



Chapter 24

拥塞控制和服务质量

24-1 数据通信量

对拥塞控制和服务质量关注的重要方面是数据通信量。在拥塞控制中，要设法避免通信量拥塞。在服务质量中，要设法为通信量创造合理的环境。因此，在讨论拥塞控制和服务质量之前，首先讨论数据通信量本身。

本节要点

通信量描述符
通信量特征值

图 24.1 通信量描述符

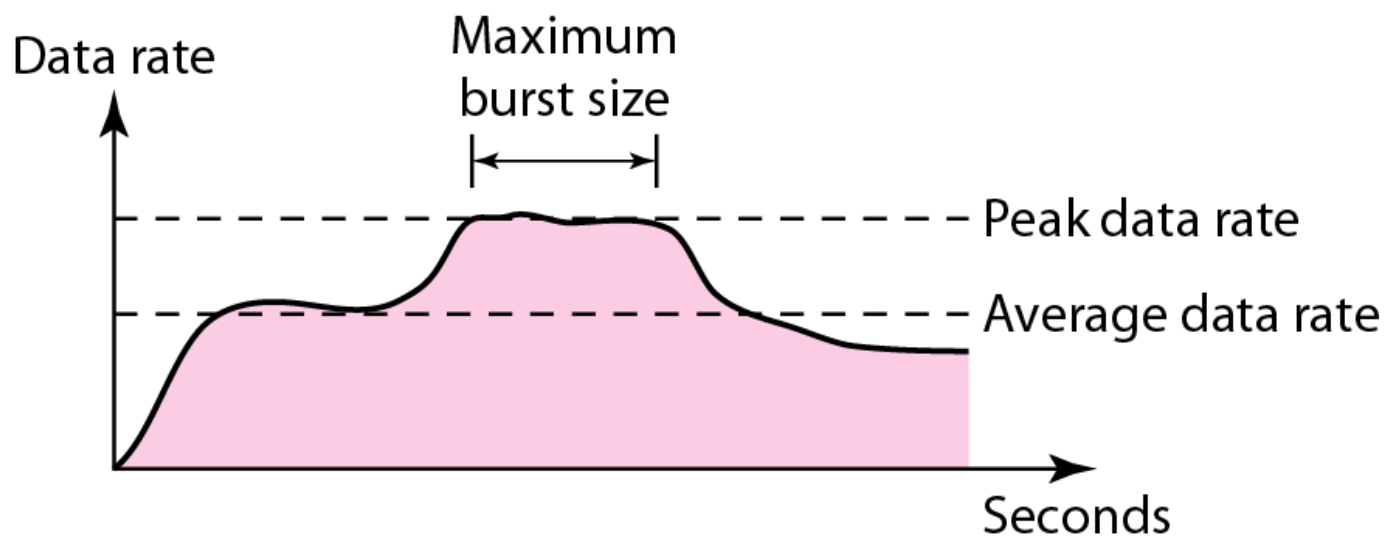
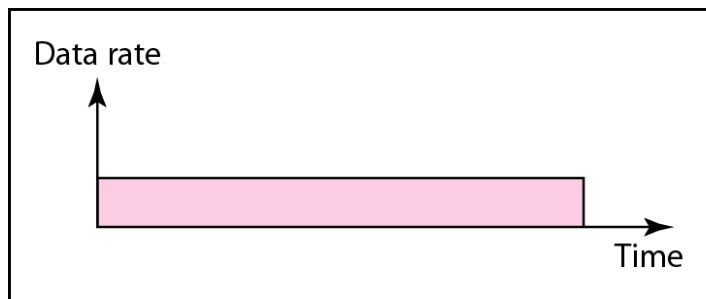
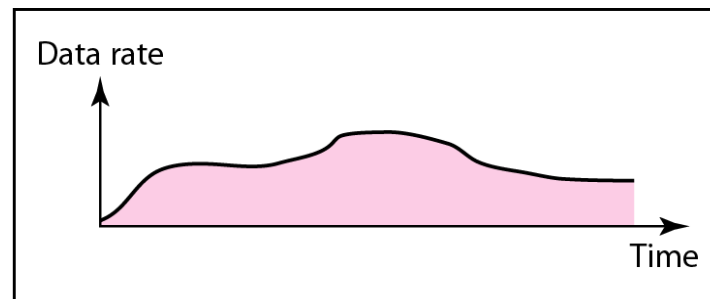


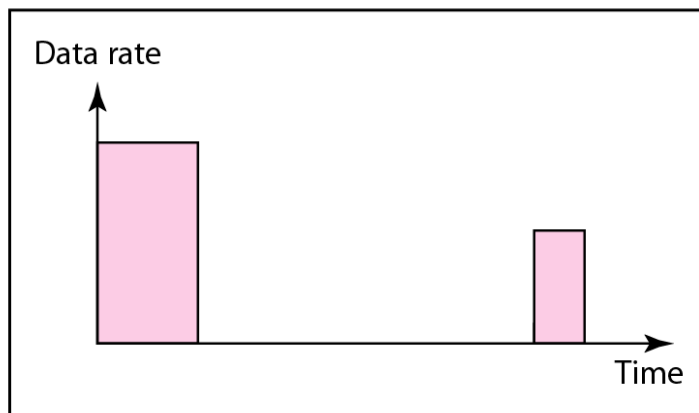
图 24.2 三个通信量特征值



a. Constant bit rate



b. Variable bit rate



c. Bursty

24-2 拥塞

拥塞(congestion) 是分组交换网络中的一个重要问题。如果网络中的载荷(load) , 即发送到网络中的分组数量, 超过了网络的容量, 即网络中能处理的分组数量, 那么在网络中就可能发生拥塞。拥塞控制(congestion control) 指的是控制拥塞和使载荷低于网络容量的机制和技术。

本节要点:

网络的性能

图 24.3 路由器中的队列

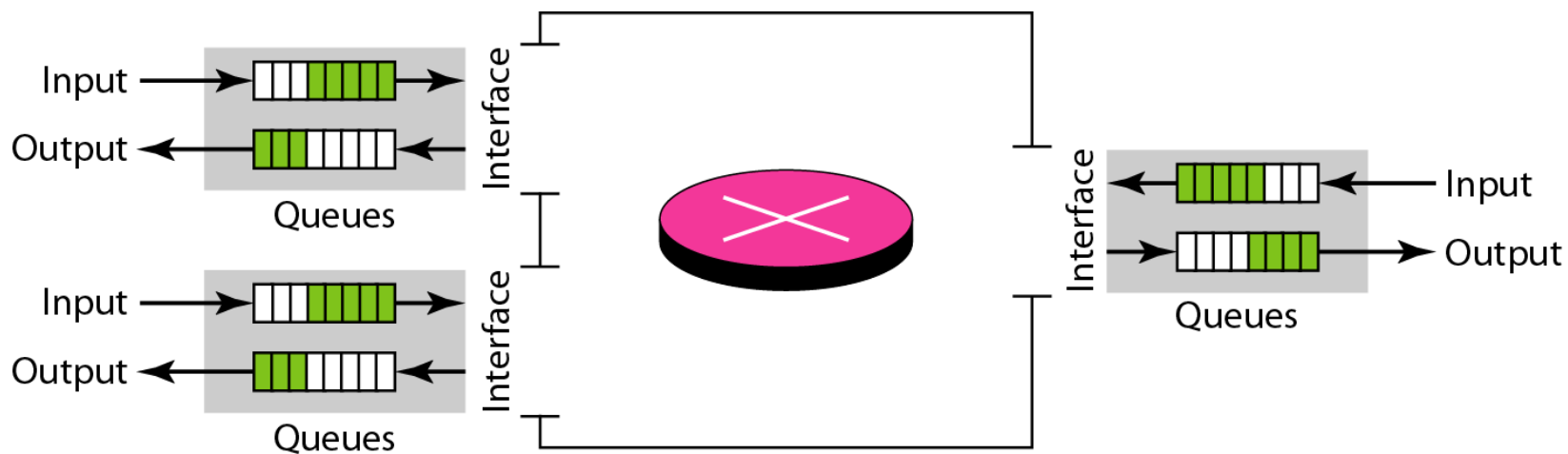
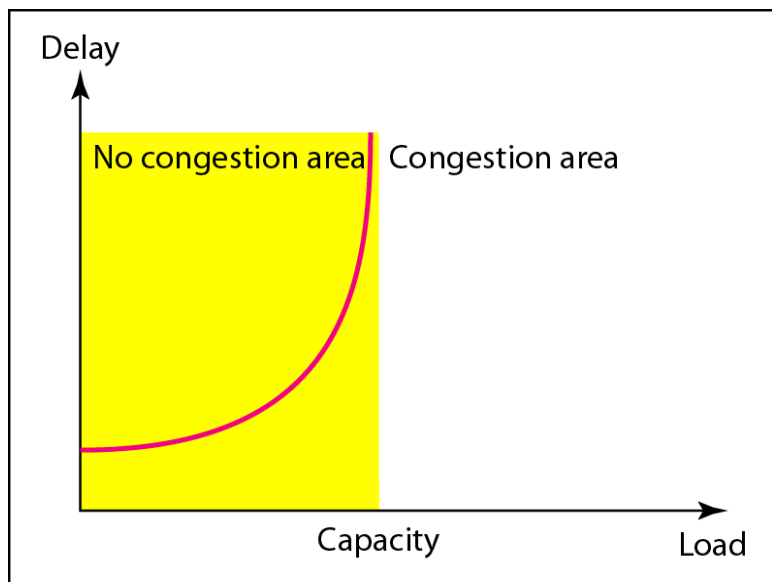
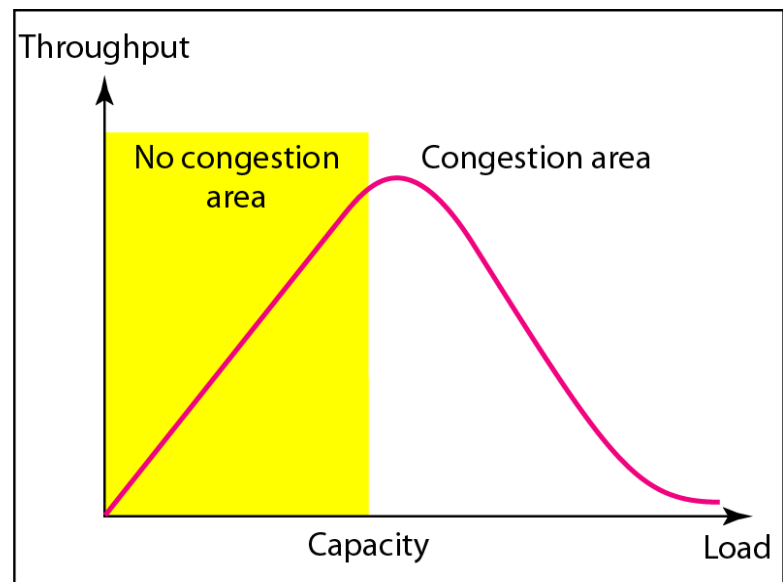


