

inst.eecs.berkeley.edu/~cs61c
CS61C : Machine Structures

Lecture 36
I/O : Networks

2008-04-25



TA Brian Zimmer

`inst.eecs/~cs61c-th`

NASA To Develop Small Satellites



NASA has announced it will... produce small satellites, called 'nanosats,'... The satellites will work together in 'constellations' and facilitate networking in space...it will 'develop a fifth gen. telecomm. and networking system for Internet protocol-based and related services.'



<http://science slashdot.org/science/08/04/24/235208.shtml>

CS61C L36 I/O : Networks (1)

Zimmer © UCB

I/O Review

- I/O gives computers their **5 senses**
- I/O speed range is **12.5-million to one**
- Differences in processor and I/O speed → synchronize with I/O devices before use
- **Polling** works, but expensive
 - processor *repeatedly queries devices*
- **Interrupts** works, more complex
 - device *causes an exception*, causing OS to run and deal with the device
- I/O control leads to **Operating Systems**



Why Networks?

- Originally *sharing I/O devices* between **computers**
ex: printers
- Then *communicating* between **computers**
ex: file transfer protocol
- Then *communicating* between **people**
ex: e-mail
- Then *communicating* between **networks of computers**
ex: file sharing, www, ...



How Big is the Network (2007)?

~30 in 273 Soda

~525 in inst.cs.berkeley.edu

~6,400 in eecs & cs .berkeley.edu

(1999) ~50,000 in berkeley.edu (outdated)

~10,700,000 in .edu (2005: ~9,000,000)

~295,305,818 in US (05: ~217E6; 06: ~286.5E6; 07:~259E6)
(.net .com .edu .arpa .us .mil .org .gov)

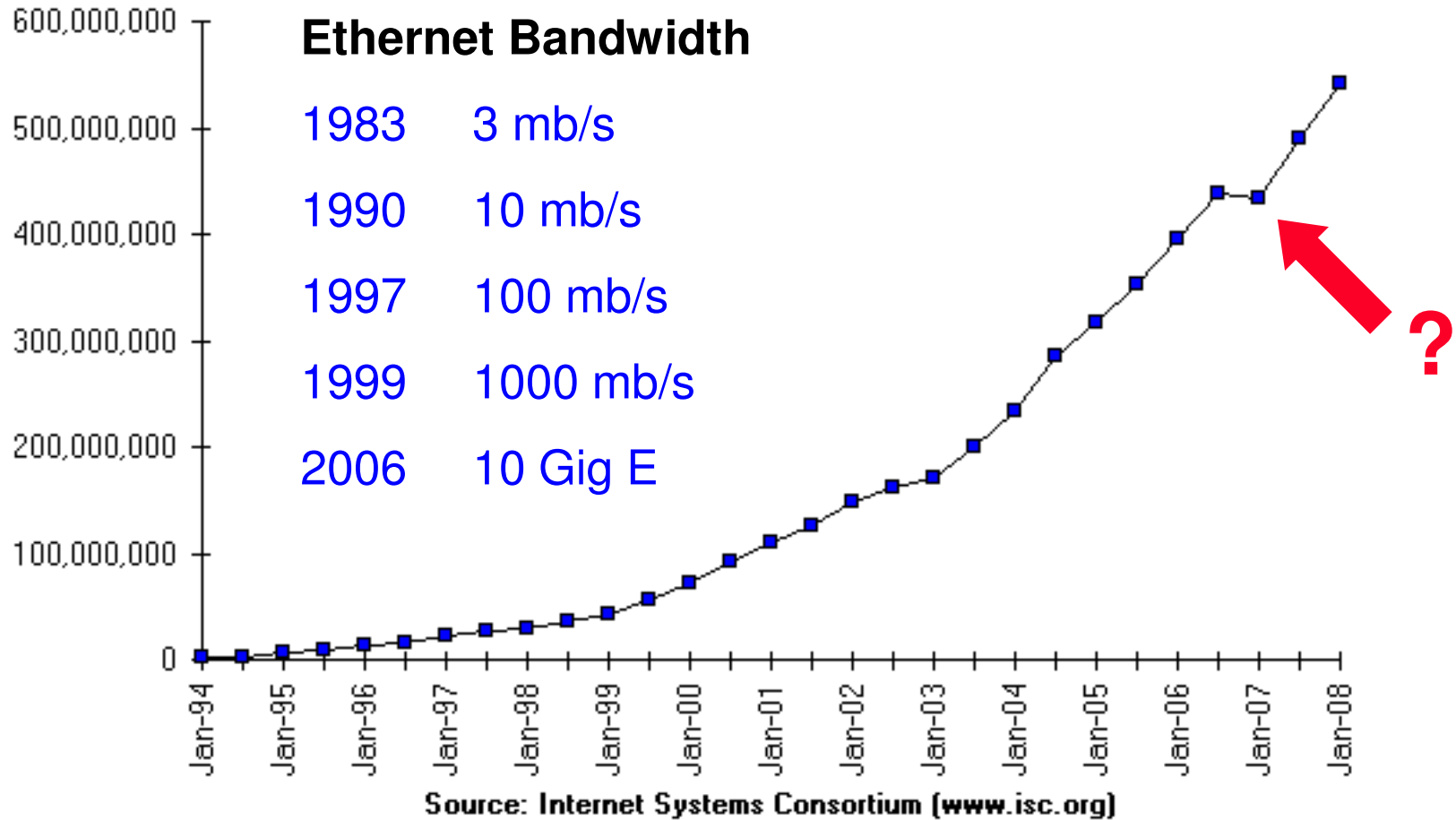
~541,677,360 in the world

(2005:~317,000,000; 2006: ~439,000,000; 2007: ~489,000,000)



