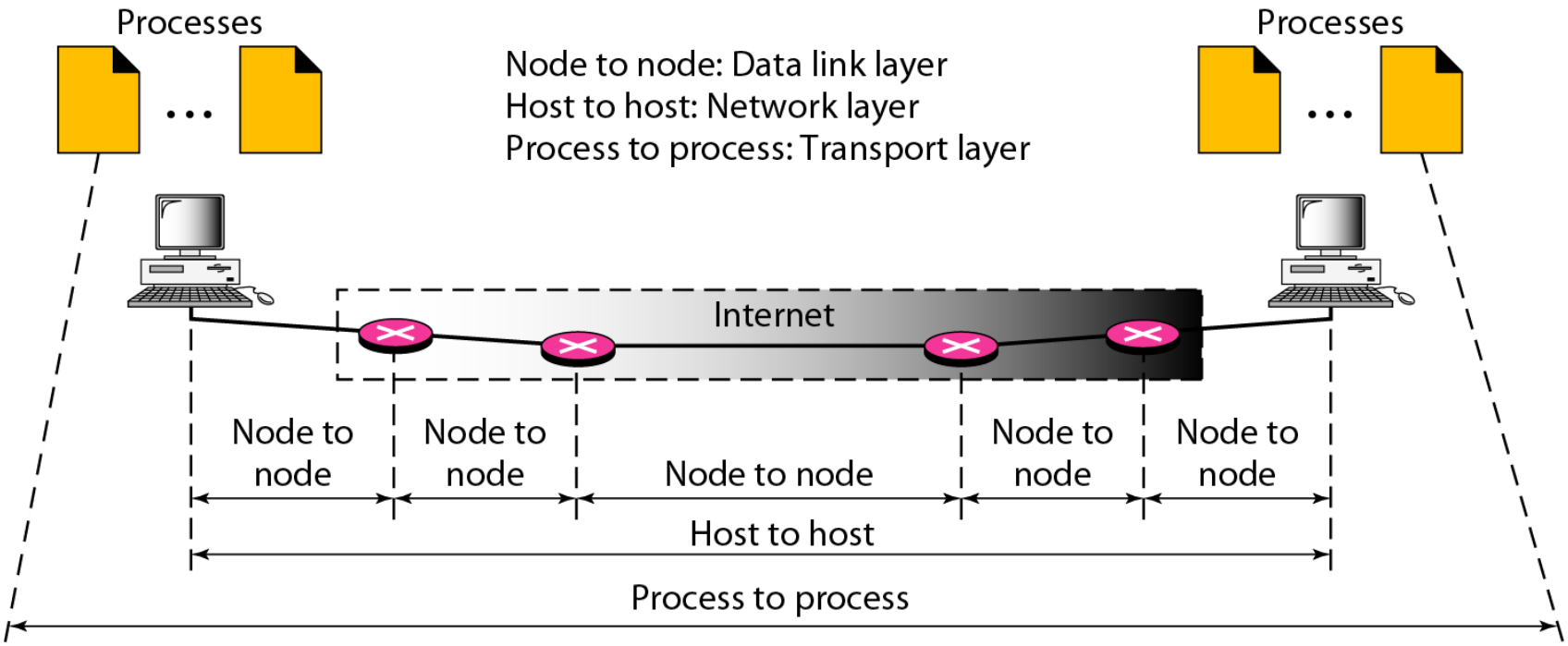
The Transport Layer: TCP, UDP

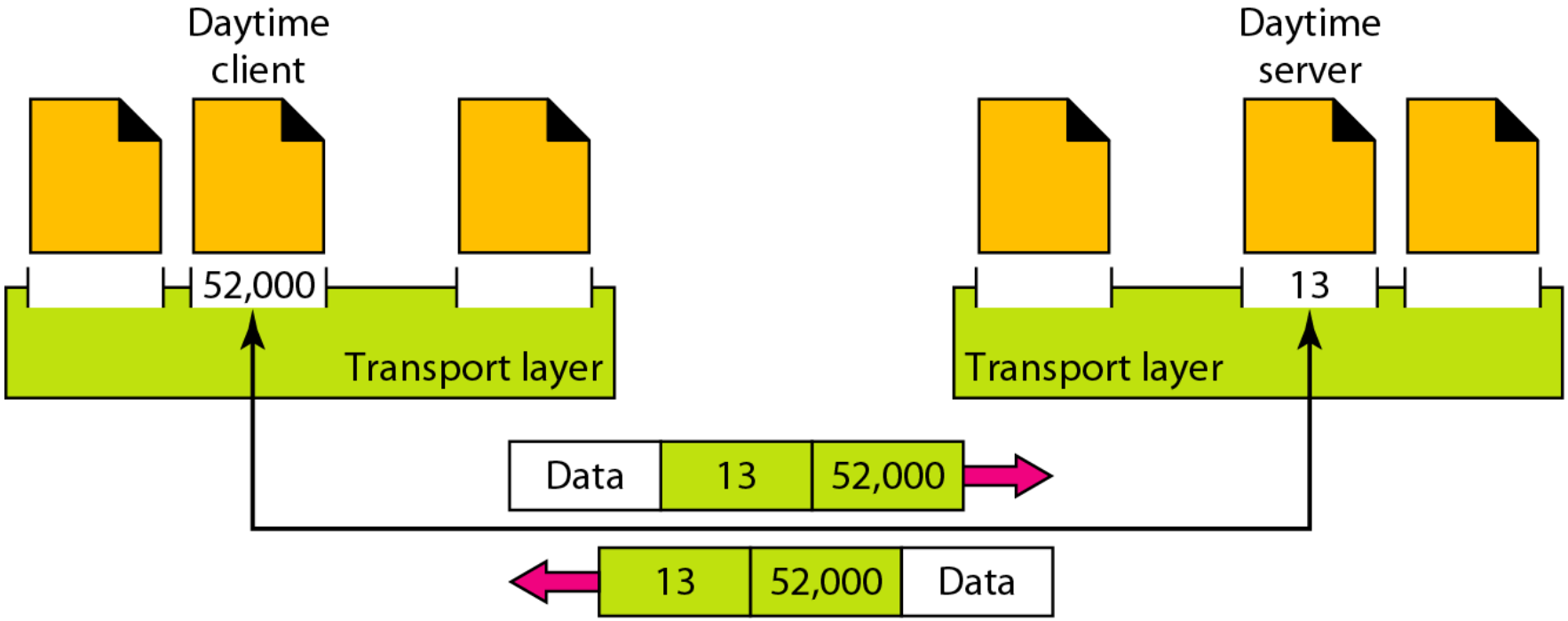
PROCESS-TO-PROCESS DELIVERY

- *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.*

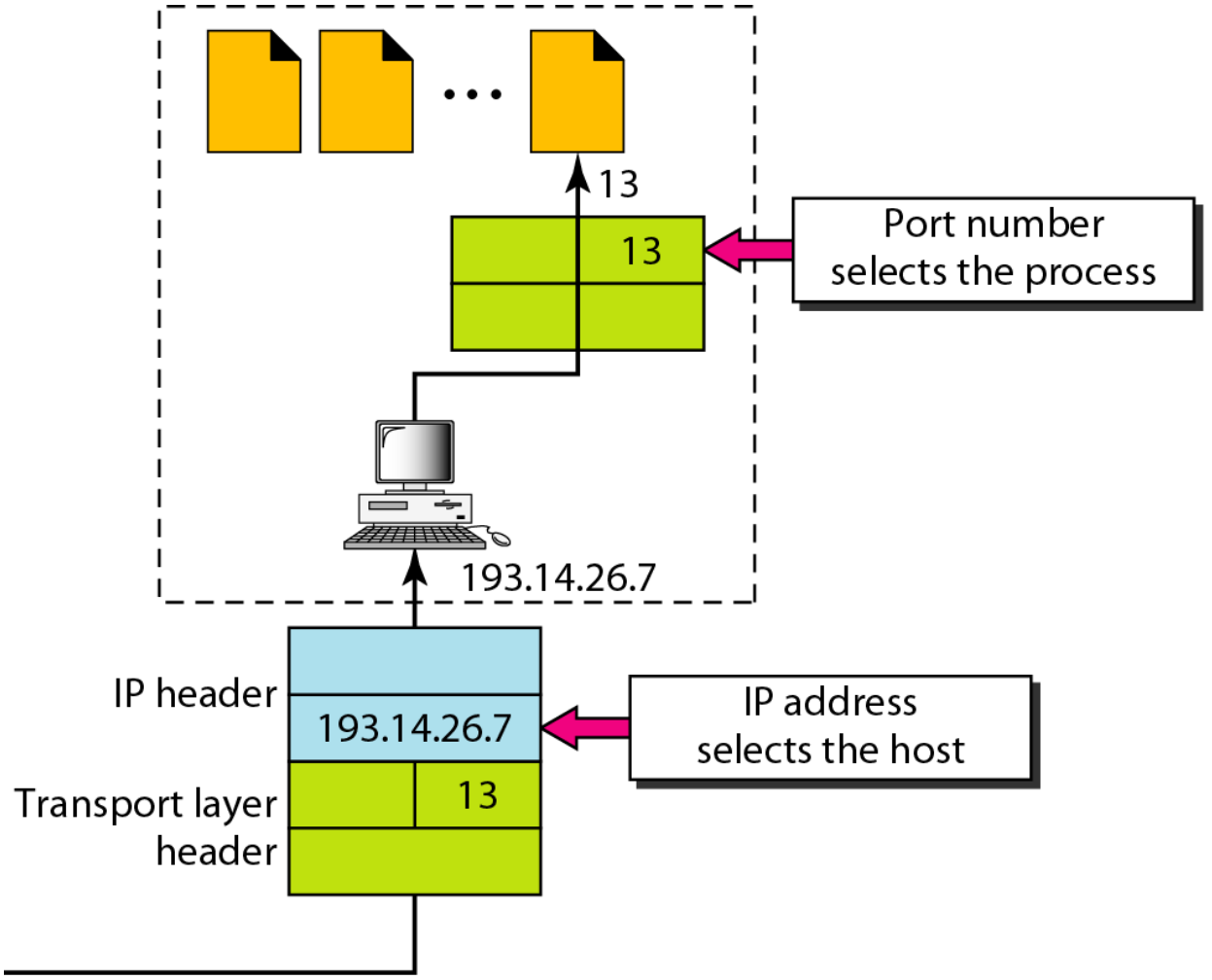
Types of data deliveries



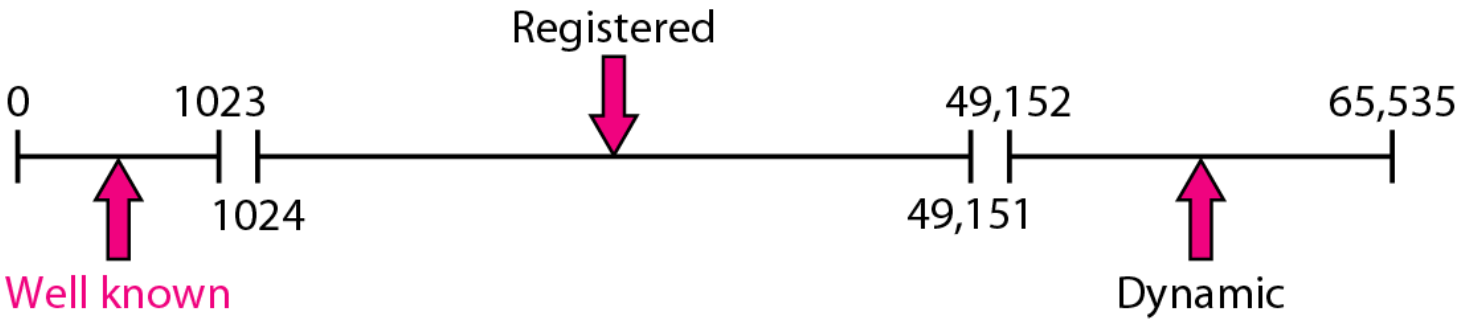
Port numbers



