CS 43: Computer Networks

23: Traffic Management and the Link Layer

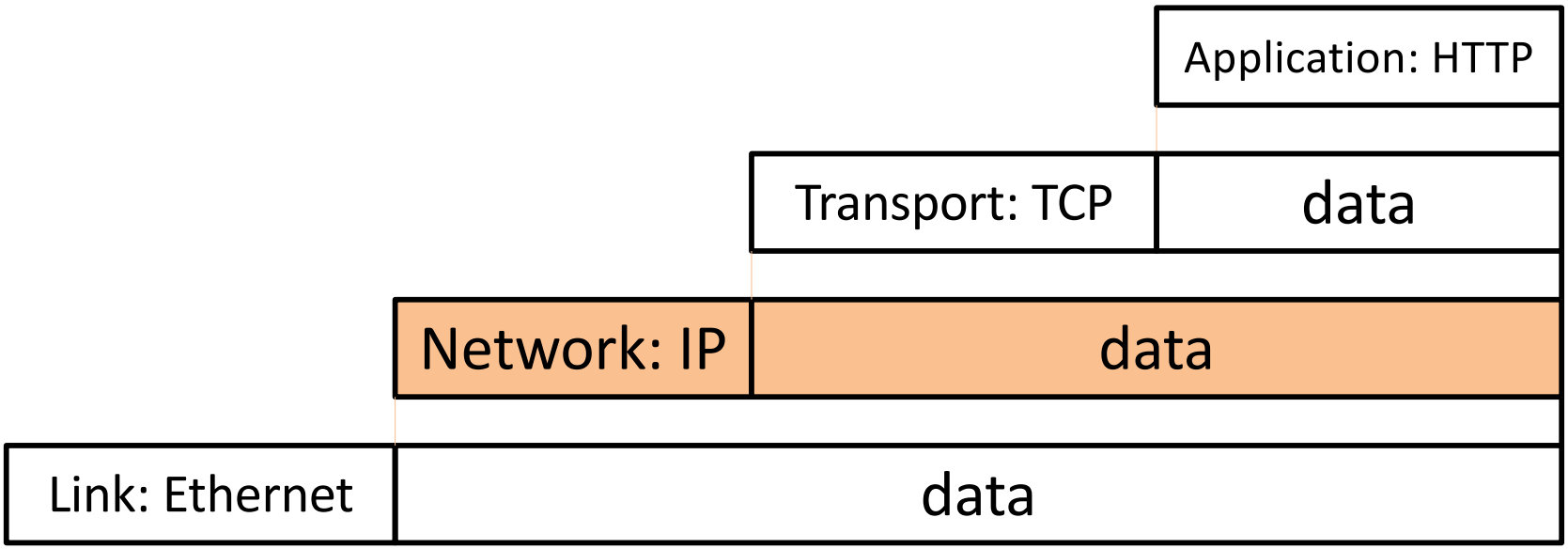
Dec 3, 2019

Adapted from Slides by: J.Kurose, J.Rexford, K. Webb

Reading Quiz

Network Layer

- Function: Route packets end-to-end on a network, through multiple hops



Network Layer Functions

- Forwarding*:* move packets from router's input to appropriate router output
  - Look up in a table
- Routing: determine route taken by packets from source to destination.
  - Populating the table

Net Neutrality

- how an ISP should share/allocation its resources
  - protecting innovation, free speech, and competition on the Internet

- Example: Comcast didn't like BitTorrent, started injecting RSTs into user TCP streams.

- Scarier example: You like Netflix, but your ISP has their own video service. They degrade (or block) Netflix service unless you pay $$$.

Net Neutrality

Cases for:

- End to end principle
- Prevent customer extortion
- Allow for innovation

Google, Microsoft, Yahoo, Amazon, eBay

Cases against:

- ISP owns their network
- Asymmetric application bandwidth usage
- We shouldn't legislate the Internet, it moves too fast

Cisco, many ISPs

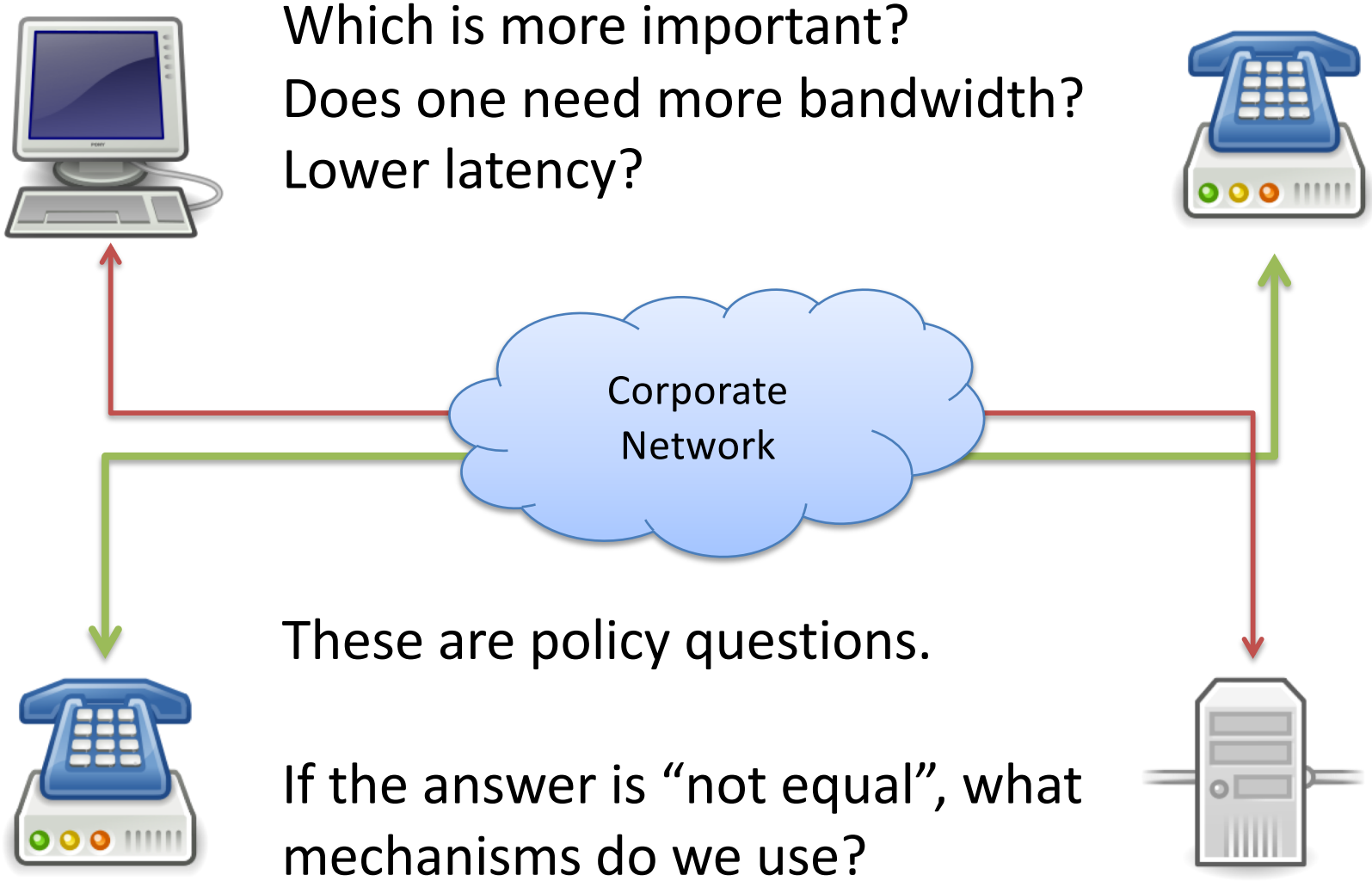
Today

- We've seen the behavior of TCP/IP, and routers
- We've joked about the option of marking packets as "urgent"
  - As a lone user, your cries for urgency will likely be ignored by one or more ISPs on the Internet
- False implication: All traffic is treated equally.

Scenarios

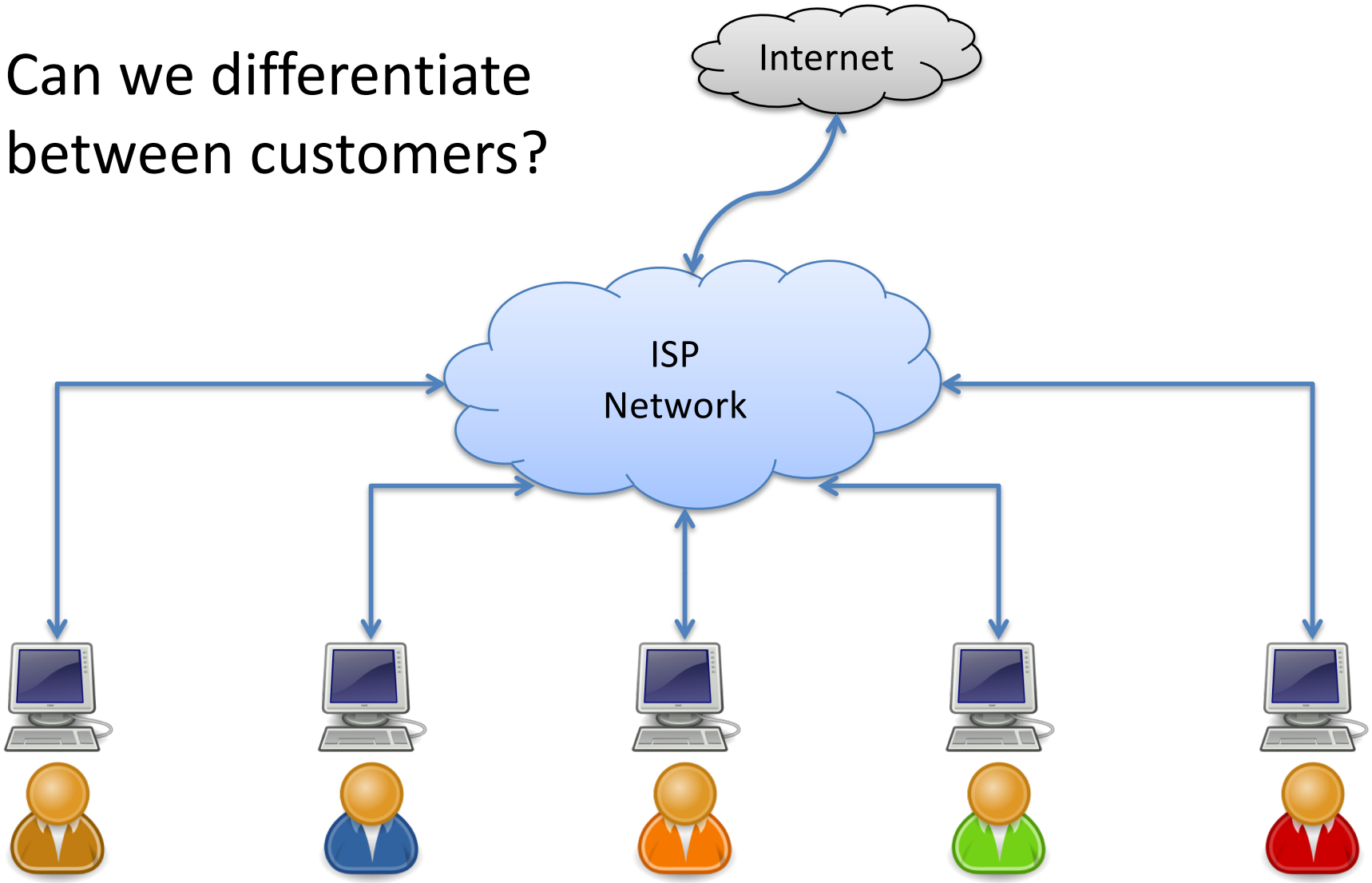
- Things we can do at the network layer to:
  - Treat traffic differently
  - Improve congestion control
- You own a private network
  - Corporate network
  - Data center
  - ISP
- You want to provide better performance to:
  - More important services
  - Customers who pay more

