CS118: Lecture 1, Introduction, Layering

George Varghese

September 26, 2022

Lecture Outline

- Administration, Syllabus
- Why Study Networking?
- Networking at UCLA
- A Hats Analogy
- Overview of a Web Transfer
- Layering More Formally

New Features

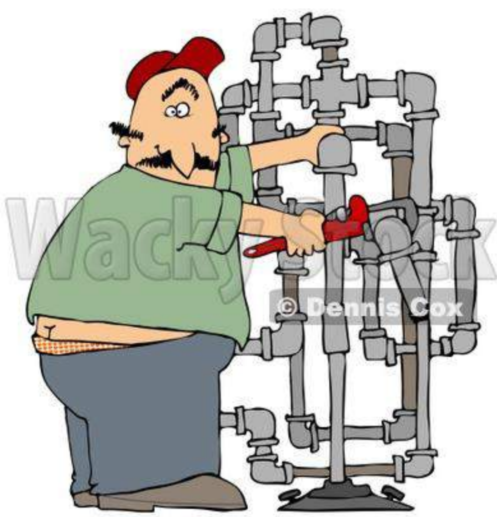
- Miniflips: short presentations of the ideas in a lecture before class begins.

- Interviews with "Reinventors of Internet". More details soon

- Other details: See syllabus

WHY NETWORKING

IS NETWORKING MERELY PLUMBING?



Network

Innovation?

Science?

Operator

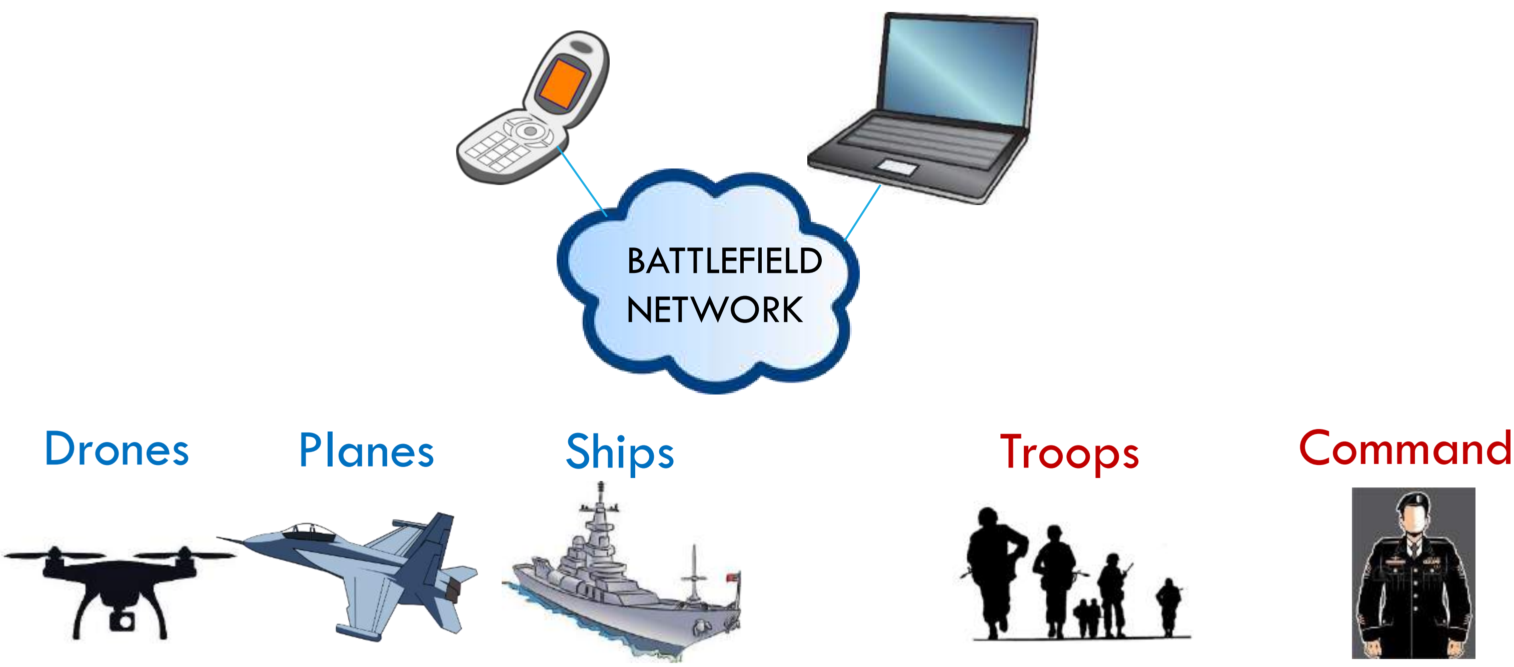
NO, ITS THE NEW COMPUTATIONAL BUS . . .

Search Shop Stay Ride

Design

Optimize

AND THE NEW AGILE BATTLEFIELD NETWORK . . .



AND STILL THE WORLD'S COMMUNICATION CHANNEL . . .

Text

Blog

Video

Friend

Share

SOME COOL FACTS. . .

June 2008 Google's index of the web consists of 1 trillion unique URLs

December 2012 Annual e-commerce sales top $1 trillion worldwide for the first time.

February 2014 45% of internet users ages 18-29 in serious relationships say the internet has had an impact on their relationship.

Summer 2014 The number of Internet users worldwide reaches 3 billion.

November 2014 Only 23% of respondents to a Pew online survey know that the "the Internet" and the "the World Wide Web" do not refer to the same thing.

From "A Very Short History of The Internet, Gil Pres, in Forbes

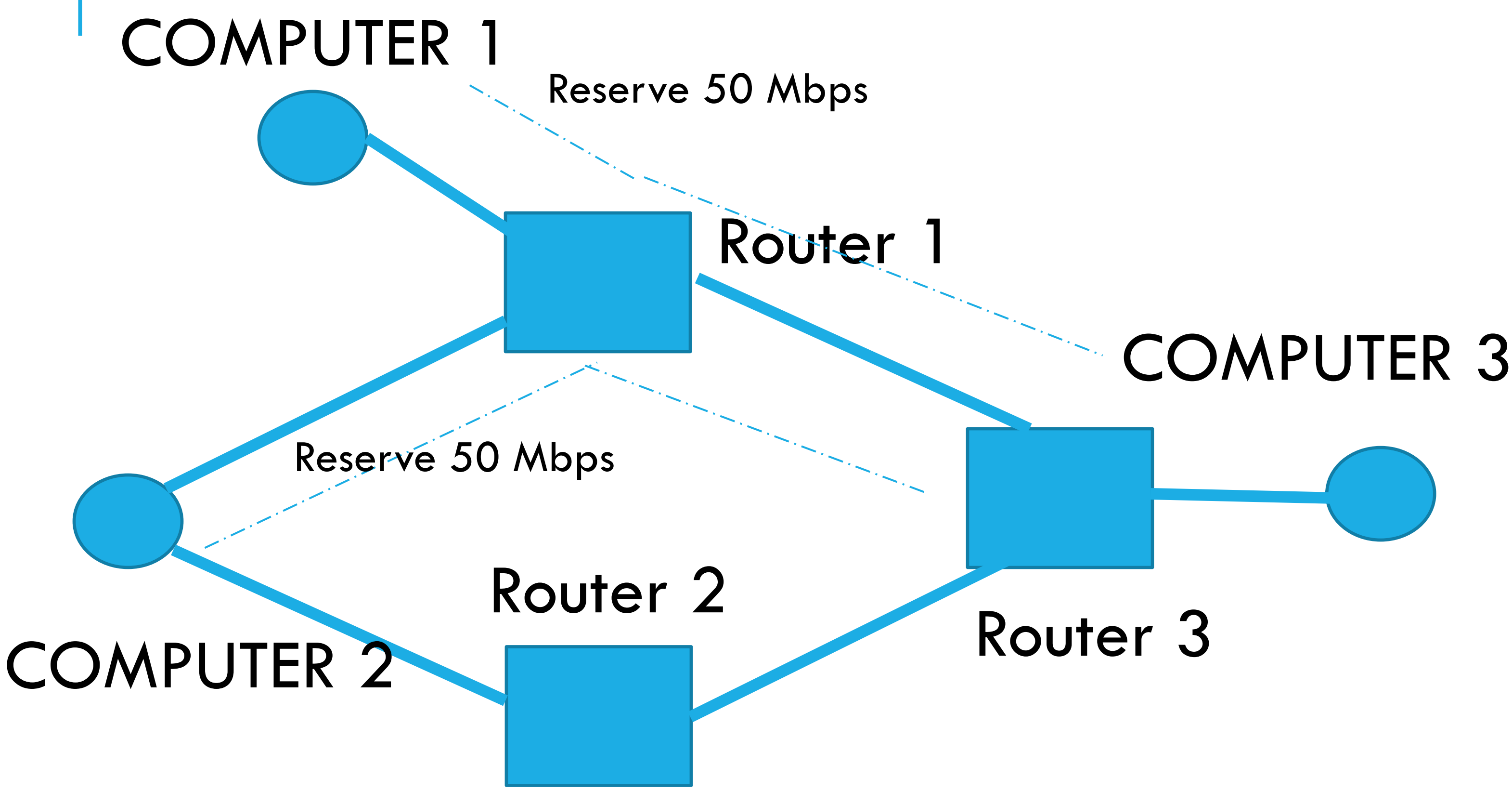
So Why Study Networking?

