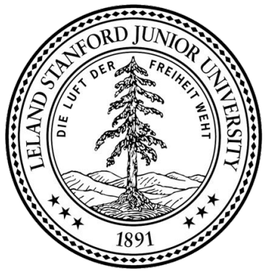
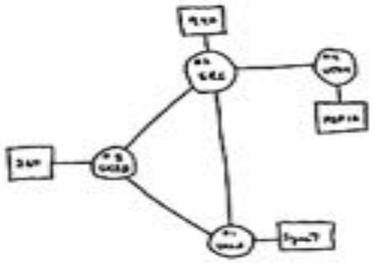
How the Internet Works

(in about an hour)



Nick McKeown
Professor of Electrical Engineering
and Computer Science, Stanford University

Four nodes connected
(UCLA, SRI, UCSB, Utah)



"*A network to survive nuclear attack.*"

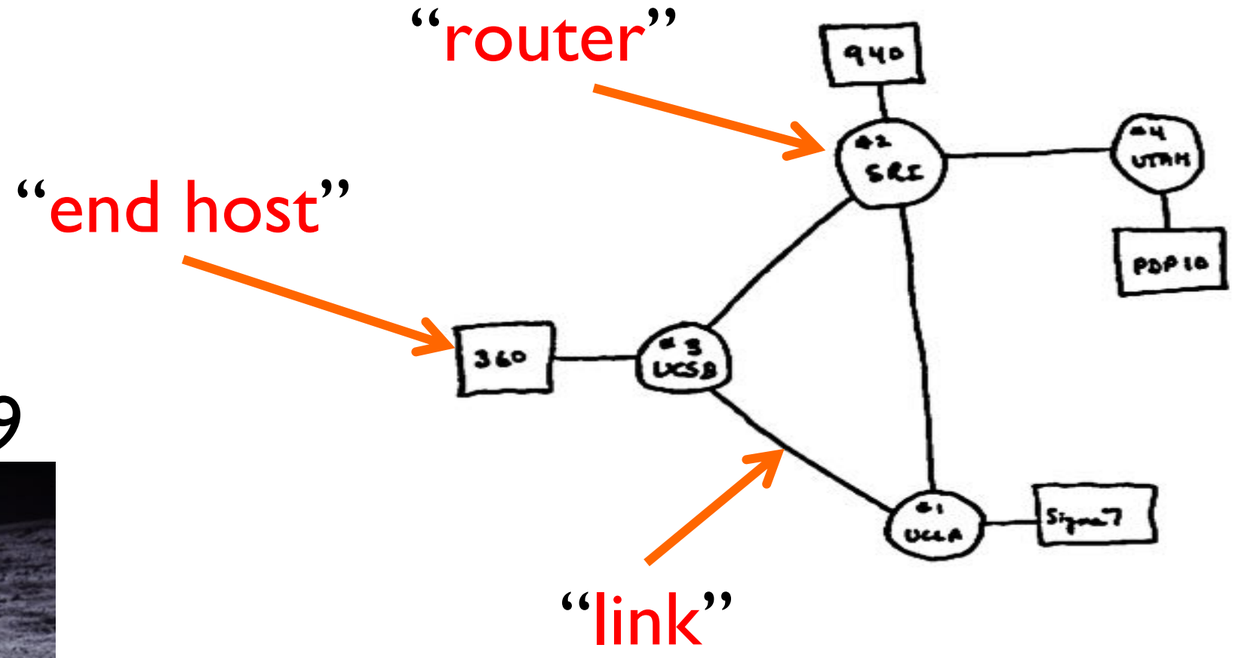
Paul Baran

US Government starts "ARPANET" project

1st network connects two computers

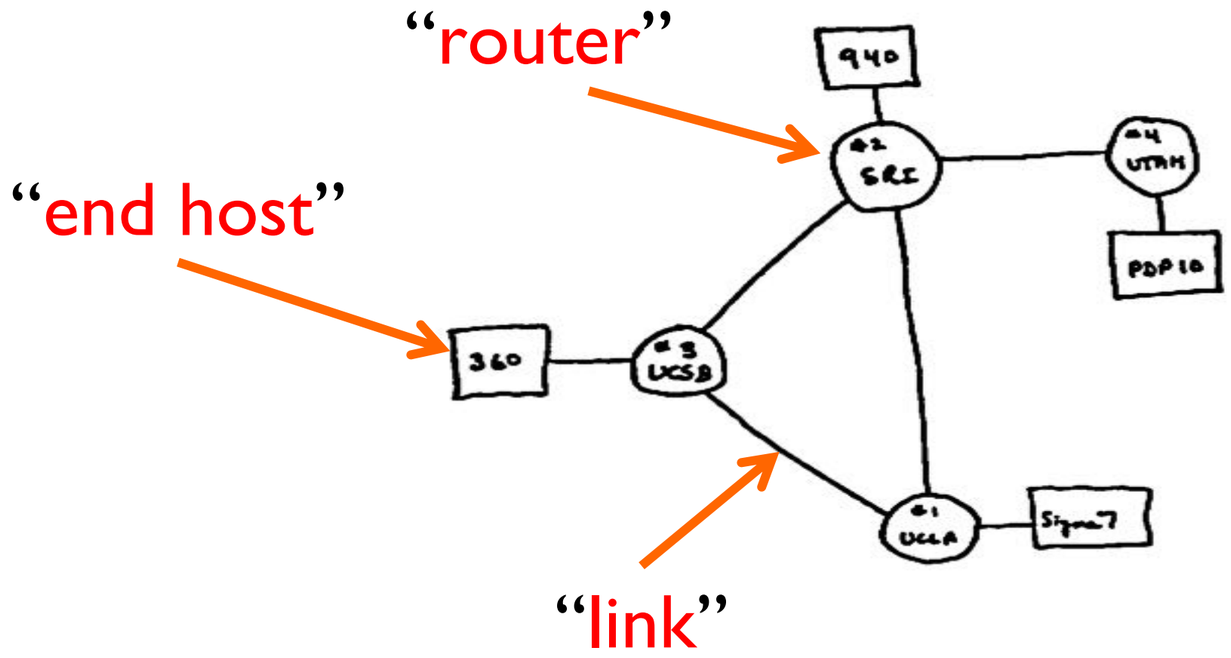
1964 1965 1966 1968 1969

The Internet in 1969



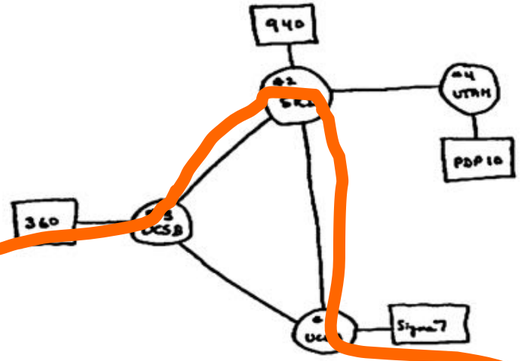
Also in 1969

What did they use it for?



1. Sending files between scientists: "*Here is a big file of astronomy data!*"
2. Email: "*Where shall we have lunch today?*"
3. Remote login to another computer.

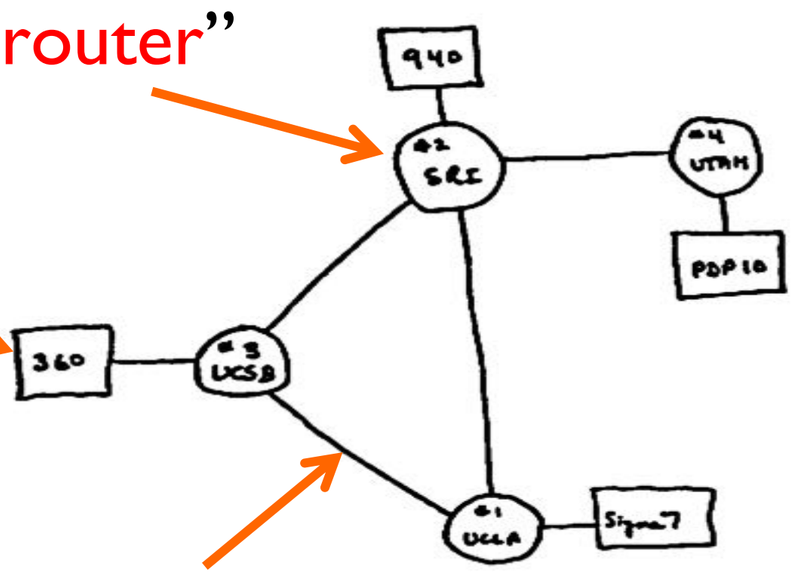
1971



First email typed here

"QWERTYUIOP"

…and printed here



"router"

"end host"

"link"

1969

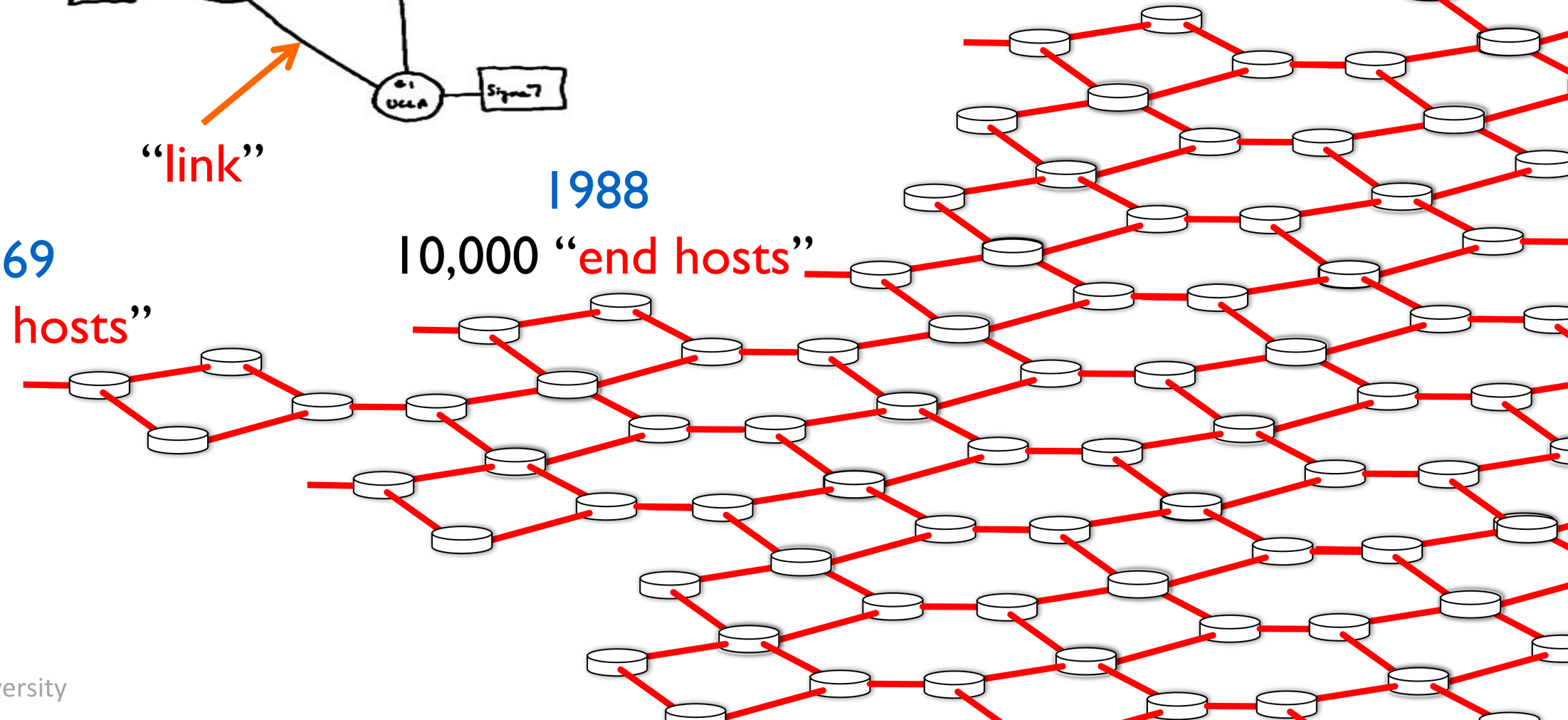
4 "end hosts"

1988

10,000 "end hosts"

1993

1,000,000 "end hosts"



Then in 1993 something
even BIGGER happened!!!

