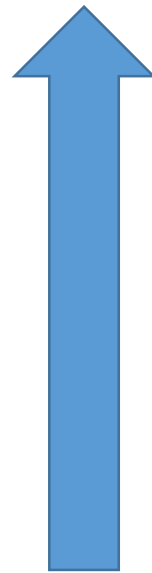
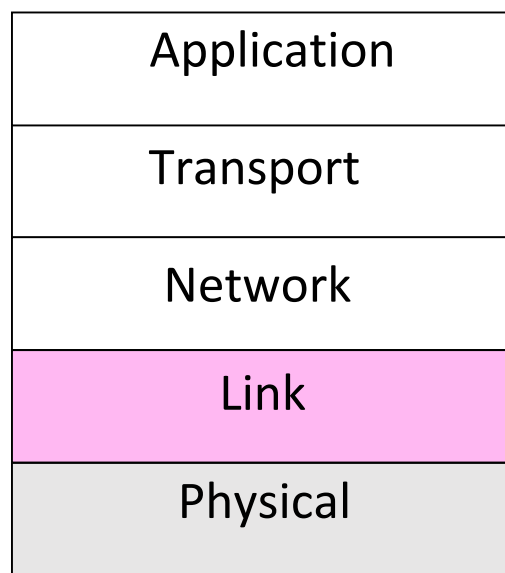


Link Layer

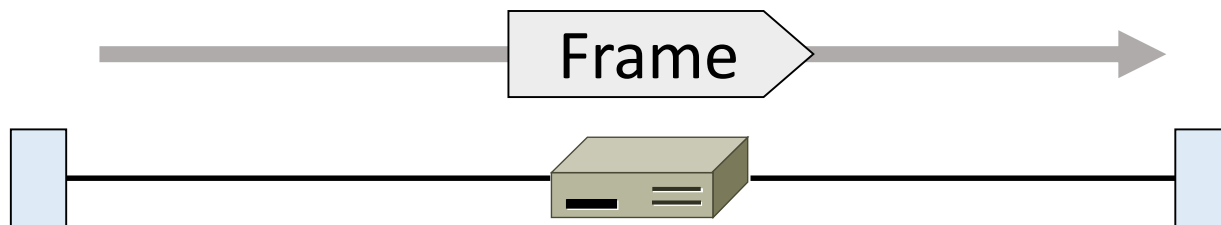
Where we are in the Course

- Moving on up to the Link Layer!