Example 1: Corporate Phones



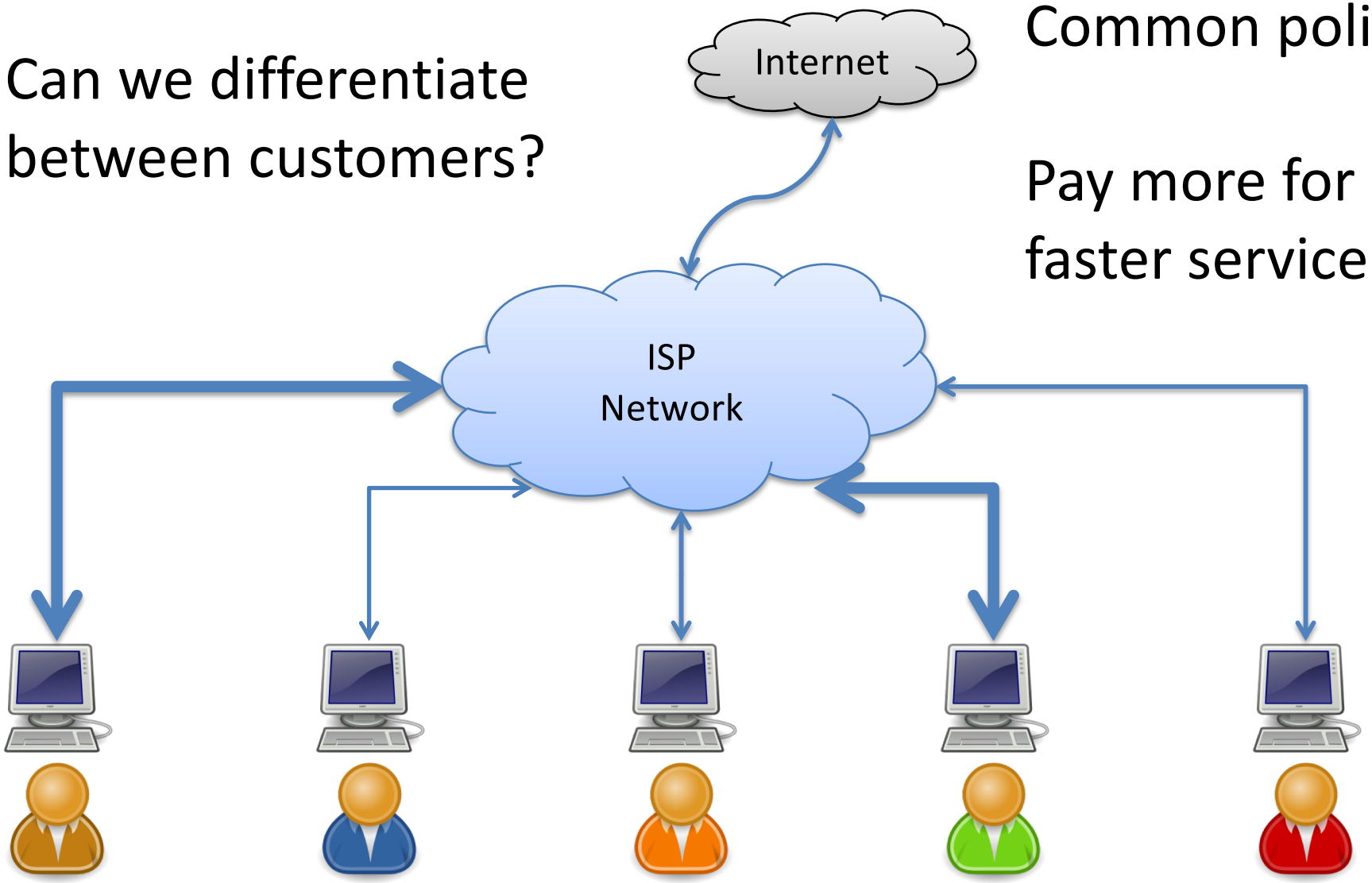
Example 2: ISP Customers

Can we differentiate between customers?



Example 2: ISP Customers

Can we differentiate between customers?



Common policy:

Pay more for faster service!

How might we enforce these types of policies?

A. Require that end-hosts police their traffic.

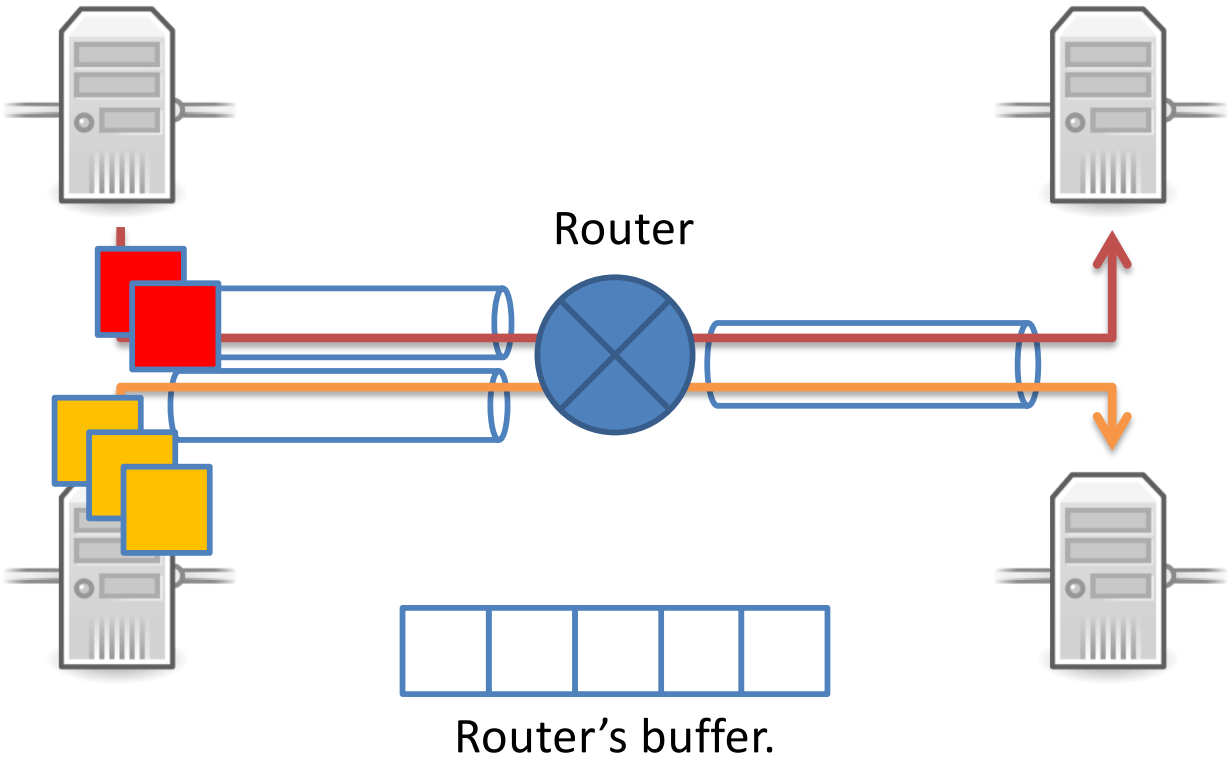
B. Change how routers queue traffic.

C. Ask users nicely to comply with policy.

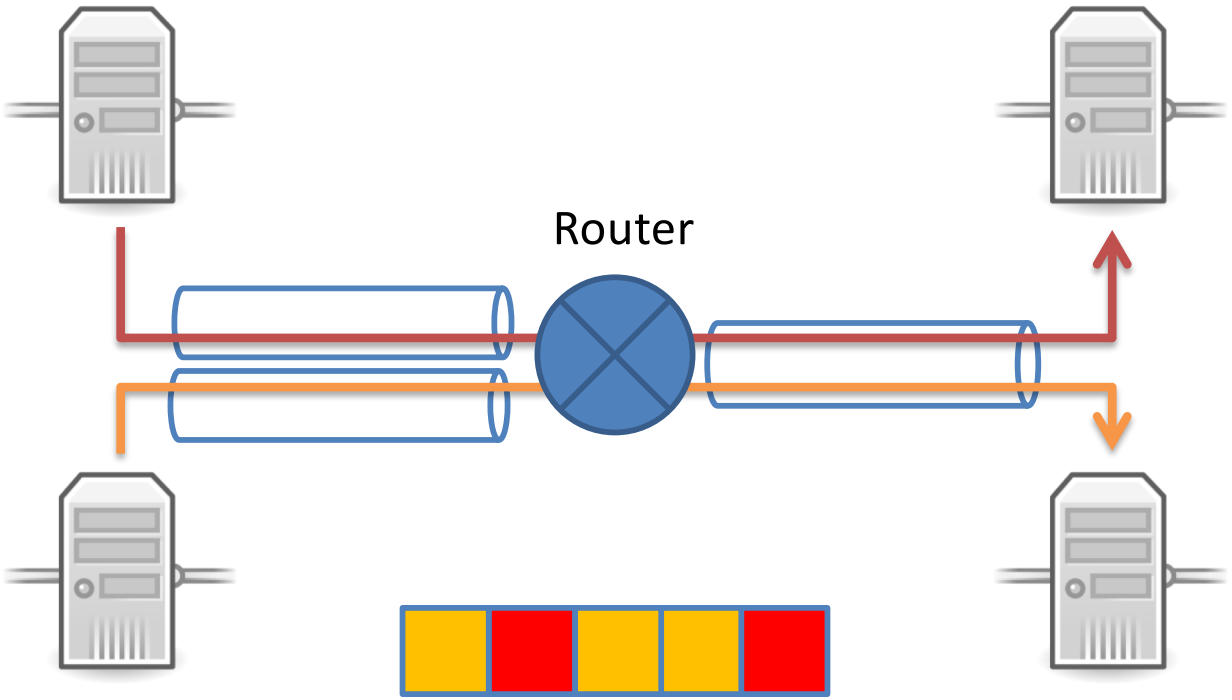
D. Enforce policies some other way.

E. There is nothing we can do.

Recall Queueing



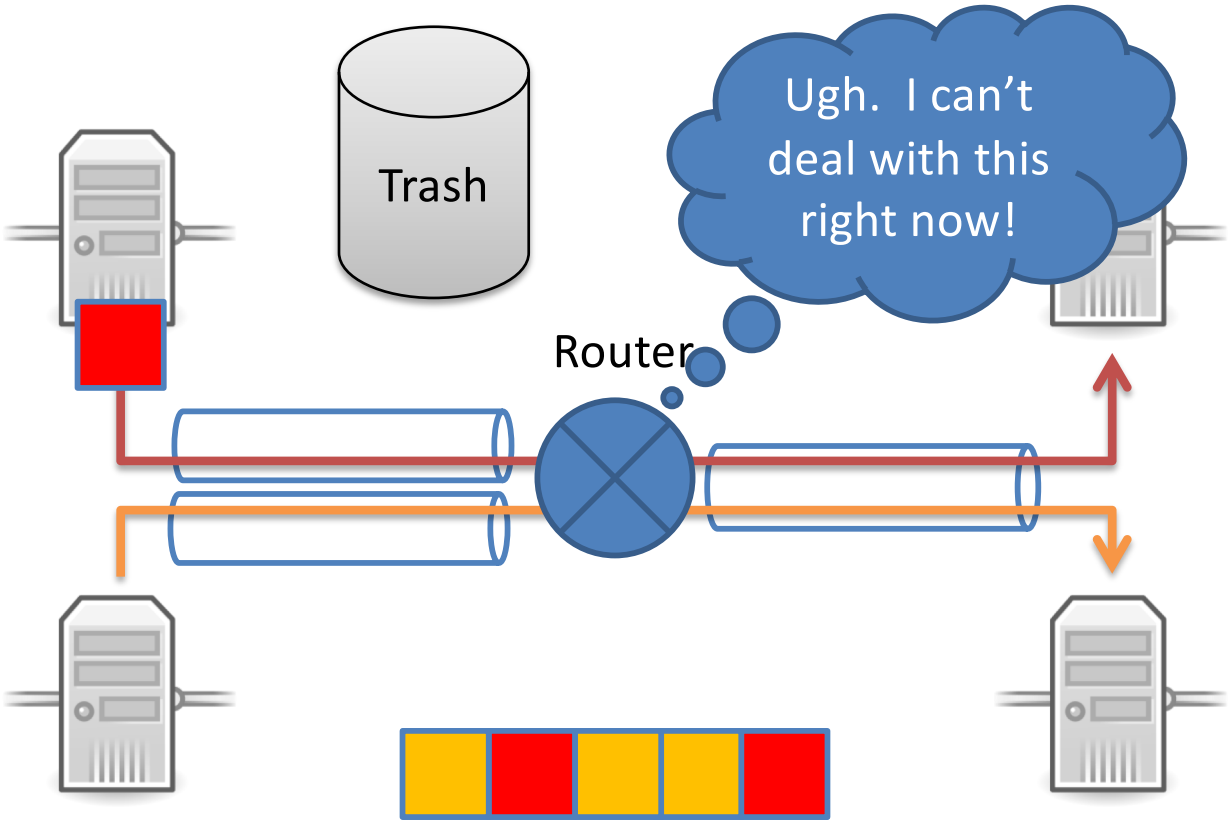
Recall Queueing



Router's buffer.

