

CS 725/825 & IT 725

Lecture 2

# Basic Concepts

---

August 28, 2024

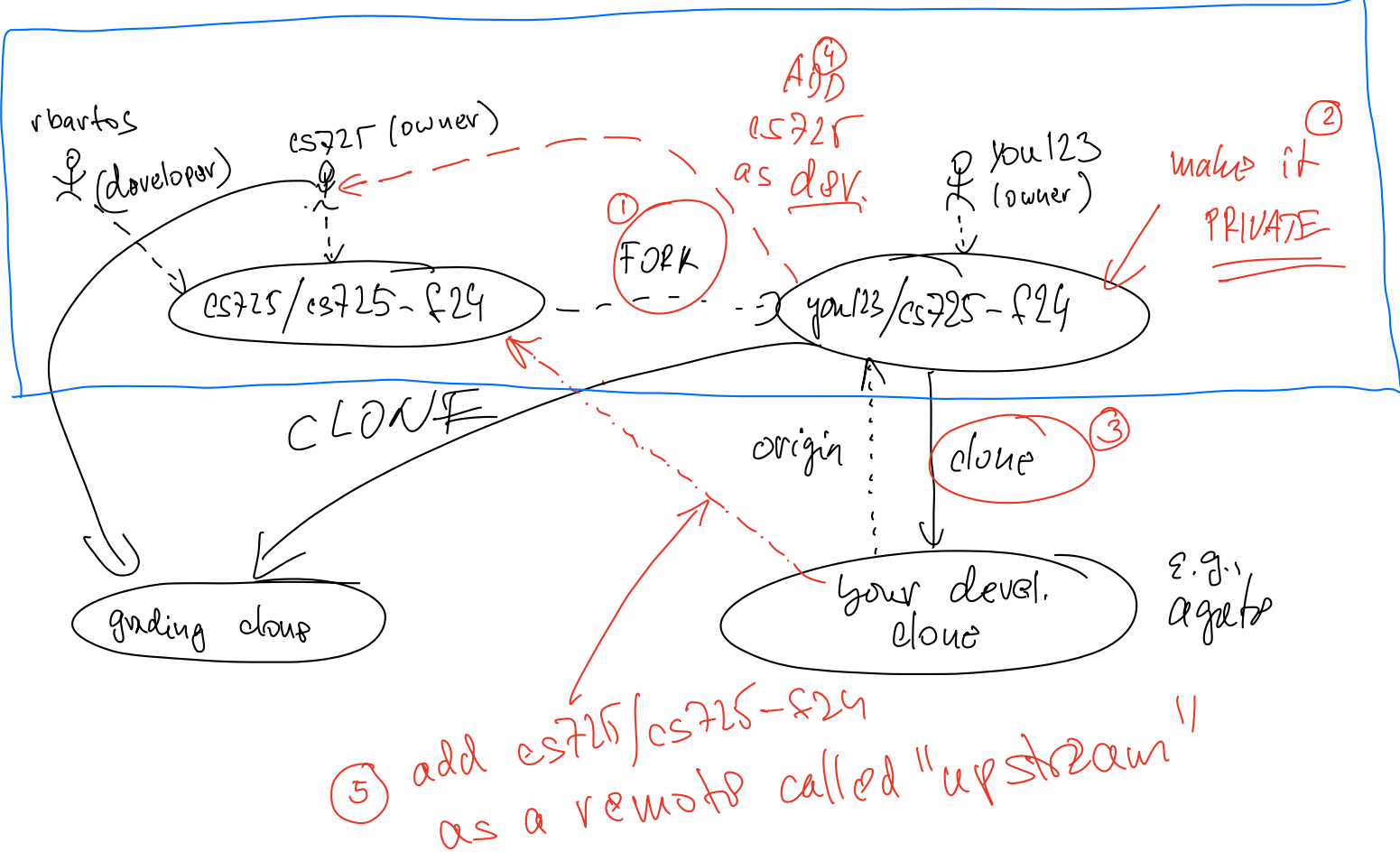
# Assignment 0

---

- ▶ **Objective:** Get ready for programming assignment submissions
  - setup your git repository (<https://www.cs.unh.edu/~cs725/assignments/gitlab.html>)
  - edit README.md files to include your name and email, tag as A0, commit changes, and push to CS GitLab
  - no due date (best done well before the first real assignment is due)

