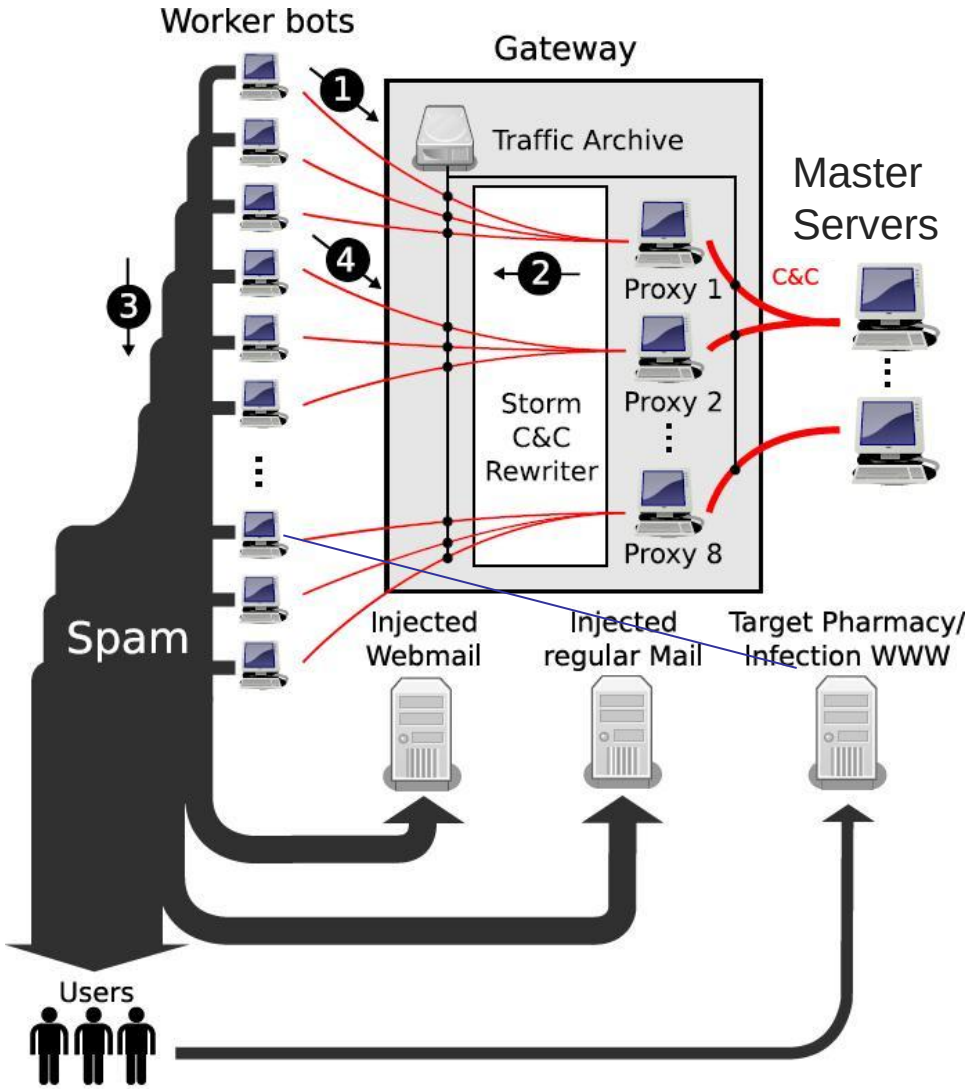
Discussion 1:
Project 1, Networking Basics

CSE 123: Computer Networks

Marti Motoyama & Chris Kanich

UCSDCSE

Spamalytics



CAPTCHAs


pursued
Vallc
3962
WK4BCP3
3RL9PGP5
340890
28951
JMHAGsd
probeholu
UBBTZWGU
TZFS R

