Computer Networks
CSE 461

https://courses.cs.washington.edu/courses/cse461/20au/

About Us | Search | Contact Info

Lectures:	MWF 12:30-01:20
Section AA:	TH 01:30-02:20
Section AB:	TH 02:30-03:20
Section AC:	TH 03:30-04:20

Who

Shyam Gollakota, Professor
gshyam at cs.washington.edu

Daniel Starikov, TA
starikov at cs.washington.edu

Mehrdad Hessar, TA
mehrdadh at cs.washington.edu

Amy Xu, TA
amyxu at cs.washington.edu

Cailin Winston, TA
cailinw at cs.washington.edu

Milin Kodnongbua, TA
milink at cs.washington.edu

Rithik Duggempudi, TA
rrd3 at cs.washington.edu

Office Hours

M 01:30-02:20

Textbooks

Computer Networks (6E 19), Peterson [Online Book]
Computer Networks (5th Edition), Andrew Tanenbaum, David Wetherall

Class mailing list

The class email is cse461a_au20@u.washington.edu. It's updated nightly from the official registration list, and uses your u.wash email address.

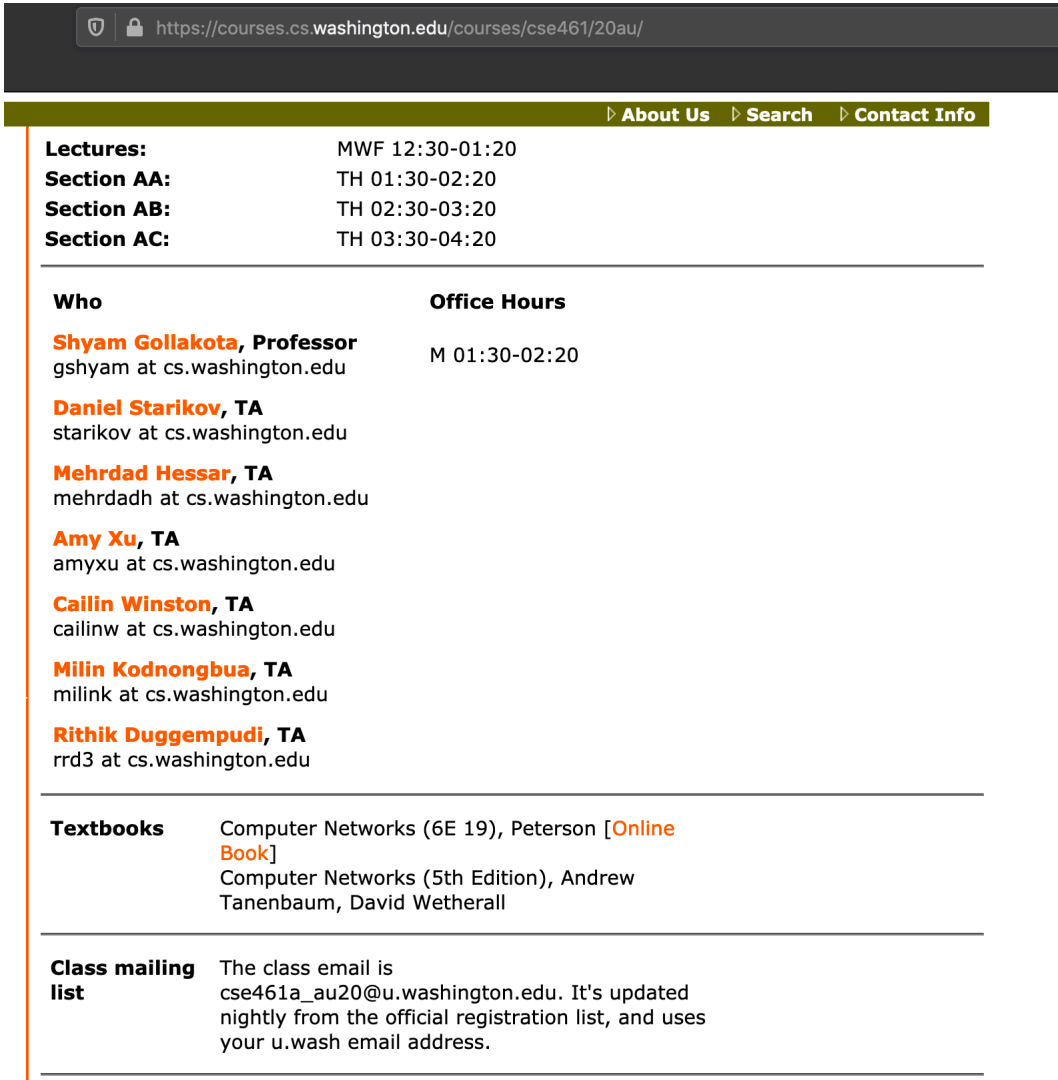
3 Projects (10+15+15%)

- Group of 3
- Can be same or different

Individual assignments (20%)

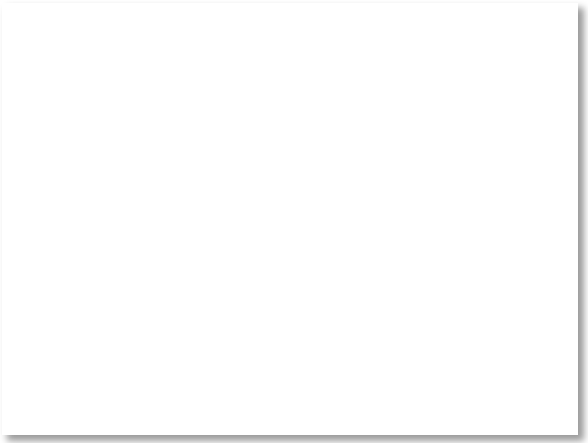
Mid term (20%)

Final (20%)



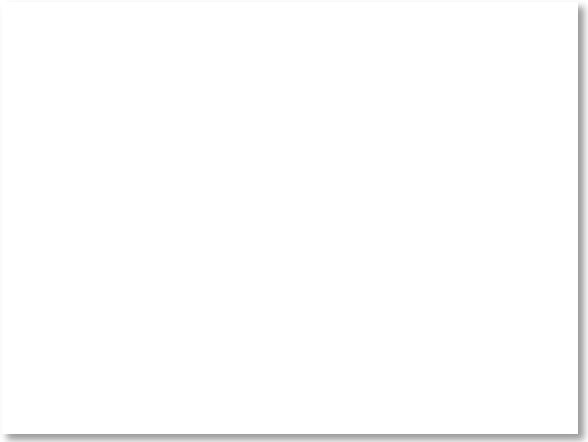
The Main Point

1. To learn how the Internet works »
   - What really happens when you "browse the web"?
   - What are TCP/IP, DNS, HTTP, NAT, VPNs, 802.11 etc. anyway?
2. To learn the fundamentals of computer networks