# ASSIGNMENT 0

CS GT LAB



# Basic Concepts

# Basic Terms

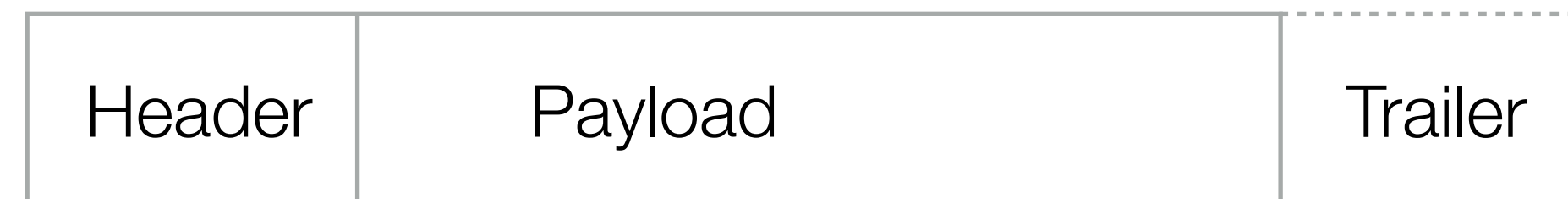
---

## ▶ Protocol

- An agreement on how a communication is to proceed

## ▶ Packet (frame, message, datagram, cell, ....)

- header, data (payload), trailer



## ▶ ?-cast

- unicast, multicast, broadcast, anycast, ...

## ▶ Single hop vs. multihop

# Communication medium types

---

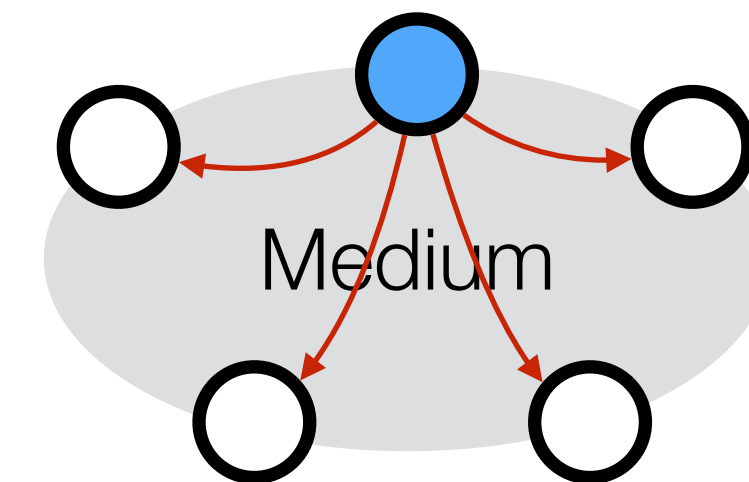
## ▶ Point to point

- between two participants
- simplex, duplex, full duplex
- no need for addressing



## ▶ Broadcast and select

- multiple nodes attached to a shared medium
- everyone hears every transmissions (*broadcast*)
- addresses needed to *select* transmission intended for a node