Incoming rate is faster than outgoing link can support.

Recall Queueing



Router's buffer.

Incoming rate is faster than outgoing link can support.

Basic Buffer Management

- FIFO + drop-tail
  - Simplest choice
  - Used widely in the Internet
- FIFO (first-in-first-out)
  - Traffic queued in first-come, first-served fashion
- Drop-tail
  - Arriving packets get dropped when queue is full
- Important distinction:
  - FIFO: queueing (scheduling) discipline
  - Drop-tail: drop policy

FIFO/Drop-Tail Problems

- Doesn't differentiate between flows/users
- No policing: send more, get more service
- Leaves responsibility of congestion control completely to the edges (e.g., TCP)
- Synchronization: hosts react to same events

FIFO/Drop-Tail Problems

- Doesn't differentiate between flows/users
- No policing: send more, get more service

QoS

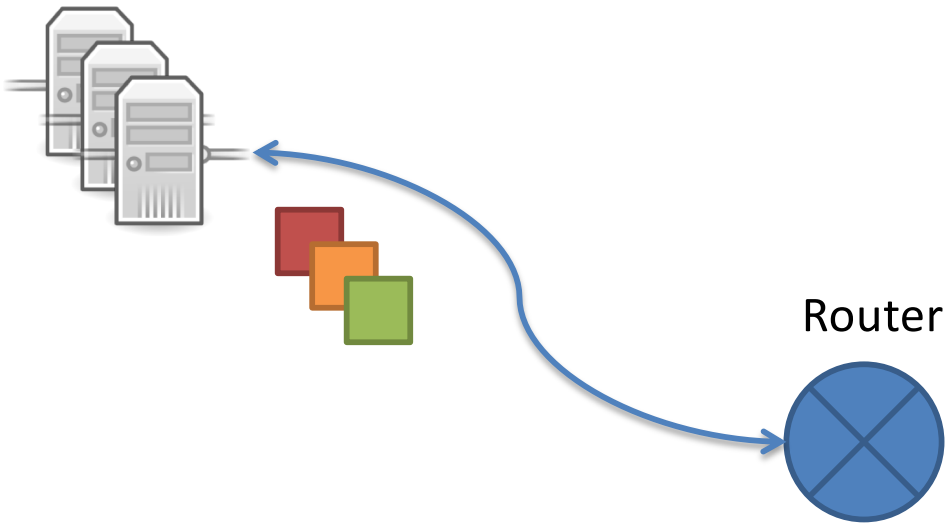
- Leaves responsibility of congestion control completely to the edges (e.g., TCP)
- Synchronization: hosts react to same events

AQM

Quality of Service (QoS)

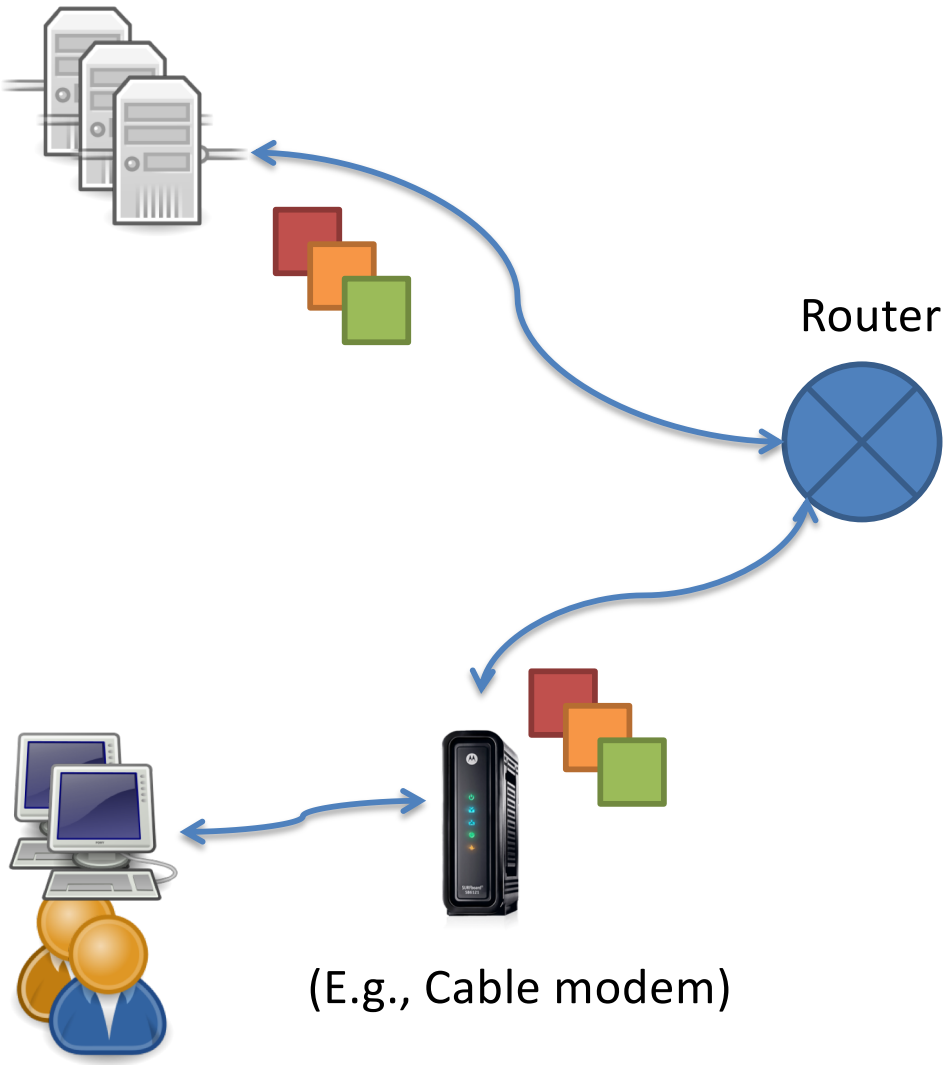
- QoS is a broad topic! We're going to discuss:
  - Mechanism for differentiating users/flows
  - Mechanism for enforcing rate limits
  - Mechanism for prioritizing traffic

Differentiating Users



- If you control end hosts:
  - Mark packets in OS according to policy.
- Take advantage of IP's class of service or options header fields

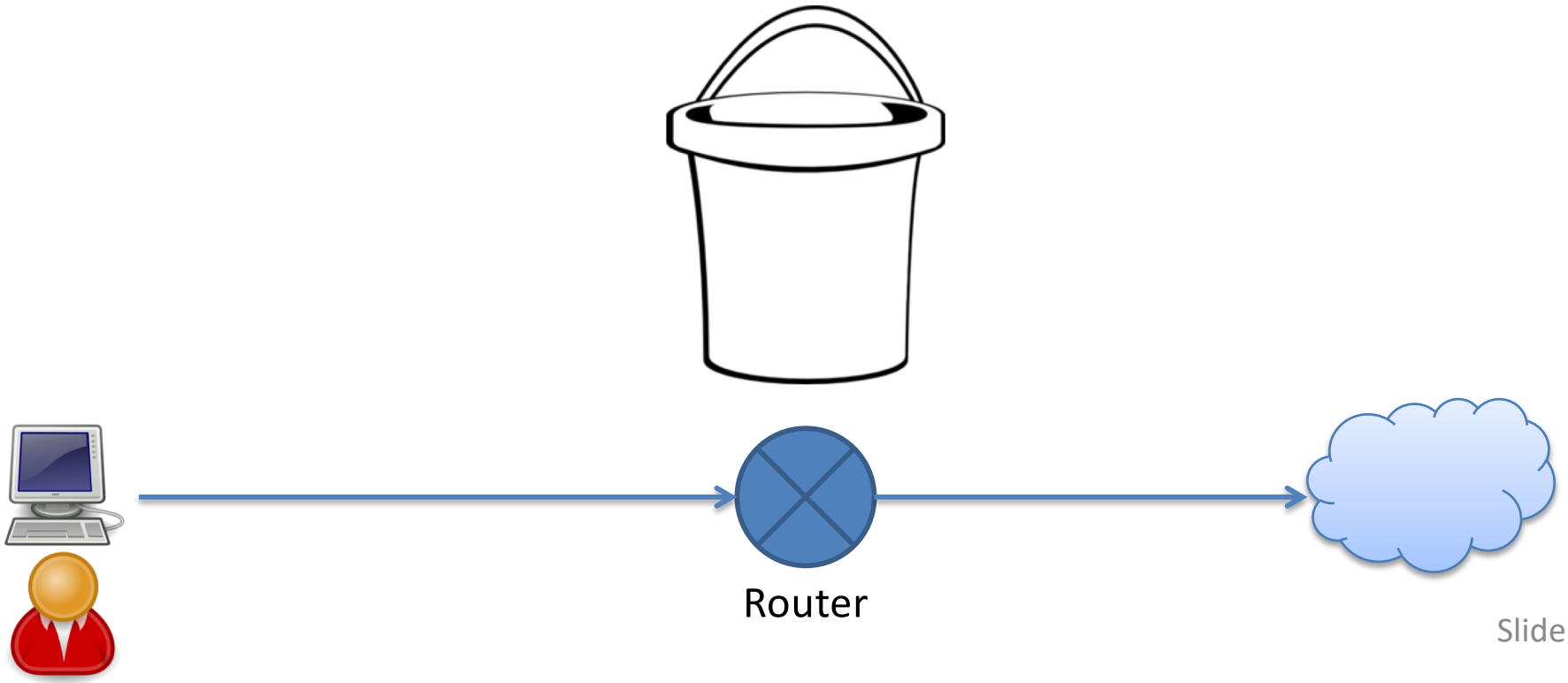
Differentiating Users



- If you control end hosts:
  - Mark packets in OS according to policy.
- Take advantage of IP's class of service or options header fields
- Otherwise:
  - Introduce an intermediate device you trust.

