CSC 358 – Principles of Computer Networks

Handout # 2:
Course Logistics and Introduction

Joe Lim

Department of Mathematical and Computational Sciences

joe.lim@utoronto.ca

Today

- Outline
  - What this course is about
- Logistics
  - Course structure, assignments, evaluation
  - What is expected from you
  - What you can expect from this course
- Review
  - Simple example – mail vs. FTP
- Foundations and basic concepts

What is This Course About?

- Computer networks
  - Basics: Layers, naming, and addressing, network (socket) programming, packet switching, routing, congestion control, …
  - Advanced networking: HTTP, web, peer-to-peer, routers and switches, security, multimedia, online social networks, software-defined networking, …

Logistics – Prerequisites, Readings

- Prerequisites
  - Algorithms
  - Basic probability theory
  - Strong background in *C* programming and Unix environment
  - CSC209H5, 258H5, 263H5, 290H5
- Readings
  - Will be posted on Blackboard
  - Read before class
- Textbook
  - James Kurose and Keith Ross. Computer Networking: *A Top-Down Approach (7th ed)*. Addison Wesley, 2017. 9780133594140

Logistics – Hours, Announcements, TA

- Office hours
  - DH 3095, Wednesdays, 11:30AM – 12:30PM
  - Email: joe.lim@utoronto.ca [Prefix subject line with: CSC358]

- Please check the class web page on Blackboard, and the bulletin board regularly for announcements.

- Teaching Assistant
  - Zhancheng Song

Logistics – Mailing List, Bulletin Board

- Bulletin board
  - Stay tuned... Bulletin Board to be selected
  - Post any questions related to the course.
  - Check previous posts before asking a question.
  - We guarantee to respond within 48 hours.

Logistics – Grading

- Grading
  - Assignments: 50%
    - Problem sets: 20%
    - Programming: 30%
  - Midterm exam: 20% - Rooms TBA
  - Final exam: 30% - TBA

Logistics - Deadlines

- Assignment deadlines
  - 10% deduction for each day late
  - Up to 20%
  - Assignment not accepted after two days (48 hours)
- Special accommodations
  - You will need to submit the proper form and documentation to your instructor

Logistics – Programming Assignments

- To be completed groups of 2.

- You can submit your assignment before the deadline
  - And get a feedback from the autotester within 10-20 mins.
  - **Your last submission before the deadline will be marked.**

- Socket Programming

- MiniNet
  - Your very own virtual network!
  - You will create and program your own network
  - VM available on DH2020 machines
  - More detail on this later.

- This is a *heavy* course, but *manageable*!

Logistics – Academic Integrity

- Academic Integrity
  - All submissions must present original, independent work.
  - We take academic offenses very seriously.
  - Please read
    - Handout # 1 (course information sheet)
    - “Guideline for avoiding plagiarism”
    - http://www.cs.toronto.edu/~fpitt/documents/plagiarism.html
    - “Advice about academic offenses”
    - http://www.cs.toronto.edu/~clarke/acoffences/

Logistics - Accessibility

- Accessibility Needs
  - The University of Toronto Mississauga is committed to accessibility. If you require accommodations or have any accessibility concerns, please visit http://www.utm.utoronto.ca/accessability/ as soon as possible.

Acknowledgements

- Special thanks to:
  - Nick McKeown from Stanford University
  - Jennifer Rexford from Princeton University
  - David Wetherall from University of Washington
  - Nick Feamster from Georgia Tech
  - Yashar Ganjali from University of Toronto

Quick Survey

- Have you taken CSC358 before?
- Have you taken any networking course?
- Are you familiar with
  - Socket programming?
  - Ethernet, framing, encoding, error detection/correction?
  - UDP, TCP and congestion control?
  - DNS, SNMP, BGP?
  - BitTorrent?
  - Voice and video over IP?
  - Control plane vs. data path?
  - Network security?
  - Software-defined networking?

Questions?

What else do you want to know about this course?

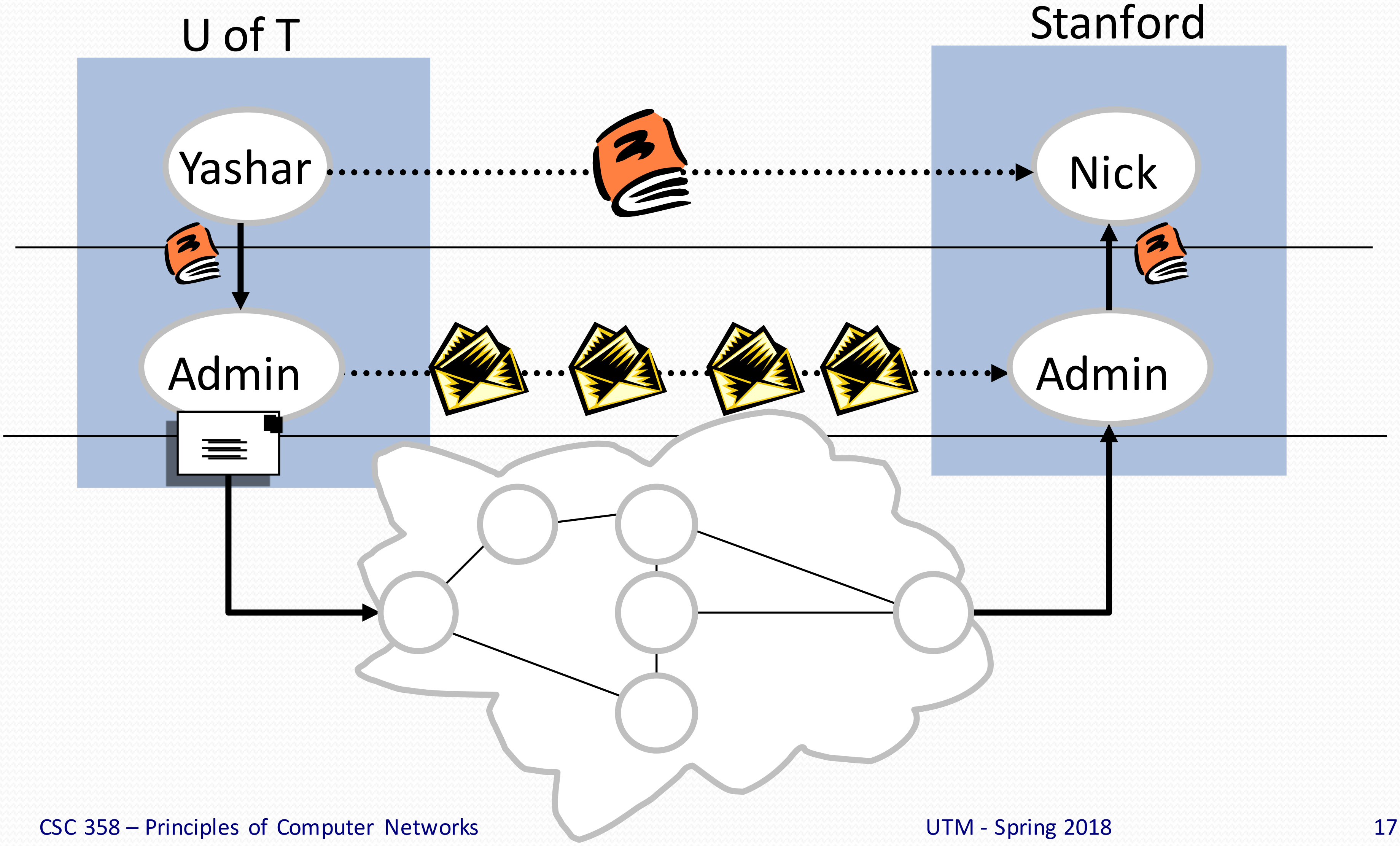
Announcement

- Tutorial Starts Next Week (January 12th)
- PA1, PA2, PS1, PS2 will be posted on Blackboard end of this week
- Start early!
- Form your group – use the discussion board to look for a team mate.

