Computer Networks
Introduction,
Network Architecture, Layers

2025/2026, 1st semester

Dr. Szilveszter Kovacs

E-mail: szilveszter.kovacs@uni-miskolc.hu

www.iit.uni-miskolc.hu/~szkovacs

Institute of Information Technology 107/a.

Phone: +36 46 565-111 / 21-07

Általános INFORMATIKAI Tanszék

Computer network

- **Computer network:**
  An interconnected network of autonomous computers
  - **Non-subordinate connection (e.g. computer to peripheral)**
  - **Any communication subsystem can connect them together (electronic information exchange)**
- **Distributed system** (as opposed to network) **:**
  A single virtual system whose elements cooperate to accomplish a task
  - **The specific location and functions of each element are hidden**
  - **also can be implemented on a computer network**

The purpose of the network

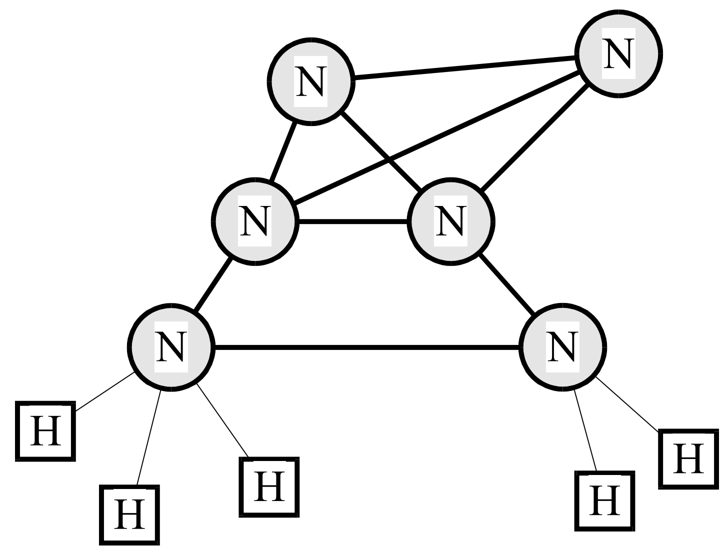
- Resource pooling/sharing
  - All resources should be accessible to anyone, regardless of their physical location
- Increase Reliability
  - Multiple resources with the same function, redundancy, data security,
- Increasing economy
  - For example, instead of one expensive supercomputer, several, smaller, cheaper ones – GRID computing, cloud computing
- New (special) services: communication
  - E.g. email, chat

Classification by extent

- **Local area network (LAN)**
  - in the past : ≈ 0-1 km, room-building group, short distance, high speed, today: up to several km – Campus network
- **Metropolitan Area Network (MAN)**
  - <10 km, medium distance, medium speed
- **Wide Area Network (WAN):**
  - continents, long distance, in the past : medium or low speed
- **Interconnected long-distance network**
  - planetary
- **Speed limits are disappearing! LAN-MAN –WAN, today more Gbps in size (optics DWDM)**

Network structures

- A computer network is a communication network connecting hosts
- Goal: Message passing between hosts
- Its elements:
  - Transmission lines – channels
  - Switching elements
    - Interface Message Processor (IMP), or
    - Packet (line) switching node, or
    - Node

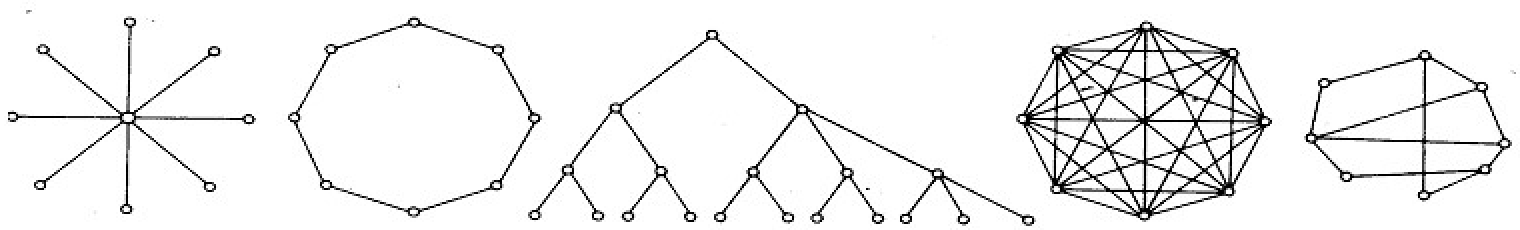


- Nodes
  - A computer (~Host) that is connected to multiple transmission lines.
  - Its task is to forward messages (or packets) (traffic control)

There are two major groups of networks

Built from point-to-point connections

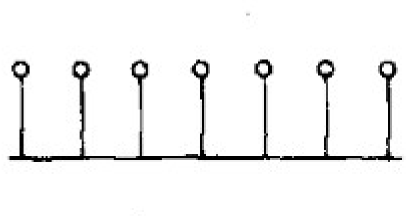
- Topologies: Hub and spokes (star), ring, tree, complete, irregular
- **On a channel there are always two** nodes communicating.
- Usually long distances, WAN, MAN (now also LAN (UTP))
- The messages (packets) are stored and forwarded (store and forward) by the nodes in the new direction. Also known as packet switched network.

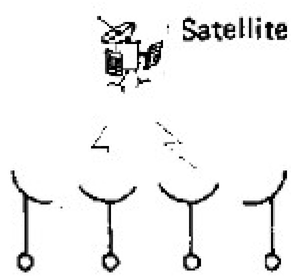


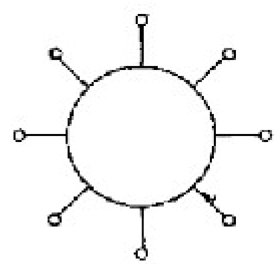
The other group

Broadcast Channel Network

- **A single channel shared by all nodes**.
- **A packet sent by one node is received by all others**.
- Address field inside the package: destination, source
  - from the destination they know who is the recipient (the rest hosts are discarging), (identification is based on the address field) – (unicast).
  - Special addresses
    - Message to all machines (broadcast)
    - Group address (multicast)
- Topologies: bus, satellite or radio, ring.
- Typically LAN or wireless network





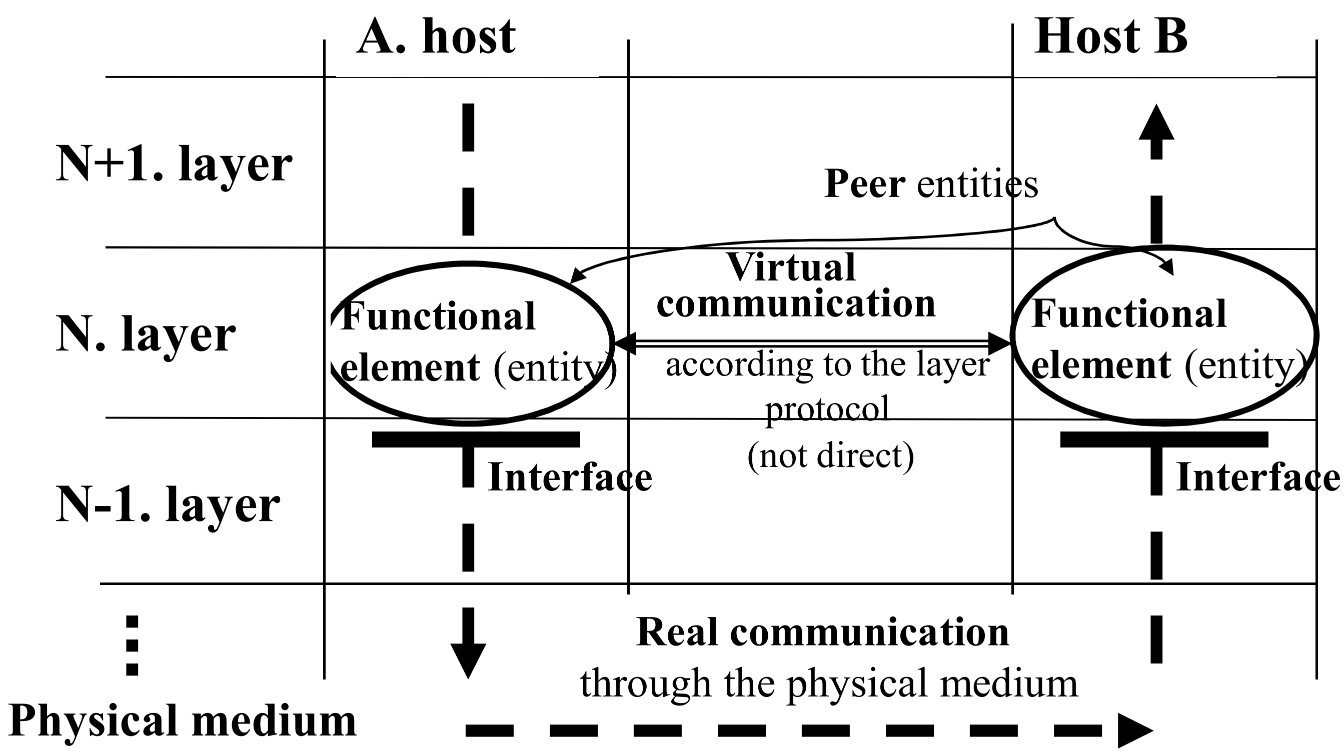


Broadcast channel

- **Problem: Only one station can transmit at a time**
  - **channel allocation problem.**
- **The channel allocation can be**
  - **Static**
    - e.g.: cyclic multiplexing (selection), (e.g.: round robin)
    - **Problem: if there is nothing to send → unused channel.**
  - **Dynamic**
    - **only those who want to broadcast compete** (better channel utilization).
    - **It can be:**
      - **centralized** :
        an arbitration (scheduling) unit decides who is next.
      - **decentralized** (distributed): each station decides for itself whether to transmit (distributed algorithm)

