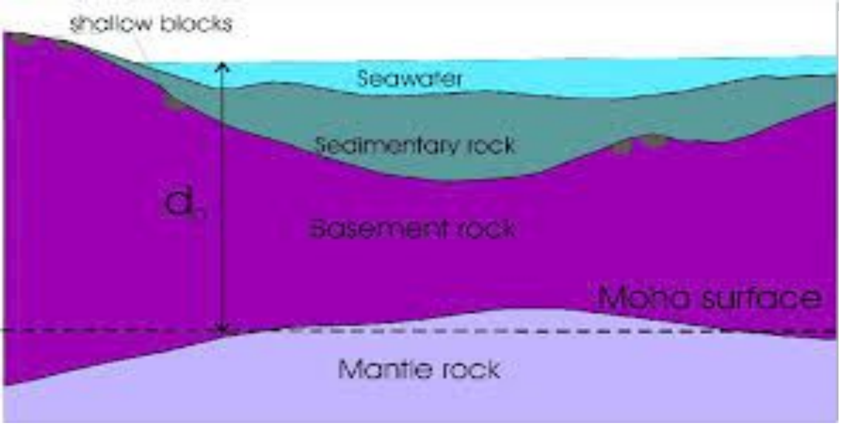
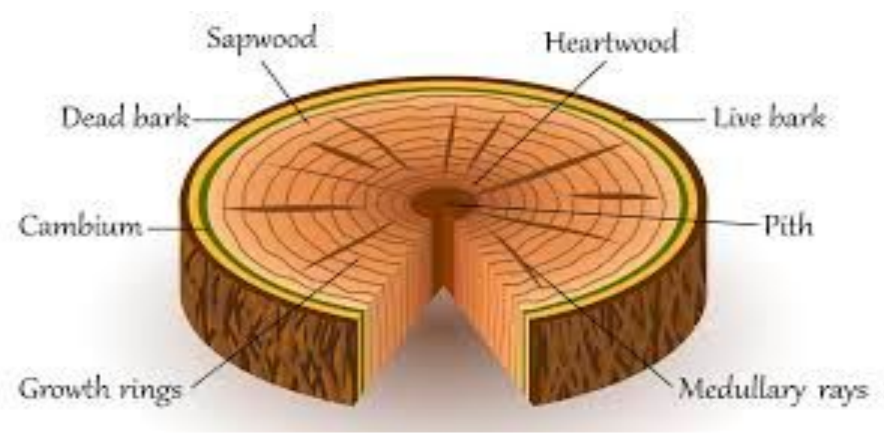
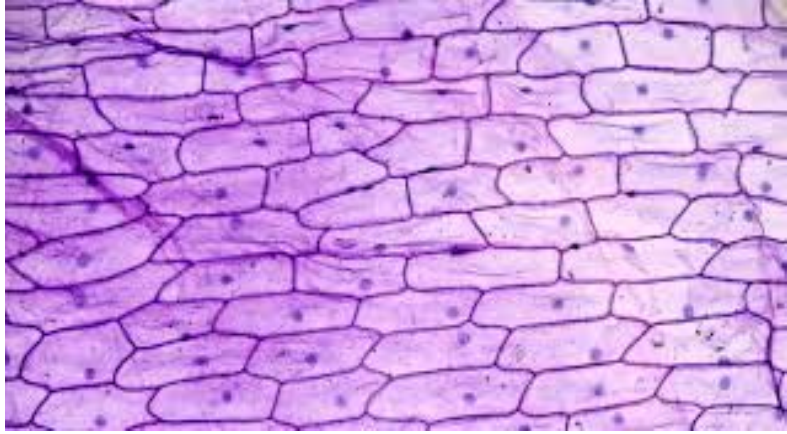






Network Layers

Jennifer Rexford

COS 316 Guest Lecture



Modularity in System Design

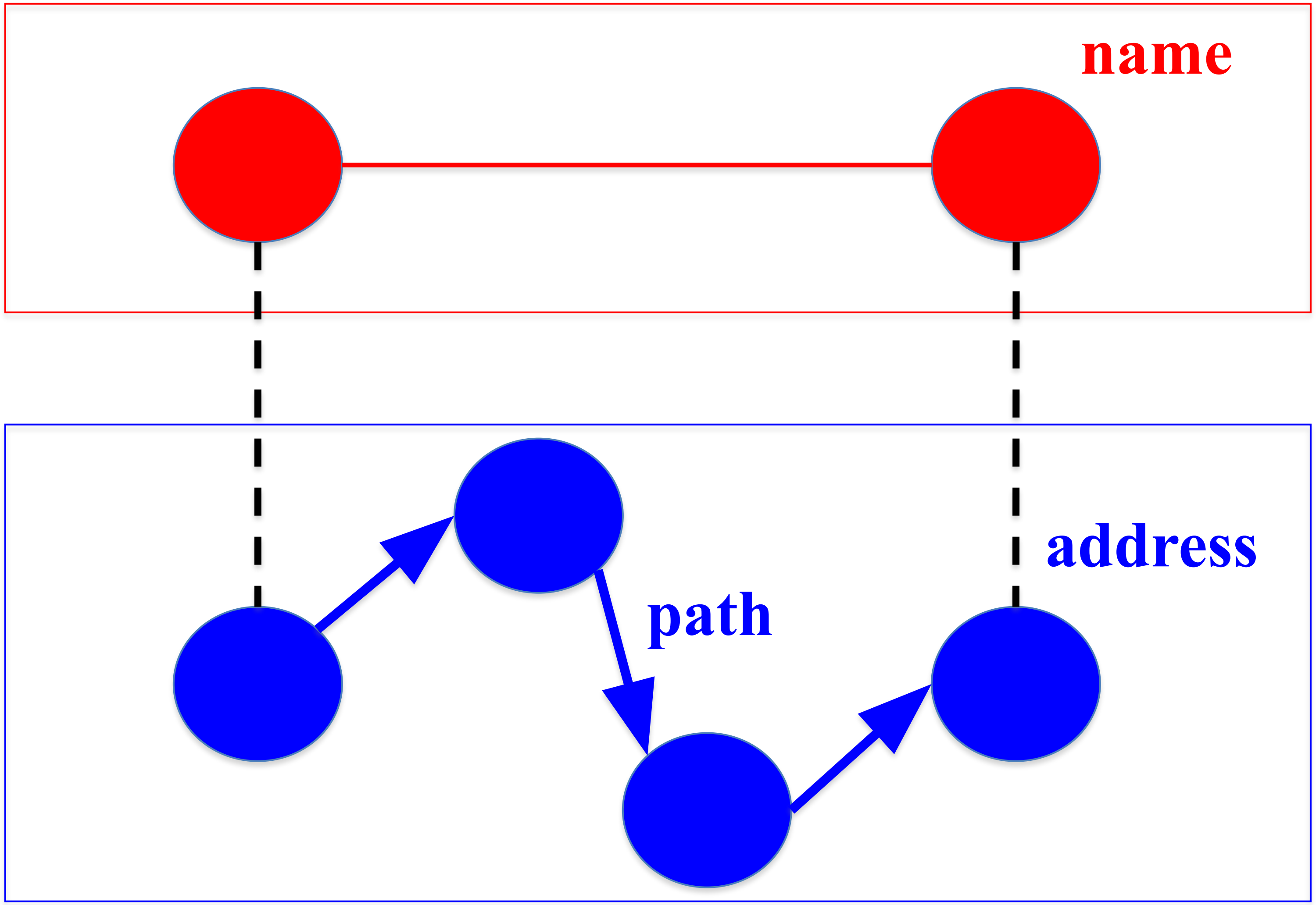
Barbara Liskov,
MIT

"Modularity based on abstraction is the way things get done."