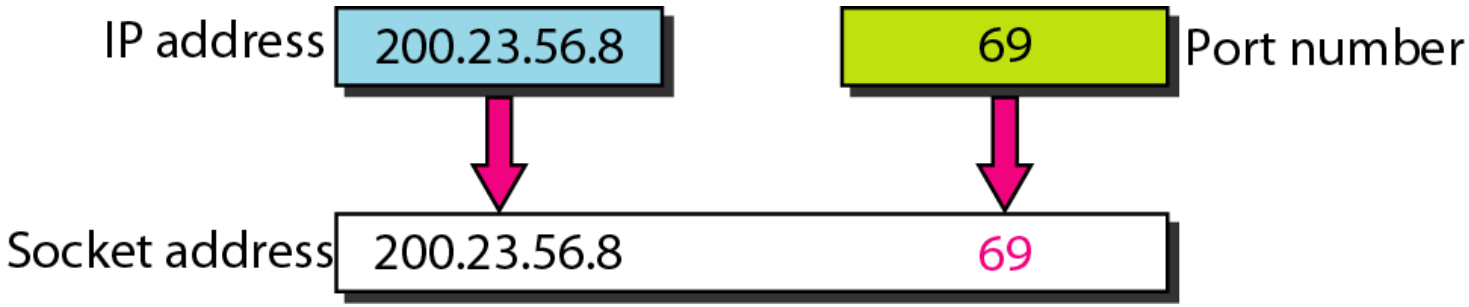
IP addresses versus port numbers



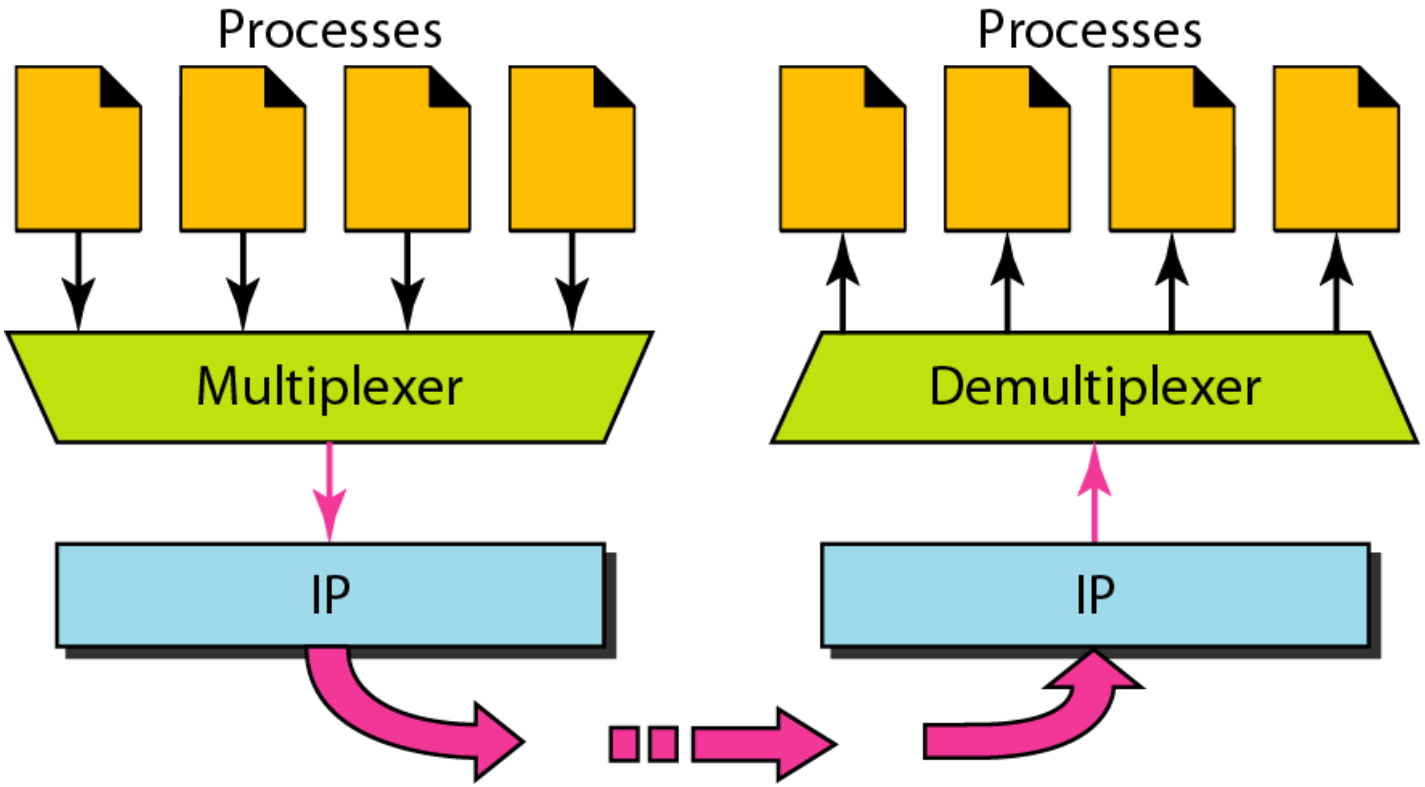
IANA ranges


Registered
0
1023
49,152
65,535
1024
49,151
Well known
Dynamic

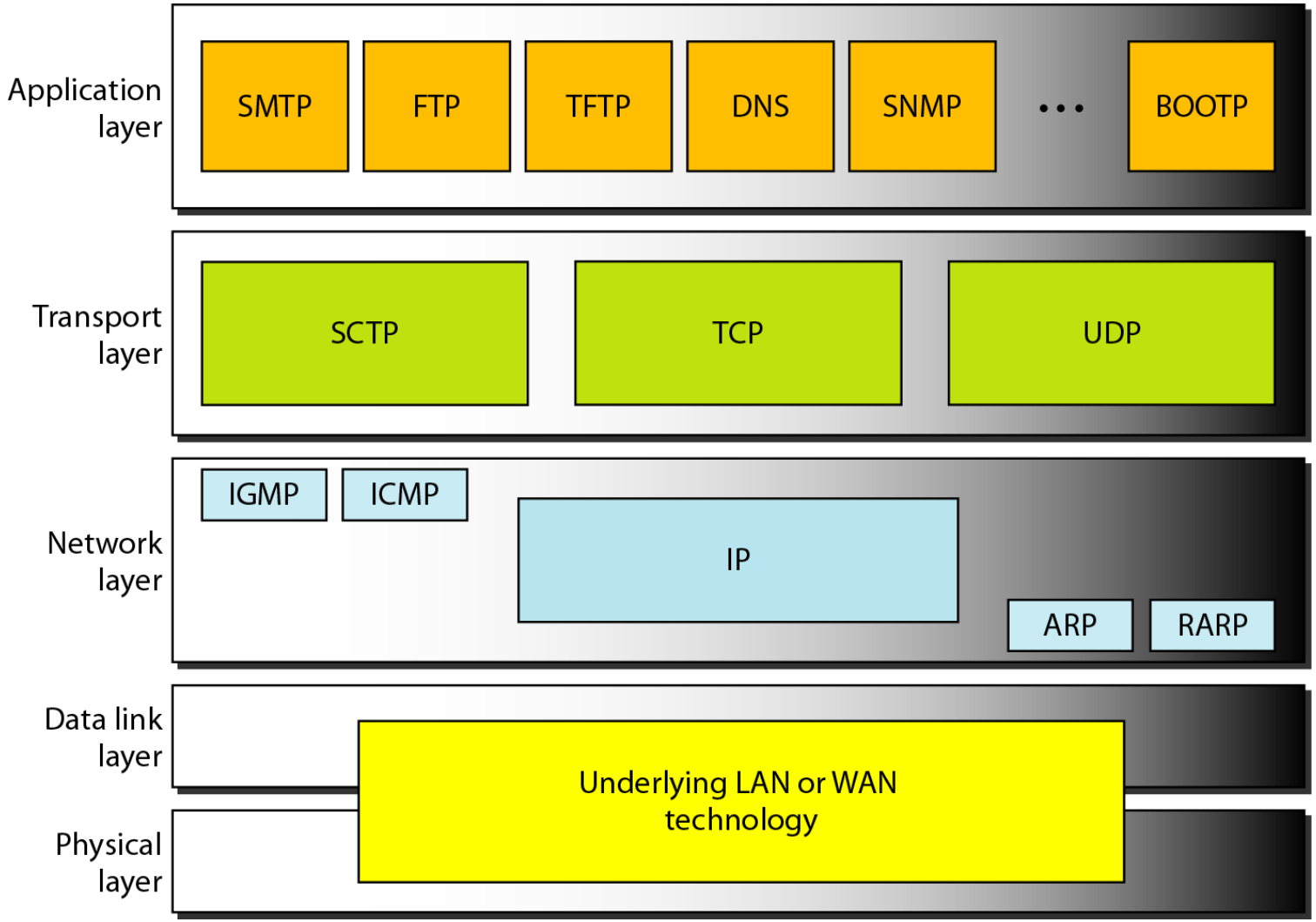
Socket address



Multiplexing and demultiplexing



Position of UDP, TCP, and SCTP in TCP/IP suite