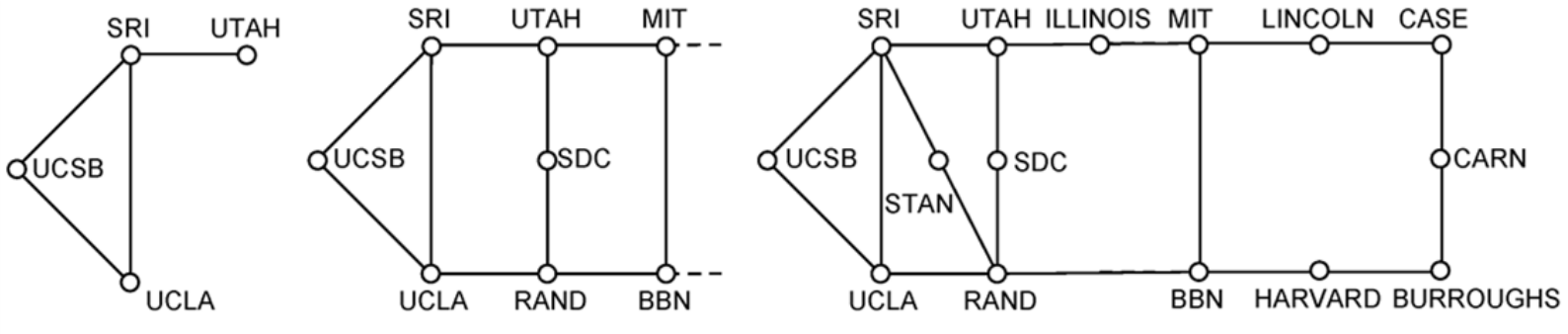
Why learn about the Internet?

1. Curiosity »
2. Impact on our world »
3. Job prospects!



From this experimental network …

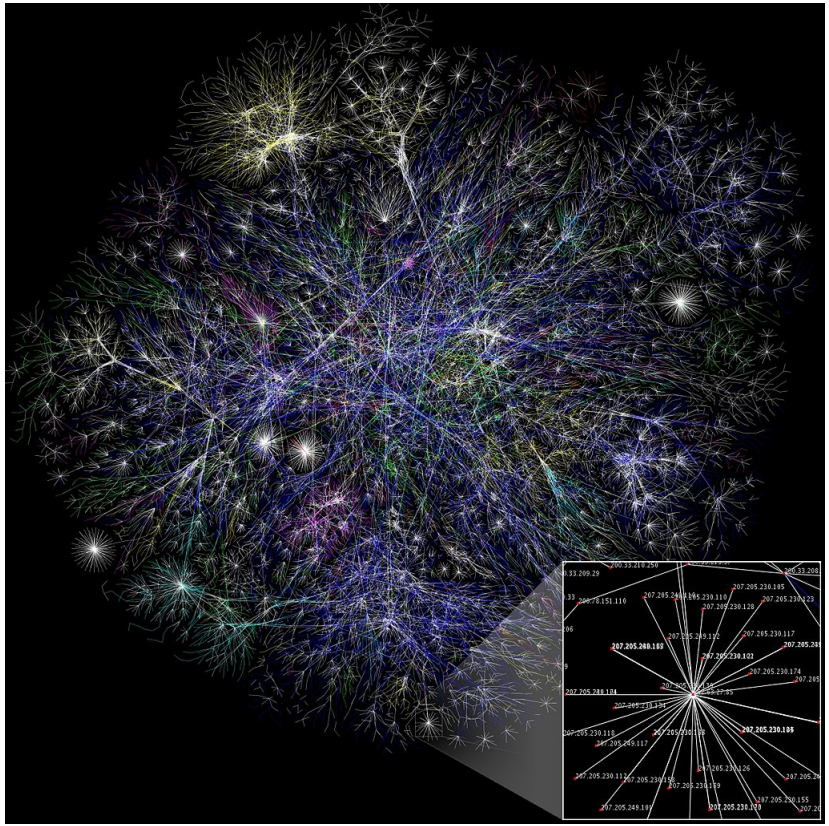
ARPANET ~1970



(a) Dec. 1969. (b) July 1970. (c) March 1971.

Internet ~2005

- An everyday institution used at work, home, and on-the-go
- Visualization contains millions of links



Attribution: By The Opte Project [CC-BY-2.5], via Wikimedia Commons

Internet – Societal Impact

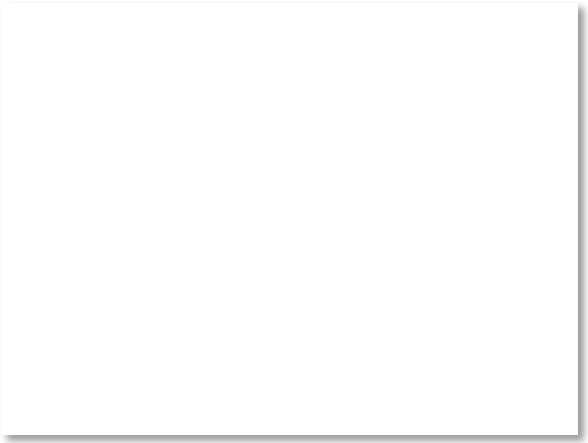
- An enabler of societal change
  - Easy access to knowledge
  - Electronic commerce
  - Personal relationships
  - Discussion without censorship

Internet – Economic impact

- An engine of economic growth
  - Advertising-sponsored search
  - “Long tail” online stores
  - Online marketplaces
  - Crowdsourcing

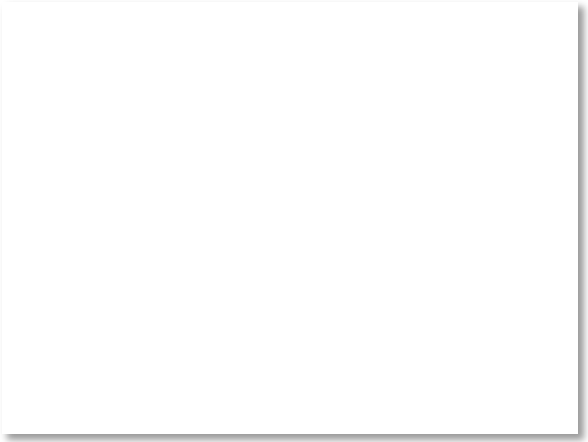
The Main Point (2)

1. To learn how the Internet works
2. To learn the fundamentals of computer networks
   - What hard problems must they solve?
   - What design strategies have proven valuable?



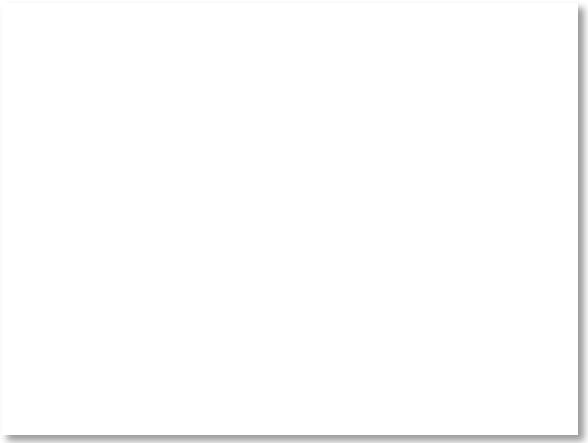
Why learn the Fundamentals?

