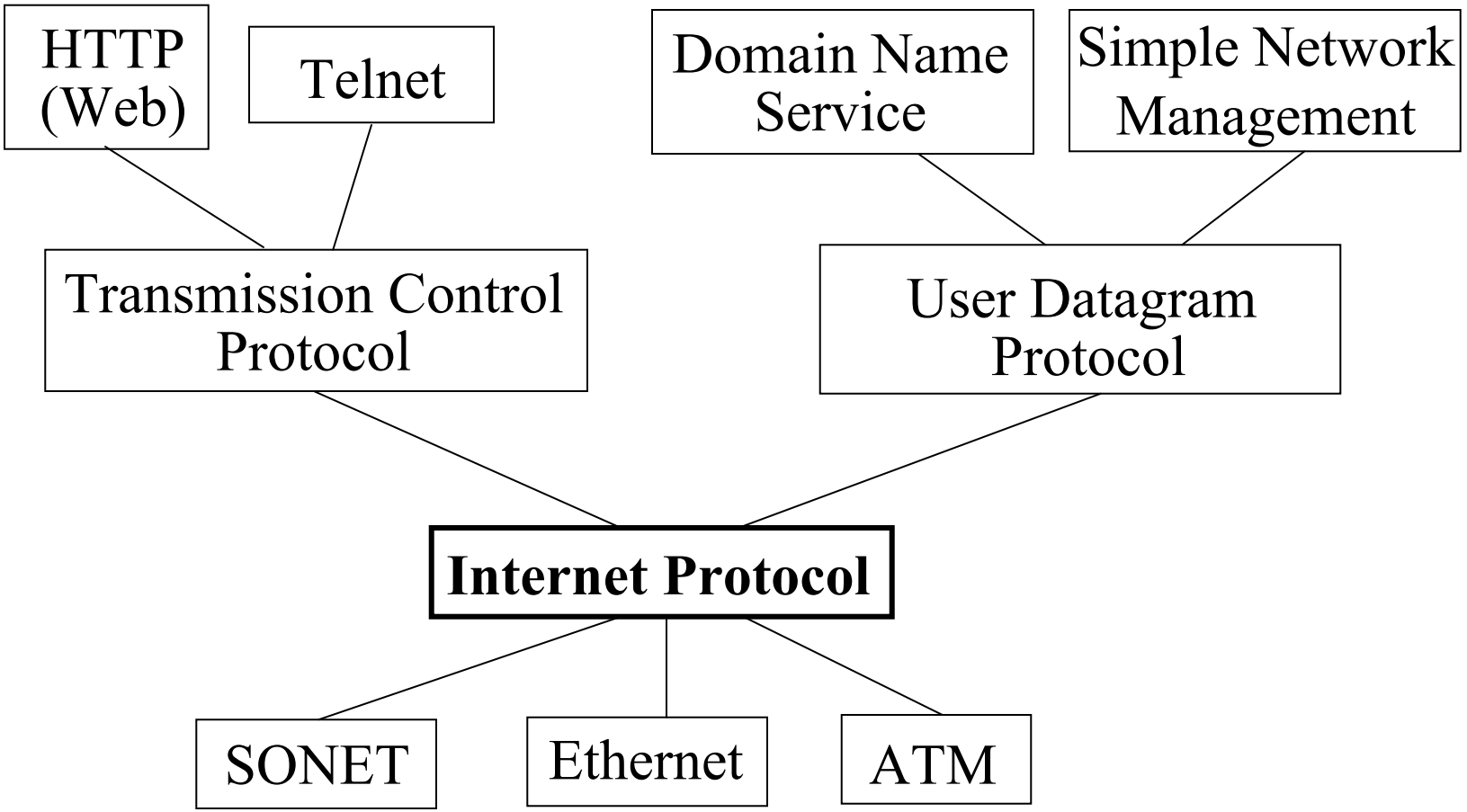
CPSC156a: The Internet
Co-Evolution of Technology and Society

