CPE 490: Information Systems Engineering I: Computer Networking

Chap. 6 - The Transport Layer

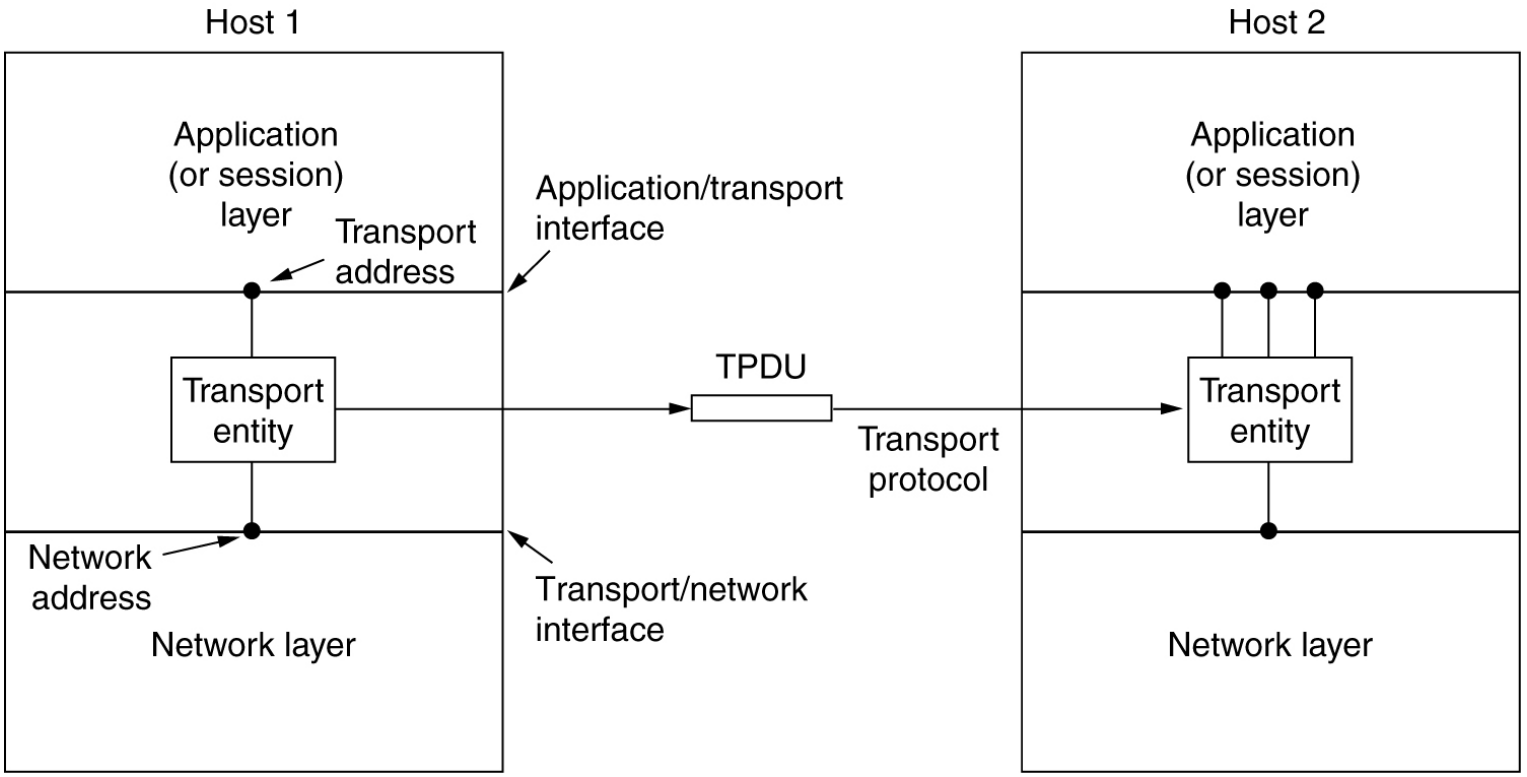
Professor Du

Department of Electrical and Computer Engineering
Stevens Institute of Technology
Email: xdu16@stevens.edu

The Transport Service

- Services Provided to the Upper Layers
- Transport Service Primitives
- Berkeley Sockets

Services Provided to the Upper Layers



The network, transport, and application layers.

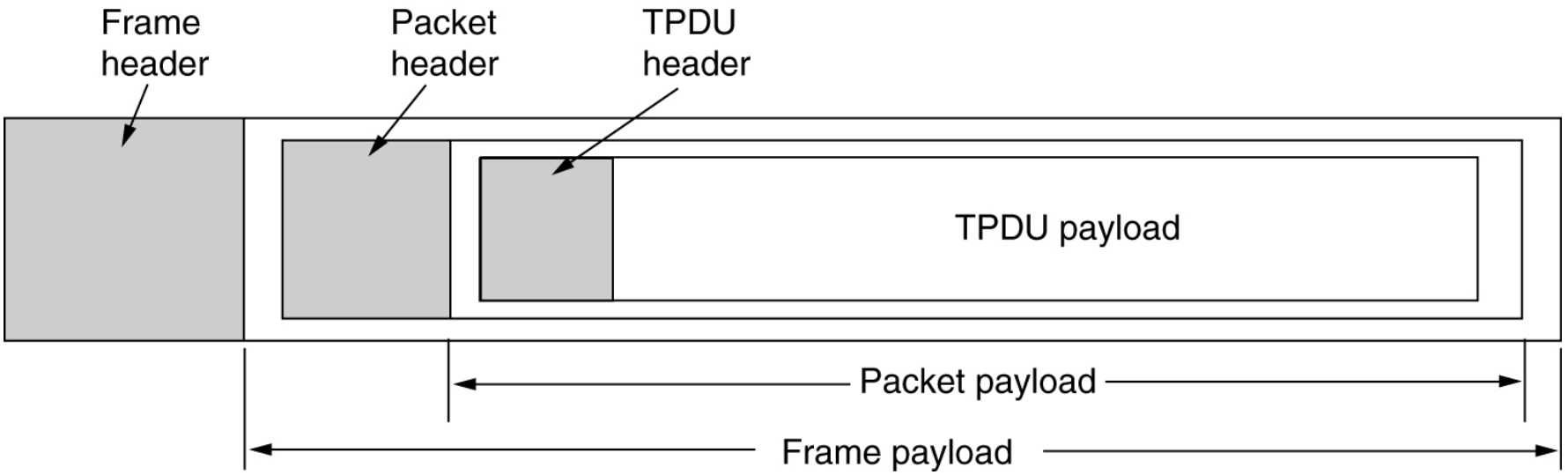
Transport Service Primitives

Primitive	Packet sent	Meaning
LISTEN	(none)	Block until some process tries to connect
CONNECT	CONNECTION REQ.	Actively attempt to establish a connection
SEND	DATA	Send information
RECEIVE	(none)	Block until a DATA packet arrives
DISCONNECT	DISCONNECTION REQ.	This side wants to release the connection

The primitives for a simple transport service.

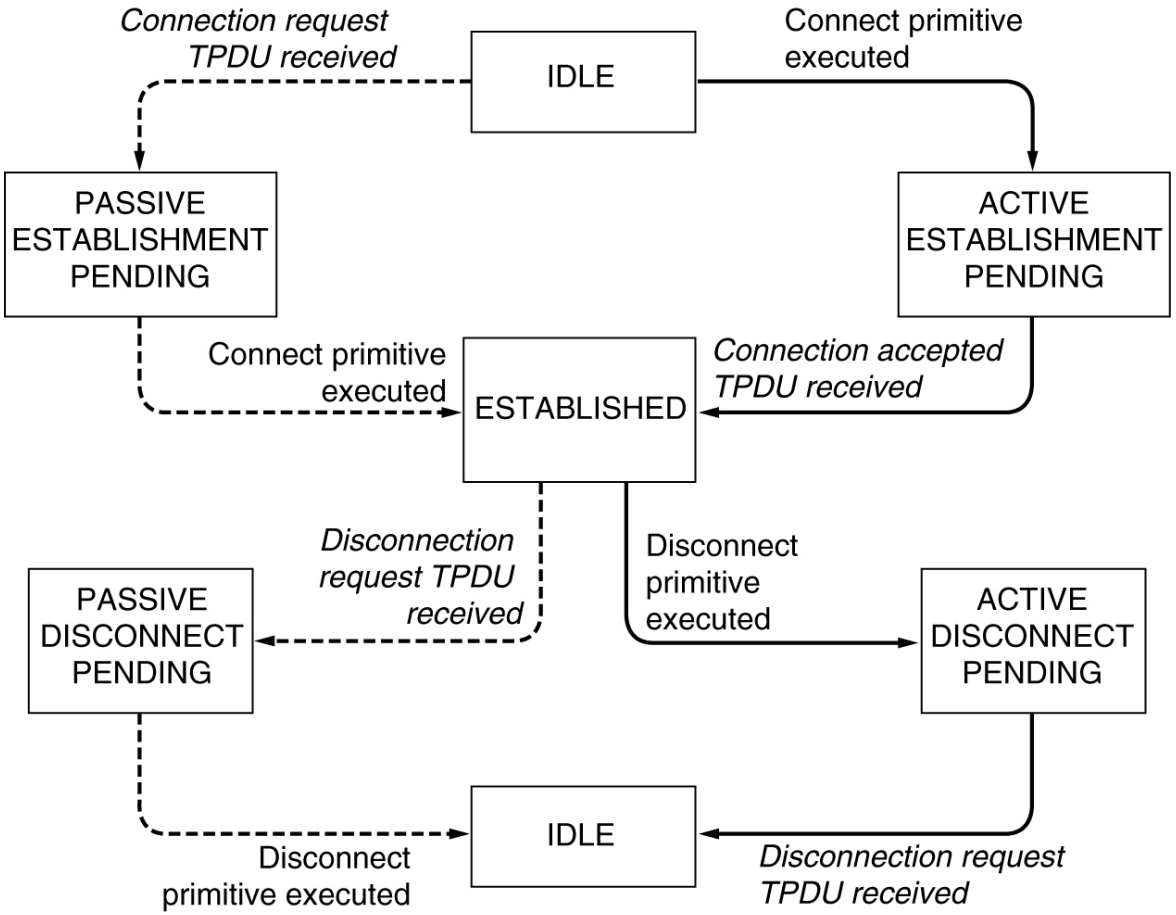
Transport Service Primitives (2)

- ✓ TPDU – Transport Protocol Data Unit
- ✓ Packet header – The network layer
- ✓ Frame header – The data link layer



The nesting of TPDUs, packets, and frames.

Transport Service Primitives (3)



A state diagram for a simple connection management scheme. Transitions labeled in italics are caused by packet arrivals. The solid lines show the client's state sequence. The dashed lines show the server's state sequence.

Berkeley Sockets

- ✓ The socket primitives used in Berkeley UNIX for TCP.

Primitive	Meaning
SOCKET	Create a new communication end point
BIND	Attach a local address to a socket
LISTEN	Announce willingness to accept connections; give queue size
ACCEPT	Block the caller until a connection attempt arrives
CONNECT	Actively attempt to establish a connection
SEND	Send some data over the connection
RECEIVE	Receive some data from the connection
CLOSE	Release the connection

The socket primitives for TCP.

