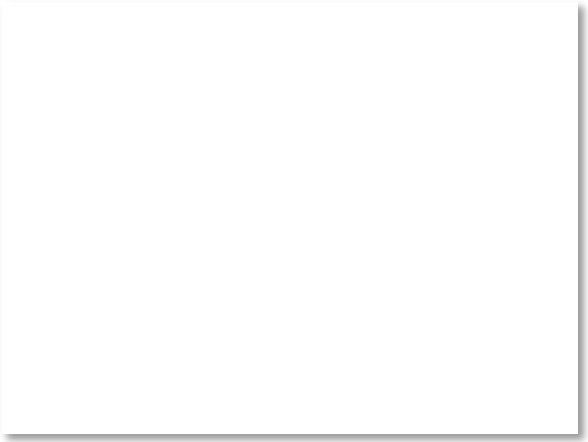
Networks Need Modularity

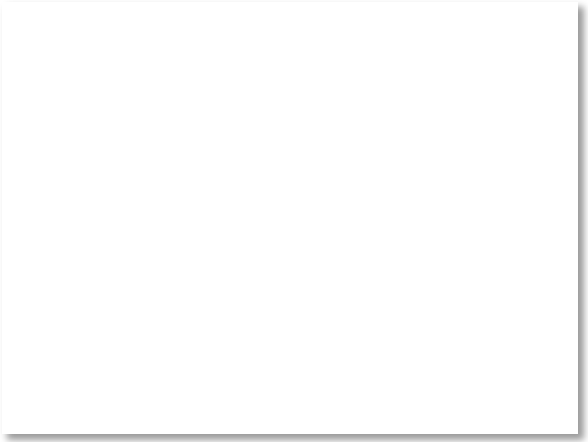
- The network does much for apps:
  - Make and break connections
  - Find a path through the network
  - Transfers information reliably
  - Transfers arbitrary length information
  - Send as fast as the network allows
  - Shares bandwidth among users
  - Secures information in transit
  - Lets many new hosts be added
  - …



Networks Need Modularity

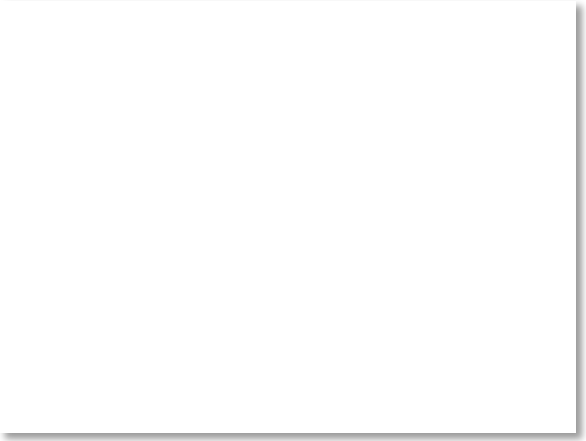
- The network does much for apps:
  - Make and break connections
  - ork
  - mation
  - ows
  - s
  - Secures information in transit
  - Lets many new hosts be added
  - …

We need a form of modularity, to help manage complexity and support reuse



Protocols and Layers

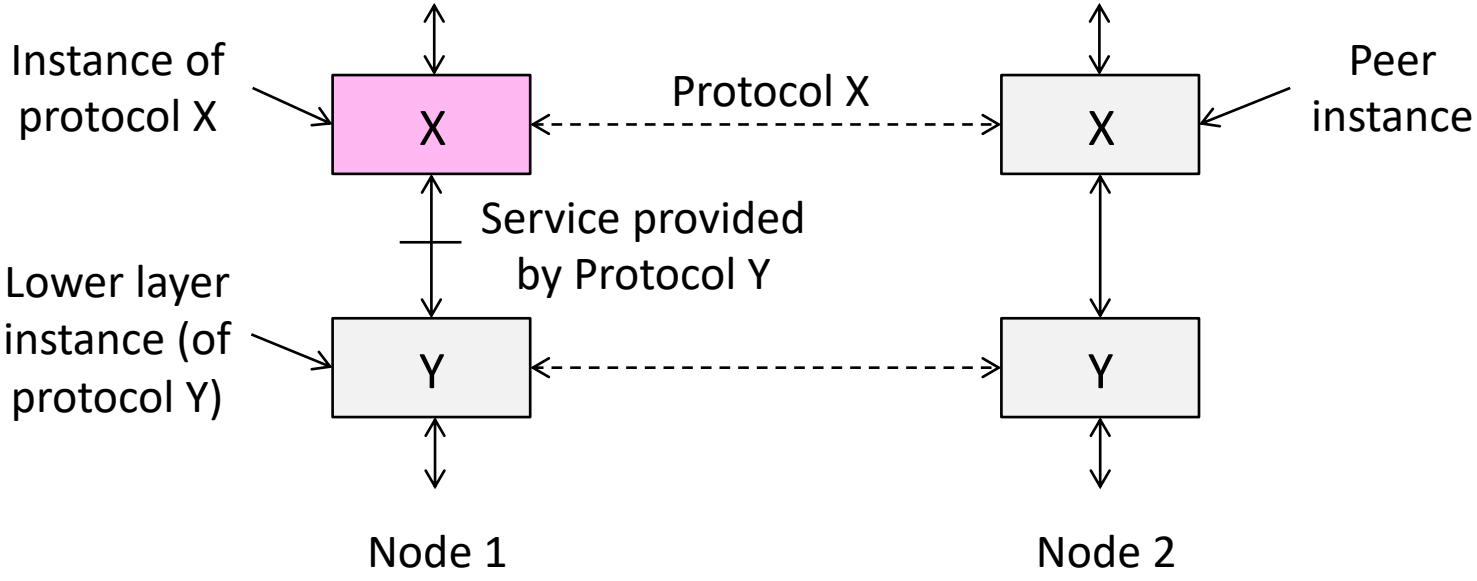
- Protocols and layering is the main structuring method used to divide up network functionality
  - Each instance of a protocol talks virtually to its peer using the protocol
  - Each instance of a protocol uses only the services of the lower layer



Protocols and Layers (2)

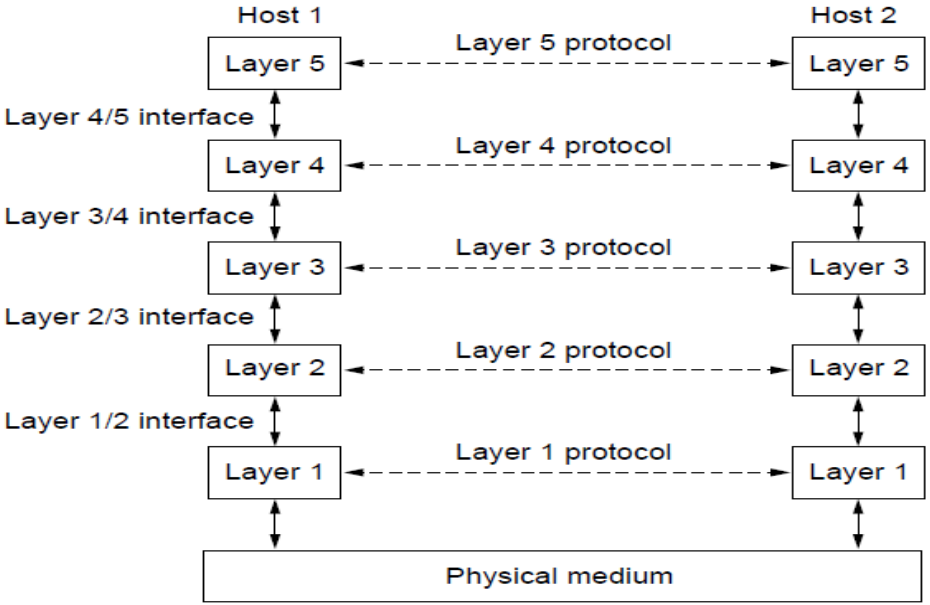
Protocols and Layers (3)

