Welcome to

***CS 3516**:*

Advanced Computer Networks

Prof. Yanhua Li

Time: 9:00am –9:50am M, T, R, and F
Location: AK219
Fall 2018 A-term

Lab assignment 1

We are grading it.

Quiz 1

Grades are available in Canvas.

Quiz 2

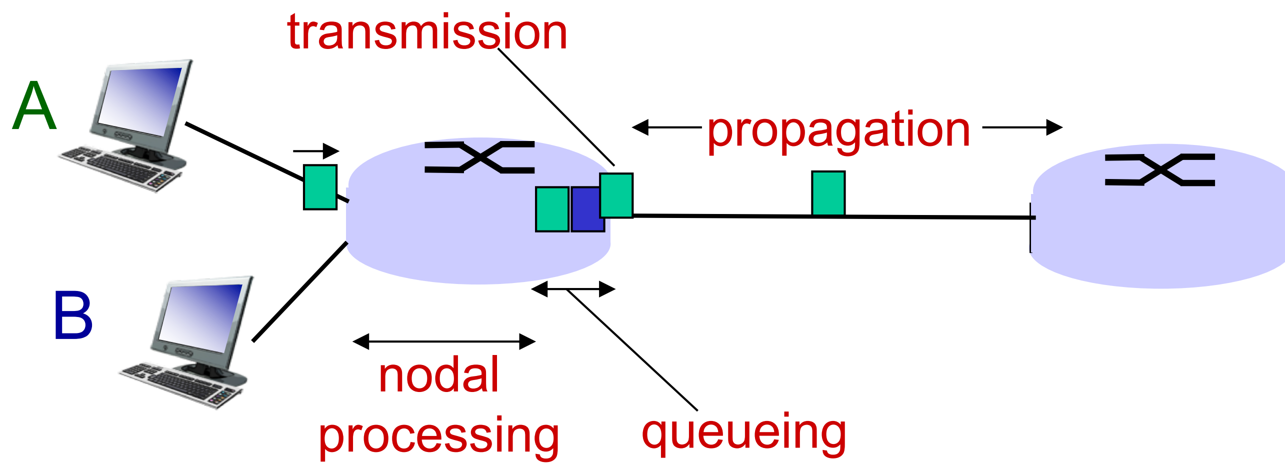
Friday tomorrow

Chapter 1: roadmap

1.4 network performance in packet-switched networks

- **delay**
- loss
- throughput

Four sources of packet delay



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

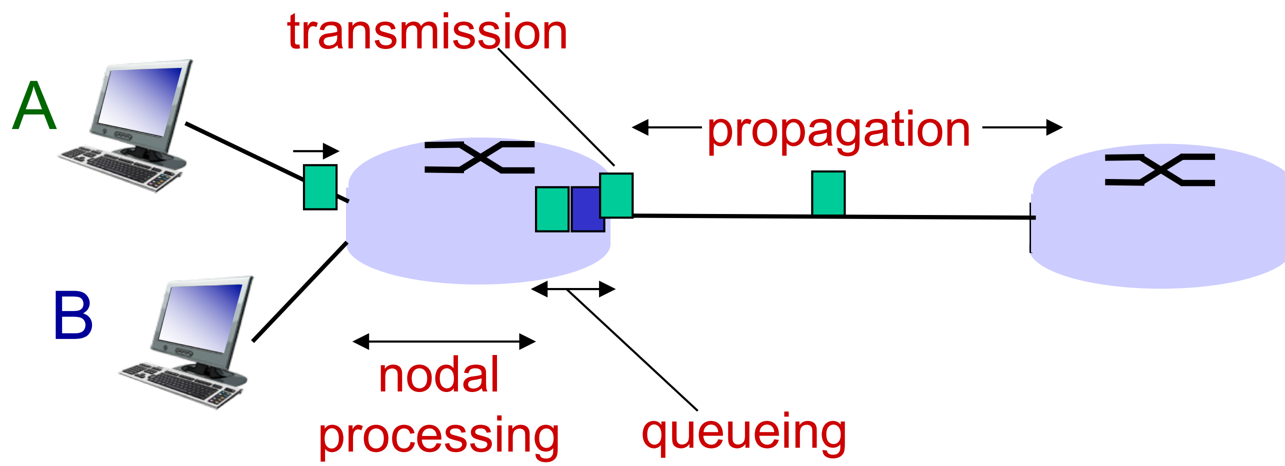
d_{proc}: nodal processing

- check bit errors
- determine output link
- typically < micro-sec

d_{queue}: queueing delay

- time waiting at output link for transmission
- depends on congestion level of router
- micro-sec to milli-sec

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:

