CS 3800: Computer Networks

Lecture 2: Packet Loss & Delay, Protocol Stack

Instructor: John Korah

Acknowledgement

- The following slides include material from author resources for:
  - KR Text book
  - “Data and computer communications,” William Stallings, Tenth edition

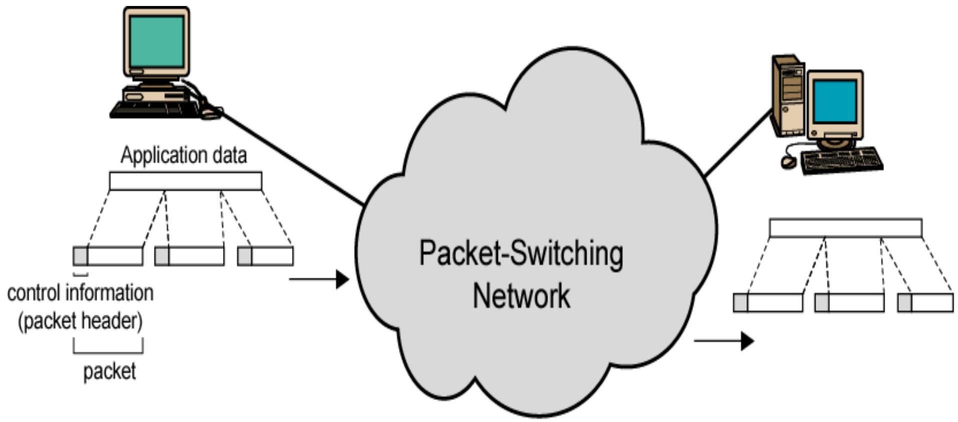
Learning Goals

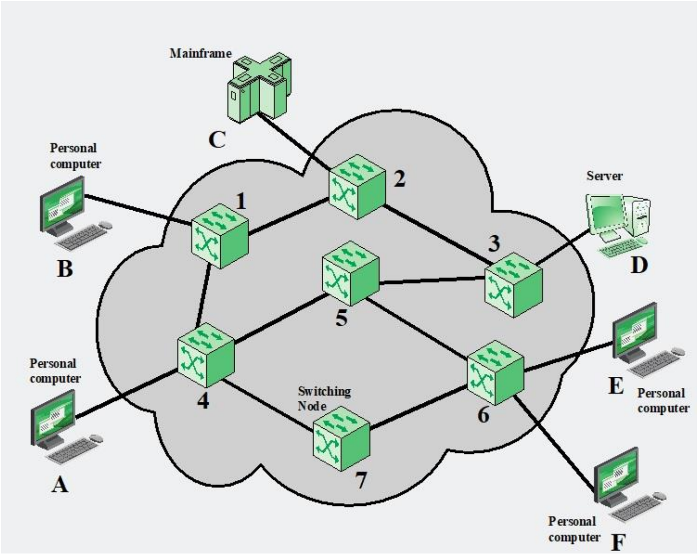
- Understand key performance metrics in packet switching networks such as packet loss and delay.
- Overview of the internet protocol stack
  - Why is protocols layering needed?
  - What is the function of each layer?