Let’s Begin

- An introduction to the mail system
- An introduction to the Internet

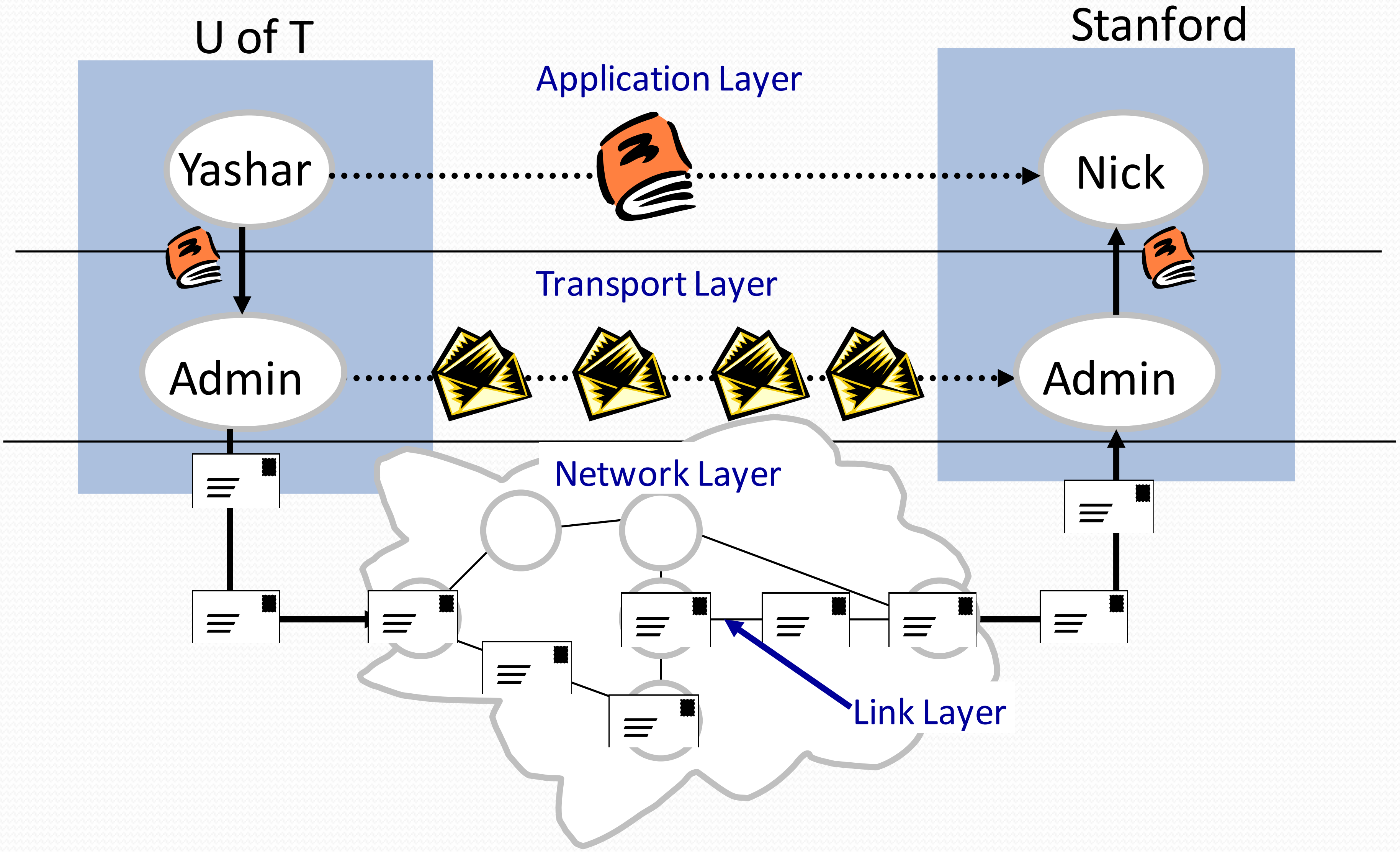
An Introduction to the Mail System



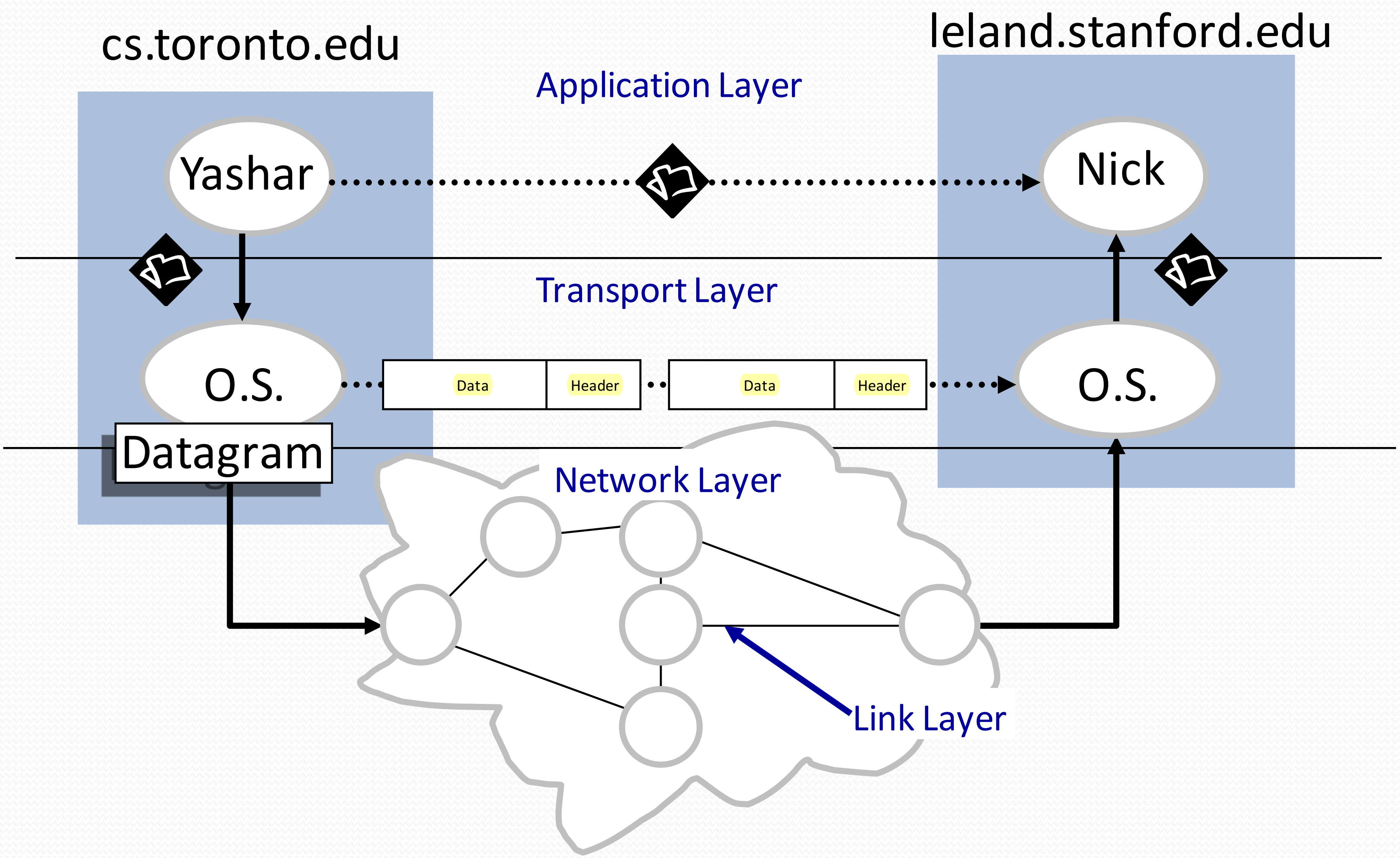
Characteristics of the Mail System

- Each envelope is individually routed.
- No time guarantee for delivery.
- No guarantee of delivery in sequence.
- No guarantee of delivery at all!
  - Things get lost
  - How can we acknowledge delivery?
  - Retransmission
    - How to determine when to retransmit? Timeout?
    - Need local copies of contents of each envelope.
    - How long to keep each copy.
    - What if an acknowledgement is lost?

An Introduction to the Mail System



An Introduction to the Internet



Characteristics of the Internet

- Each packet is individually routed.
- No time guarantee for delivery.
- No guarantee of delivery in sequence.
- No guarantee of delivery at all!
  - Things get lost
  - Acknowledgements
  - Retransmission
    - How to determine when to retransmit? Timeout?
    - Need local copies of contents of each packet.
    - How long to keep each copy?
    - What if an acknowledgement is lost?

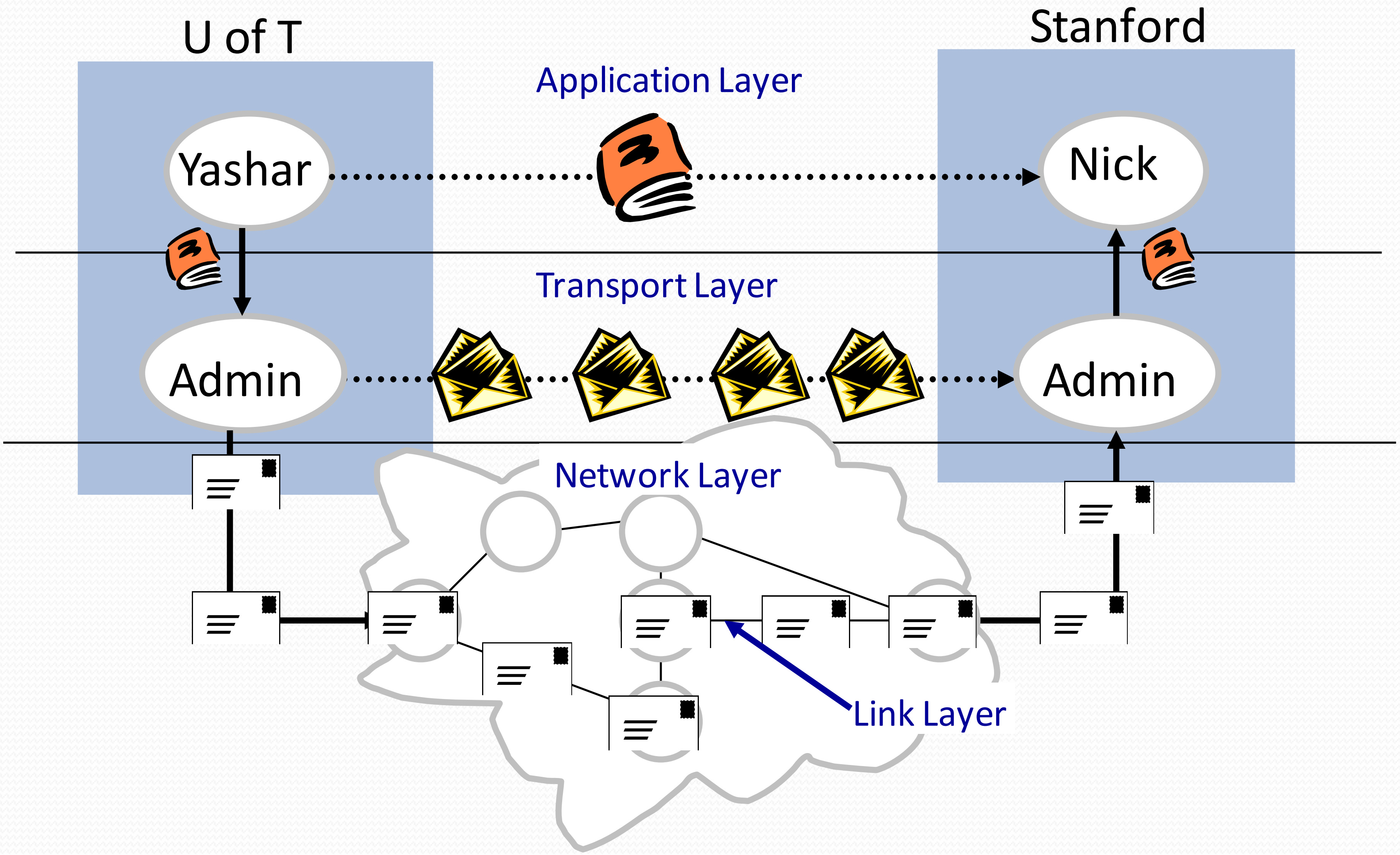
Characteristics of the Internet – Cont'd

- No guarantee of integrity of data.
- Packets can be fragmented.
- Packets may be duplicated.

Layering in the Internet

- Transport Layer
  - Provides reliable, in-sequence delivery of data from end-to-end on behalf of application.
- Network Layer
  - Provides “best-effort”, but unreliable, delivery of datagrams.
- Link Layer
  - Carries data over (usually) point-to-point links between hosts and routers; or between routers and routers.

An Introduction to the Mail System



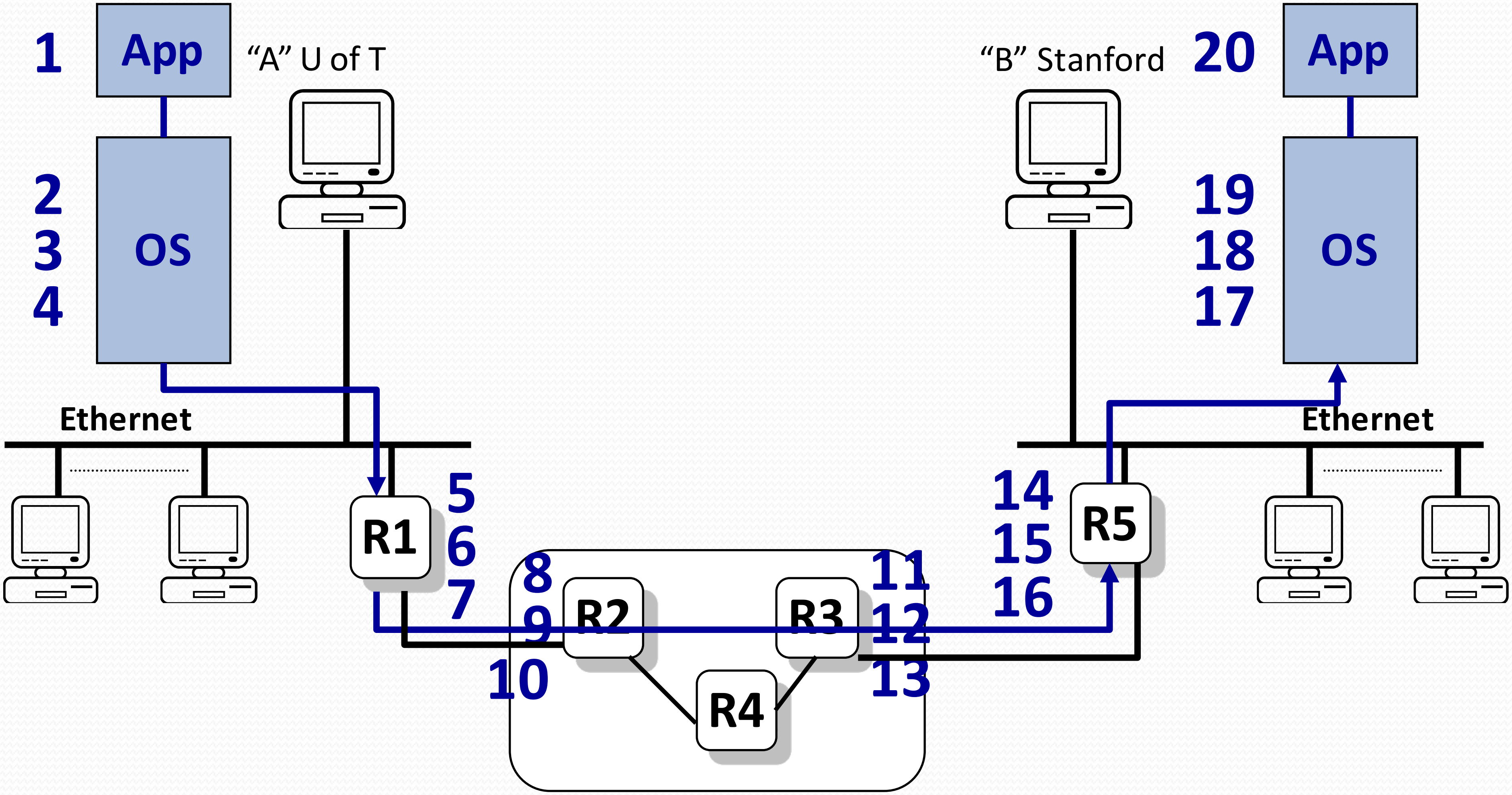
Some Questions About the Mail System

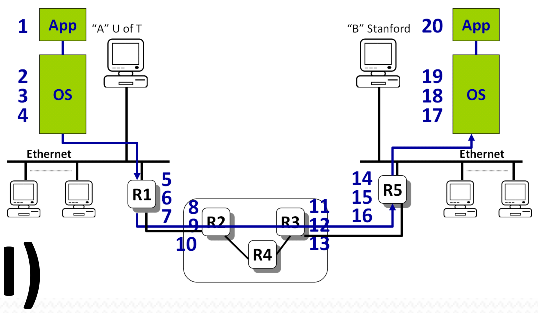
- How many sorting offices are needed and where should they be located?
- How much sorting capacity is needed?
  - Should we allocate for Mother’s Day?
- How can we guarantee timely delivery?
  - What prevents delay guarantees?
  - Or delay variation guarantees?
- How do we protect against fraudulent mail deliverers, or fraudulent senders?