- Because its already part of your life: the computational bus for services you need and the communication channel to stay connected,
- Because its the hip place to be: the Internet is constantly growing Wouldn't you like it to be all geek (as opposed to greek) to you?
- Because there's lots of cool work to do. Google, Facebook, Uber, and even Airbnb are all hiring. Internet startups (see Nicira.com) continue to redefine networking. Wouldn't like to know enough to get in the door of a great startup?

Now for a whirlwind tour . . . Fasten your seatbelts

PACKET SWITCHING: THE NEW PARADIGM USED IN THE INTERNET

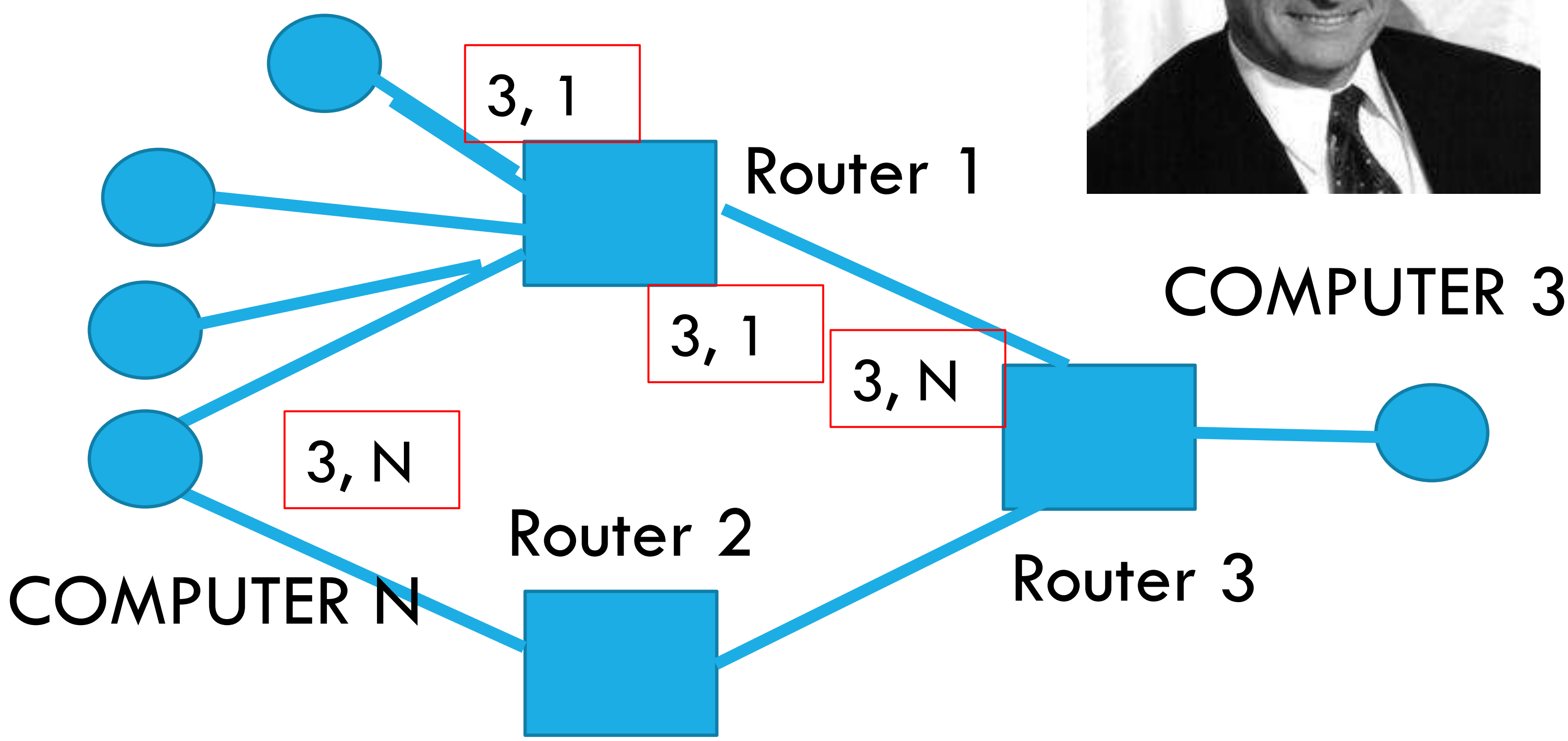
CIRCUIT SWITCHING



All links 100 Mbps

Way phone networks were built in 1960s and way AT&T wanted to do data

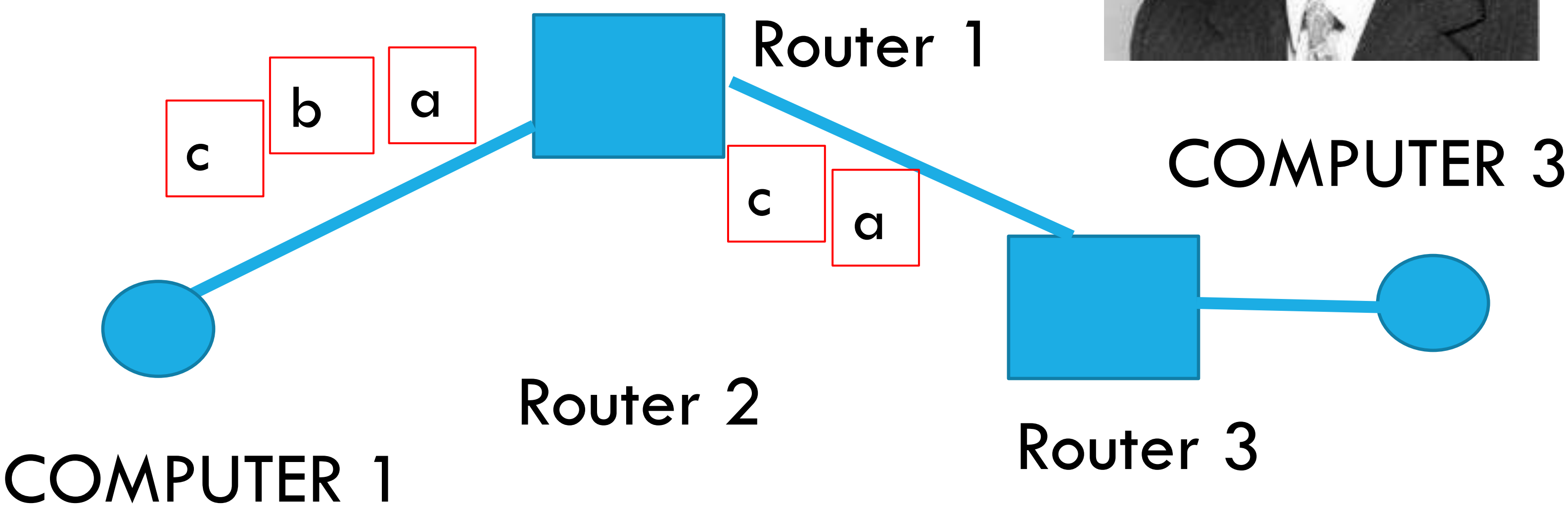
PACKET SWITCHING 1: LEN KLEINROCK



All links 100 Mbps

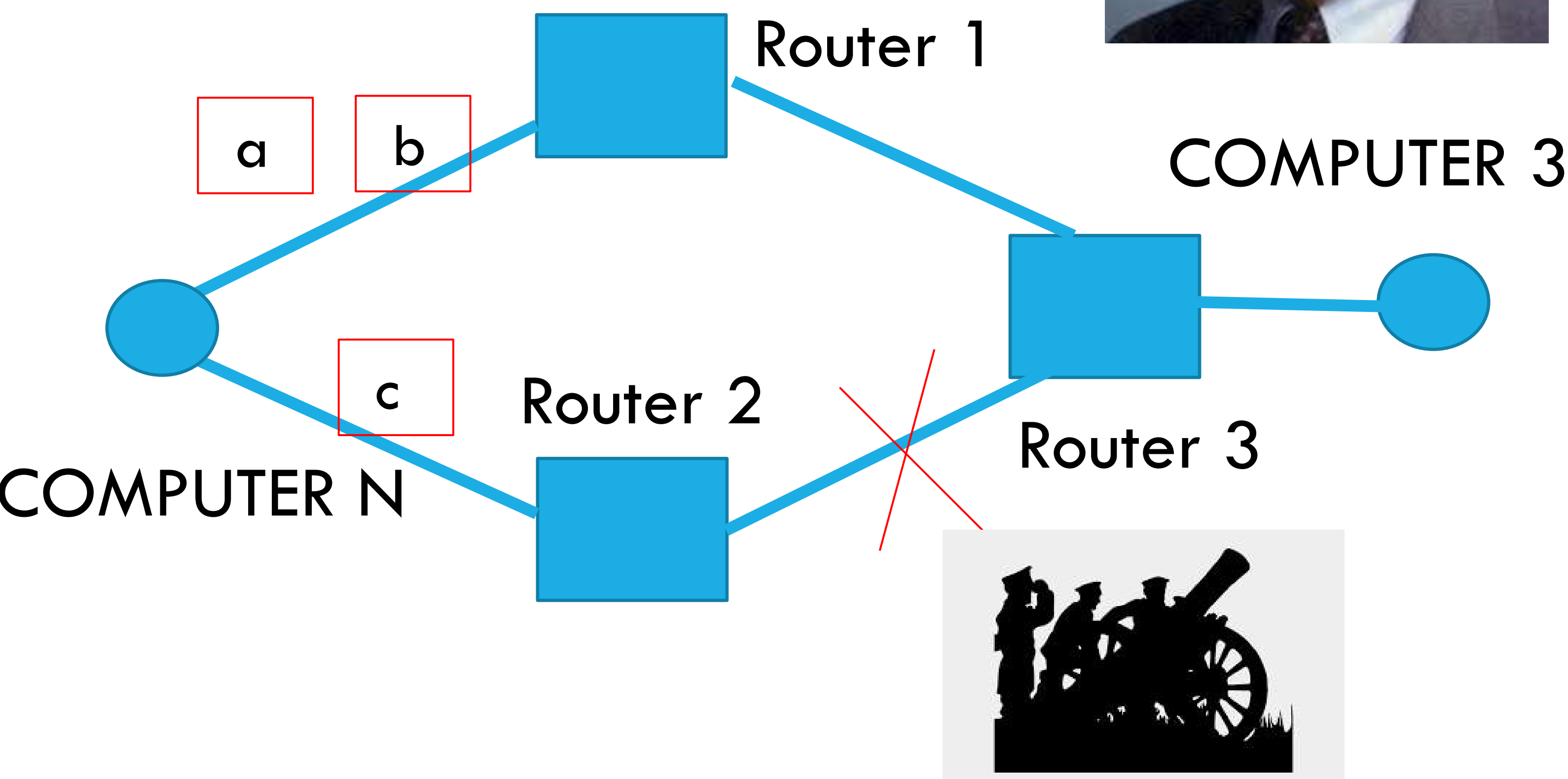
