



# Data Communications and Networking

Fourth Edition

Forouzan

## Chapter 23

# Process-to-Process Delivery: UDP, TCP, and SCTP

# 23-1 PROCESS-TO-PROCESS DELIVERY

*The transport layer is responsible for process-to-process delivery—the delivery of a packet, part of a message, from one process to another. Two processes communicate in a client/server relationship, as we will see later.*

## **Topics discussed in this section:**

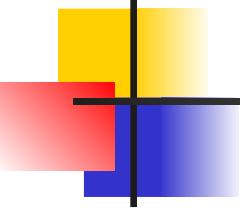
Client/Server Paradigm

Multiplexing and Demultiplexing

Connectionless Versus Connection-Oriented Service

Reliable Versus Unreliable

Three Protocols



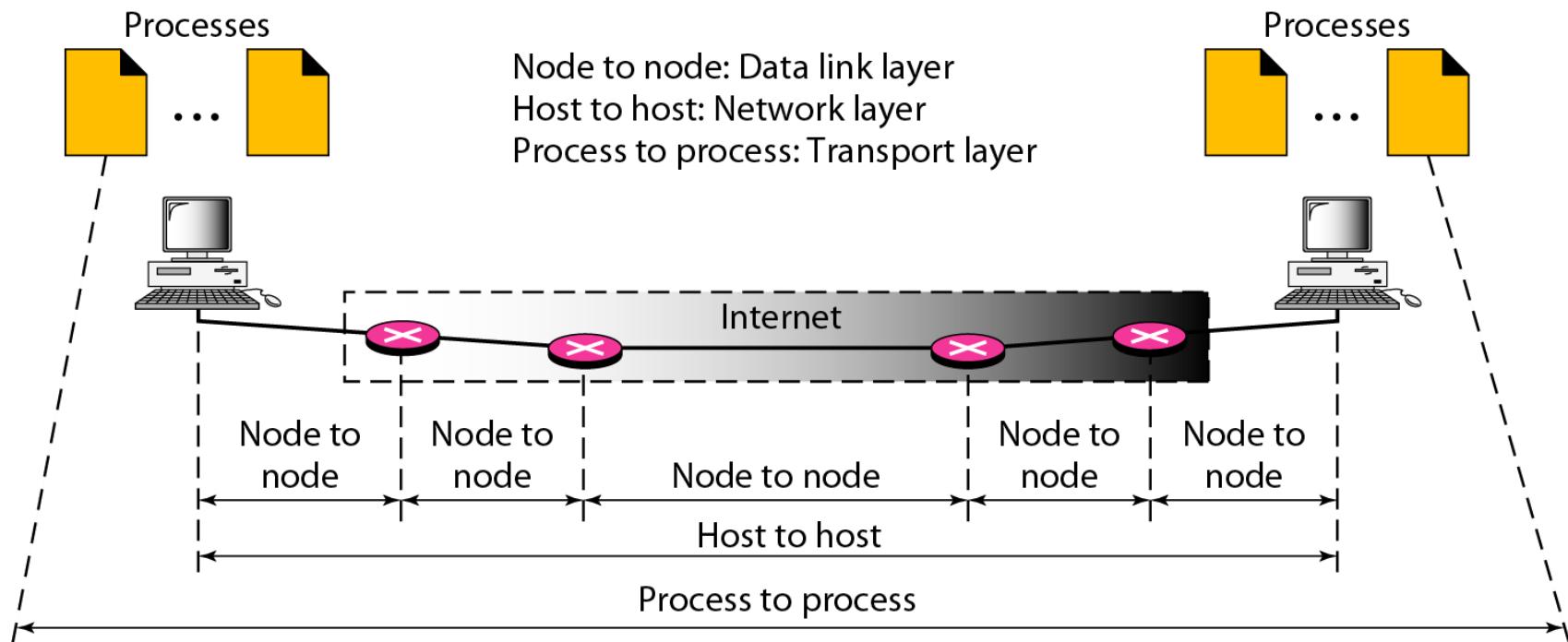
**Note**

---

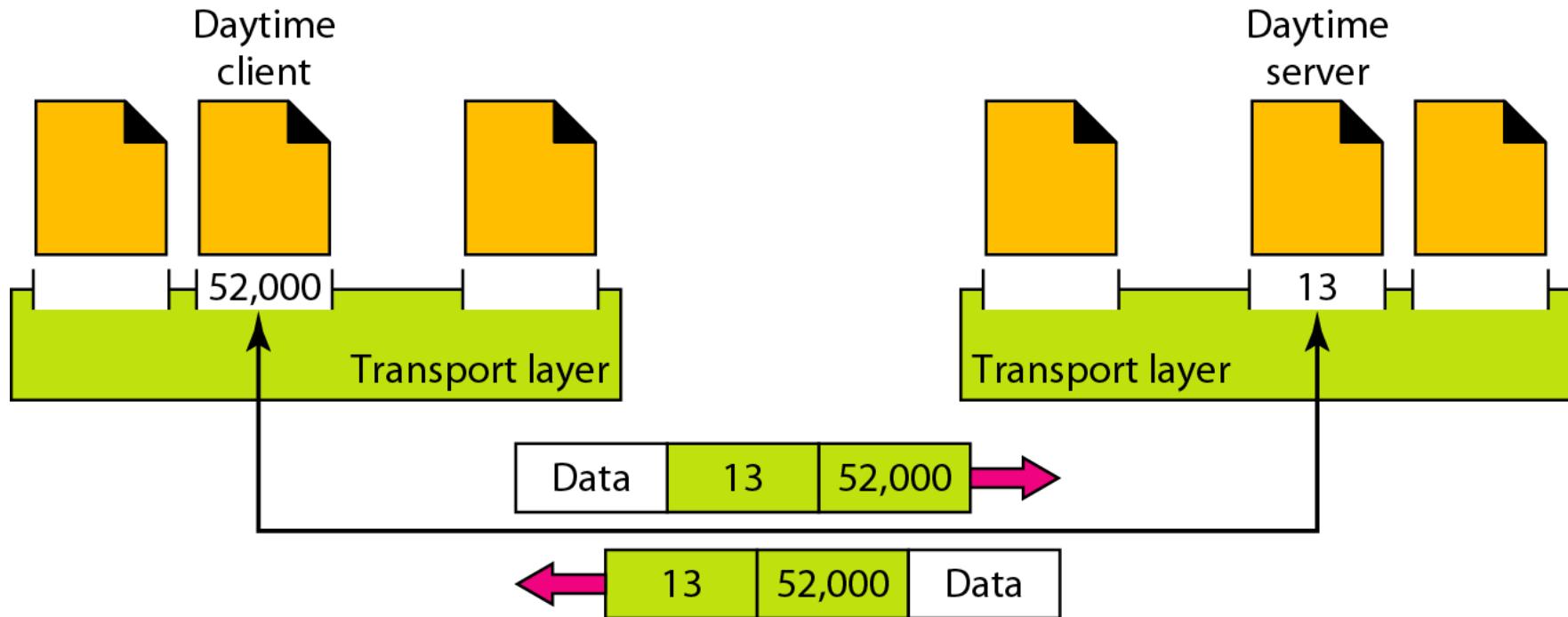
**The transport layer is responsible for process-to-process delivery.**