# Layered models

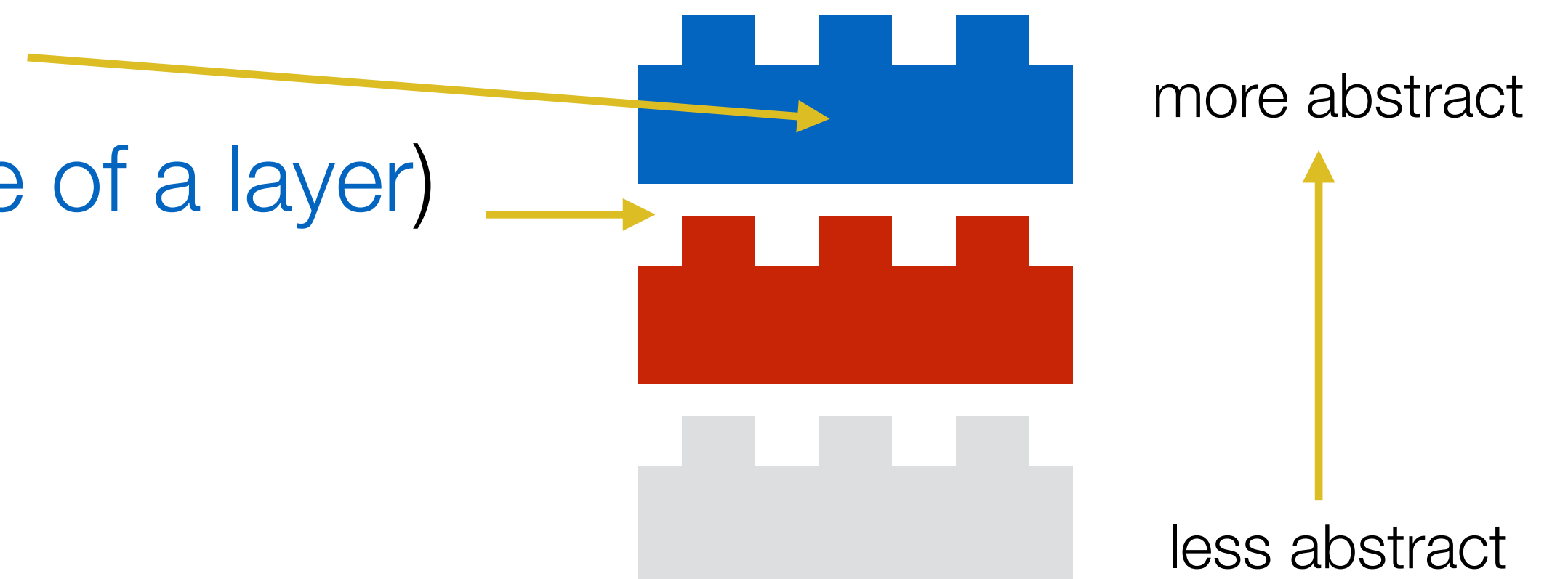
---

## ► Motivation

- networks require many different types of expertise
- need to mix-and-match

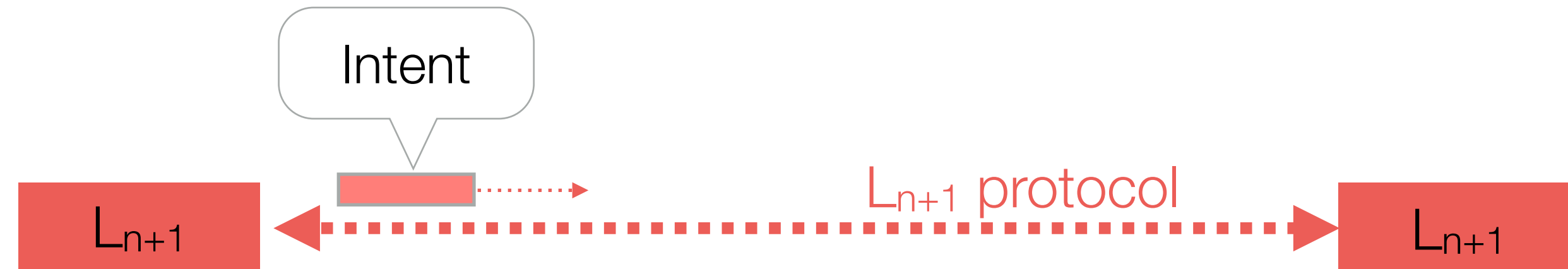
## ► Characteristics

- black box functionality (abstraction)
- simple, well defined interfaces (**service of a layer**)
- vertically stacked



