24-1 DATA TRAFFIC

The main focus of congestion control and quality of service is data traffic. In congestion control we try to avoid traffic congestion. In quality of service, we try to create an appropriate environment for the traffic. So, before talking about congestion control and quality of service, we discuss the data traffic itself.

Figure 24.1 *Traffic descriptors*

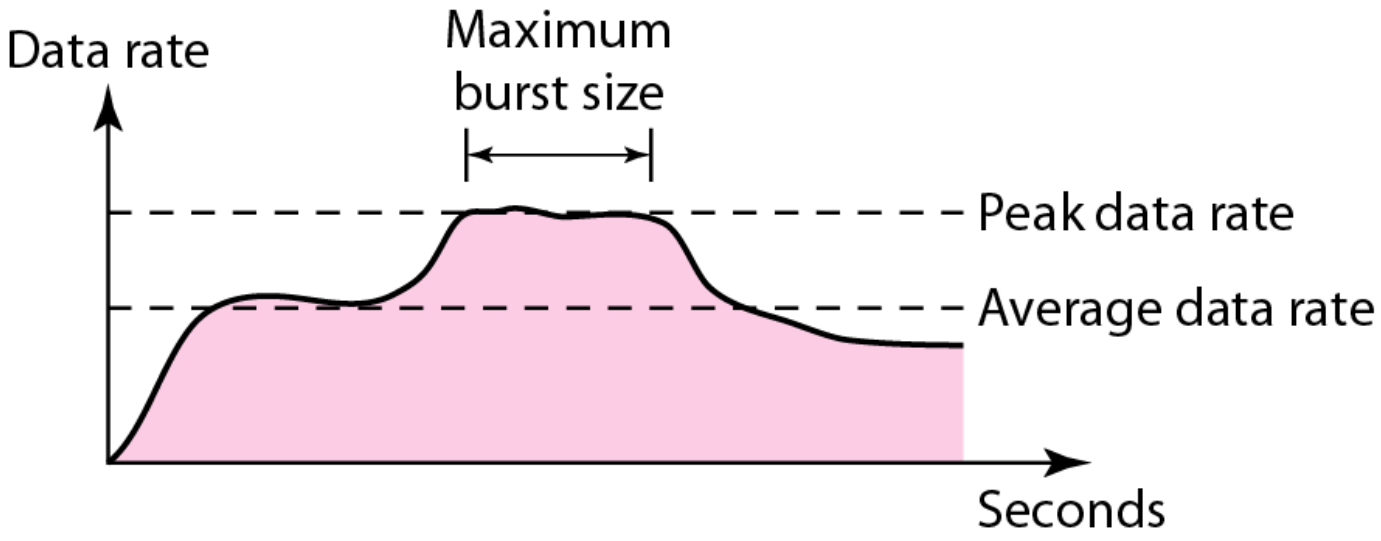
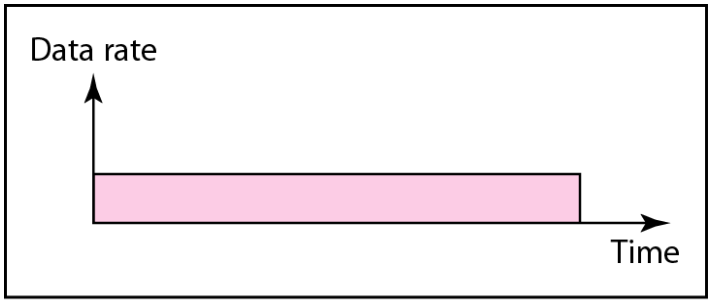
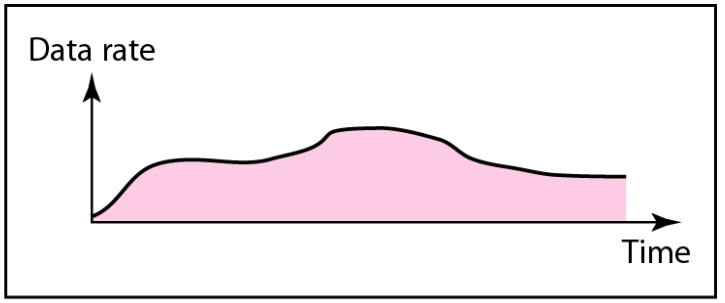


