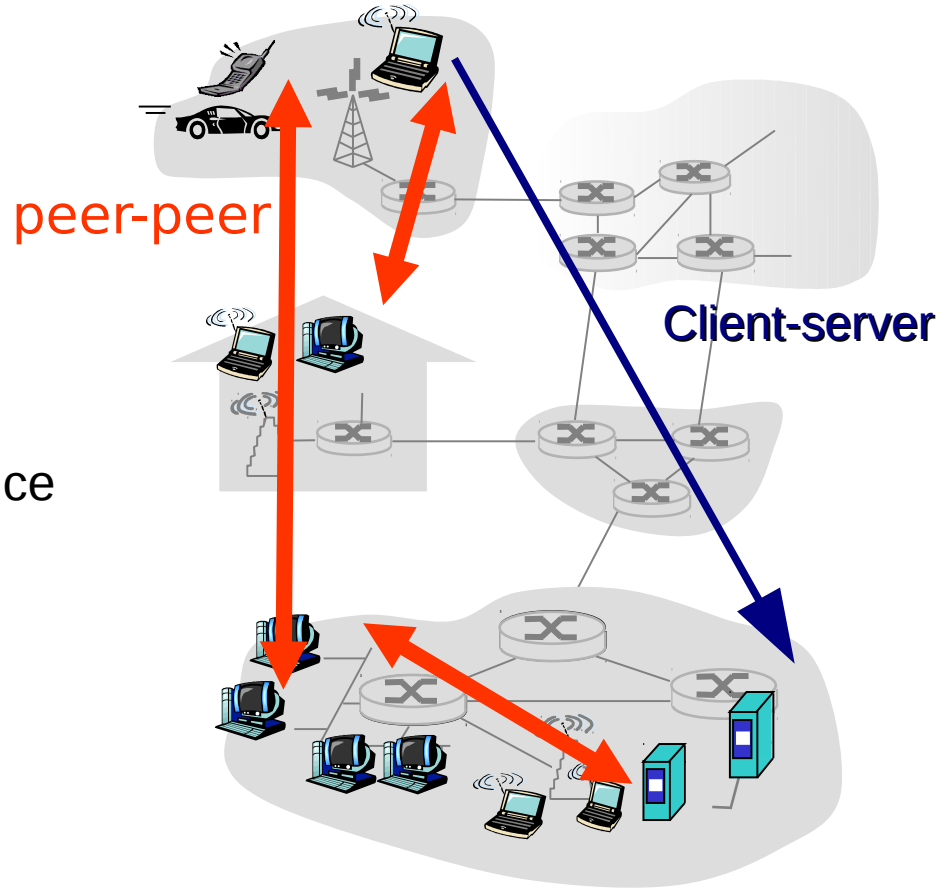
A closer look at network structure:

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



The network edge : Hosts

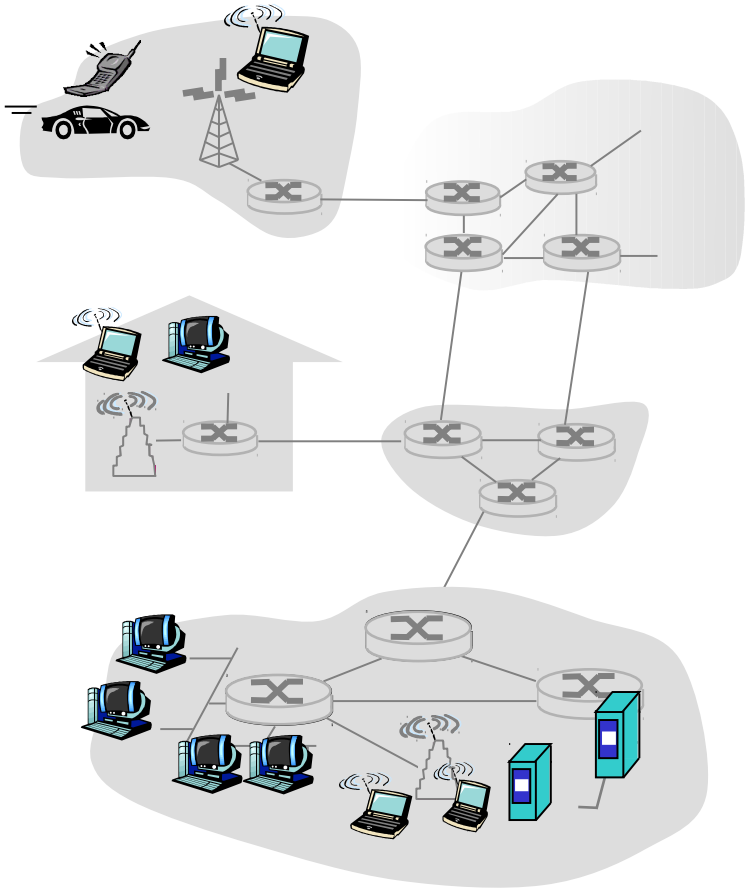
- End systems (hosts):
  - run application programs
  - e.g. Web, email
  - at “edge of network”
- Client-Server model
  - client host requests, receives service from always-on server
  - e.g. Web browser/server; email client/server
- Peer-peer model
  - minimal (or no) use of dedicated servers
  - e.g.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



Network Edge: Access network

Q: How to connect end systems to edge router?

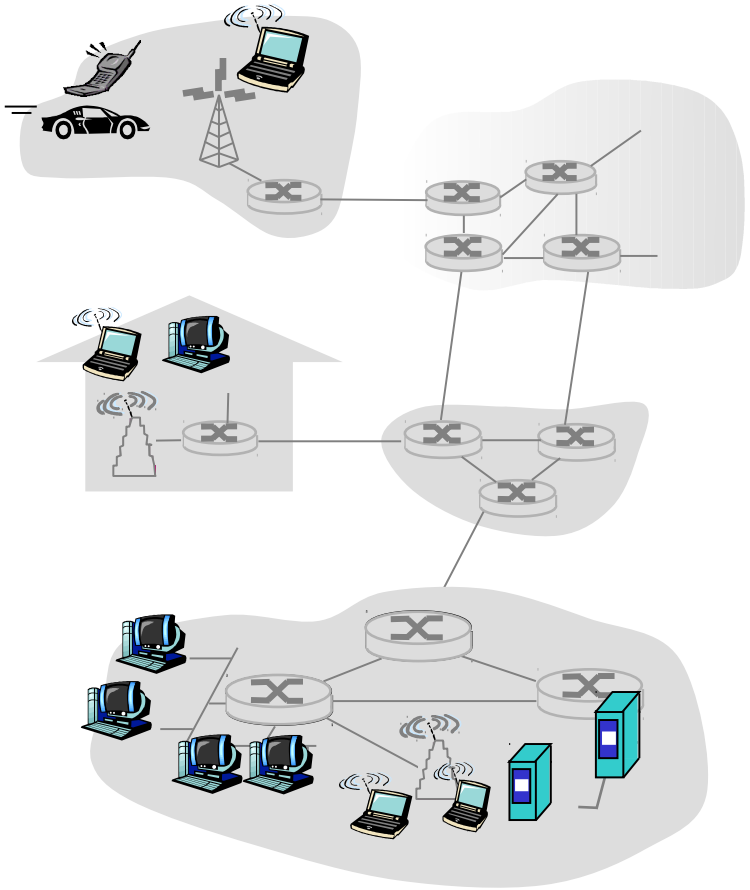
- residential access nets
- institutional access networks (school, company)
- mobile access networks



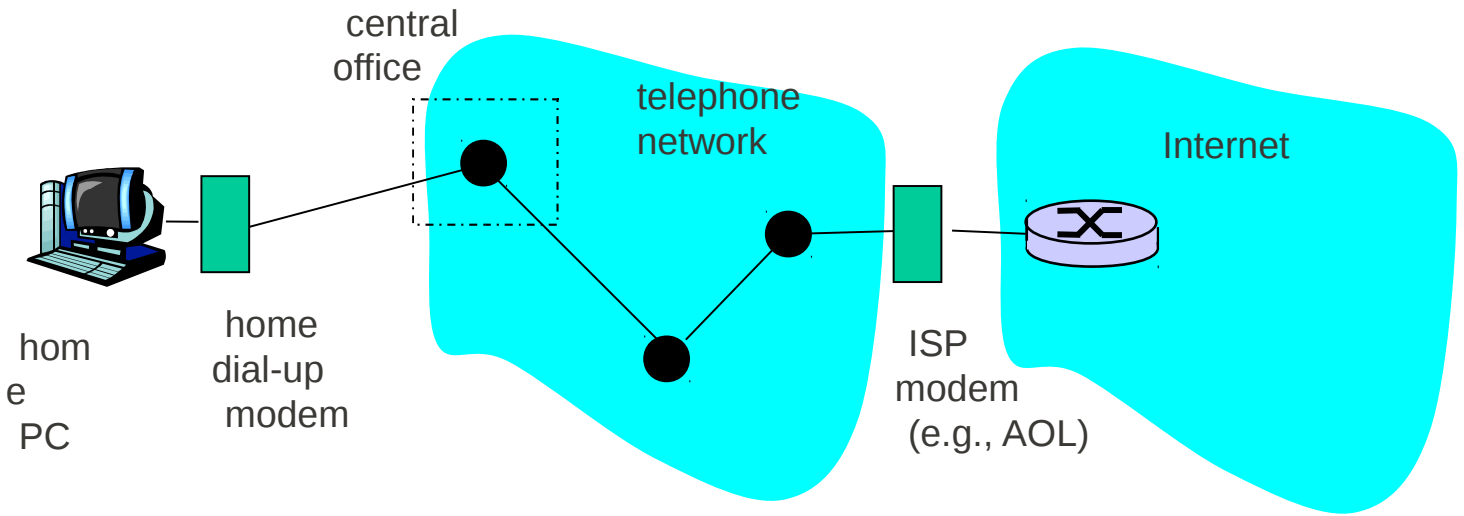
Network Edge: Access network

Q: How to connect end systems to edge router?