# Protocol hierarchy

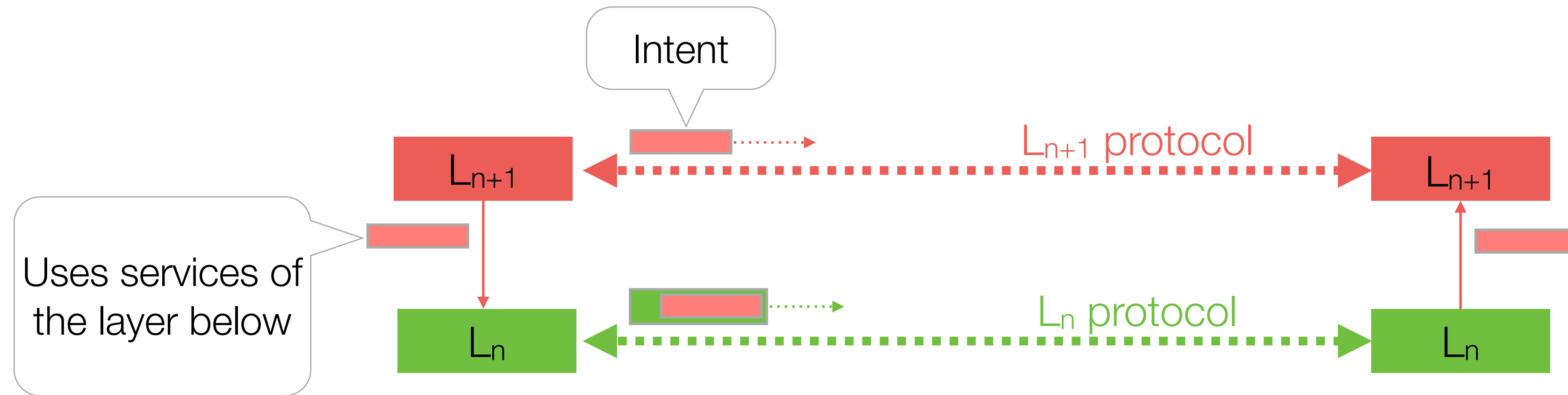
---





# Protocol hierarchy