Modularity Through Layering

Or, “Can I Explain How the Internet Works in One Lecture?”

Modularity Through Protocol Layering



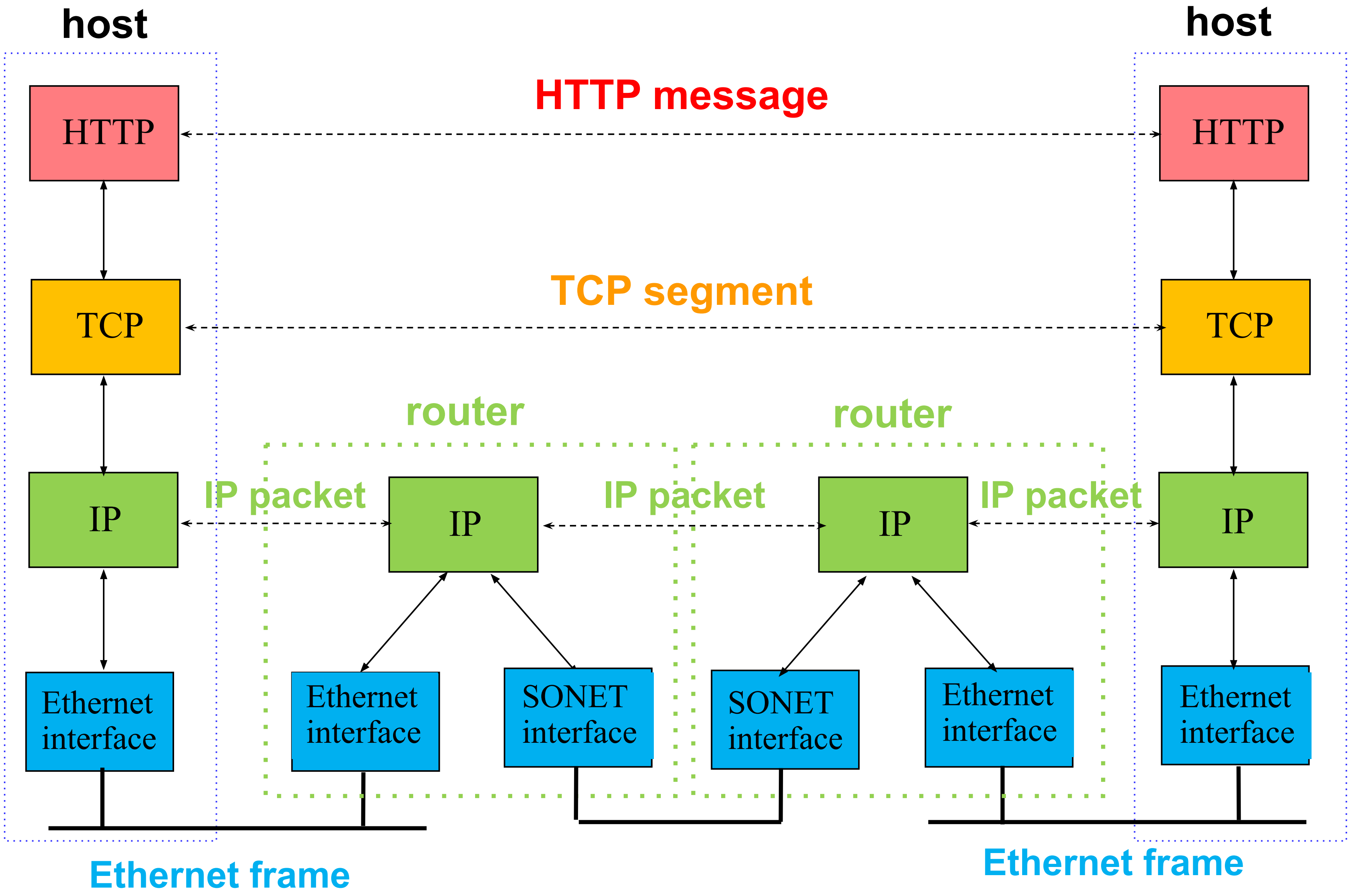
Internet Protocol Layers

Application	Between software applications running on hosts
Transport	Between hosts while maintaining quality-of-service
Network	Between nodes in different networks
Link	Between physically-connected devices
Physical	Between a device and the physical medium