- Protocols are horizontal, layers are vertical



Protocols and Layers (4)

- Set of protocols in use is called a protocol stack

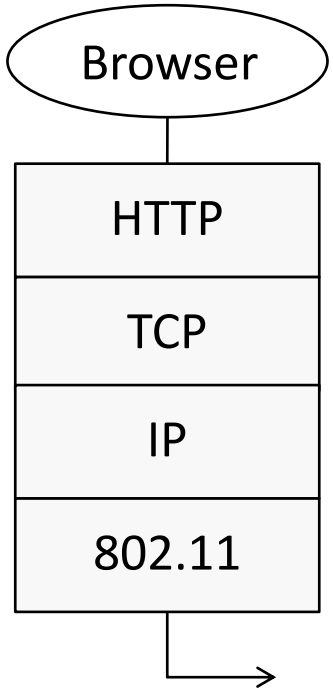


Protocols and Layers (5)

- Protocols you've probably heard of:
  - TCP, IP, 802.11, Ethernet, HTTP, SSL, DNS, … and many more
- An example protocol stack
  - Used by a web browser on a host that is wirelessly connected to the Internet

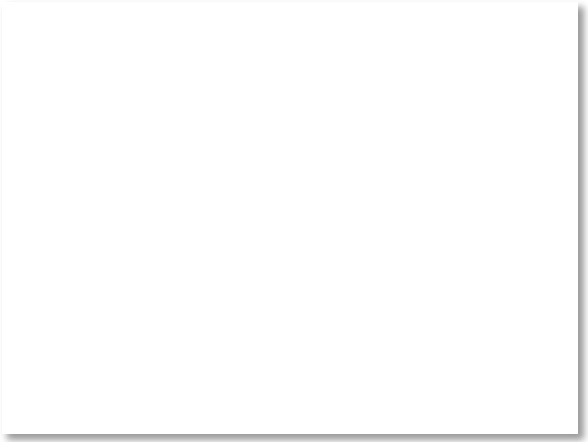
Protocols and Layers (6)

- Protocols you've probably heard of:
  - TCP, IP, 802.11, Ethernet, HTTP, SSL, DNS, … and many more
- An example protocol stack
  - Used by a web browser on a host that is wirelessly connected to the Internet

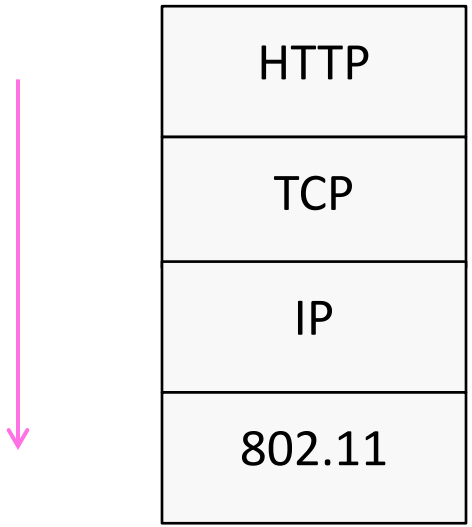


Encapsulation

- Encapsulation is the mechanism used to effect protocol layering
  - Lower layer wraps higher layer content, adding its own information to make a new message for delivery
  - Like sending a letter in an envelope; postal service doesn't look inside

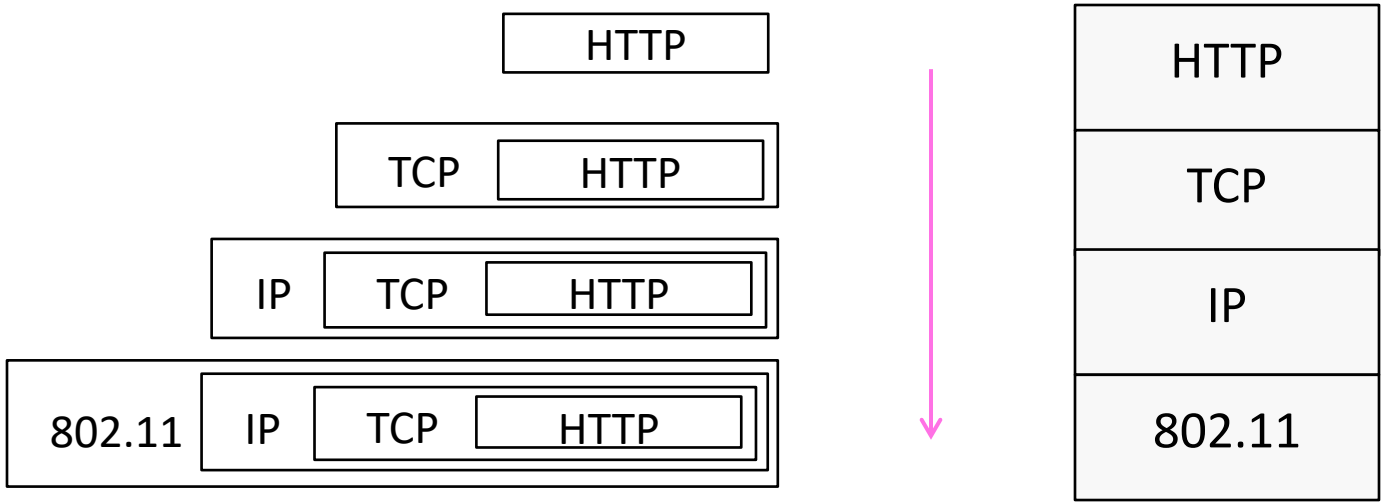


Encapsulation (2)