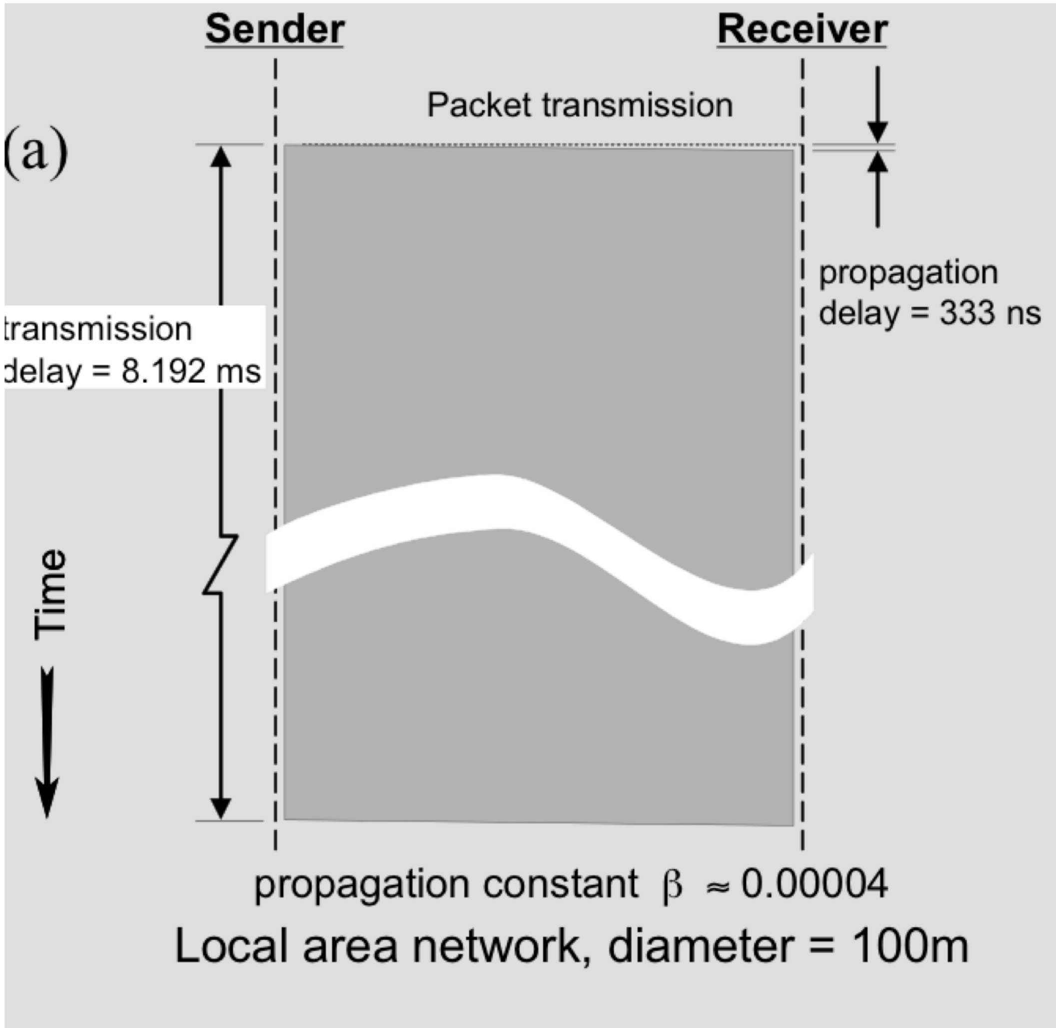
- *d*: length of physical link
- *s*: propagation speed in medium (~2x10[8] m/sec)
- $d_{prop} = d/s$

When a packet can be transmitted on a link?

- 1) no other pkt transmitted on the link (in practice)
- 2) no other pkt preceding it in the queue

Propagation constant β

Ratio of propagation delay vs. packet transmission time



Wireless LAN 0.00004.   Ethernet: 0.01

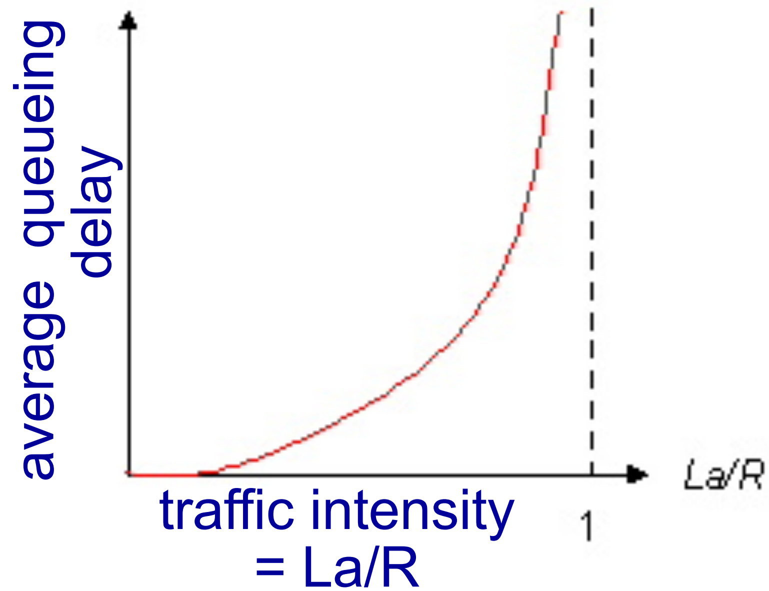
Chapter 1: roadmap

1.4 network performance in packet-switched networks
- delay
- loss
- throughput

1.5 protocol layers, service models
2.1 Overview of application layer

Queueing delay (revisited)

