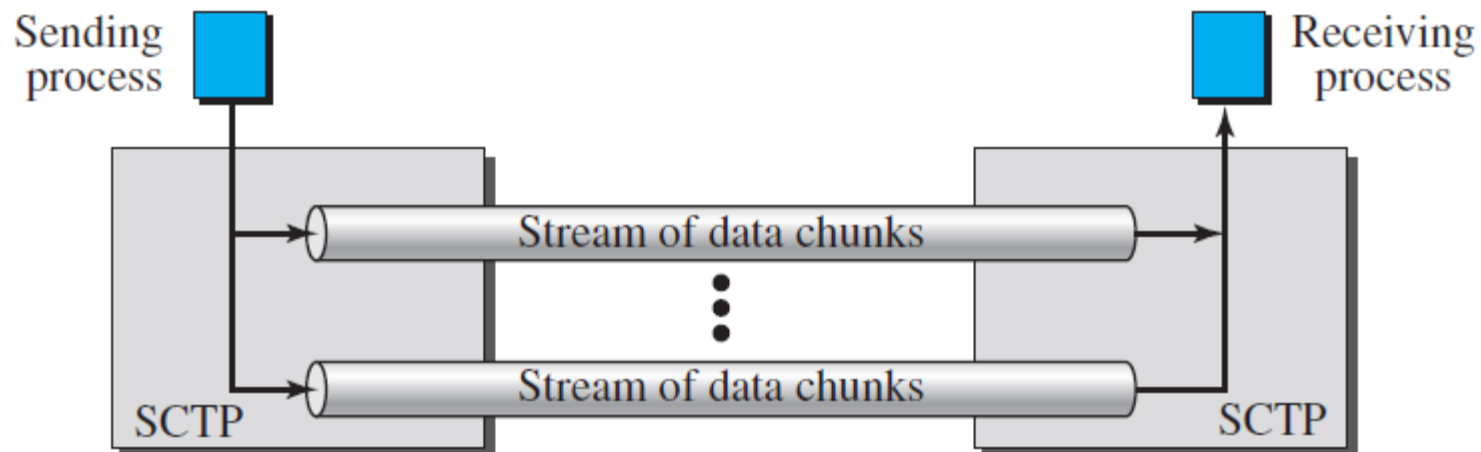


SCTP

Stream Control Transmission Protocol

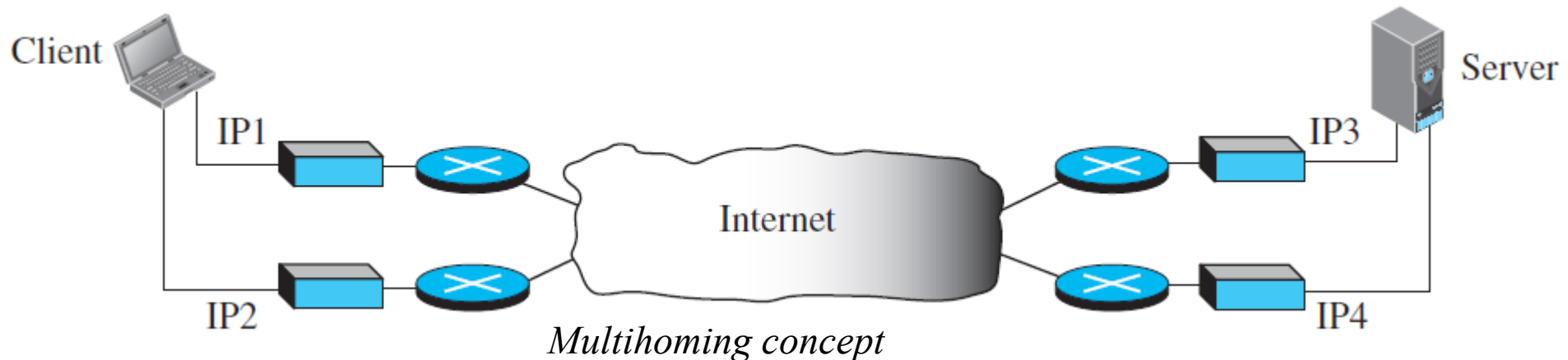
Stream Control Transmission Protocol (SCTP)

- A new transport-layer protocol
- Designed to **combine some features of UDP and TCP** in an effort to create a better protocol for multimedia communication.
- Provides ***process-to-process*** communication
- SCTP allows **multistream service** in each connection, which is called ***association***
- If one of the streams is blocked, the other streams can still deliver



SCTP

- An SCTP association supports multihoming service. The sending and receiving host can **define multiple IP addresses** in each end for an association.
- Helpful when we are sending and receiving a real-time payload
- SCTP offers **full-duplex service ; reliable transport protocol**. It uses an acknowledgment mechanism



SCTP Features

Transmission Sequence Number (TSN)

- The unit of data in SCTP is a data chunk, uses a transmission sequence number (TSN) to number them.
- TSNs are 32 bits long and randomly initialized between 0 and $2^{32} - 1$.
- Each data chunk must carry the corresponding TSN in its header.

