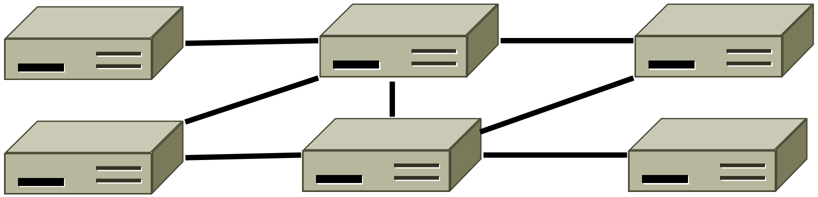
# Where we are in the Course

- Starting the Network Layer!
  - Builds on the link layer. Routers send packets over multiple networks

| |
|---|
| Application |
| Transport |
| Network |
| Link |
| Physical |

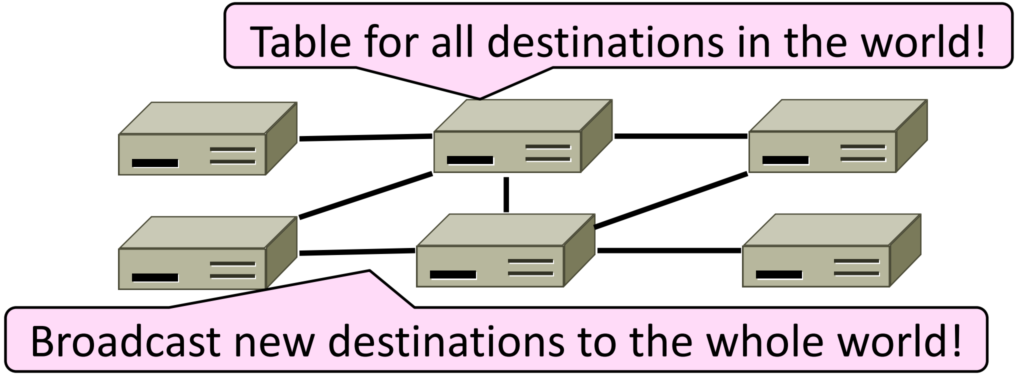
# Why do we need a Network layer?

- We can already build networks with links and switches and send frames between hosts …



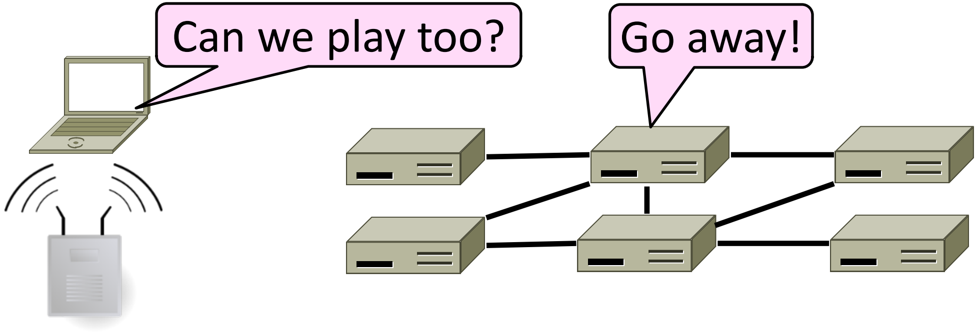
# Shortcomings of Switches

1. Don’t scale to large networks
   - Blow up of routing table, broadcast



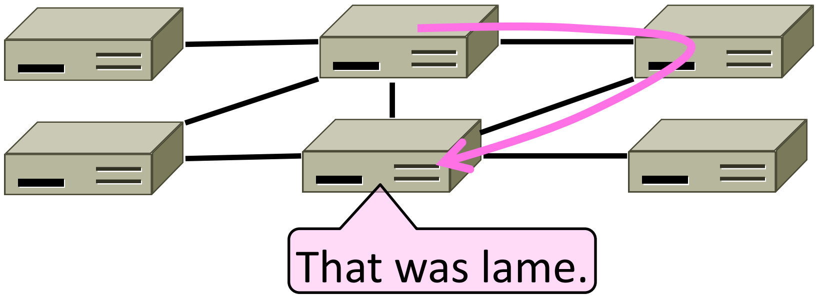
# Shortcomings of Switches (2)

2. Don’t work across more than one link layer technology
   – Hosts on Ethernet + 3G + 802.11 …



# Shortcomings of Switches (3)

3. Don’t give much traffic control
   – Want to plan routes / bandwidth



# Network Layer Approach

- Scaling:
  - Hierarchy, in the form of prefixes
- Heterogeneity:
  - IP for internetworking
- Bandwidth Control:
  - Lowest-cost routing
  - Later QOS (Quality of Service)