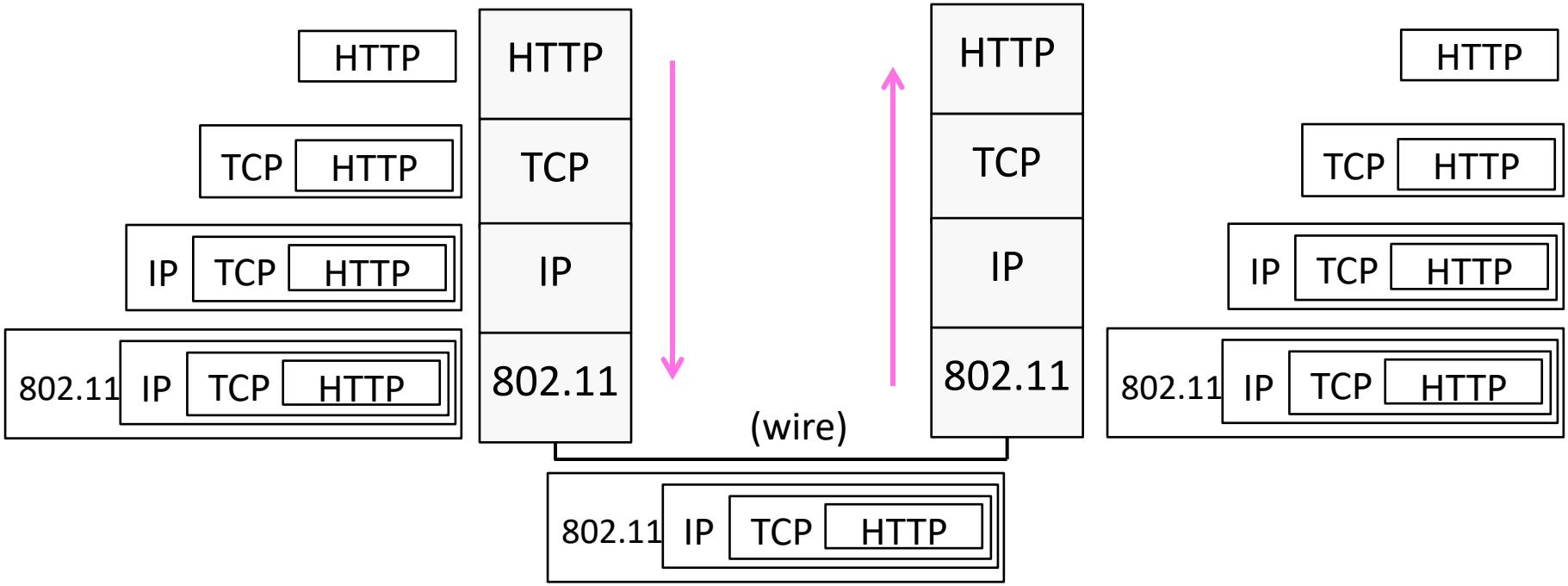


Encapsulation (3)

- Message “on the wire” begins to look like an onion
  - Lower layers are outermost

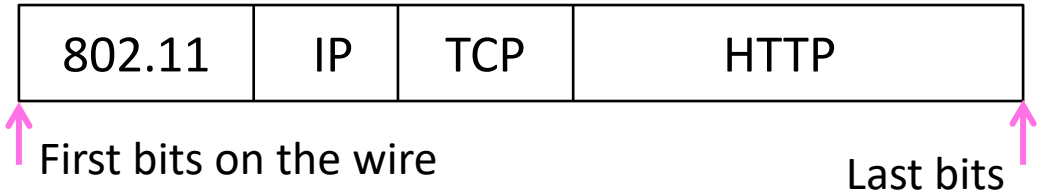


Encapsulation (4)



Encapsulation (5)

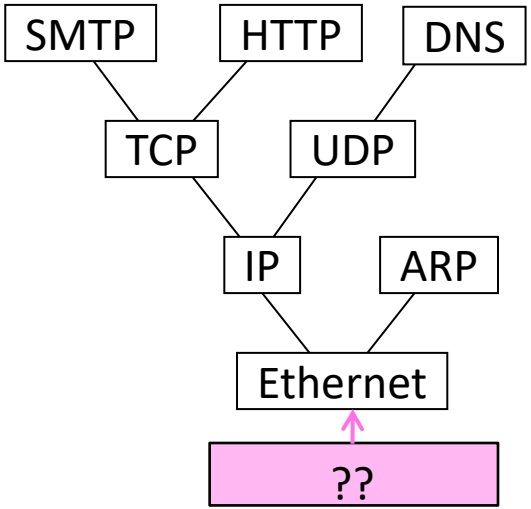
- Normally draw message like this:
  - Each layer adds its own header



- More involved in practice
  - Trailers as well as headers, encrypt/compress contents
  - Segmentation (divide long message) and reassembly

