Introduction to Transport Layer

Computer Networks(CS31204)

Prof. Sudip Misra

Department of Computer Science and Engineering

Indian Institute of Technology Kharagpur

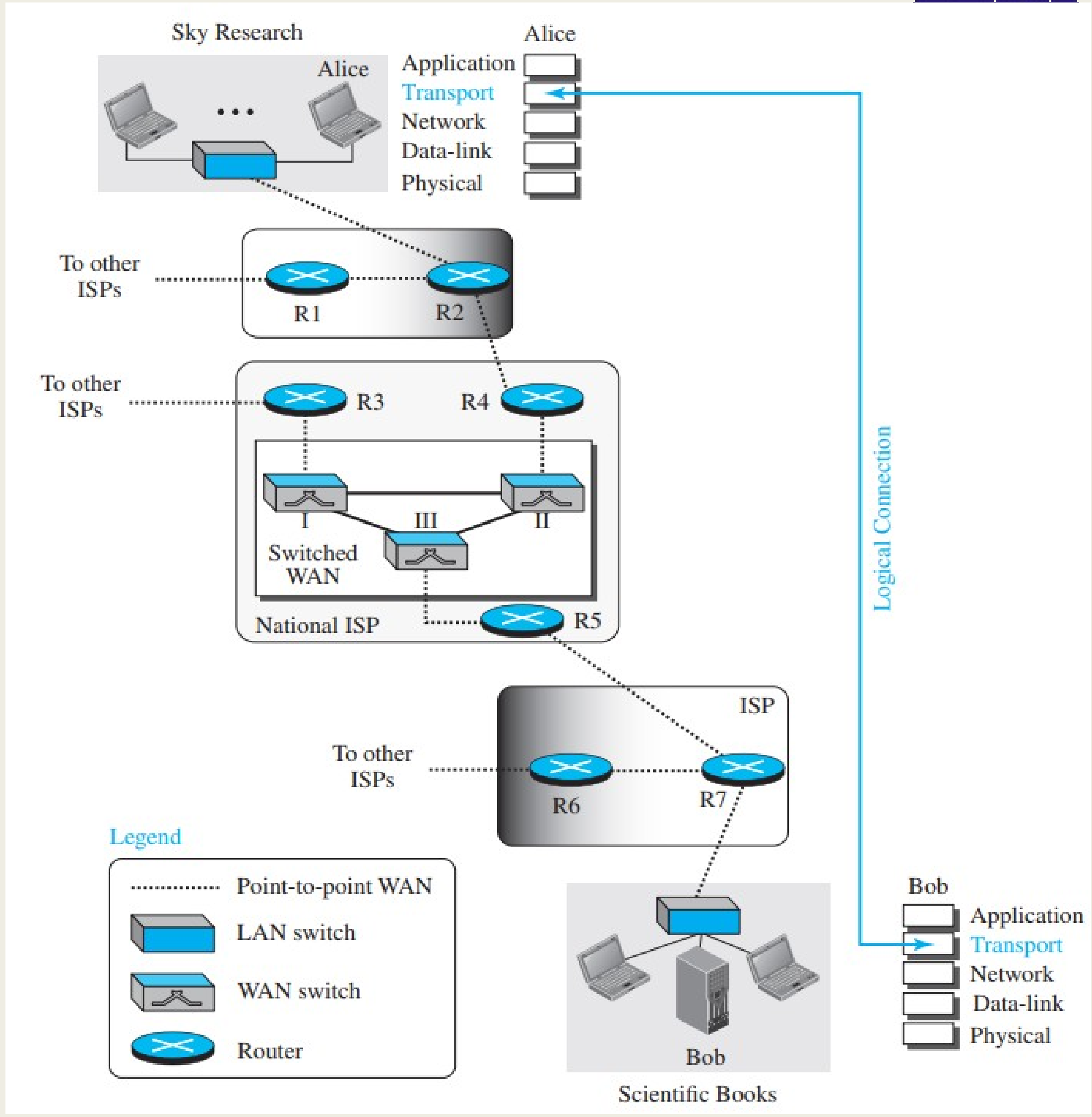
Email: smisra@sit.iitkgp.ernet.in

Website: http://cse.iitkgp.ac.in/~smisra/

Research Lab: cse.iitkgp.ac.in/~smisra/swan/

Introduction

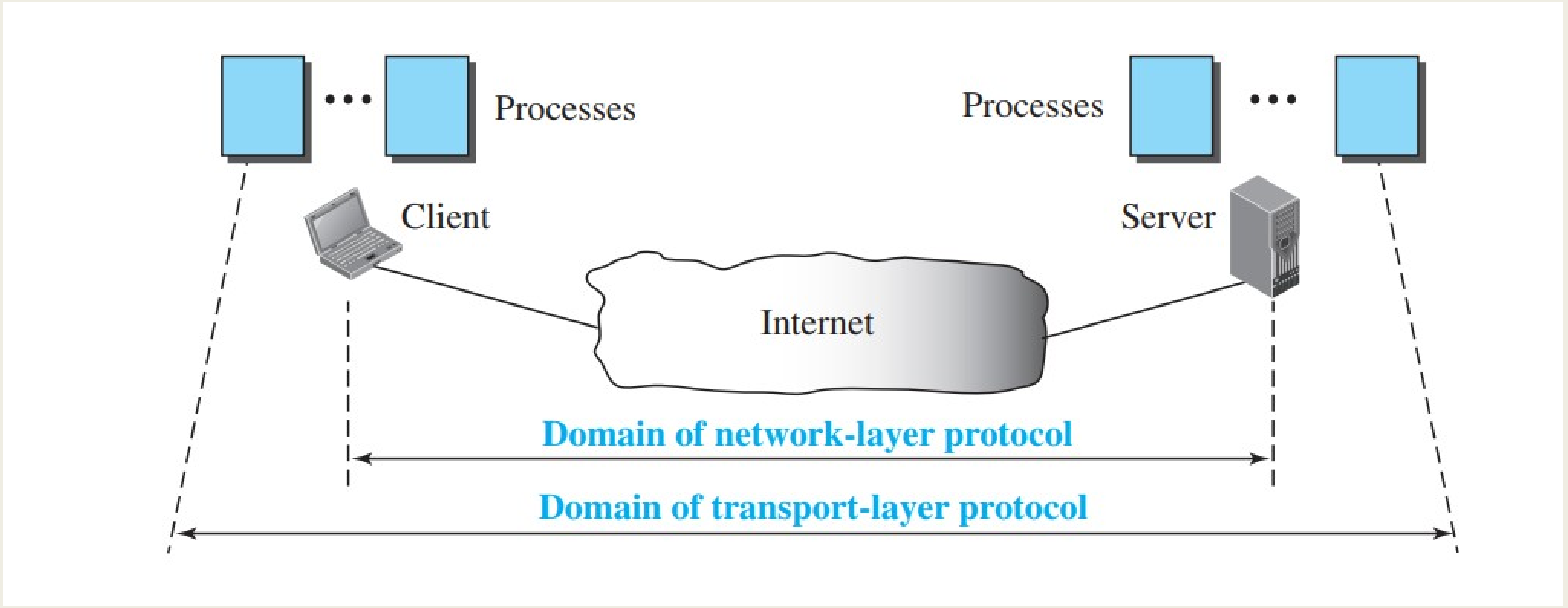
- Provides a process-to-process communication between two application layers, one at the local host and the other at the remote host.
- Communication is provided using a logical connection.
- Two application layers, which can be located in different parts of the globe, assume that there is an imaginary direct connection through which they can send and receive messages.