MIT PhD thesis: No reservations! Can multiplex more bursty traffic. Analysis

PACKET SWITCHING 2: DONALD DAVIES



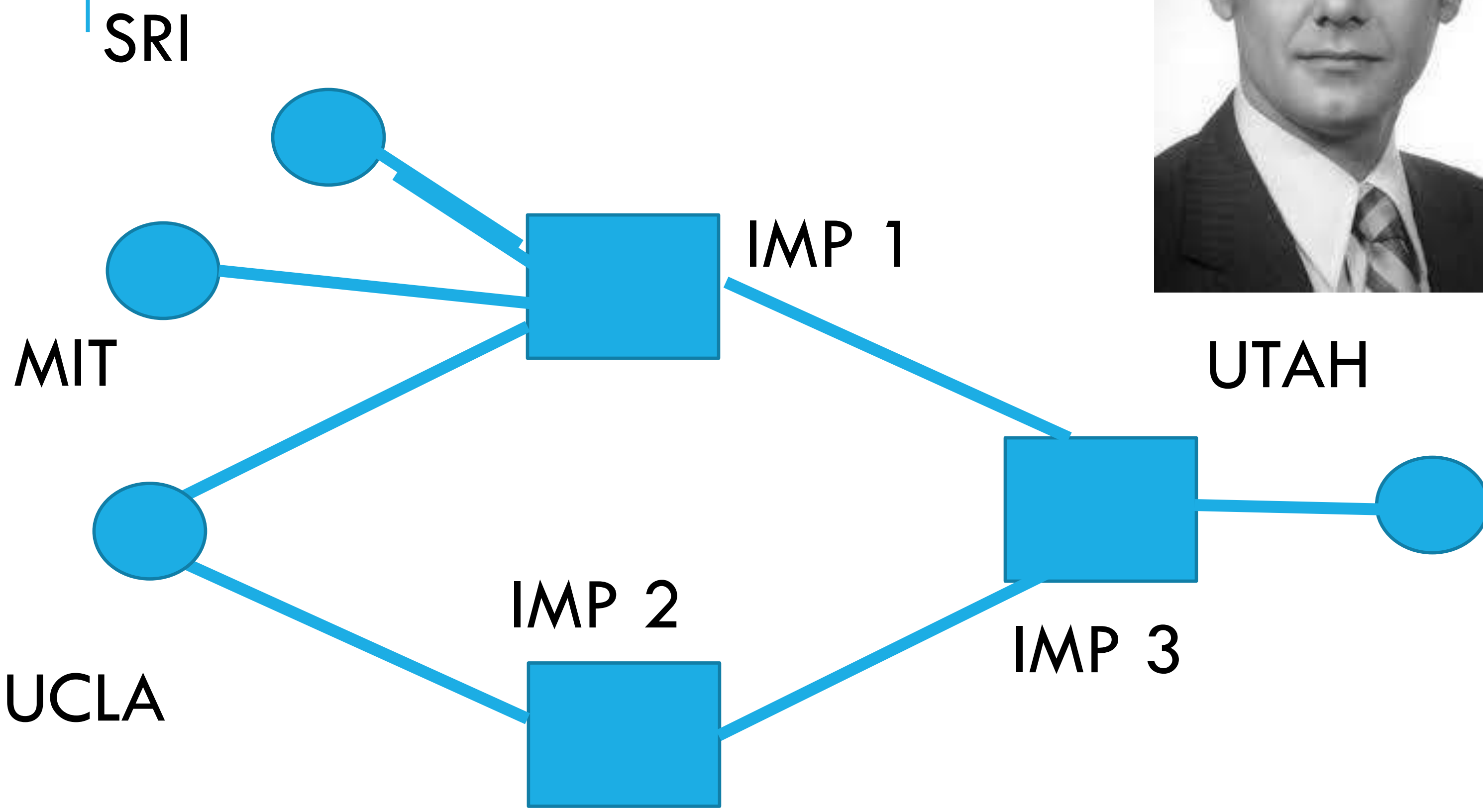
At England's NPL: <u>easier reliability</u>. If packet b gets dropped, only repeat b

PACKET SWITCHING 3: PAUL BARAN



Rand Report: More survivable. If enemy can attack some paths, can still send data

PACKET SWITCHING 4: LARRY ROBERTS



ARPA: Resource Sharing: could share expensive graphics computer at Utah

Contracts with BBN to build IMP and with UCLA to do measurement and first IMP

https://en.wikipedia.org/wiki/Packet_switching

NETWORKING@UCLA

REVOLUTION 1: IN 1969 A UCLA TEAM LED BY LEN KLEINROCK SENT THE FIRST INTERNET MESSAGE VIA THE FIRST INTERNET ROUTER. VISIT 3420 BOELTER HALL

REVOLUTION 2: FROM 2002-2012 DEBORAH ESTRIN AND GREG POTTIE SET UP UCLA CENS THAT ANTICIPATED THE WIRELESS SENSOR NETWORK AGE

REVOLUTIONS BREWING:

LIXIA ZHANG: NAMED DATA NETWORKING.