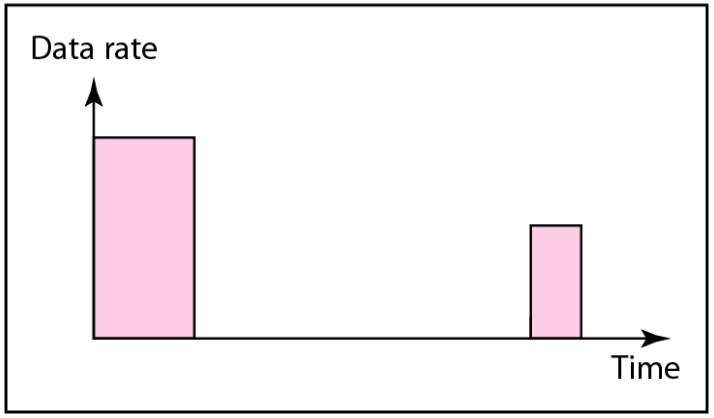
Figure 24.2 *Three traffic profiles*



a. Constant bit rate



b. Variable bit rate



c. Bursty

Poll

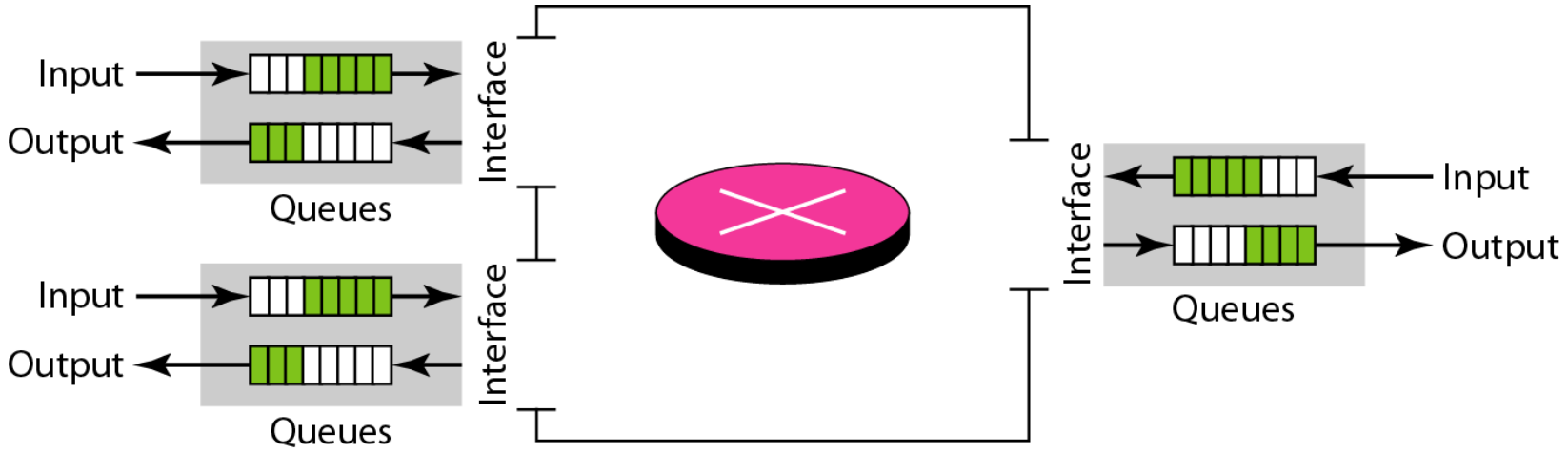
Traffic profiles are

A. Constant
B. Variable
C. Bursty
D. All of the above

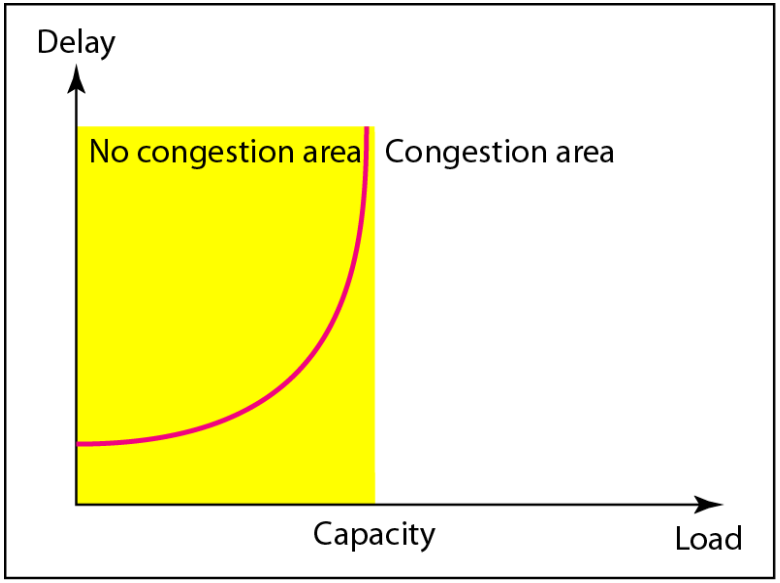
24-2 CONGESTION

Congestion in a network may occur if the load on the network—the number of packets sent to the network—is greater than the capacity of the network—the number of packets a network can handle. Congestion control refers to the mechanisms and techniques to control the congestion and keep the load below the capacity.

