

# **Transport Layer Protocols**

# Topics of Discussion

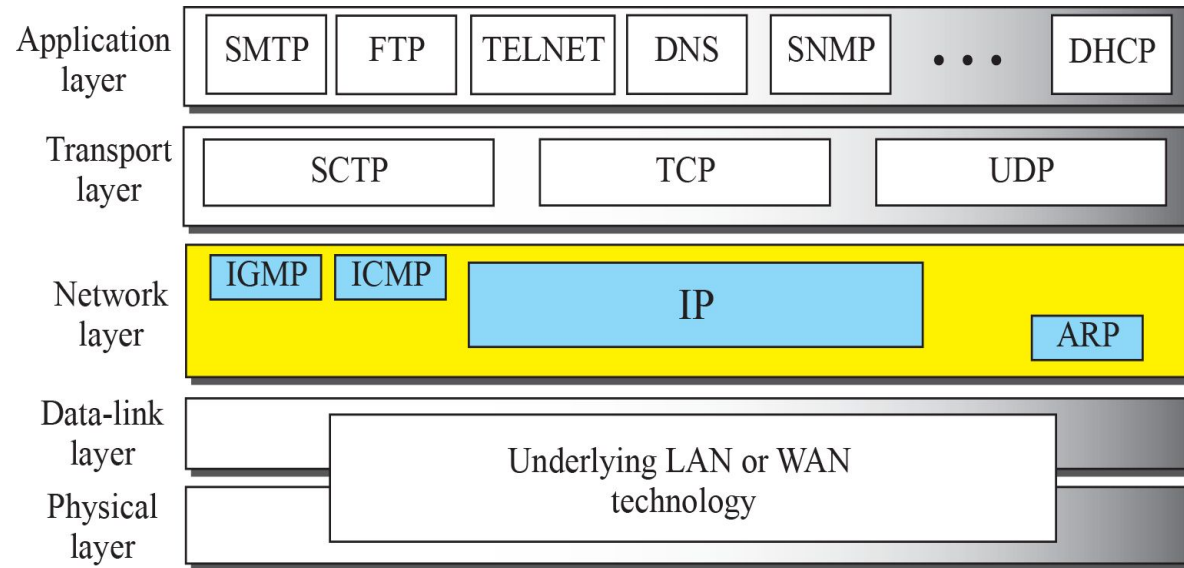
---

In this session we will learn the following

- ✓ UDP protocol
- ✓ TCP protocol
- ✓ SCTP protocol

# Transport Layer Protocols

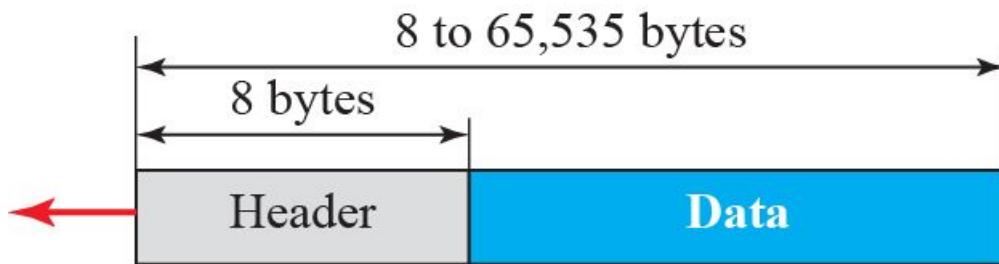
- A transport-layer protocol usually has several responsibilities.
- One is to create a **process-to-process** communication; these protocols use **port numbers** to accomplish this.
- Port numbers provide **end-to-end addresses** at the transport layer and allow multiplexing and demultiplexing at this layer, just as IP addresses do at the network layer.