# Topics

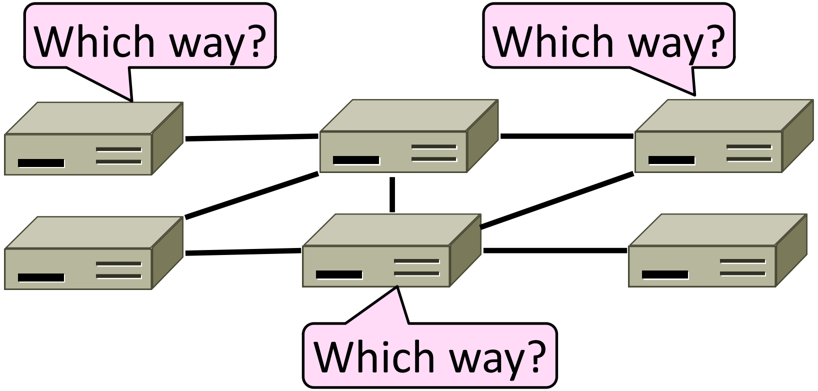
- Network service models
  - Datagrams (packets), virtual circuits
- IP (Internet Protocol)
  - Internetworking
  - Forwarding (Longest Matching Prefix)
  - Helpers: ARP and DHCP
  - Fragmentation and MTU discovery
  - Errors: ICMP (traceroute!)
- IPv6, the future of IP
- NAT, a “middlebox”
- Routing algorithms

This time

Next time

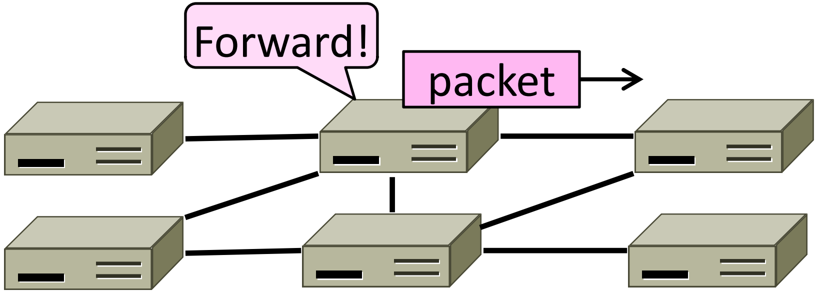
# Routing vs. Forwarding

- Routing is the process of deciding in which direction to send traffic
  - Network wide (global) and expensive



# Routing vs. Forwarding (2)

- <u>Forwarding</u> is the process of sending a packet on its way
  - Node process (local) and fast

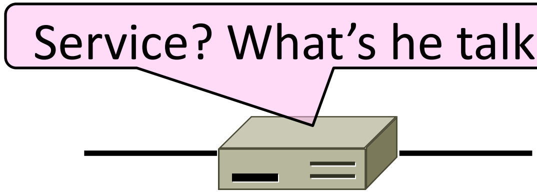


# Our Plan

- Forwarding this time
  - What routers do with packets
- Routing next time
  - Logically this comes first
  - But ignore it for now

# Topic

- What kind of service does the Network layer provide to the Transport layer?
  - How is it implemented at routers?



# Two Network Service Models

- Datagrams, or connectionless service
  - Like postal letters
  - (This one is IP)

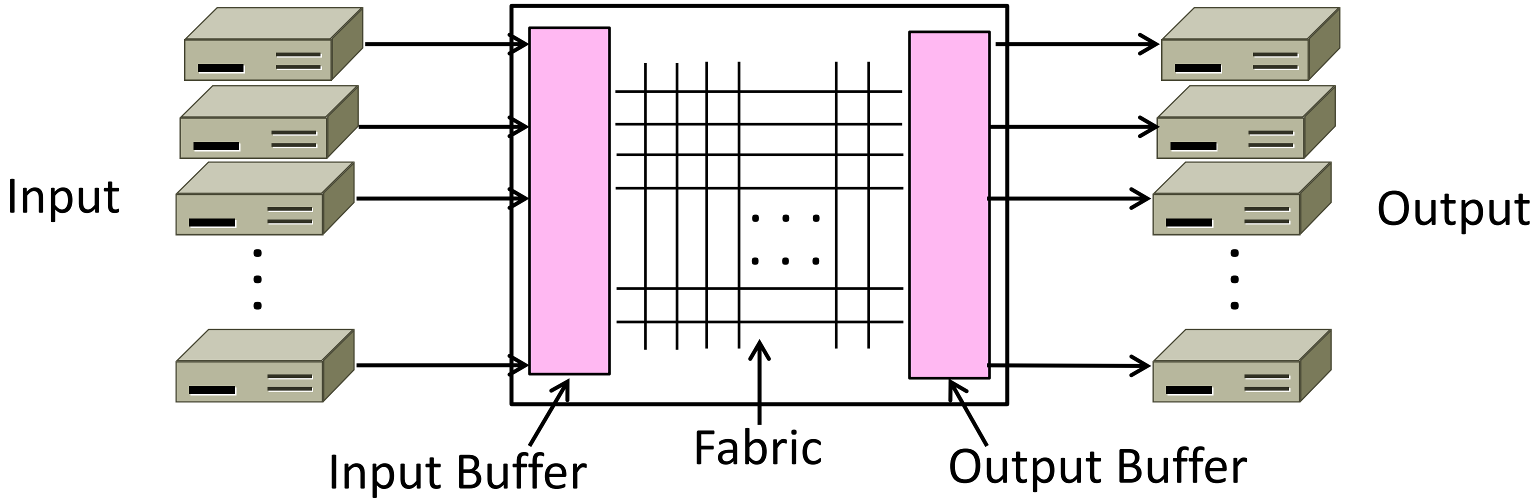
- Virtual circuits, or connection-oriented service
  - Like a telephone call