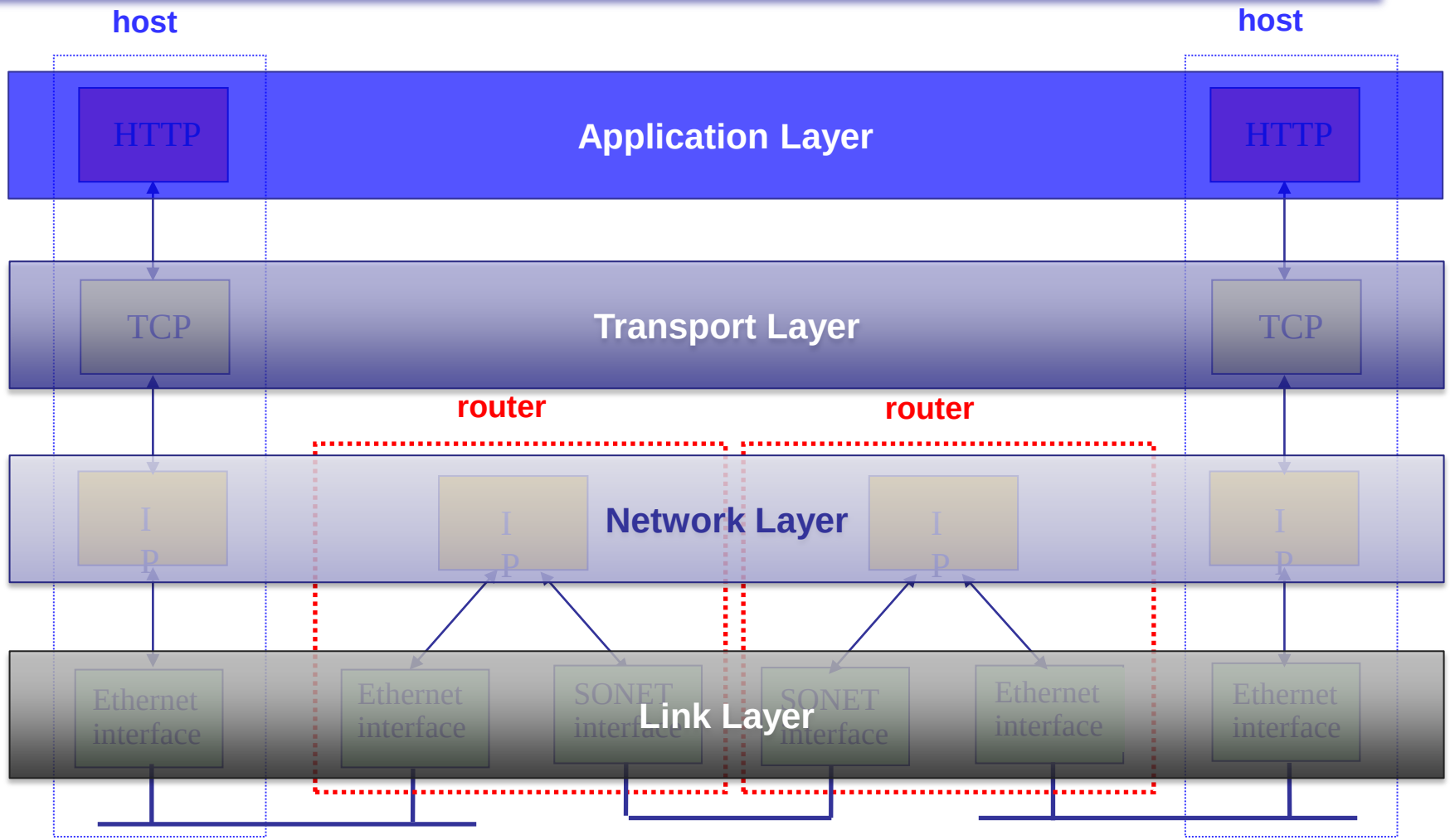
Measures of success

- How fast?
  - Bandwidth measured in bits per second
  - Often talk about KBps or Mbps – Bytes vs bits
- How long was the wait?
  - Delay (one-way or round trip) measured in seconds
- How efficiently?
  - Overhead measured in bits or seconds or cycles or…
- Any mistakes?
  - Error rate measured in terms of probability of flipped bit

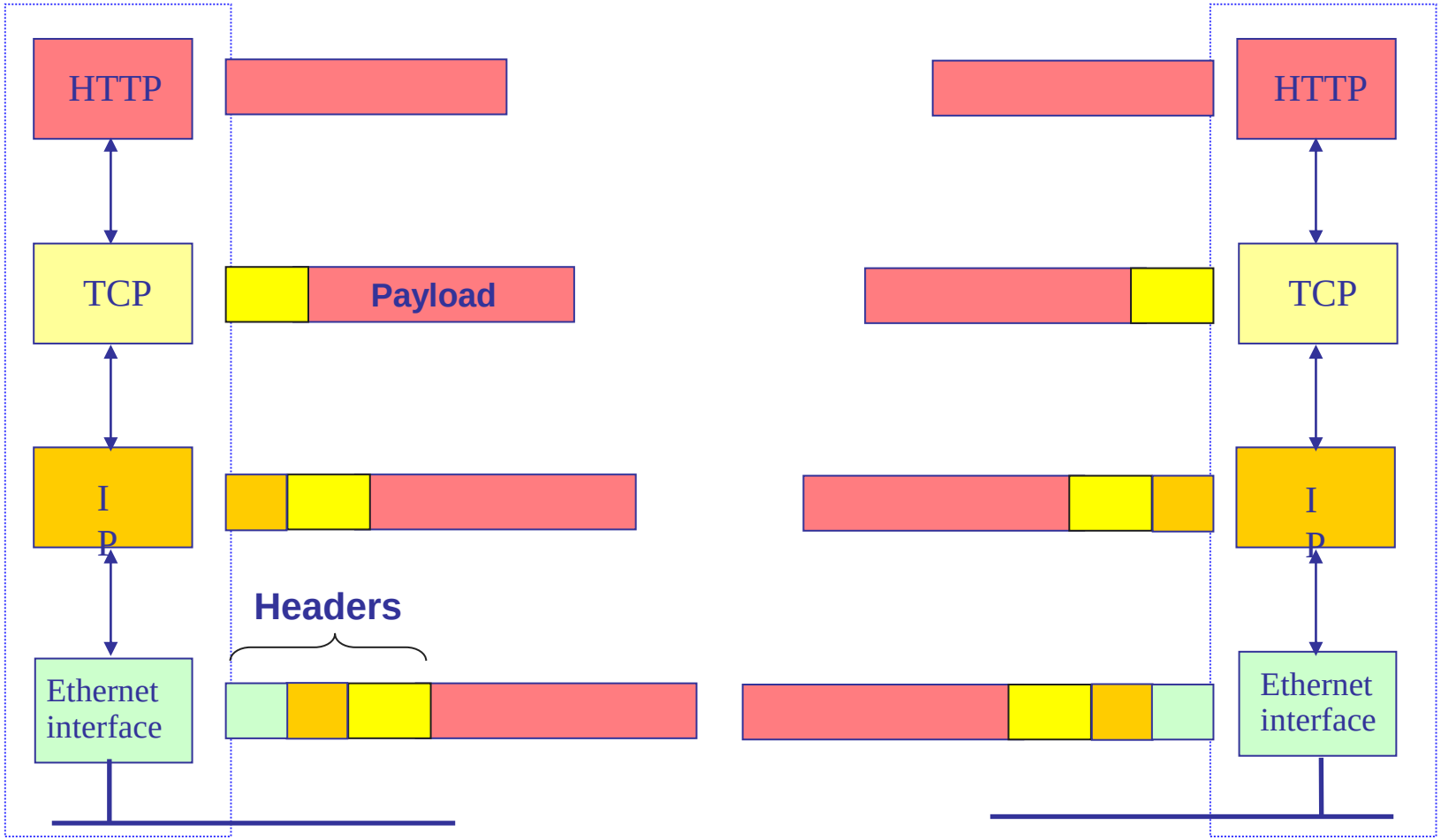
How long to send a message?