Lecture 4: September 16, 2003

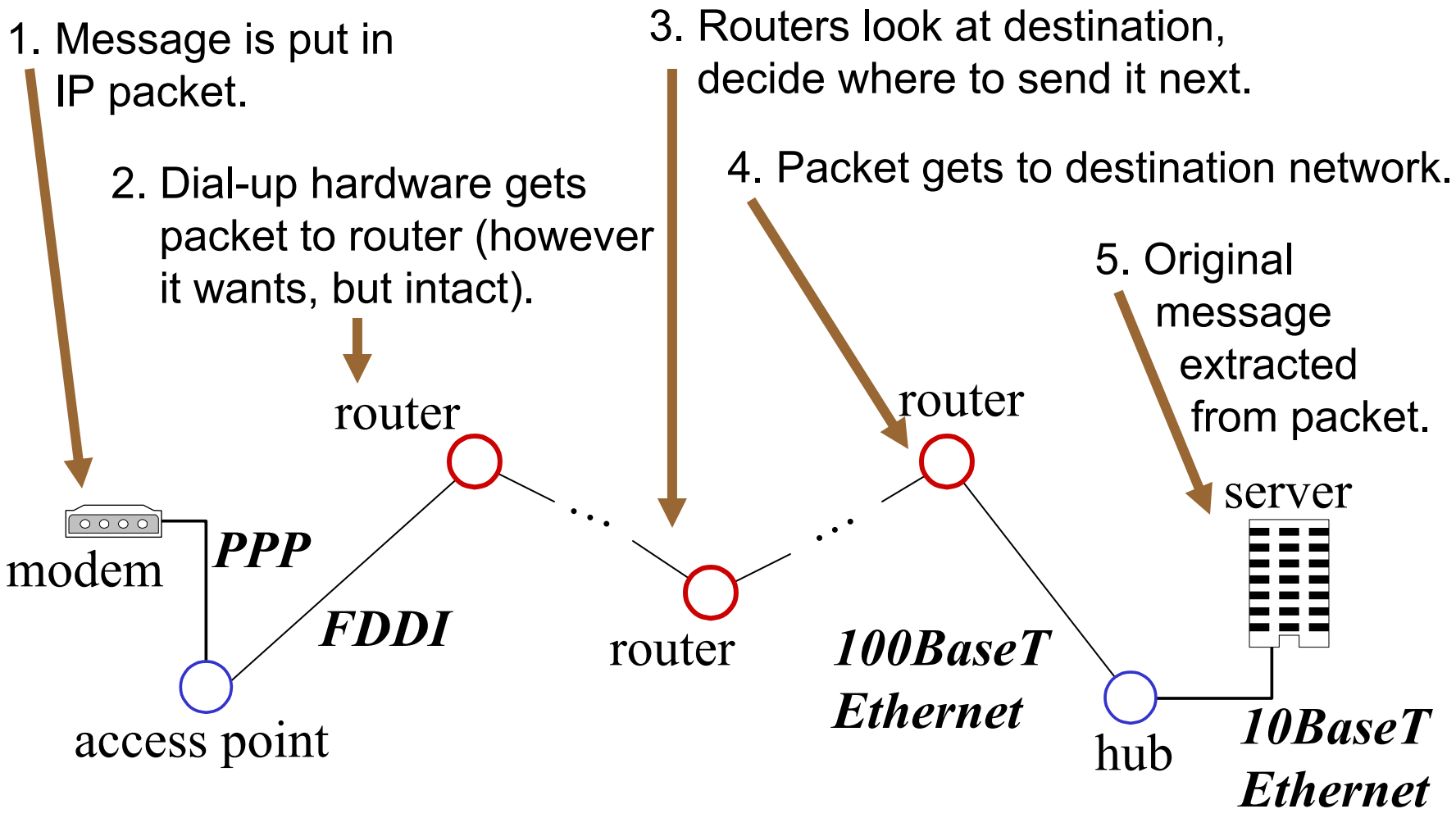
Internet Layers and "the Web"