Elements of Transport Protocols

- Addressing
- Connection Establishment
- Connection Release
- Flow Control and Buffering
- Multiplexing
- Crash Recovery

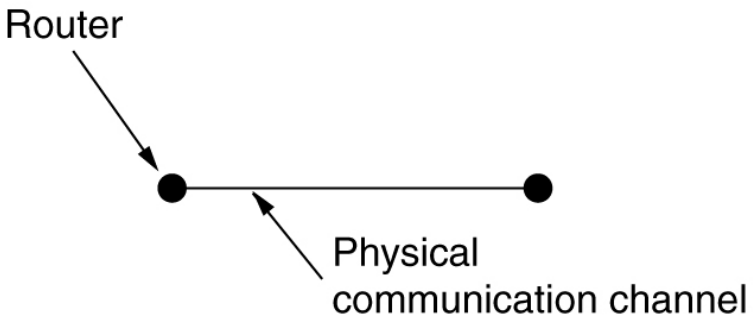
Transport Protocol

(a) Environment of the data link layer.

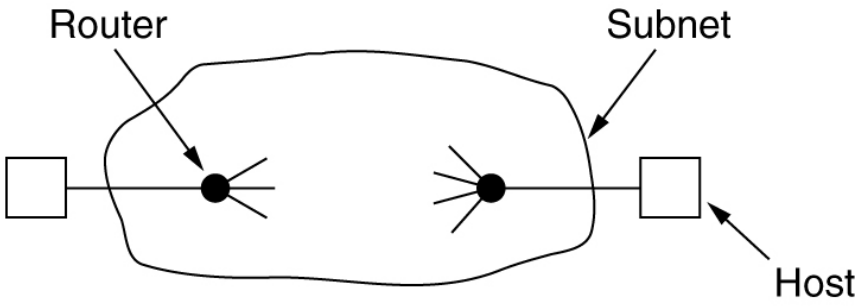
- Point-to-point connection
- The other end is always there

(b) Environment of the transport layer.

- End-to-end connection
- Explicit addressing of the destination is required.
- Connection needs to be set up.



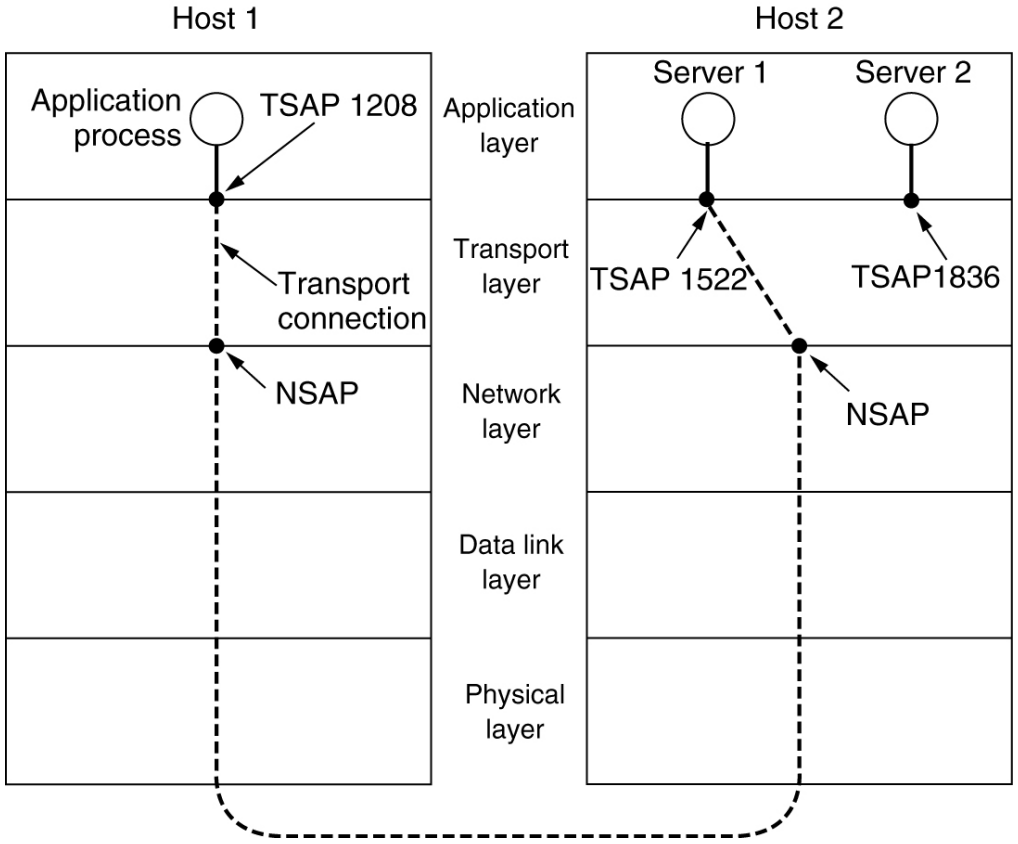
(a)



(b)

Addressing

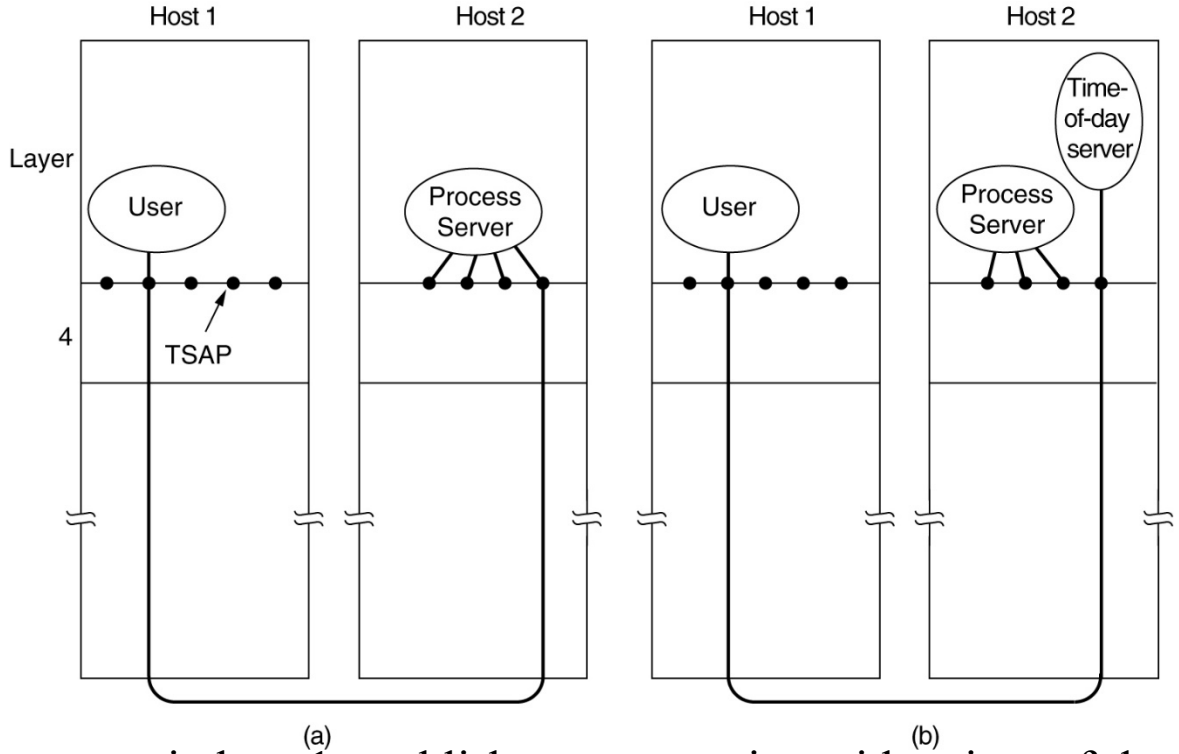
- TSAP (Transport Service Access Point) – Port number
  - specify which remote application process
- NSAP (Network Service Access Point) – IP addresses



TSAPs, NSAPs and transport connections.

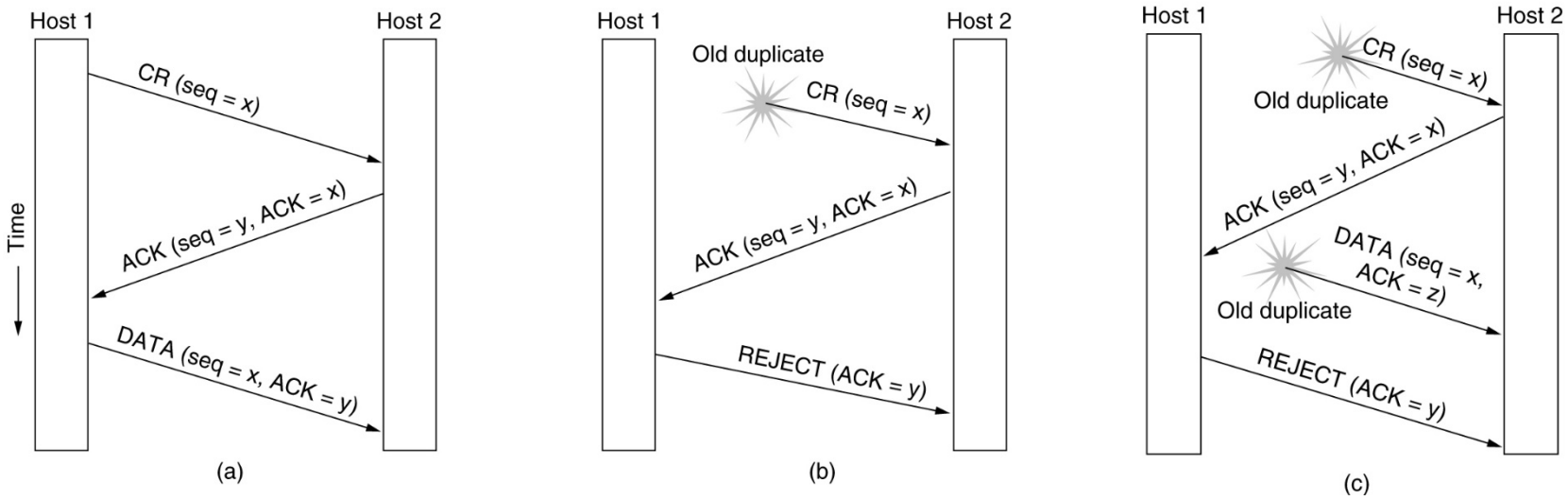
Connection Establishment

- ✓ A special process server listens to a set of ports at the same time.
- ✓ A user sends a CONNECT request with the TSAP.
- ✓ If no server is waiting for the request, the user gets a connection with the process server.
- ✓ Then the process server calls the requested (time-of-day) server, allowing it to inherit the existing connection with the user.



How a user process in host 1 establishes a connection with a time-of-day server in host 2.

Connection Establishment (2)



Three protocol scenarios for establishing a connection using a three-way handshake. CR denotes CONNECTION REQUEST.