- Transmit time T = M/R + D
  - 10 Mbps Ethernet LAN (M=1KB)
    - » M/R=1ms, D ~=5us
  - 155 Mbps cross country ATM link (M=1KB)
    - » M/R = 50us, D ~= 40-100ms
- Where are the bits in the mean time?
  - In transit inside the network
- R*D is called the bandwidth delay product
  - How many bits can be “stored” be stored in transit
  - Colloquially, we say “fill the pipe”

TCP/IP Protocol Stack



Encapsulation



123.test - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: !(eth.dst == ff:ff:ff:ff:ff:ff) Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
26	3.237933	132.239.0.202	128.54.1.79	ICMP	Destination unreachable (Port unreachable)
33	3.886033	128.54.1.79	132.239.0.252	DNS	Standard query A www.facebook.com
34	3.888454	132.239.0.252	128.54.1.79	DNS	Standard query response A 66.220.147.33
35	3.889163	128.54.1.79	132.239.0.252	DNS	Standard query AAAA www.facebook.com
36	3.890698	132.239.0.252	128.54.1.79	DNS	Standard query response

```
⊞ Frame 33 (76 bytes on wire, 76 bytes captured)
⊟ Ethernet II, Src: IntelCor_ae:1b:2b (00:1f:3b:ae:1b:2b), Dst: Cisco_4a:70:00 (00:21:56:4a:70:00)
  ⊞ Destination: Cisco_4a:70:00 (00:21:56:4a:70:00)
  ⊞ Source: IntelCor_ae:1b:2b (00:1f:3b:ae:1b:2b)
    Type: IP (0x0800)
⊟ Internet Protocol, Src: 128.54.1.79 (128.54.1.79), Dst: 132.239.0.252 (132.239.0.252)
    Version: 4
    Header length: 20 bytes
  ⊞ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
    Total Length: 62
    Identification: 0x4fa3 (20387)
  ⊞ Flags: 0x00
    Fragment offset: 0
    Time to live: 128
    Protocol: UDP (0x11)
  ⊞ Header checksum: 0xe39b [correct]
    Source: 128.54.1.79 (128.54.1.79)
    Destination: 132.239.0.252 (132.239.0.252)
⊟ User Datagram Protocol, Src Port: 62619 (62619), Dst Port: domain (53)
    Source port: 62619 (62619)
    Destination port: domain (53)
    Length: 42
  ⊞ Checksum: 0x8bd0 [validation disabled]
⊞ Domain Name System (query)
```