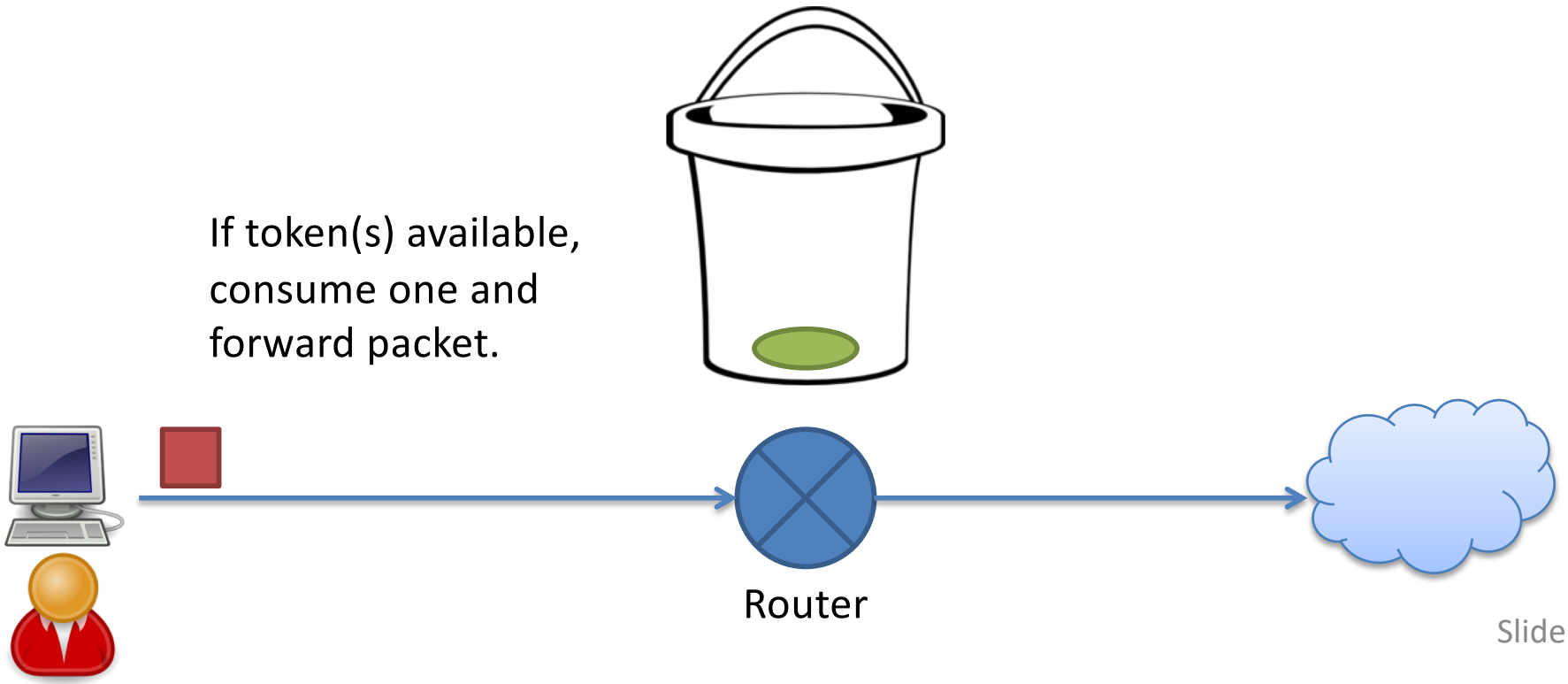
Enforcing (Policing) Rate Limits

- Example: the red user gets at most 10 Mbps
- Solution: Token bucket



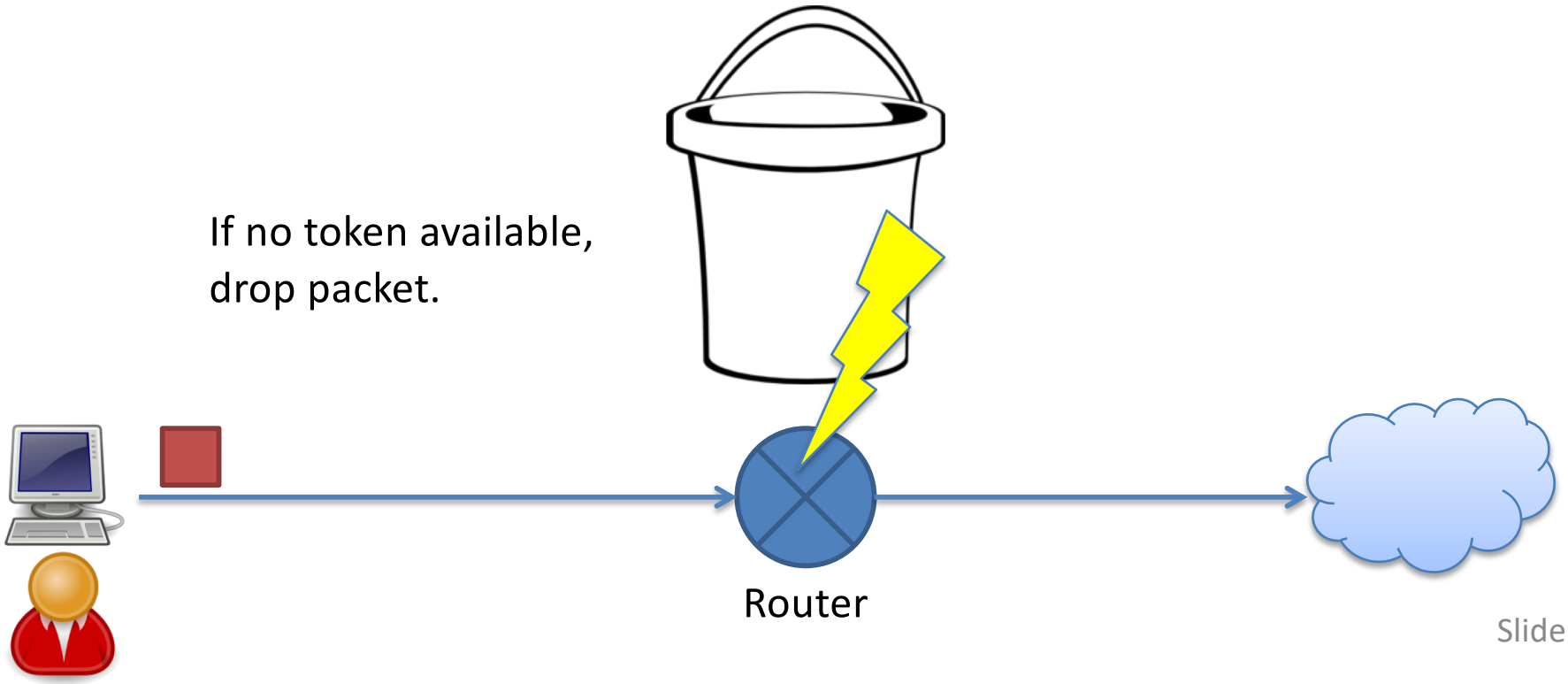
Enforcing (Policing) Rate Limits

- Example: the red user gets at most 10 Mbps
- Solution: Token bucket



Enforcing (Policing) Rate Limits

- Example: the red user gets at most 10 Mbps
- Solution: Token bucket

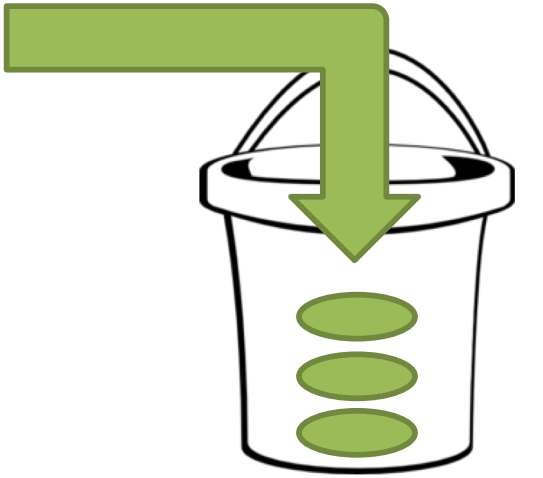


Enforcing (Policing) Rate Limits

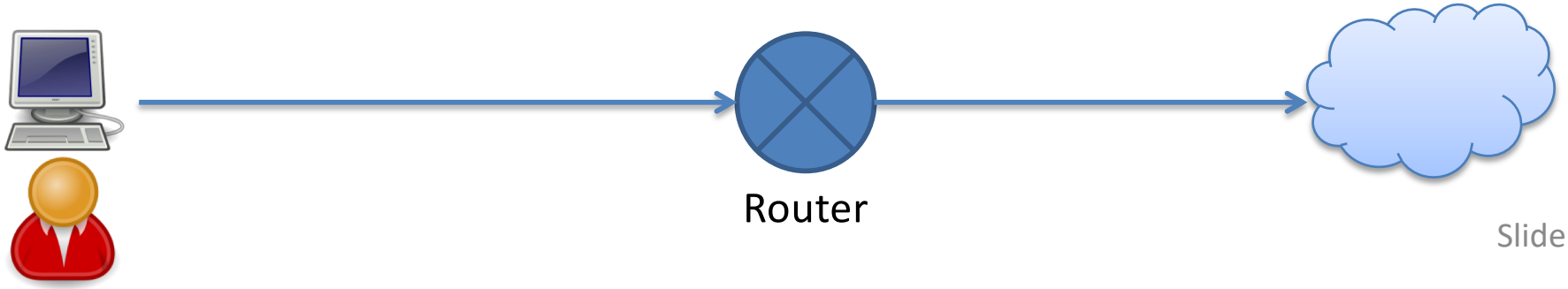
- Example: the red user gets at most 10 Mbps
- Solution: Token bucket

No matter how fast user sends, limited by number of tokens, which replenish at controlled rate!

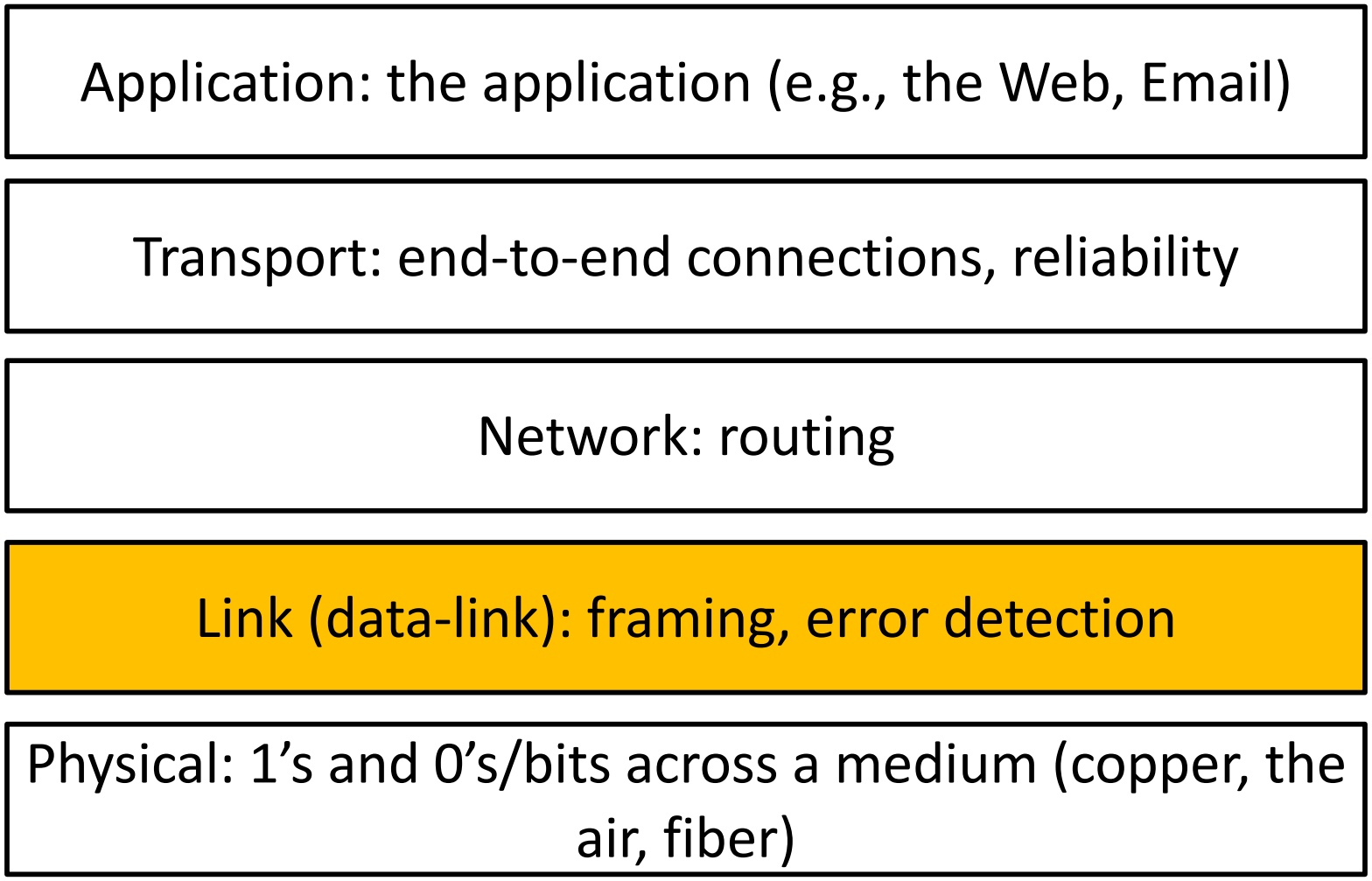
Router adds tokens at specified rate. (10 Mbps)



Bucket depth determines burst size.

