

Basic Concepts (I)

Basic Terms

- **Protocol**
 - An agreement about how communication is to proceed
 - A protocol specifies things like:
 - What messages can be sent?
 - What format do they have?
 - In what order are they sent?
 - What should the receiver do when a message arrives?
 - What happens when something goes wrong?
 - Protocols are everywhere
 - When you open the CS homepage “[https:// cs.colby.edu](https://cs.colby.edu),” your computer may use several protocols: DNS (find CS’s IP), QUIC (establish communication), TLS (secure the communication), HTTP/3 (Request the webpage), IP (deliver packets across networks), WiFi (reach the local access point)
 - No single protocol “does the Internet.” The Internet works because **many protocols cooperate**.
- **Packet** (message, segment, datagram, frame, ...)
 - A general term for a unit of data transmitted through networks
 - Has different names depending on the layer a packet is at
 - Application layer — message, transport layer — segment, network layer — datagram, data link layer — frame
 - composed of header (for control information), data (aka payload), trailer (for error checking, not all packets have trailer, ethernet frame has)



- ***-cast**
 - unicast, one -to-one
 - multicast, one-to-group
 - broadcast, one-to-all in a broadcast domain
 - anycast, one-to-anyone in a group (examples include DNS, CDNs, etc. routing directs your request towards an appropriate instance. Why might this improve both performance and reliability?)
- **Single hop** vs. **multi-hop**
 - single hop: two devices communicate directly over one link ([traceroute d122-03 from d122-01](#))
 - multi-hop: communication passes through multiple network devices ([traceroute google.com](#)) (Why doesn’t your packet travel directly from my laptop to Google?)

Network Communication Characteristics

- **Point-to-point** Communication
 - Transmissions are between two participants
 - Three transmission modes: simplex, half-duplex, and full-duplex, defining the direction of the signal flow between two connected devices.
 - Simplex operation
 - Communication flows in only one direction. “one-way street”.
 - E.g., keyboard/mouse/monitor → computer.
 - Half-duplex operation
 - Both ends can transmit, but not simultaneously.
 - One example is that printers to send messages to the computer. The printer cannot send the messages to the computer while the computer is sending data to the printer. But when the computer stops sending, the printer can send messages back.
 - Another one is walkie-talkies
 - Full-duplex operation
 - Both ends can transmit simultaneously. (e.g., telephone line, switched Ethernet).
 - Provide the potential greatest performance
 - WiFi is a useful counterexample. People often assume modern WiFi must be full-duplex because it is fast. It generally isn't. WiFi devices share a wireless medium and ordinarily cannot transmit and receive on the same channel simultaneously. So, high speed isn't equivalent to full duplex.
- **Broadcast and select**
 - multiple nodes attached to as a shared medium
 - everyone hears every transmission (*broadcast*)
 - addresses needed to *select* transmission intended for a node

OSI 7-Layer Model

- OSI (Open System Interconnection) model defines a networking framework to implement protocols in seven layers.
 - L7 - Application
 - Support network applications: DNS, SMTP, HTTP
 - HTTP now has multiple important generations: HTTP/1.1, HTTP/2, HTTP/3. HTTP/3 runs over QUIC rather than directly over TCP.
 - **Protocols evolve while the layered architecture remains useful.**
 - L6 - Presentation
 - Translate data of application to network format, and vice versa: encoding (UTF-8, JPEG, JSON)
 - L5 - Session
 - Establish, manage and terminate connections between applications

