Fall 2018-2019: EE471/CS471/CS573

Computer Networks: Principles & Practices

Slide set 03

Tariq Jadoon and Zartash Afzal Uzmi
SBA School of Science and Engineering
LUMS

Material with thanks to Jennifer Rexford, Sylvia Ratnasamy, Ion Stoica, K&R, Scott Shenker, Dave Anderson and others

Comparing similar systems: Mail delivery versus eMail delivery

Decomposition of tasks

Inspiration…

- CEO A writes letters to CEO B
  - Folds letters and hands those to the secretary
- Secretary:
  - Puts letters in envelopes with CEO B's full name
  - Takes to FedEx (or post office)
- FedEx Office
  - Puts letters in larger "**FedEx envelopes**"
  - Puts name and street address on FedEx envelopes
  - Puts packages on FedEx delivery truck
- FedEx delivers to other company

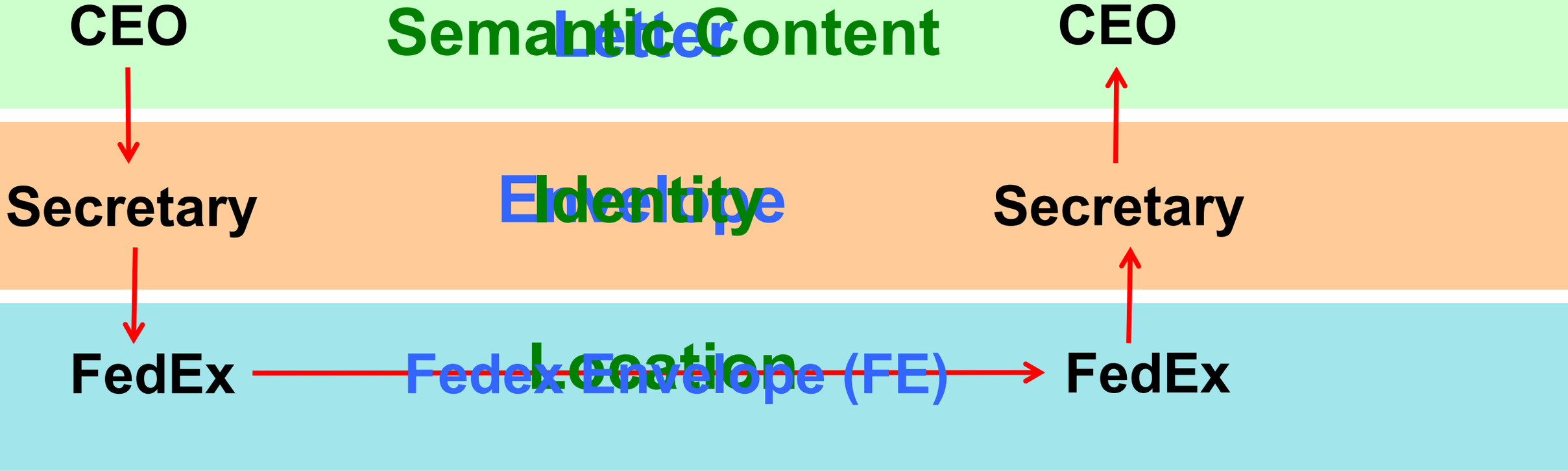
Dear John,

Your days are numbered.

--Pat

The Path of the Letter

- "Peers" in the same layer understand the same things
- No one else needs to
- Lowest level has **most packaging**



In the Internet: decomposition

Applications

Reliable (or unreliable) transport

Best-effort *global* packet delivery

Best-effort local *packet* delivery

Physical transfer of bits

In the Internet: organization

Applications

…built on… (using socket interface)