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*

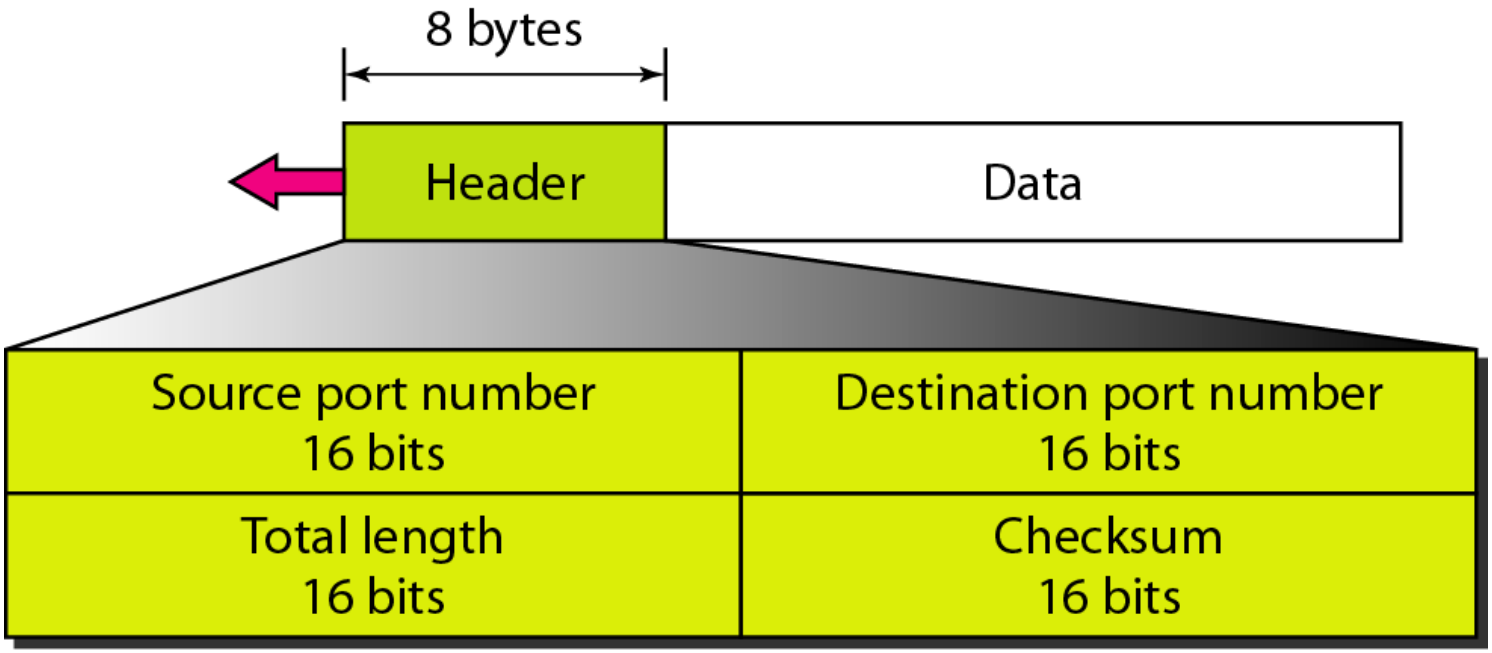
Well-known ports used with UDP

Port	*Protocol*	*Description*
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)

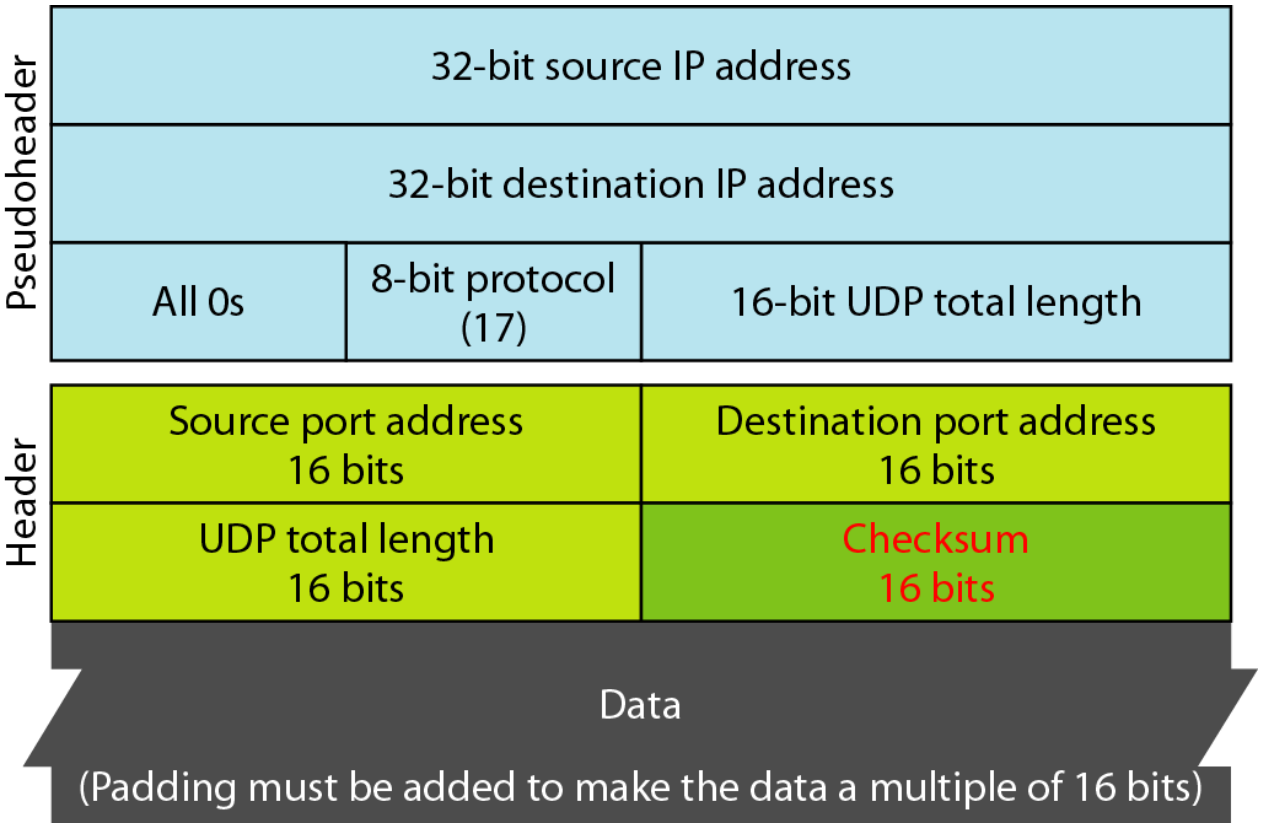
- *In UNIX, the well-known ports are stored in a file called /etc/services.*
- *Each line in this file gives the name of the server and the well-known port number.*
- *We can use the grep utility to extract the line corresponding to the desired application.*
- *The following shows the port for FTP.*
- *Note that FTP can use port 21 with either UDP or TCP.*

```
$ grep    ftp   /etc/services
ftp          21/tcp
ftp          21/udp
```