Layering in the IP Protocols



The Role of the IP Layer

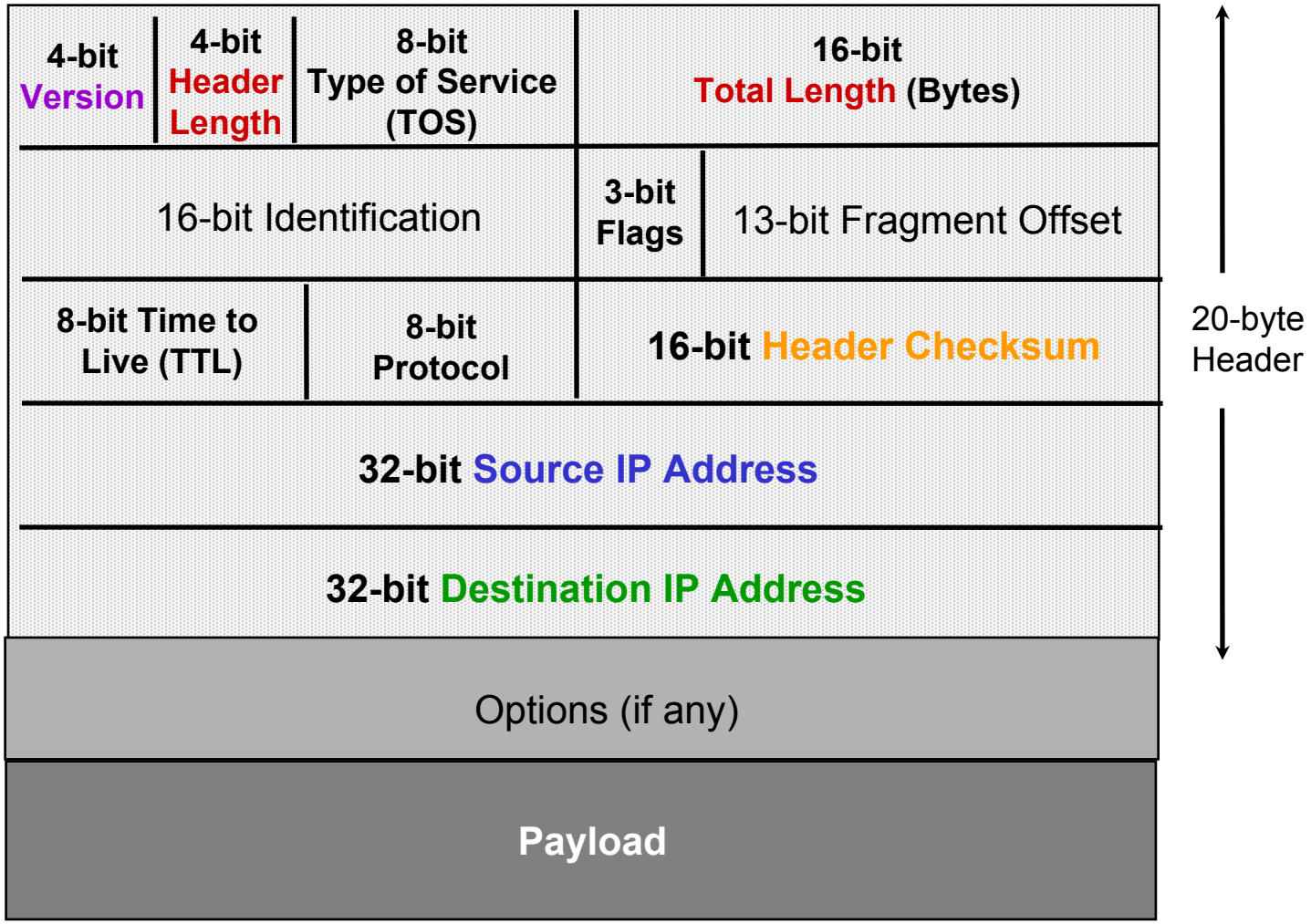
- **Internet Protocol (IP)**: gives a standard way to "package" messages across different hardware types.