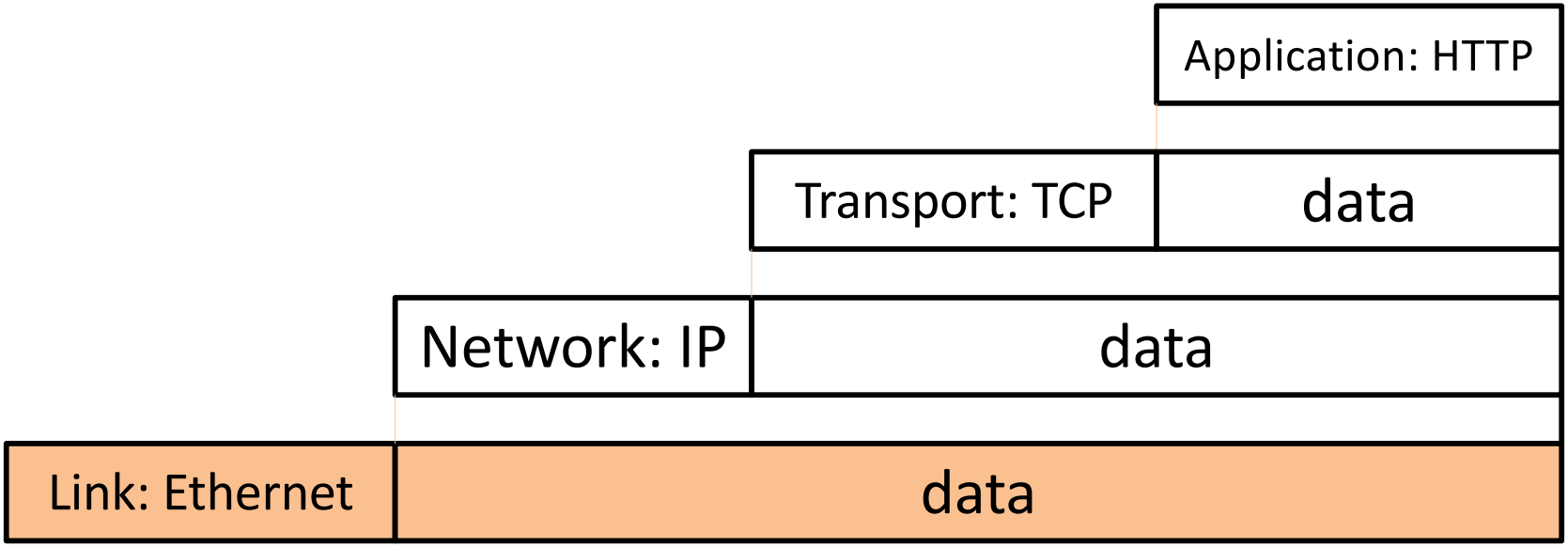


The Link Layer



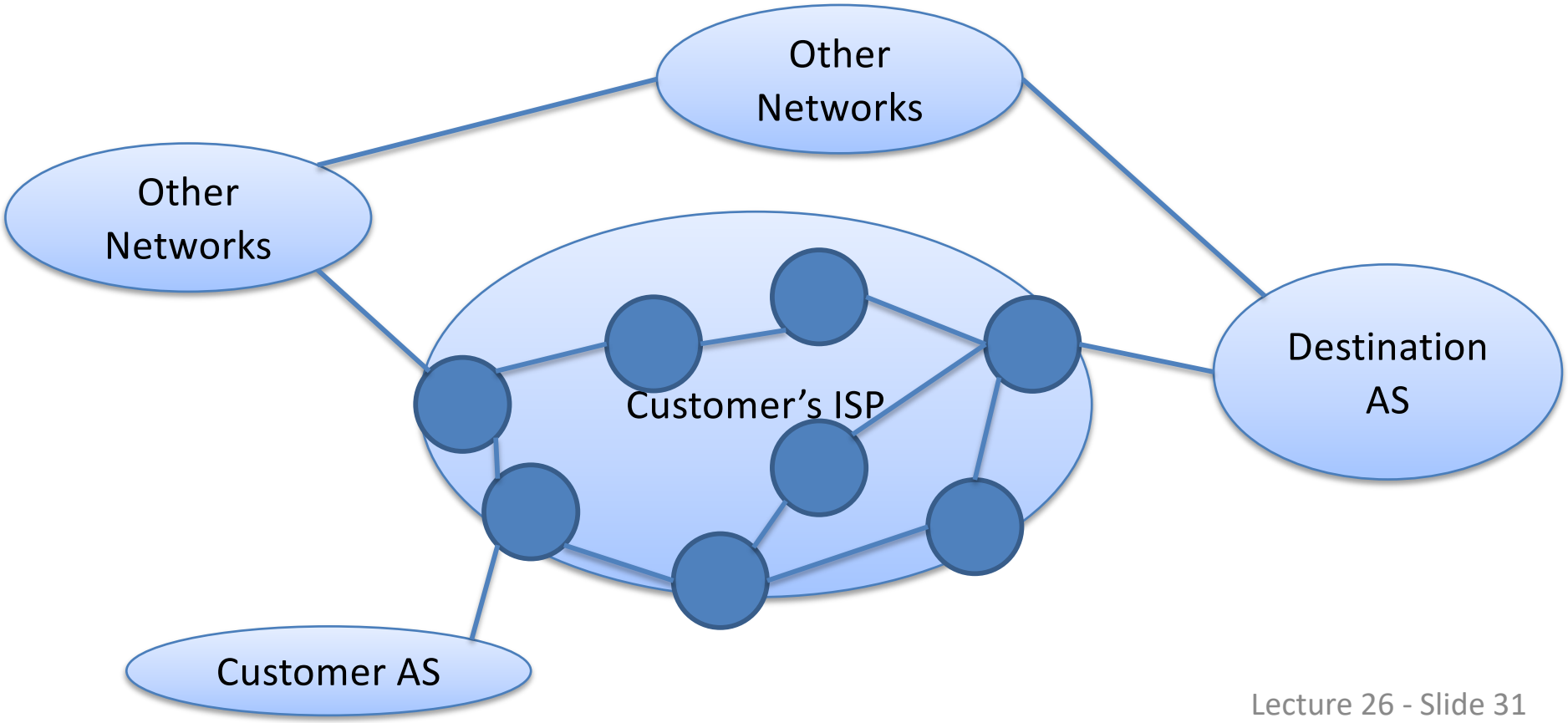
Network Layer

- Function: Route packets end-to-end on a network, through multiple hops

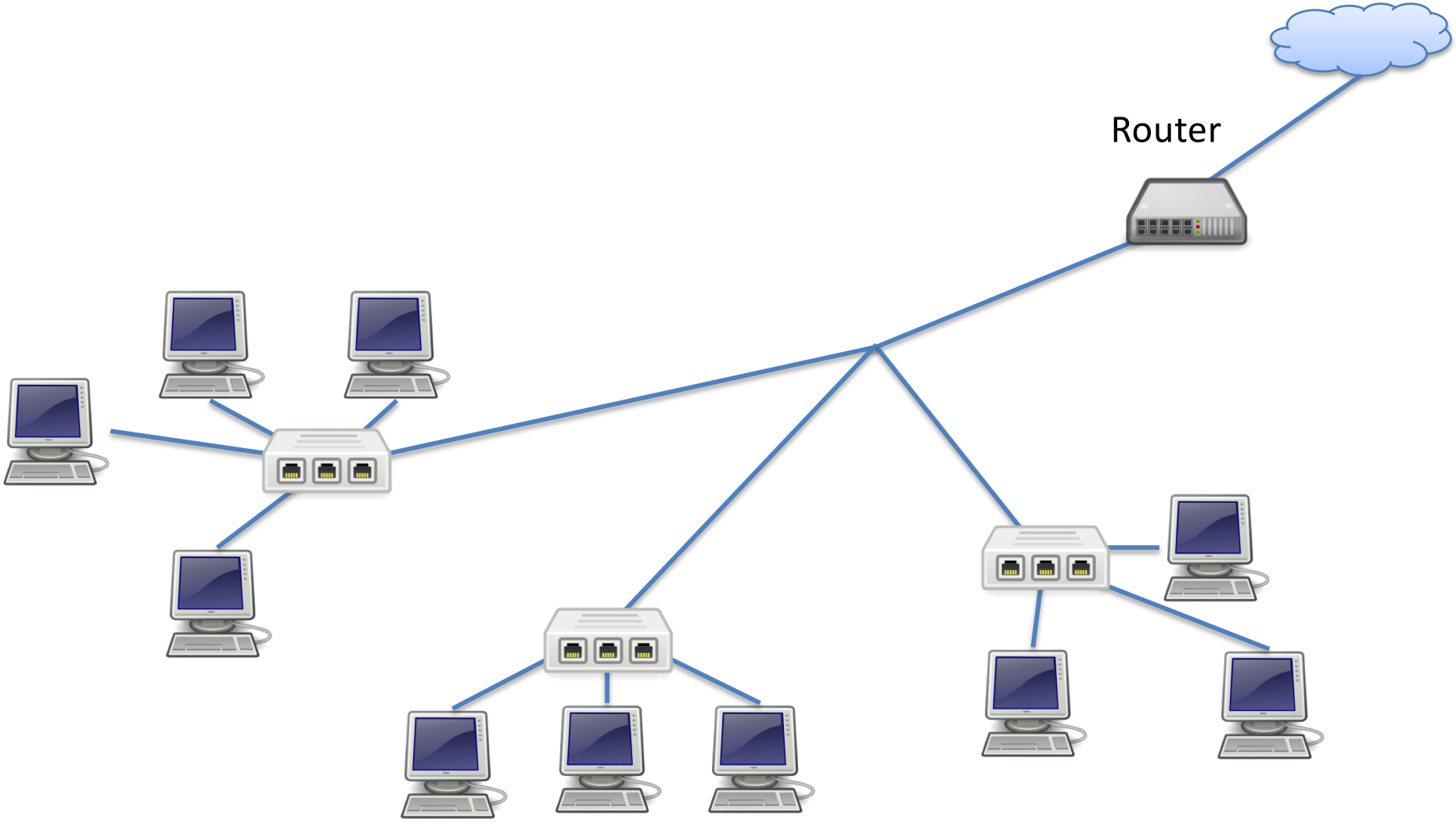


From Macro- to Micro-

- Previously, we looked at Internet scale…



Within a Subnet



What is a Link?