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

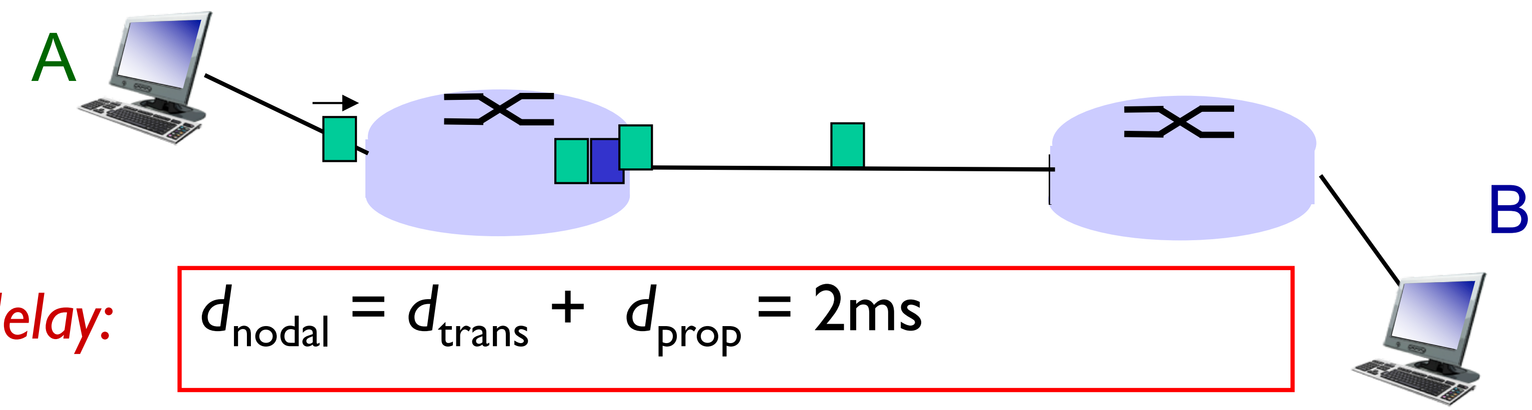


- *La/R: Traffic Intensity (pkts)*
- *La/R* ~ 0: avg. queueing delay small
- *La/R* -> 1: avg. queueing delay large
- *La/R* > 1: more "work" arriving than can be serviced, average delay infinite!


La/R ~ 0


La/R -> 1

End-to-end delay



Nodal delay: $d_{nodal} = d_{trans} + d_{prop} = 2\text{ms}$

Ten packets are sent from A to B:

Assumption: the time at which the 2nd packet is received at the first router is equal to the time at which the 1st packet is received at the second router.

Nodal delay: single packet end-to-end delay 6ms.

$$D_{\text{end-to-end, single}} = 3{*}d_{nodal} = 6 \text{ ms}$$

Nodal delay: ten packet end-to-end delay.