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Process to Process Communication

- A transport-layer protocol is responsible for delivery of the message to the appropriate process.


Processes
Processes
Client
Server
Internet
Domain of network-layer protocol
Domain of transport-layer protocol

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Addressing

- A process on the local host, called a *client,* needs services from a process usually on the remote host, called a *server.*
- A remote computer can run several server programs at the same time.
- The local host and the remote host are defined using IP addresses.
- To define the processes, we need an identifiers, called ***port numbers.***
- In the TCP/IP protocol suite, the port numbers are integers between 0 and 65,535 (16 bits).

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

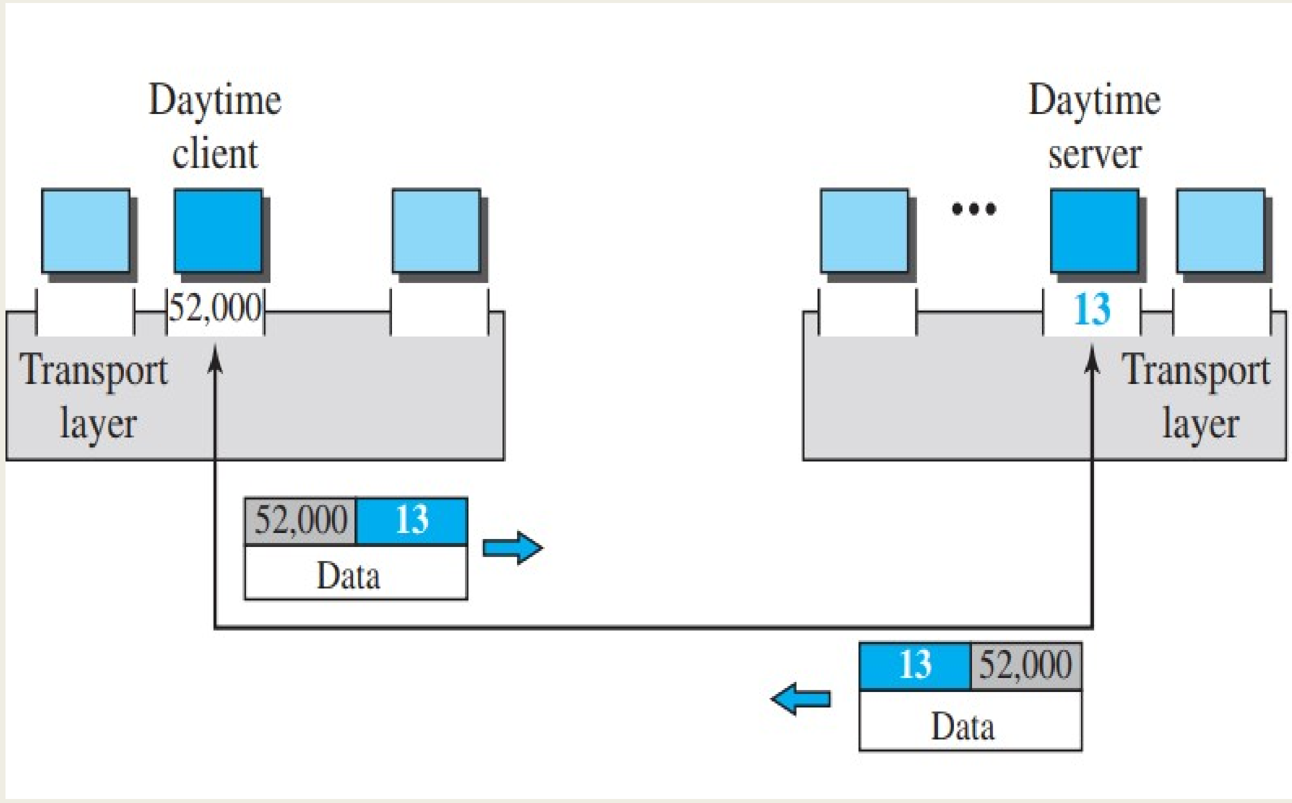
Port Numbers

Ephemeral Port number

- The client program defines itself with a port number, called the **ephemeral port number.**
- The word ephemeral means “short-lived” and is used because the life of a client is normally short.

Well Known Port number

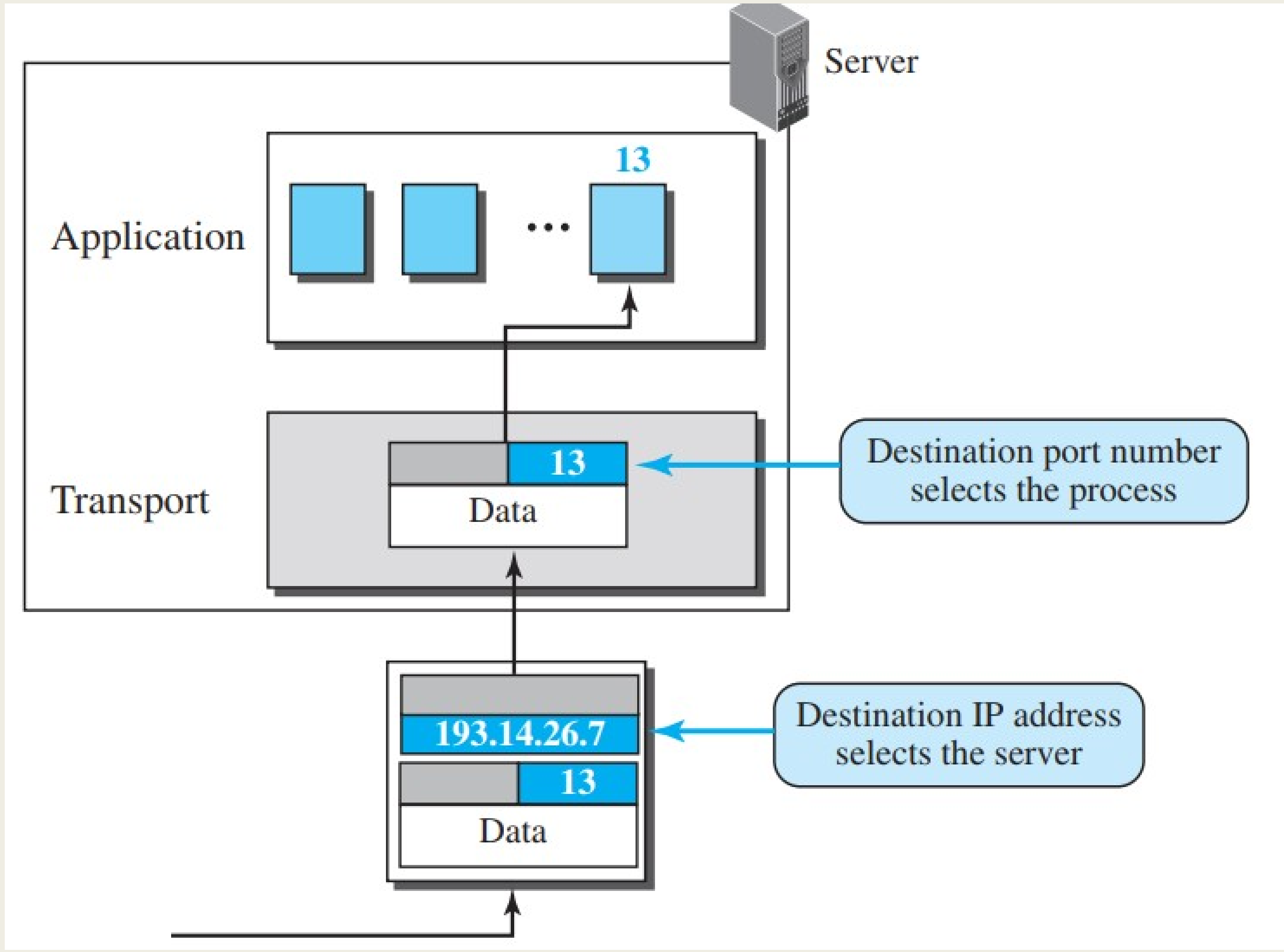
- TCP/IP has decided to use universal port numbers for servers; these are called **well-known port numbers.**