1. Apply to all computer networks
2. Intellectual interest »
3. Change / reinvention »



Fundamentals – Intellectual Interest

- Example key problem: Reliability!
  - Any part of the Internet might fail
  - Messages might be corrupted
  - So how do we provide reliability?
- Reliability solutions
  - Codes to detect/correct errors
  - Routing around failures ...



Fundamentals – Intellectual Interest (2)

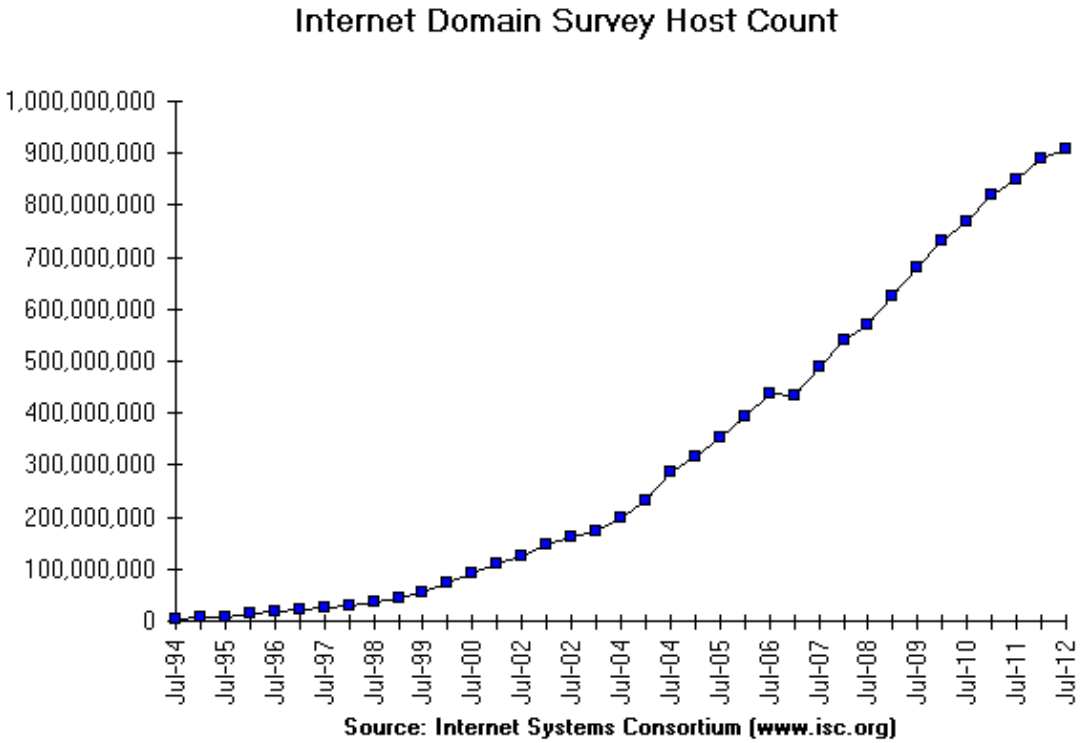
Key problem	**Example solutions**
Reliability despite failures	Codes for error detection/correction Routing around failures
Network growth and evolution	Addressing and naming Protocol layering
Allocation of resources like bandwidth	Multiple access Congestion control
Security against various threats	Confidentiality of messages Authentication of communicating parties

Fundamentals – Reinvention

- The Internet is constantly being re-invented!
  - Growth over time and technology trends drive upheavals in Internet design and usage »
- Today's Internet is different from yesterday's
  - And tomorrow's will be different again
  - But the fundamentals remain the same

Fundamentals – Reinvention (2)

- At least a billion Internet hosts and growing …



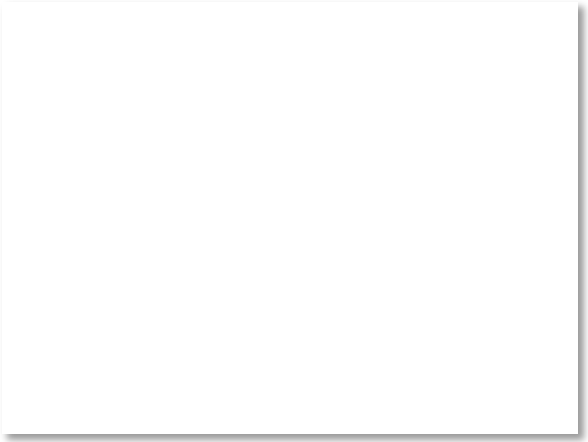
Fundamentals – Reinvention (3)

- Examples of upheavals in the past 1-2 decades

Growth / Tech Driver	**Upheaval**
Emergence of the web	Content Distribution Networks
Digital songs/videos	Peer-to-peer file sharing
Falling cost/bit	Voice-over-IP calling
Many Internet hosts	IPv6
Wireless advances	Mobile devices

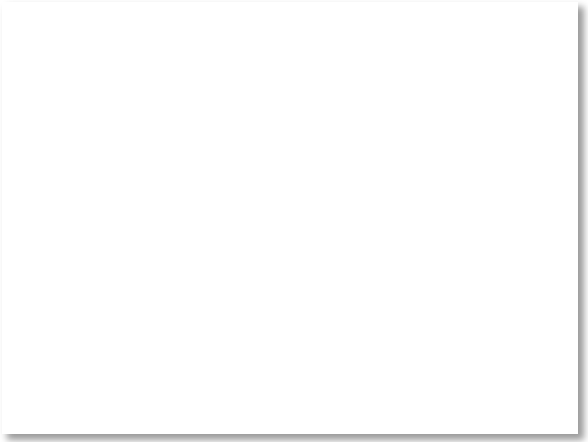
Not a Course Goal

- To learn IT job skills
  - How to configure equipment
    - e.g., Cisco certifications
  - But course material is relevant, and we use hands-on tools



Example Uses of Networks

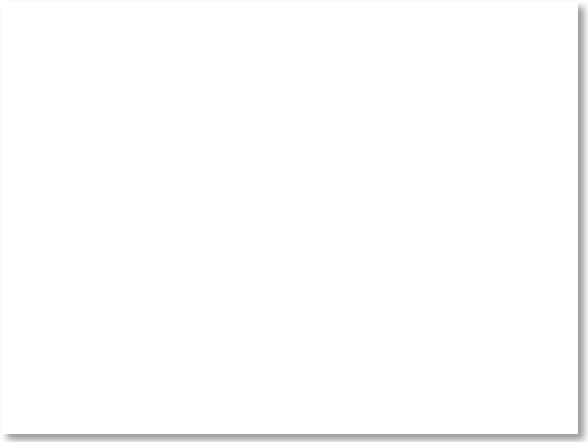
- Work:
  - Email, file sharing, printing, …
- Home:
  - Movies / songs, news, calls / video / messaging, e-commerce, …
- Mobile:
  - Calls / texts, games, videos, maps, information access …



Example Uses of Networks

- Work:
  - Email, file sharing, printing, …
- Hom
  - M s / video / , …
- Mobile:
  - Calls / texts, games, videos, maps, information access …

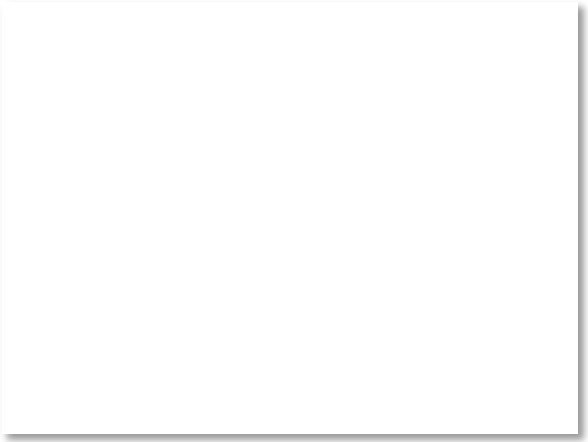
What do these uses tell us about why we build networks?