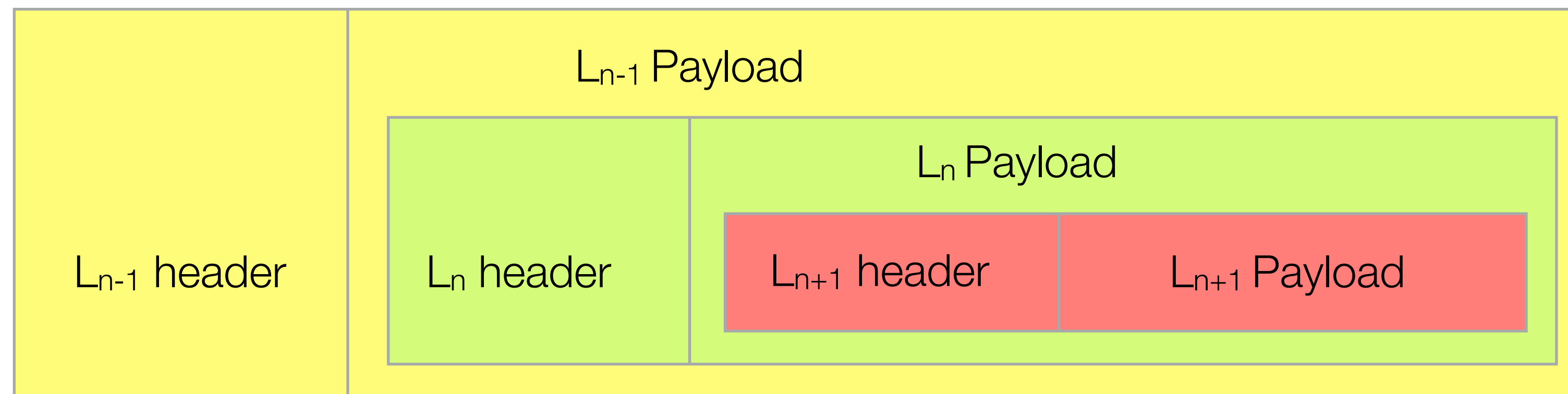
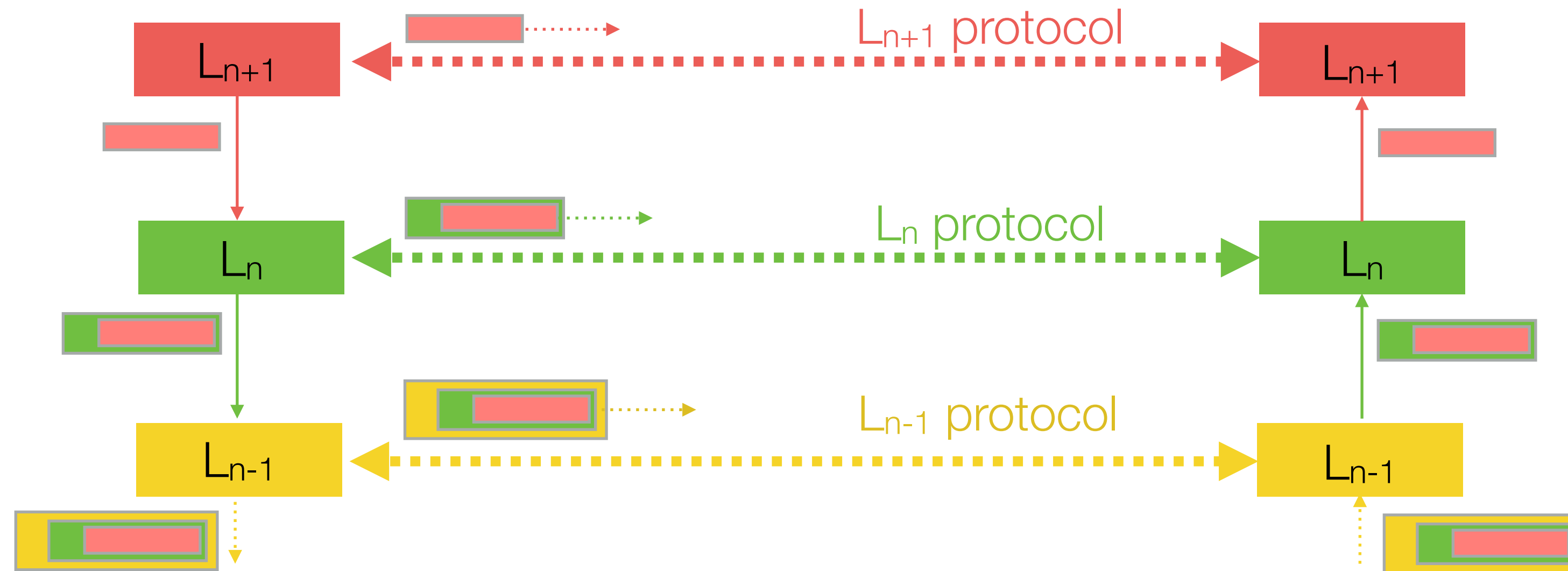
---





