Traceroute Project Guide

CS 168 @ UC Berkeley

Slides credit: Sylvia Ratnasamy, Rob Shakir, Peyrin Kao

Last updated: Fall 2024

Protocols

Traceroute Project Guide, CS 168

Protocols

The Internet is all about designing **protocols**.

- Protocol: A specification on how to communicate.
  - Syntax: Format of messages. What do the 1s and 0s mean?
  - Semantics: What actions should I take in response to certain messages?
- Example: Protocol for asking a question in lecture?
  - Raise your hand.
  - Wait for speaker to call on you.
  - Ask your question after speaker calls on you.
  - If speaker doesn't see you after some time, say "Excuse me!"

Building the Internet

Traceroute Project Guide, CS 168

Layer 1 – Moving Bits Across Space

We need some **physical** technology to move data across space.

- Postal analogy: Mailman, Pony Express, carrier pigeon, etc.

IP over Avian Carriers

From Wikipedia, the free encyclopedia

In computer networking, **IP over Avian Carriers** (**IPoAC**) is a proposal to carry Internet Protocol (IP) traffic by birds such as homing pigeons. IP over Avian Carriers was initially described in RFC 1149, a Request for Comments (RFC) issued by the Internet Engineering Task Force (IETF), written by D. Waitzman, and released on April 1, 1990. It is one of several April Fools' Day Request for Comments.

Waitzman described an improvement of his protocol in RFC 2549, *IP over Avian Carriers with Quality of Service* (1 April 1999). Later, in RFC 6214—released on 1 April 2011, and 13 years after the introduction of IPv6—Brian Carpenter and Robert Hinden published *Adaptation of RFC 1149 for IPv6*.[1]

IPoAC has been successfully implemented, but for only nine packets of data, with a packet loss ratio of 55% (due to operator error),[2] and a response time ranging from 3,000 seconds (≈50 minutes) to over 6,000 seconds (≈1.77 hours). Thus, this technology suffers from poor latency. Nevertheless, for large transfers, avian carriers are capable of high average throughput when carrying flash memory devices, effectively implementing a sneakernet. During the last 20 years, the information density of storage media and thus the bandwidth of an avian carrier has increased 3 times as fast as the bandwidth of the Internet.[3] IPoAC may achieve bandwidth peaks of orders of magnitude more than the Internet when used with multiple avian carriers in rural areas. For example: If 16 homing pigeons are given eight 512 GB SD cards each, and take an hour to reach their destination, the throughput of the transfer would be 145.6 Gbit/s, excluding transfer to and from the SD cards.

Are pigeons faster than the Internet?

Under RFC 1149, a homing pigeon (exemplar in Scheßlitz) can carry Internet Protocol traffic.

Layer 1 – Moving Bits Across Space

We need some **physical** technology to move bits across space.

- Voltages on electrical wire.
- Light signals on optical fiber.
- Wireless radio waves.

Won't go into detail in this class.

Risks [edit]

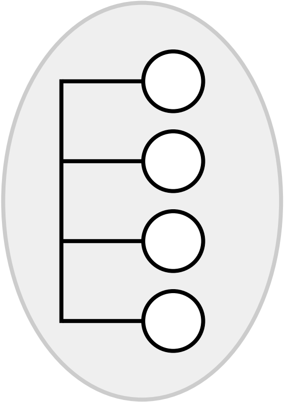
Although collisions are unlikely, packets can be lost, particularly to raptors.

Layer 2 – Local Networks

Postal analogy: Use our physical technology to connect everybody in the local town.

Forming a local network:

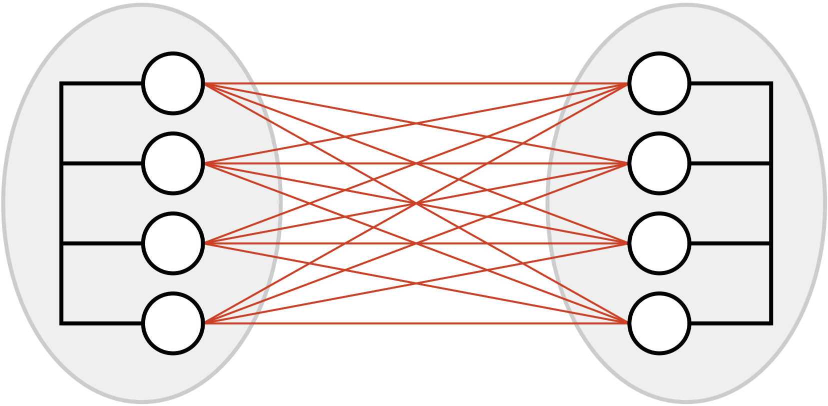
- Use physical technology to create a **link** between machines.
- Use links to connect all machines in a local area.
- Machines can exchange **packets**: A group of bits representing a message.



Layer 3 – Connecting Local Networks

Postal analogy: How do we connect houses from different towns?

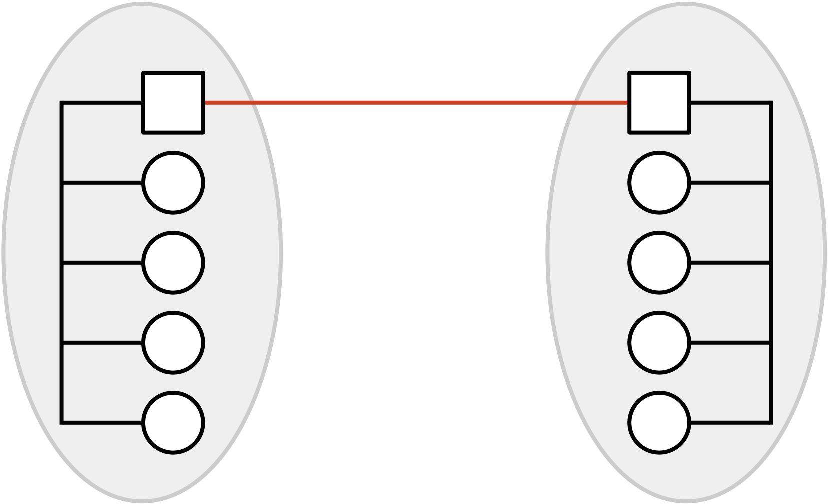
- Adding new links between every pair of houses is inefficient.



Layer 3 – Connecting Local Networks

Postal analogy: How do we connect houses from different towns?

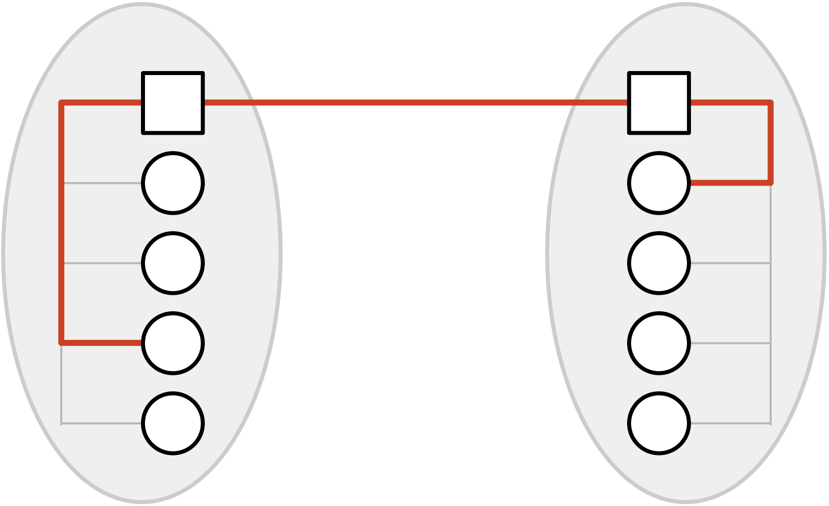
- Solution: Introduce a *post office* in each town.
- Just connect the two post offices.



Layer 3 – Connecting Local Networks

To send a letter to the other town:

- You send the packet to...
- Your local post office, which sends the packet to...
- The other town's post office, which sends the packet to...
- The final destination.



Properties of the Internet

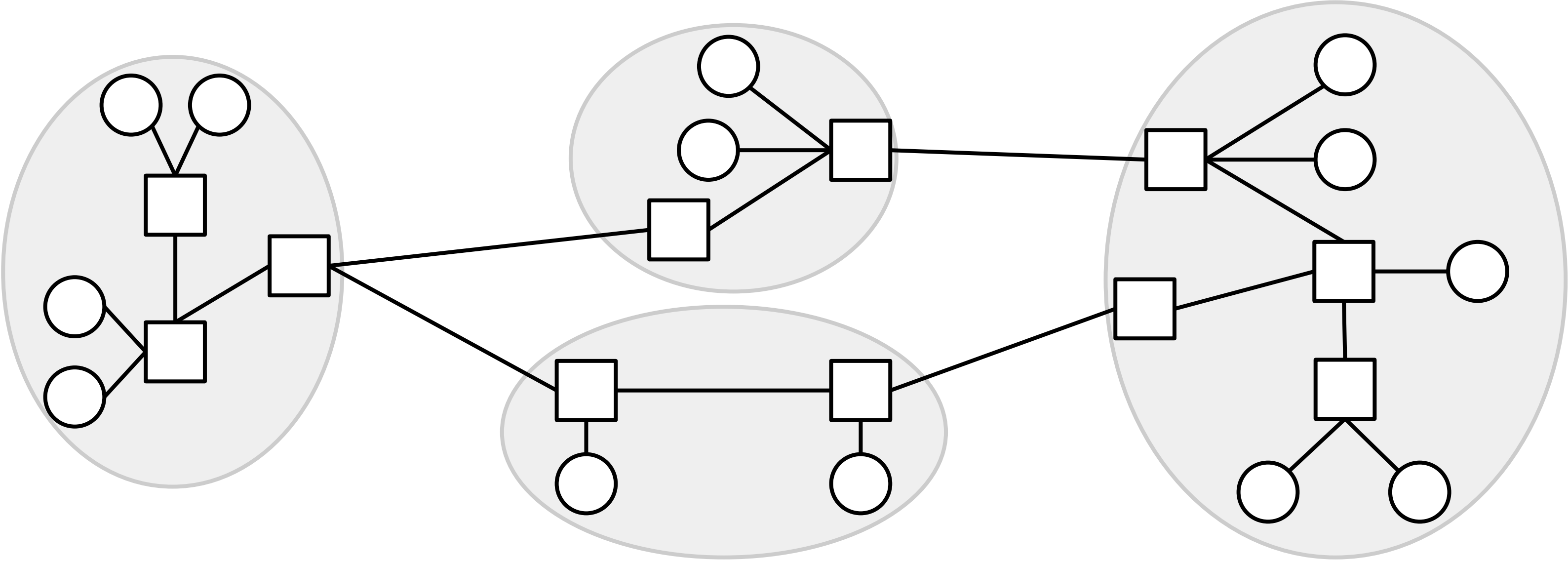
Traceroute Project Guide, CS 168

Network of Networks

With enough post offices, we can connect all the towns in the world!