For User Communication

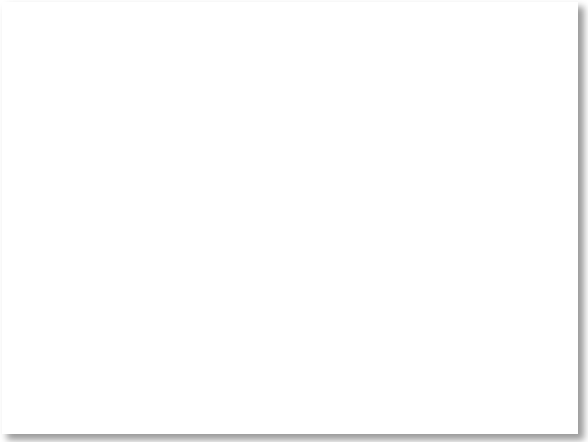
- From the telephone onwards:
  - VoIP (voice-over-IP)
  - Video conferencing
  - Instant messaging
  - Social networking

→Enables remote communication
  - Need low latency for interactivity



For Resource Sharing

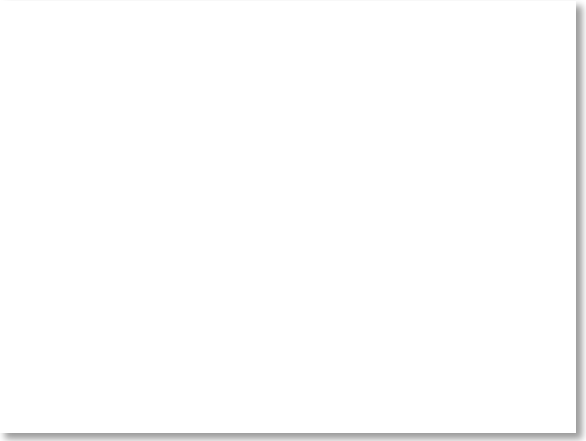
- Many users may access the same underlying resource
  - E.g., 3D printer, search index, machines in the cloud



For Computer Communication

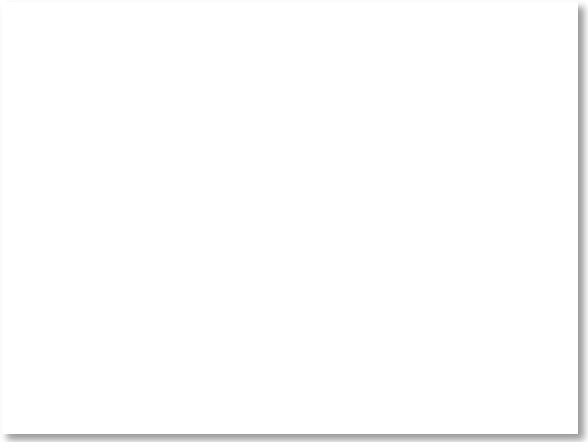
- To let computers interact with other computers
  - E.g., e-commerce, reservations

→ Enables automated information processing across different parties

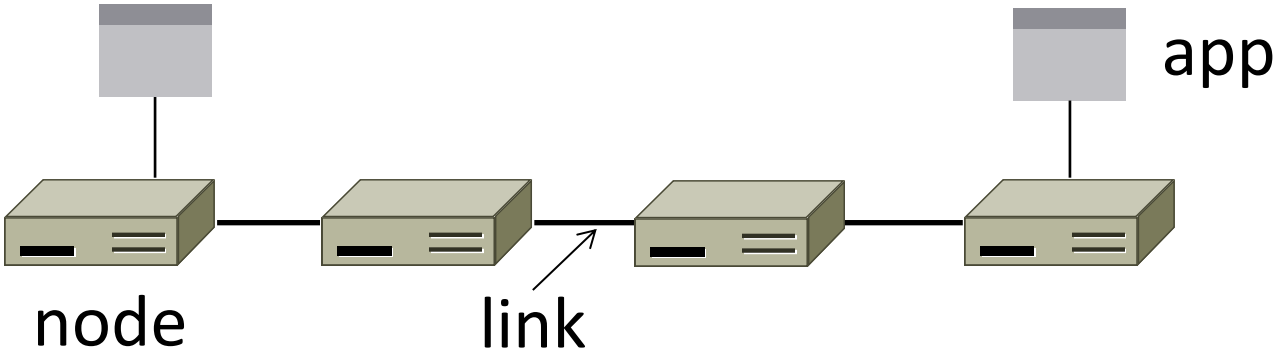


To Connect Computers to the Physical World

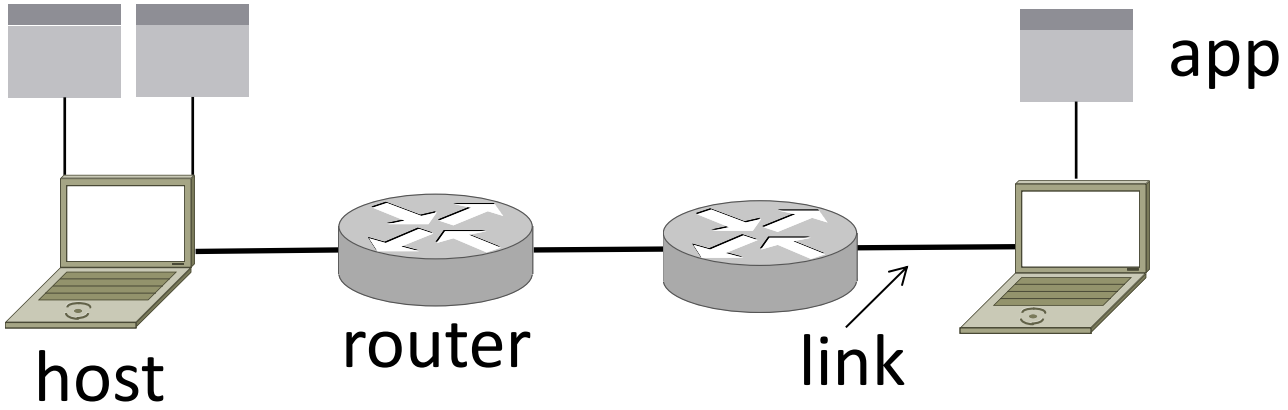
- For gathering sensor data, and for manipulating the world
  - E.g., webcams, location on mobile phones, door locks, …
- This is a rich, emerging usage