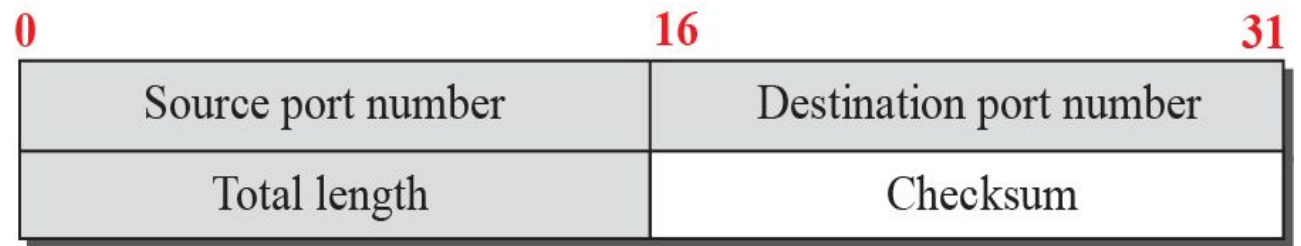
Port	Protocol	UDP	TCP	Description
7	Echo	√		Echoes back a received datagram
9	Discard	√		Discards any datagram that is received
11	Users	√	√	Active users
13	Daytime	√	√	Returns the date and the time
17	Quote	√	√	Returns a quote of the day
19	Chargen	√	√	Returns a string of characters
20, 21	FTP		√	File 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, 162	SNMP		√	Simple Network Management Protocol

# UDP protocol

- The User Datagram Protocol (UDP) is a **connectionless, unreliable transport protocol**.
- UDP is a very simple protocol using a minimum of overhead.
- UDP packets, called user datagrams, have a fixed-size header of 8 bytes made of four fields, each of 2 bytes (16 bits).