Link Layer

Network Layer

Transport Layer

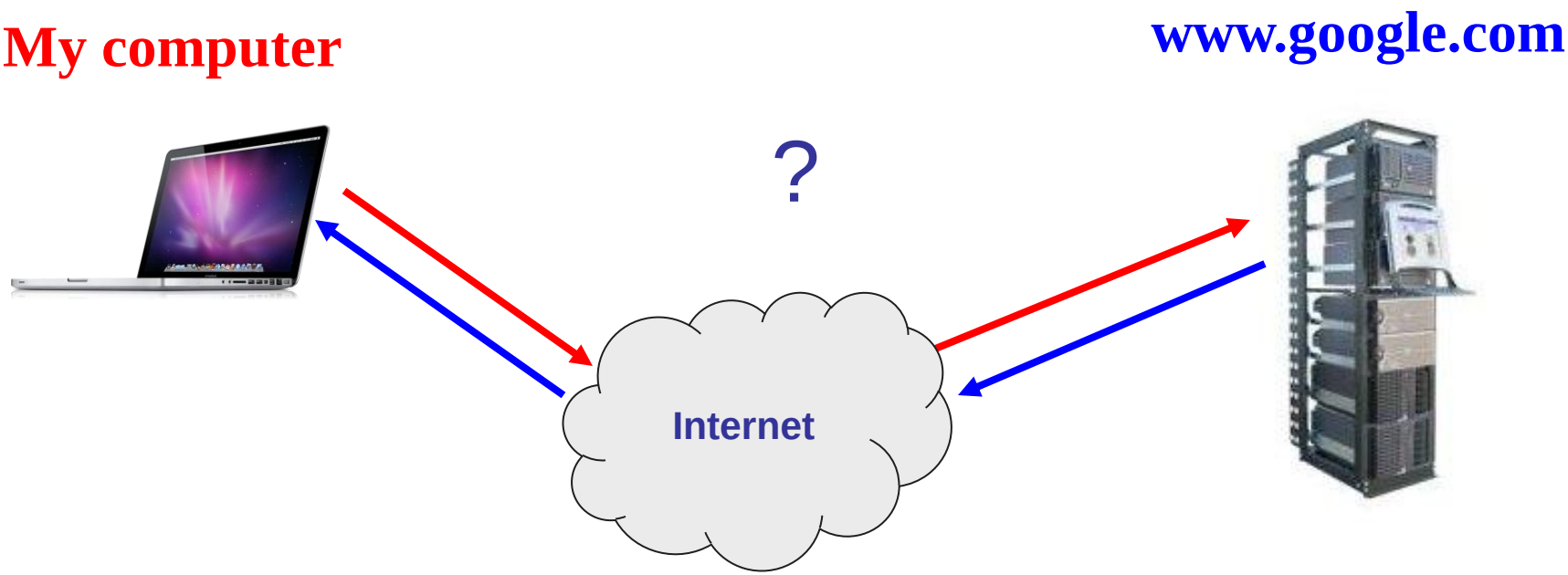
Application Layer

```
0000  00 21 56 4a 70 00 00 1f  3b ae 1b 2b 08 00 45 00   .!VJp... ;..+..E.
0010  00 3e 4f a3 00 00 80 11  e3 9b 80 36 01 4f 84 ef   .>O..... ...6.O..
0020  00 fc f4 9b 00 35 00 2a  8b d0 81 88 01 00 00 01   .....5.* ........
0030  00 00 00 00 00 00 03 77  77 77 08 66 61 63 65 62   .......w ww.faceb
0040  6f 6f 6b 03 63 6f 6d 00  00 01 00 01               ook.com. ....
```

Frame (frame), 76 bytes | Packets: 702 Displayed: 629 Marked: 0 | Profile: Default

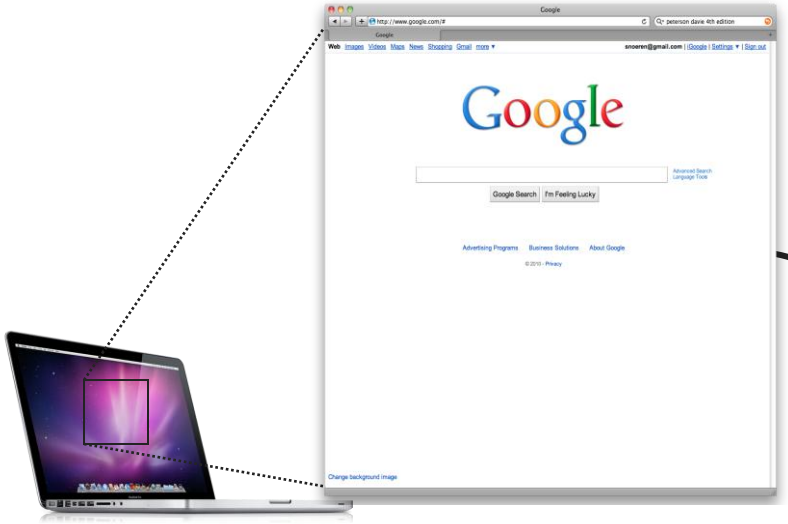
Putting this all together

- **ROUGHLY**, what happens when I click on a Web page from UCSD?



Web request (HTTP)

- Turn click into HTTP request



GET http://www.google.com/ HTTP/1.1
Host: www.google.com
Connection:keep-alive
…

Name resolution (DNS)

- Where is www.google.com?

My computer (132.239.9.64)

Local DNS server (132.239.51.18)