Parts of a Network (2)



Parts of a Network (3)

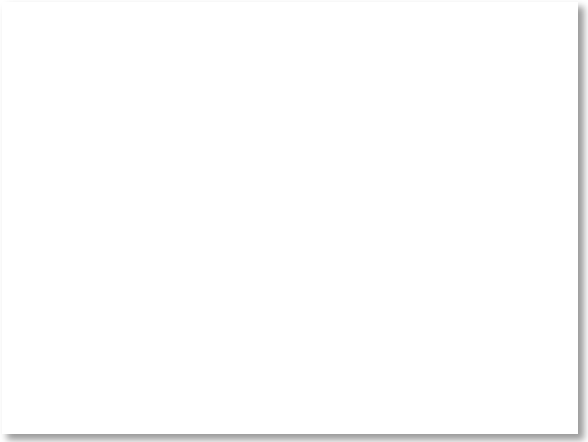


Component Names

Component	**Function**	**Example**
Application, or app, user	Uses the network	Skype, iTunes, Amazon
Host, or end-system, edge device, node, source, sink	Supports apps	Laptop, mobile, desktop
Router, or switch, node, hub, intermediate system	Relays messages between links	Access point, cable/DSL modem
Link, or channel	Connects nodes	Wires, wireless

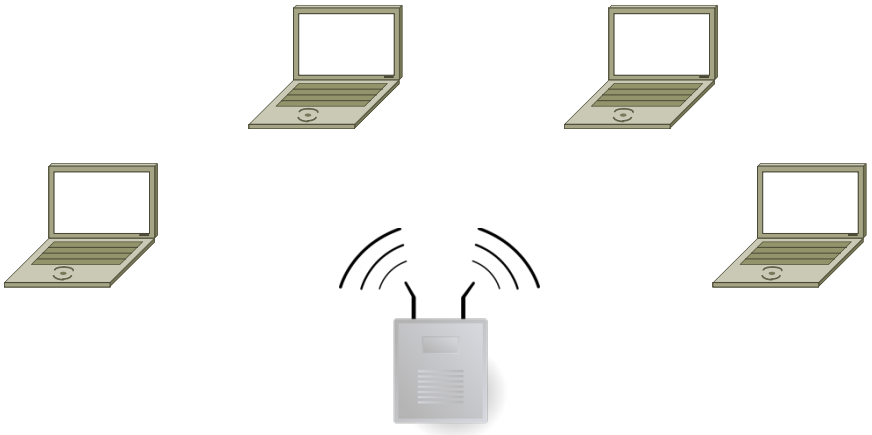
Types of Links

- Full-duplex
  - Bidirectional
- Half-duplex
  - Bidirectional
- Simplex
  - unidirectional



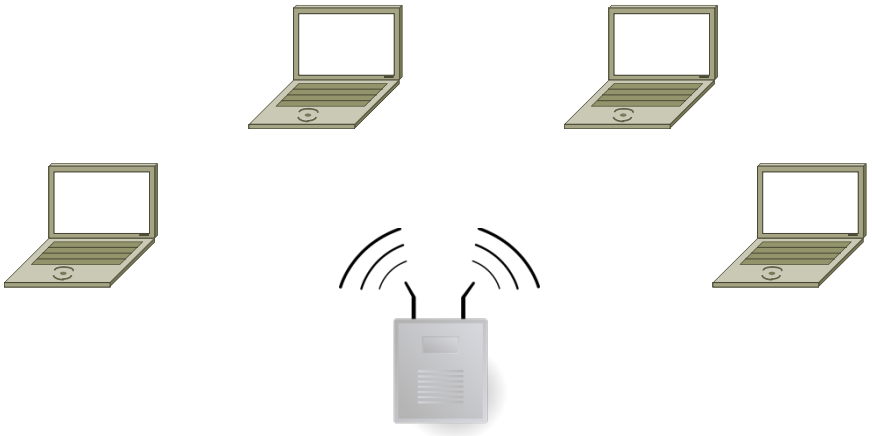
Wireless Links

- Message is broadcast
  - Received by all nodes in range
  - Not a good fit with our model



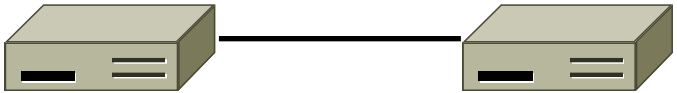
Wireless Links (2)