Figure 24.4 分组延迟和吞吐量是网络载荷的函数



a. Delay as a function of load



b. Throughput as a function of load

吞吐量：单位时间内通过网络的分组数

24-3 拥塞控制

拥塞控制是指在拥塞发生之前预防拥塞、或在拥塞发生之后消除拥塞的技术和机制。通常，我们把拥塞控制分为两大类：开环拥塞控制(预防)和闭环拥塞控制(消除)。

本节要点:

开环拥塞控制
闭环拥塞控制

图 24.5 拥塞控制分类

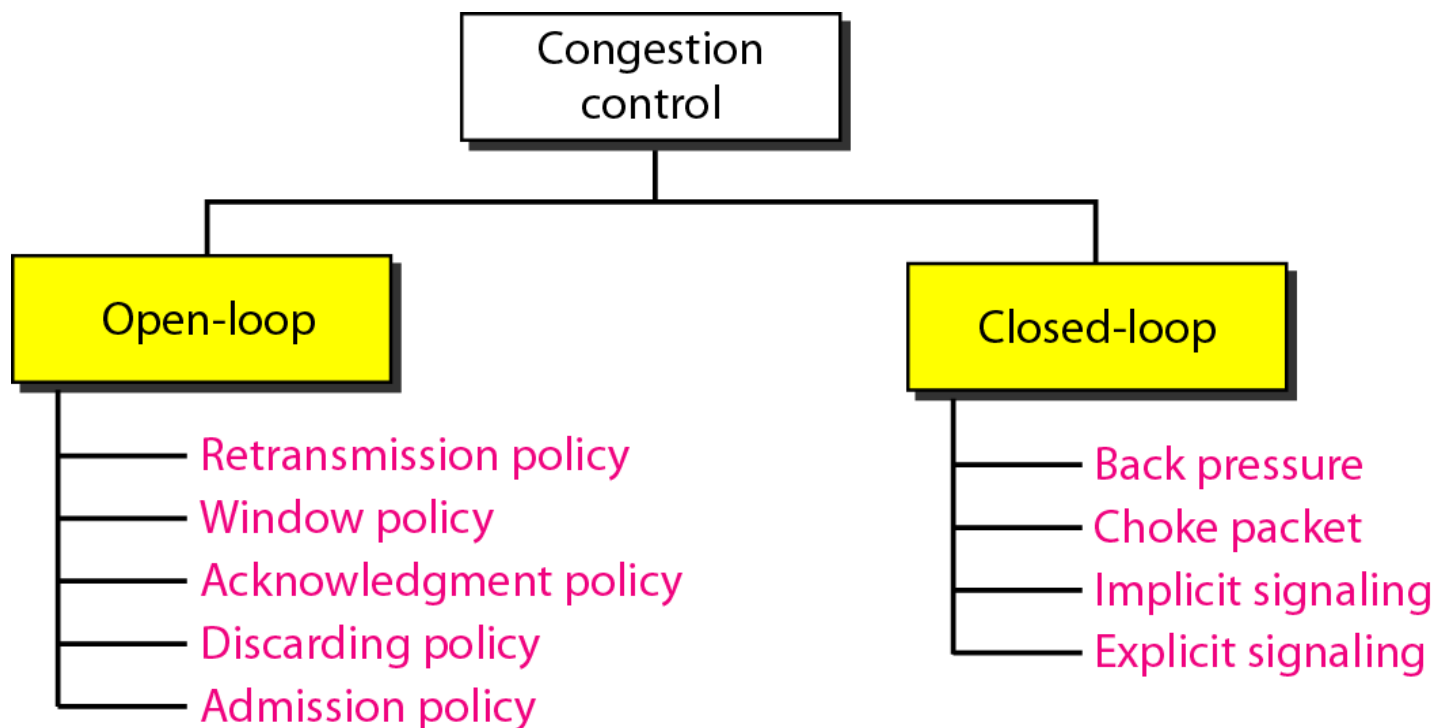


Figure 24.6 减轻拥塞背压（反压）方法

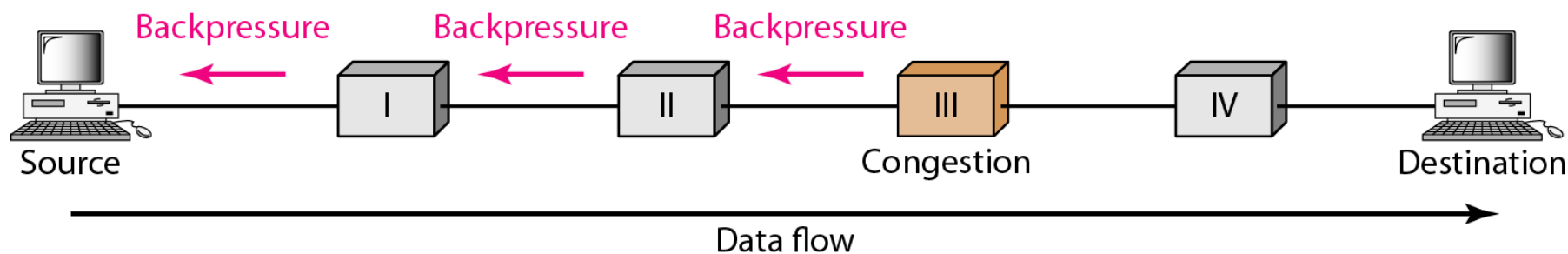
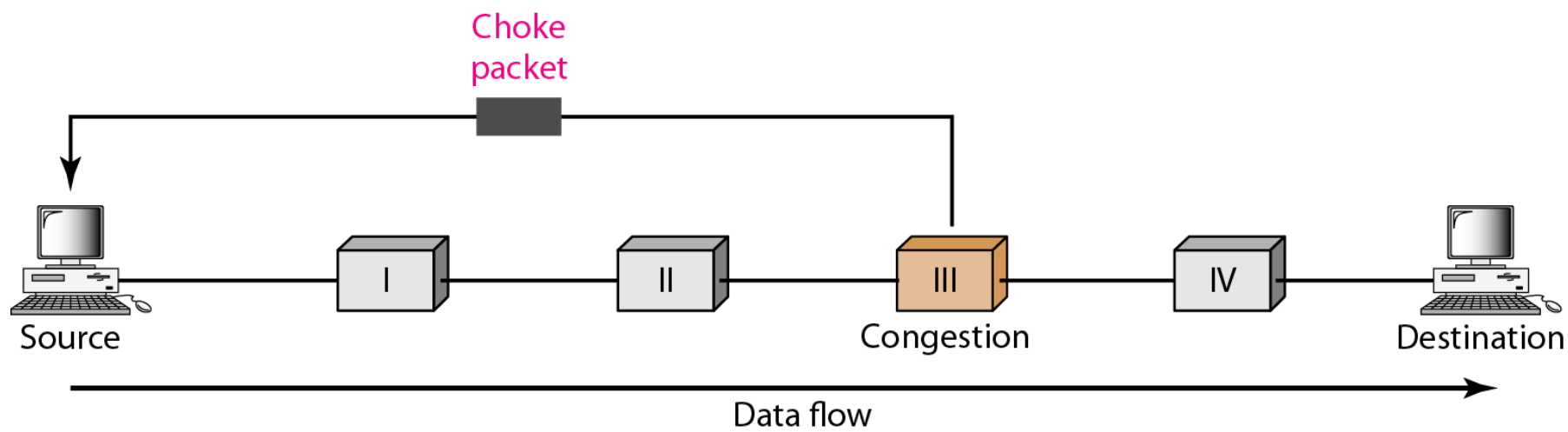


图 24.7 抑制分组



24-4 两个示例

为了更好地理解拥塞控制的概念，我们举两个例子：一个是**TCP** 中的拥塞控制，另一个是帧中继中的拥塞控制。

本节要点:

TCP 中的拥塞控制
帧中继中的拥塞控制

拥塞窗口

- 接收方窗口 rwnd
- 拥塞窗口 cwnd
- 实际窗口 $= \min(\text{rwnd}, \text{cwnd})$
- 拥塞策略：
 - 慢启动
 - 拥塞避免
 - 拥塞检测



注意

在慢速启动算法中，
拥塞窗口大小按指数规律增长
直到达到阈值。



注意

在拥塞避免算法中，
拥塞窗口大小是加性增加的直到检测到拥塞。
。



注意

用下列方法之一对拥塞检测做出反应:

- 如果检测址:计时器到时， 那么一个新的慢速启动阶段开始。
- 如果检测出三个**ACK**， 那么一个新的拥塞避免阶段开始。

图 24.11 拥塞例子

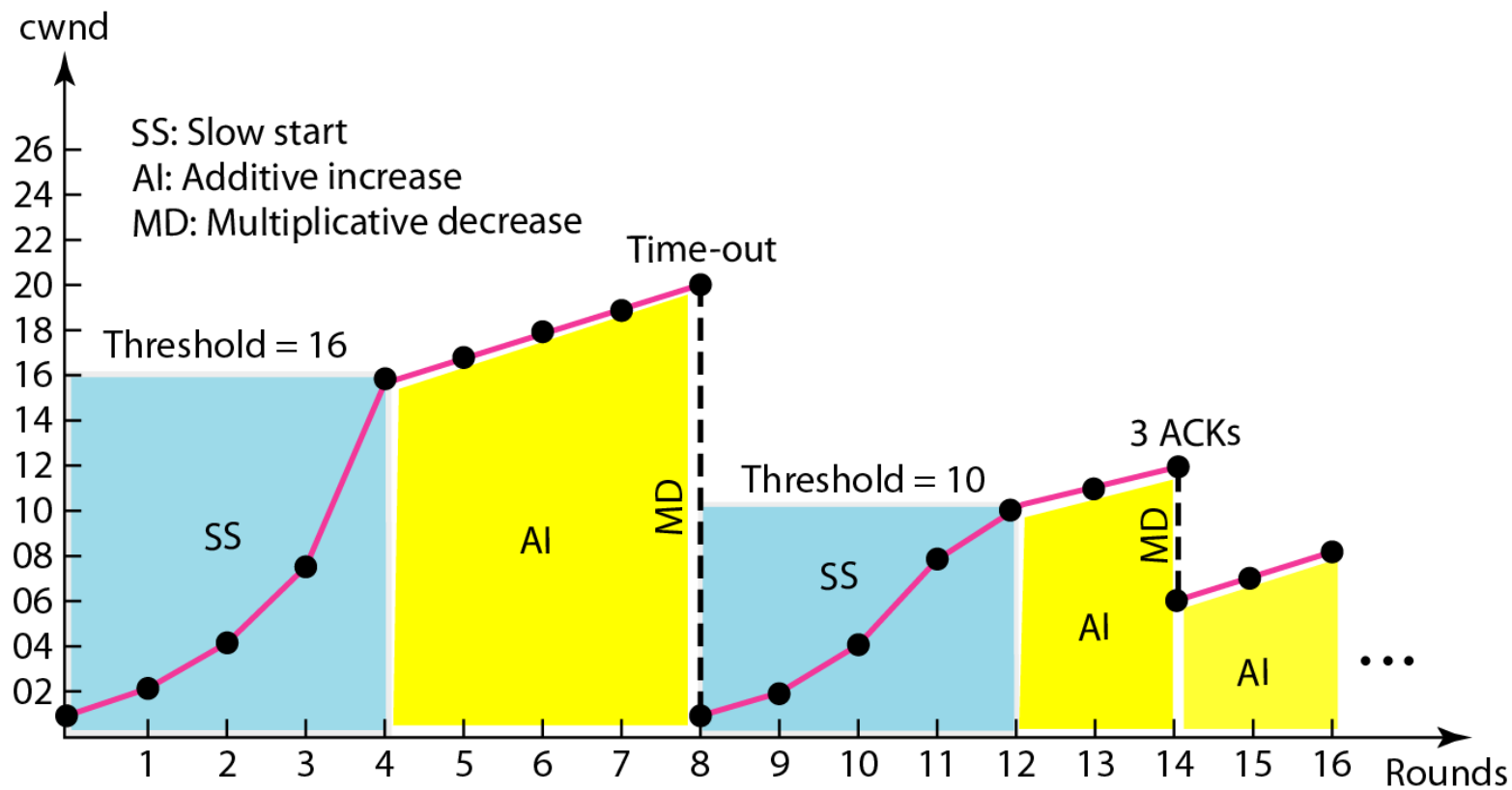


图 24.8 慢速启动: 指数增加

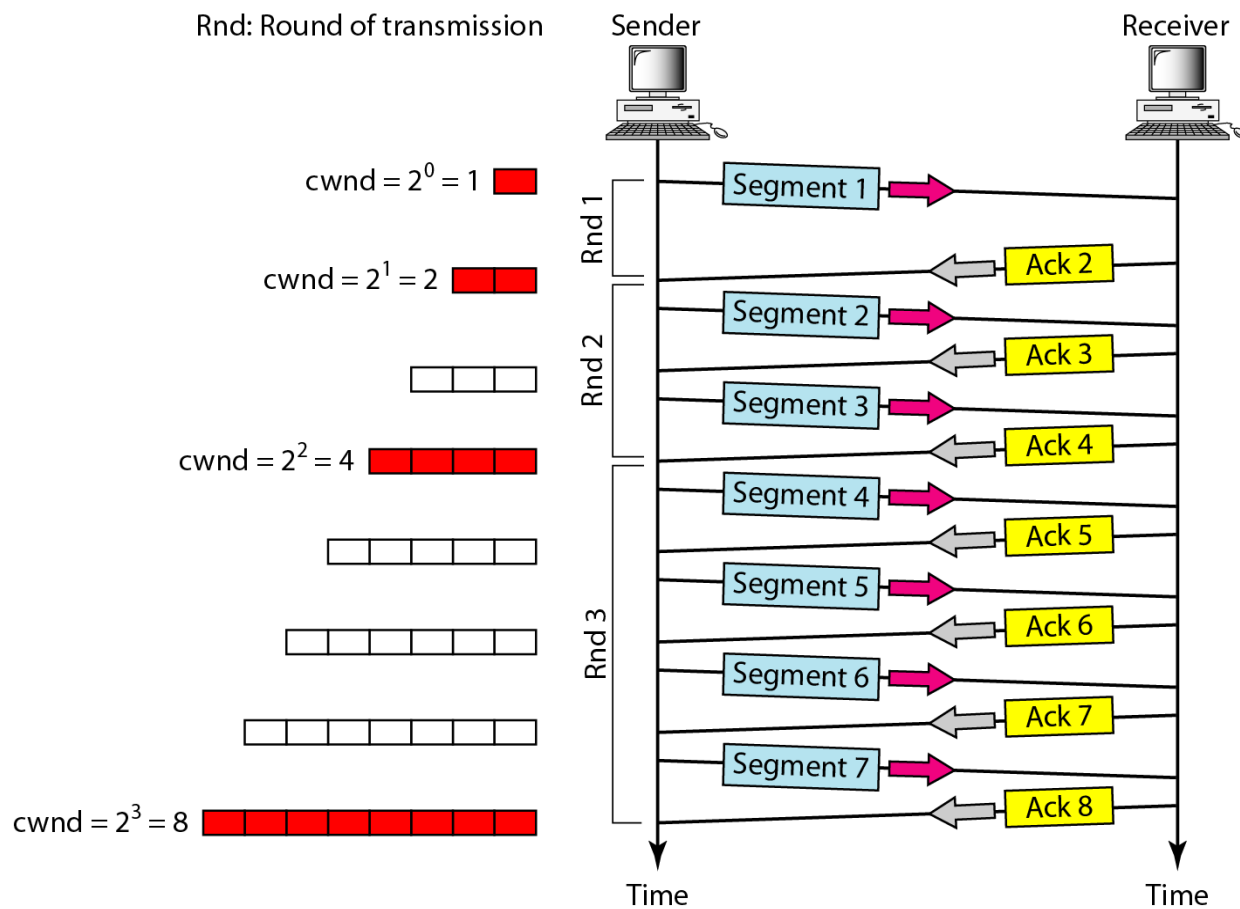
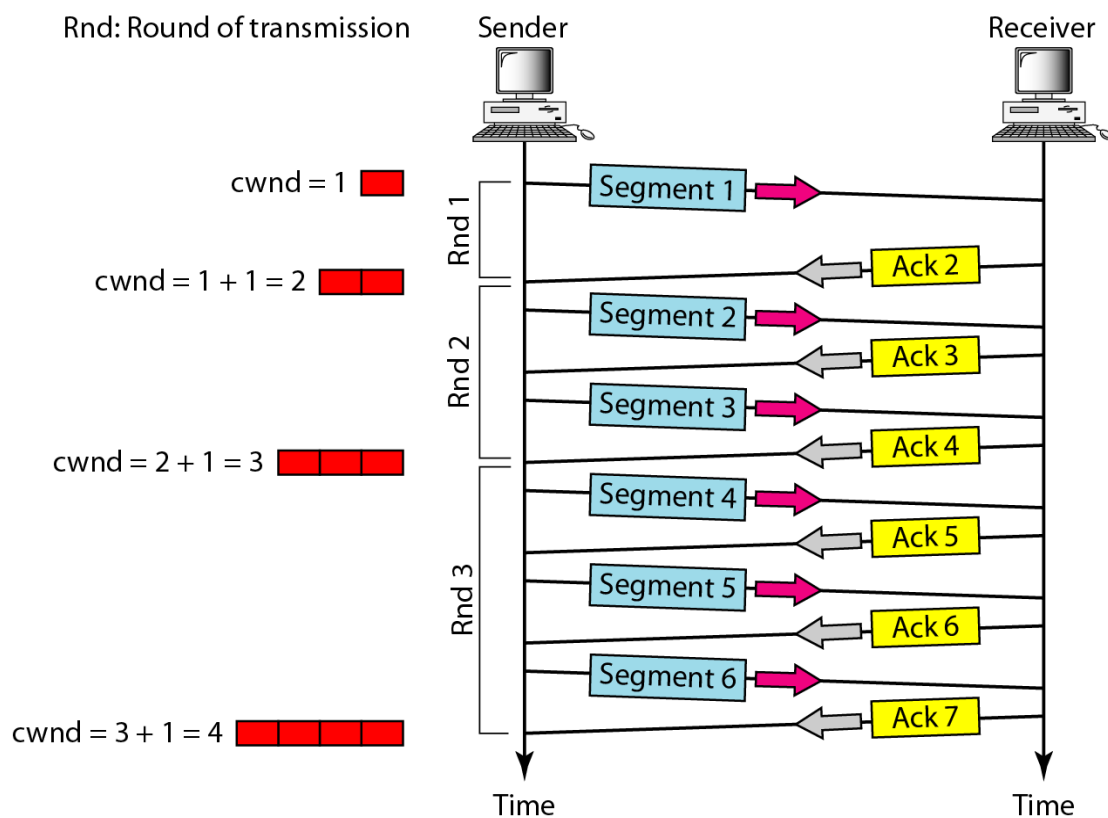