- residential access nets
  - Dial-Up
  - DSL
  - Cable
  - FTTH

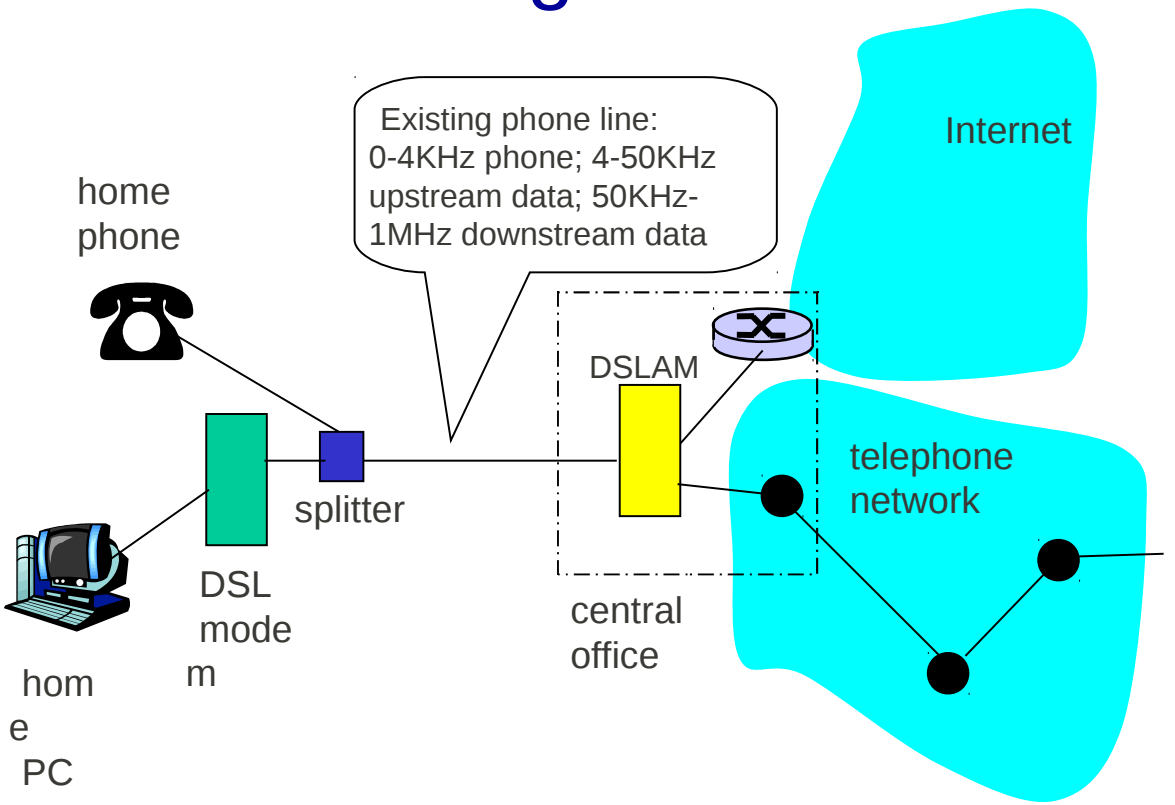


Residential Access: Dial-up Modem



- Uses existing telephony infrastructure
  - home directly-connected to central office
- up to 56Kbps direct access to router (often less)
- can't surf, phone at same time: not "always on"

Residential access: Digital Subscriber Line (DSL)

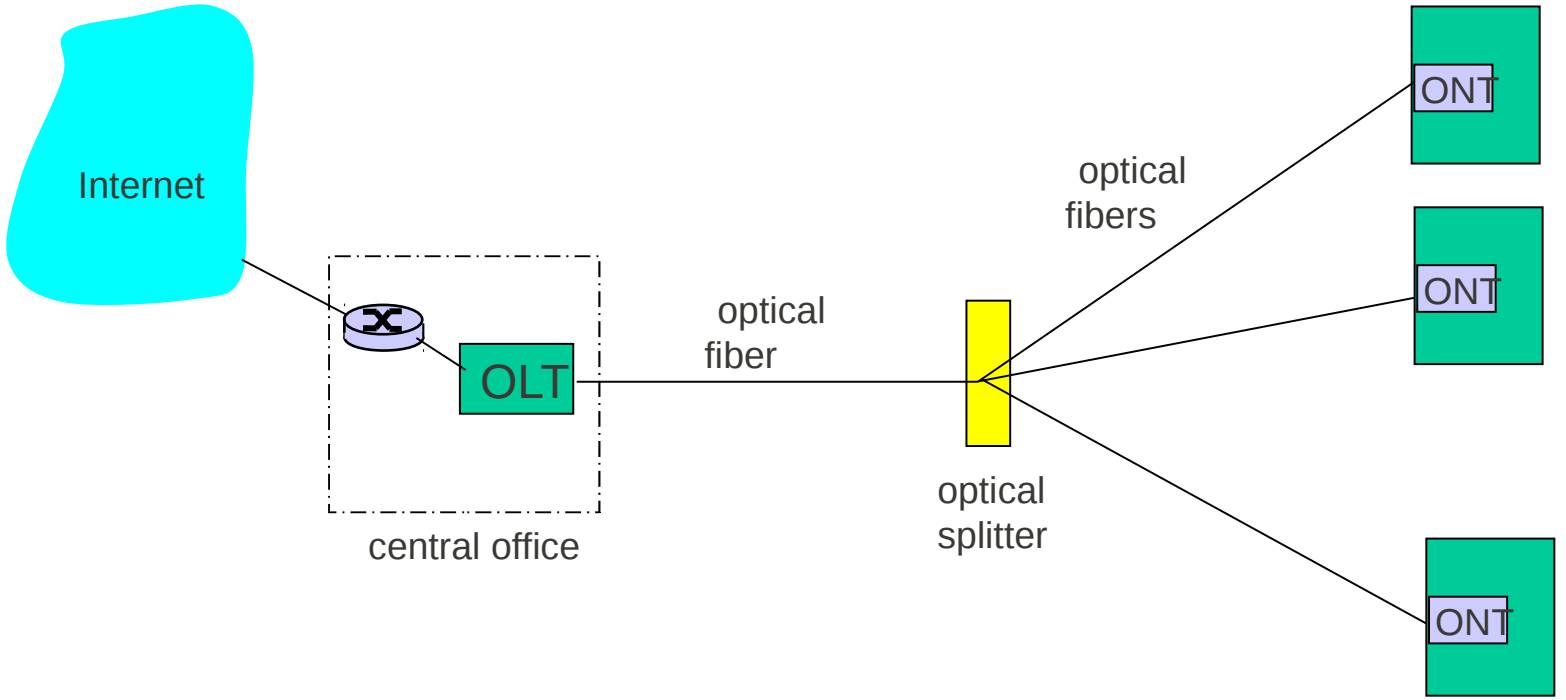


- Uses existing telephony infrastructure
- up to 1 Mbps upstream (today typically < 256 kbps)
- up to 8 Mbps downstream (today typically < 1 Mbps)
- dedicated physical line to telephone central office.

Residential Access: Cable

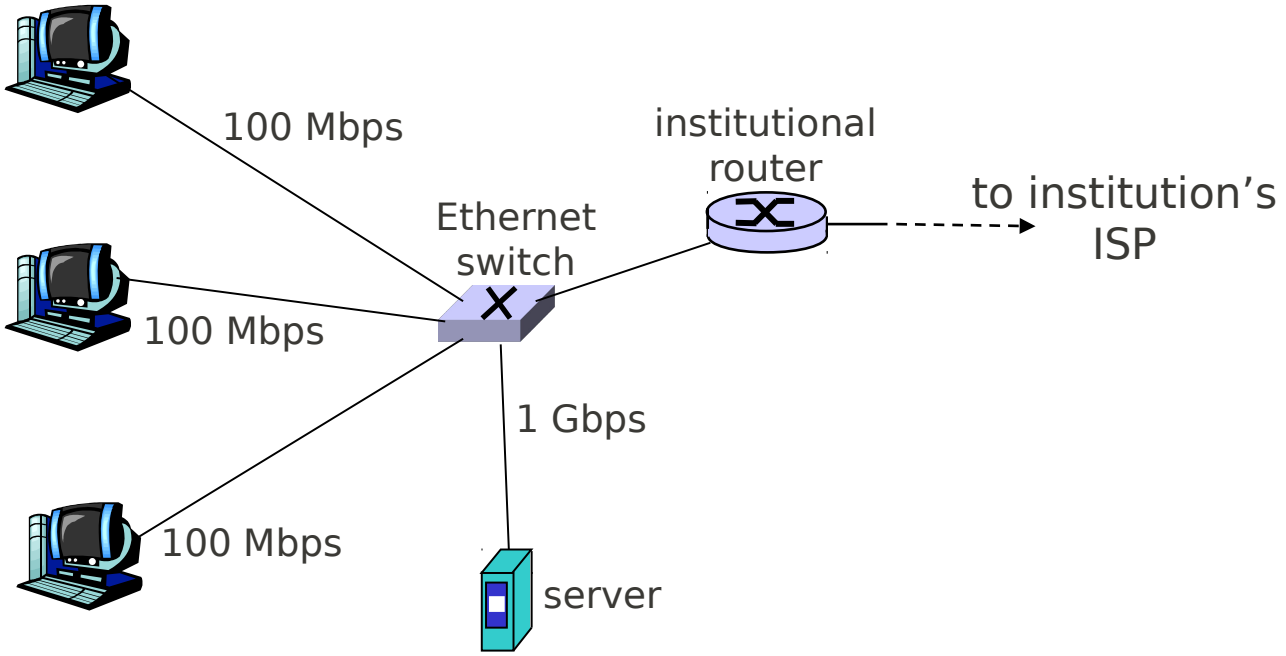
- Uses cable TV infrastructure, rather than telephone infrastructure