Growth Rate

Internet Domain Survey Host Count



en.wikipedia.org/wiki/10_gigabit_ethernet

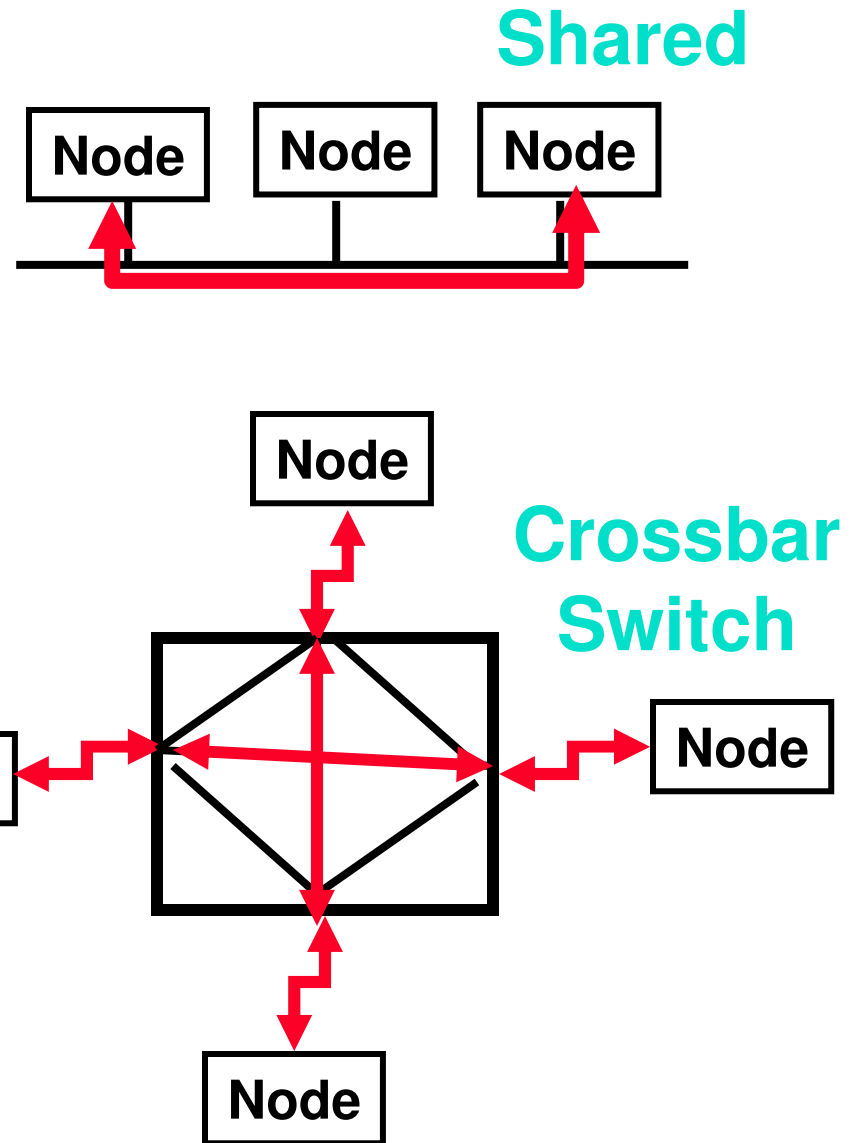
Shared vs. Switched Based Networks

- **Shared vs. Switched:**

- **Switched:** pairs (“[point-to-point](#)” connections) communicate at same time
- **Shared:** 1 at a time (CSMA/CD)

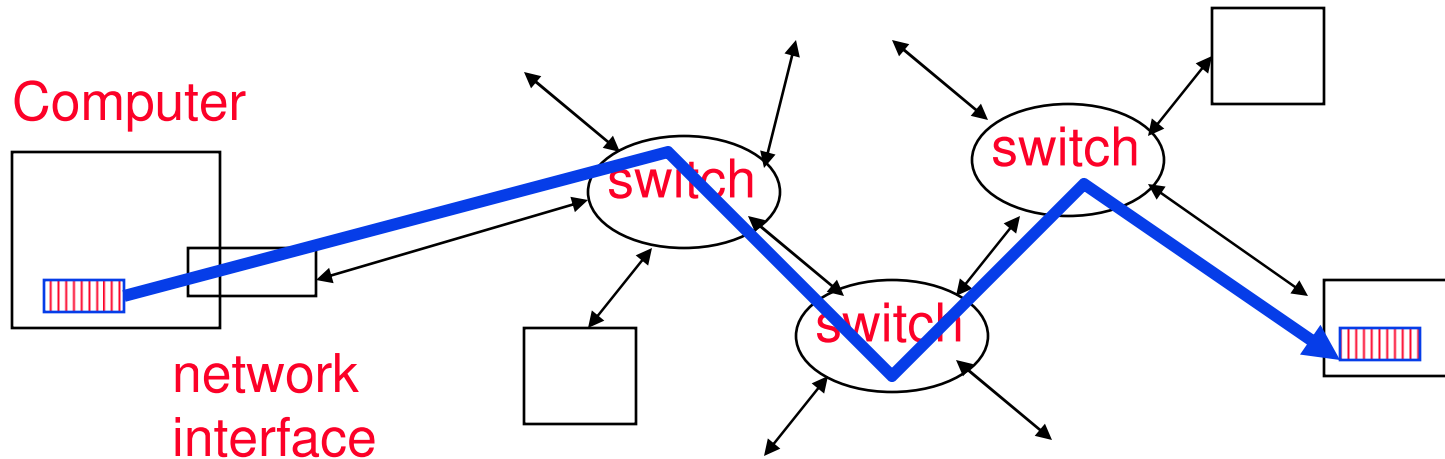
- **Aggregate bandwidth (BW) in switched network is many times shared:**

- point-to-point faster since **no arbitration**, simpler interface



What makes networks work?

- **links** connecting **switches** to each other and to computers or devices



- ability to **name** the components and to **route** packets of information - messages - from a source to a destination



- Layering, redundancy, protocols, and encapsulation as means of **abstraction** (61C big idea)



Typical Types of Networks

- **Local Area Network (Ethernet)**
 - Inside a building: Up to 1 km
 - (peak) Data Rate: 10 Mbits/sec, 100 Mbits/sec, 1000 Mbits/sec (1.25, 12.5, 125 MBytes/s)
 - Run, installed by network administrators
- **Wide Area Network**
 - Across a continent (10km to 10000 km)
 - (peak) Data Rate: 1.5 Mb/s to 10000 Mb/s
 - Run, installed by telecommunications companies (Sprint, UUNet[MCI], AT&T)



Wireless Networks (LAN), ...