$$D_{\text{end-to-end, 10}} = 3{*}d_{nodal} + 9 * d_{nodal} = 24 \text{ ms}$$

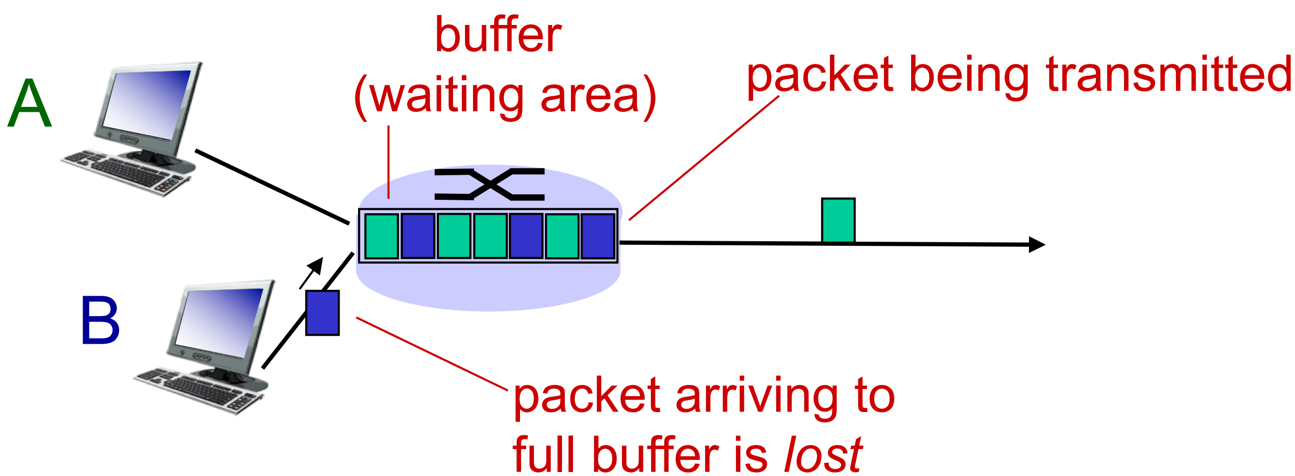
Chapter 1: roadmap

1.4 network performance in packet-switched networks

- delay
- **loss**
- throughput

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



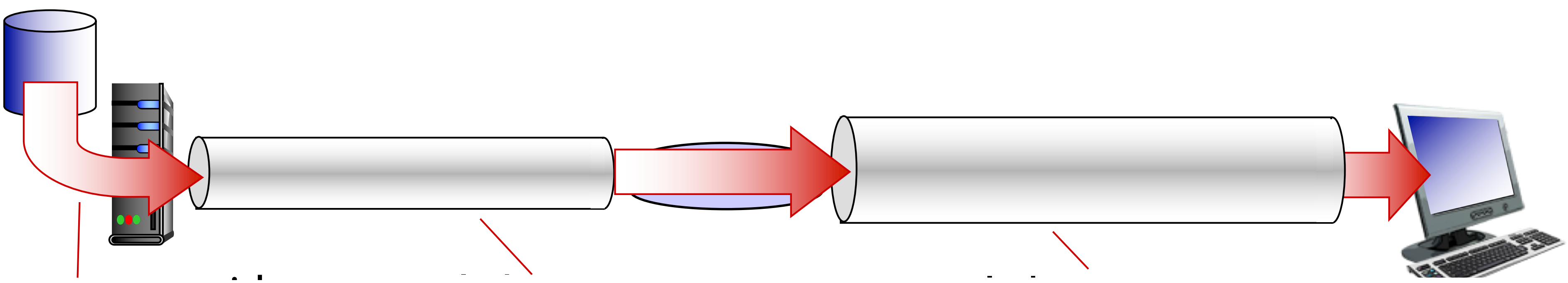
Chapter 1: roadmap

1.4 network performance in packet-switched networks

- delay
- loss
- **throughput**

Throughput (an end-to-end measure)

- *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

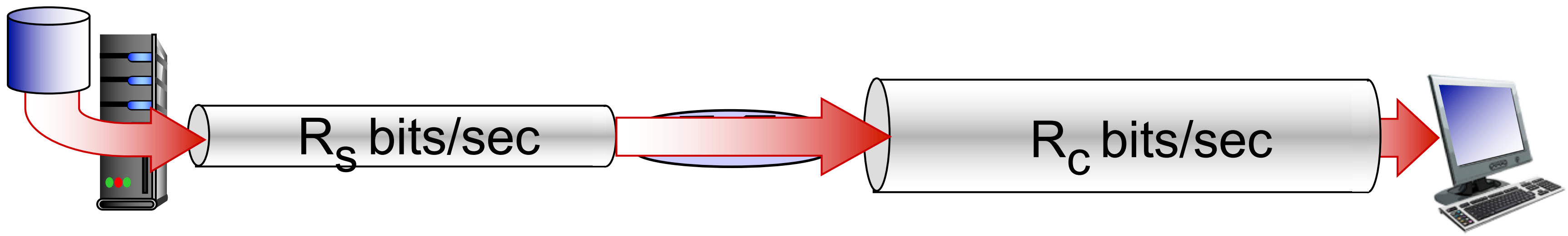
pipe that can carry fluid at rate R_s bits/sec)

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

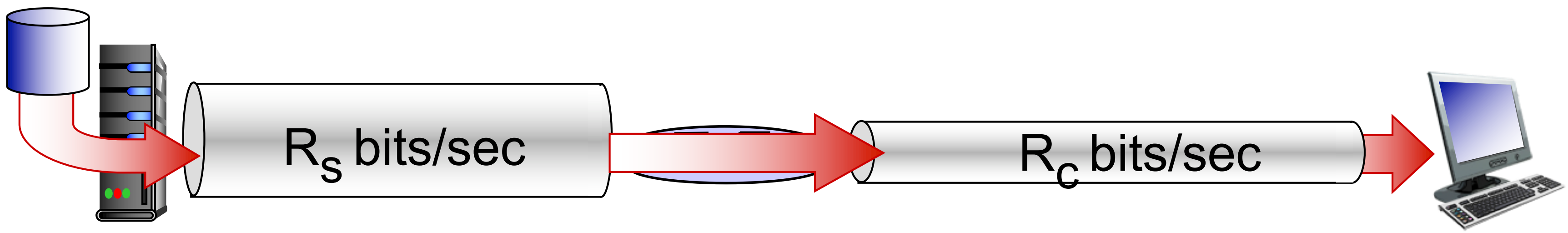
Throughput (more)

