The Saigon CTT

Chapter 12,13,14

TRANSPORT

SESSION

PRESENTATION

LAYERS

TRANSPORT LAYER

Why we need a transport layer?

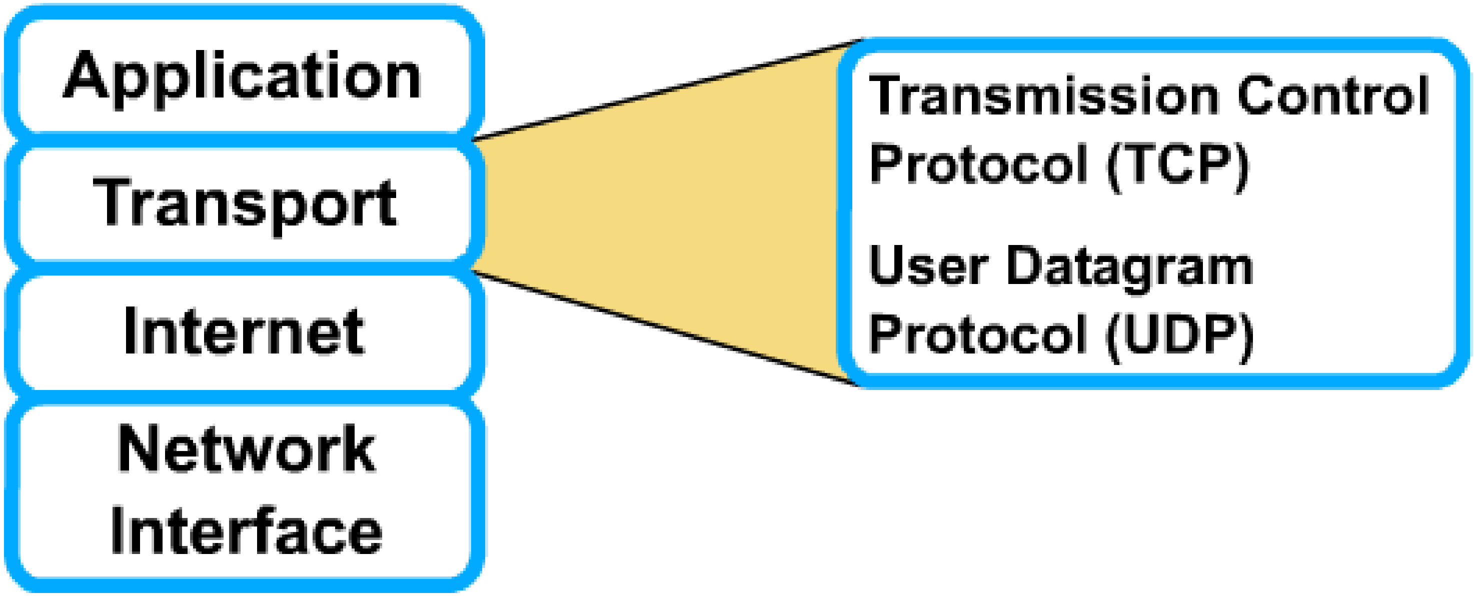
- Layer 1 allows bit streams to be created and to travel.
-
- layers in packets and makes routing and network delivery possible.

But they made no provision
for assuring our data reliably
travels end-to-end across
the often vast network path.

Purpose of the transport layer

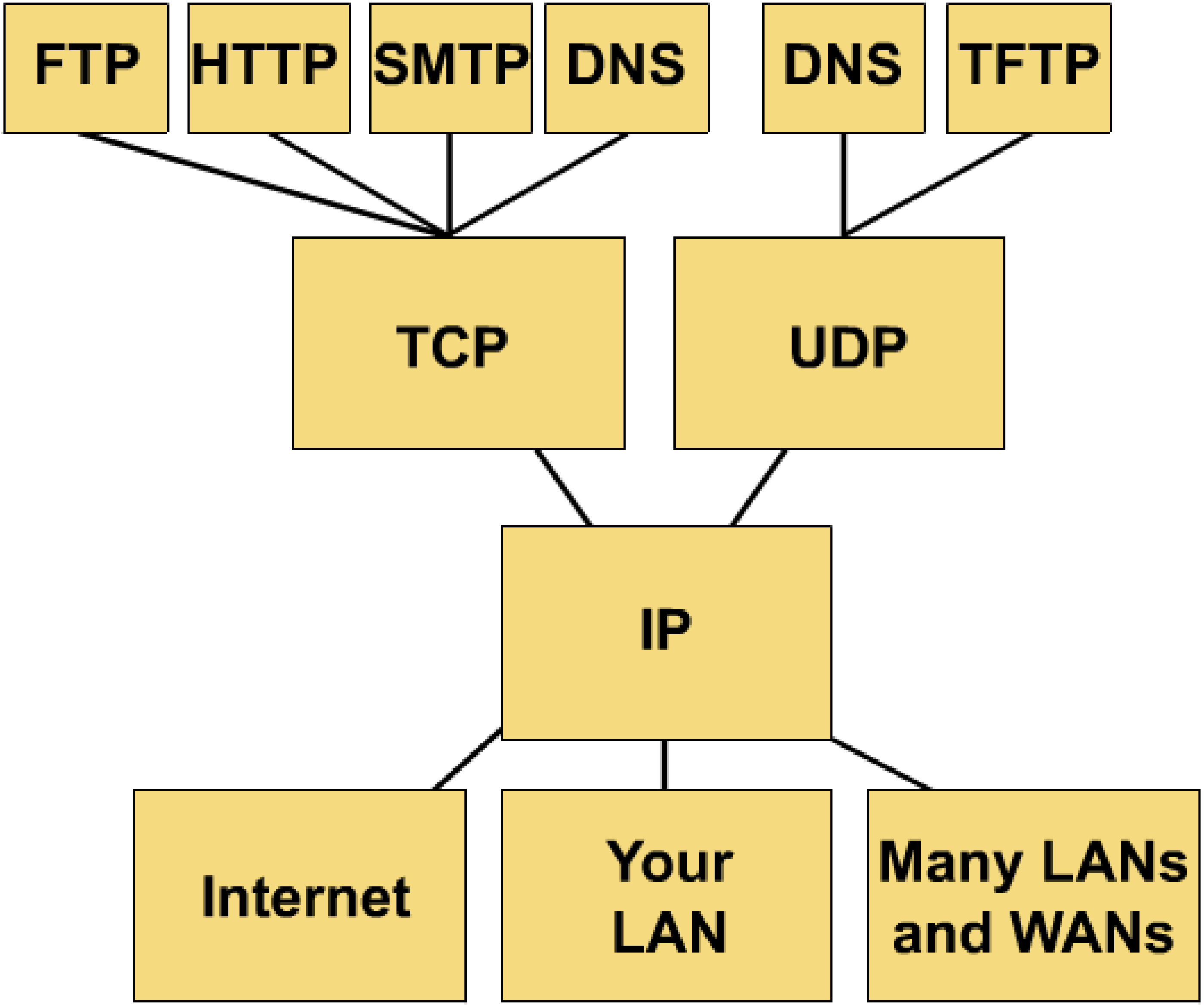
- Transport and regulate the flow of information from source to destination, reliably and accurately.
- The end-to-end control:
  - Sliding windows.
  - Sequencing numbers.
  - Acknowledgments.
  - Segmentation.
  - Multiplexing.

TCP and UDP

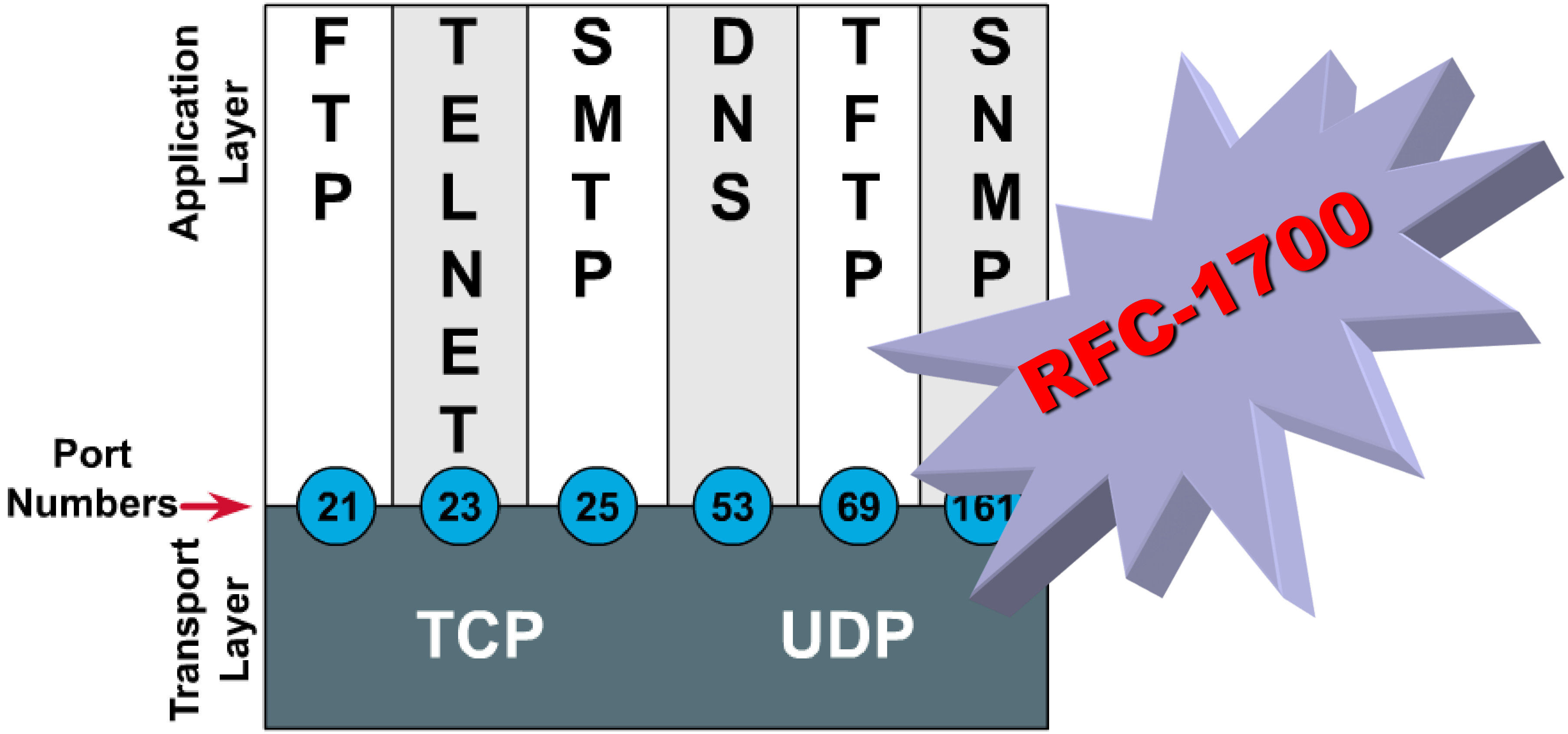

Application
Transport
Internet
Network Interface
Transmission Control Protocol (TCP)
User Datagram Protocol (UDP)

- The emphasis of this curriculum is on TCP/IP Ethernet networks.
- The TCP/IP protocol of the OSI model Layer 4 (transport layer) has two protocols - TCP and UDP.