Today's Topics

- Delay, Loss, Throughput
- Protocol Layers

Recall: Packet Switching



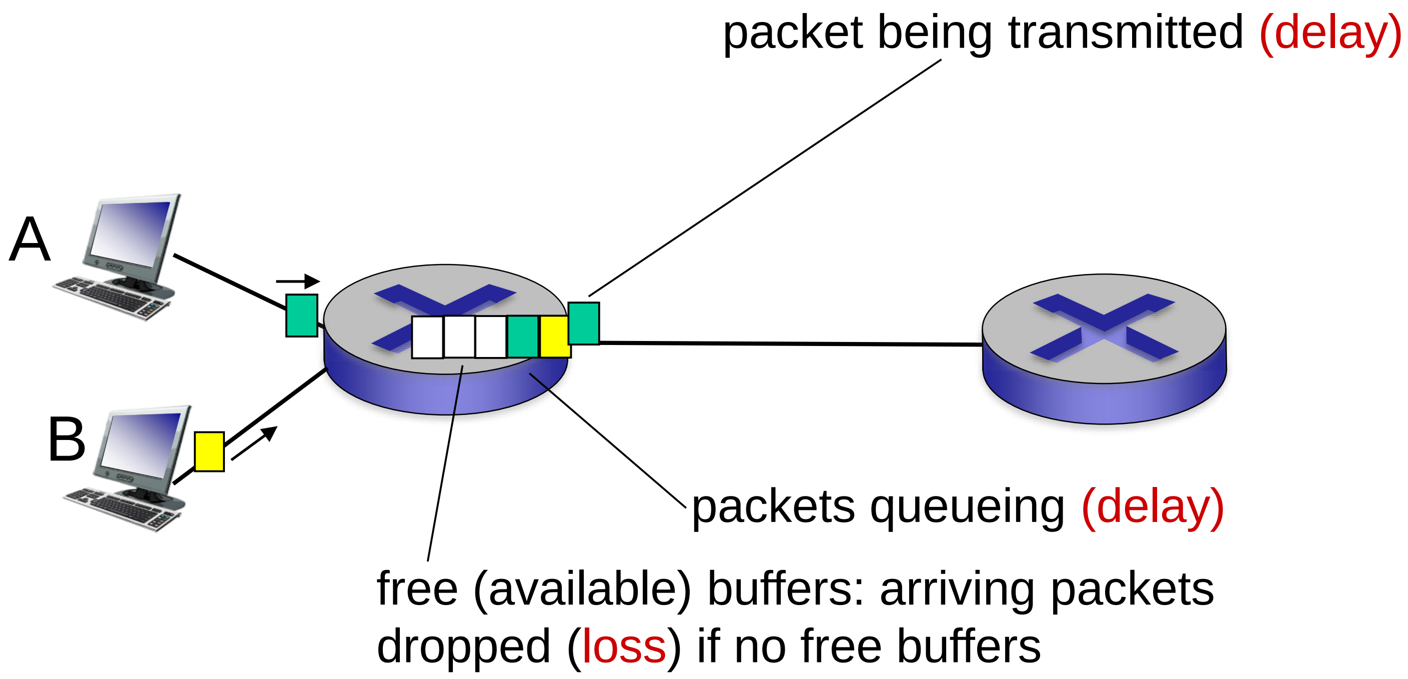


Packet delay and loss are the effects of congestion

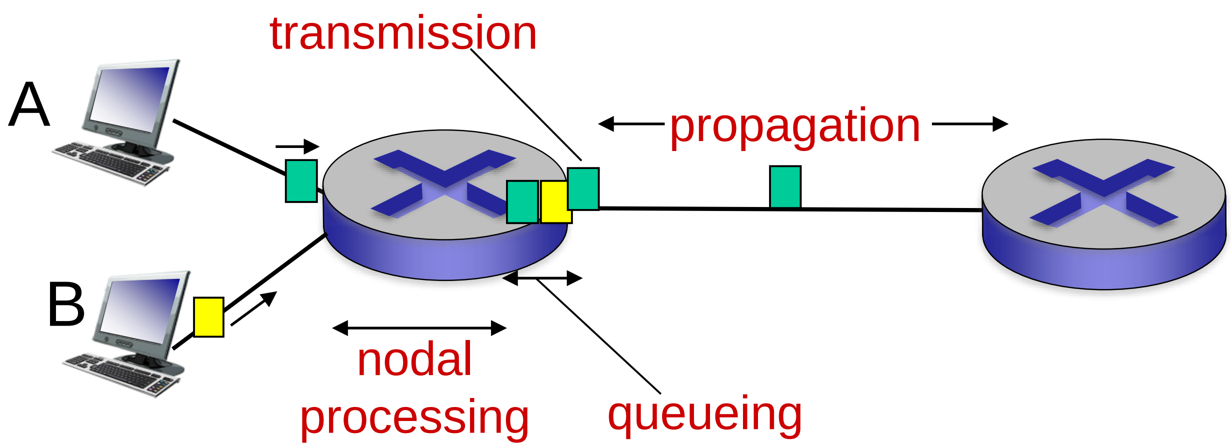
How do loss and delay occur?

packets *queue* in router buffers

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



Four sources of packet delay



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

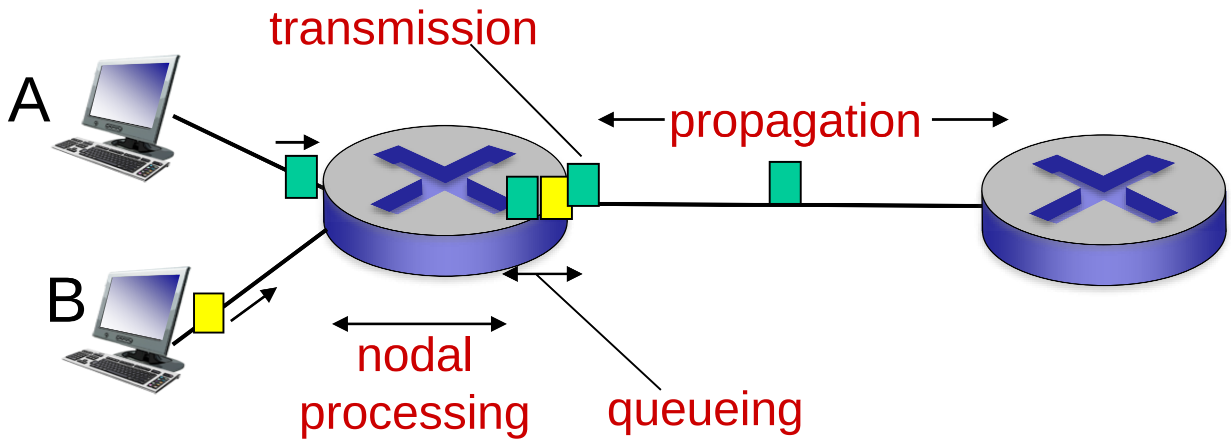
d_{proc}: nodal processing

- 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

Four 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)*
- $d_{trans} = L/R$

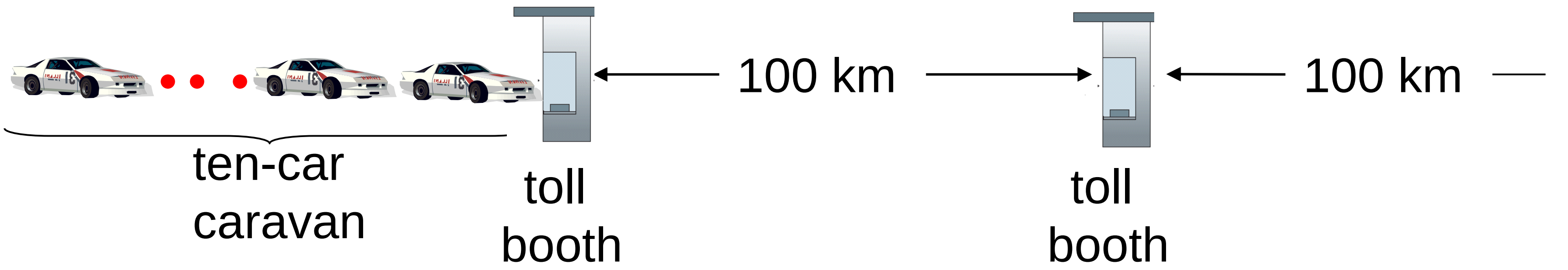
d_{trans} and d_{prop} *very* different

d_{prop}: propagation delay:

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

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

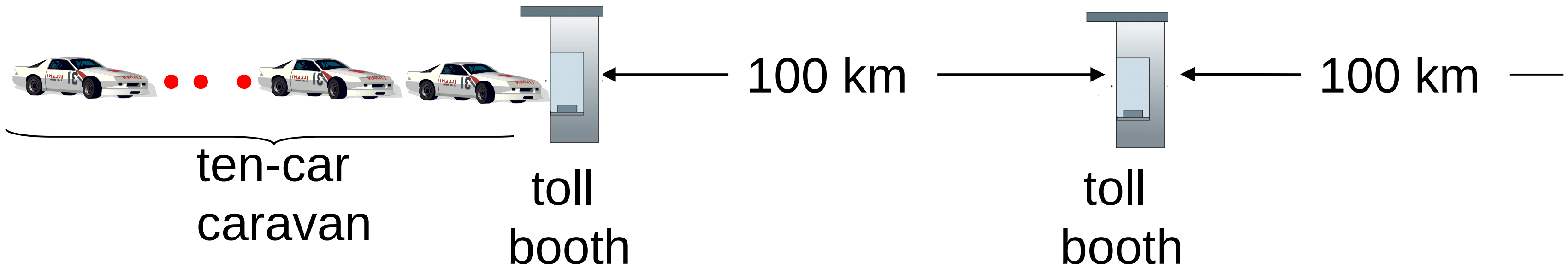
Caravan analogy



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

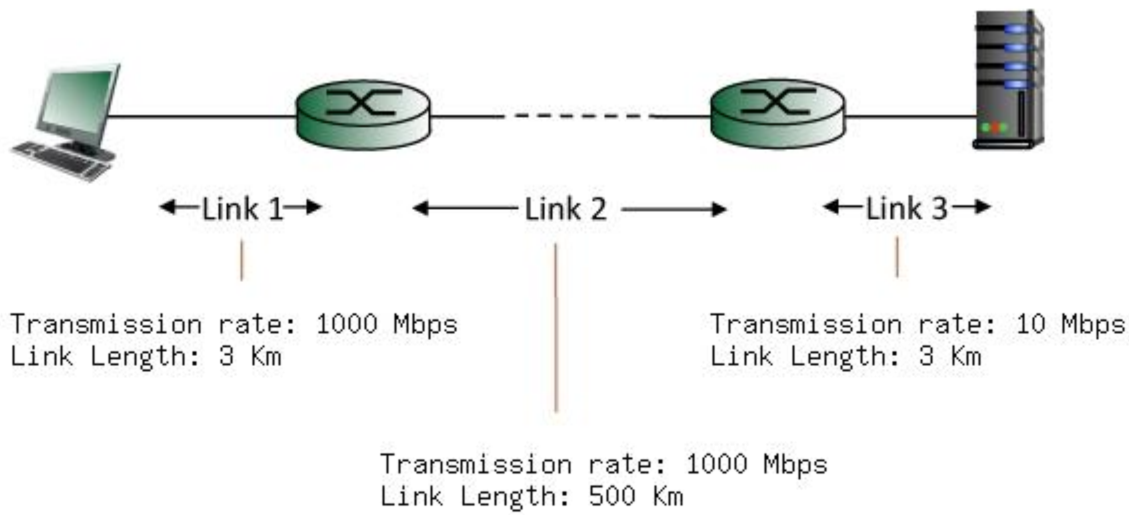
- time to "push" entire caravan through toll booth onto highway = 12*10 = 120 sec
- time for last car to propagate from 1st to 2nd toll both: 100km/(100km/hr)= 1 hr
- *A:* 62 minutes

Caravan analogy (more)



- suppose cars now "propagate" at 1000 km/hr
- and suppose toll booth now takes one min to service a car
- *Q:* Will cars arrive to 2nd booth before all cars serviced at first booth?
    - *A: Yes!* after 7 min, first car arrives at second booth; three cars still at first booth

Problem



- Find the end-to-end delay (including the transmission delays and propagation delays on each of the three links but ignoring queueing delays and processing delays) from when the left host begins transmitting the first bit of a packet to the time when the last bit of that packet is received at the server at the right. The speed of light propagation delay on each link is 3×10^8 m/sec. Note that the transmission rates are in Mbps and the link distances are in km. Assume a packet length of **12000** bits. Give your answer in milliseconds.

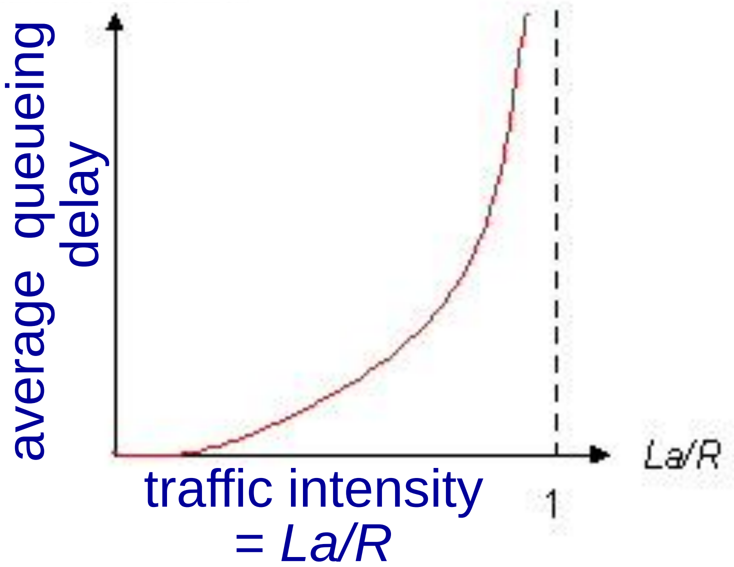
Queueing delay (revisited)

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

- *What happens when*
  - $La/R \sim 0$
  - $La/R \rightarrow 1$
  - $La/R > 1$

Queueing delay (revisited)