a. UDP user datagram

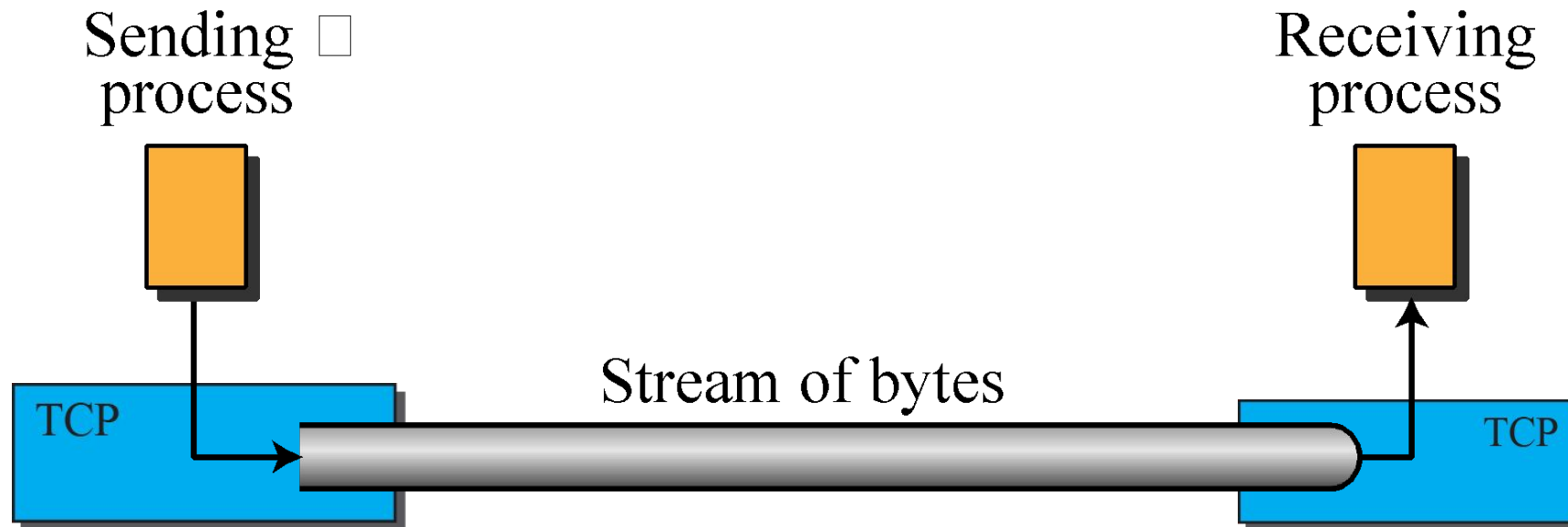


b. Header format

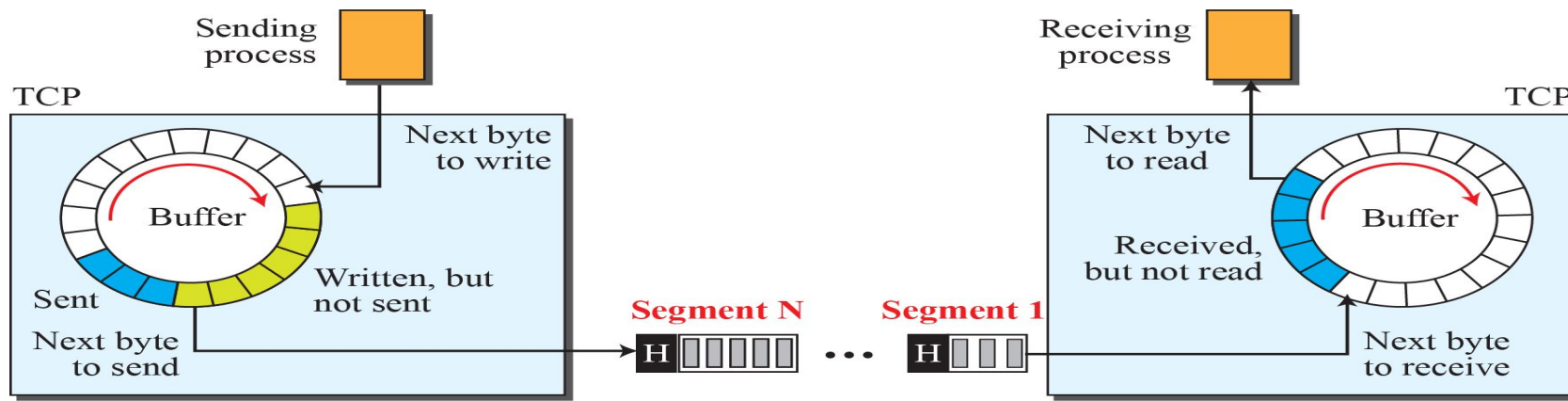
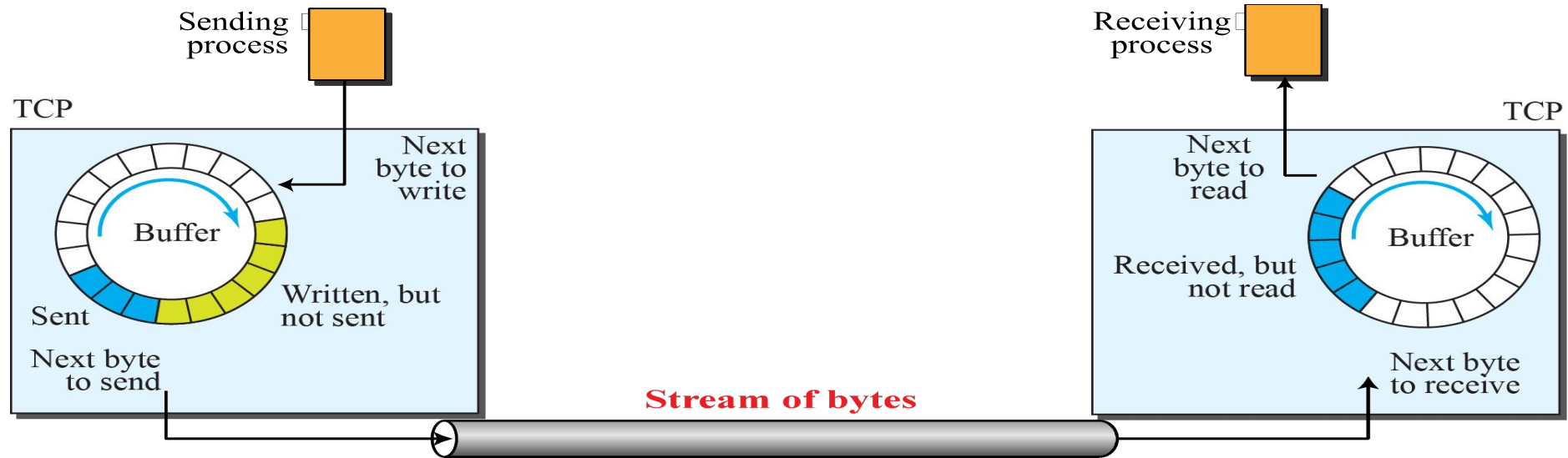
- The first two fields define the source and destination port numbers.
- The third field defines the total length of the user datagram, header plus data.
- The 16 bits can define a total length of 0 to 65,535 bytes.
- However, the total length needs to be less because a UDP user datagram is stored in an IP datagram with the total length of 65,535 bytes.
- The last field can carry the optional checksum (explained later).