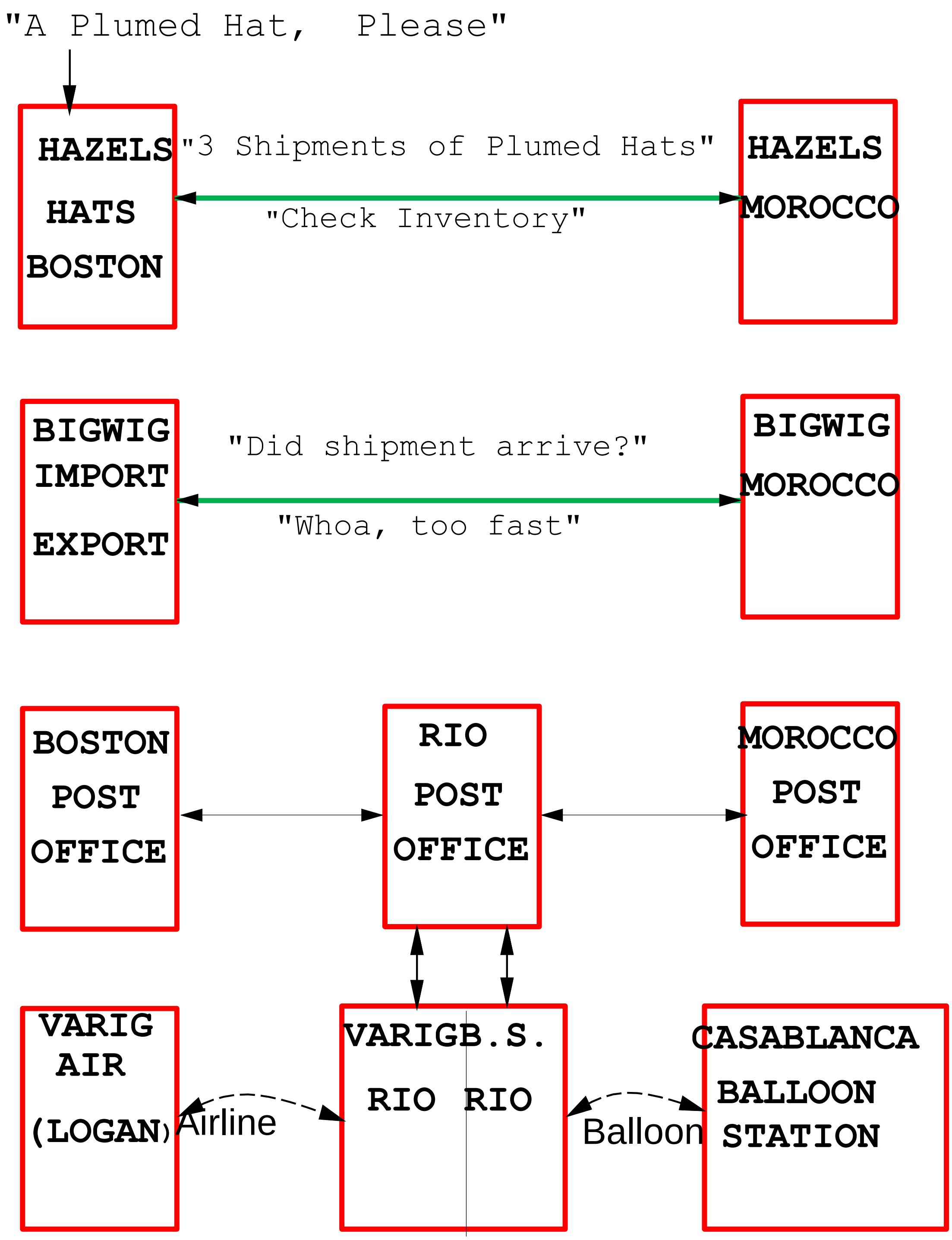
SONGWU LU: CELLULAR VERIFICATION

ME: NETWORK DESIGN AUTOMATION

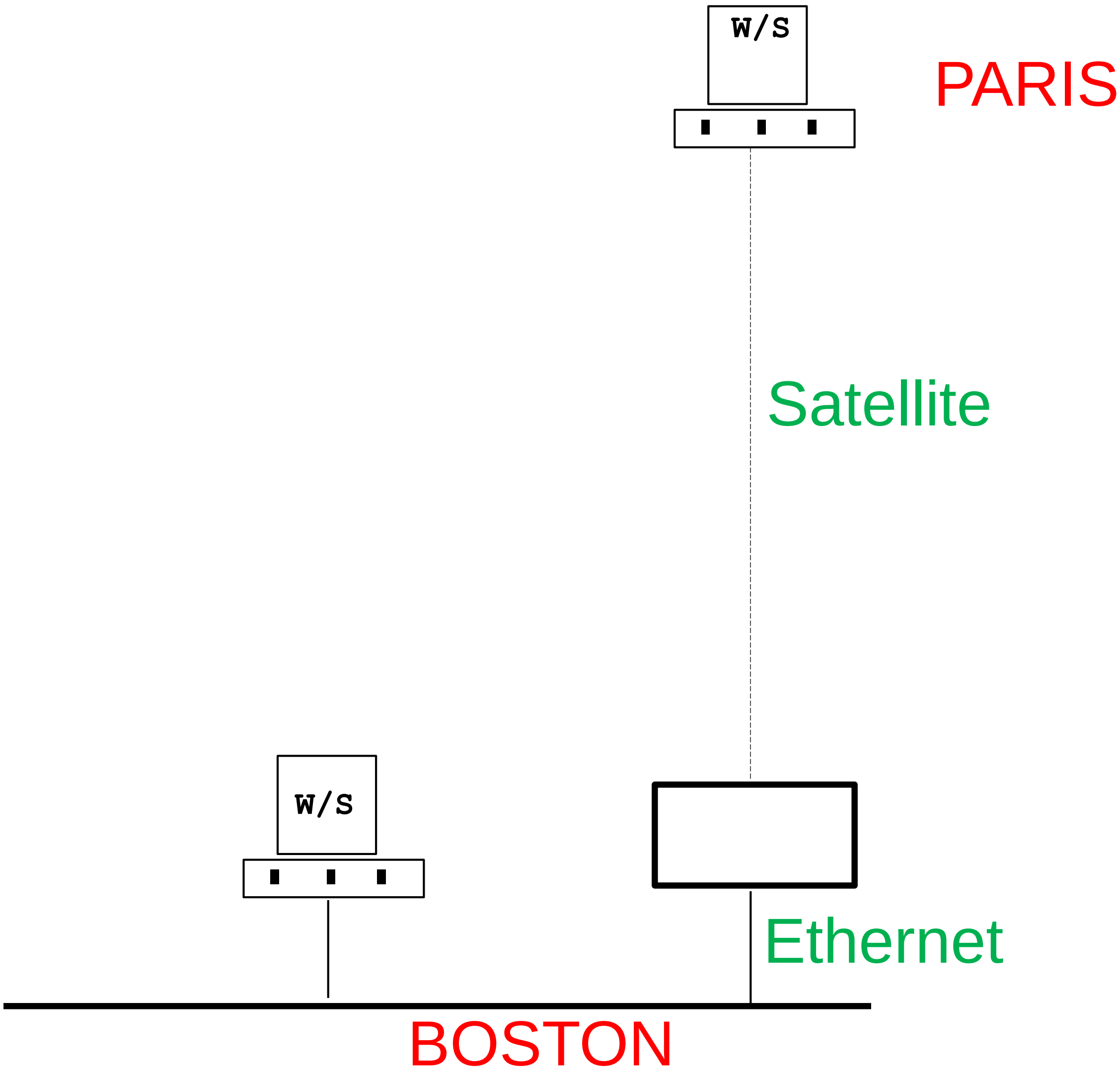
YOU TOO!