TCP/IP


FTP
HTTP
SMTP
DNS
DNS
TFTP
TCP
UDP
IP
Internet
Your LAN
Many LANs and WANs

Port



- Both TCP and UDP use port (or socket) numbers to pass information to the upper layers.

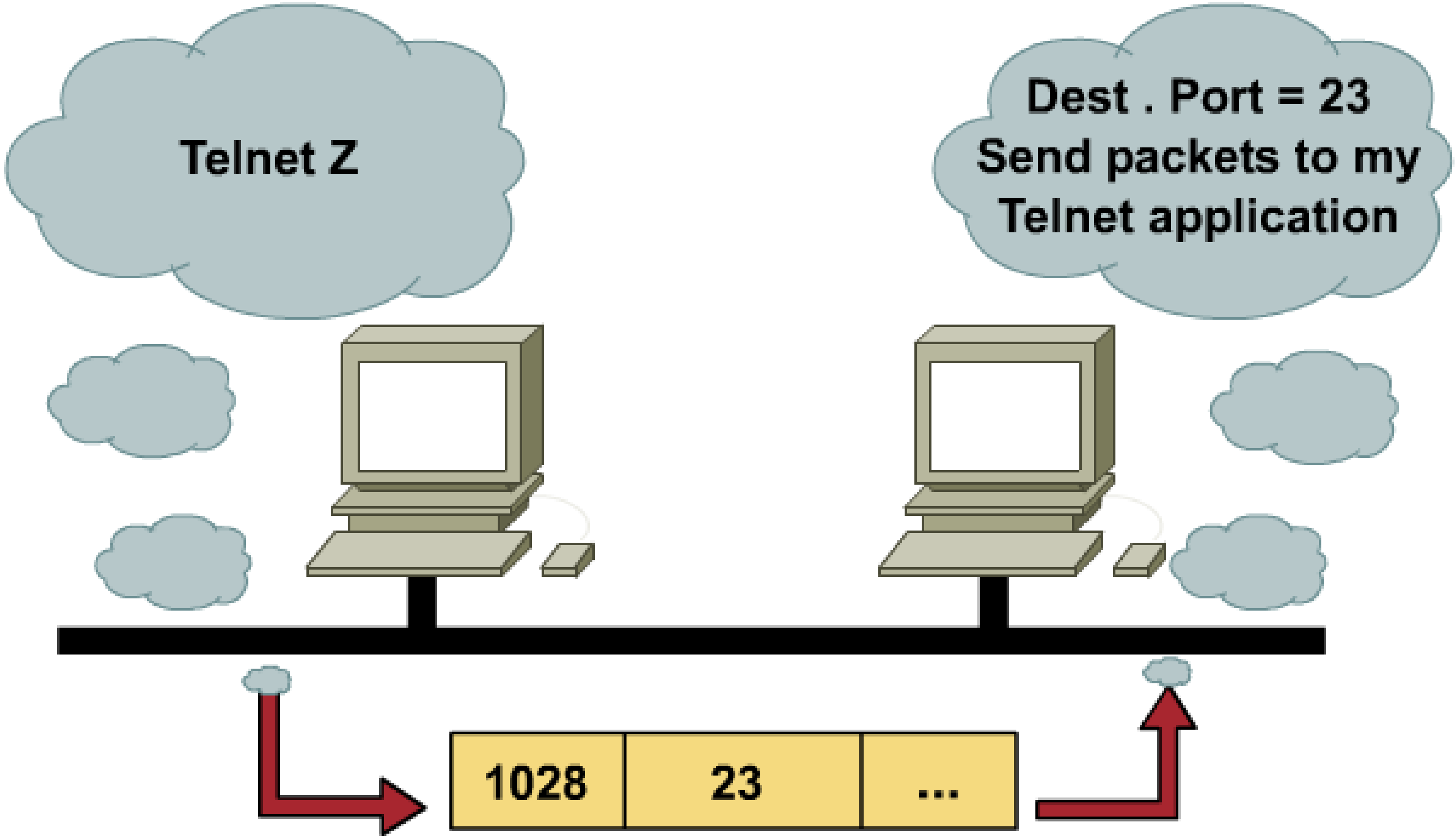
Range of ports

- 2 bytes: 0 – 65535.
  - Numbers below 255 : for public applications.
  - Numbers from 255 - 1023 : assigned to companies for marketable applications.
  - Numbers above 1023 : are unregulated.
- End systems use port numbers to select proper applications.
- Originating source port numbers are dynamically assigned by the source host; usually, it is a number larger than 1023.

Telnet port number

The Saigon CTT

Source Port	Destination Port	...



Telnet Z

Dest . Port = 23
Send packets to my
Telnet application

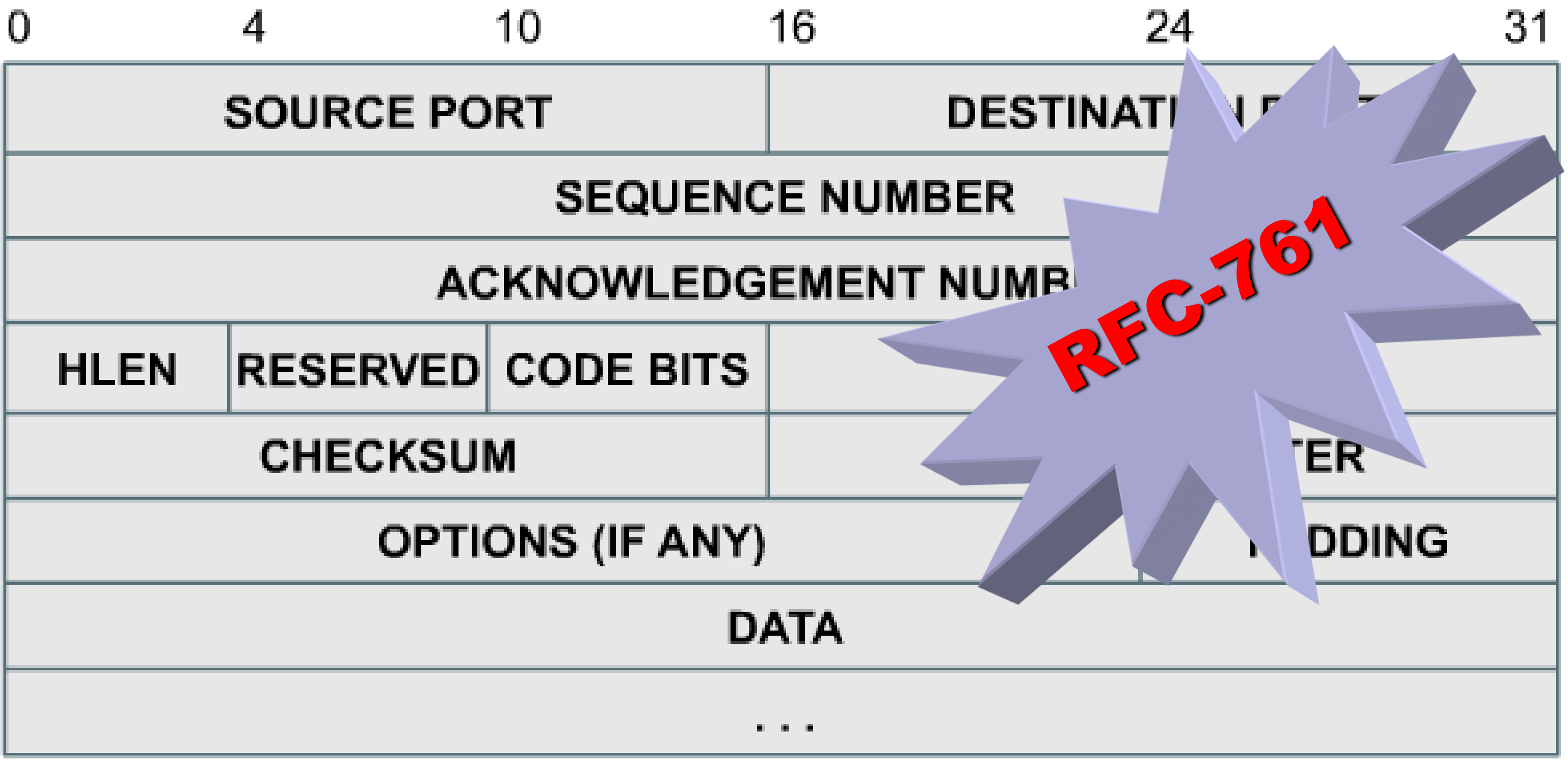
1028	23	...

TCP AND UDP

TCP

- TCP supplies a virtual circuit between end-user applications. These are its characteristics:
  - connection-oriented.
  - reliable.
  - divides outgoing messages into segments.
  - reassembles messages at the destination station.
  - re-sends anything not received.
  - reassembles messages from incoming segments.