Outline – Foundations & Basic Concepts

- A detailed FTP example
- Layering
- Packet switching and circuit switching

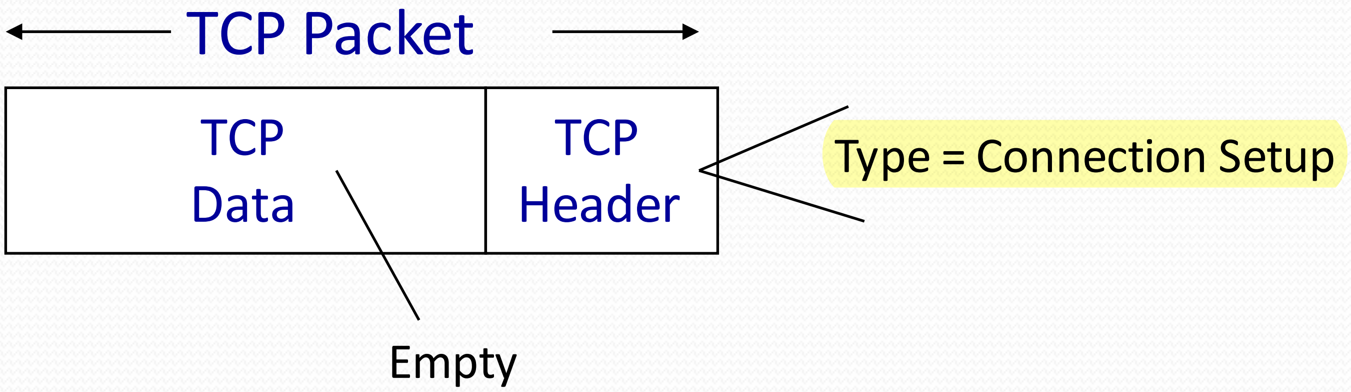
Example: FTP over the Internet Using TCP/IP and Ethernet





In the Sending Host

1. **Application-Programming Interface (API)**
   - Application requests TCP connection with "B"
2. **Transmission Control Protocol (TCP)**
   - Creates TCP "Connection setup" packet
   - TCP requests IP packet to be sent to "B"



TCP Packet

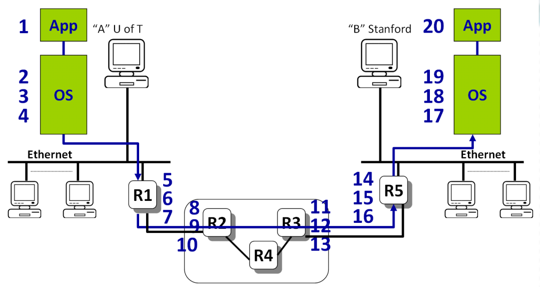
TCP Data

TCP Header

Type = Connection Setup

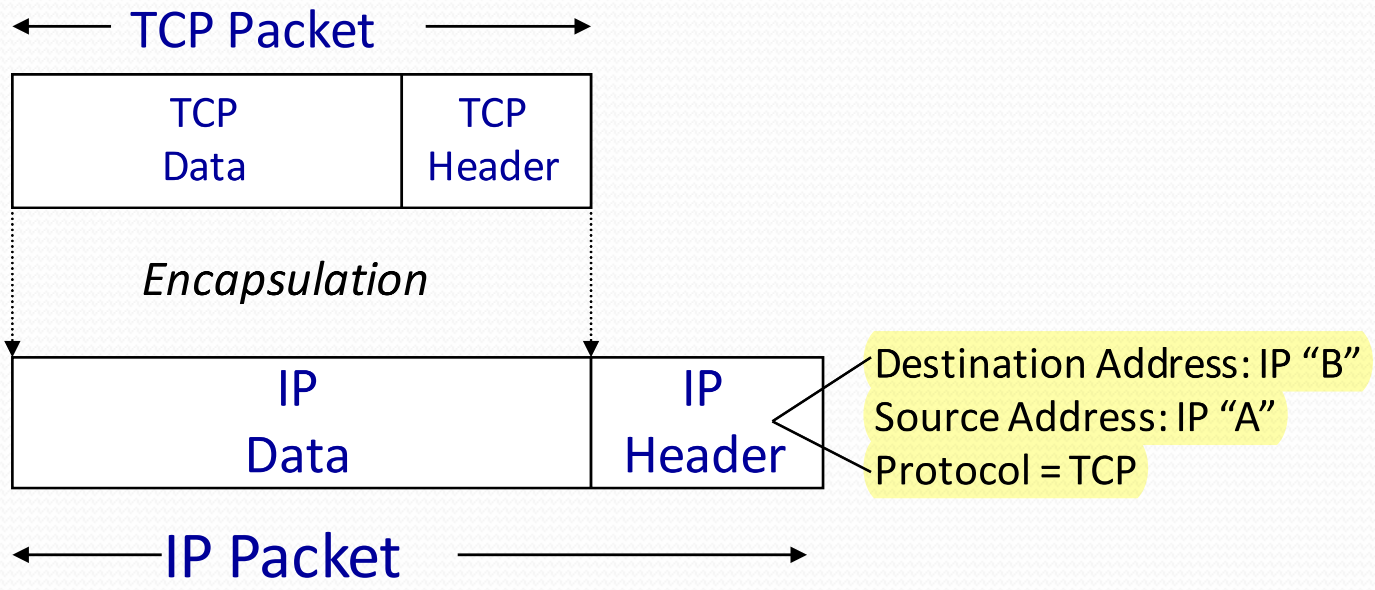
Empty

In the Sending Host – Cont’d



3. Internet Protocol (IP)

- Creates IP packet with correct addresses.
- IP requests packet to be sent to router.



TCP Packet

TCP Data

TCP Header

Encapsulation

IP Data

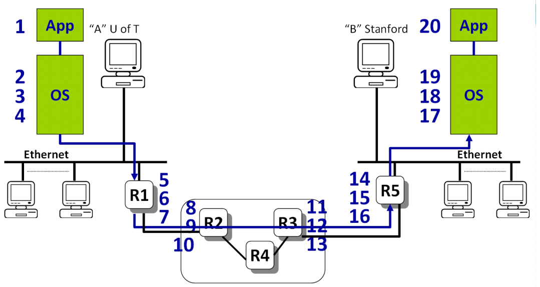
IP Header

Destination Address: IP “B”

Source Address: IP “A”

Protocol = TCP

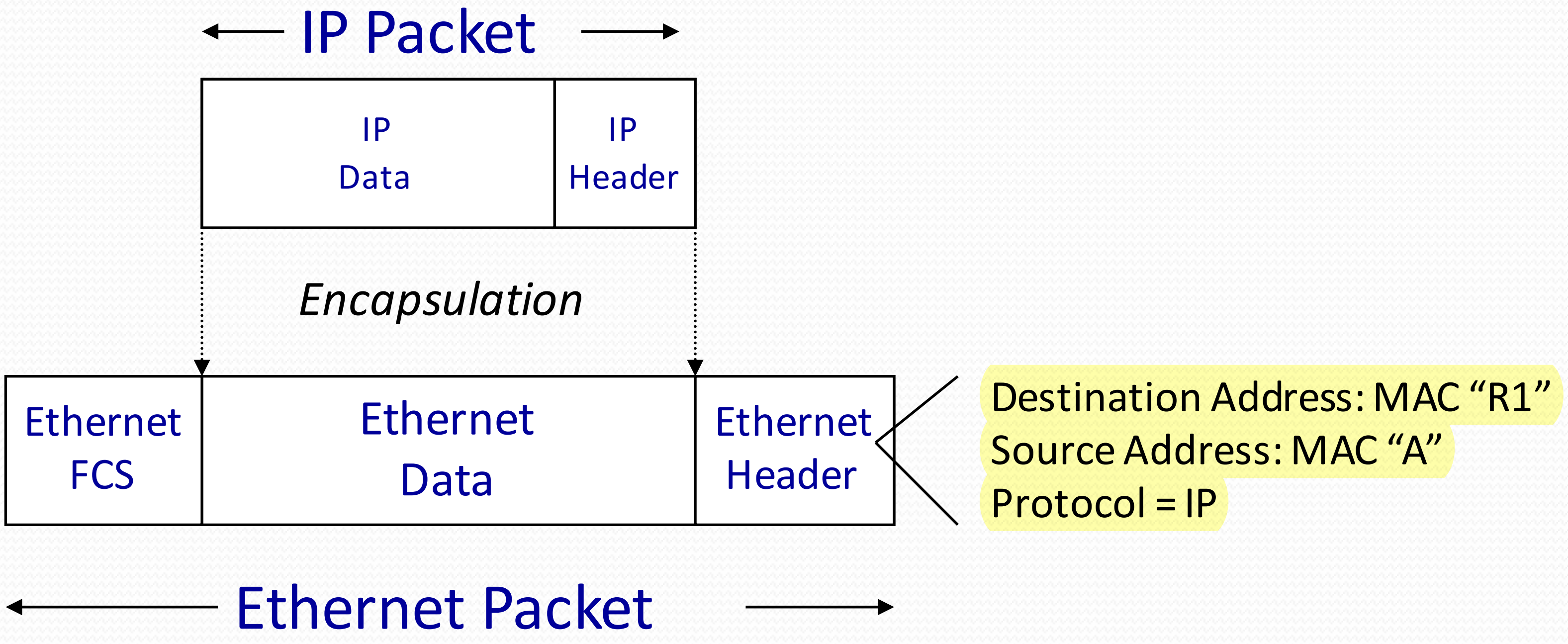
IP Packet



In the Sending Host – Cont'd

4. Link ("MAC" or Ethernet) Protocol

- Creates MAC frame with Frame Check Sequence (FCS).
- Wait for Access to the line.
- MAC requests PHY to send each bit of the frame.


IP Packet

IP Data | IP Header

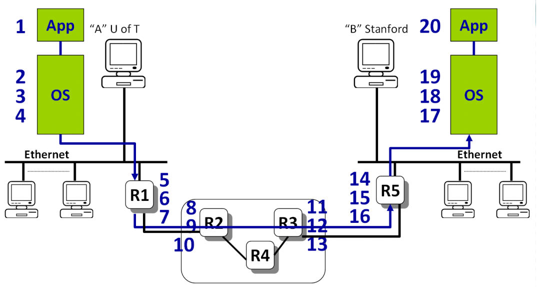
Encapsulation

Ethernet FCS | Ethernet Data | Ethernet Header

Destination Address: MAC "R1"
Source Address: MAC "A"
Protocol = IP

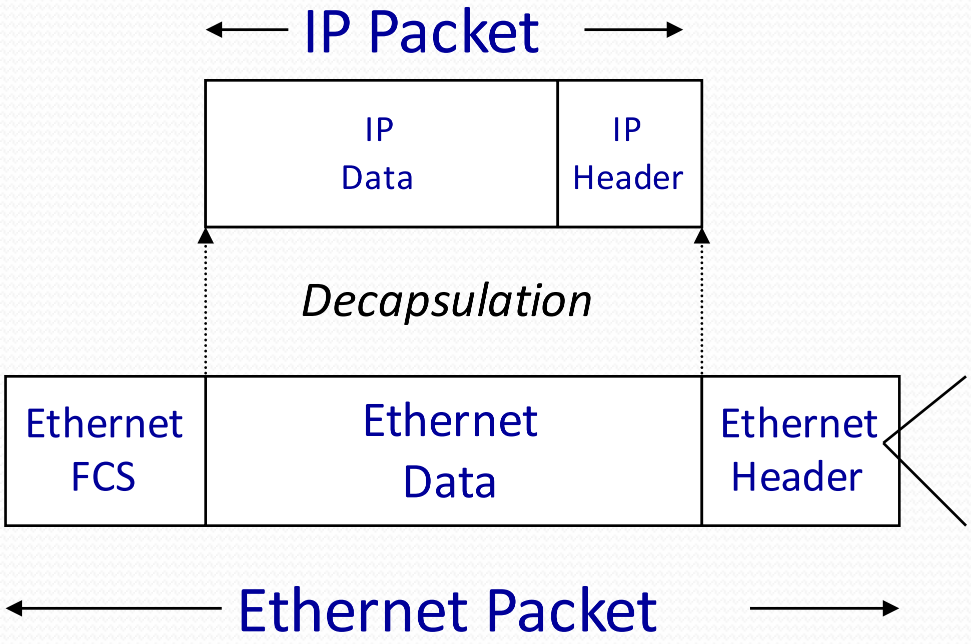
Ethernet Packet

In Router R1



5. Link ("MAC" or Ethernet) Protocol

- Accept MAC frame, check address and Frame Check Sequence (FCS).
- Pass data to IP Protocol.