# Store-and-Forward Packet Switching

- Both models are implemented with <u>store-and-forward packet switching</u>
  - Routers receive a complete packet, storing it temporarily if necessary before forwarding it onwards
  - We use statistical multiplexing to share link bandwidth over time

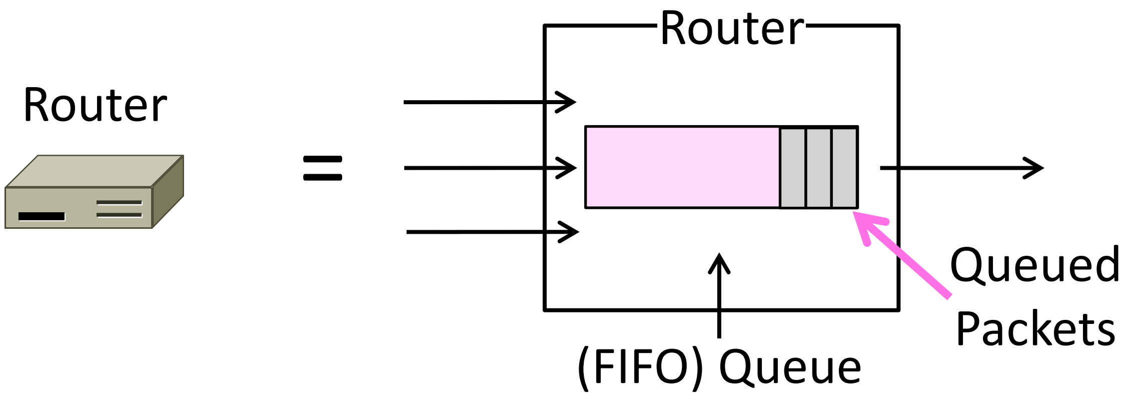
# Store-and-Forward (2)

- Switching element has internal buffering for contention



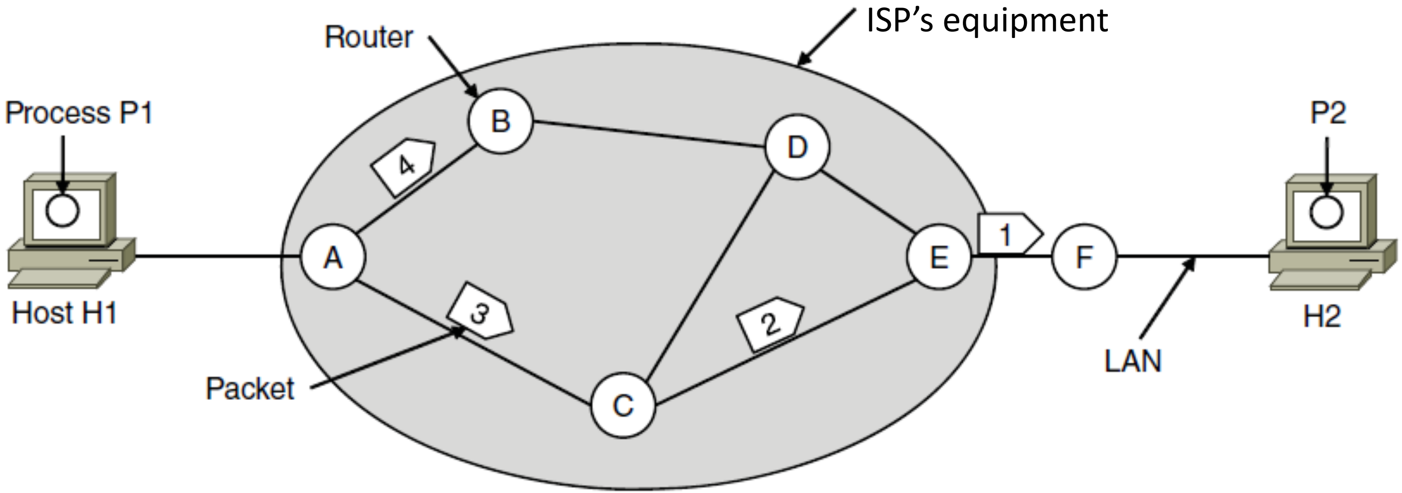
# Store-and-Forward (3)

- Simplified view with per port output buffering
  - Buffer is typically a FIFO (First In First Out) queue
  - If full, packets are discarded (congestion, later)



# Datagram Model

- Packets contain a destination address; each router uses it to forward each packet, possibly on different paths