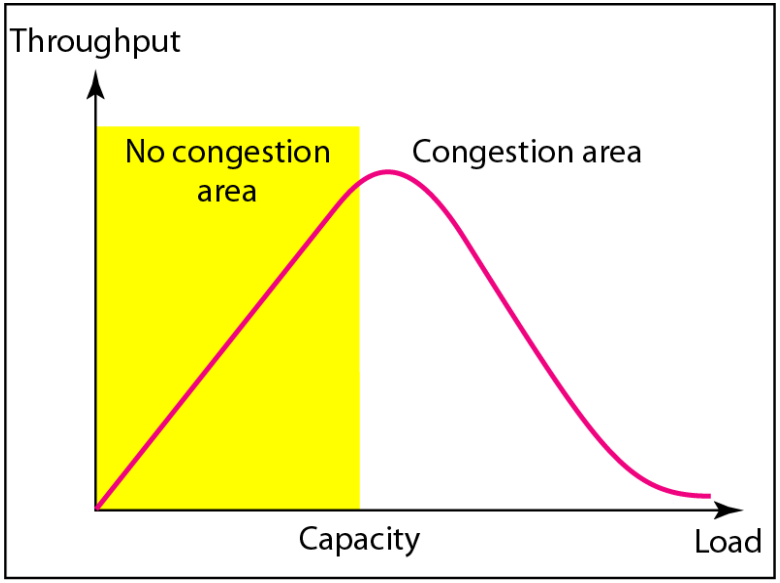
Figure 24.3 *Queues in a router*



Network Performance: *Delay and throughput as functions of load*



a. Delay as a function of load



b. Throughput as a function of load

Poll

Congestion control refers

A. *keep the load below the capacity*
B. *keep the load above the capacity*
C. A and b both
D. None of the above

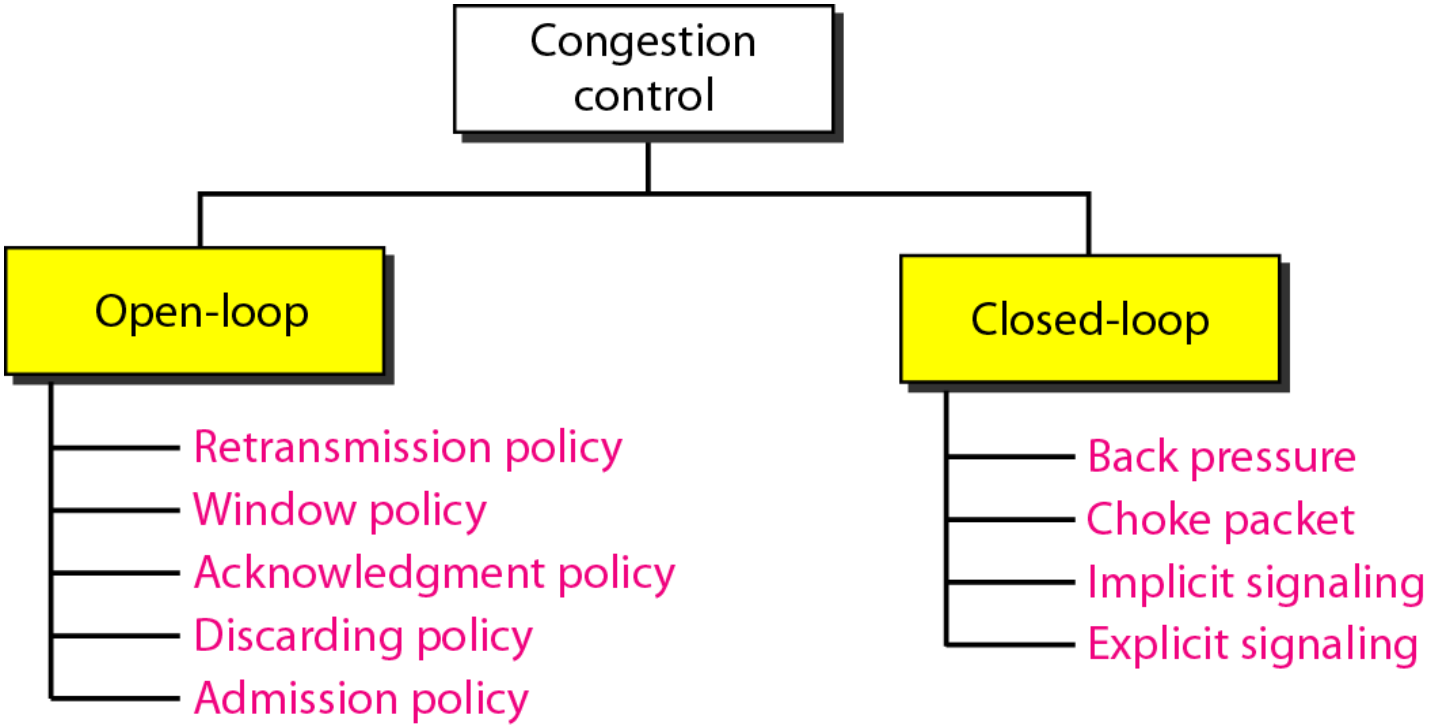
24-3 CONGESTION CONTROL

Congestion control refers to techniques and mechanisms that can either ***prevent congestion****, before it happens, or* ***remove congestion****, after it has happened. In general, we can divide congestion control mechanisms into two broad categories:* ***open-loop congestion control (prevention)*** *and* ***closed-loop congestion control (removal).***

Topics discussed in this section:

Open-Loop Congestion Control
Closed-Loop Congestion Control

Figure 24.5 *Congestion control categories*



Retransmission Policy

- When to retransmit
  - Lost Frame
  - Lost Ack
  - Damaged Frame

After Timer expiration

timer value too small– more retransmissions

timer value too large– delay in response to lost segment.