HAT TRANSFER ANALOGY

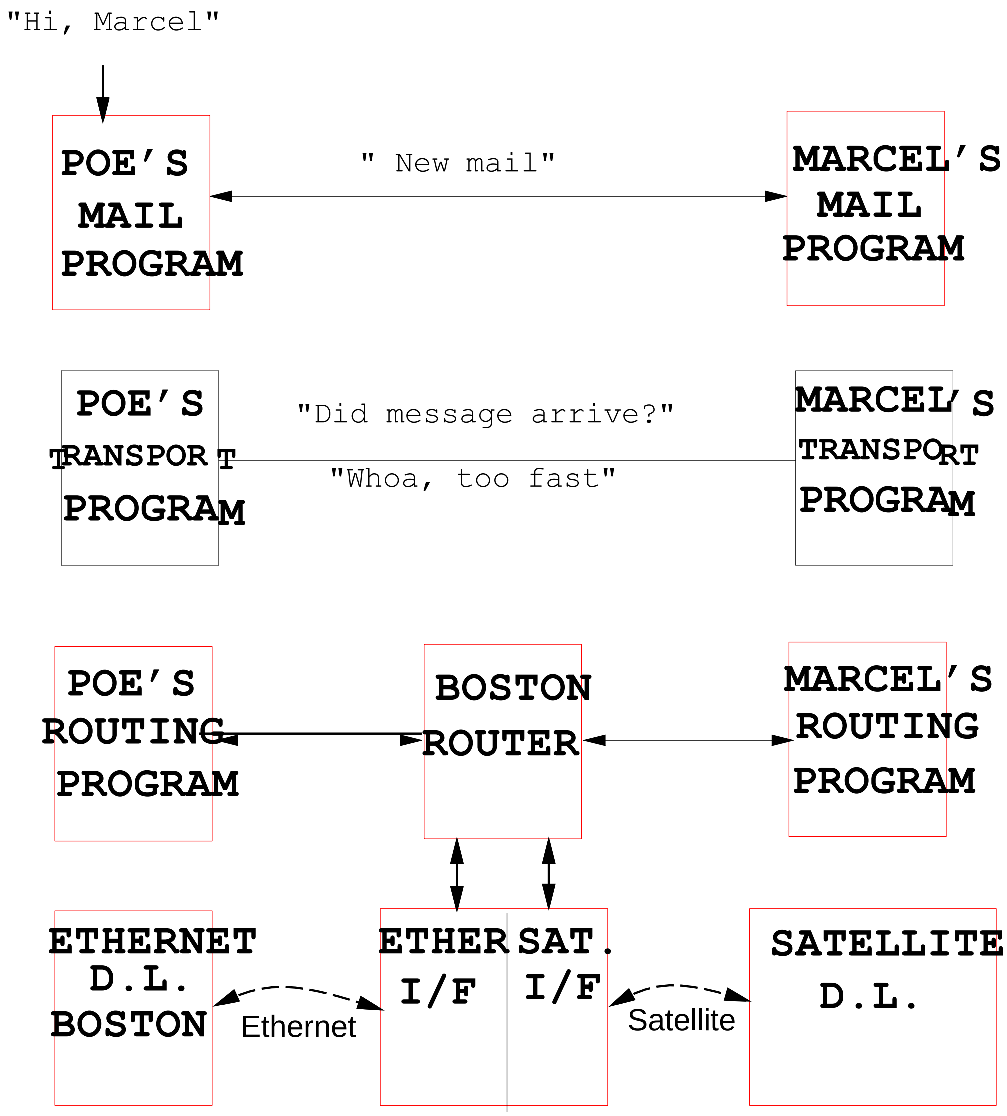
HAT TRANSFER ANALOGY



MAIL FROM BOSTON TO PARIS



MAIL: POE TO PROUST



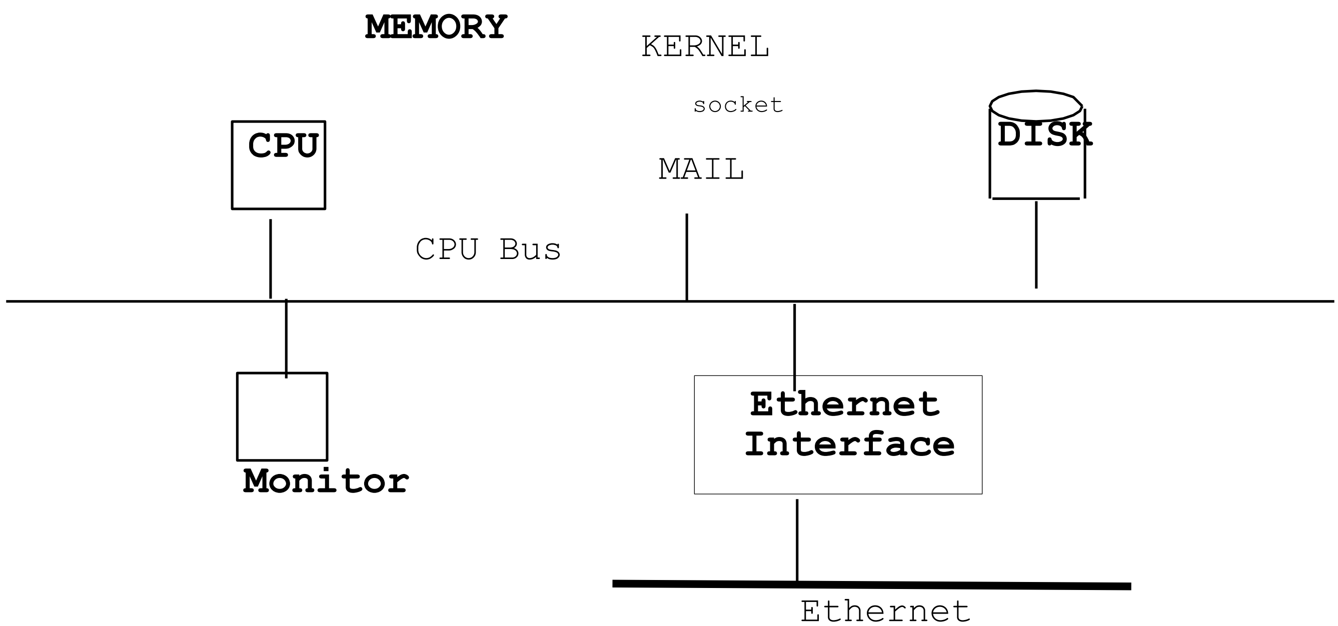
Deeper Issues with the Analogy

- Equivalents of Immigration and customs?
- Equivalents of postage?
- Chopping up hats into slices as in Data Link?
- Out-of-band versus in-band communication between layers
- Discuss this in your breakout rooms