Stream Identifier (SI)

- Each stream in SCTP needs to be identified using a stream identifier (SI).
- Each data chunk must carry the SI in its header so that when it arrives at the destination, it can be properly placed in its stream.
- The SI is a 16-bit number starting from 0.

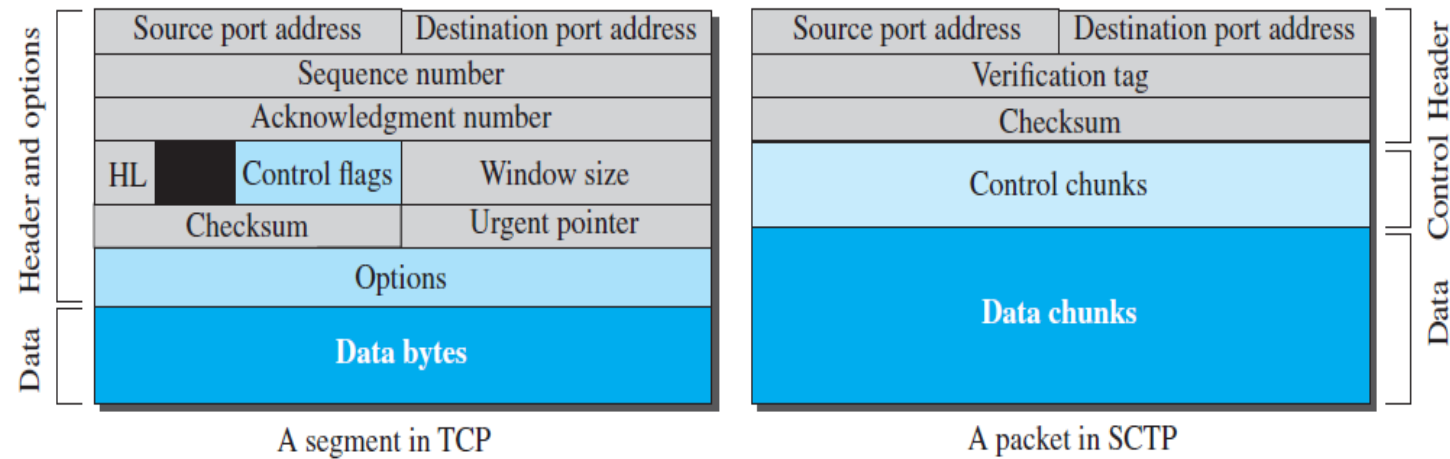
SCTP Features

Stream Sequence Number (SSN)

- When a data chunk arrives at the destination SCTP, it is delivered to the appropriate stream and in the proper order.
- In addition to an SI, SCTP defines each data chunk in each stream with a stream sequence number (SSN).