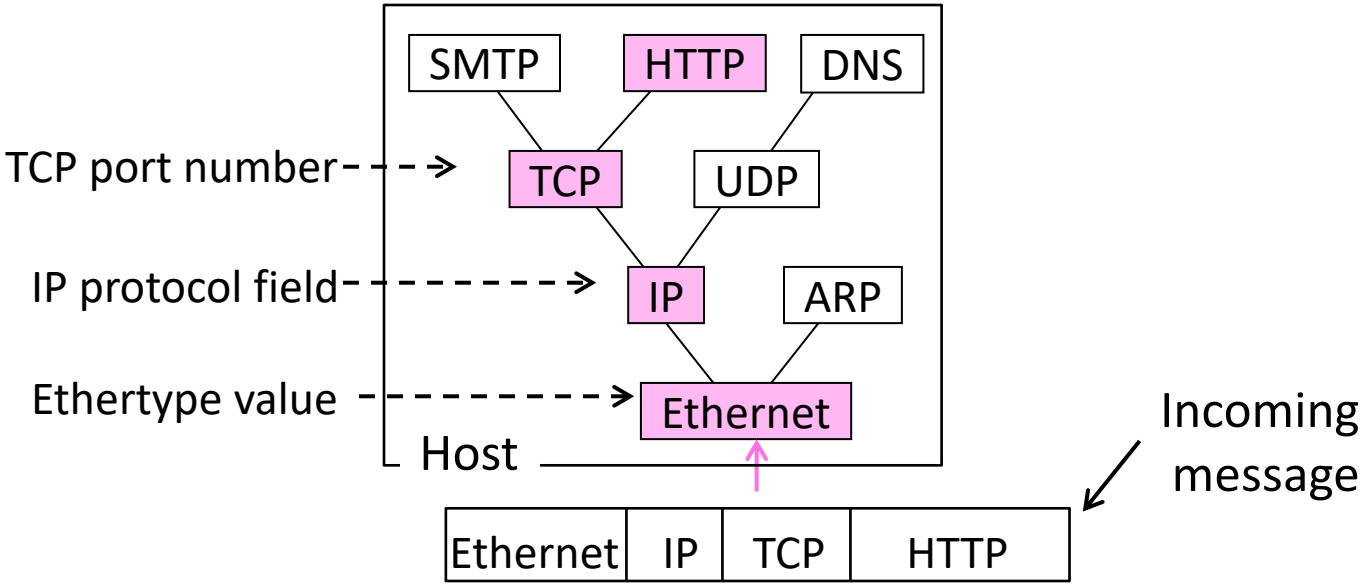
Demultiplexing

- Incoming message must be passed to the protocols that it uses



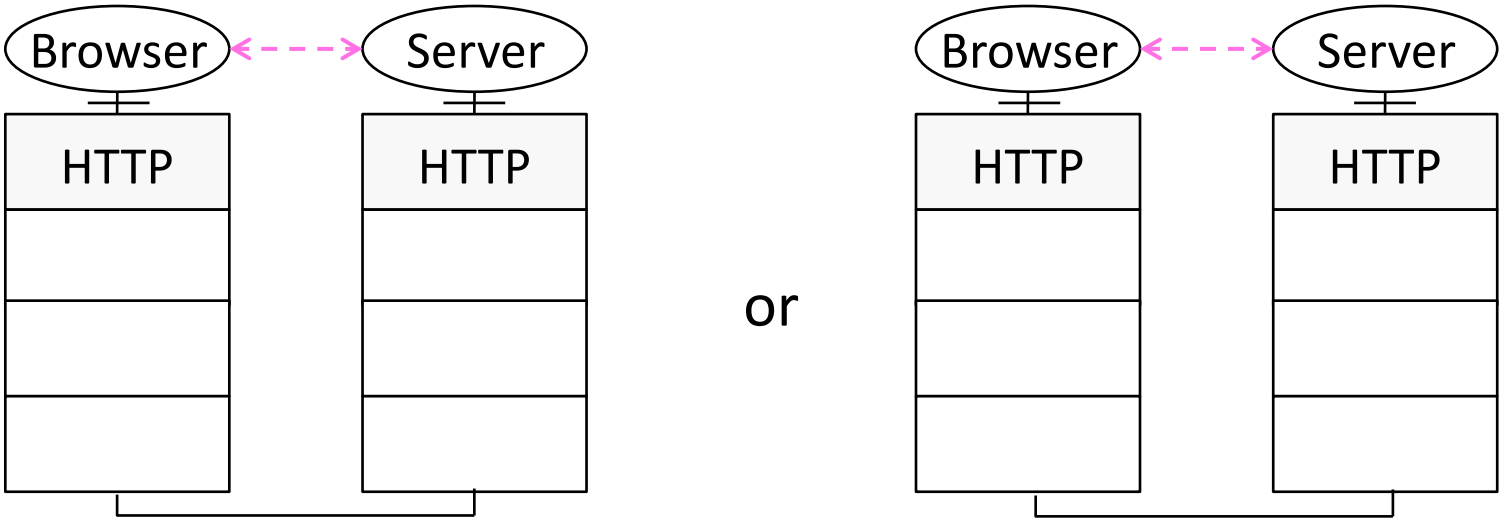
Demultiplexing (2)

- Done with demultiplexing keys in the headers



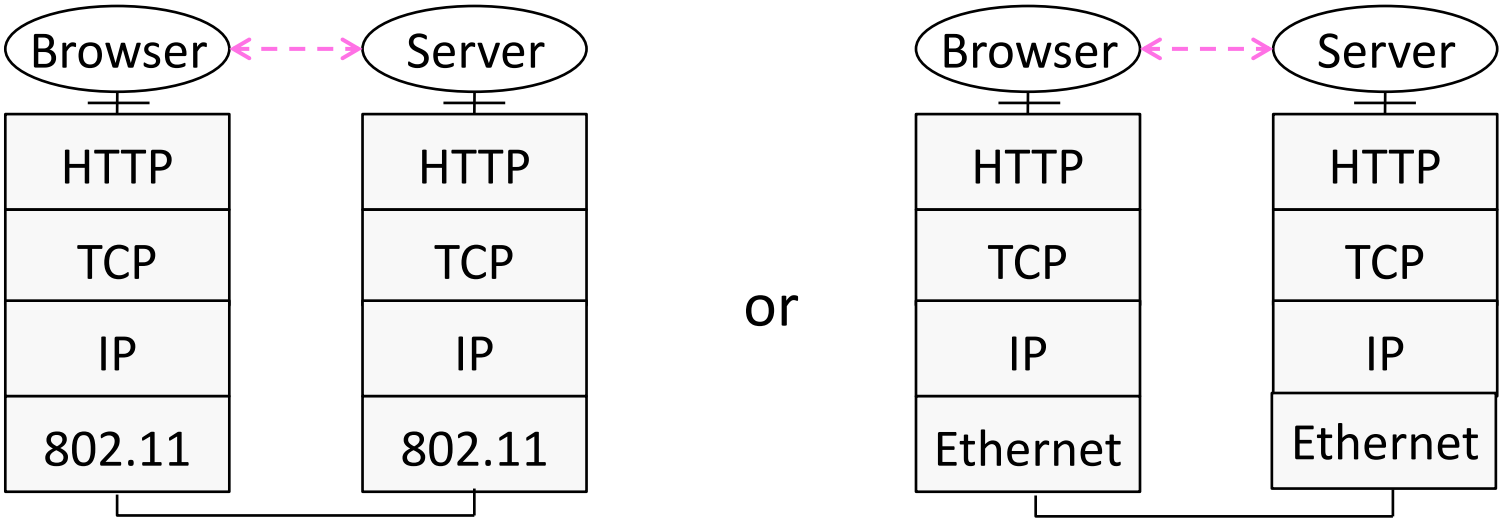
Advantage of Layering

- Information hiding and reuse



Advantage of Layering (2)

- Information hiding and reuse

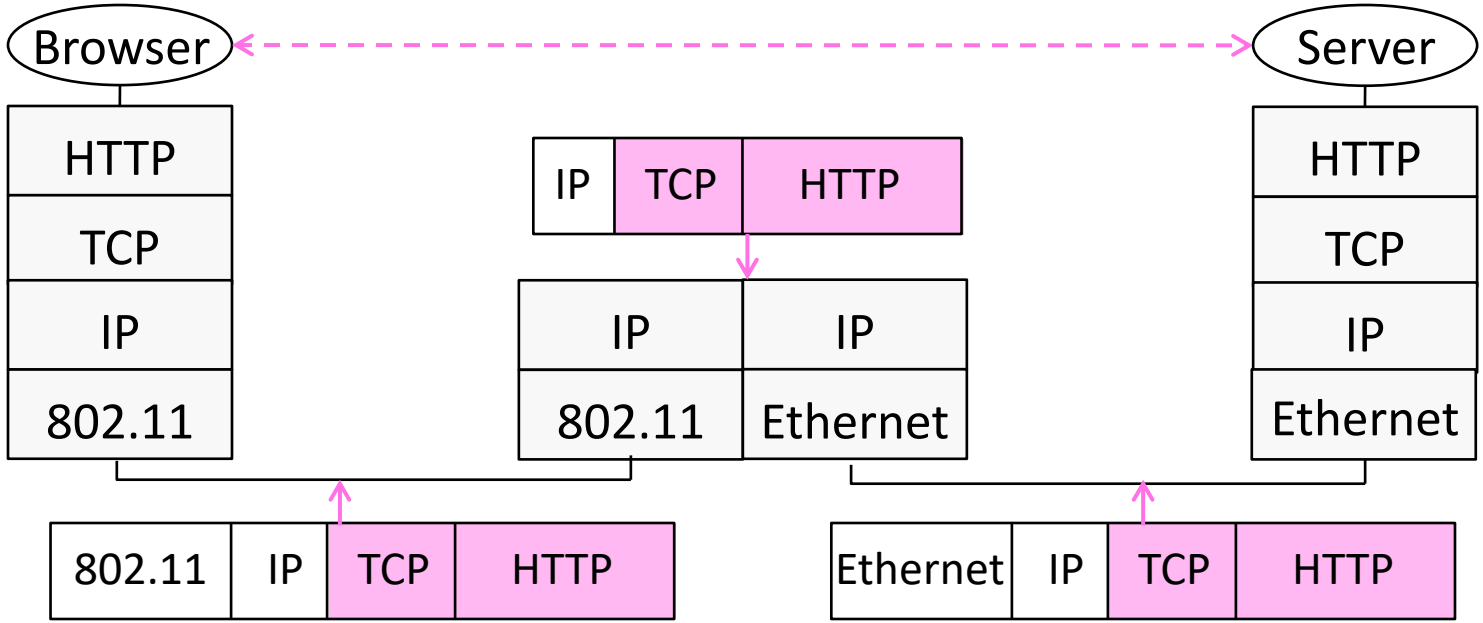


Advantage of Layering (3)

