CS 372 Lecture #38

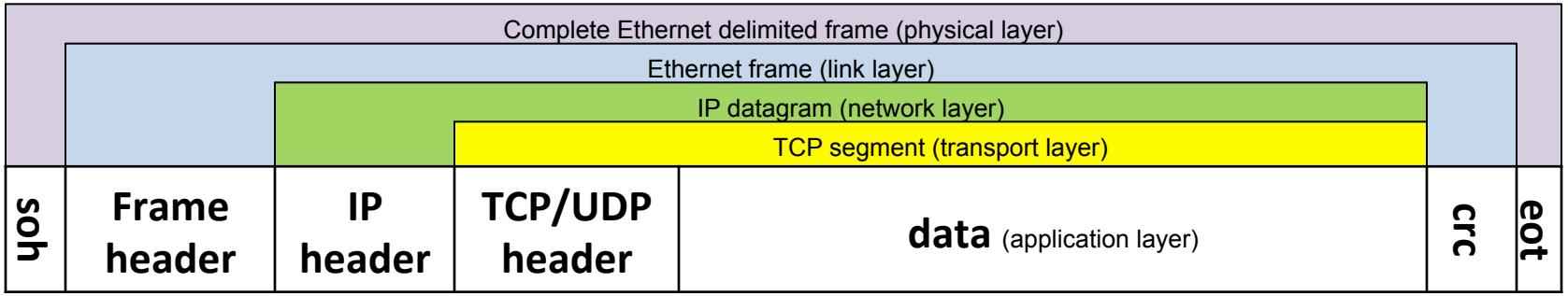
Ethernet

- framing delimiters
- byte stuffing
- CSMA/CD

Note: Many of the lecture slides are based on presentations that accompany *Computer Networking: A Top Down Approach,* 6th edition, by Jim Kurose & Keith Ross, Addison-Wesley, 2013.

Frames, delimiters

- *Frame* (*hardware frame*) denotes a packet with a specific format on a specific hardware technology
- Each hardware technology defines standard *delimiters* (*start*/*end*) to indicate the beginning and end of the frame
- Can be used to detect transmission problems:
  - Missing *end* indicates sending computer crashed
  - Missing *start* indicates receiving computer missed beginning of message
  - Bad frame is discarded

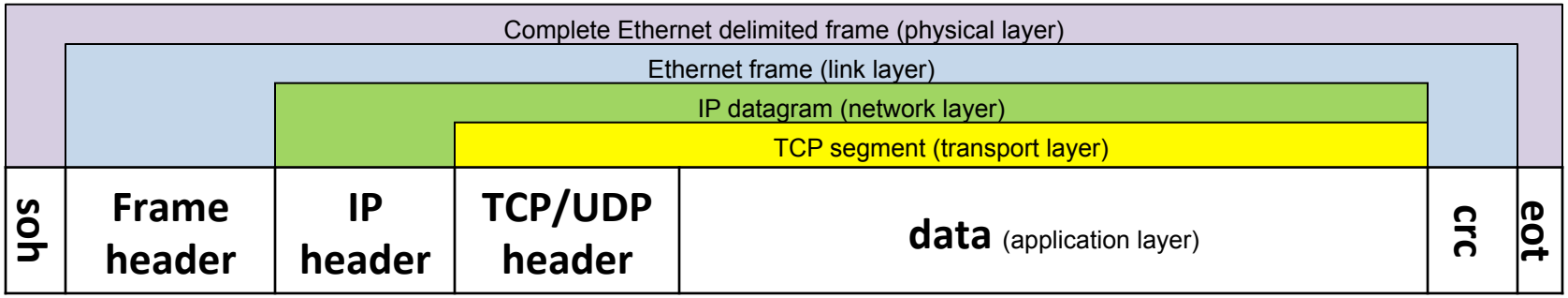


Defining the delimiter standard

- Can choose unused data values for delimiters
  - If data is limited to printable ASCII, can use non-printable characters (control codes), e.g.
    - "start of header" (*soh*, ASCII character #1)
    - "end of text" (*eot*, ASCII character #4)