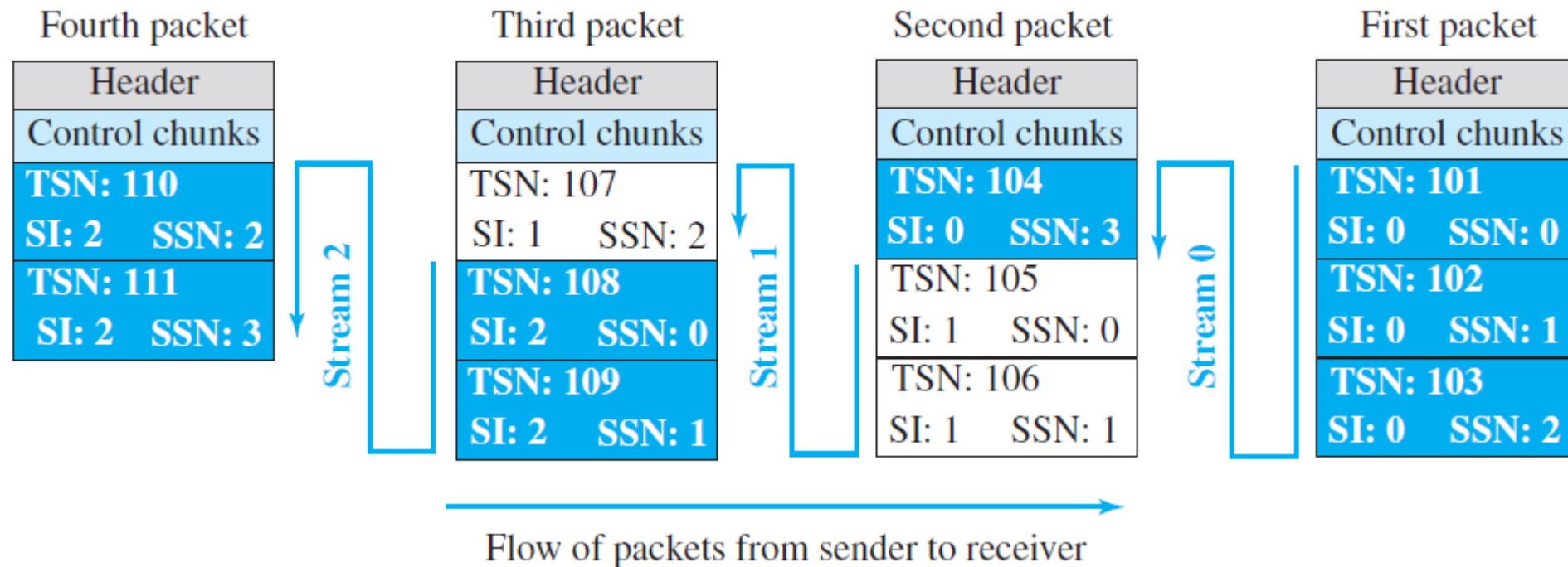
Packets

- Data are carried as data chunks, control information as control chunks.
- Several control chunks and data chunks can be packed together in a packet.
- A packet in SCTP plays the same role as a segment in TCP



Packets

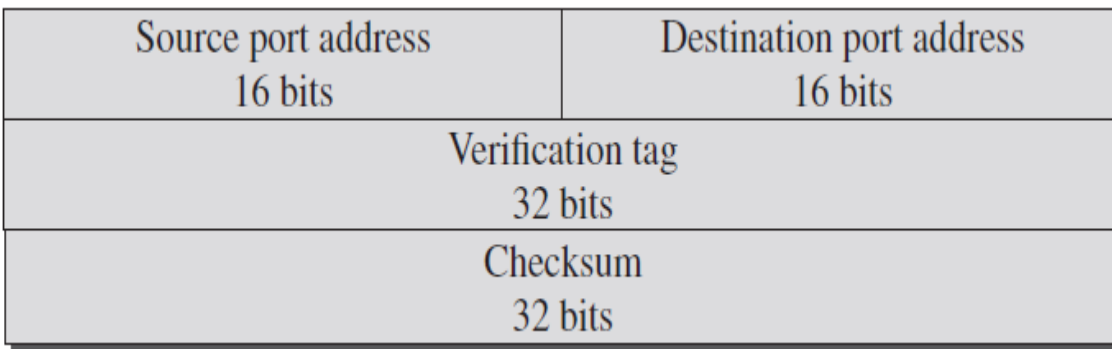
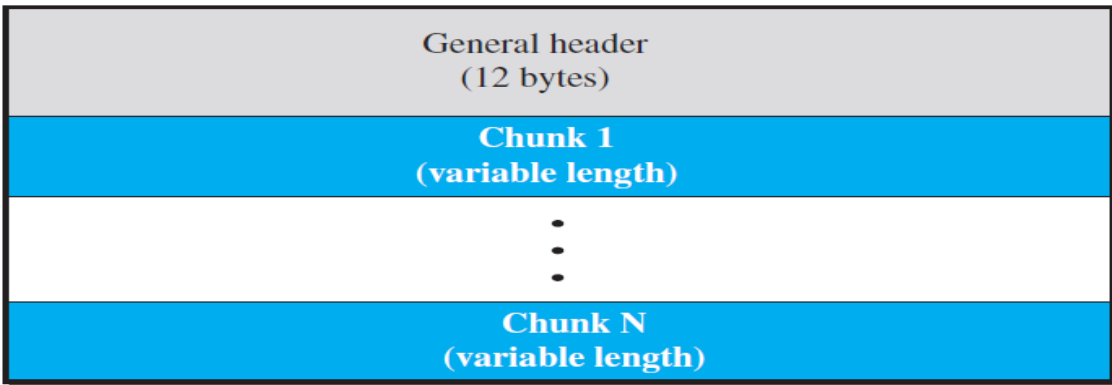
- An association may send many packets, a packet may contain several chunks, and chunks may belong to different streams.



