CS423: Lecture 2, Layering

George Varghese

September 24, 2001
Layering

- Recall that in Hazel’s Hats analogy, we simplified a complex task (transferring hats) by outsourcing reliable transmission to an import-export agency who in turn outsourced package delivery to the post office who in turn outsourced single-hop package transmission to a carrier etc.

- Similarly, in networking, an email transfer is simplified by subcontracting reliable delivery to a transport like TCP who subcontracts packet delivery to the network layer who subcontracts to the Data Link etc.

- This division of labor in networking is called layering. Each horizontal slice (layer) is given a number starting with 1 for physical, 2 for data link etc. While TCP only uses bottom 4 layers, most general model is OSI/ISO model shown next.
application
  ↓
presentation
  ↓
session
  ↓
transport (e.g. TCP)
  ↓
network (e.g., IP)
  ↓
data link (e.g., Ethernet)
  ↓
physical

application
  ← APDU
  ↓
presentation
  ← PPDU
  ↓
session
  ← SPDU
  ↓
transport
  ← TPDU
  ↓
network
  ← NPDU
  ↓
data link
  ← DPDU
  ↓
physical

(e.g., bits as voltages)
Some layering concepts and terminology

- **Protocol**: Rules governing *horizontal* communication between peer layer entities.

- **Interface**: Rules governing *vertical* communication between a Layer N entity and a Layer N+1 entity on the same computer.

- **PDU**: Protocol Data Units are the messages that are exchanged between peer entities. N-PDU between Layer N-entities. In Internet, TPDU = Segment, NPDU = Packet, DPDU = Frame.

- **SDU**: Data Unit passed across an interface. N-SDU passed to and from layer N from Layer N+1.

- **PDU versus SDU**: Normally, N-PDU is N-SDU together with a Layer N header. However, one SDU can be split into multiple PDUs if the protocol allows only small PDUs.
Watch those headers!

- Communication between layer entities shares physical medium by using a layer header for each layer in each message. Think of data in envelope with transport header, stuffed in envelope with routing header, stuffed in envelope with DL header.

- Sharing headers saves postage and also trivially coordinates headers with corresponding data (compared to out-of-band transmission between layers).

- **Strict Layering:** Each layer only looks at its header and interface data to do its job. Software engineering: changes to one layer do not cause other layers to be reimplemented. Information can be passed between layers via interface.

- As data moves down the layers, each layer adds its header. As data moves up, each layer strips off its header. Read text, 1.1 – 1.3
From Files to Voltage Levels

- File Transfer implemented by two FTP processes on each machine. Shared queue is simplest asynchronous interface, which is what TCP provides.

- TCP implements shared queue abstraction by sending numbered segments, retransmitting if acks are not received. Requires being able to send segments to arbitrary destinations, which is what IP provides.

- IP computes routes (routing) and then forwards packet hop-by-hop. At each hop, IP requires sending a frame to directly connected neighbor, which is what Data Link provides.

- Data Link uses physical layer to send each bit of a frame; then puts together bits at receiver to form a frame and does error checks. Physical layer sends bits by transforming 0s and 1s into physical energy that can travel distance.
So why a 20 lecture class?

All layers have common problems: synchronization in the face of errors and asynchrony, addressing, multiplexing, interconnection. Sample problems you will learn the answer to:

- Transport:
  - Congestion Control. How does a TCP sender know how to speed up or slow down depending on current Internet speed? Slow-start.
  - Connection Management: How does TCP prevent old conversations between the same pair of machines from mixing in with new conversations. 3-way handshakes

- Routing:
  - CIDR: How does IP allow various sizes of networks in allocating addresses.
  - BGP: How does IP calculate routes between multiple competing provides?
• **Data Link:**

  – **Min Packet Sizes:** How does Ethernet ensure that if one node detects a collision, all nodes do?

  – **Dynamic Backoff:** How can Ethernet sort out 2 sender collisions quickly while being able to sort out even 32 sender collisions?

• **Physical Layer:**

  – **Clock Recovery:** How does a receiver reconstruct bits from physical signals despite speed differences?

  – **Media Issues:** When should a manager use wireless versus fiber versus satellite?