Reliable (or unreliable) transport

…built on…

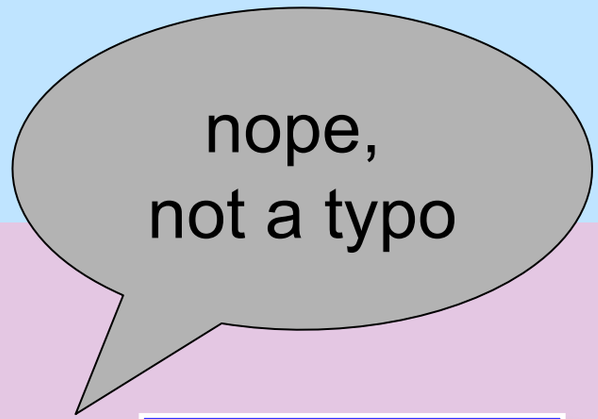
Best-effort global packet delivery

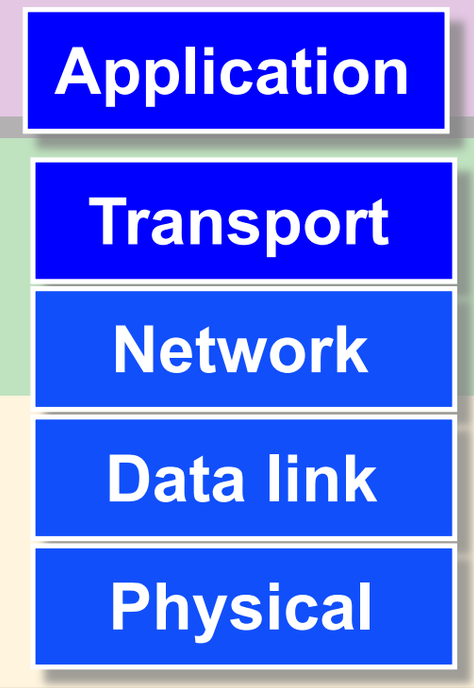
…built on…

Best-effort local packet delivery

…built on…

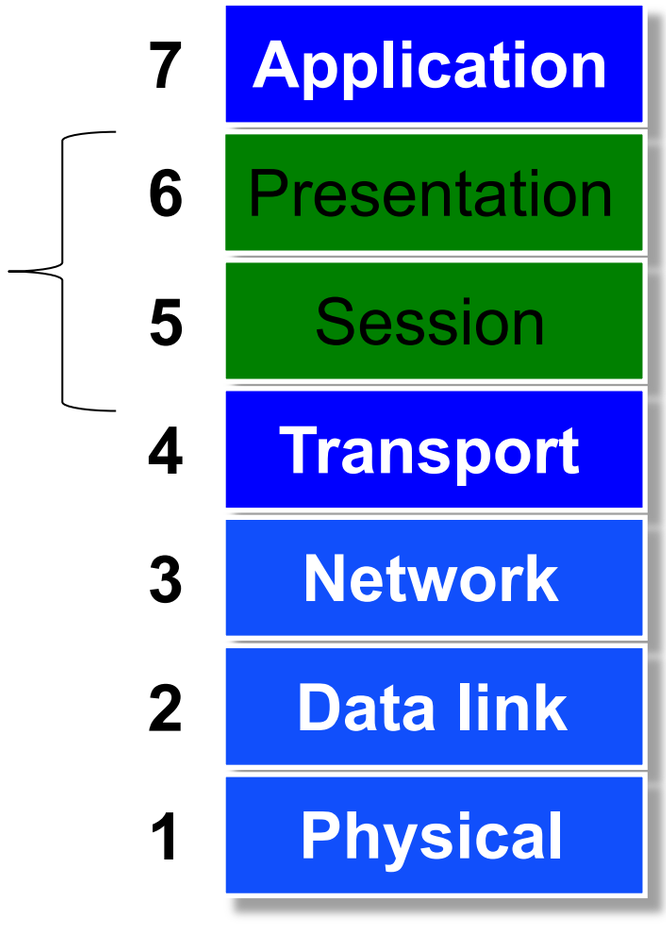
Physical transfer of bits





In the context of the Internet

The Open Systems Interconnect (OSI) model developed by the ISO included two additional layers that are often implemented as part of the application



Layers

- Layer = a part of a system with well-defined interfaces to other parts

- One layer interacts only with **layer above** and **layer below** (Does CEO talk to the FedEx guy?)

- Interaction only through the interface between them (e.g., **socket interface**, remember?)

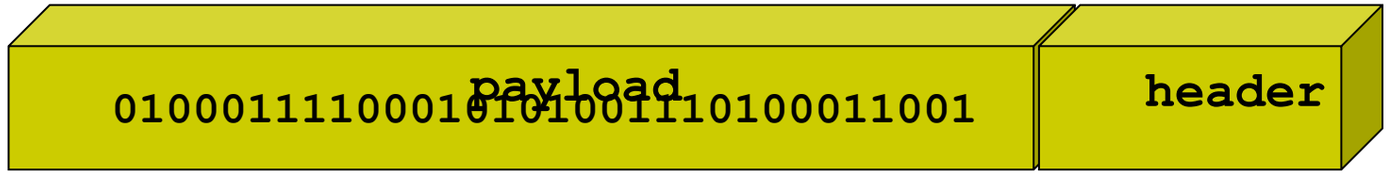
Protocols and Layers

Communication between **peer layers** on different systems is defined by **protocols**

What is a Protocol?

- An agreement between parties on how to communicate

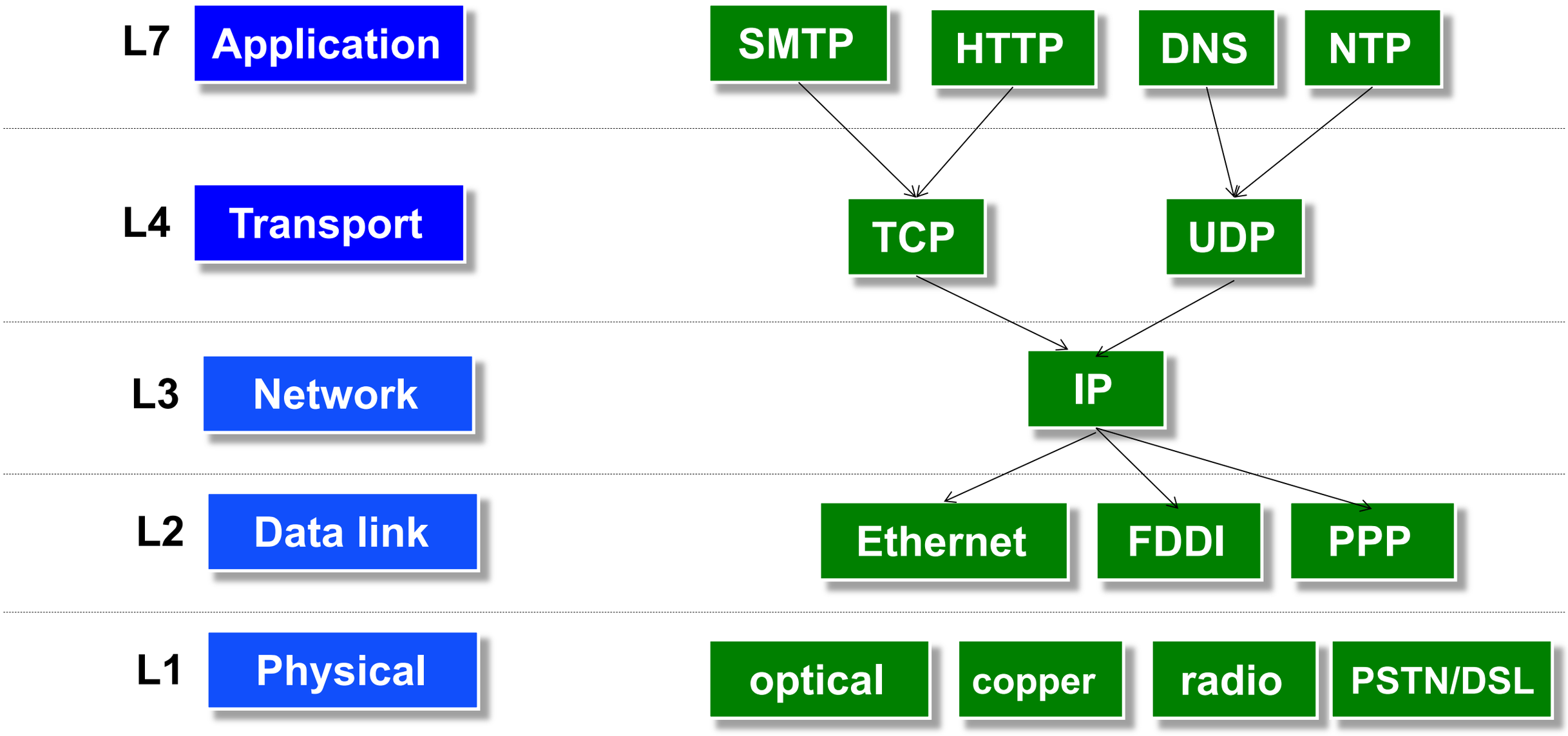
- Defines the syntax of communication
  - header → instructions for how to process the payload
  - Each protocol defines the format of its packet **headers**
    - *e.g. "the first 32 bits carry the destination address"*