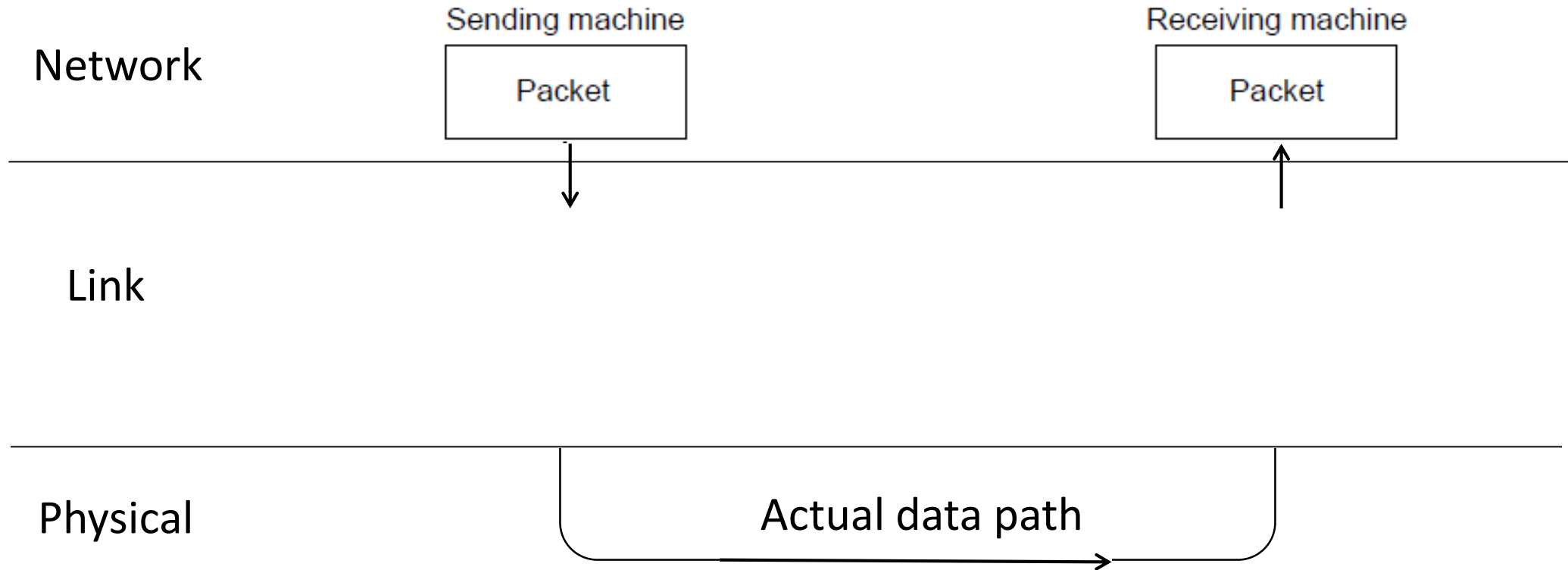


Scope of the Link Layer

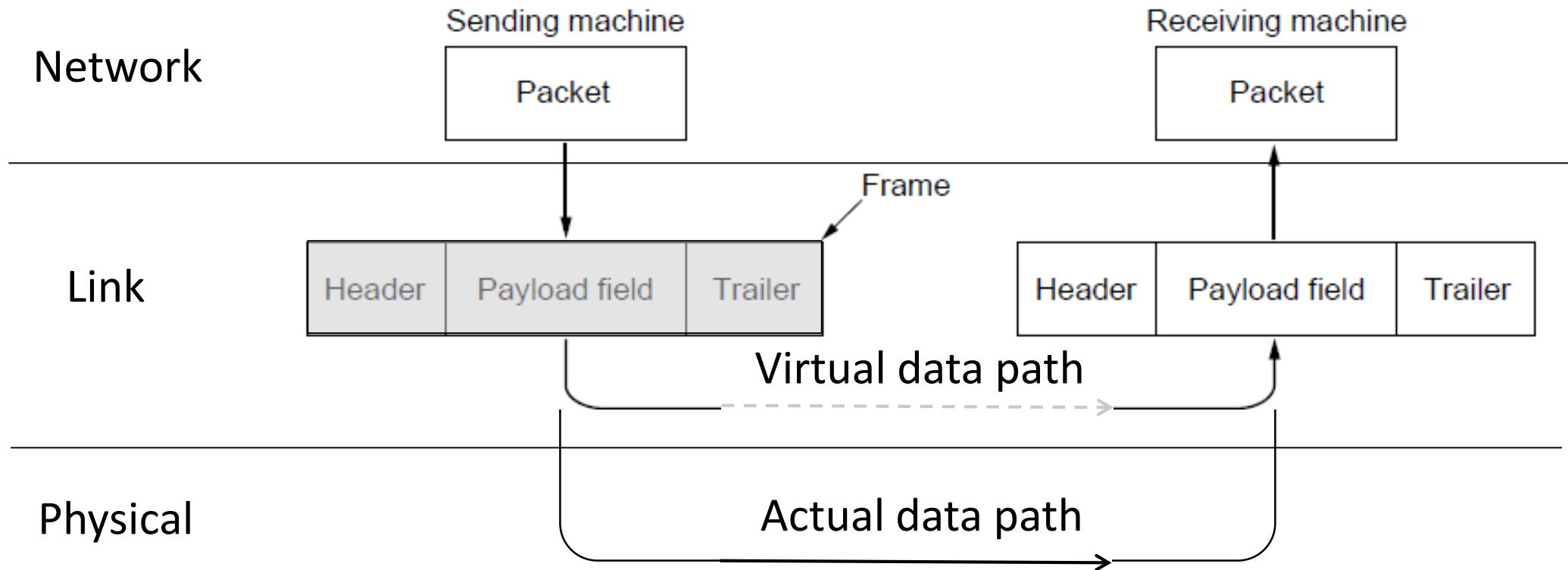
- Concerns how to transfer messages over one or more connected links
 - Messages are frames, of limited size
 - Builds on the physical layer