The Internet is a **network of networks**.

- Each operator runs its own local network.
- The local networks connect to each other to form the Internet.



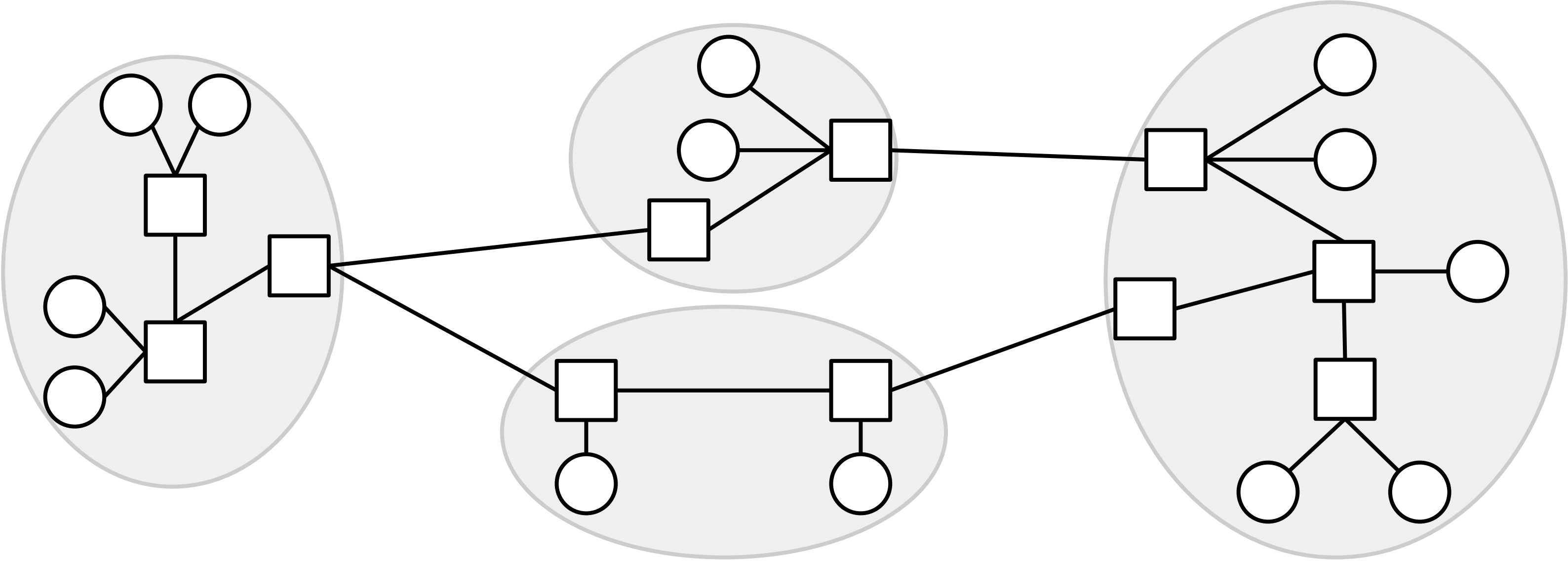
Hosts vs. Switches

End hosts are the machines communicating over the Internet.

- Analogy: Houses.
- Examples: Your laptop, your phone, Google's server.

Switches (aka **routers**) receive packets and forward them toward their destination.

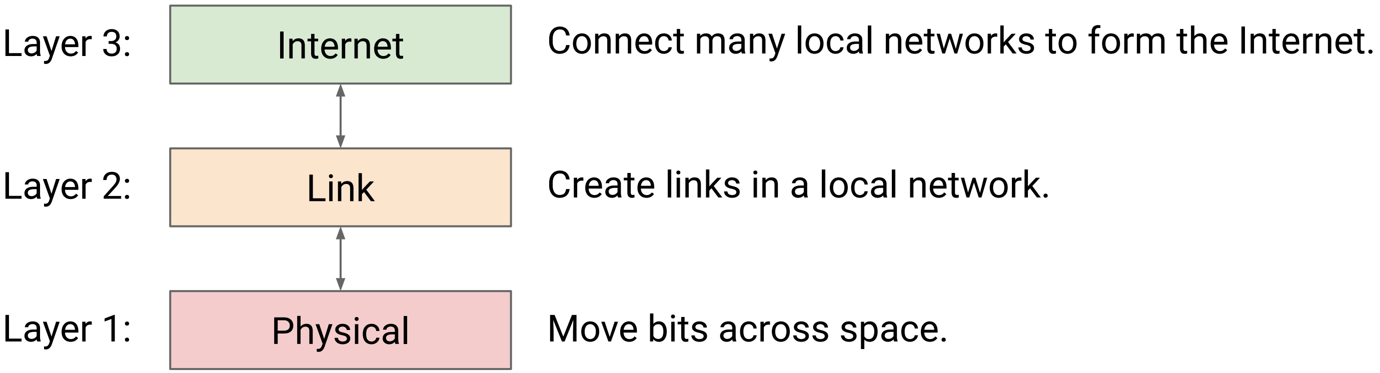
- Analogy: Post offices.



Layers of Abstraction

Modularity: In our design, we decomposed the system into layers of abstraction.

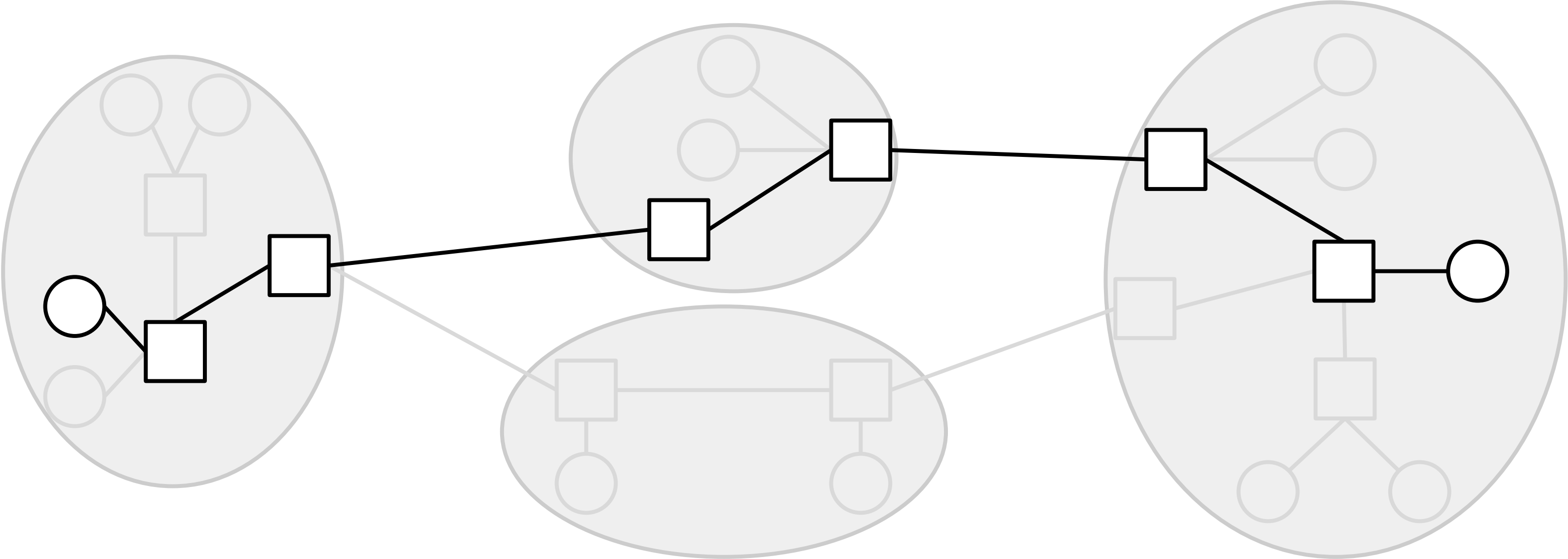
- Each layer relies on services from the layer below.
- Each layer provides services to the layer above.



Global Delivery at Layer 3

A packet can take multiple **hops** to reach its destination.

- Each router needs to **forward** the packet closer to its destination.



Layer 3 is Best-Effort

Layer 3 offers a **best-effort** service model.

- Packets are limited in size.
- Packets could get lost, reordered, corrupted, etc.
- The network will try its best to deliver your packet, but no guarantee.
- The network won't tell you if the delivery failed.

We need to build more layers if we want to guarantee packet delivery.