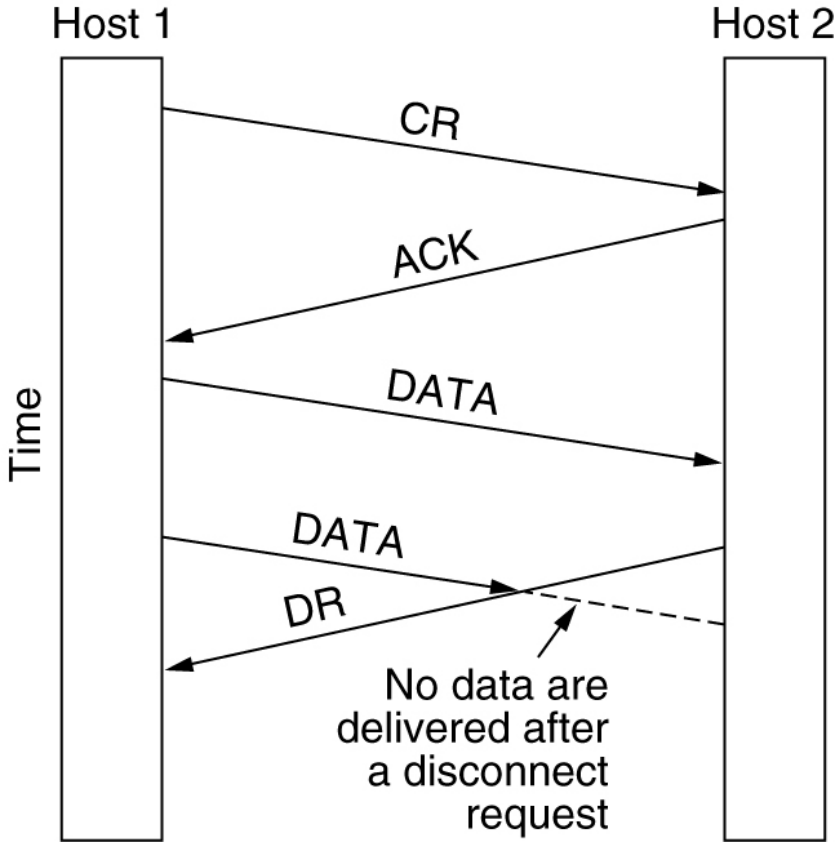
(a) Normal operation,

(b) Old CONNECTION REQUEST appearing out of nowhere.

- Host 2 replies an ACK to verify the connection request.
- Host 2 realizes that the CR was an old message when receiving the REJECT.

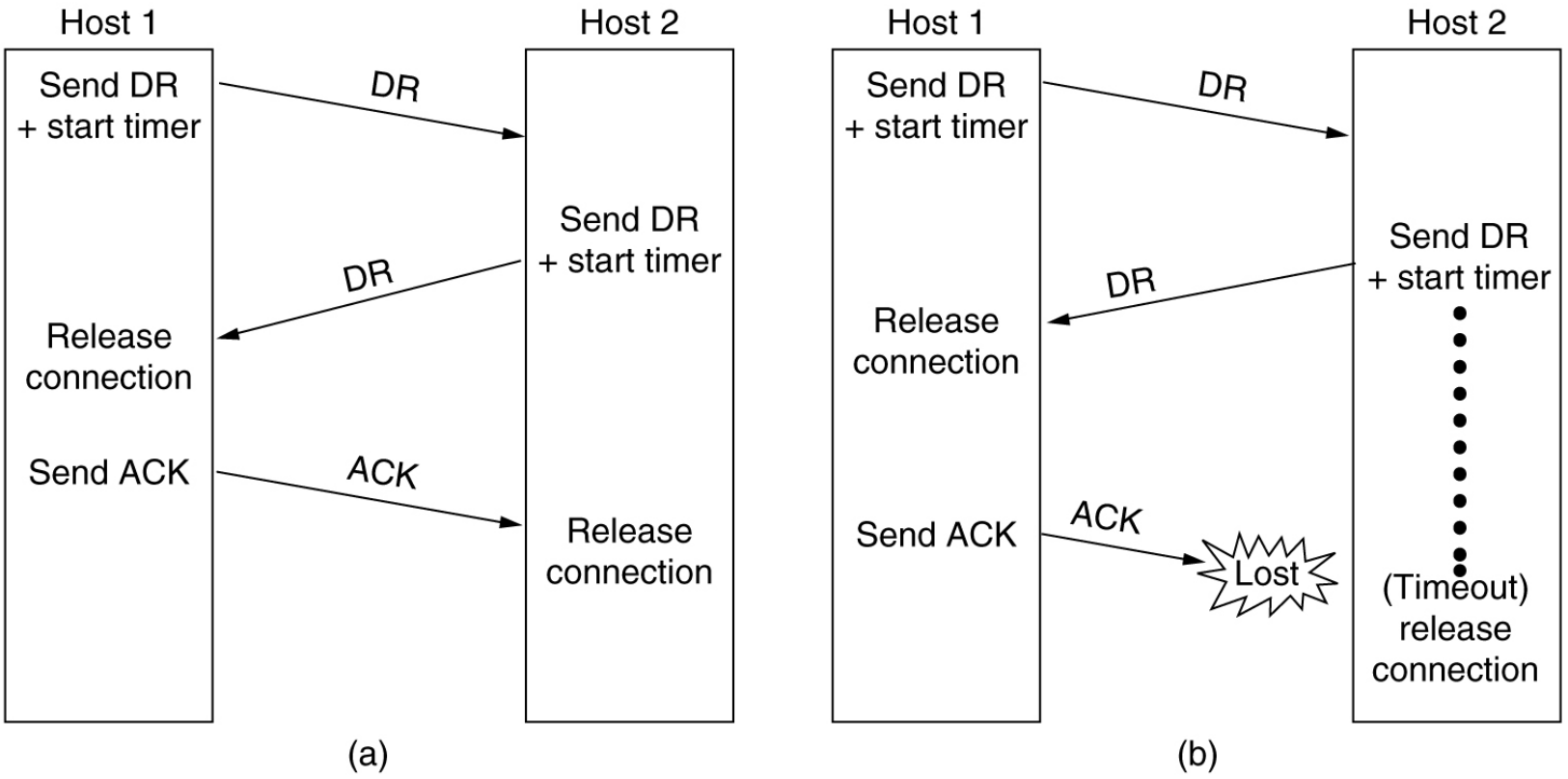
(c) Duplicate CONNECTION REQUEST and duplicate ACK.

Connection Release



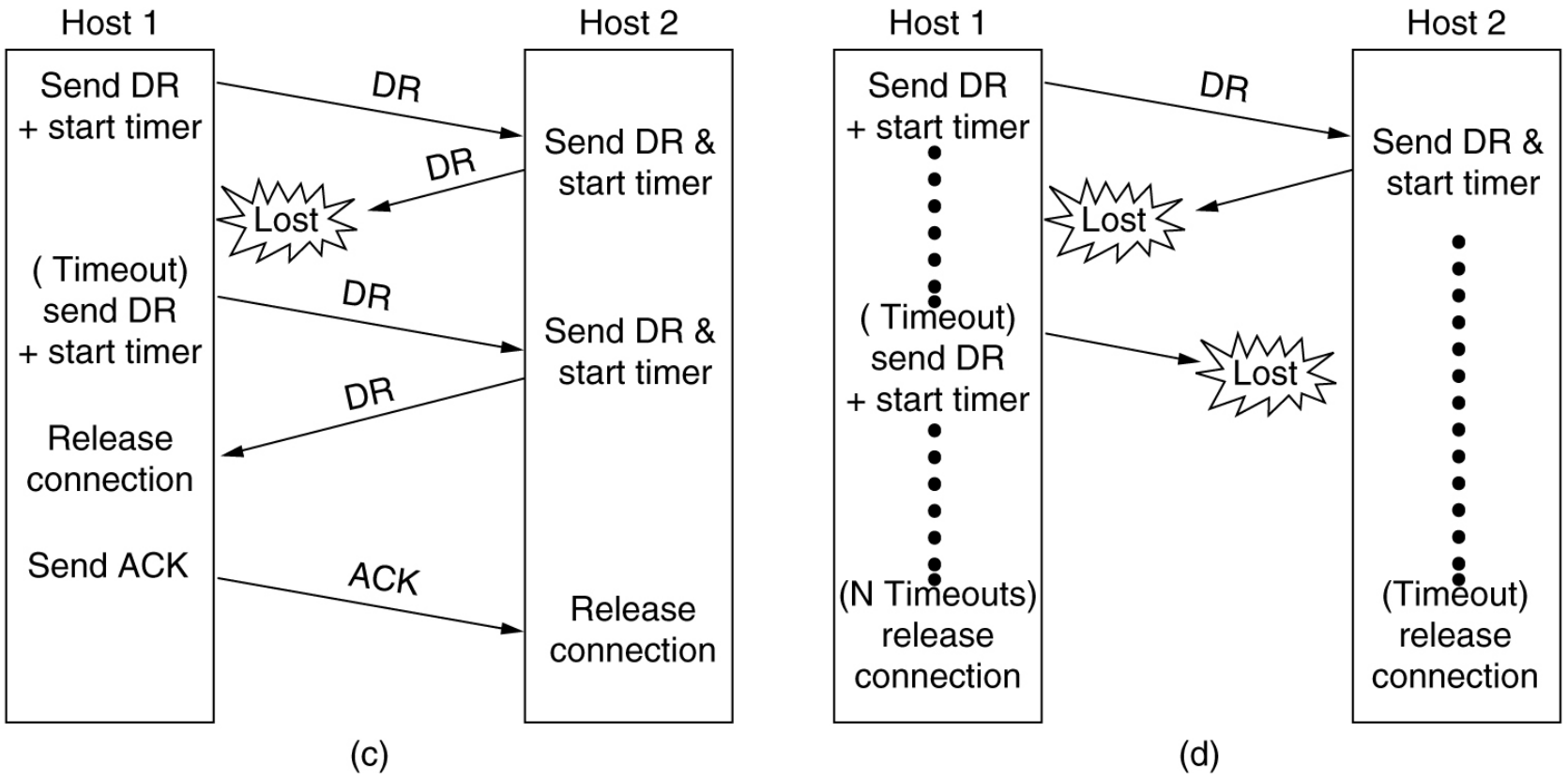
Abrupt disconnection with loss of data.

Connection Release



Four protocol scenarios for releasing a connection. (a) Normal case of a three-way handshake. (b) final ACK lost.

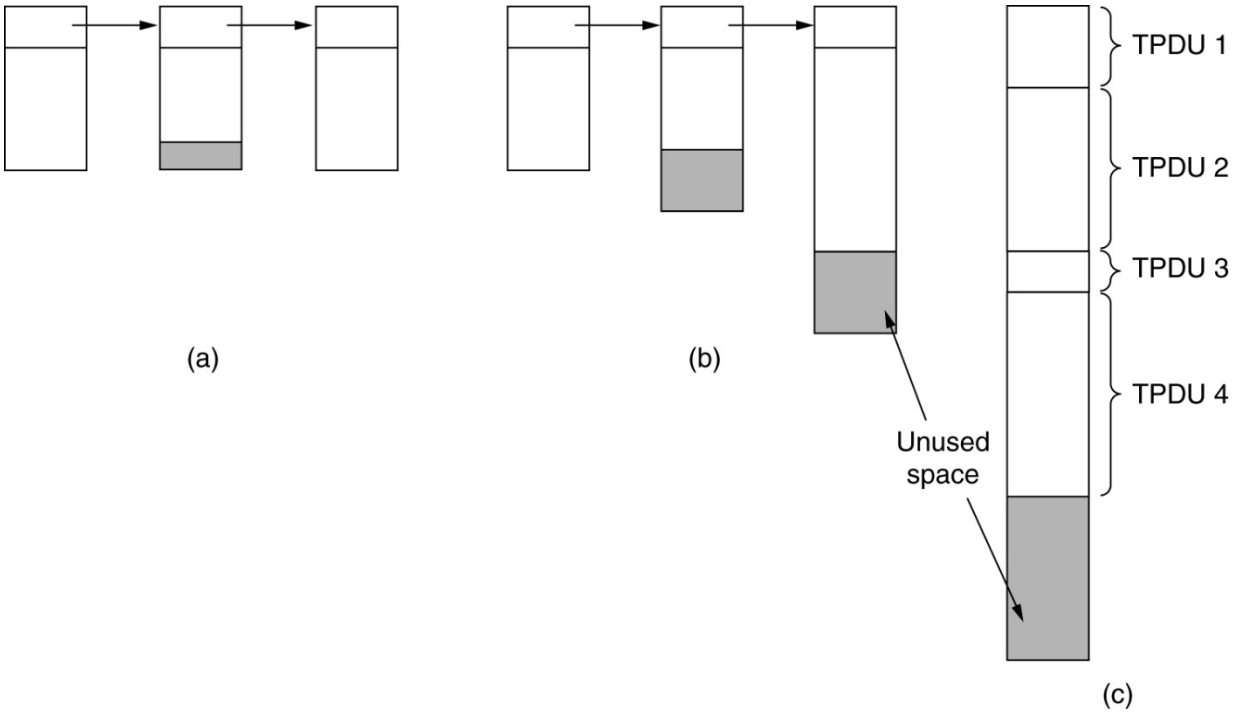
Connection Release



(c) Response lost. (d) Response lost and subsequent DRs lost.