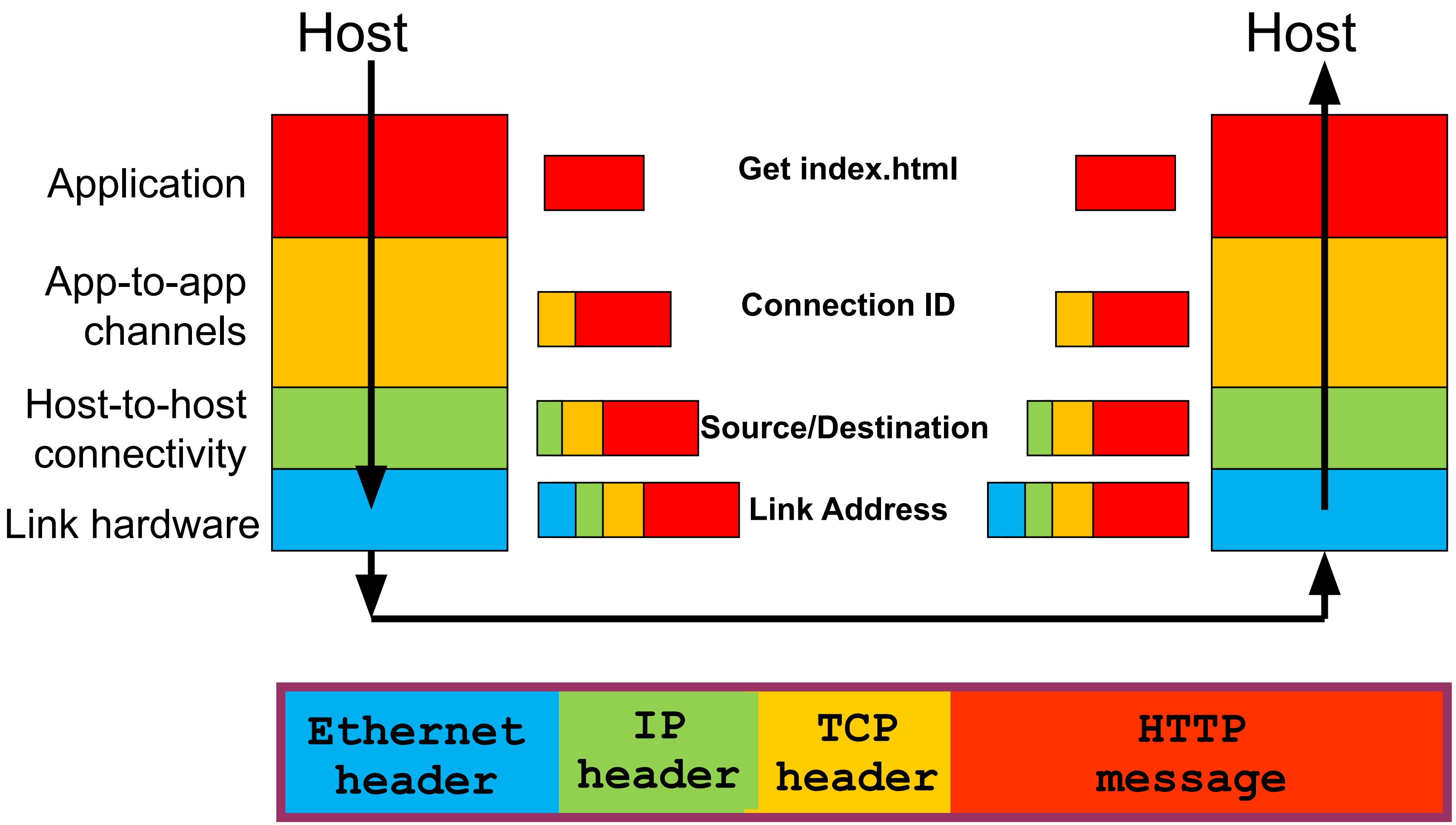
Internet Protocol Layers

Layer	Function	Protocols
Application	Application Messages	HTTP, SMTP, FTP, Skype, etc.
Transport	Reliable streams / Datagrams	TCP, UDP
Network	Best-effort *global* packet delivery	IP
Link	Best-effort *local frame* delivery	Ethernet, WiFi, etc.
Physical	Bit delivery	

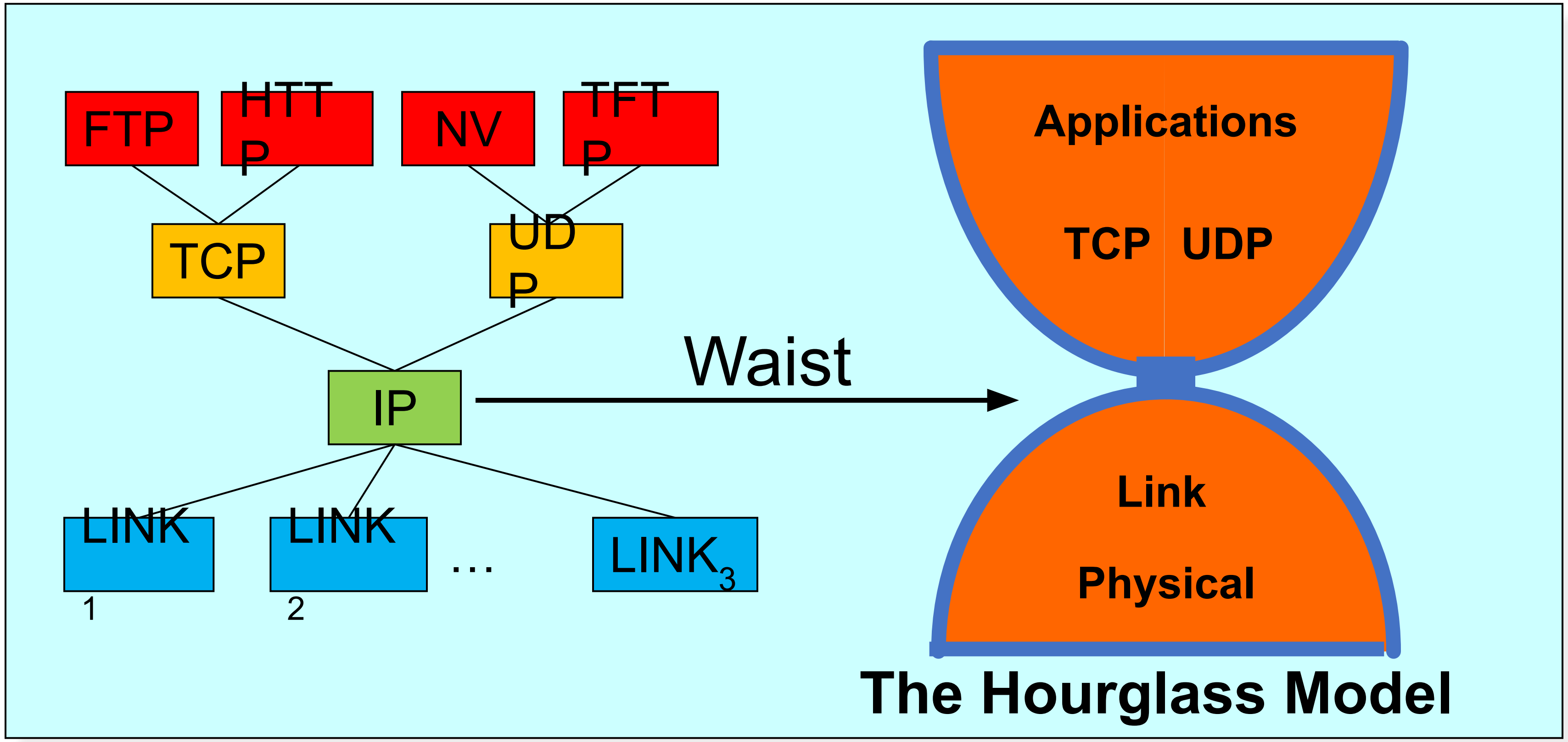
Layers in Action



Encapsulation: Layers of Headers



The Internet Hourglass with “Narrow Waist”



The “narrow waist” facilitates interoperability

Network Protocols

What is a Network Protocol?

- Rules that govern communication
  - How to identify the devices and establish connectivity
  - Message format (syntax) and meaning (semantics)
- A distributed solution to a problem
  - Deliver an ordered, reliable stream of bytes to another end-point
  - Share network bandwidth fairly
  - Compute shortest paths on a graph
  - Automatically learn how to reach remote hosts
  - Share an individual link's bandwidth fairly

Application: HyperText Transfer Protocol (HTTP)

Request

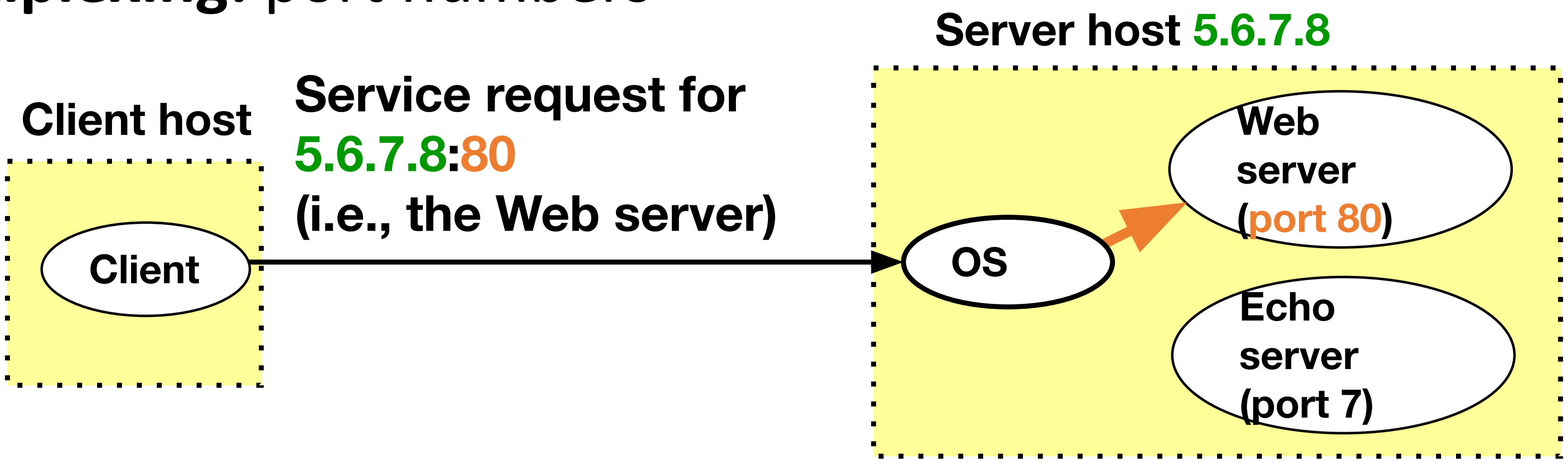
```
GET /courses/archive/spr12/cos461/ HTTP/1.1
Host: www.cs.princeton.edu
User-Agent: Mozilla/4.03
CRLF
```