# TCP Protocol

- Transmission Control Protocol (TCP) is a connection-oriented, reliable protocol.
- TCP explicitly defines connection establishment, data transfer, and connection teardown phases to provide a connection-oriented service.
- TCP uses a combination of GBN(Go-Back-N) and SR(Selective Repeat) protocols to provide reliability.

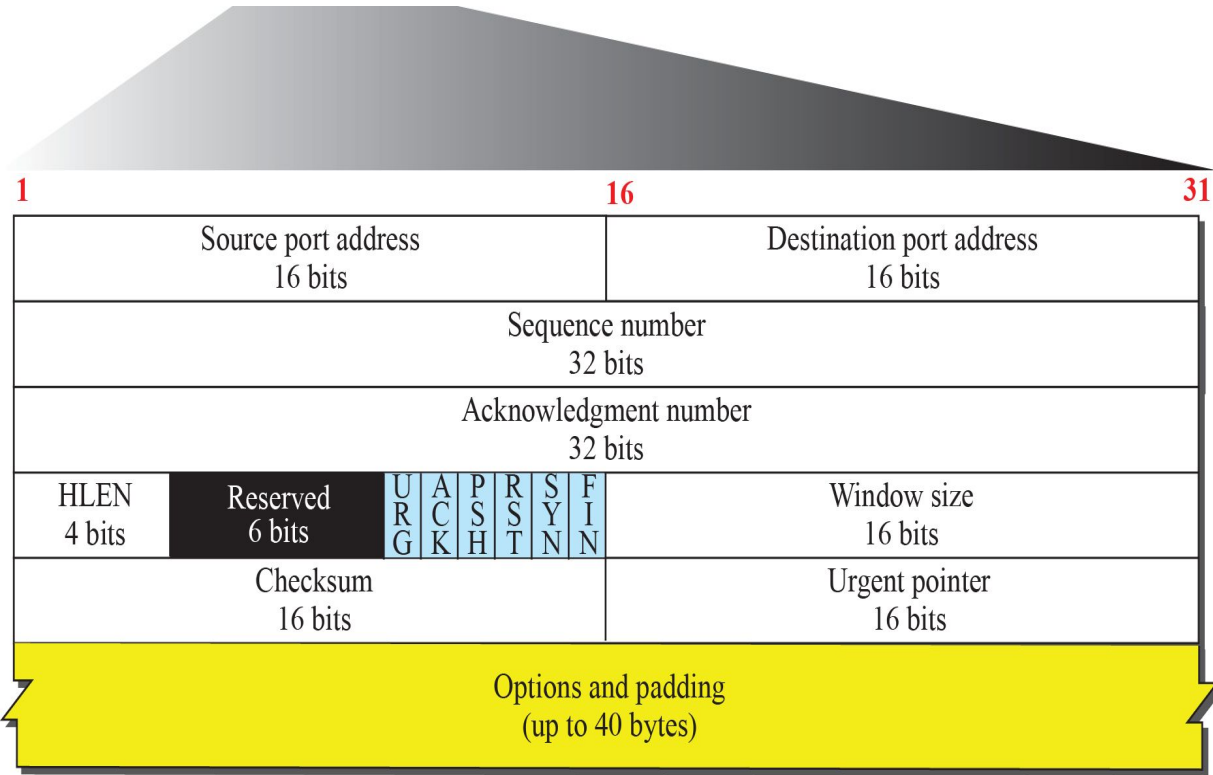
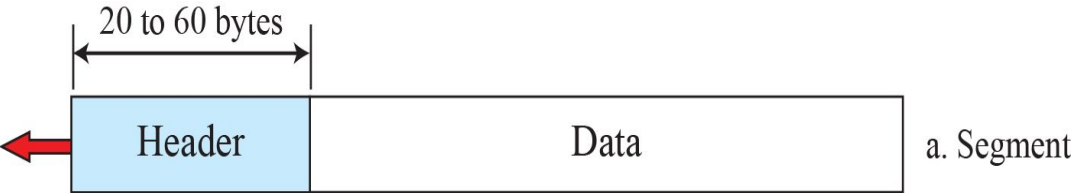