What is a Protocol?

- An agreement between parties on how to communicate
- Defines the syntax of communication
- And semantics
  - “first a hullo, then a request…”
  - we’ll study many protocols later in the semester
- Protocols exist at many levels, hardware and software
  - defined by a variety of standards bodies (IETF, IEEE, ITU)

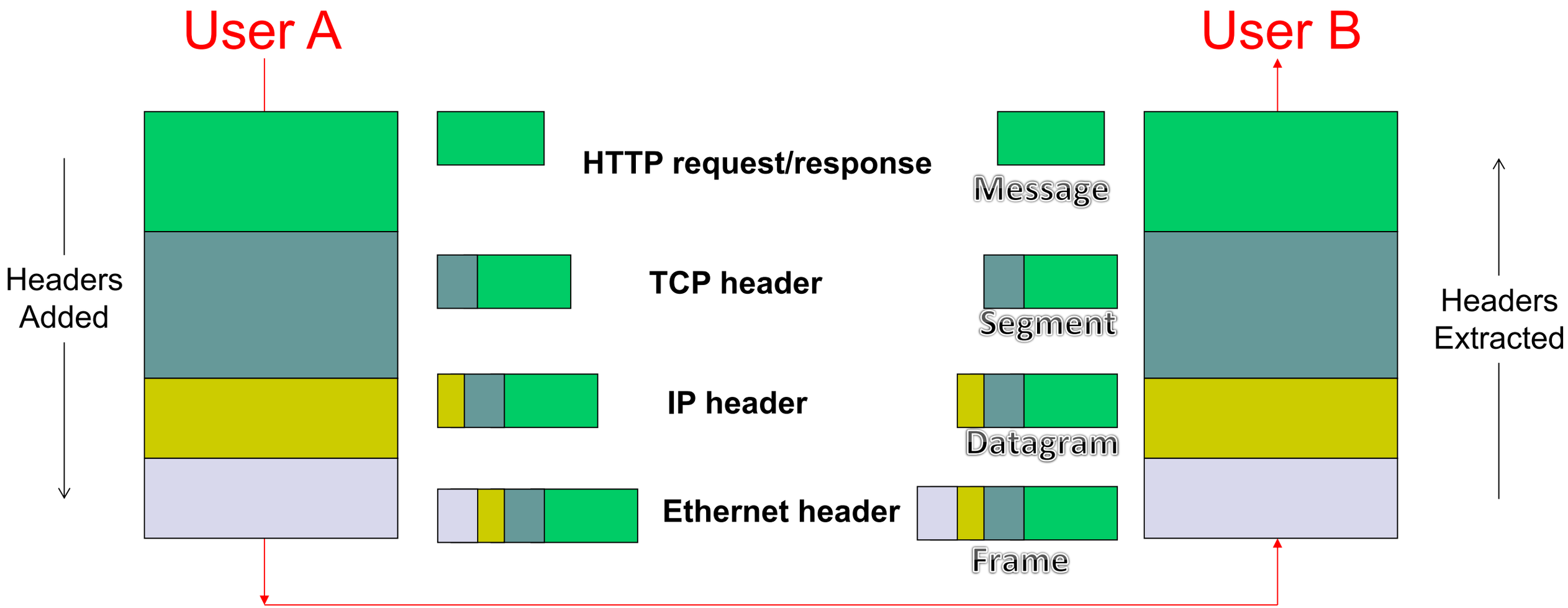
Protocols at different layers



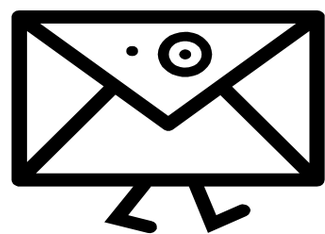
There is just one network-layer protocol!

Layer Encapsulation: Protocol Headers

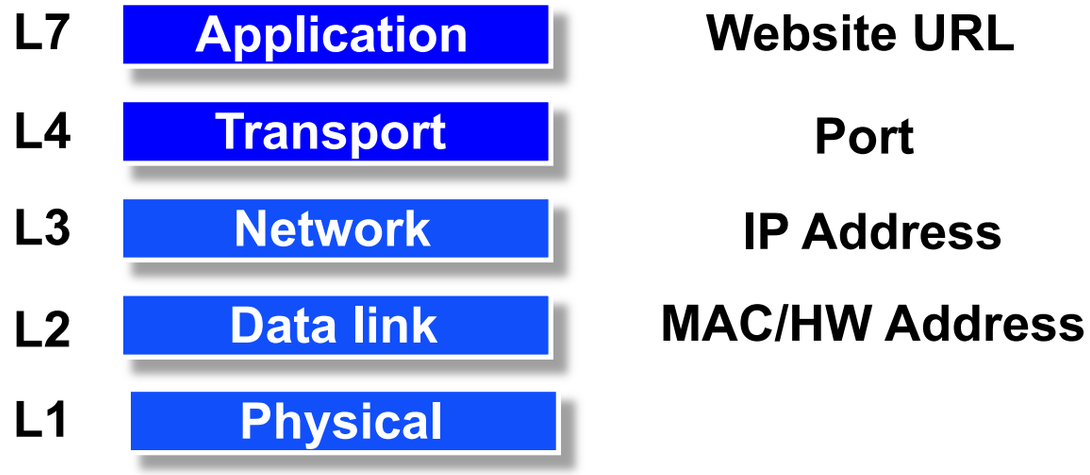
Just like a secretary adding address label of recipient