General network architecture

- **For simpler (structured) design, computer networks are organized into layers.**
  - **Network architecture: a set of layers and protocols**



Network architecture

- **Layer:**
  - Provides well-defined services to the layer above it
  - Hides the details of service implementation (“black box” design)
- **Interface:**
  - Definition of elementary operations and services provided by the lower layer to the upper one
  - (Control information and data are transmitted through the interface (down and up))
- **Functional element (entity):**
  - Implementation of the functions of the given layer
  - The functional element connects the services below and above the layer.

Peer entities

- **Peer entities:**
  - **Functional elements in corresponding layers on different machines.**
- **Virtual communication:**
  - **Communication of the peer entities**
- **Physical communication:**
  - **Transfer of data and control information down and up through the interfaces between layers**
    - **If the n^{th} layer of one machine communicates with the n^{th} layer of another machine, virtual communication occurs, while real communication occurs in the physical layer.**
- **Protocol:**
  - **A set of rules and conventions used in communication.**

PL: (multi-level communication)

Two philosophers want to communicate (layer 3). Their location : Kenya, Indonesia; Their language: Suhaeli, Indonesian

Their layer protocol: they exchange sentences in turns.

3/2 IF: verbally/ Suhaeli, on paper/ Indonesian (here the language is IF protocol).

Layer 2: interpreters; Their layer protocol is English (what they translate from).

2/1 IF: verbally, on paper.

Layer 1: Technicians. Their physical protocol: telex (how to send, receive).

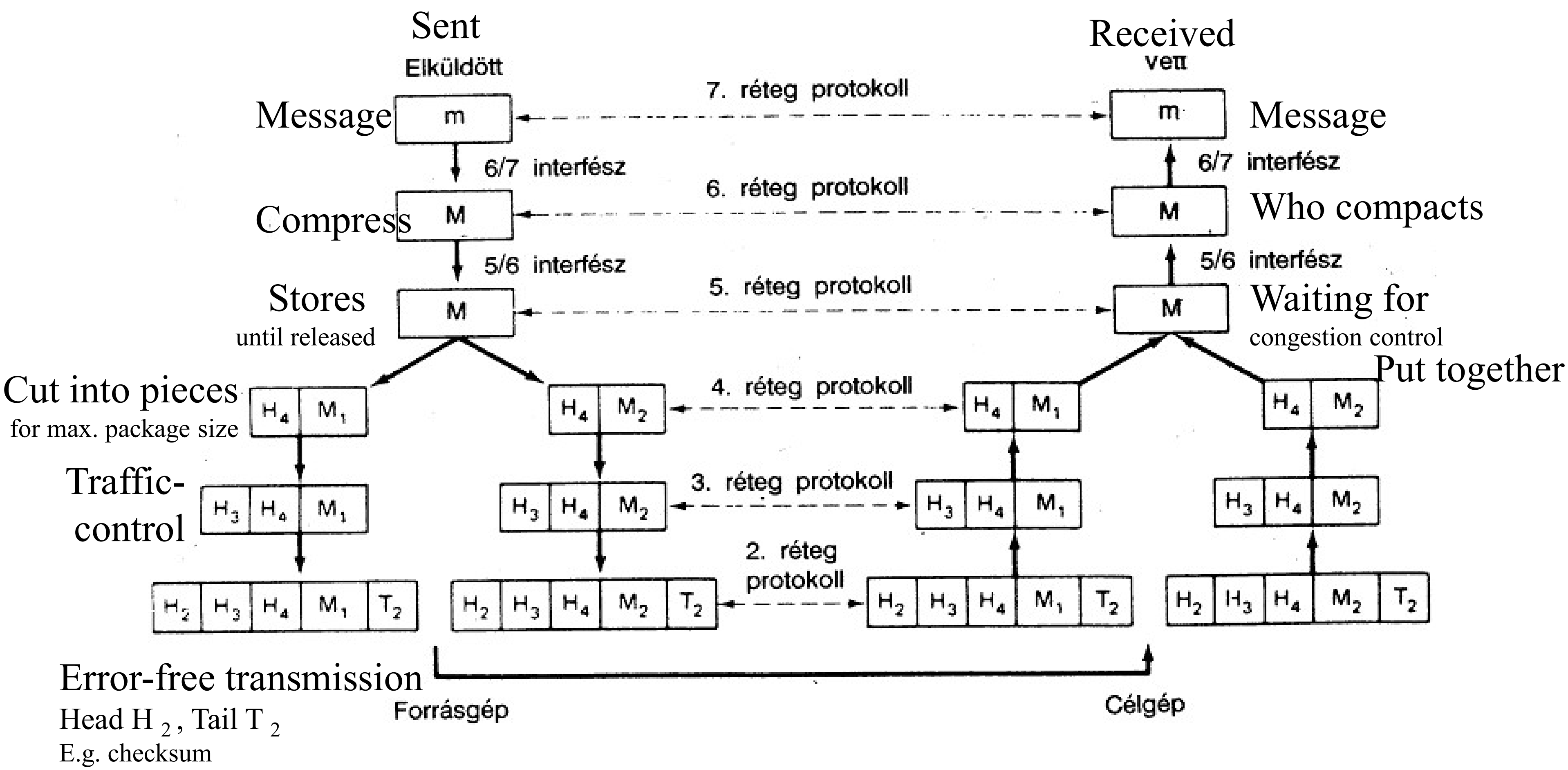
Why is layering good?

For example, the protocol or interface of any layer can change without affecting the upper layers.

Network architecture

- **Set of layers and protocols**
- **Enough information to implement**
- **It does not include either the detailed implementation or the specification of the interfaces (it is a designer's decision during the specific implementation).**

Network architecture



- Each layer can have a connection establishment and teardown mechanism
- The direction of data transmission can be: simplex, half-duplex, duplex

The ISO-OSI network reference model

- **International Standards Organization (ISO)** recommendation:

Open System Interconnection (OSI) reference model

- **The reference model: 7-layer structure**

  The OSI model is not a network architecture!
  It does not define specific protocols or services in each layer.
  It only defines functions.

ISO/OSI layer design

- **Considerations for designing layers:**
  - **Layers should represent different levels of abstraction**
    and perform well-defined tasks
  - **Create standards**
  - **Minimal information exchange** between layers
  - **Number of layers**
    (not too many – simple,
    not too few – only a few tasks should be placed in one layer)

The recommended 7 layers

7: **Application layer – widely used protocols** (e.g. file transfer, mail, virtual terminal)

6: **Presentation layer – code conversion, encryption, compression** (data format management)

5: **Session layer** (e.g. organization of conversations, synchronization, interaction management)

4: **Transport layer** (providing end-to-end connectivity on a large network)

3: **Network layer – route selection**

2: **Data Link layer – transmission of data units, error checking, delimitation, correction** (ensure media access, channel sharing (if necessary))

1: **Physical layer – connects to physical medium** (ensure transmission of bitstream)

Basic concepts

- **Protocol**
  - **A set of rules that govern the exchange of data between two separate elements (entities)**
    (using the same "language")
    E.g. Conventions about the subject, how, etc. of communication.
    Protocols have syntax, semantics
  - **Layer protocol**
    - **Protocol among peer-entities**
- **Interface**
  - **It is located between two layers.**
  - **It describes the services provided by the lower layer, the data and control information required to request them, the information providing the results of the services, and the method of access.**