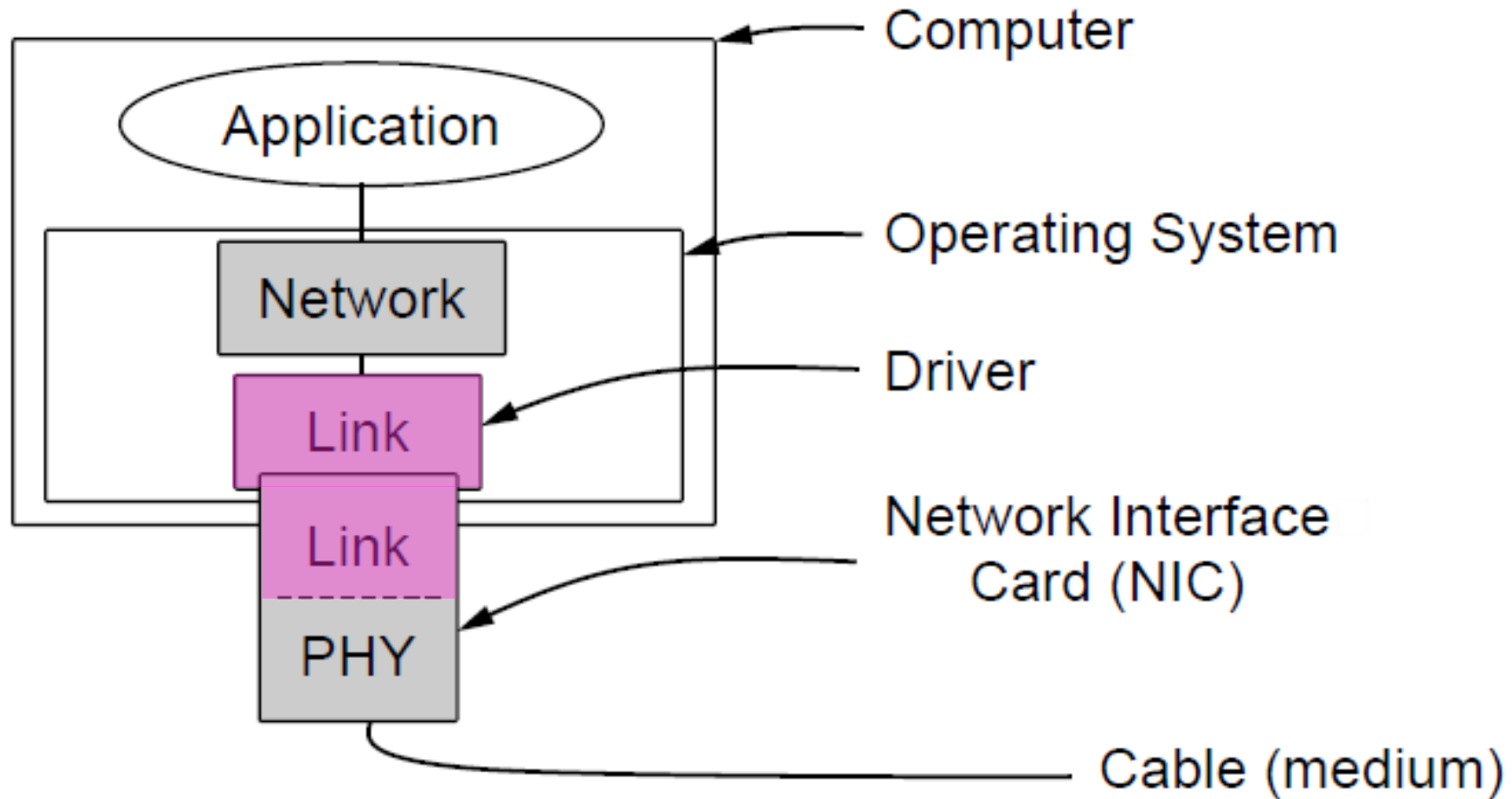
In terms of layers ...