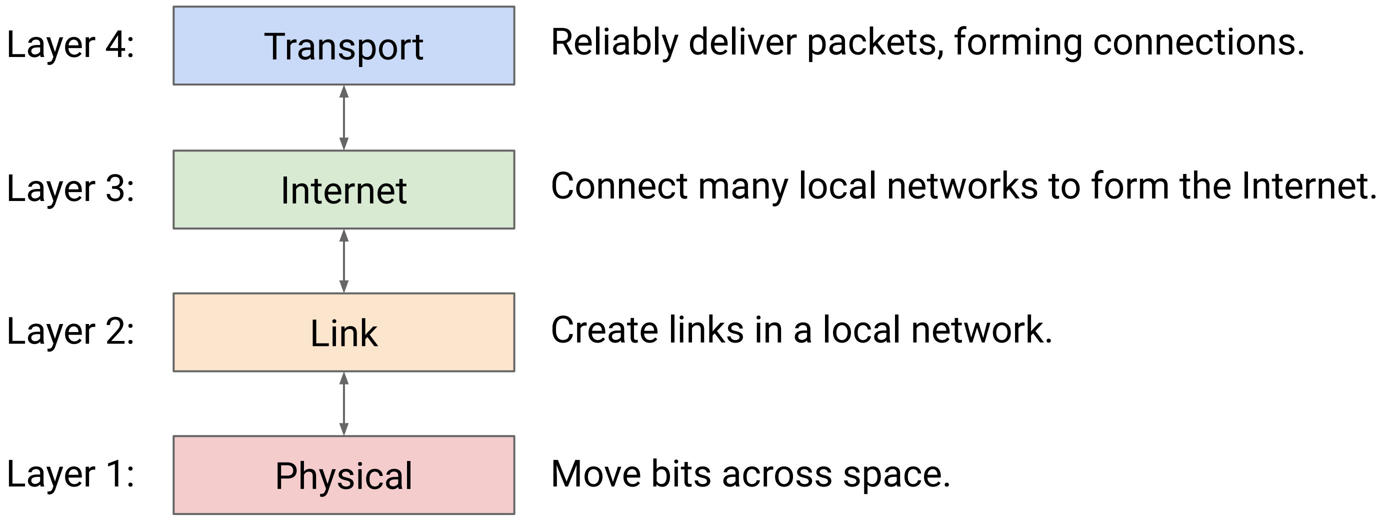
Higher Layers

Traceroute Project Guide, CS 168

Layer 4 – Reliability

Transport layer builds on top of Layer 3 (global packet delivery).

- Adds extra mechanisms (e.g. re-sending lost packets) for reliable packet delivery.
- Splits up large data into packets to send them. Reassembles received packets.
- Instead of individual packets, can think about **flows** (aka **connections**): A stream of packets exchanged between two endpoints.

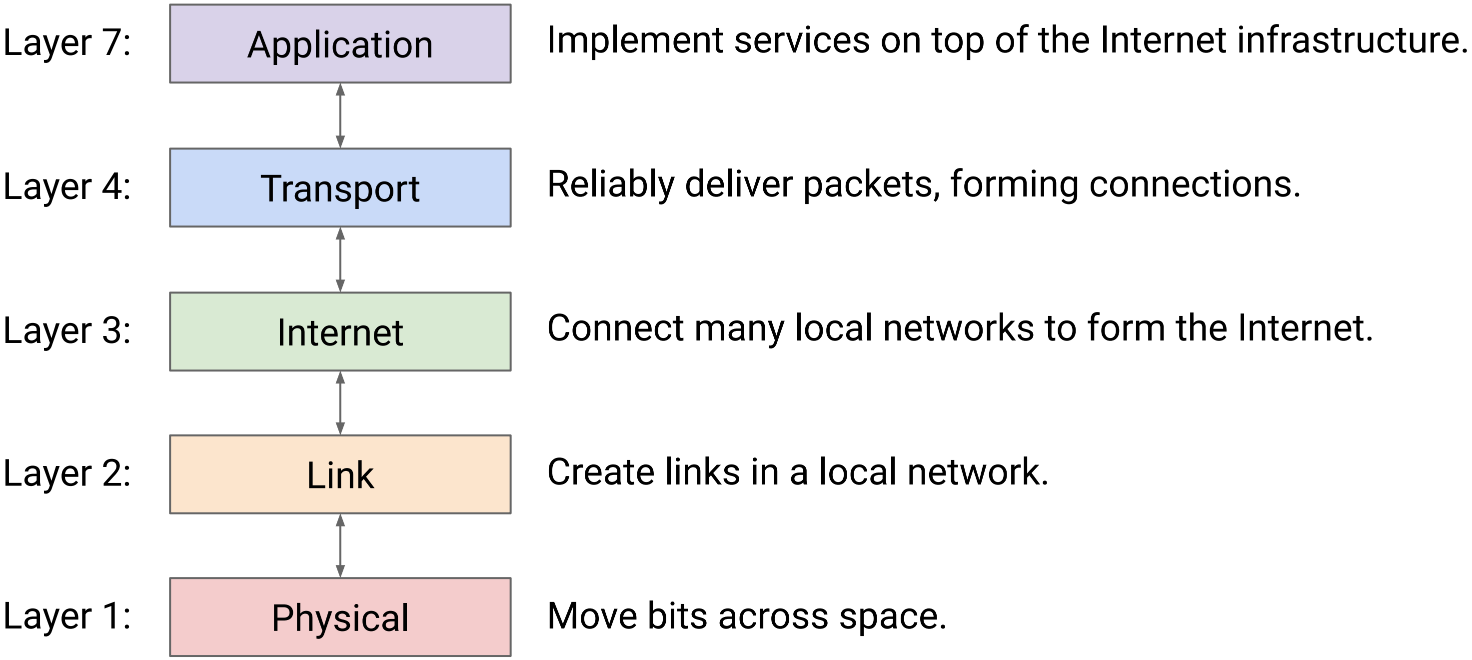


Layer 7 – Application

Application layer builds services (e.g. websites, video streaming) on top of Layer 4.

- This design lets us build different services, all on the same infrastructure.

Note: Layers 5 and 6 are now obsolete.



Headers

Traceroute Project Guide, CS 168

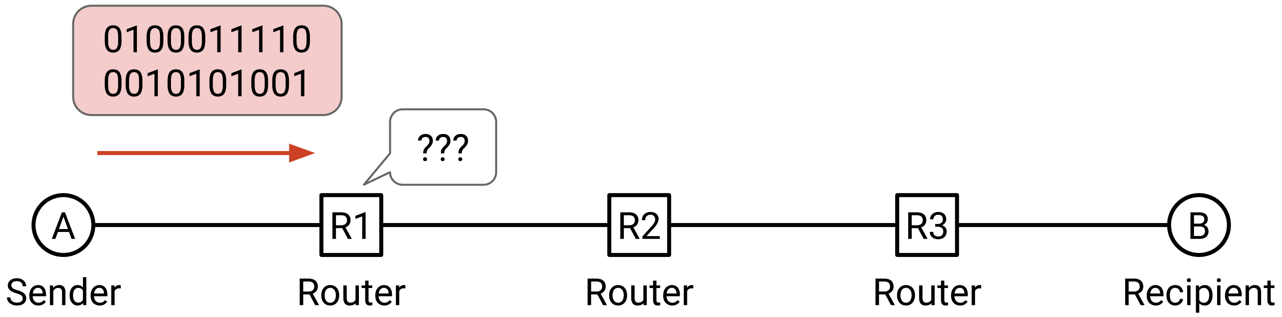
Why Do We Need Headers?

Suppose A wants to send an image to B.

- A forms a packet with the bits of the image. *(May need to split image into multiple packets.)*
- A sends the packet to the next router.
- The router has no idea what these bits are for!

The packet needs some extra **metadata**, to tell us what to do with the packet.

- Analogy: Letter needs to be put in an envelope.
  Envelope describes what to do with the letter.



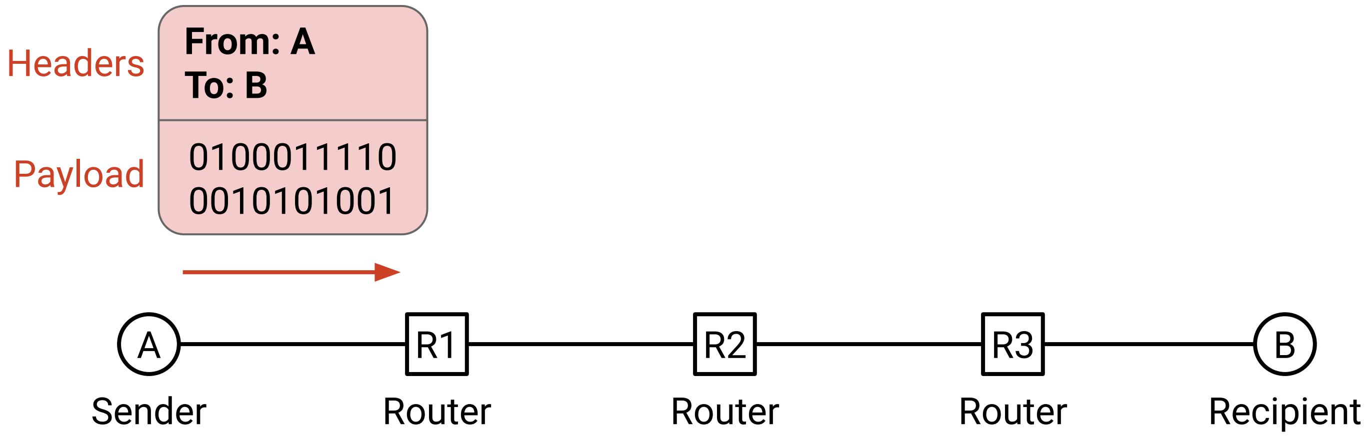
Common Header Fields

The packet **header** contains metadata describing how the data should be sent.

Some common fields in a header:

- Destination address: Required to deliver the packet.
- Source address: Useful if the recipient wants to send replies back.