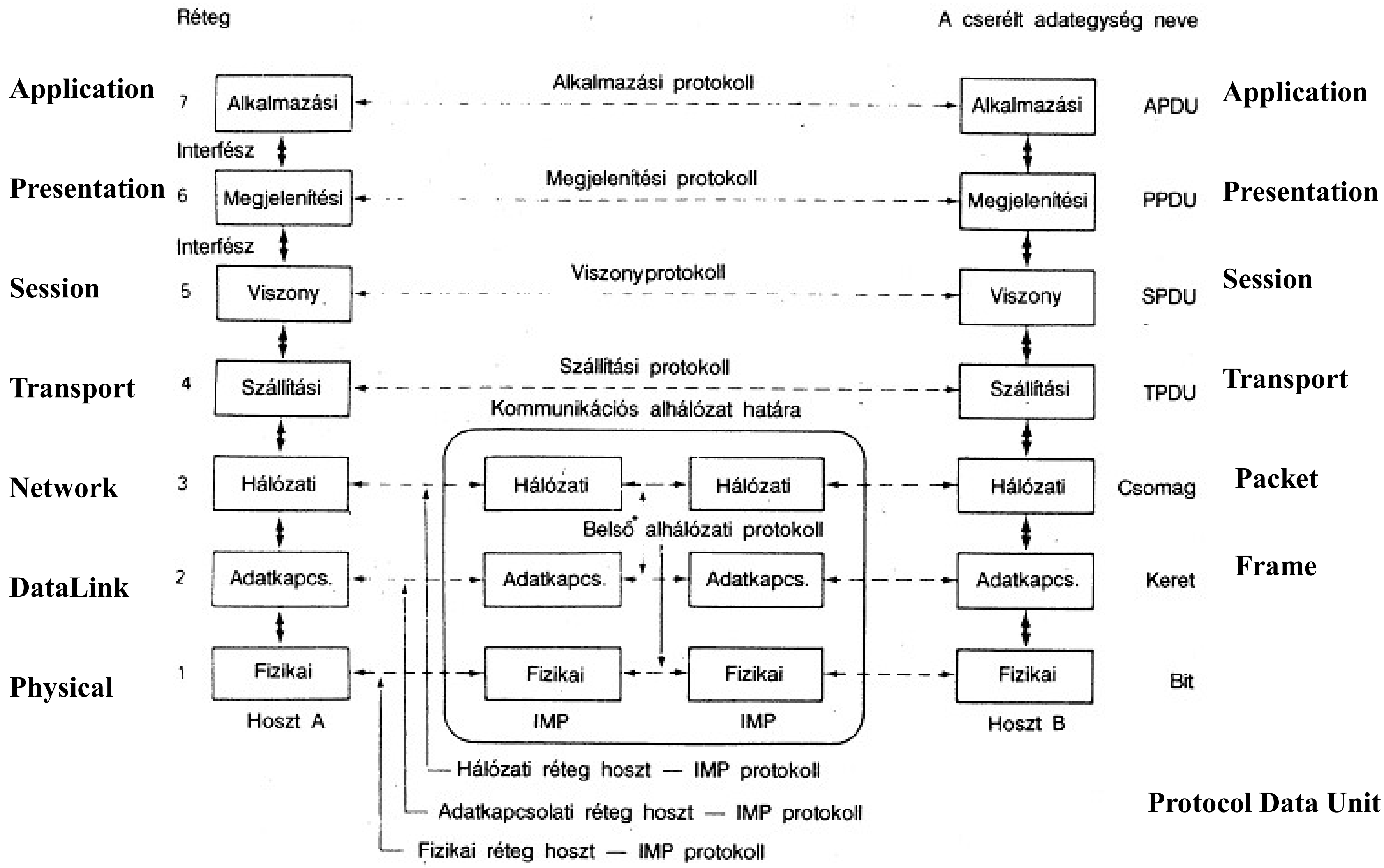
Basic concepts

- **Reference model**
  - It defines the recommended number of layers and the functions of the layers, but does not specify specific protocols and interfaces!
- **Network architecture**
  - A set of layers and protocols, which is enough information for implementation. However, the implementation itself is not part of it, not even the specification of the interfaces!

Architecture based on ISO OSI model

- **ISO also produces standards for each layer (but these are not part of the reference model - e.g. IEEE 802.3 for the data link layer)**
- **The following figure is an example of a network architecture based on the OSI model.**

Network architecture based on the ISO OSI model



1. The physical layer: <u>bitstream</u>

- **It is responsible for the transmission of bits over a communication channel.**
- **Questions to be resolved:**
  - **Representation of individual bits** (signals, e.g. voltage)
  - **Definition of data transmission directions** (duplex, half-duplex, simplex)
  - **Building and breaking connections**
  - **Type of the medium and connectors** (how many pins, what kind of plug)
- **The physical layer protocol**
  - **mechanical,**
  - **electric,**
  - **functional,**
  - **procedure specifications.**

2. Data Link Layer: Frames

- **Ensures error-free transmission for the network layer.**
- **Tasks:**
  - **Framing and delimitation** (according to the physical layer).
  - **Checking and correcting errors (** coding, sending and receiving acknowledgements).
  - **Data flow control** (slow receiver, source shutdown) , **traffic control.**
  - **Channel sharing if necessary** (in case of access to a broadcast medium – e.g. LLC, MAC).

3. Network layer: packets

- **Controls the operation of the communication network**
  - **Determining the route of packets between source and destination.**
    - **Route can be chosen**
      - **static,**
      - **dynamic.**
  - **Congestion control**
    (no overloaded parts of the network)
  - **Connecting heterogeneous networks**
    (a node contains at least 3 layers)

4. Transport layer: <u>datagram, segment</u>

- **Its task is to transmit messages from the session layer.**
- **True end-to-end layer**
  - while in lower layers, peer entities are not necessarily the real source-target elements, here they are real).
- **Tasks**
  - **Breaking or composing messages**
  - **Creating connections**
  - **Multiplexing multiple unidirectional requests (downstream beaming)**
    - one network connection for multiple transport connections
  - **Multiple network connections for one transport connection**
    - increase throughput
  - **Data flow control** (slow processing, stop the source)
  - **Error handling**
  - **Connection types e.g.:**
    - error-free, correct order,
    - error-free not in sending order

5. Session layer: messages

- **Establishing user relationships between different machines**
- **Tasks:**
  - **organizing dialogues**
    - **establishing one-way or two-way connections**
  - **Synchronization**
    - **(synchronization points in case of large amounts of data transfer, so that in case of failure the data transfer can only be repeated from there)**
  - **interaction management**
    - **(both sides should not attempt the same operation at the same time)**