# Protocol hierarchy

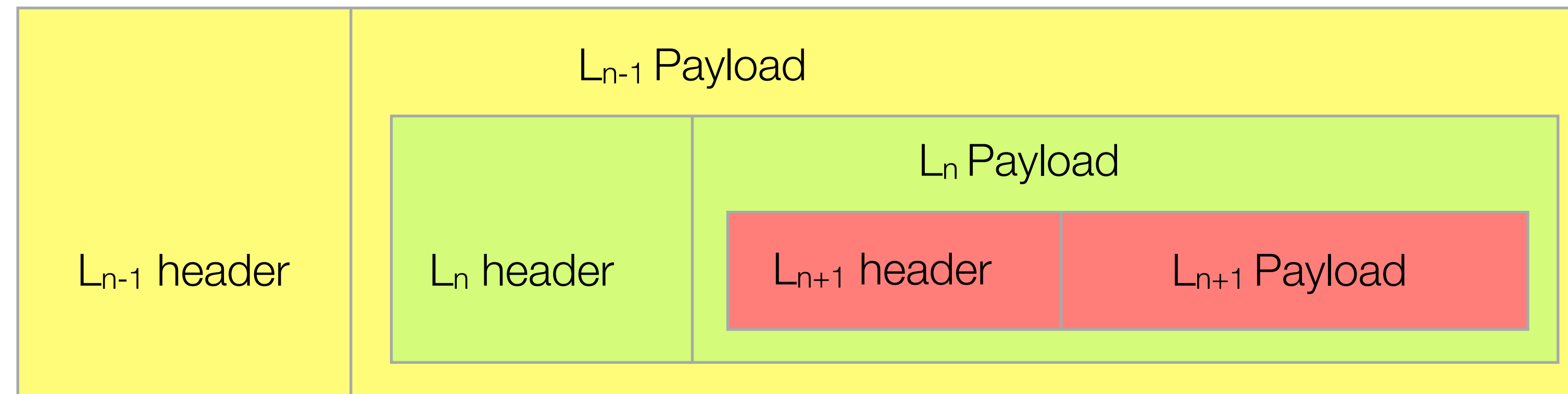
---



# Protocol hierarchy

---

## ► Conceptual view

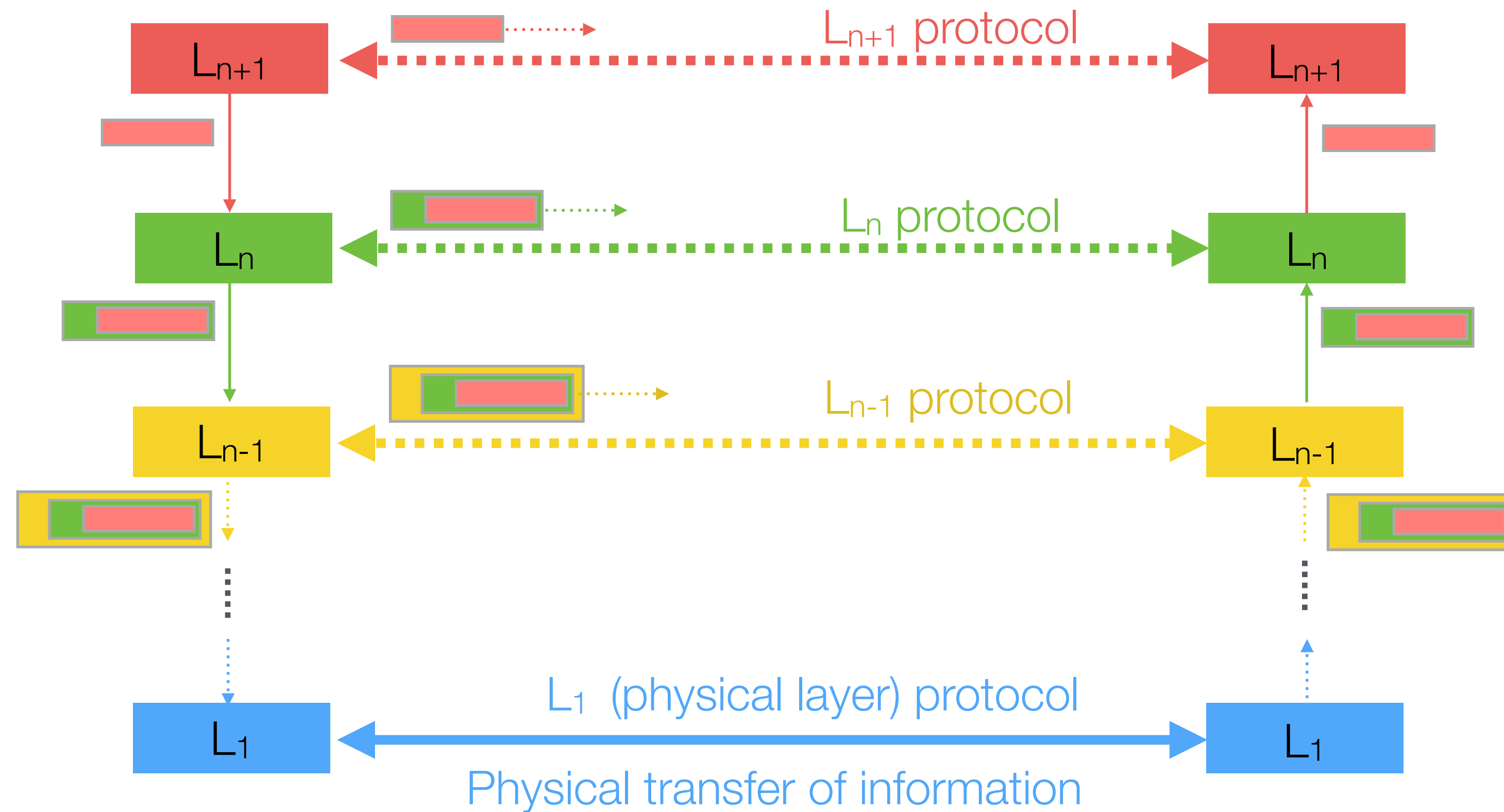


## ► Real protocol example



# Protocol hierarchy

---

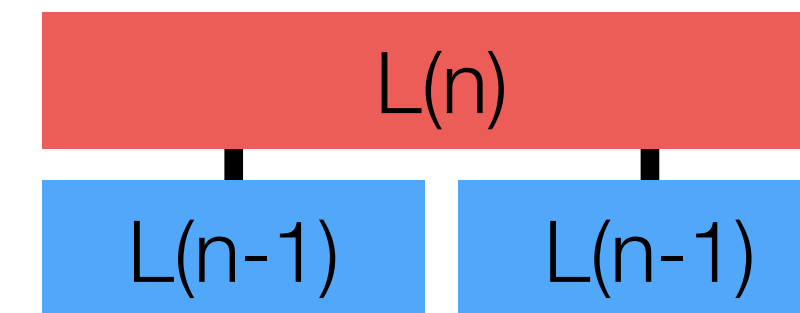


# Multiple layer implementations

---

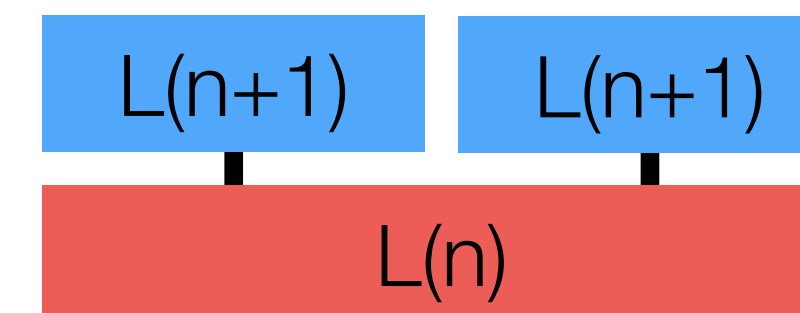
- ▶ Multiple lower layers

- Example: wired and wireless Ethernet



- ▶ Multiple higher layers

- Example: email and web



# OSI 7-Layer Model

---

- ▶ L7 - **Application**
- ▶ L6 - **Presentation** - data representation
- ▶ L5 - **Session** - open/close/maintain session
- ▶ L4 - **Transport** - end-to-end error and flow control
- ▶ L3 - **Network** - end-to-end delivery (routing)
- ▶ L2 - **Link** - node-to-node delivery (single hop)
- ▶ L1 - **Physical** - send bits over a physical channel

# “Internet” layers today

---

- ▶ L7 - Application
  - ▶ L6 - Presentation
  - ▶ L5 - Session
  - ▶ L4 - Transport
  - ▶ L3 - Network
  - ▶ L2 - Link
  - ▶ L1 - Physical
- ▶ L7 - Application
  - ▶ L4 - Transport
  - ▶ L3 - Network
  - ▶ L1 & L2 - Link & Physical



# “Internet” layers today

---

- ▶ L7 - Application
  - ▶ L6 - Presentation
  - ▶ L5 - Session
  - ▶ L4 - Transport
  - ▶ L3 - Network
  - ▶ L2 - Link
  - ▶ L1 - Physical
- ▶ L7 - Application
  - ▶ “security layer”
  - ▶ L4 - Transport
  - ▶ L3 - Network
  - ▶ L1 & L2 - Link & Physical