Residential Access: File-To-The-Home(FTTH)



- Optical links from central office to the home
- Two competing optical technologies:
  - Passive Optical network (PON)
  - Active Optical Network (PAN)
- Much higher Internet rates; fiber also carries television and phone services

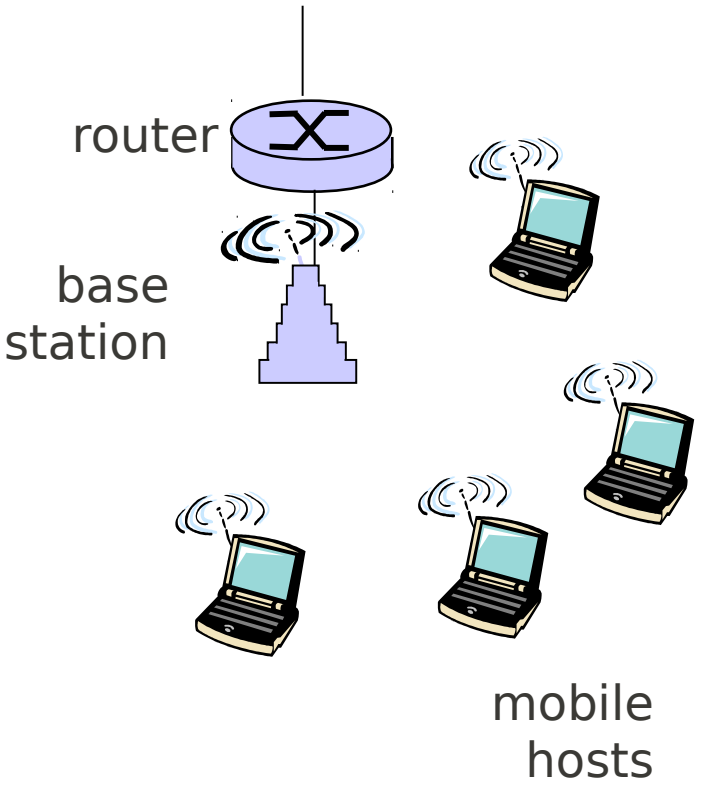
Company Access: Ethernet



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

Wireless access network

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



Chapter1 : Introduction

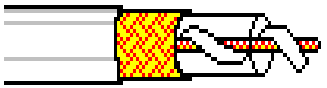
Overview:

- What's the Internet?
- What's a protocol?
- Network edge:
  - Hosts
  - Access net
  - Physical media
- Network core
- Performance: loss, delay, throughput
- Protocol layers, service models

Network edge: 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



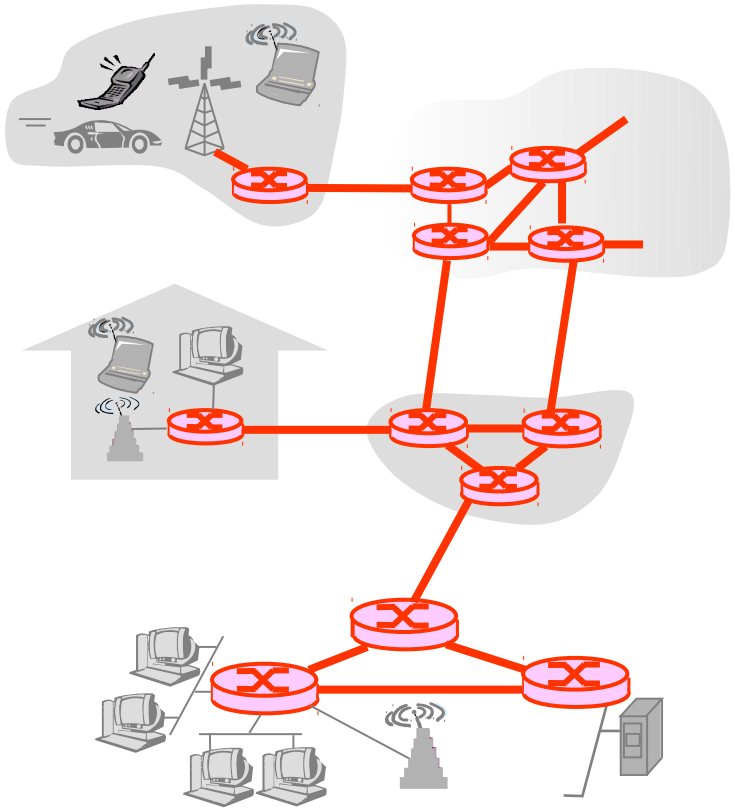
Chapter1 : Introduction

Roadmap:

- What's the Internet?
- What's a protocol?
- Network edge
- Network core
  - Circuit switching
  - Packet switching
  - Network structure
- Performance: loss, delay, throughput
- Protocol layers, service models

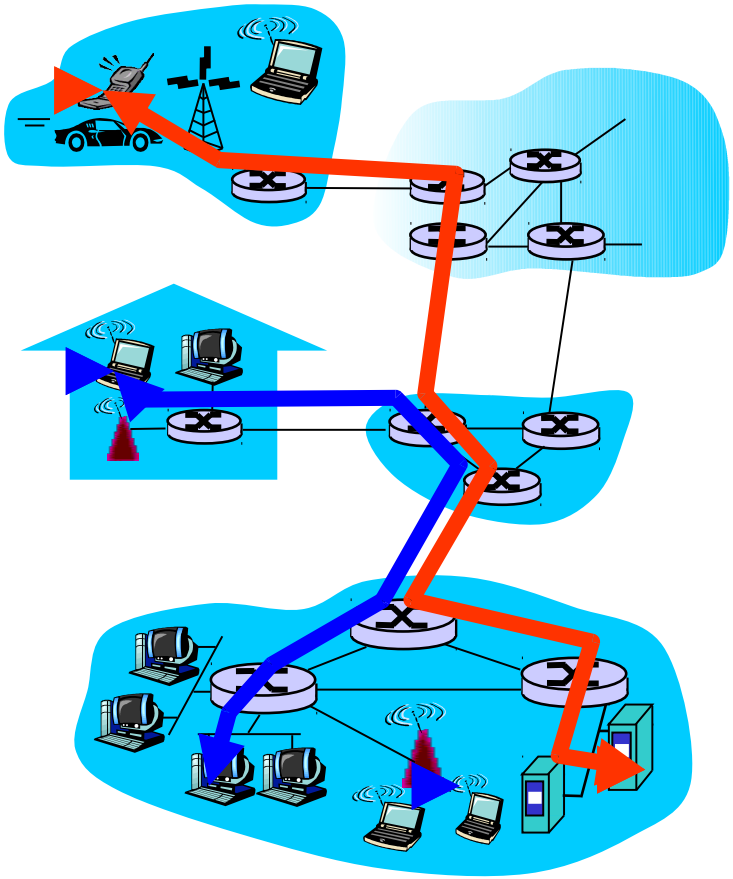
Network Core

- What is the network core? A mesh of interconnected routers
- How is data transferred through the network?
  - Circuit switching: dedicated circuit per call: telephone net
  - Packet switching: data sent thru net in discrete “chunks”



Network Core: Circuit Switching

- Circuit: A connection must be established between sender and the receiver.
  - The needed resources are reserved along the path.
  - Dedicated resources: no sharing
  - Performance guaranteed
  - Call setup is 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)*

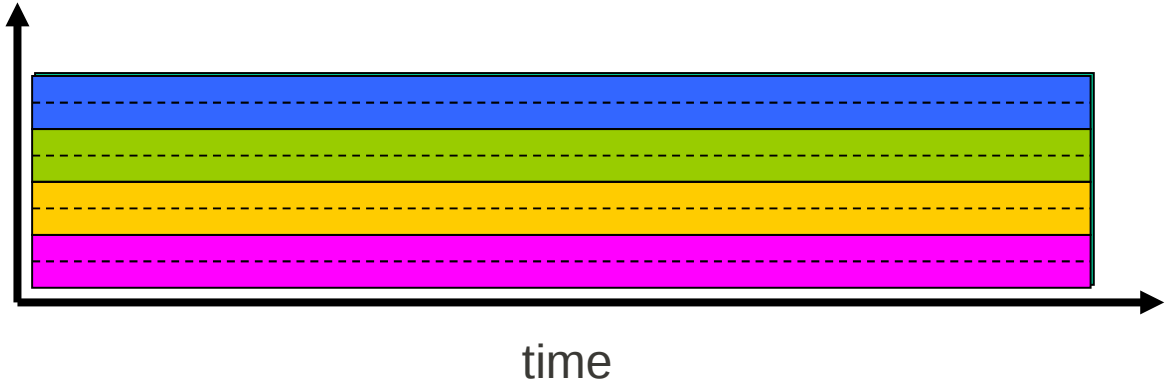
- Dividing link bandwidth into pieces:
  - frequency division
  - time division