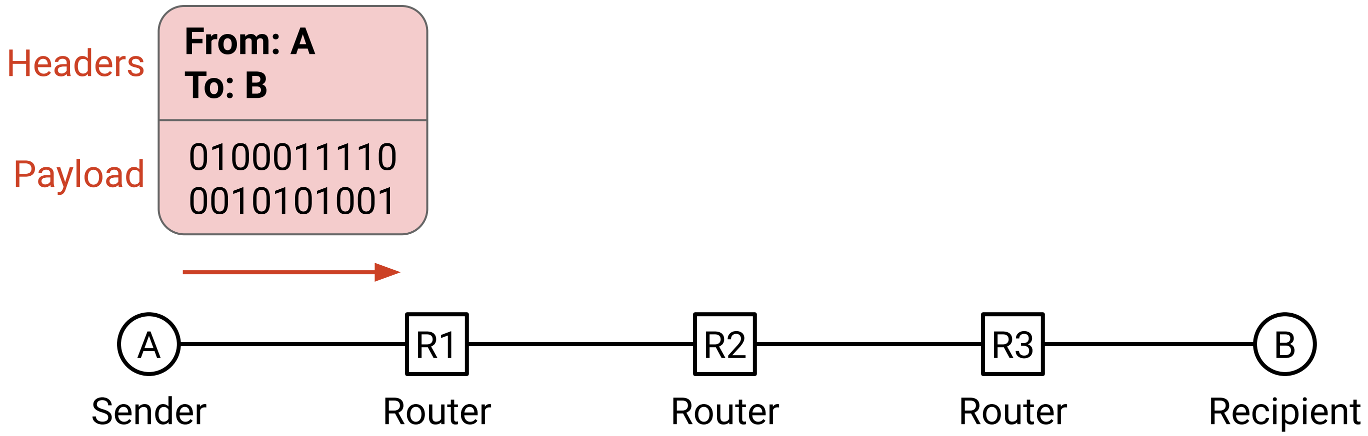
The actual data in the packet is called the **payload**.



Headers are Standardized

Everybody needs to agree on the format of the header.

- "First 8 bits are the source, next 8 bits are the destination..."
- If we use a different format, others won't understand the header.



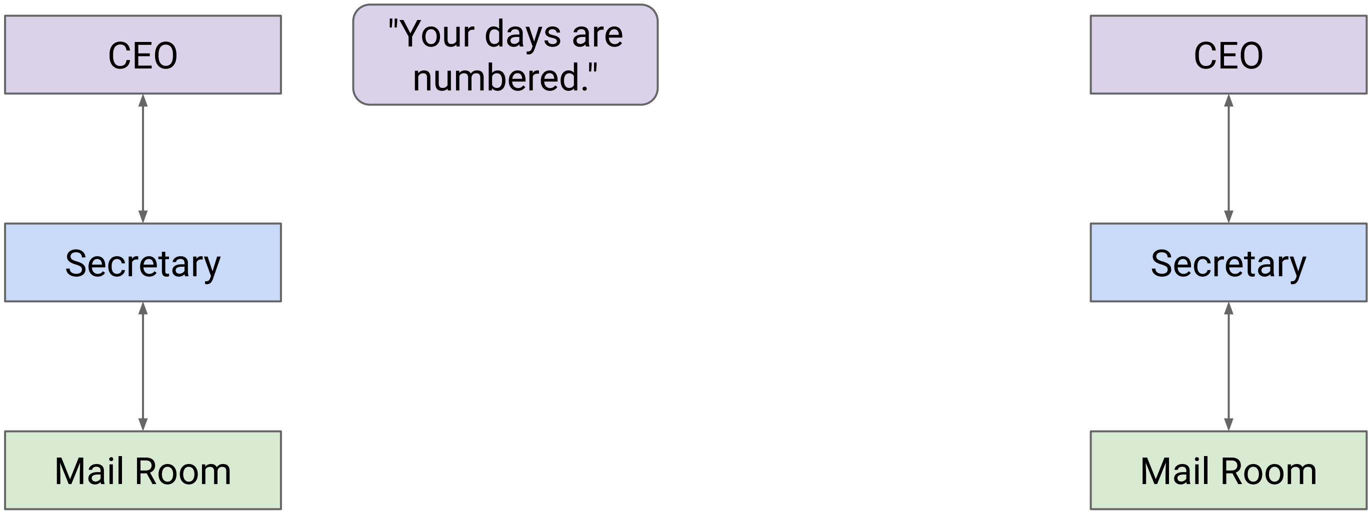
Multiple Headers

Traceroute Project Guide, CS 168

Postal Analogy

CEO Alice wants to send a message to CEO Bob.

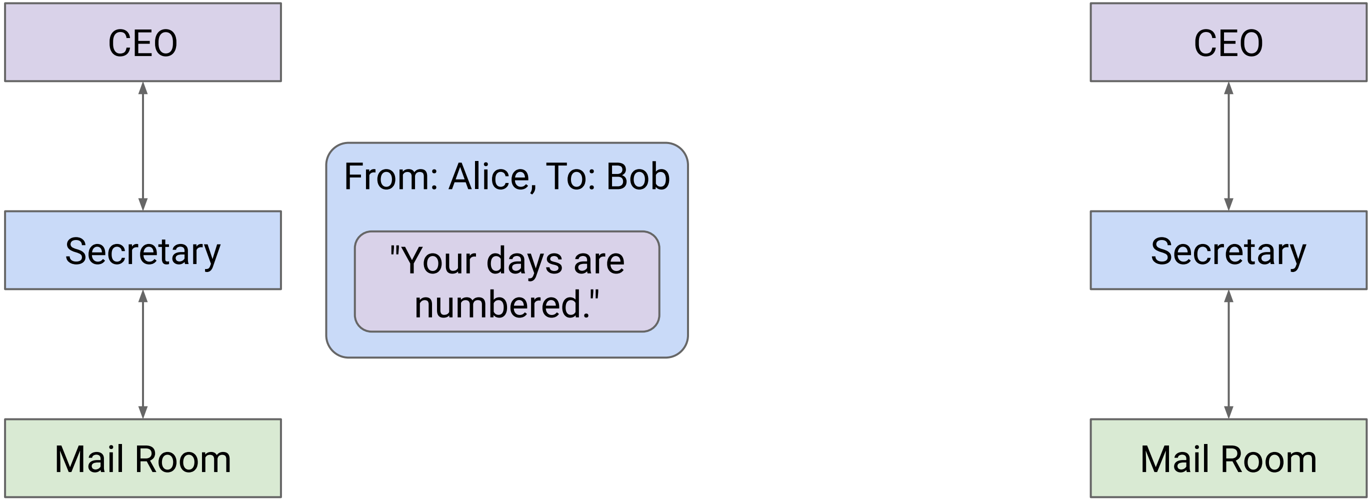
Alice writes a letter.



Postal Analogy

Alice passes the letter down to her secretary.

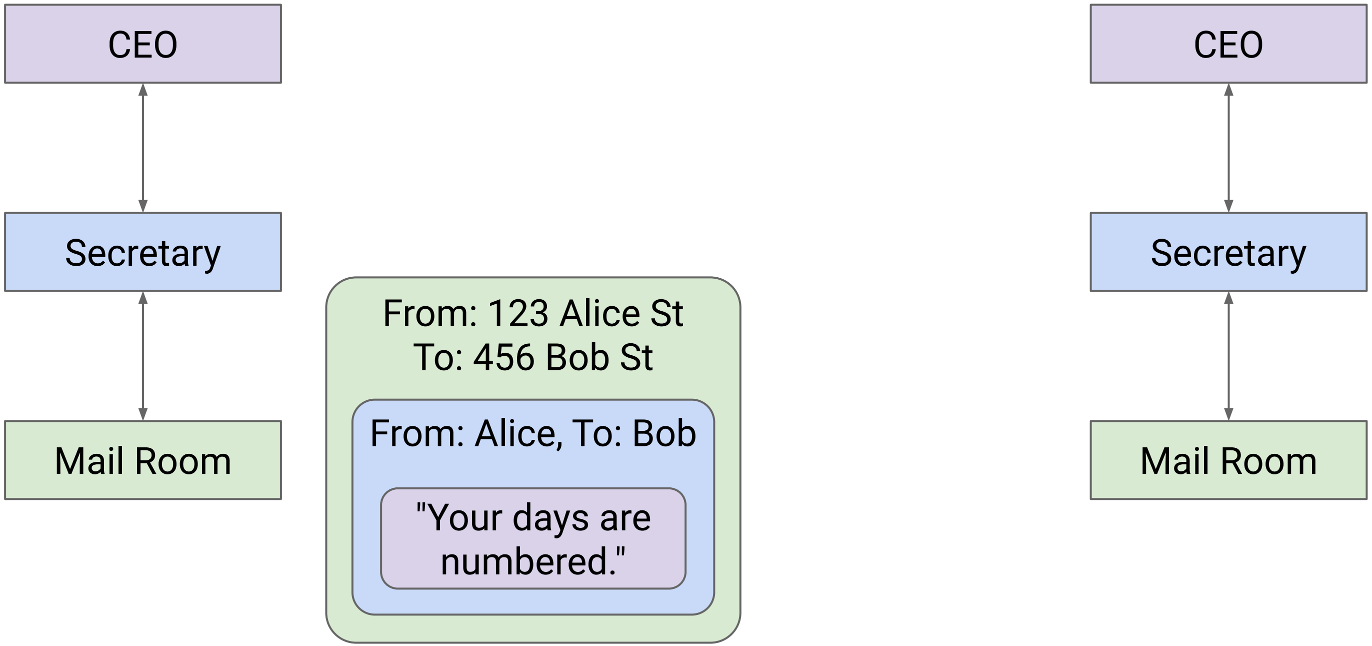
Her secretary puts the letter in an envelope.