- Using information hiding to connect different systems

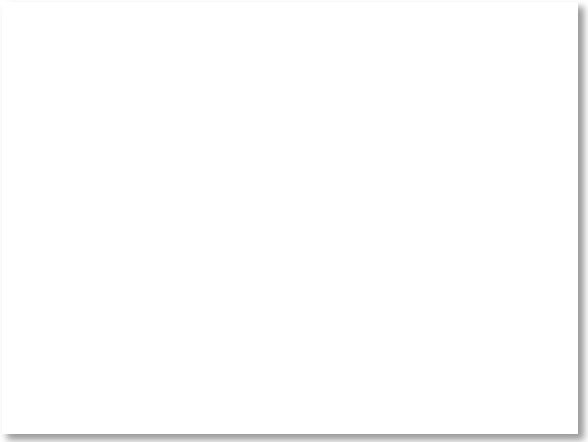
Advantage of Layering (4)

- Using information hiding to connect different systems



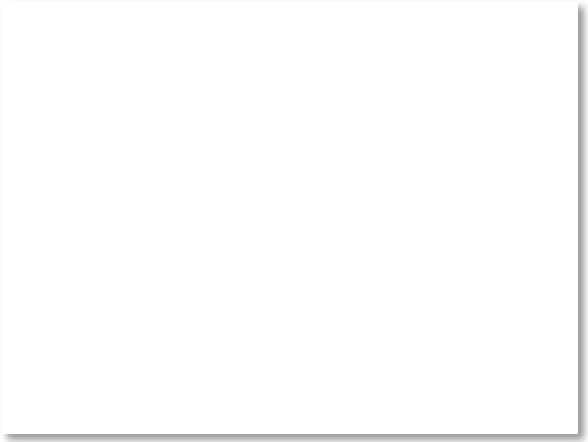
Disadvantage of Layering

- Adds overhead
  - But minor for long messages
- Hides information
  - App might care whether it is running over wired or wireless!



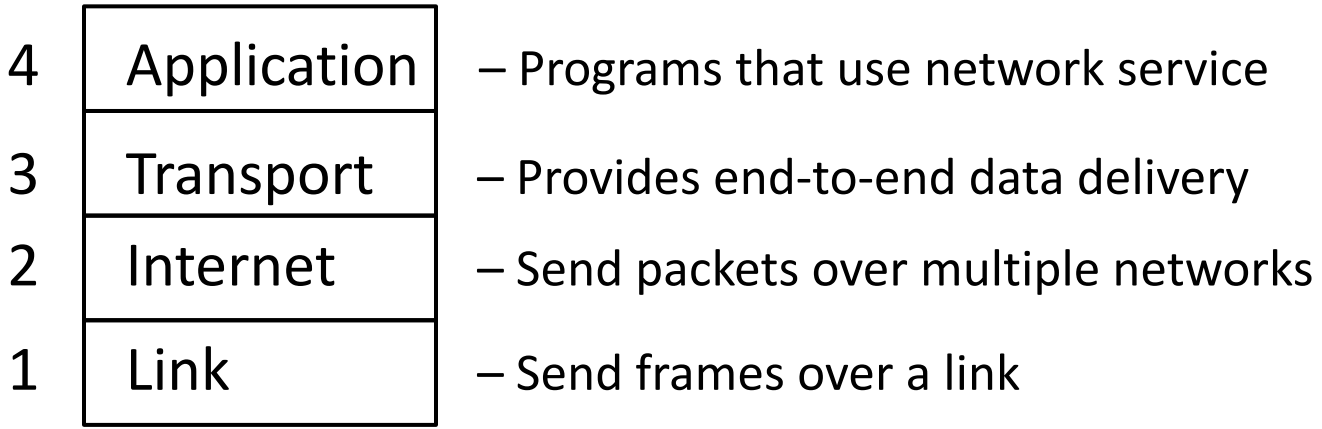
A Little Guidance Please …

- What functionality should we implement at which layer?
  - This is a key design question
  - Reference models provide frameworks that guide us »



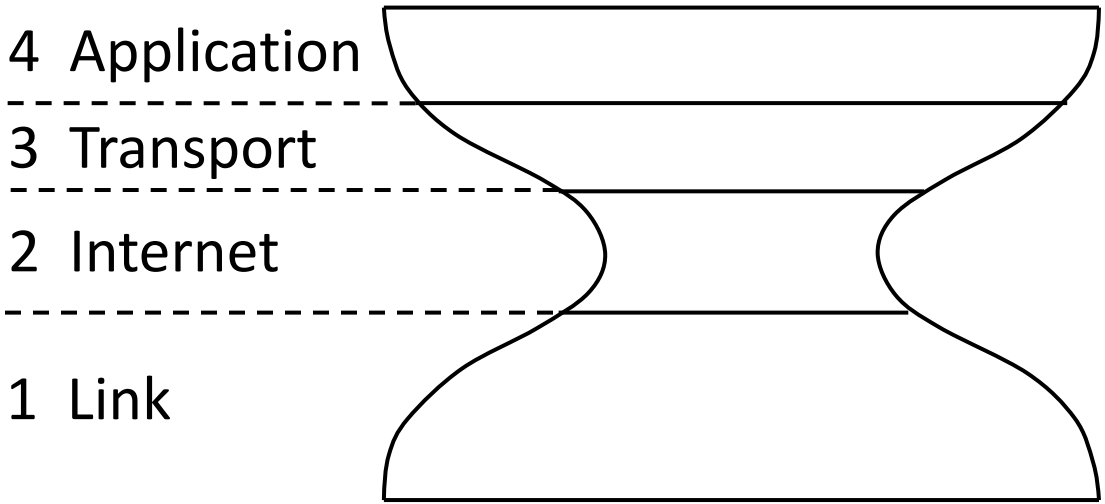
Internet Reference Model

- A four layer model based on experience; omits some OSI layers and uses IP as the network layer.