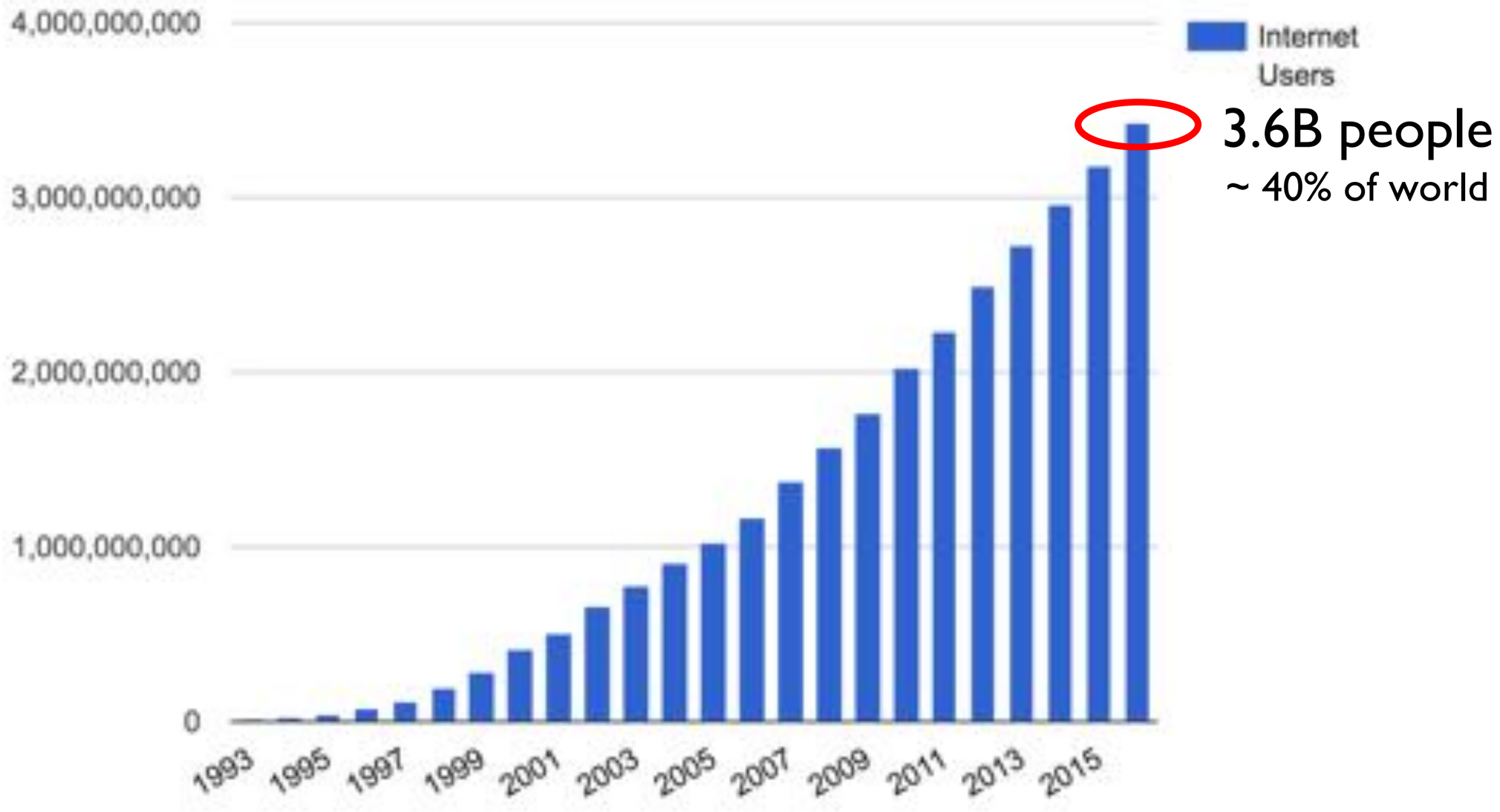
1993: The first web browser (Mosaic)

Marc Andreessen

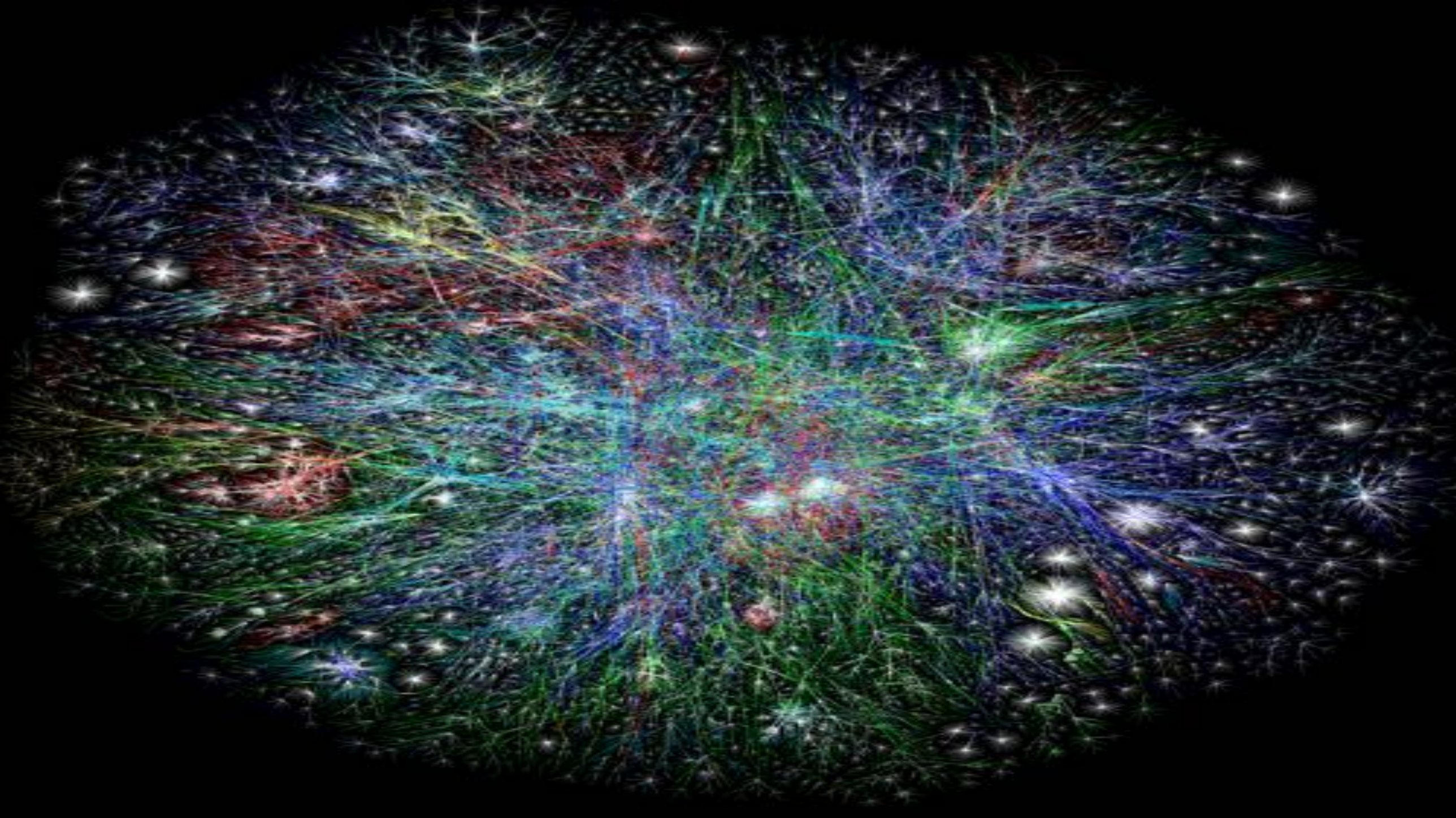
1993

President
Bill Clinton

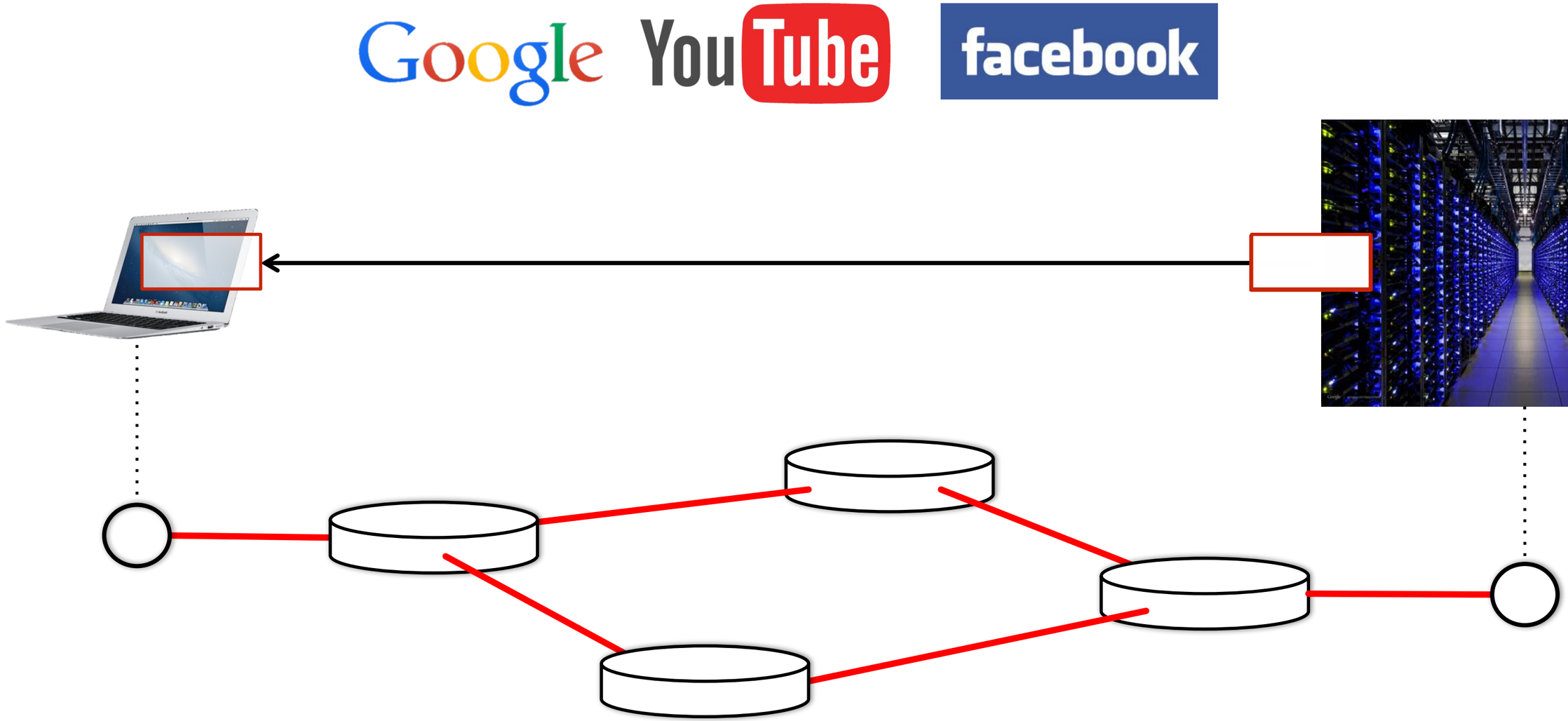
The number of Internet users in the world



