ulm university universität ulm

Prof. Dr. Hans Peter Großmann mit M. Rabel sowie
H. Hutschenreiter und T. Nau | Sommersemester 2012 |
Institut für Organisation und Management von
Informationssystemen

Thomas Nau, kiz

Lecture Computer Networks

TCP and UDP

TCP and UDP

- Transmission Control Protocol (TCP)
  - Motivation and Overview
  - Header
  - Ports and User Processes
  - Connection orientation

- User Datagram Protocol (UDP)
  - Header

Motivation

We have

a connectionless and unreliable packet delivery system:

IP (Internet Protocol)

Overview

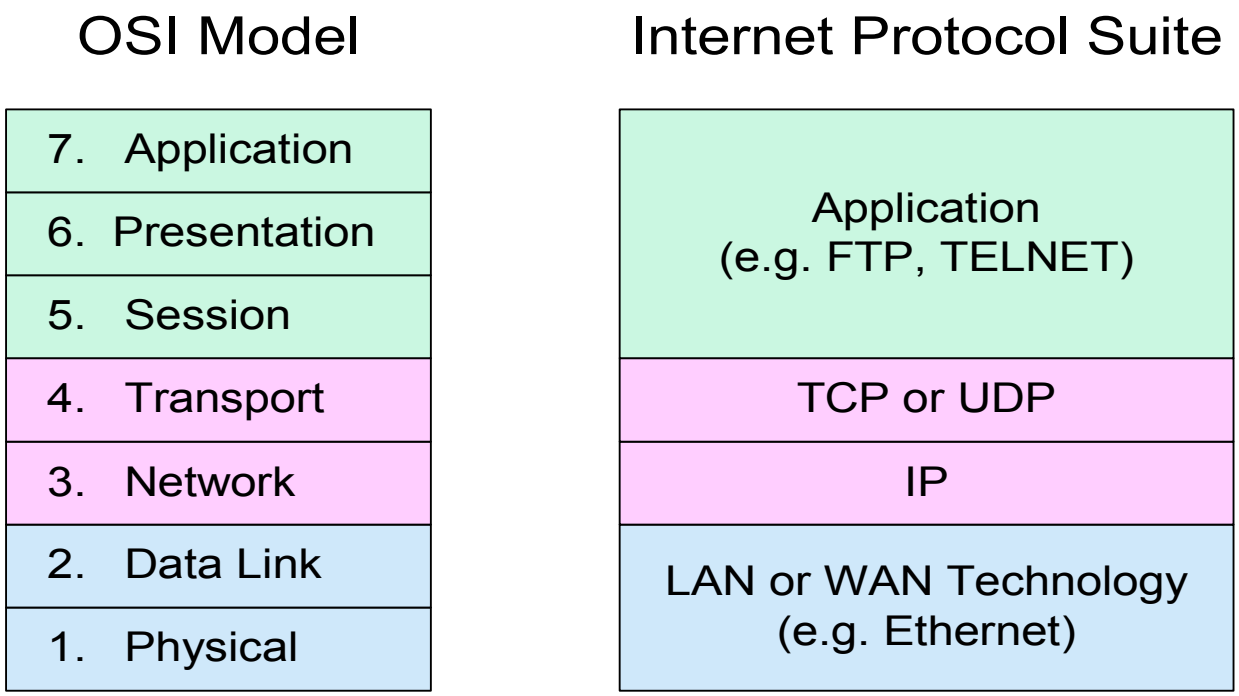
TCP

provides a connection-oriented, reliable, full-duplex, byte-stream service to an application program