Flow Control and Buffering

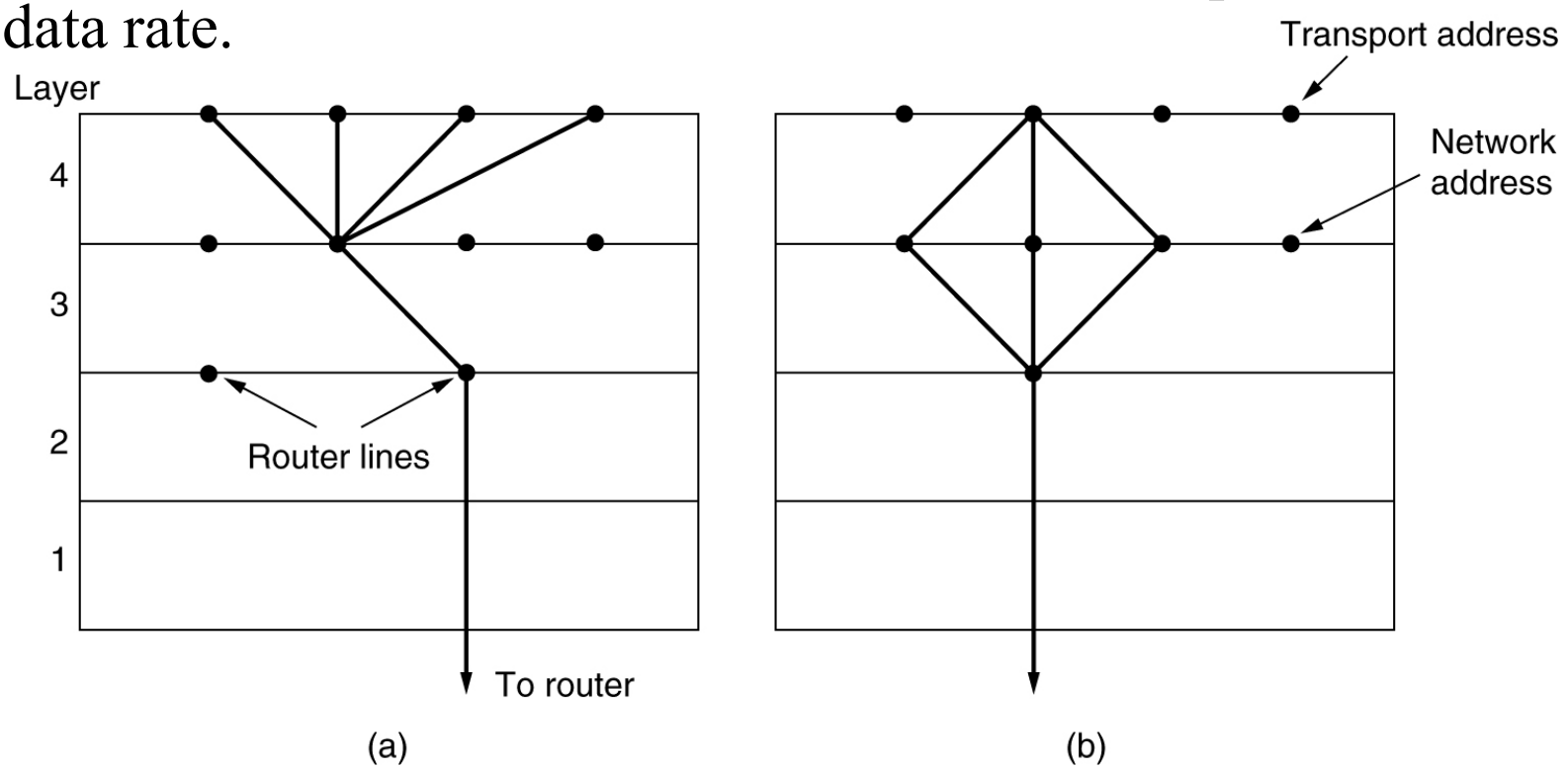
- ✓ Flow control in transport layer has similarity and difference to flow control in data link layer.
- ✓ Difference: a router has relatively few lines, whereas a host may have lots of connections ➔ buffering scheme should be different.



(a) Chained fixed-size buffers. (b) Chained variable-sized buffers.
(c) One large circular buffer per connection.

Multiplexing

- ✓ Upward multiplexing: Several TPDUs are multiplexed together – e.g., sharing one IP address.
- ✓ Downward multiplexing: E.g., VC in network layer imposes a max. data rate. Several connections can be set up to increase the data rate.



(a) Upward multiplexing. (b) Downward multiplexing.

Crash Recovery

Strategy used by sending host	Strategy used by receiving host: First ACK, then write			Strategy used by receiving host: First write, then ACK		
	AC(W)	AWC	C(AW)	C(WA)	W AC	WC(A)
Always retransmit	OK	DUP	OK	OK	DUP	DUP
Never retransmit	LOST	OK	LOST	LOST	OK	OK
Retransmit in S0	OK	DUP	LOST	LOST	DUP	OK
Retransmit in S1	LOST	OK	OK	OK	OK	DUP

OK = Protocol functions correctly
DUP = Protocol generates a duplicate message
LOST = Protocol loses a message

Client can be in one of the two states: S1 – one TPDU outstanding; S0 – no outstanding TPDU.

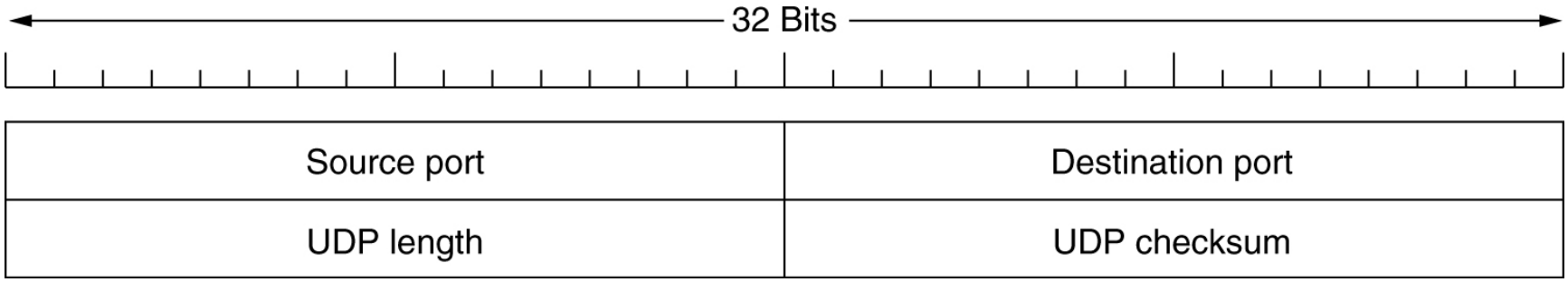
A – Ack; C – Crash; W – Write.

Different combinations of client and server strategy.

The Internet Transport Protocols: UDP

- Introduction to UDP
- Remote Procedure Call
- The Real-Time Transport Protocol

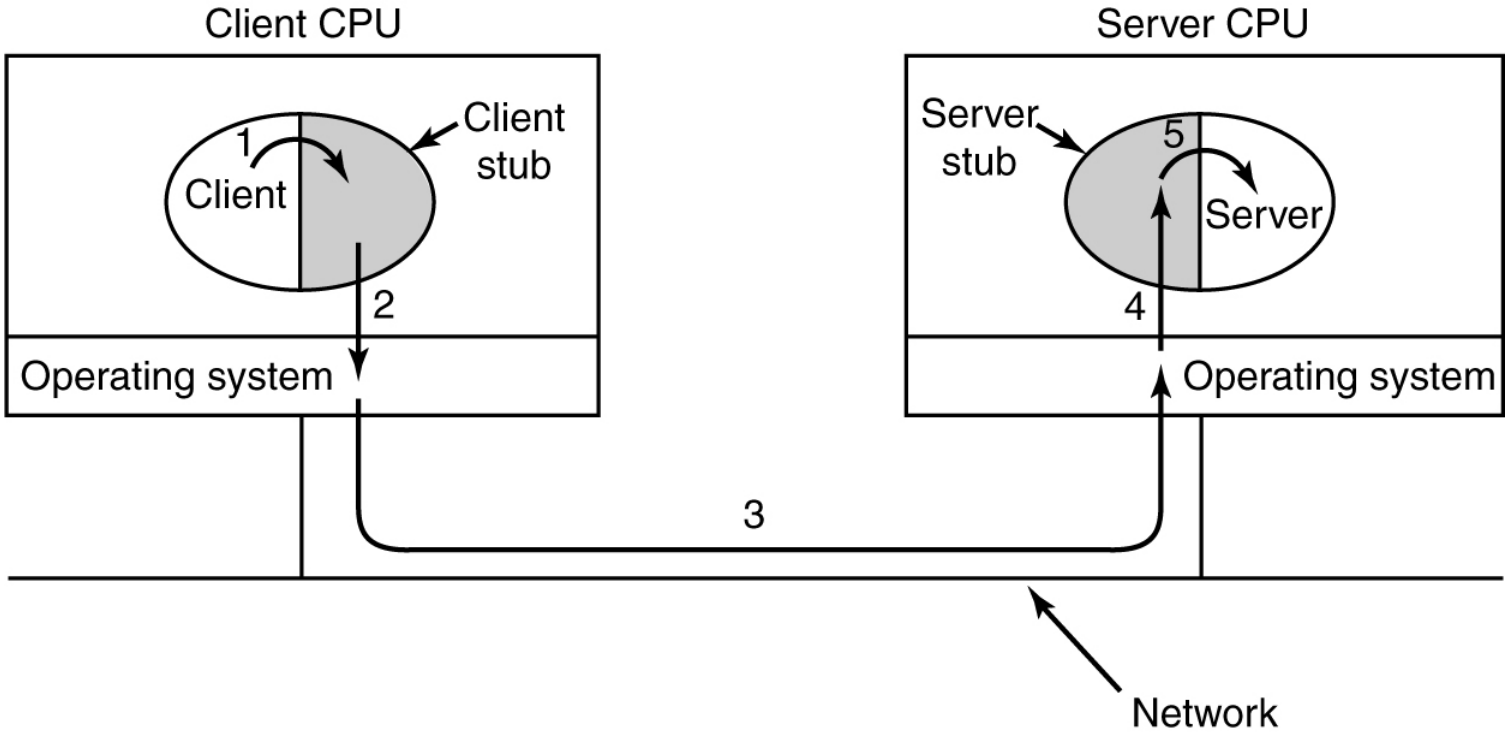
Introduction to UDP



The UDP header.