Window policy

- What will be maximum size of window- in case of sliding window protocol

  Either you implement

  - Go-Back N ARQ

  - Selective Repeat ARQ

```
Generally
Maximum window size = 1 + 2*a where a = Tp/Tt
```

Transmission Delay (Tt)

Propagation Delay (Tp)

Acknowledgement Policy

- Selective Ack
- Cumlative Ack

Discarding policy

- In case of congestion
  - Which packet to discard
    - Low priority
    - Newer Packet
    - Discard the packet which is NOT nearest to destination
    - When to discard
      - Make sure , you are NOT discarding those packets which are just about to enter the destination's network.

Admission Policy

- If there are insufficient resources to handle the transmission, then do not accept any more packets from sender.

- A router can deny establishing a virtual circuit connection if there is a congestion.

Solutions to Open Loop Congestion

Selective repeat window is better

Should send cumulative acknowledgement

Higher priority packets should not be discard

Closed Loop Congestion Control

- Backpressure
- Choke packet
- Implicit Signaling
- Explicit Signaling

Figure 24.6 *Backpressure method for alleviating congestion*

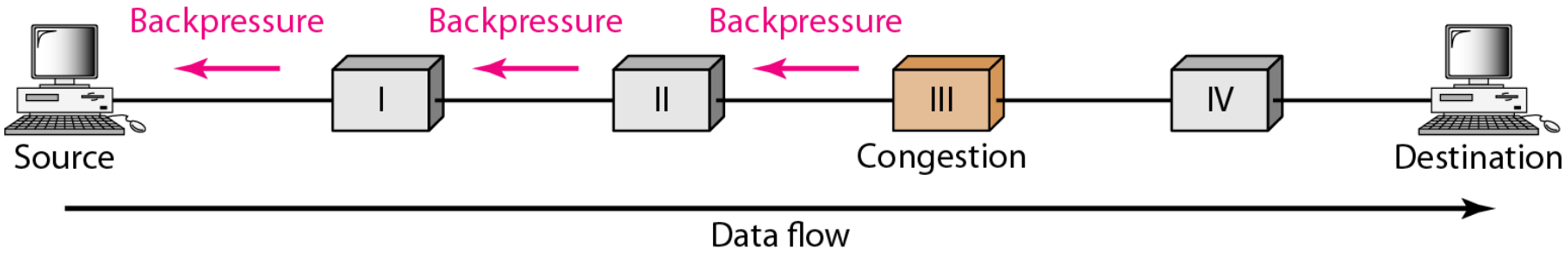
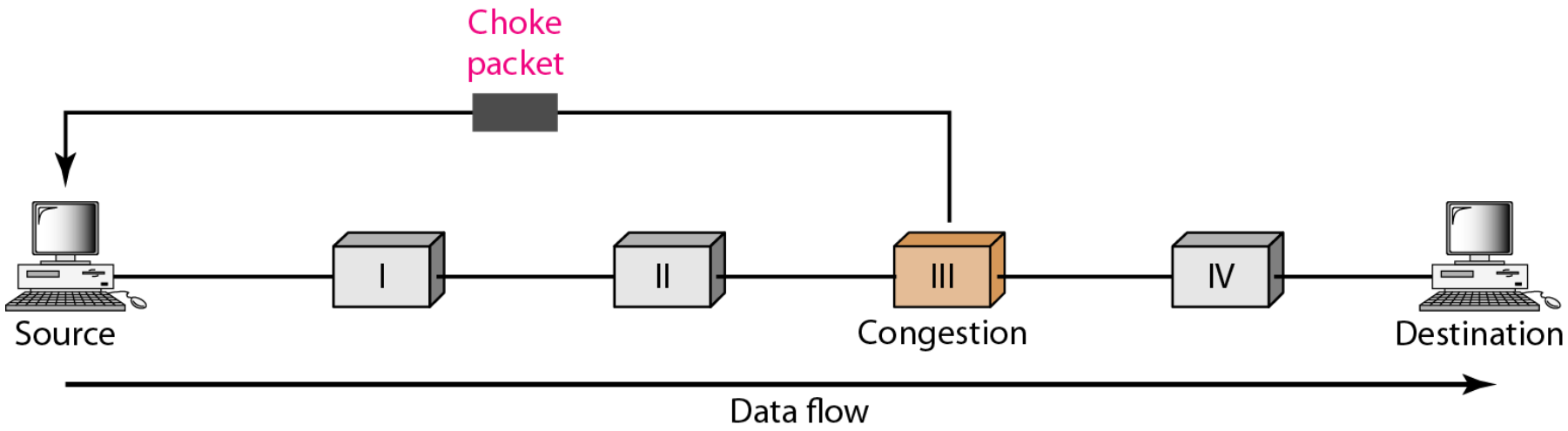


Figure 24.7 *Choke packet*



Congestion leads to –

discard of segments

transmission delay increase

Implicit Signaling- by symptom ***source*** will aware about congestion in network. No communication is send to source from congested node. For example, through delayed acknowledgments.