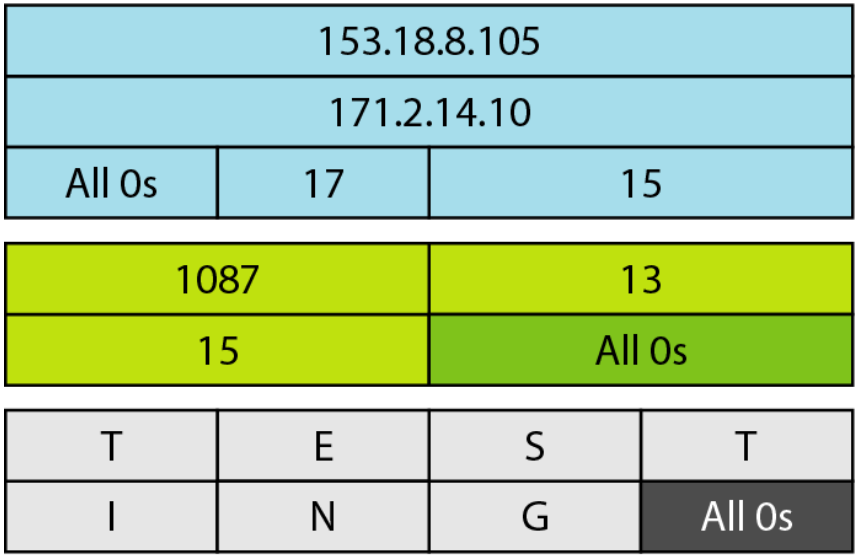
User datagram format



Pseudoheader for checksum calculation

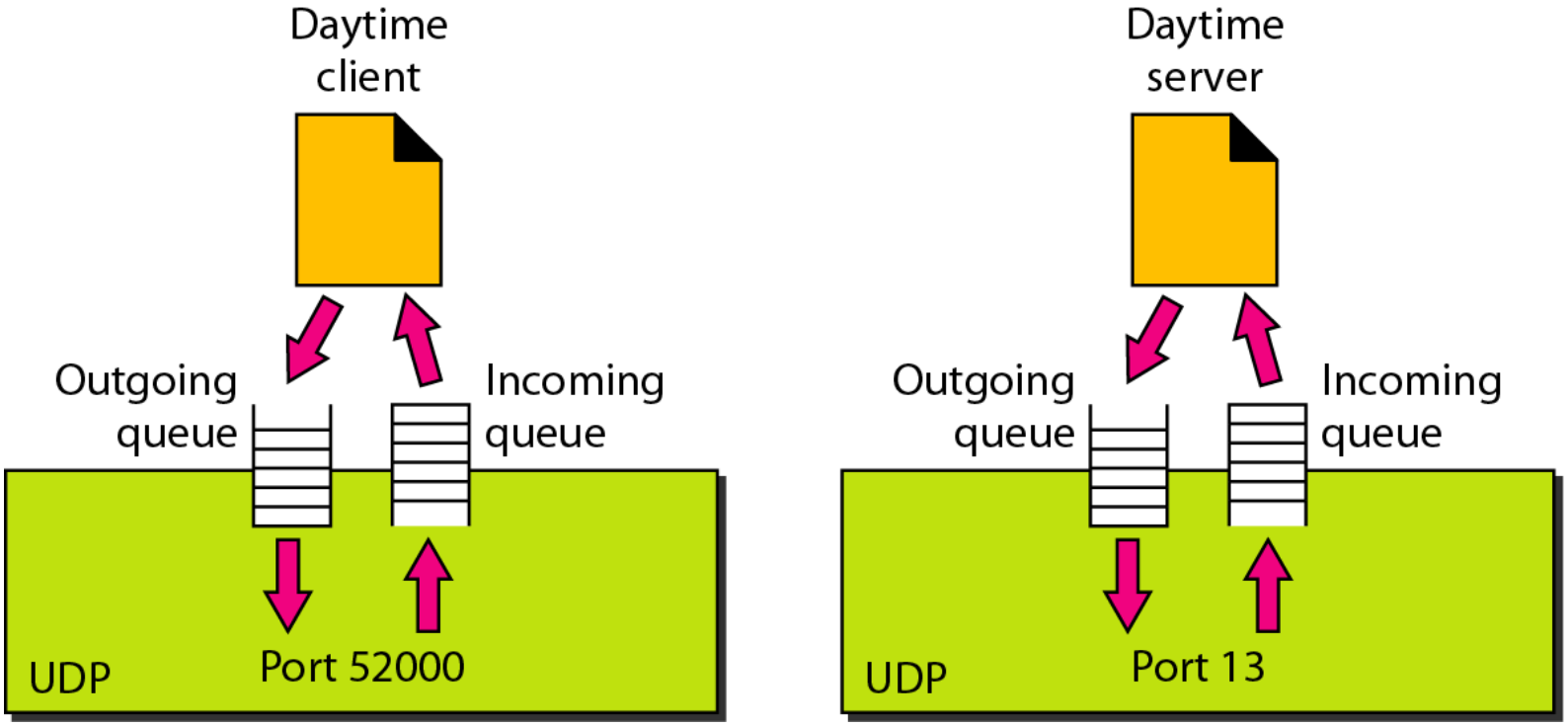


Checksum calculation of a simple UDP user datagram



10011001 00010010	153.18
00001000 01101001	8.105
10101011 00000010	171.2
00001110 00001010	14.10
00000000 00010001	0 and 17
00000000 00001111	15
00000100 00111111	1087
00000000 00001101	13
00000000 00001111	15
00000000 00000000	0 (checksum)
01010100 01000101	T and E
01010011 01010100	S and T
01001001 01001110	I and N
01000111 00000000	G and 0 (padding)
10010110 11101011	Sum
01101001 00010100	Checksum