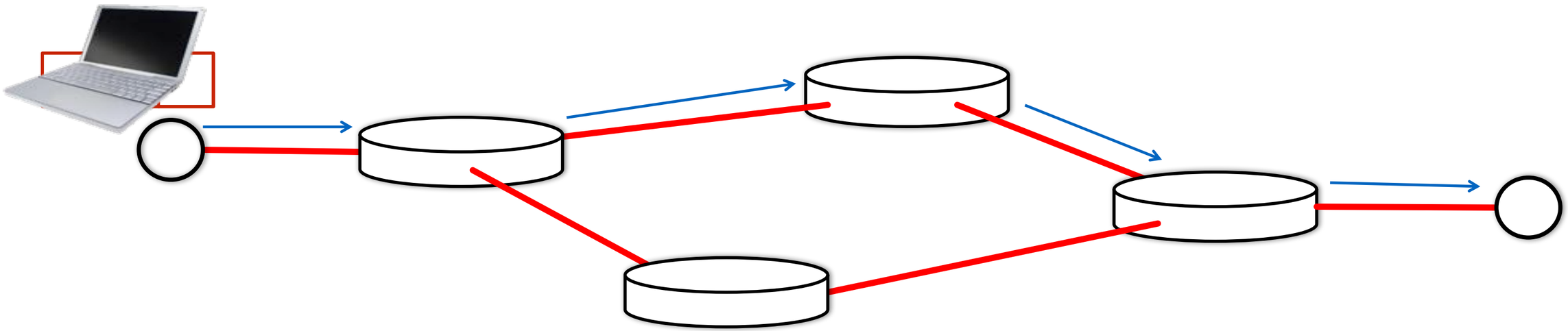
Source: http://www.internetlivestats.com/

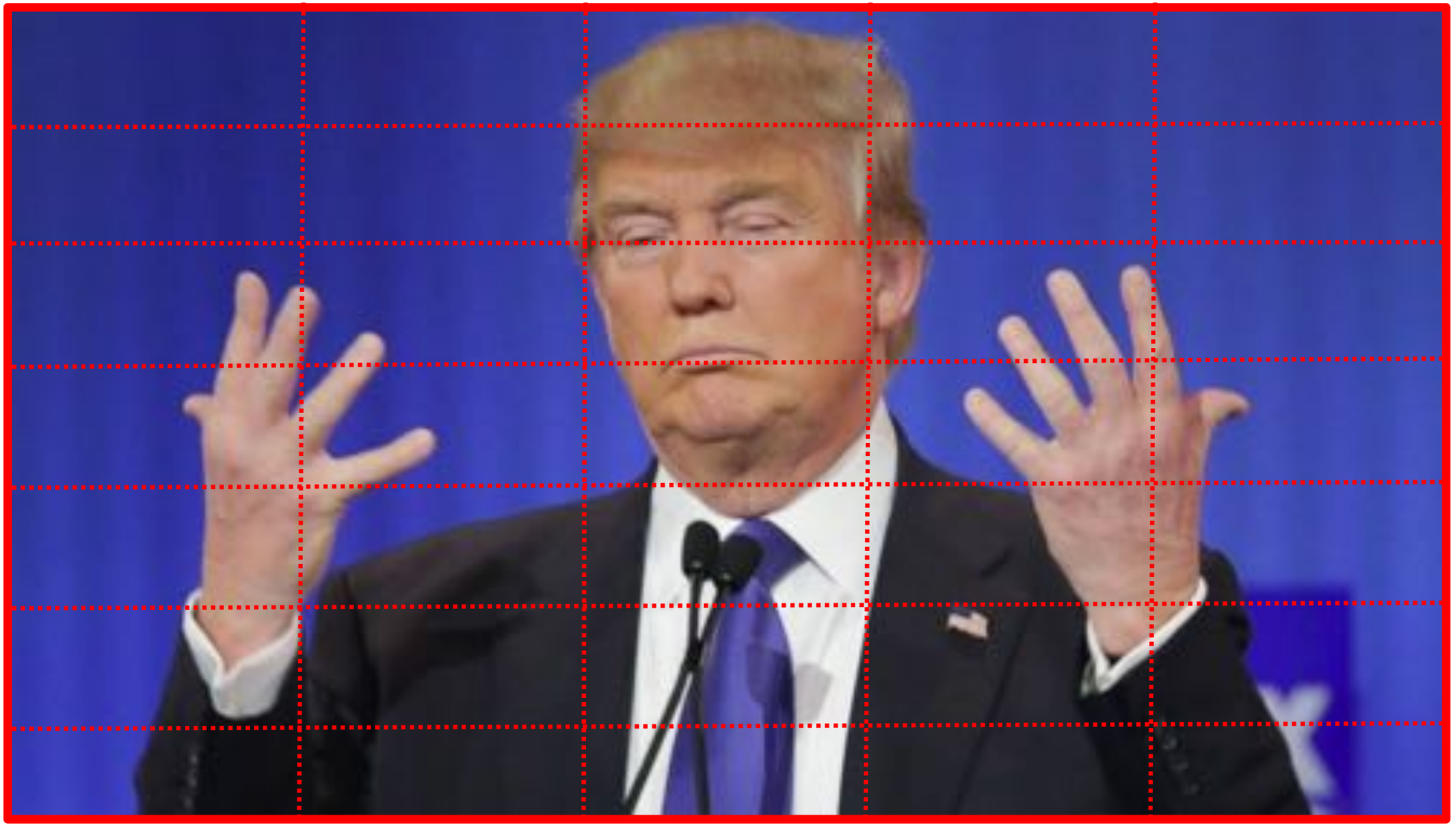


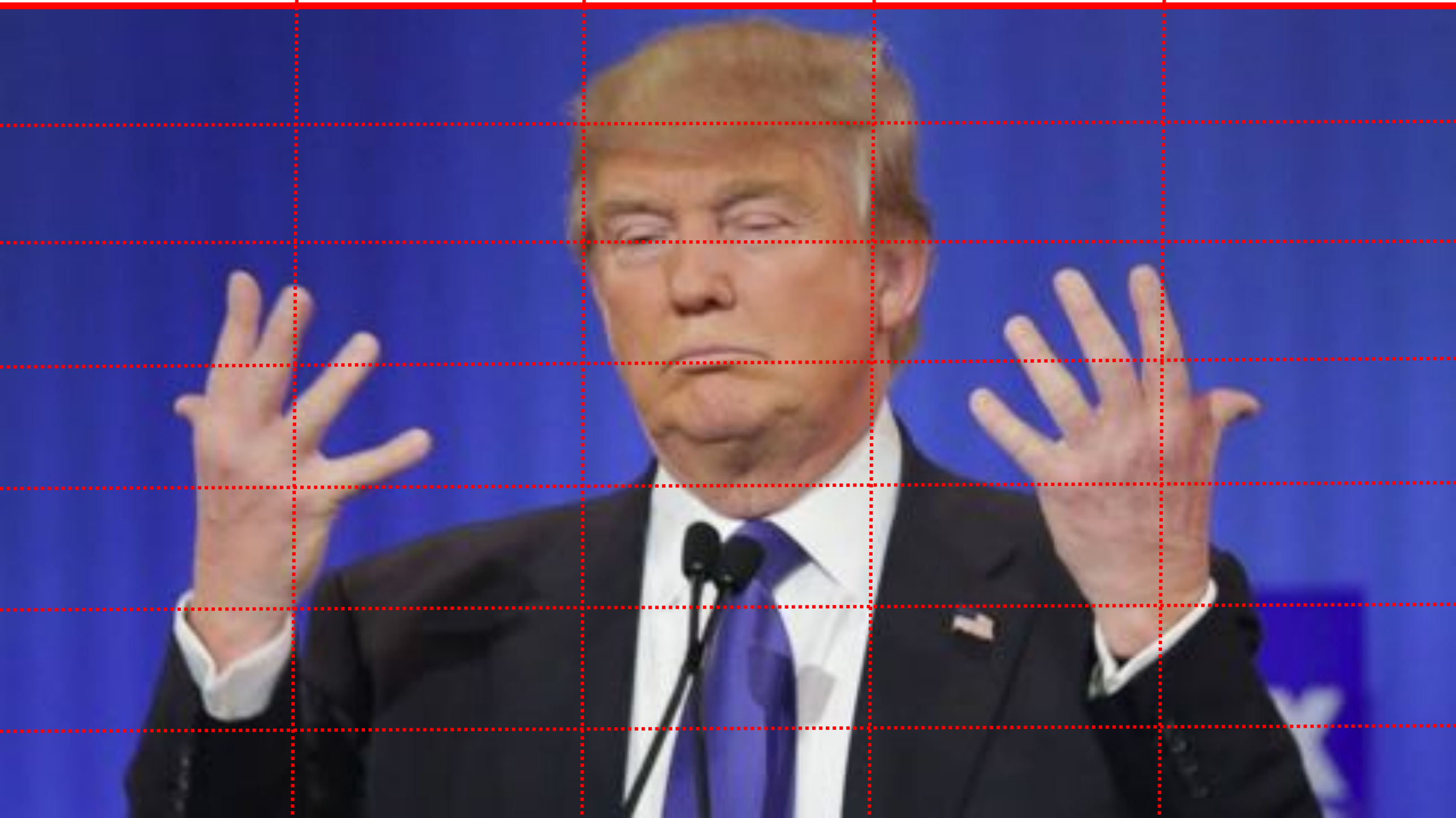
How does it all work?