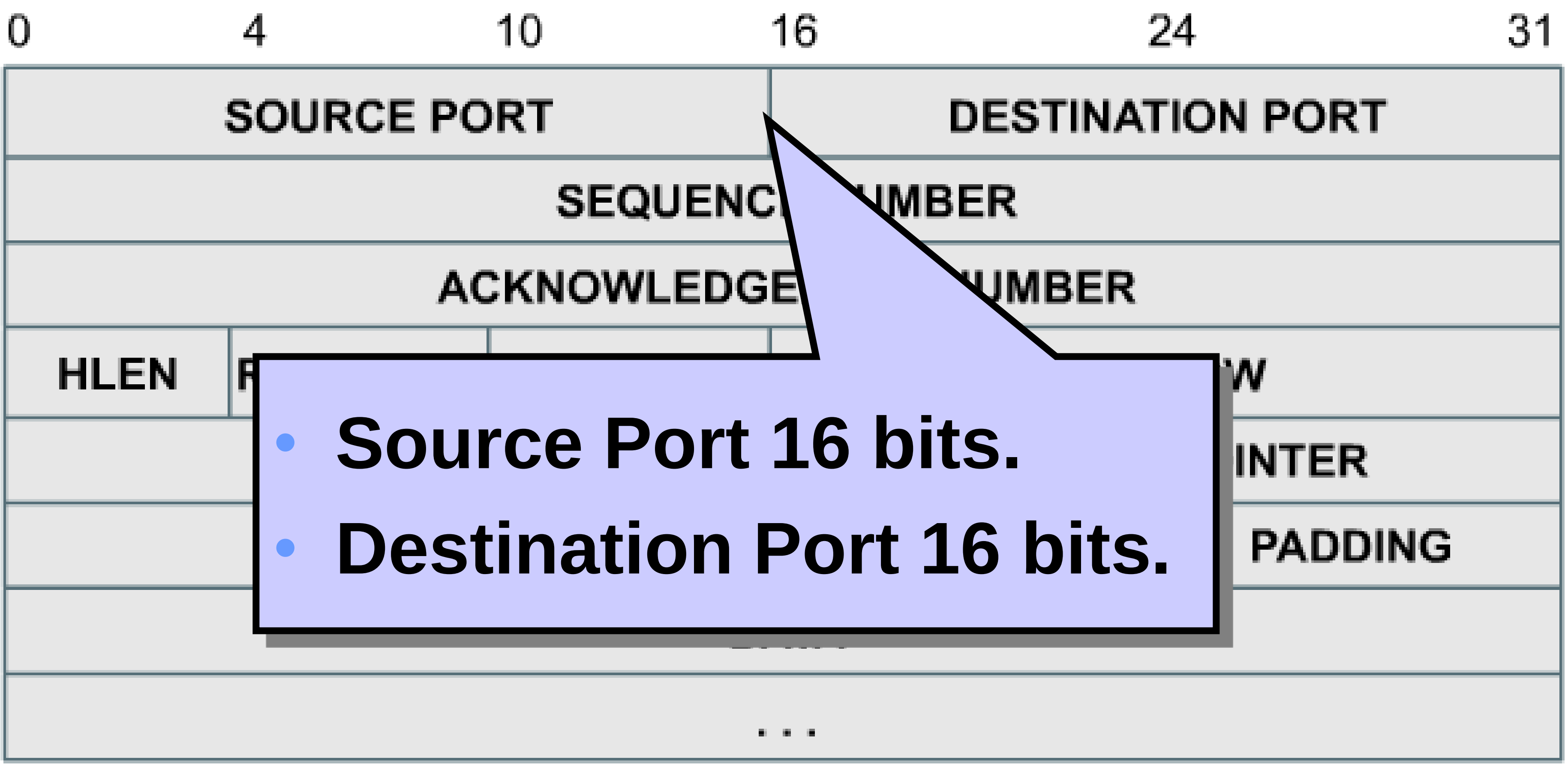
TCP Header format



0	4	10	16	24	31
SOURCE PORT			DESTINATI		
SEQUENCE NUMBER					
ACKNOWLEDGEMENT NUMB					
HLEN	RESERVED	CODE BITS			
CHECKSUM				TER	
OPTIONS (IF ANY)				DDING	
DATA					
. . .					

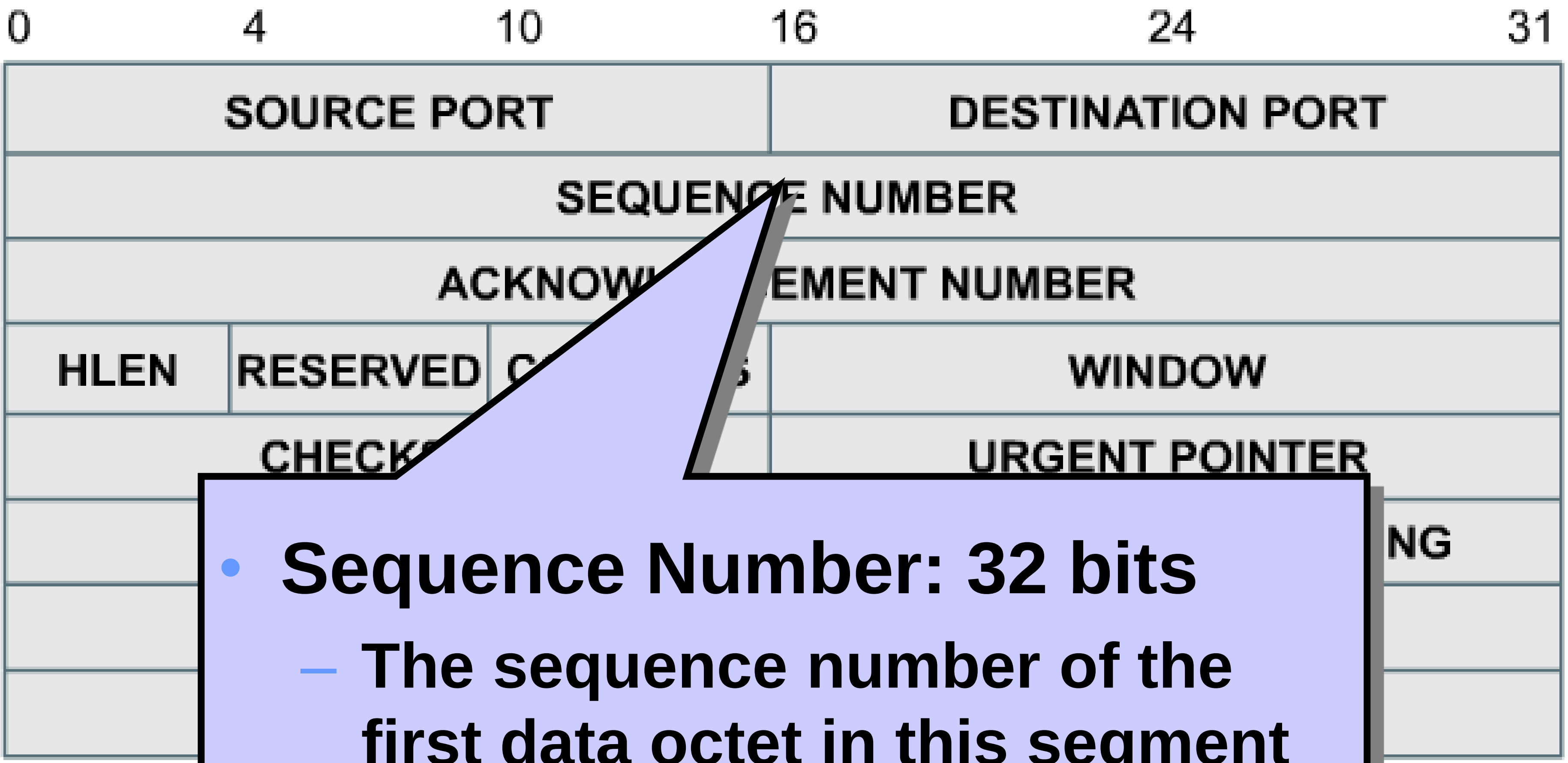
- Protocol that provides reliable full-duplex data transmission.

TCP Header format: Port number



- Source Port 16 bits.
- Destination Port 16 bits.

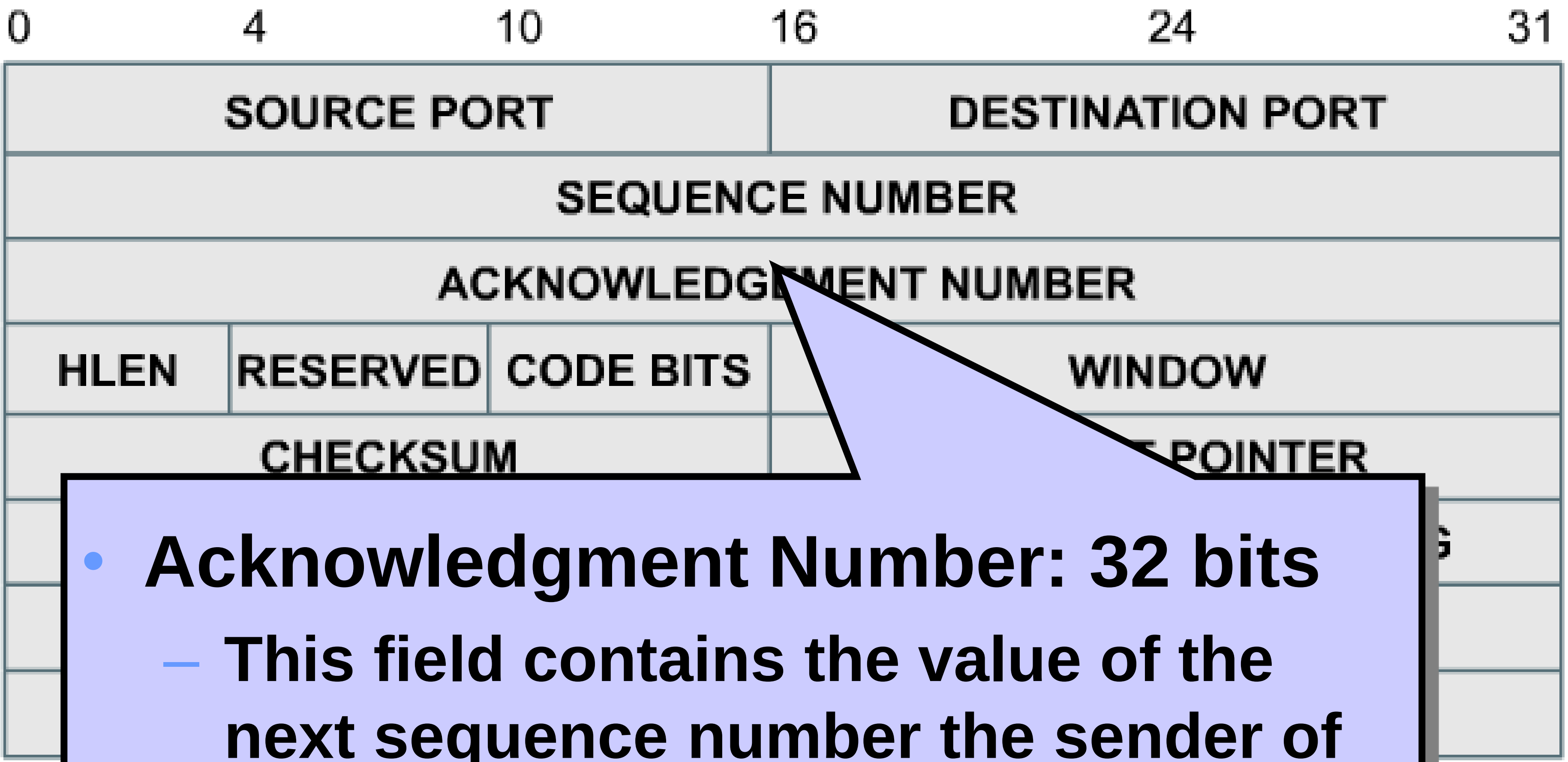
TCP Header format: Sequence



0	4	10	16	24	31
SOURCE PORT			DESTINATION PORT		
SEQUENCE NUMBER					
ACKNOWL EMENT NUMBER					
HLEN	RESERVED	C S	WINDOW		
CHECK			URGENT POINTER		
NG					