In terms of layers (2)



Typical Implementation of Layers (2)



Topics

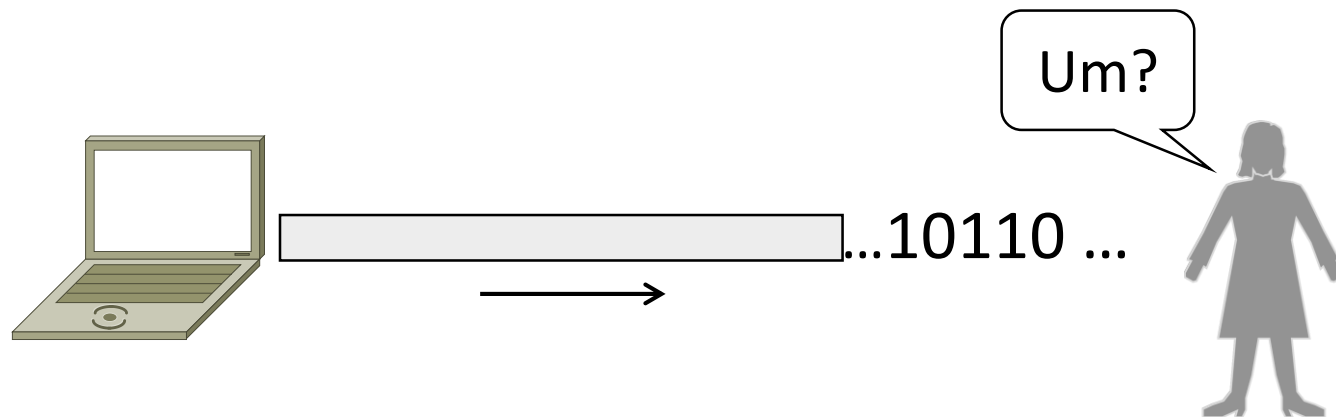
1. Framing
 - Delimiting start/end of frames
2. Error detection and correction
 - Handling errors
3. Retransmissions
 - Handling loss
4. Multiple Access
 - 802.11, classic Ethernet
5. Switching
 - Modern Ethernet

Framing

Delimiting start/end of frames

Topic

- The Physical layer gives us a stream of bits. How do we interpret it as a sequence of frames?