Remote Procedure Call (RPC)

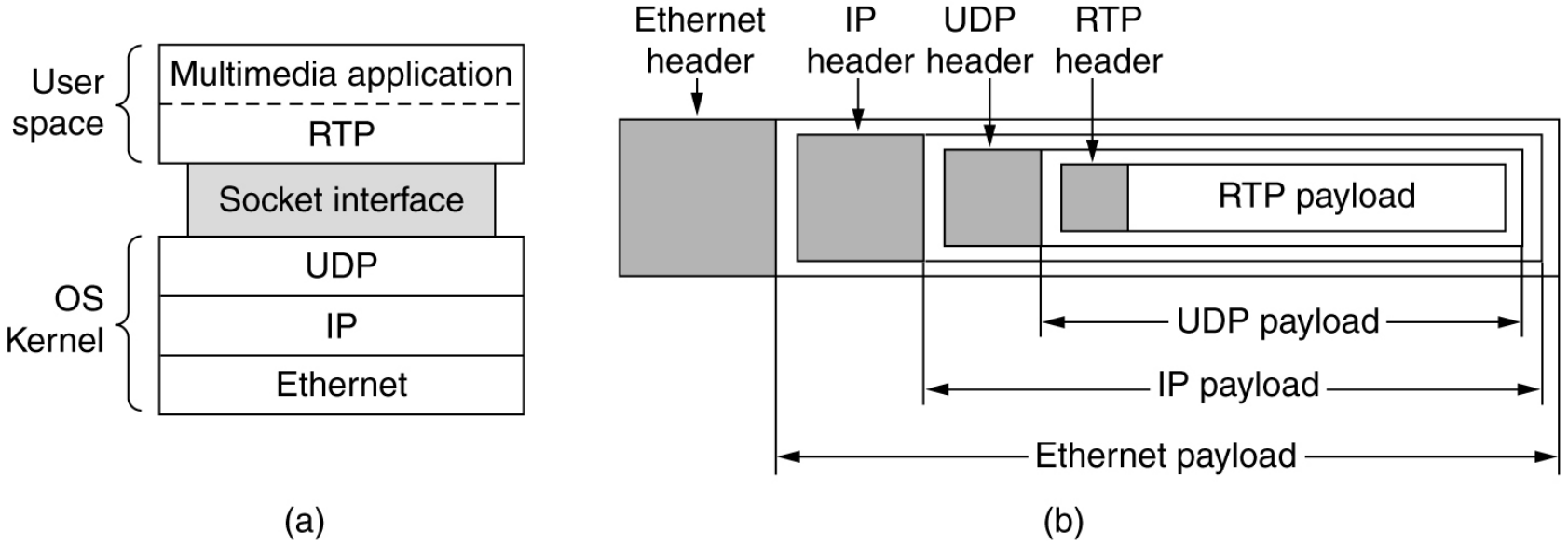
- ✓ RPC - Information can be transported from the caller to the callee in the parameters and can come back in the procedure result.
- ✓ Client stub – A small library procedure that represents the server procedure in the client's address space.



Steps in making a remote procedure call. The stubs are shaded.

The Real-Time Transport Protocol

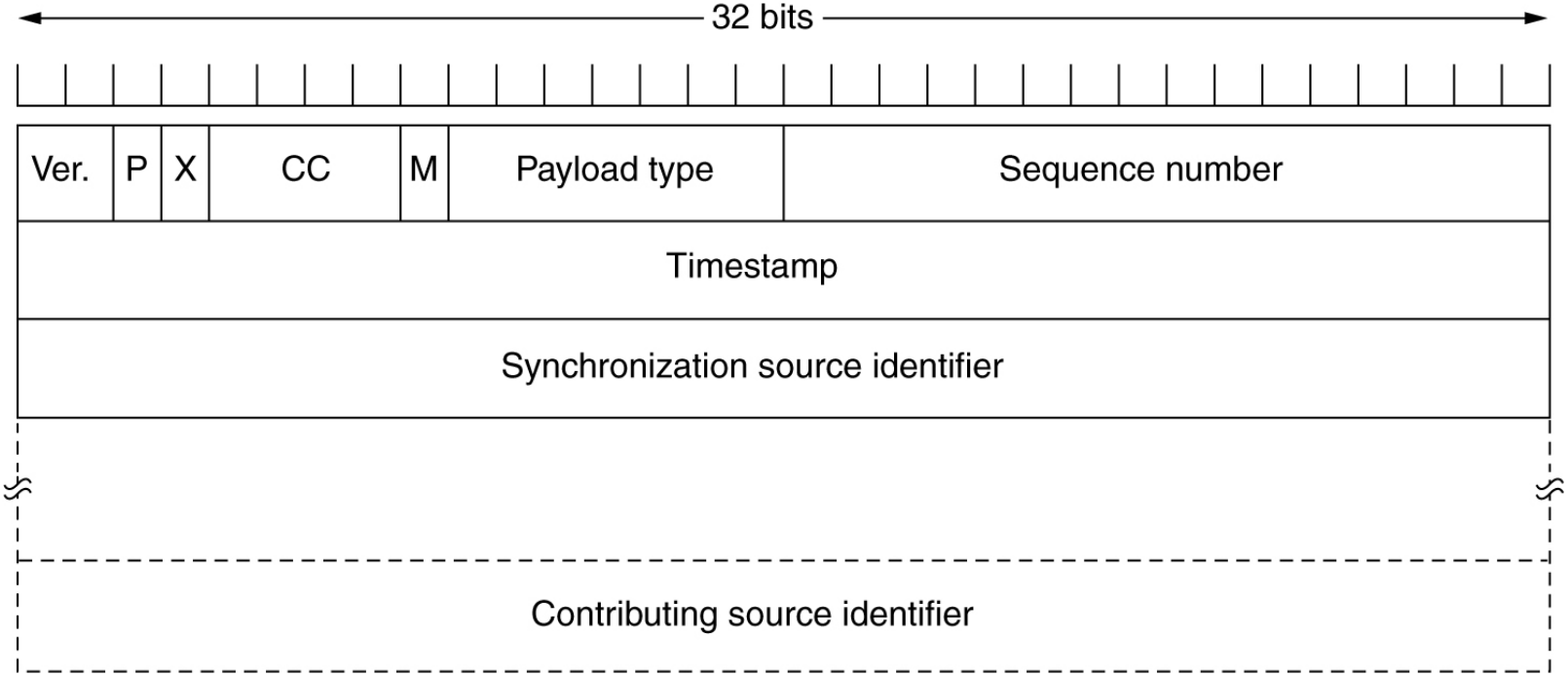
- ✓ RTP – A real time transport protocol for multimedia applications.
- ✓ RTP is put in the user space and runs over UDP.



(a) The position of RTP in the protocol stack. (b) Packet nesting.

The Real-Time Transport Protocol (2)

- P – the packet has been padded to a multiple of 4 bytes.
- X – an extension header is present.
- CC – tells how many contribution sources are present
- M – an application-specific marker bit, e.g., the start of a video frame.
- Payload type – tells which encoding alg. is used, e.g., MP3, MPEG-4.
- Contributing source identifier – when mixers are present in the studio.



The RTP header.

The Internet Transport Protocols: TCP

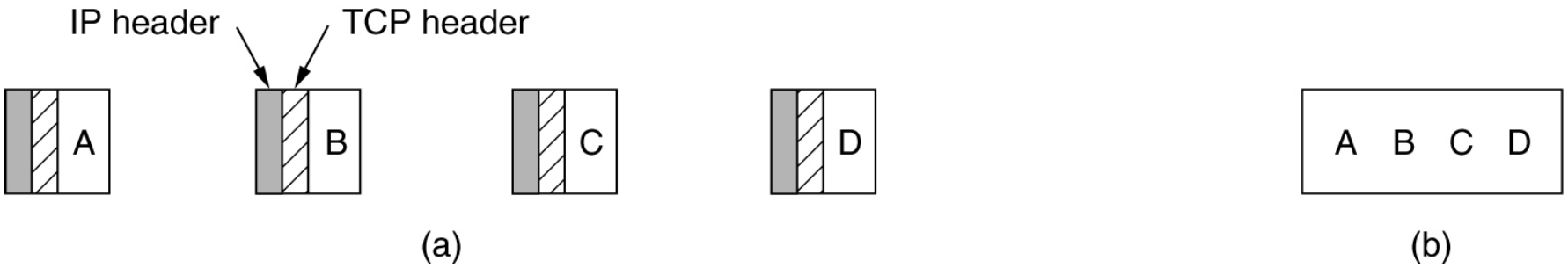
- Introduction to TCP
- The TCP Service Model
- The TCP Protocol
- The TCP Segment Header
- TCP Connection Establishment
- TCP Connection Release
- TCP Connection Management Modeling
- TCP Transmission Policy
- TCP Congestion Control
- TCP Timer Management
- Wireless TCP and UDP
- Transactional TCP

The TCP Service Model

Port	Protocol	Use
21	FTP	File transfer
23	Telnet	Remote login
25	SMTP	E-mail
69	TFTP	Trivial File Transfer Protocol
79	Finger	Lookup info about a user
80	HTTP	World Wide Web
110	POP-3	Remote e-mail access
119	NNTP	USENET news

Some assigned ports.

The TCP Service Model (2)



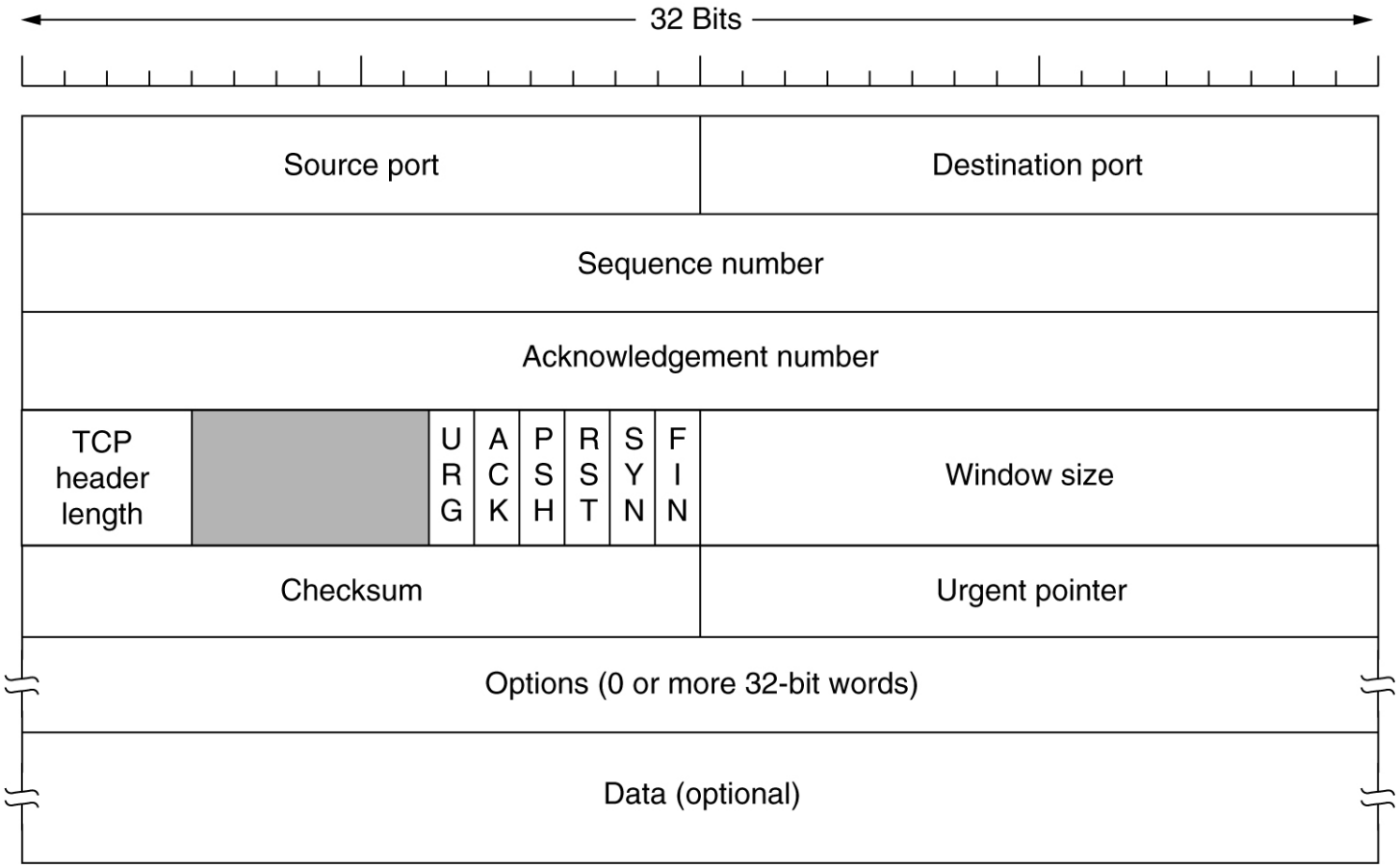
(a) Four 512-byte segments sent as separate IP datagrams.