IP Connectionless Paradigm

- No error detection or correction for packet data
  - Higher-level protocol can provide error checking
- Successive packets may not follow the same path
  - Not a problem as long as packets reach the destination
- Packets can be delivered out-of-order
  - Receiver can put packets back in order (if necessary)
- Packets may be lost or arbitrarily delayed
  - Sender can send the packets again (if desired)
- No network congestion control (beyond "drop")
  - Send can slow down in response to loss or delay

IP Packet Structure

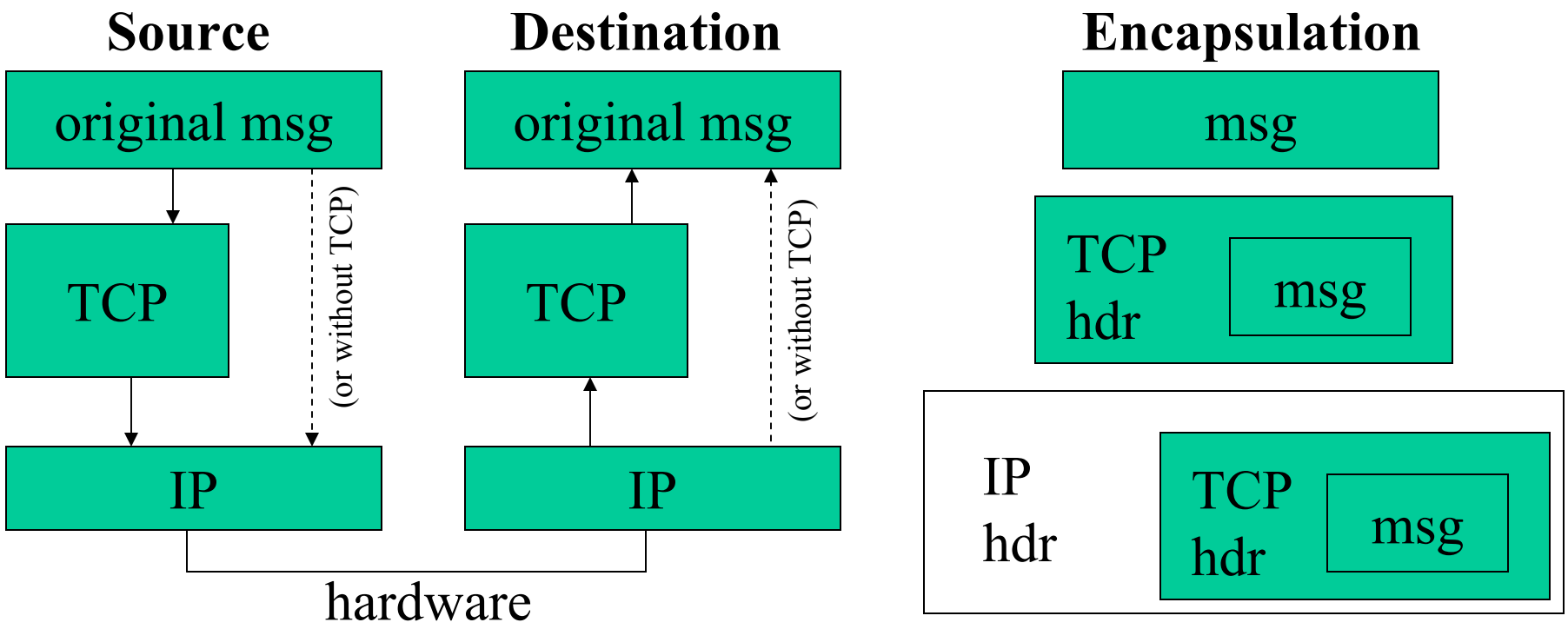


Main IP Header Fields

- **Version number** (e.g., version 4, version 6)
- **Header length** (number of 4-byte words)
- **Header checksum** (error check on header)
- **Source** and **destination** IP addresses
- Upper-level protocol (e.g., TCP, UDP)
- **Length** in bytes (up to 65,535 bytes)
- IP options (security, routing, timestamping, etc.)
- TTL (prevents messages from looping around forever; packets "die" if they "get lost")