Acknowledgment Number

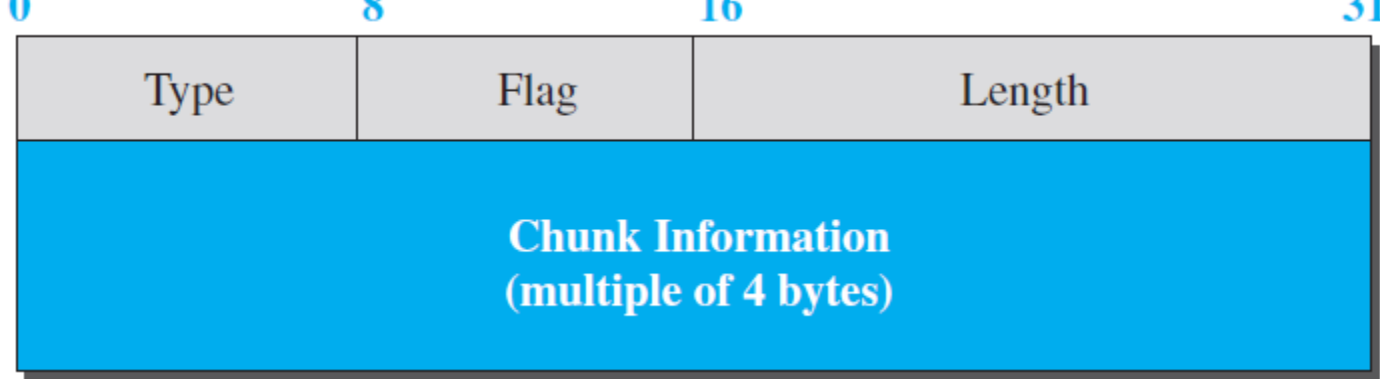
- SCTP acknowledgment numbers are chunk-oriented. They refer to the TSN.
- In SCTP, the control information is carried by control chunks, which do not need a TSN.
- These control chunks are acknowledged by another control chunk of the
- appropriate type (some need no acknowledgment).
- For example, an INIT control chunk is acknowledged by an INIT-ACK chunk.

Packet Format



- The general header (packet header) defines the end points of each association
- The verification tag is a 32-bit field that matches a packet to an association.
- It serves as an identifier for the association
- SCTP uses the CRC-32 checksum

Chunks



- The type field can define up to 256 types of chunks.
- The flag field defines special flags that a particular chunk may need.
- The length field defines the total size of the chunk, in bytes, including the type, flag, and length fields.
- If a chunk carries no information, the value of the length field is 4 (4 bytes).
- If the value is not a multiple of 4, the receiver knows there is padding.