- L4 - Transport
 - Provide complete data transfer between end hosts (may provide end-to-end error recovery and flow control): TCP, UDP
 - Modern Internet applications increasingly use QUIC. QUIC runs over UDP but provides capabilities traditionally associated with TCP + TLS, including reliable delivery, congestion control, encryption, multiple streams, faster connection establishment. So instead of HTTP -> TCP -> IP, we can now also have HTTP3 -> QUIC -> UDP -> IP.
 - This is a great example of why networking is **not a finished field**.
- L3 - Network
 - Route datagrams from source to destination: IP (Internet Protocol) is the major protocol
 - Important concepts include: IP addresses, forwarding, routing
 - Device: Router
- L2 - Data Link
 - Communication between neighboring devices over a link/network segment.
 - Examples: Ethernet (computer cluster, servers, data centers), WiF 6/6E/7 (wireless LAN)
 - Device: Switch
- L1 - Physical
 - transmits bits as physical signals
 - possible media: copper, fiber, radio, satellite radio links

Internet Layers

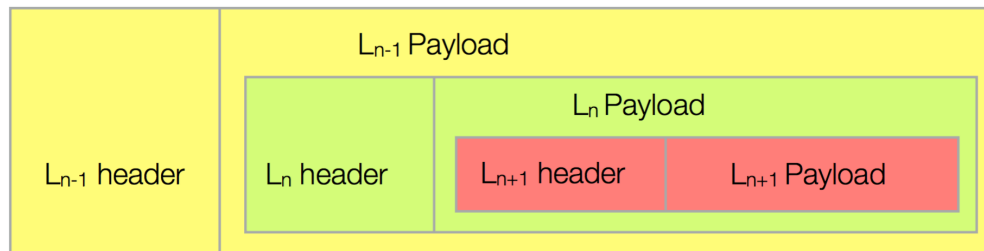
- The OSI model is primarily a conceptual model. The Internet is more commonly described using a simpler stack:

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

- Why layers?
 - Suppose you switch from Colby WiFi to Ethernet, does Chrome need to be rewritten? No
 - OR switch from WiFi to 5G, does HTTP need a new protocol? No
 - Layering allows technologies at one layer to evolve while preserving interfaces to other layers.