$\min(R_c, R_s)$

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



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



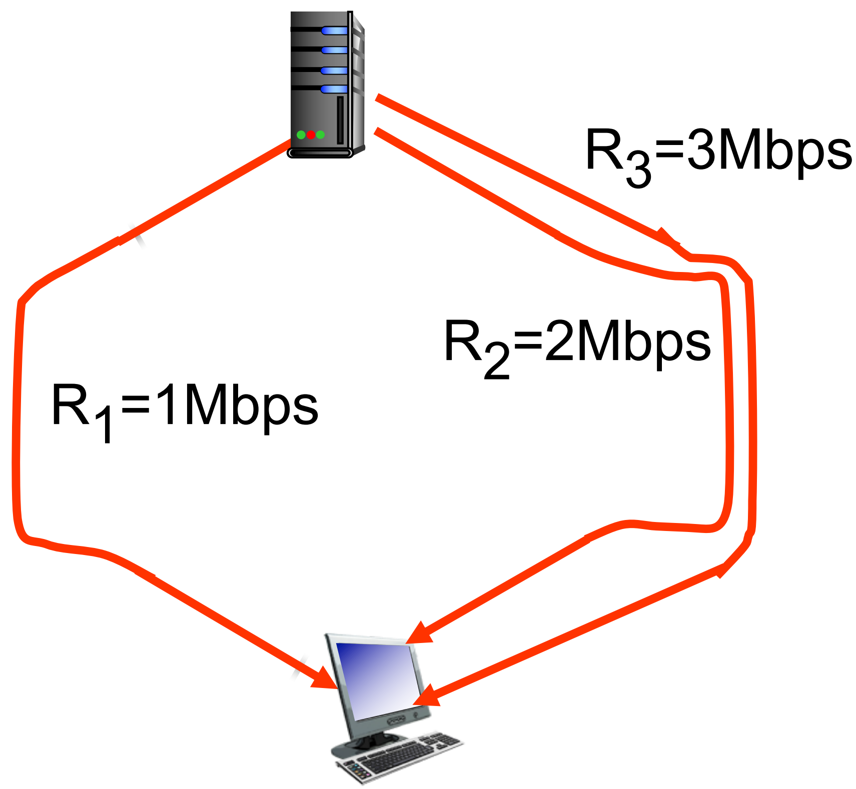
bottleneck link

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

Throughput

- per-connection end-end throughput:

$\max(R_1,R_2,R_3)=R_3=3\text{Mbps}$

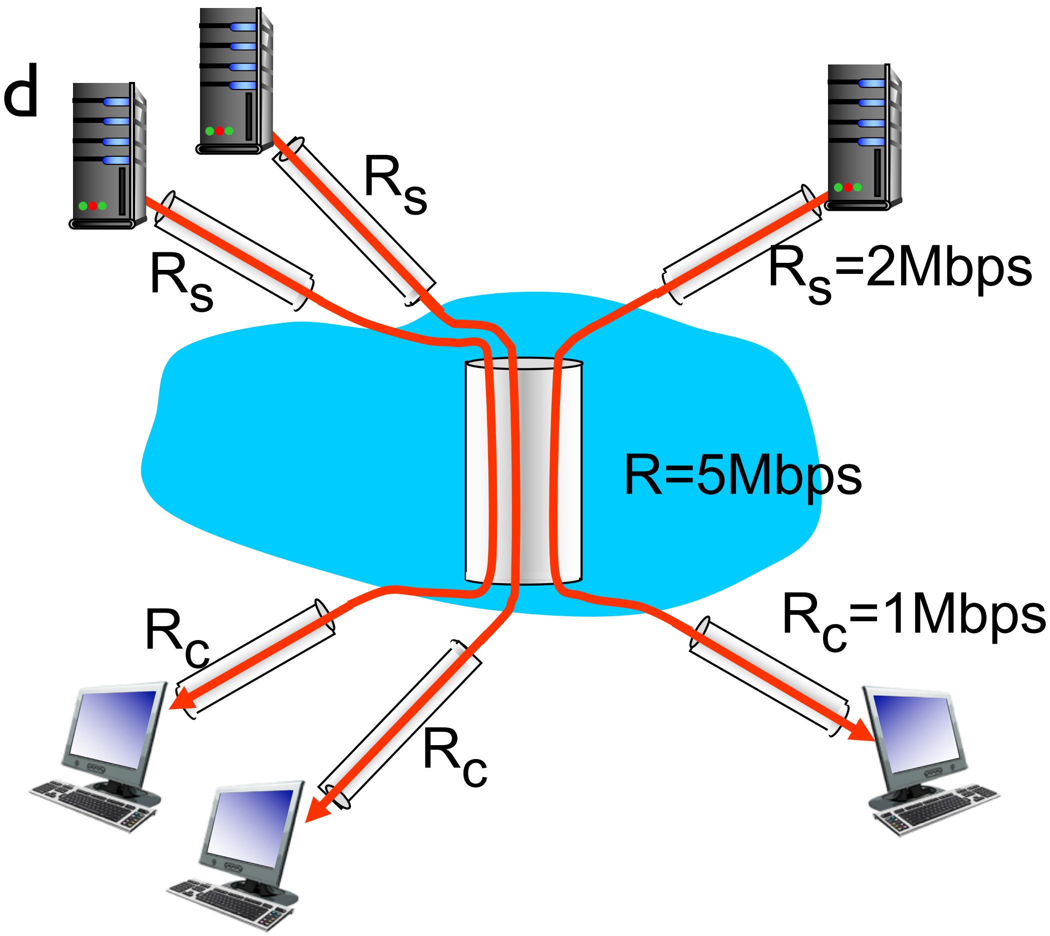


Throughput: Internet scenario

- per-connection end-end throughput:

$\min(R_c, R_s, R/10) = 500\text{kbps}$

- in practice: R_c or R_s is often bottleneck



10 connections (fairly) share backbone bottleneck link R bits/sec

Chapter 1: roadmap

1.4 delay, loss, throughput in networks
1.5 protocol layers, service models

protocol layers,
service models

Protocol "layers"

Networks are complex, with many "pieces":

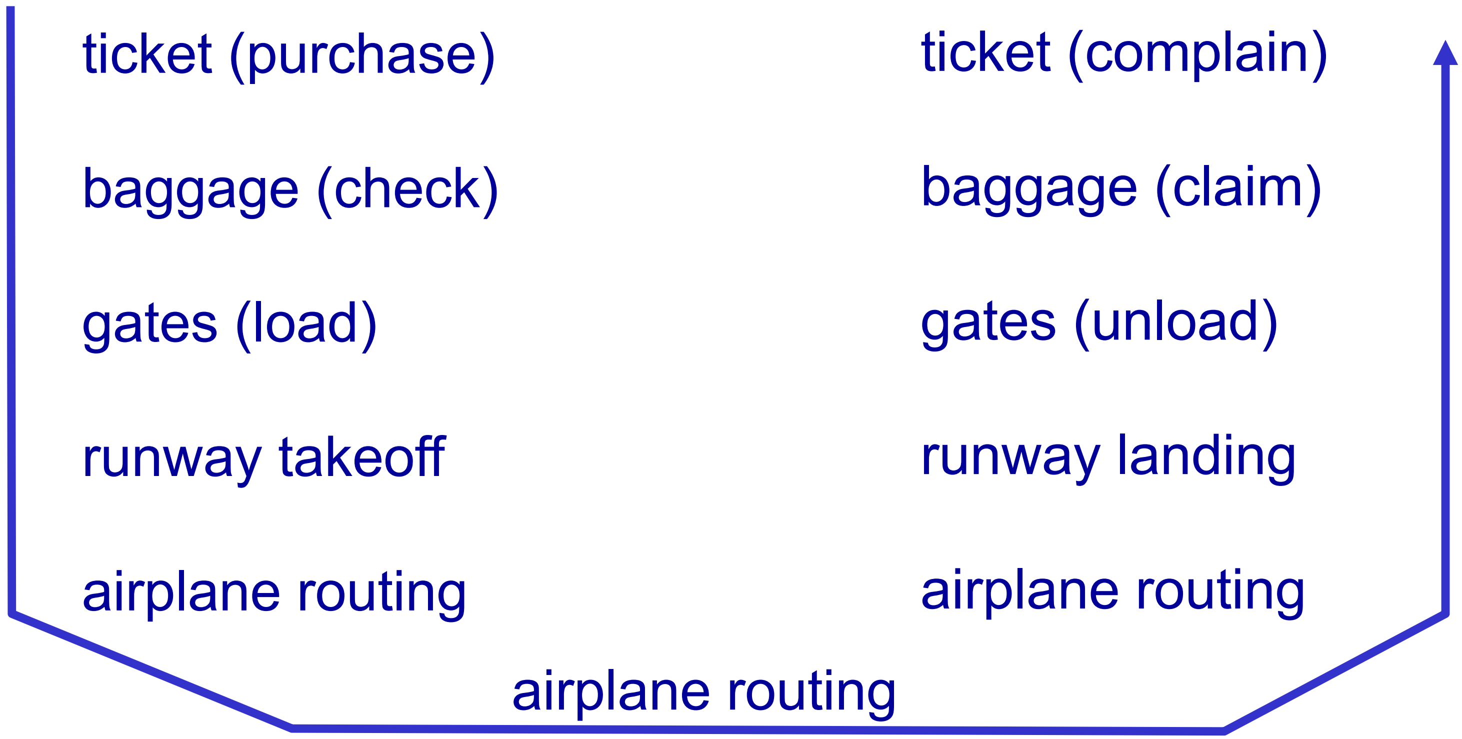
- hosts
- routers
- links of various media
- applications
- protocols
- hardware, software

Question:

is there any hope of *organizing* network processes in a structured way?