# Datagram Model (2)

- Each router has a forwarding table keyed by address
  - Gives next hop for each destination address; may change

A's table (initially)

| Dest. | Line |
|---|---|
| A | |
| B | B |
| C | C |
| D | B |
| E | C |
| F | C |

A's table (later)

| | |
|---|---|
| A | |
| B | B |
| C | C |
| D | B |
| E | B |
| F | B |

C's Table

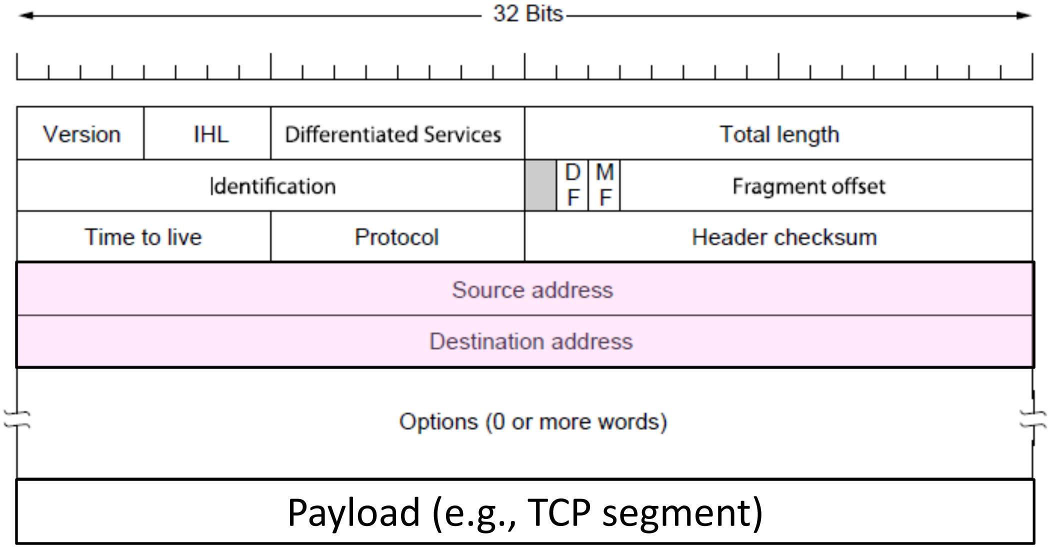
| | |
|---|---|
| A | A |
| B | A |
| C | |
| D | E |
| E | E |
| F | E |

E's Table

| | |
|---|---|
| A | C |
| B | D |
| C | C |
| D | D |
| E | |
| F | F |

# IP (Internet Protocol)

- Network layer of the Internet, uses datagrams (next)
  - IPv4 carries 32 bit addresses on each packet (often 1.5 KB)



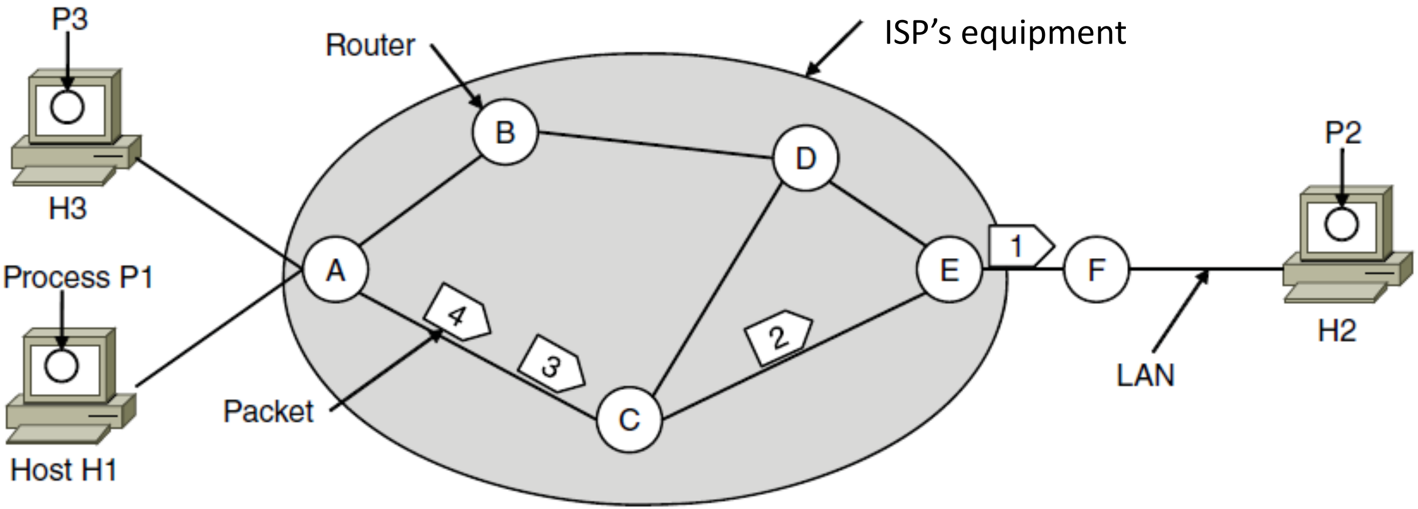
# Virtual Circuit Model

- Three phases:
  1. Connection establishment, circuit is set up
     - Path is chosen, circuit information stored in routers
  2. Data transfer, circuit is used
     - Packets are forwarded along the path
  3. Connection teardown, circuit is deleted
     - Circuit information is removed from routers