Google
YouTube
facebook


My Program
Someone else's Program

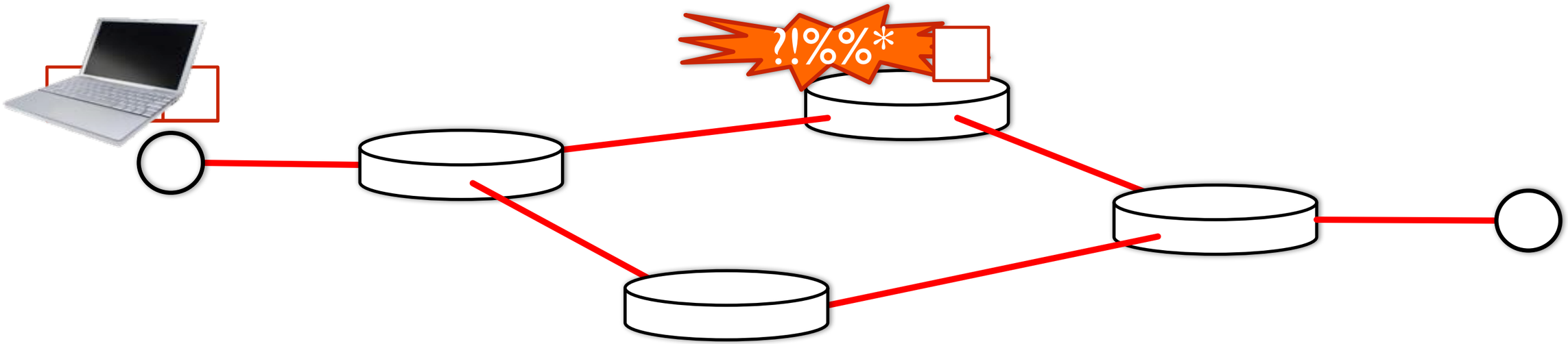




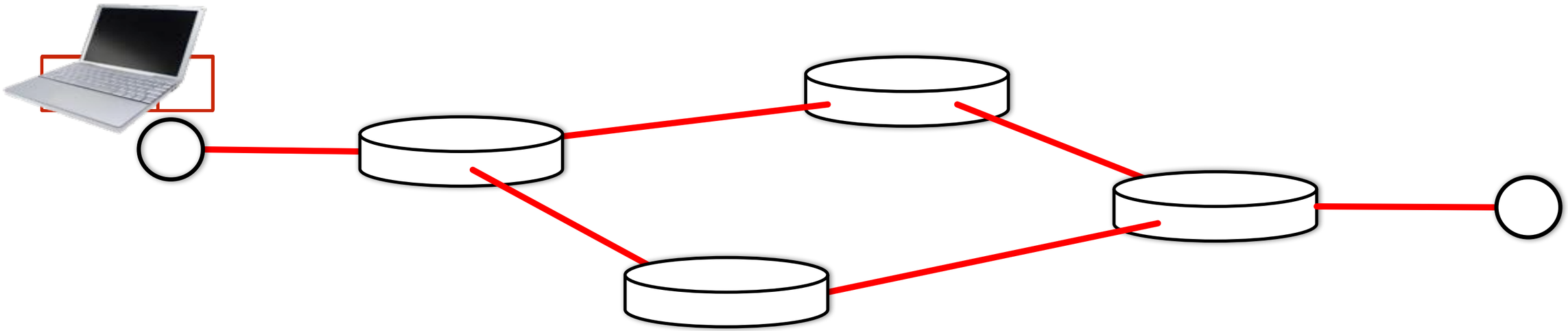



1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35

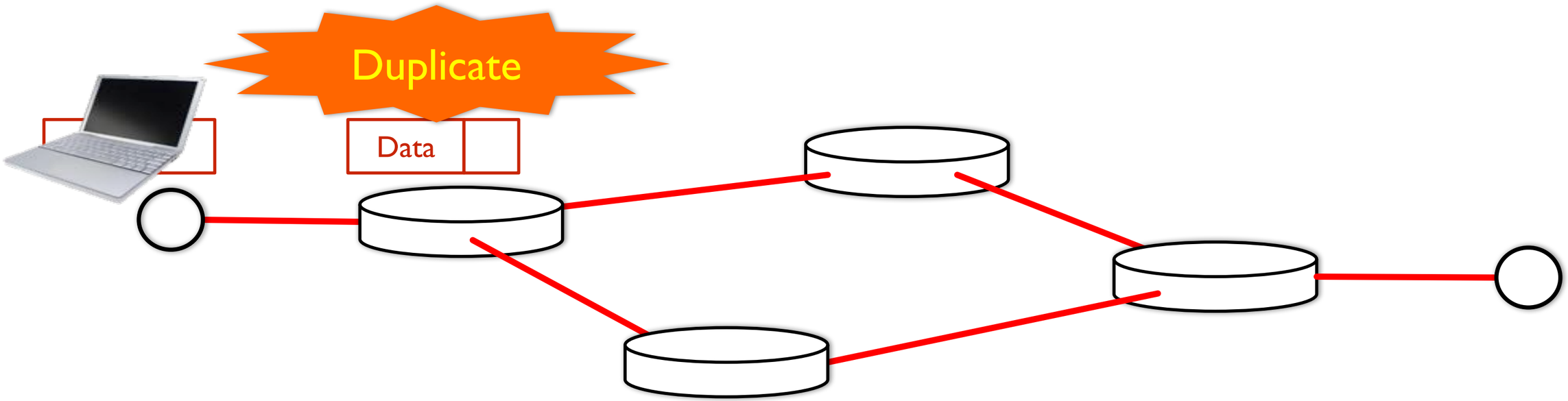
Packets may be damaged



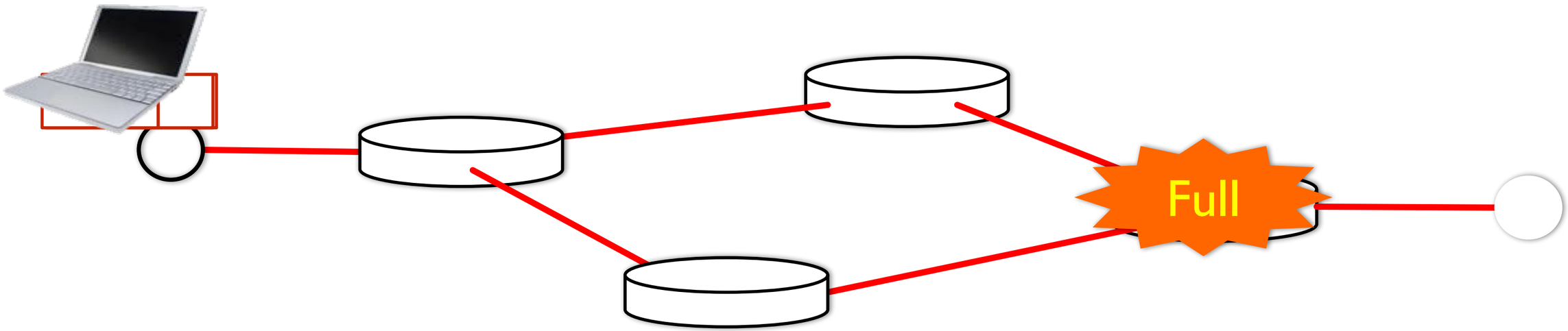
Packets may arrive out of order



Packets may be duplicated

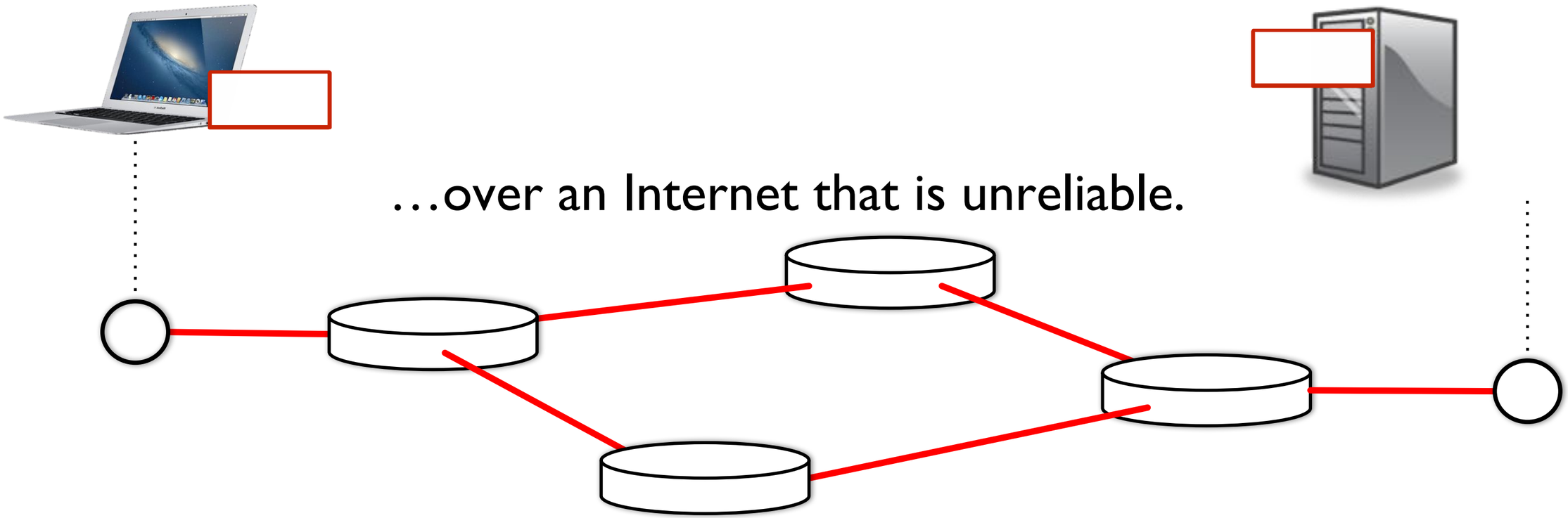


They may not arrive at all!



Summary so far

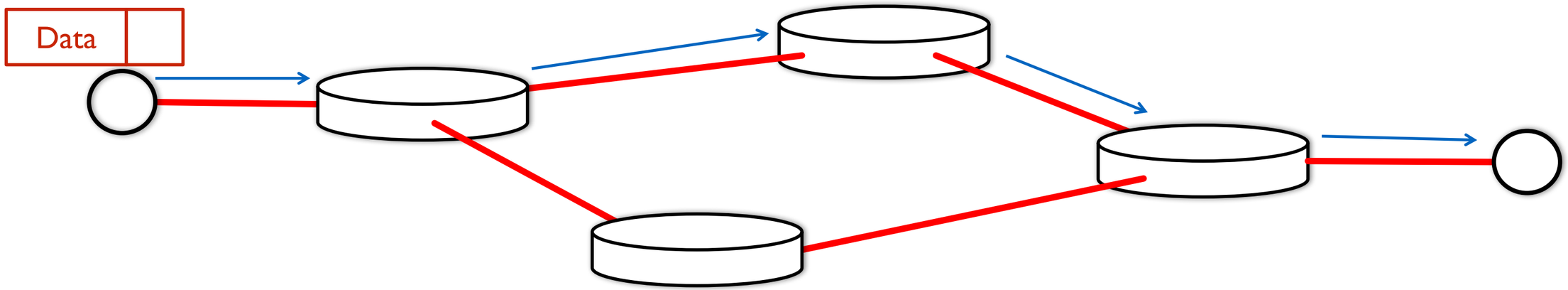
Applications send and receive data in packets….



…over an Internet that is unreliable.

What do Internet packets look like?

Internet addresses



Internet Addresses ("IP address")

All Internet packets carry a destination IP address.

We usually write the IP address like this:

171.64.74.58

Internet “IP” Addresses

The IP address tells each router where to send the packet next.