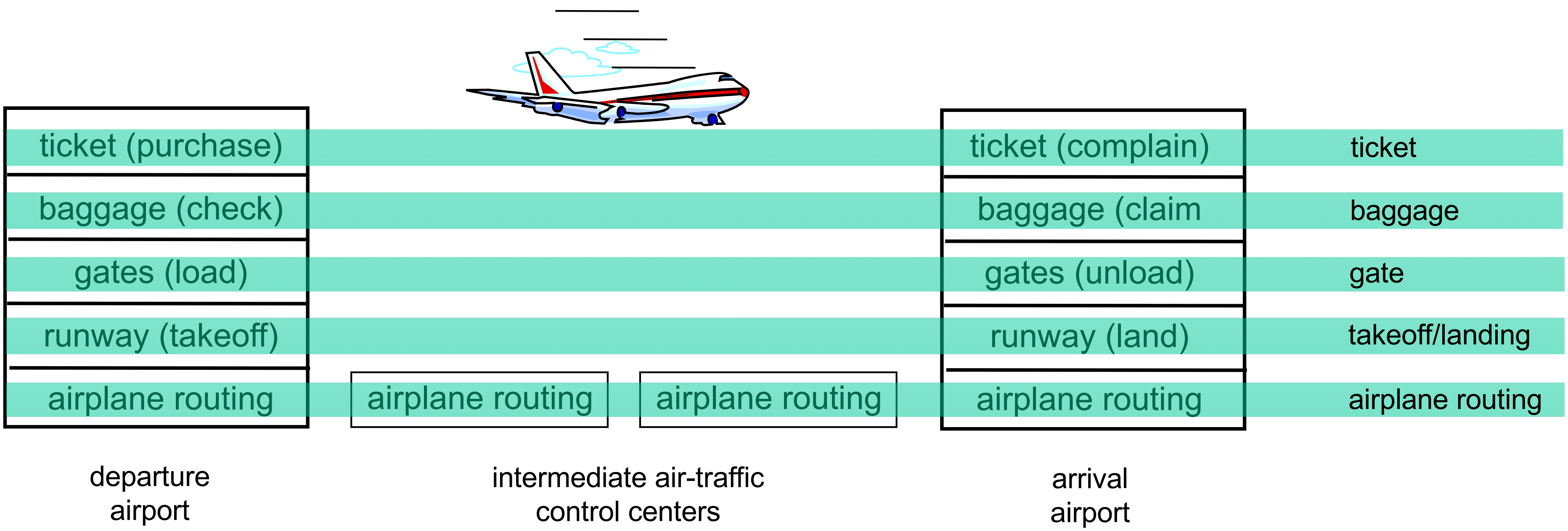
…. or at least our discussion of networks?

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

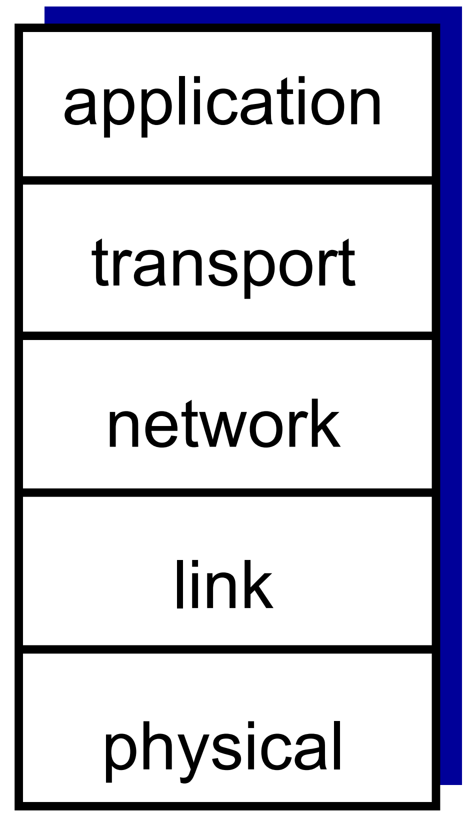
Why layering?

dealing with complex systems:

- Structured way to discuss the of complex system
  - 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?
  - Duplicated functionalities at different layers, like error check
  - Functionality at one layer needs info from other layers

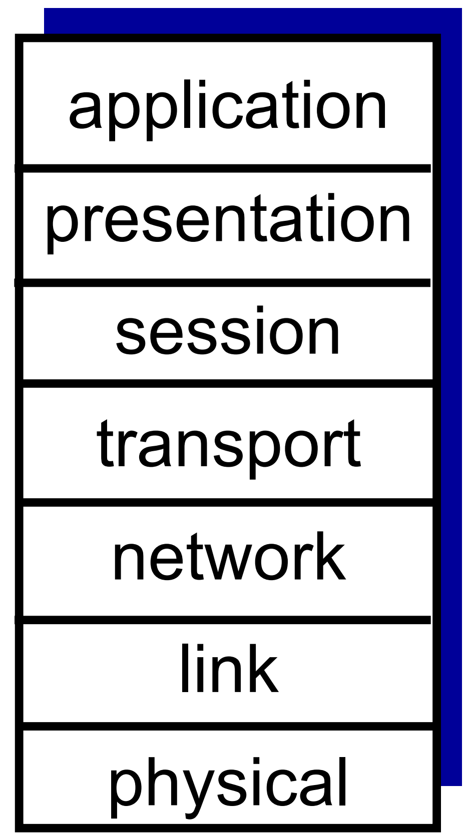
Internet protocol stack

- *application:* supporting network applications
  - FTP, SMTP, HTTP, DNS
- *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), PPP
- *physical:* bits "on the wire"