Source: B. A. Forouzan, “ Data Communications and Networking ,” *McGraw-Hill Forouzan Networking Series*,5E.

Ip Addresses Versus Port Numbers

- The destination IP address defines the host among the different hosts in the world.
- After the host has been selected, the port number defines one of the processes on this particular host.



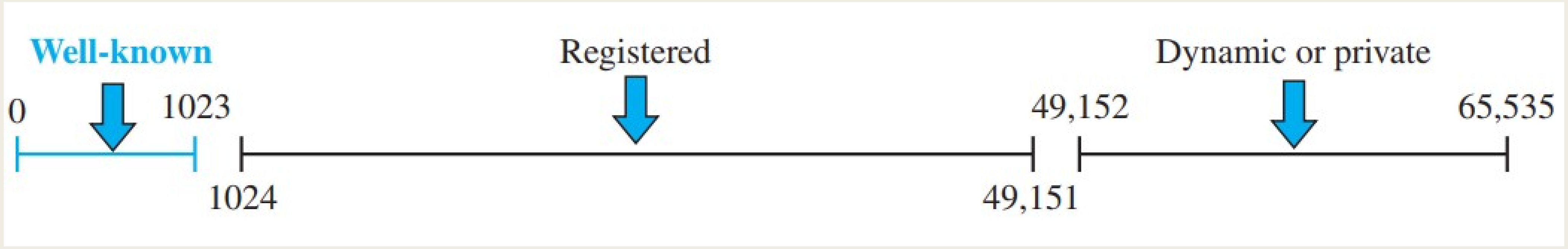
Server
Application
13
Transport
13
Data
Destination port number selects the process
193.14.26.7
13
Data
Destination IP address selects the server

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

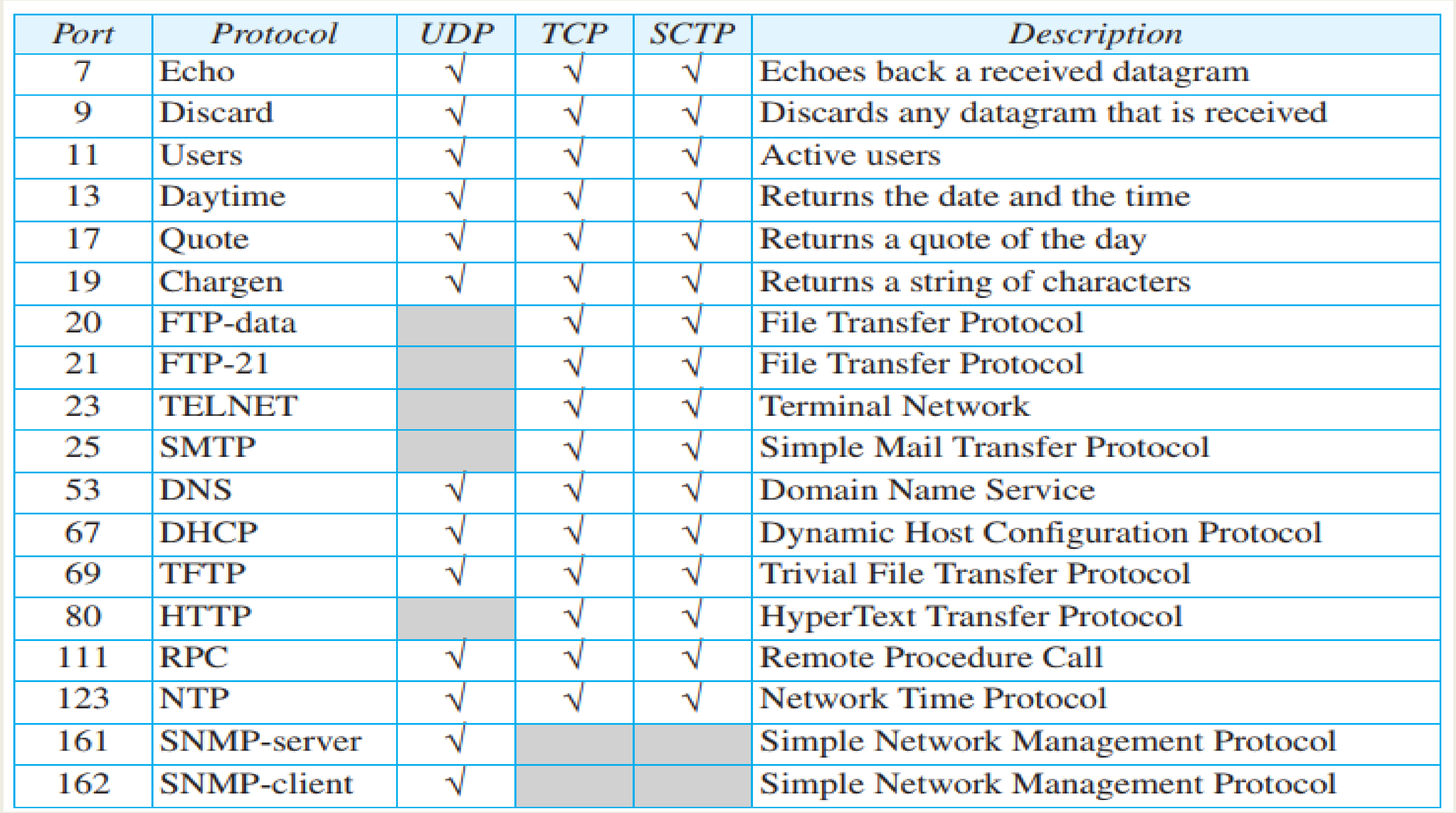
ICANN Ranges

ICANN has divided the port numbers into three ranges: well-known, registered, and dynamic (or private).