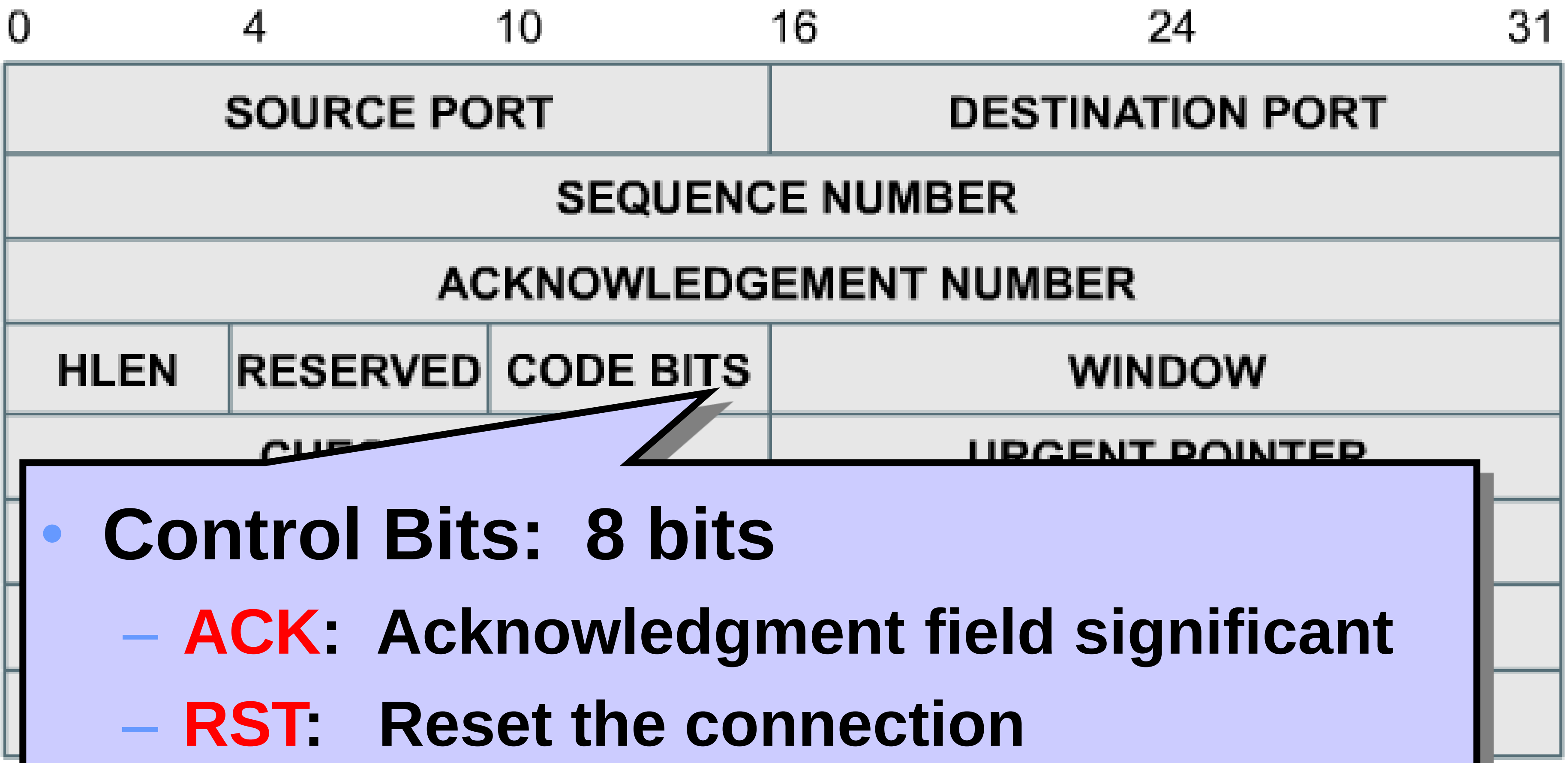
- **Sequence Number: 32 bits**
  - **The sequence number of the first data octet in this segment (except when SYN is present).**

TCP Header format: Acknowledgment



- **Acknowledgment Number: 32 bits**
  - **This field contains the value of the next sequence number the sender of the segment is expecting to receive.**

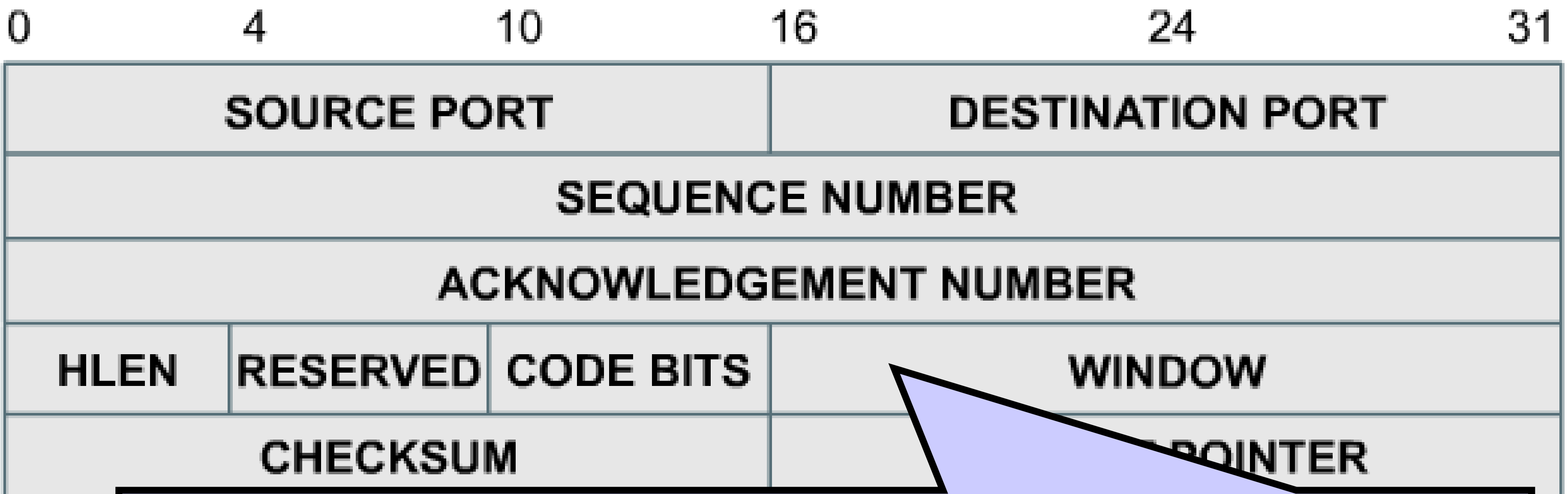
TCP Header format: Code bits



0	4	10	16	24	31
SOURCE PORT			DESTINATION PORT		
SEQUENCE NUMBER					
ACKNOWLEDGEMENT NUMBER					
HLEN	RESERVED	CODE BITS	WINDOW		
CHE			URGENT POINTER		

- Control Bits: 8 bits
  - ACK: Acknowledgment field significant
  - RST: Reset the connection
  - SYN: Synchronize sequence numbers
  - FIN: No more data from sender

TCP Header format: Window



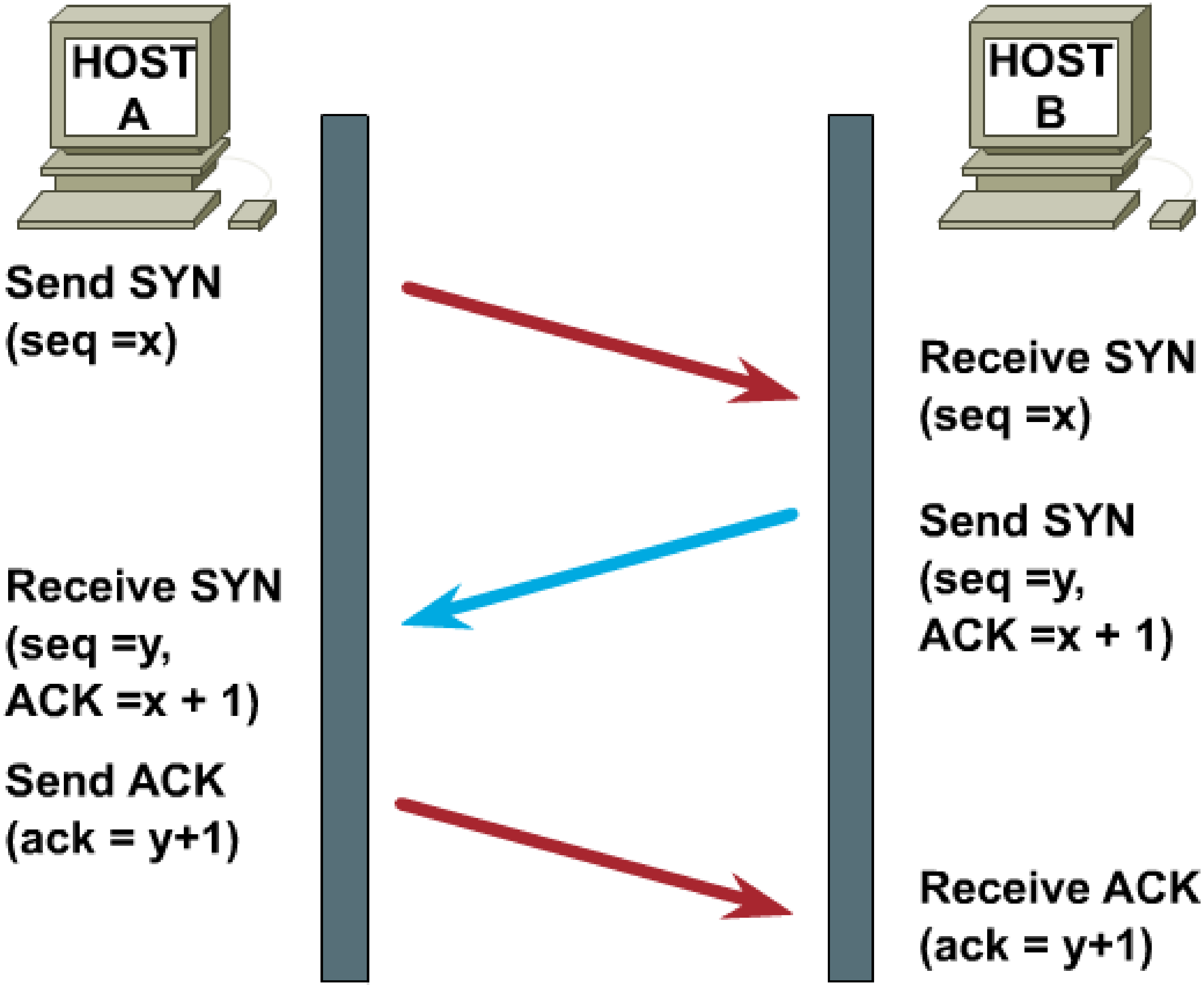
0 4 10 16 24 31

SOURCE PORT			DESTINATION PORT
SEQUENCE NUMBER			
ACKNOWLEDGEMENT NUMBER			
HLEN	RESERVED	CODE BITS	WINDOW
CHECKSUM			POINTER