---

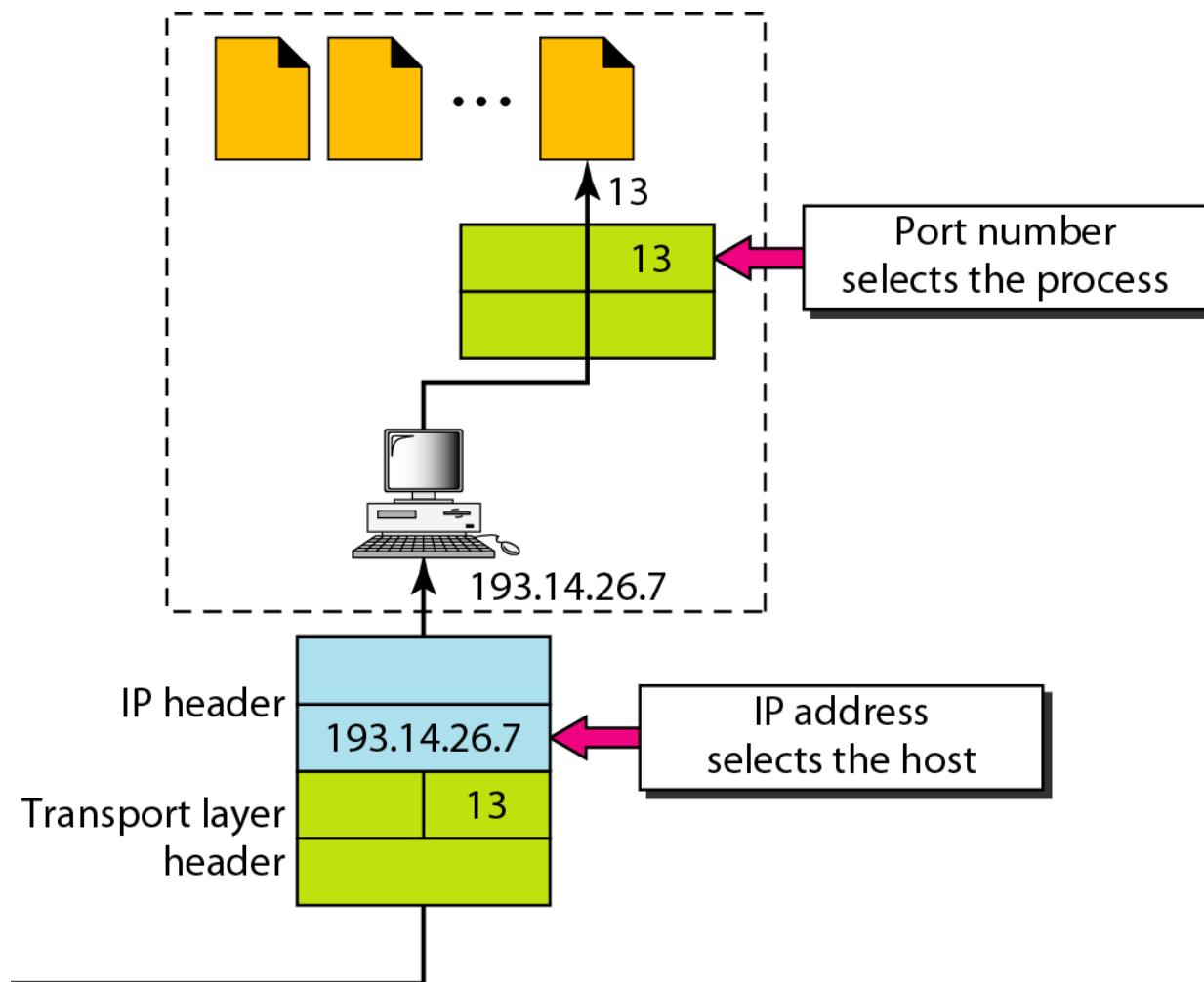
## Figure 23.1 Types of data deliveries



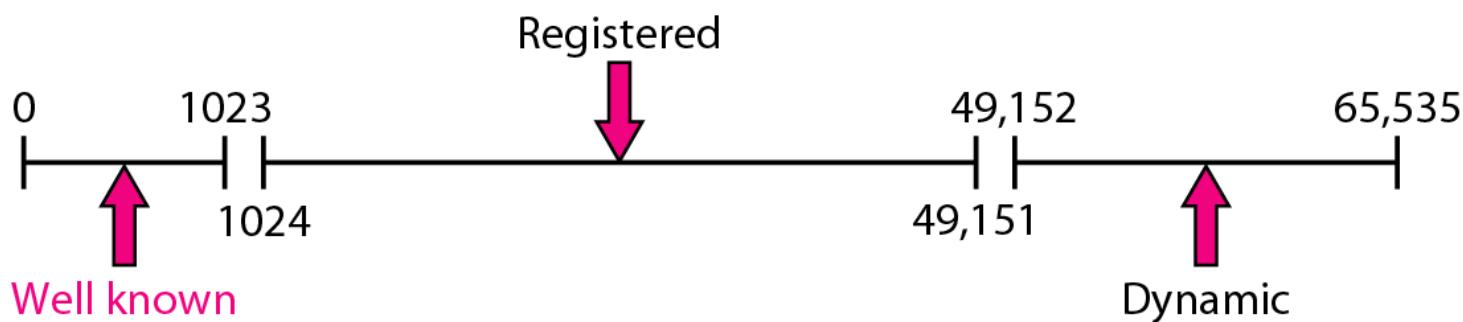
## Figure 23.2 Port numbers



## Figure 23.3 IP addresses versus port numbers

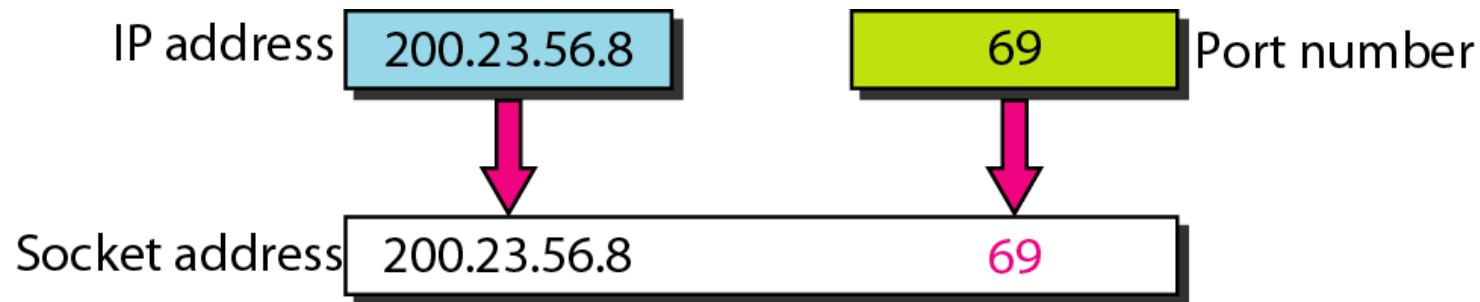


## Figure 23.4 IANA ranges



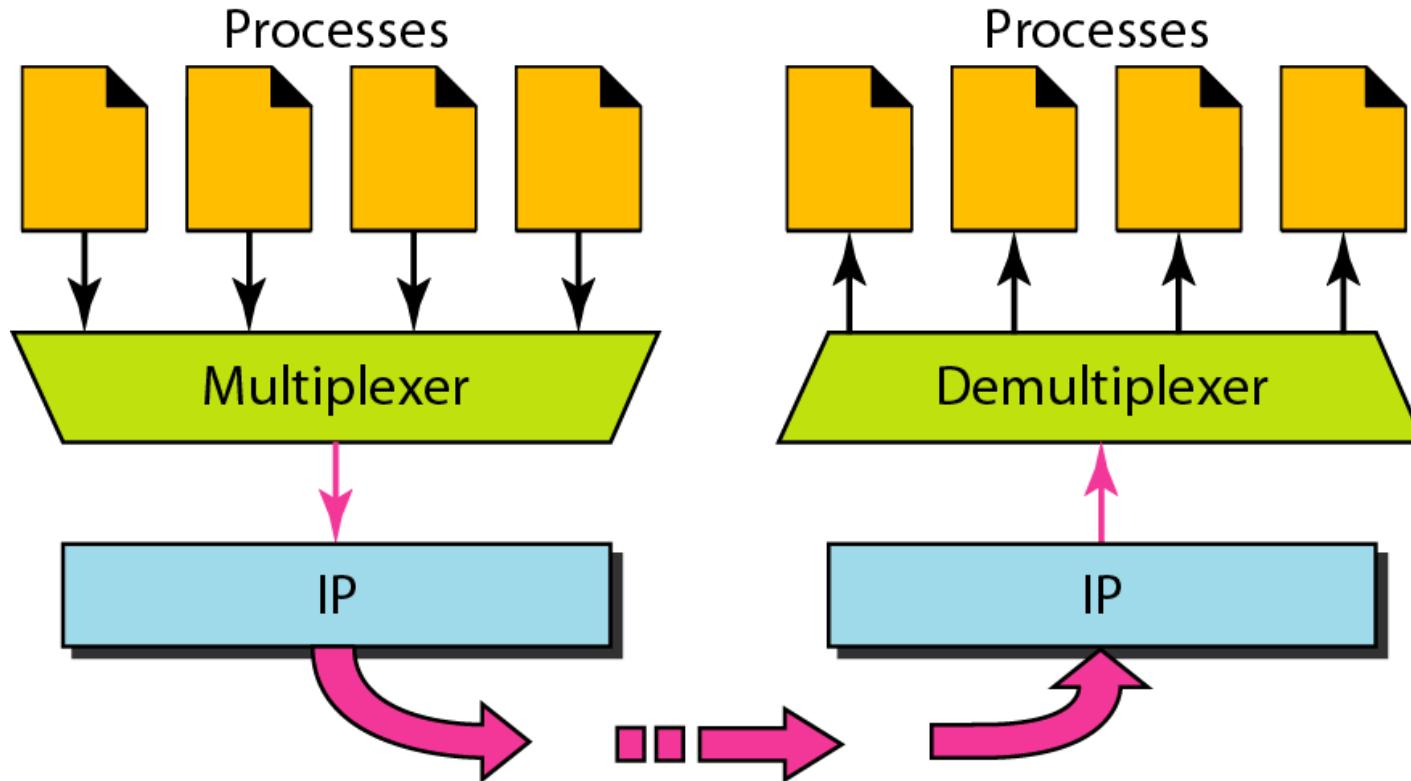
## Figure 23.5 *Socket address*

---

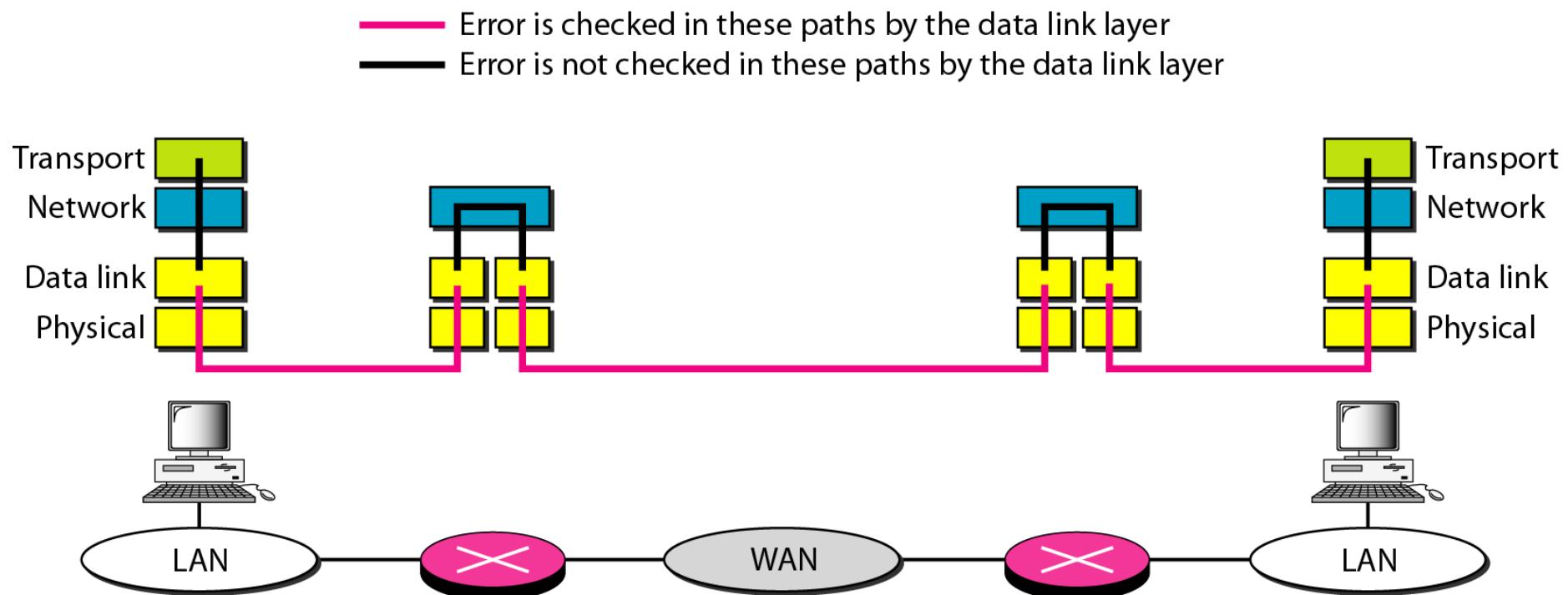


**Figure 23.6 Multiplexing and demultiplexing**

---



## Figure 23.7 Error control



## **23-2 USER DATAGRAM PROTOCOL (UDP)**

*The User Datagram Protocol (UDP) is called a connectionless, unreliable transport protocol. It does not add anything to the services of IP except to provide process-to-process communication instead of host-to-host communication.*

### **Topics discussed in this section:**

**Well-Known Ports for UDP**

**User Datagram**

**Checksum**

**UDP Operation**

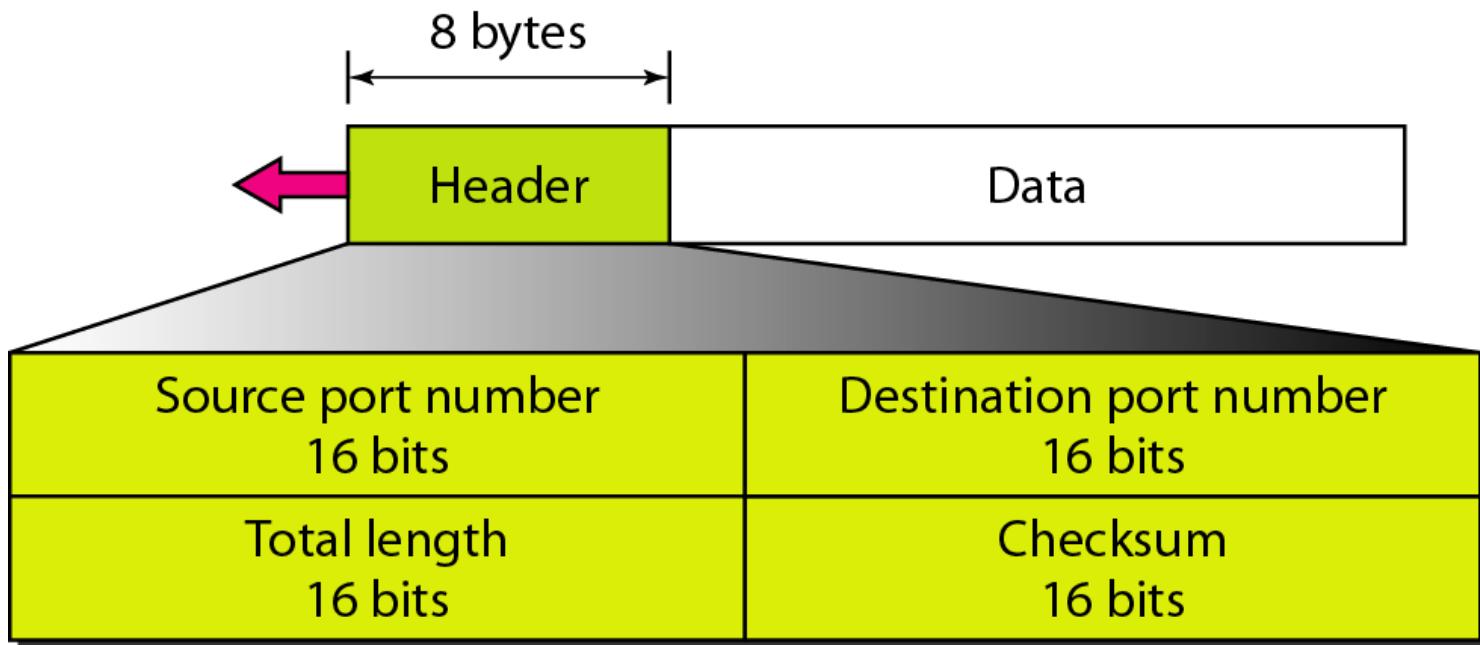
**Use of UDP**

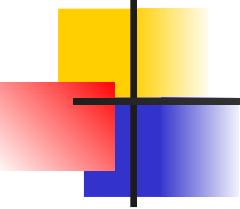
**Table 23.1 Well-known ports used with UDP**

<i>Port</i>	<i>Protocol</i>	<i>Description</i>
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
53	Nameserver	Domain Name Service
67	BOOTPs	Server port to download bootstrap information
68	BOOTPc	Client port to download bootstrap information
69	TFTP	Trivial File Transfer Protocol
111	RPC	Remote Procedure Call
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management Protocol
162	SNMP	Simple Network Management Protocol (trap)

**Figure 23.9** *User datagram format*

---





*Note*

---

**UDP length  
= IP length – IP header's length**

---

## 23-3 TCP

***TCP is a connection-oriented protocol; it creates a virtual connection between two TCPS to send data. In addition, TCP uses flow and error control mechanisms at the transport level.***