- **Well-known ports.** The ports ranging from 0 to 1023 are assigned and controlled by ICANN. These are the well-known ports.
- **Registered ports.** The ports ranging from 1024 to 49,151 are not assigned or controlled by ICANN. They can only be registered with ICANN to prevent duplication.
- **Dynamic ports.** The ports ranging from 49,152 to 65,535 are neither controlled nor registered. They can be used as temporary or private port numbers.



Some Well Known Port Numbers



Port	*Protocol*	*UDP*	*TCP*	*SCTP*	*Description*
7	Echo	√	√	√	Echoes back a received datagram
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
21	FTP-21		√	√	File Transfer Protocol
23	TELNET		√	√	Terminal Network
25	SMTP		√	√	Simple Mail Transfer Protocol
53	DNS	√	√	√	Domain Name Service
67	DHCP	√	√	√	Dynamic Host Configuration Protocol
69	TFTP	√	√	√	Trivial File Transfer Protocol
80	HTTP		√	√	HyperText Transfer Protocol
111	RPC	√	√	√	Remote Procedure Call
123	NTP	√	√	√	Network Time Protocol
161	SNMP-server	√			Simple Network Management Protocol
162	SNMP-client	√			Simple Network Management Protocol

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Stream and Segment

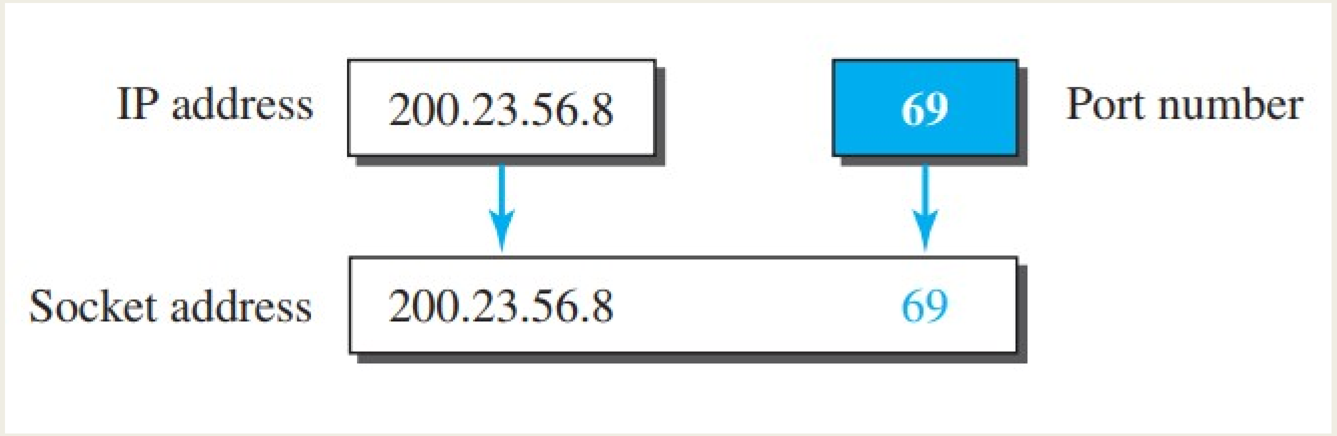
- Data viewed as a byte stream, i.e., a sequence of bytes
- Segment – the unit of transfer between TCP s/w on two machines
- The stream of bytes is divided into segments, each segment is given a TCP header, and transmitted
- Usually 1 segment is encapsulated in 1 IP datagram
- Segments may not contain any data
- Ex – segments used to establish/terminate connections, send acks etc.
- Sequence Number – used to specify position within the stream
- Each TCP segment will contain a 32-bit sequence no. to identify its position in the stream

Maximum Segment Size

- Maximum size of a segment (excluding header)
- Ideally, should be (Minimum MTU of any link from the source to the destination –TCP header size – IP header size)
- Avoids fragmentation and reassembly at IP layer, which is costly
- Hard to know minimum MTU of links end-to-end, though can be known in some cases, for ex, if all links are Ethernet
- Default MSS = 536
- All IP based networks must support a MTU of 576
- Can be changed during connection establishment time using TCP options
- Cannot be changed once connection is established

Socket Addresses

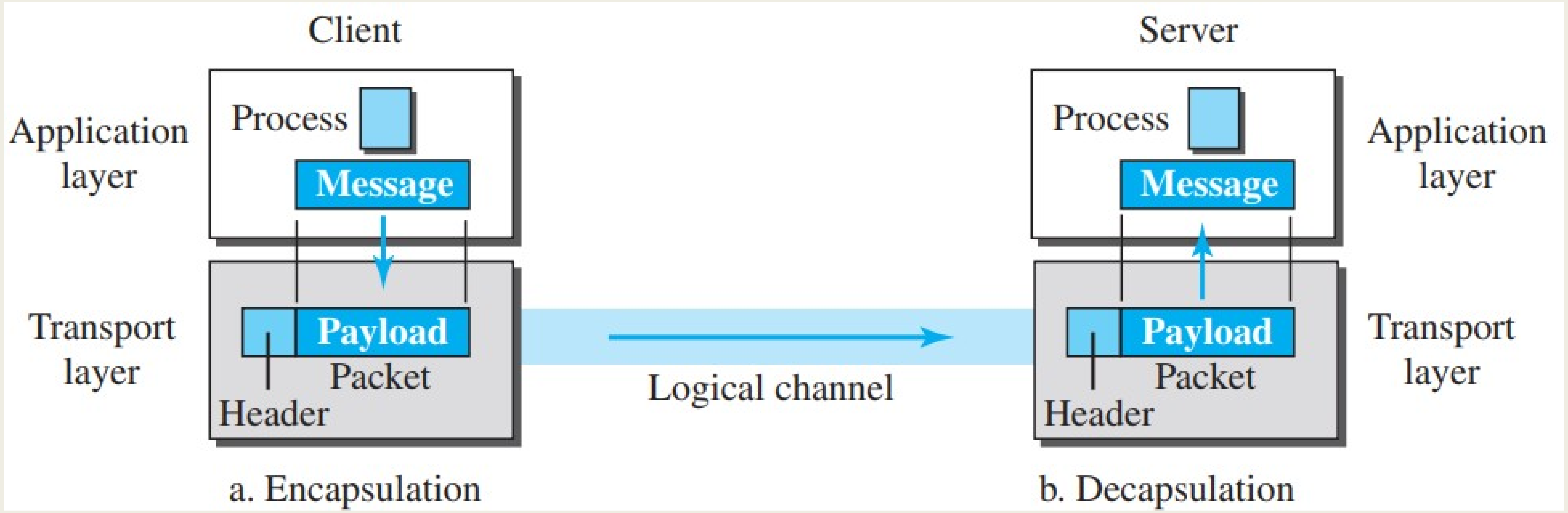
- A transport-layer protocol in the TCP suite needs both the IP address and the port number, at each end, to make a connection.
- The combination of an IP address and a port number is called a ***socket address.***
- Socket address defines the process uniquely



