Computer Networks

Layered architectures

Józef Woźniak
jowoz@eti.pg.edu.pl
jozef.wozniak@eti.....
Pok. 149 WETI, tel. +347 2223
Katedra Teleinformatyki
Wydział ETI Politechniki Gdańskiej

POLITECHNIKA GDAŃSKA

Computer networks - sample literature

- Sieci LAN, MAN, WAN: Protokoły komunikacyjne – J. Woźniak, K, Nowicki
- Computer Networks – Sieci Komputerowe – Andrew Tanenbaum *& David Wetherall*
- Data and Computer Communications - William Stallings
- Computer Networks and Internets – Douglas Comer

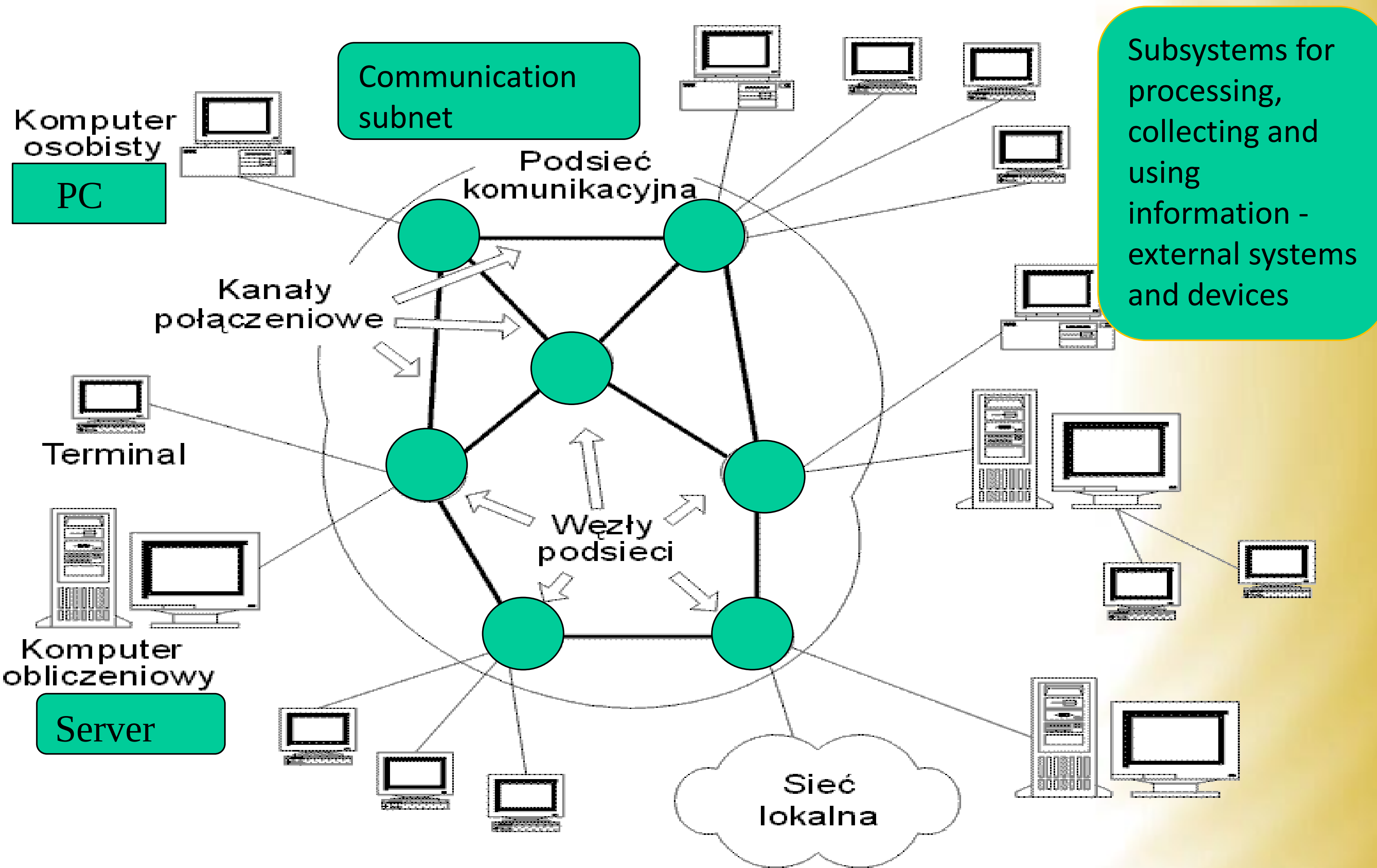
Computer networks

- Objectives of development and services provided
- Network classification
- Establishing connections in networks
- **Layered logical architectures:**