Protocol Hierarchy in a Packet: Encapsulation

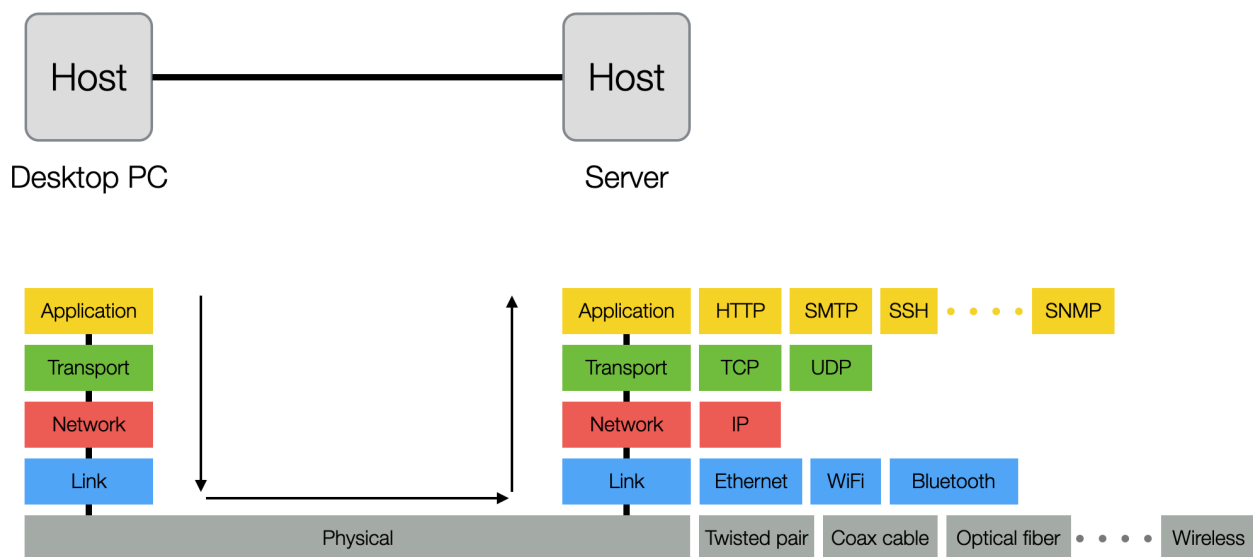
- Conceptual view



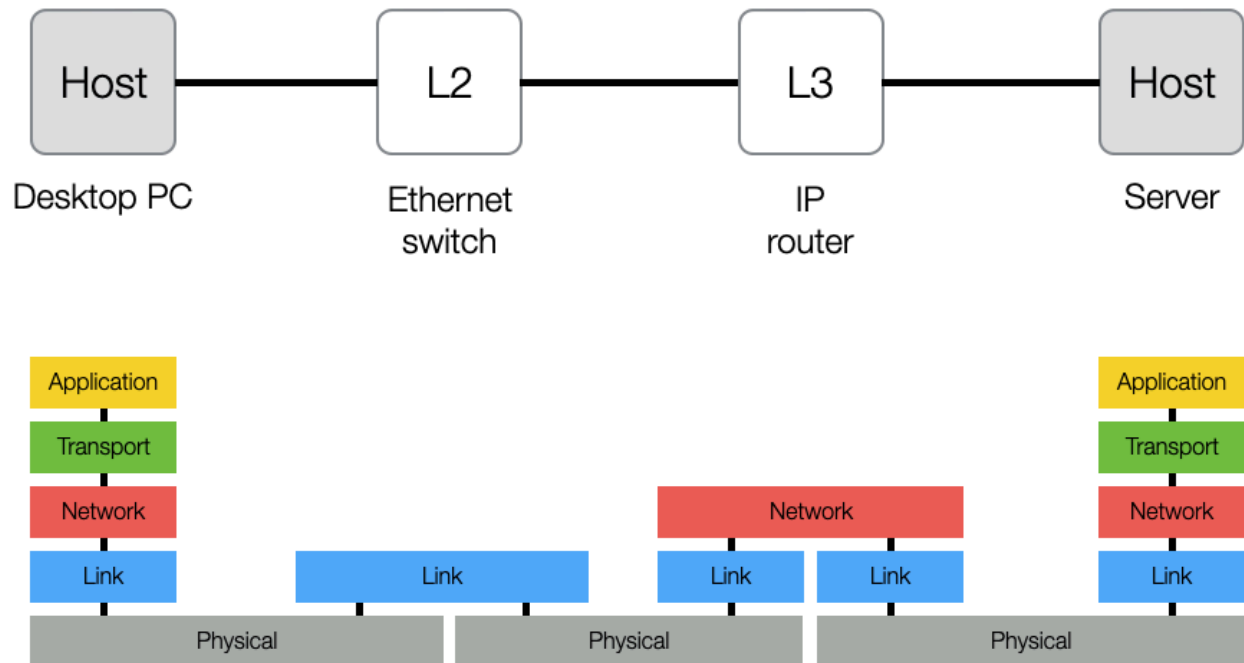
- The packet from the above layer is the payload of the packet at the current layer. E.g., the packet of application layer is encapsulated in the payload of the transport layer packet.

A Layer Examples

- In the layered model, control is passed from one layer to the next, starting at the application layer at one station, and proceeding to the bottom layer, over the channel to the next station and back up the hierarchy.
- Single hop



- Multi-hop



- Use Wireshark to examine the protocol hierarchy of a packet
 - Open Wireshark and sniff on an active interface
 - `nslookup cs.colby.edu` (find the IP address of cs.colby.edu)
 - filter for DNS
 - `ping cs.colby.edu`
 - filter `ip.addr == 137.146.213.6 && icmp` (pings)
 - For the `nslookup` example, identify: Ethernet/WiFi information, IP address, UDP, DNS. Can you physically see the layers (encapsulation) we just discussed inside a real packet?

Disclaimer: Notes adapted from the textbook and online resources.