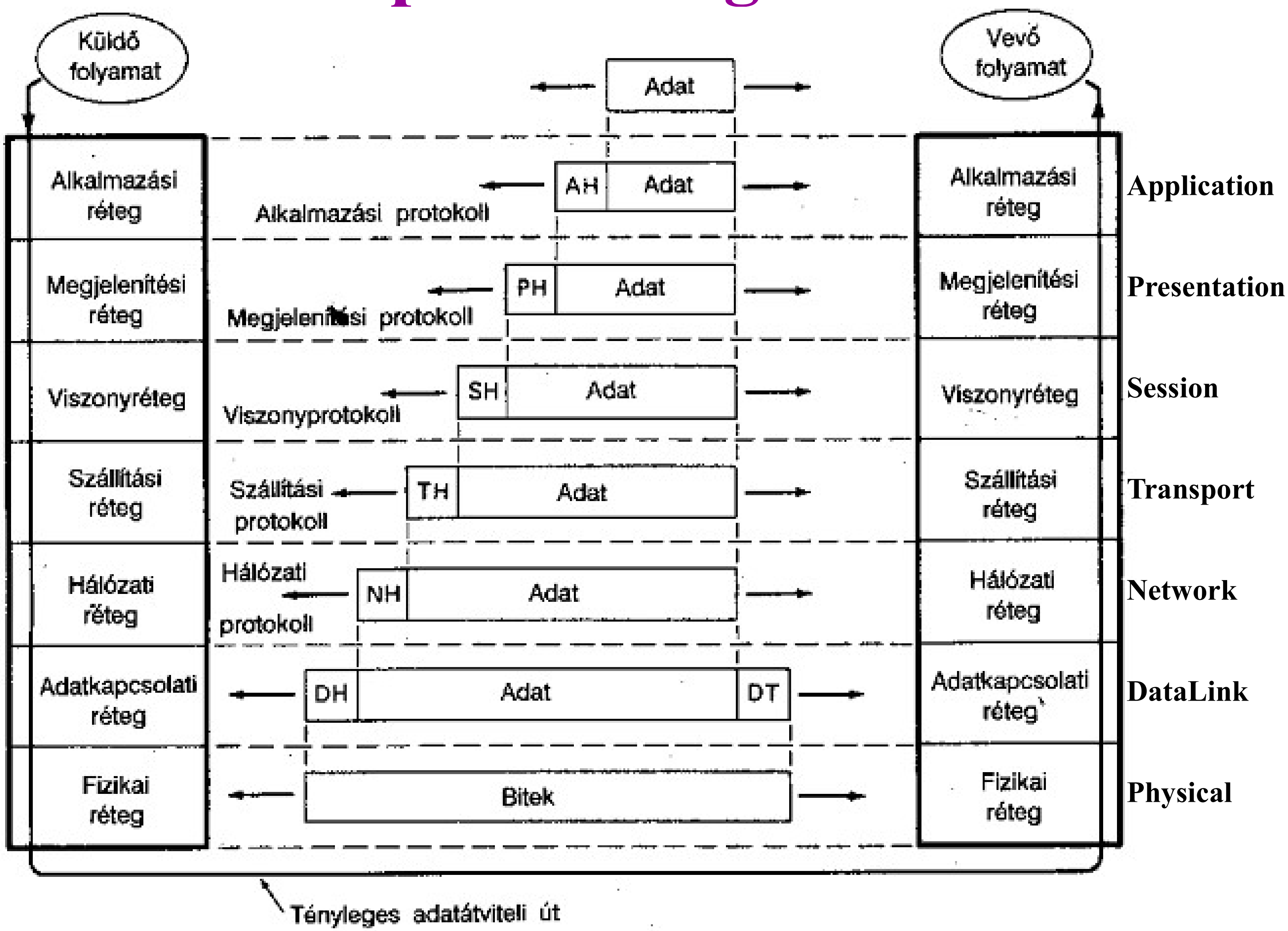
6. Presentation layer: messages

- **It deals with the syntax and semantics of the information to be transmitted.**
- **Performing common general-purpose tasks**
- **Tasks e.g.:**
  - **code conversion** (e.g. different standard encodings)**,**
  - **encryption,**
  - **compression.**

7. Application layer: messages

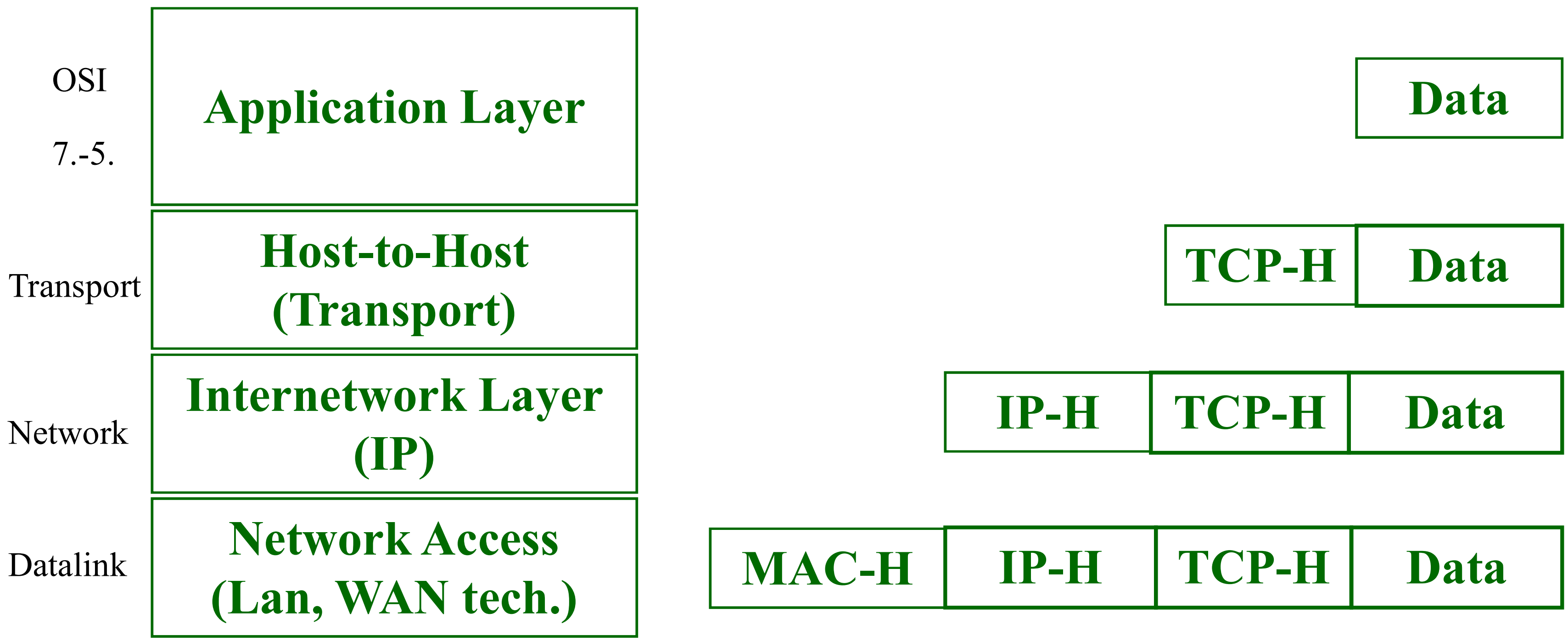
- **It includes widely requested protocols, e.g.:**
  - **File and Print Services**
    - File transfer (ftp), storage and migration, archiving
    - Remote printing, drawing, fax
  - **Communication services**
    - E.g. electronic mail (SMTP, MIME), mailing lists, news
  - **Directory services**
    - Querying addresses, phone numbers, services, network objects, etc.
  - **Application service**
    - Unified terminal description: virtual terminal, remote machine use (ssh), browsers (WWW)
    - Database service

Example of using OSI



Other models …

- **The DoD (Department of Defense) model**
  - **ARPANET, TCP/IP protocol stack**

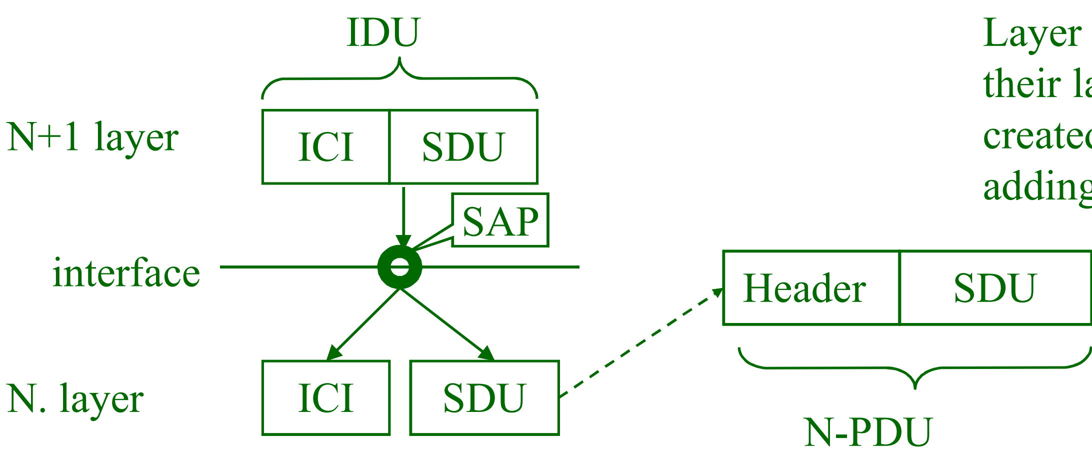


Services

- **The task of each layer of the OSI model is to provide well-defined services to the layer above it.**
  - builds on what is underneath
  - services are provided by functional elements
- **Service Access Point (SAP) concept:**
  - **The services are accessible through these.**
  - **Every SAP has a unique identifier .**
    - E.g. Berkley Unix Socket addresses

The general model

- **E.g.: A functional element at layer N+1 sends an interface data unit (IDU) to the functional element at layer N via the SAP.**



Layer N elements exchange N-PDUs in their layer N protocols. An N-PDU is created by possibly splitting an SDU and adding a header.

ICI: Interface Control Information

SDU: service data unit

PDU: protocol data unit

Types of services

- **Connection-oriented service**
  - **connection establisment,**
  - **use,**
  - **disconnection.**

  → Keeping the sending order.