### **Topics discussed in this section:**

TCP Services

TCP Features

Segment

A TCP Connection

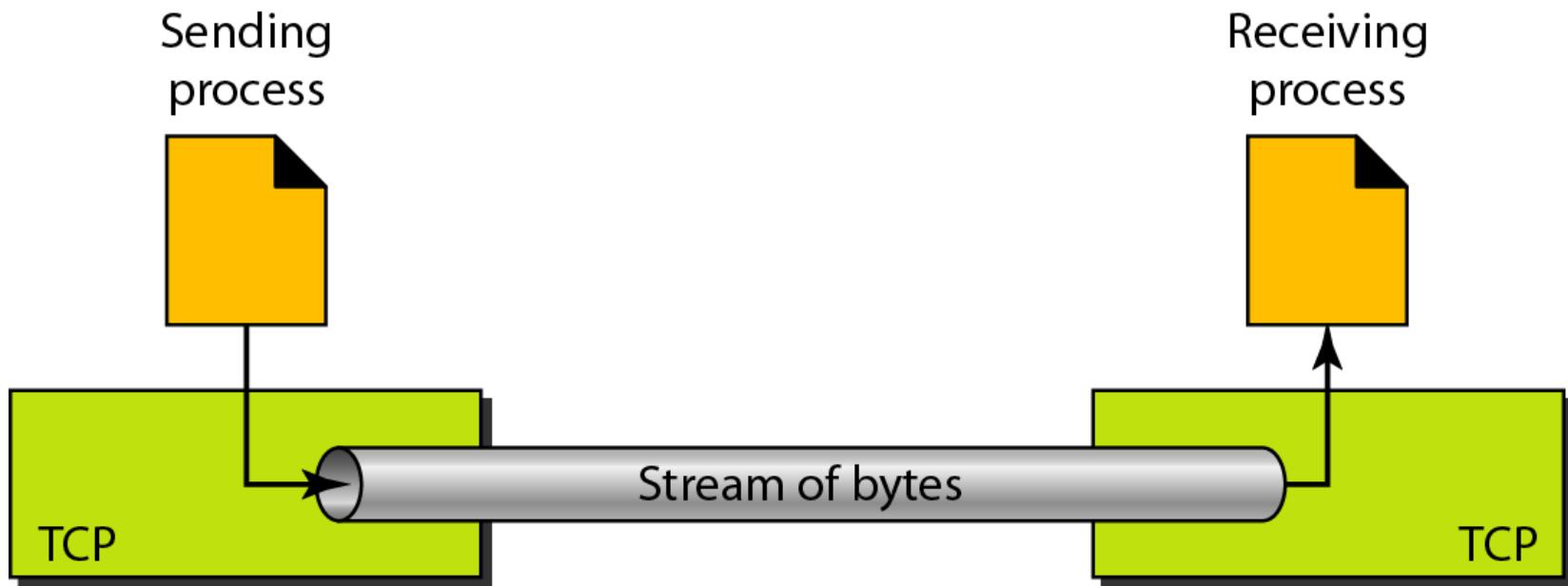
Flow Control

Error Control

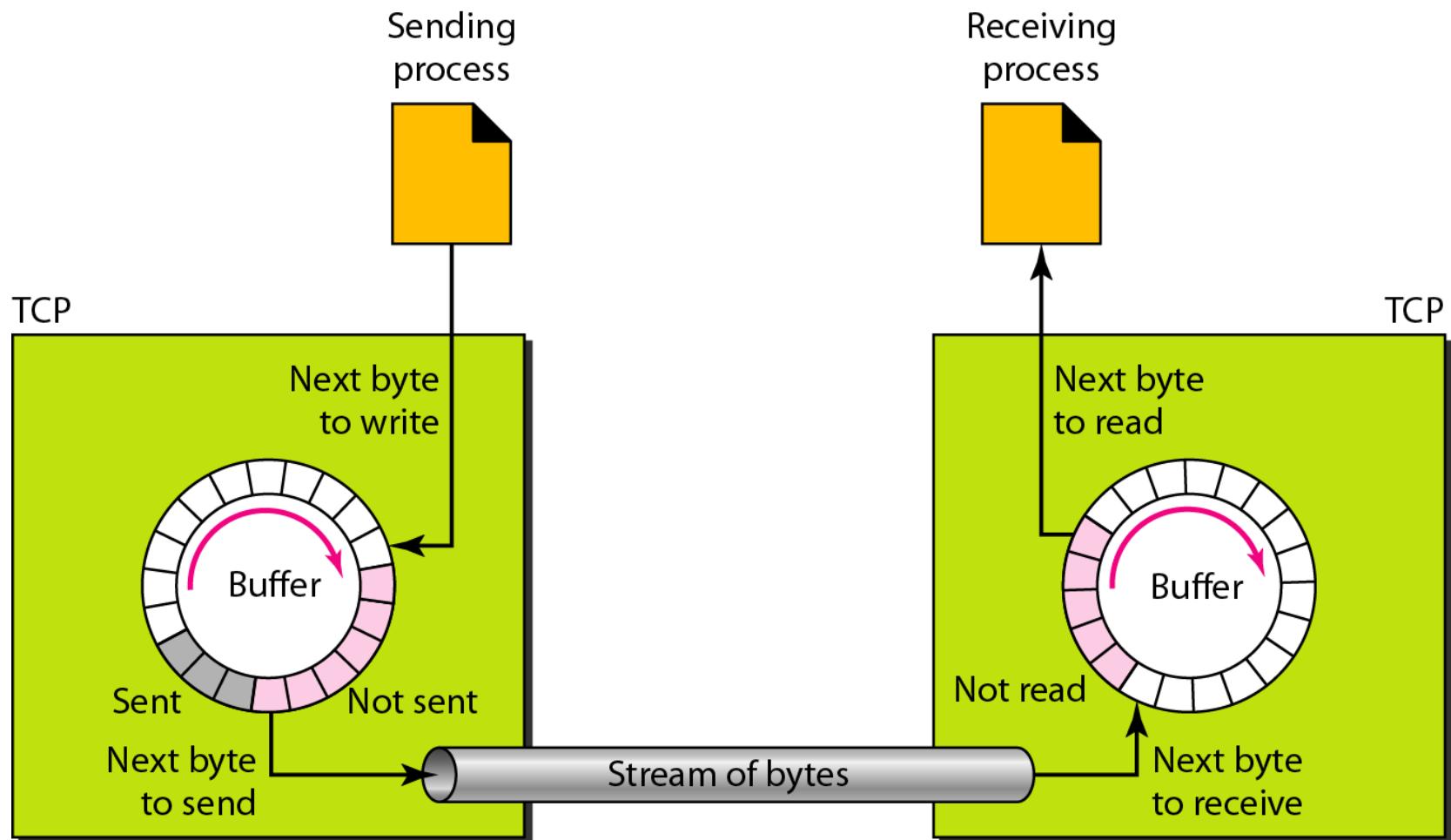
**Table 23.2 Well-known ports used by TCP**

<i>Port</i>	<i>Protocol</i>	<i>Description</i>
7	Echo	Echoes a received datagram back to the sender
9	Discard	Discards any datagram that is received
11	Users	Active users
13	Daytime	Returns the date and the time
17	Quote	Returns a quote of the day
19	Chargen	Returns a string of characters
20	FTP, Data	File Transfer Protocol (data connection)
21	FTP, Control	File Transfer Protocol (control connection)
23	TELNET	Terminal Network
25	SMTP	Simple Mail Transfer Protocol
53	DNS	Domain Name Server
67	BOOTP	Bootstrap Protocol
79	Finger	Finger
80	HTTP	Hypertext Transfer Protocol
111	RPC	Remote Procedure Call

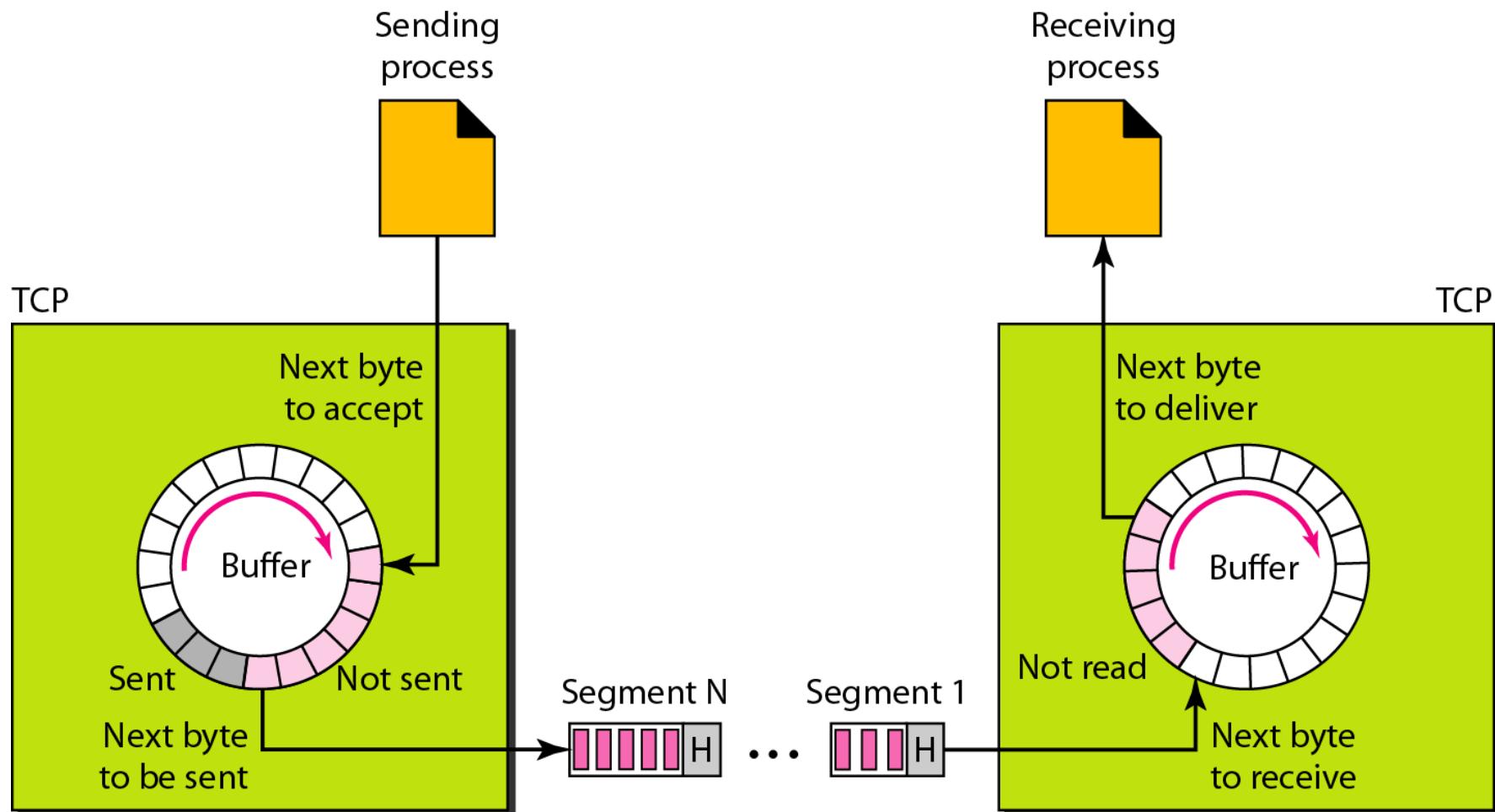
**Figure 23.13 Stream delivery**

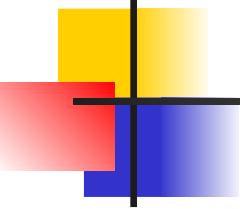


**Figure 23.14** *Sending and receiving buffers*



## Figure 23.15 TCP segments

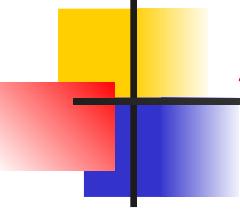




## **Note**

---

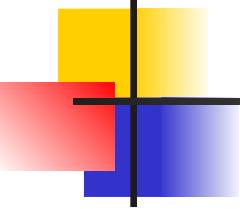
**The bytes of data being transferred in each connection are numbered by TCP. The numbering starts with a randomly generated number.**



## **Example 23.3**

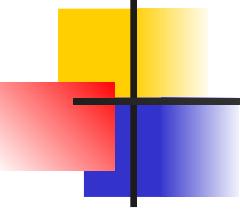
***The following shows the sequence number for each segment:***

Segment 1	➡	Sequence Number: 10,001 (range: 10,001 to 11,000)
Segment 2	➡	Sequence Number: 11,001 (range: 11,001 to 12,000)
Segment 3	➡	Sequence Number: 12,001 (range: 12,001 to 13,000)
Segment 4	➡	Sequence Number: 13,001 (range: 13,001 to 14,000)
Segment 5	➡	Sequence Number: 14,001 (range: 14,001 to 15,000)



## **Note**

**The value in the sequence number field of a segment defines the number of the first data byte contained in that segment.**