(b) The 2048 bytes of data delivered to the application in a single READ CALL.

The TCP Segment Header

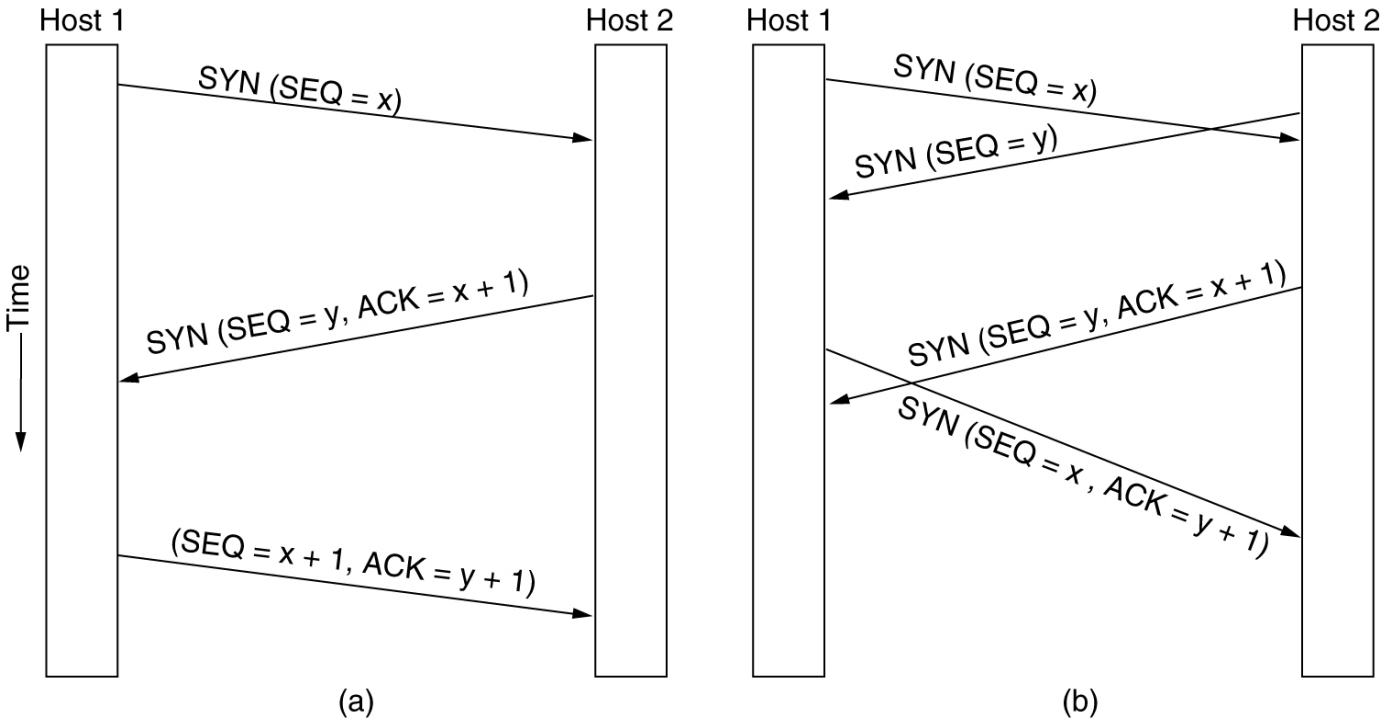
- ✓URG =1: The Urgent pointer is used.
- ✓ACK = 1: The Ack # is valid.
- ✓PSH =1: PUSHed data.
- ✓RST =1: Reset a connection.
- ✓SYN is used to establish a connection.
- ✓FIN bit is used to release a connection.
- ✓Window size: how many bytes may be sent before Ack.

✓Urgent pointer is used to indicate a byte offset from the current seq. at which urgent data are to be found.



TCP Header.

TCP Connection Establishment



(a) TCP connection establishment in the normal case.

(b) Call collision.

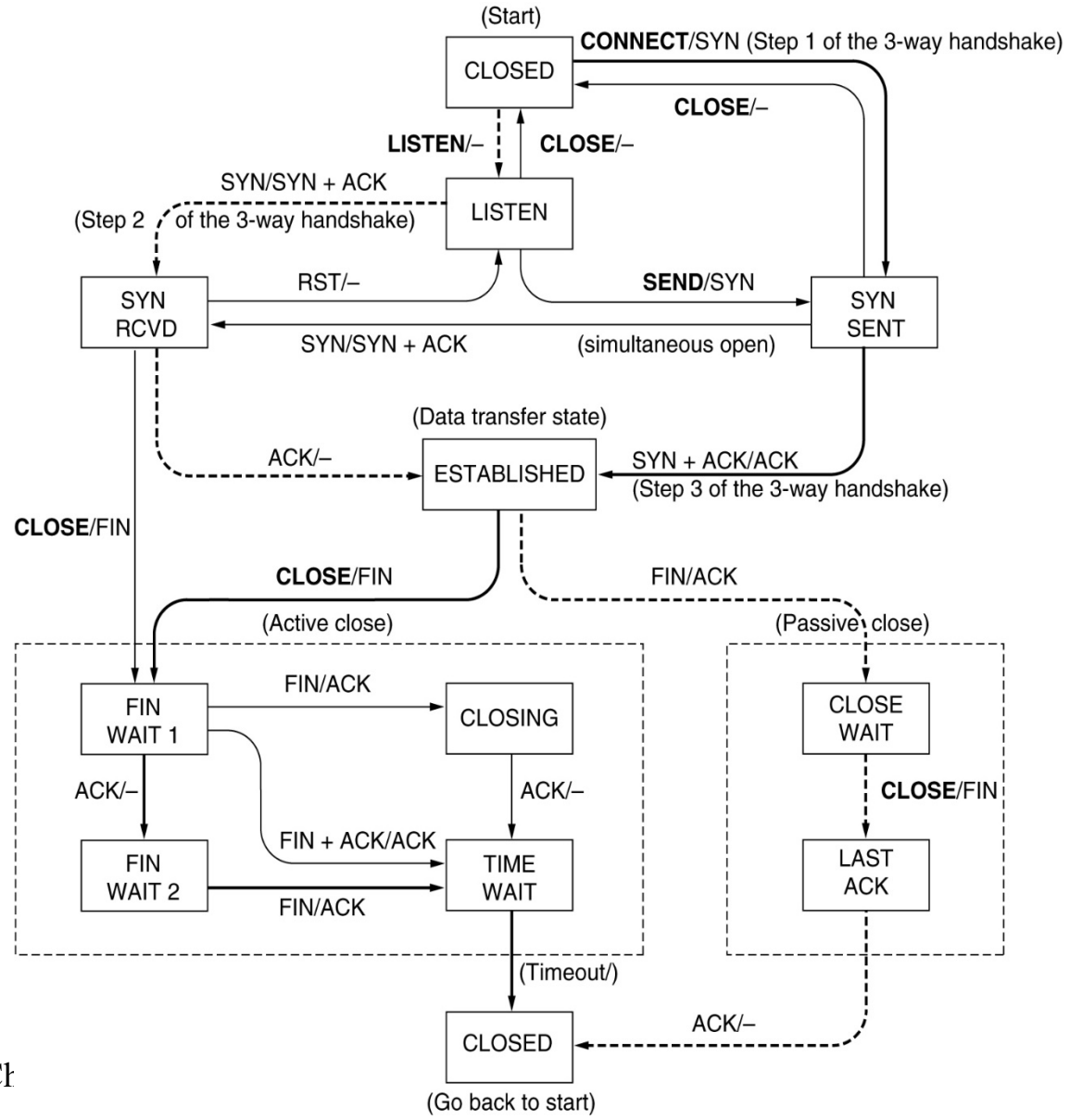
TCP Connection Management Modeling

State	Description
CLOSED	No connection is active or pending
LISTEN	The server is waiting for an incoming call
SYN RCVD	A connection request has arrived; wait for ACK
SYN SENT	The application has started to open a connection
ESTABLISHED	The normal data transfer state
FIN WAIT 1	The application has said it is finished
FIN WAIT 2	The other side has agreed to release
TIMED WAIT	Wait for all packets to die off
CLOSING	Both sides have tried to close simultaneously
CLOSE WAIT	The other side has initiated a release
LAST ACK	Wait for all packets to die off

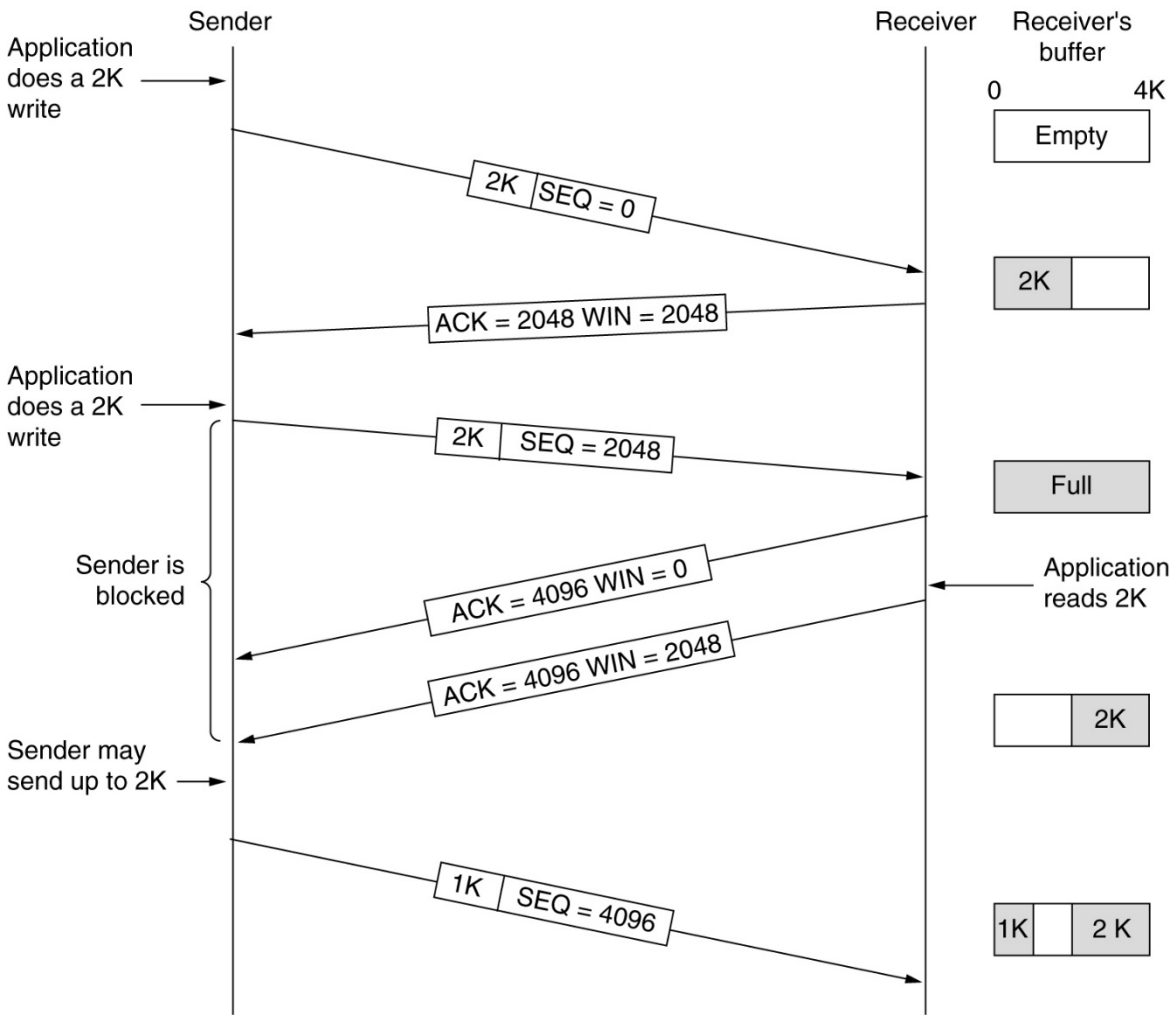
The states used in the TCP connection management finite state machine.