  - **Two variants:**
    - **message sequence (message boundaries are preserved),**
    - **byte sequence (no message boundaries).**

Types of services

- **Connectionless service**
  - **Messages (message pieces) contain a destination and a sender address,**
  - **they are transmitted independently of each other.**
  - Result:
    → **do not keep the sending order (datagram) connection .**

It can be both acknowledged (reliable) or **unacknowledged** (unreliable)

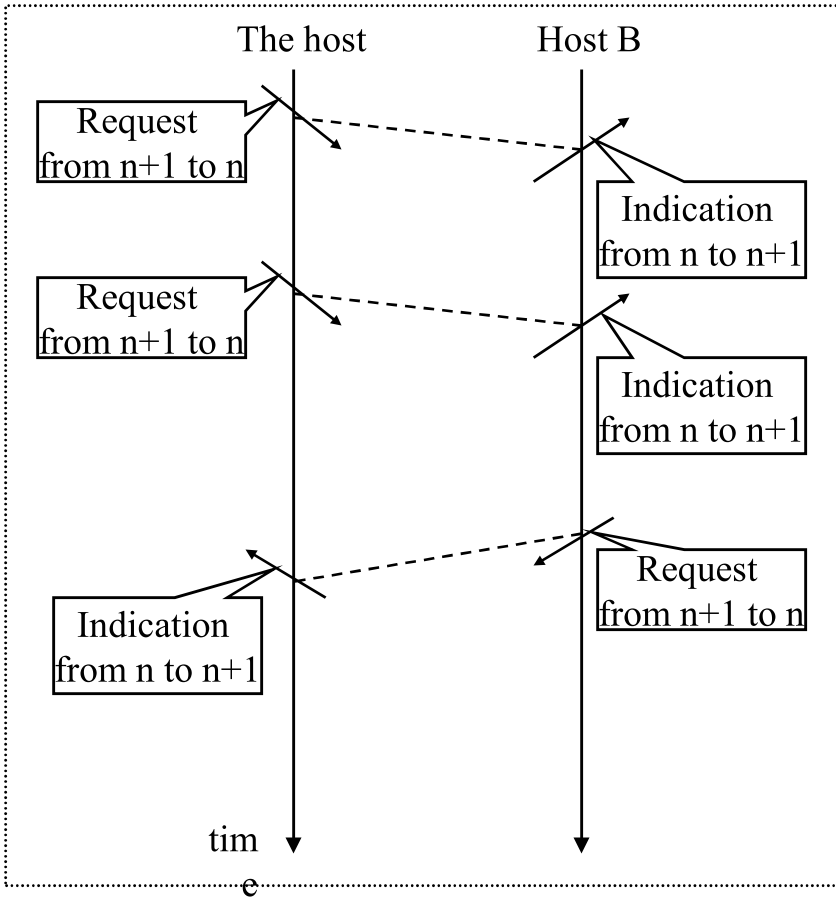
Service primitives

- **A service can be described by a set of primitives, i.e. operations.**
- **primitive classes in the OSI model:**
  - **Request**: a functional element requests the execution of an action (↓).
  - **Indication**: a functional element needs to be informed about an event (↑).
  - **Response**: a functional element wants to respond to an event (↓).
  - **Confirm**: a functional element must be informed about the request **(** ↑ **)** .

Service Primitive Examples

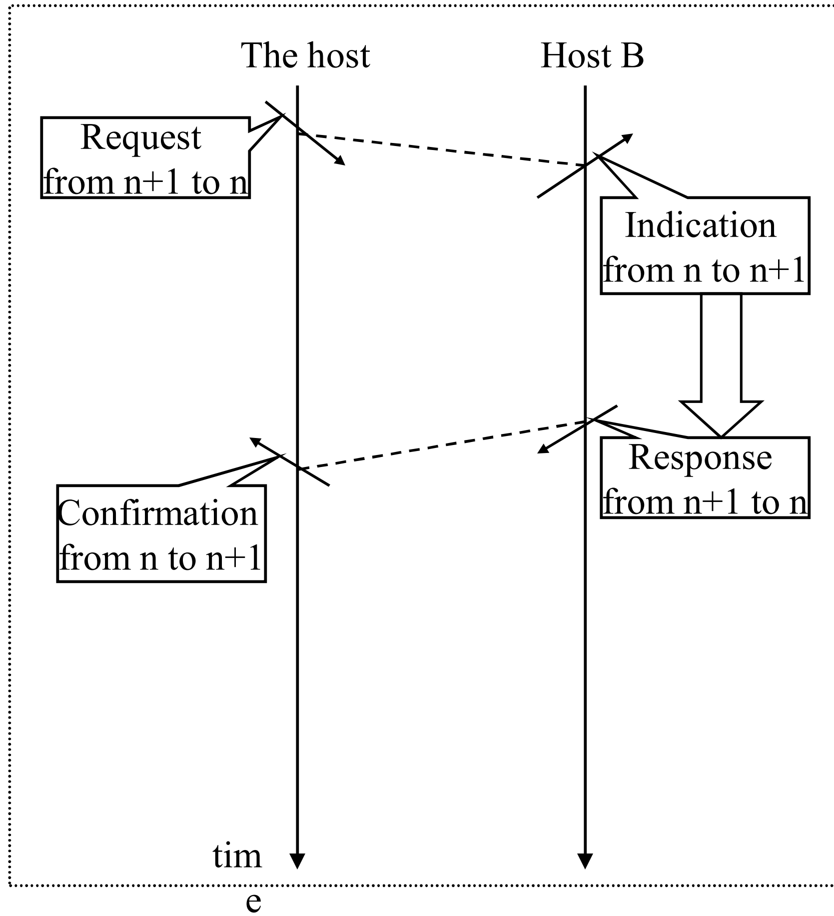
Unconfirmed service

(without confirmation)
(only request, indication)



Reinforced service

(acknowledged)
(request, indication, response, confirm)



An "everyday" example...

I invite Aunt Milli to tea

- **Confirmed connection establishment, unconfirmed data transmission and connection termination [Tanenbaum, p.44]**
- **Markings:**
  CK: Connect.Request
  CB: Connect.Indication
  CV: Connect.Response
  CM: Connect.Confirm
  DK: Data.Request
  DB: Data.Indication
  DC.K: Disconnect.Request
  DC.B: Disconnect.Indication

I'm inviting Aunt Milli to tea.

1.CK: I'm dialing...	
	2.CB: The phone rings
	3.CV: Milli picks up
4.CM: Ringing stops	
5.DK: I invite you for tea...	
	6.DB: Milli hears …
	7.DK: Tell me, he's coming...
8.DB: I hear...	
9.DC.K: I'm hanging up.	
	10.DC.B: Hey, you hung up...

- **Confirmed connection establishment,** **(request, indication, response, confirmation)**
- **unconfirmed data transmission,** **(request, indication)**
- **unconfirmed connection termination.** **(request, indication)**

Physical layer - the page concepts

- **Transmitting data as analog and digital signals**
- **The data transfer (bit rate) of the channel**

$$\frac{\text{Data transferred}\,[\text{bit}]}{\text{Transferring time}\,[\text{sec}]}$$

  - **amount of data transferred in one second: [bit/sec], [bps]**
- **Channel signaling rate**

$$\frac{\text{Signals transferred}\,[\text{pcs}]}{\text{Transferring time}\,[\text{sec}]}$$

  - **signal changes (e.g. voltage, phase) within one second: [baud]**
- **Channel bandwidth**
  - **The difference between the highest and lowest transmitted frequencies.**
  - **A real channel is bandwidth-limited**
    - **Due to power loss (relationship between the energy carried by the signal at a given frequency and the Fourier coefficients)**
    - **There may also be built-in filters.**

Maximum data transfer rate

- **Maximum data transfer rate achievable on a noise-free, band-limited channel:**
- **H. Nyquist (1924) proved**
  - **From an arbitrary signal of bandwidth H, taken at 2H samples per second, the original signal can be completely recovered. From this:**
  - **Max_data_rate [bps] = $2 \cdot H \cdot \log_2 V$**
    where

    H: channel bandwidth

    V: the number of discrete values of the signal (number of possible signals).

    (That is, the value V can be encoded in $\log_2 V$ bits.)

(E.g. a noiseless 3 KHz channel for binary signals (V=2) gives a maximum transmission speed of 6Kbps.)

In case of noisy bandwidth-limited channel

- **Claude Shannon (1948) determined the theoretical maximum data rate for channels loaded with random (thermal) noise (based on information theory considerations)**

Max_available_data_rate [bps] = $H \cdot \log_2 (1+S/N)$

where

- **H: channel bandwidth;**
- **S/N: signal-to-noise ratio**
  - **S: signal power;**
  - **N: noise power,**
  - **in decibels [dB] : $S/N_{dB} = 10 \cdot \log_{10} S/N$**