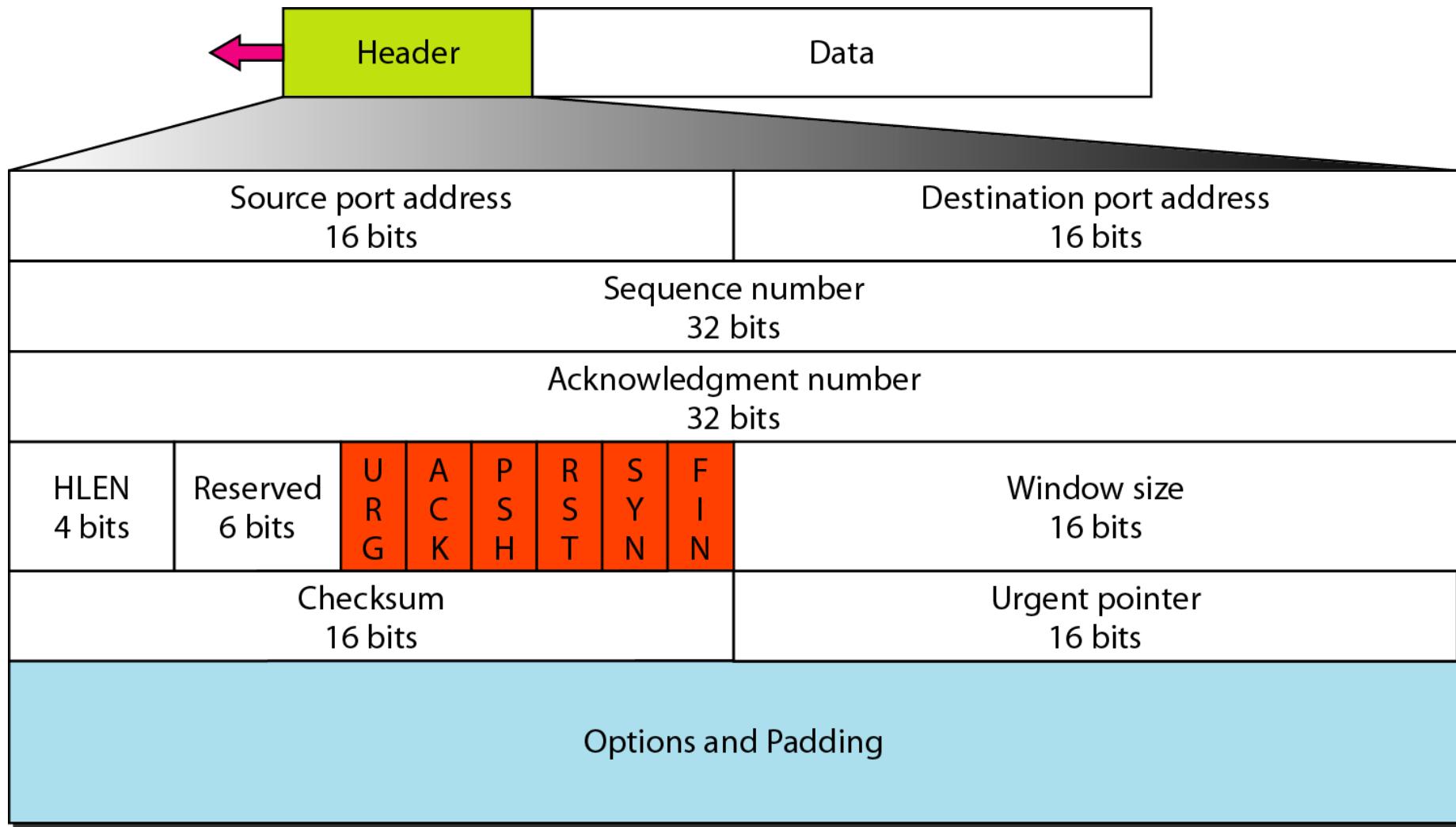


## **Note**

**The value of the acknowledgment field in a segment defines the number of the next byte a party expects to receive.**

**The acknowledgment number is cumulative.**

**Figure 23.16 TCP segment format**



## Figure 23.17 Control field

URG: Urgent pointer is valid

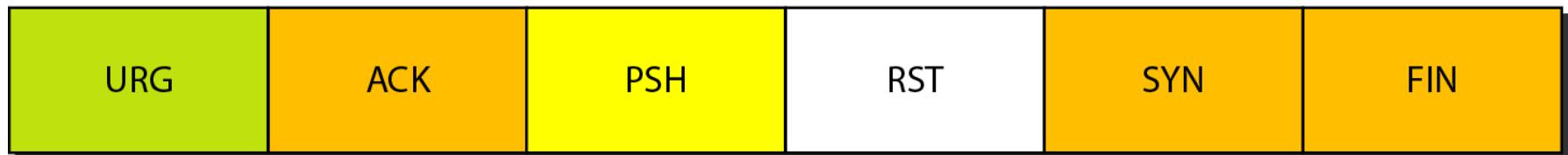
ACK: Acknowledgment is valid

PSH: Request for push

RST: Reset the connection

SYN: Synchronize sequence numbers

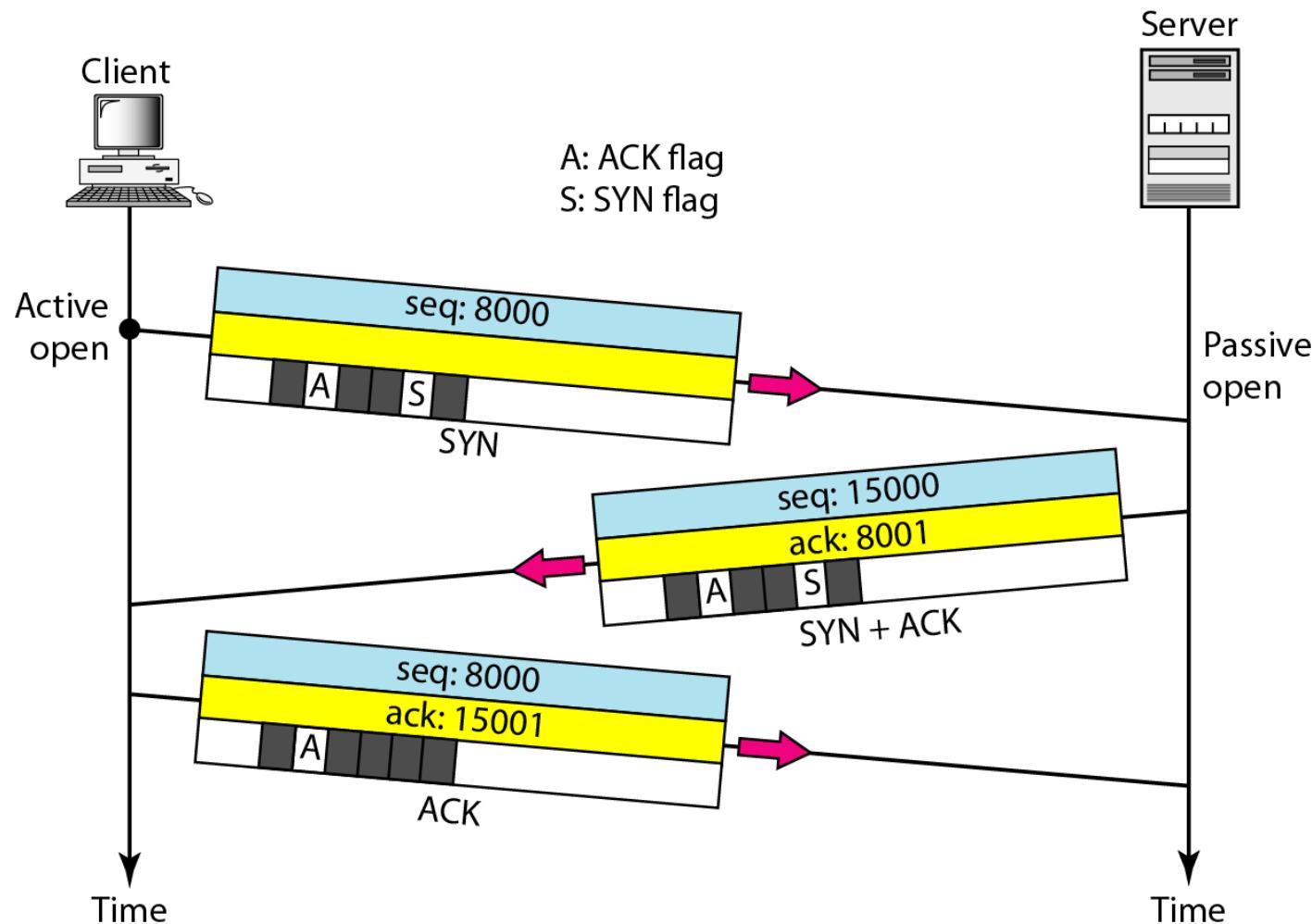
FIN: Terminate the connection

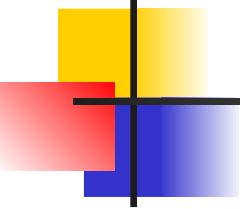


**Table 23.3** *Description of flags in the control field*

<i>Flag</i>	<i>Description</i>
URG	The value of the urgent pointer field is valid.
ACK	The value of the acknowledgment field is valid.
PSH	Push the data.
RST	Reset the connection.
SYN	Synchronize sequence numbers during connection.
FIN	Terminate the connection.