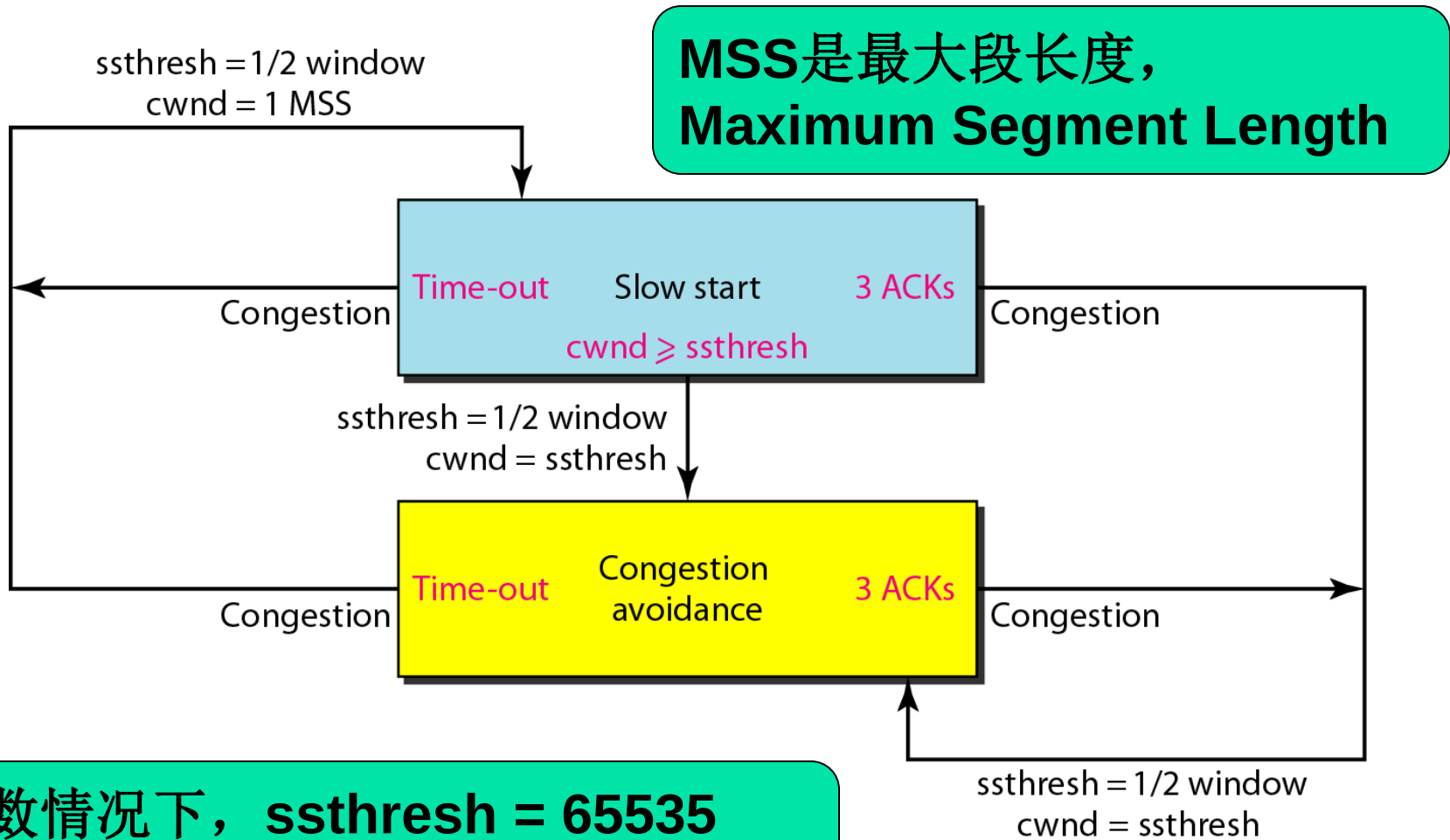
图 24.9 拥塞避免:加性增加



拥塞检测：乘性减少

- 发生拥塞后，拥塞窗口必须减少
- 发送方推测出发生拥塞的唯一方法就是通过重传段。
- 重传在两种情况下发生：
 - 重传计时器倒是
 - 接收到了三个ACK
- 此时，阈值降到一半，即就是乘性减少。

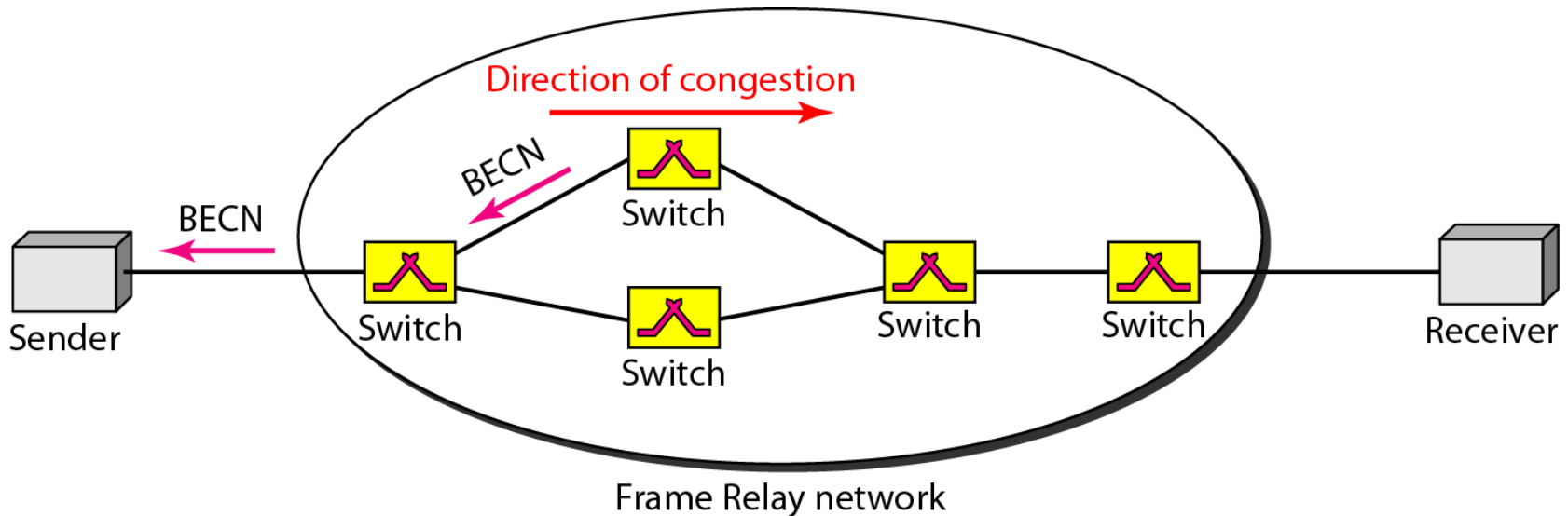
图 24.10 TCP拥塞策略总结



多数情况下, $ssthresh = 65535$

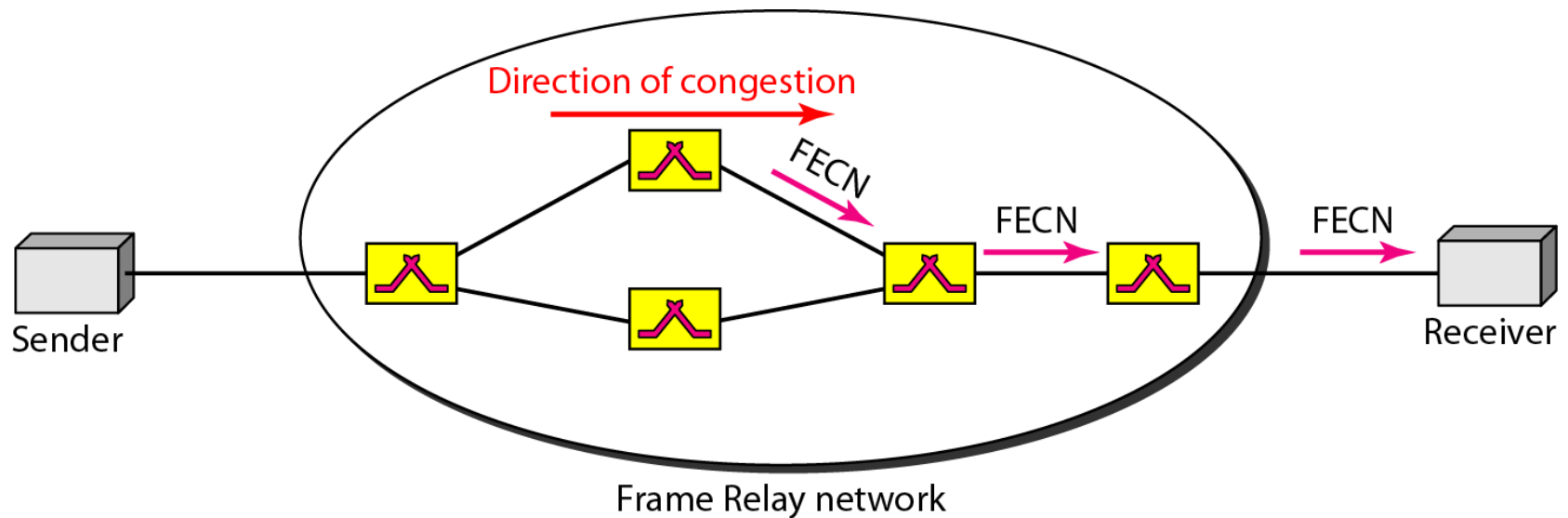
图 24.12 后向显式拥塞通

24.4.2 帧中继中的拥塞控制



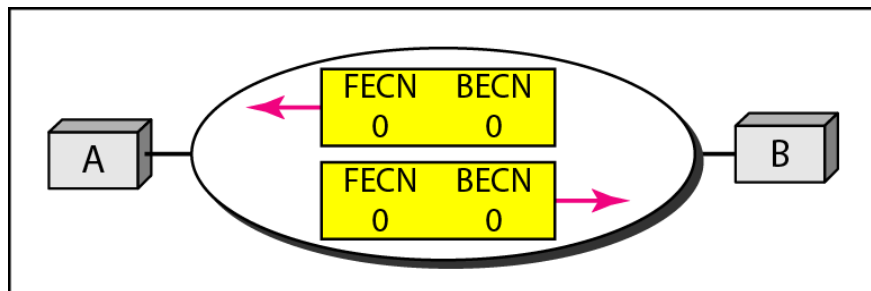
BECN,反向显式拥塞通知, backward explicit congestion notification

