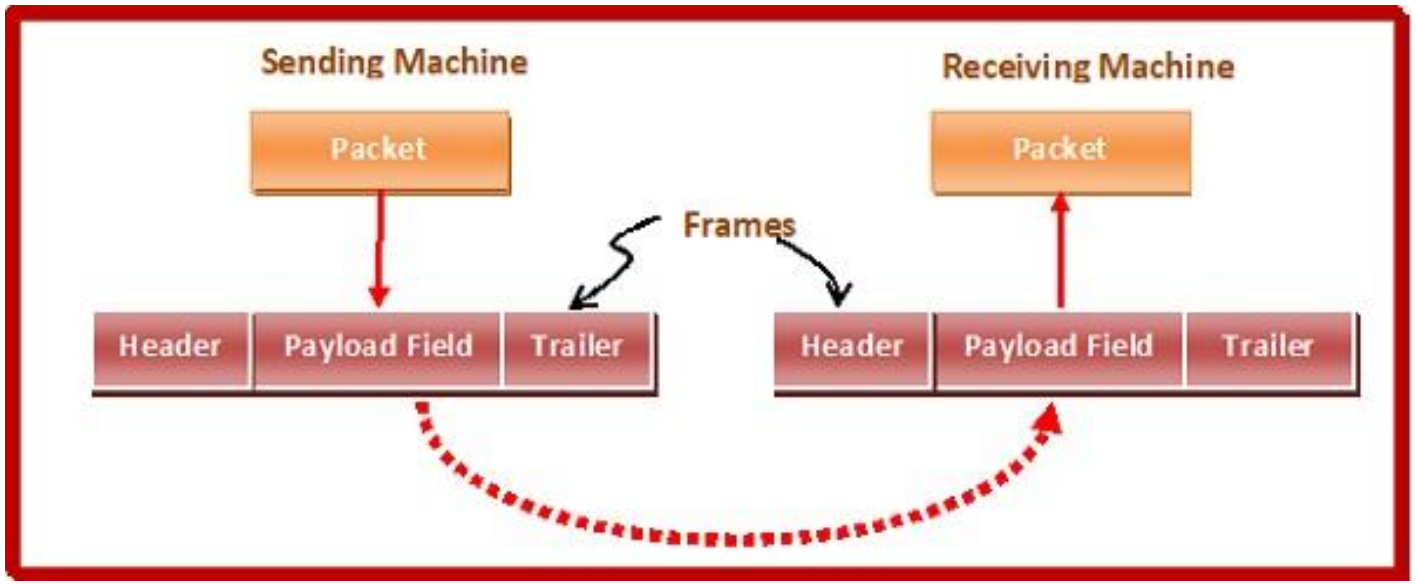
FRAMING

- Framing in the data link layer separates a message from one source to a destination, or from other messages to other destinations, by adding a sender address and a destination address. The destination address defines where the packet is to go; the sender address helps the recipient acknowledge the receipt.

- Although the whole message could be packed in one frame, that is not normally done. One reason is that a frame can be very large, making flow and error control very inefficient. When a message is carried in one very large frame, even a single-bit error would require the retransmission of the whole message. When a message is divided into smaller frames, a single-bit error affects only that small frame.

FRAMING

- Data-link layer takes the packets from the Network Layer and encapsulates them into frames.



Parts of a Frame

A frame has the following parts −

1. Frame Header − It contains the source and the destination addresses of the frame.
2. Payload field − It contains the message to be delivered.
3. Trailer − It contains the error detection and error correction bits.
4. Flag − It marks the beginning and end of the frame.

Types of Framing

Framing can be of two types :

1. Fixed sized framing and
2. Variable sized framing.

Fixed-sized Framing

- Here the size of the frame is fixed and so the frame length acts as delimiter of the frame. Consequently, it does not require additional boundary bits to identify the start and end of the frame.