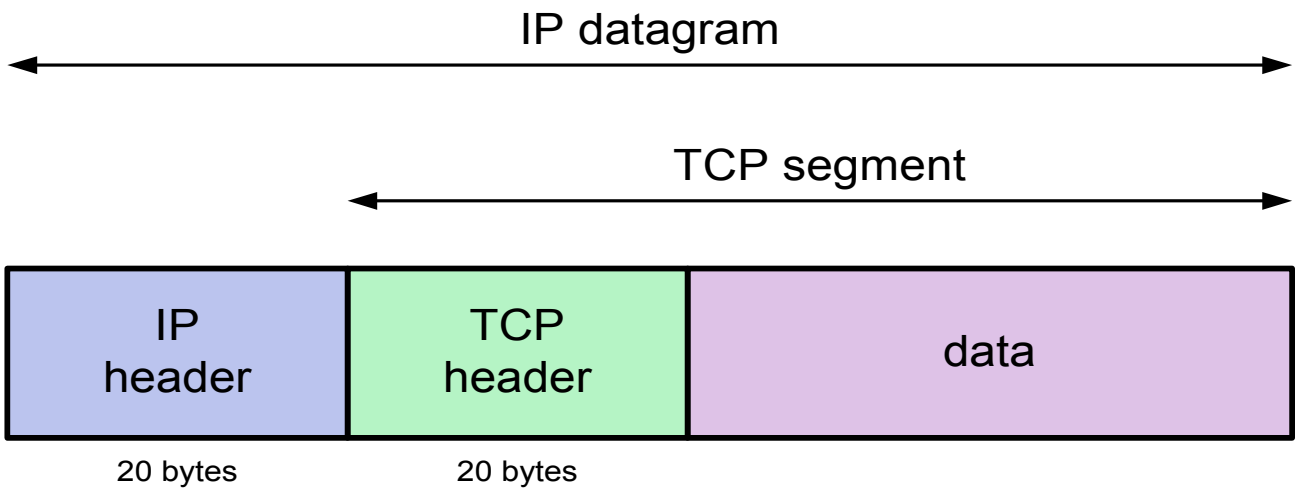
UDP

provides a connectionless, unreliable datagram service

OSI Model

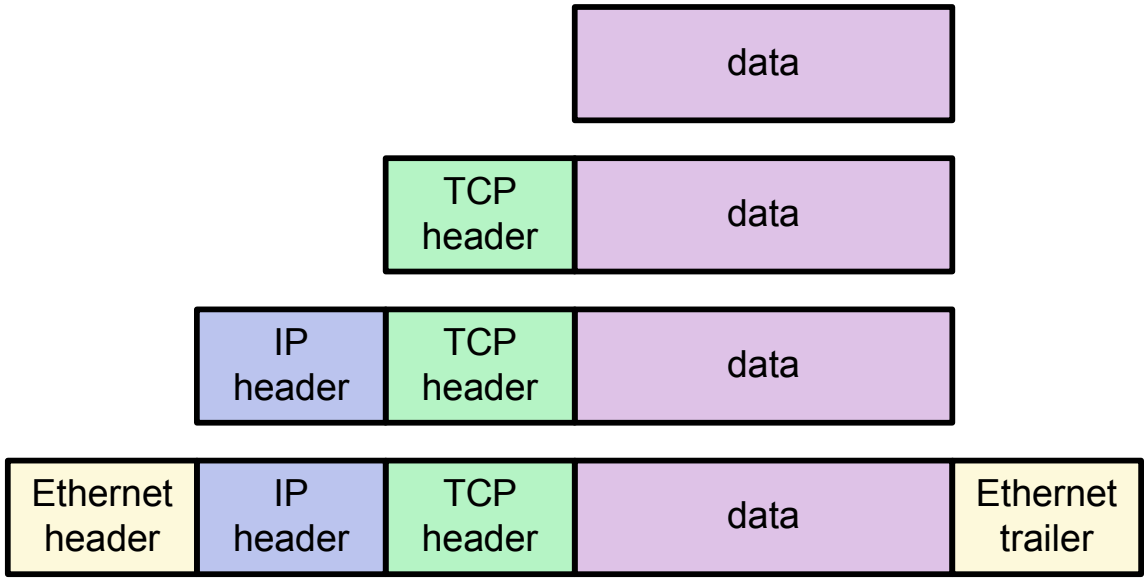

OSI Model
7. Application
6. Presentation
5. Session
4. Transport
3. Network
2. Data Link
1. Physical
Internet Protocol Suite
Application
(e.g. FTP, TELNET)
TCP or UDP
IP
LAN or WAN Technology
(e.g. Ethernet)