Response

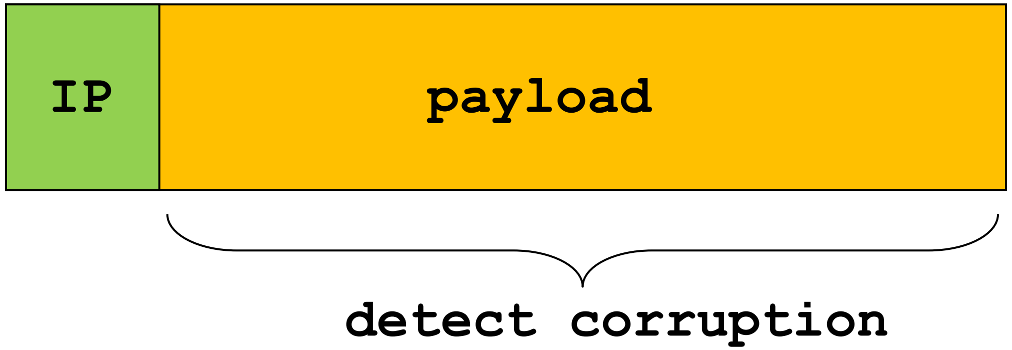
```
HTTP/1.1 200 OK
Date: Mon, 6 Feb 2012 13:09:03 GMT
Server: Netscape-Enterprise/3.5.1
Last-Modified: Mon, 7 Feb  2011 11:12:23 GMT
Content-Length: 21
CRLF
Site under construction
```

Transport Protocols: TCP and UDP

- **Demultiplexing:** port numbers



- **Error detection:** checksums



Transport: Transmission Control Protocol (TCP)

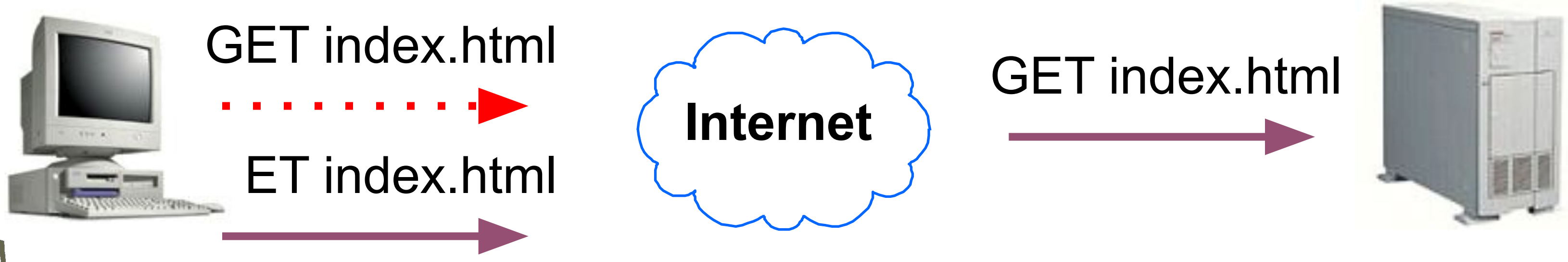
- Ordered, reliable stream of bytes
  - Built on top of best-effort packet delivery at the network layer (IP)

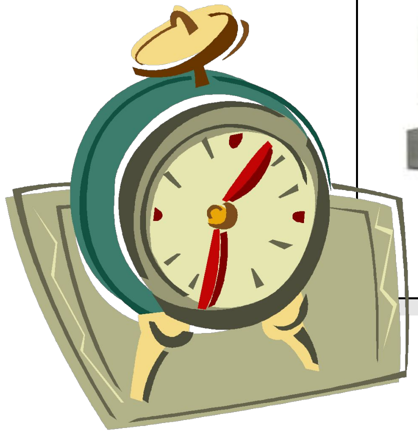
- Challenges with IP
  - Lost or delayed packets
  - Corrupted packets
  - Out-of-order packet arrivals
  - Receiver that runs out of space
  - Network that cannot handle the load