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



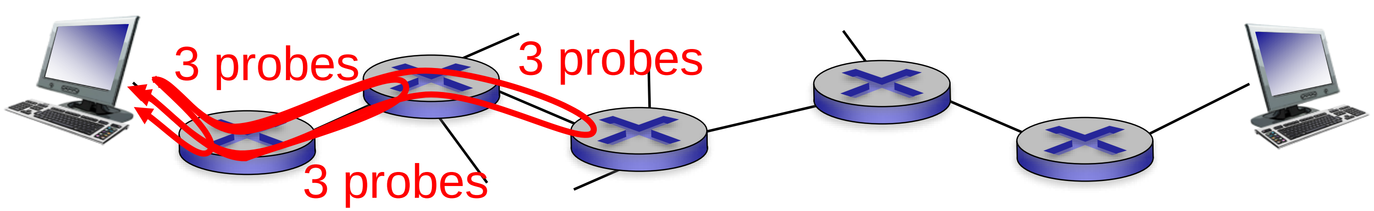
- *La/R* ~ 0: avg. queueing delay small
- *La/R* → 1: avg. queueing delay gradually grows large
- *La/R* > 1: more "work" arriving than can be serviced, average delay infinite!

La/R ~ 0

La/R -> 1

"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 all router *i*:
  - 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, routes

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

3 delay measurements 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

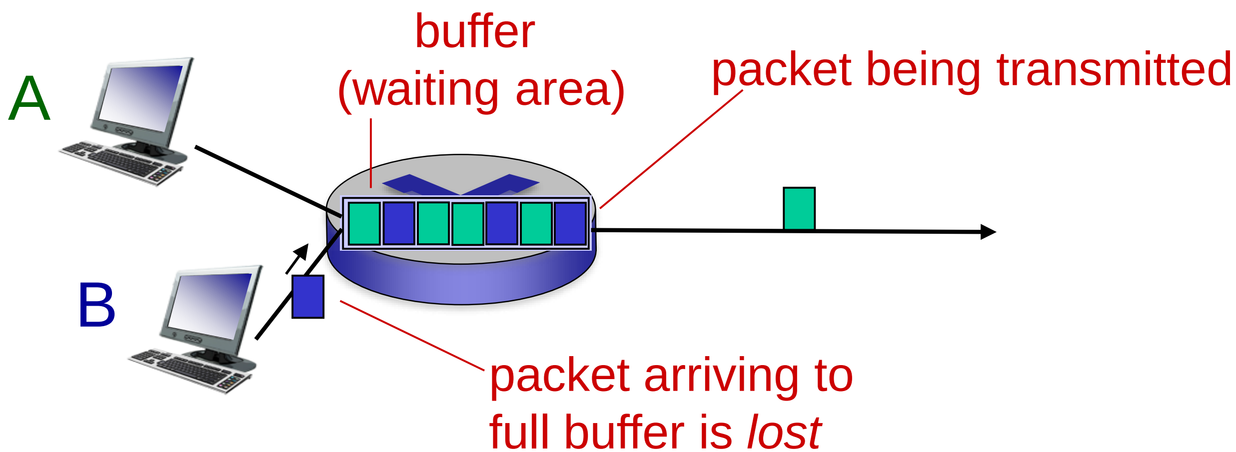
trans-oceanic link

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

* Exercise: Do some traceroutes from exotic countries at www.traceroute.org

Packet loss

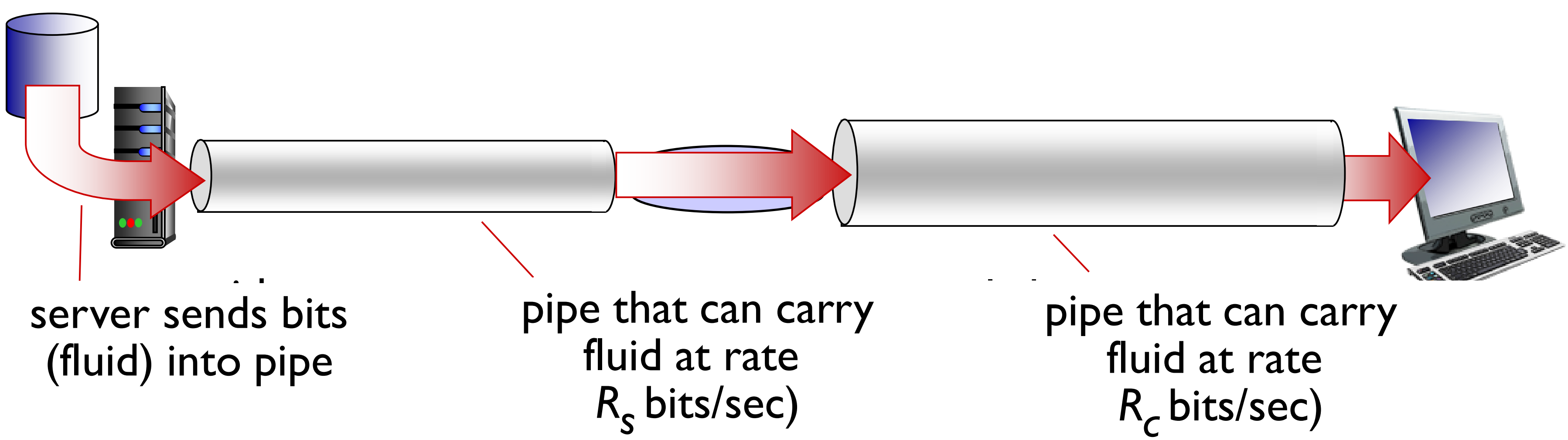
- queue (aka buffer) preceding link in buffer has finite capacity
- packet arriving to full queue dropped (aka lost)
- lost packet may be retransmitted by previous node, by source end system, or not at all