Example

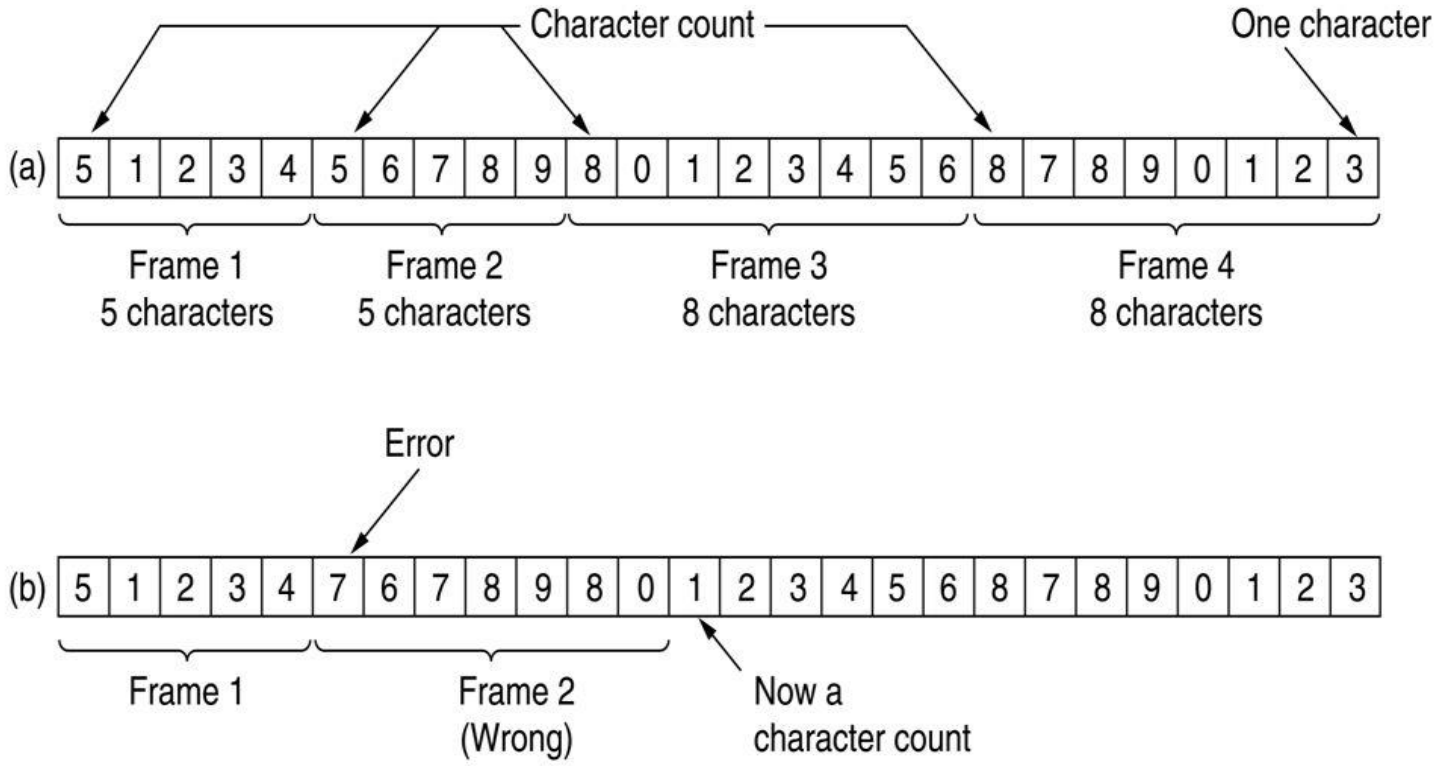


Fig : Character Count a) With out Errors b) With Errors

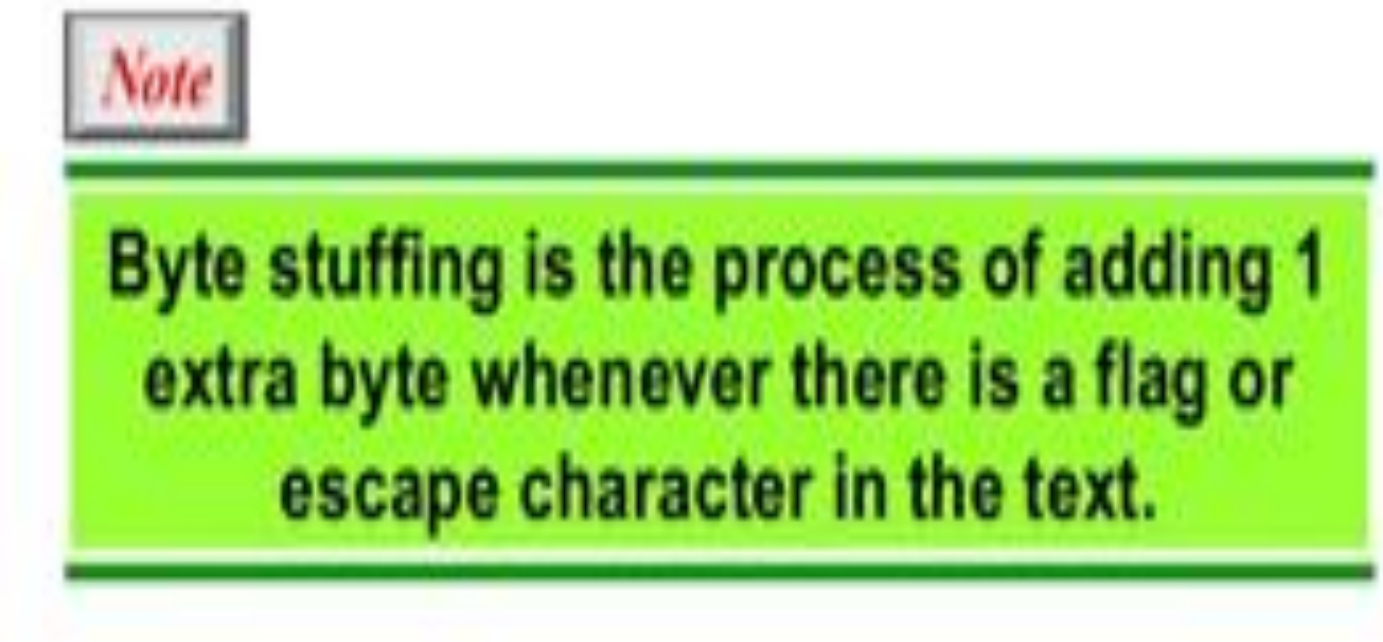
Variable – Sized Framing

- Here, the size of each frame to