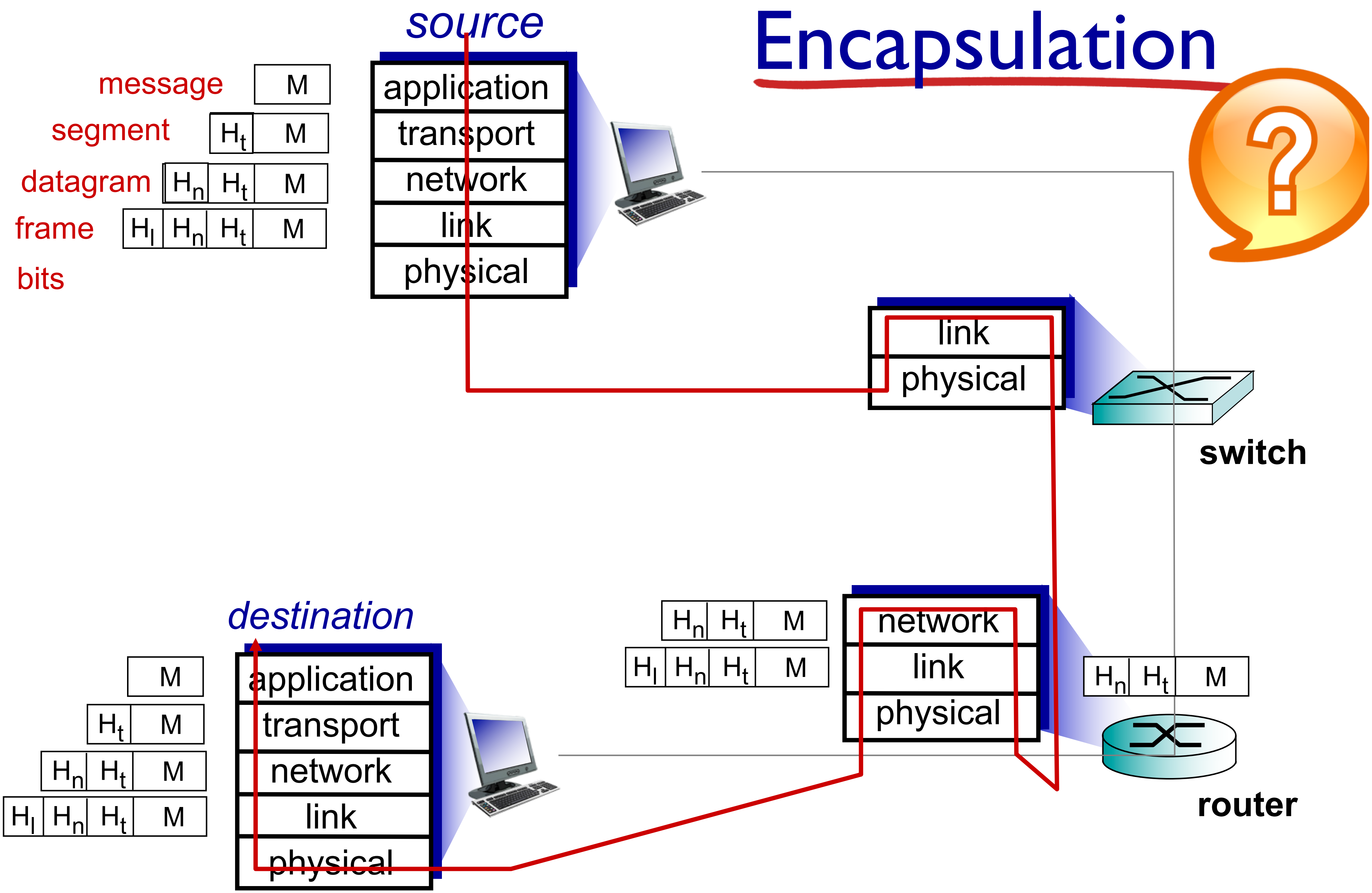


ISO/OSI reference model

- *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



Introduction: summary

covered a "ton" of material!

- S1: Internet overview
- S1: what's a protocol?
- S2: network edge, core, access network
  - packet-switching versus circuit-switching
  - Internet structure
- S3: Socket programming
- S4: performance: loss, delay, throughput
- S5: layering, service models

you now have:

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

Course Progression

- Week 1-2: Overview
- Week 2-4: Application Layer Protocols
  - P2P, HTTP, SMTP, DNS
- Week 4-5: Transport Layer Protocols
  - UDP and TCP
- Week 6: IP, Routing Protocols
- Week 7: Link Layer Protocols
- Week 8: Wireless & Data Center Networking
- Slides for the lecture will be posted on the website

Questions?