What's the address for www.google.com

Oh, you can find it at 66.102.7.104

123.test - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: !(eth.dst == ff:ff:ff:ff:ff:ff) Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
26	3.237933	132.239.0.202	128.54.1.79	ICMP	Destination unreachable (Port unreachable)
33	3.886033	128.54.1.79	132.239.0.252	DNS	Standard query A www.facebook.com
34	3.888454	132.239.0.252	128.54.1.79	DNS	Standard query response A 66.220.147.33
35	3.889163	128.54.1.79	132.239.0.252	DNS	Standard query AAAA www.facebook.com
36	3.890698	132.239.0.252	128.54.1.79	DNS	Standard query response
37	3.959996	128.54.1.79	66.220.147.33	TCP	53792 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460
38	3.977502	66.220.147.33	128.54.1.79	TCP	http > 53792 [SYN, ACK] Seq=0 Ack=1 Win=4380 Len=0 MSS=1460
39	3.977671	128.54.1.79	66.220.147.33	TCP	53792 > http [ACK] Seq=1 Ack=1 Win=64240 Len=0
40	3.981512	128.54.1.79	66.220.147.33	HTTP	GET / HTTP/1.1
42	4.177980	66.220.147.33	128.54.1.79	TCP	http > 53792 [ACK] Seq=1 Ack=558 Win=4937 Len=0
43	4.178344	66.220.147.33	128.54.1.79	TCP	[TCP segment of a reassembled PDU]
45	4.390254	128.54.1.79	66.220.147.33	TCP	53792 > http [ACK] Seq=558 Ack=1461 Win=64240 Len=0
46	4.407936	66.220.147.33	128.54.1.79	TCP	[TCP segment of a reassembled PDU]
47	4.417271	128.54.1.79	132.239.0.252	DNS	Standard query A static.ak.fbcdn.net
48	4.419163	132.239.0.252	128.54.1.79	DNS	Standard query response CNAME static.ak.facebook.com.edgesuit
49	4.419929	128.54.1.79	132.239.0.252	DNS	Standard query AAAA static.ak.fbcdn.net
50	4.421474	132.239.0.252	128.54.1.79	DNS	Standard query response CNAME static.ak.facebook.com.edgesuit

```
⊞ Frame 33 (76 bytes on wire, 76 bytes captured)
⊞ Ethernet II, Src: IntelCor_ae:1b:2b (00:1f:3b:ae:1b:2b), Dst: Cisco_4a:70:00 (00:21:56:4a:70:00)
⊞ Internet Protocol, Src: 128.54.1.79 (128.54.1.79), Dst: 132.239.0.252 (132.239.0.252)
⊞ User Datagram Protocol, Src Port: 62619 (62619), Dst Port: domain (53)
⊞ Domain Name System (query)
```

```
0000  00 21 56 4a 70 00 00 1f  3b ae 1b 2b 08 00 45 00   .!VJp... ;..+..E.
0010  00 3e 4f a3 00 00 80 11  e3 9b 80 36 01 4f 84 ef   .>O..... ...6.O..
0020  00 fc f4 9b 00 35 00 2a  8b d0 81 88 01 00 00 01   .....5.* ........
0030  00 00 00 00 00 00 03 77  77 77 08 66 61 63 65 62   .......w ww.faceb
0040  6f 6f 6b 03 63 6f 6d 00  00 01 00 01               ook.com. ....
```

File: "C:\Users\ckanich\Documents\123.test" 38... Packets: 702 Displayed: 629 Marked: 0 Profile: Default

Data transport (TCP)

- Break message into packets (TCP segments)
- Should be delivered reliably & in-order

123.test - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: !(eth.dst == ff:ff:ff:ff:ff:ff) Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
26	3.237933	132.239.0.202	128.54.1.79	ICMP	Destination unreachable (Port unreachable)
33	3.886033	128.54.1.79	132.239.0.252	DNS	Standard query A www.facebook.com
34	3.888454	132.239.0.252	128.54.1.79	DNS	Standard query response A 66.220.147.33
35	3.889163	128.54.1.79	132.239.0.252	DNS	Standard query AAAA www.facebook.com
36	3.890698	132.239.0.252	128.54.1.79	DNS	Standard query response
37	3.959996	128.54.1.79	66.220.147.33	TCP	53792 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460
38	3.977502	66.220.147.33	128.54.1.79	TCP	http > 53792 [SYN, ACK] Seq=0 Ack=1 Win=4380 Len=0 MSS=1460
39	3.977671	128.54.1.79	66.220.147.33	TCP	53792 > http [ACK] Seq=1 Ack=1 Win=64240 Len=0
40	3.981512	128.54.1.79	66.220.147.33	HTTP	GET / HTTP/1.1
42	4.177980	66.220.147.33	128.54.1.79	TCP	http > 53792 [ACK] Seq=1 Ack=558 Win=4937 Len=0
43	4.178344	66.220.147.33	128.54.1.79	TCP	[TCP segment of a reassembled PDU]
45	4.390254	128.54.1.79	66.220.147.33	TCP	53792 > http [ACK] Seq=558 Ack=1461 Win=64240 Len=0
46	4.407936	66.220.147.33	128.54.1.79	TCP	[TCP segment of a reassembled PDU]
47	4.417271	128.54.1.79	132.239.0.252	DNS	Standard query A static.ak.fbcdn.net
48	4.419163	132.239.0.252	128.54.1.79	DNS	Standard query response CNAME static.ak.facebook.com.edgesuit
49	4.419929	128.54.1.79	132.239.0.252	DNS	Standard query AAAA static.ak.fbcdn.net
50	4.421474	132.239.0.252	128.54.1.79	DNS	Standard query response CNAME static.ak.facebook.com.edgesuit