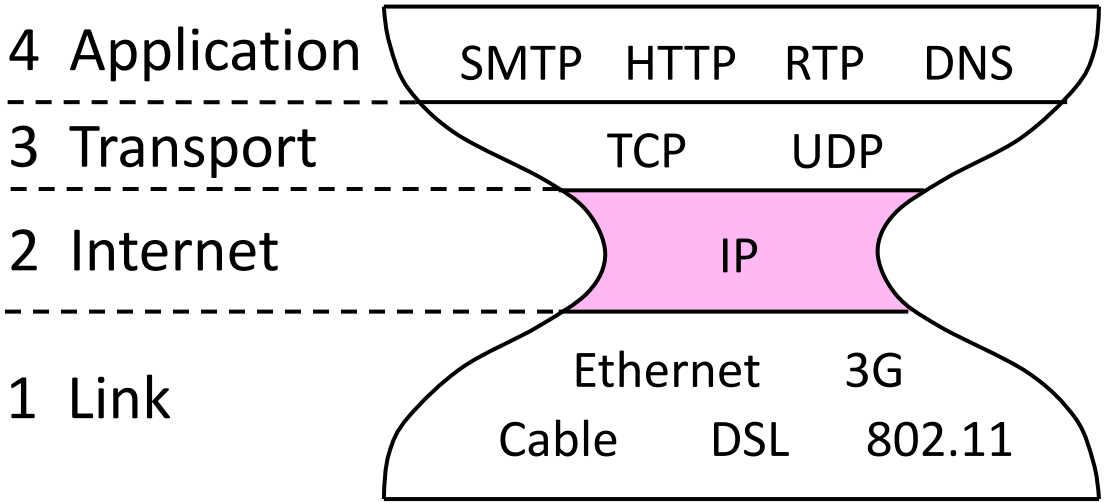
Internet Reference Model (2)

- With examples of common protocols in each layer



Internet Reference Model (3)

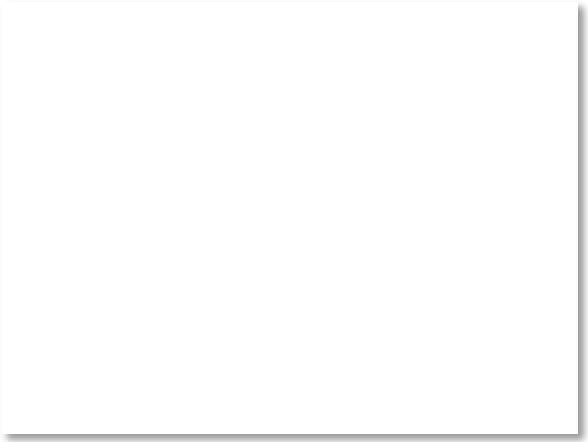
- IP is the “narrow waist” of the Internet
  - Supports many different links below and apps above



Standards Bodies

- Where all the protocols come from!
  - Focus is on interoperability

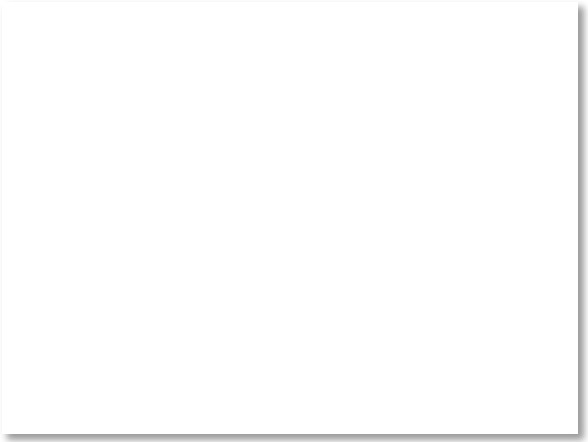
Body	Area	Examples
ITU	Telecom	G.992, ADSL H.264, MPEG4
IEEE	Communications	802.3, Ethernet 802.11, WiFi
IETF	Internet	RFC 2616, HTTP/1.1 RFC 1034/1035, DNS
W3C	Web	HTML5 standard CSS standard



Layer-based Names

- For units of data:

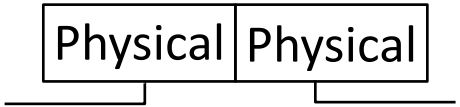
Layer	Unit of Data
Application	Message
Transport	Segment
Network	Packet
Link	Frame
Physical	Bit



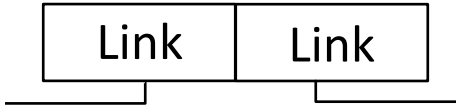
Layer-based Names (2)

- For devices in the network:

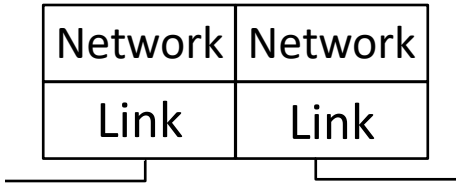
Repeater (or hub)



Switch (or bridge)



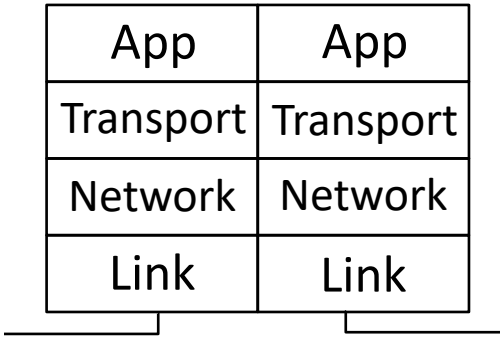
Router



Layer-based Names (3)

- For devices in the network:

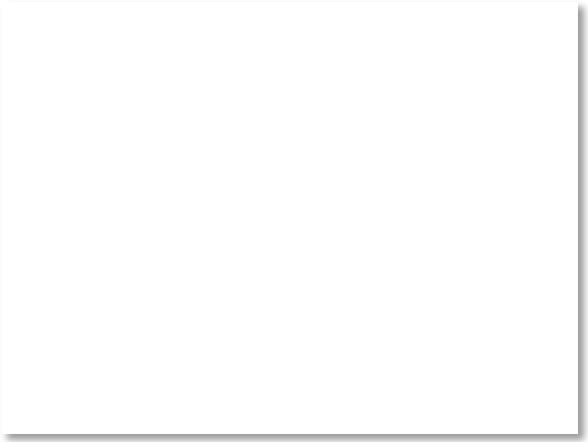
Proxy or middlebox or gateway



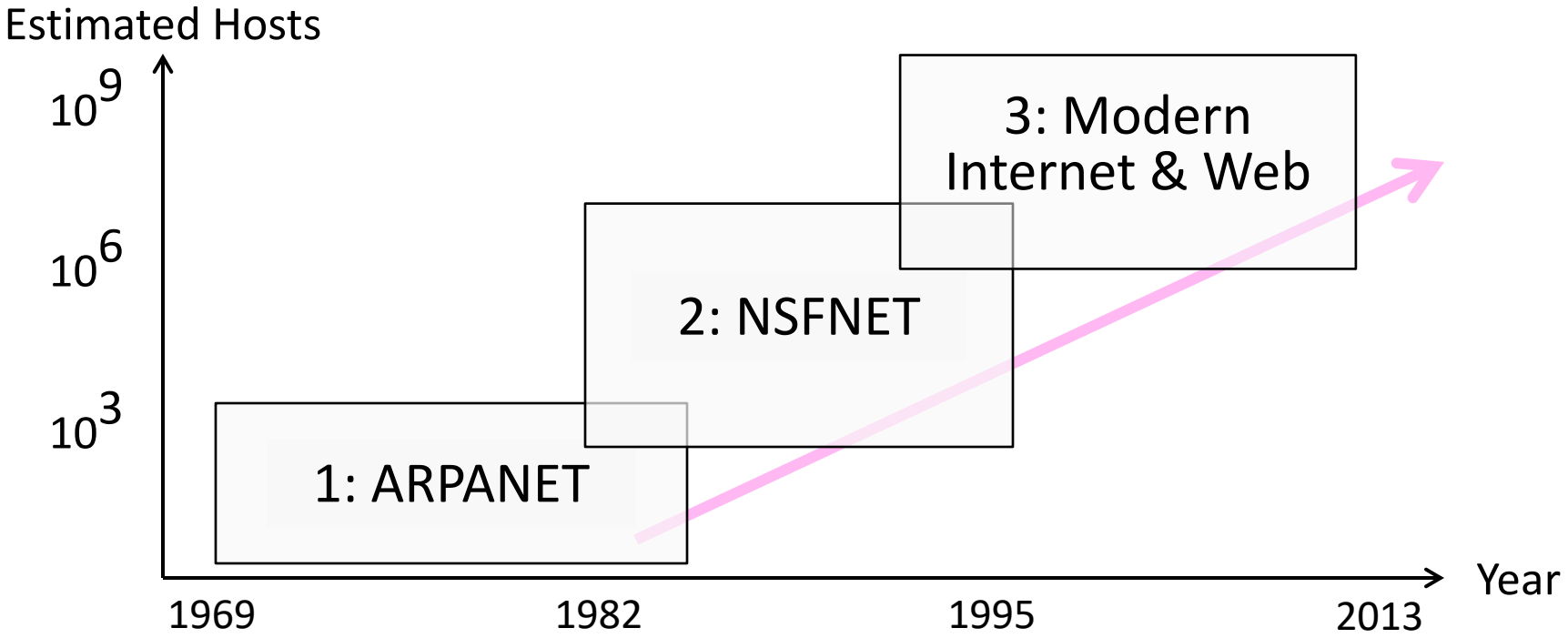
But they all look like this!

A Note About Layers

- They are guidelines, not strict
  - May have multiple protocols working together in one layer
  - May be difficult to assign a specific protocol to a layer

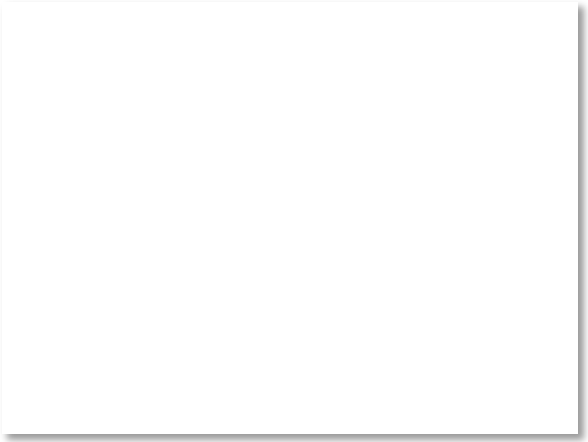


Rough Internet Timeline



The Beginning – ARPANET

- ARPANET by U.S. DoD was the precursor to the Internet
  - Motivated for resource sharing
  - Launched with 4 nodes in 1969, grew to hundreds of hosts
  - First “killer app” was email



ARPANET – Influences

- Leading up to the ARPANET (1960s):
  - Packet switching (Kleinrock, Davies), decentralized control (Baran)

Paul Baran

Credit: Internet Hall of Fame

Donald Davies

Credit: Internet Hall of Fame

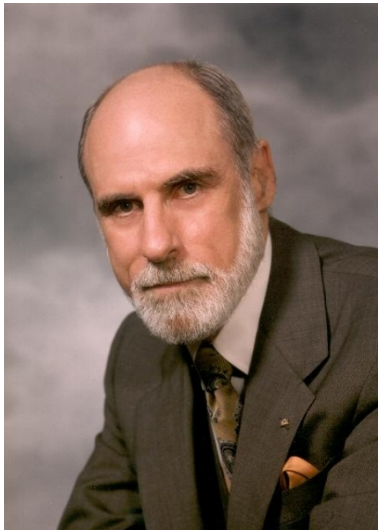
Len Kleinrock

Credit: Internet Hall of Fame

ARPANET – Influences (2)

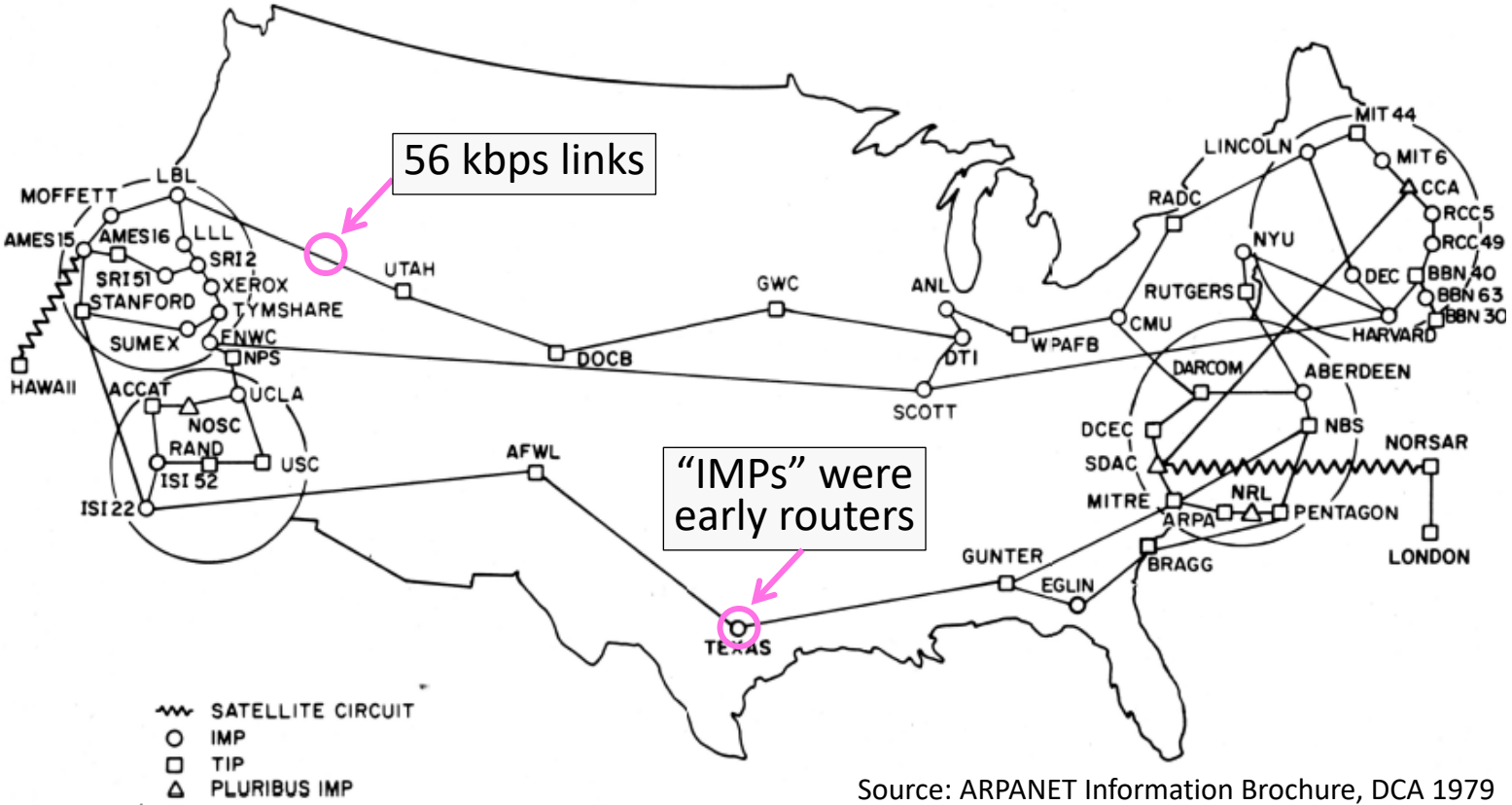
- In the early ARPANET
  - Internetworking became the basis for the Internet
  - Pioneered by Cerf & Kahn in 1974, later became TCP/IP
  - They are popularly known as the “fathers of the Internet”