Postal Analogy

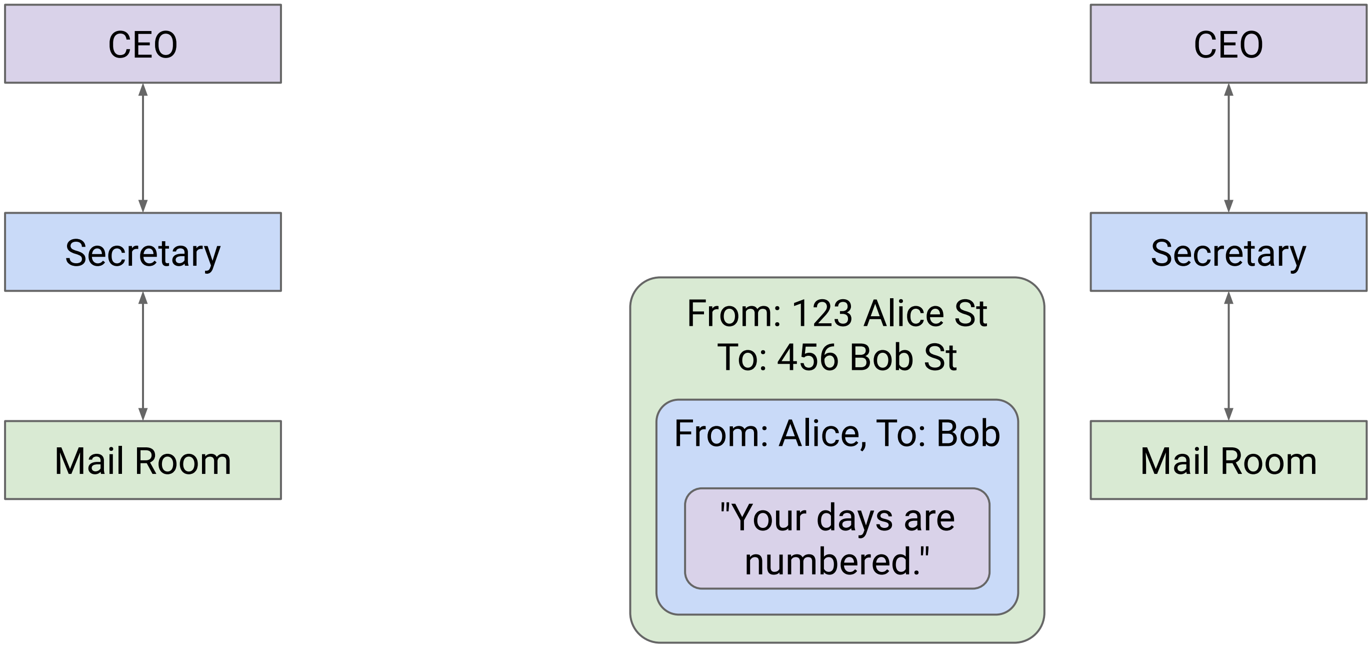
Her secretary passes the letter down to the mailman.

The mailman puts the envelope in a box.



Postal Analogy

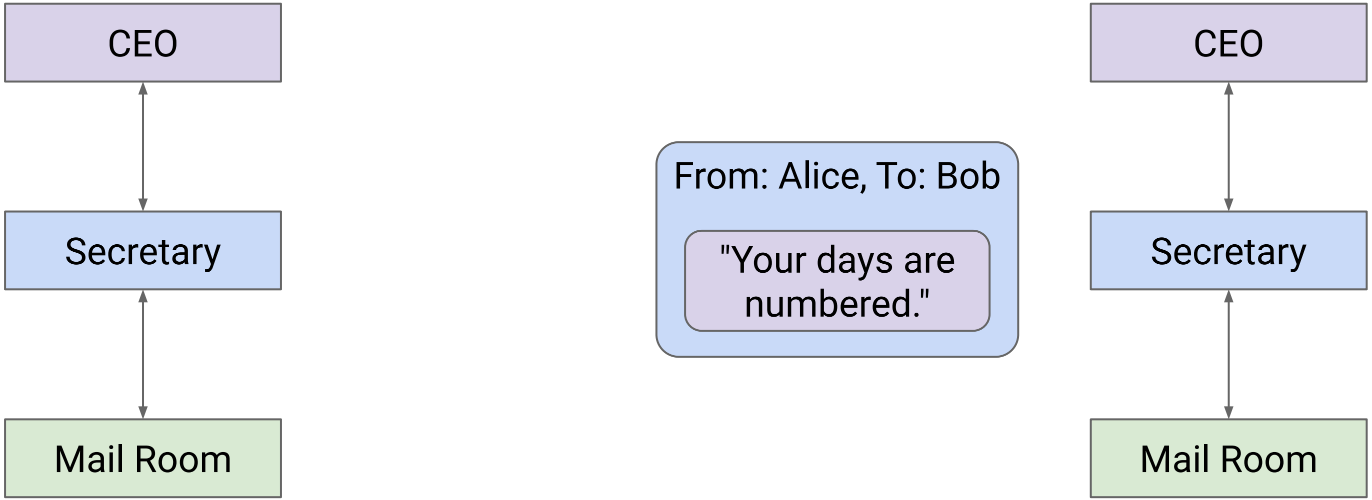
The packet travels through the postal system, to Bob's building.



Postal Analogy

The mailman *unwraps* the box, revealing the envelope inside.

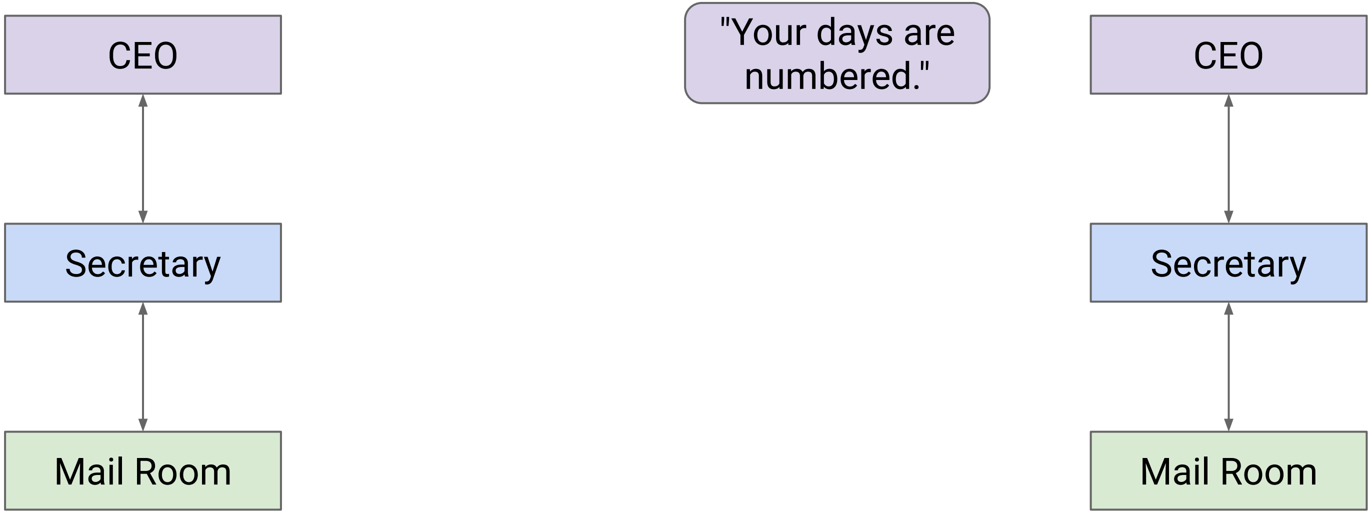
The mailman passes the envelope up to the secretary.



Postal Analogy

The secretary *unwraps* the envelope, revealing the letter inside.

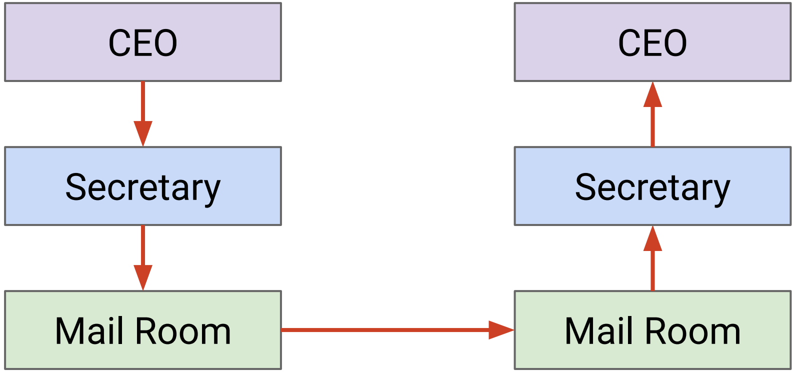
The secretary passes the letter up to Bob.

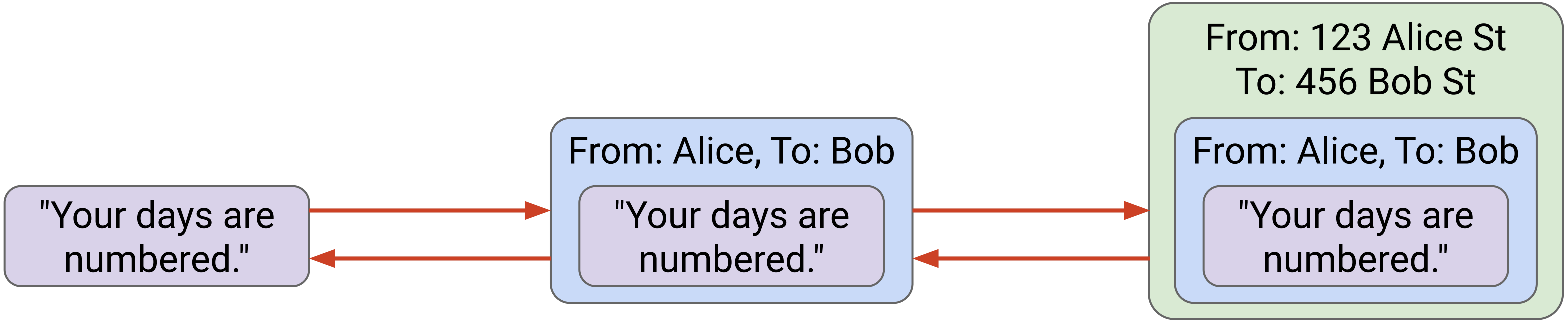


Postal Analogy

As we move to lower layers, we wrap additional headers around the packet.