TCP: Lost or Delayed Packets

Problem: Lost or Delayed Data

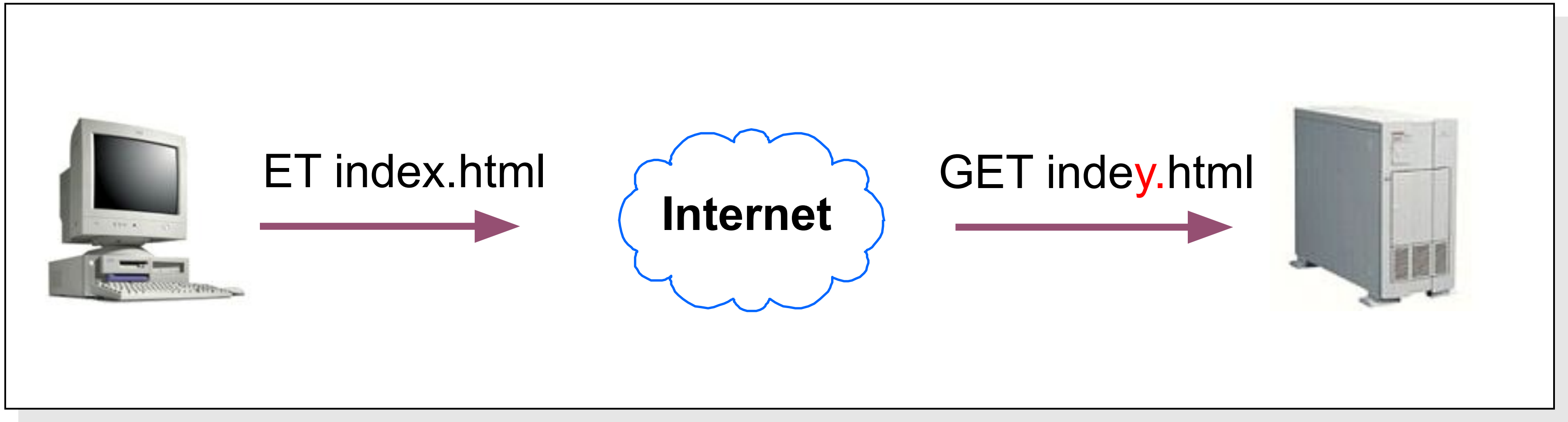
Solution: Timeout and Retransmit





Waiting for an acknowledgment

TCP: Corrupted Data



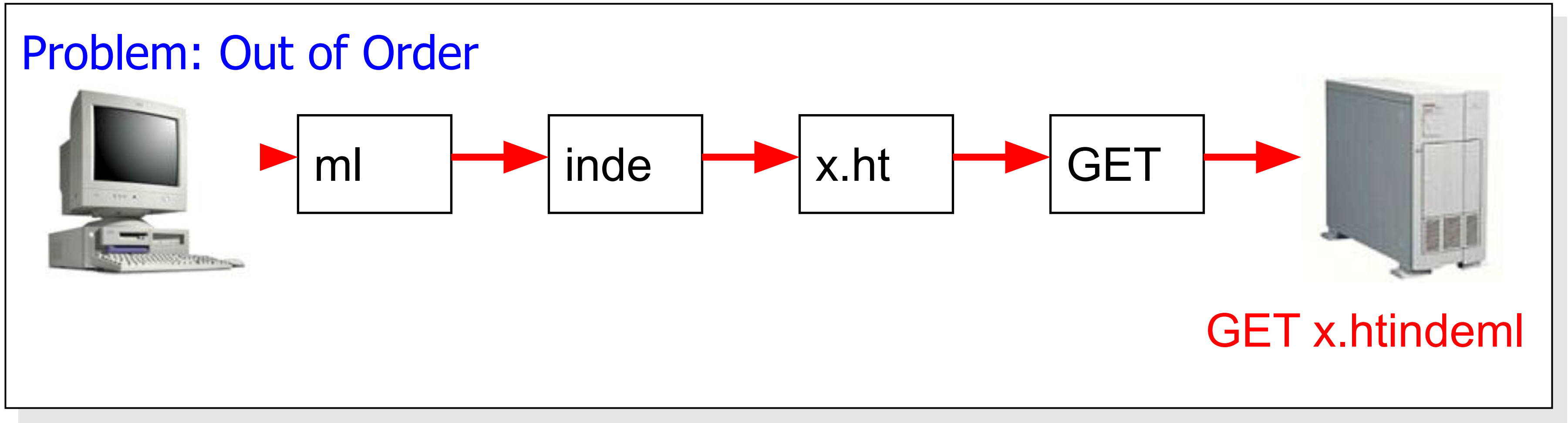
- Sender computes a checksum
  - Sender sums up all of the bytes
  - And sends the sum to the receive
- Receiver checks the checksum
  - Received sums up all of the bytes
  - And compares against the checksum

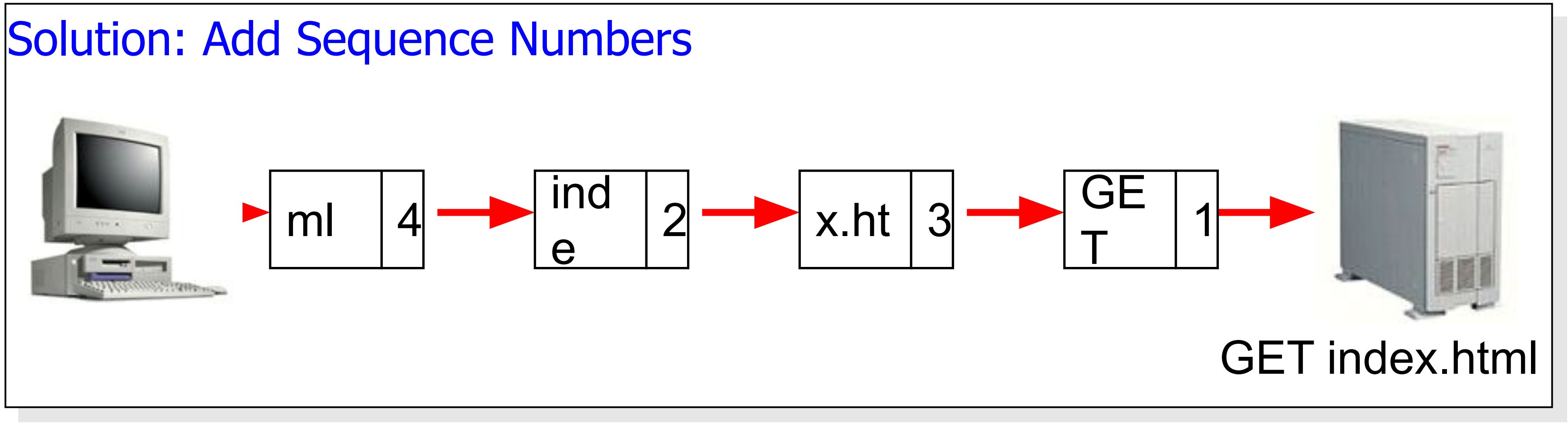
```
  134
+ 212
= 346
```

```
  134
+ 216
= 350
```

Then what?

TCP: Out-of-Order Packet Arrivals





TCP: Receiver that Runs Out of Space

?

- Receiver maintains a *window size*
  - Amount of data it can buffer
- Advertises window to the sender
  - Amount sender can send without acknowledgment
- Ensures that sender does not send too much
  - While still sending as much as possible

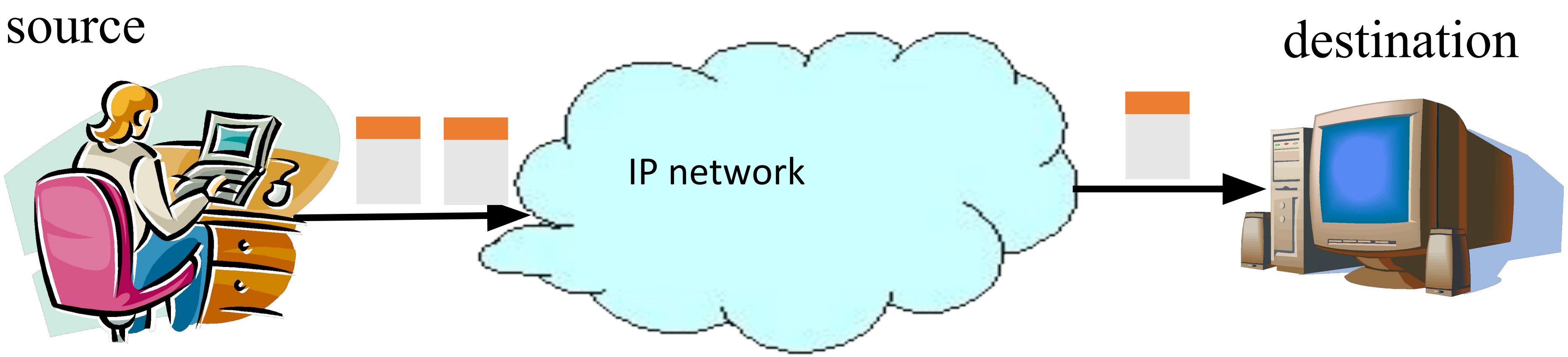
Flow control!

Network that Cannot Handle the Load

- Some TCP senders need to slow down…
- TCP congestion control (future lecture)!