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

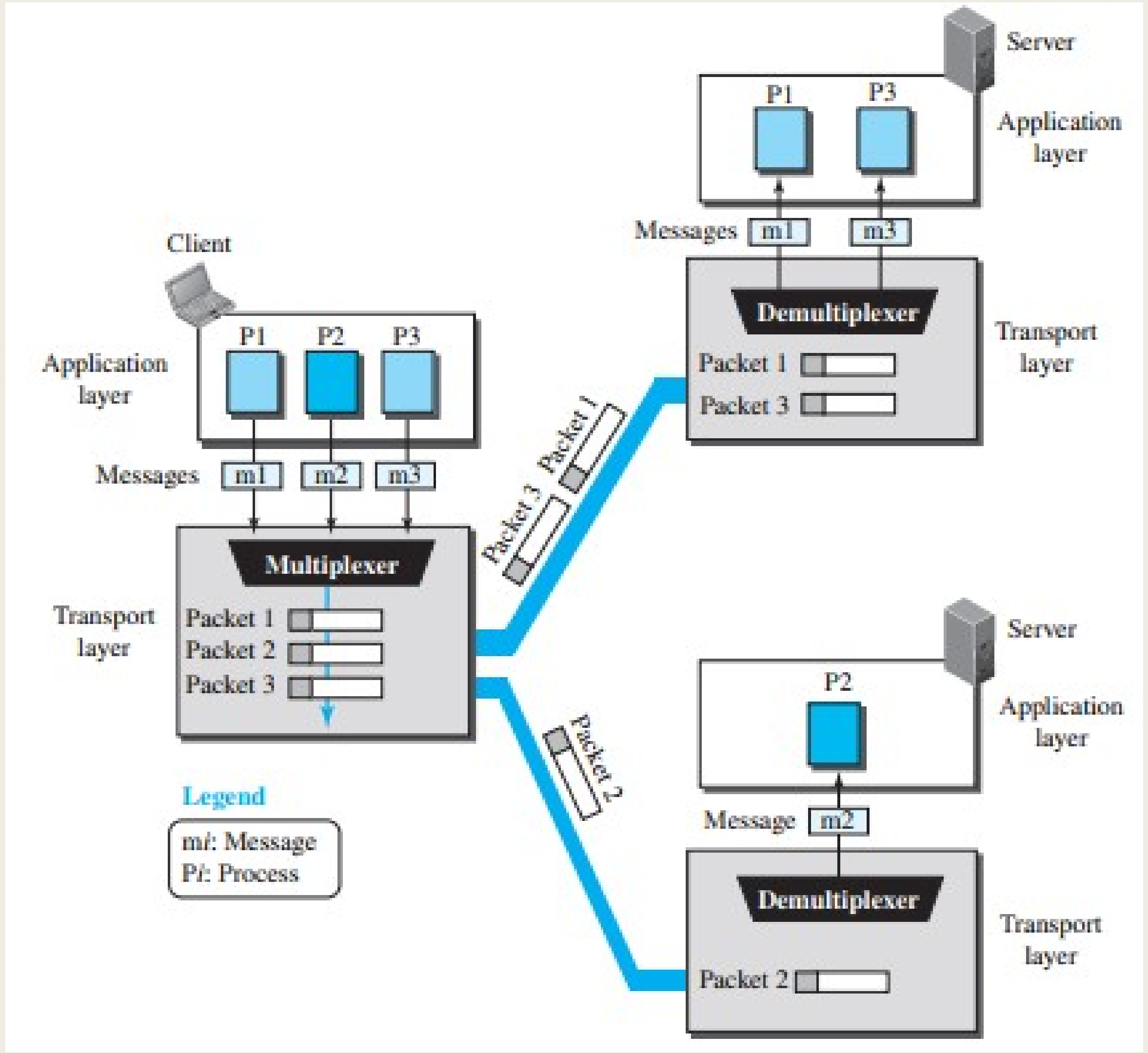
Encapsulation and Decapsulation

- Encapsulation happens at the sender site.
- The transport layer receives the data and adds the transport-layer header. The packets at the transport layer in the Internet are called *segments.*
- Decapsulation happens at the receiver site.
- When the message arrives at the destination transport layer, the header is dropped and the transport layer delivers the message to the process running at the application layer.



Multiplexing and Demultiplexing

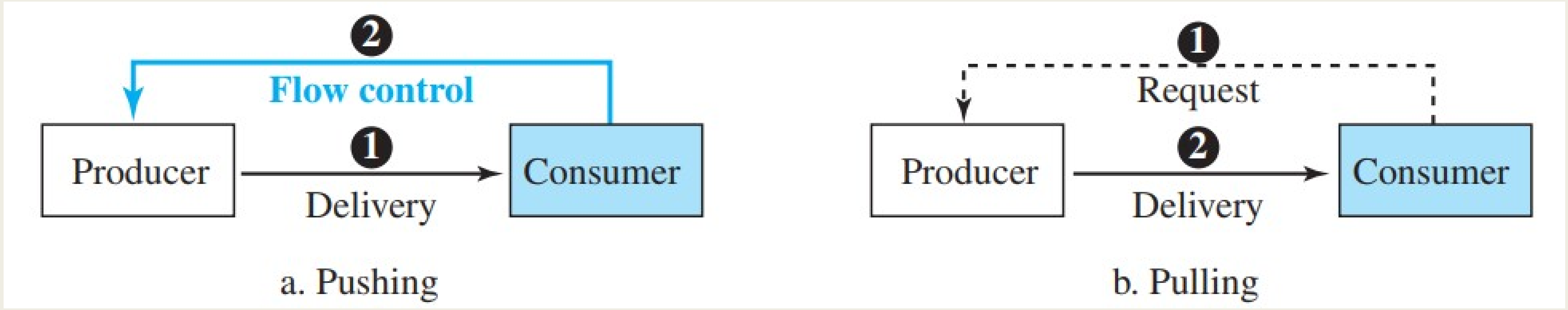
- Whenever an entity accepts items from more than one source, this is referred to as **multiplexing** (many to one).
- Whenever an entity delivers items to more than one source, this is referred to as **demultiplexing** (one to many).
- The transport layer at the source performs multiplexing; the transport layer at the destination performs demultiplexing.