* Check online interactive animation on queuing and loss here

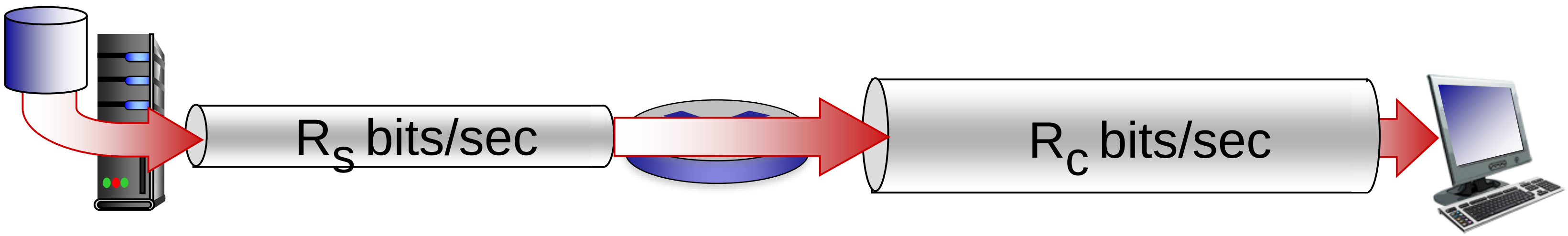
Throughput

- *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

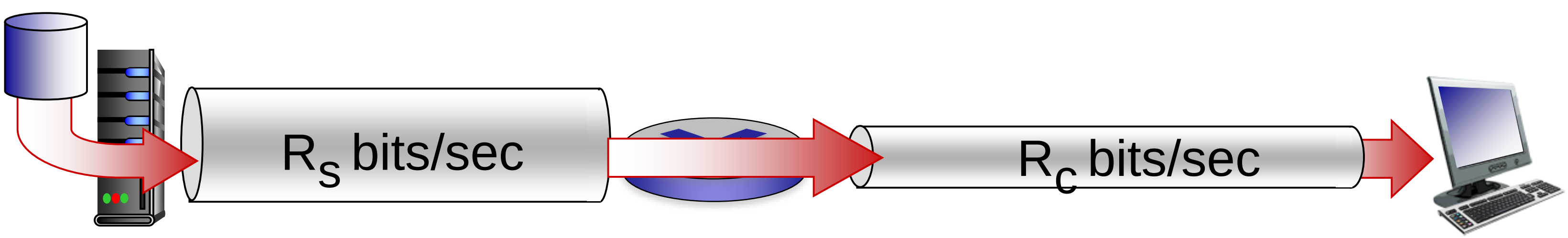


Throughput (more)

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



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

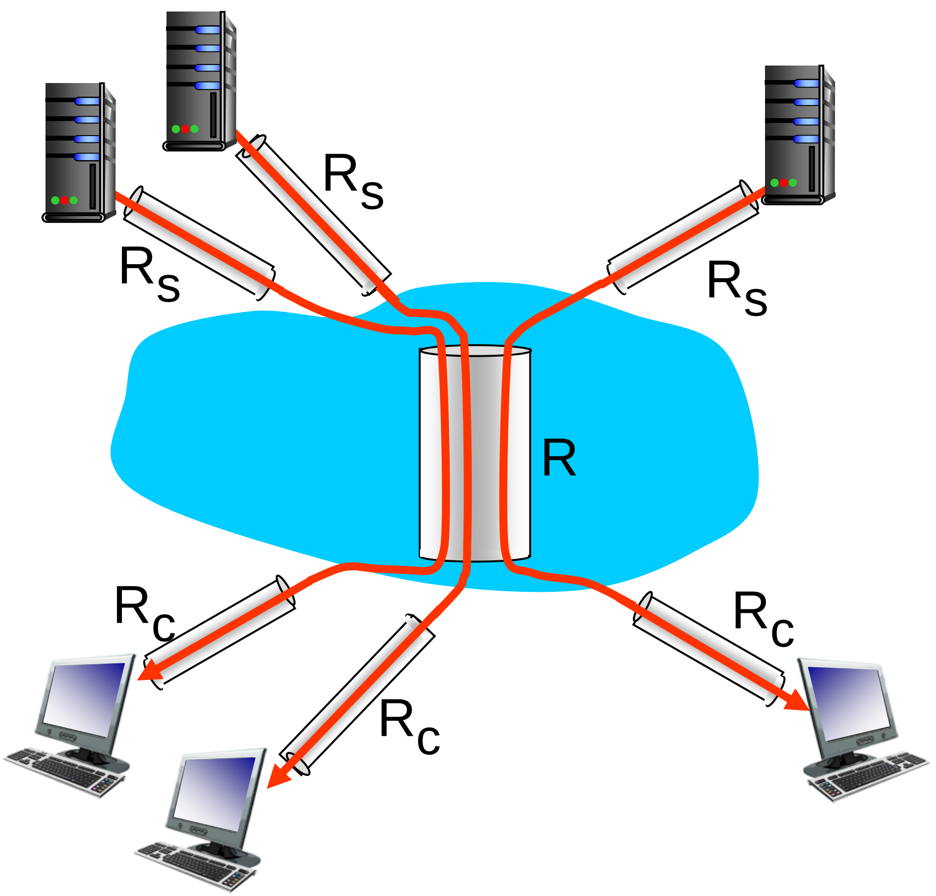


bottleneck link

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

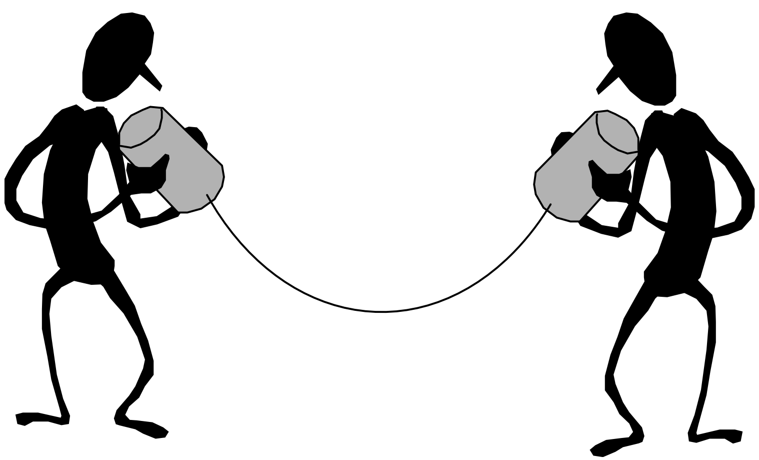
Today's Topics

- Delay, Loss, Throughput
- Protocol Layers

Network Protocols

To destroy communication completely, there must be no rules in common between transmitter and receiver—neither of alphabet nor of syntax.