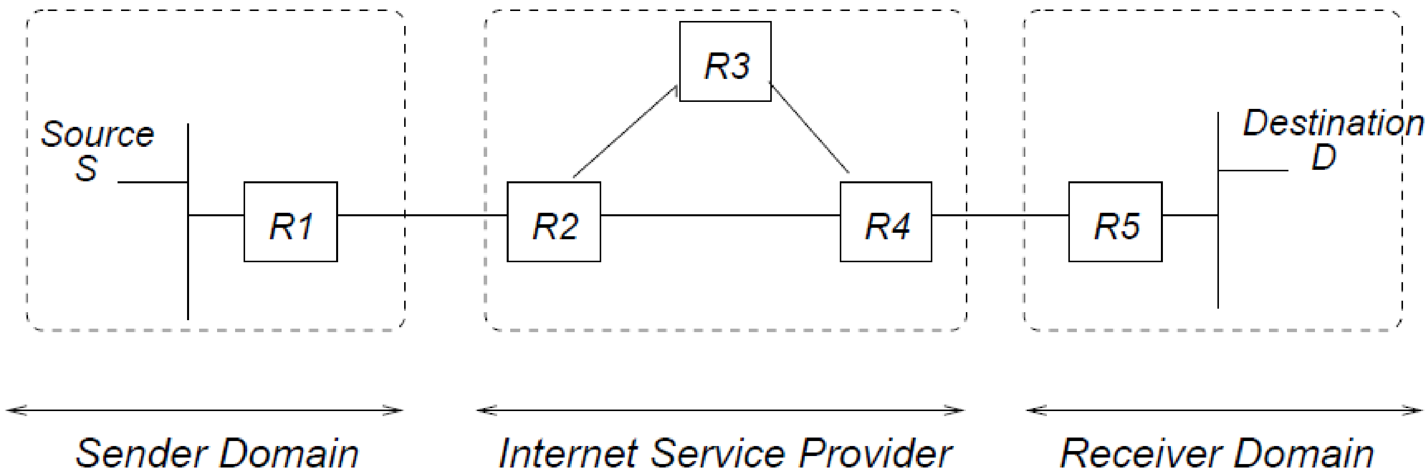
ANATOMY OF A WEB TRANSFER

Context 1: End Device Networking

A TYPICAL LAPTOP

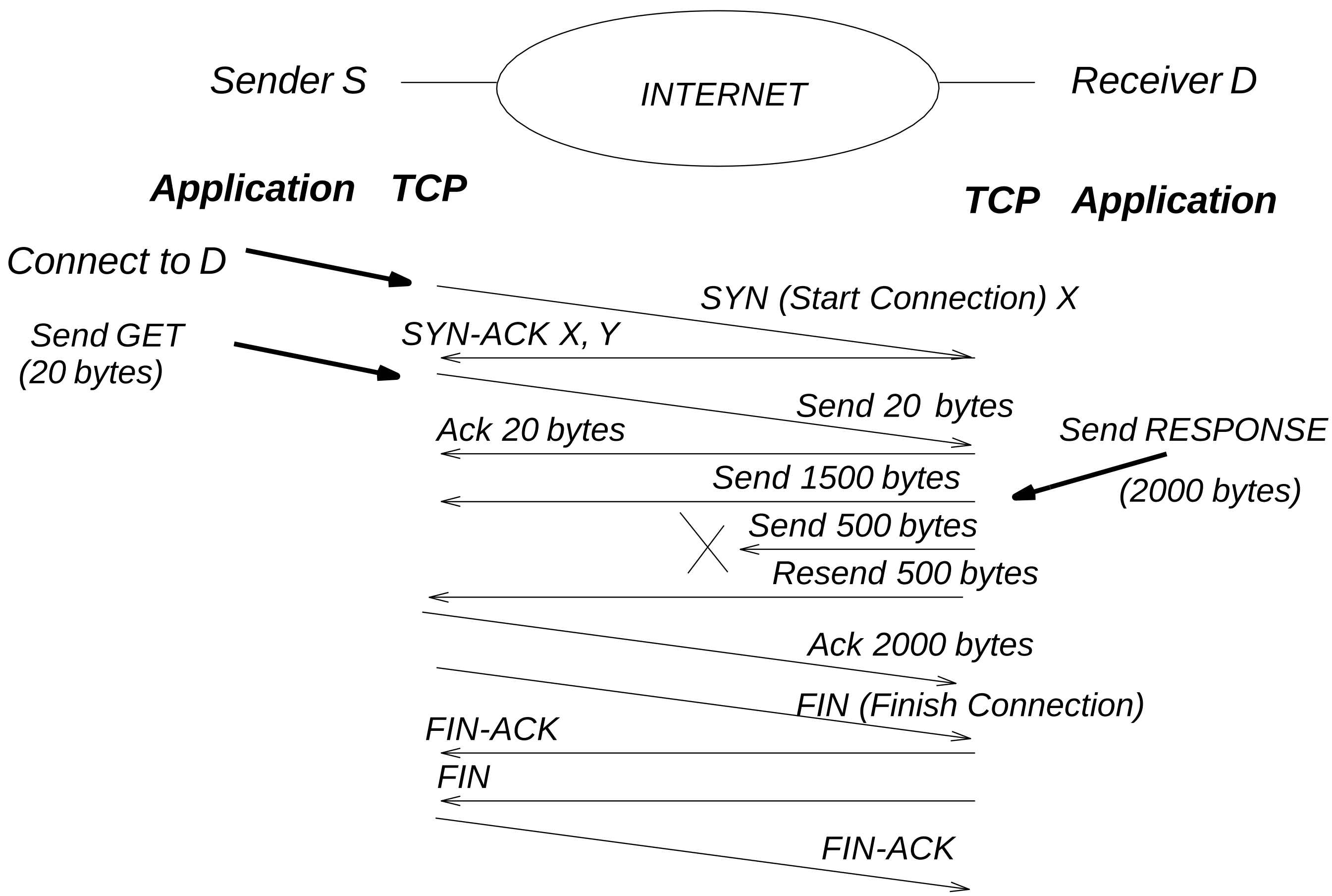


Context 2: IP Routing



- Forwarding: routers forward packets based on looking up destination addresses in a forwarding table
- Routing: Routers work together to build a forwarding table. OSPF (within domains), BGP (between ISPs)

A Web Transfer

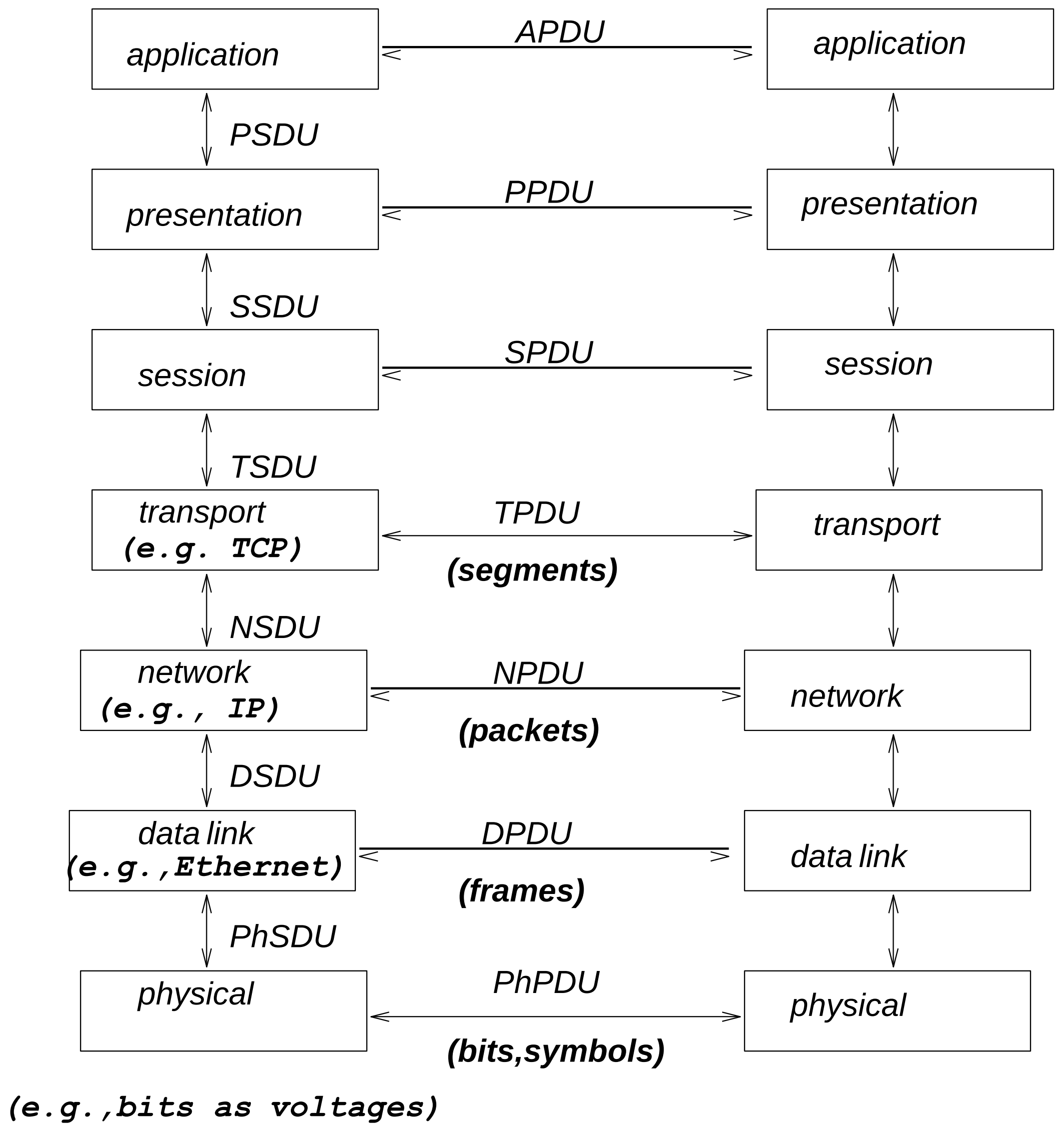


- Watch the steps next time you access a web page. Looking up host name, transferring data, etc.