Communication Medium

Network Adapter

Adapted from J.Rexford

Link Layer Goal

- Get from one node to it's adjacent neighbor.
- Abstract the details of the underlying network technology from the protocols above it (IP).
- Lots of media with different characteristics:
  - Copper cable
  - Fiber optic cable
  - Radio/electromagnetic broadcast
  - Satellite

Challenges

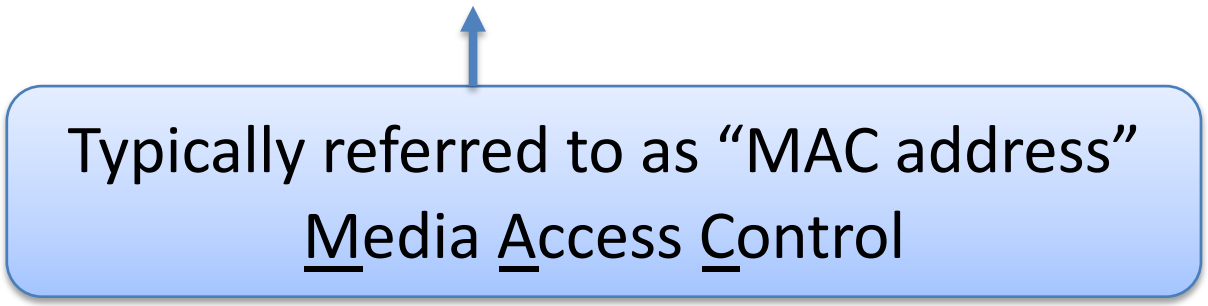
- Even with one medium:
  - Potentially many ways to format & signal data.
  - Multiple users may contend to transmit.
  - How do we address endpoints?
  - How do we locate destinations?

Link Layer Functions

Addressing: identifying endpoints

- Must be able to uniquely identify each host on the network. Can't assume IP.

- Implication: each host on the Internet will have **two** addresses: IP & link-layer



Why Not Just Use IP Addresses?

- Links can support *any* network protocol
  - Not just for IP (e.g., IPX, Appletalk, X.25, …)
  - Different addresses on different kinds of links
- An adapter may move to a new location
  - So, cannot simply assign a static IP address
  - Instead, must reconfigure the adapter's IP address
- Must identify the adapter during bootstrap
  - Need to talk to the adapter to assign it an IP address

Adapted from J.Rexford

Addressing

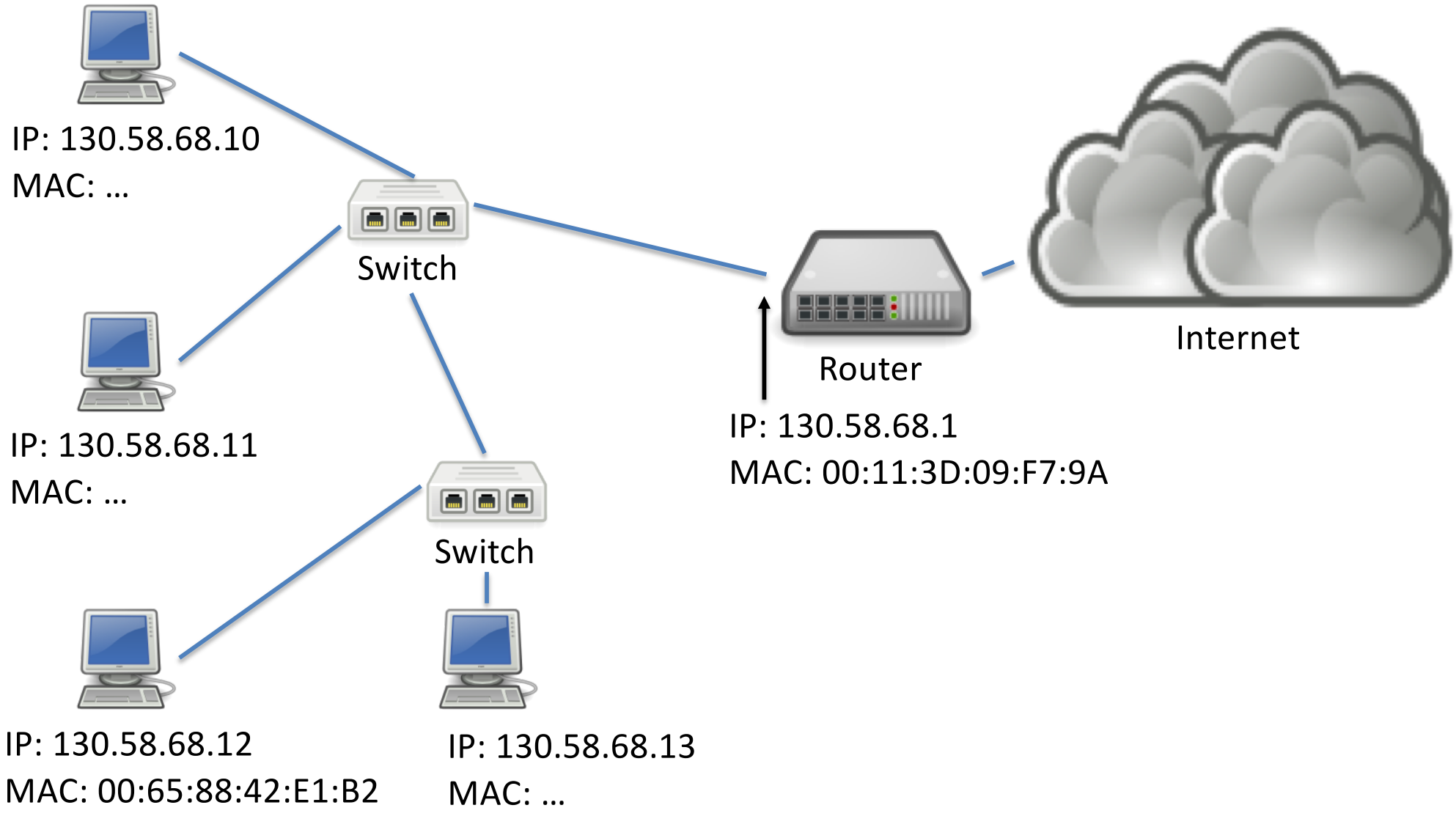
- Typically, humans deal in IP addresses (or DNS names that resolve to them)
- Network needs a mechanism to determine corresponding MAC address for local sending

ARP: Address Resolution Protocol

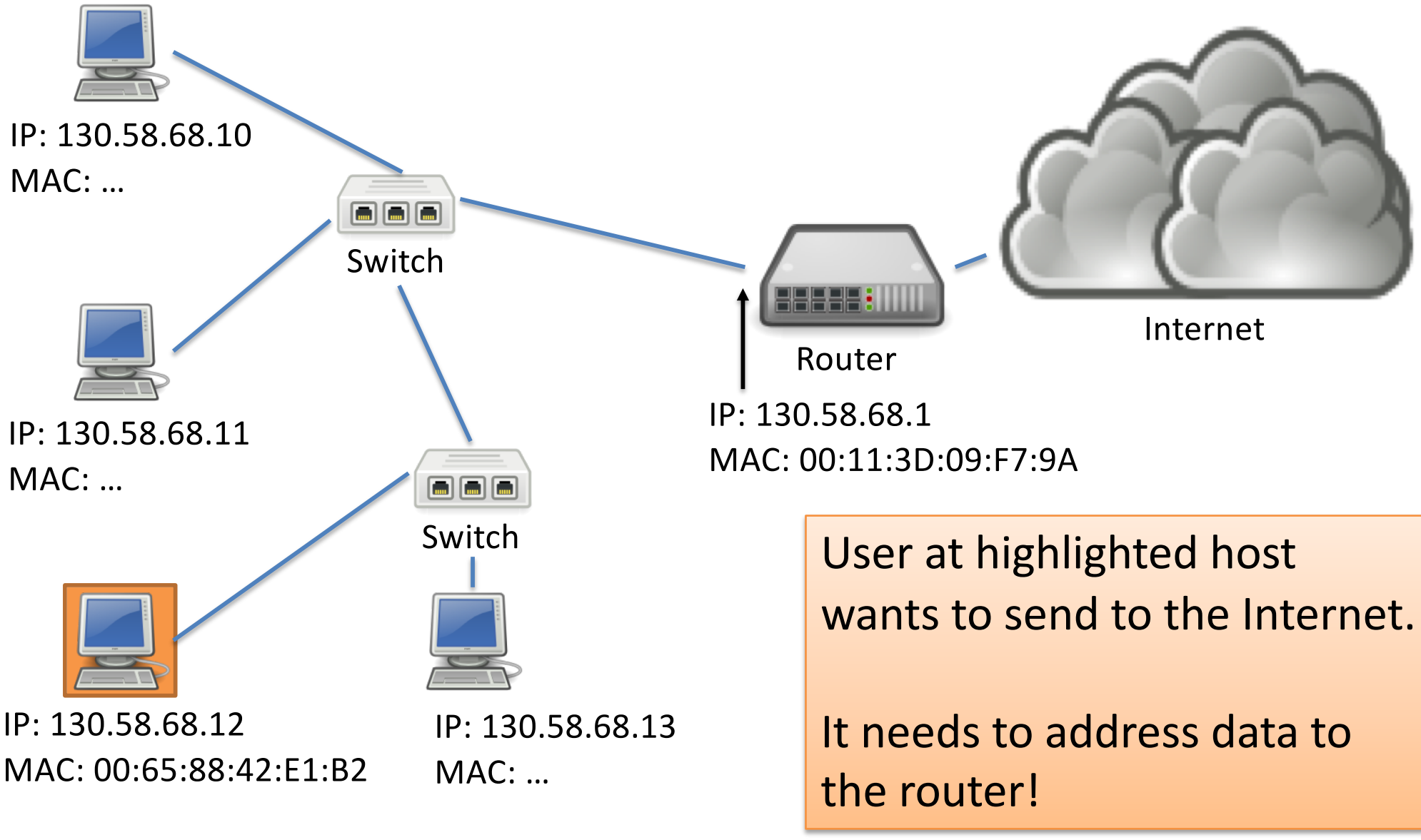
- Common in networks you use: Ethernet, WiFi
- Broadcast to entire local network:
  - "I'm looking for the MAC address of the host with IP address A.B.C.D. If you're out there, please respond to me!"

- You will implement this in lab 7!

ARP Example


IP: 130.58.68.10
MAC: …
Switch
IP: 130.58.68.11
MAC: …
Switch
Router
IP: 130.58.68.1
MAC: 00:11:3D:09:F7:9A
Internet
IP: 130.58.68.12
MAC: 00:65:88:42:E1:B2
IP: 130.58.68.13
MAC: …