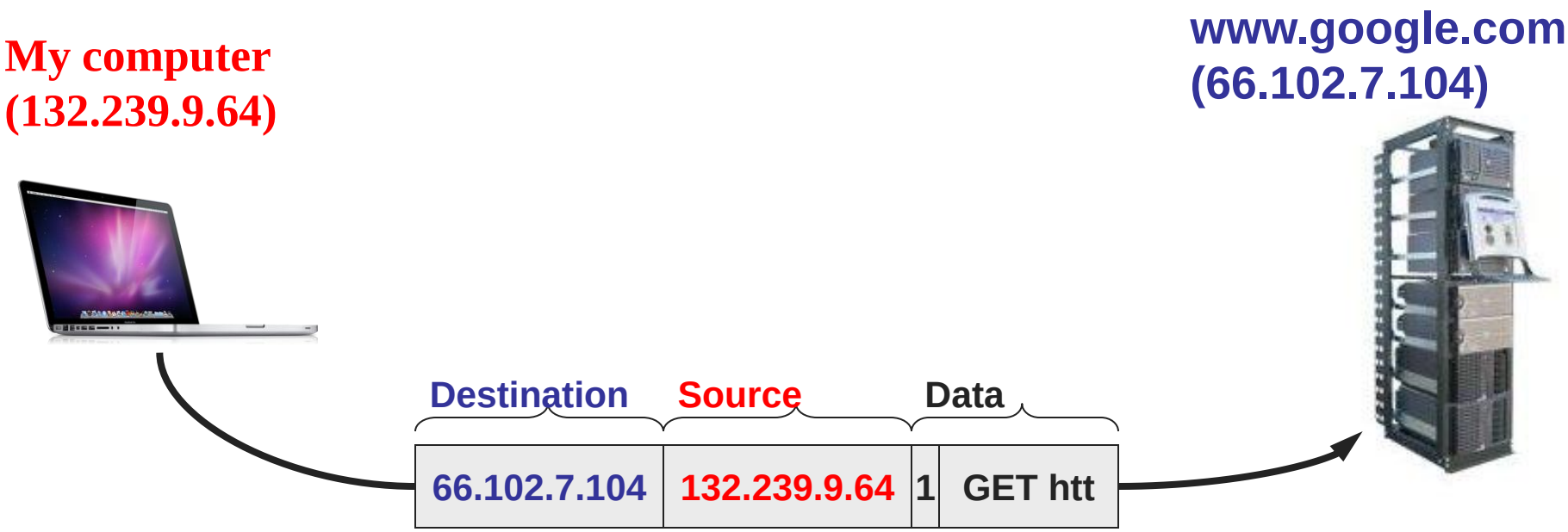
```
⊞ Frame 43 (1514 bytes on wire, 1514 bytes captured)
⊞ Ethernet II, Src: Cisco_4a:70:00 (00:21:56:4a:70:00), Dst: IntelCor_ae:1b:2b (00:1f:3b:ae:1b:2b)
⊞ Internet Protocol, Src: 66.220.147.33 (66.220.147.33), Dst: 128.54.1.79 (128.54.1.79)
⊞ Transmission Control Protocol, Src Port: http (80), Dst Port: 53792 (53792), Seq: 1, Ack: 558, Len: 1460
```

```
0000  00 1f 3b ae 1b 2b 00 21  56 4a 70 00 08 00 45 00   ..;..+.! VJp...E.
0010  05 dc 3a 1d 40 00 f3 06  f0 7b 42 dc 93 21 80 36   ..:.@... .{B..!.6
0020  01 4f 00 50 d2 20 a8 05  b6 f8 7c 64 82 09 50 18   .O.P. .. ..|d..P.
0030  13 49 13 3b 00 00 48 54  54 50 2f 31 2e 31 20 32   .I.;..HT TP/1.1 2
0040  30 30 20 4f 4b 0d 0a 43  61 63 68 65 2d 43 6f 6e   00 OK..C ache-Con
0050  74 72 6f 6c 3a 20 70 72  69 76 61 74 65 2c 20 6e   trol: pr ivate, n
0060  6f 2d 63 61 63 68 65 2c  20 6e 6f 2d 73 74 6f 72   o-cache,  no-stor
0070  65 2c 20 6d 75 73 74 2d  72 65 76 61 6c 69 64 61   e, must- revalida
0080  74 65 0d 0a 45 78 70 69  72 65 73 3a 20 53 61 74   te..Expi res: Sat
0090  2c 20 30 31 20 4a 61 6e  20 32 30 30 30 20 30 30     01 Jan  2000 00
```

File: "C:\Users\ckanich\Documents\123.test" 38... Packets: 702 Displayed: 629 Marked: 0 Profile: Default