- Just like a telephone circuit, but virtual in the sense that no bandwidth need be reserved; statistical sharing of links

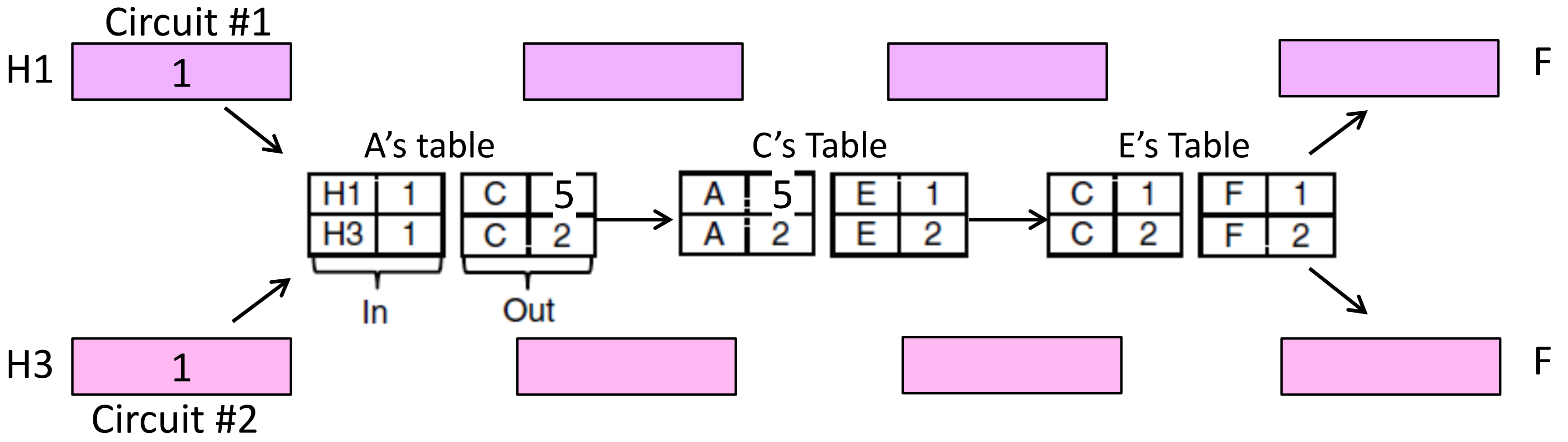
# Virtual Circuits (2)

- Packets only contain a short label to identify the circuit
  - Labels don't have any global meaning, only unique for a link



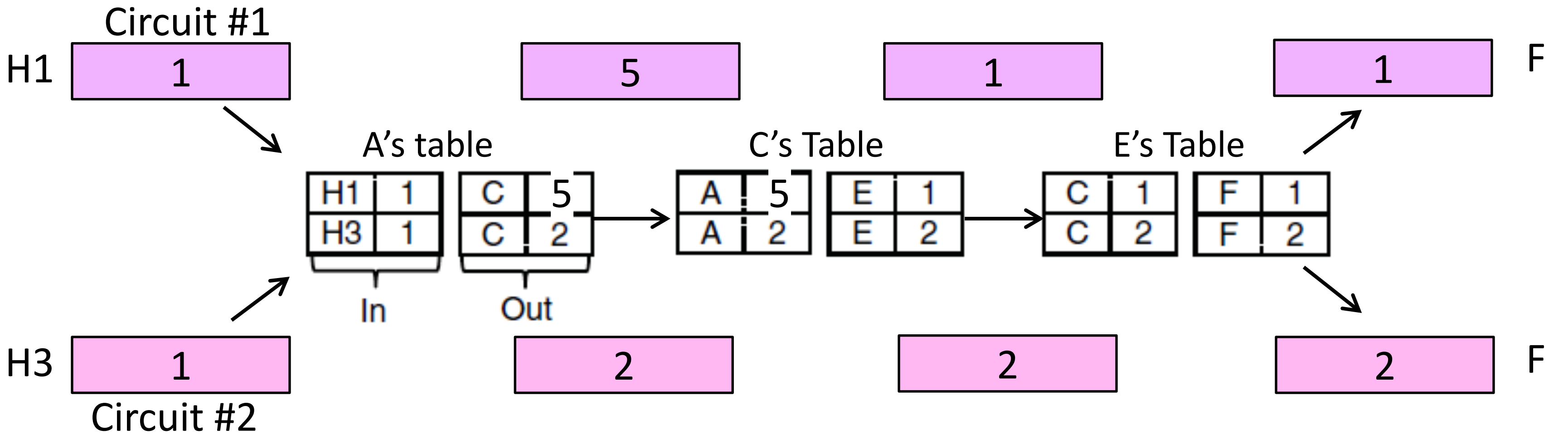
# Virtual Circuits (3)