As we move to higher layers, we peel off headers, revealing the inner headers.





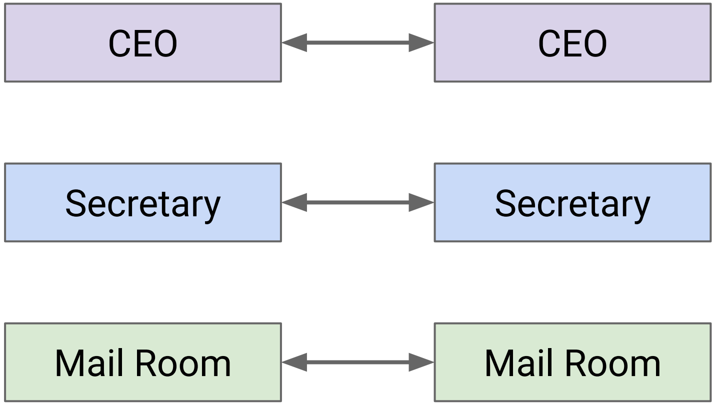
Postal Analogy

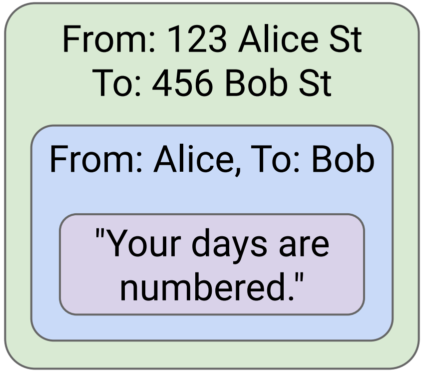
Each person only cares about the headers at their layer.

- Mailman reads the green header, ignores all the payload inside.

Each person communicates with its peers at the same layer.

- Alice's secretary writes the blue header, for Bob's secretary to read.
- A protocol at a specific layer only makes sense to people at that layer.





Mailman only cares about this.

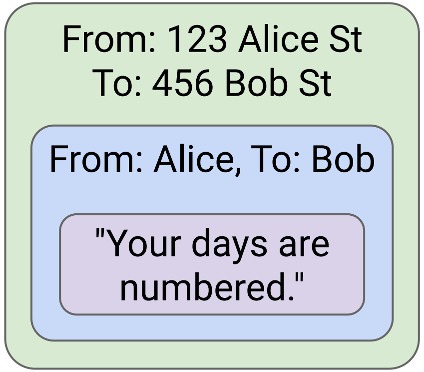
Secretary only cares about this.

CEO only cares about this.

Addressing at Different Layers

Notice: Different layers use different addressing schemes.

- Inside a building: "413 Soda Hall."
- In the postal system: "2551 Hearst Ave, Berkeley, CA."

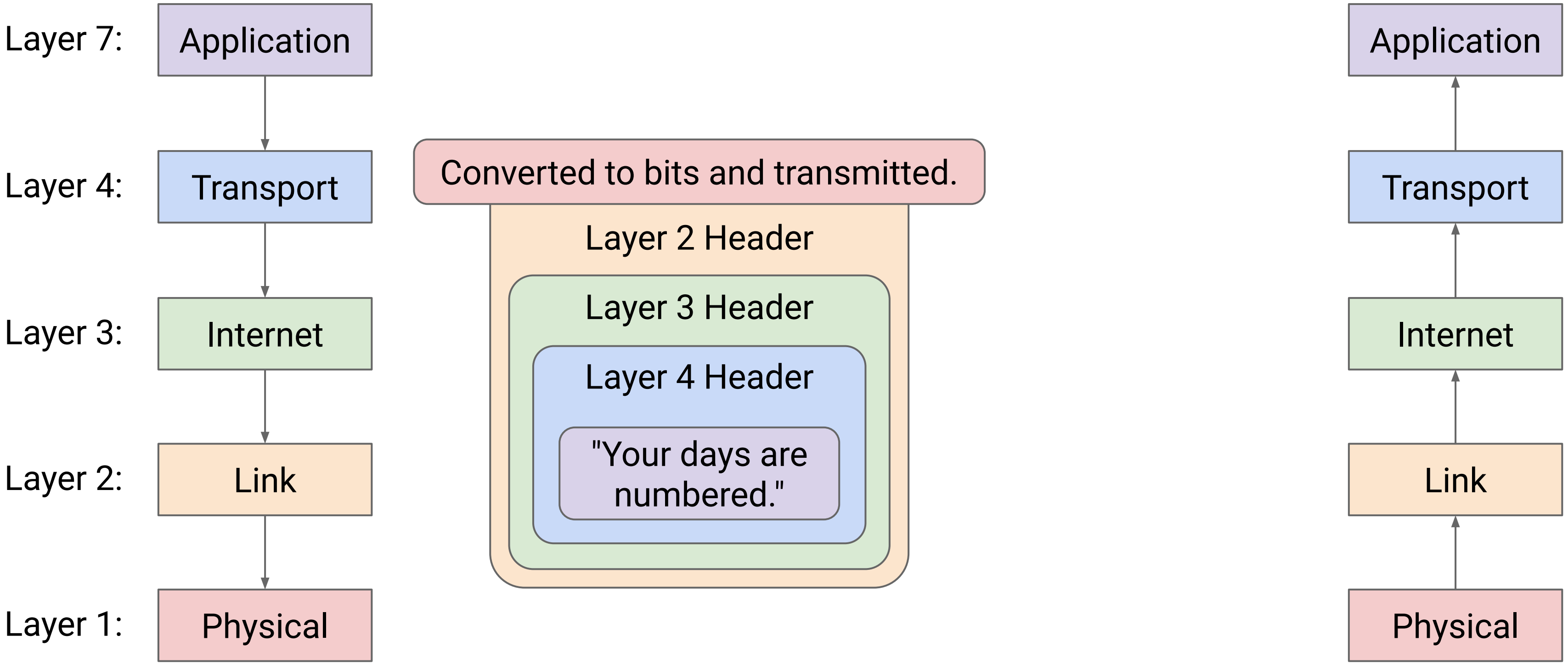


These addresses make sense to the mailman.

These names make sense to the secretary.

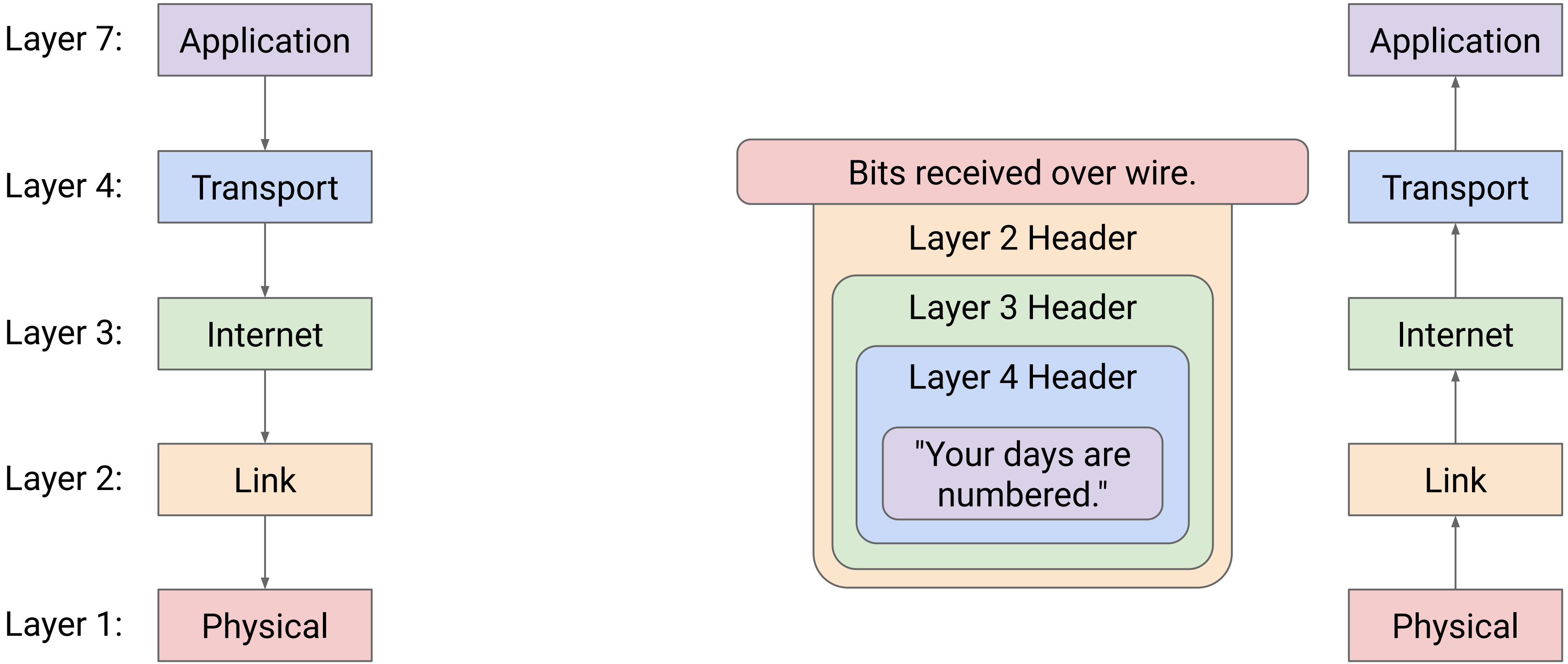
Multiple Headers

As we move to lower layers, we wrap additional headers around the packet.