Global Network Addressing

- Address each packet so it can traverse network and arrive at host



Project Overview

Project Objective

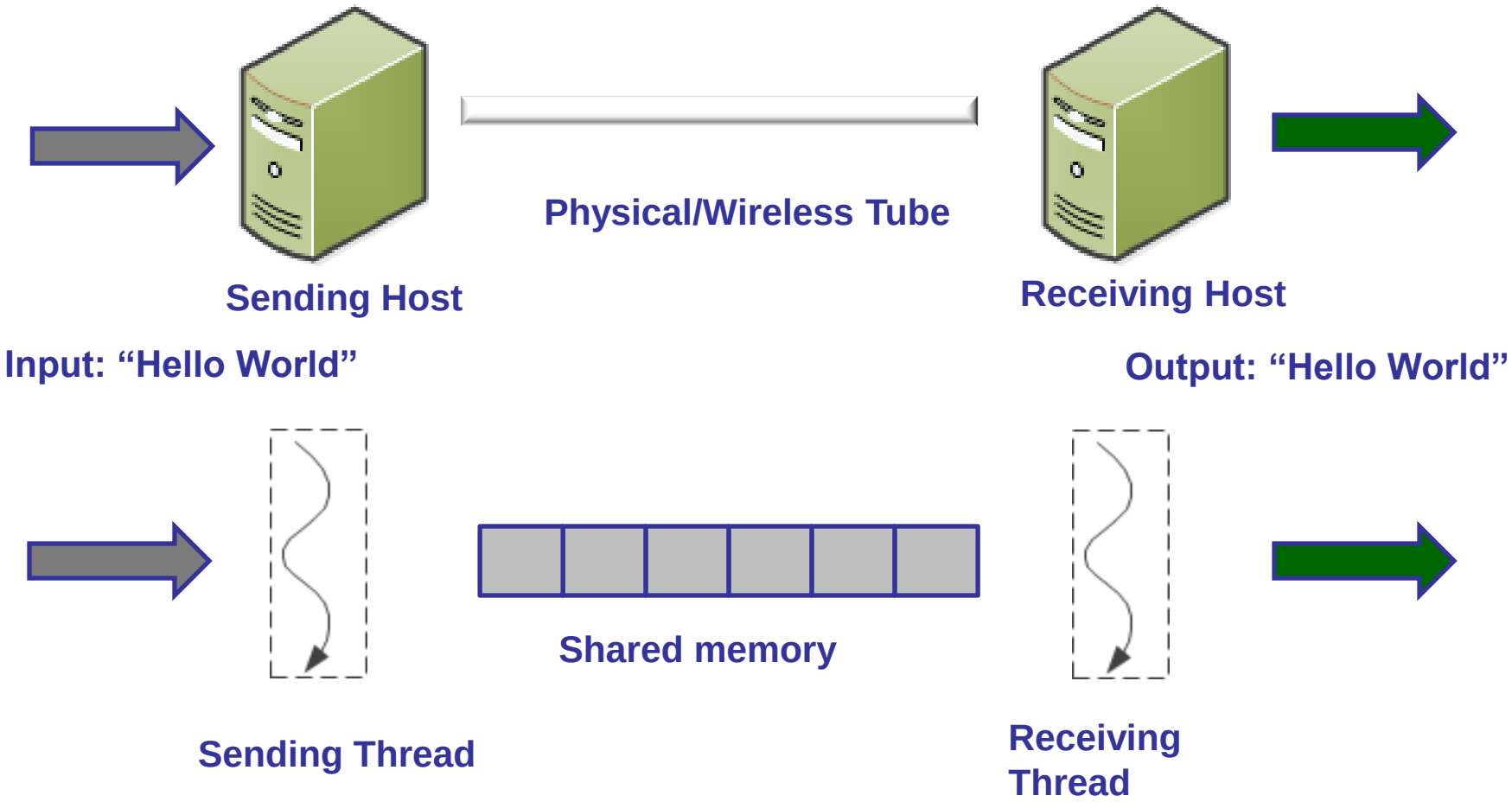
- Goal: Facilitate communication between sender and receiver threads over a lossy, unreliable communication channel.

- Your task: Implement a version of the Sliding Window Protocol (SWP) for communication between threads
  - See Sections 2.4 and 2.5 of P&D

Recapping Threads

- Wikipedia definition: The smallest unit of processing that can be scheduled by the OS
- Multiple threads can run inside the same process and share memory
  - Protect critical sections using mutexes/condition variables/etc
- FYI, how to create a thread in C: pthread_create
  - Arguments passed in:
    - » pthread data type
    - » IGNORE
    - » function name
    - » function input parameter