—*On Human Communication*,
Colin Cherry



Key Features of a Protocol

A protocol is a set of rules or conventions that allow peer layers to communicate

The key features of a protocol are:

Syntax	• Format of data blocks
Semantics	• Control information for coordination and error handling
Timing	• Flow control and sequencing

Protocol "layers"

Networks are complex, with many "pieces":

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

Question:

- How do we make it work?

Protocol "layers"

Networks are complex, with many "pieces":

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

Question:

- How do we make it work?

- We will employ a divide and conquer approach
- How do we divide the complexity?

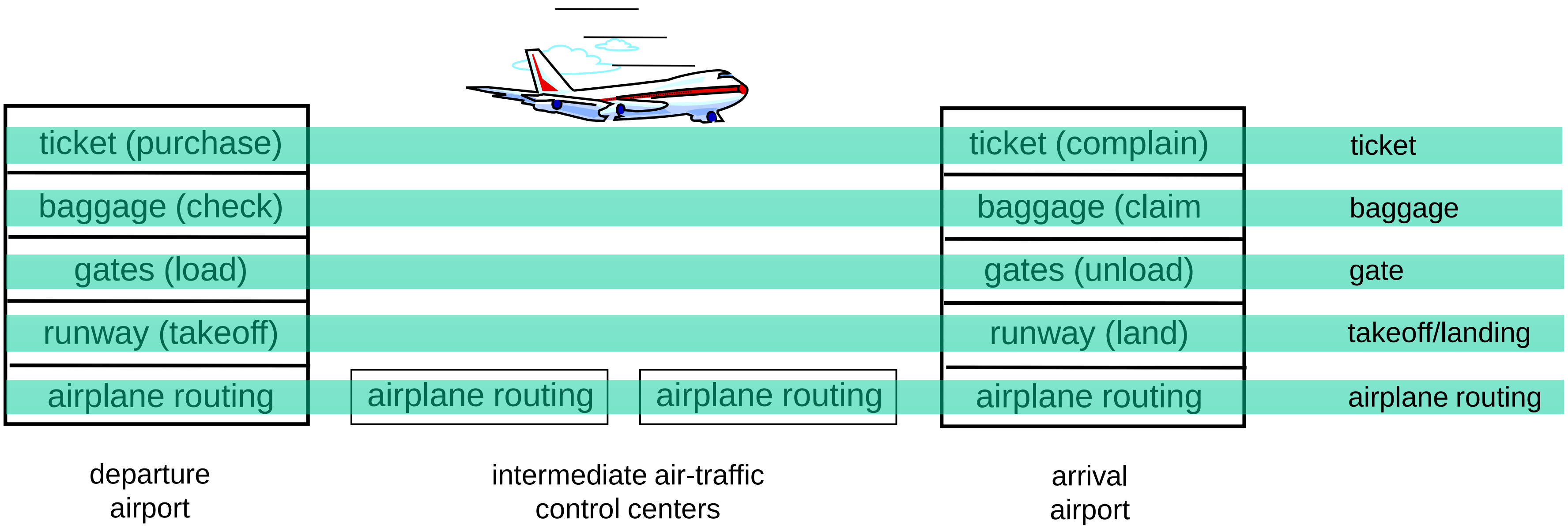
Organization of air travel

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

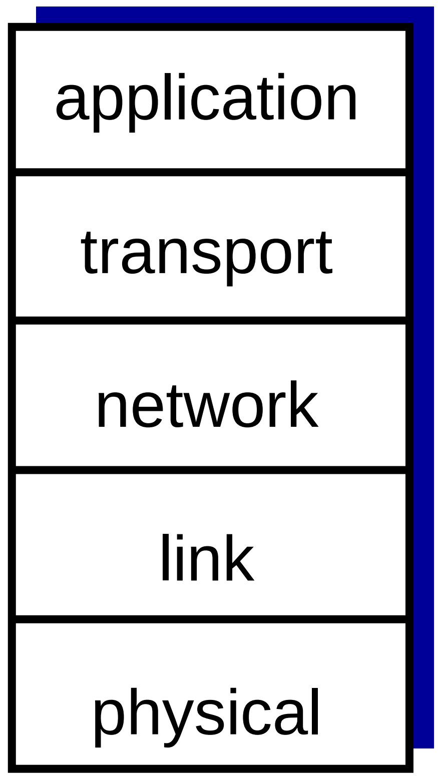
- via its own internal-layer actions
- relying on services provided by layer below

Why layering?

dealing with complex systems:

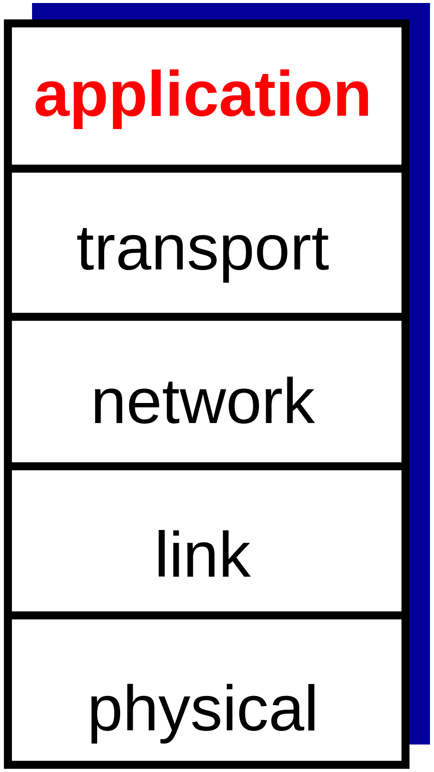
- explicit structure allows identification, relationship of complex system's pieces
  - layered *reference model* for discussion
- modularization eases maintenance, updating of system
  - change of implementation of layer's service transparent to rest of system
  - e.g., change in gate procedure doesn't affect rest of system
- layering considered harmful?