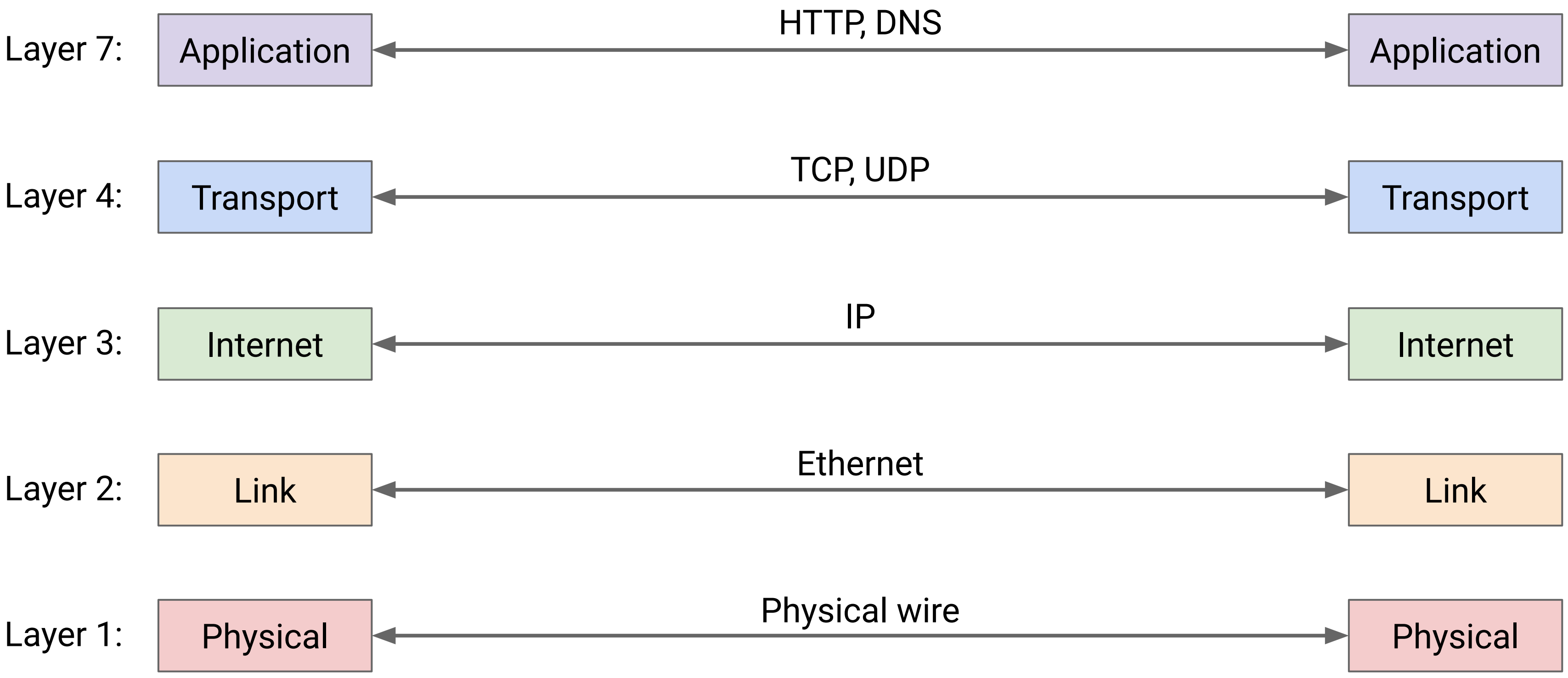
Multiple Headers

As we move to higher layers, we peel off headers, revealing the inner headers.



Multiple Headers

Peers at the same layer communicate with each other using the header at that layer.



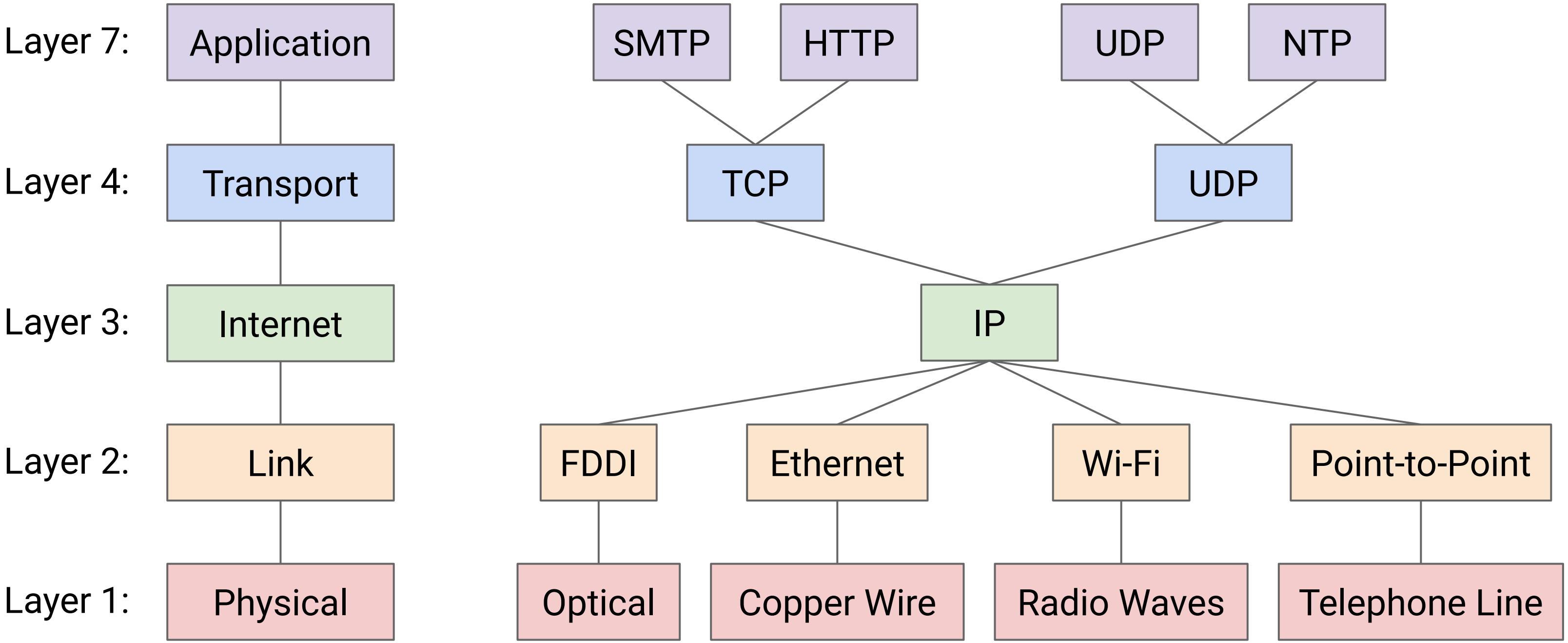
Demultiplexing

Traceroute Project Guide, CS 168

Protocols at Different Internet Layers

Multiple protocols exist at each layer.

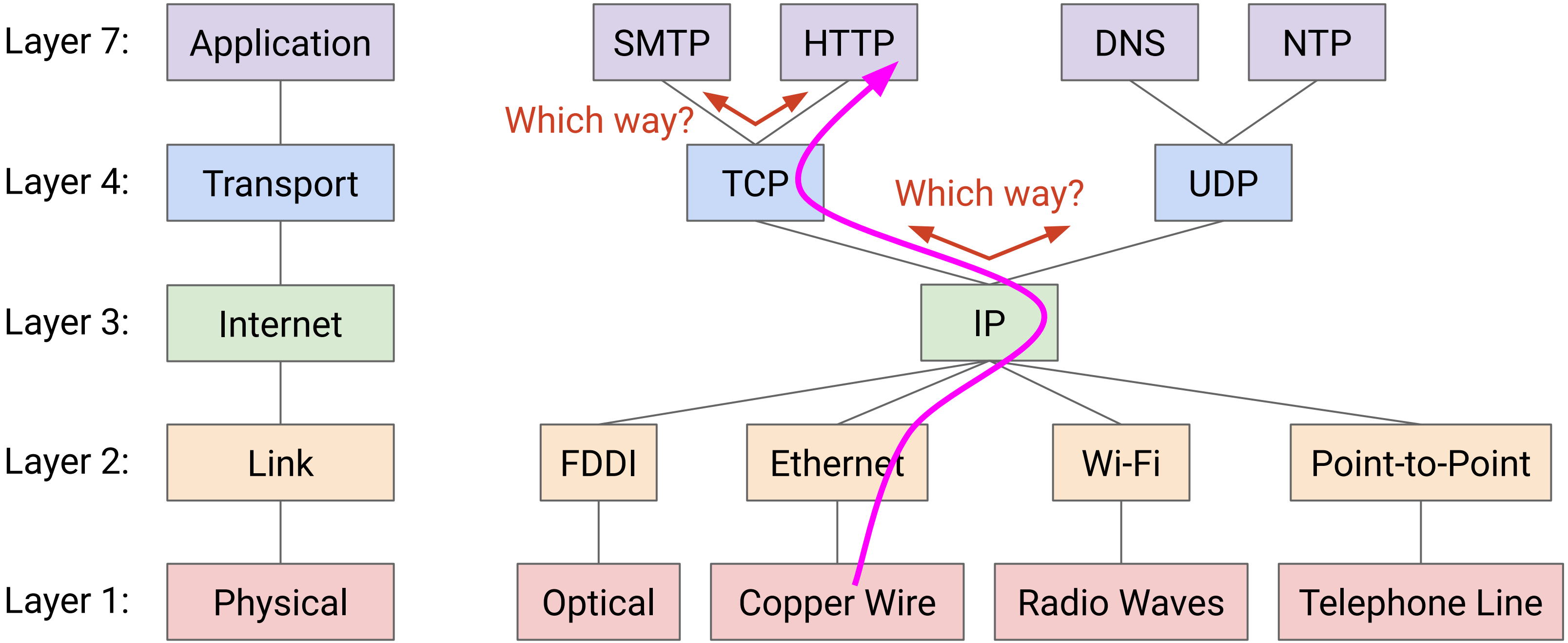
- End hosts can agree on the L4 and L7 protocols they want to use.
- Routers on each link can agree on the L1 and L2 protocols they want to use.