- Each router has a forwarding table keyed by circuit
  - Gives output line and next label to place on packet



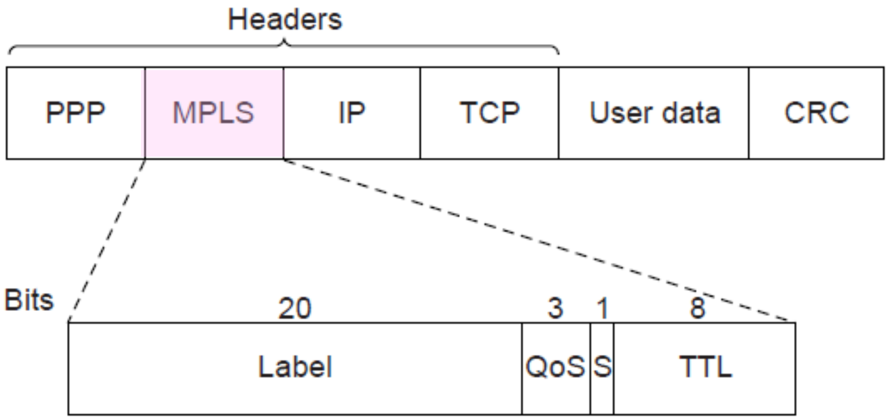
# Virtual Circuits (4)

- Each router has a forwarding table keyed by circuit
  - Gives output line and next label to place on packet



# MPLS (Multi-Protocol Label Switching, §5.6.5)

- A virtual-circuit like technology widely used by ISPs
  - ISP sets up circuits inside their backbone ahead of time
  - ISP adds MPLS label to IP packet at ingress, undoes at egress



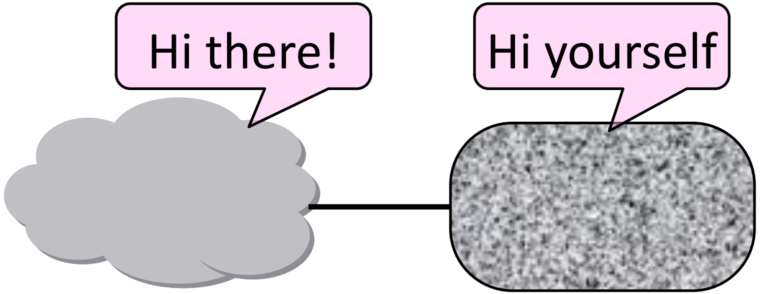
# Datagrams vs Virtual Circuits

- Complementary strengths

| Issue | Datagrams | Virtual Circuits |
|---|---|---|
| Setup phase | Not needed | Required |
| Router state | Per destination | Per connection |
| Addresses | Packet carries full address | Packet carries short label |
| Routing | Per packet | Per circuit |
| Failures | Easier to mask | Difficult to mask |
| Quality of service | Difficult to add | Easier to add |

# Topic

- How do we connect different networks together?
  - This is called <u>internetworking</u>
  - We’ll look at how IP does it

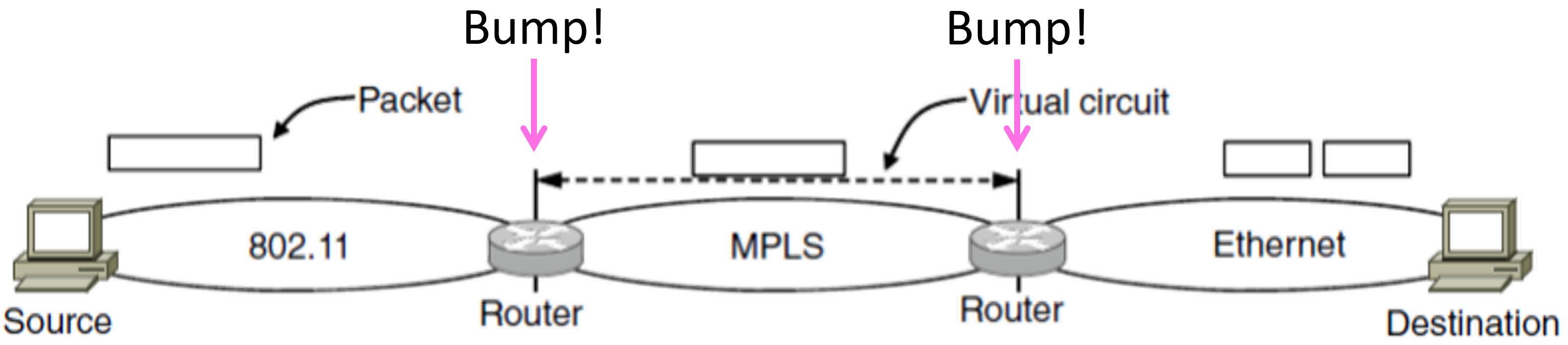


# How Networks May Differ

- Basically, in a lot of ways:
  - Service model (datagrams, VCs)
  - Addressing (what kind)
  - QOS (priorities, no priorities)
  - Packet sizes
  - Security (whether encrypted)
- Internetworking hides the differences with a common protocol. (Uh oh.)