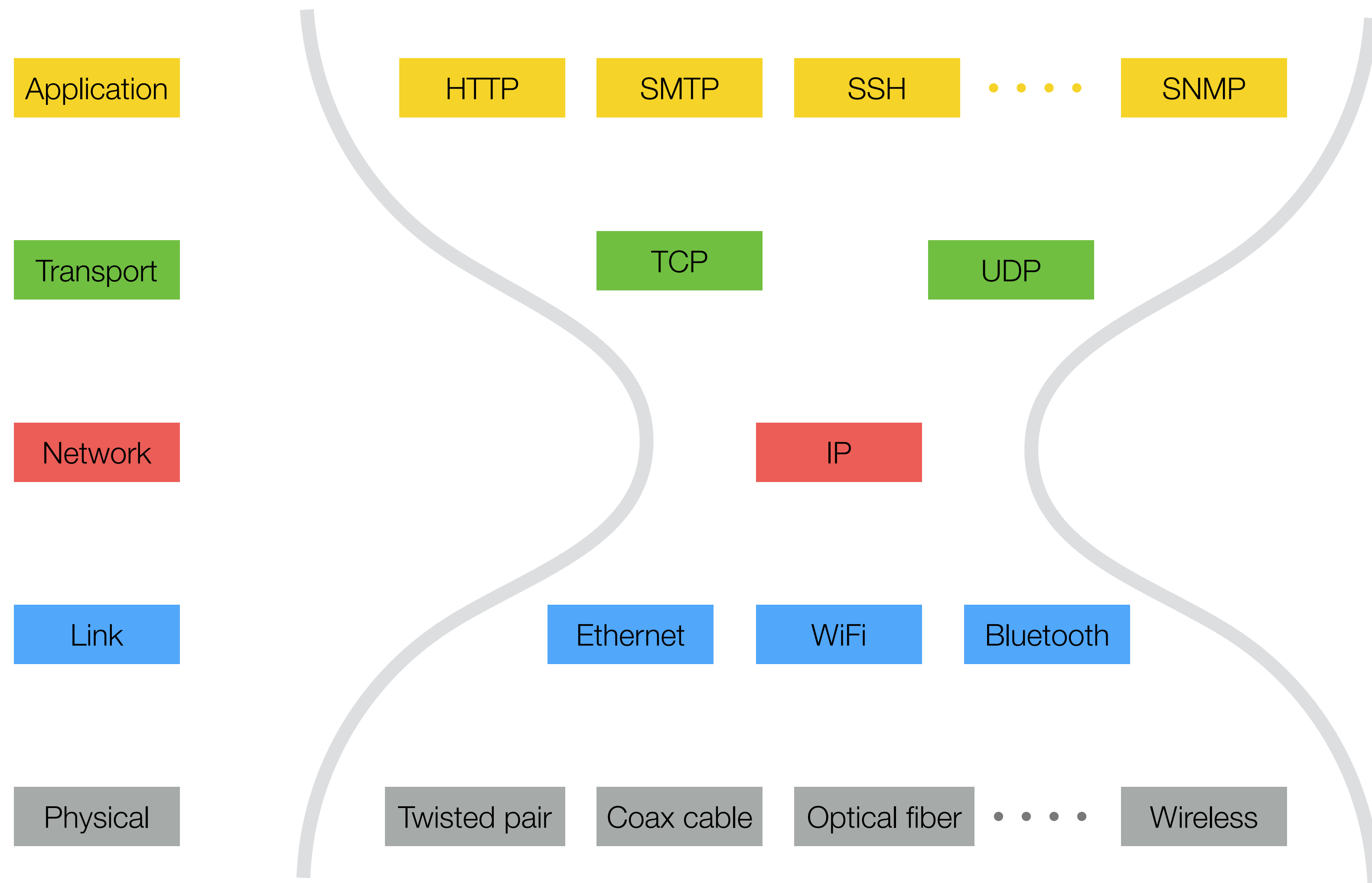
# Internet protocol examples

---

- ▶ **Application** layer
  - HTTP/**HTTPS**, SMTP (email), streaming, messaging, etc.
- ▶ **Security** layer: **TLS** (a.k.a. SSL)
- ▶ **Transport** layer
  - **TCP**, UDP
- ▶ **Network** layer
  - IPv4, **IPv6**
- ▶ **Link & Physical** layer
  - Wired Ethernet, WiFi, etc.

# Internet “Hourglass”

---



# Protocols going forward

---

## ▶ QUIC and HTTP/3

- Enhanced HTTP over TLS 1.3 over UDP over IP ...
- Developed by Google and becoming widely deployed
- Design goal: reduction of transaction latency
- Implemented in the user space (application)

# Common Layer Functions

---

- ▶ Addressing
- ▶ Error control
  - error detection
  - error correction
- ▶ Flow control (traffic management, congestion control)
- ▶ Quality of Service (QoS)
- ▶ (new) Security