TCP Segment



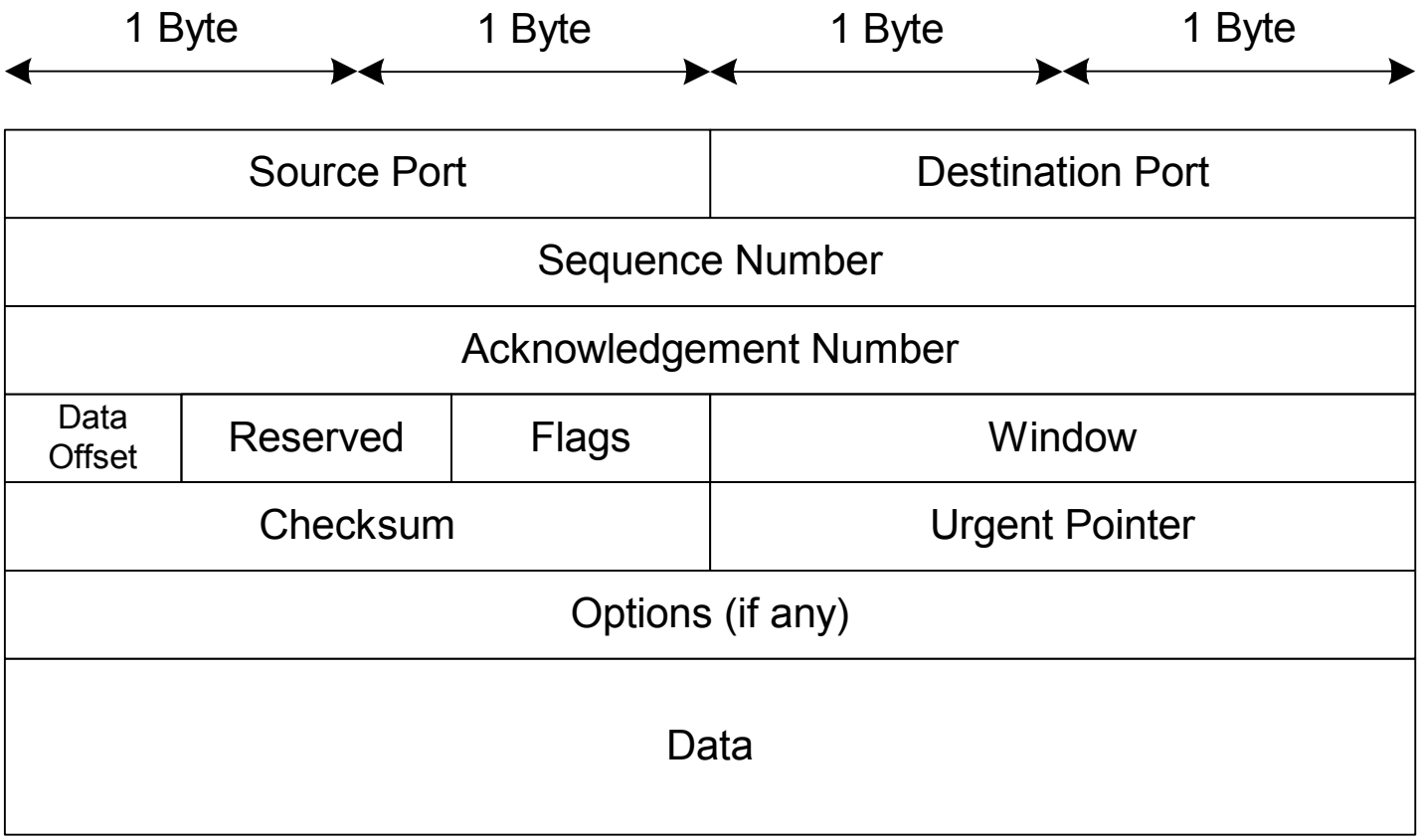
TCP divides the application data into packets, so that they have the best size to send.