**Figure 23.18 Connection establishment using three-way handshaking**



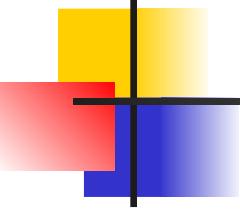


## *Note*

---

**A SYN segment cannot carry data, but it consumes one sequence number.**

---

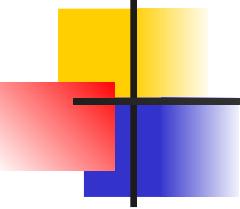


## *Note*

---

**A SYN + ACK segment cannot carry data, but does consume one sequence number.**

---



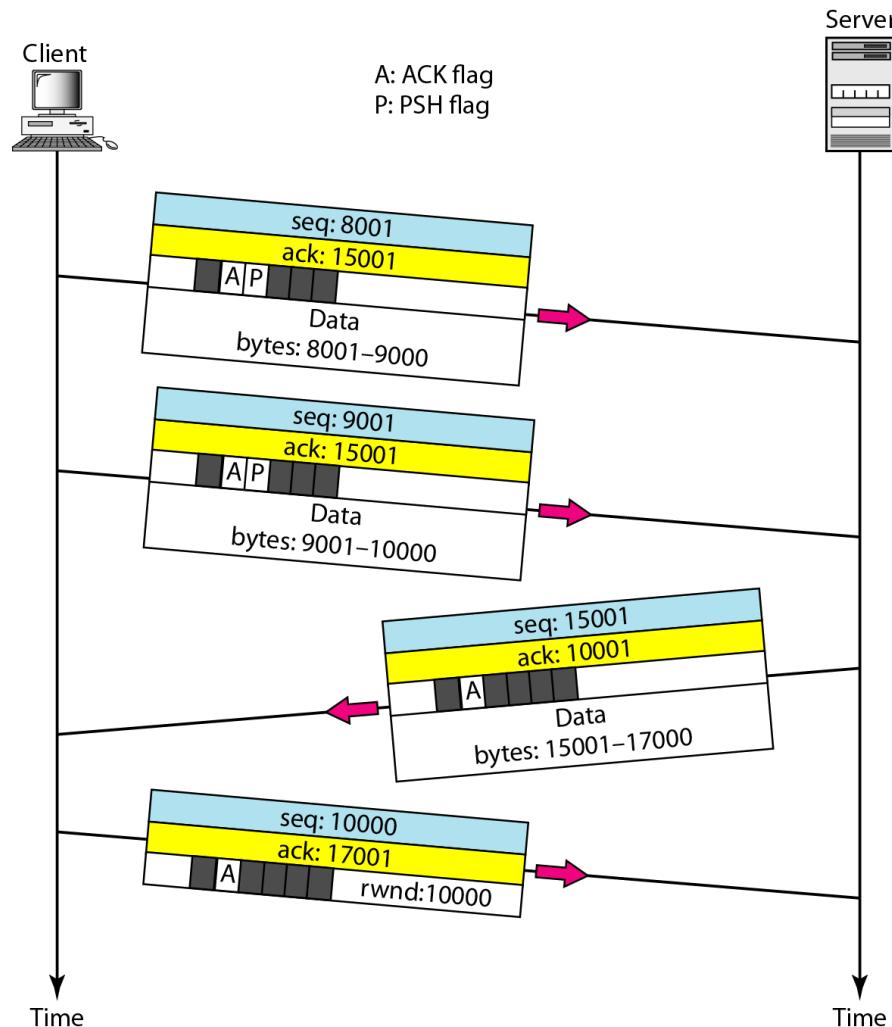
## *Note*

---

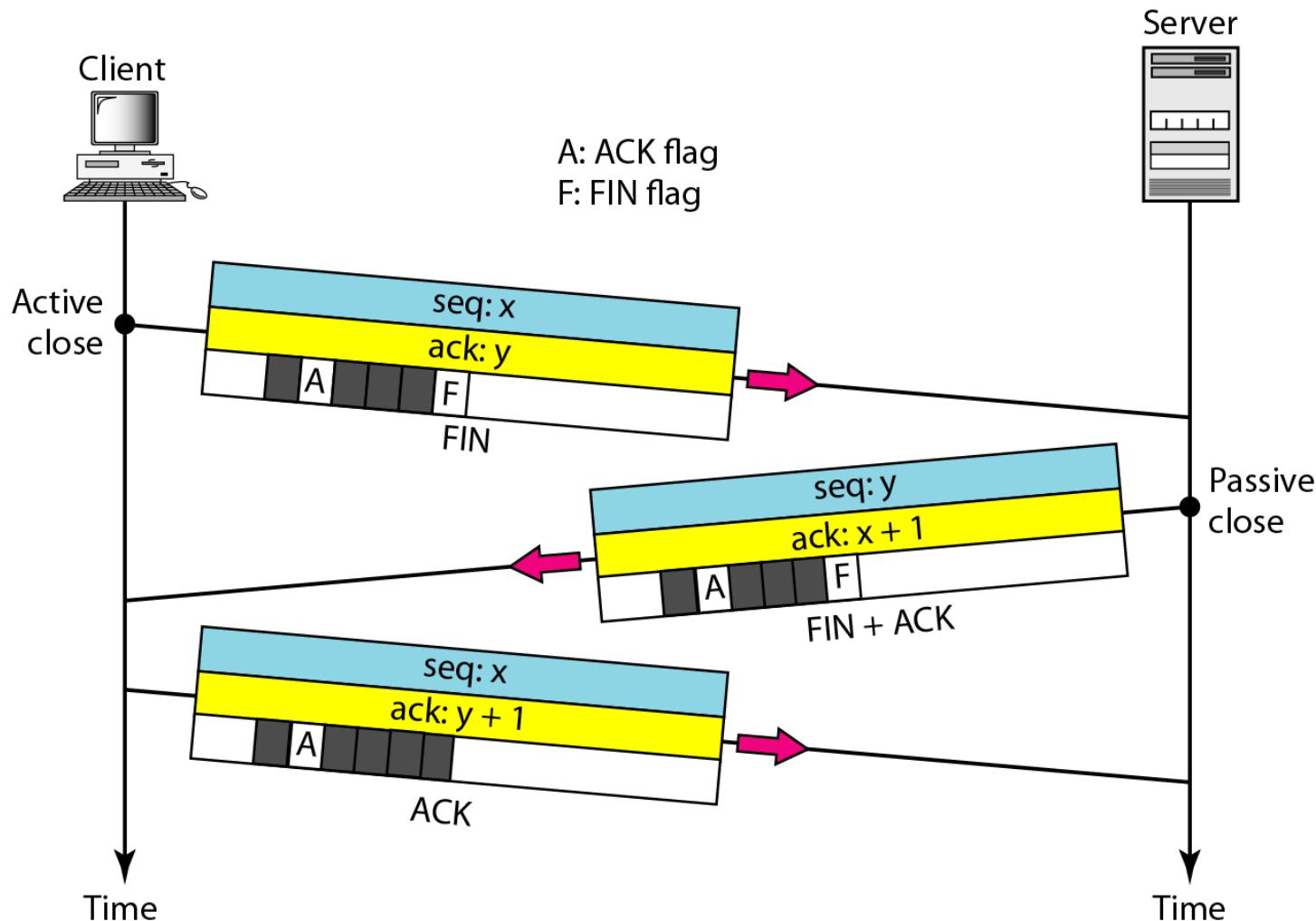
**An ACK segment, if carrying no data,  
consumes no sequence number.**

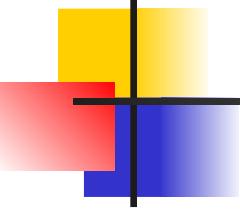
---

## Figure 23.19 Data transfer



**Figure 23.20 Connection termination using three-way handshaking**



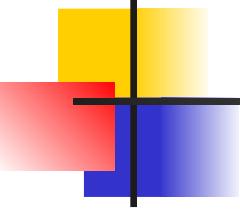


## *Note*

---

**The FIN segment consumes one sequence number if it does not carry data.**

---



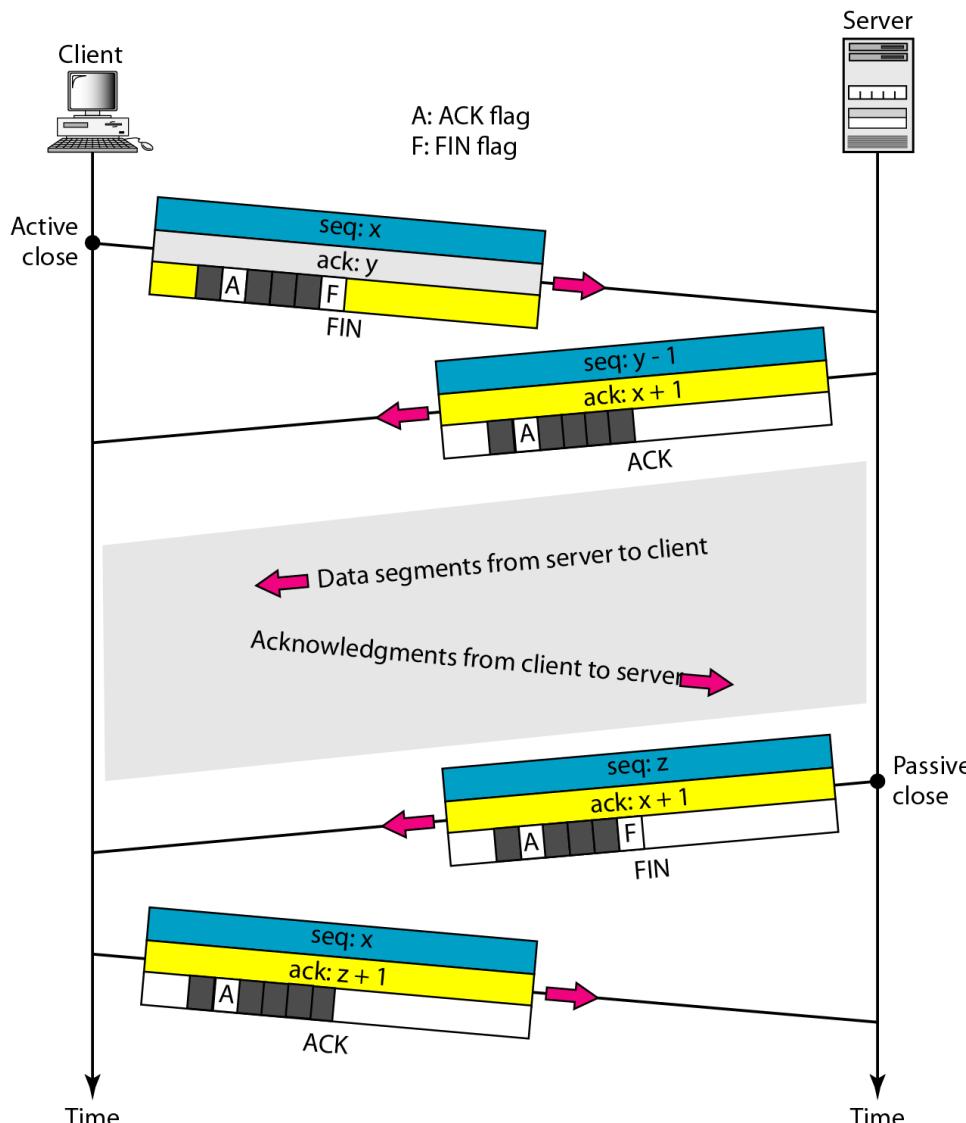
**Note**

---

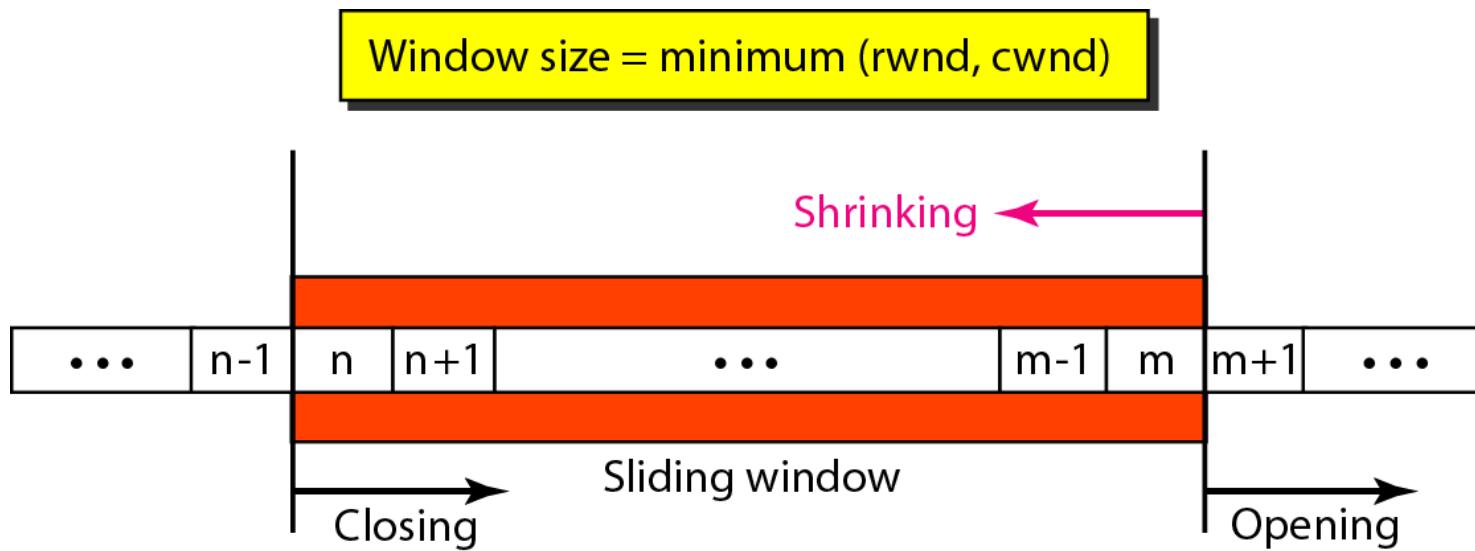
**The FIN + ACK segment consumes  
one sequence number if it  
does not carry data.**

---

## Figure 23.21 Half-close

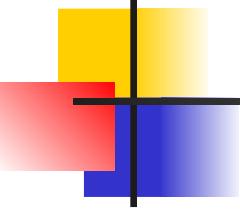


## Figure 23.22 Sliding window



## **Some points about TCP sliding windows:**

- The size of the window is the lesser of rwnd and cwnd.**
- The source does not have to send a full window's worth of data.**
- The window can be opened or closed by the receiver, but should not be shrunk.**
- The destination can send an acknowledgment at any time as long as it does not result in a shrinking window.**
- The receiver can temporarily shut down the window; the sender, however, can always send a segment of 1 byte after the window is shut down.**

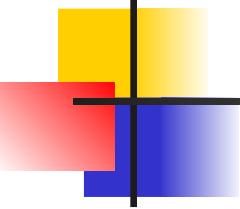


## *Note*

---

**ACK segments do not consume sequence numbers and are not acknowledged.**

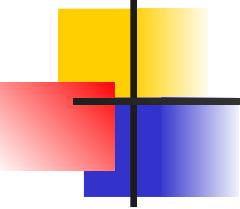
---



## **Note**

---

**In modern implementations, a retransmission occurs if the retransmission timer expires or three duplicate ACK segments have arrived.**