Network Layer: Internet Protocol (IP)

- Best-effort global packet delivery
  - Packet delivery: each packet handled independently
  - Best-effort: allow loss, delay, corruption, and out-of-order delivery



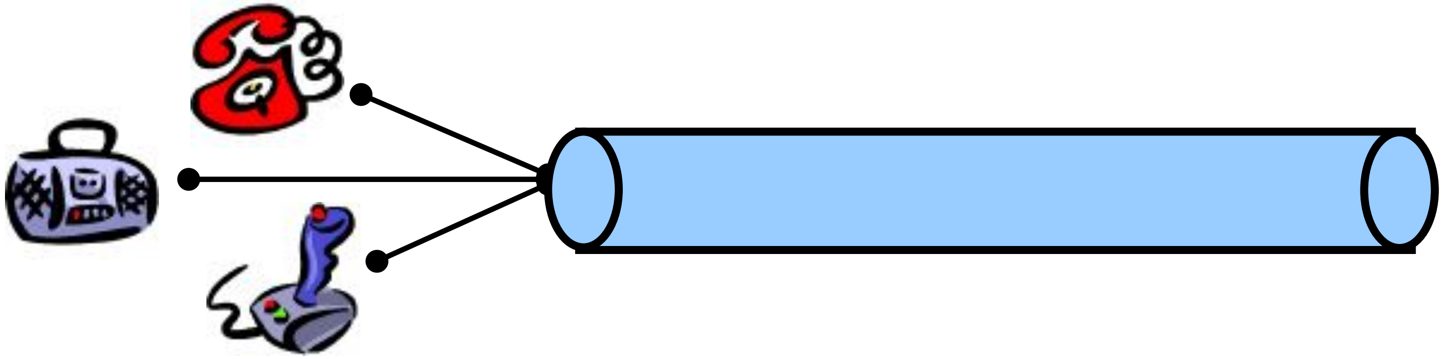
IP: Best-Effort Packet Delivery is Simpler

- Never having to say you're sorry…
  - Don't reserve bandwidth and memory
  - Don't do error detection and correction
  - Don't remember anything from one packet to next
- Easier to survive failures
  - Transient disruptions are okay during failover
- Can run on nearly any link technology
  - Greater interoperability and evolution

RFC 1149: IP Datagrams over Avian Carriers

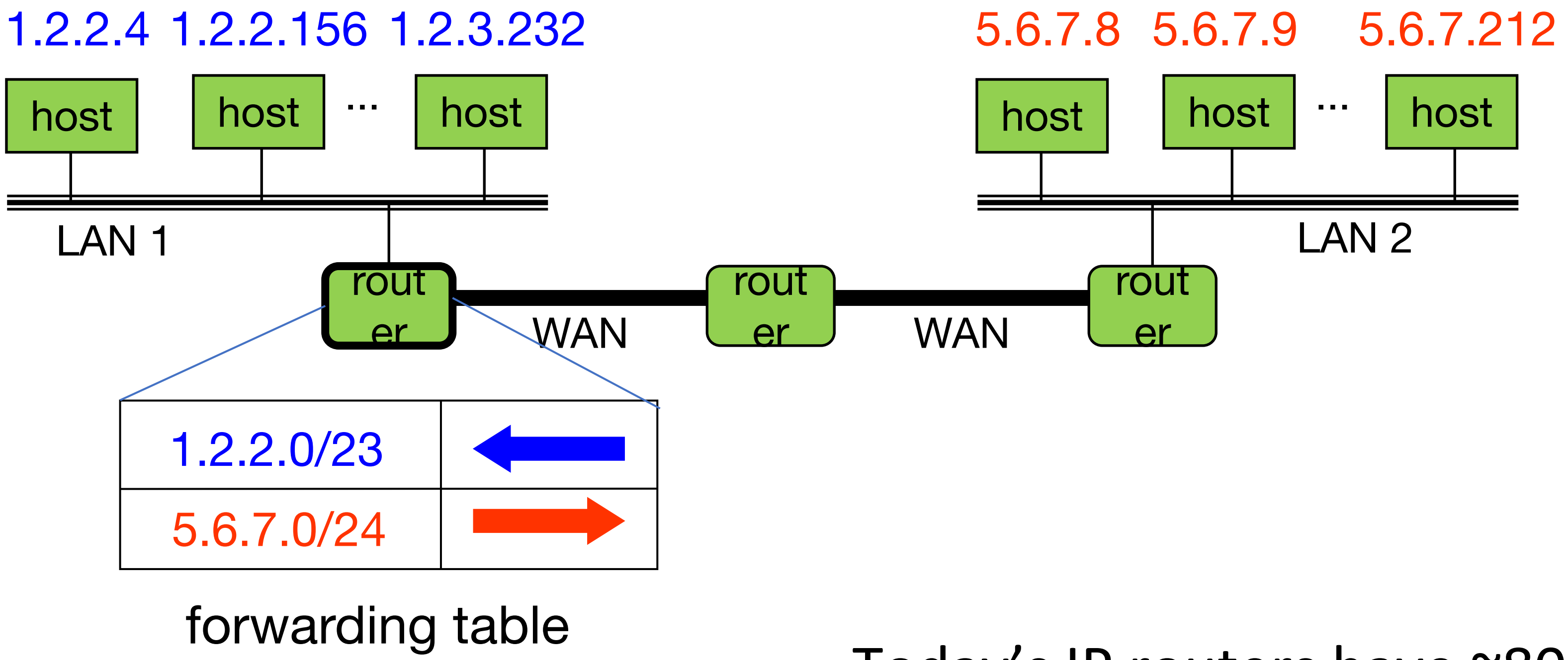
IP: Statistical Multiplexing

- Data traffic is bursty
  - Logging in to remote machines
  - Exchanging e-mail messages
- Don’t waste bandwidth
  - No traffic exchanged during idle periods
- Better to allow multiplexing
  - Different transfers share access to same links



IP: Scalable Global Packet Delivery

- Hierarchical IP addresses (“zip code”)
  - Variable-length prefix, identified by mask length