# Sending and Receiving Buffers

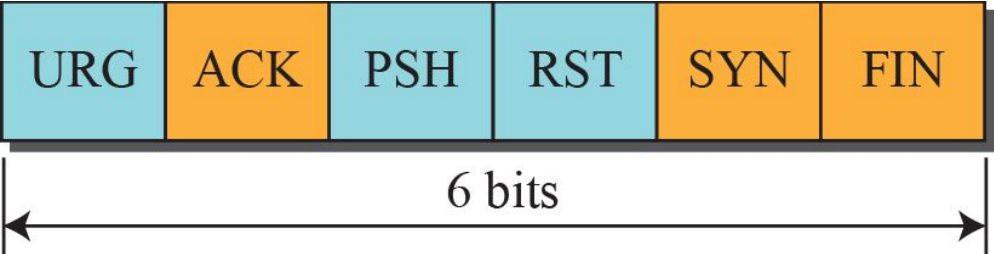


**TCP Segment**

# TCP Segment Format



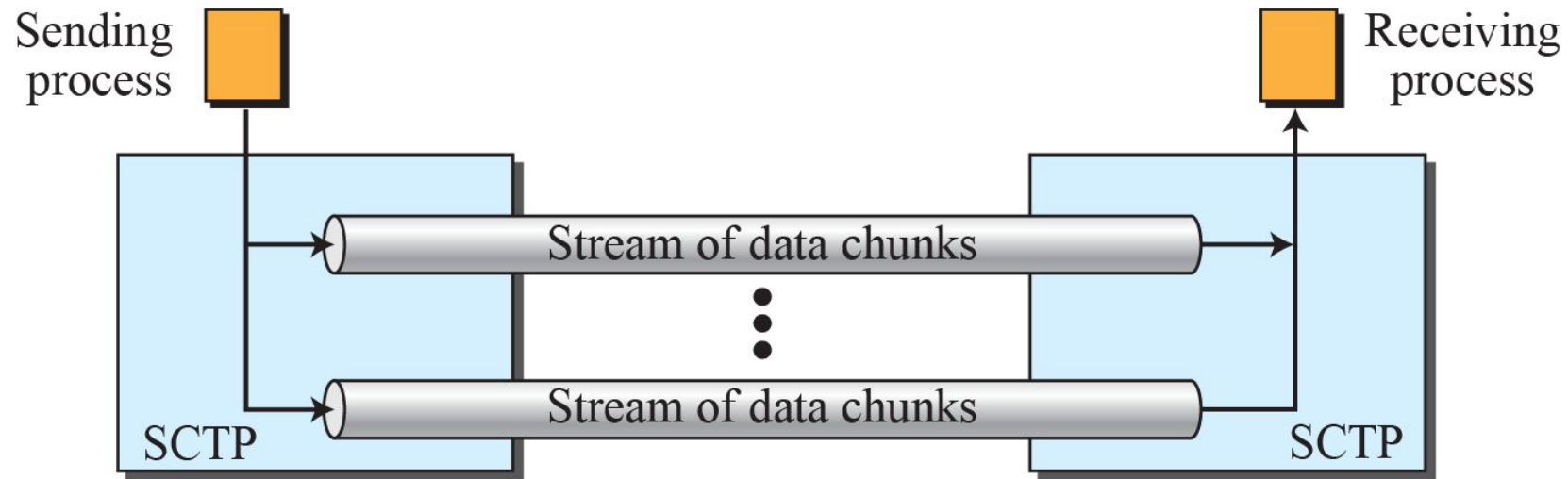
b. Header



- 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

# SCTP Protocol

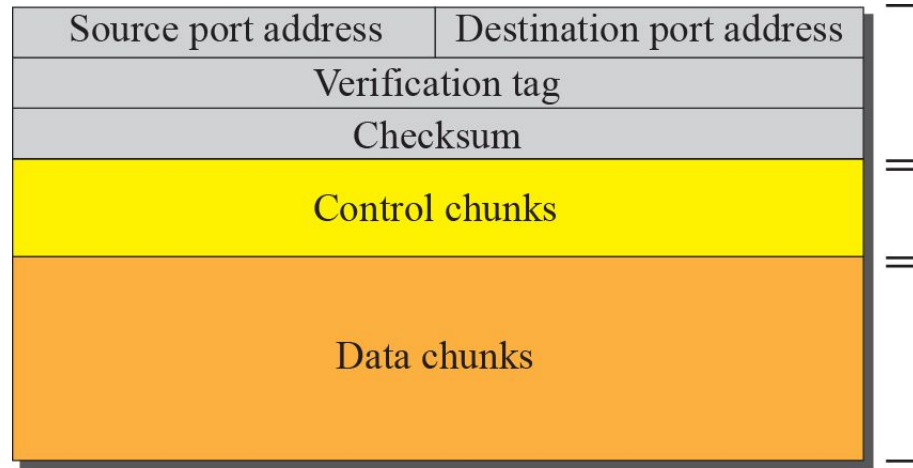
Stream Control Transmission Protocol (SCTP) is a new transport-layer protocol designed to combine some features of UDP and TCP in an effort to create a protocol for multimedia communication.



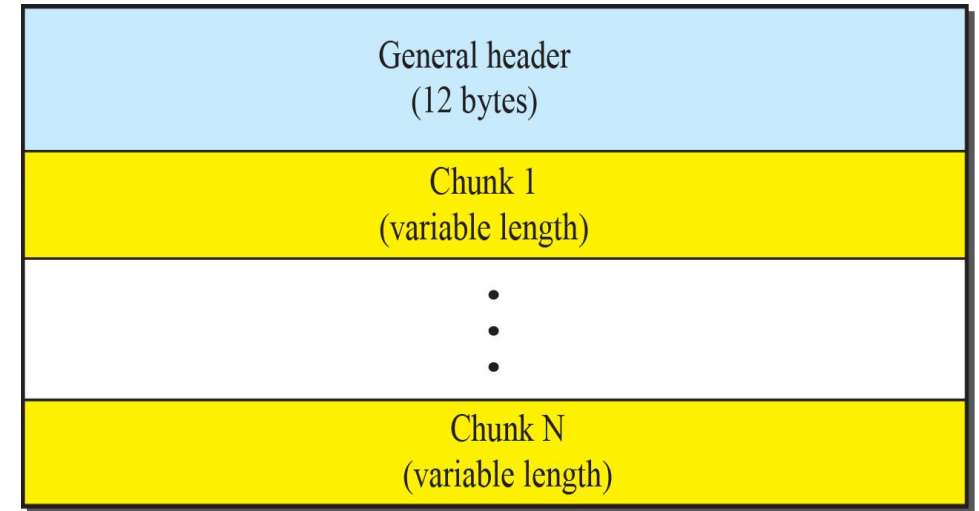
**Multiple-stream concept**



# SCTP Frame format and Chunks



A packet in SCTP



- An SCTP packet has a mandatory general header and a set of blocks called chunks.
- There are two types of chunks: **control chunks and data chunks**.
- A control chunk controls and maintains the association; a data chunk carries user data.
- In a packet, the control chunks come before the data chunks.
- Figure above shows the general format of an SCTP packet.

# Chunks

---

<i>Type</i>	<i>Chunk</i>	<i>Description</i>
0	DATA	User data
1	INIT	Sets up an association
2	INIT ACK	Acknowledges INIT chunk
3	SACK	Selective acknowledgment
4	HEARTBEAT	Probes the peer for liveliness
5	HEARTBEAT ACK	Acknowledges HEARTBEAT chunk
6	ABORT	Aborts an association
7	SHUTDOWN	Terminates an association
8	SHUTDOWN ACK	Acknowledges SHUTDOWN chunk
9	ERROR	Reports errors without shutting down
10	COOKIE ECHO	Third packet in association establishment
11	COOKIE ACK	Acknowledges COOKIE ECHO chunk
14	SHUTDOWN COMPLETE	Third packet in association termination
192	FORWARD TSN	For adjusting cumulating TSN

# Summary

---

In this section we have discussed the following:

- ✓ TCP, UDP and SCTP working and frame formats

Thank  
you!