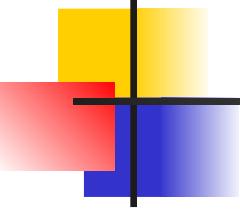


*Note*

---

**No retransmission timer is set for an  
ACK segment.**

---

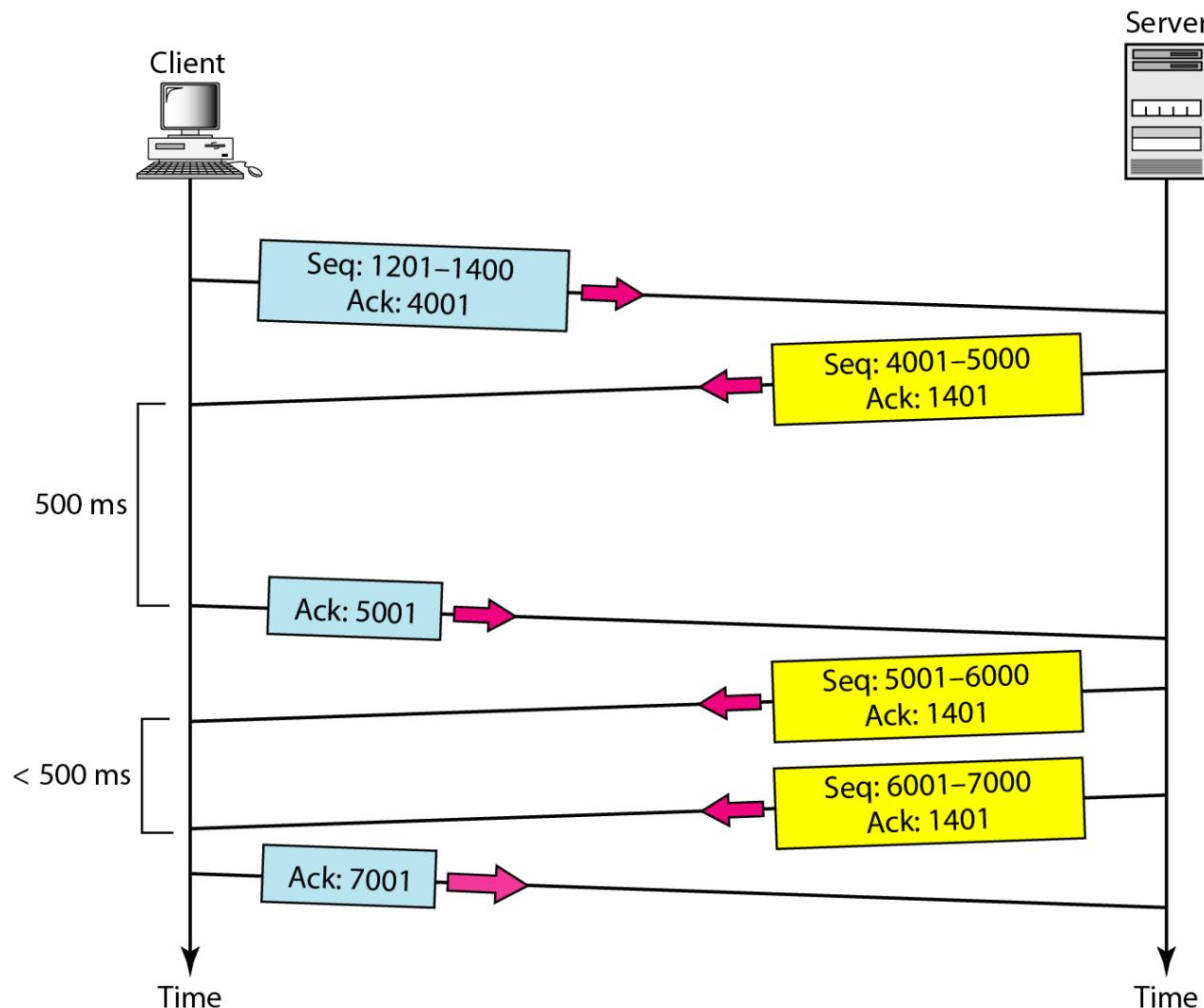


## **Note**

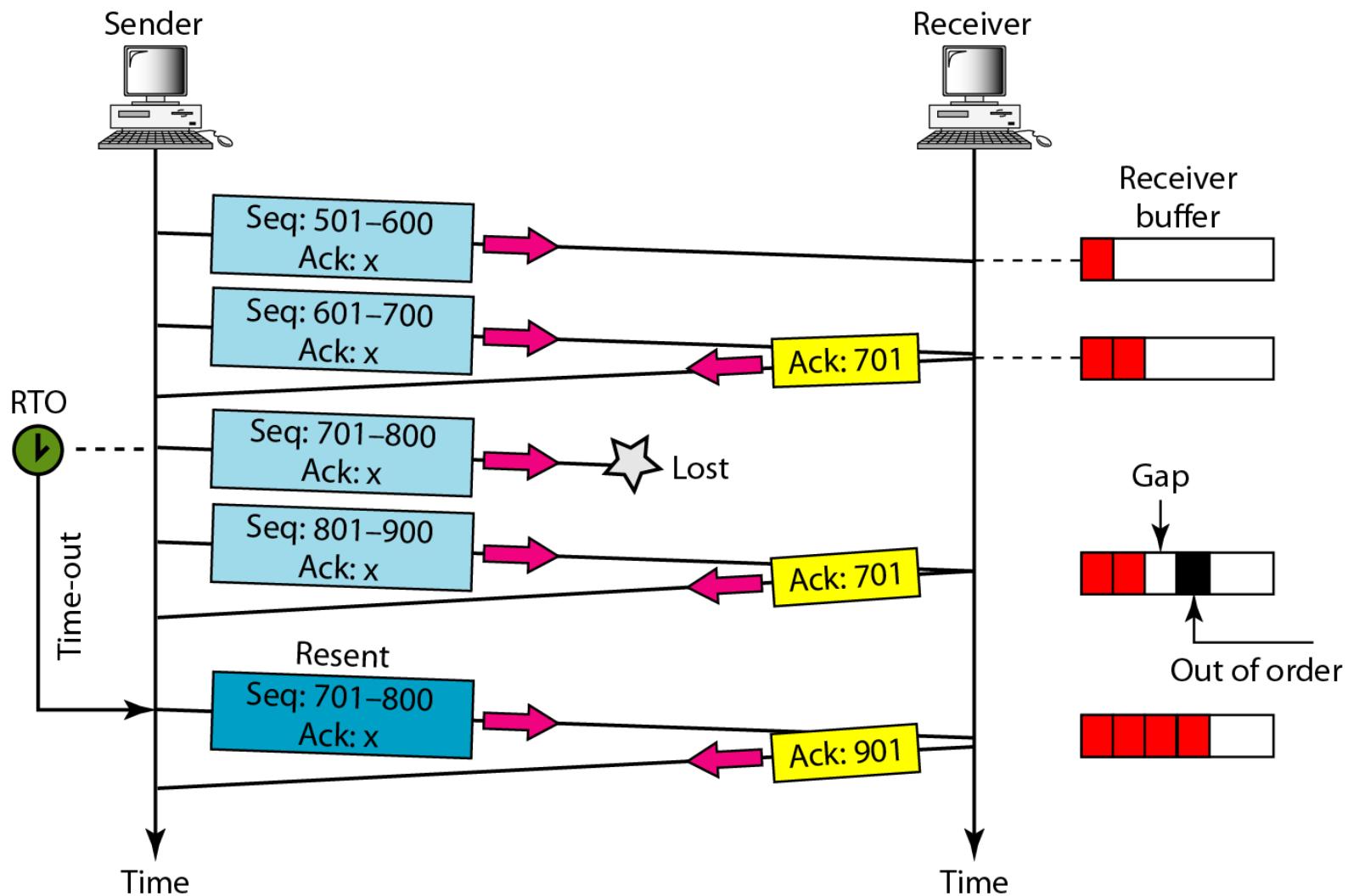
---

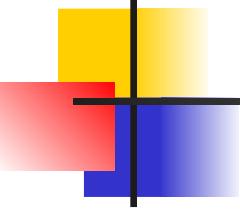
**Data may arrive out of order and be temporarily stored by the receiving TCP, but TCP guarantees that no out-of-order segment is delivered to the process.**

**Figure 23.24 Normal operation**



**Figure 23.25 Lost segment**





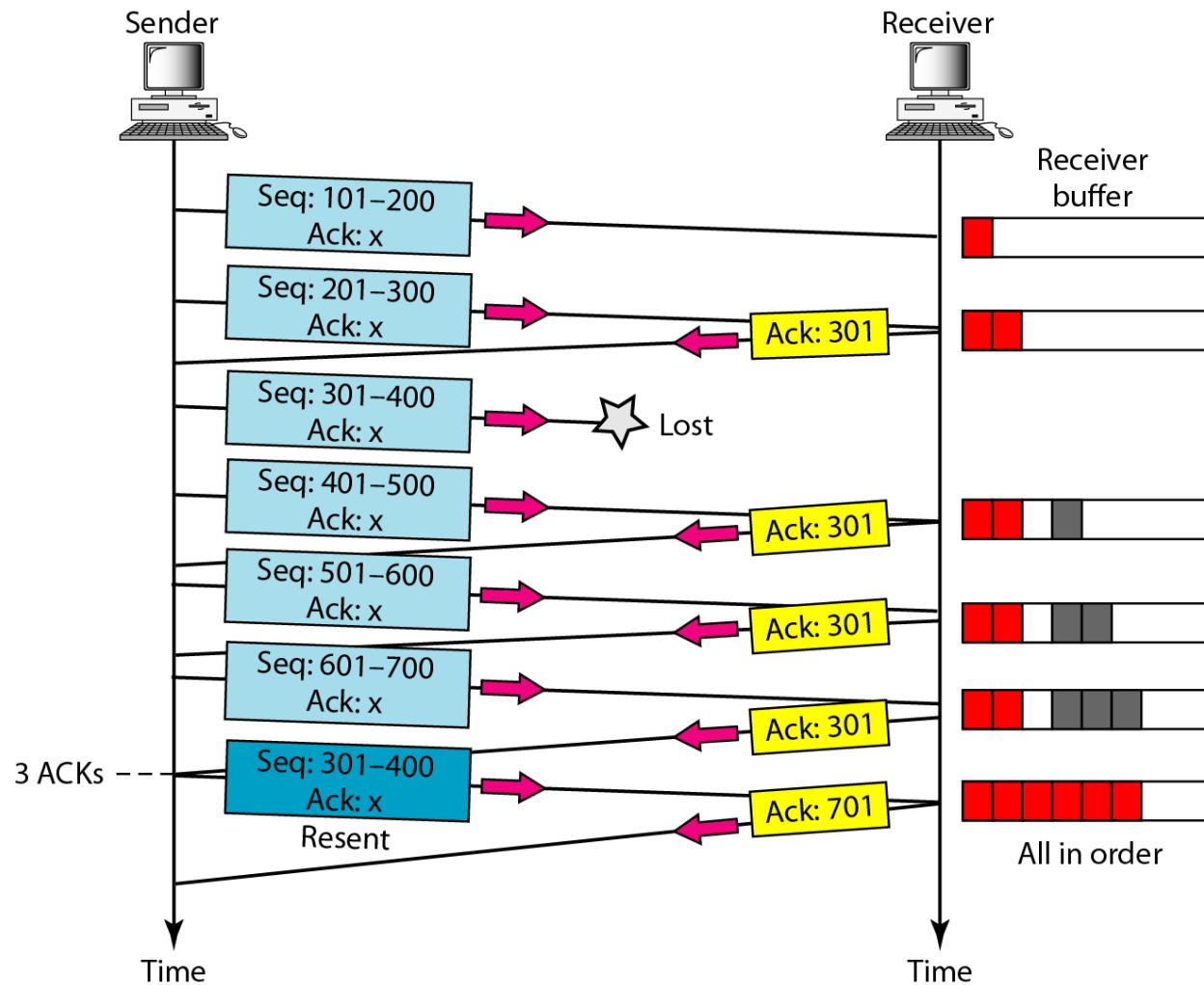
## *Note*

---

**The receiver TCP delivers only ordered data to the process.**

---

## Figure 23.26 Fast retransmission





# Data Communications and Networking

Fourth Edition

Forouzan

## Chapter 24

# Congestion Control and Quality of Service

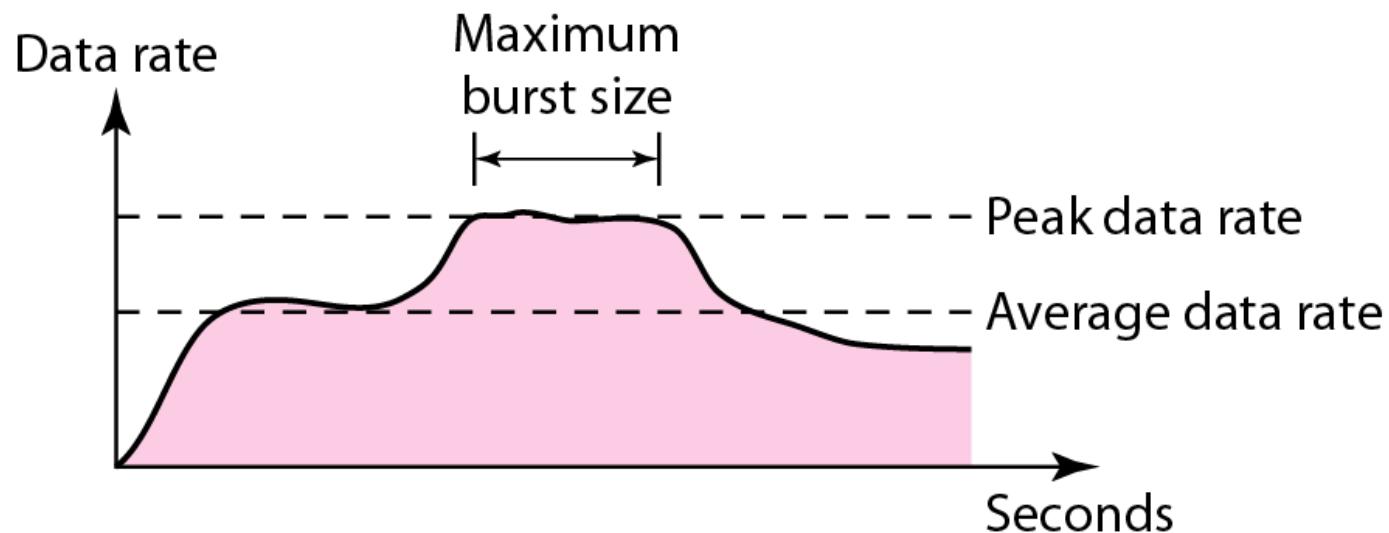
## 24-1 DATA TRAFFIC

*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.*

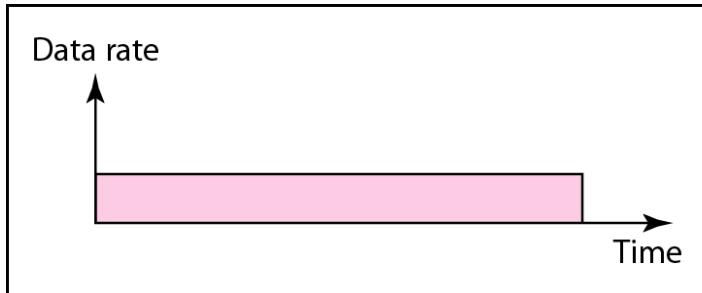
**Topics discussed in this section:**

Traffic Descriptor  
Traffic Profiles

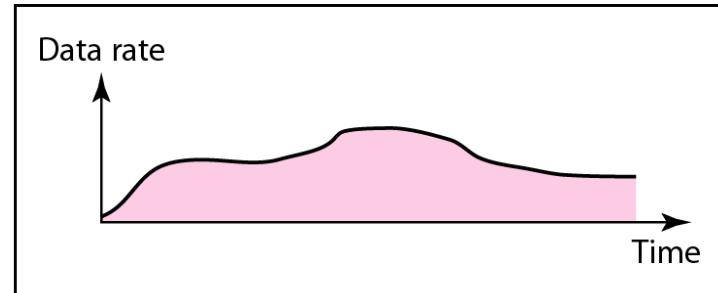
**Figure 24.1** *Traffic descriptors*



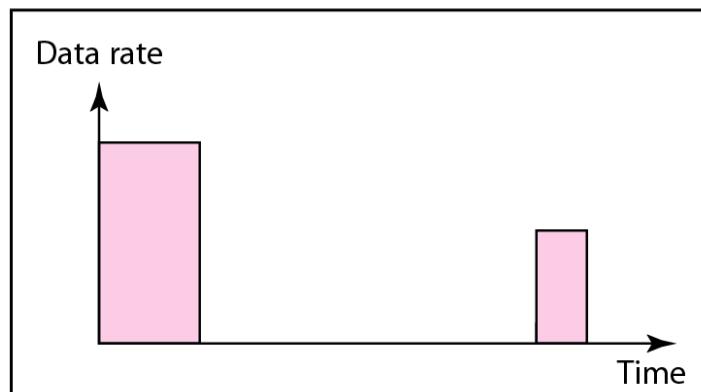
**Figure 24.2** *Three traffic profiles*