Encapsulation



Example for a TCP Encapsulation

TCP Header


1 Byte
1 Byte
1 Byte
1 Byte
Source Port
Destination Port
Sequence Number
Acknowledgement Number
Data Offset
Reserved
Flags
Window
Checksum
Urgent Pointer
Options (if any)
Data

Ports, Data Offset and Checksum

- Source Port (16 bits): The source port number.
- Destination Port (16 bits): The destination port number.
- Data Offset (4 bits): The number of 32-bit words in the TCP Header
  - This indicates where the data begins.
  - The TCP header length (even one including options) is an integral number of 32 bits.
  - (minimum header length: 20 bytes, maximum header length: 60 bytes)
- Checksum (16 bits): The checksum of header and data

Sequence Number and Acknowledgement Number

- Sequence Number (32 bits):
  - The sequence number of the first data byte in this segment.
  - (IP datagrams and with them the TCP segments can arrive out of order and TCP must resequence the data if necessary)

- Acknowledgment Number (32 bits):

  If the ACK control bit is set, this field contains the value of the next sequence number, which the sender of the acknowledgement is expecting to receive. Once a connection is established this number is always sent.

  = Sequence Number (of the last successfully received byte of data) + 1

Flags

- Reserved (6 bits):

  Reserved for future use. Must be zero.

- Control Bits (6 bits):

  From left to right

URG:	Urgent Pointer field is valid
ACK:	Acknowledgment field is valid
PSH:	Push Function (The receiver should pass this data to the application as soon as possible)
RST:	Reset the connection
SYN:	Synchronize sequence numbers (for connection establishment)
FIN:	The sender is finished sending data (for connection termination)

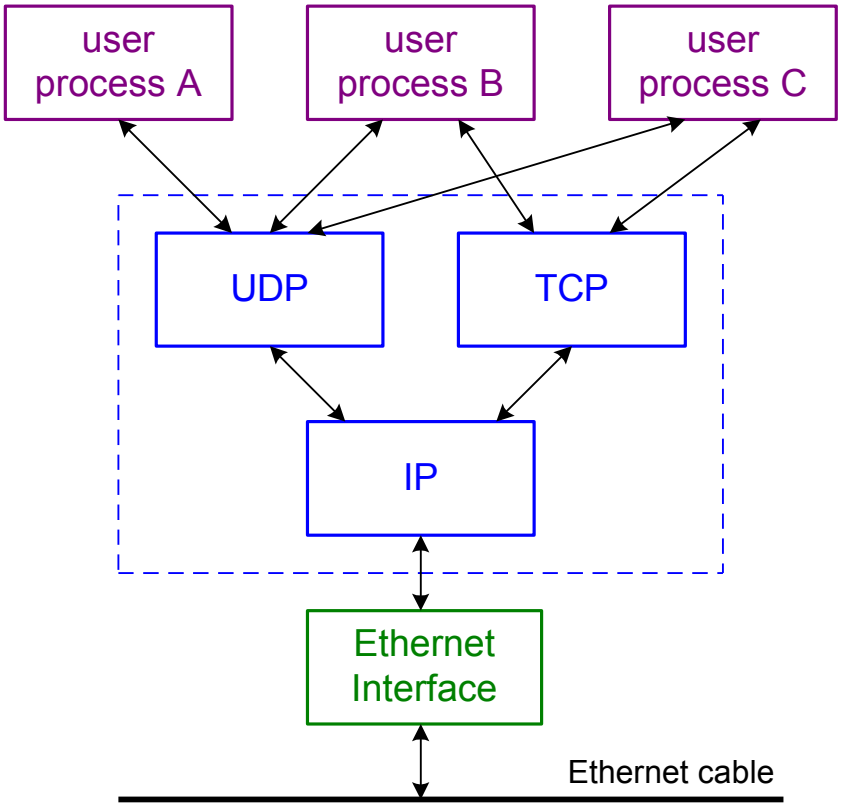
Port Numbers

At a time more than one user process can be using
TCP and / or UDP

- ⇨ a method for identifying the data associated with each user process is required

- ⇨ TCP and UDP use 16-bit integer *port numbers* for this identification

User Processes



Well-known Ports

When a client process wants to contact a server,
the client must know the port number of the server process

Therefore TCP and UDP have defined a group of *well-known ports*

Examples:	21	FTP (File Transfer Protocol)
	23	TELNET
	25	SMTP (Simple Mail Transfer Protocol)