Circuit Switching: FDM and TDM

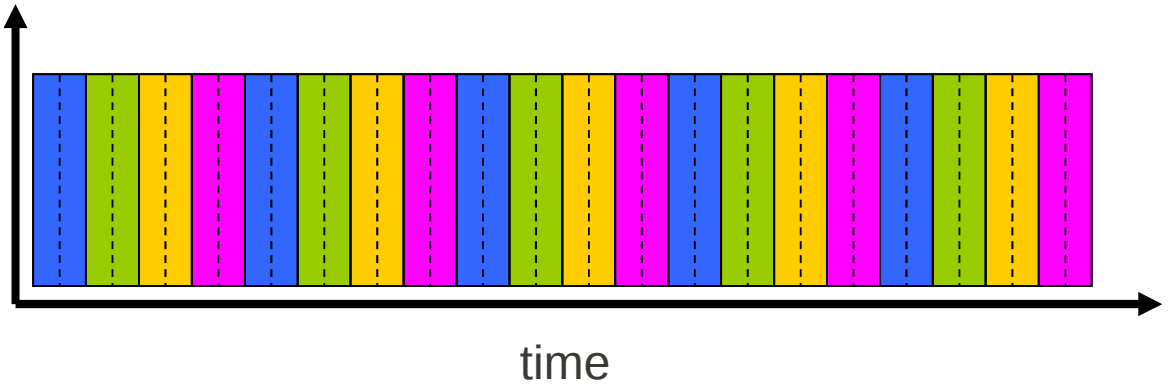
Example:

4 users

FDM



TDM



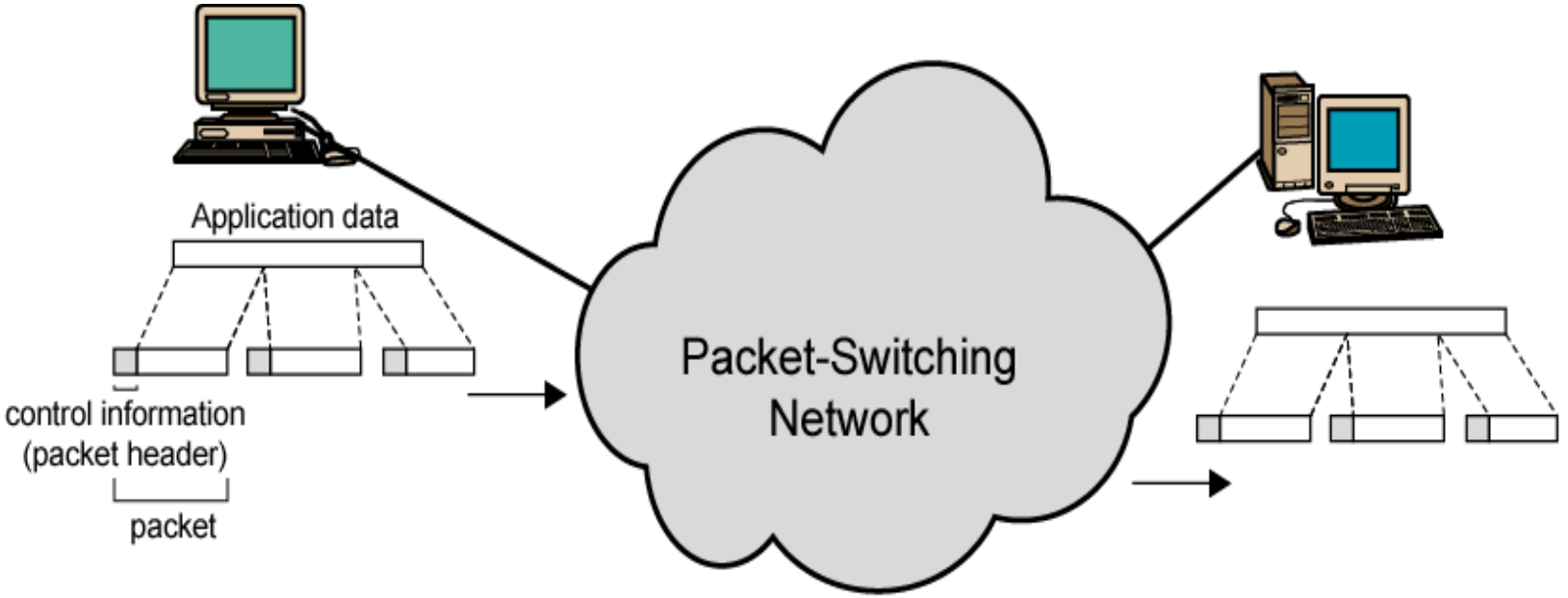
Numerical example

- How long does it take to send a file of 640,000 bits from host A to host B over a circuit-switched network?
  - all link speeds: 1.536 Mbps
  - each link uses TDM with 24 slots/sec
  - 500 msec to establish end-to-end circuit

Let’s work it out!

Network Core: Packet Switching

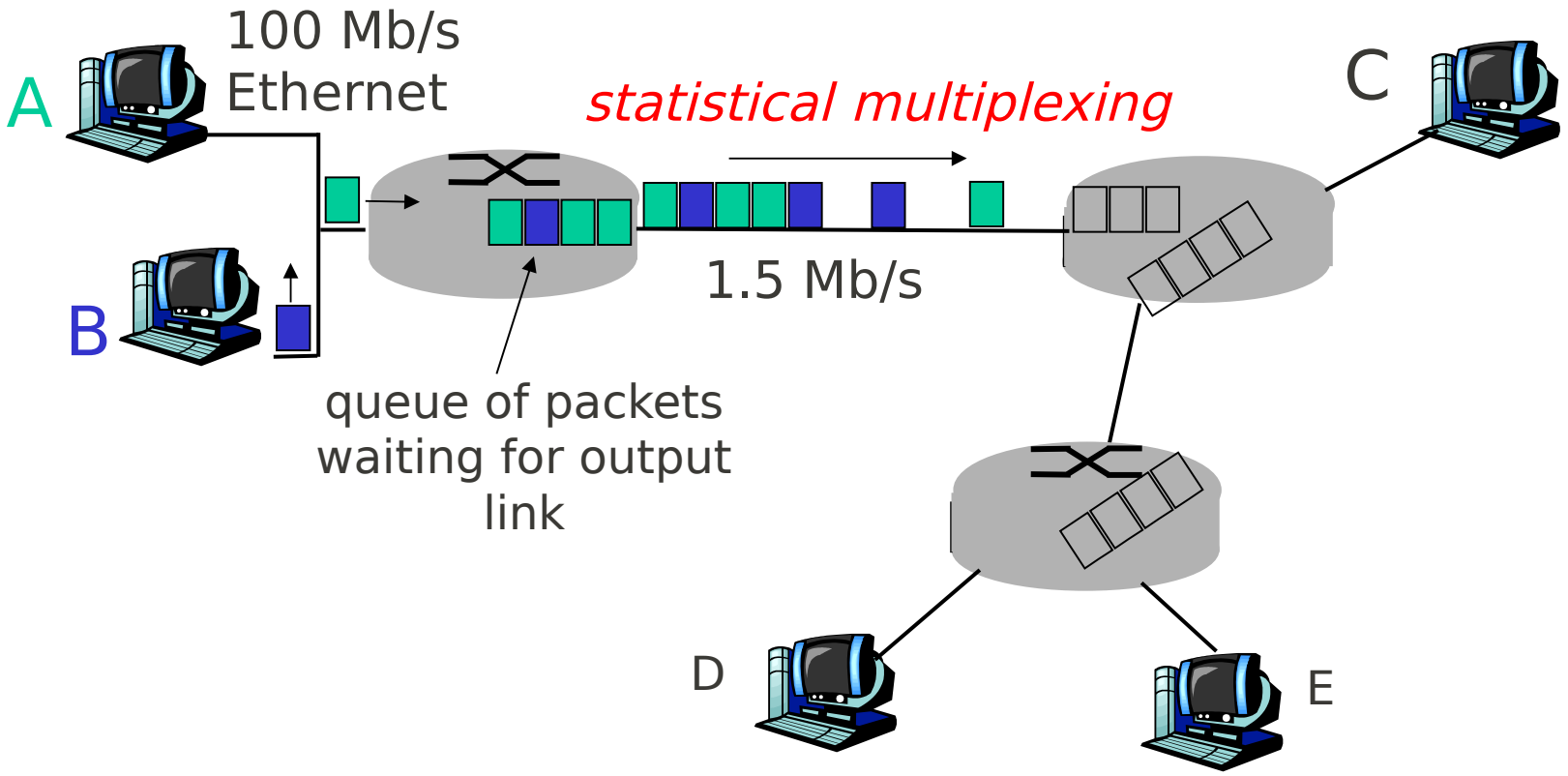
- Each end-end data stream divided into *packets*
- User A, B packets *share* network resources
- Each packet uses full link bandwidth
- Resources used *as needed*



Network Core: Packet Switching

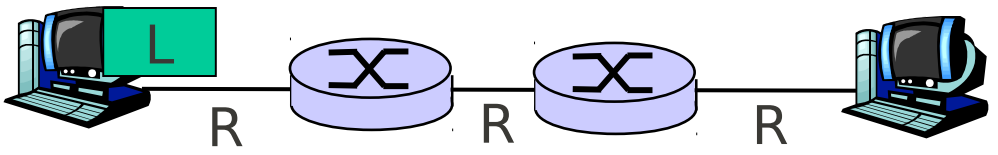
- Resource contentions
  - aggregate resource demand can exceed amount available
  - congestion: packets queue, wait for link use
  - store and forward: packets move one hop at a time
  - node receives complete packet before forwarding

Packet Switching: Statistical Multiplexing



- sequence of A & B packets has no fixed timing pattern
  - bandwidth shared on demand: *statistical multiplexing*.