- Window: 16 bits
  - The number of data octets beginning with the one indicated in the acknowledgment field which the sender of this segment is willing to accept.

TCP: Three way handshaking

The Saigon CTT



Send SYN
(seq =x)

Receive SYN
(seq =x)

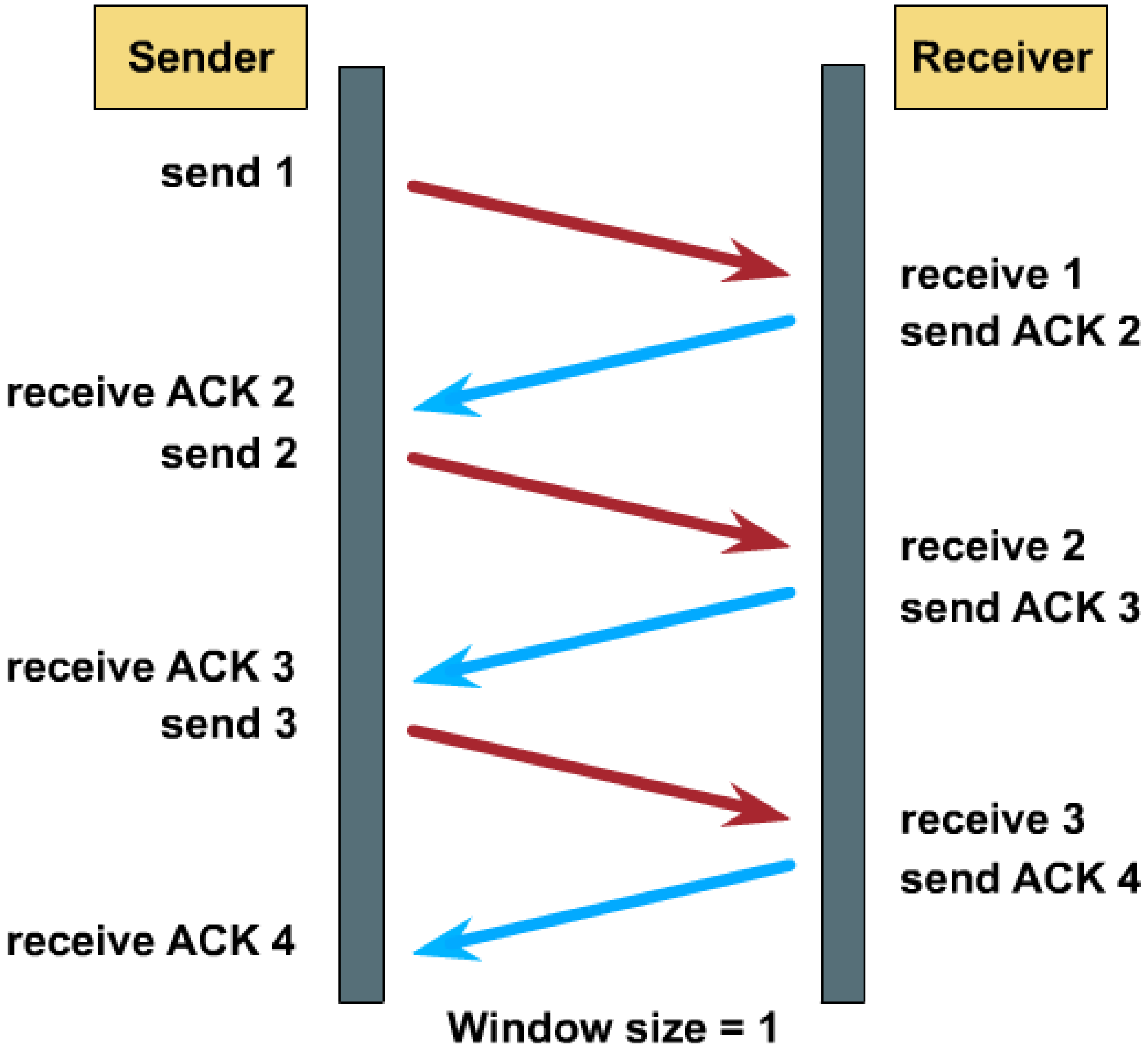
Send SYN
(seq =y,
ACK =x + 1)

Receive SYN
(seq =y,
ACK =x + 1)

Send ACK
(ack = y+1)

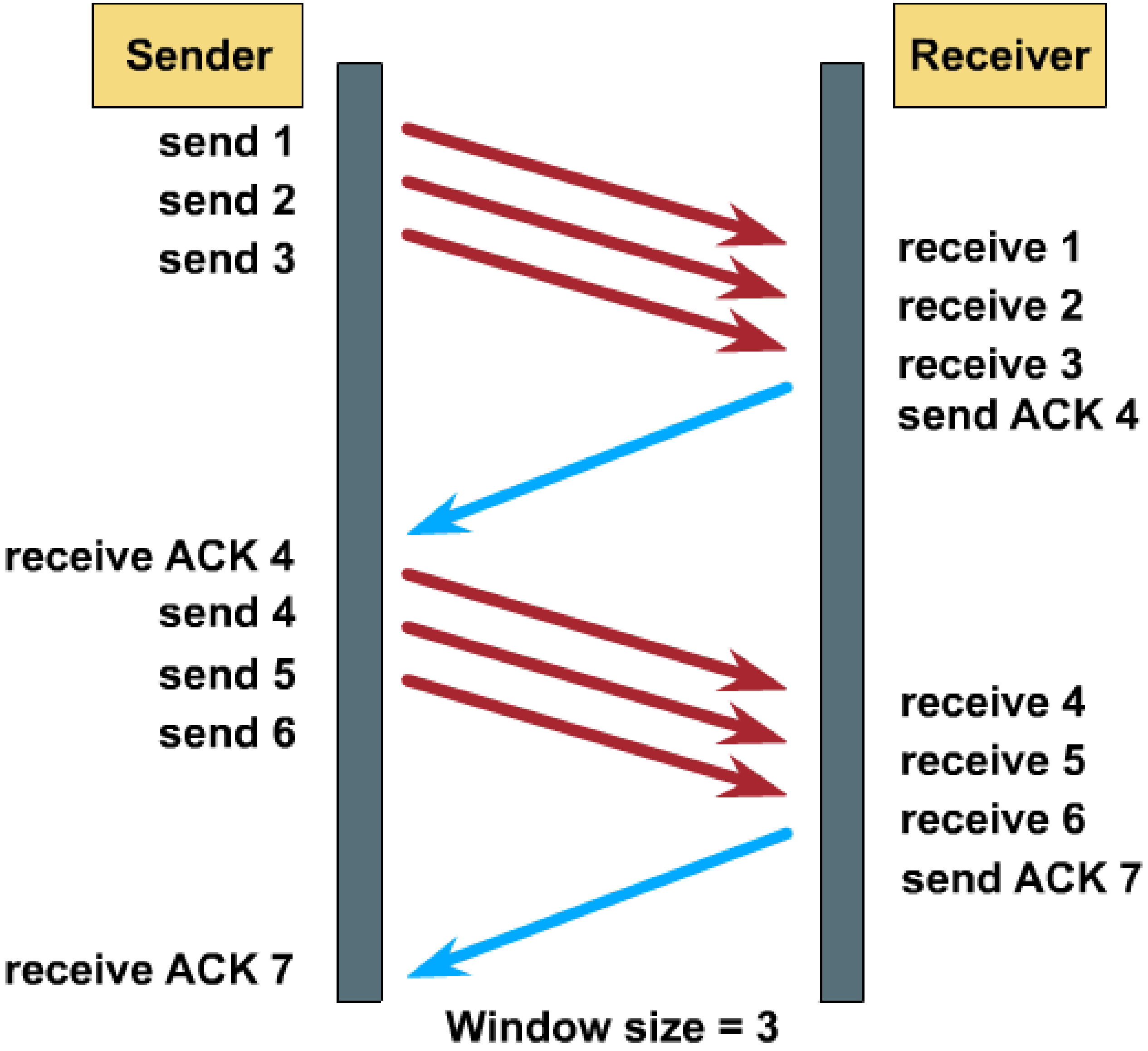
Receive ACK
(ack = y+1)