S/N	S/N_{dB}
0	-inf
1	0
10	10
100	20
1000	30

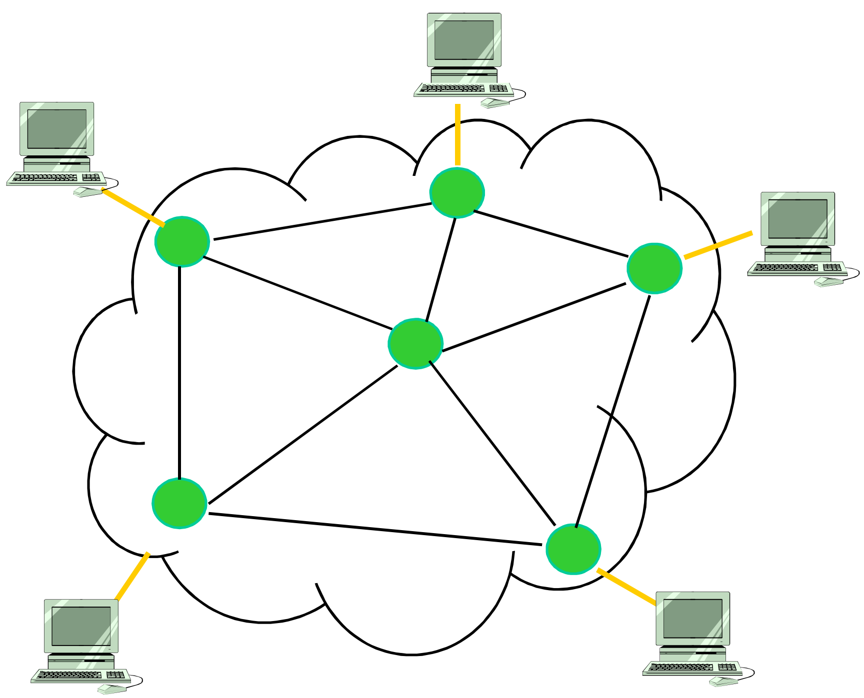
Ex: H=3000Hz; S/N_{dB} = 30dB; ($\log_2 1001=9.967$); max_seb=30Kbps

The Shannon limit

- **In practice, even approaching the Shannon limit is difficult!**
- **On a noise-free band-limited channel, the maximum data rate is infinite ($2 \cdot H \cdot \log_2 V$, $V \to \infty$)**
- **The Shannon limit originates from information theory considerations and has an extremely wide range of validity.**
- **S/N=1, bit error probability 50%,**
  max. bit rate: $H \cdot \log_2 (1+1) = H$
- $2 \cdot H \cdot \log_2 V_{max} = H \cdot \log_2 (1+S/N)$

  $V_{max} = (1+S/N)^{1/2}$

Connected networks



Communication Network

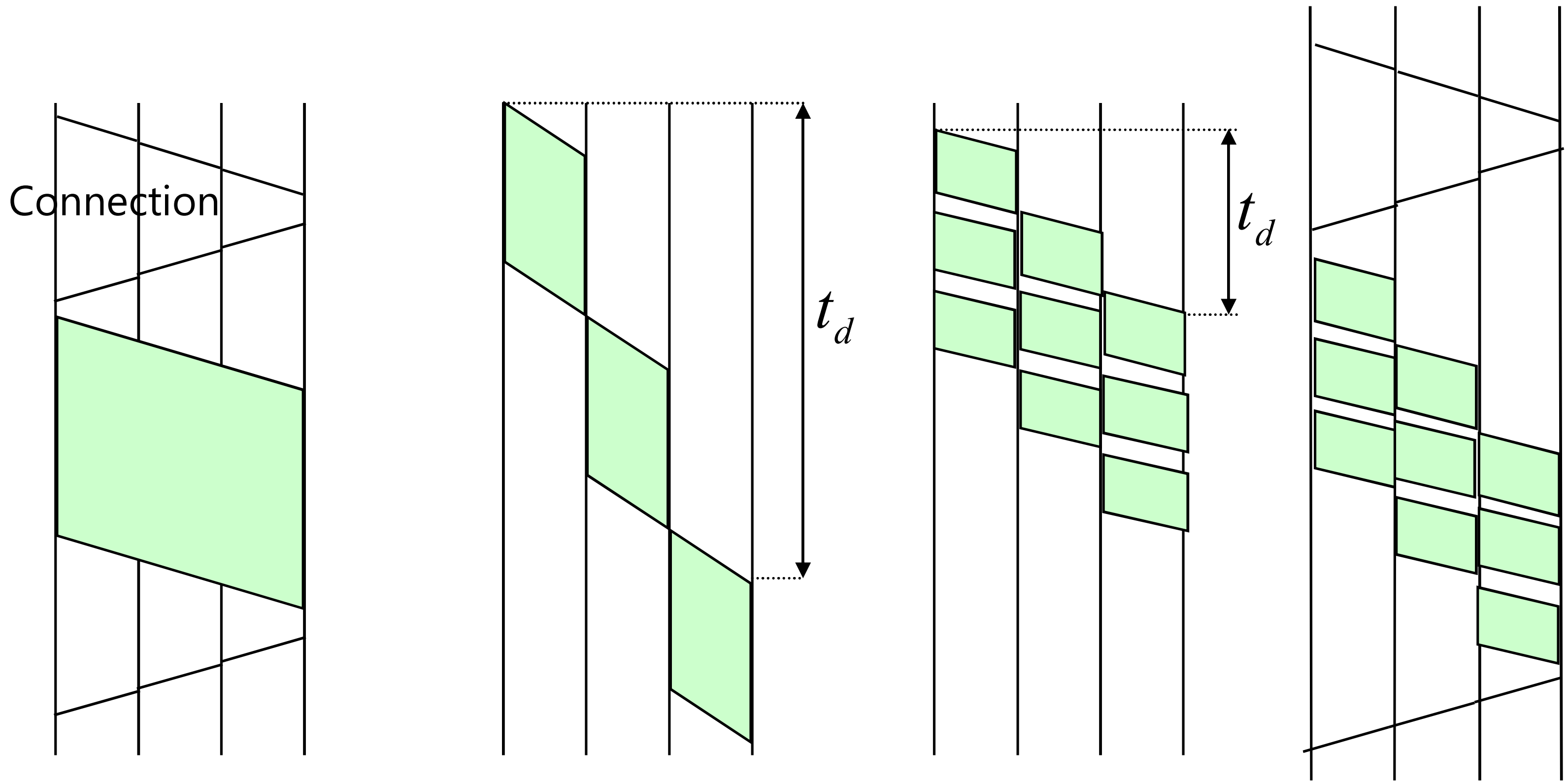
- Communication node (switch)
- Network endpoint

- Data exchange between endpoints through switching nodes

Switching technologies

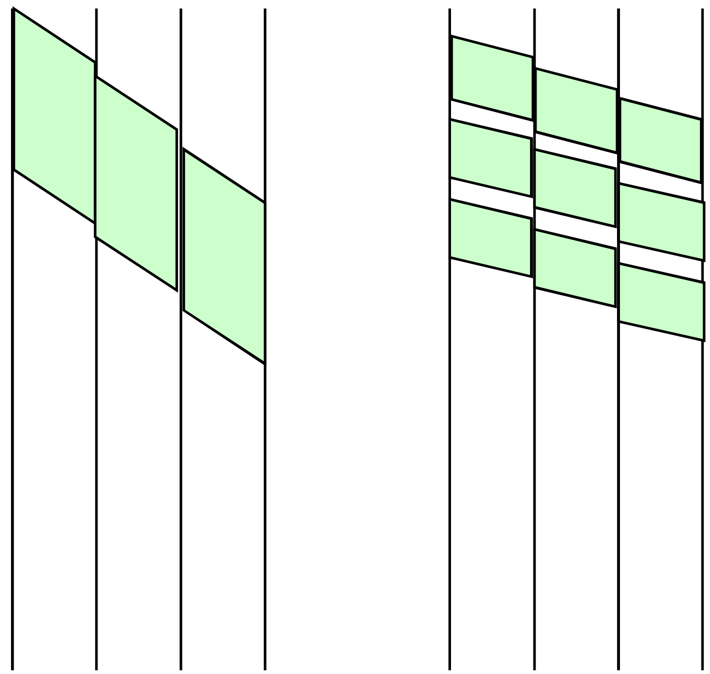
- **Circuit Switching (CS)**
- **Message Switching (MS)**
- **Datagram or Packet Switching (PS)**
- **Virtual Circuit Switching (VCS)**

Switching technologies



Circuit Switching | **Message Switching** | **Packet Switching** | **Virtual Circuit Switching**

Switching technologies – overlapped operation



Message Switching

Packet Switching

Circuit switching

- Circuit-switched network
  - Establishing a connection between the endpoints (disadvantage: this can be time-consuming),
  - communication between endpoints on the dedicated line (advantage: no link access delay, no congestion),
  - finally disconnection.
  - Not favorable in the case of pulse-type (burst) traffic (unutilization may occur - it is reserved but not used).
  - For example, the old public switched telephone network was one such network.