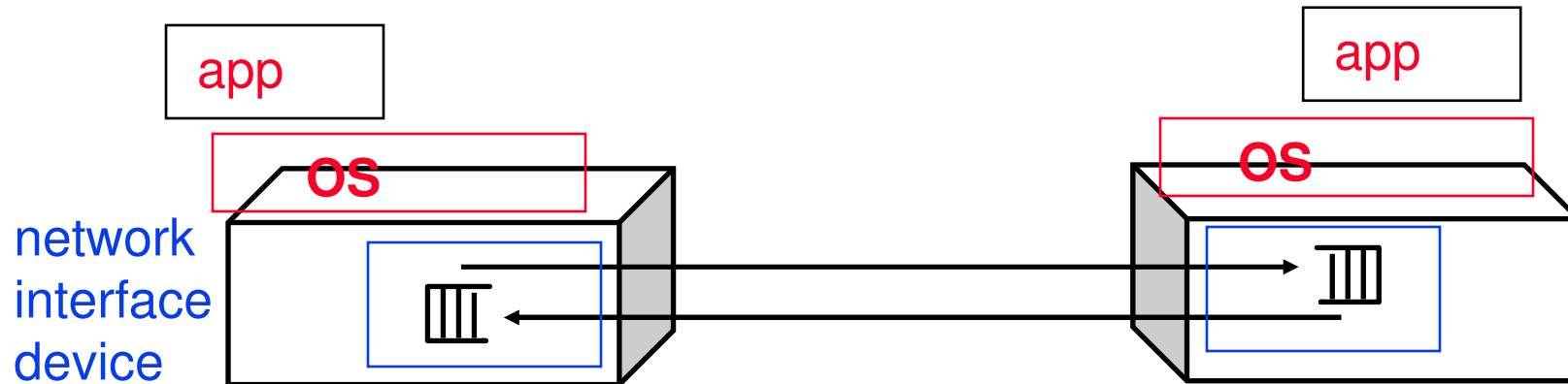
Administrivia

- **Crunch time**
 - **Last Lecture and Course Surveys on Monday 2008-05-12**
 - **Final Review Session on Sun 2008-05-18**
 - **3 weeks + 3 days until the Final ...**
 - **Final Exam on Monday 2008-05-19**
5 - 8 PM @ TBA - IS MANDATORY!
- **Project 4 is out!**
 - **May work in pairs.**
- **Do the performance competition!**
 - **Work in pairs!**



ABCs of Networks: 2 Computers

- **Starting Point: Send bits between 2 computers**

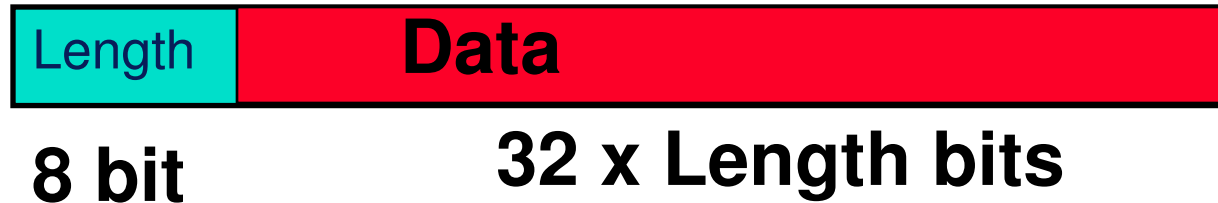


- Queue (First In First Out) on each end
- Can send both ways (“**Full Duplex**”)
 - One-way information is called “**Half Duplex**”
- Information sent called a “**message**”
 - Note: Messages also called **packets**



A Simple Example: 2 Computers

- **What is Message Format?**
 - Similar idea to Instruction Format
 - Fixed size? Number bits?



- **Header (Trailer)**: information to deliver message
- **Payload**: data in message
- What can be in the data?
 - anything that you can represent as bits
 - values, chars, commands, addresses...



Questions About Simple Example

- What if more than 2 computers want to communicate?
 - Need computer “address field” in packet to know:
 - which computer should receive it (destination)
 - which computer to reply to (source)
 - Just like envelopes!