TCP: Simple acknowledgment



TCP: Sliding acknowledgment

The Saigon CTT



Sender

Receiver

send 1
send 2
send 3

receive 1
receive 2
receive 3
send ACK 4

receive ACK 4
send 4
send 5
send 6

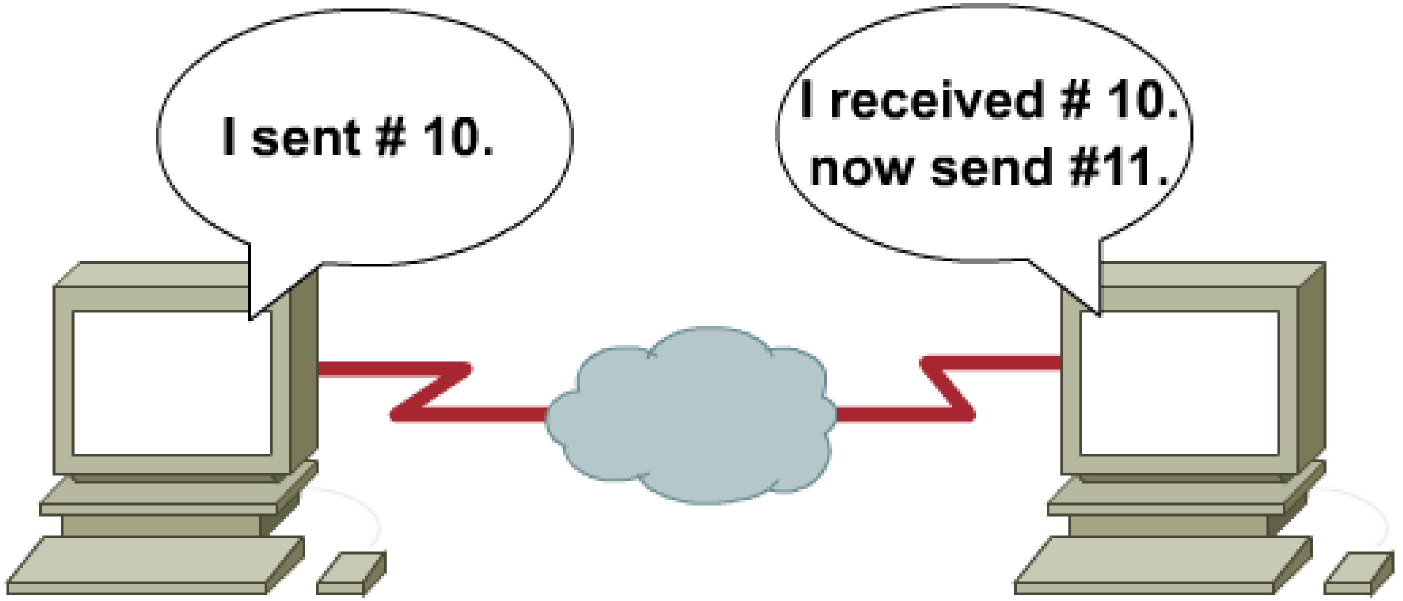
receive 4
receive 5
receive 6
send ACK 7

receive ACK 7

Window size = 3

TCP: Sequence and acknowledgment

Source Port	Destination Port	Sequence Number	Acknowledgment Numbers	...



Source	Des.	Seq.	Ack.	
1028	23	10	1	...

Source	Des.	Seq.	Ack.	
1028	23	1	11	...

Source	Des.	Seq.	Ack.	
1028	23	11	2	...

UDP

- UDP transports data unreliably between hosts. Following are the characteristics:
  - Connectionless.
  - Unreliable.
  - Transmit messages (called user datagrams).
  - Provides no software checking for message delivery (unreliable).
  - Does not reassemble incoming messages.
  - Uses no acknowledgements.

UDP Header format

# of Bits	16	16	16	1	
	Source Port	Destination Port	Length		

RFC-768

- UDP is a simple protocol that exchanges datagrams, without acknowledgments or guaranteed delivery.

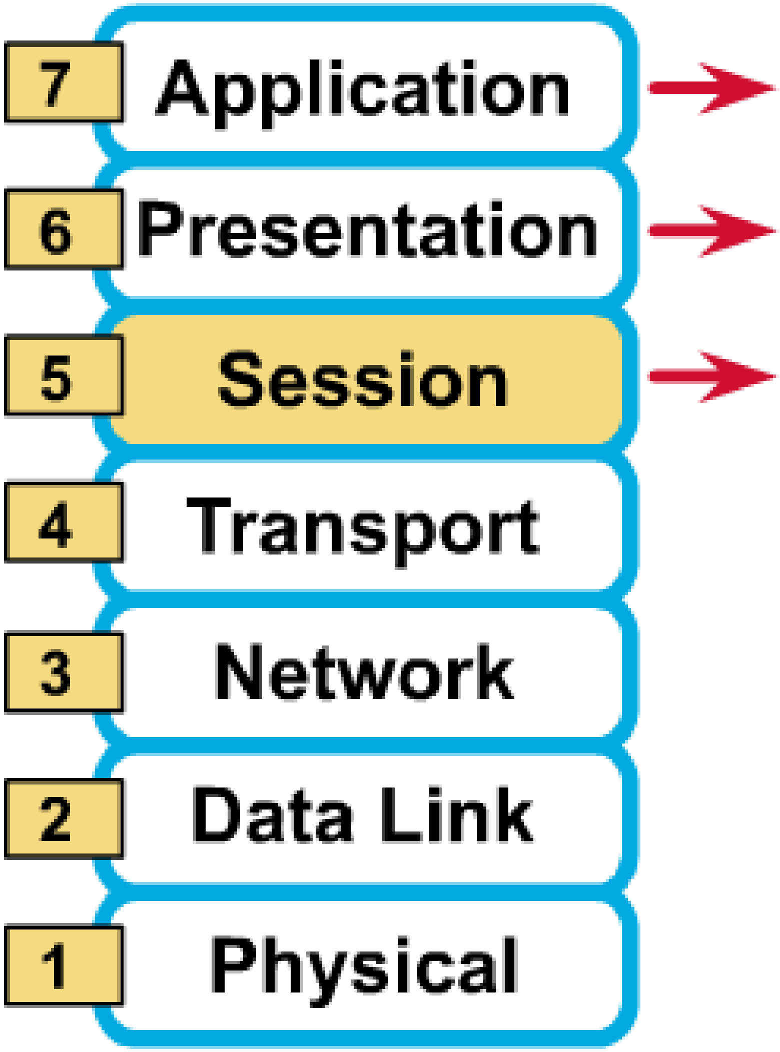
Preparation for LAB

- Lab companion:
  - 12.4.1: Protocol Inspector and TCP

The Saigon CTT

SESSION LAYER

Functions



Network Processes to Applications

Data Representation

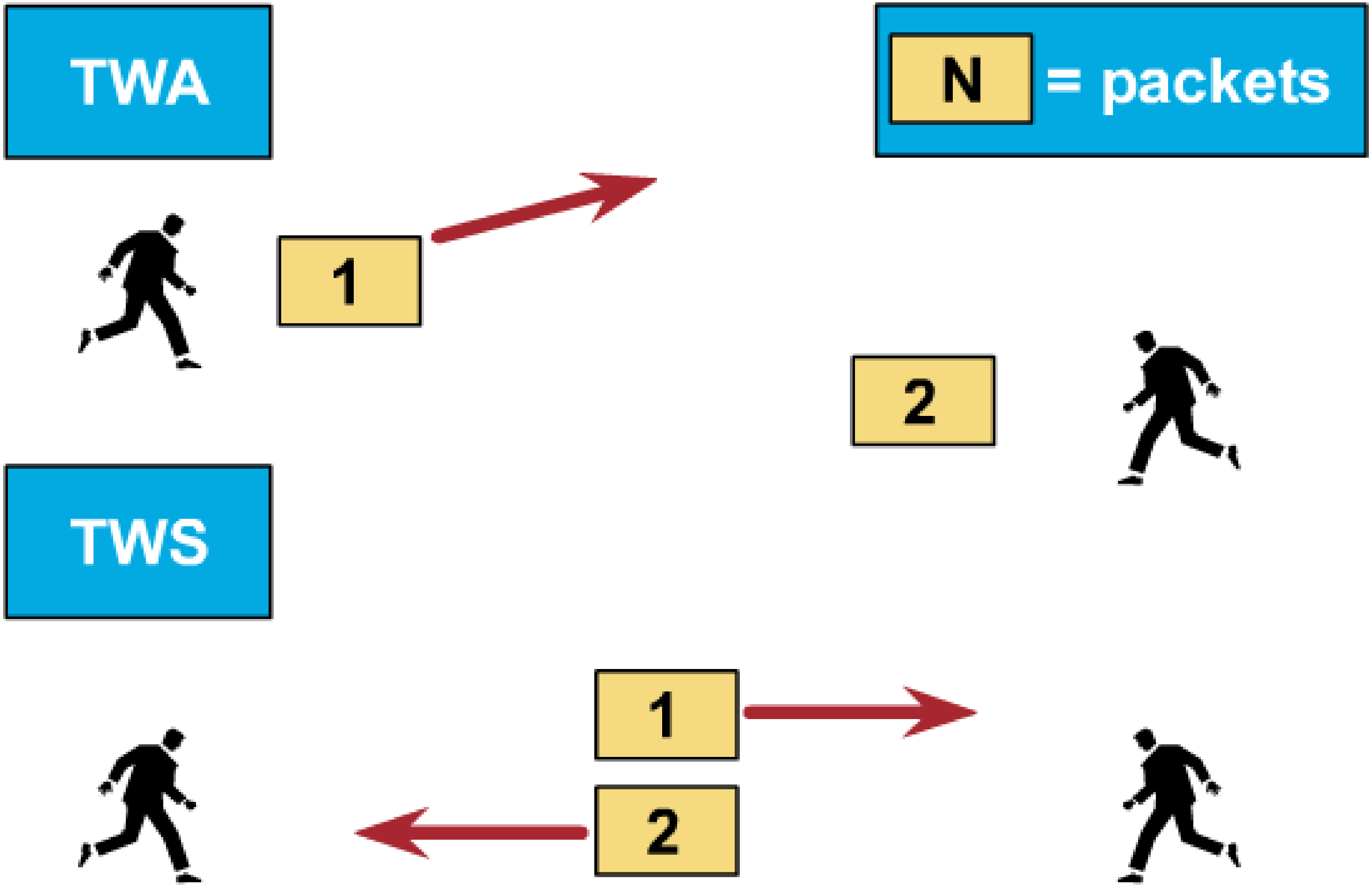
Interhost Communication

- Establishes, manages, and terminates sessions between applications

Process

- Dialog control:
  - In the conversation, each host plays dual roles: requesting service and replying with service. Dialogue control determining which role they are playing at any given moment.
  - Agree a set of guidelines to use during the communicating with each other.
- Dialog separation:
  - Order to Initiate, terminate and manage of communication.
  - Using a checkpoint to synchronize.