A network in the CS Department at Stanford University

171.64.74.58

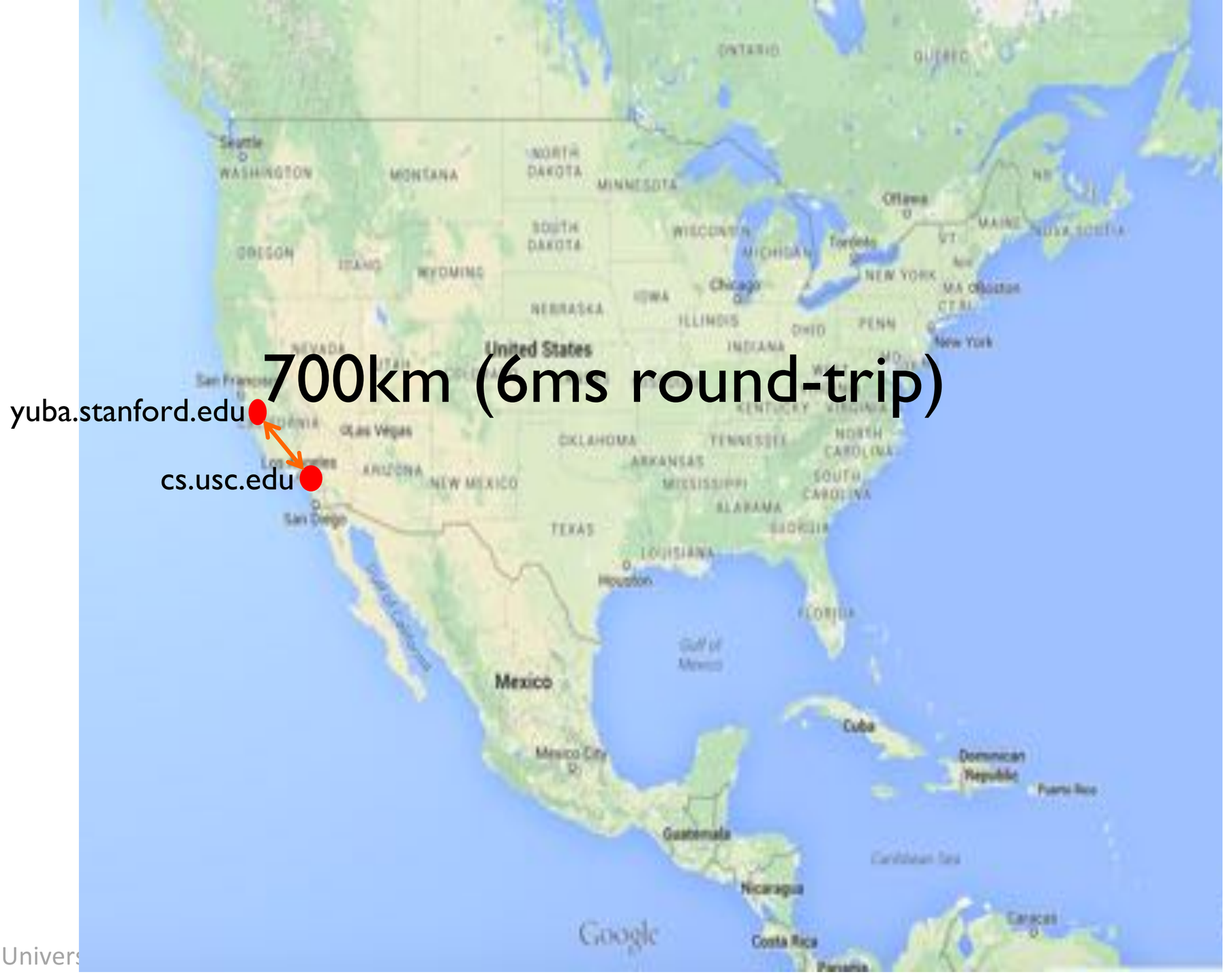
The computer yuba.stanford.edu

Can we see the path our packets take?

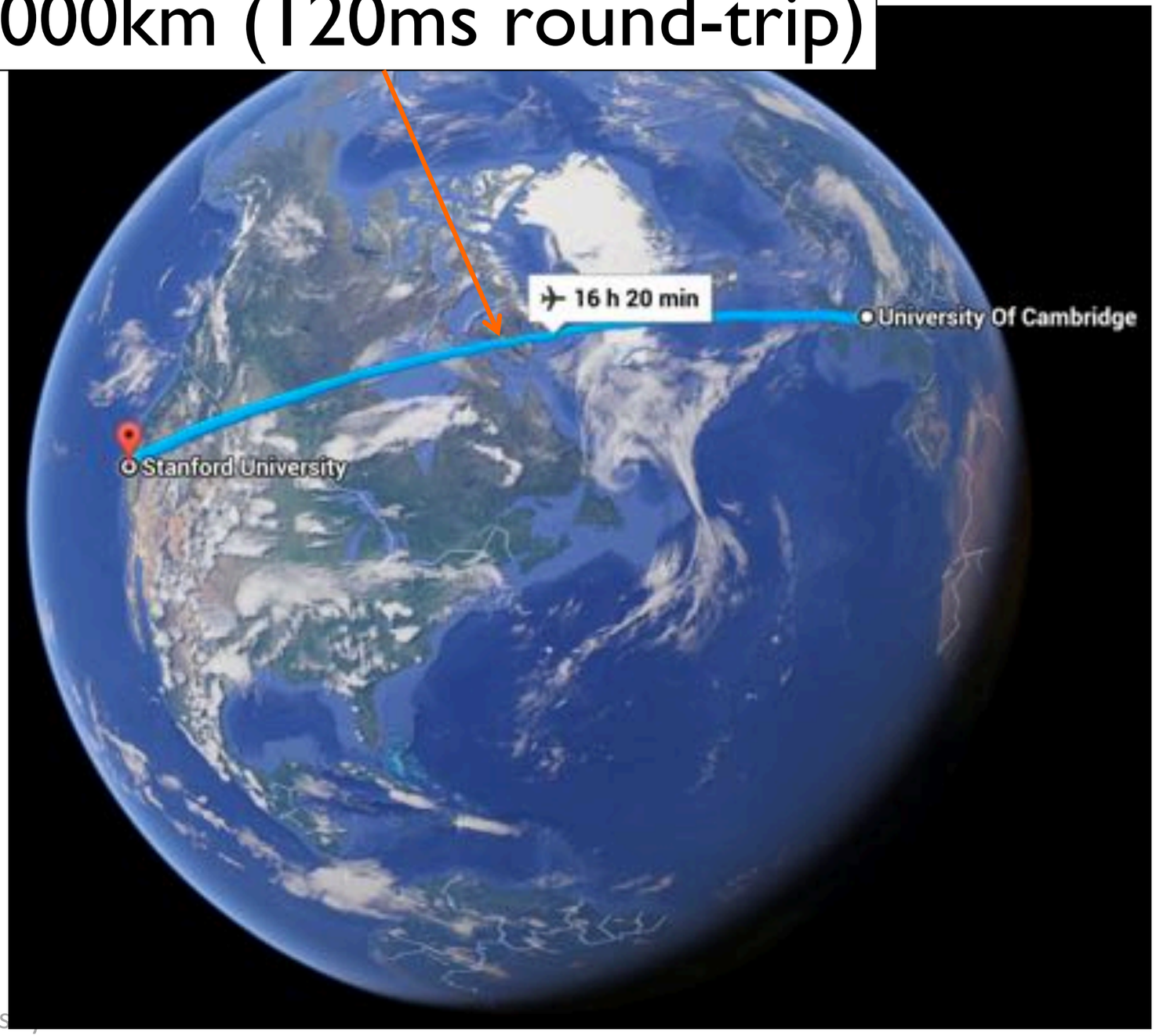
Yes!

On your computer, try:

```
traceroute -q1 yuba.stanford.edu
```


700km (6ms round-trip)
yuba.stanford.edu
cs.usc.edu
Seattle
WASHINGTON
MONTANA
NORTH DAKOTA
MINNESOTA
ONTARIO
QUEBEC
Ottawa
OREGON
IDAHO
WYOMING
SOUTH DAKOTA
WISCONSIN
MICHIGAN
Toronto
MAINE
NOVA SCOTIA
NEW YORK
Chicago
IOWA
NEBRASKA
ILLINOIS
OHIO
PENN
New York
NEVADA
United States
INDIANA
San Francisco
Las Vegas
OKLAHOMA
TENNESSEE
NORTH CAROLINA
ARIZONA
NEW MEXICO
ARKANSAS
MISSISSIPPI
SOUTH CAROLINA
ALABAMA
San Diego
TEXAS
LOUISIANA
Houston
FLORIDA
Gulf of Mexico
Mexico
Cuba
Mexico City
Dominican Republic
Puerto Rico
Guatemala
Caribbean Sea
Nicaragua
Caracas
Google
Costa Rica
Panama

15,000km (120ms round-trip)



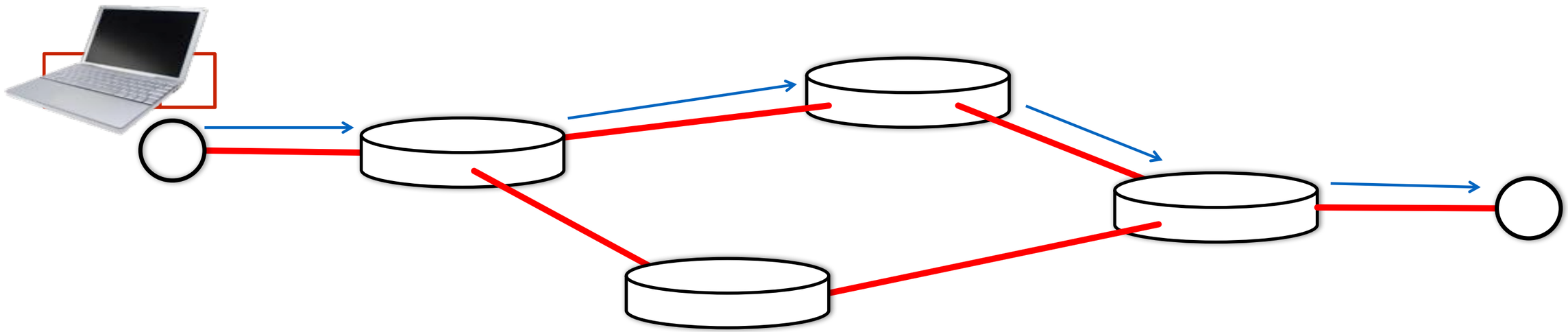
Try traceroute to….

yuba.stanford.edu, www.google.com, www.ntua.gr, …

How packets find their way across the Internet

Routers forward packets **one at a time**.

Routers look at IP addresses, then send packets to a router closer to the destination.



IP Addresses

The IP address tells a router where to send the packet next.

IP addresses have *structure*

A network in the CS department at Stanford University

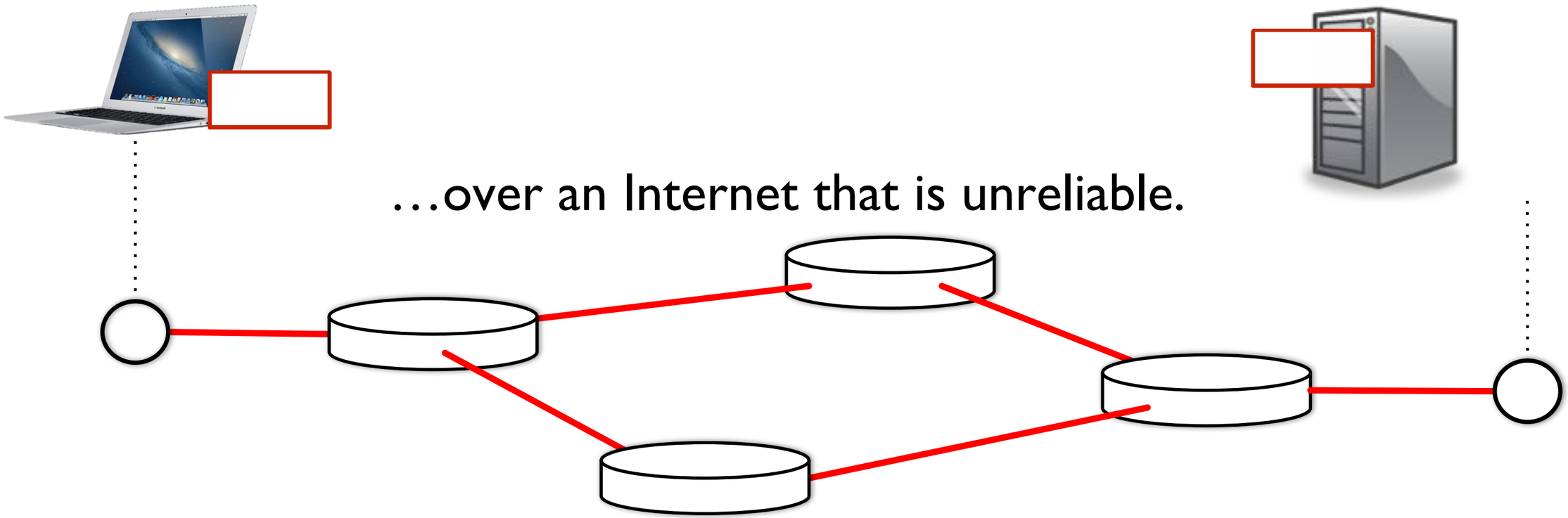
171.64.74.58

An address managed by RIPE (European IP Networks)

88.255.96.208

Summary so far

Applications send and receive data in packets….