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

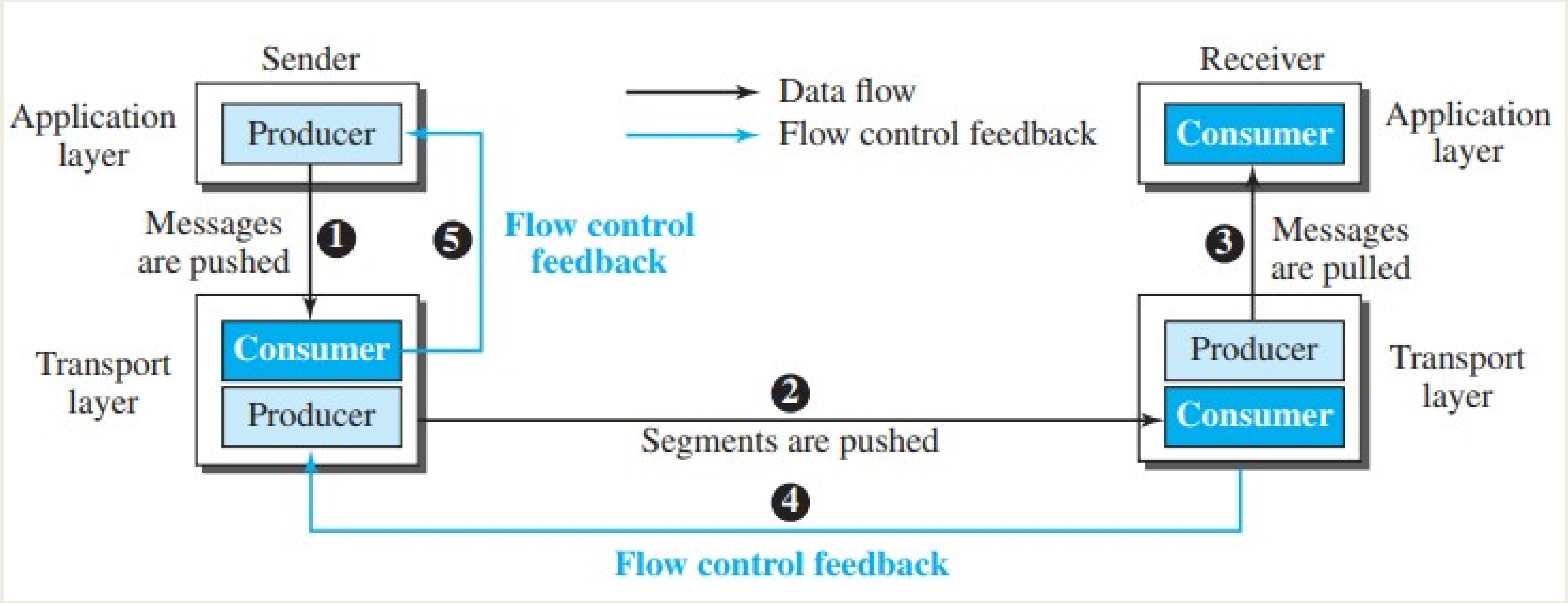
Pushing and Pulling

- Delivery of items from a producer to a consumer can occur in one of two ways: pushing or pulling.
- If the sender delivers items whenever they are produced- without a prior request from the consumer- the delivery is referred to as pushing.
- If the producer delivers the items after the consumer has requested them, the delivery is referred to as pulling.



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Flow Control



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Error Control

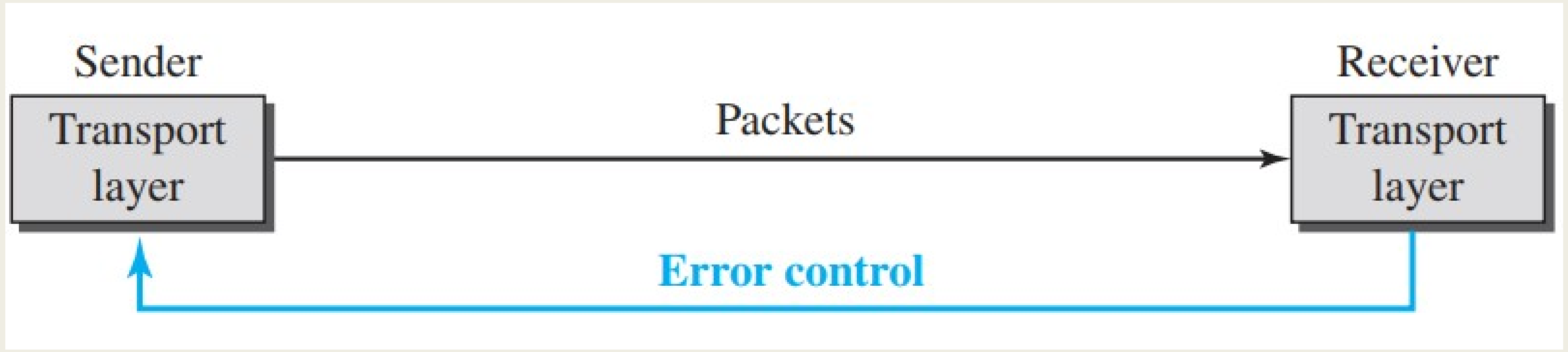
Error control at the transport layer is responsible for

1. Detecting and discarding corrupted packets.

2. Keeping track of lost and discarded packets and resending them.

3. Recognizing duplicate packets and discarding them.

4. Buffering out-of-order packets until the missing packets arrive.



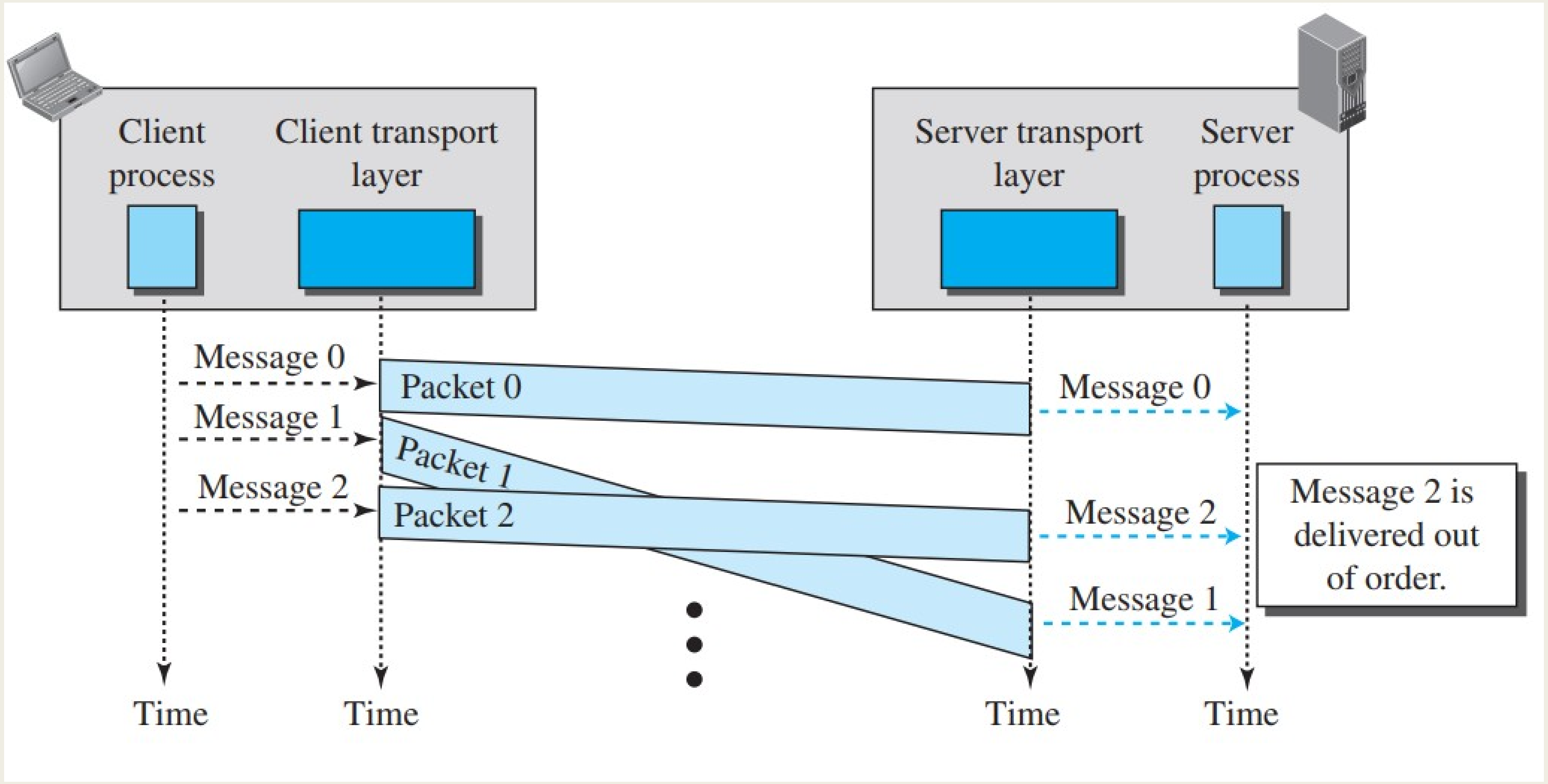
Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Congestion Control