Message switching

- Message-switched network
  - The entire message is sent,
  - the nodes store and then forward the message.
  - There is no limit to message size.
    - Arbitrarily high latency (non-interactive, cannot be real-time), although priorities can be set,
    - Arbitrarily large storage capacity requirement on nodes.
  - Congestion is controllable, making good use of the medium

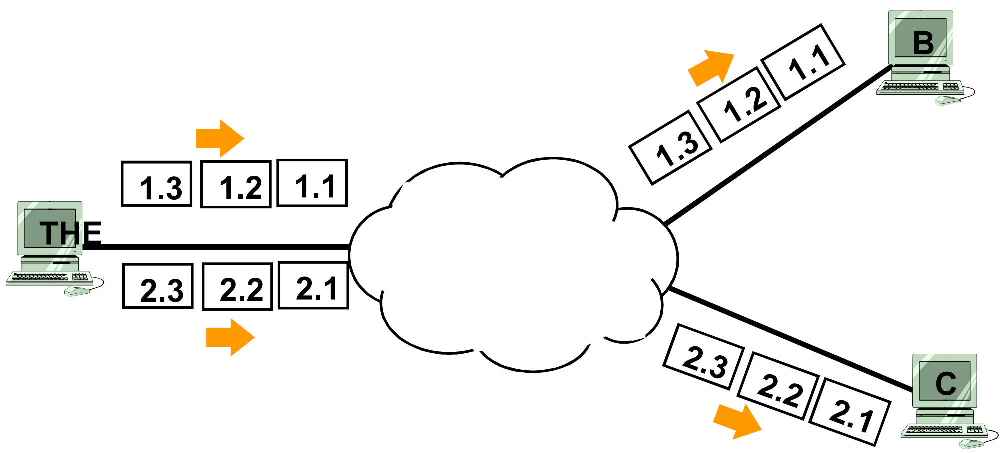
Packet switching

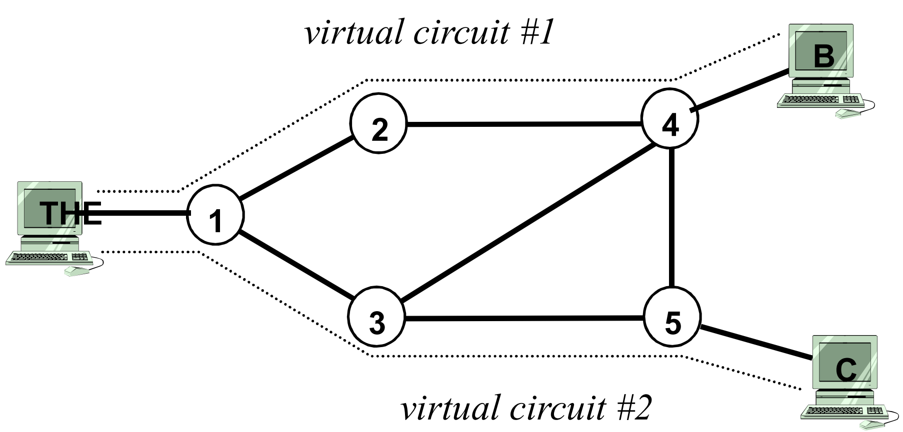
- **Packet-switched network**
  - They produce upper limited sized packets from by fragmenting messages.
  - Packets are “dynamically share” on links between nodes
  - Limited storage capacity requirements at the nodes,
  - lower delay is possible (also suitable for interactive communication).
  - The throughput could be higher.
    - Interleaved operation is possible (the first packets of a longer message are already processed while the rest are still being transmitted).

Virtual circuit switching

- Packet switching, but
  - a logical path is formed between the endpoints,
  - the packets use the same route (therefore they arrive in the order they were sent).
  - Similar to circuit switching, but the route is not dedicated (other packages share some links).
  - Connection to establish the logical path need a build!
- Compared to packet switching:
  - each packet is forwarded independently,
  - order can be “overturned” (ordering protocol required),
  - no connection needed structure.

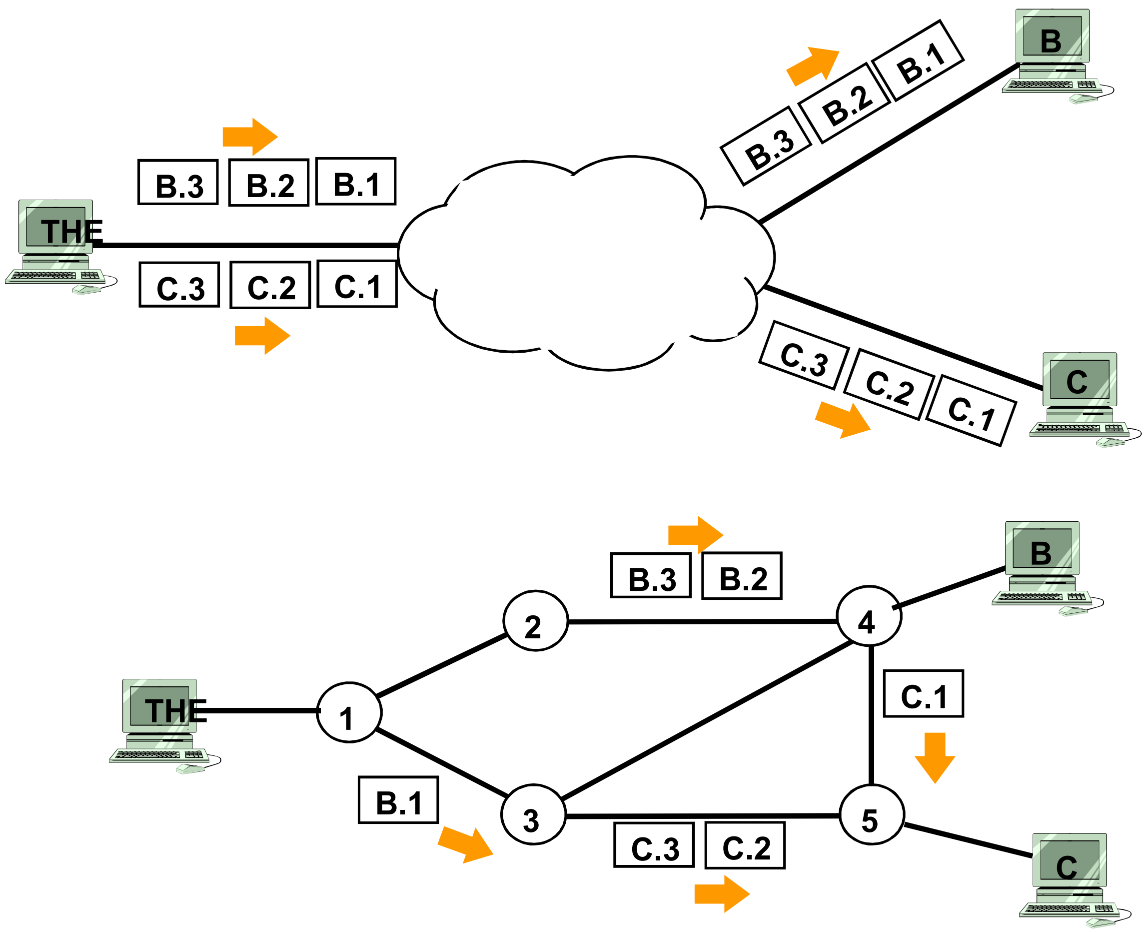
Virtual Circuit Switching





- A logical connection (virtual circuit: VC) is established between two stations.
- Packages are labeled with the VC number and their serial number

Packet Switching



- Each packet is forwarded independently
- The packets are labeled with the destination address and their sequence number. Their order can be "overturned".

Exercise 1

Task 1: Transporting 1.4 10^6 Bytes of data on a floppy disk

The floppy disk has a write/read speed of 20 10^3 Bytes/sec. I write, check it by reading it back, then transport it to the destination at 50 Km/h for 1 hour and read it there.

Questions:

How big is the

a. Total data transfer time?

b. The average data transfer rate?

c. Data transmission delay time

d. What is the average signaling rate in baud if we consider a signal to be 8 bits?

Exercise 1

Solution :

Writing/checking time: $2 \cdot \frac{1.4 \cdot 8 \cdot 10^6}{2 \cdot 10^4 \cdot 8} = 140\,\text{sec}$

Reading time $\frac{1.4 \cdot 8 \cdot 10^6}{2 \cdot 10^4 \cdot 8} = 70\,\text{sec}$

Delivery time 3600 sec

Total transfer time 3600 +140+70 = 3810 seconds

Delay time 3600 +140 = 3740 seconds

Average transfer speed $\frac{1.4 \cdot 8 \cdot 10^6}{3810} = 2939.2\ \text{bps}$

Signaling speed in baud $\frac{1.4 \cdot 8 \cdot 10^6}{8 \cdot 3810} = 367.4\ \text{baud}$

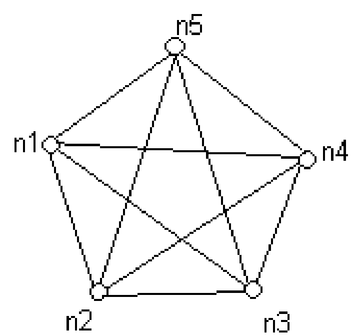
And the signal propagation speed?

approx. 50 Km/h (I could adjust this with the readout/check scan time...)

Exercise 2

Task 2 Topologies

In how many possible ways can 5 nodes (n) be connected in a fully connected network, if we deploy point-to-point connections that can be high-, medium-, and low-speed?