- Source computer
  - encapsulates a hardware frame between delimiters

- Destination computer
  - interprets/discards *soh*
  - manages hardware frame
  - interprets/discards *eot*



Transmitting arbitrary data

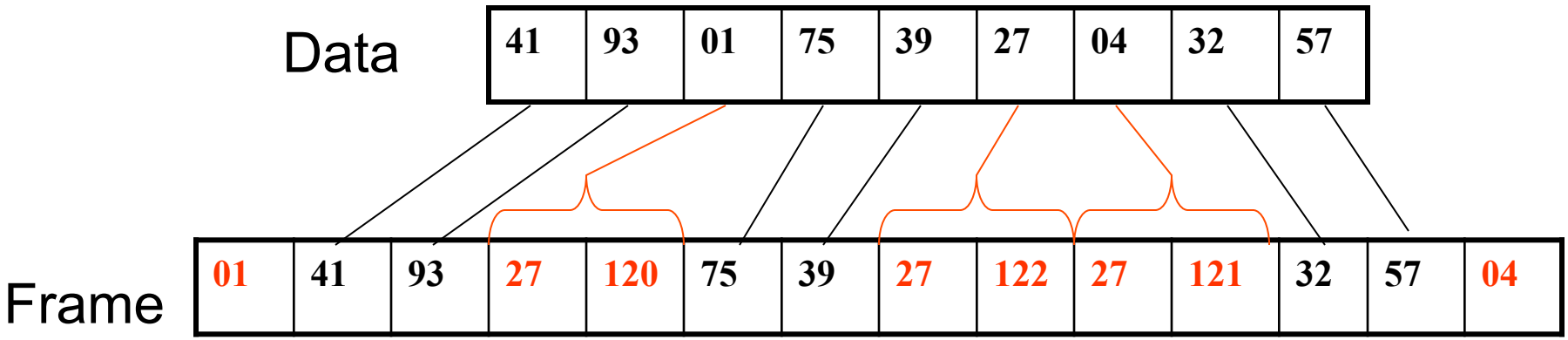
- Transmission might have no unused characters available for delimiters
  - E.G., transmitting binary data, such as graphics files
- If *soh* (01) and *eot* (04) are used inside the data, they will be misinterpreted as delimiters
- Sender and receiver must agree on encoding special characters for unambiguous transmission

Byte Stuffing

- Choose a third "special" character
  - e.g., esc(27)
- soh(01) and eot(04) still delimit the frame
- Whenever one of the delimiters or the third special character appear in the actual data …
  - special character is also “stuffed” into the data
  - the character is replaced per byte-stuffing protocol

Byte-stuffing example

Character in data	Characters sent
soh (01)	esc x (27 120)
eot (04)	esc y (27 121)
esc (27)	esc z (27 122)



Ethernet CSMA/CD algorithm

1. NIC receives datagram from network layer, creates frame
2. If NIC senses channel idle
   - starts frame transmission

   If NIC senses channel busy
   - waits until channel idle, then transmits
3. If NIC transmits entire frame without detecting another transmission
   - NIC is done with frame

   If NIC detects another transmission while transmitting
   - collision: aborts and sends jam signal
   - after aborting, NIC enters **exponential backoff**:

Ethernet's CSMA/CD (more)

Exponential backoff:

- adaptive retransmission attempts to estimate current load
- heavy load: random wait will be longer
- after *m*th collision, NIC chooses *K* at random from {0,1,2,…,2^m-1}.
  - NIC waits K·512 *bit times**
  - returns to Step 2

Choosing random K:

•first collision: choose K from {0,1}; delay is K· 512 bit times*

•after second collision: choose K from {0,1,2,3}…

•after ten collisions, choose K from {0,1,2,3,4,…,1023}

•etc.

*Bit time: time to transmit one bit

- 0.0001 ms for 10 Mbps Ethernet
- for K=1023, wait time is about 52 ms

Summary Lecture #38

- Ethernet
  - legacy, durable, still most popular
  - frame (link layer), delimiters (physical layer)
    - byte-stuffing
  - CSMA/CD
    - exponential backoff