Types of Chunks

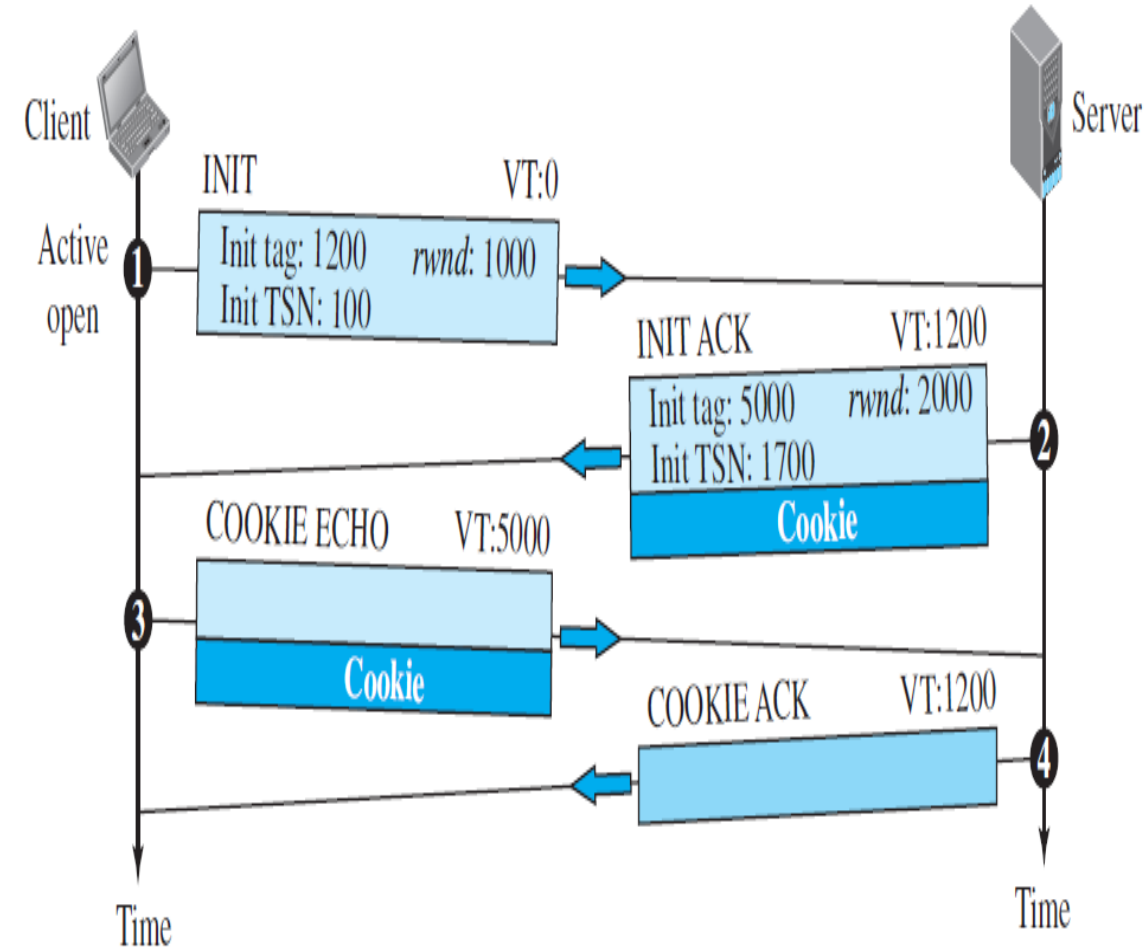
<i>Type</i>	<i>Chunk</i>	<i>Description</i>
0	DATA	User data
1	INIT	Sets up an association
2	INIT ACK	Acknowledges INIT chunk
3	SACK	Selective acknowledgment
4	HEARTBEAT	Probes the peer for liveliness
5	HEARTBEAT ACK	Acknowledges HEARTBEAT chunk
6	ABORT	Aborts an association
7	SHUTDOWN	Terminates an 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

SCTP Association

- A connection in SCTP is called an association.

Association Establishment

- Requires a four-way handshake.
- A process, normally a client, wants to establish an association with another process, normally a server, using SCTP as the transport-layer protocol.
- The SCTP server needs to be prepared to receive any association (passive open).
- Association establishment, however, is initiated by the client (active open).



Data Transfer

- SCTP supports piggybacking
- SCTP, recognizes and maintains boundaries.
- Each message coming from the process is treated as one unit and inserted into a DATA chunk .
- A message received from a process becomes a DATA chunk, or chunks if fragmented, by adding a DATA chunk header to the message.
- Each DATA chunk formed by a message or a fragment of a message has one TSN.
- Only DATA chunks use TSNs and only DATA chunks are acknowledged by SACK chunks

Multihoming Data Transfer

- Only one of multiple addresses can be defined as the primary Address; the rest are alternative addresses.
- The primary address is defined during association establishment.
- A source defines the primary address for a destination.
- Data transfer, uses the primary address of the destination.
- If the primary is not available, one of the alternative addresses is used.