a. Constant bit rate



b. Variable bit rate



c. Bursty

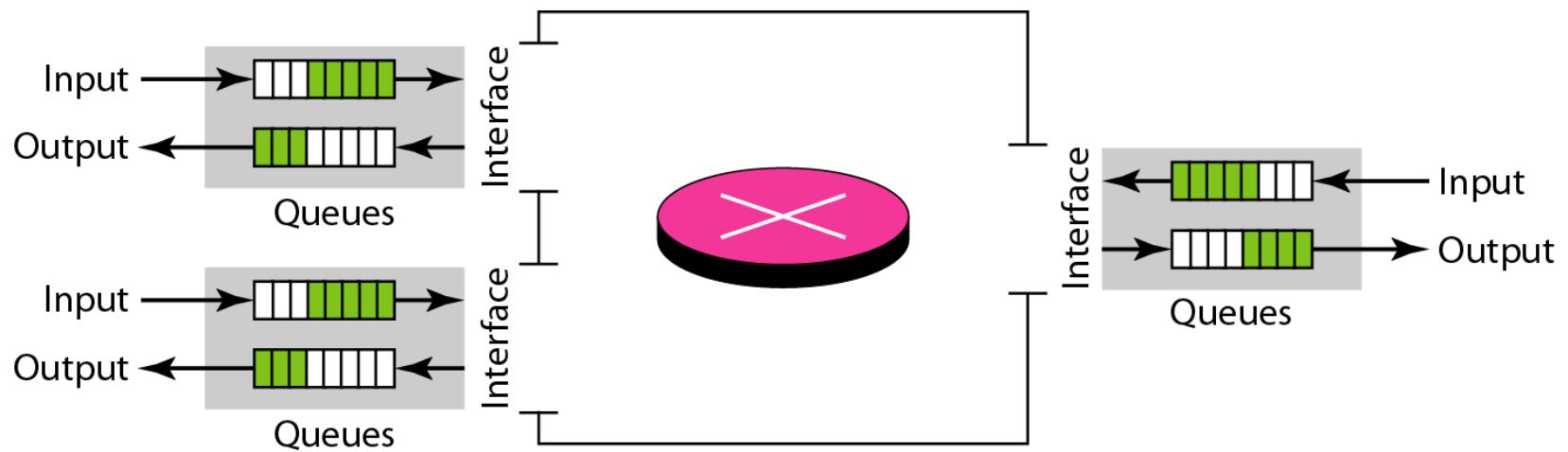
## **24-2 CONGESTION**

*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.*

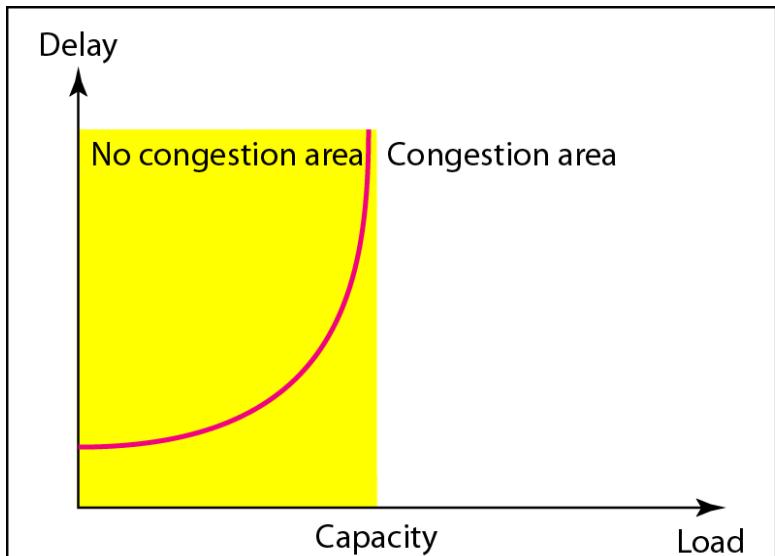
**Topics discussed in this section:**

**Network Performance**

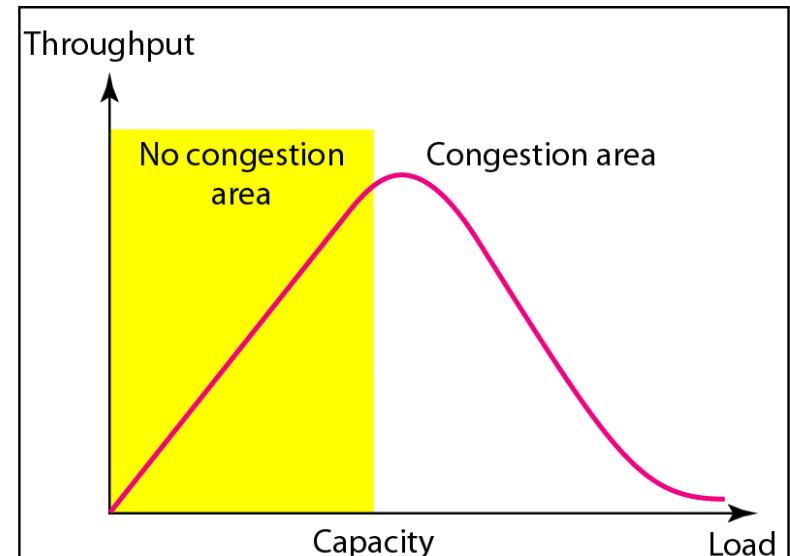
**Figure 24.3** *Queues in a router*



**Figure** *Packet delay and throughput as functions of load*



a. Delay as a function of load



b. Throughput as a function of load

## 24-3 CONGESTION CONTROL

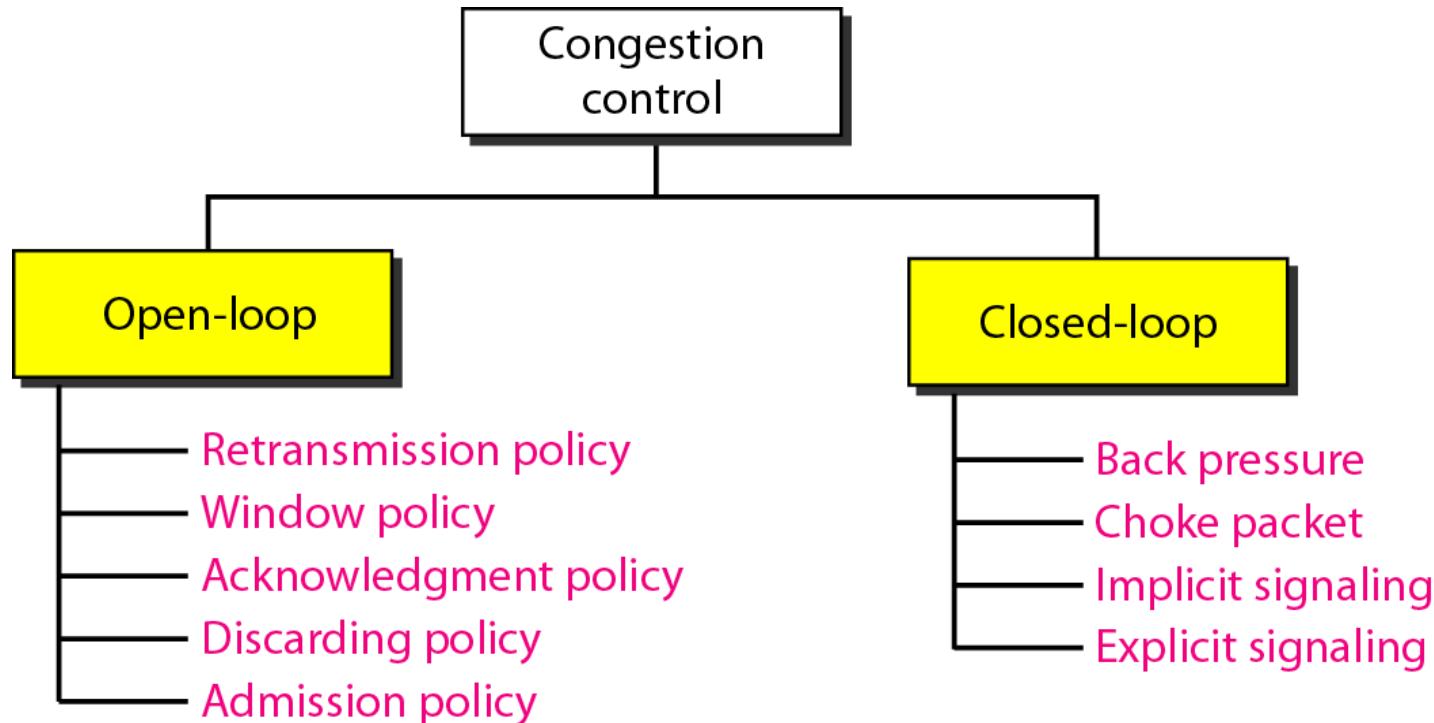
*Congestion control refers to techniques and mechanisms that can either prevent congestion, before it happens, or remove congestion, after it has happened. In general, we can divide congestion control mechanisms into two broad categories: open-loop congestion control (prevention) and closed-loop congestion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