Internet protocol stack



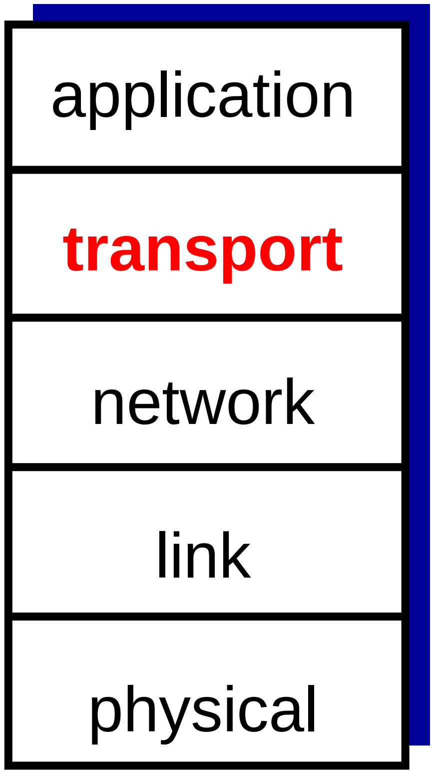
Application Layer

- *Responsibilities*
  - Exchange information between hosts
  - Data unit is called **message**
- Examples
  - HTTP
  - DNS
  - SMTP
  - FTP
  - BitTorrent



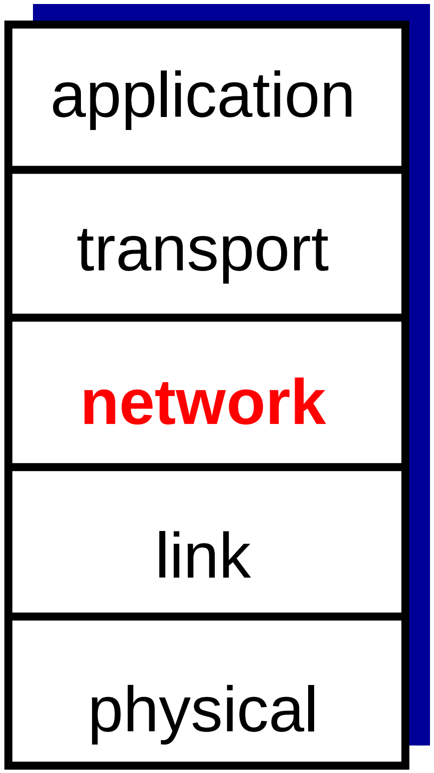
Transport Layer

- *Responsibilities*
  - Exchange packets between hosts
  - Error recovery
  - Congestion and flow control
  - Data unit is called **segment**
- Examples
  - TCP
  - UDP



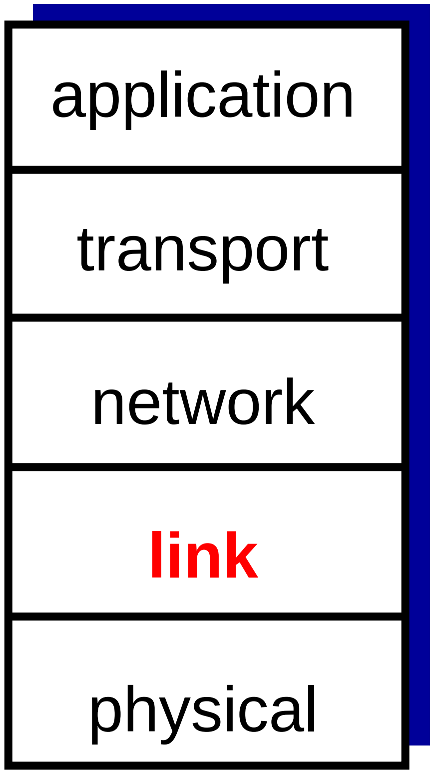
Network Layer

- *Responsibilities*
  - Route packets from source host to destination host
  - Data unit is called **datagram**
- Examples
  - IP
  - Routing table update: RIP, OSPF, BGP



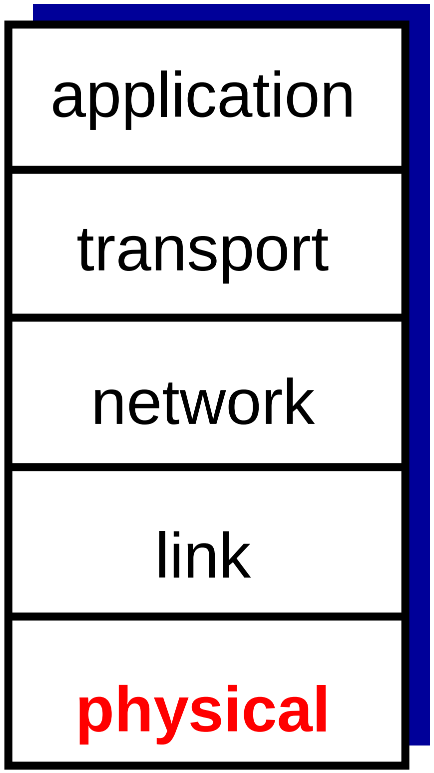
Link Layer

- *Responsibilities*
  - Deliver packets from one end of a transmission channel to the other (i.e., between two nodes [host or switching device])
  - Error recovery, flow control
  - Coordinate transmission channel sharing
  - Data unit is called **frame**
- Examples
  - Ethernet
  - WiFi