# Connecting Datagram and VC networks

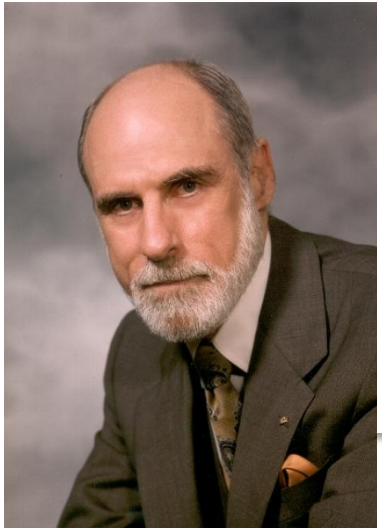
- An example to show that it's not so easy
  - Need to map destination address to a VC and vice-versa
  - A bit of a "road bump", e.g., might have to set up a VC



# Internetworking – Cerf and Kahn

- Pioneered by Cerf and Kahn, the "fathers of the Internet"
  - In 1974, later led to TCP/IP
- Tackled the problems of interconnecting networks
  - Instead of mandating a single network technology

Vint Cerf



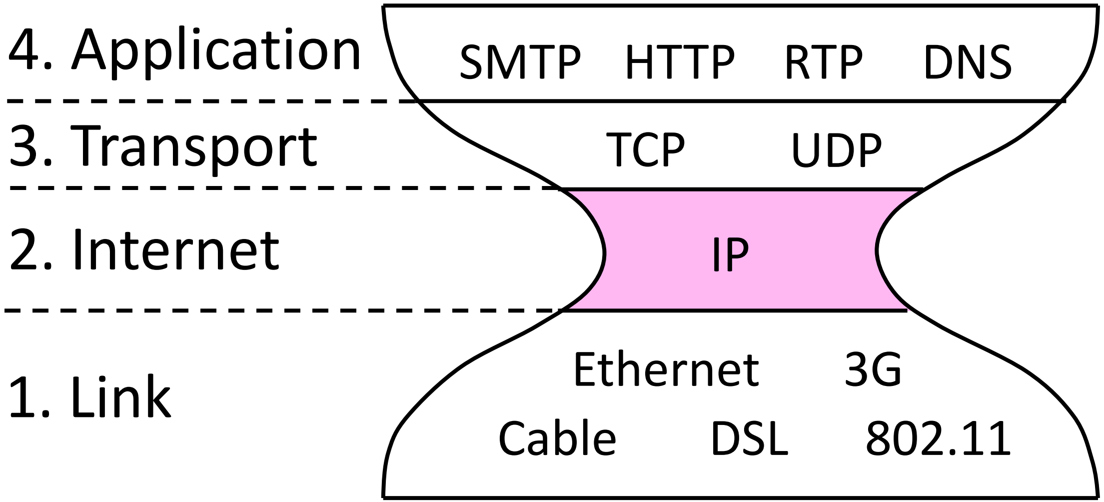
© 2009 IEEE

Bob Kahn

© 2009 IEEE

# Internet Reference Model

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

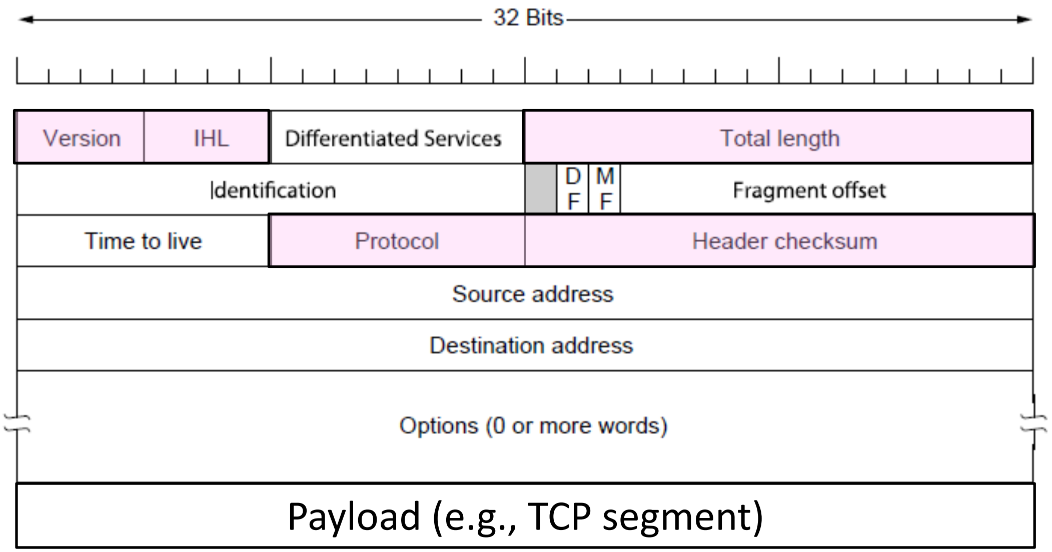


# IP as a Lowest Common Denominator

- Suppose only some networks support QOS or security etc.
  - Difficult for internetwork to support
- Pushes IP to be a “lowest common denominator” protocol
  - Asks little of lower-layer networks
  - Gives little as a higher layer service