IP Packet

IP Data | IP Header

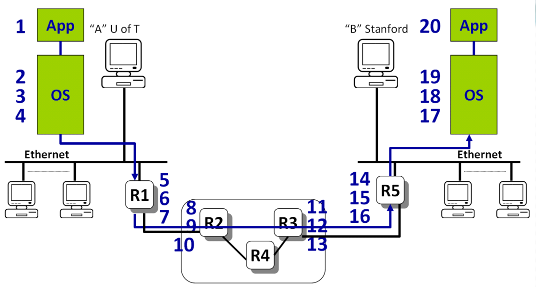
Decapsulation

Ethernet FCS | Ethernet Data | Ethernet Header

Ethernet Packet

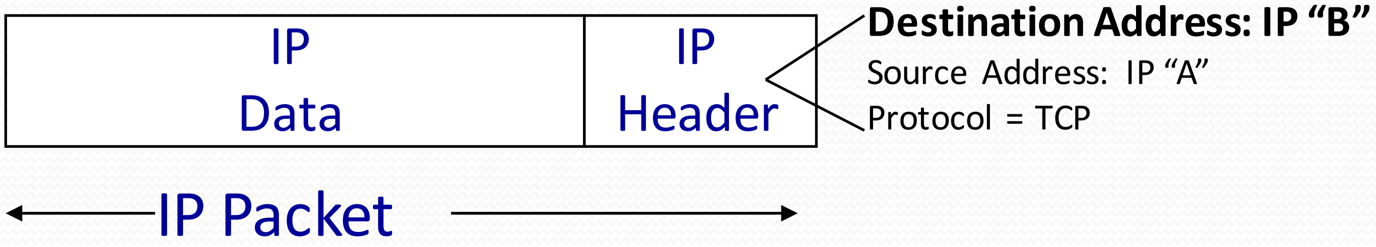
Destination Address: MAC "R1"
Source Address: MAC "A"
Protocol = IP

In Router R1



6. Internet Protocol (IP)

- Use IP destination address to decide where to send packet next ("next-hop routing").
- Request Link Protocol to transmit packet.

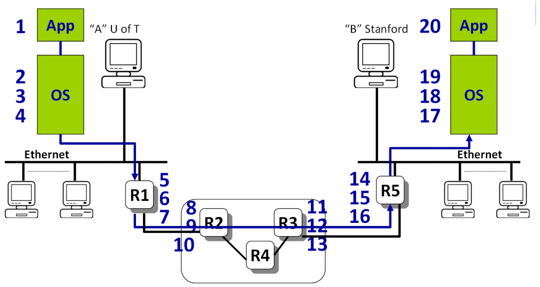


Destination Address: IP "B"

Source Address: IP "A"

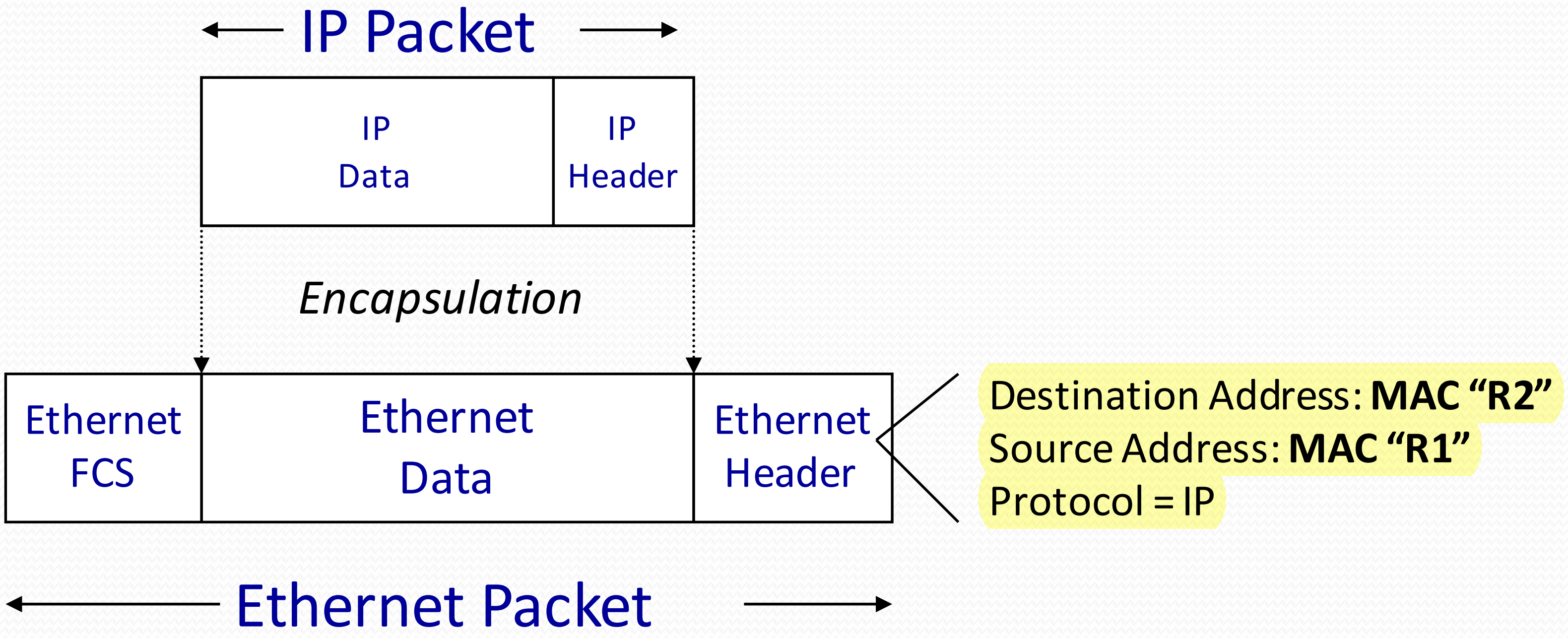
Protocol = TCP

In Router R1



7. Link ("MAC" or Ethernet) Protocol

- Creates MAC frame with Frame Check Sequence (FCS).
- Wait for Access to the line.
- MAC requests PHY to send each bit of the frame.



IP Packet

IP Data

IP Header

Encapsulation

Ethernet FCS

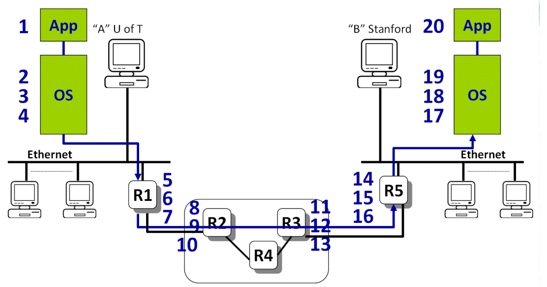
Ethernet Data

Ethernet Header

Destination Address: **MAC "R2"**
Source Address: **MAC "R1"**
Protocol = IP

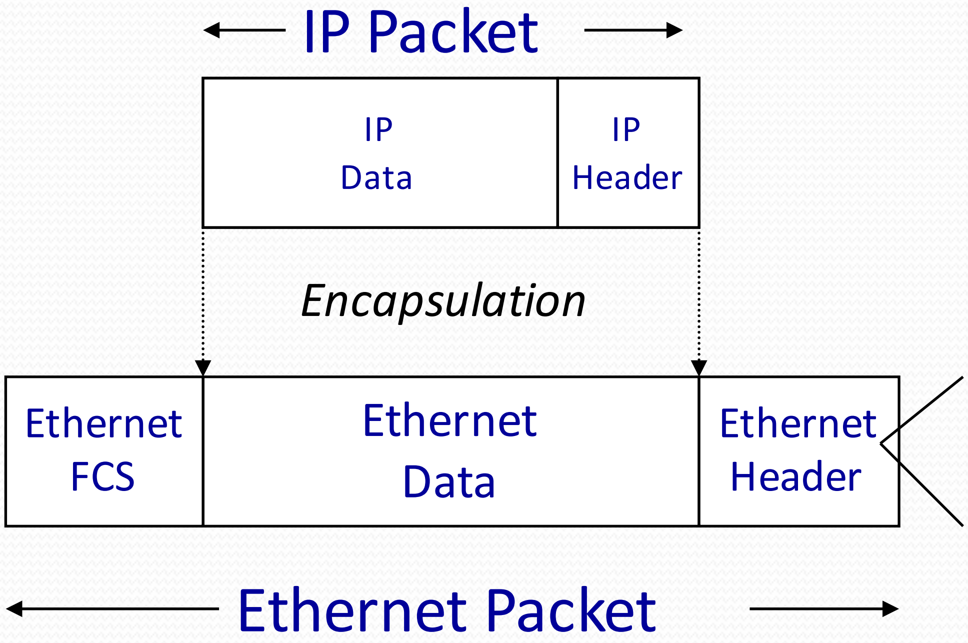
Ethernet Packet

In Router R5



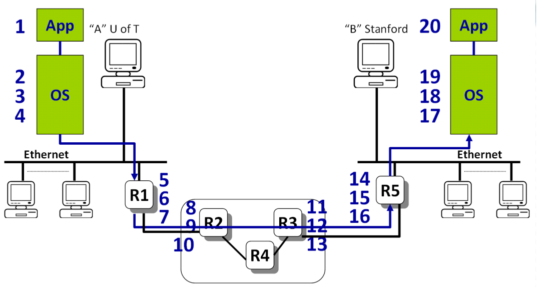
16. Link ("MAC" or Ethernet) Protocol

- Creates MAC frame with Frame Check Sequence (FCS).
- Wait for Access to the line.
- MAC requests PHY to send each bit of the frame.



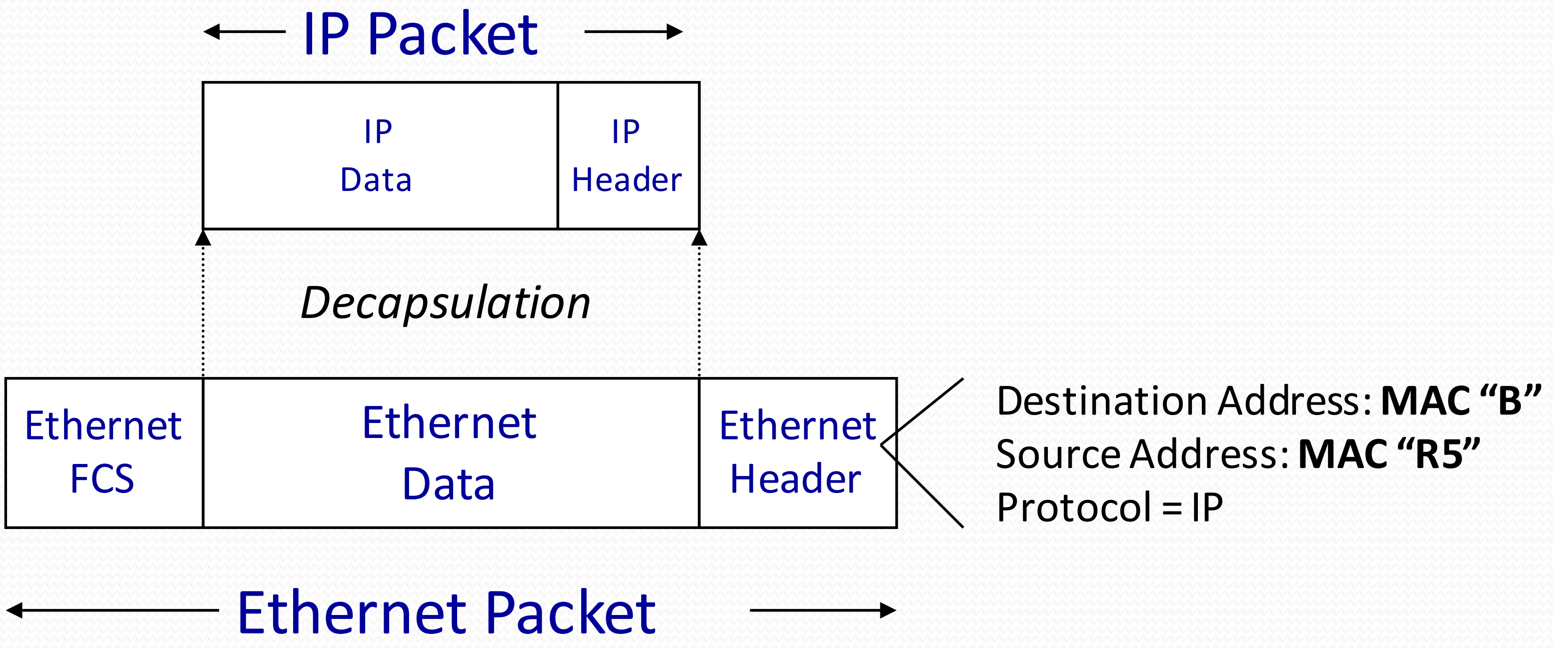
Destination Address: **MAC "B"**
Source Address: **MAC "R5"**
Protocol = IP

In the Receiving Host



17. Link ("MAC" or Ethernet) Protocol

- Accept MAC frame, check address and Frame Check Sequence (FCS).
- Pass data to IP Protocol.