be transmitted may be different. So additional mechanisms are kept to mark the end of one frame and the beginning of the next frame.

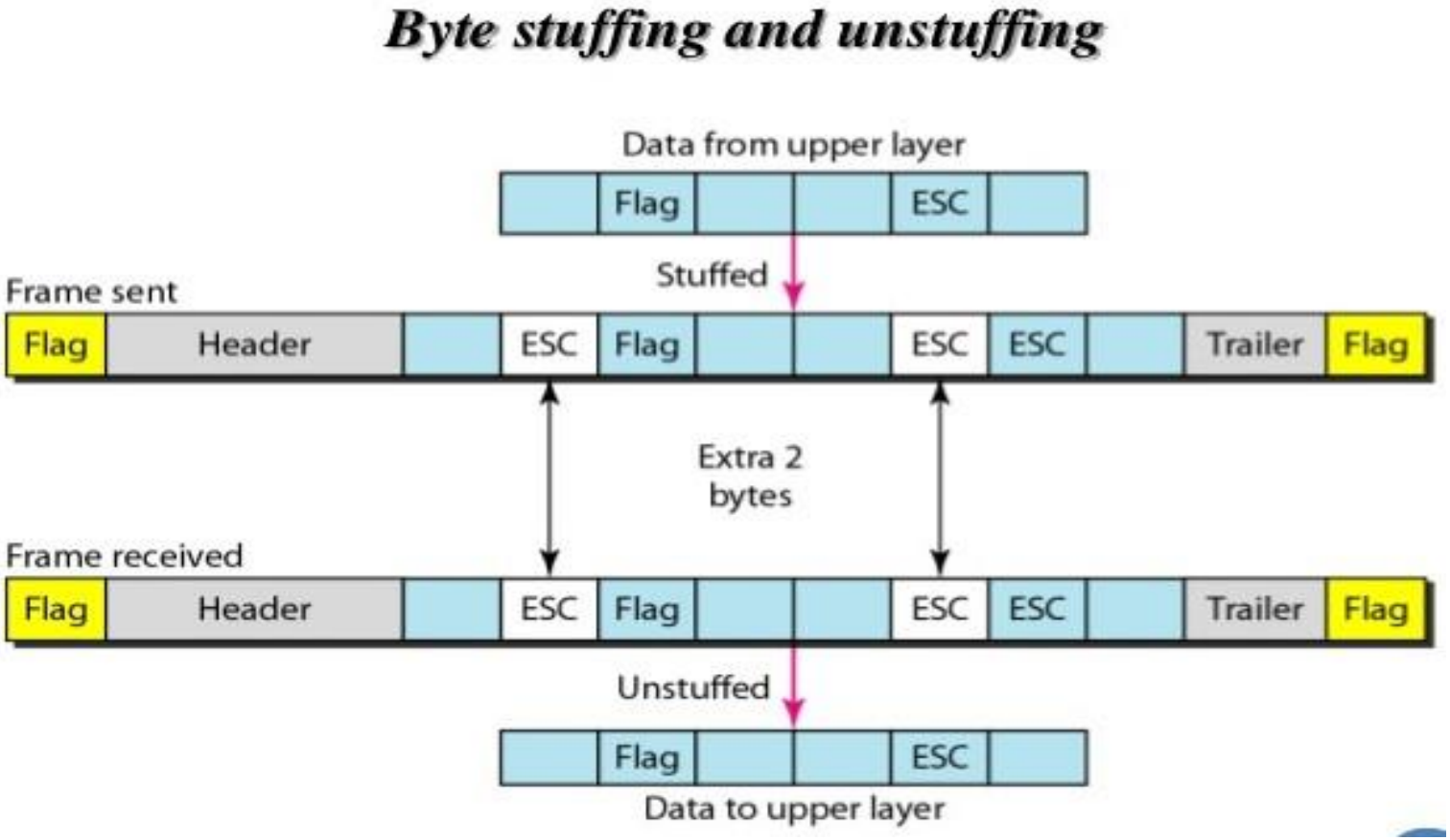
- It is used in local area networks.