Vint Cerf



Bob Kahn

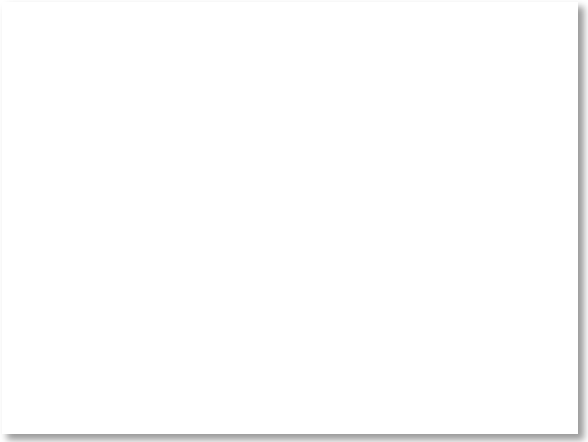
ARPANET Geographical Map (Dec. 1978)



Source: ARPANET Information Brochure, DCA 1979

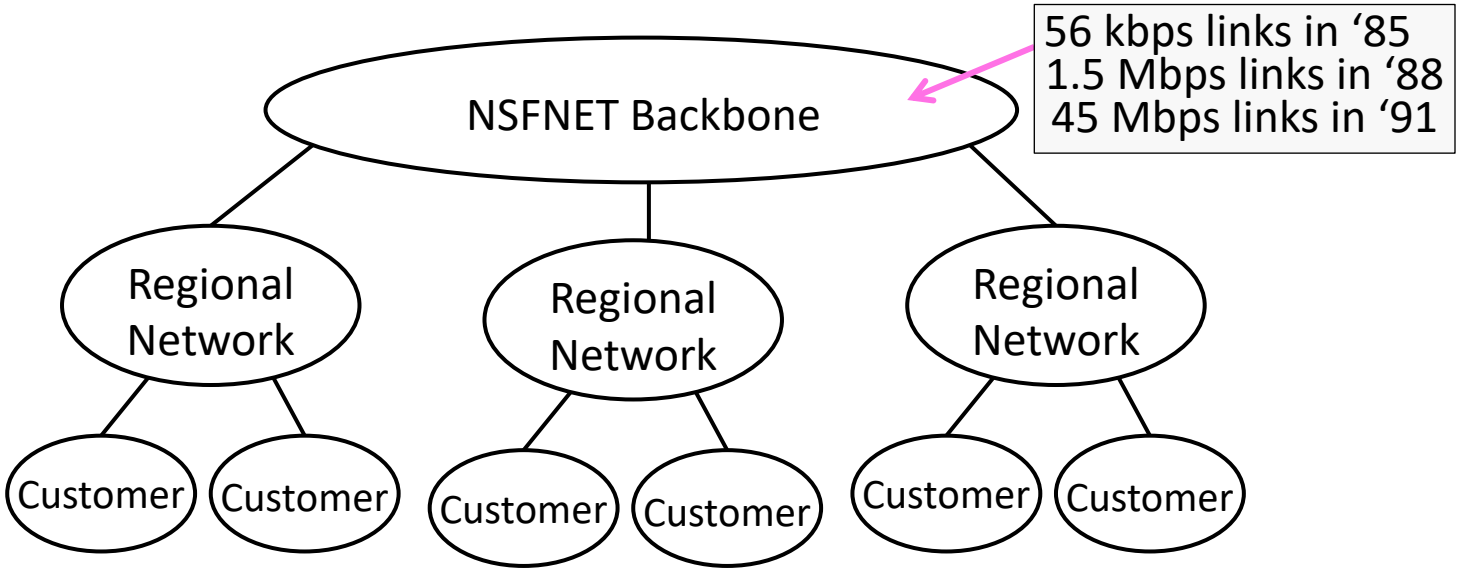
Growing Up – NSFNET

- NSFNET ’85 supports educational networks
  - Initially connected supercomputer sites, but soon became the backbone for all networks
- Classic Internet protocols we use emerged
  - TCP/IP (transport), DNS (naming), Berkeley sockets (API) in ’83, BGP (routing) in ’93
- Much growth from PCs and Ethernet LANs
  - Campuses, businesses, then homes
  - 1 million hosts by 1993 …



Early Internet Architecture

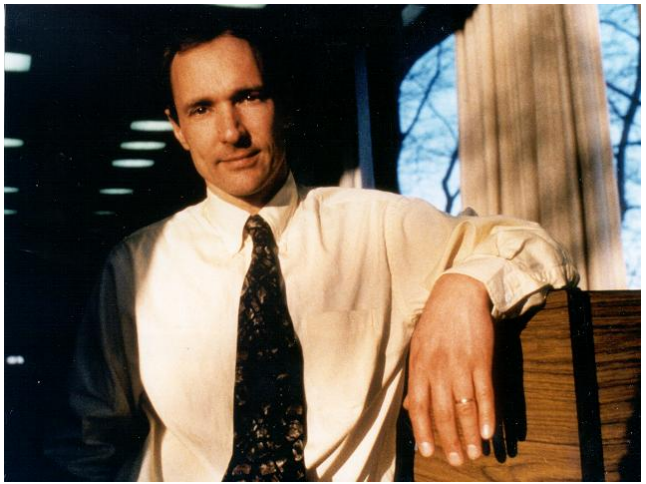
- Hierarchical, with NSFNET as the backbone



Modern Internet – Birth of the Web

- After ’95, connectivity is provided by large ISPs who are competitors
  - They connect at Internet eXchange Point (IXP) facilities
  - Later, large content providers connect
- Web bursts on the scene in ’93
  - Growth leads to CDNs, ICANN in ‘98
  - Most bits are video (soon wireless)
  - Content is driving the Internet

Tim Berners-Lee



Modern Internet Architecture

- Complex business arrangements affect connectivity
  - Still decentralized, other than registering identifiers