Dest. Source Len



8 bits 8 bits 8 bits

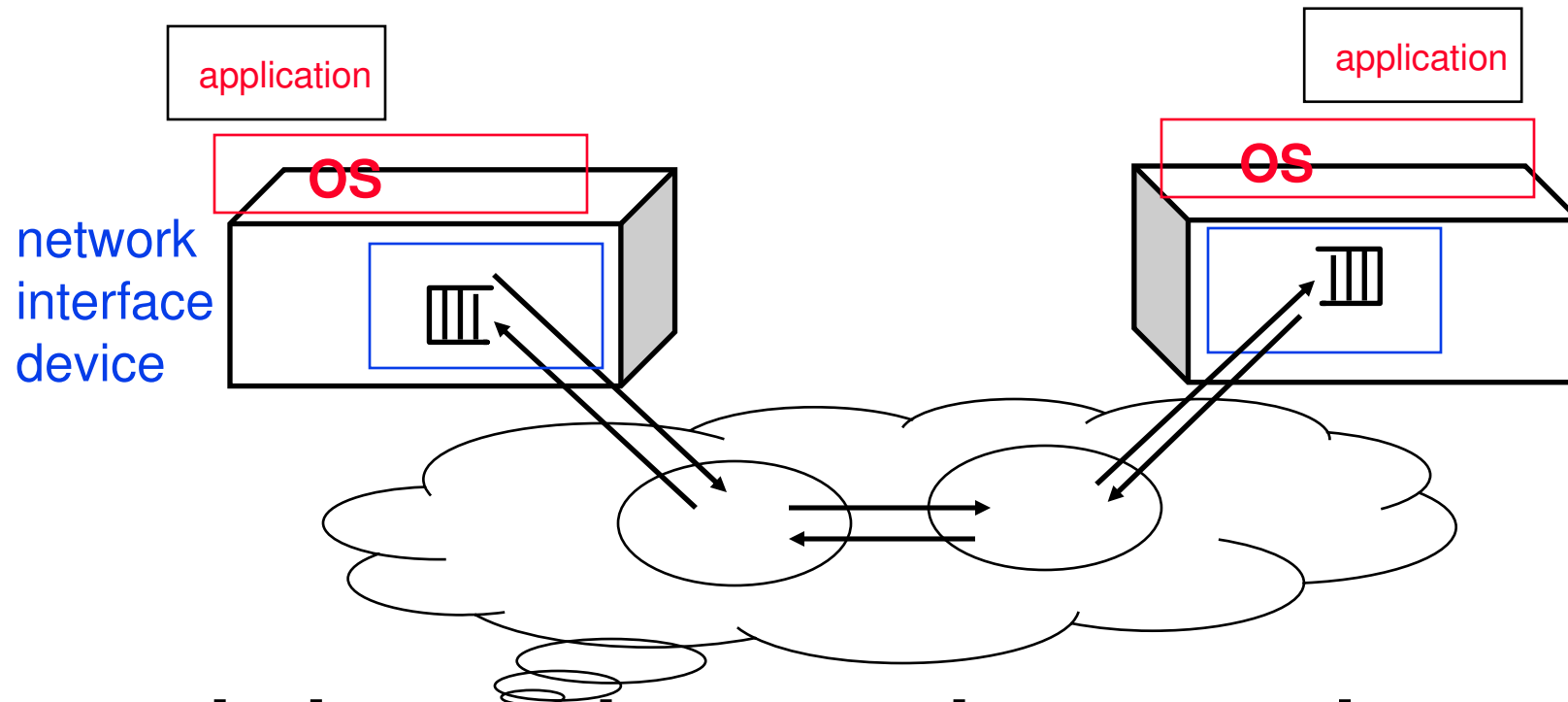
32*n bits

Header

Payload



ABCs: many computers



- **switches and routers interpret the header in order to deliver the packet**
- **source encodes and destination decodes content of the payload**

Questions About Simple Example

- What if message is garbled in transit?
- Add redundant information that is checked when message arrives to be sure it is OK
- 8-bit sum of other bytes: called “**Checksum**”; upon arrival compare checksum to sum of rest of information in message. **xor** also popular.

Checksum



Header

Payload

Trailer



Learn about Checksums in Math 55/CS 70...

Questions About Simple Example

- What if message never arrives?
- Receiver tells sender when it arrives
 - Send an ACK (ACKnowledgement) [like registered mail]
 - Sender retries if waits too long
- Don't discard message until it is ACK'ed
- If check sum fails, don't send ACK

Checksum



Header

Payload

Trailer



Observations About Simple Example

- **Simple questions (like those on the previous slides) lead to:**
 - **more complex procedures to send/receive message**
 - **more complex message formats**
- **Protocol: algorithm for properly sending and receiving messages (packets)**
 - ...an agreement on how to communicate



Software Protocol to Send and Receive

- **SW Send steps**

- 1: Application copies data to OS buffer

- 2: OS calculates checksum, starts timer

- 3: OS sends data to network interface HW and says start

- **SW Receive steps**

- 3: OS copies data from network interface HW to OS buffer

- 2: OS calculates checksum, if OK, send ACK; if not, delete message (sender resends when timer expires)

- 1: If OK, OS copies data to user address space, & signals application to continue



Protocol for Networks of Networks?

- **Abstraction** to cope with **complexity of communication** (compare to Abstraction for complexity of computation)

- **Networks are like onions**

- **Hierarchy of layers:**

- **Application (chat client, game, etc.)**
- **Transport (TCP, UDP)**
- **Network (IP)**
- **Physical Link (wired, wireless, etc.)**



Networks are like onions.

They stink?

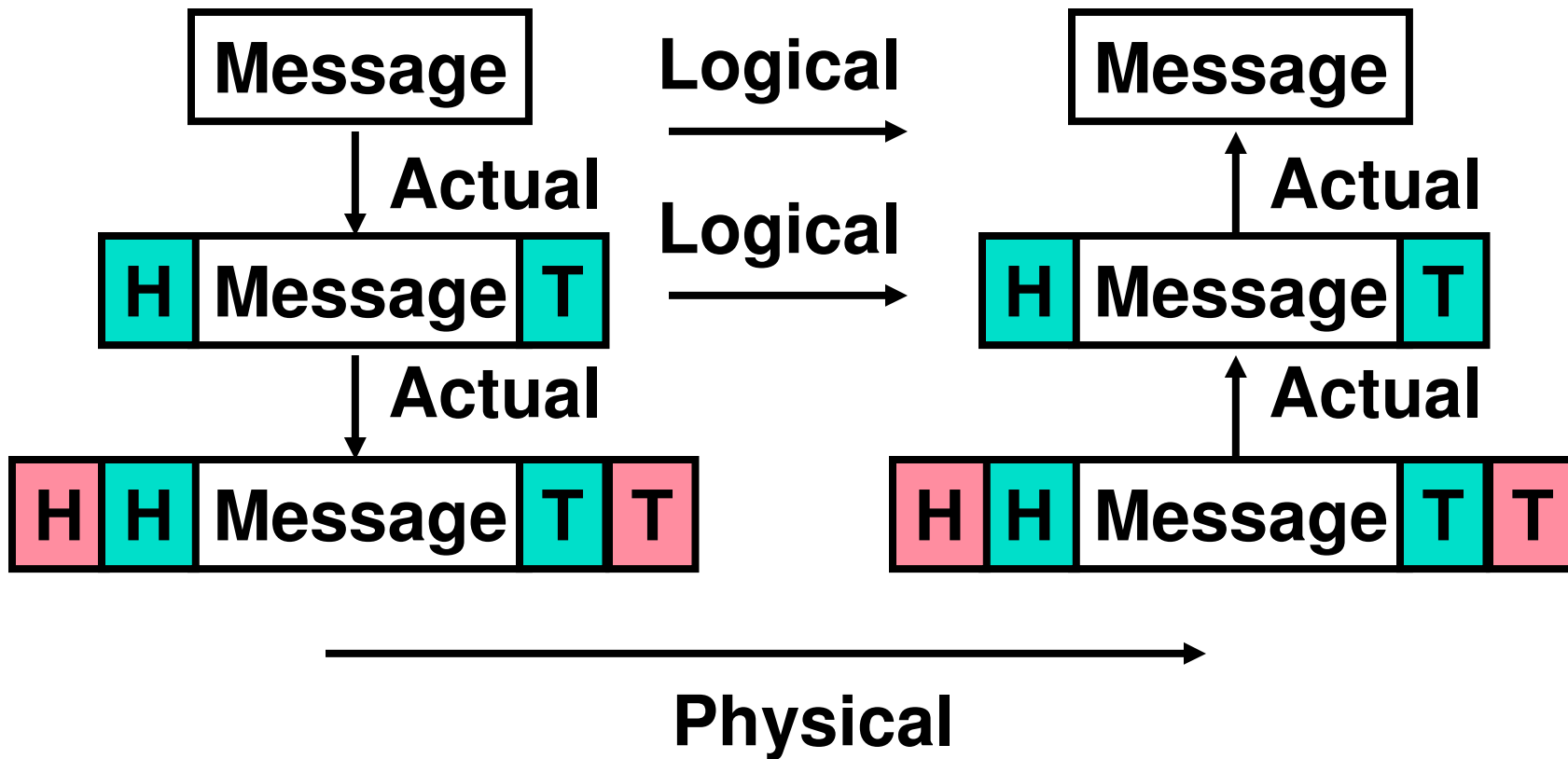
Yes. No!