**Topics discussed in this section:**

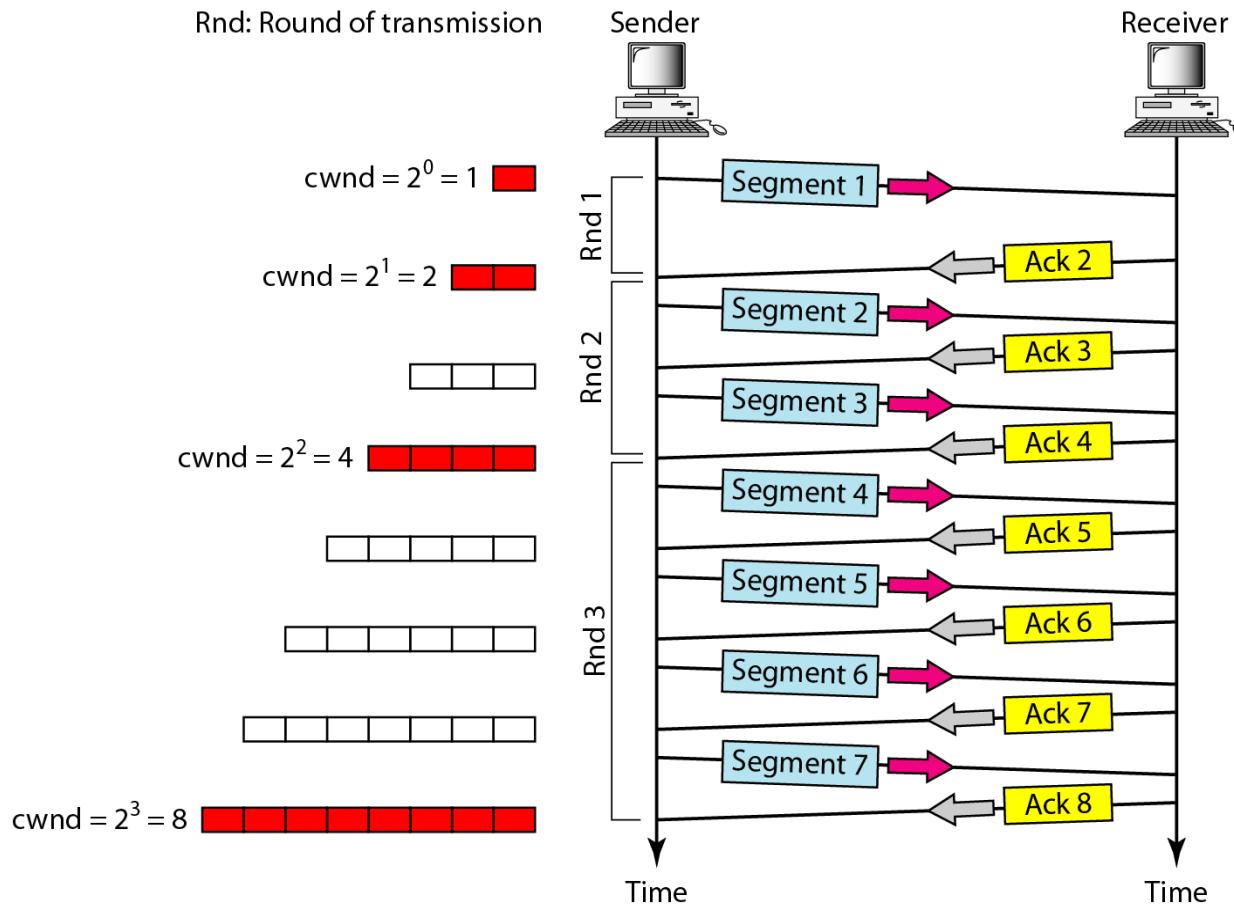
Open-Loop Congestion Control

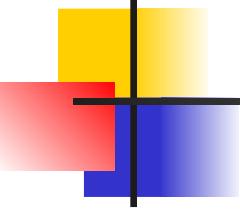
Closed-Loop Congestion Control

**Figure 24.5** *Congestion control categories*



**Figure 24.8 Slow start, exponential increase**





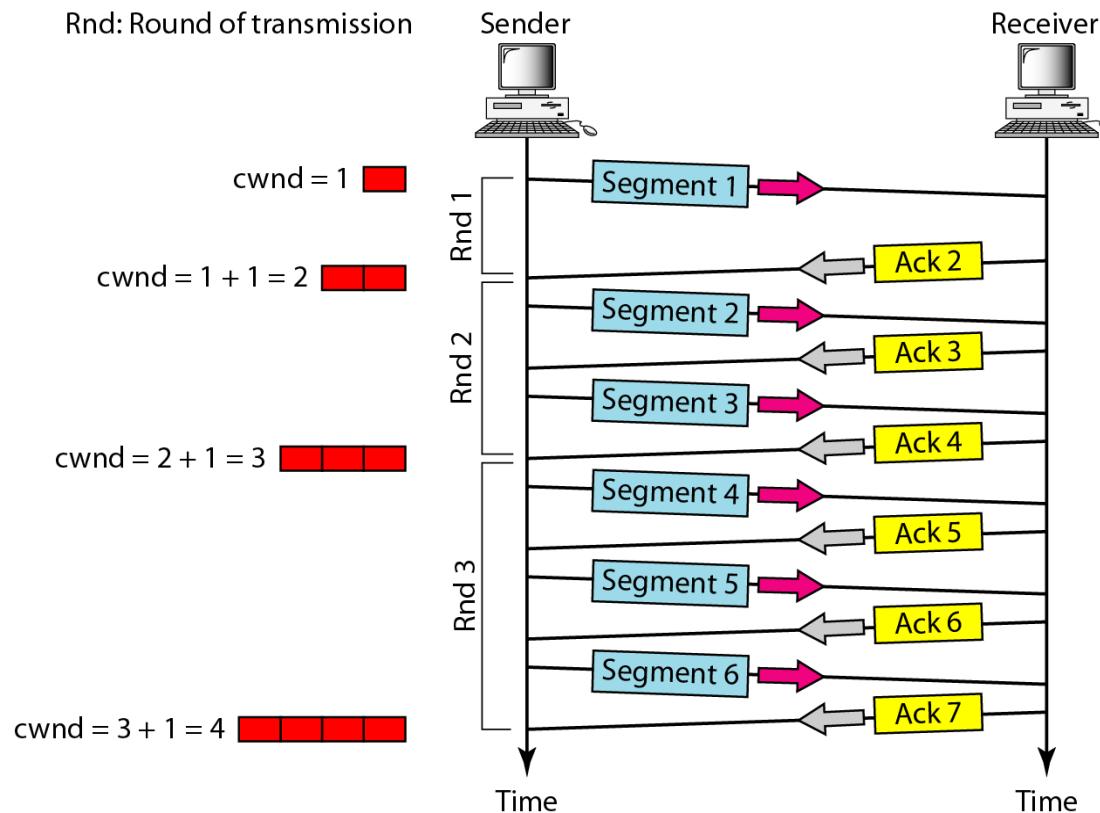
## ***Note***

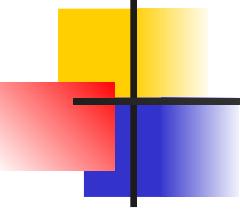
---

**In the slow-start algorithm, the size of the congestion window increases exponentially until it reaches a threshold.**

---

**Figure 24.9 Congestion avoidance, additive increase**

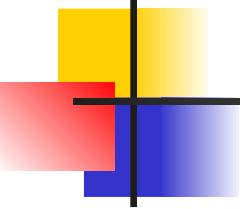




## **Note**

---

**In the congestion avoidance algorithm,  
the size of the congestion window  
increases additively until  
congestion is detected.**



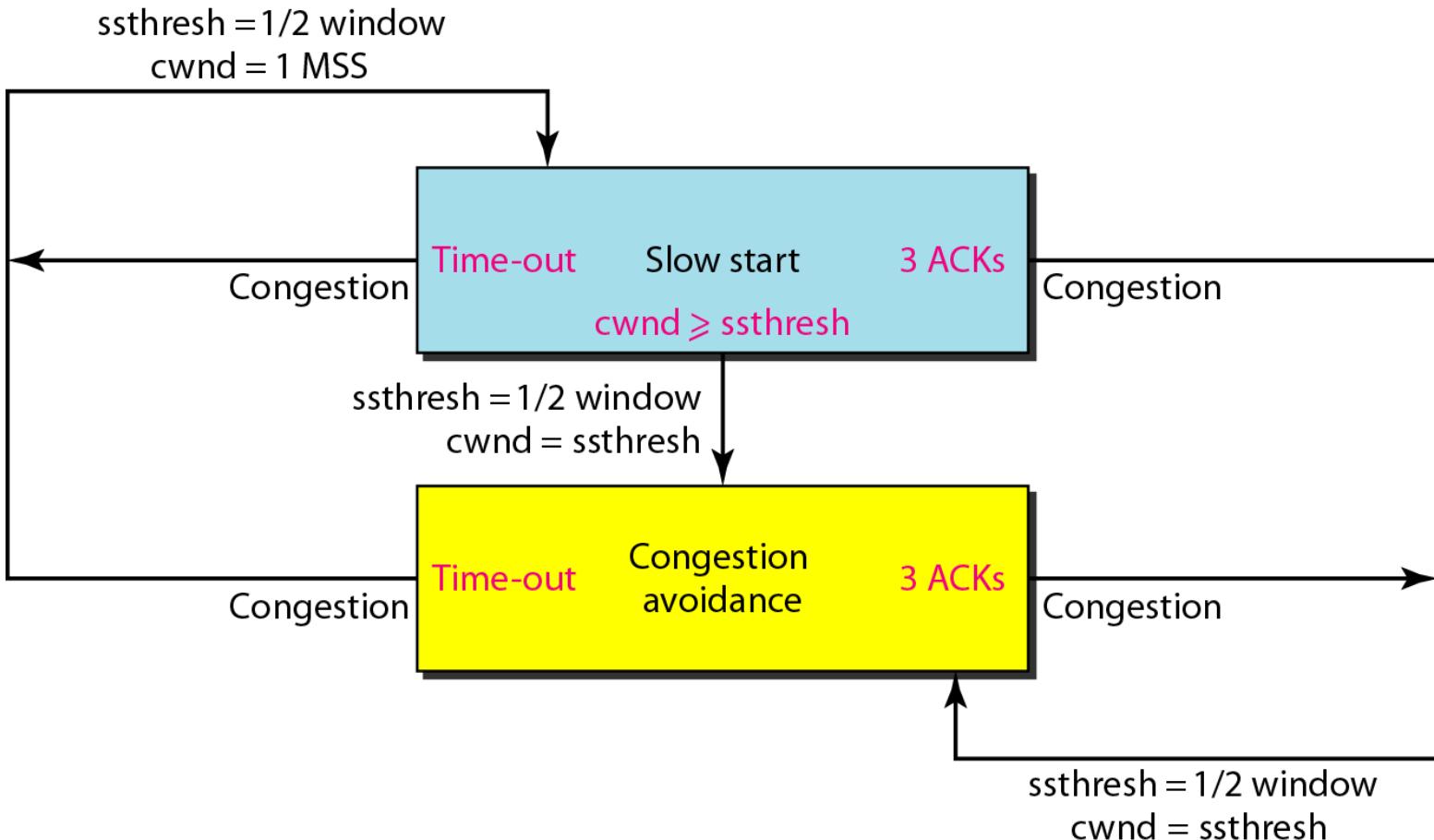
## **Note**

---

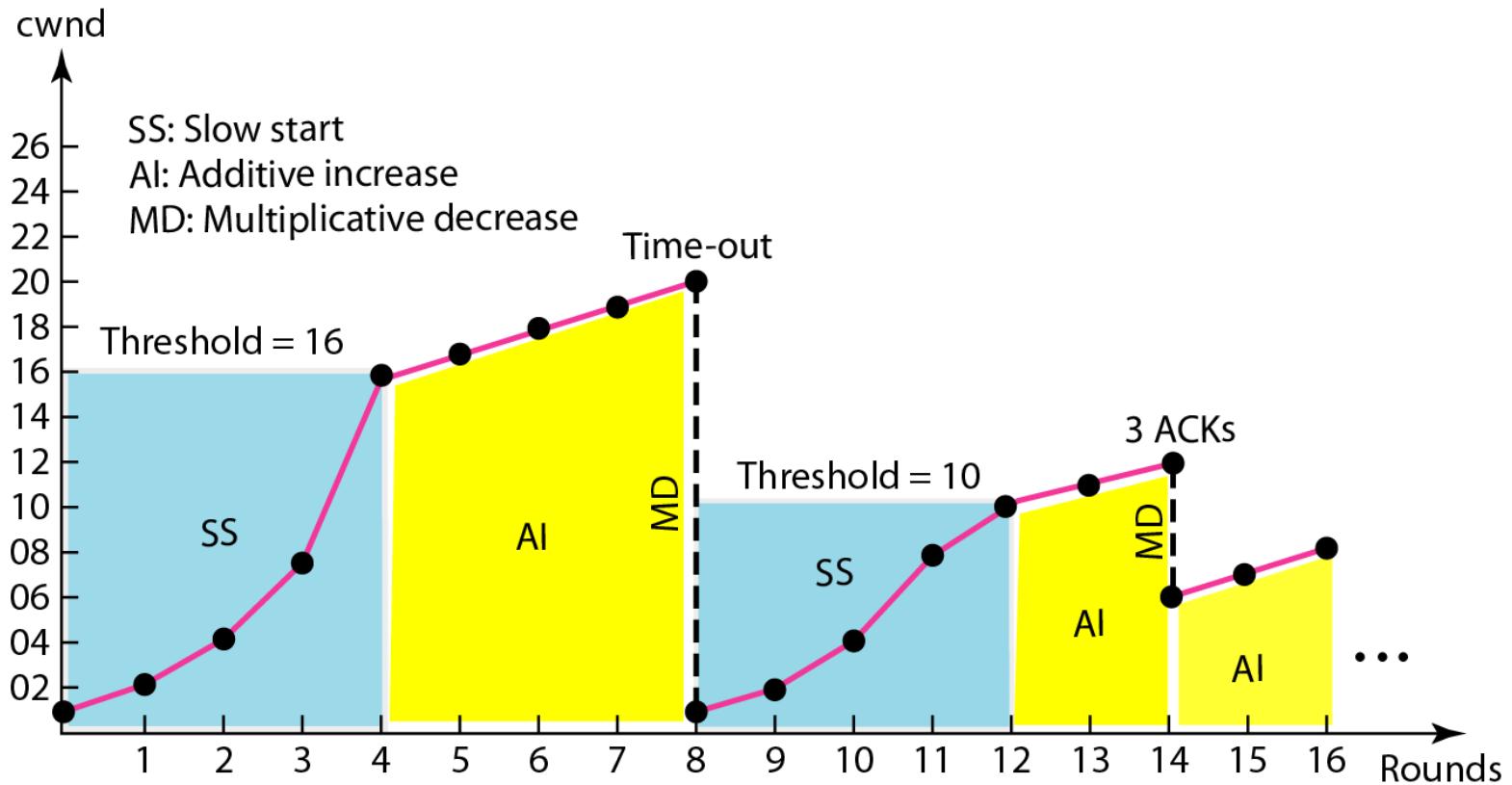
**An implementation reacts to congestion detection in one of the following ways:**

- If detection is by time-out, a new slow start phase starts.**
  - If detection is by three ACKs, a new congestion avoidance phase starts.**
-

**Figure 24.10** TCP congestion policy summary



**Figure 24.11 Congestion example**



## **24-5 QUALITY OF SERVICE**

*Quality of service (QoS) is an internetworking issue that has been discussed more than defined. We can informally define quality of service as something a flow seeks to attain.*

### **Topics discussed in this section:**

Flow Characteristics

Flow Classes

**Figure 24.15** *Flow characteristics*