Explicit signal- network alerts sender or destination to slow down the rate of transmission by sending signal that is included in packet that carry data. That is why its different from choke packet.

- Backward signaling- warning source (opposite direction to congestion) and take appropriate congestion corrective measures

- Forward signaling- warning destination (same direction to congestion) and take appropriate congestion avoidance measures. Such as receiver can slow down the acknowledgment

Quality of service (QoS)

- To provide good quality of service is to build a network with enough capacity for whatever traffic will be thrown at it. The name for this solution is **over-provisioning**.

- Quality of service mechanisms let a network with less capacity meet application requirements just as well at a lower cost.

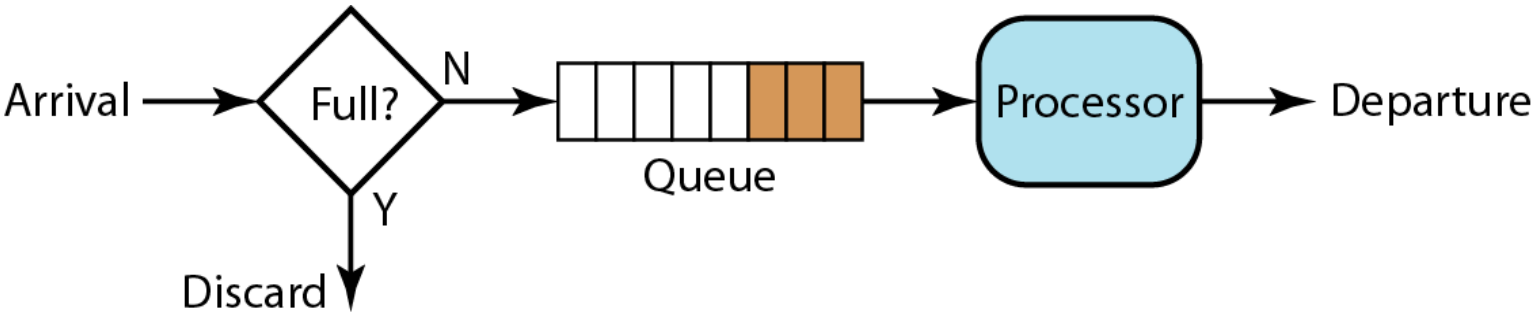
Flow Characteristics

- Reliability
- Delay
- Jitter– variation in delay
- Bandwidth

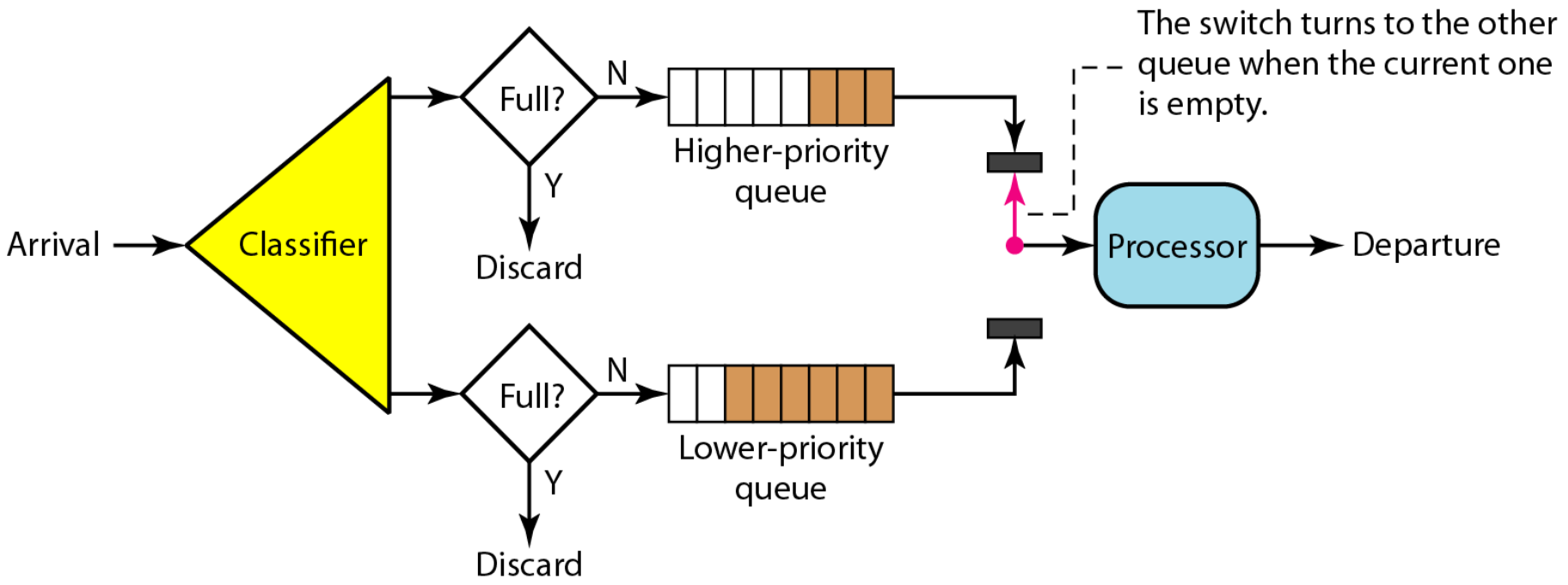
TECHNIQUES TO IMPROVE QoS

- Scheduling
  - FIFO Queuing
  - Priority Queuing
  - Weighted Fair Queuing
- Traffic Shaping
  - Leaky Bucket
  - Token Bucket
- Resource Reservation
- Admission Control