图 24.13 前向显式拥塞通知

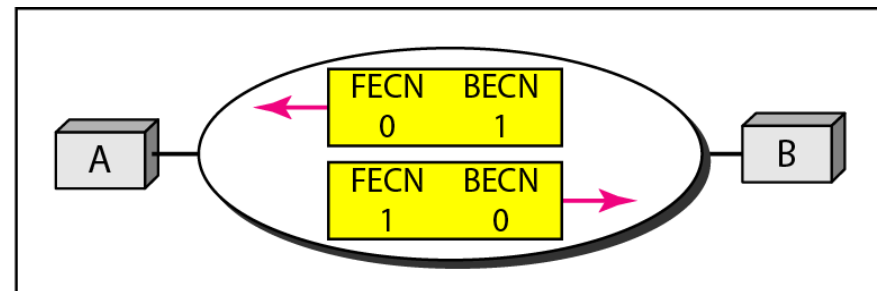


FECN,前向显式拥塞通知, forward explicit congestion notification

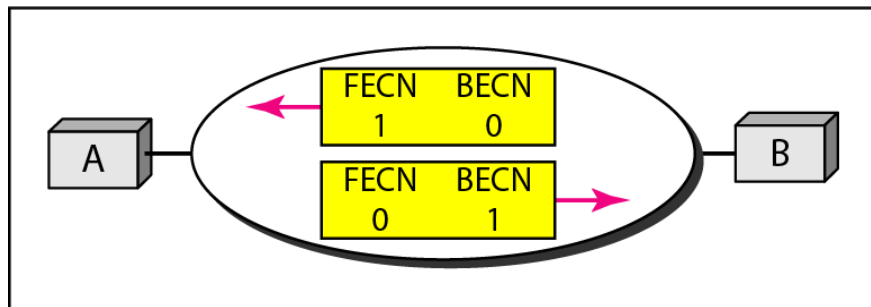
图 24.14 拥塞的四种情况



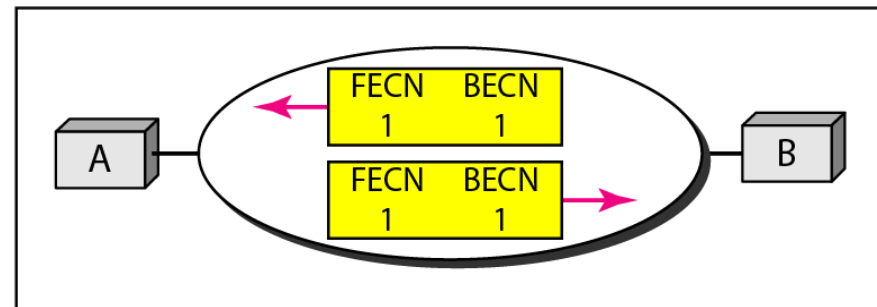
a. No congestion



b. Congestion in the direction A-B



c. Congestion in the direction B-A



d. Congestion in both directions

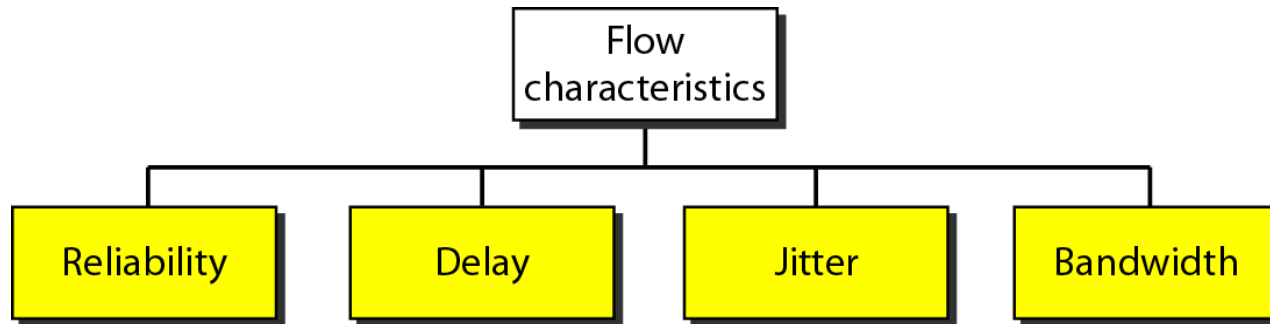
24-5 服务质量

服务质量(quality of service , QoS) 是一个网络互联问题, 对该问题的讨论已经远远超出对它的定义。我们可以非形式地将服务质量定义为数据流所追求的某种目标。

本节要点:

数据流特性
数据流类型

图 24.15 数据流特性



24-6 改进QoS 的技术

在第24.5 节中，根据数据流的特性对QoS 进行了定义。本节将讨论一些用于改进服务质量的方法。这里简要地讨论四种常用方法:调度、通信量整形、许可控制和资源预留。

本节要点:

- 调度
- 通信量整形
- 资源预留
- 许可控制

图 24.16 *FIFO* 队列

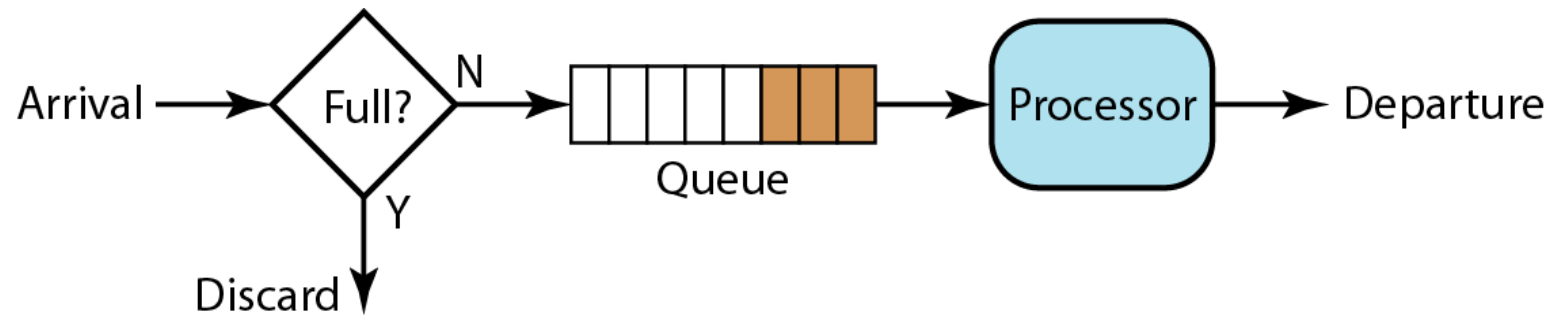


图 24.17 优先级队列

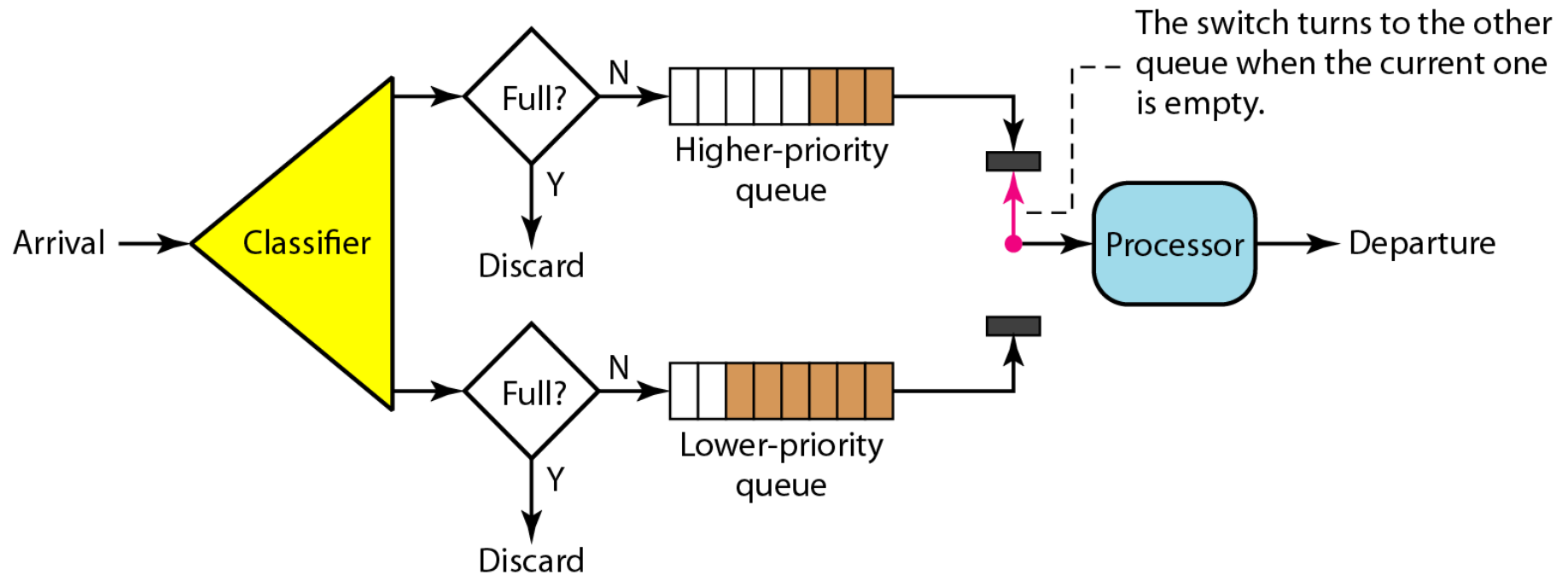


图 24.18 加权公平队列

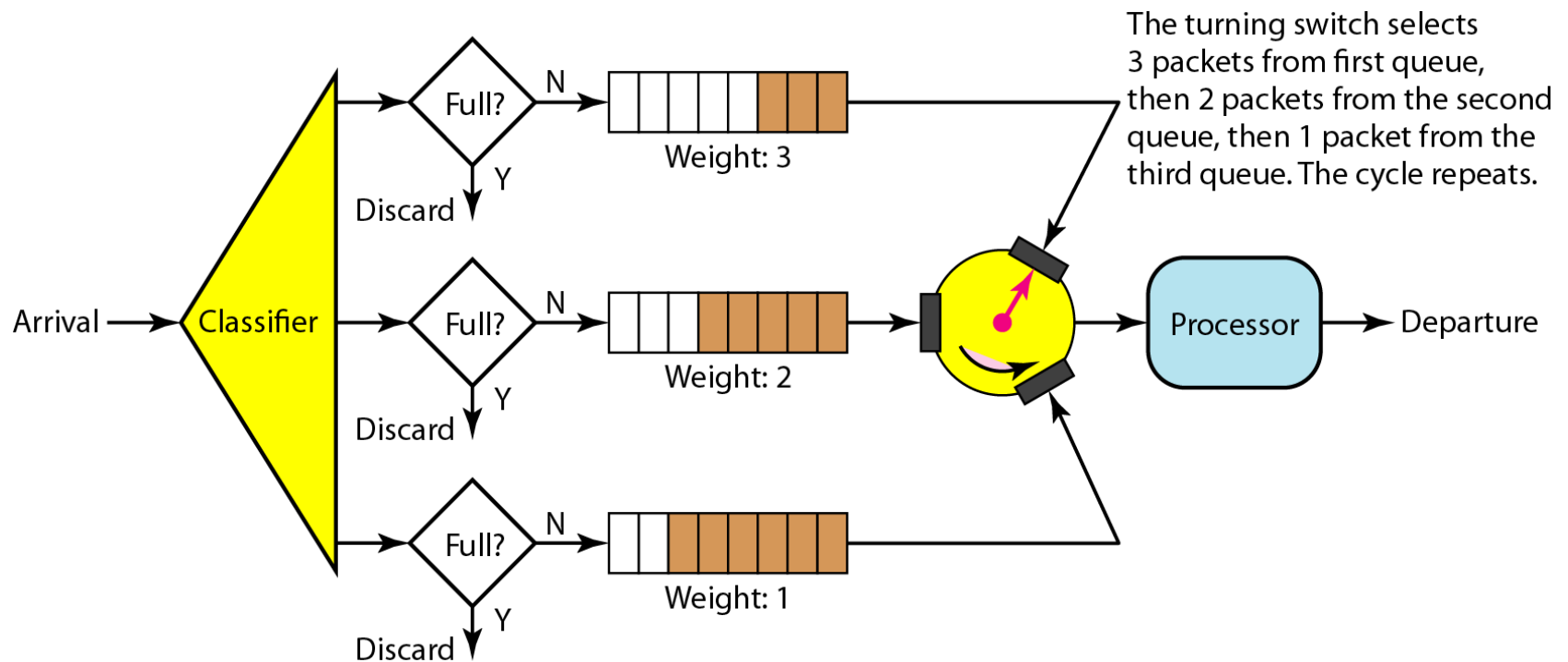


图 24.19 漏桶

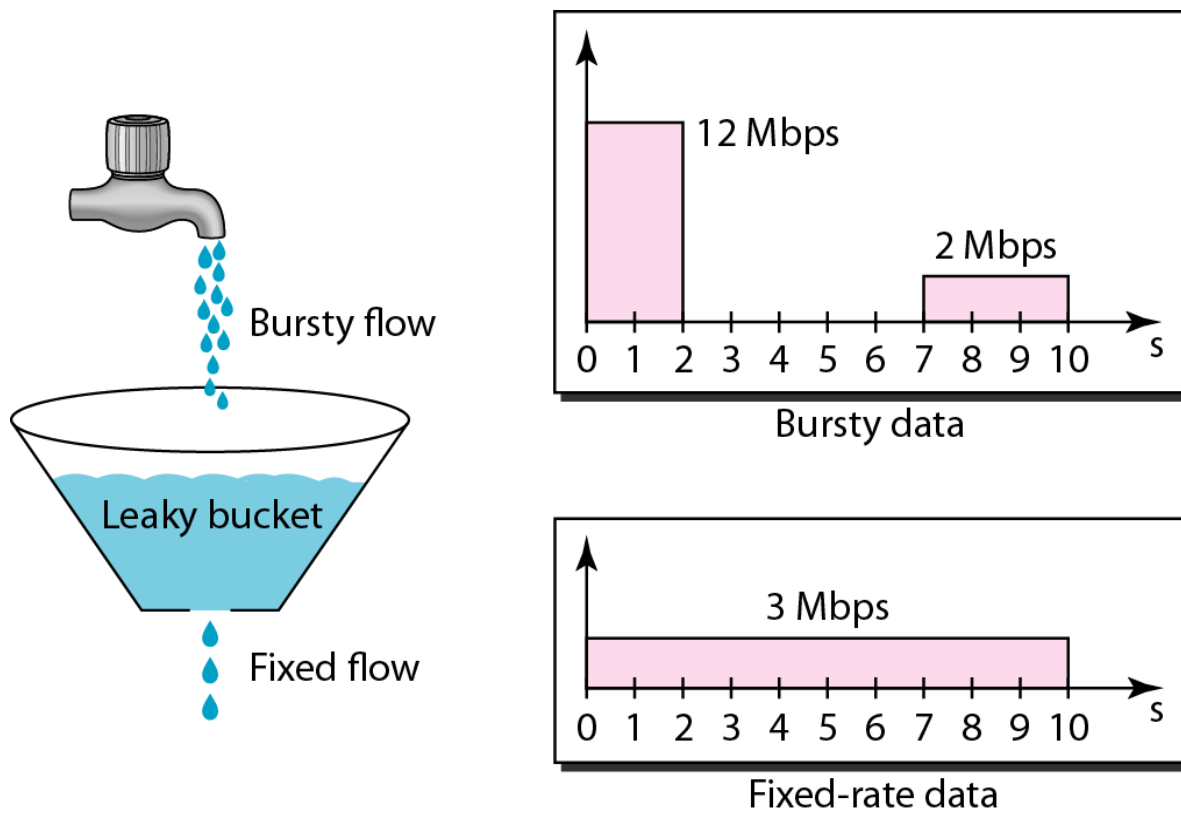
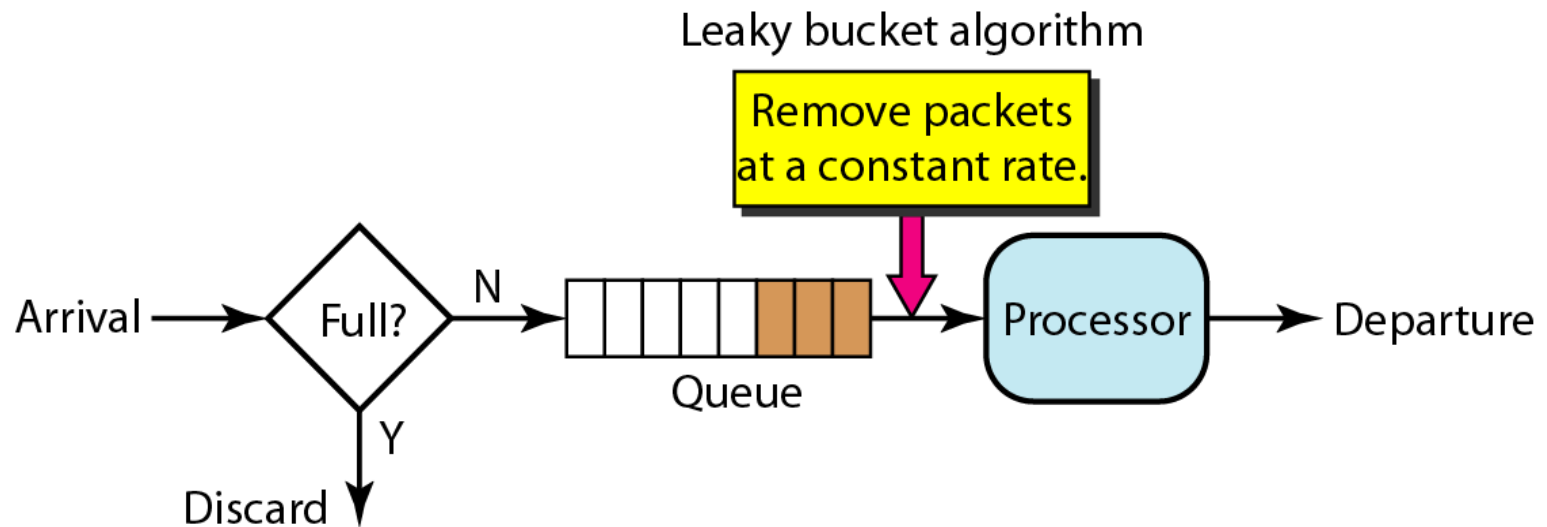