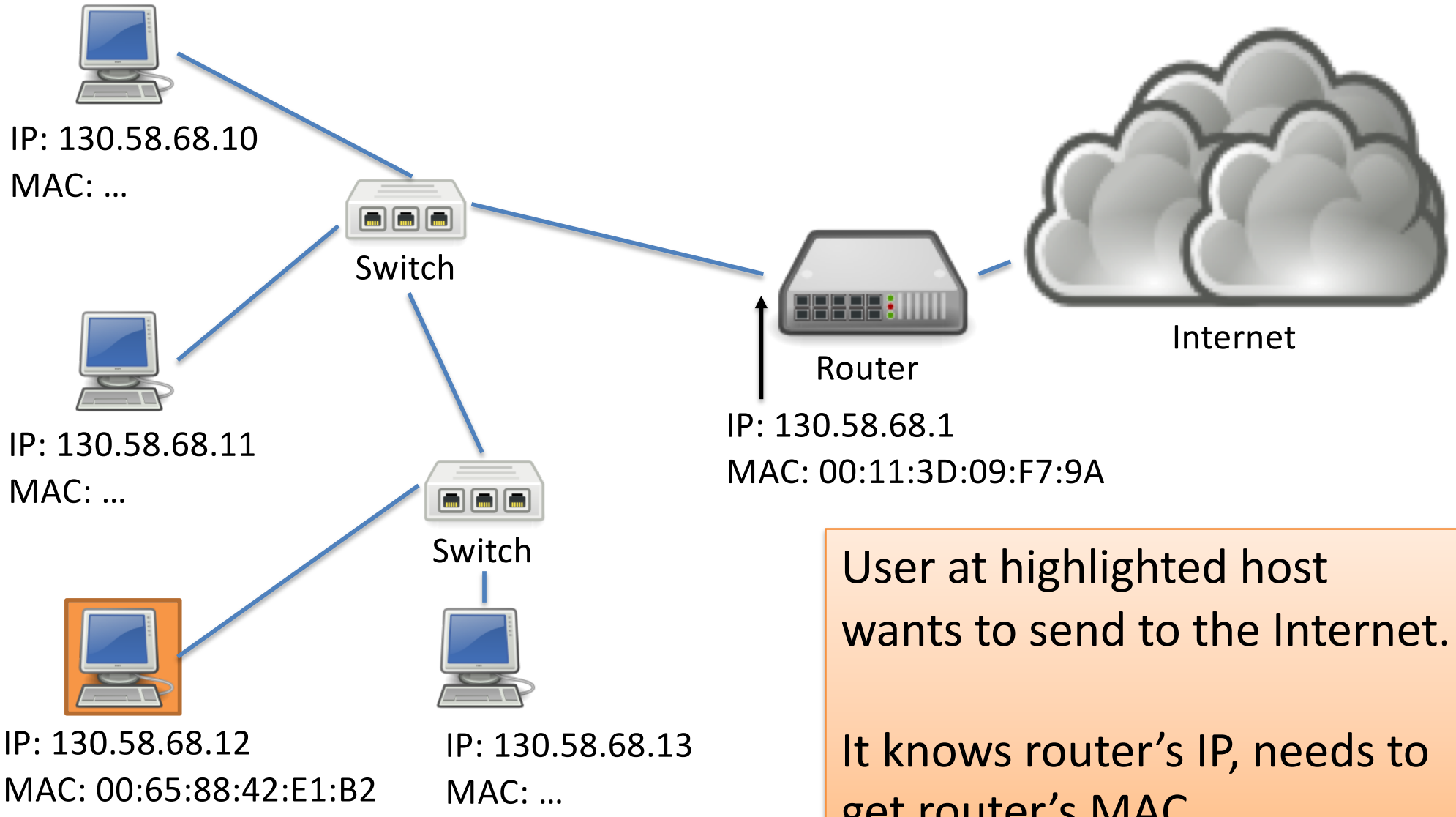
ARP Example



User at highlighted host wants to send to the Internet.

It needs to address data to the router!

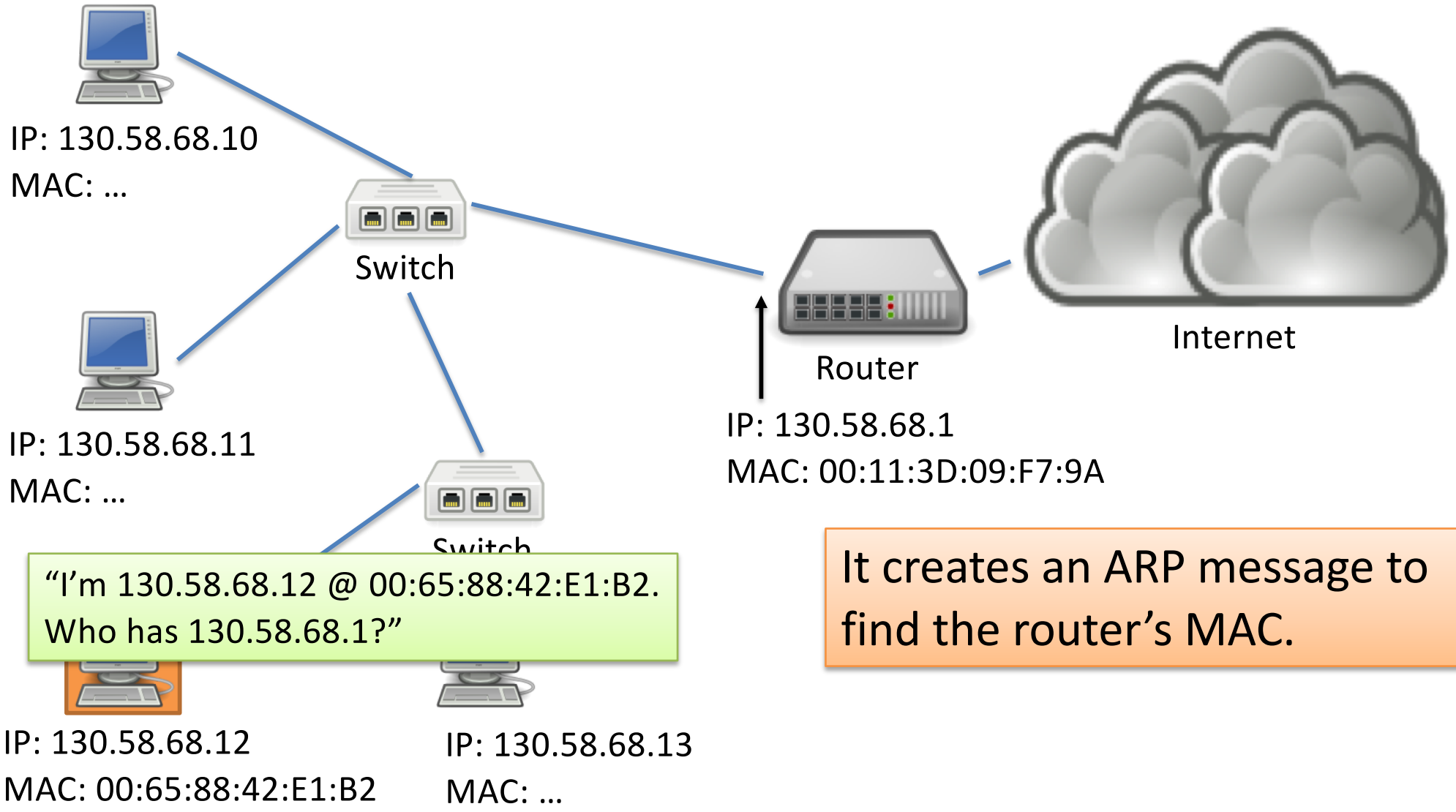
ARP Example



User at highlighted host wants to send to the Internet.

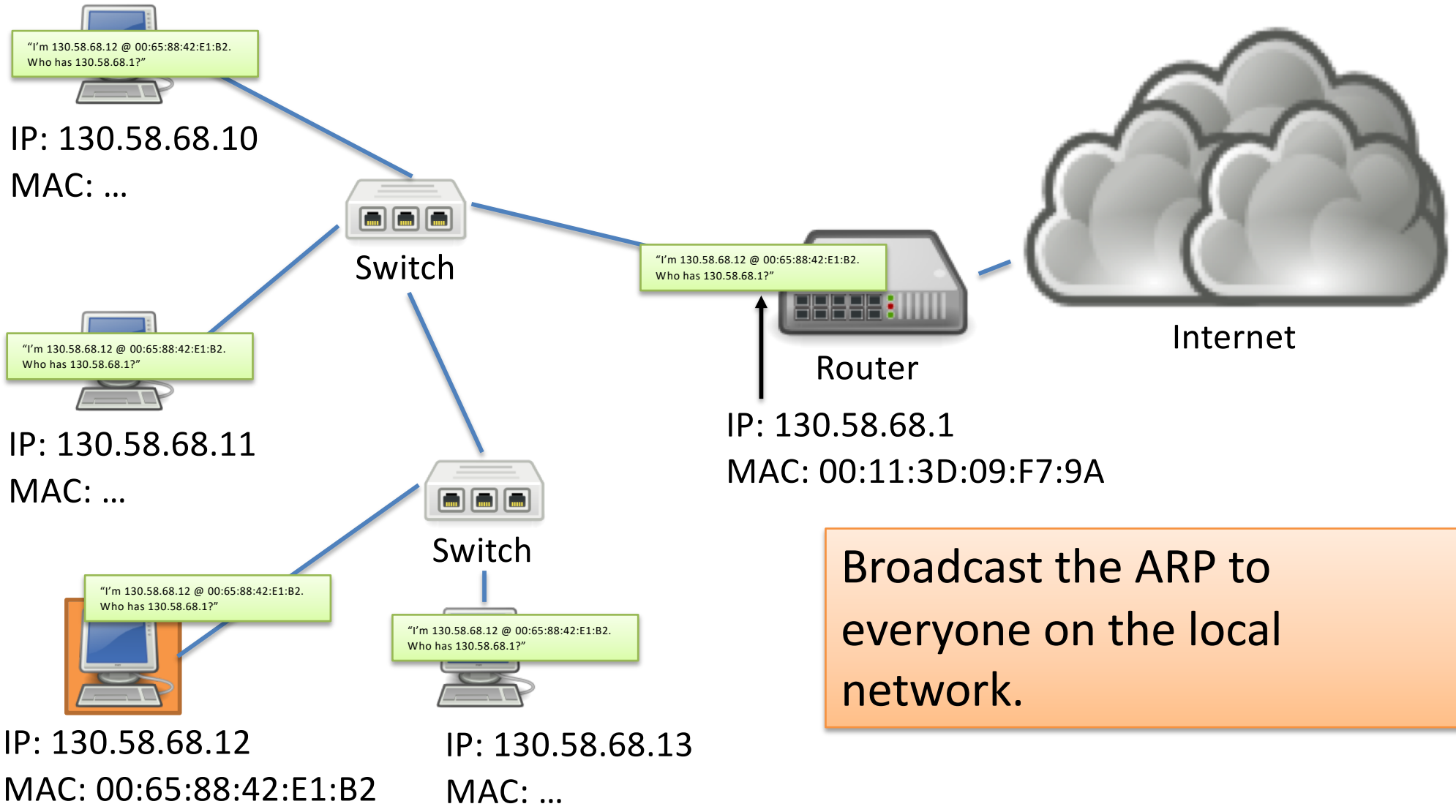
It knows router's IP, needs to get router's MAC.

ARP Example



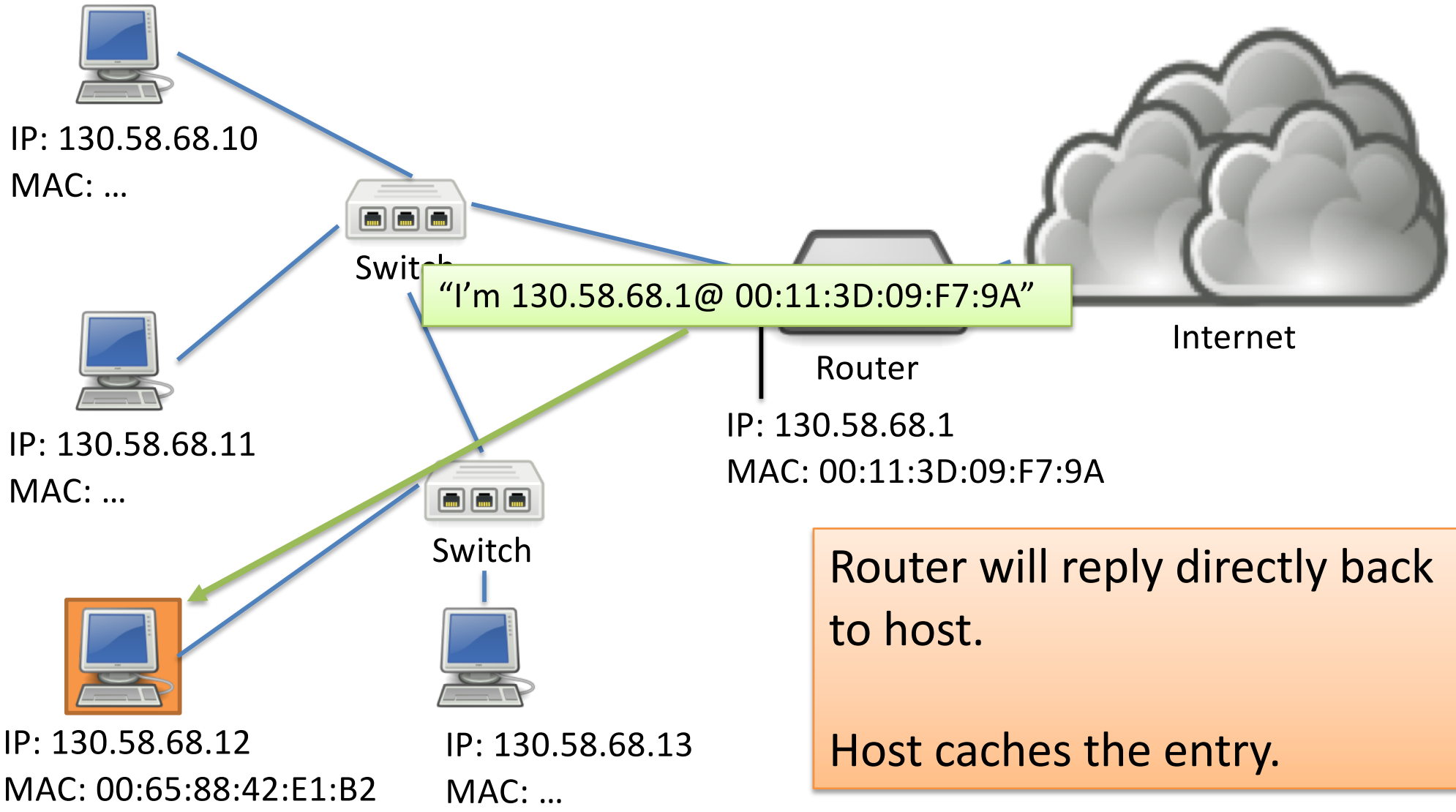
It creates an ARP message to find the router's MAC.