IP Packet

IP Data | IP Header

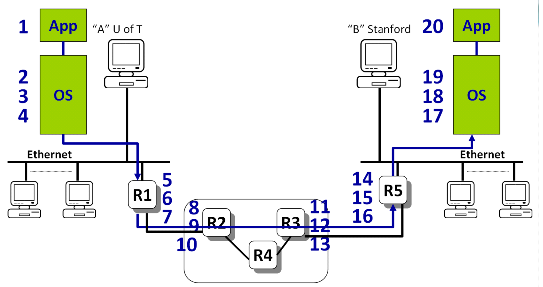
Decapsulation

Ethernet FCS | Ethernet Data | Ethernet Header

Destination Address: **MAC "B"**
Source Address: **MAC "R5"**
Protocol = IP

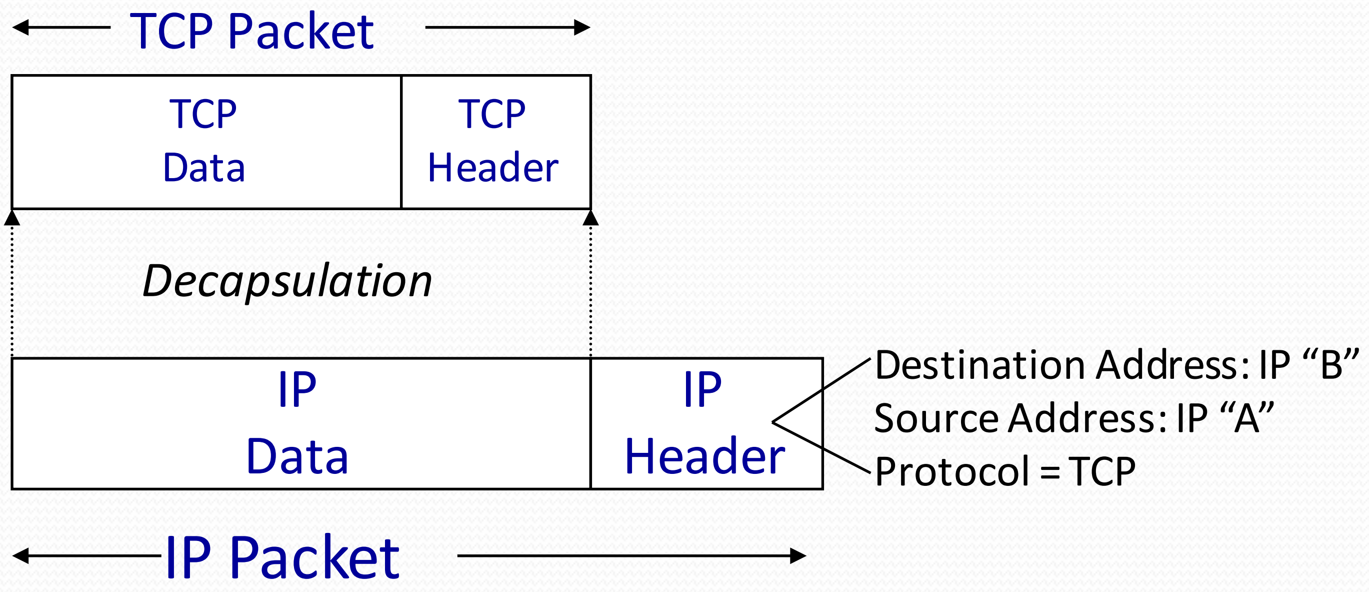
Ethernet Packet

In the Receiving Host - Cont’d



18. Internet Protocol (IP)

- Verify IP address.
- Extract/decapsulate TCP packet from IP packet.
- Pass TCP packet to TCP Protocol.



TCP Packet

TCP Data

TCP Header

Decapsulation

IP Data

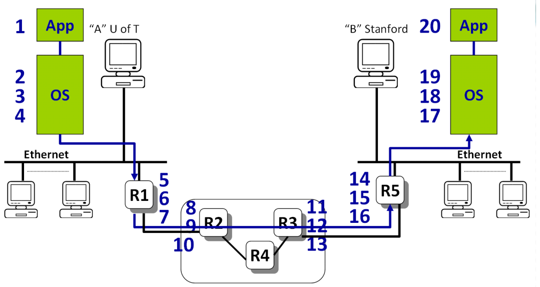
IP Header

Destination Address: IP “B”

Source Address: IP “A”

Protocol = TCP

IP Packet



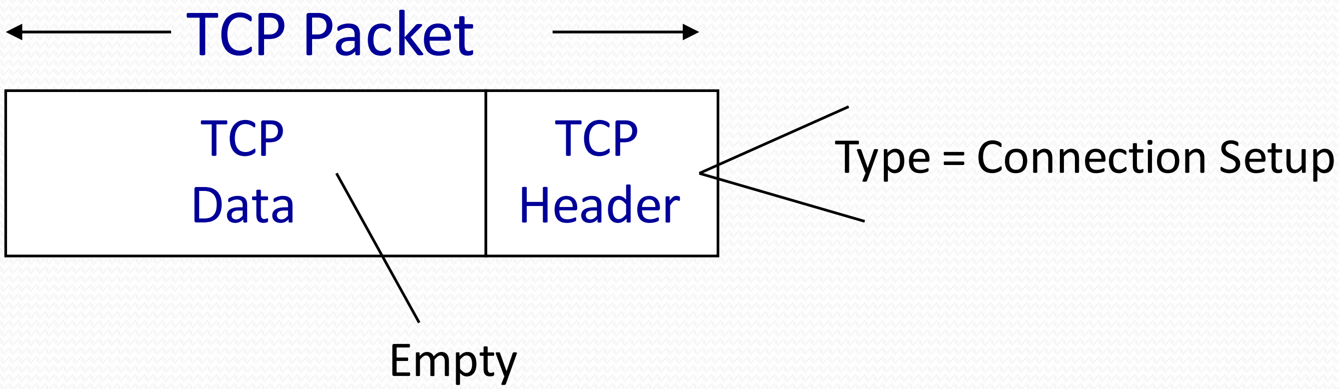
In the Receiving Host - Cont'd

19. Transmission Control Protocol (TCP)

- Accepts TCP "Connection setup" packet
- Establishes connection by sending "Ack".

20. Application-Programming Interface (API)

- Application receives request for TCP connection with "A".



TCP Packet

TCP Data

TCP Header

Type = Connection Setup

Empty

Outline – Foundations & Basic Concepts

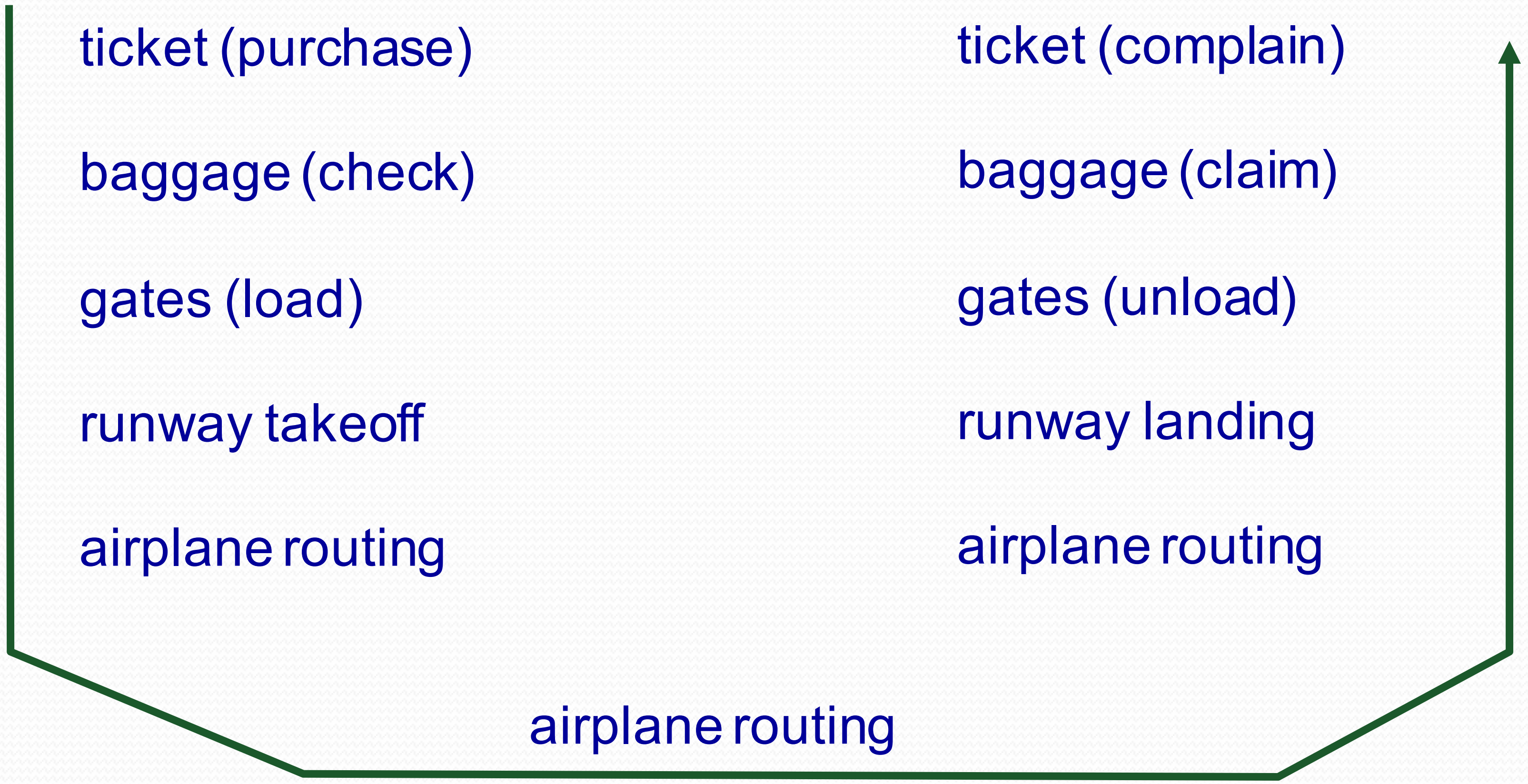
- A detailed FTP example
- Layering
- Packet switching and circuit switching

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?

Organization of air travel

- a series of steps



ticket (purchase)

baggage (check)

gates (load)

runway takeoff

airplane routing

airplane routing

ticket (complain)