图 24.20 漏桶的实现过程





注意

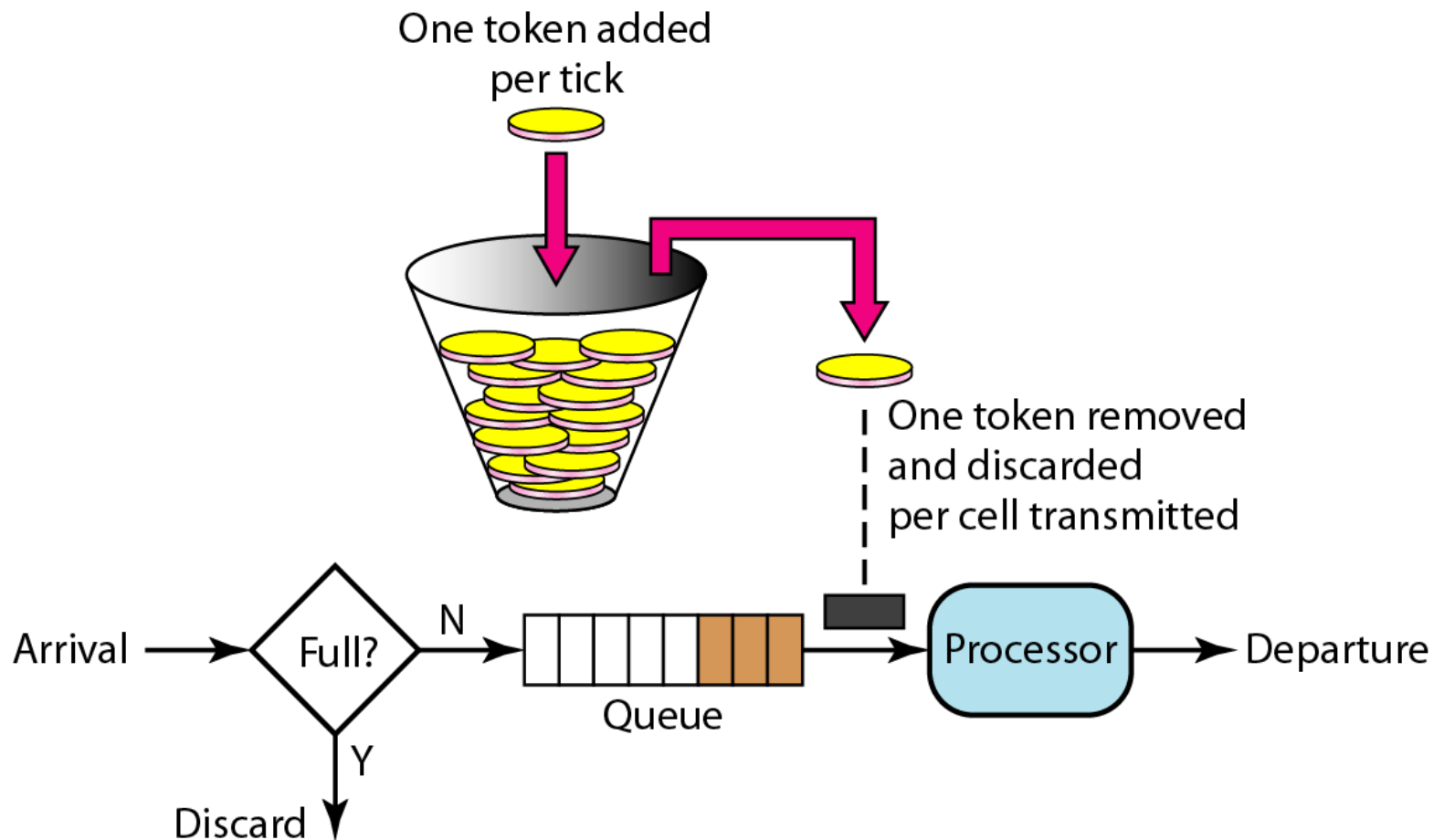
漏桶算法通过均分数据速率将突发性通信量整形为固定速率的通信量。如果桶已满，就可能丢失分组。



注意

令牌桶允许规定的最大速率发送突发性通信量。

图 24.21 令牌桶算法的原理



24-7 综合业务

在因特网中已经设计出了两种用于提供服务质量的模型:综合业务和差分业务。本章节我们讨论第一种模型。

本节要点:

信令

数据流规范

许可

业务类型

RSVP

综合业务所存在的问题



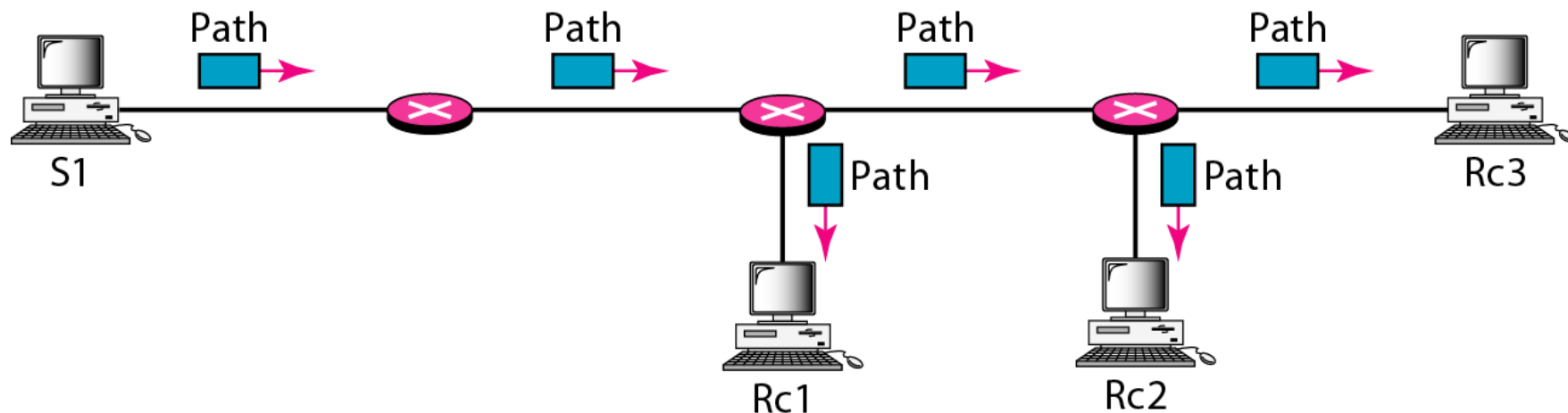
注意

综合业务是设计用于**IP** 的基于数据流的**QoS**模型。

RSVP

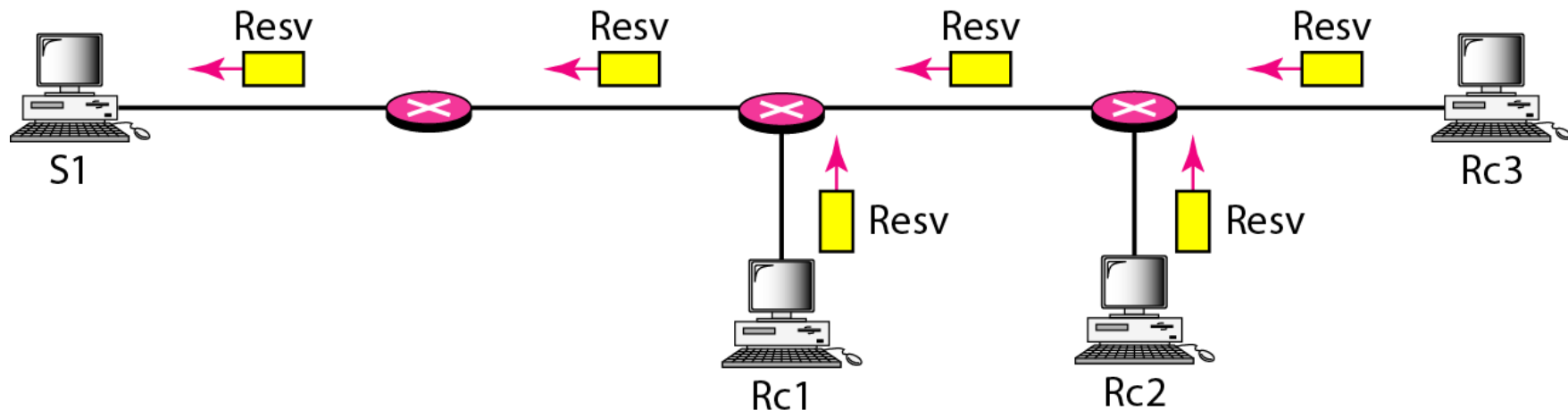
- **RSVP, Resource Reservation Protocol**
- **RSVP**是一种信令协议，它帮助**IP**创建一条数据流服务
- **RSVP**是一种多播的信令系统
- **RSVP**是基于接收方的预留
- 有多种报文，最主要的有两种：
 - 路径(**Path**)报文
 - 预留(**Resv**)报文

图 24.22 路径报文



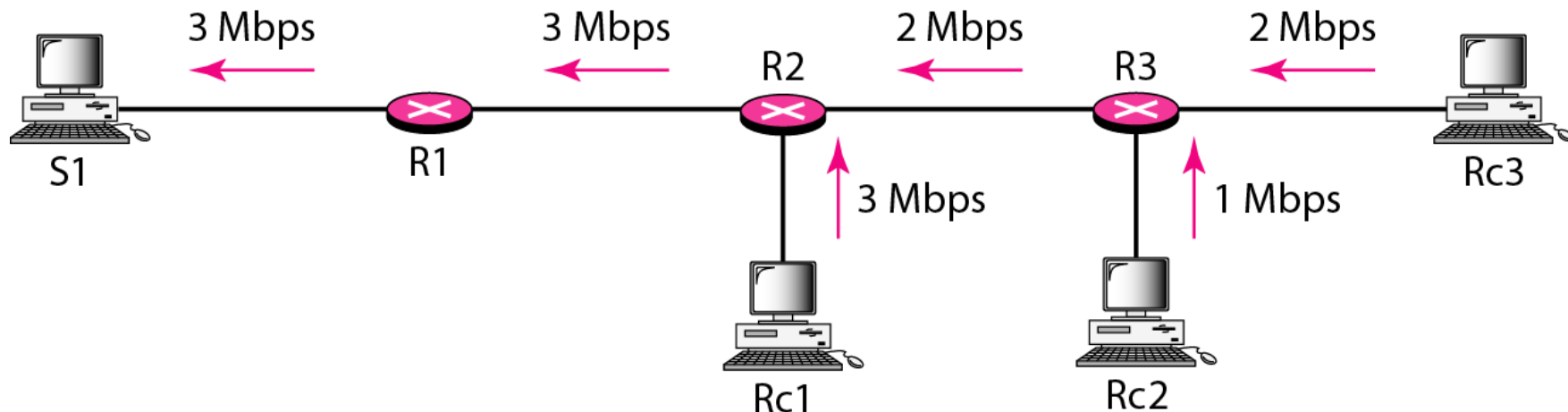
- (1) 预留资源之前，接收方并不知道分组经过的路径，但预留资源是需要知道路径的；
- (2) 路径报文从发送方送出，到达多播路径中的所有接收方