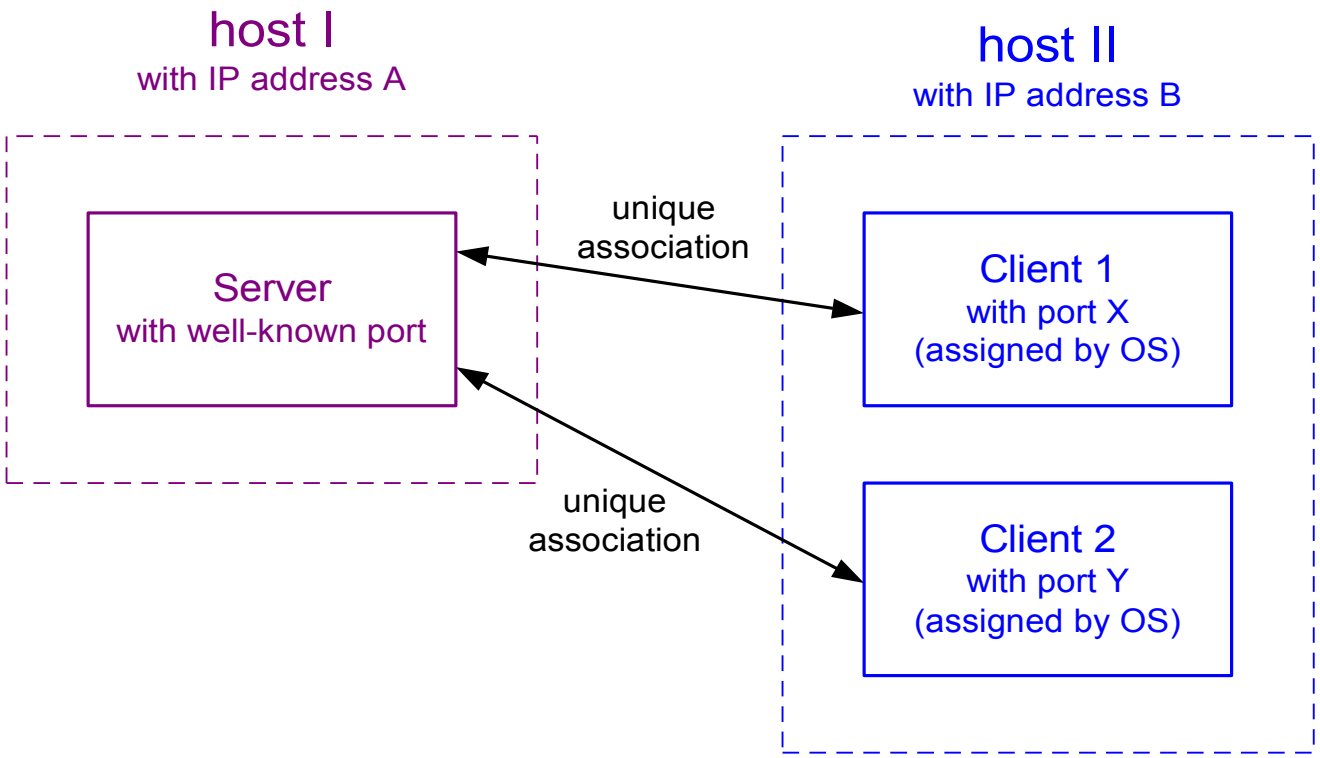
Defining a Process - Process Connection

Each connection from user process to user process is defined by the following combination:

- the protocol (UDP or TCP)
- the local host´s Internet address
- the local host´s port number
- the foreign host´s Internet address
- the foreign host´s port number

(The combination of an IP address and a port number is sometimes called a socket)