Packet-switching: store-and-forward



- Takes L/R seconds to transmit (push out) packet of L bits on to link at R bps
- *Store and forward:* entire packet must arrive at router before it can be transmitted on next link
- Delay = 3L/R (assuming zero propagation delay)

  Example:

  L = 7.5 Mbits

  R = 1.5 Mbps

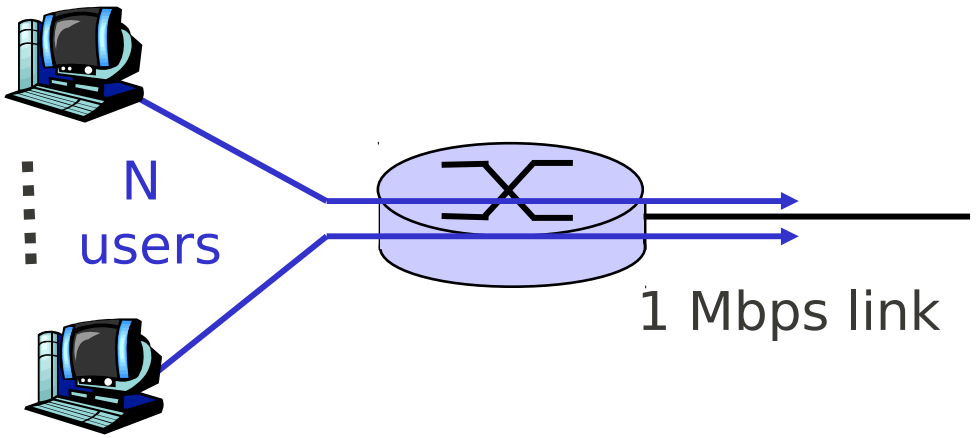
  transmission delay = 15 sec

Packet switching versus circuit switching

Packet switching allows more users to use network!

Example:

- 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 at same time is less than .0004

Q: how did we get value 0.0004?

Q: what happens if > 35 users ?

Packet switching versus circuit switching

Is packet switching a “slam dunk winner?”

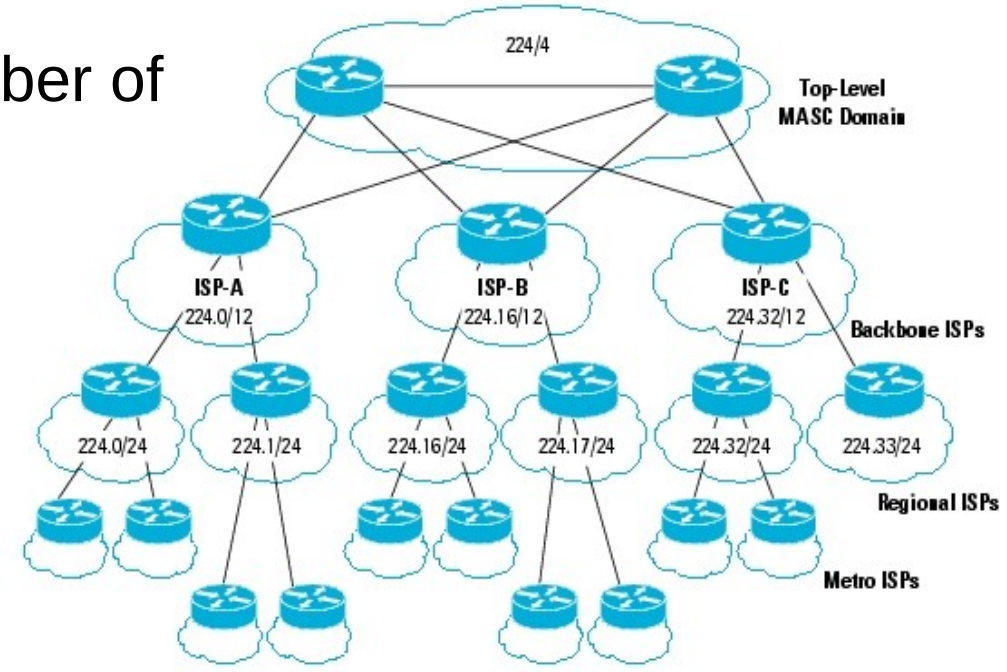
- great for bursty data
  - resource sharing
  - simpler, no call setup

- excessive congestion: packet delay and loss
  - protocols needed for reliable data transfer, congestion control

Internet structure

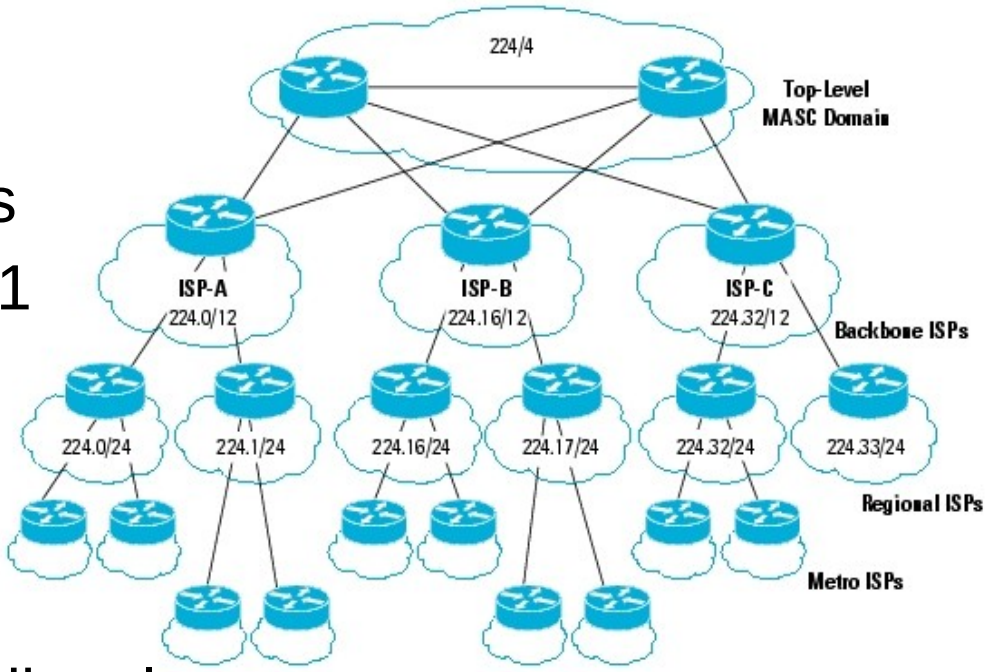
- Top Level (Tier-1 ISPs)
  - Internet backbones (AT&T,NTT,...)
  - Directly connected to each of other tier-1 ISP's
  - Connected to a large number of tier-2 ISPs
  - International in coverage

    simpler, no call setup



Internet structure

- Tier-2 ISPs
  - smaller (often regional) ISPs
  - connect to one or more tier-1 *(provider)* ISPs
  - each tier-1 has many tier-2 *customer nets*
  - tier 2 pays tier 1 provider
  - tier-2 nets sometimes peer directly with each other (bypassing tier 1) , or at IXP

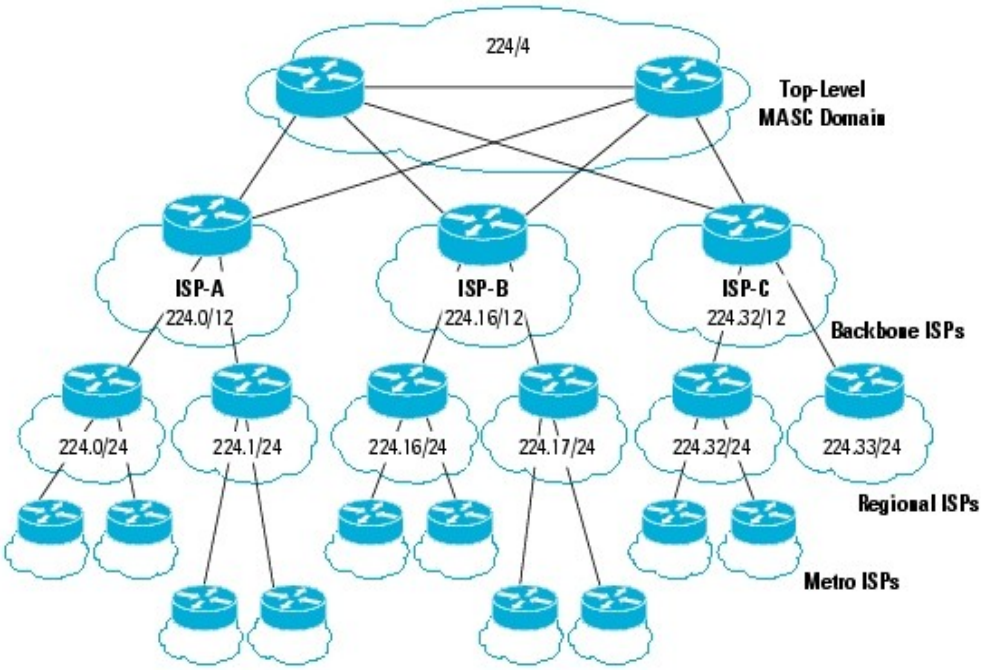


Internet structure

- Tier-3 ISPs or Local ISPs

customer of tier 1 or tier 2 network

last hop (“access”) network (closest to end systems)