图 24.23 预留报文



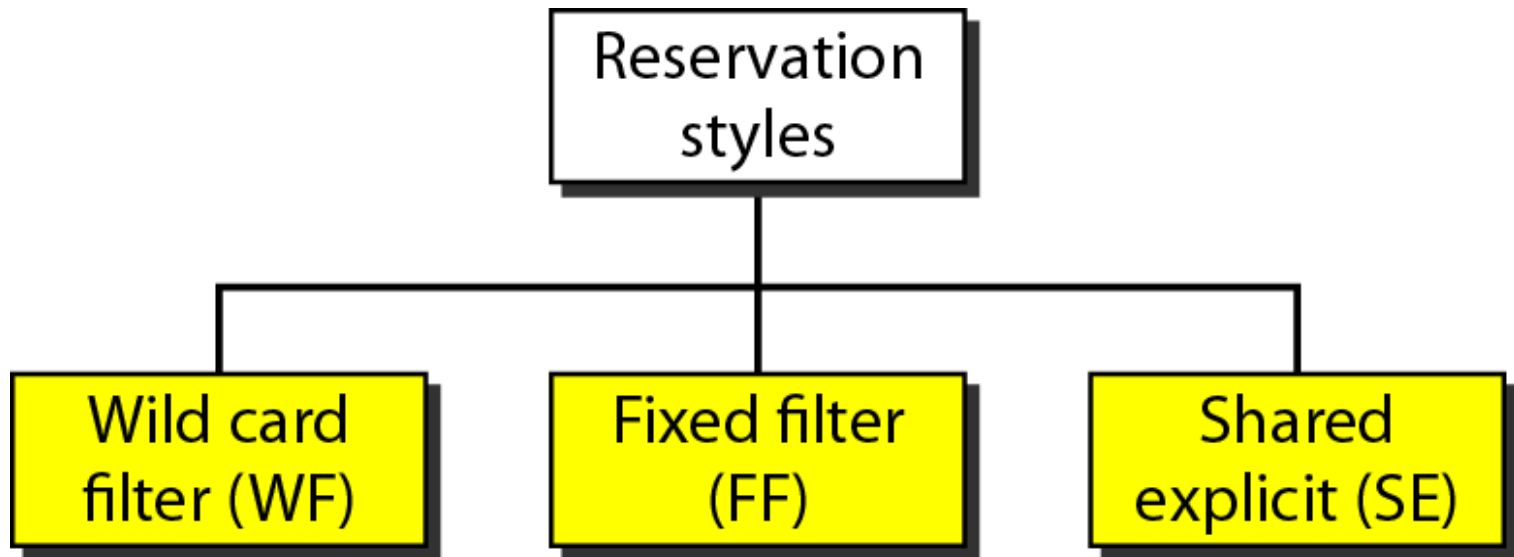
- (1) 预留资源之前，接收方并不知道分组经过的路径，但预留资源是需要知道路径的；
- (2) 路径报文从发送方送出，到达多播路径中的所有接收方

图 24.24 预留合并



- (1) 在接收方收到路径报文之后，接收方发送预留报文；
- (2) 预留报文朝着发送方传送，在支持**RSVP**的路由器上预约资源

图 24.25 预留方式



1. 通过过滤器方式，路由器为所有发送方创建一个预留，适用于来自不同方的数据流不同时出现情况
2. 固定过滤器，路由器为每个数据流创建一个不同的预留，适用于来自不同发送方的数据流同时出现的概率很高的情况
3. 共享显式方式，创建一个预留，该预留可以被一组数据流共享

24-8 差分业务（服务）

差分业务 (Differentiated Services , DS Diffserv) 是由因特网工程任务组 (Internet Engineering Task Force , IETF) 提出的用于弥补综合业务缺陷的一种业务。

本节要点:

差分业务字段



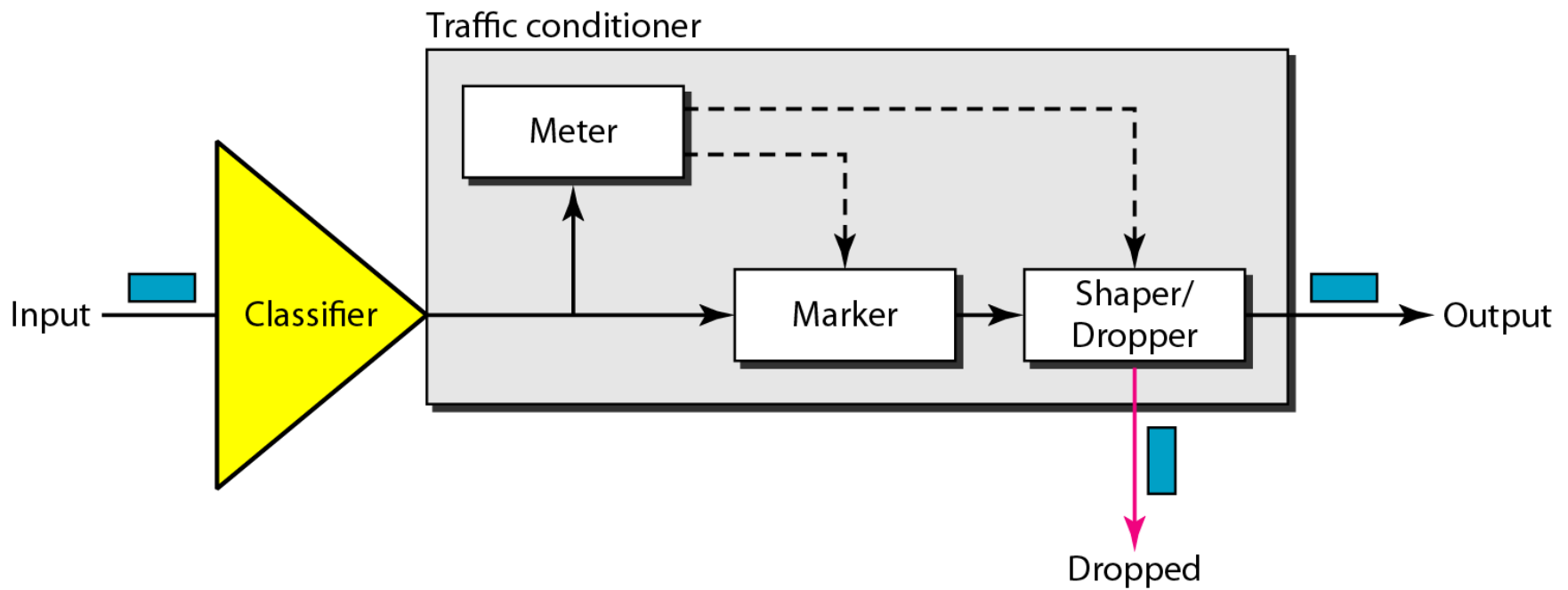
注意

- 1.差分业务是用于**IP** 的基于类的**QoS** 模型。
- 2.主要处理工程从网络核心转移到网络边缘，解决了可伸缩性的问题，路由器不需要存储有关数据流的信息
- 3.将针对每条数据流的业务改变为针对每种类型的业务。

图 24.26 差分业务字段



图 24.27 通信量调节器



24-9 交换网络中的QoS

讨论了IP协议中提出的QoS 模型之后，现在来讨论一下用于两个交换网络中的QoS：帧中继和ATM。这两个网络都是虚电路网络，它们需要一个诸如RSVP 的信令协议。

本节要点:

帧中继中的QoS
ATM 中的QoS

本节不要求

补充作业：

■ 2009 第38题。

- 38.主机甲和主机乙间已建立一个TCP连接，主机甲向主机乙发送了两个连续的TCP段，分别包含300字节和500字节的有效载荷，第一个段的序列号为200，主机乙正确接收到两个段后，发送给主机甲的确认序列号是

A. 500 B.700 C.800 D.1000

■ 2010 第39题。

- 39、主机甲和主机乙之间已建立一个TCP连接，TCP最大段长度为1000字节，若主机甲的当前拥塞窗口为4000字节，在主机甲向主机乙连接发送2个最大段后，成功收到主机乙发送的第一段的确认段，确认段中通告的接收窗口大小为2000字节，则此时主机甲还可以向主机乙发送的最大字节数是（ ）

■ A: 1000 B: 2000 C: 3000 D: 4000

■ 2009 第39题。

- 39. 一个TCP连接总是以1KB的最大段发送TCP段，发送方有足够多的数据要发送。当拥塞窗口为16KB时发生了超时，如果接下来的4个RTT（往返时间）时间内的TCP段的传输都是成功的，那么当第4个RTT时间内发送的所有TCP段都得到肯定应答时，拥塞窗口大小是

A. 7KB B. 8KB C. 9KB D. 16KB

■ 2010 第39题。

- 39、主机甲和主机乙之间已建立一个TCP连接，TCP最大段长度为1000字节，若主机甲的当前拥塞窗口为4000字节，在主机甲向主机乙连接发送2个最大段后，成功收到主机乙发送的第一段的确认段，确认段中通告的接收窗口大小为2000字节，则此时主机甲还可以向主机乙发送的最大字节数是（ ）

■ A: 1000 B: 2000 C: 3000 D: 4000

39. 主机甲向主机乙发送一个(SYN=1, seq=11220)的TCP段, 期望与主机乙建立TCP连接, 若主机乙接受该连接请求, 则主机乙向主机甲发送的正确的TCP段可能是
- A. (SYN=0, ACK=0, seq=11221, ack=11221)
 - B. (SYN=1, ACK=1, seq=11220, ack=11220)
 - C. (SYN=1, ACK=1, seq=11221, ack=11221)
 - D. (SYN=0, ACK=0, seq=11220, ack=11220)
40. 主机甲与主机乙之间已建立一个TCP连接, 主机甲向主机乙发送了3个连续的TCP段, 分别包含300字节、400字节和500字节的有效载荷, 第3个段的序号为900。若主机乙仅正确接收到第1和第3个段, 则主机乙发送给主机甲的确认序号是
- A. 300
 - B. 500
 - C. 1200
 - D. 1400