TCP Connection Management Modeling (2)

TCP connection management finite state machine. The heavy solid line is the normal path for a client. The heavy dashed line is the normal path for a server. The light lines are unusual events. Each transition is labeled by the event causing it and the action resulting from it, separated by a slash.

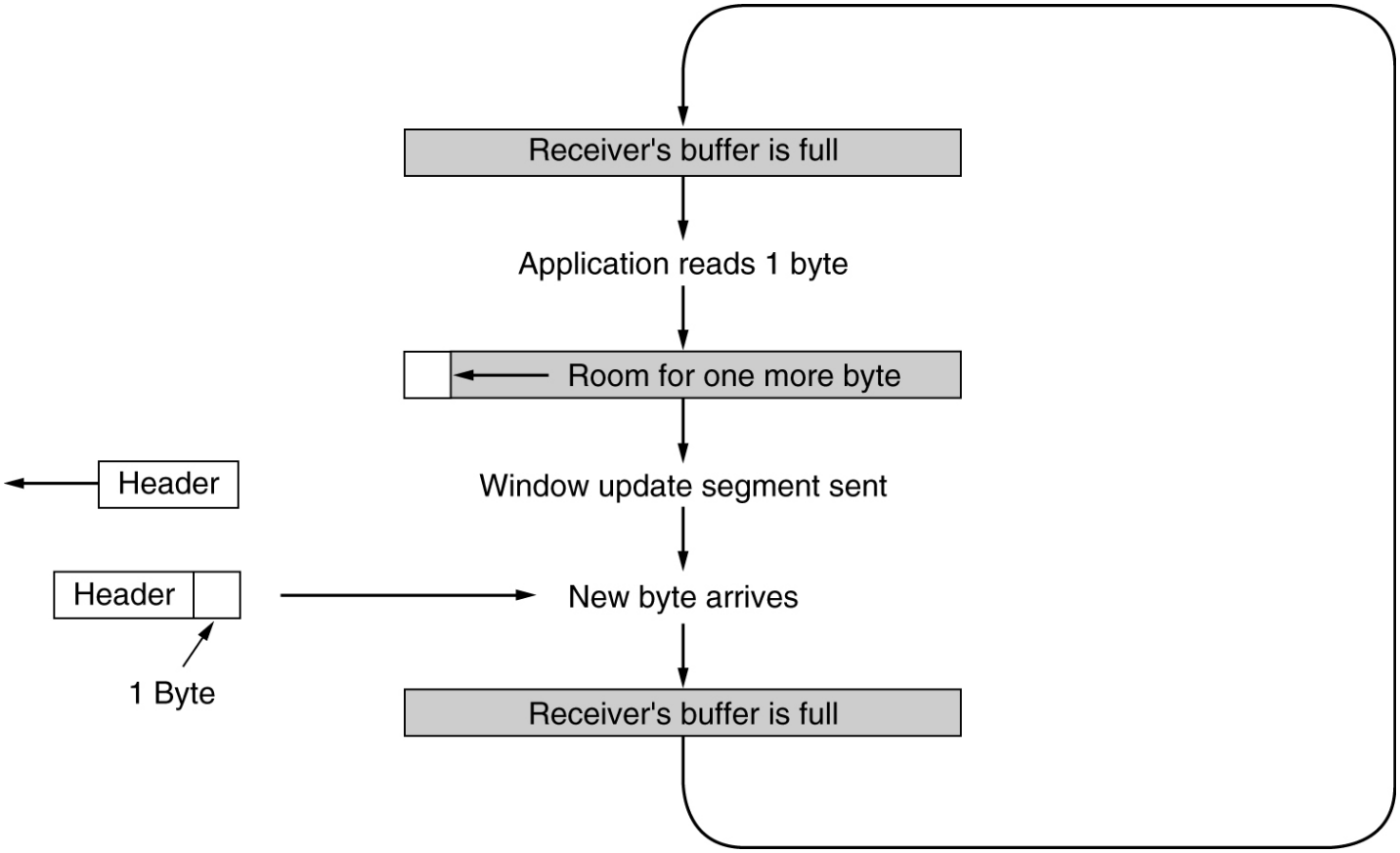


TCP Transmission Policy



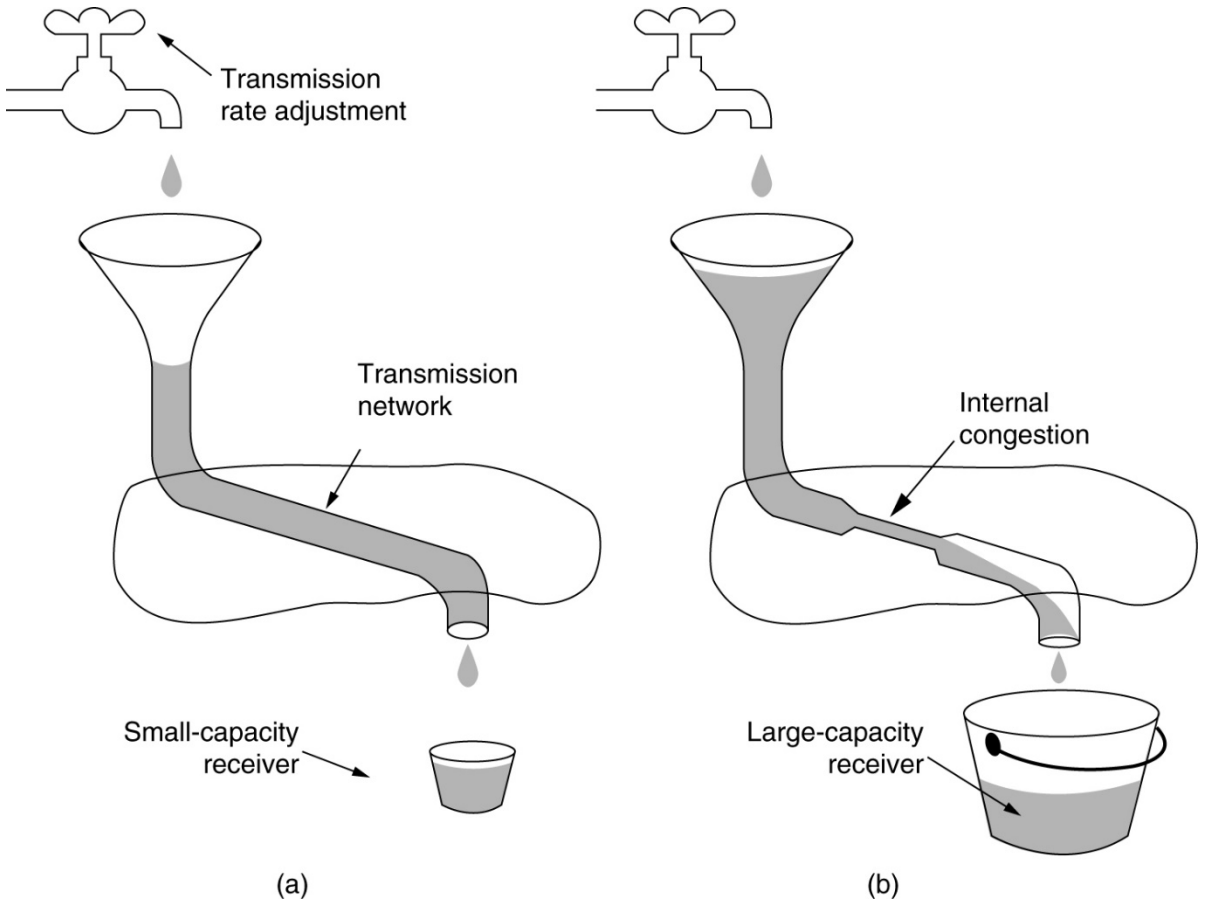
Window management in TCP.

TCP Transmission Policy (2)



Silly window syndrome.

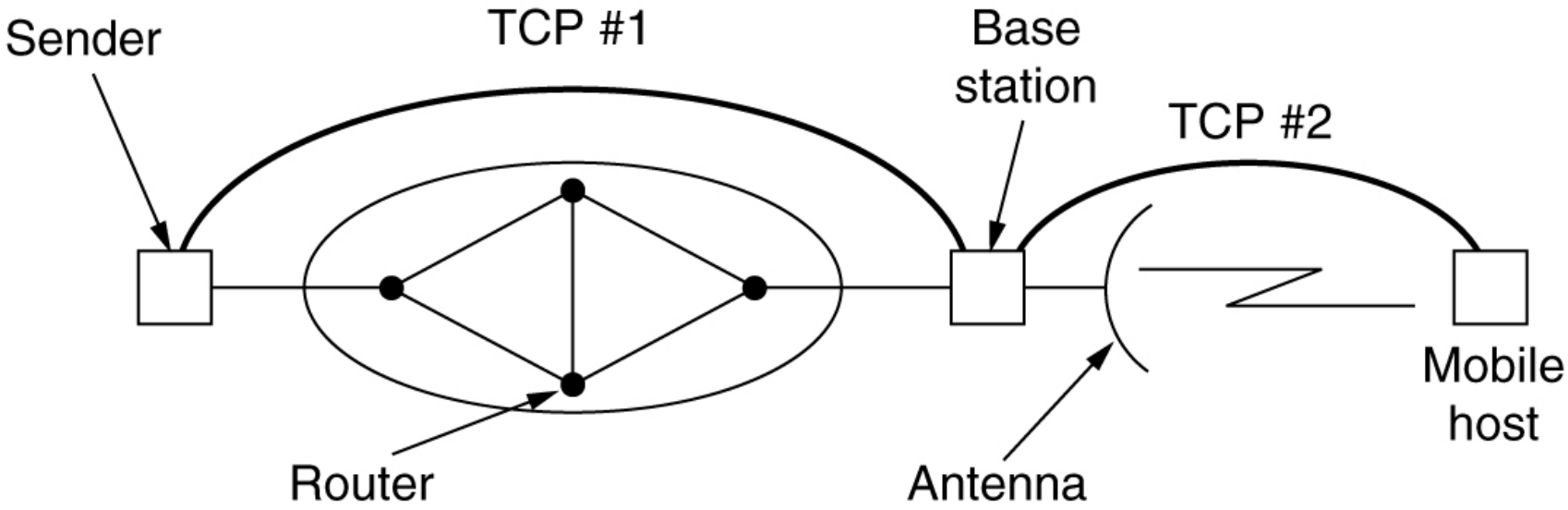
TCP Congestion Control



(a) A fast network feeding a low capacity receiver.