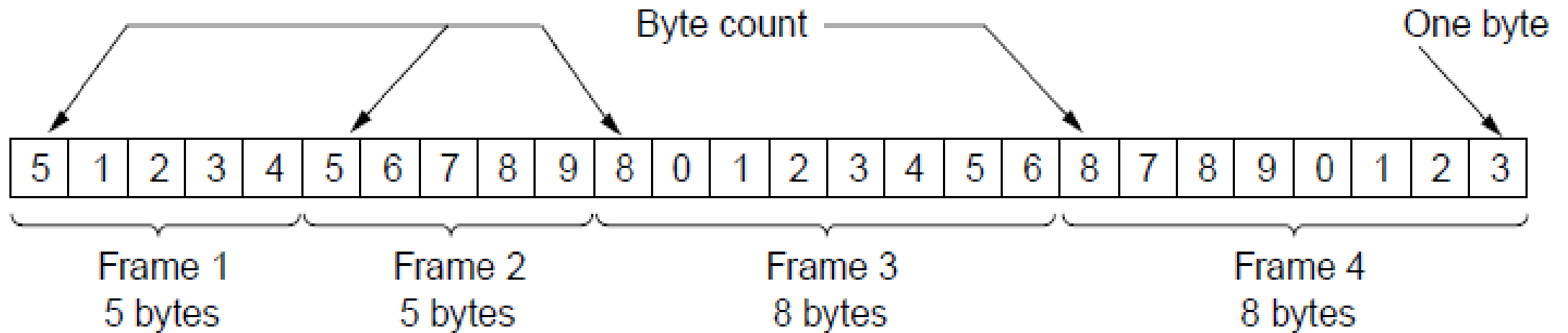
Framing Methods

- We'll look at:
 - Byte count (motivation)
 - Byte stuffing
 - Bit stuffing
- In practice, the physical layer often helps to identify frame boundaries
 - E.g., Ethernet, 802.11

Byte Count

- First try:
 - Let's start each frame with a length field!
 - It's simple, and hopefully good enough ...

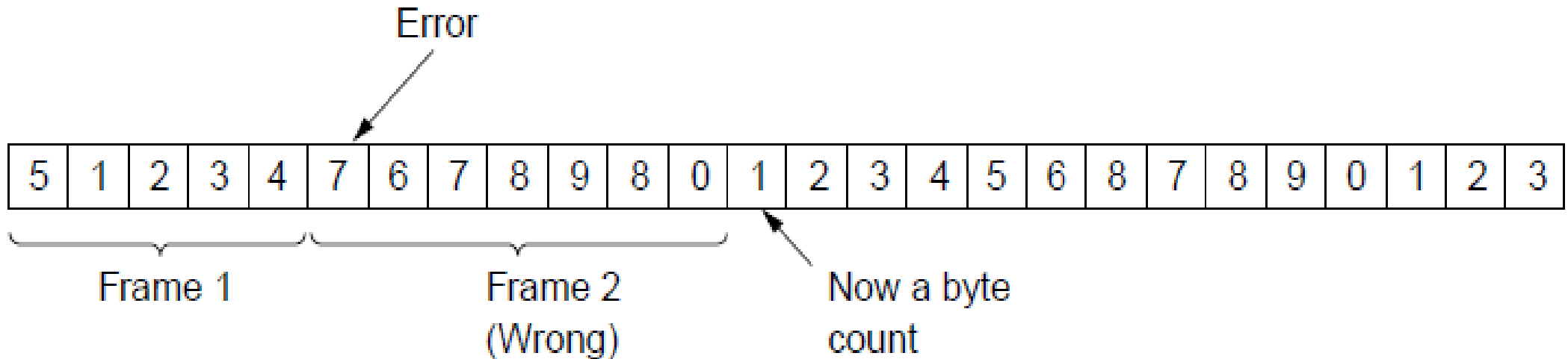
Byte Count (2)



- How well do you think it works?

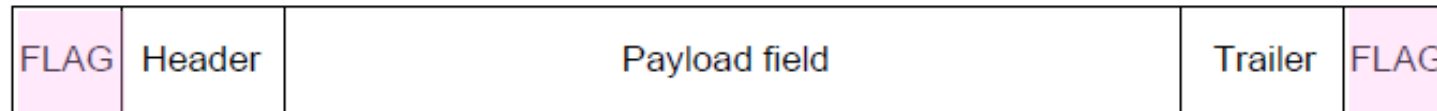
Byte Count (3)

- Difficult to re-synchronize after framing error
 - Want a way to scan for a start of frame