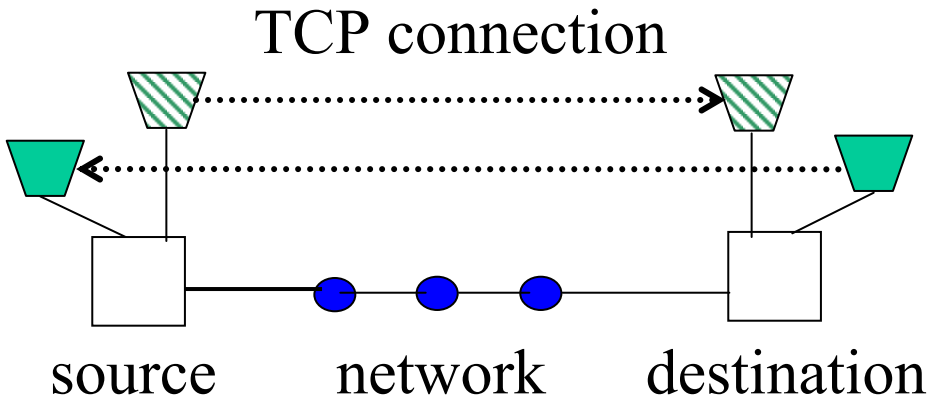
Adding Some Functionality

- More guarantees, *e.g.*, that packets go in order, require more work at both ends.
- Solution: add another layer (*e.g.*, TCP)