- Historically, two approaches were used for this purpose: a character-oriented approach and a bit-oriented approach.

Character-oriented approach OR Byte Stuffing



Byte Stuffing Example

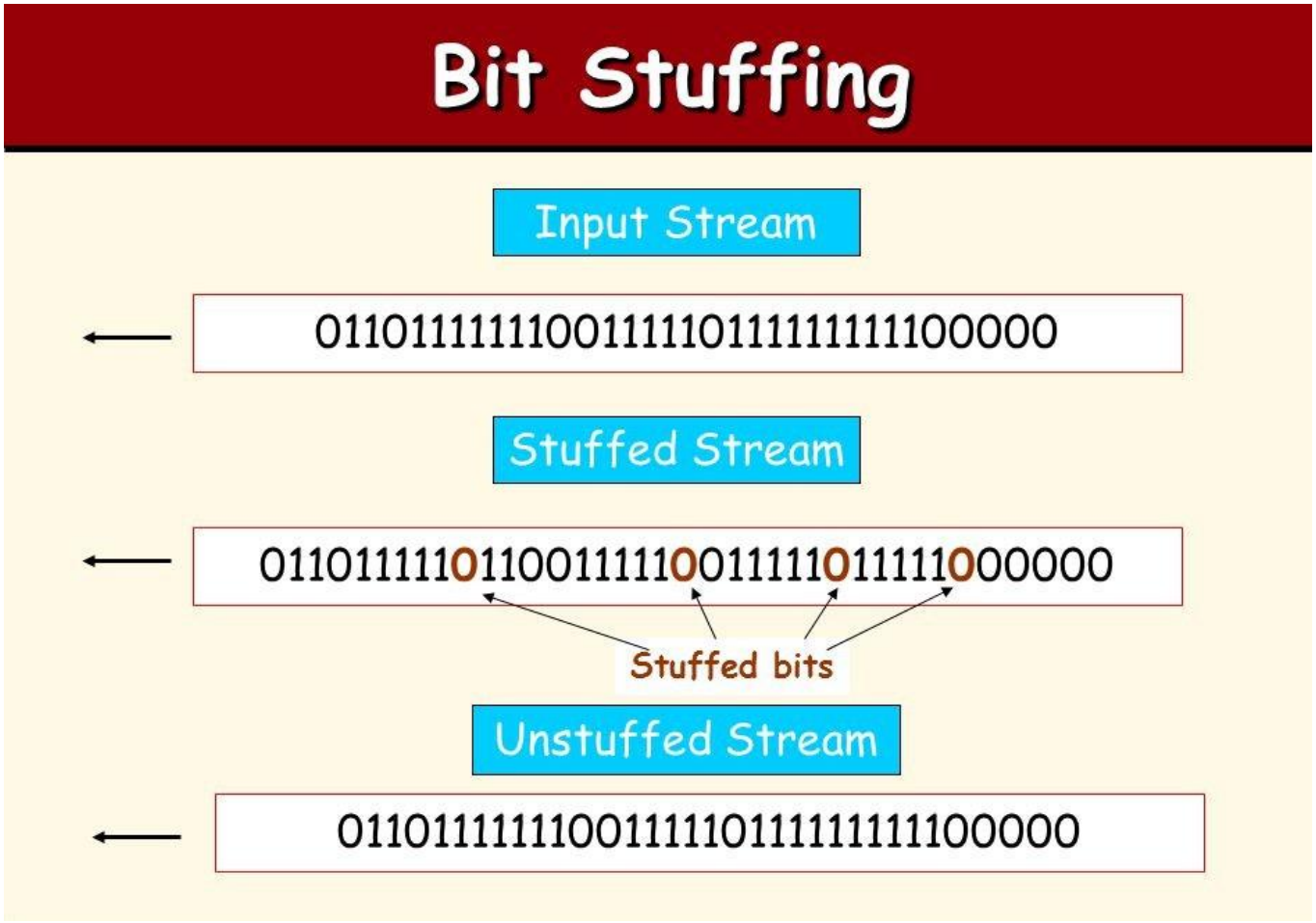


Bit Stuffing

Note

Bit stuffing is the process of adding one extra 0 whenever five consecutive 1s follow a 0 in the data, so that the receiver does not mistake the pattern 0111110 for a flag.

Bit Stuffing Example



Bit Stuffing

- Process of adding extra bit to ensure flag sequence does not appear in the data