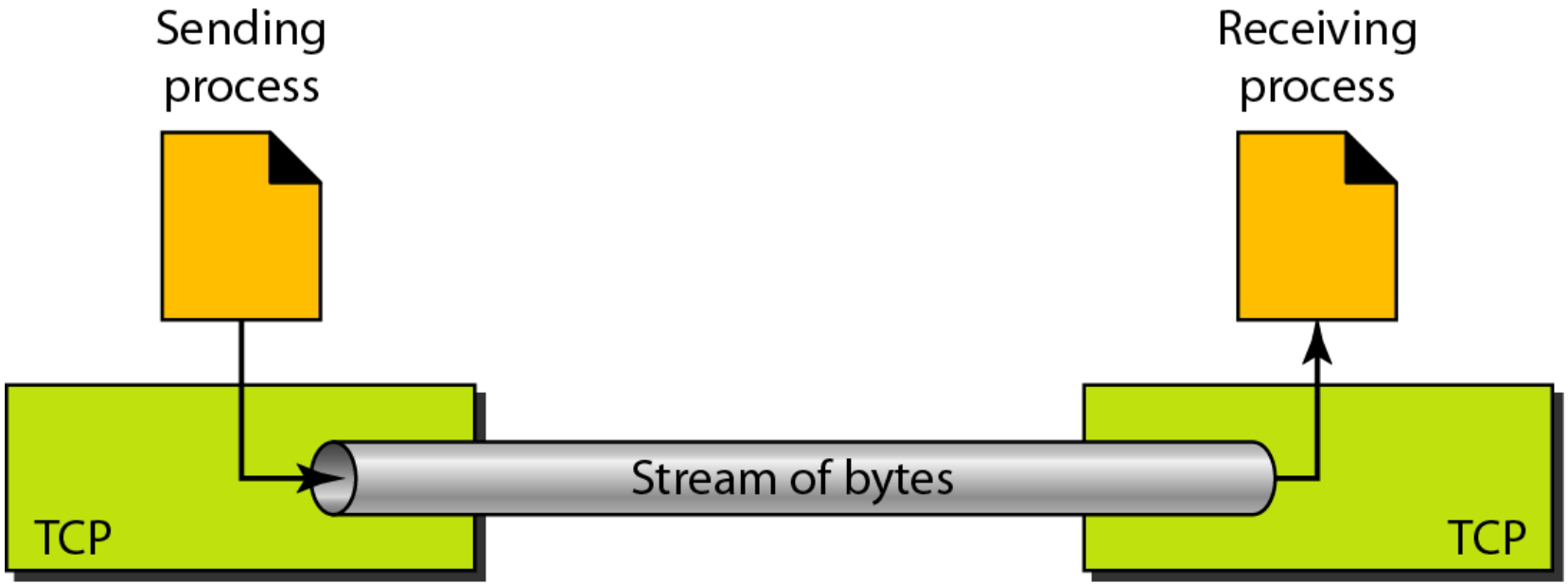
Queues in UDP



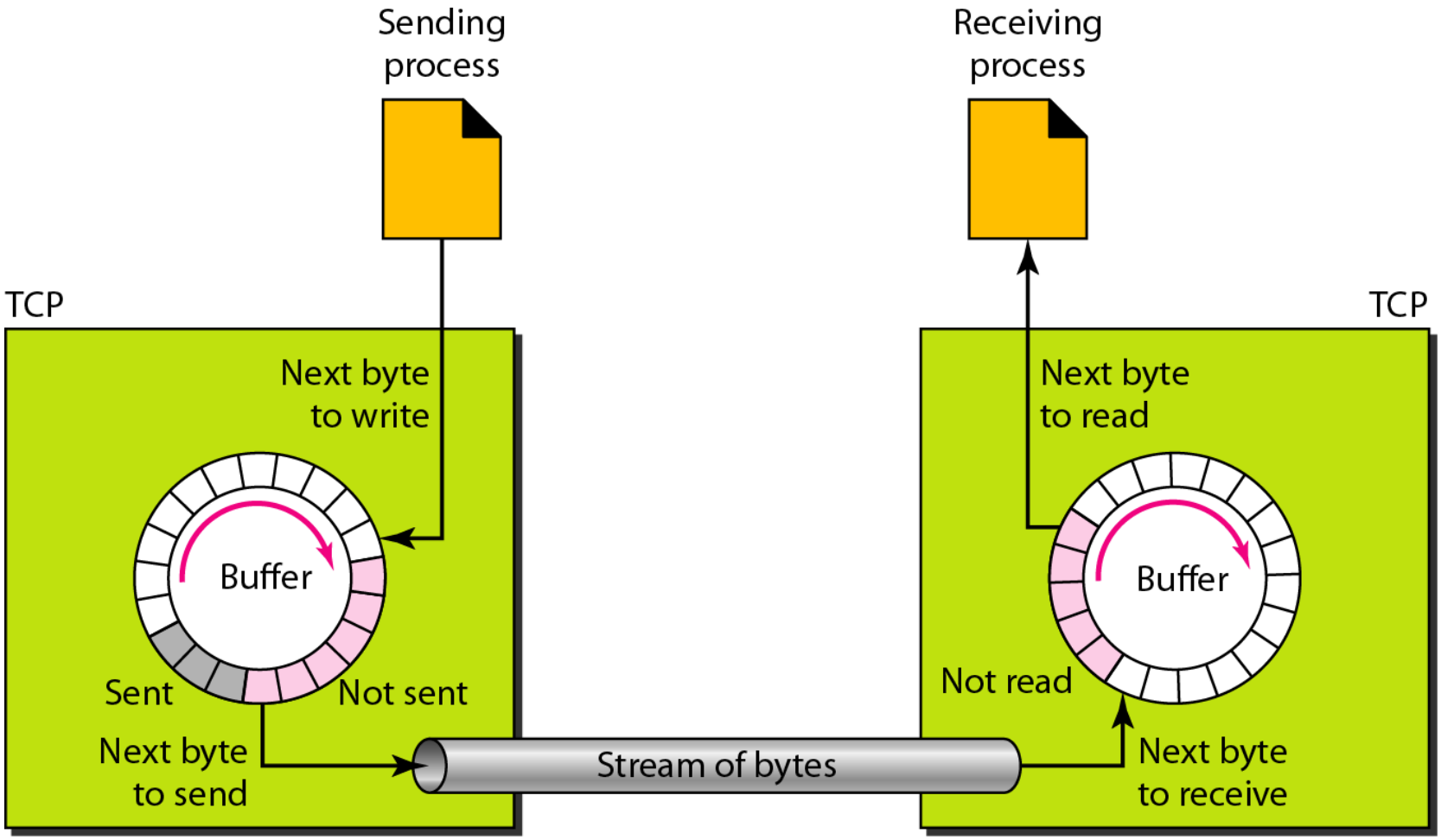
TCP

- *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*

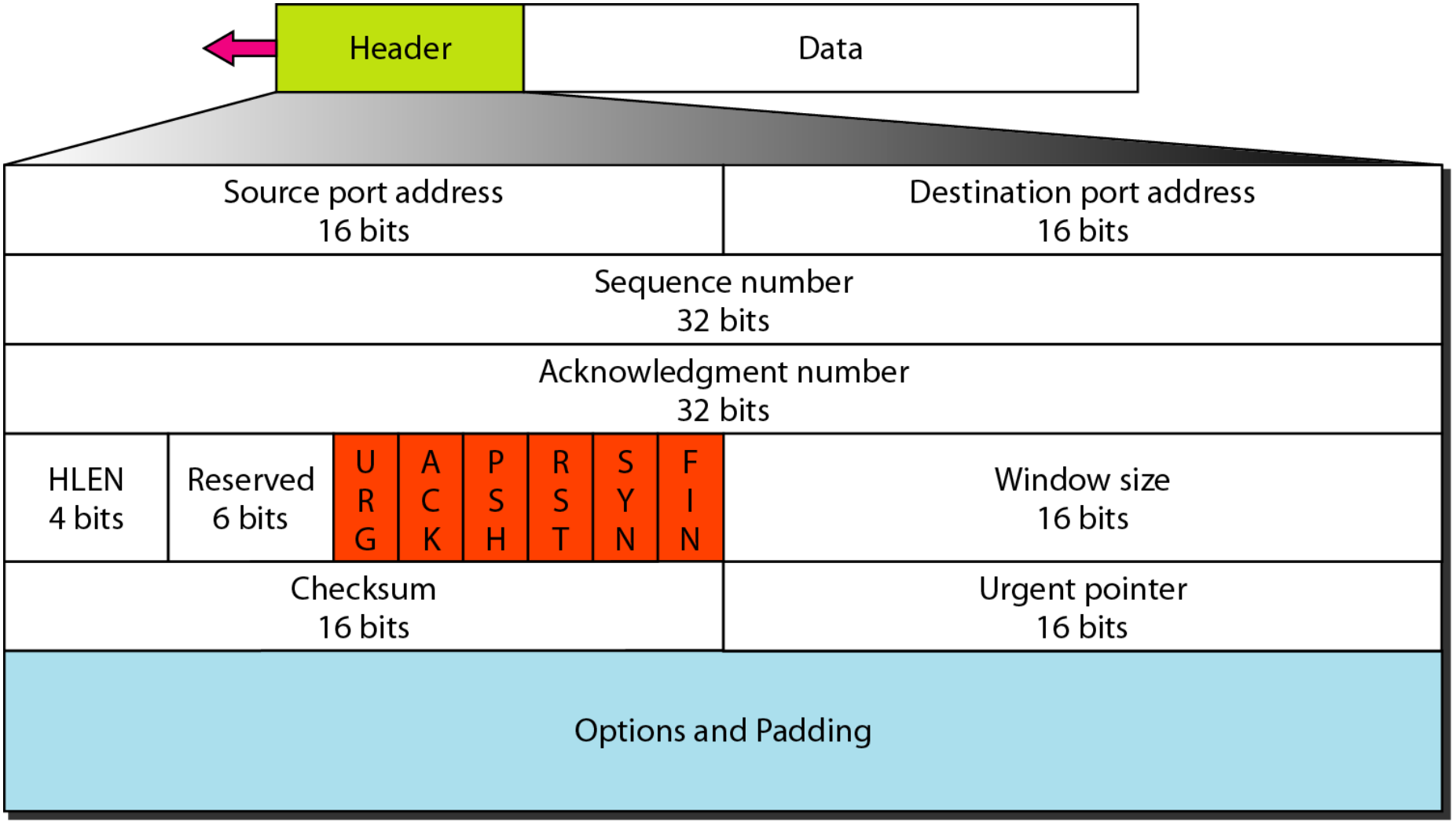
Stream delivery



Sending and receiving buffers



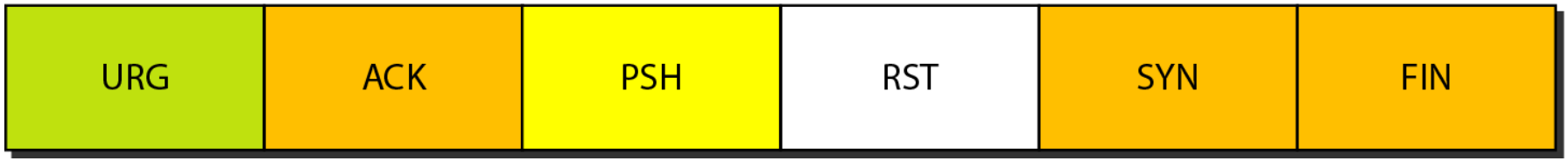