- 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 that control the congestion and keep the load below the capacity.

Connectionless Services



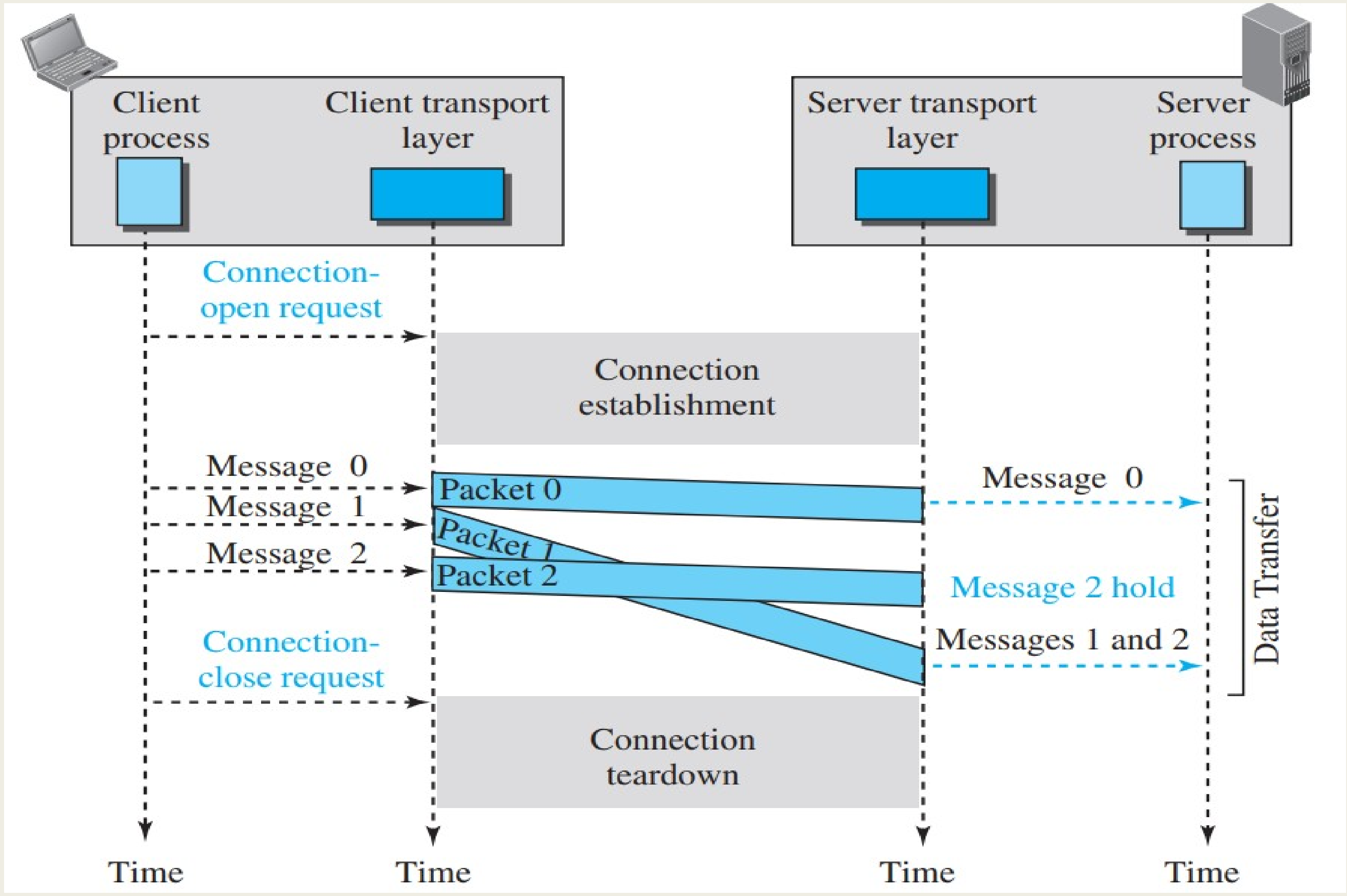
Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Connectionless Services

- In a connectionless service, the source process (application program) needs to divide its message into chunks of data of the size acceptable by the transport layer and deliver them to the transport layer one by one.
- The transport layer treats each chunk as a single unit without any relation between the chunks.
- When a chunk arrives from the application layer, the transport layer encapsulates it in a packet and sends it.
- To show the independency of packets, assume that a client process has three chunks of messages to send
  to a server process.
- The chunks are handed over to the connectionless transport protocol in order.
- However, since there is no dependency between the packets at the transport layer, the packets may arrive out of order at the destination and will be delivered out of order to the server process.