- Often show logical links
  - Not all possible connectivity



A Small Network

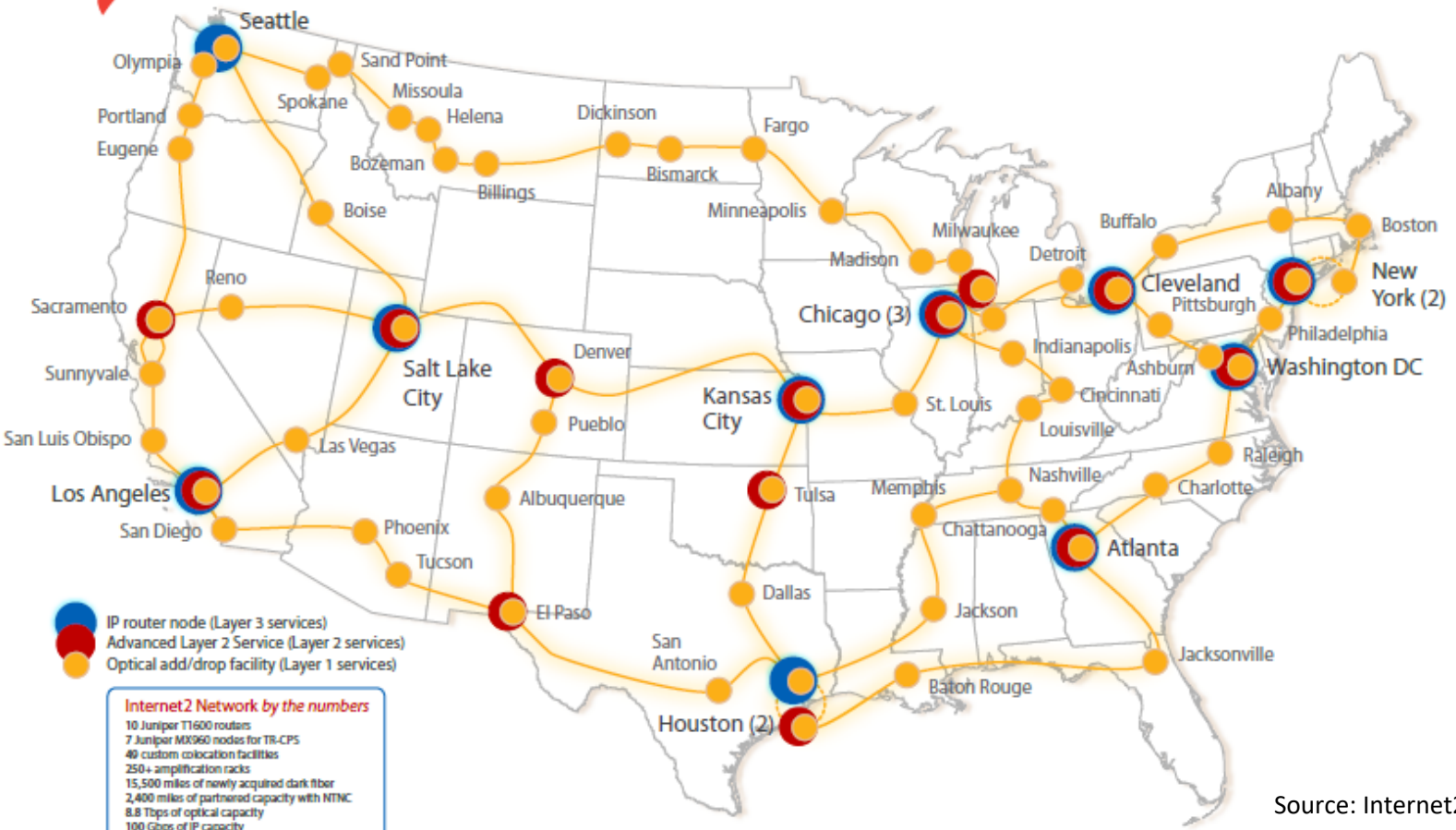
- Connect a couple of computers



- Next, a large network …

Internet2 Network Infrastructure Topology

October 2012



Internet2 Network *by the numbers*

- 10 Juniper T1600 routers
- 7 Juniper MX960 nodes for TR-CPS
- 40 custom colocation facilities
- 250+ amplification racks
- 15,500 miles of newly acquired dark fiber
- 2,400 miles of partnered capacity with NTNC
- 8.8 Tbps of optical capacity
- 100 Gbps of IP capacity
- 300+ Ciena ActiveFlex 6500 network elements

IN SUPPORT OF U.S.UCAN

NETWORK PARTNERS

ciena

cisco

INDIANA UNIVERSITY

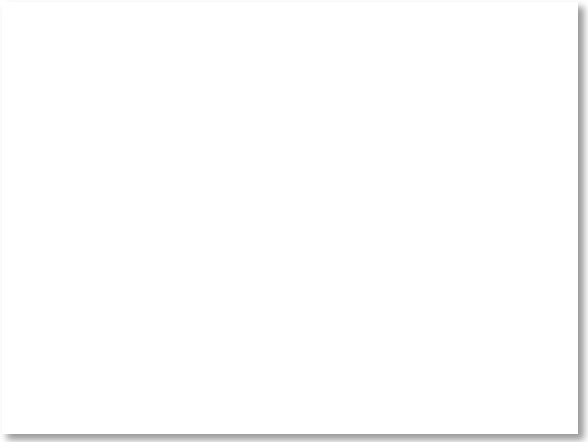
Infinera

JUNIPER NETWORKS

Source: Internet2

Example Networks (2)

- WiFi (802.11)
- Enterprise / Ethernet
- ISP (Internet Service Provider)
- Cable / DSL
- Mobile phone / cellular (2G, 3G, 4G)
- Bluetooth
- Telephone
- Satellite ...

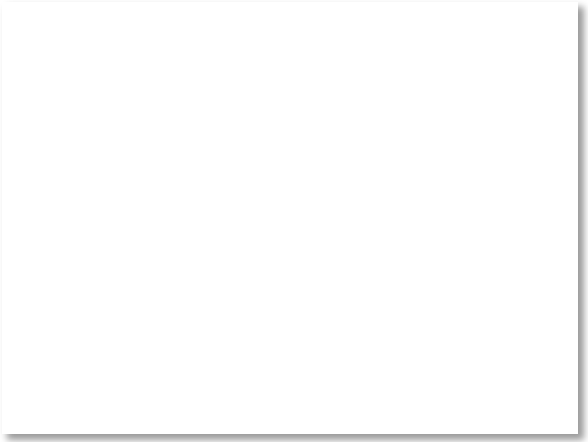


Network names by scale

Scale	Type	Example
Vicinity	PAN (Personal Area Network)	Bluetooth (e.g., headset)
Building	LAN (Local Area Network)	WiFi, Ethernet
City	MAN (Metropolitan Area Network)	Cable, DSL
Country	WAN (Wide Area Network)	Large ISP
Planet	The Internet (network of all networks)	The Internet!

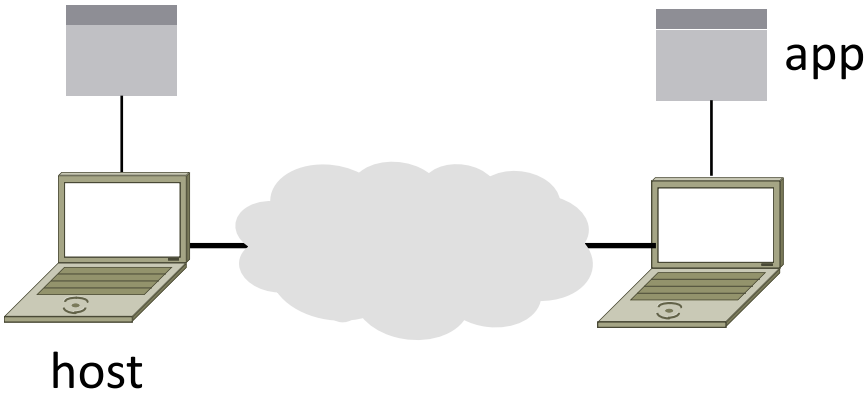
Internetworks

- An internetwork, or internet, is what you get when you join networks together
  - Just another network
- The Internet (capital “I”) is the internet we all use



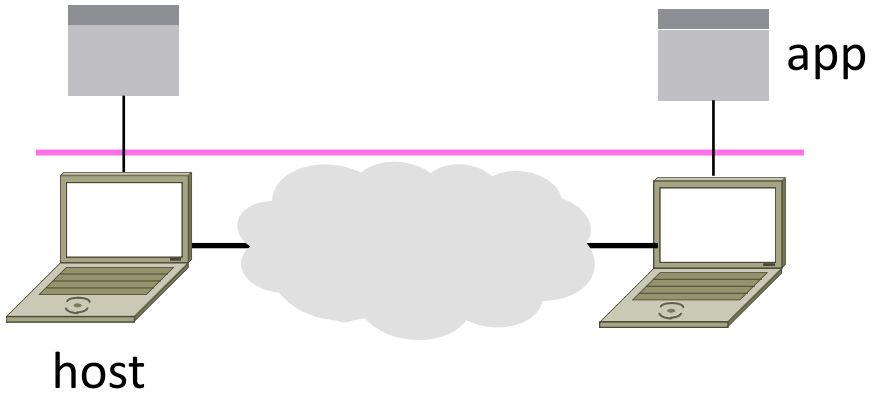
Key Interfaces

- Between (1) apps and network, and (2) network components
  - More formal treatment later on