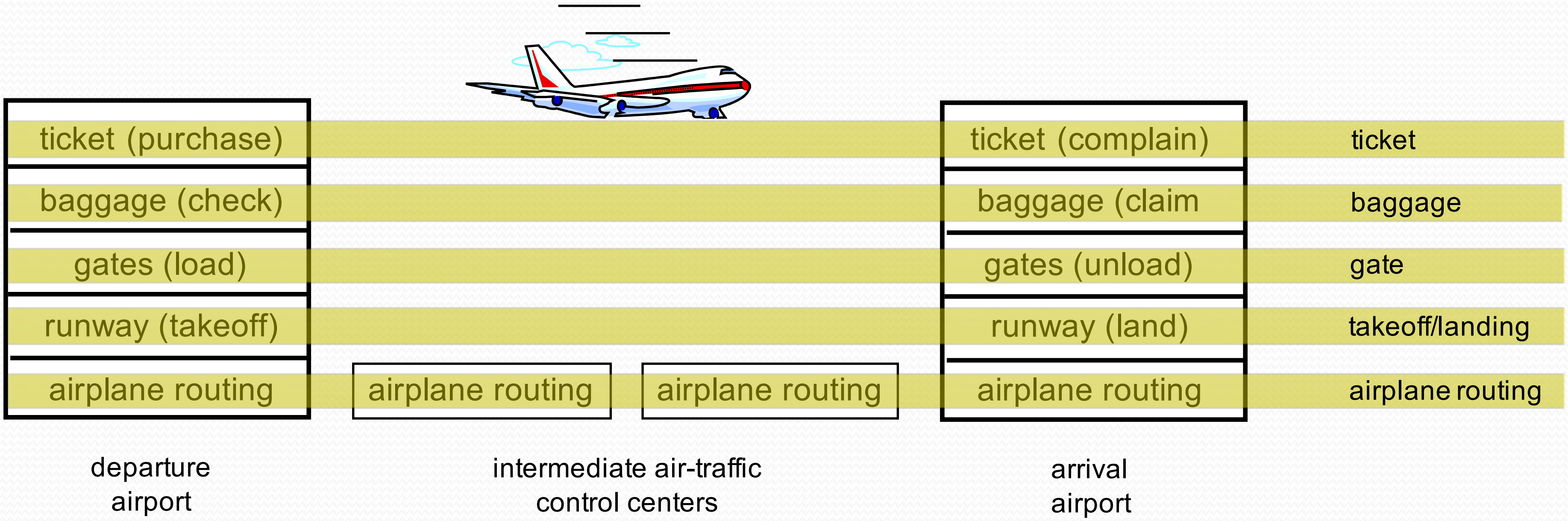
baggage (claim)

gates (unload)

runway landing

airplane routing

Layering of airline functionality



ticket (purchase)
baggage (check)
gates (load)
runway (takeoff)
airplane routing

airplane routing
airplane routing

ticket (complain)
baggage (claim
gates (unload)
runway (land)
airplane routing

ticket
baggage
gate
takeoff/landing
airplane routing

departure airport

intermediate air-traffic control centers

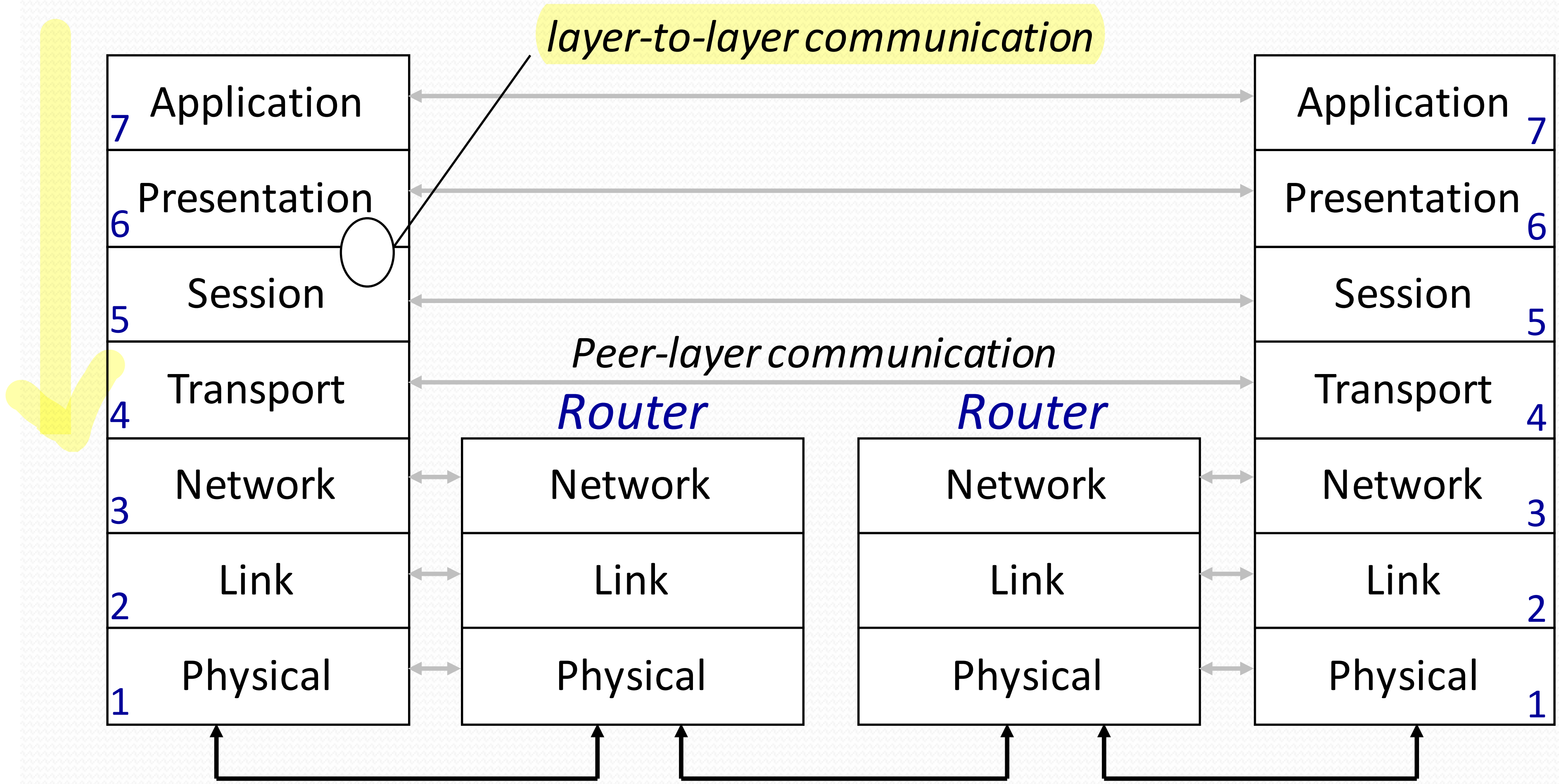
arrival airport

- Layers: each layer implements a service
  - via its own internal-layer actions
  - relying on services provided by layer below

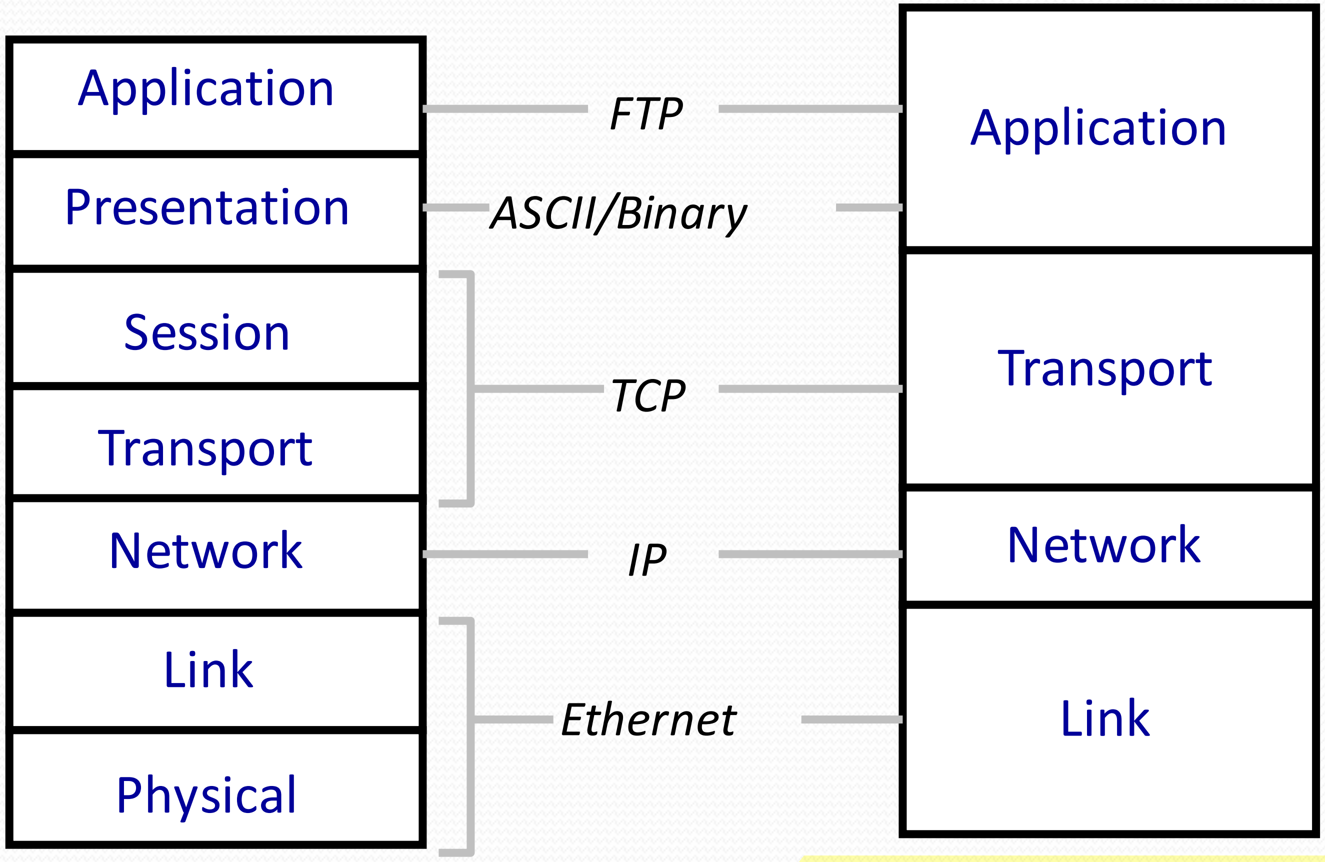
Why layering?

- Deal with complex systems:
  - explicit structure allows identification, relationship of complex system's pieces
  - 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?