Connection Oriented Services

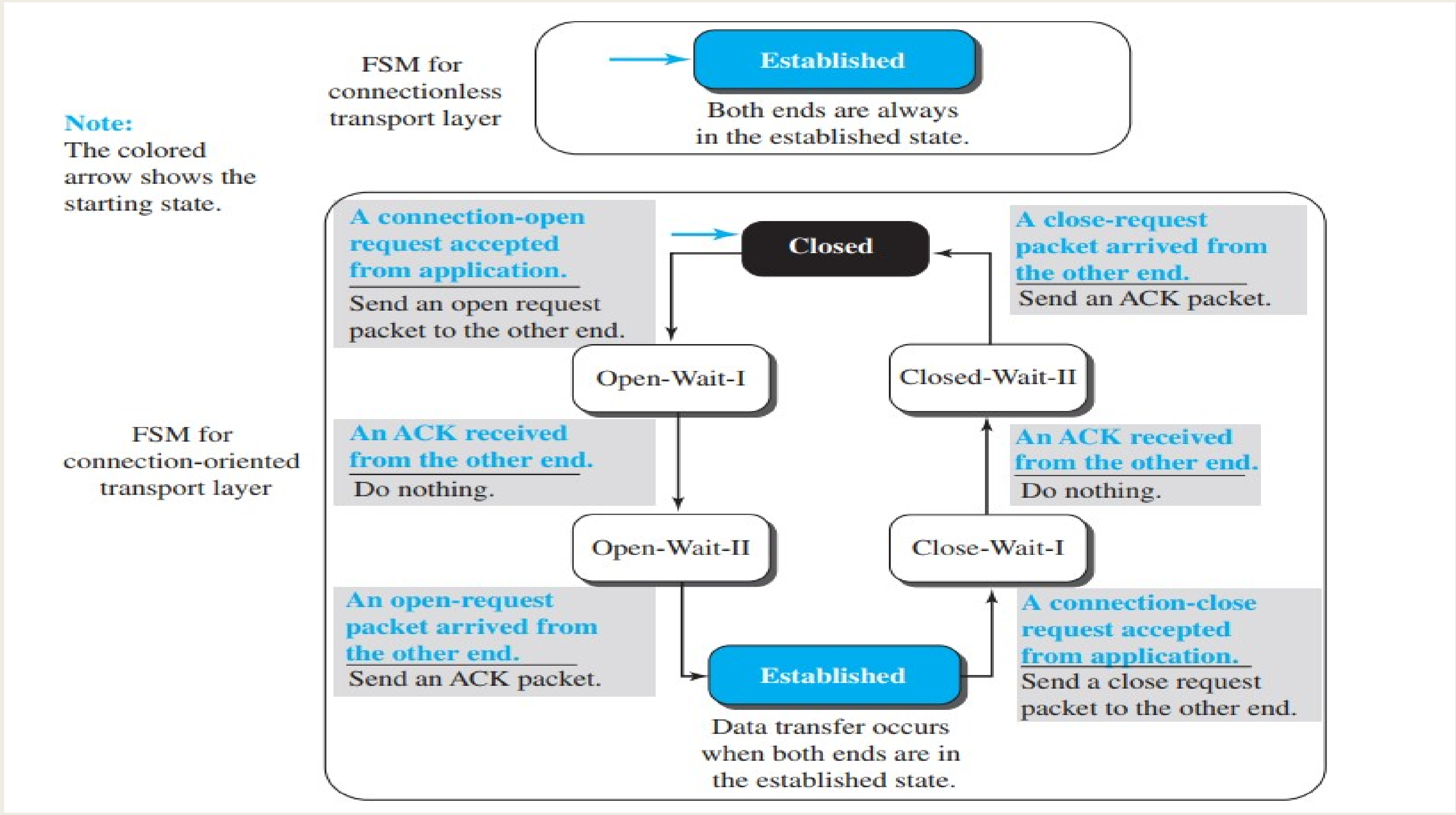
- In a connection-oriented service, the client and the server first need to establish a logical connection between themselves.
- The data exchange can only happen after the connection establishment.
- After data exchange, the connection needs to be torn down



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Connectionless and connection-oriented service represented as FSM

- FSM in connectionless transport layer with only one state: the established state.
- The machine on each end (client and server) is always in the established state, ready to send and receive transport-layer packets.
- An FSM in a connection-oriented transport layer, on the other hand, needs to go through three states before reaching the established state.

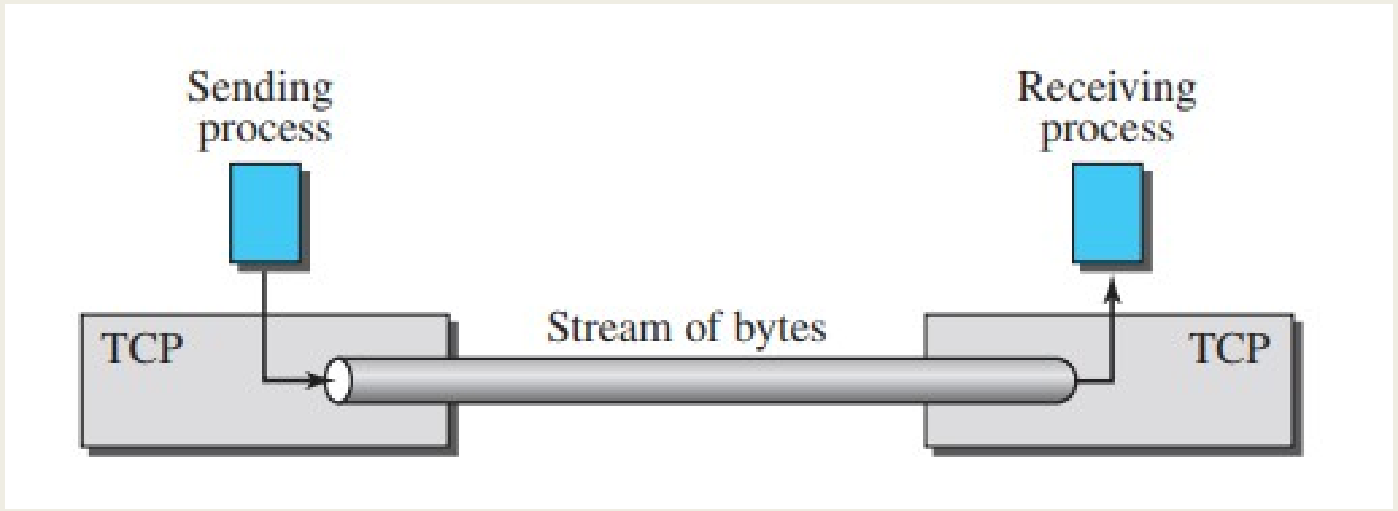


Note:
The colored arrow shows the starting state.
FSM for connectionless transport layer
Established
Both ends are always in the established state.
FSM for connection-oriented transport layer
A connection-open request accepted from application.
Send an open request packet to the other end.
Closed
A close-request packet arrived from the other end.
Send an ACK packet.
Open-Wait-I
Closed-Wait-II
An ACK received from the other end.
Do nothing.
An ACK received from the other end.
Do nothing.
Open-Wait-II
Close-Wait-I
An open-request packet arrived from the other end.
Send an ACK packet.
Established
A connection-close request accepted from application.
Send a close request packet to the other end.
Data transfer occurs when both ends are in the established state.

Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Stream Delivery Service

- Deliver data as a stream of bytes and allows the receiving process to obtain data as a stream of bytes.
- The sending process produces (writes to) the stream and the receiving process consumes (reads from) it.



Source: B. A. Forouzan, " Data Communications and Networking ," *McGraw-Hill Forouzan Networking Series*,5E.

Thank You!!!