Demultiplexing

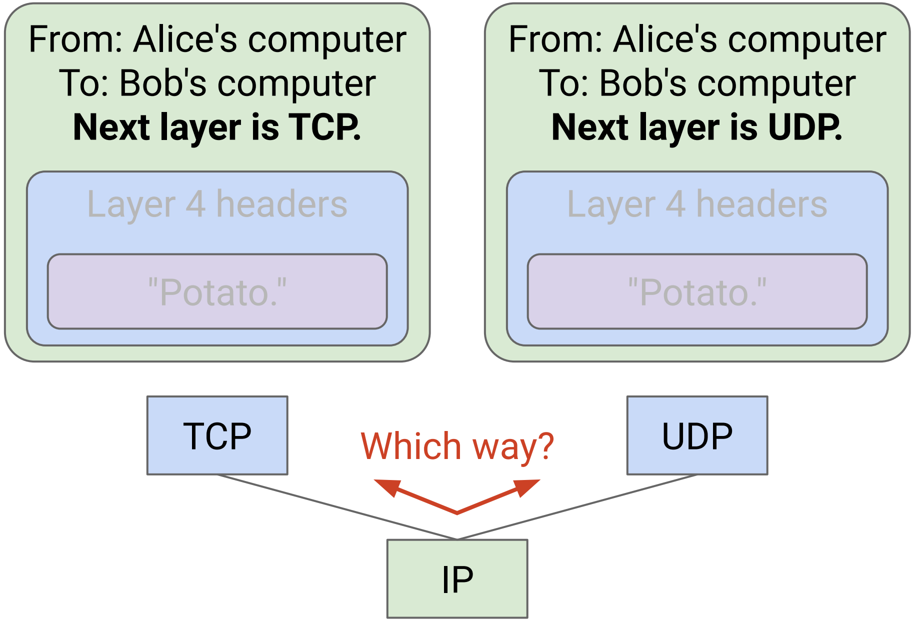
When you receive a packet, you pass it up the stack, to higher-layer protocols.

- How did IP know to pass up to TCP, not UDP?
- How did TCP know to pass up to HTTP, not SMTP?



Demultiplexing:

- Add a new header field that tells us what the next (higher) layer protocol is.
- Allows the IP code to pass the rest of the packet to the appropriate L4 code.



Demultiplexing with Ports

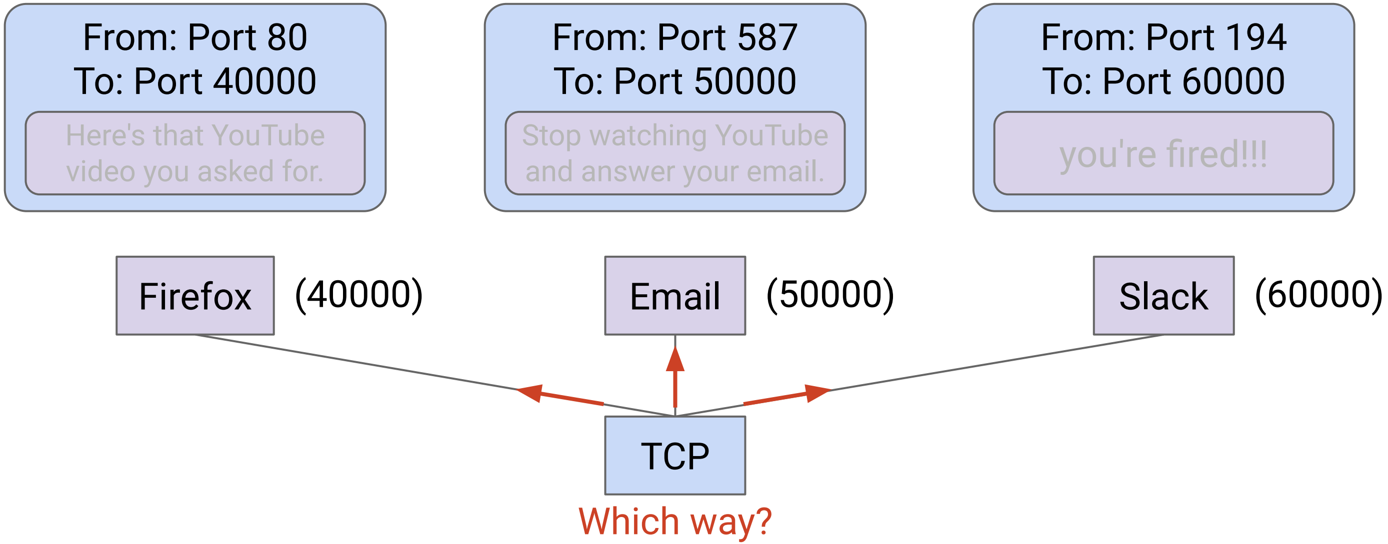
Traceroute Project Guide, CS 168

Demultiplexing at Layer 4

Demultiplexing also works at Layer 4.

More specifically, each open connection on your computer.

- Each running application on your computer is associated with a **port number**.
- When L4 receives a packet, it uses the port number to pass the packet to the corresponding application.



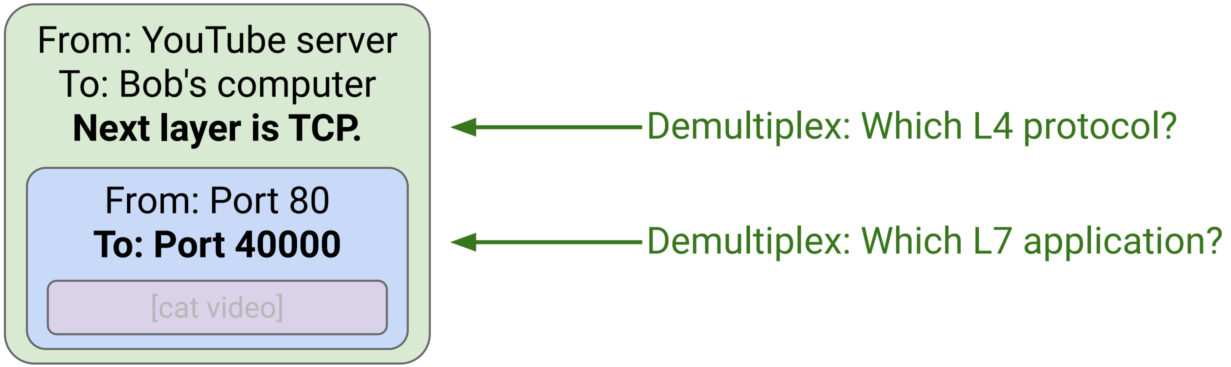
Ports at Layer 4

Port numbers help us distinguish between applications on the same computer.

- IP address (Layer 3) for all the applications is the same.
- But each connection is associated with a different port number.

Analogy: Room numbers.

- You and your housemate both have the same street address.
- If someone sends a letter to your house, who is it for?
- Distinguish by assigning room numbers to each housemate.



Ports at Layer 4

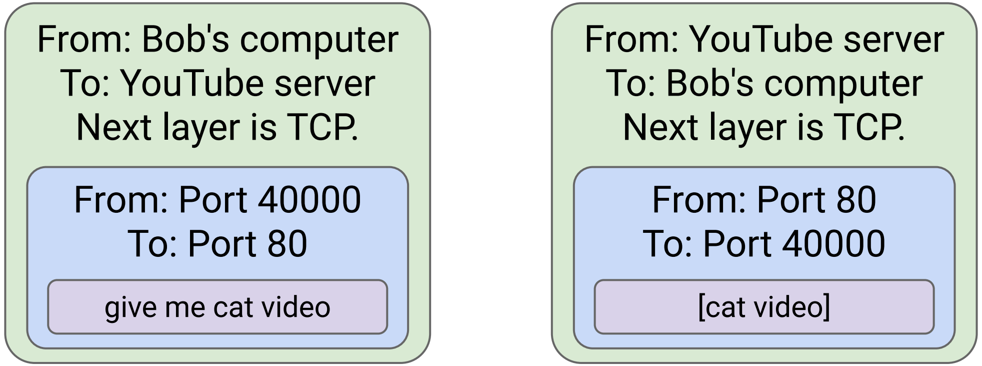
Both end hosts in a connection have a port number.

- A private client (e.g. your computer) can use a randomly-generated port number.
- A public server (e.g. YouTube) must use a fixed, well-known port number.

Analogy: Room numbers.

- Pick any number for your bedroom. No one cares.
- Public room numbers (e.g. in Soda Hall) must be fixed and well-known.

Outgoing packet: Bob picks a random port number, but sends to YouTube's fixed port, 80.

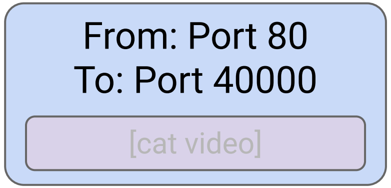


Incoming reply: YouTube replies to Bob's chosen port. Bob's computer passes the packet to the correct application (Firefox, not Slack).

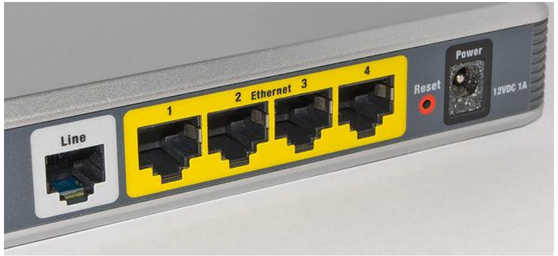
Caution – Terminology Conflict

In networking, there are two different things, both called "ports."

- If it's unclear, we will specify "logical port" or "physical port."



Logical port: A number identifying an application. Exists in software.



Physical port: The hole you plug a cable into. Exists in hardware.

Under construction,
should be done by
September 1, 2024 or so.

Time-to-Live

Traceroute Project Guide, CS 168