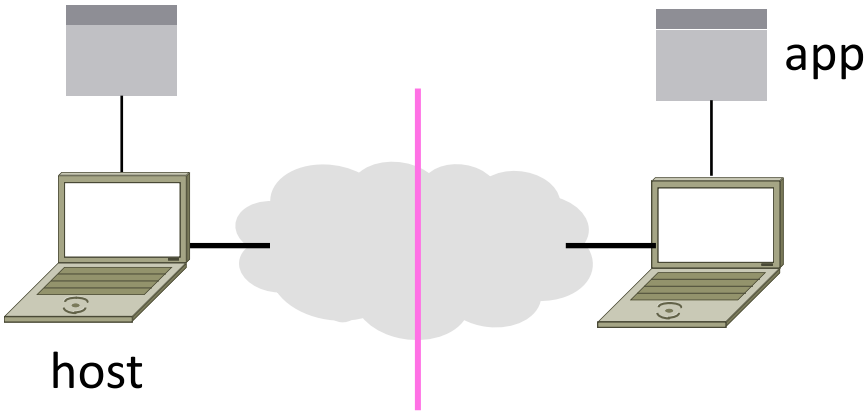
Key Interfaces (2)

1. Network-application interfaces define how apps use the network
   - Sockets are widely used in practice



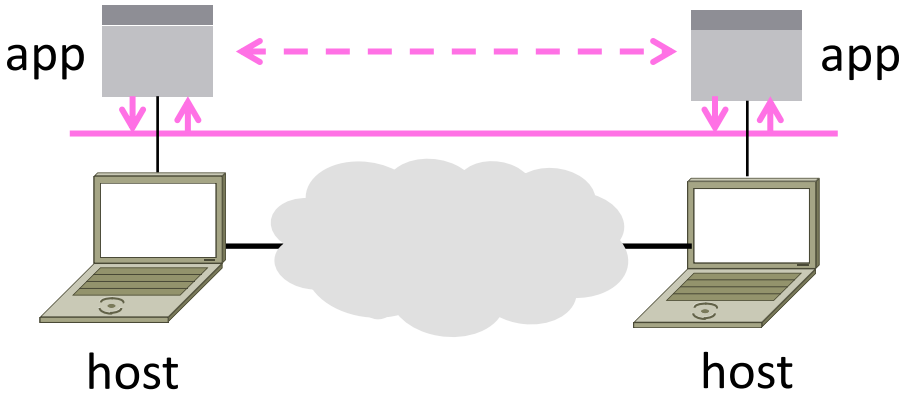
Key Interfaces (3)

2. Network-network interfaces define how nodes work together
   - Traceroute can peek in the network



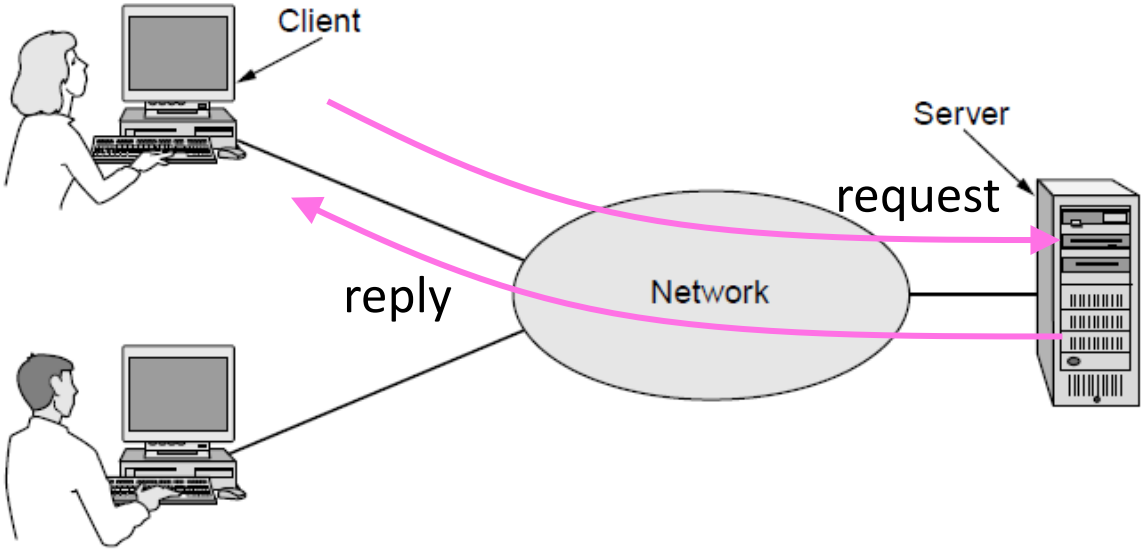
Network-Application Interface

- Defines how apps use the network
  - Lets apps talk to each other via hosts; hides the details of the network



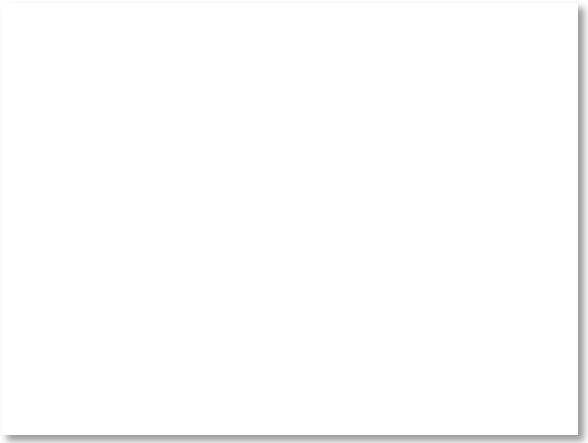
Motivating Application

- Simple client-server setup



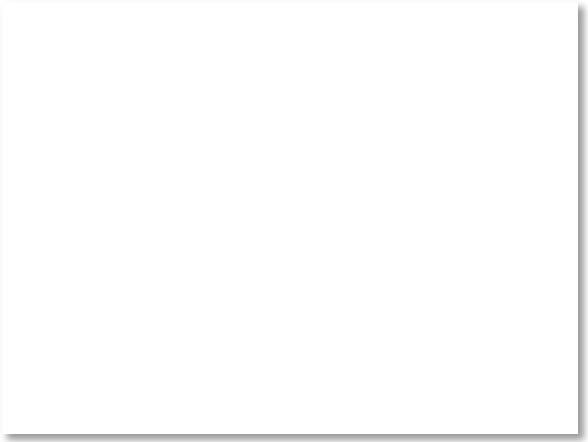
Motivating Application (2)

- Simple client-server setup
  - Client app sends a request to server app
  - Server app returns a (longer) reply
- This is the basis for many apps!
  - File transfer: send name, get file (§6.1.4)
  - Web browsing: send URL, get page
  - Echo: send message, get it back
- Let's see how to write this app …