Project Analogy



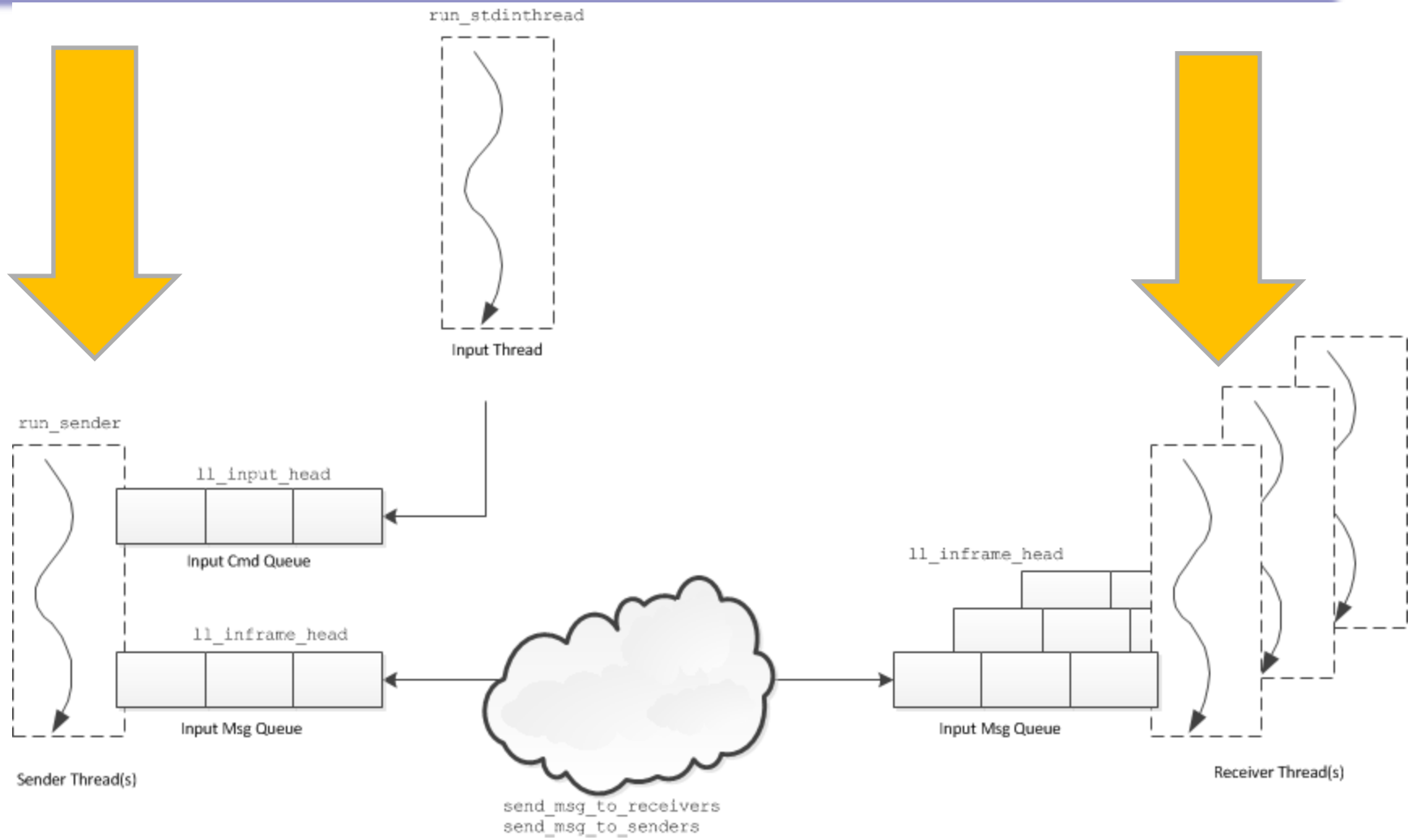
Project Analogy (cont)

- Sender thread: A node attempting to transmit data over a link
- Receiver thread: A node waiting to receive data over the link
- What problems do nodes face when transmitting data?
  - Packet losses
  - Data corruption
  - Collisions

Project Overview

- Sender threads: Take input from command line and transmit to receivers
  - msg <sender_id> <receiver_id> <stdin message>
  - `msg 0 1 hello world!`
- Receiver threads: Print out messages intended for them
  - `<RECV_1>[hello world!]`

Project Overview (diagram)

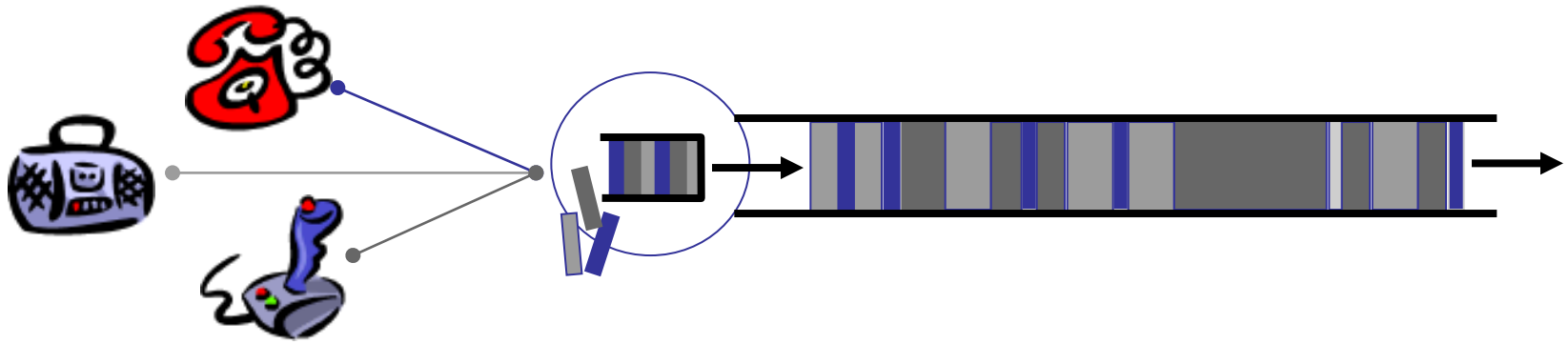


Project Overview (cont)

- You will have to handle:
  - Message drops
  - Corrupted messages
  - Message buffering/segmenting
  - Multiple outstanding messages

- To tackle this project:
  - Read P&D Section 2.4/2.5

Resource Allocation: Queues



- Sharing access to limited resources
  - E.g., a link with fixed service rate
- Simplest case: first-in-first out queue
  - Queue/serve packets in the order they arrive
  - Drop packets when the queue is full
- Anybody hear of “Network Neutrality”?