Flow Control

- In computer Networking, **flow control** is the process of managing the rate of data transmission between two nodes to prevent a fast sender from outrunning a slow receiver.
- It provides a mechanism for the receiver to control the transmission speed, so that the receiving node is not overwhelmed with data from tranceiving nodes.

Hardware flow control

- In common RS 232 there are pairs of control lines:
- **RTS flow control**, RTS (Request To Send)/CTS (Clear To Send) and
- **DTR flow control**, DTR (Data Terminal Ready)/DSR (Data Set Ready),

Software flow control

- **Open-loop flow control**

This simple means of control is widely used.
The allocation of resources must be a “prior reservation” or “hop-to-hop” type.

Resource allocation is made at connection setup using a CAC (Connection Admission Control).

Example: ATM CBT, VBR, ABR, UBR & QoS

- **Closed-loop flow control**
- This information is then used by the transmitter in various ways to adapt its activity to existing network conditions.
- Characterized by ability of the congestion of the network to report pending network congestion.
- Congestion Procedures. Drop Tail, RED, WRED etc

Congestion Avoidance Procedures

Drop Tail

Tail drop is quite straight forward.

Principle

- Traffic is not classified or differentiated everything is treated the same.
- When a queue starts fill, any arriving packet dropped until potential congestion is eliminated.
- When a packet arrives at a full buffer, the arriving packet is discarded

Advantages

- Easy to implement can limit the number of packet losses for large buffer

Disadvantages

- No distinction between the various flows

Random Early Detection (RED)

- **Goals:**
- Work was done on TCP early 1990s to address the problems of congestions.
- The idea is to be proactive to the potential problems rather than just only reactive, as a the tail drop procedure.
- Proactive stance requires that TCP sender slow down its transmission rate, when packets are lost en-rout the receiver.
- This idea is called RED.
- Even though VOIP does not use TCP, it will be helpful to show how TCP/RED operates as Precursor or to the discussion on WRED.

- RED inform a sending host that:

(a) Received traffic is correct.

(b) Some traffic is missing (or in error)

(c) Not additional data has been received but the receiver host is alive.

(d) Perhaps the host should slow down or speed up its transmission.

Weighted Random Early Detection (WRED)

- Uses the concept of RED and IP procedure field to support preferential (Special Treatment of high priority traffic.
- It can selectively discard low priority traffic during the period of high traffic load.
- Thus it performs different services for different classes of traffic.
- During the congestion WRED will drop flows other than RSVP flows.
- Assuming it has sufficient available queue depth.

Weighted Fair Queuing" **scheduler (WFQ).**

- To put a packet into a queue:
- Class one code Best-effort" services (queues)
- Class second code to select the "*premium queue"* (higher weighted than Best-effort"

- The premium class can be kept low, since WFQ will try to transmit premium packets as soon as possible.
- If premium load ~10% . behaves as if premium traffic is running on an under-loaded network.
- If premium load ~30% . behaves like a highly loaded network.
- Just as in WRED, we can generalize this WFQ-based approach to allow more than two classes represented by different code points.
- We can also combine the idea of a queue selector with a drop preference