Example

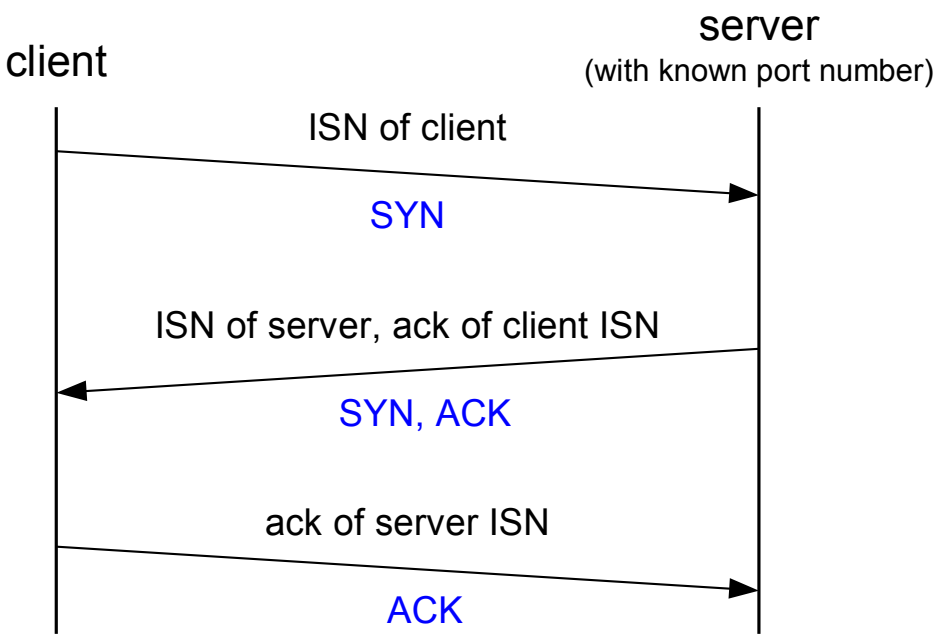

host I
with IP address A
host II
with IP address B
Server
with well-known port
unique
association
unique
association
Client 1
with port X
(assigned by OS)
Client 2
with port Y
(assigned by OS)

TCP is connection-oriented

There are three steps involved:

- connection establishment
- data transfer
- connection termination

Connection Establishment (full-duplex)



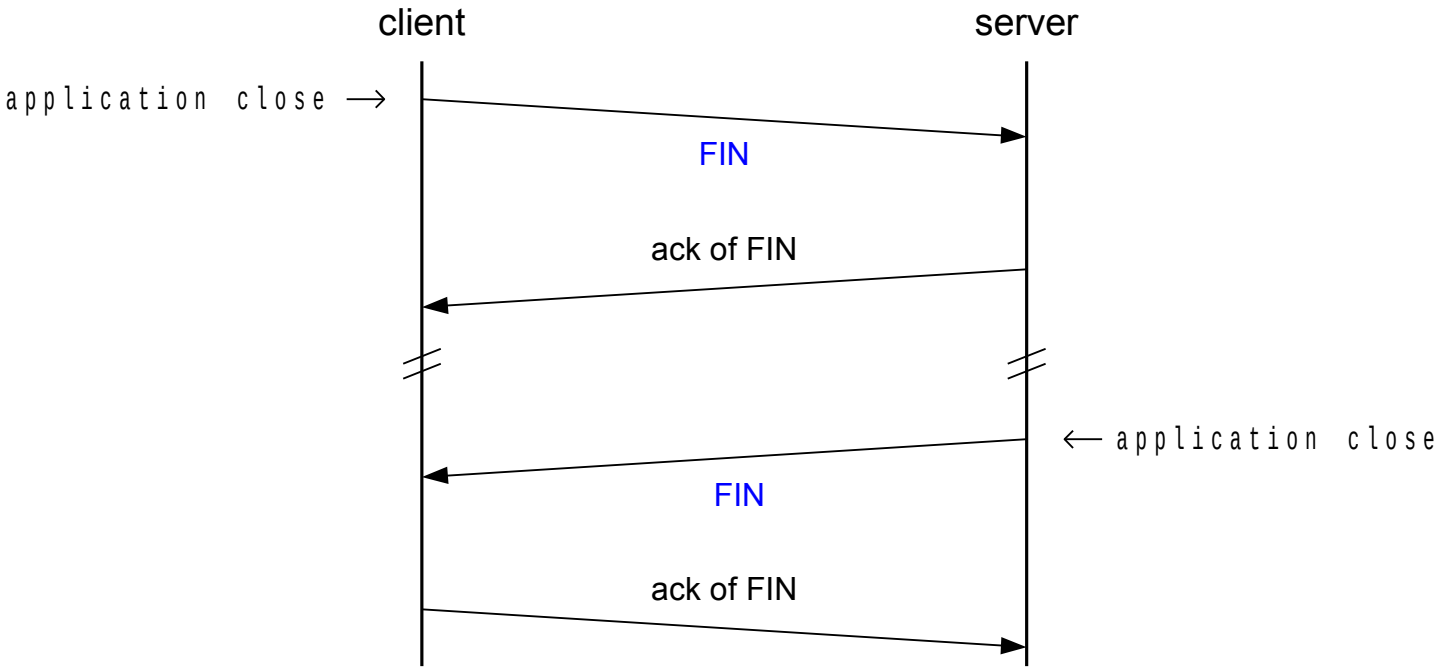
ISN : initial sequence number
ack : acknowledge