Oh, they make you cry.

No!... Layers. Onions have layers.

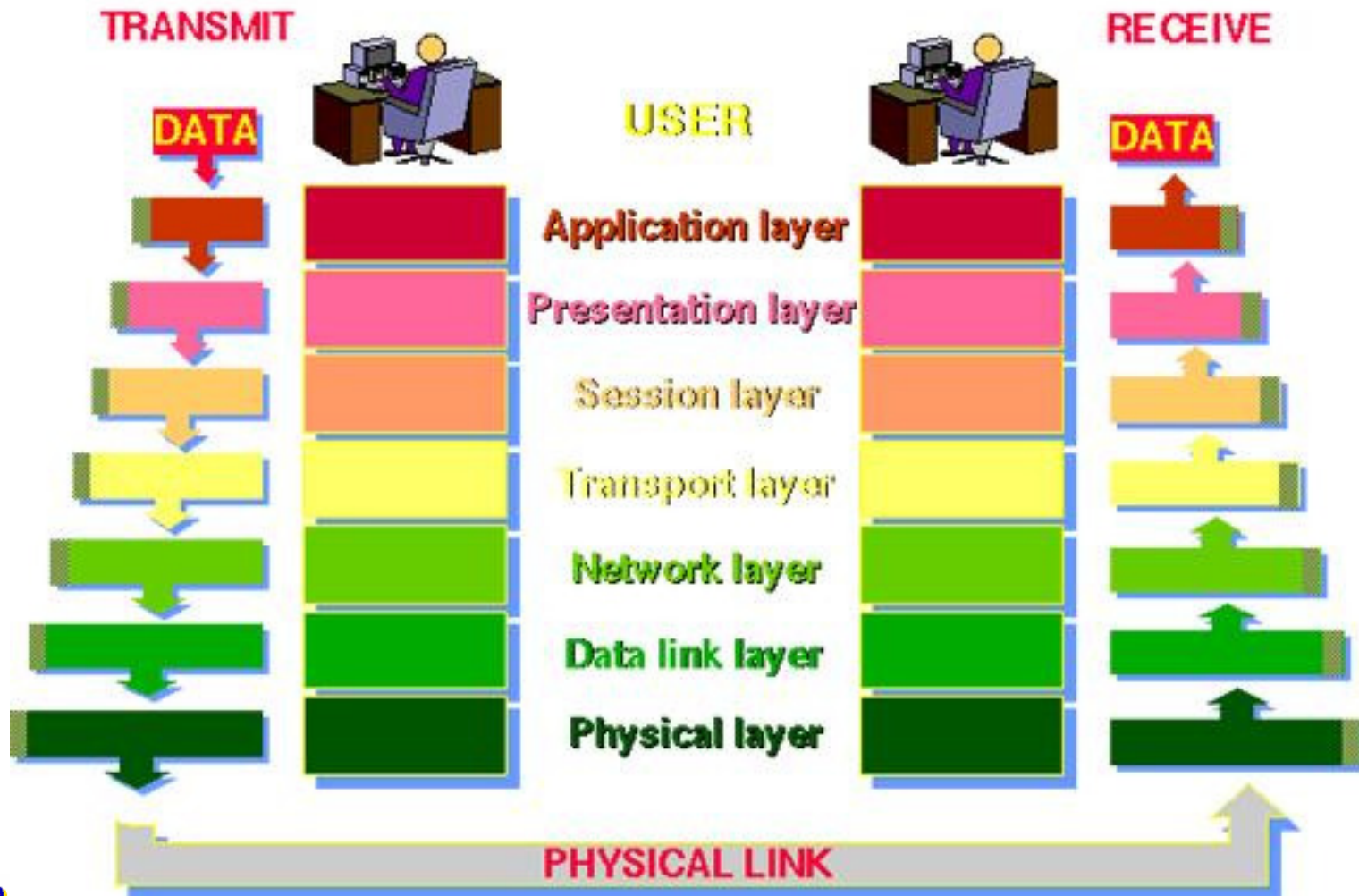
Networks have layers.