Packet == Message or Segment or Datagram or Frame



Addresses



- Peer layers use addresses to talk to each other
  - Buried within respective headers

- *IP addresses* at Network Layer (Layer 3)
- MAC or Physical or Hardware addresses at L2
- What about Transport layer (L4)?
  - Ports – a network application runs at a machine (IP) on a port
- Application layer also uses addresses (e.g., URLs)

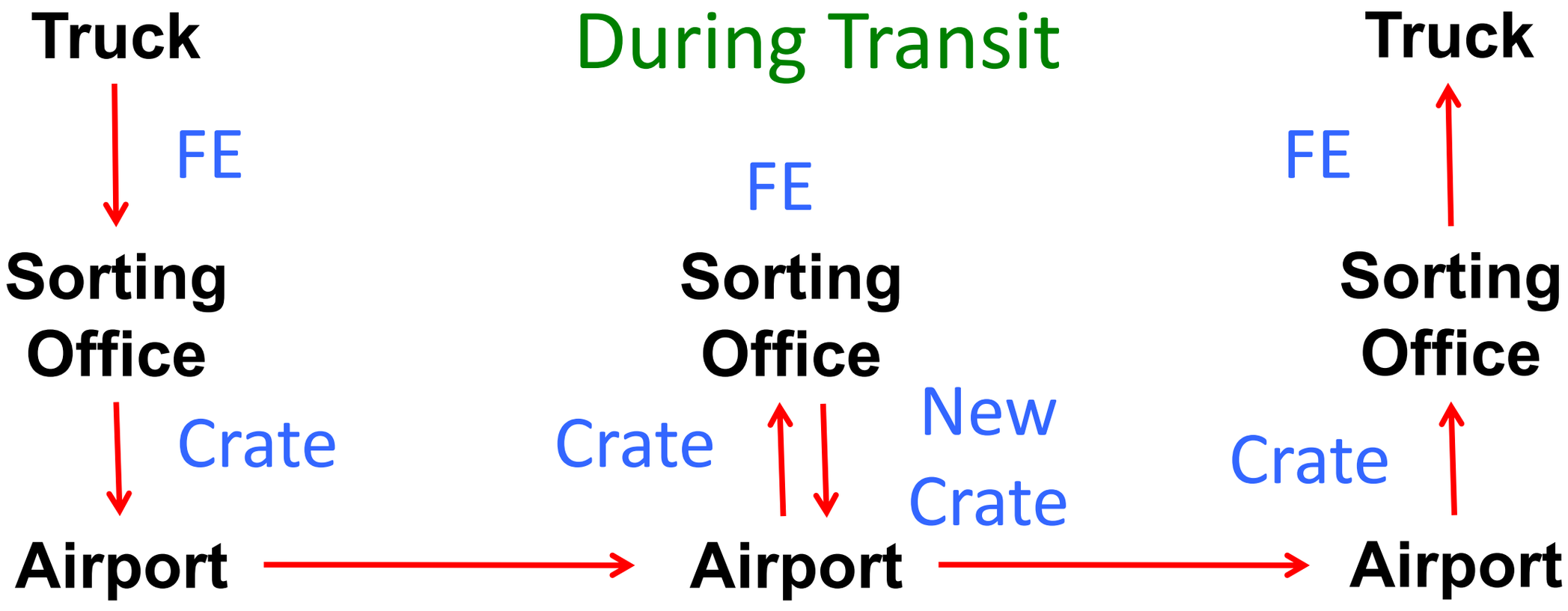
Three steps in Layering

- Decomposition
- Organization
- Assignment

The Path Through FedEx

Higher "Stack" at Ends

Partial "Stack" During Transit



Deepest Packaging (Envelope+FE+Crate) at the Lowest Level of Transport

What gets implemented where?

What gets implemented at the end systems?

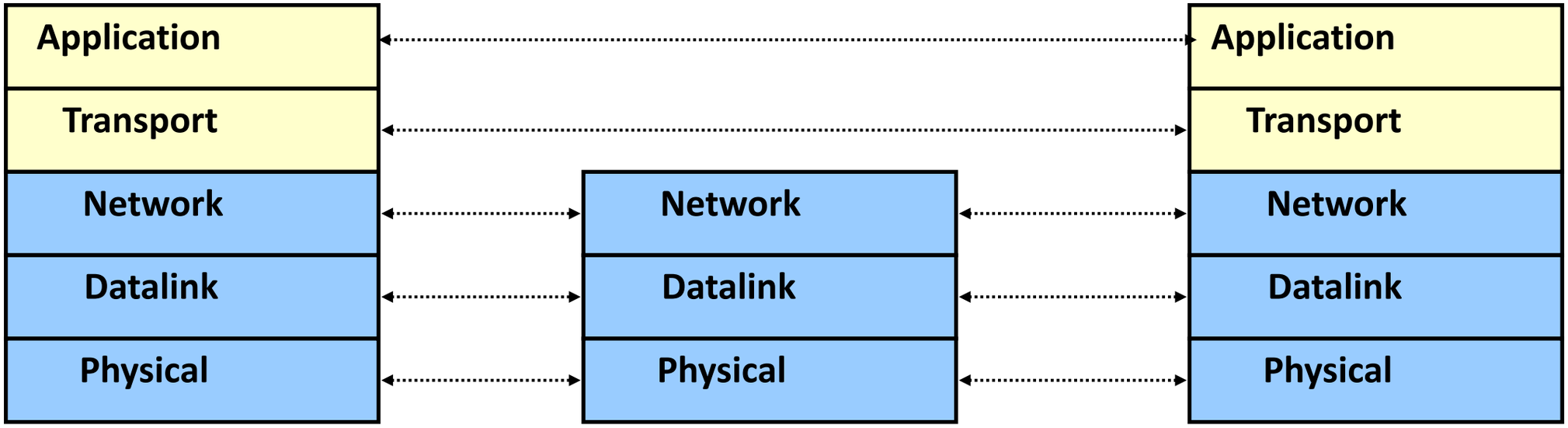
- Bits arrive on wire, must make it up to application
- Therefore, all layers must exist at host!