Layering – *The OSI Model*



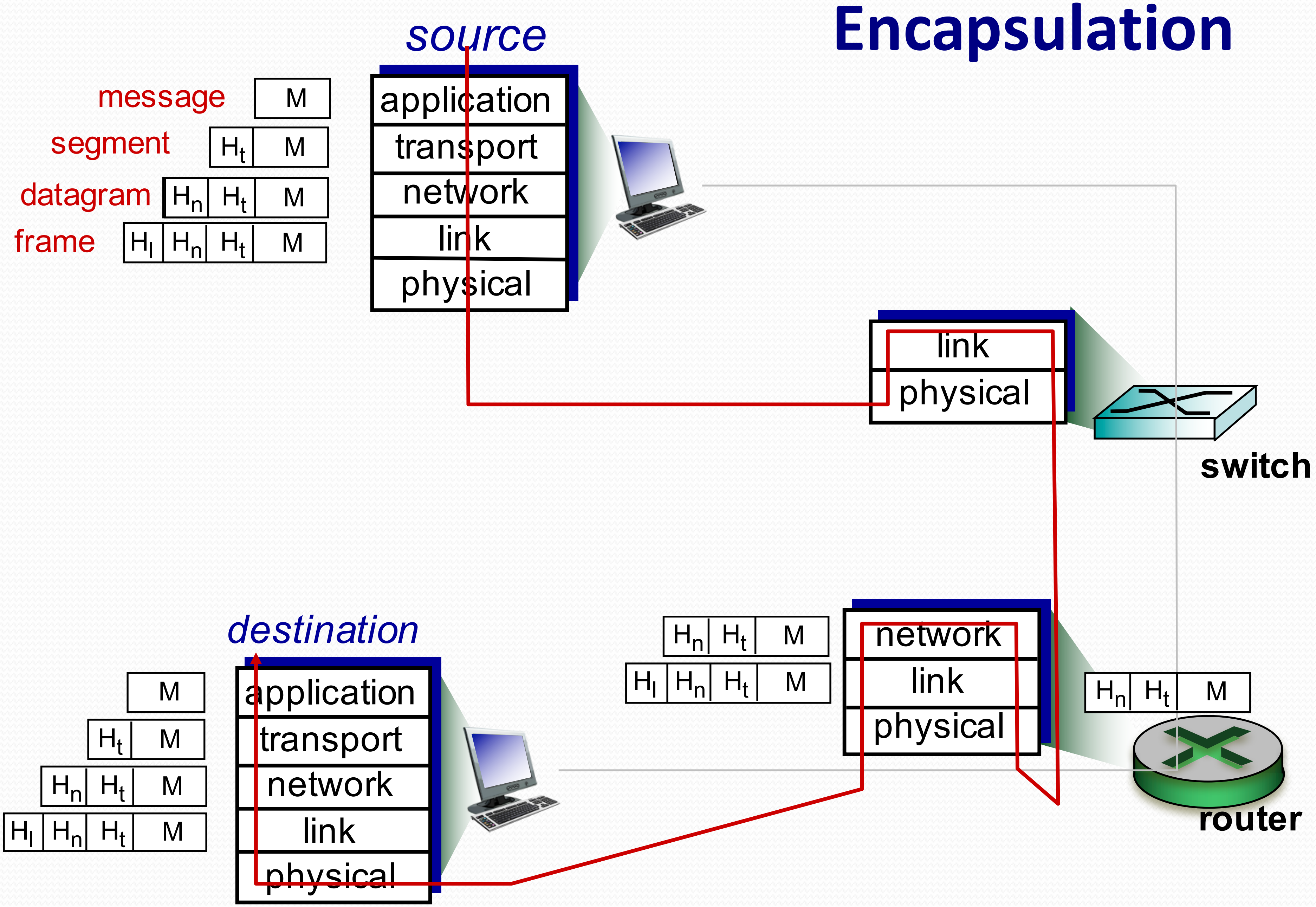
Layering – *Our FTP Example*



The 7-layer OSI Model

The 4-layer Internet model

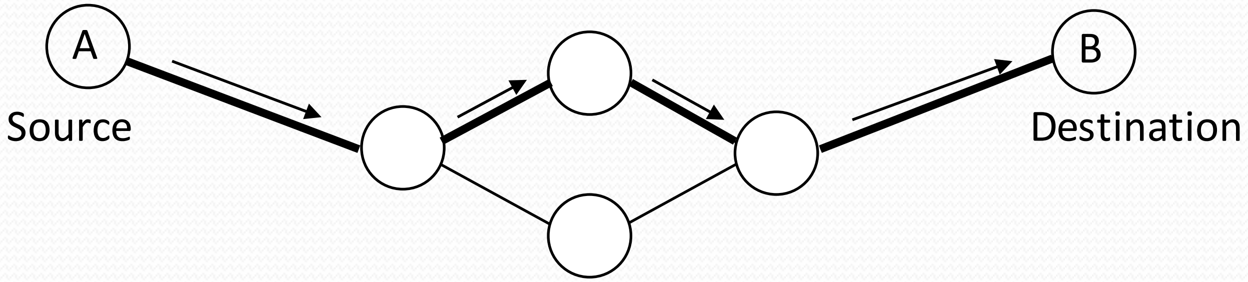
Encapsulation



Outline – Foundations & Basic Concepts

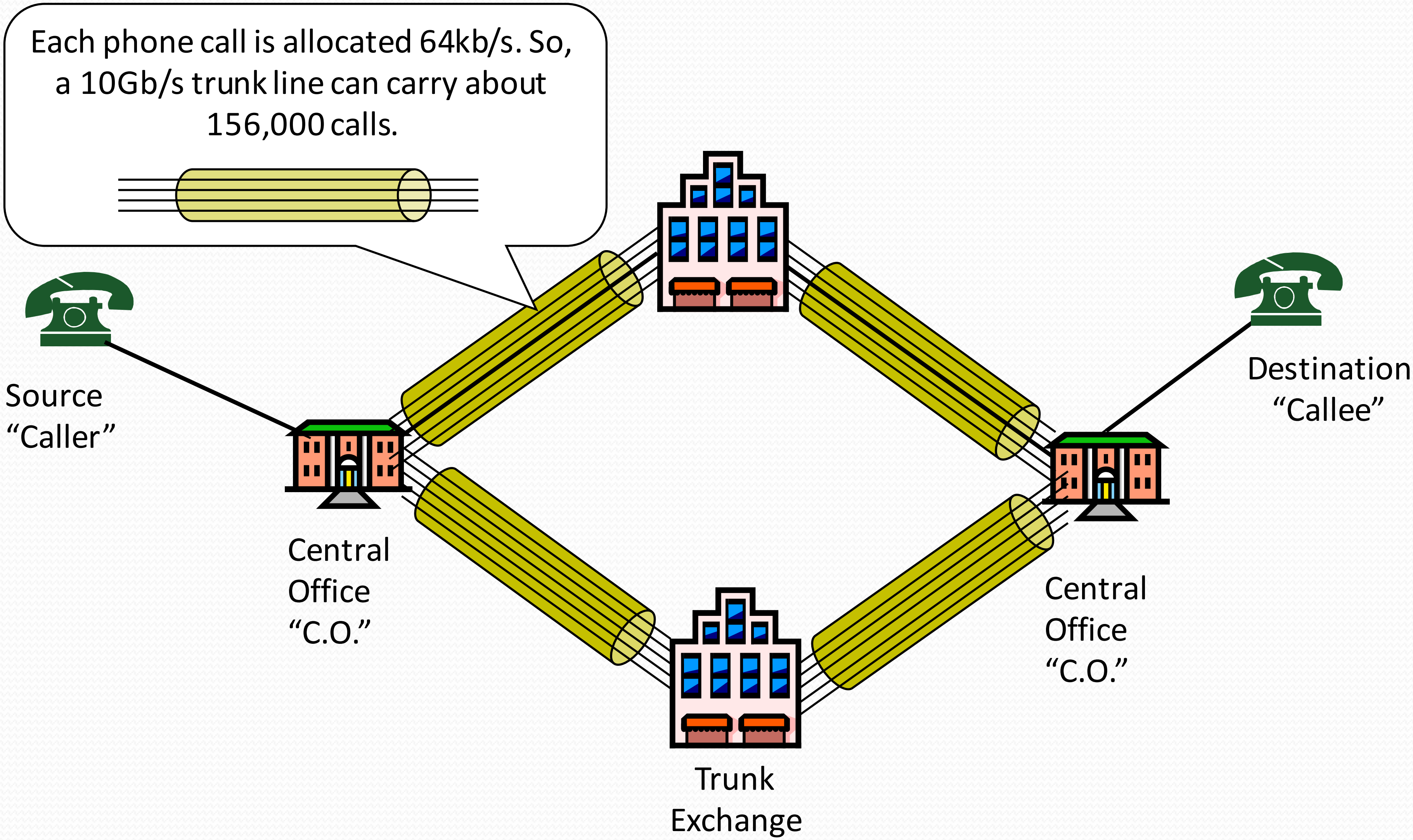
- A detailed FTP example
- Layering
- Packet switching and circuit switching

Circuit Switching

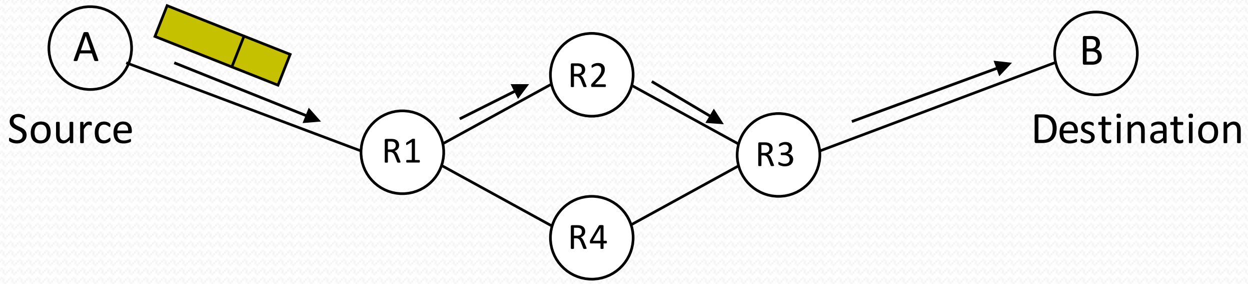


- It's the method used by the telephone network.
- A call has three phases:
  - Establish circuit from end-to-end ("dialing"),
  - Communicate,
  - Close circuit ("tear down").
- Originally, a circuit was an end-to-end physical wire.
- Nowadays, a circuit is like a virtual private wire: each call has its own private, guaranteed data rate from end-to-end.

Circuit Switching – *Telephone Network*

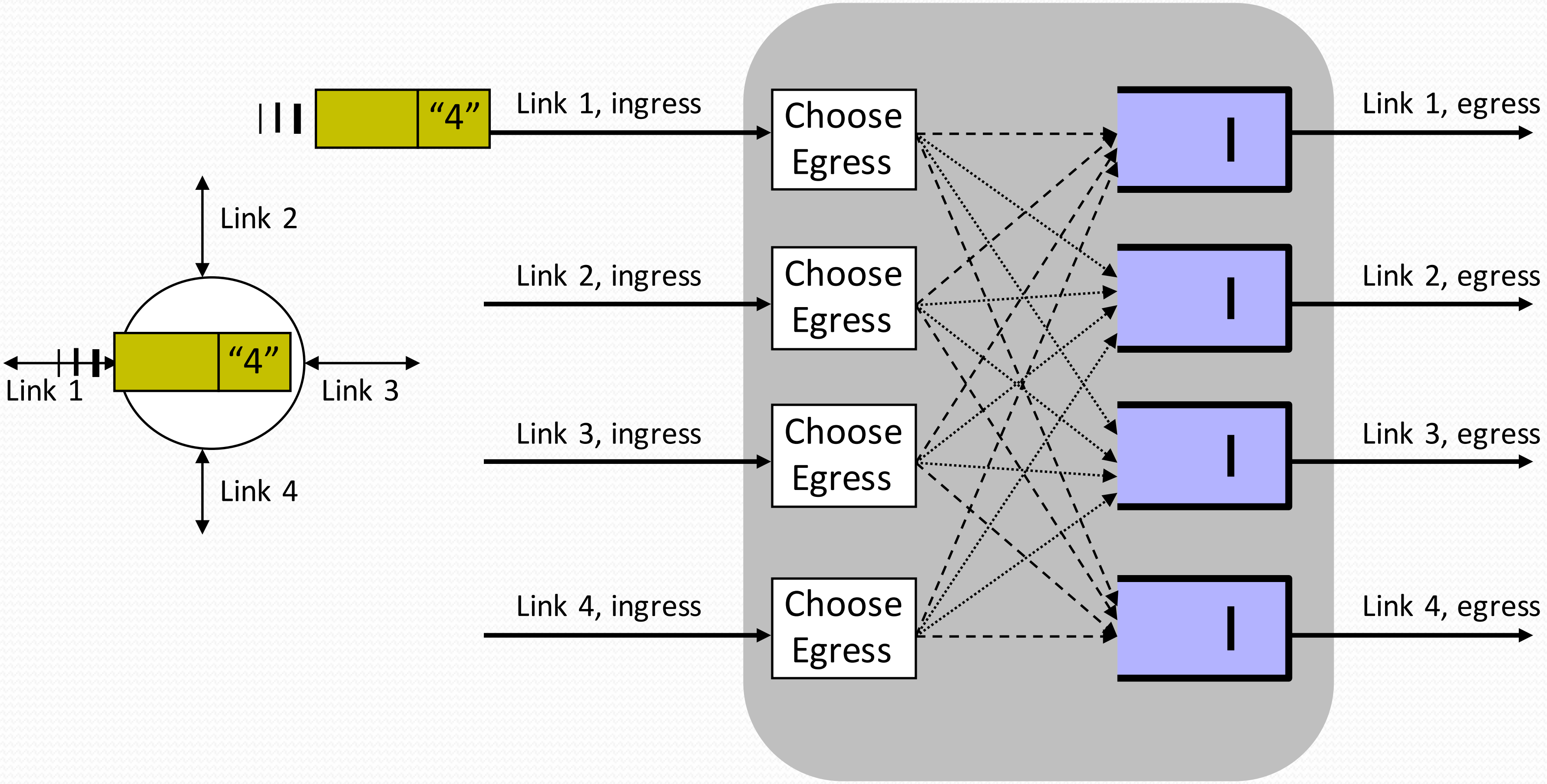

Each phone call is allocated 64kb/s. So, a 10Gb/s trunk line can carry about 156,000 calls.
Source "Caller"
Central Office "C.O."
Trunk Exchange
Central Office "C.O."
Destination "Callee"

Packet Switching



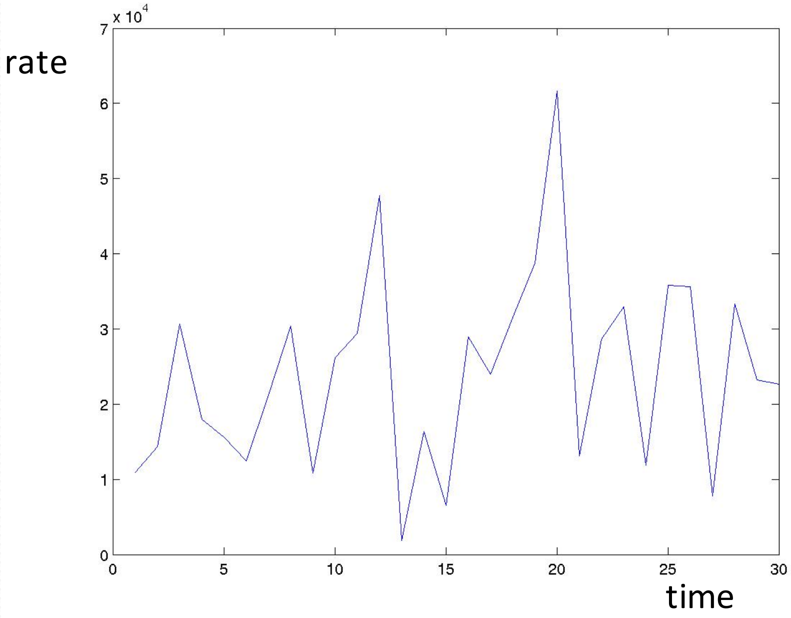
- It’s the method used by the Internet.
- Each packet is individually routed packet-by-packet, using the router’s local routing table.
- The routers maintain no per-flow state.
- Different packets may take different paths.
- Several packets may arrive for the same output link at the same time, therefore a packet switch has buffers.