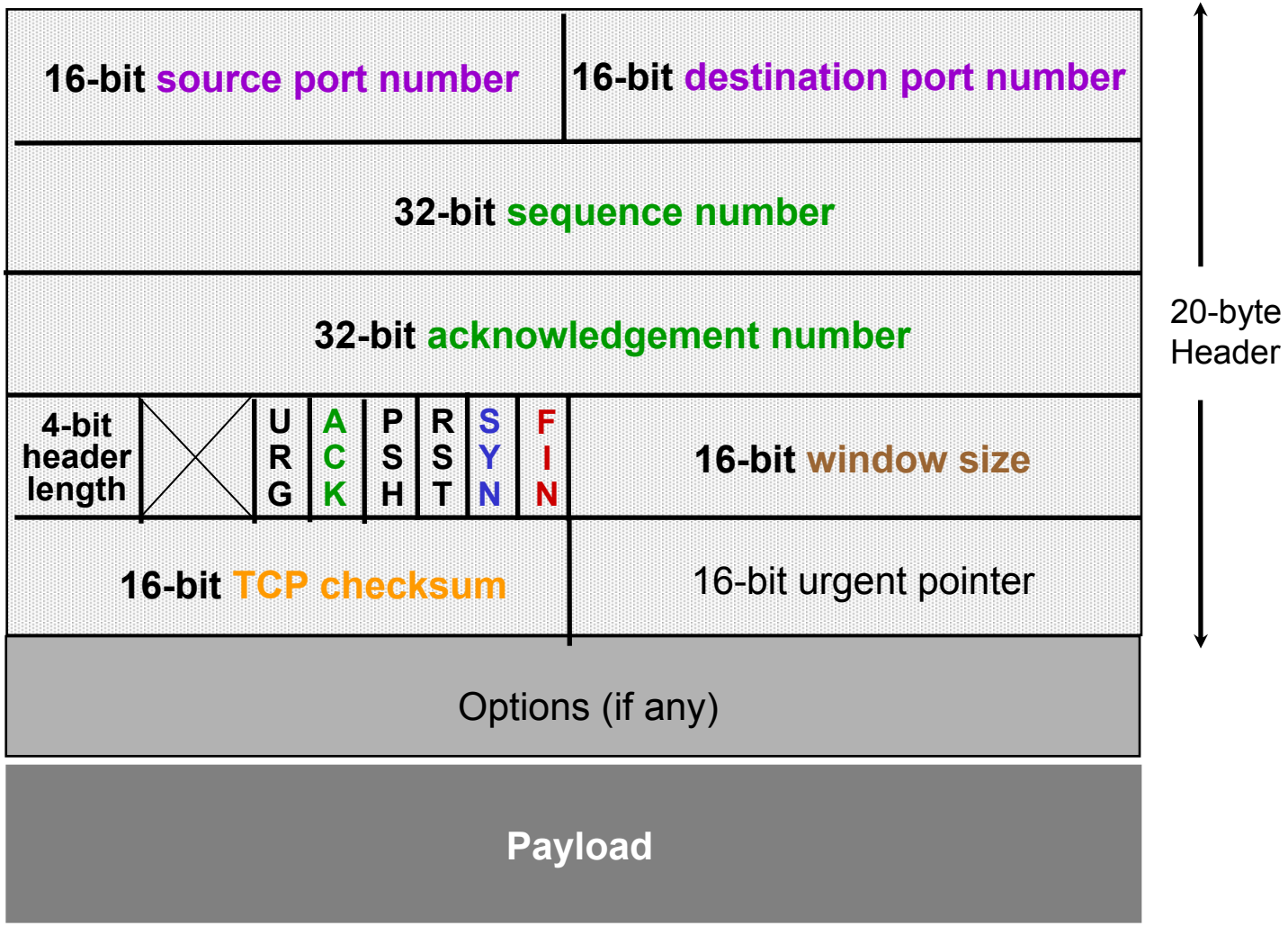


Transmission Control Protocol (TCP)

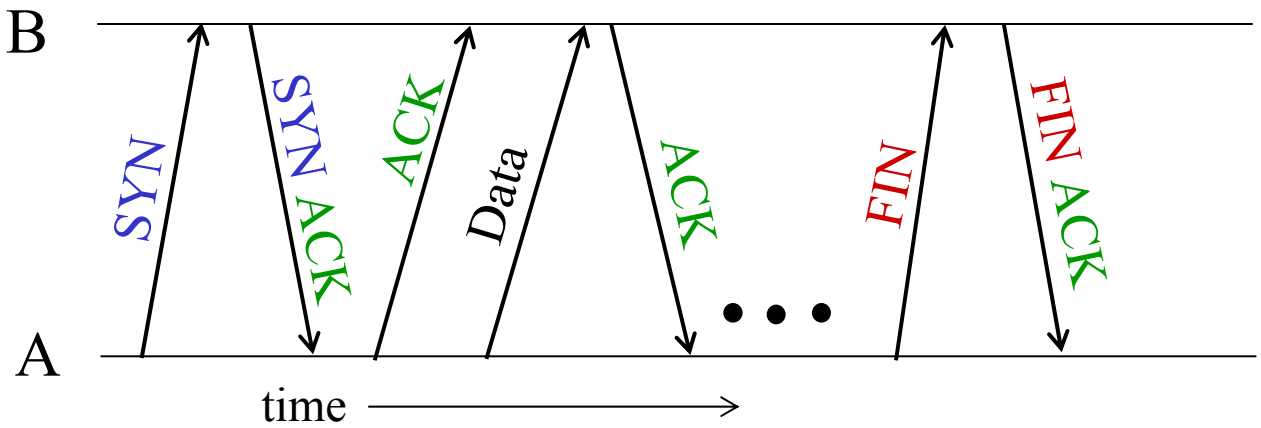
- Byte-stream socket abstraction for applications
- **Retransmission** of lost or corrupted packets
- **Flow-control** to respond to network congestion
- Simultaneous transmission in both directions
- **Multiplexing** of multiple logical connections