Solution:

Number of edges (connections): 4+3+2+1=10

So the total number of possible cases is: $3^{(4+3+2+1)} = 3^{10}$

Exercise 3

Task 3: Topologies

Let us consider $2^n - 1$ nodes (a node is denoted by N_i). Let the topology be

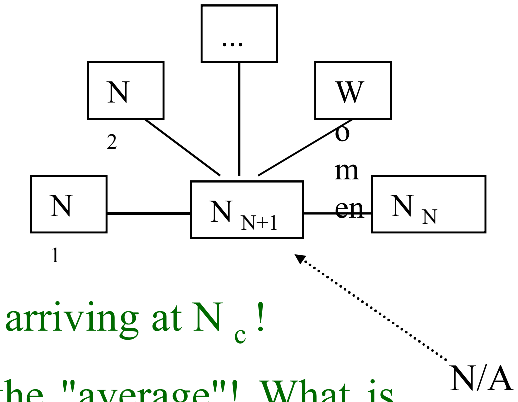
a. hub and spokes (also a node in the middle, N_c);

b. ring;

c. fully connected.

Let us give the average node-to-node "hop" number (the average number of edges touched by a node-to-node packet) (n_T) for all three cases!

Exercise 3



Solution

a. Hub and spoke topology

All transiting packets affect N_c, except those originating from N_c and arriving at N_c!

First - considering this topology - let's give some interpretation to the "average"! What is this average?

... Anyone sends to anyone, let's count all the jump numbers and divide by all the cases!

For Nc: 1 jump (N) node-to: 1 case

For Nx: 2 jumps (N-1) for node: (N) case +

1 jump 1 node-to: (N) case

$$n_T = \frac{\sum \text{jump}}{\sum \text{case}} = \frac{[N \cdot 1] \cdot 1 + [(N-1) \cdot 2 + 1 \cdot 1] \cdot N}{N \cdot 1 + (N - 1 + 1) \cdot N} =$$

$$= \frac{N + (2N-1) \cdot N}{(1+N) \cdot N} = \frac{1 + 2N - 1}{1+N} = \frac{2 \cdot N}{1+N} = \frac{2}{1 + \frac{1}{N}} \cong 2 \text{ if } n >> 1$$

Exercise 3

Solution

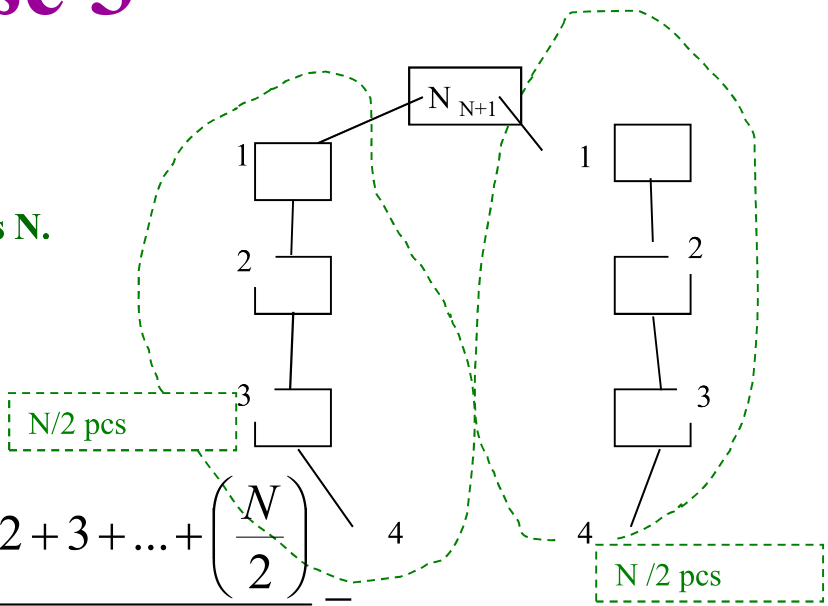
b. Ring topology

“symmetric” with respect to N_{N+1}. The number of cases is N.

The ring can be divided into 2 rows, each with N/2 rows,

and so the number of jumps to one row is

1 + 2 + ... + N /2.



$$n_T = \frac{\sum \text{jump}}{\sum \text{case}} = \frac{2 \cdot \left(1+2+3+\ldots+\frac{N}{2}\right)}{N} = \frac{1+2+3+\ldots+\left(\frac{N}{2}\right)}{\frac{N}{2}} =$$

$$= \frac{\frac{N}{2} \cdot \frac{1+\frac{N}{2}}{2}}{\frac{N}{2}} = \frac{1+\frac{N}{2}}{2} = \frac{1}{2} + \frac{N}{4}$$

arithmetic sequence : $S_k = k\dfrac{a_1 + a_k}{2}$ és itt $k = \dfrac{N}{2}$

c. Fully connected topology (everyone connected to everyone)

Symmetric: it is enough to examine one. There is 1 jump everywhere, so the average is also 1.

$\boldsymbol{n_T = 1}$

Exercise 4

Task 4 Frame retransmission number probability

Suppose the data link layer handles errors by requesting retransmission of corrupted frames. Let **p be** the probability that a frame **is corrupted** (then the probability that the frame **is not corrupted is 1-p**). Assume that the acknowledgement is never corrupted.

What is the expected value of the restart number n of a frame, n_v ? (How many times is it expected to be restarted?)

$$\sum_{j=1}^{k} a_j = k\frac{a_1 + a_k}{2} \qquad \text{(arithmetic sequence)}$$

$$\sum_{j=1}^{k} q_j = q\frac{q^k - 1}{q - 1} \qquad \text{(geometric sequence)}$$

Solution

Exercise 4

Let's look at the probabilities of retransmissions in a table:

	Probability of success	Error probability
It passes right away.	1-p	p
Reissued 1 time	p(1-p)	p^2
Reissued 2 times	$p^2(1-p)$	Friday ^{3}
...	...	
reissued n times	$p^n(1-p)$	
etc.		

Let's interpret the expected value of the retransmission number (n_v),

the product of the retransmission number and its probability

sum as !

$$n_v = \sum_{n=1}^{\infty} n\left(p^n(1-p)\right) = (1-p)\sum_{n=1}^{\infty} np^n$$

Here n is the retransmission number and $p^n(1-p)$ is the retransmission probability.

The amount can also be written as				
1*p +	p^1			
2*p^2+	p^2+	p^2		
3*p^3+	p^3+	p^3+	Friday ^{3}	
	...	...	...	...
	↓	↓	↓	...
	Σp^n	$p \Sigma p^n$	$p^2 \Sigma p^n$	...
	↓	↓	↓	...
	p/(1-p)	p2 $^/$ (1-p)	p^3/(1-p)	...

$$n_v = \frac{p}{1-p}$$

geometric sequence :

$$\sum_{j=1}^{k} q^j = q\frac{q^k - 1}{q-1}$$

$$\sum_{j=1}^{\infty} q^j = q\frac{-1}{q-1} = \frac{q}{1-q}$$

	p	n_v
No damage	0	0
Hardly damaged	0.1	1/9
Medium	0.5	1
Very	0.9	9
Always	1	∝

$$n_v = (1-p)\sum_{n=1}^{\infty} np^n = (1-p)\left(\sum_{n=1}^{\infty} p^n + \sum_{n=2}^{\infty} p^n + \sum_{n=3}^{\infty} p^n + \ldots\right) = (1-p)\left(p\cdot\frac{1}{1-p} + p^2\cdot\frac{1}{1-p} + p^3\cdot\frac{1}{1-p} + \ldots\right) = p + p^2 + p^3 + \ldots = \sum_{n=1}^{\infty} p^n = \frac{p}{1-p}$$

Exercise 5,6

Task 5 Data transfer rate

A television channel has a bandwidth of 6MHz. How many bits can be sent per second (what is the data rate) using quadruple signals?

Solution

The maximum data transfer rate can be calculated using the Nyquist equation:

Data transfer speed = $2 H \log_2 V$ [bps]

Here now $H = 6 * 10^6$ Hz; $V = 4$; Thus $\log_2 V = 2$;

So **Data transfer rate** = $2 * 6 * 10^6 * 2$ = $\mathbf{24 * 10^6}$ **bps** = **24 Mbps**

Task 6 Data transfer rate

What is the theoretical maximum data rate that can be achieved on a 3 KHz channel with a 20 dB signal-to-noise ratio?

Solution

Shannon's theorem : Max_available_data_rate = $H \log_2 (1 + S/N)$ [bps];

and $S/N_{dB} = 10 * \log_{10}(S/N)$, now $H = 3 * 10^3$ Hz; 20 dB = $10 * \log_{10}(S/N)$; of which $S/N = 100$;

So **Max_available_data_rate =** $3 * 10^3 * \log_2 101 = 6.666 * 3 * 10^3$ **≈ 20 Kbps**