What gets implemented in the network? (at routers)

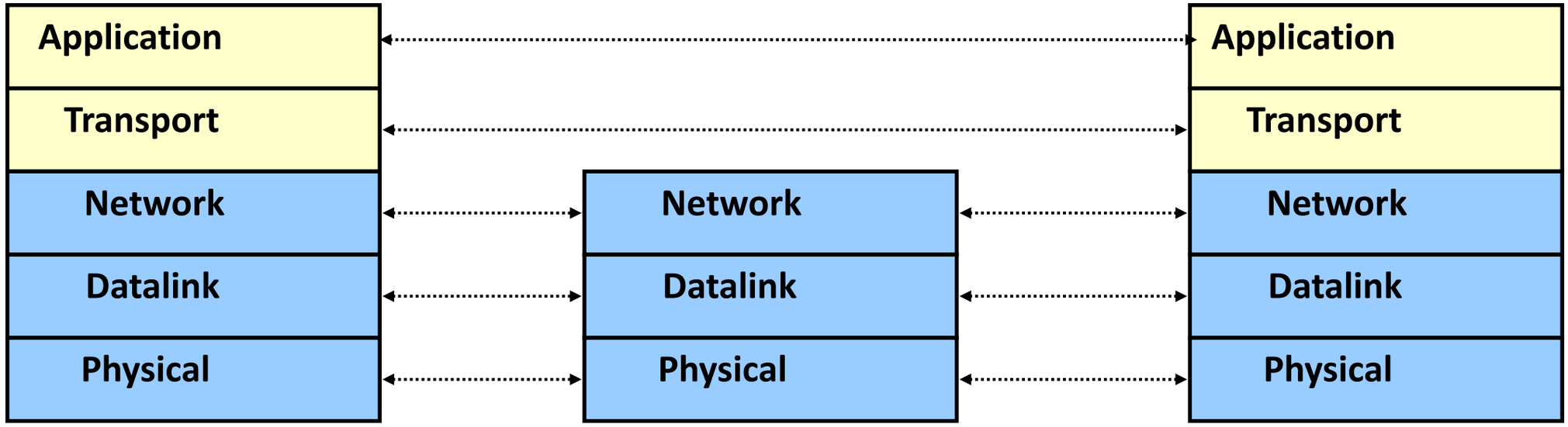
- Bits arrive on wire → physical layer (L1)
- Packets must be delivered across links and local networks → datalink layer (L2)
- Packets must be delivered between “networks” **<u>for global delivery</u>** → network layer (L3) : IP
- The network does not support reliable delivery
  - Transport layer (and above) ***<u>not</u>*** supported
  - No need for secretary in the FedEx network!

Simple Diagram – global delivery

- Lower three layers implemented everywhere
- Top two layers implemented only at hosts