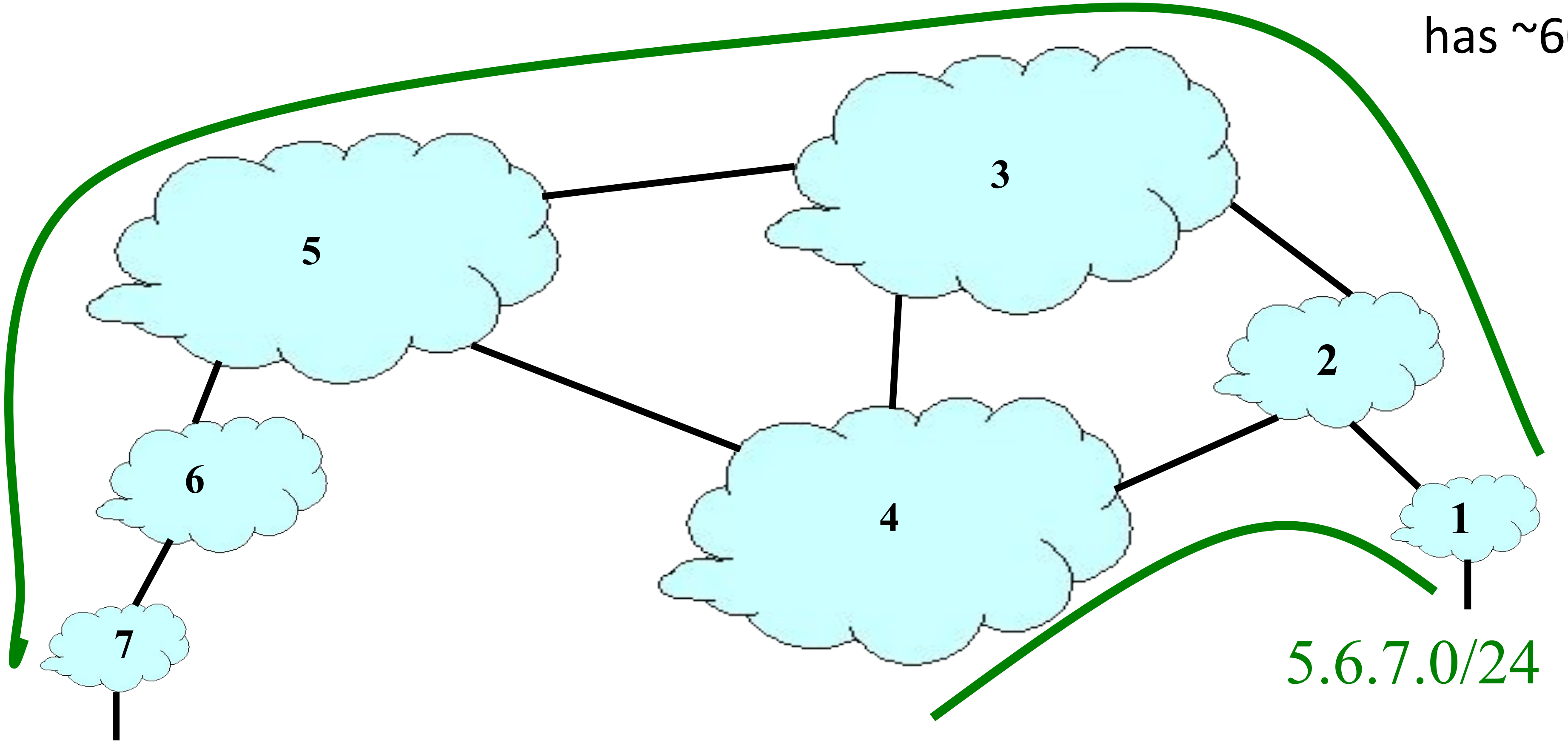


Today’s IP routers have ~800,000 prefixes

IP: Scalable Global Packet Delivery

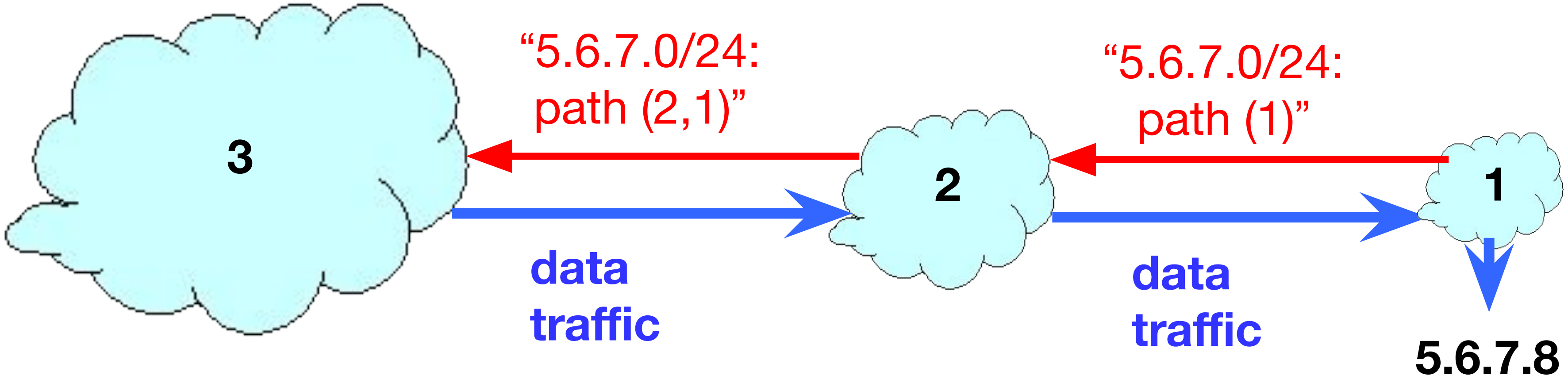
- Distributed global IP routing
  - Internet: a “network of networks” (Autonomous Systems)

Today’s Internet has ~60,000 ASes



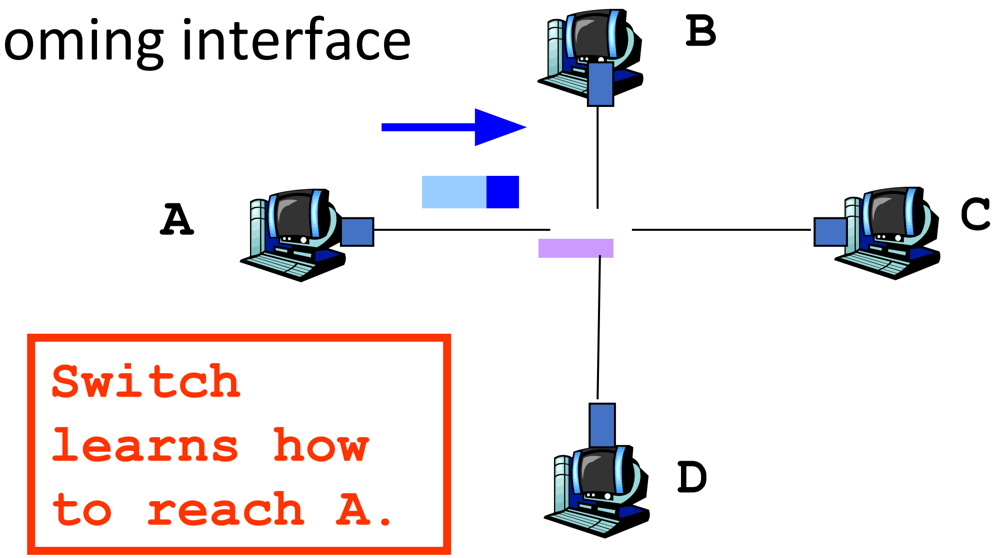
IP: Scalable Global Packet Delivery

- Distributed, policy-based IP routing
  - Interdomain routing: diffusion of IP prefixes (Border Gateway Protocol)



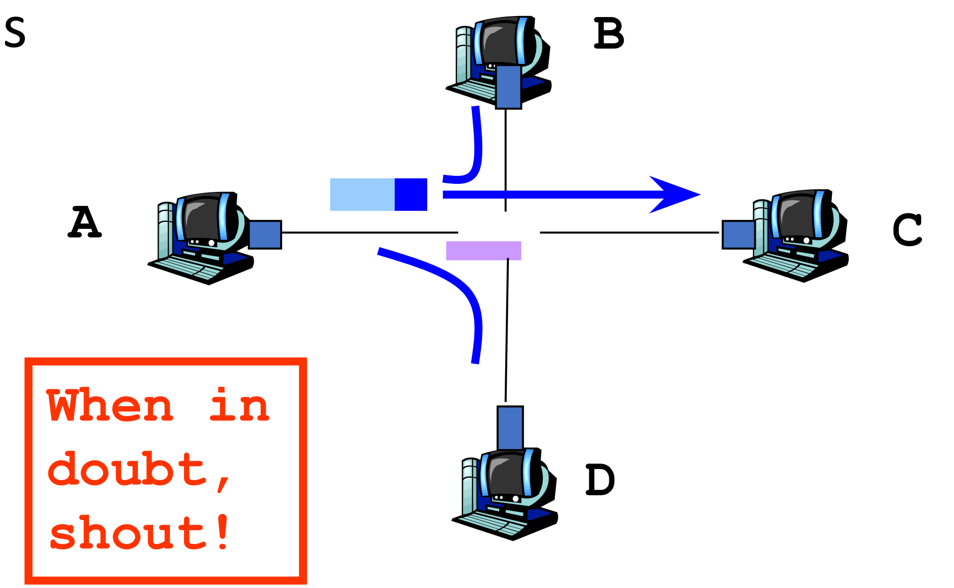
Link Layer: Ethernet Local Area Networks