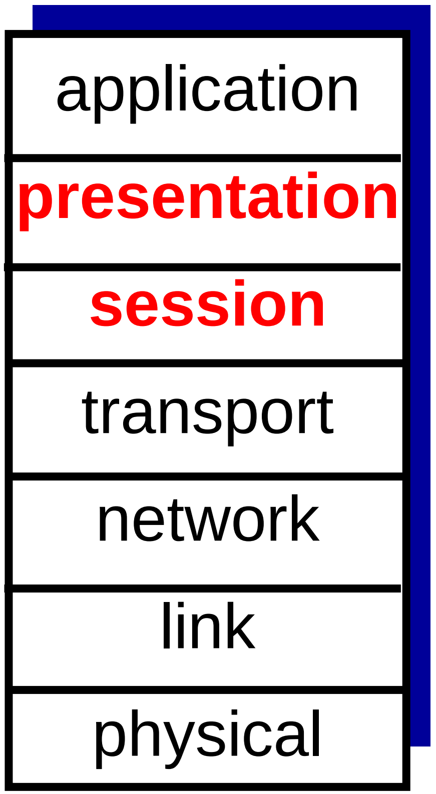
Physical Layer

- *Responsibilities*
- Define electrical or optical signals that represent bit sequences
- Data units are **bits**
- Examples
  - Cable Modem
  - ON-OFF Keying for fiber optics

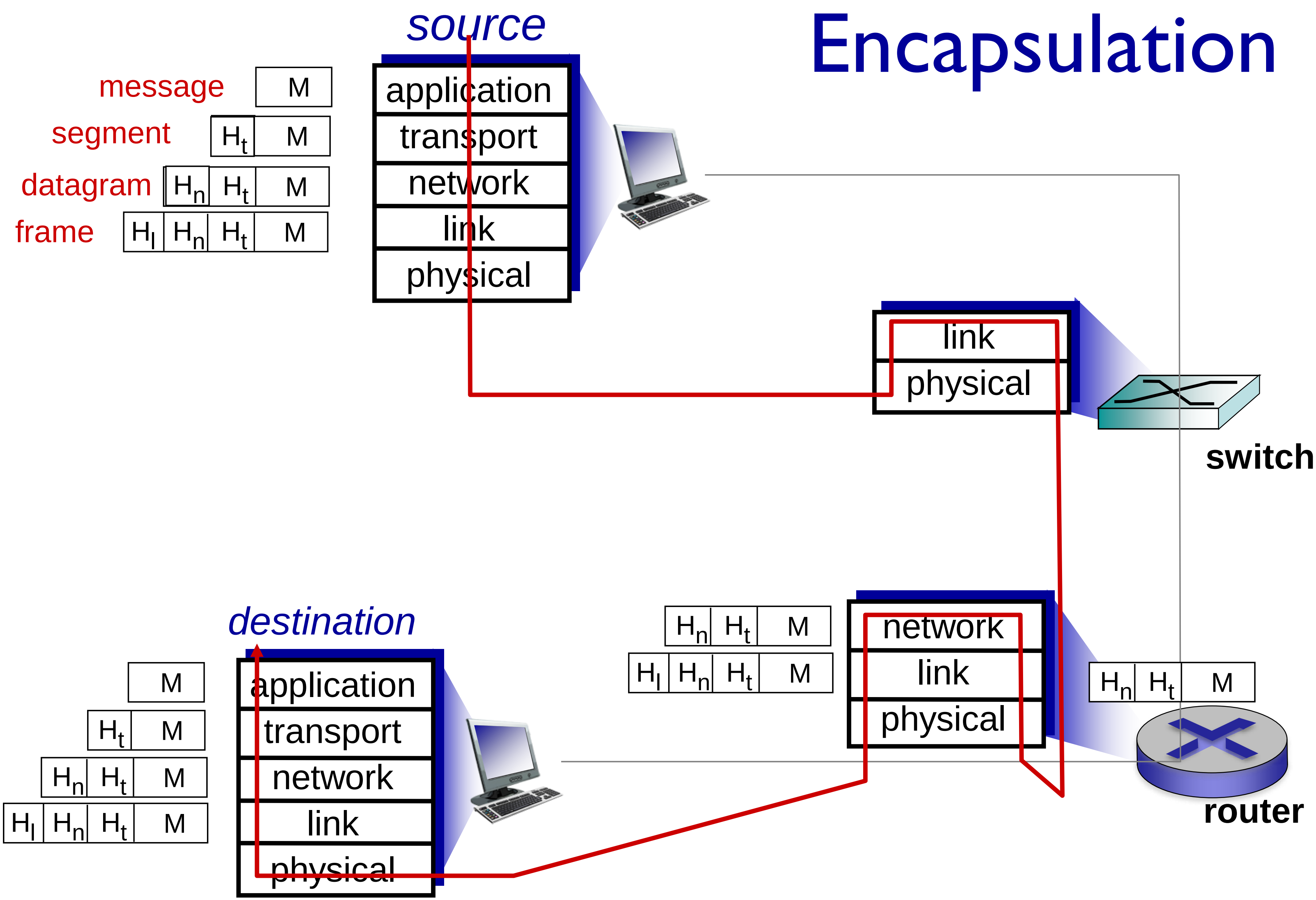


ISO/OSI reference model

- *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



Encapsulation



So Far…

- Internet overview
- What is a protocol?
- network edge, core, access network
  - packet-switching versus circuit-switching
  - Internet structure
- performance: loss, delay, throughput
- layering, service models

you now have:

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