Delivery within “Local” network

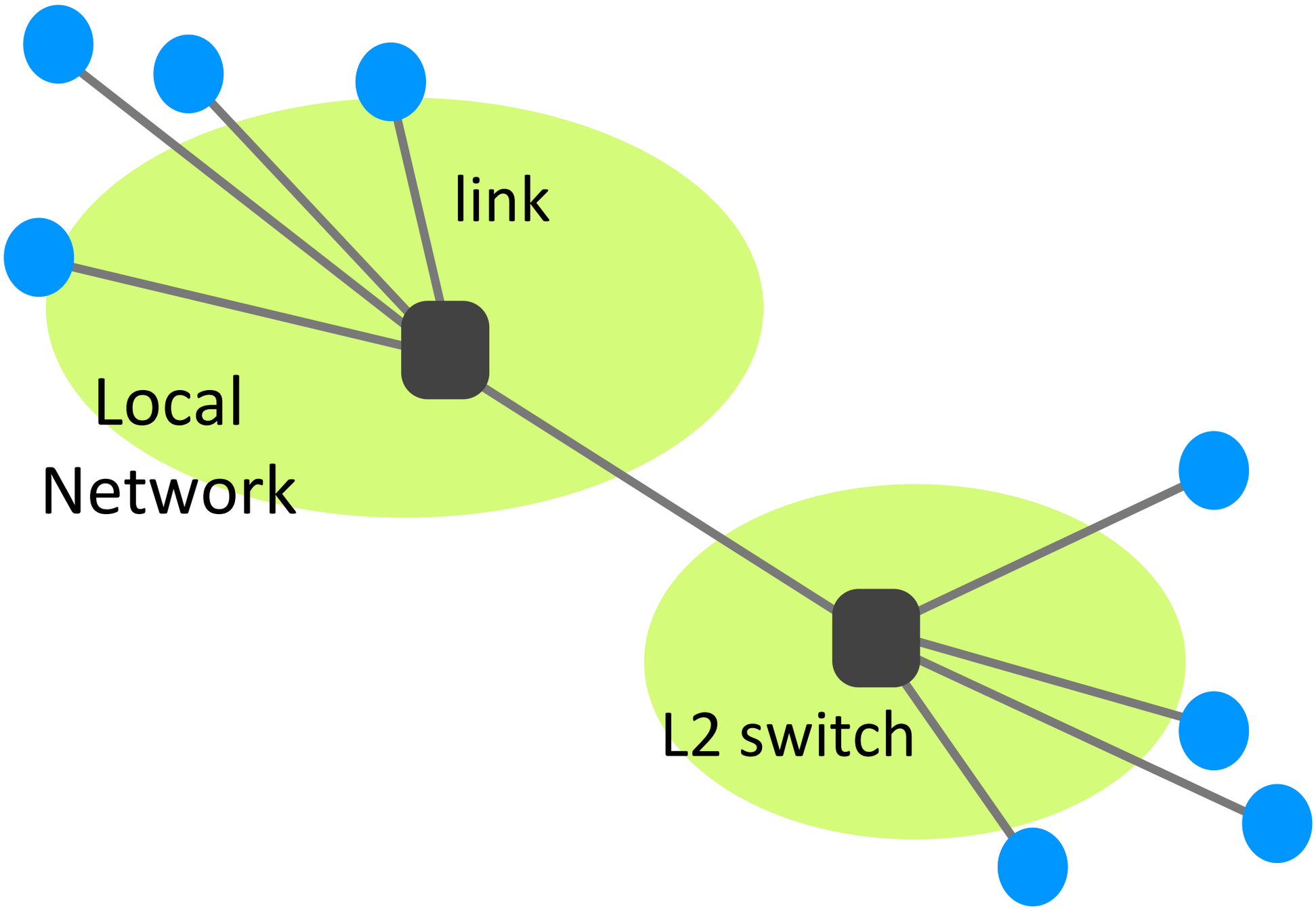


End system Switch End system

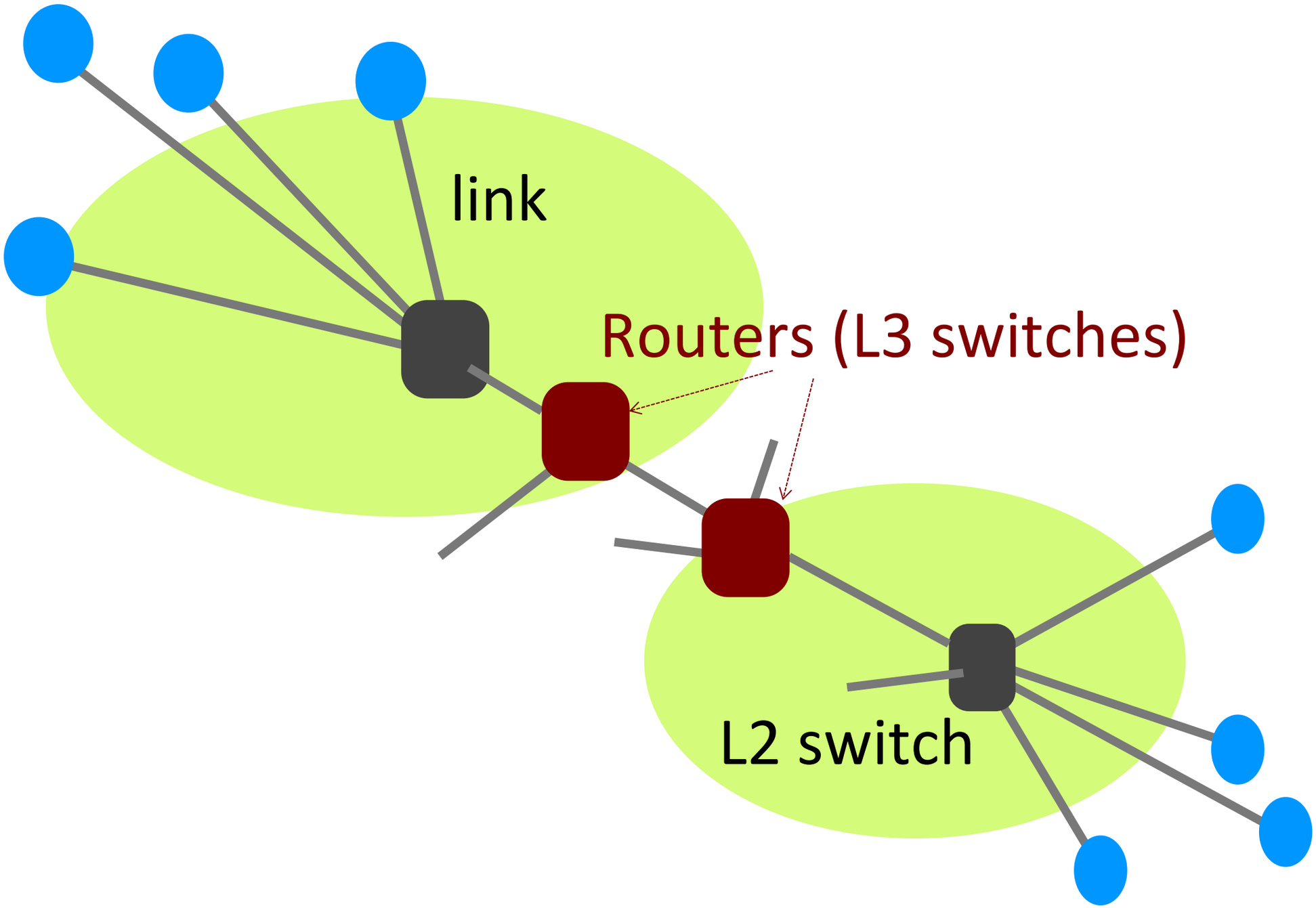
A closer look: end-system

- Application
  - Web server, browser, mail, game
- Transport and network layer
  - typically part of the operating system
- Datalink and physical layer
  - hardware/firmware/drivers

A closer look: network



A closer look: network



What gets implemented in the network?