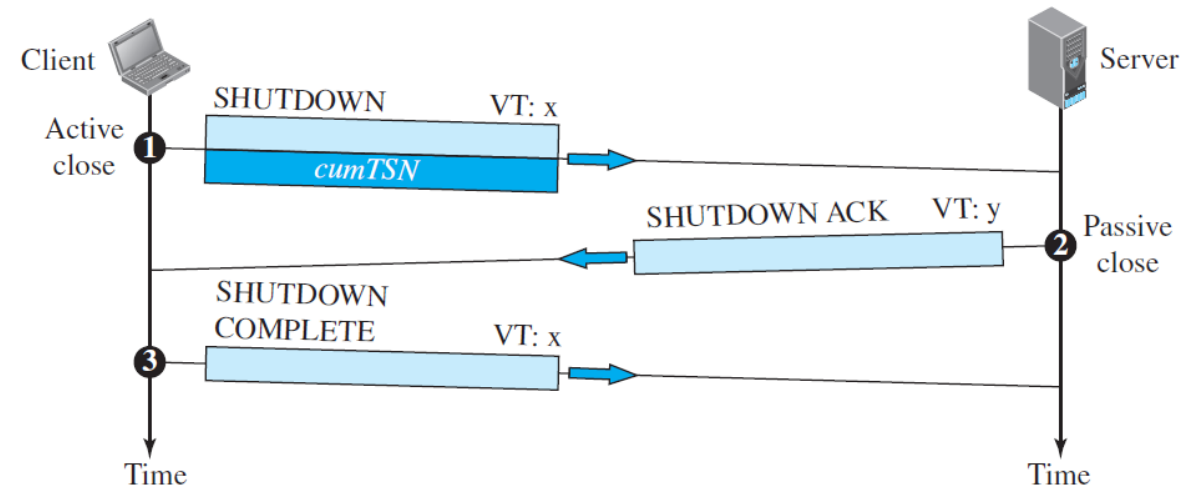
Multistream Delivery

- SCTP supports two types of data delivery in each stream: ordered (default) and unordered.
- In ordered data delivery, data chunks in a stream use stream sequence numbers (SSNs) to define their order in the stream.
- When the chunks arrive at the destination, SCTP is responsible for message delivery according to the SSN defined in the chunk.
- This may delay the delivery because some chunks may arrive out of order.
- In unordered data delivery, the data chunks in a stream have the U flag set, but their SSN field value is ignored.
- They do not consume SSNs.
- When an unordered data chunk arrives at the destination SCTP, it delivers the message carrying the chunk to the application without waiting for the other messages.

Fragmentation

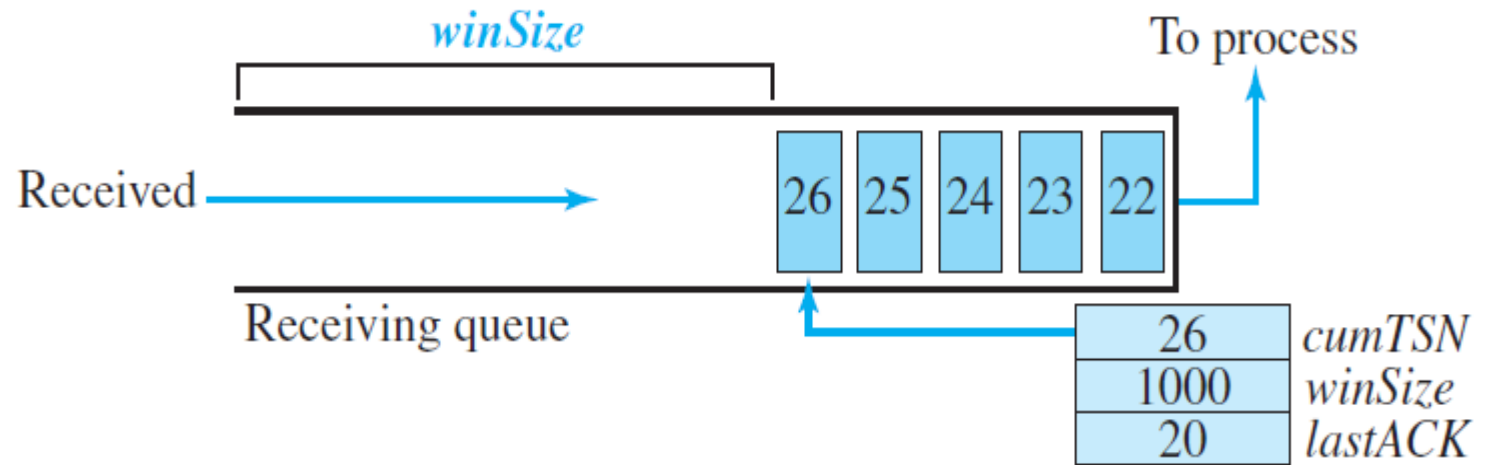
- SCTP preserves the boundaries of the message from process to process when creating a DATA chunk from a message if the size of the message does not exceed the MTU of the path.
- The size of an IP datagram carrying a message can be determined by
 - adding the size of the message, in bytes,
 - data chunk header, necessary SACK chunks, SCTP general header,
 - IP header.
- If the total size exceeds the MTU, the message needs to be fragmented.