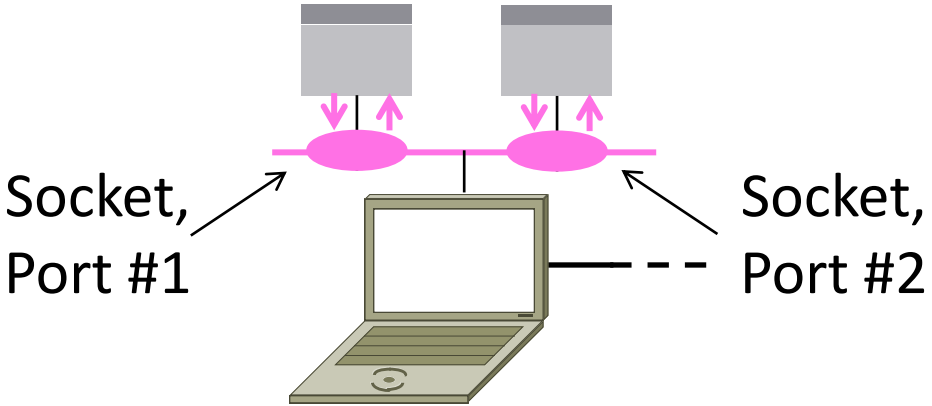
Socket API

- Simple abstraction to use the network
  - The network service API used to write all Internet applications
  - Part of all major OSes and languages; originally Berkeley (Unix) ~1983
- Supports two kinds of network services
  - Streams: reliably send a stream of bytes »
  - Datagrams: unreliably send separate messages. (Ignore for now.)



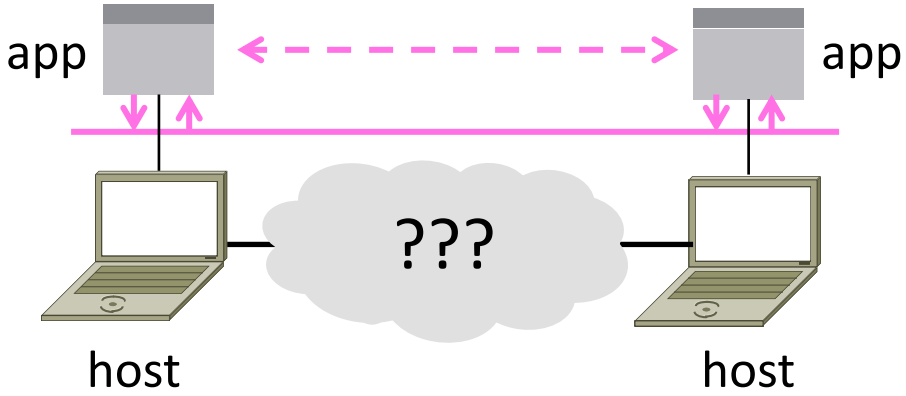
Socket API (2)

- Sockets let apps attach to the local network at different ports



Network Service API Hides Details

- Apps talk to other apps with no real idea of what is inside the network
  - This is good! But you may be curious …



Traceroute

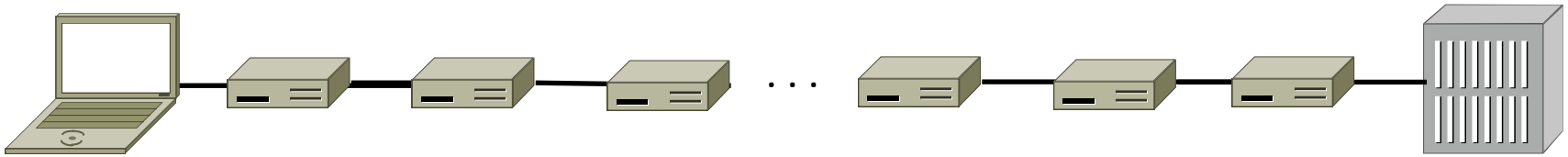
- Widely used command-line tool to let hosts peek inside the network
  - On all OSes (tracert on Windows)
  - Developed by Van Jacobson ~1987
  - Uses a network-network interface (IP) in ways we will explain later

Van Jacobson

: Credit: Wikipedia (public domain)

Traceroute (2)

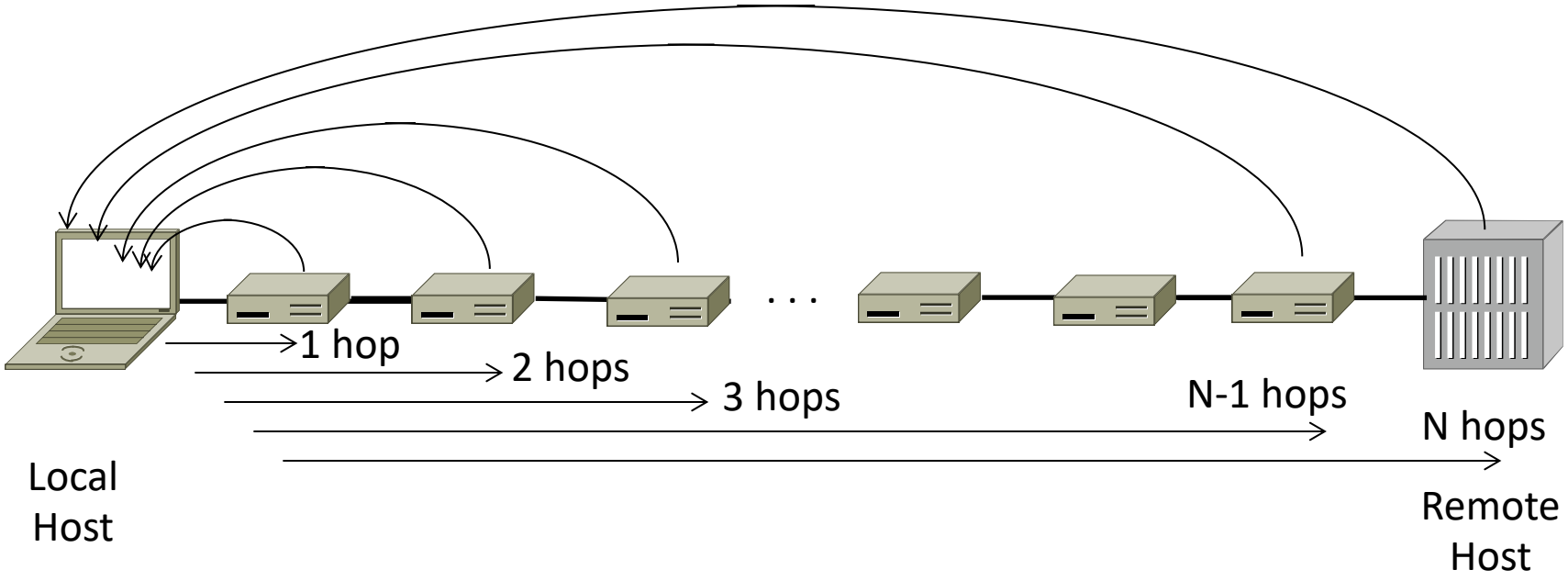
- Probes successive hops to find network path



Local Host

Remote Host

Traceroute (3)



Using Traceroute

Using Traceroute (2)

- ISP names and places are educated guesses