- Bits arrive on wire → physical layer (L1)
- Packets must be delivered across links and local networks → datalink layer (L2)
- Packets must be delivered between networks for global delivery → network layer (L3) : IP

- Hence:
  - switches: implement physical and datalink layers (L1, L2)
  - routers: implement physical, datalink, network layers (L1, L2, L3)

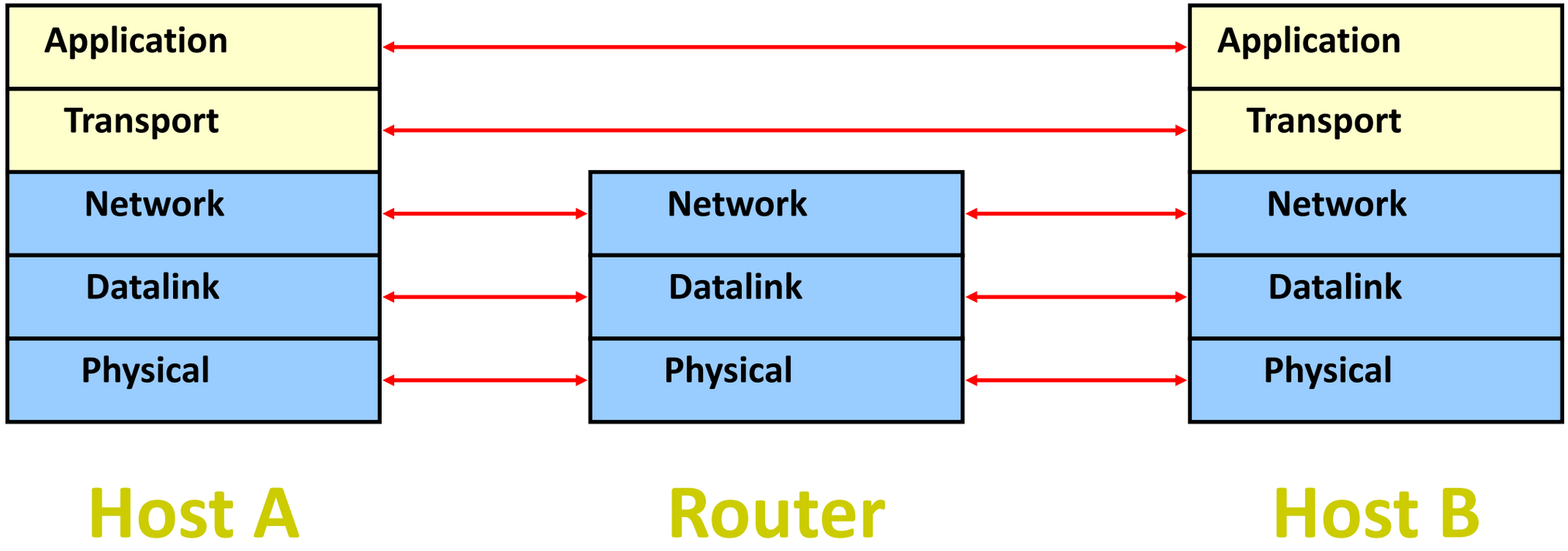
Switches *vs.* Routers

- Switches do what routers do but don't participate in global delivery, just local delivery
  - L2 switches only need to support L1, L2
  - routers support L1, L2, L3

- Won't focus on the router/switch distinction
  - When we say switch, we mostly mean a router
    - We sometimes use "L3 switch" to explicitly mean a router
  - almost all boxes support network layer these days