Association Termination



- Either of the two parties involved in exchanging data can close the connection.
- SCTP does not allow a “halfclosed” association.
- If one end closes the association, the other end must stop sending new data.
- If any data are left over in the queue of the recipient of the termination request, they are sent and the association is closed.
- Association termination uses three packets,

Flow Control



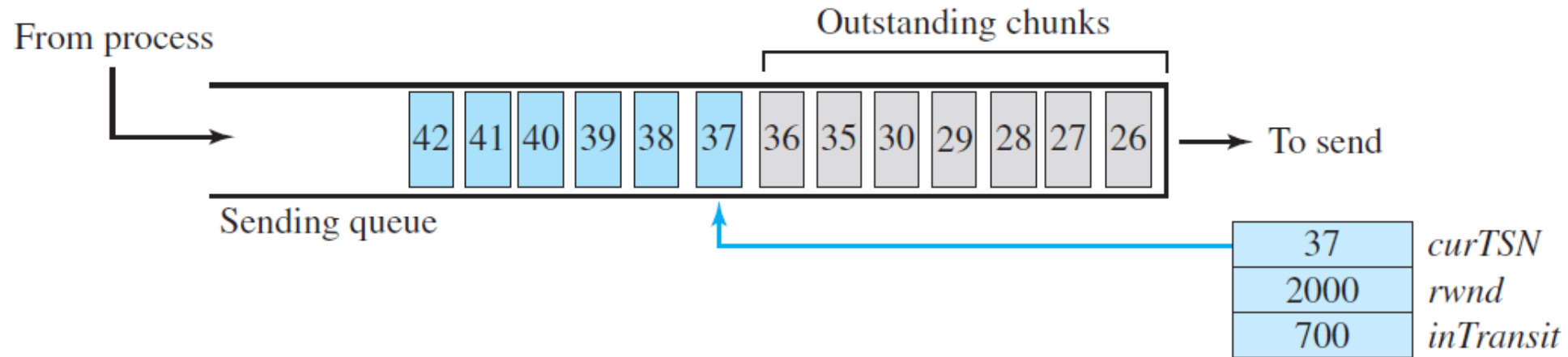
Receiver Site

- The receiver has one buffer and 3 variables.
- The queue holds the received data chunks that have not yet been read by the process.
- The first variable holds the last TSN received, *cumTSN*.
- The second variable holds the available buffer size, *winSize*.
- The third variable holds the last cumulative acknowledgment, *lastACK*.

Receiver Site

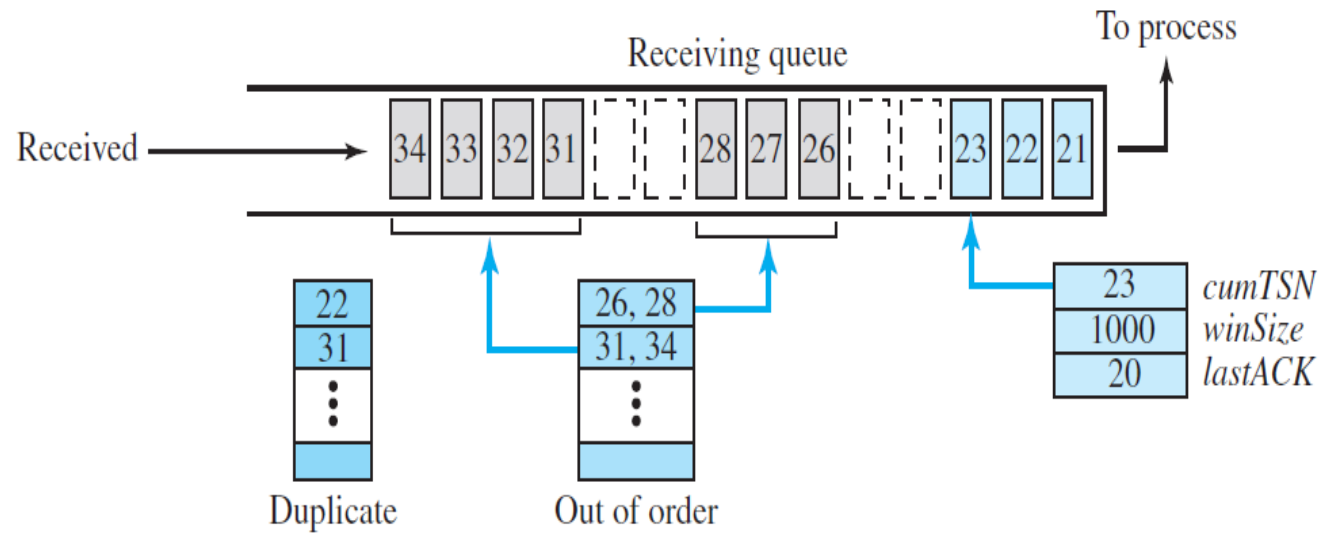
1. When the site receives a data chunk, it stores it at the end of the buffer (queue) and subtracts the size of the chunk from winSize.
 - The TSN number of the chunk is stored in the cumTSN variable.
2. When the process reads a chunk, it removes it from the queue and adds the size of the removed chunk to winSize
3. When the receiver decides to send a SACK, it checks the value of lastAck;
 - if it is less than cumTSN, it sends a SACK with a cumulative TSN number equal to the cumTSN.
 - It also includes the value of winSize as the advertised window size.
 - The value of lastACK is then updated to hold the value of cumTSN.