ARP Example



Broadcast the ARP to everyone on the local network.

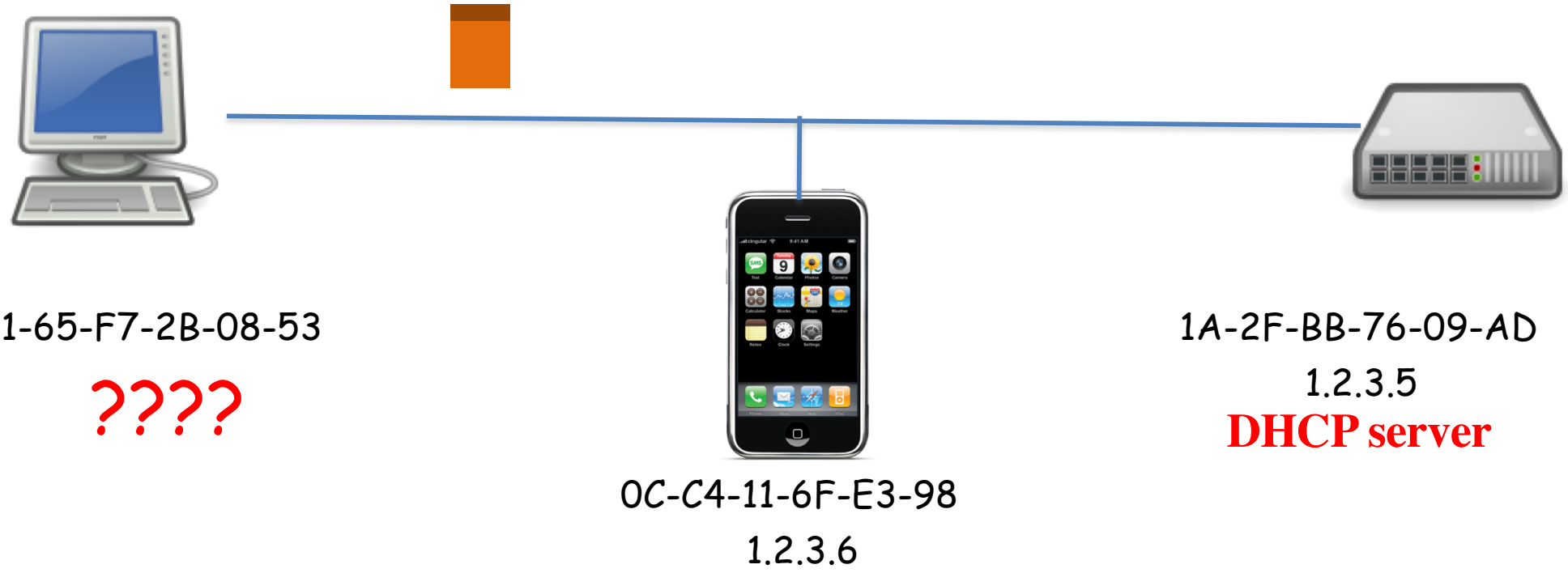
ARP Example



Router will reply directly back to host.

Host caches the entry.

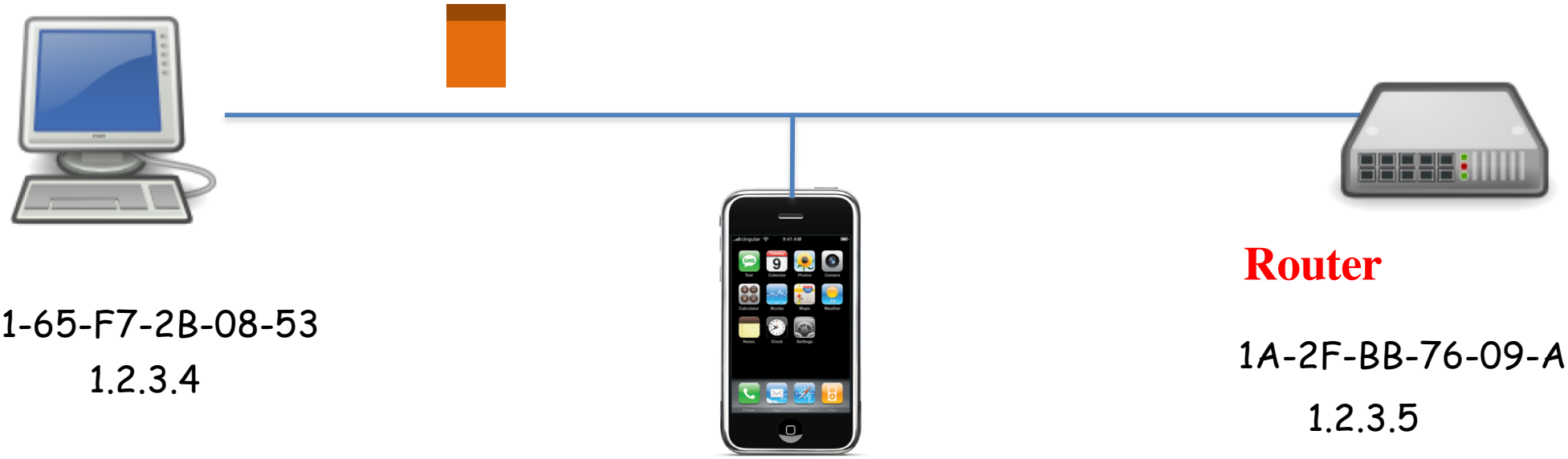
Putting it together: Who Am I: Acquiring an IP Address



- Dynamic Host Configuration Protocol (DHCP)
  - Broadcast "I need an IP address, please!"
  - Response "You can have IP address 1.2.3.4."

Adapted from J.Rexford

Putting it together: Who Are You: Discovering the Receiver



- Address Resolution Protocol (ARP)
  - Broadcast "who has IP address 1.2.3.6?"
  - Response "0C-C4-11-6F-E3-98 has 1.2.3.6!"

Link Layer Functions

1. Addressing: identifying endpoints

2. Framing: Dividing data into pieces that are sized for the network to handle.

- Data pieces:
  - Transport: Segments
  - Network: Datagrams (or packets)
  - Link: Frames
  - Physical: Bits

Link Layer Functions

1. Addressing: identifying endpoints

2. Framing: Dividing data into pieces that are sized for the network to handle.

- Data pieces:
  - Transport: Segments
  - Network: Datagrams (or packets)
  - Link: Frames
  - Physical: Bits

"Big freaking deal, Sherlock!"

Why do we put a limit on the size of a frame?

A. To keep one user from hogging the channel.

B. To make signaling message boundaries easier.

C. To achieve higher performance

D. Some other reason.

Link Layer Functions

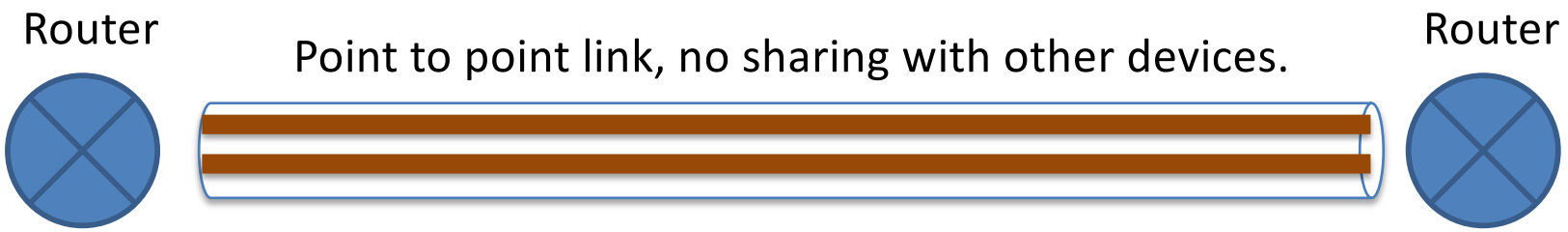
1. Addressing: identifying endpoints

2. Framing: Dividing data into pieces that are sized for the network to handle.

3. Link access: Determining how to share the medium, who gets to send, and for how long.

Link Access

- Some networks may not require much.



Example 1: Single copper wire, only one of them can send at a time.

Example 2: Two copper wires in cable, each can send on one simultaneously.

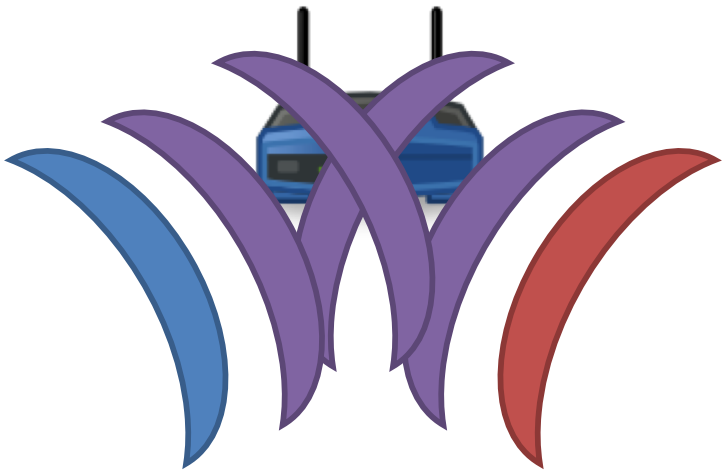
Link Access

- For other networks, this is a huge challenge.

Link Access

- For other networks, this is a huge challenge.

Collision!



How should we handle collisions in general (for WiFi and other link media)?

A. Enforce at the end hosts that only one sender transmit at a time.

B. Enforce in the network that only one sender transmit at a time.

C. Detect collisions and retransmit later.

D. Something else.

Link Layer Functions

1. Addressing: identifying endpoints

2. Framing: Dividing data into pieces that are sized for the network to handle.

3. Link access: Determining how to share the medium, who gets to send, and for how long.

4. Error detection/correction and reliability.

Reliability in the link layer seems at odds with the E2E principle. Why would we add reliability here?

A. Legacy reasons: reliability was done at the link layer first, E2E came later.

B. It improves performance.

C. It's necessary for correctness.

D. Some other reason.

E. It's completely unnecessary.