Chapter1 : Introduction

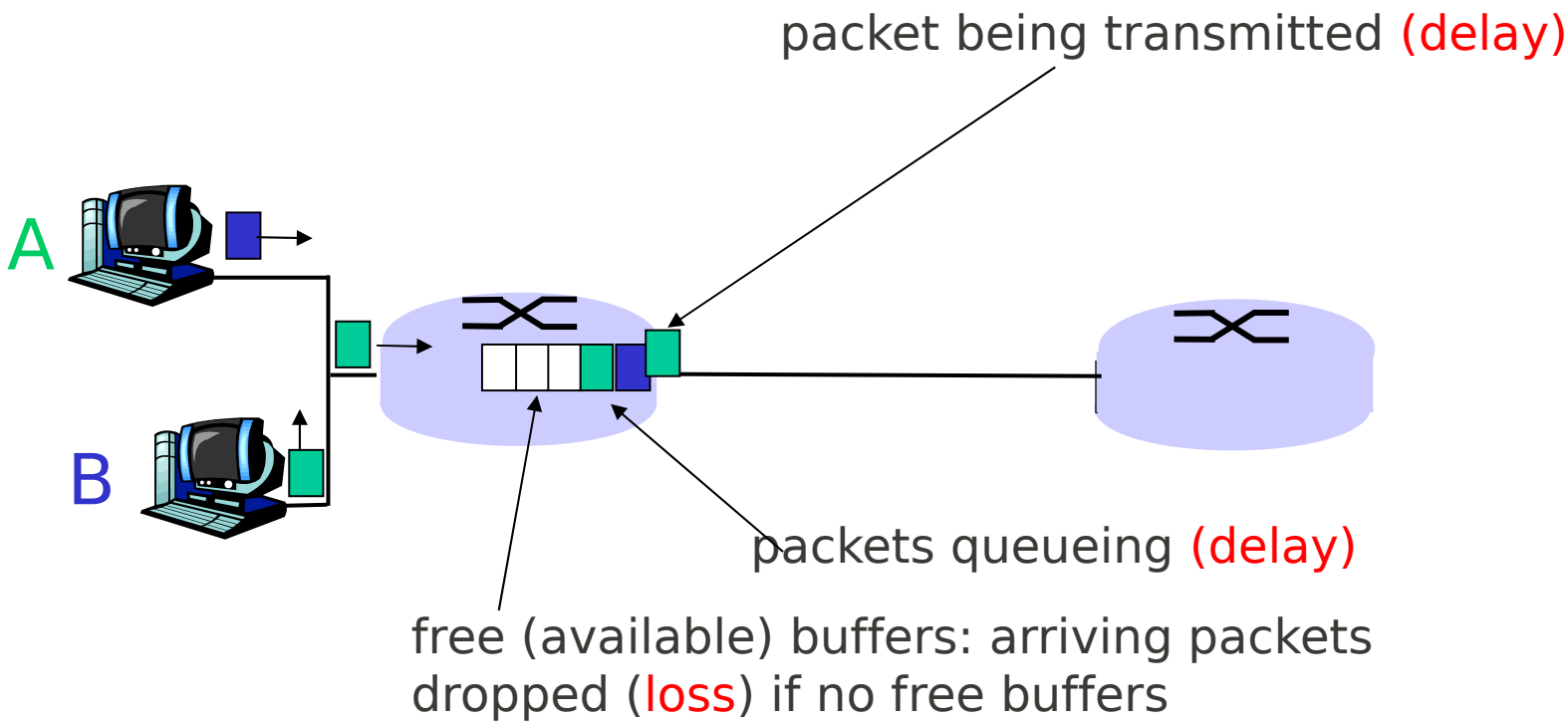
Overview:

- What's the Internet?
- What's a protocol?
- Network edge
- Network core
- Loss, delay, throughput in packet switching
- Protocol layers, service models

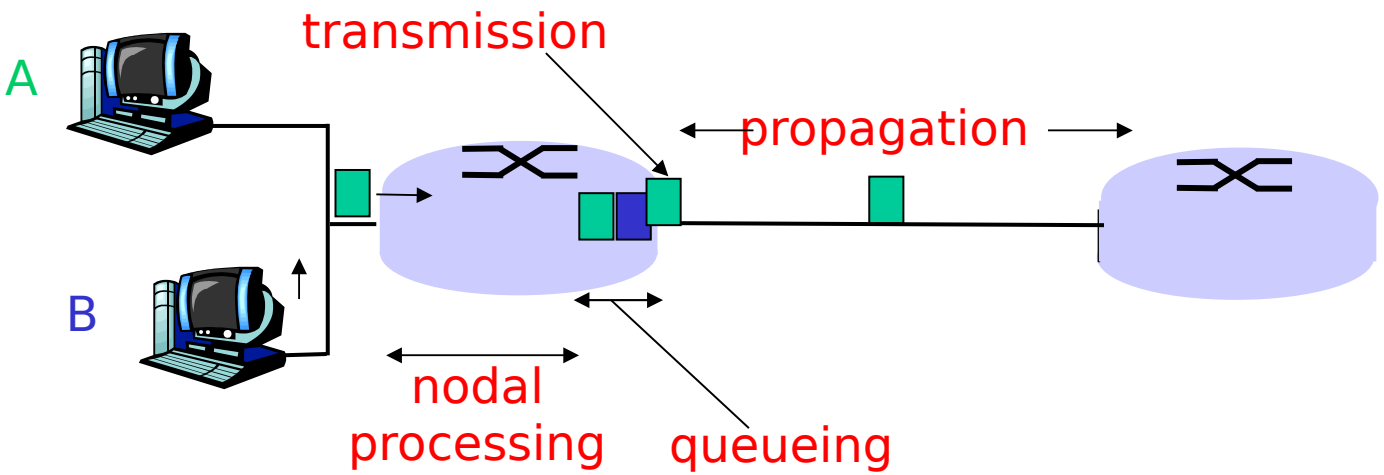
How do loss and delay occur?

packets *queue* in router buffers

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



Four sources of packet delay



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

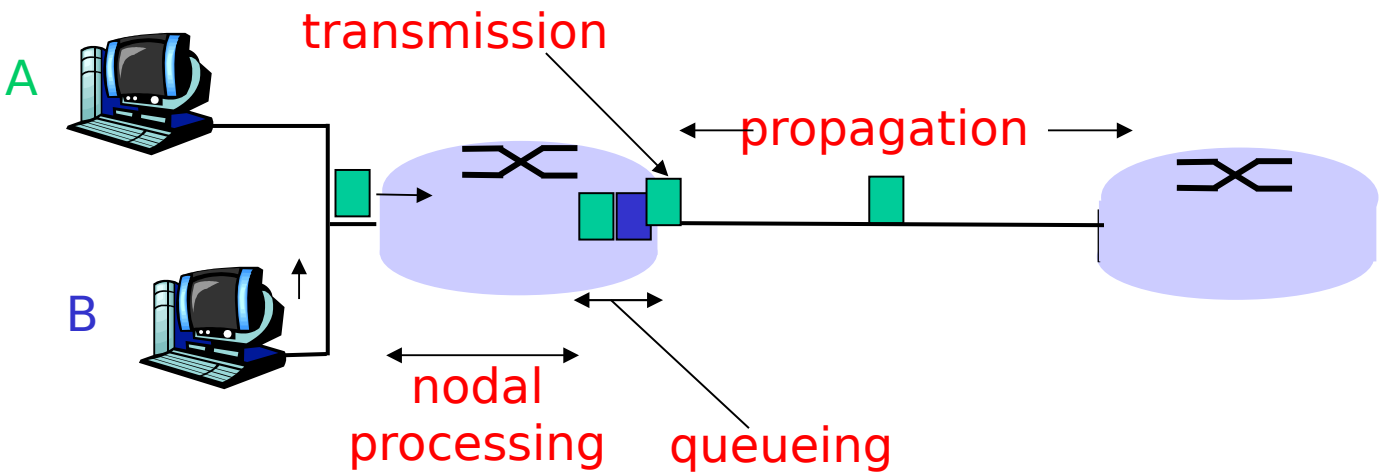
d_{proc}: nodal processing

- 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

Types of Delay



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

d_{trans}: transmission delay:

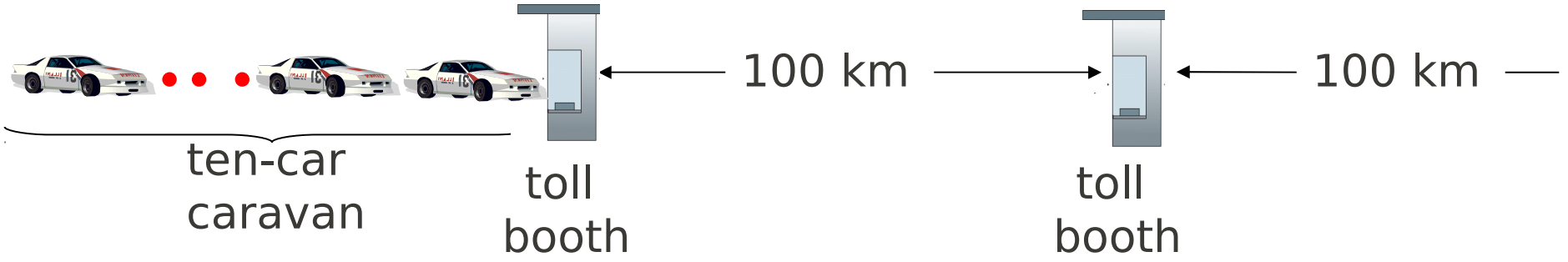
- L: packet length (bits)
- R: link bandwidth (bps)
- d_{trans} = L/R

d_{prop}: propagation delay:

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

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

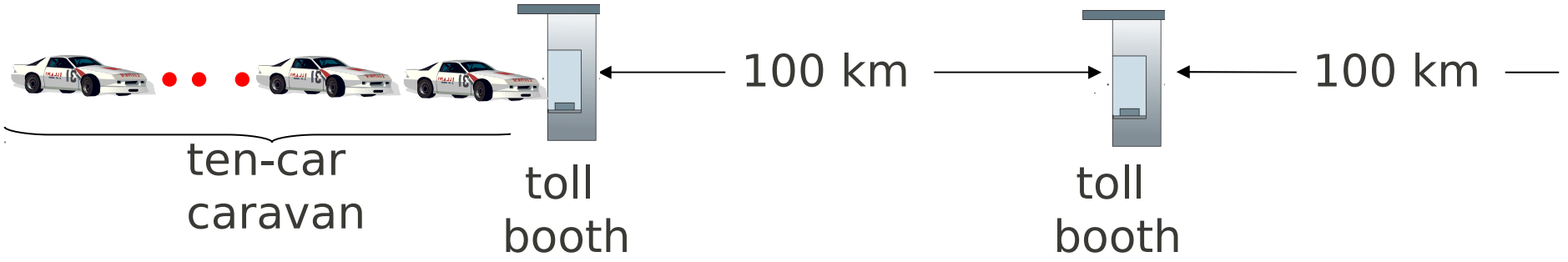
Transmission and Propagation delay



- cars “propagate” at 100 km/hr
- toll booth takes 12 sec to service car (transmission time)
- car~bit; caravan ~ packet
- 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
- Q: How long until caravan is lined up before 2nd toll booth?