Dialog control



TWA

N = packets

1

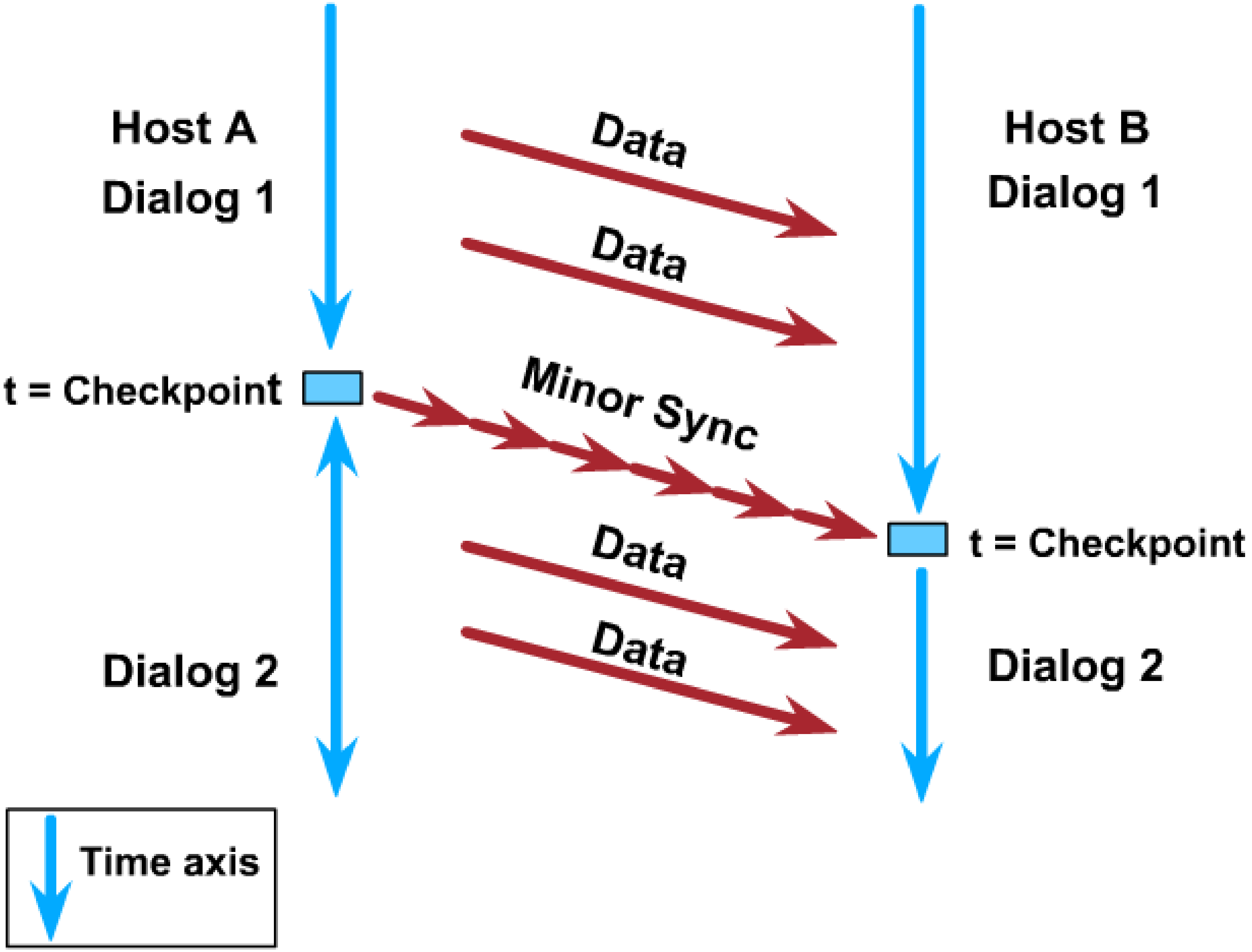
2

TWS

1

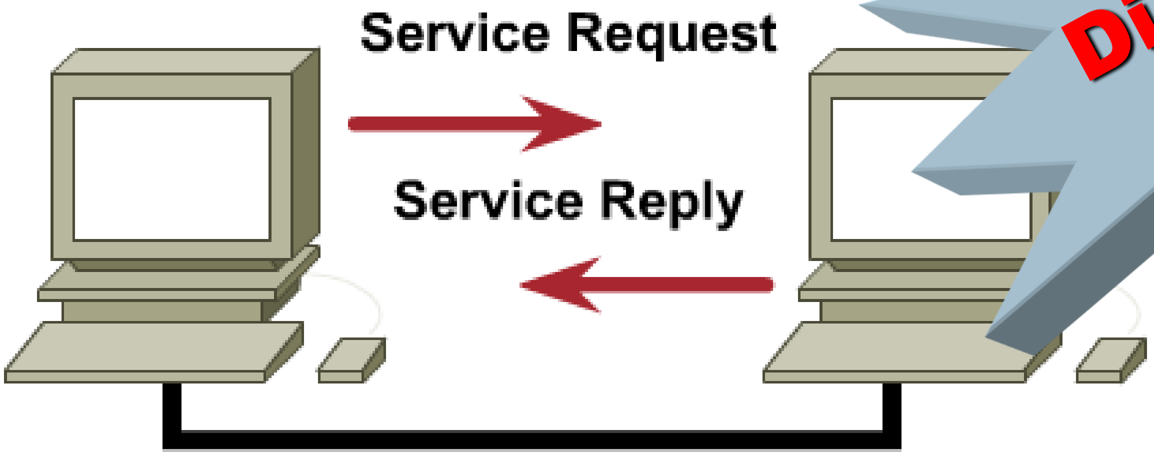
2

Dialog separation



Session layer protocols

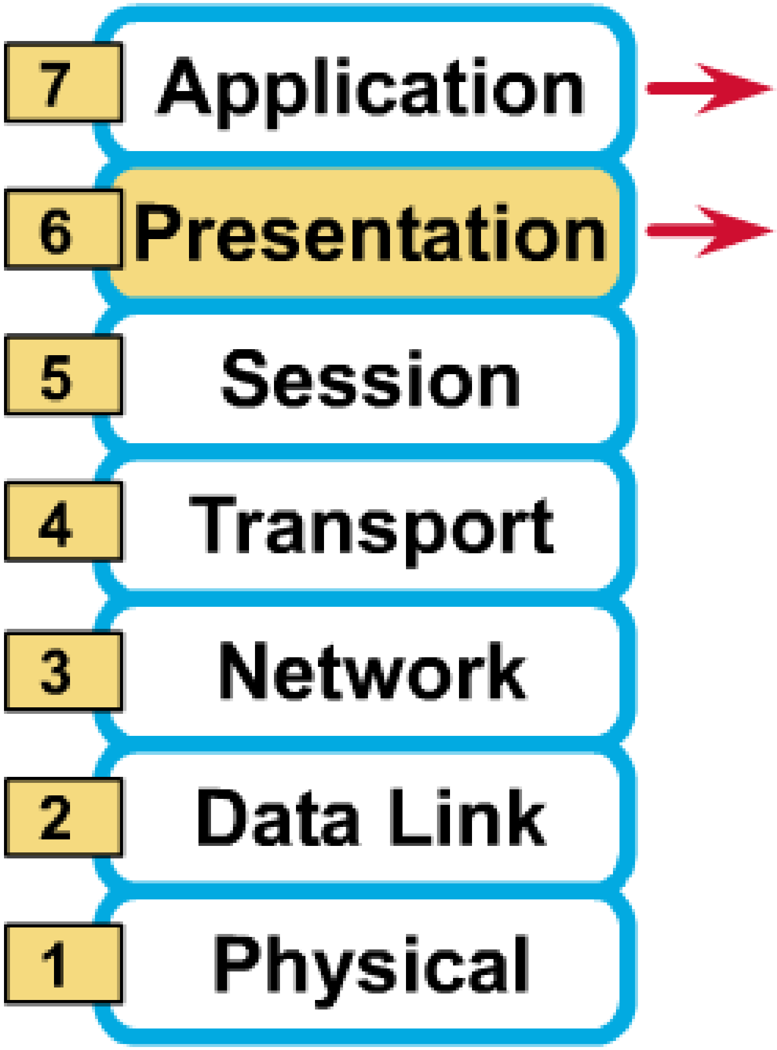
- Network File System (NFS)
- Structured Query Language (SQL)
- Remote-Procedure Call (RPC)
- X Window System
- AppleTalk Session Protocol (ASP)
- DNA Session Control Protocol (SCP



Discussion

PRESENTATION LAYER

Functions

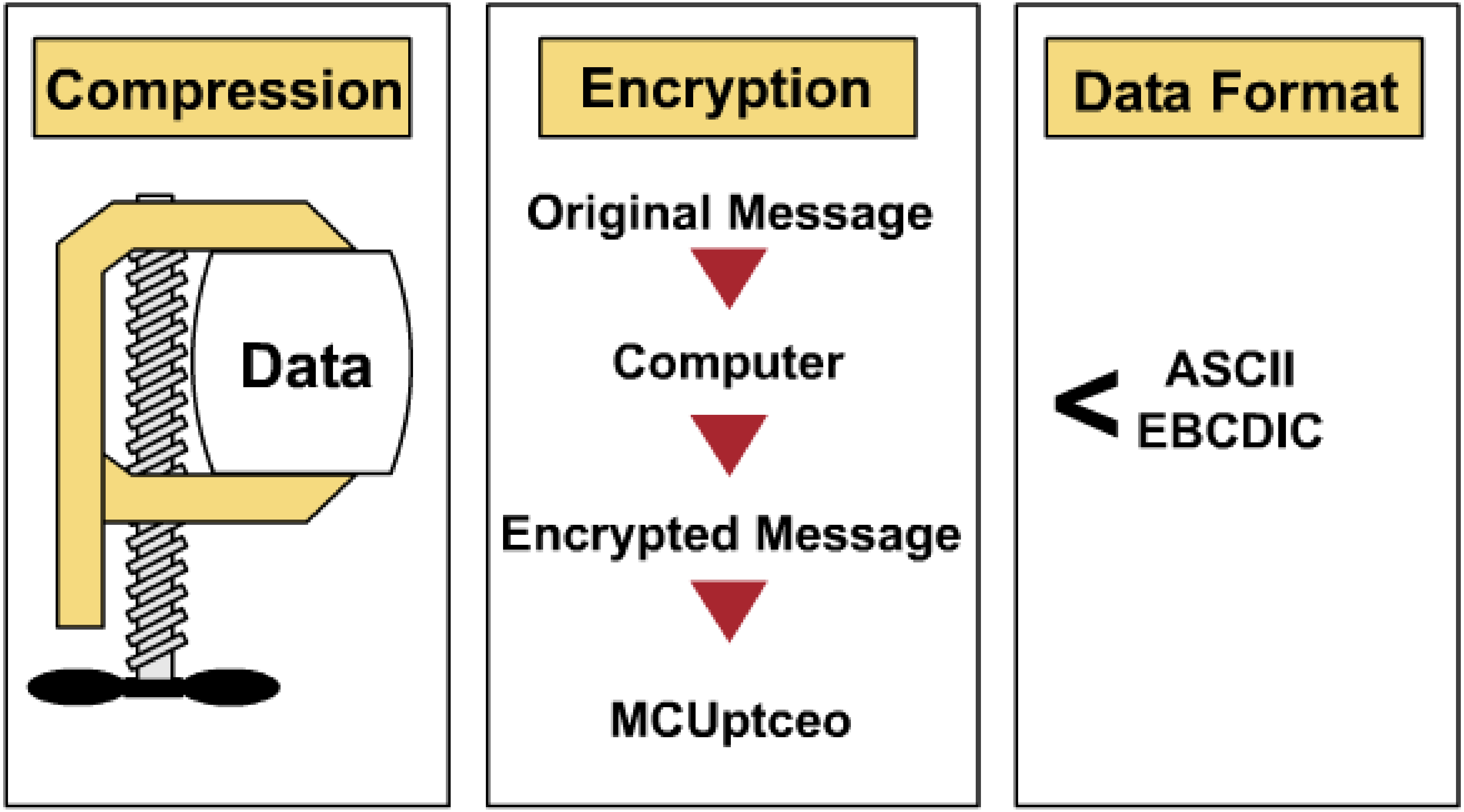


Network Processes to Applications

Data Representation

- Insure data is readable by receiving system
- Format of data
- Data structures
- Negotiates data transfer syntax for application layer