Scheduling : *FIFO queue*

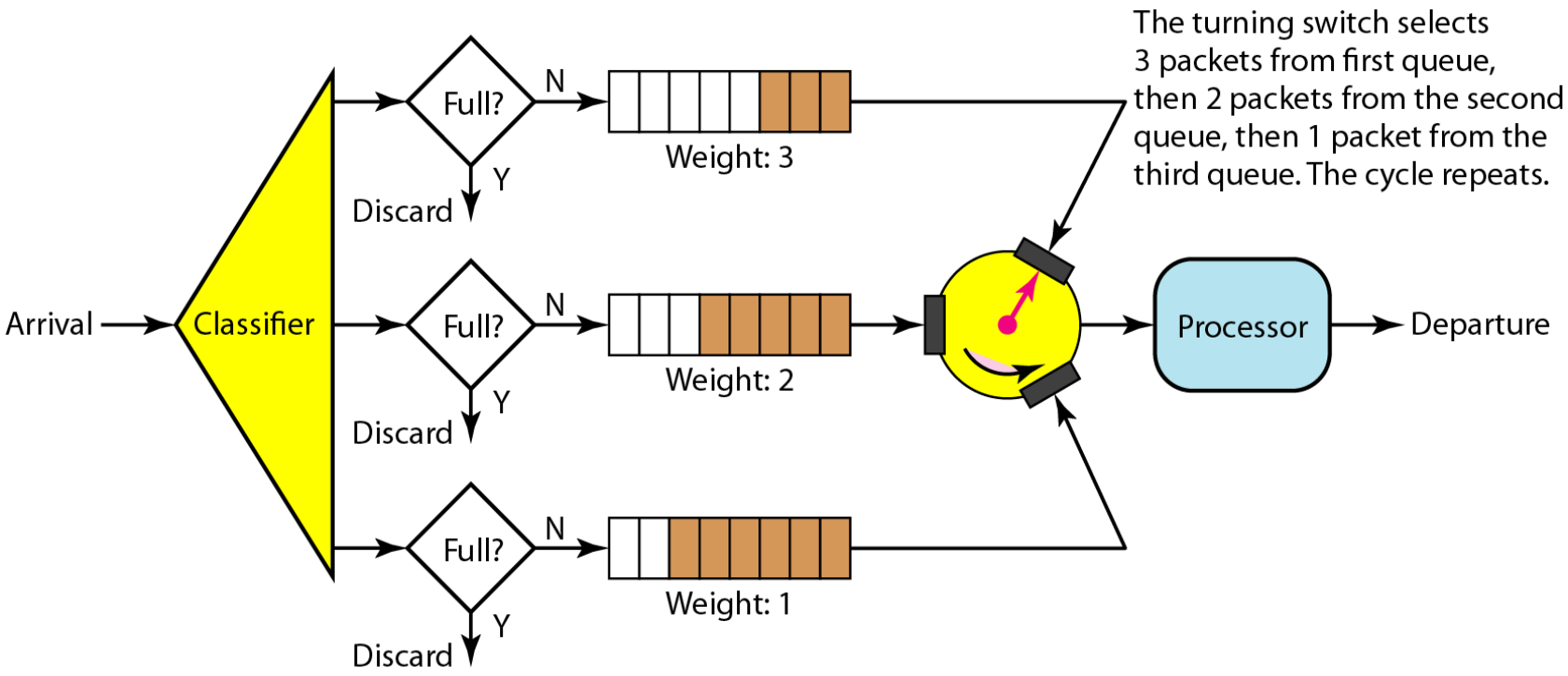


Scheduling : *Priority queuing*



- ***Priority queuing have problem of starvation***

Scheduling : *Weighted fair queuing*



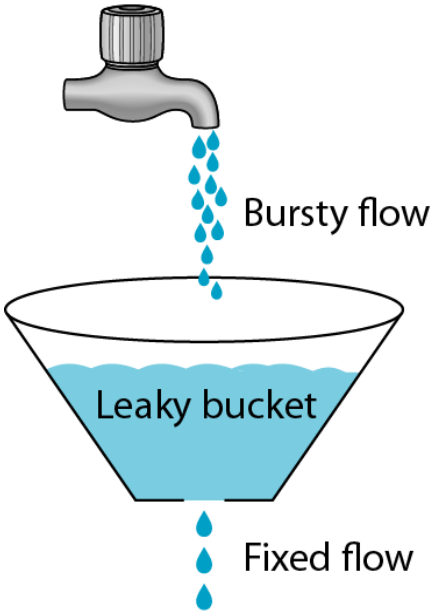
Traffic Shaping

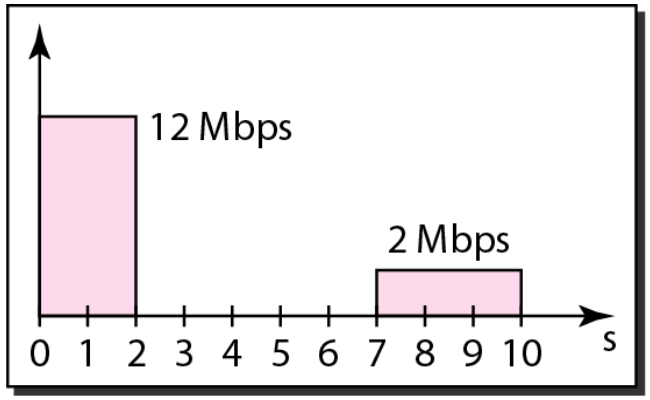
- **Traffic shaping** is a technique for regulating/ controling the average rate and burst-ness of a flow of data that enters the network.
- The goal is to allow applications to transmit a wide variety of traffic that suits their needs, including some bursts.
- **SLA** (**Service Level Agreement**)

Traffic Policing

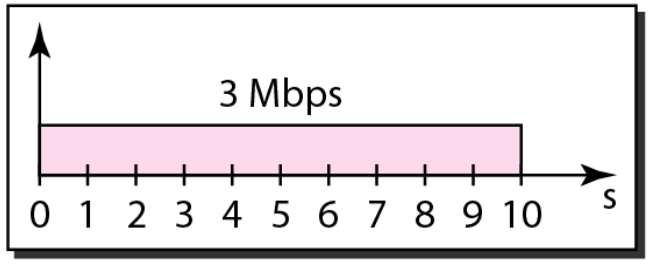
- Traffic shaping reduces congestion and thus helps the network live up to its promise.
- There is also the issue of how the provider can tell if the customer is following the agreement and what to do if the customer is not.
- Packets in excess of the agreed pattern might be dropped by the network, or they might be marked as having lower priority.
- Monitoring a traffic flow is called **traffic policing**.