TCP segment format



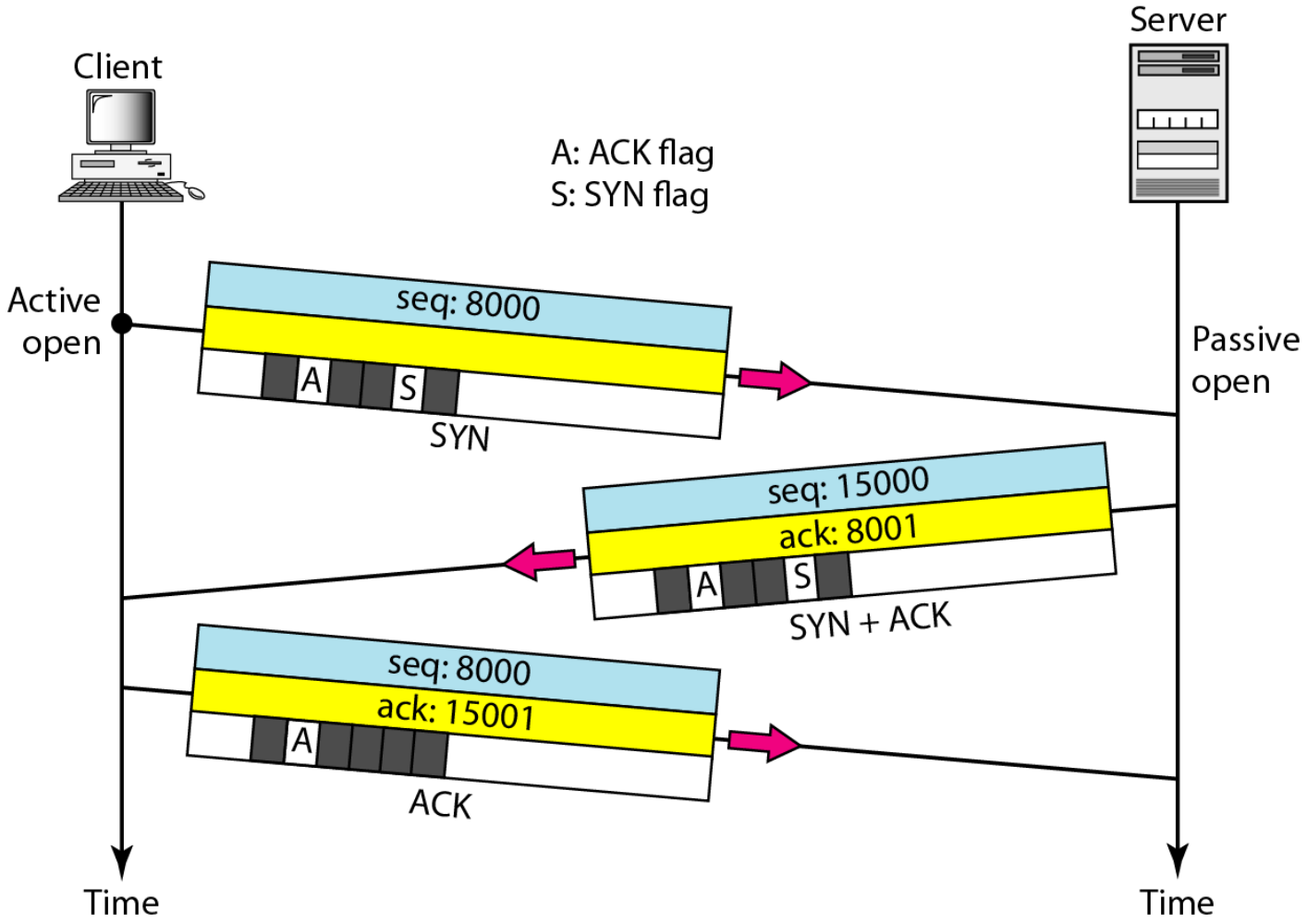
Control field

URG: Urgent pointer is valid
ACK: Acknowledgment is valid
PSH: Request for push

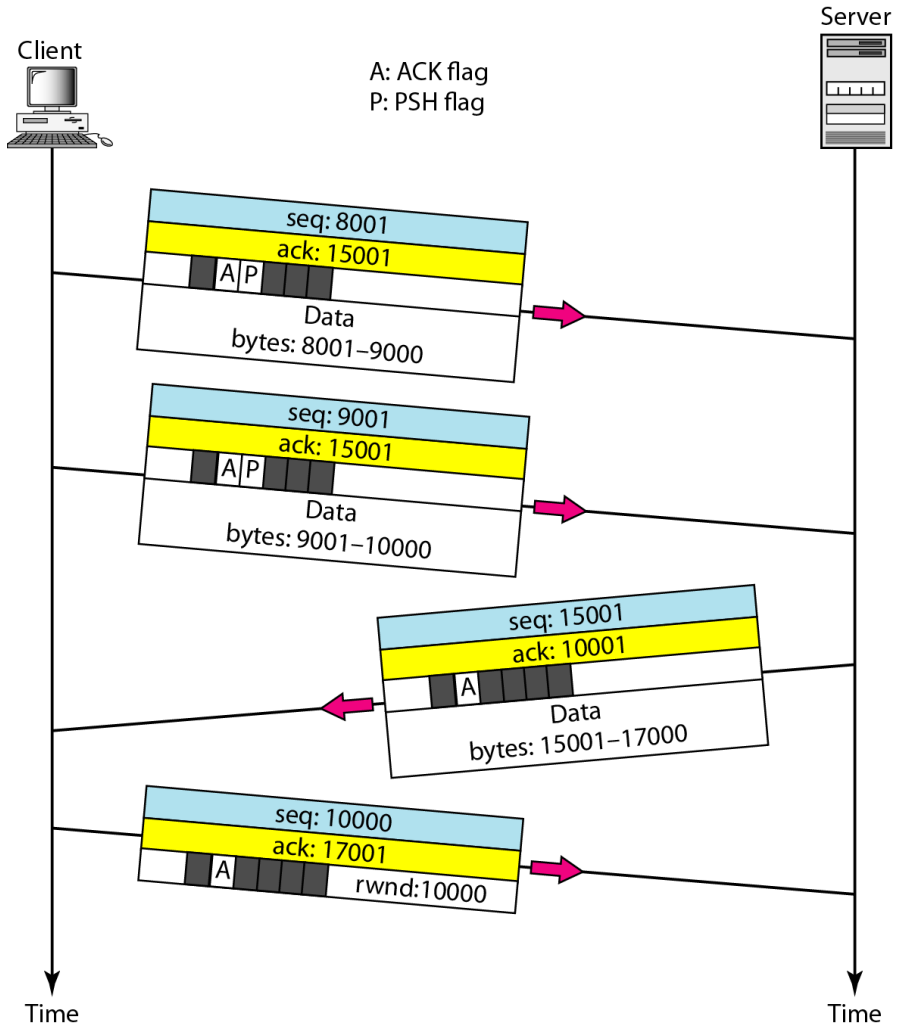
RST: Reset the connection
SYN: Synchronize sequence numbers
FIN: Terminate the connection