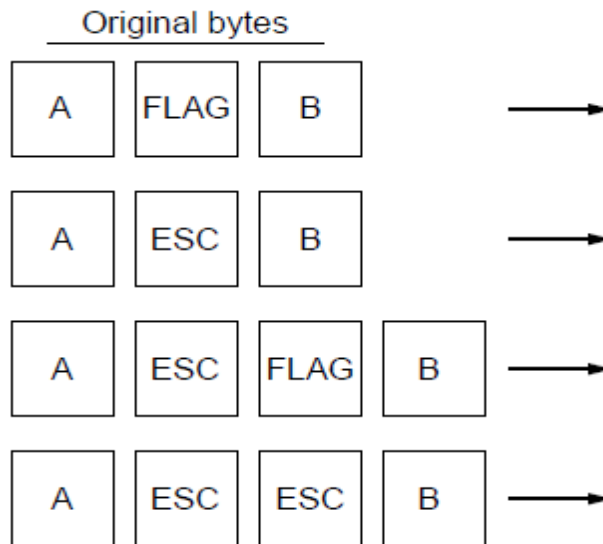
Byte Stuffing

- Better idea:
 - Have a special flag byte value for start/end of frame
 - Replace (“stuff”) the flag with an escape code
 - Complication: have to escape the escape code too!



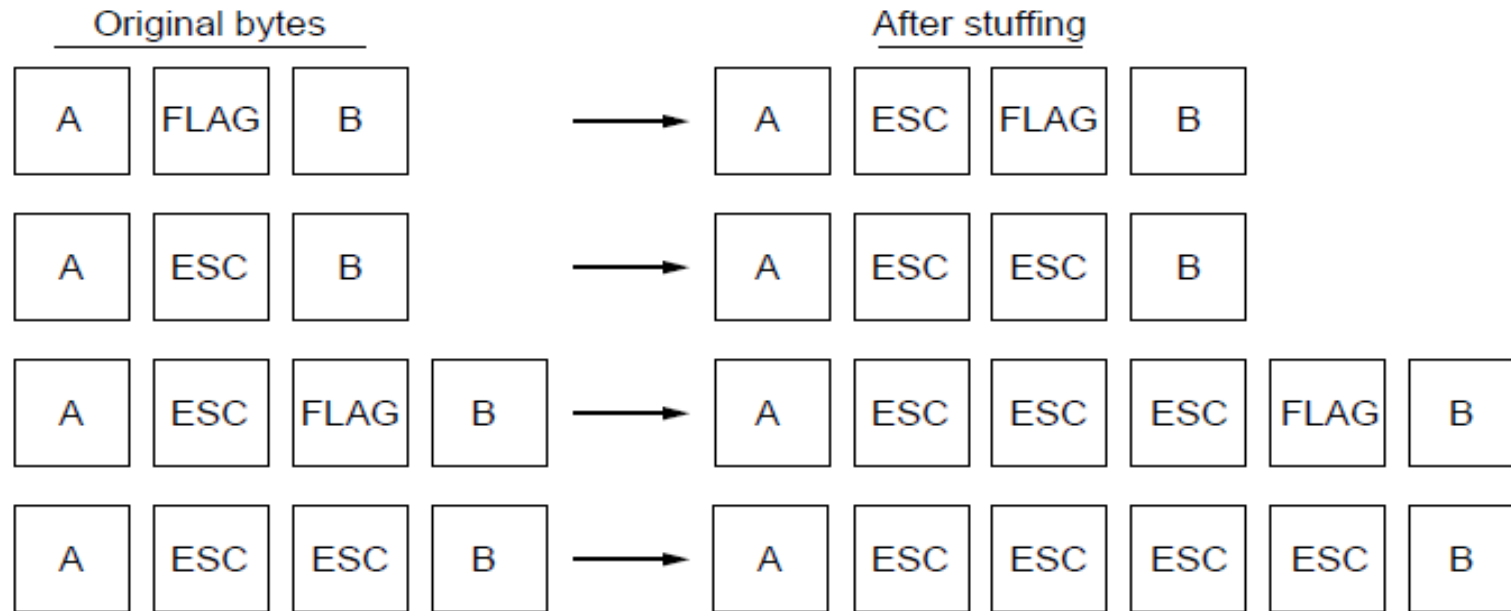
Byte Stuffing (2)