LAYERING MORE FORMALLY

Layering

- Recall that in Hazel's Hats analogy, we simplified a complex task (transferring hats) by outsourcing reliable transmission to an import-export agency who in turn outsourced package delivery to the post office who in turn outsourced single-hop package transmission to a carrier etc.
- Similarly, in networking, an email transfer is simplified by subcontracting reliable delivery to a transport like TCP who subcontracts packet delivery to the network layer who subcontracts to the Data Link etc.
- This division of labor in networking is called *layering*. Each horizontal slice (layer) is given a number starting with 1 for physical, 2 for data link etc. While TCP only uses bottom 4 layers, most general model is OSI/ISO model shown next.



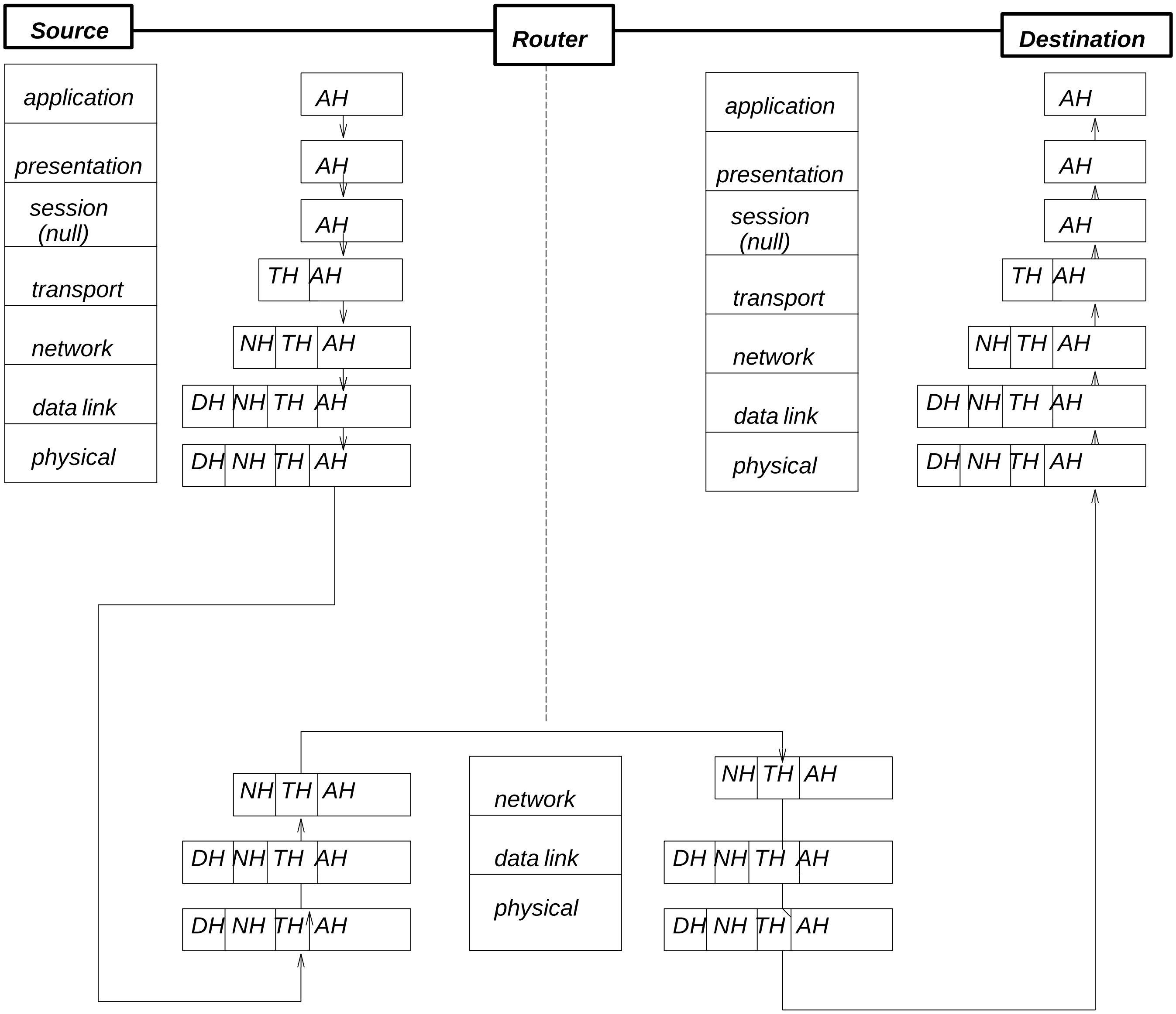
THE OSI MODEL: DATED BUT CONCEPTUALLY USEFUL

Layering concepts

- *Protocol:* Rules governing *horizontal* communication between peer layer entities.
- *Interface:* Rules governing *vertical* communication between a Layer N entity and a Layer N+1 entity on the same computer.
- *PDU:* Protocol Data Units are the messages that are exchanged between peer entities. N-PDU between Layer N-entities. In Internet, TPDU = Segment, NPDU = Packet, DPDU = Frame.
- *SDU:* Data Unit passed across an interface. N-SDU passed to and from layer N from Layer N+1.
- *PDU versus SDU:* Normally, N-PDU is N-SDU together with a Layer N header. However, one SDU can be split into multiple PDUs if the protocol allows only small PDUs.

Watch those headers!

- Communication between layer entities shares physical medium by using a layer header for each layer in each message. Think of data in envelope with transport header, stuffed in envelope with routing header, stuffed in enevelope with DL header.
- Sharing headers saves postage and also trivially coordinates headers with corresponding data (compared to out-of-band transmission between layers).
- Strict Layering: Each layer only looks at its header and interface data to do its job. Software engineering: changes to one layer do not cause other layers to be reimplemented. Information can be passed between layers via interface.
- As data moves down the layers, each layer adds its header. As data moves up, each layer strips off its header.



HEADERS: WATCH HOW THEY ARE ADDED AND REMOVED. STRICT LAYERING – THE DEFINITION

LAYERING IS MODULAR DESIGN

Sub-divide the problem
- Each layer relies on services from layer below
- Each layer exports services to layer above

Interface between layers defines interaction
- Hides implementation details (encapsulation)
- Layers can change without disturbing other layers (modularity)

Interface among peers in a layer is a protocol
- If peers speak same protocol, they can interoperate

KEY DESIGN DECISION IN IP

How do you divide functionality across the layers?