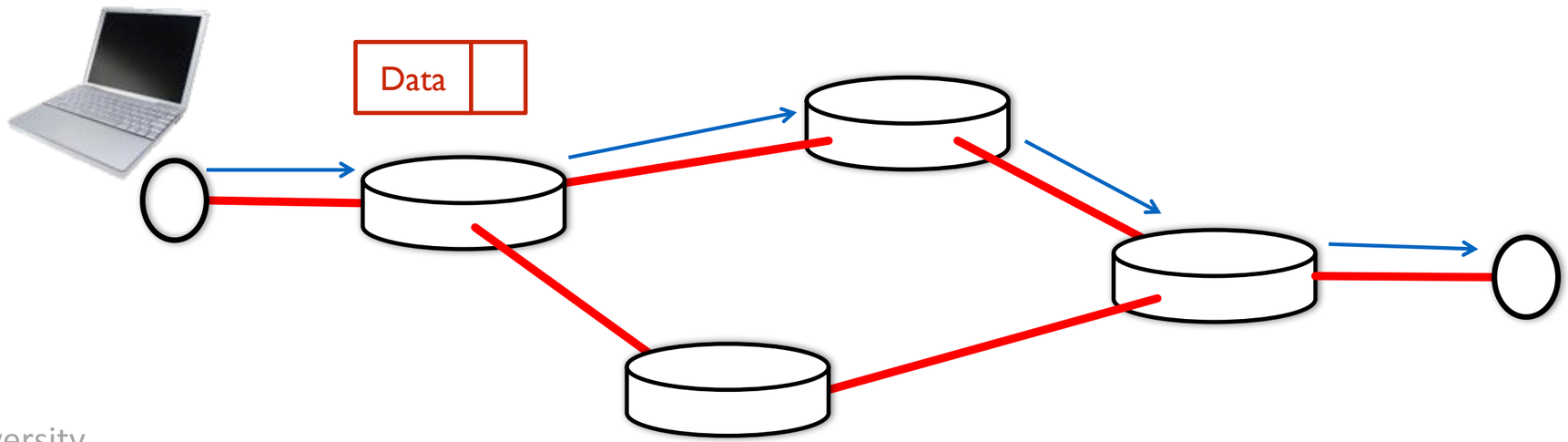
…over an Internet that is unreliable.

Packets are forwarded hop-by-hop based on the final destination address.

The Internet cannot be trusted!!

The Internet doesn't promise to deliver packets in order.
It doesn't promise to deliver packets quickly, or on time.
It doesn't even promise to deliver them at all!

It just makes a "best-effort" attempt.



Sending data reliably over an Internet that is unreliable

How Network Applications Communicate

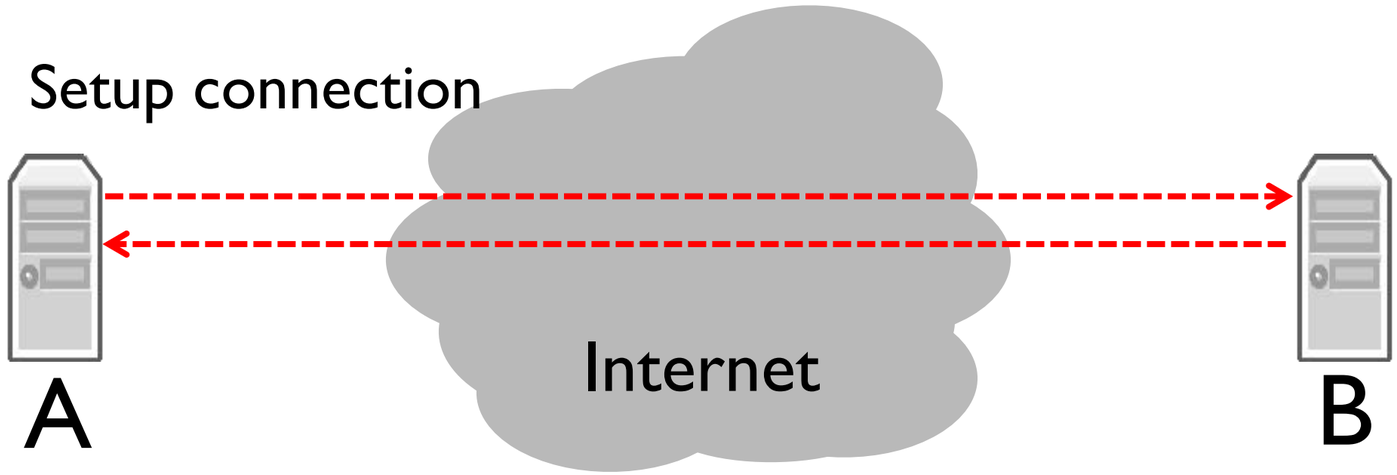
The most common method:

- Communication is in both directions – "bidirectional".
- Communication is reliable
  (if there is a working path between the two computers).

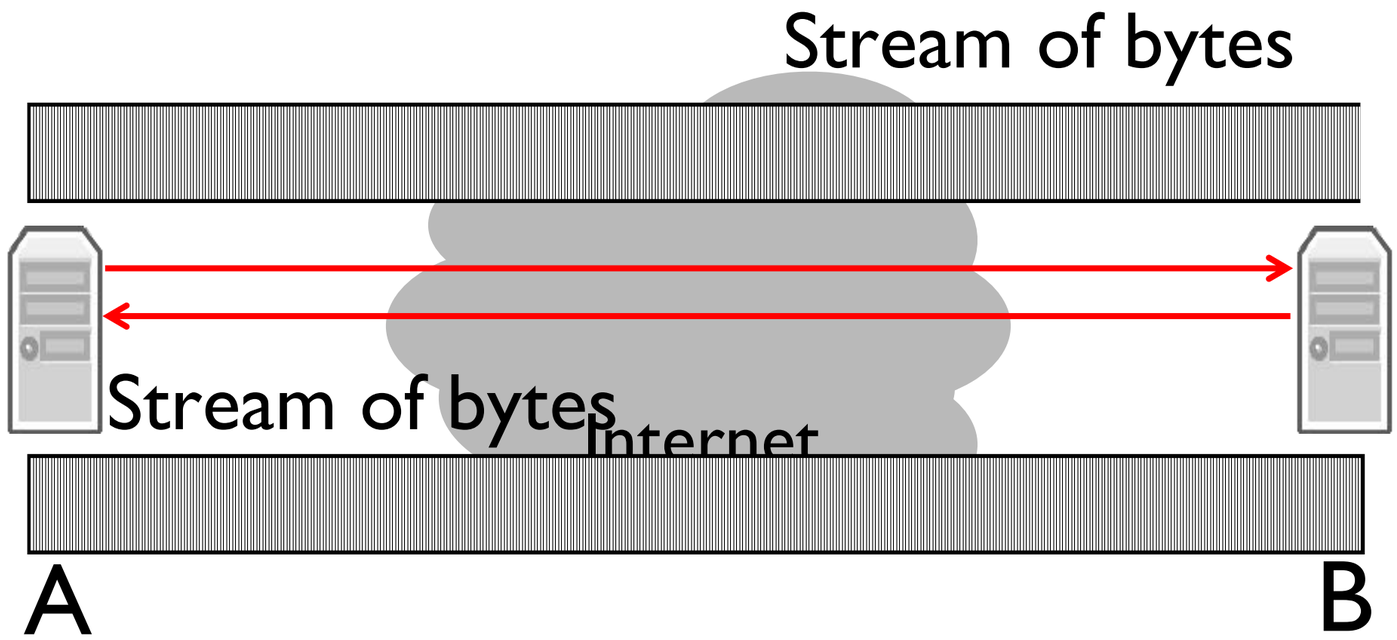
It's like an unformatted pipe:

- You push data in at one end, and it pops out correctly at the other end.
- The applications decide how the data is formatted inside the pipe.