TCP Header


16-bit source port number
16-bit destination port number
32-bit sequence number
32-bit acknowledgement number
4-bit header length
URG
ACK
PSH
RST
SYN
FIN
16-bit window size
16-bit TCP checksum
16-bit urgent pointer
Options (if any)
Payload
20-byte Header

Establishing a TCP Connection



- Three-way handshake to establish connection
  - Host A sends a SYN (open) to the host B
  - Host B returns a SYN acknowledgment (ACK)
  - Host A sends an ACK to acknowledge the SYN ACK
- Closing the connection
  - Finish (FIN) to close and receive remaining bytes (and other host sends a FIN ACK to acknowledge)
  - Reset (RST) to close and not receive remaining bytes

Lost and Corrupted Packets

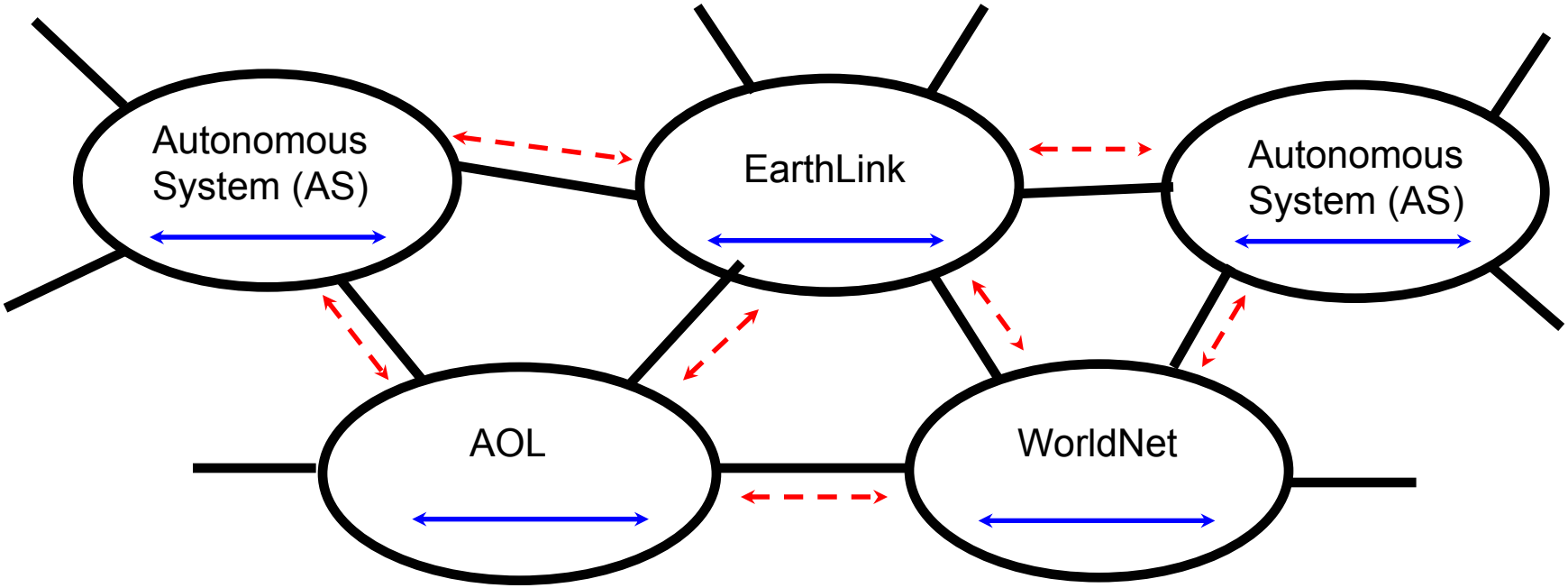
- Detecting corrupted and lost packets
  - Error detection via **checksum** on header and data
  - Sender sends packet, sets timeout, and waits for ACK
  - Receiver sends ACKs for received packets
- Retransmission from sender
  - Sender retransmits lost/corrupted packets
  - Receiver reassembles and reorders packets
  - Receiver discards corrupted and duplicated packets

Packet loss rates are high (e.g., 10%), causing significant delay (especially for short Web transfers)!