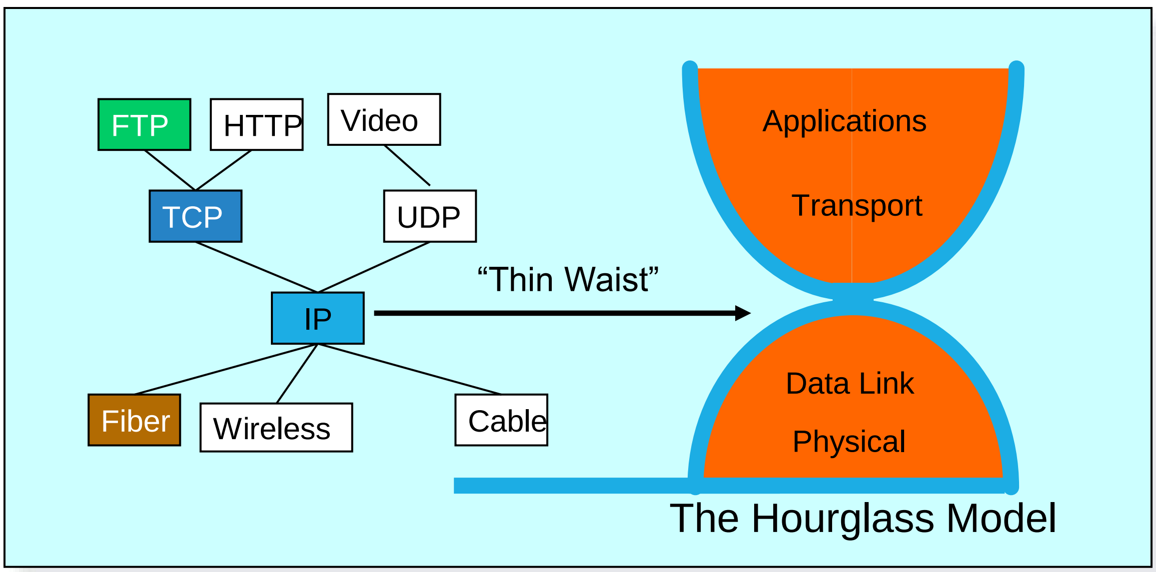
End-to-end argument [Saltzer84]

- Functionality should be implemented at a lower layer iff it can be **correctly** and **completely** implemented there. Incomplete versions of a function can be used as a performance enhancement, but not for correctness

Early, and still relevant, example

- ARPAnet provided reliable link transfers
- Was this enough for reliable communication?
- No, packets could still get corrupted on host-switch link, or inside of the switches
- Hence, still need reliability at higher layers (TCP)

TCP SUCCESSFUL > 40 years



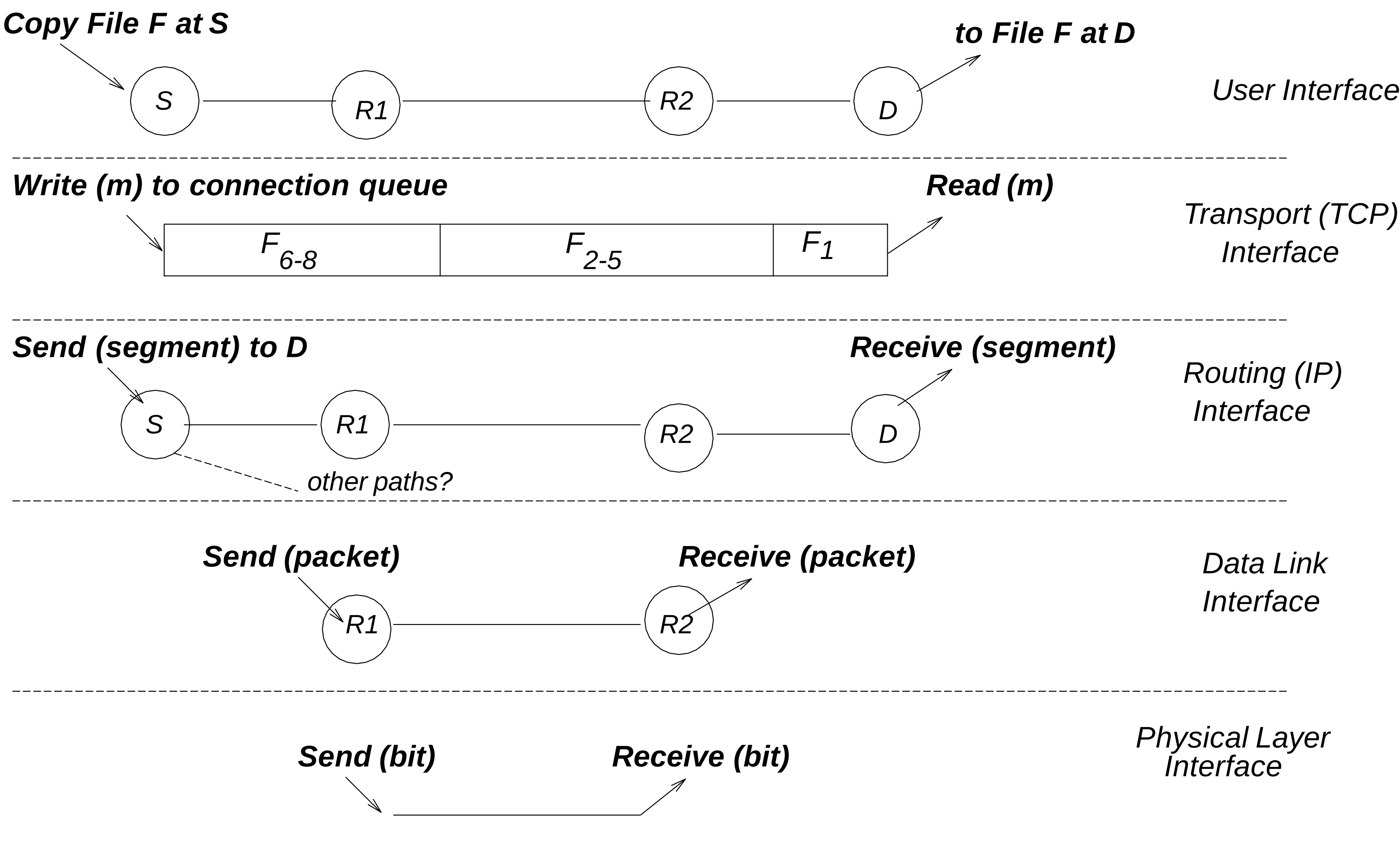
Q: Why was TCP/IP successful?
A: IP addressing (Lecture 12)
Socket interface (Lecture 17)

FEYNMAN ON PHYSICS

We have a way of discussing the world, when we talk of it at various hierarchies, or levels . . . I will indicate, by describing a set of ideas, what I mean by hierarchies of ideas.

The great mass of workers in between, connecting one step to another, are improving all the time our understanding of the world, . . . and in that way we are gradually understanding this tremendous world of interconnecting hierarchies.

Which End Is Closer To God?
Richard Feynman



THE IP ABSTRACTIONS: Each layer provides a service to the layer above it

From Files to Voltage Levels

- File Transfer implemented by two FTP processes on each machine. Shared queue is simplest asynchronous interface, which is what TCP provides.
- TCP implements shared queue abstraction by sending numbered segments, retransmitting if acks are not received. Requires being able to send segments to arbitrary destinations, which is what IP provides.
- IP computes routes (routing) and then forwards packet hop-by-hop. At each hop, IP requires sending a frame to directly connected neighbor, which is what Data Link provides.
- Data Link uses physical layer to send each bit of a frame; then puts together bits at receiver to form a frame and does error checks. Physical layer sends bits by transforming 0s and 1s into physical energy that can travel distance.

So why a 20 lecture class?

All layers have common problems: synchronization in the face of errors and asynchrony, addressing, multiplexing, interconnection. Sample problems you will learn the answer to:

- *Transport:*
  - Congestion Control. How does a TCP sender know how to speed up or slow down depending on current Internet speed? *Slow-start.*
  - Connection Management: How does TCP prevent old conversations between the same pair of machines from mixing in with new conversations. *3-way handshakes*
- *Routing:*
  - CIDR: How does IP allow various sizes of networks in allocating addresses.
  - BGP: How does IP calculate routes between multiple competing providers?

- *Data Link:*
  - Min Packet Sizes: How does Ethernet ensure that if one nodes detects a collision, all nodes do?
  - Dynamic Backoff: How can Ethernet sort out 2 sender collisions quickly while being able to sort out even 32 sender collisions?
- *Physical Layer:*
  - Clock Recovery: How does a receiver reconstruct bits from physical signals despite speed differences?
  - Media Issues: When should a manager use wireless versus fiber versus satellite?