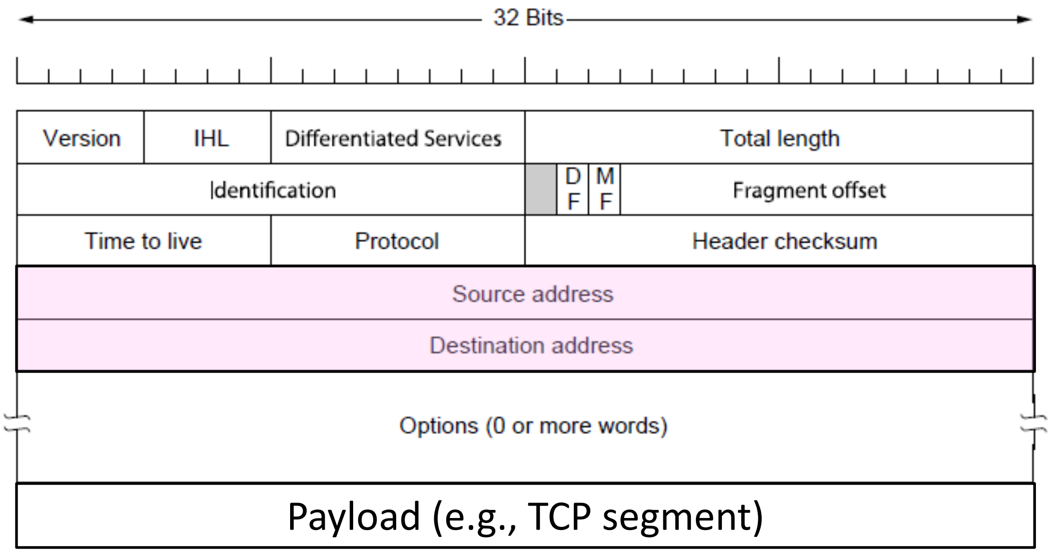
# IPv4 (Internet Protocol)

- Various fields to meet straightforward needs
  - Version, Header (IHL) and Total length, Protocol, and Header Checksum



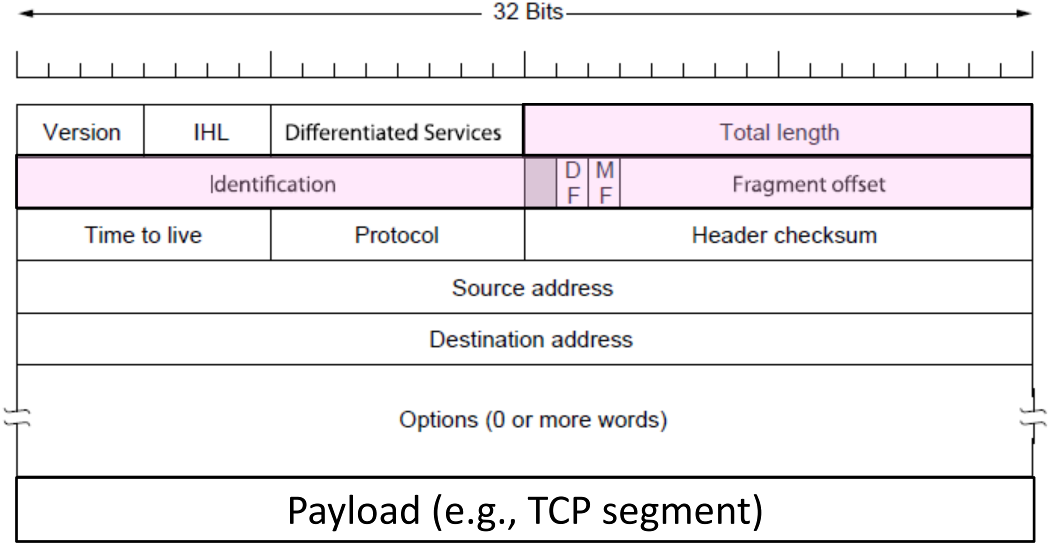
# IPv4 (2)

- Network layer of the Internet, uses datagrams
  - Provides a layer of addressing above link addresses (next)



# IPv4 (3)

- Some fields to handle packet size differences (later)
  - Identification, Fragment offset, Fragment control bits



# IPv4 (4)

- Other fields to meet other needs (later, later)
  - Differentiated Services, Time to live (TTL)