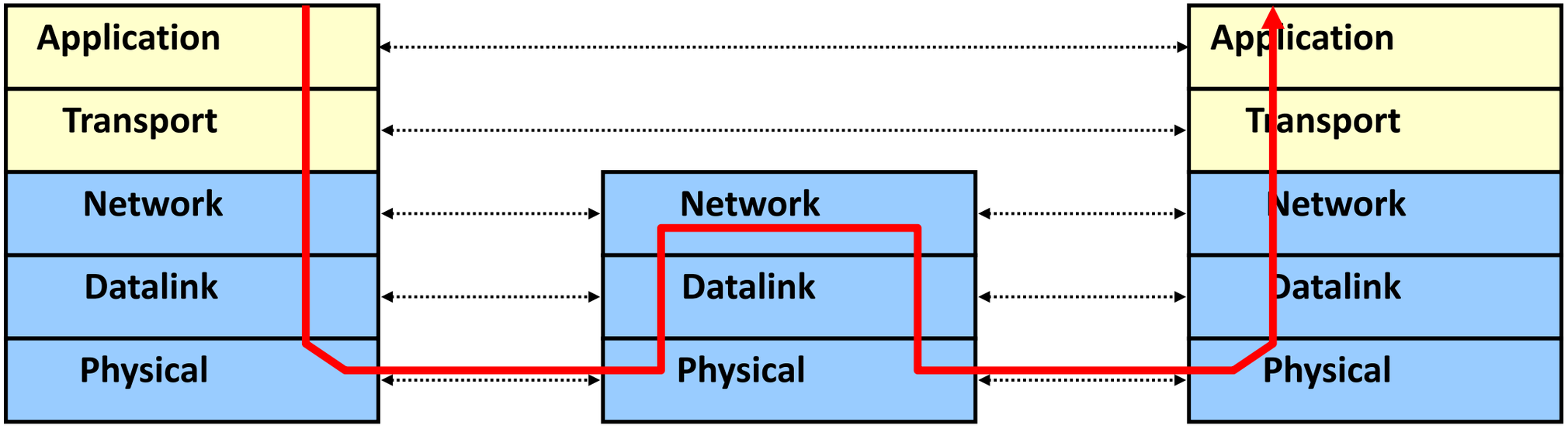
Logical Communication

- Layers interact with their peer layers

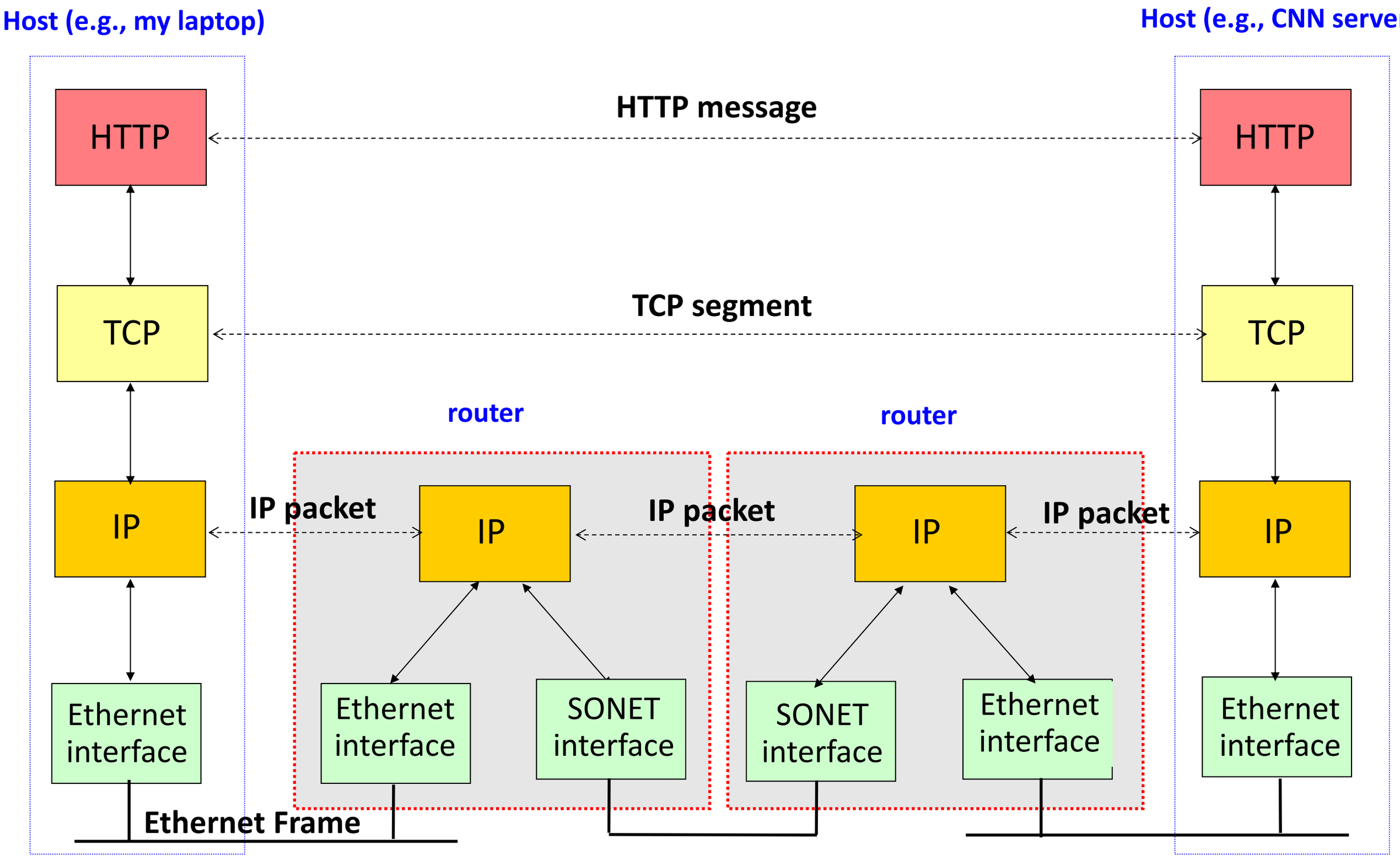


Physical Communication

- Communication goes down to physical network
- Then up to relevant layer



A Protocol-Centric Diagram



Why layers?

- Reduce complexity
- Improve flexibility

Why not?

- sub-optimal performance
- cross-layer information often useful
  - *several “layer violations” in practice*

- *To layer or not? What layers? Where?*

  Would I like to put a secretary in the network who would keep an eye and monitor if things are going as planned!

- *End-to-End Principle (E2E)*

  Hugely influential paper: "End-to-End Arguments in System Design" by Saltzer, Reed, and Clark ('84)

Basic Observation

- Some application requirements can only be correctly implemented **end-to-end**
  - reliability, security, *etc.*
- Implementing these in the network is hard
  - every step along the way must be fail proof
- Hosts
  - **Can** satisfy the requirement without network's help
  - **Will/must** do so, since they can't rely on the network

Summary of End-to-End Principle

- Implementing functionality (e.g., reliability) in the network
  - Doesn't reduce host implementation complexity
  - Does increase network complexity
  - Probably increases delay and overhead on all applications even if they don't need the functionality (e.g. VoIP)

- However, implementing in the network can improve performance in some cases
  - e.g., consider a very lossy link

Recap

- Layering is a good way to organize networks
- Unified Internet layer (L3 or Network layer) decouples apps from networks
- E2E argument encourages us to keep IP simple

When bad things happen:
Mail delivery versus eMail delivery

- Pages (packets) lost
- Pages (packets) out-of-order
- Pages (packets) damaged

Some questions?

- What happens when a page/packet is lost?
  - Local end keeps a copy, so we are safe
- How do we know when a page is lost?
  - Receiver knows when a page is missing (sequence?)
- But the page may arrive later at the receiver?
  - Maybe receiver should wait a bit before declaring a page as lost

Some questions?

- How does the sender know of lost page?
  - A feedback is needed from the receiver

- What if the feedback itself is lost?
  - Perhaps, sender re-sends anyways after a “timeout”

- How long the sender should hold on to a page?
  - Perhaps, some way of **Ack**nowledging from receiver

Big question?

- Can the mail system guarantee a delivery time?
  - They know all resources (buses, trains, etc.)
  - They know all the schedules

- We stick to schedules; still no guarantee? Why?
  - Because “stuff happens” – ***variable load***

- Mail system guaranteeing delivery time?
  - Put enough resources in the network
  - Extremely extremely expensive

Thank you!