- Rules:
 - Replace each FLAG in data with ESC FLAG
 - Replace each ESC in data with ESC ESC



Byte Stuffing (3)

- Now any unescaped FLAG is the start/end of a frame

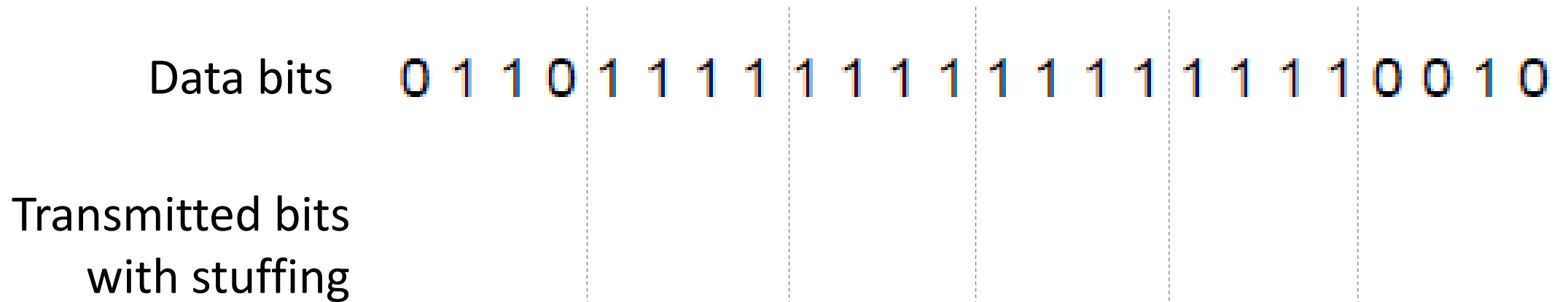


Bit Stuffing

- Can stuff at the bit level too
 - Call a flag six consecutive 1s
 - On transmit, after five 1s in the data, insert a 0
 - On receive, a 0 after five 1s is deleted

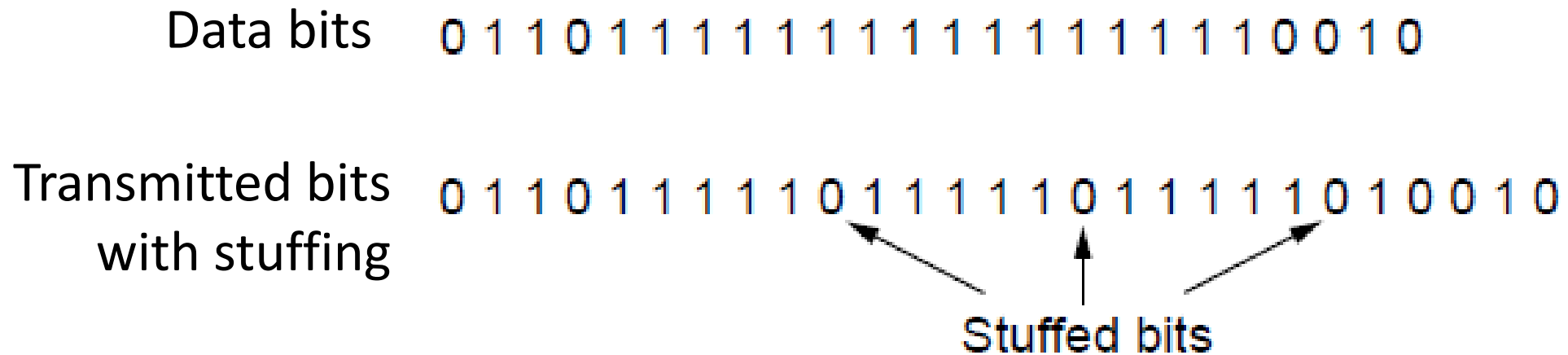
Bit Stuffing (2)

- Example:



Bit Stuffing (3)

- So how does it compare with byte stuffing?