  A: 62 minutes

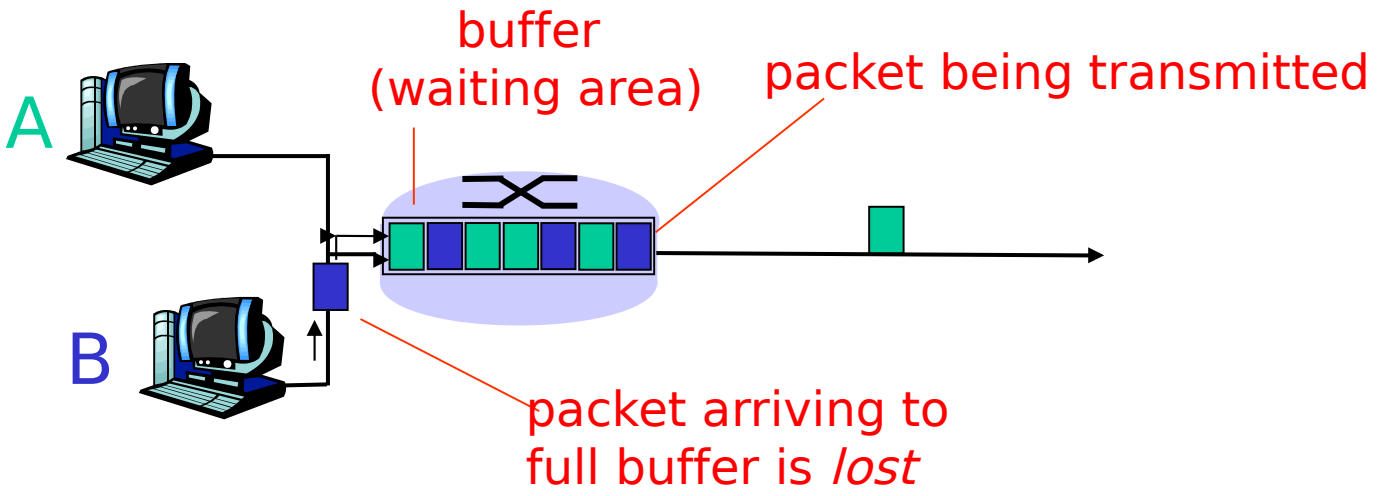
Caravan analogy (more)



- 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?

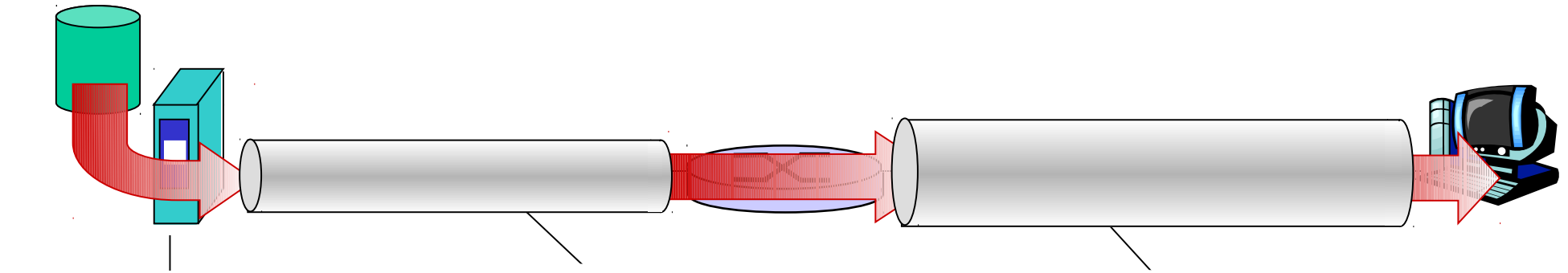
Packet loss

- 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



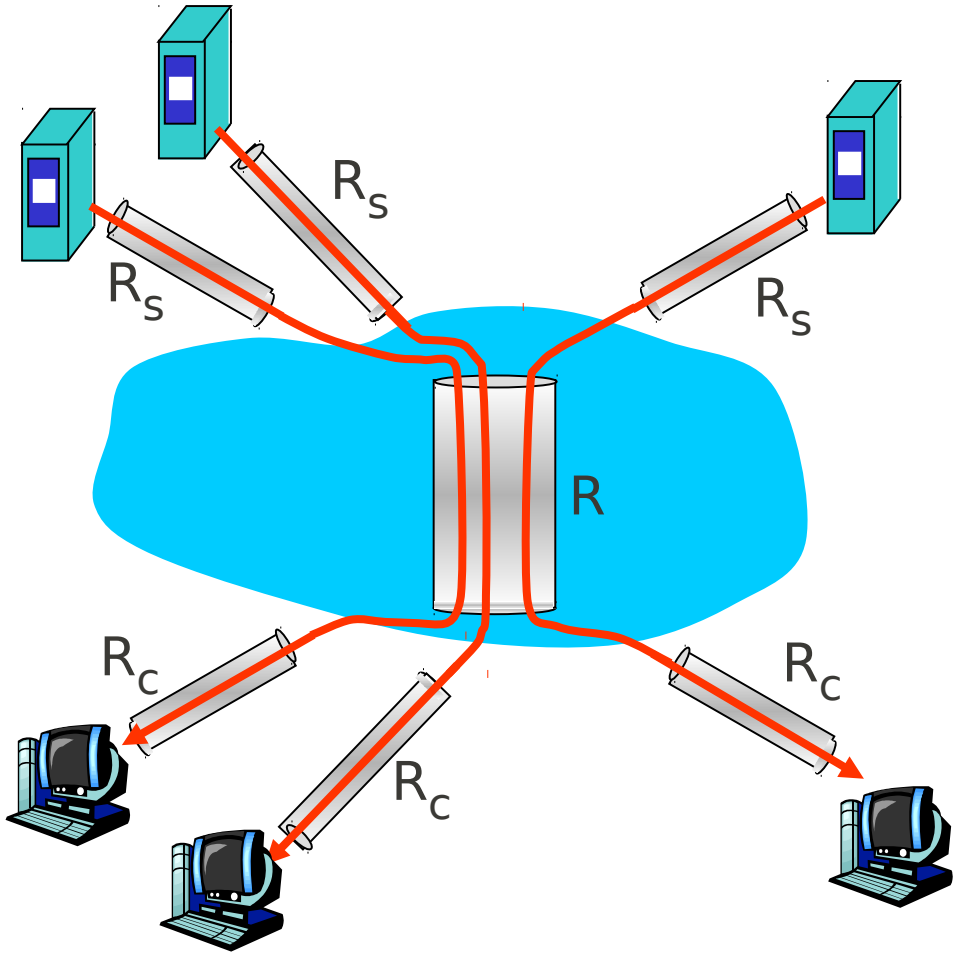
Throughput

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

Chapter1 : Introduction

Overview:

- What's the Internet?
- What's a protocol?
- Network edge
- Network core
- Loss, delay, throughput in packet switching
- Protocol layers, service models

Protocol “Layers”

Networks are complex, with many “pieces”:

- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Protocol “Layers”

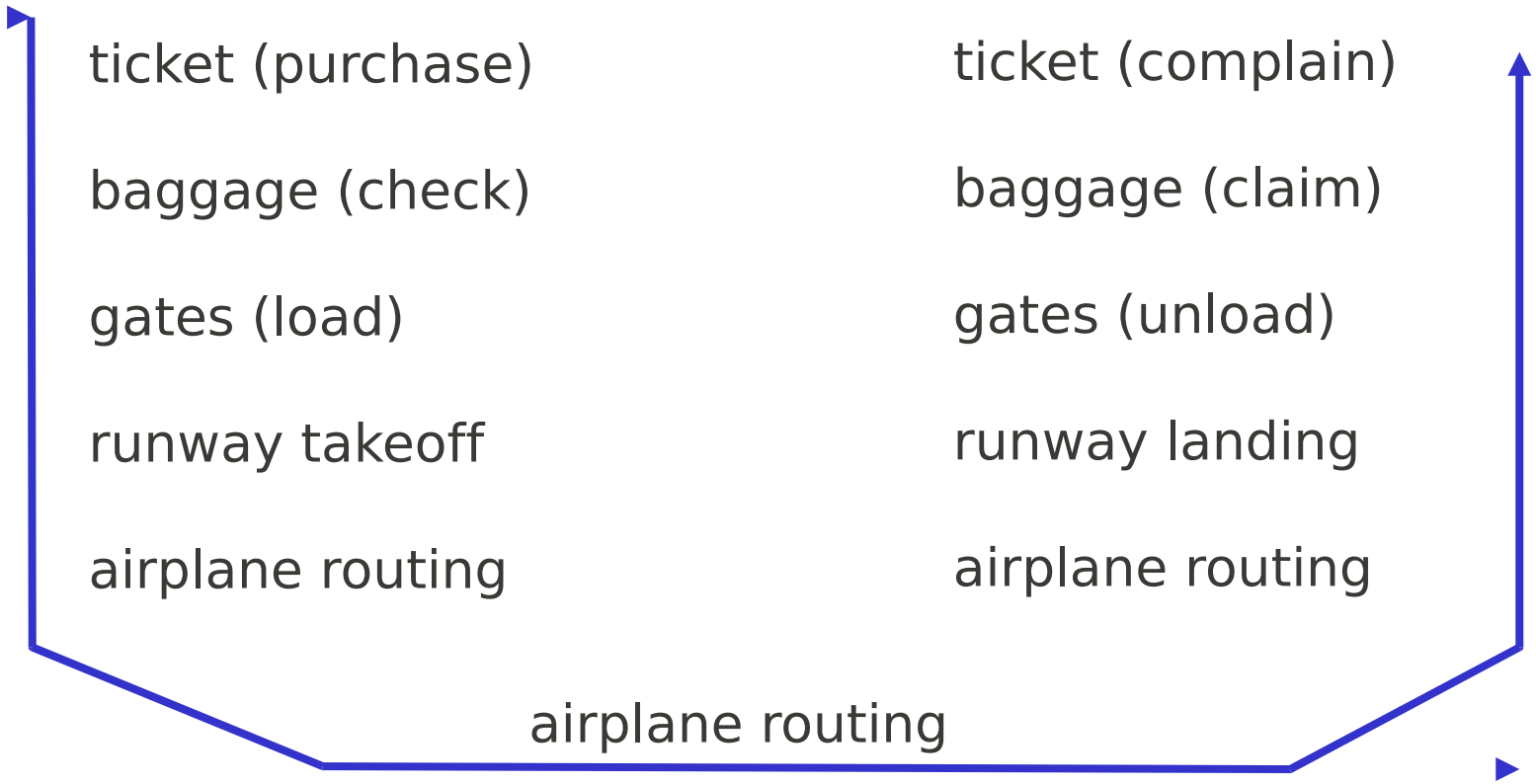
Networks are complex, with many “pieces”:

- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Question:

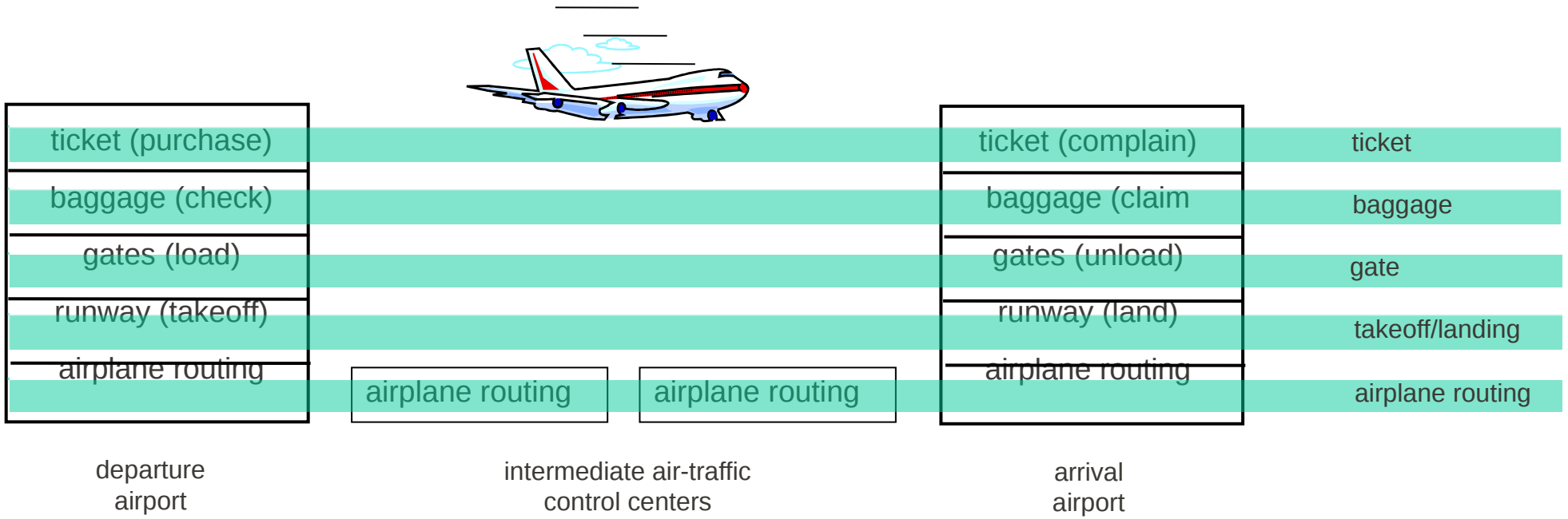
Is there any hope of *organizing* structure of network?

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

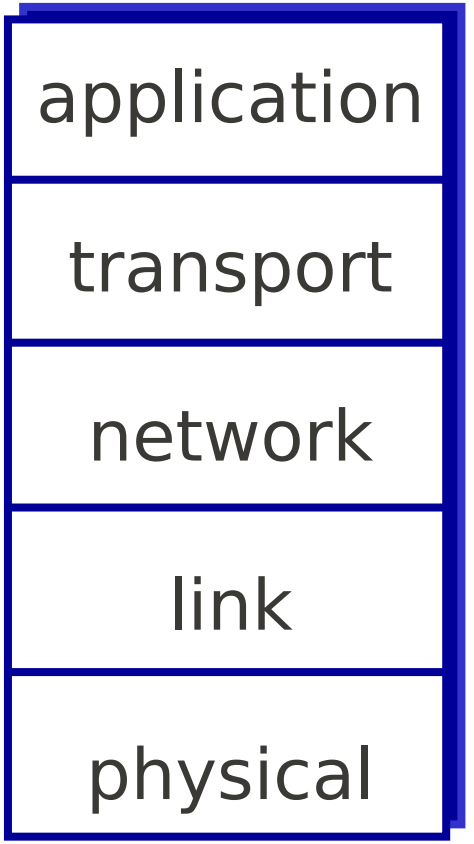
Why layering?

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
- layering considered harmful?

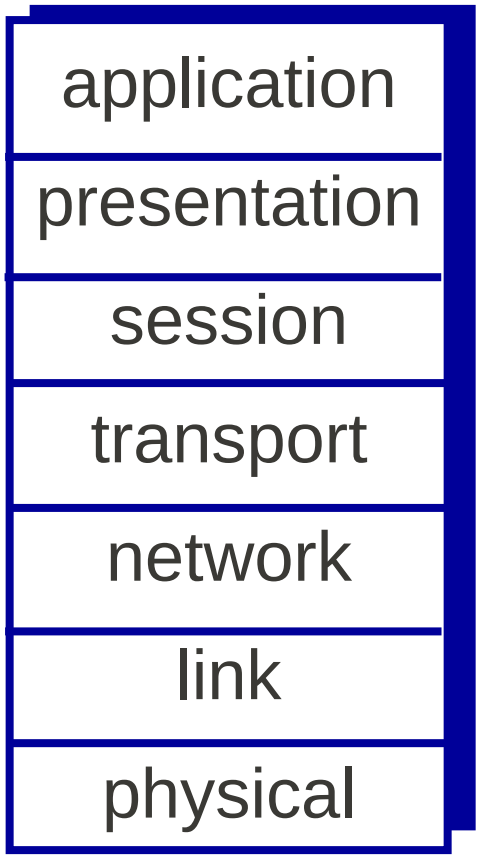
Internet protocol stack

- 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, 802.111 (WiFi), PPP
- 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?



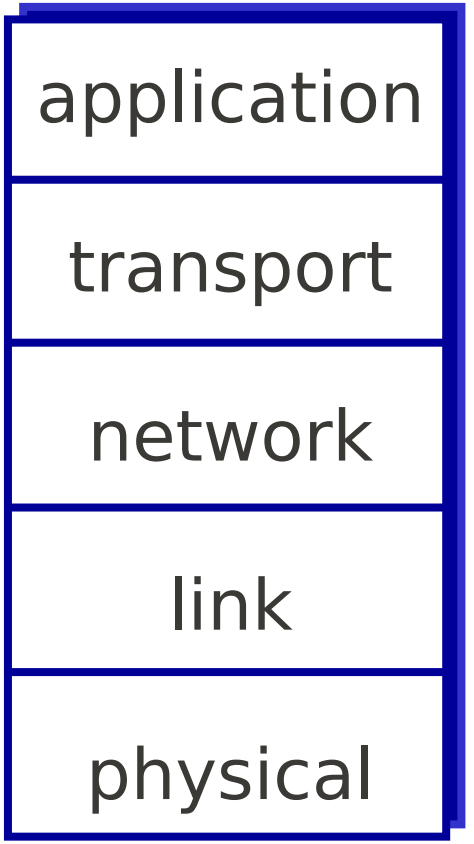
Why layering?

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
- layering considered harmful?

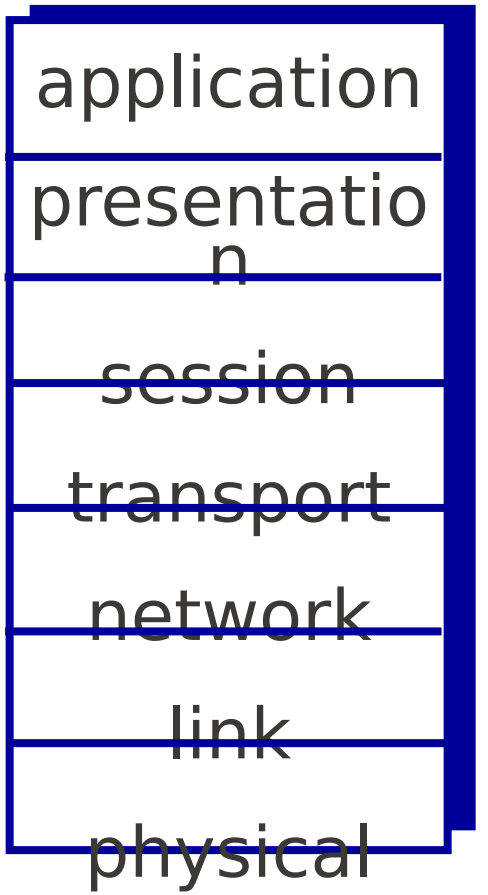
Internet protocol stack

- 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, 802.111 (WiFi), PPP
- 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