Protocol Family Concept

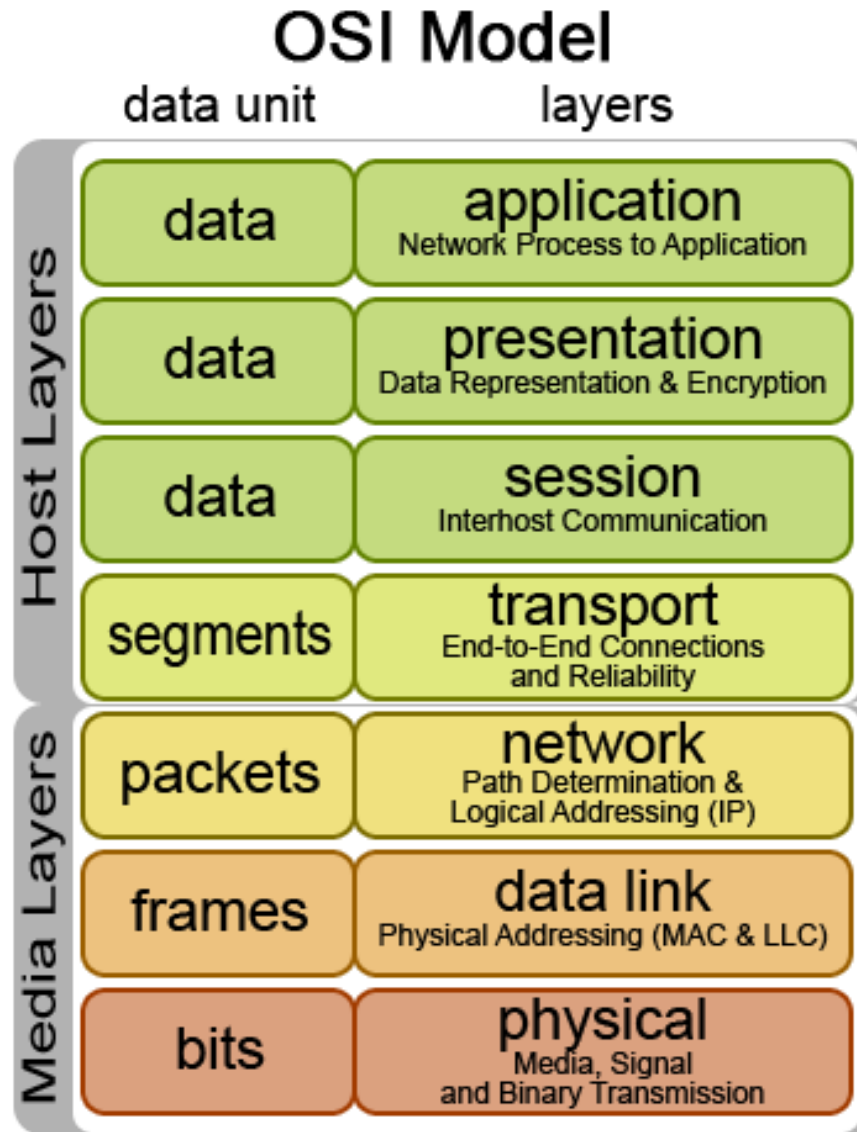


OSI Model

THE 7 LAYERS OF OSI



OSI Model



<http://wiki.go6.net/images/2/2b/Osi-model.png>



Protocol Family Concept

- Key to **protocol families** is that communication occurs **logically** at the same level of the protocol, called **peer-to-peer**...

...but is **implemented via services at the next lower level**
- **Encapsulation**: carry higher level information within lower level “envelope”
- **Fragmentation**: break packet into multiple smaller packets and reassemble



Protocol for Network of Networks

- Transmission Control Protocol/Internet Protocol (TCP/IP)

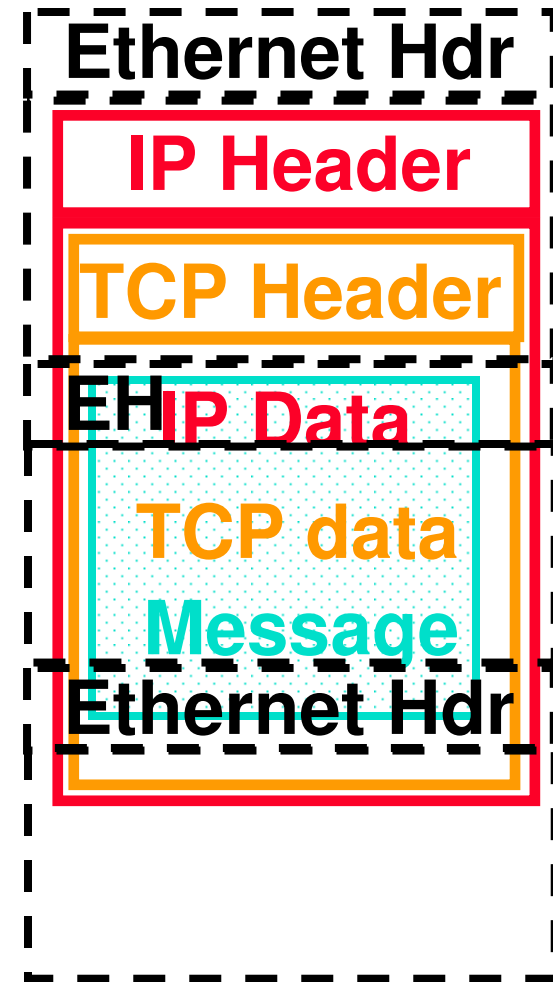
(TCP :: a Transport Layer)

- This protocol family is the **basis of the Internet**, a WAN protocol
- IP makes best effort to deliver
 - Packets can be lost, corrupted
- TCP guarantees delivery
- TCP/IP so popular it is used even when communicating locally: even across homogeneous LAN



TCP/IP packet, Ethernet packet, protocols

- Application sends message
- TCP breaks into 64KiB segments, adds 20B header
- IP adds 20B header, sends to network
- If Ethernet, broken into 1500B packets with headers, trailers (24B)
- All Headers, trailers have length field, destination,



Overhead vs. Bandwidth

- Networks are typically advertised using peak bandwidth of network link: e.g., 100 Mbits/sec Ethernet (“100 base T”)
- Software overhead to put message into network or get message out of network often limits useful bandwidth
- Assume overhead to send and receive = 320 microseconds (μs), want to send 1000 Bytes over “100 Mbit/s” Ethernet
 - Network transmission time:
 $1000\text{B} \times 8\text{b/B} / 100\text{Mb/s}$
 $= 8000\text{b} / (100\text{b}/\mu\text{s}) = 80 \mu\text{s}$
 - Effective bandwidth: $8000\text{b} / (320 + 80)\mu\text{s} = 20 \text{ Mb/s}$



And in conclusion...