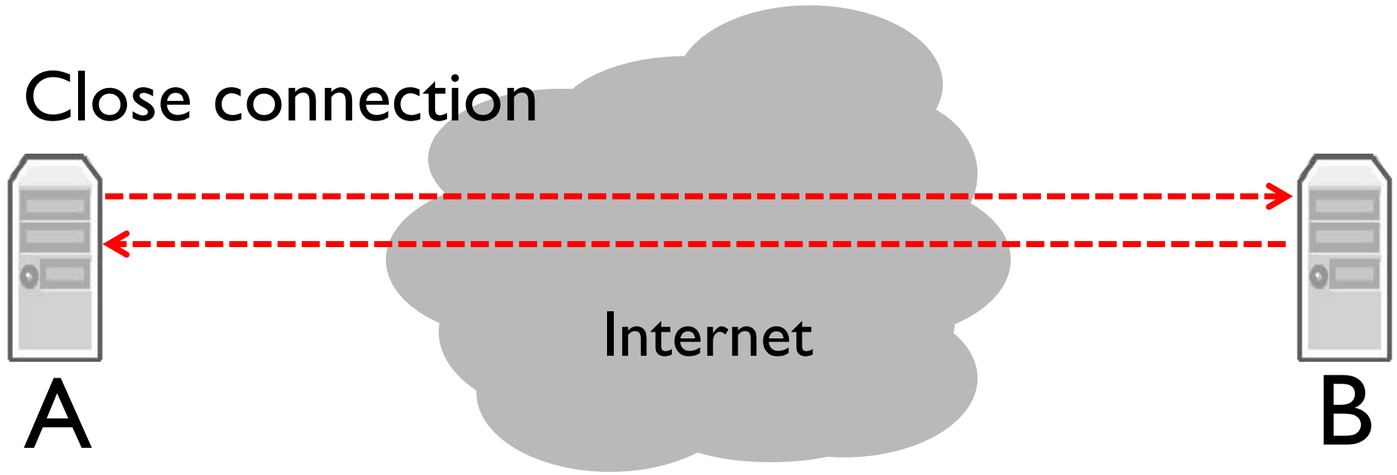
Byte Stream Model



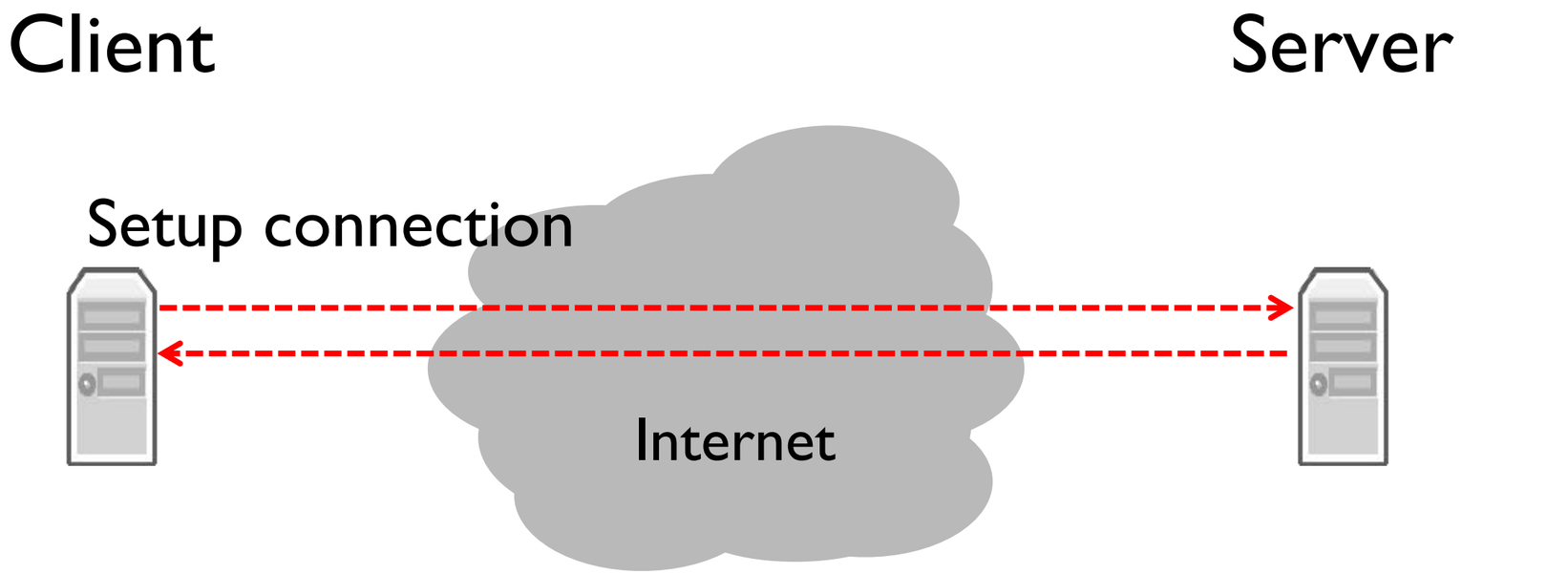
Byte Stream Model



Byte Stream Model

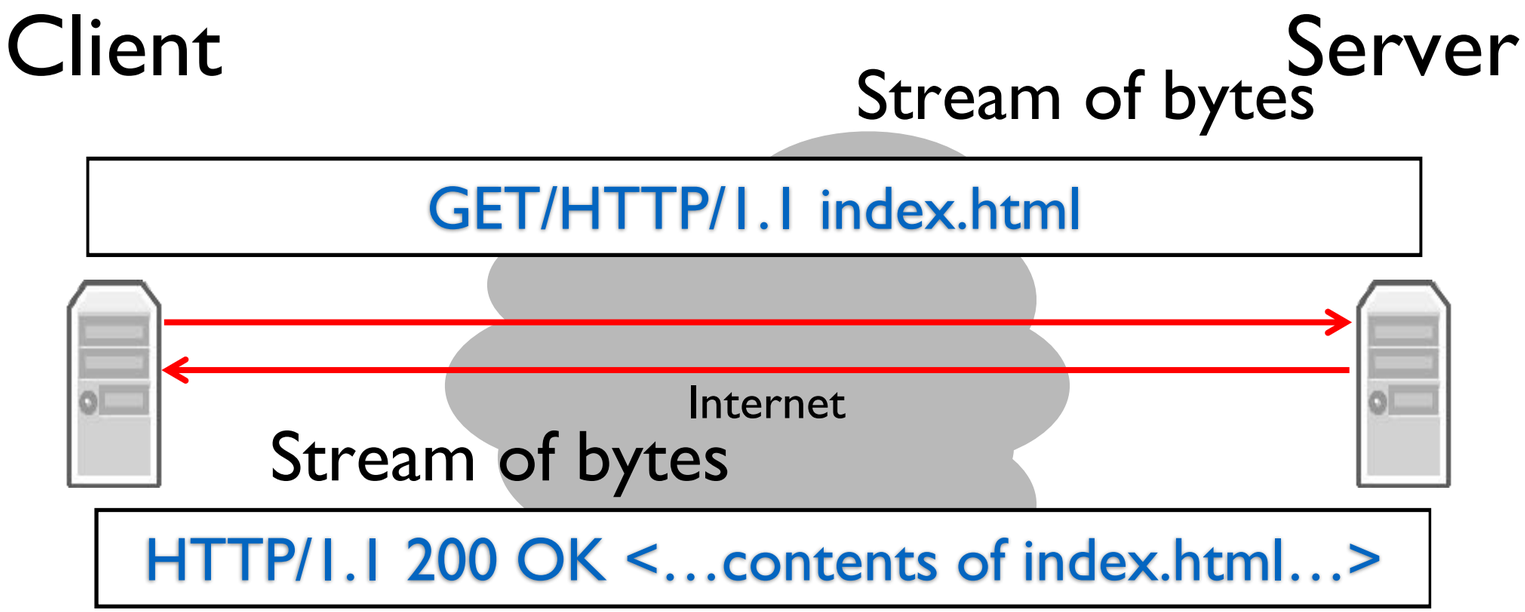


World Wide Web (HTTP)



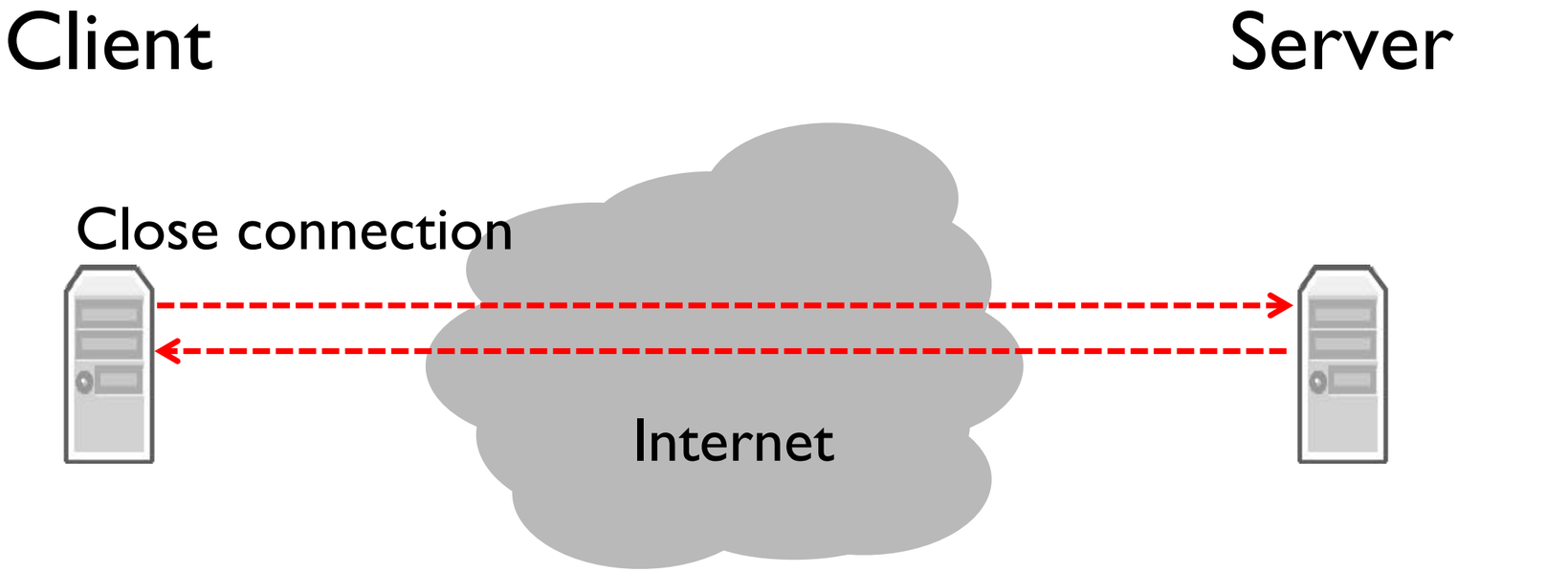
www.stanford.edu

World Wide Web (HTTP)

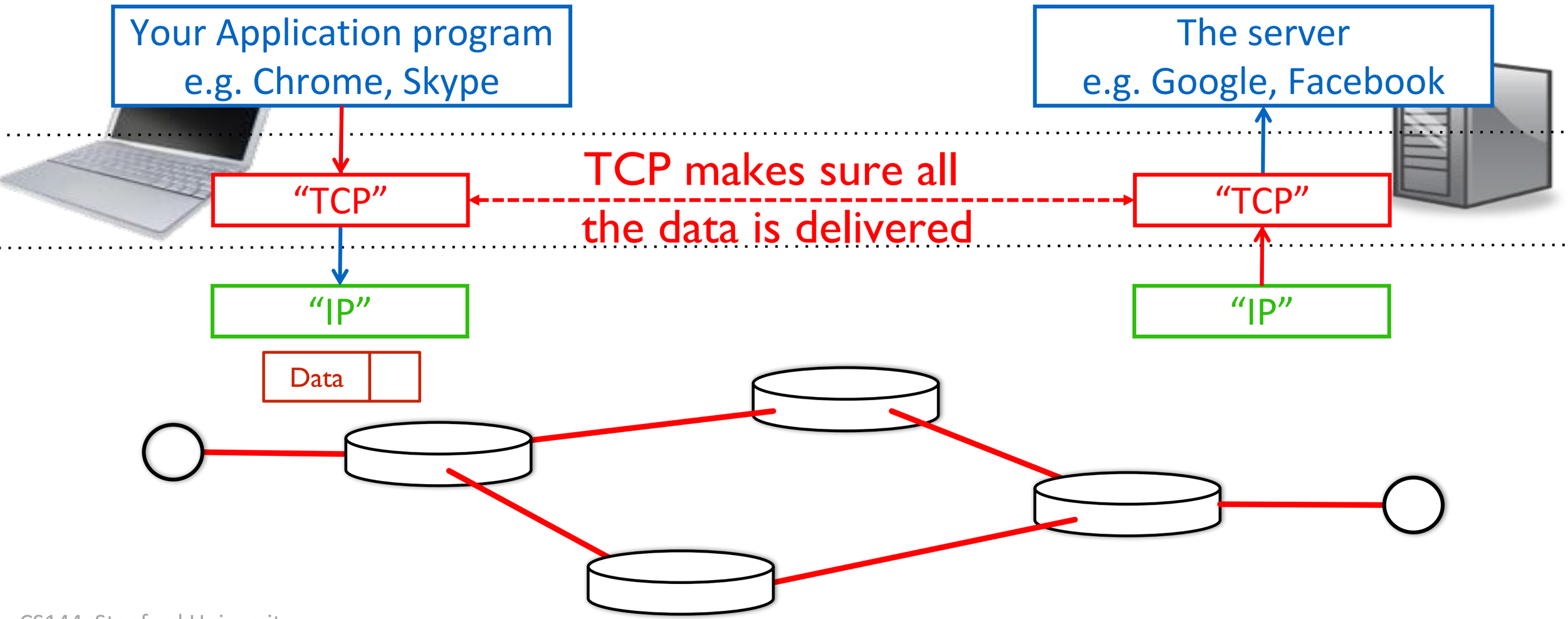


www.stanford.edu

World Wide Web (HTTP)



www.stanford.edu


Your Application program
e.g. Chrome, Skype
The server
e.g. Google, Facebook
"TCP"
TCP makes sure all
the data is delivered
"TCP"
"IP"
"IP"
Data

TCP's job

Makes sure all data is delivered correctly.