Packet Switching – *Simple Router Model*



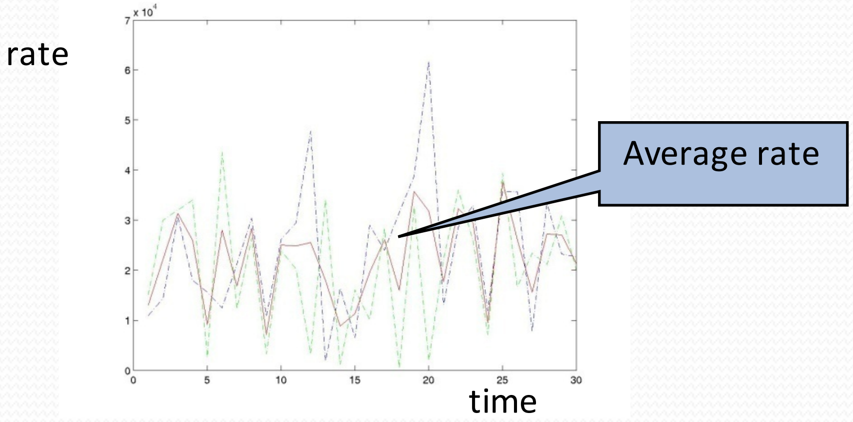
Statistical Multiplexing – *Basic Idea*

One flow

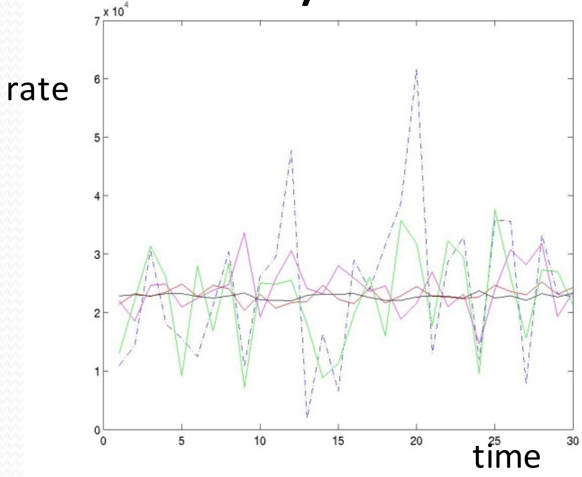


- Network traffic is bursty.
  i.e. the rate changes frequently.
- Peaks from independent flows generally occur at different times.
- Conclusion: The more flows we have, the smoother the traffic.

Two flows

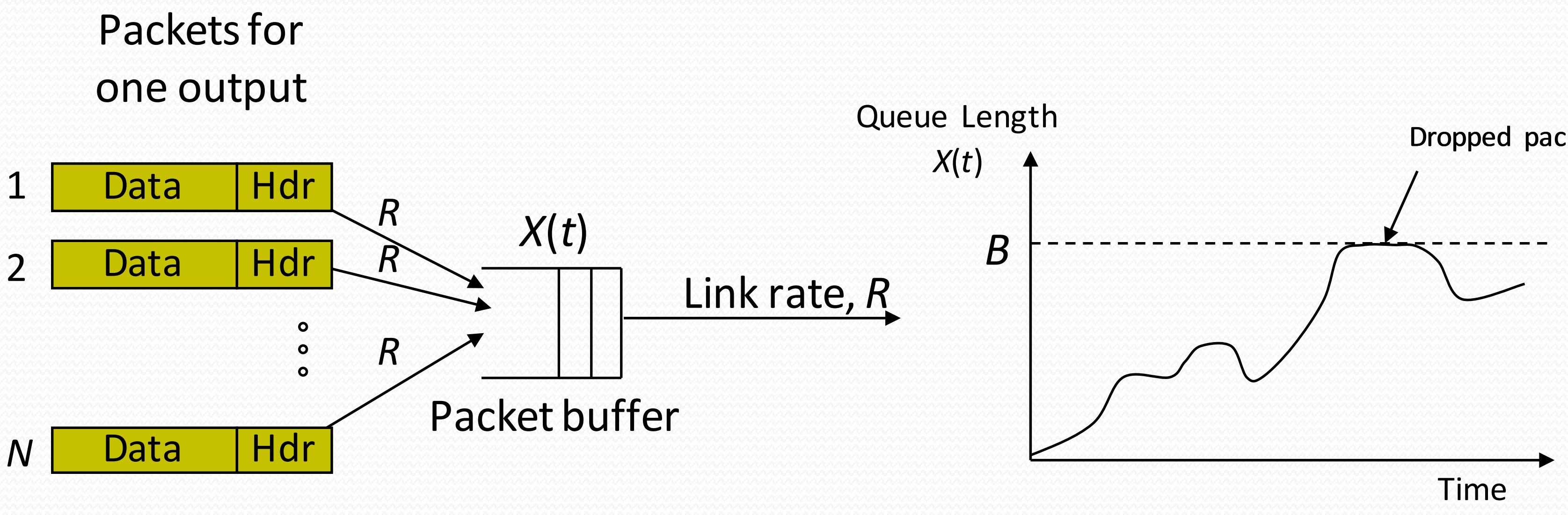


Many flows



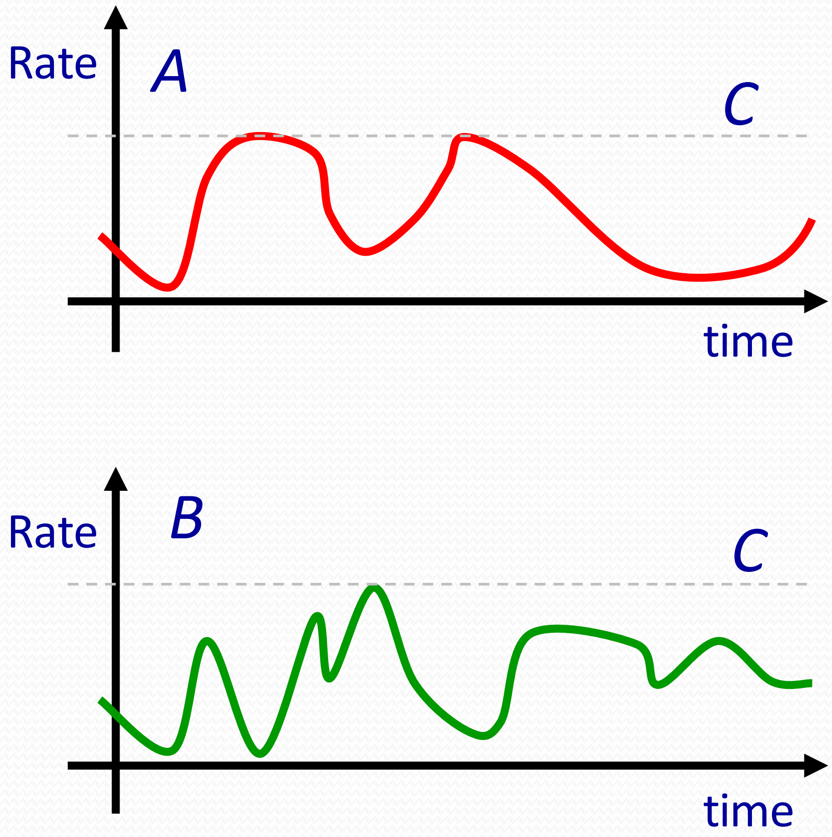
Average rates of:
1, 2, 10, 100, 1000 flows.

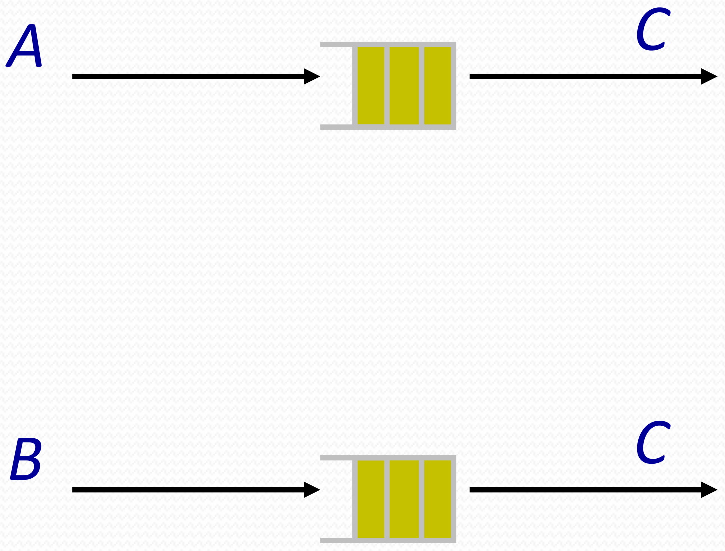
Packet Switching – *Statistical Multiplexing*



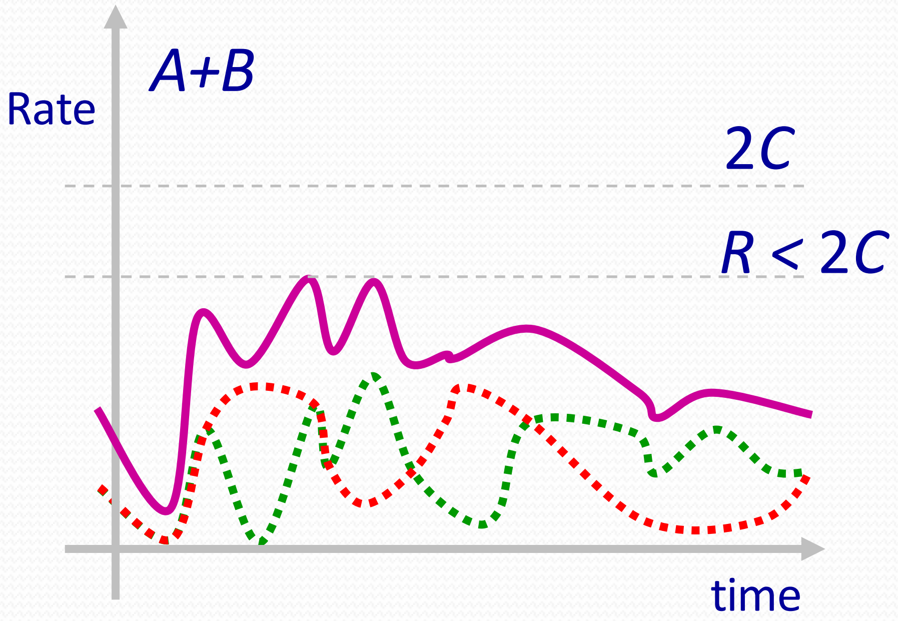
- Because the buffer absorbs temporary bursts, the egress link need not operate at rate *N.R.*
- But the buffer has finite size, *B*, so losses will occur.

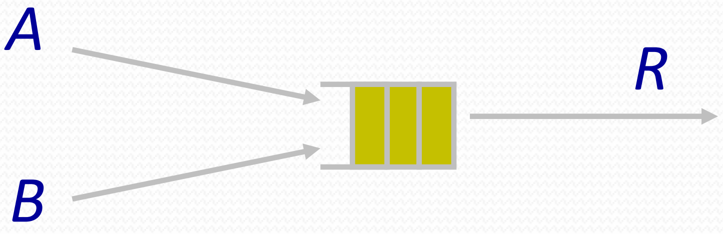
Statistical Multiplexing





Statistical Multiplexing Gain





Statistical multiplexing gain = $2C/R$

Other definitions of SMG: The ratio of rates that give rise to a particular queue occupancy, or particular loss probability.

Why Packet Switching in the Internet?

- Efficient use of expensive links:
  - The links are assumed to be expensive and scarce.
  - Packet switching allows many, bursty flows to share the same link efficiently.
  - "Circuit switching is rarely used for data networks, ... because of very inefficient use of the links" - Gallager
- Resilience to failure of links & routers:
  - "For high reliability, ... [the Internet] was to be a datagram subnet, so if some lines and [routers] were destroyed, messages could be ... rerouted" - Tanenbaum

Final Comments, Discussion

- Is layering the best approach?
  - Simplifies design
  - Yet, limited and inflexible
- Best effort service
  - Made the rapid growth of the Internet possible
  - Makes providing any guarantees very difficult
- Packet switching
  - Enables statistical multiplexing
  - We need extremely fast routers
- Routing
  - How does a router know which output port to send the packet to?

Next Week

- Link Layer
- Error Detection/Correction
- Tutorial: Intro to Mininet + Wireshark

- Readings
  - CH 5: Kurose & Rose

Questions?