SYN : SYN flag is set
ACK : Acknowledge flag is set

Also called *three-segment handshake*

Connection Termination

Each direction of the full-duplex connection must be shut down independently



FIN : FIN flag is set

Window

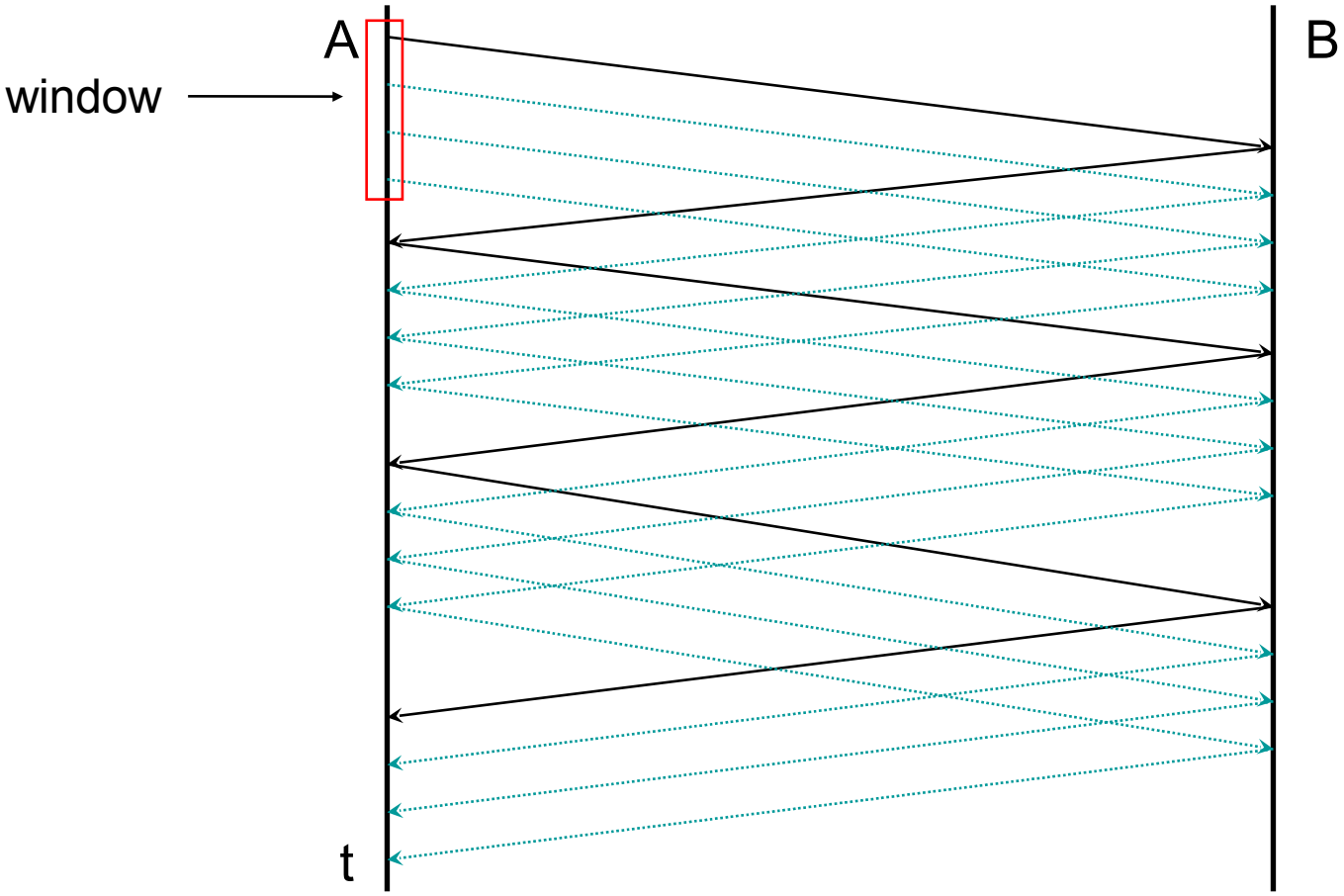
- Window (16 bits):