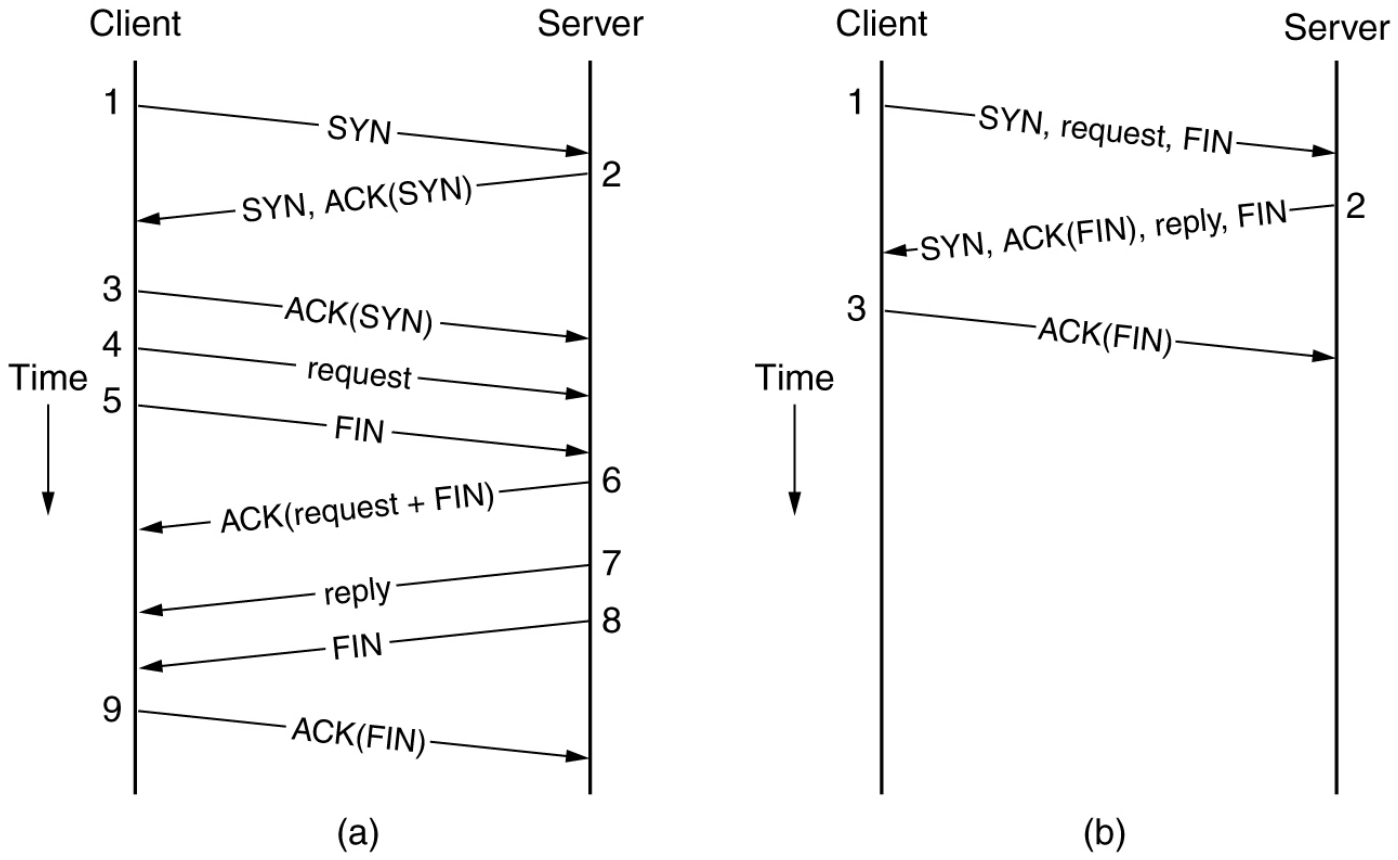
(b) A slow network feeding a high-capacity receiver.

Wireless TCP and UDP



Splitting a TCP connection into two connections.

Transitional TCP



(a) RPC using normal TPC. (b) RPC using T/TCP.

Performance Issues

- Performance Problems in Computer Networks
- Network Performance Measurement
- System Design for Better Performance

Network Performance Measurement

The basic loop for improving network performance.

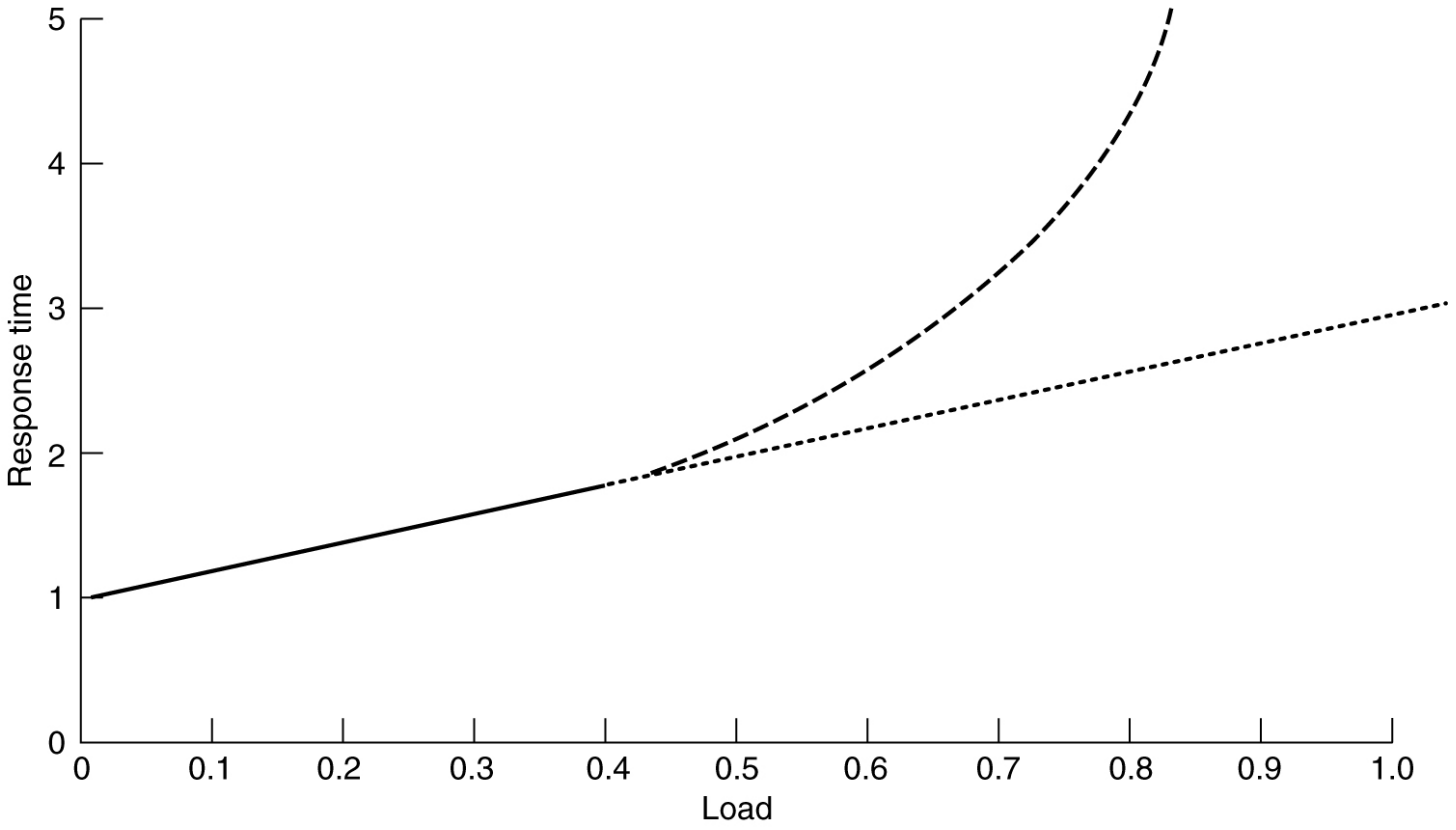
1. Measure relevant network parameters, performance.
2. Try to understand what is going on.
3. Change one parameter.

System Design for Better Performance

Rules:

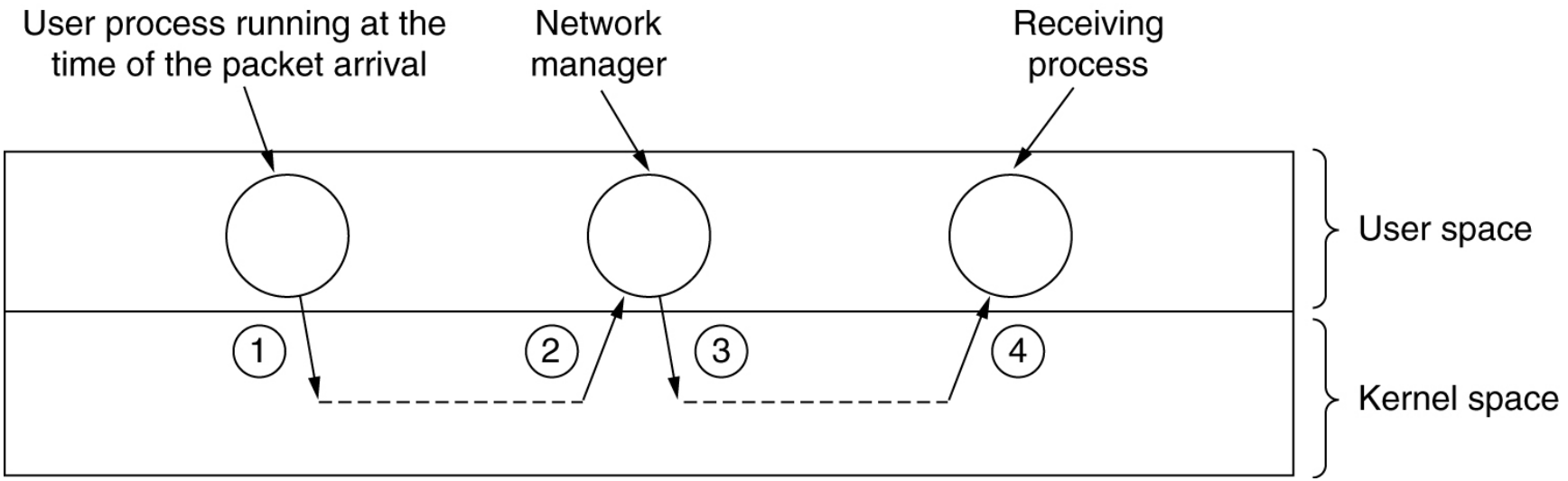
1. CPU speed is more important than network speed.
2. Reduce packet count to reduce software overhead.
3. Minimize context switches.
4. Minimize copying.
5. You can buy more bandwidth but not lower delay.
6. Avoiding congestion is better than recovering from it.
7. Avoid timeouts.

System Design for Better Performance (2)



Response as a function of load.

System Design for Better Performance (3)



Four context switches to handle one packet with a user-space network manager.