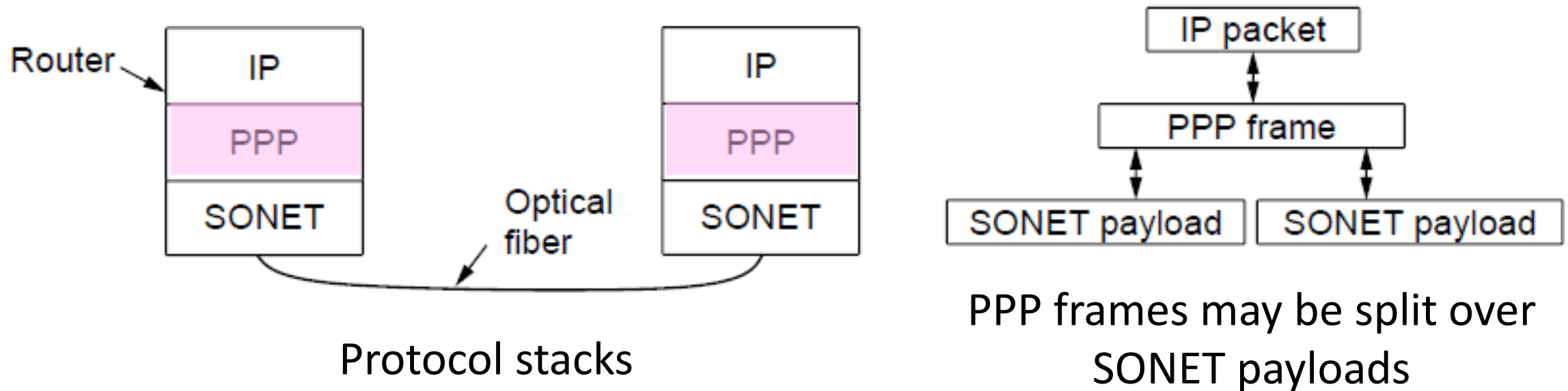


Link Example: PPP over SONET

- PPP is Point-to-Point Protocol
- Widely used for link framing
 - E.g., it is used to frame IP packets that are sent over SONET optical links

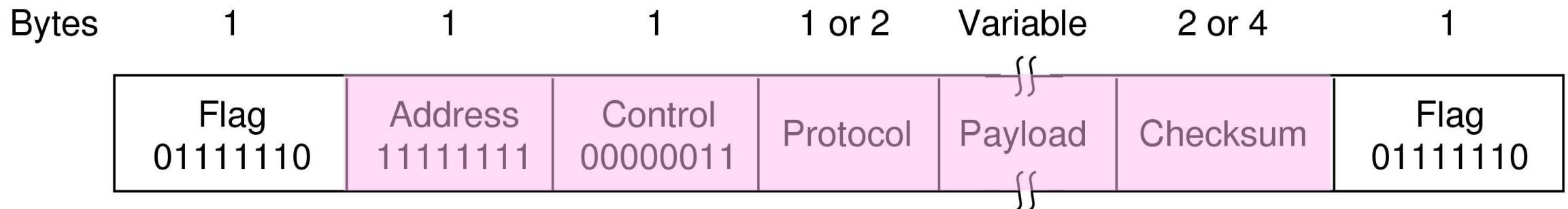
Link Example: PPP over SONET (2)

- Think of SONET as a bit stream, and PPP as the framing that carries an IP packet over the link



Link Example: PPP over SONET (3)

- Framing uses byte stuffing
 - **FLAG** is 0x7E and **ESC** is 0x7D



Link Example: PPP over SONET (4)

- Byte stuffing method:
 - To stuff (unstuff) a byte
 - add (remove) ESC (0x7D)
 - and XOR byte with 0x20
 - Removes **FLAG** from the contents of the frame