- **Protocol suites allow networking of heterogeneous components**
 - Another form of principle of abstraction
 - Protocols \Rightarrow operation in presence of failures
 - Standardization key for LAN, WAN
- **Integrated circuit (“Moore’s Law”) revolutionizing network switches as well as processors**
 - Switch just a specialized computer
- **Trend from shared to switched networks to get faster links and scalable bandwidth**
- **Interested?**



- EE122 (CS-based in Fall, EE –based in Spring)

CS61C L36 I/O : Networks (26)

Zimmer © UCB

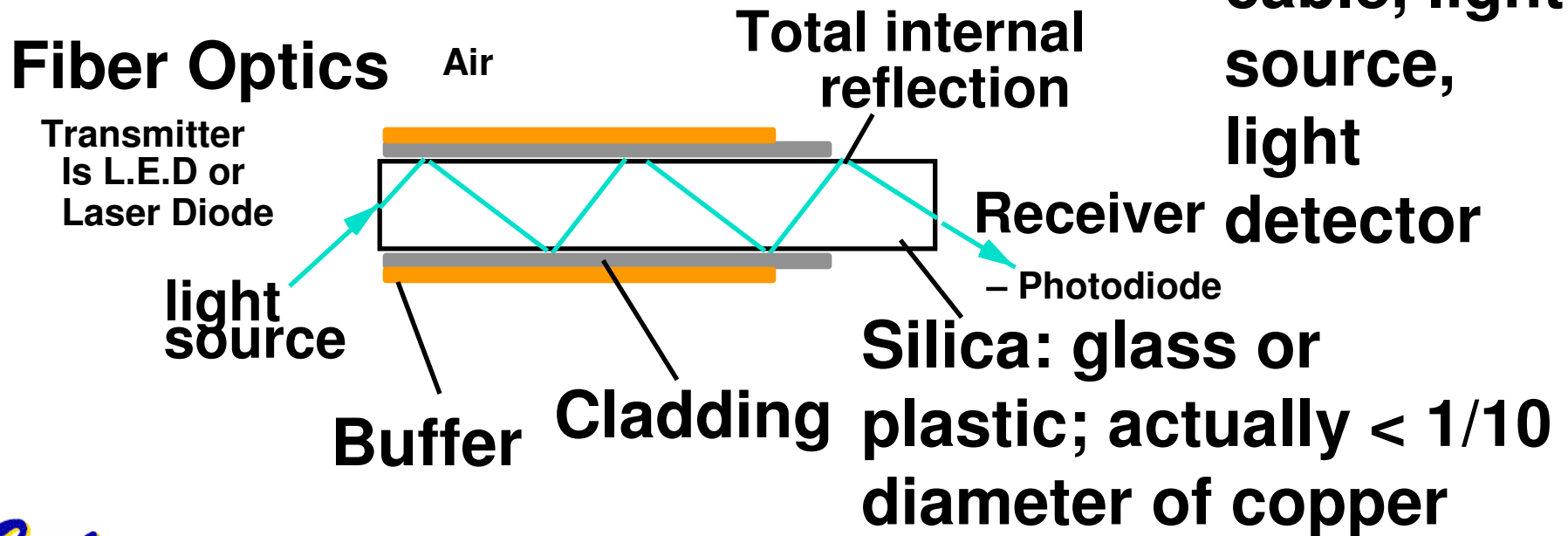
[Bonus] Example: Network Media

Twisted Pair
(“Cat 5”):