Sender Site



- The sender has one buffer and 3 variables: *curTSN*, *rwnd*, and *inTransit*,
- The buffer holds the chunks produced by the process that have either been sent or are ready to be sent.
- The first variable, *curTSN*, refers to the next chunk to be sent.
- All chunks in the queue with a TSN less than this value have been sent but not acknowledged;
- The second variable, *rwnd*, holds the last value advertised by the receiver (in bytes).
- The third variable, *inTransit*, holds the number of bytes in transit, bytes sent but not yet acknowledged.

Error Control



- It leaves spaces for any missing chunks.
- It discards duplicate messages

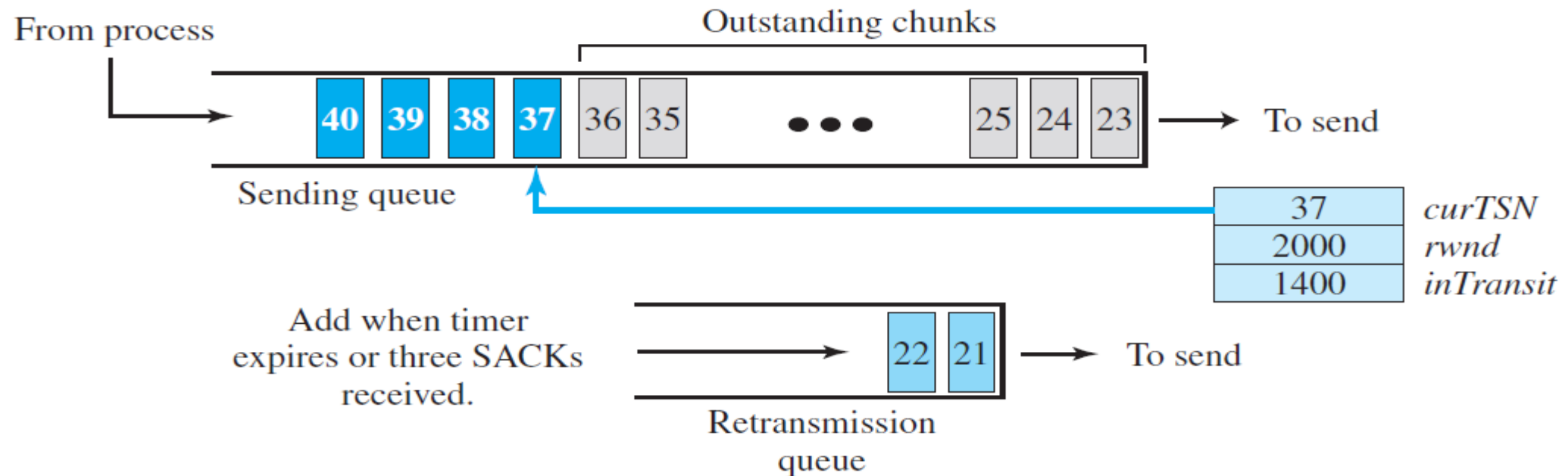
SACK chunk

Type: 3	Flag: 0	Length: 32
Cumulative TSN: 23		
Advertised receiver window credit: 1000		
Number of gap ACK blocks: 2		Number of duplicates: 2
Gap ACK block #1 start: 3		Gap ACK block #1 end: 5
Gap ACK block #2 start: 8		Gap ACK block #2 end: 11
Duplicate TSN: 22		
Duplicate TSN: 31		

Numbers are
relative to
cumTSN

Sender Site

- Sender site, demands two buffers : a sending queue and a retransmission queue. And 3 variables: **rwnd**, **inTransit**, and **curTSN**



Summary

Learnt about

- SCTP features
- Connection establishment
- Flow control
- Error control