Three main functions



Compression

Encryption

Original Message

Computer

Encrypted Message

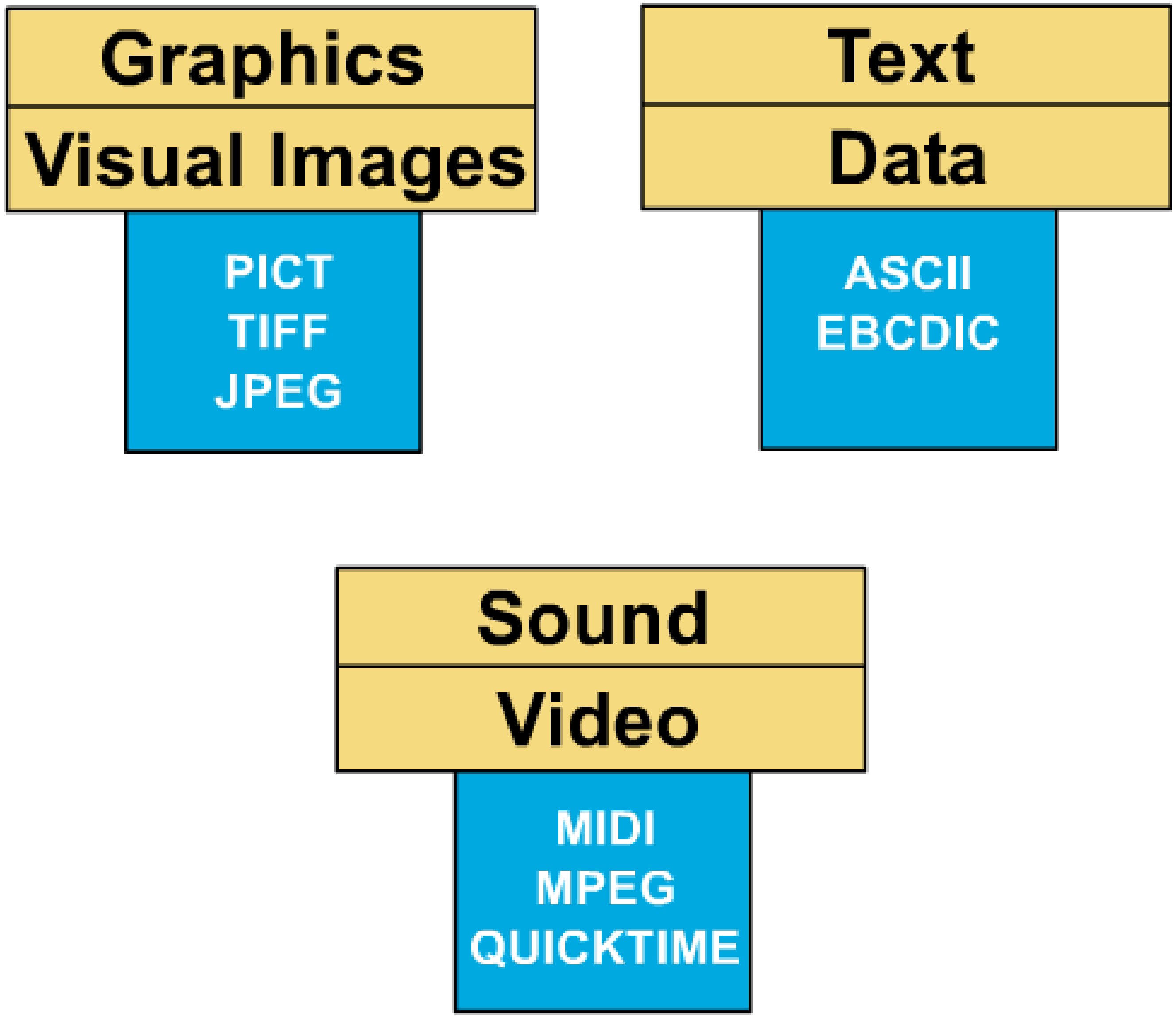
MCUptceo

Data Format

< ASCII
EBCDIC

The Saigon CTT

Data formatting



Graphics

Visual Images

PICT
TIFF
JPEG

Text

Data

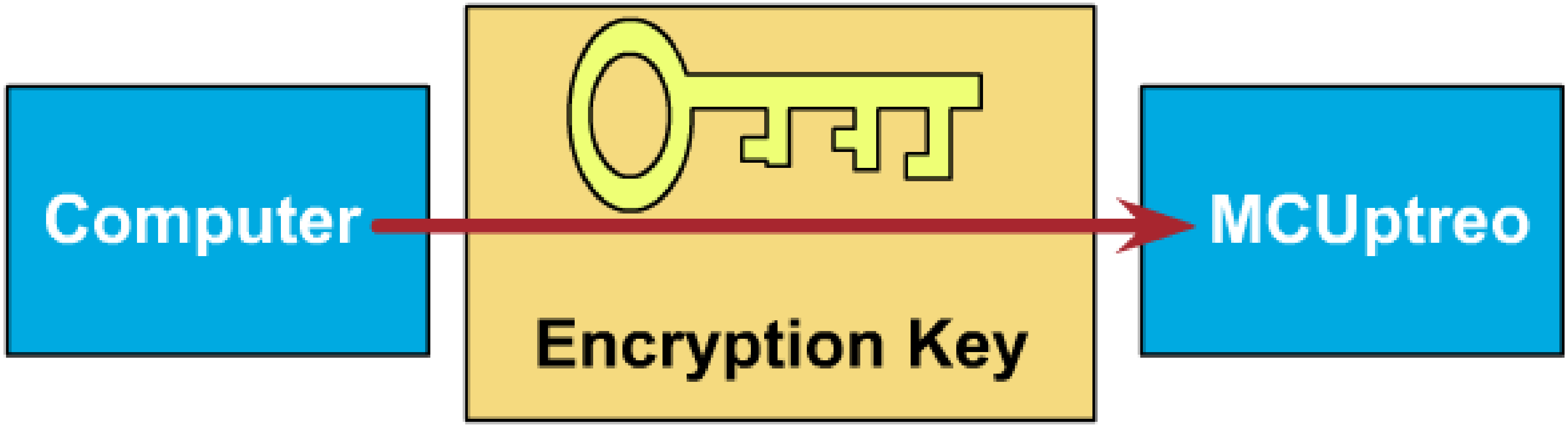
ASCII
EBCDIC

Sound

Video

MIDI
MPEG
QUICKTIME

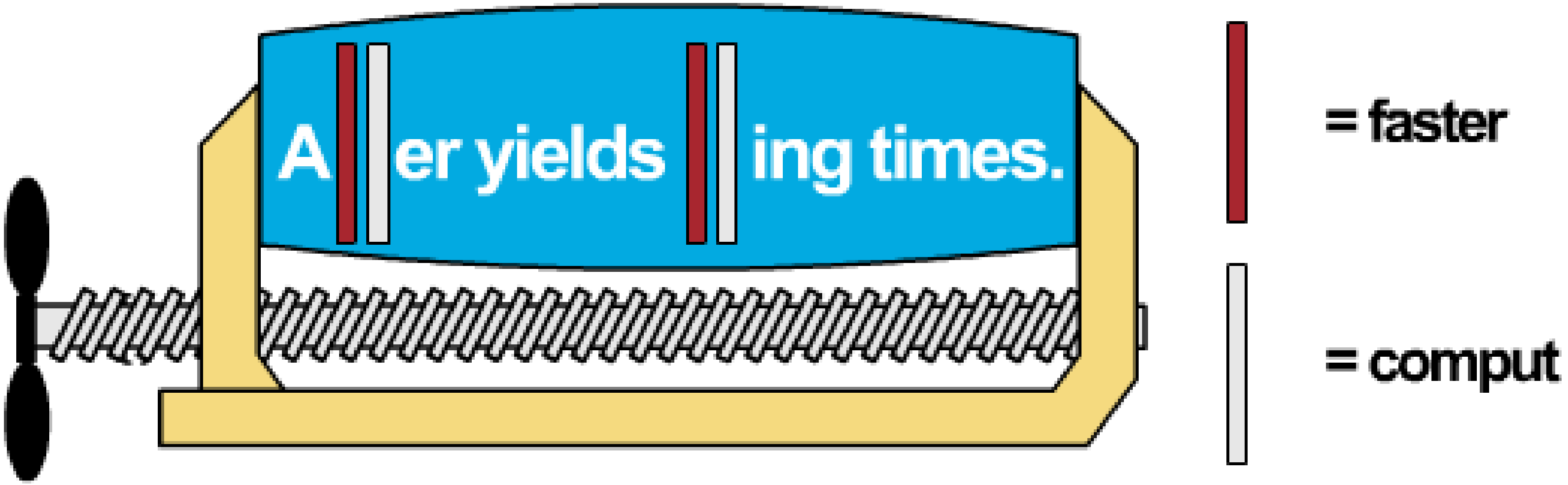
Data encryption



Data compression

A faster computer yields faster computing times.

Compressed


A
er yields
ing times.
= faster
= comput