Traffic Shaping : *Leaky bucket*

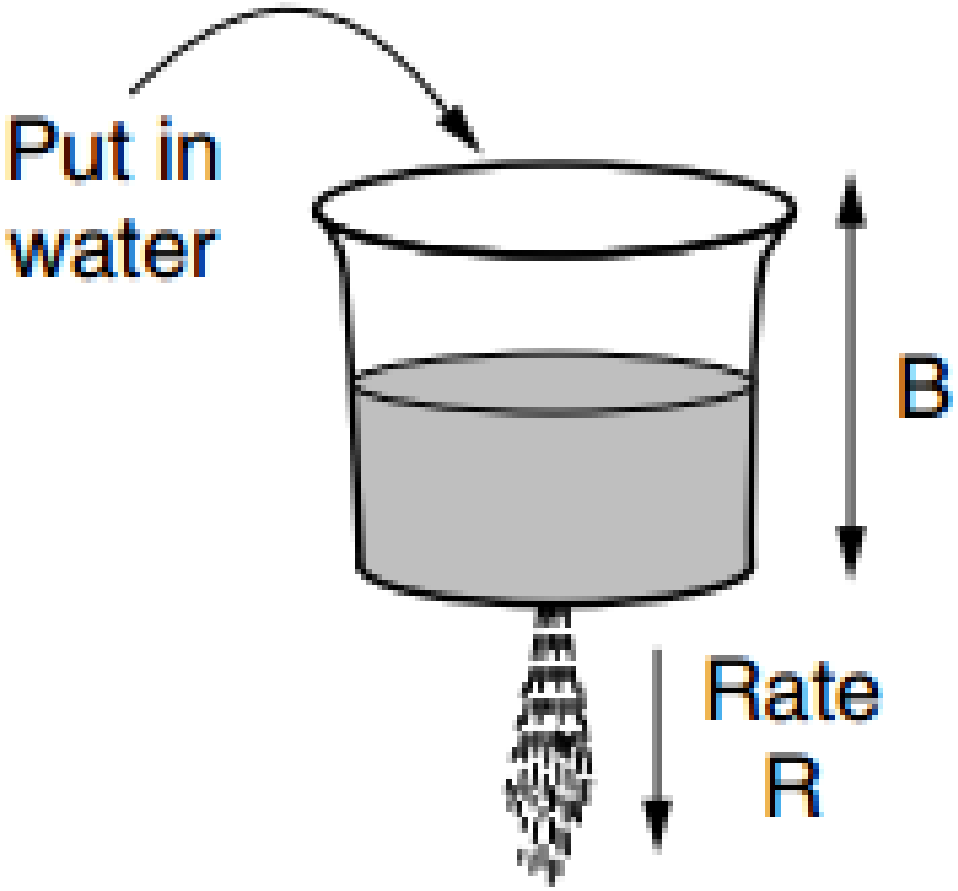




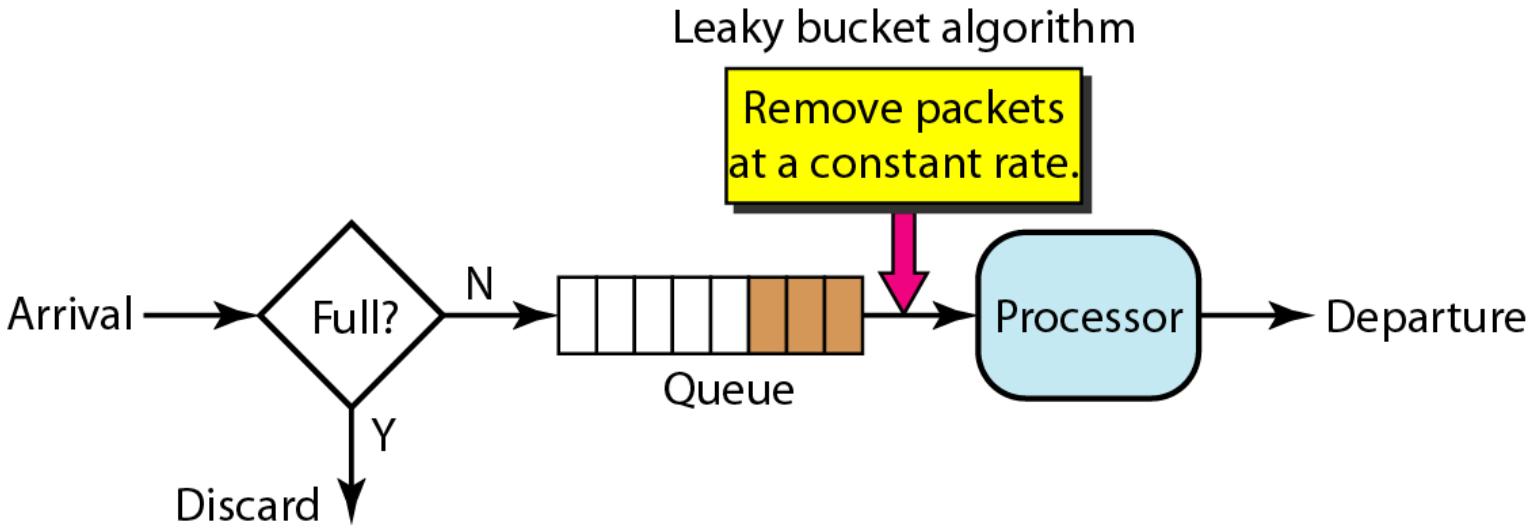
Bursty data



Fixed-rate data


Put in
water
B
Rate
R

Traffic Shaping : *Leaky bucket implementation*



Note

A leaky bucket algorithm shapes bursty traffic