User Datagram Protocol (UDP)

- Some applications do not want or need TCP
  - Don't need recovery from lost or corrupted packets
  - Don't want flow control to respond to loss/congestion
- Fraction of UDP packets is rapidly increasing
  - Commonly used for multimedia applications
  - UDP traffic interferes with TCP performance
  - But, many firewalls do not accept UDP packets
- Dealing with the growth in UDP traffic
  - Pressure for applications to apply flow control
  - Future routers may enforce "TCP-like" behavior
  - Need better mathematical models of TCP behavior

Connecting Networks



Autonomous System: A collection of IP subnets and routers under the same administrative authority.

——— Interior Routing Protocol (e.g., Open Shortest Path First)

- - - - - - Exterior Routing Protocol (e.g., Border Gateway Protocol)

Where to Go Next

- Routers contain a **forwarding table** that pairs destination with next hop (on what physical wire to send msg.).
- The table gets populated with information learned **internally** (*e.g.*, OSPF) and **externally** (*e.g.*, BGP).
- OSPF and BGP are protocols that communicate ***knowledge about destinations*** between routers.

Getting from A to B: Summary

- Need IP addresses for:
  - Self (to use as source address)
  - DNS Server (to map names to addresses)
  - Default router to reach other hosts (e.g., gateway)
- Use DNS to get destination address
- Pass message through TCP/IP handler
- Send it off! **Routers** will do the work:
  - Physically connecting different networks
  - Deciding where to next send packets

HTTP (Hypertext Transfer Protocol)

- Standard protocol for web transfer
- "Request-response" interaction between clients and servers
- Request methods: GET, HEAD, PUT, POST, DELETE,…
- Response: Status line + additional info (*e.g.*, a web page)

Example, from HW1, of a request line:

```
<form action="http://lab.zoo.cs.yale.edu/cs156/cgi-
  bin/sendform.cgi" method="post">
```

HTML (Hypertext Markup Language)

- Language in which web pages are written
- Contains formatting commands
- Tells browser what to display and how to display

```
<TITLE> Welcome to Yale </TITLE>
```

- The title of this page is "Welcome to Yale"

```
<B> Great News! </B>
```

- Set "**Great News!**" in boldface

```
<A HREF="http://www.cs.yale.edu/index.html">
Yale Computer Science Department </A>
```

- A link pointing to the web page http://www.cs.yale.edu/index.html with the text "Yale Computer Science Department" displayed.

What does "http://www.cs.yale.edu/index.html" mean?

Protocol	Host, Domain Name	Local File
http	www.cs.yale.edu	index.html

Simple HTML Commands in HW1

```
Lastname:
<input type="text" name="lastname"><br>
Firstname:
<input type="text" name="firstname"
id="firstname"><br>
```

Lastname:

Firstname:

```
<input type="submit" value="submit form"><br>
```

Submit form

Discussion Point

- Packets, decentralized control, "intelligence at the endpoints"
- Now we've built a "byte-stream socket abstraction," pairs of which look like a "conversation."

Have we gained anything?

Yes: The WWW Revolution!

- Late 1990: WWW, HTTP, HTML, "Browser" invented by Tim Berners-Lee at CERN.
- Mid-1994: Mosaic Communications founded (later renamed to Netscape Communications).
- 1995: "Browsing" has become a universal pastime. IE ships with Windows 95.
- New businesses (*e.g.*, portal companies) enabled.
- Old businesses (*e.g.*, book selling) revolutionized.
- Triumph of Internet architecture and ethos: layering, "stupid network," open standards.

Reading Assignment for this Week

Chapter 1 of Information Rules, by Shapiro and Varian.

(Available in paper form only; distributed in class.)