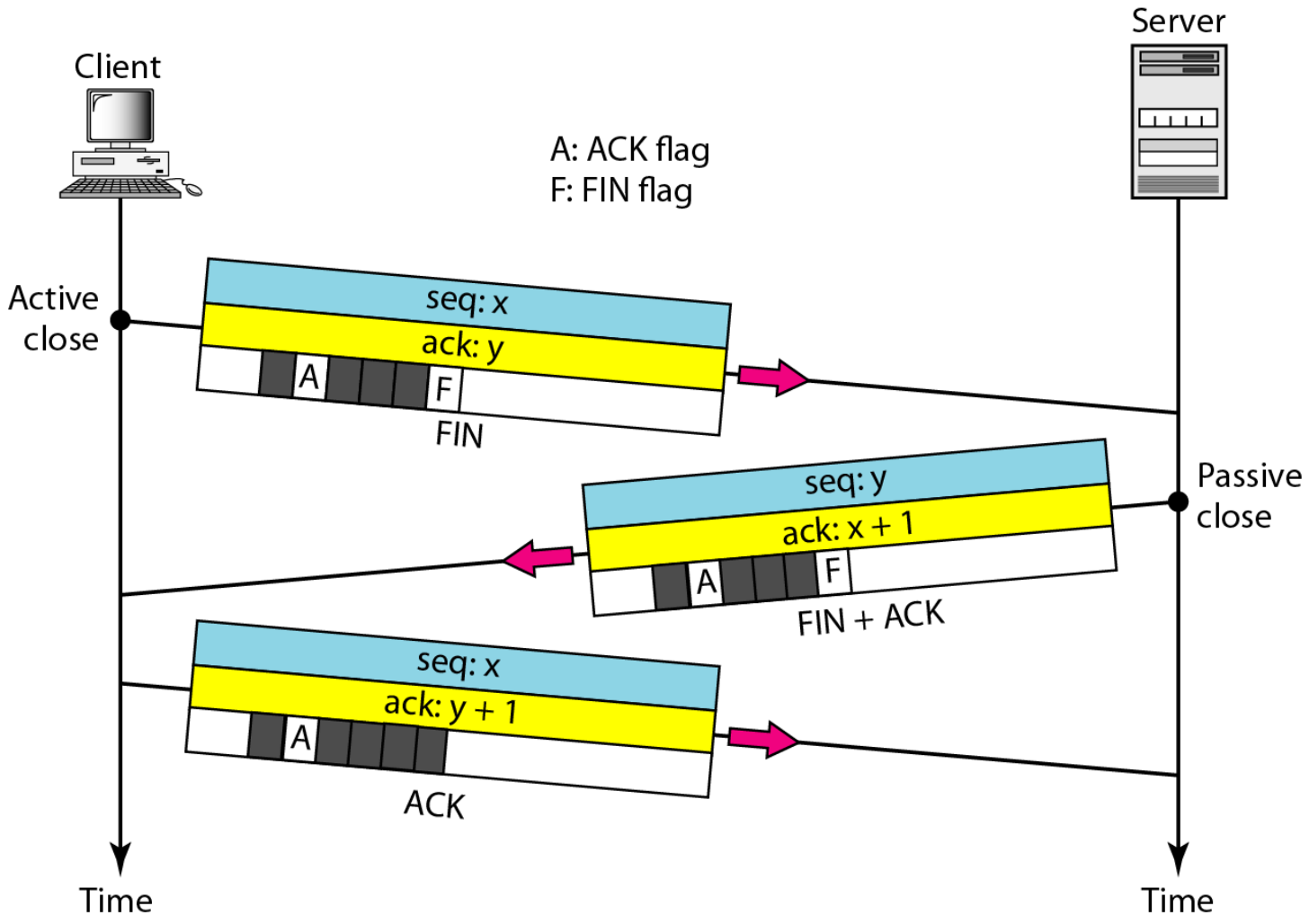
Connection establishment using three-way handshaking



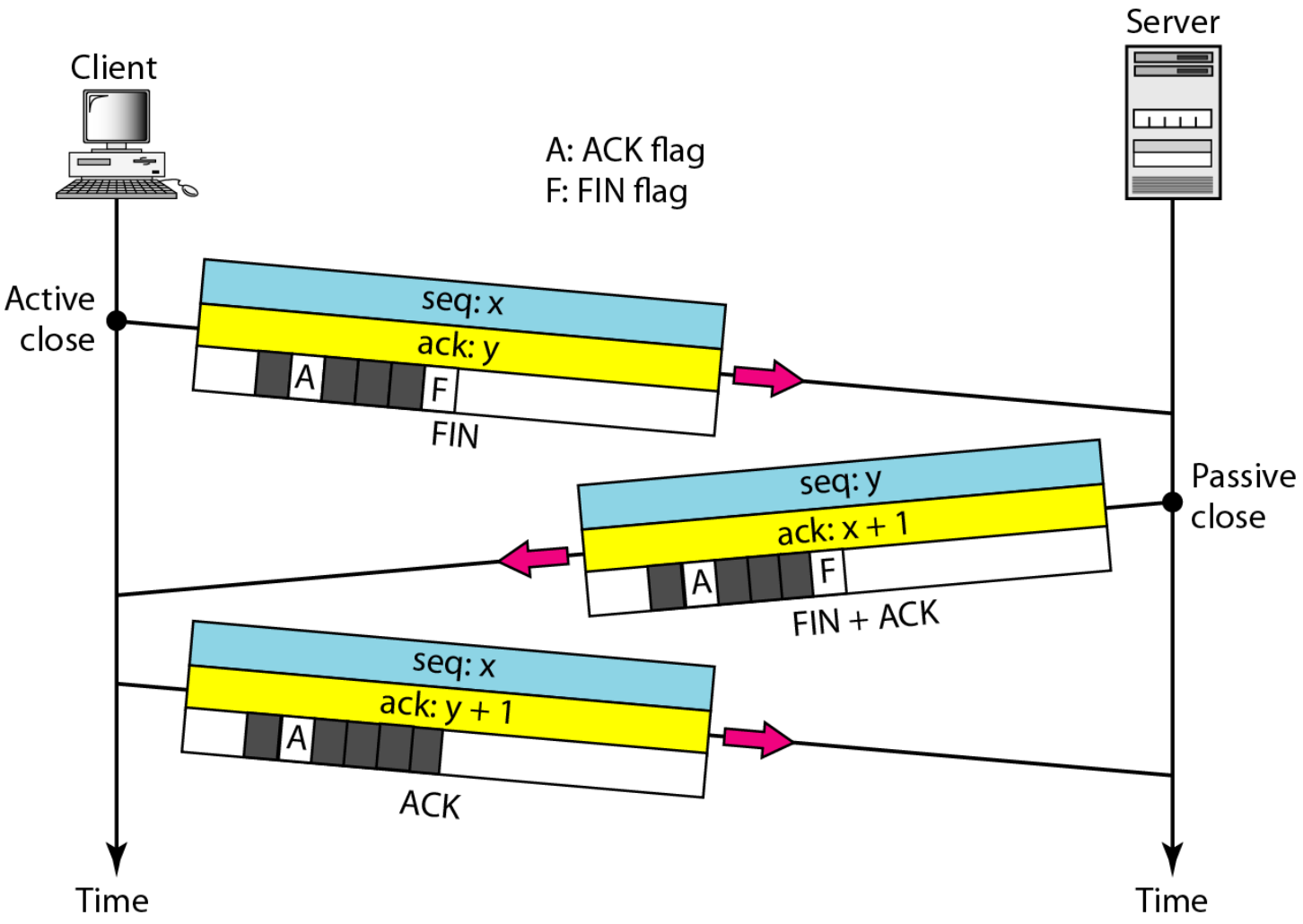
Data transfer



Connection termination using three-way handshaking



Connection termination using three-way handshaking



Thank you