into fixed-rate traffic by averaging the data rate. It may drop the packets if the bucket is full.

Note

The token bucket allows bursty traffic at a regulated maximum rate.

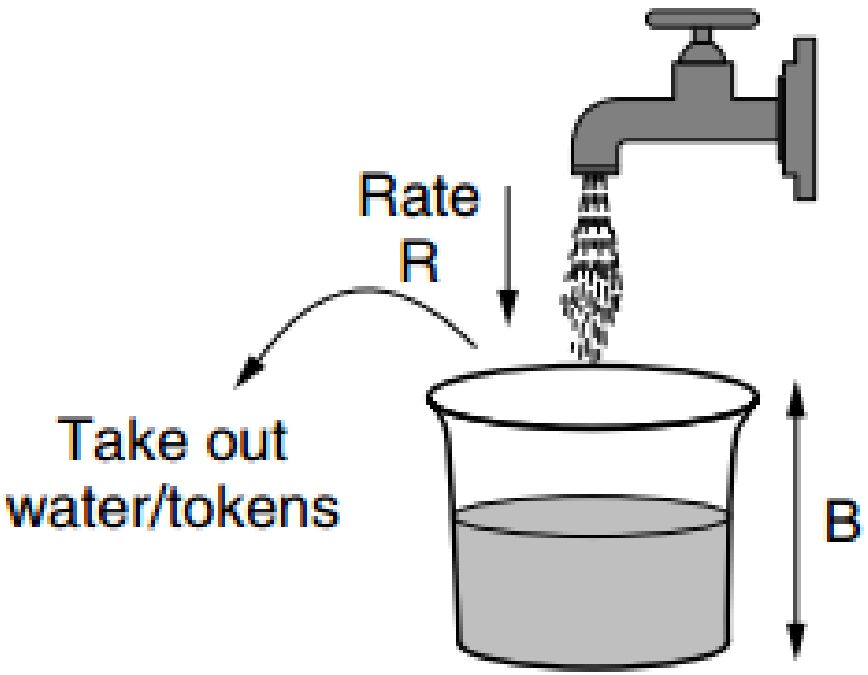

Rate
R
Take out
water/tokens
B

Figure 24.21 *Token bucket*