- Automatic bootstrapping of best-effort local frame delivery
  - MAC address in end-host network interface card
  - MAC learning to reach other hosts in the LAN
- When an Ethernet frame arrives
  - Switch inspects the *source MAC* address
  - … and associates the address with the incoming interface



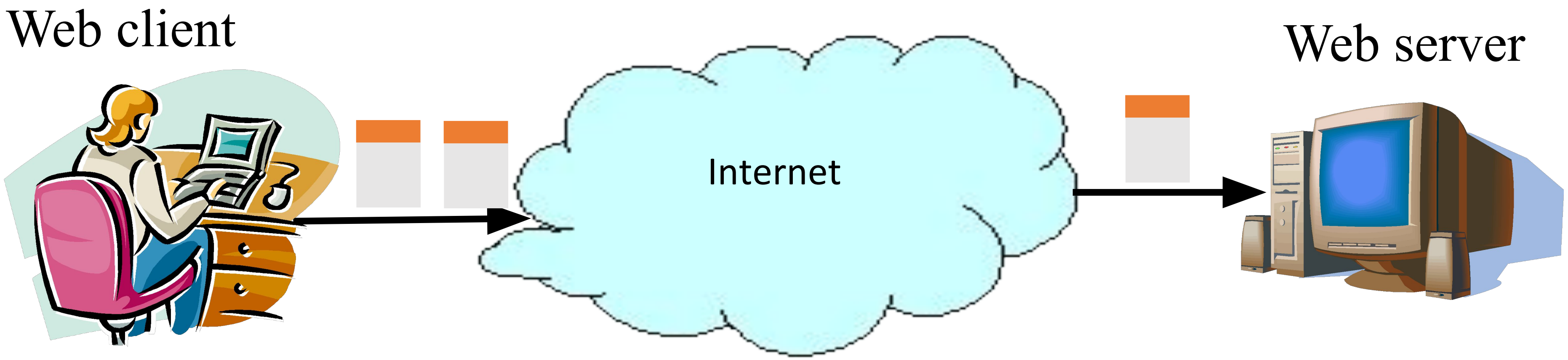
Link Layer: Ethernet Local Area Networks

- Automatic bootstrapping of best-effort local frame delivery
  - MAC address in end-host network interface card
  - MAC learning and flooding to reach other hosts in the LAN
- When the frame has an unfamiliar *destination*
  - Switch forwards the frame out all interfaces
  - … except the incoming interface



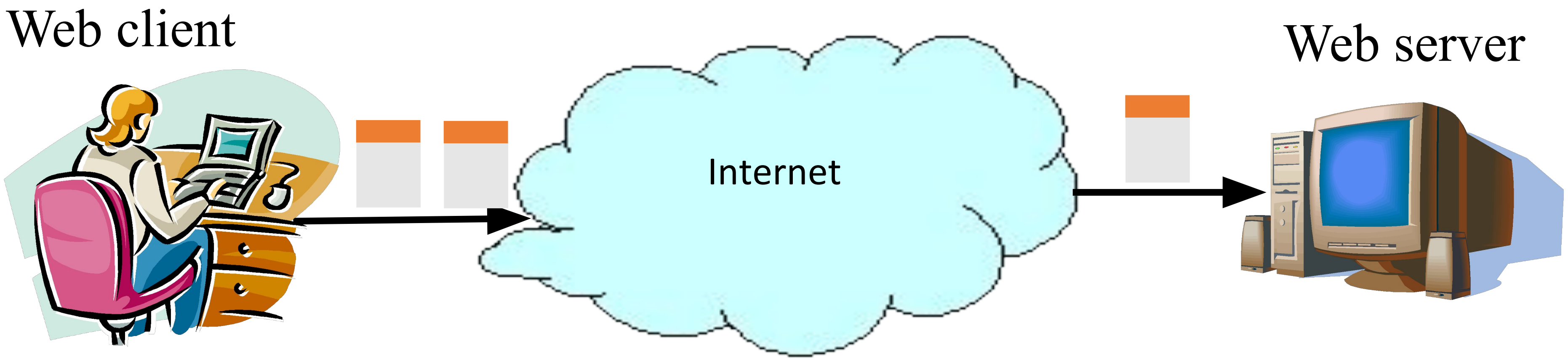
Putting it All Together: Crossing the Layers

- Crossing the layers
  - Application: HTTP request and response messages
  - Transport: TCP ordered reliable byte stream
  - Network: best-effort global IP packet delivery
  - Link: best-effort local Ethernet frame delivery



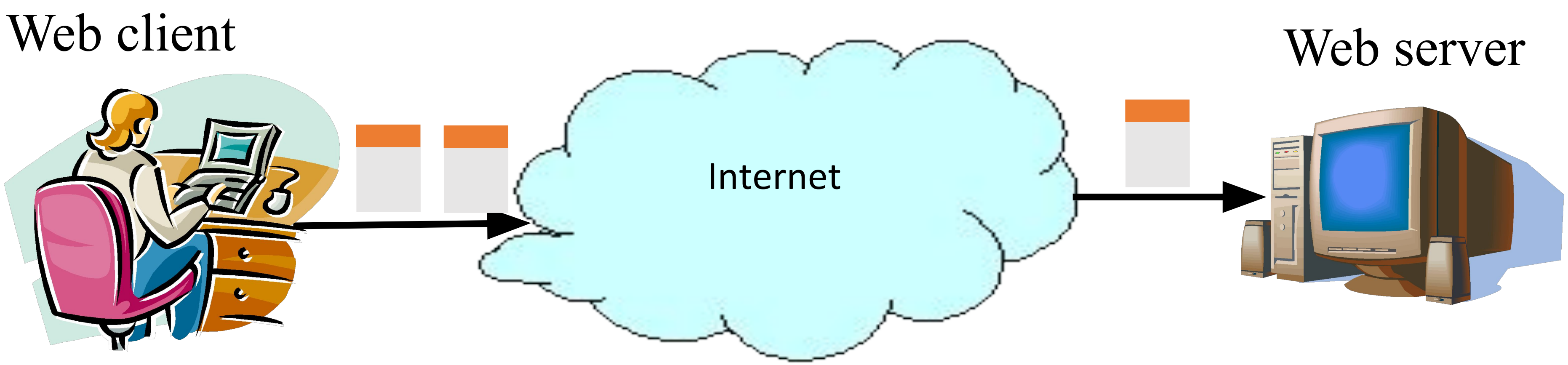
Putting it All Together: Crossing the Layers

- Naming at different layers
  - Application: http://www.cs.princeton.edu/courses/archive/spr12/cos461/
  - Transport: TCP connection to IP address 5.6.7.8 on port 80
  - Network: server host interface with IP address 5.6.7.8
  - Link: server host interface with MAC address 00:15:C5:49:04:A9



Putting it All Together: Crossing the Layers

- Routing at different layers
  - Application: maps request to local object /courses/archive/spr12/cos461/
  - Transport: directs TCP segments to specific transport port (e.g., port 80)
  - Network: directs IP packets toward IP destination prefix 5.6.7.0/24
  - Link: directs Ethernet frames to MAC address 00:15:C5:49:04:A9



Conclusion

- Modularity
  - The way to build and manage large systems
  - Protocol layering in computer networks
- Network protocol layers
  - Application, transport, network, link, and physical
  - Internet “hourglass” model with IP as the “narrow waist”
- Customized protocol designs
  - Separation of concerns that are unique to each layer
  - Content naming and delivery, ordered reliable byte stream, scalable best-effort delivery, autoconfiguration of best-effort frame delivery