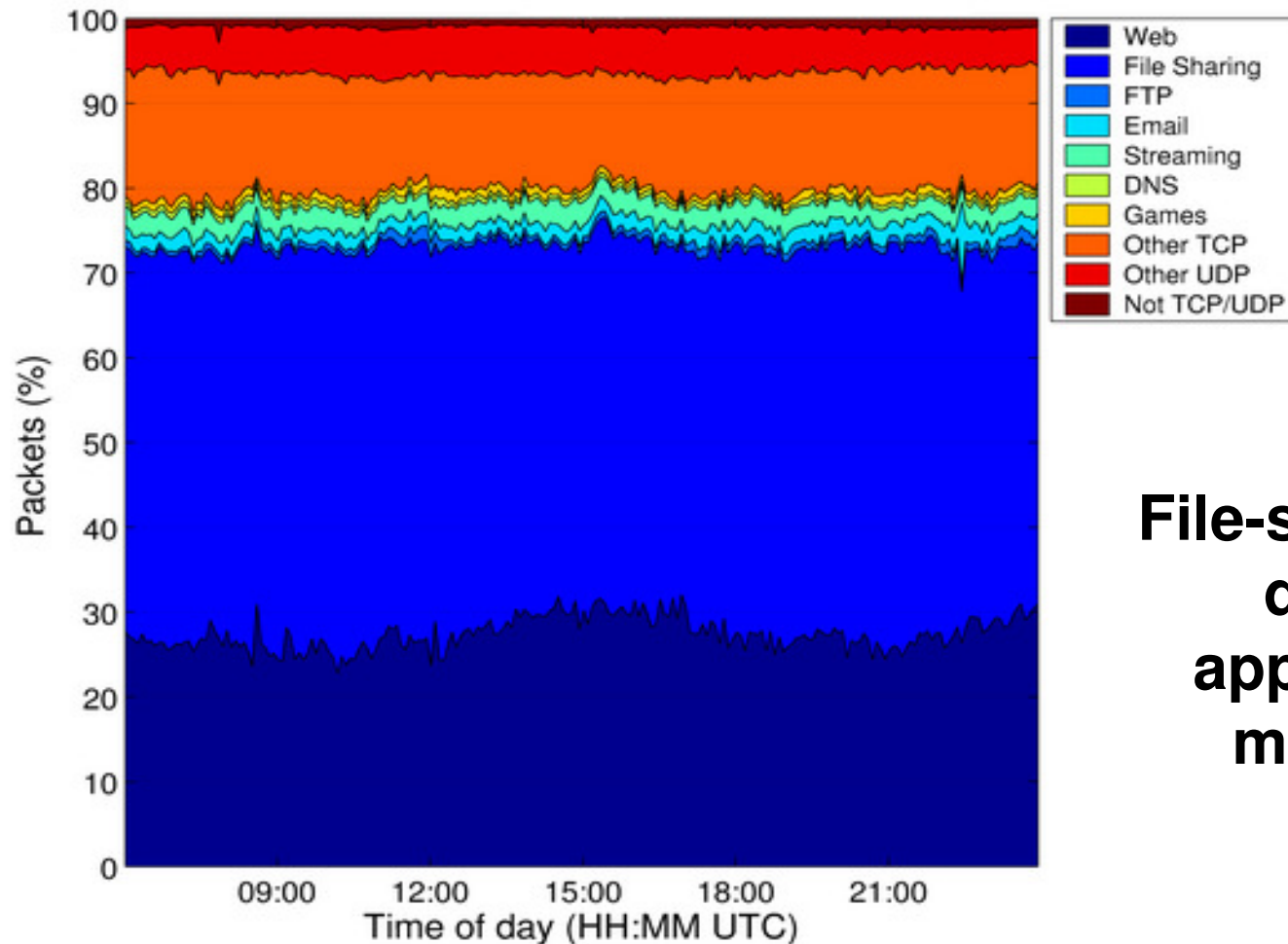


Copper, 1mm thick, twisted to avoid antenna effect

Light:
3 parts are cable, light source, light



[Bonus] Backbone Link App Composition

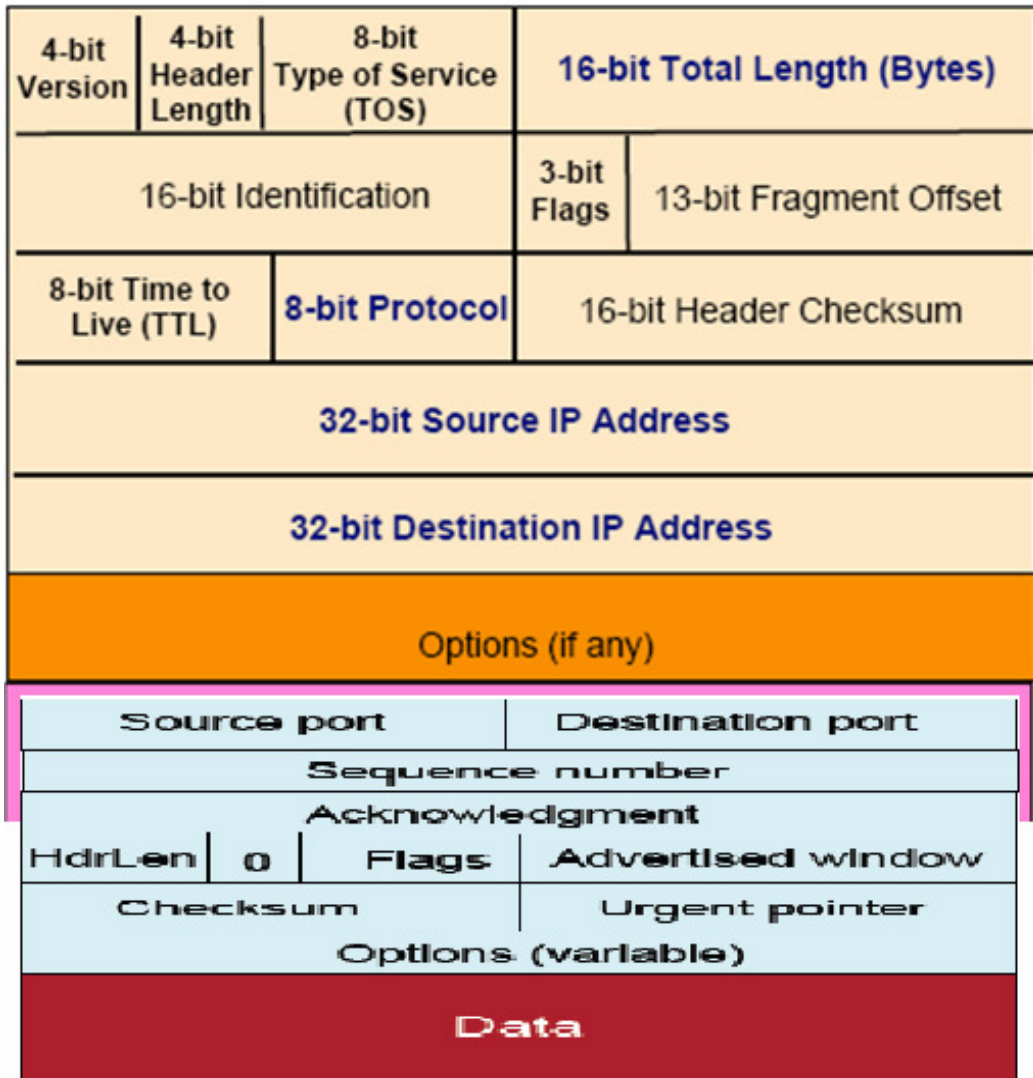


File-sharing is the dominant application on many links!



[Bonus] TCP/IP in action

Creating a Packet:



TCP
TCP Header

IP Header
IP



[Bonus] The Sprint U.S. Topology (2001)