  The number of data bytes beginning with the one indicated in the acknowledgment field which the sender of this segment is willing to accept.

  ⇨ Flow Control

  ⇨ Sliding Window

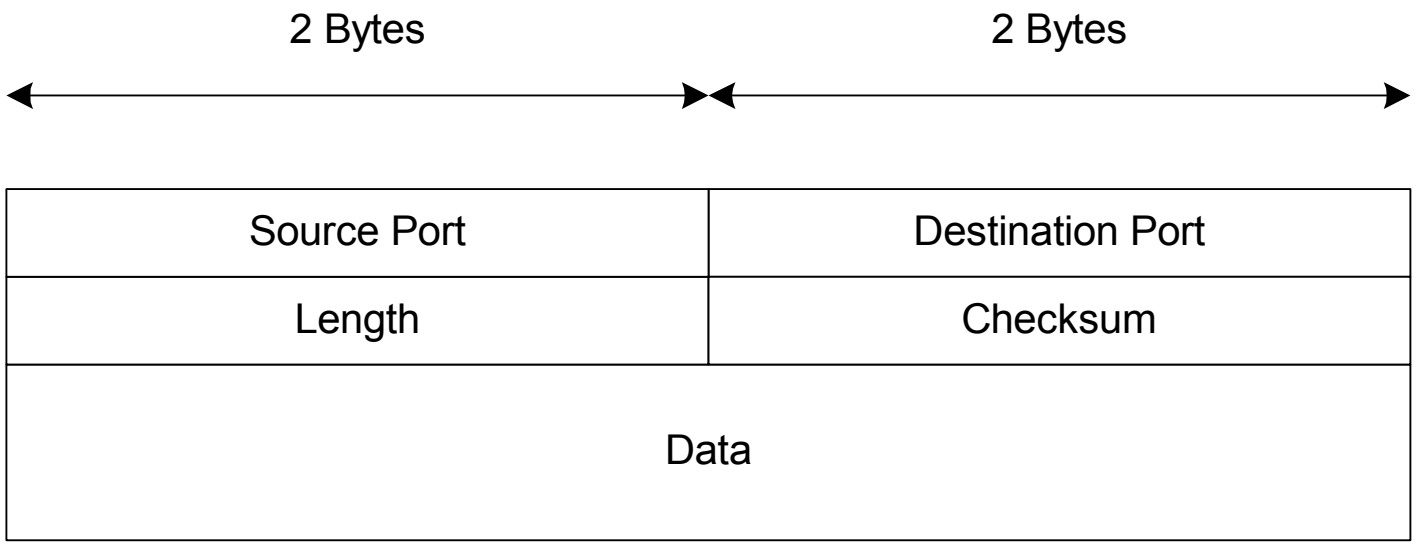
Sliding Window



Urgent Mode

- Hosts can send „urgent data“ placed into the normal stream of data (e.g. FTP - Abortion of the file transfer)
- It is up to the receiving end to decide what to do
- Urgent Pointer (16 bits):
  - This pointer is a positive offset that must be added to the sequence number field of the segment to yield the sequence number of the last byte of urgent data.
  - This field will only be interpreted in segments with the URG control bit set.

UDP Header


2 Bytes
2 Bytes
Source Port
Destination Port
Length
Checksum
Data

UDP Header Explanation

- Source Port: an optional field

  When meaningful, it indicates the port of the sending process, and may be assumed to be the port to which a reply should be addressed in the absence of any other information. If not used, a value of zero is inserted.

- Destination Port: The destination port number.

- Length: The length in octets of this user datagram including the header and the data. (minimum: 8 Byte)

- Checksum (16 bits): The checksum of header and data.