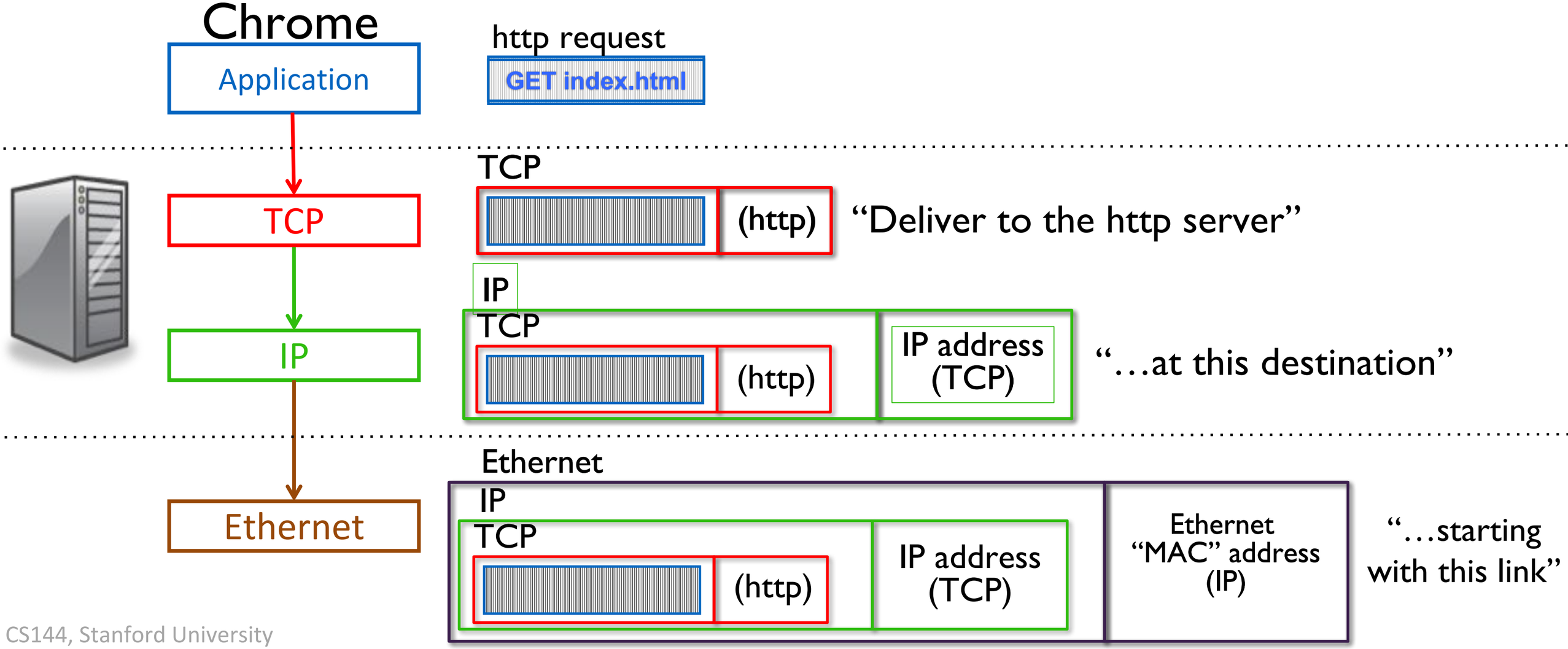
Delivers data to the application in the right order.

How?

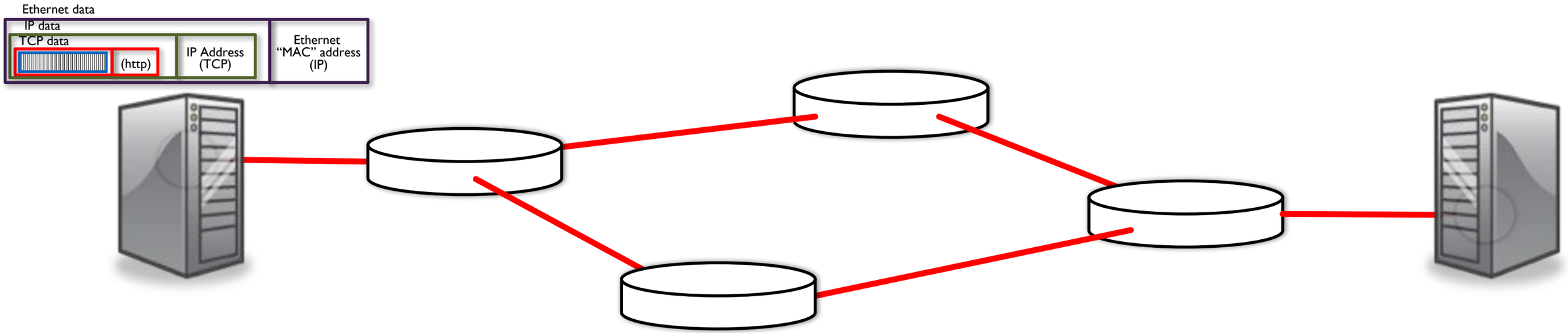
- Add sequence numbers to every packet (so the receiver can check if any are missing, and put them in right order)
- When a packet arrives, send an **acknowledgment of receipt** or "ACK" back to the sender
- If no acknowledgment is received, resend the data


1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35

http client (e.g. Chrome)



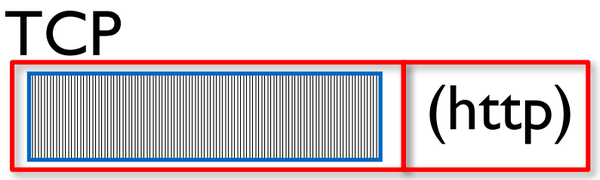
Here it goes….


Ethernet data
IP data
TCP data
(http)
IP Address
(TCP)
Ethernet
“MAC” address
(IP)

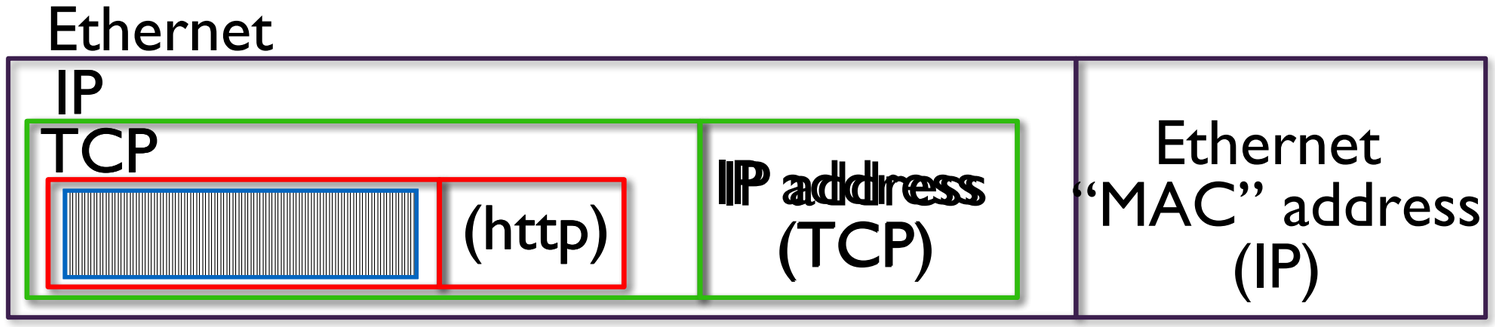
http server (e.g. www.google.com)

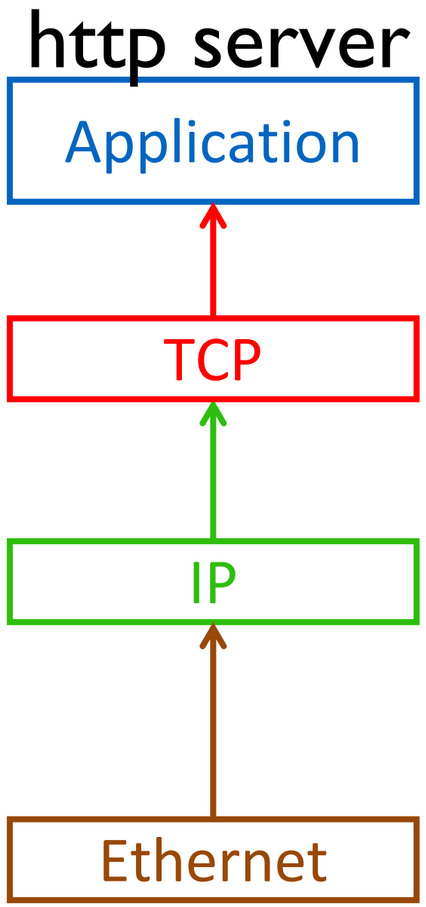
http request

"Deliver to the http server"



"Deliver to TCP"



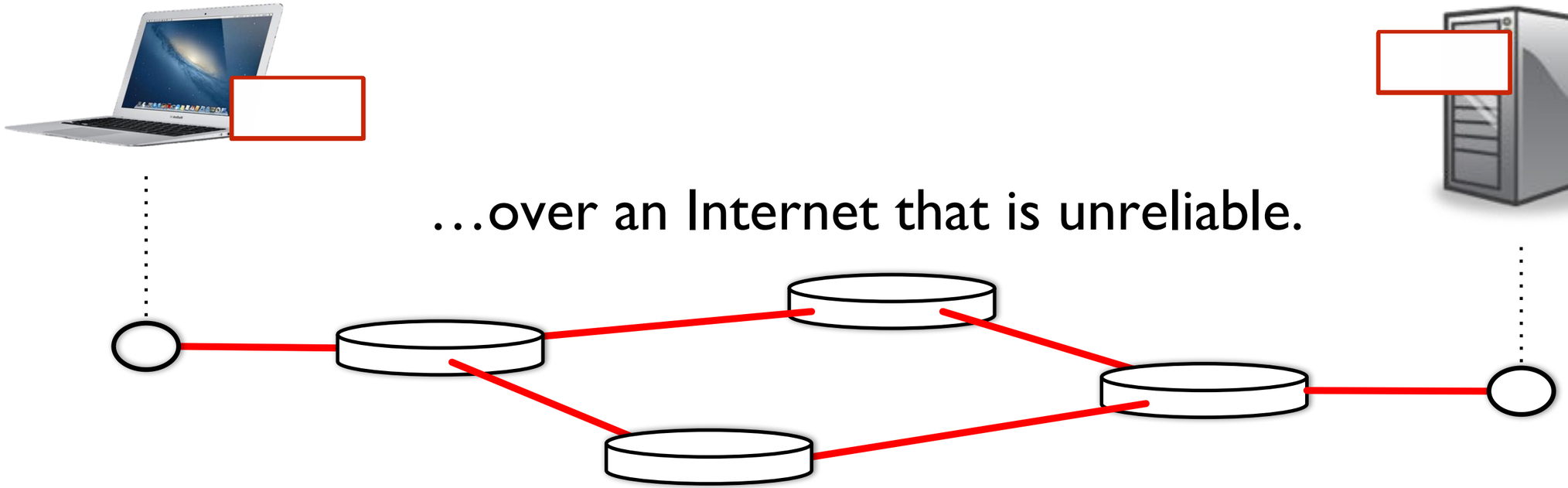


Summary of what we

My Program

Someone else's Program

Applications send and receive data in packets….



…over an Internet that is unreliable.

Packets are forwarded hop-by-hop using the IP destination address.

Our applications use TCP to make sure they are delivered and put back in the correct order.