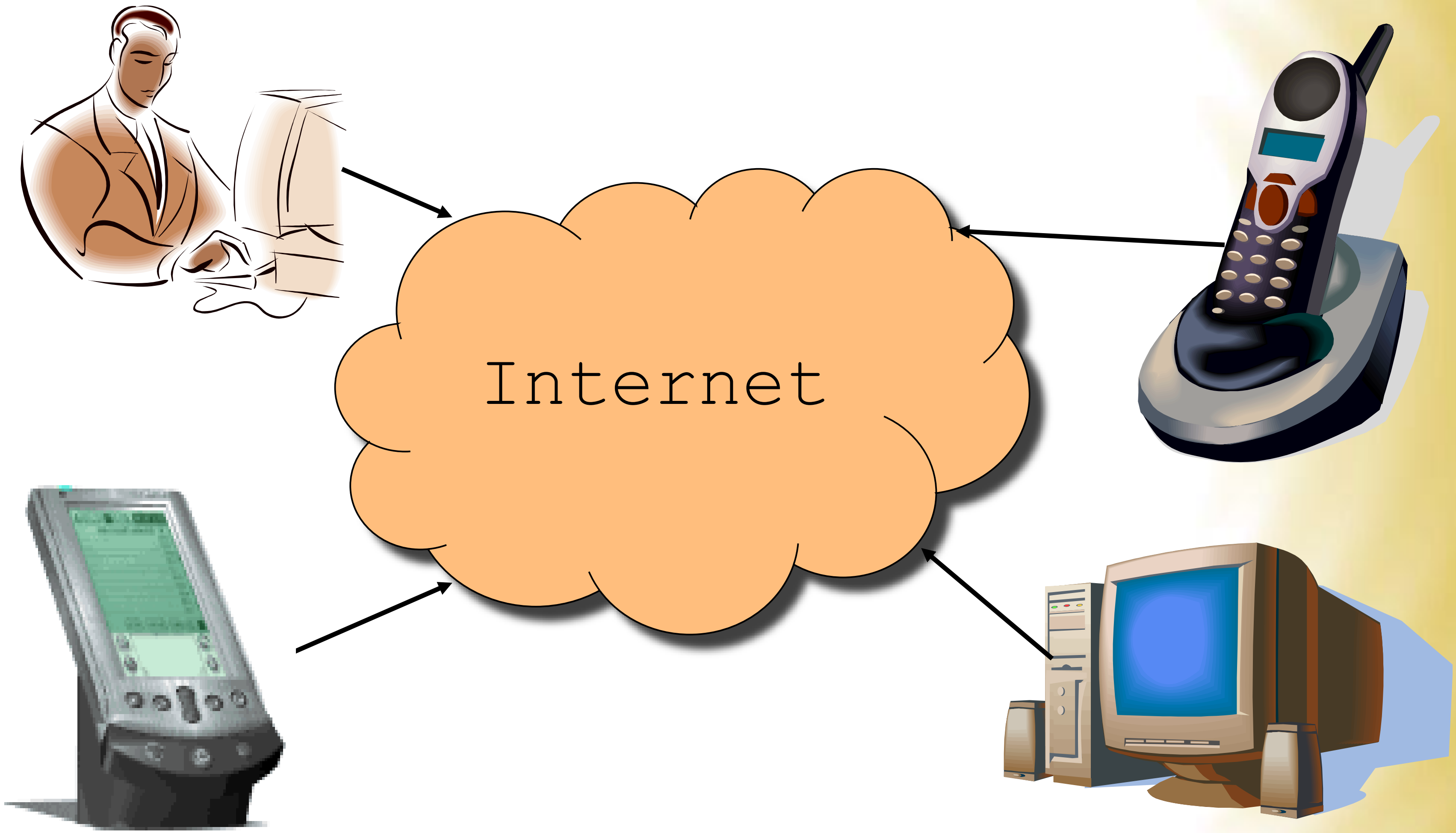
ISO-OSI model

TCP/IP model

An exemplary structure of a computer network

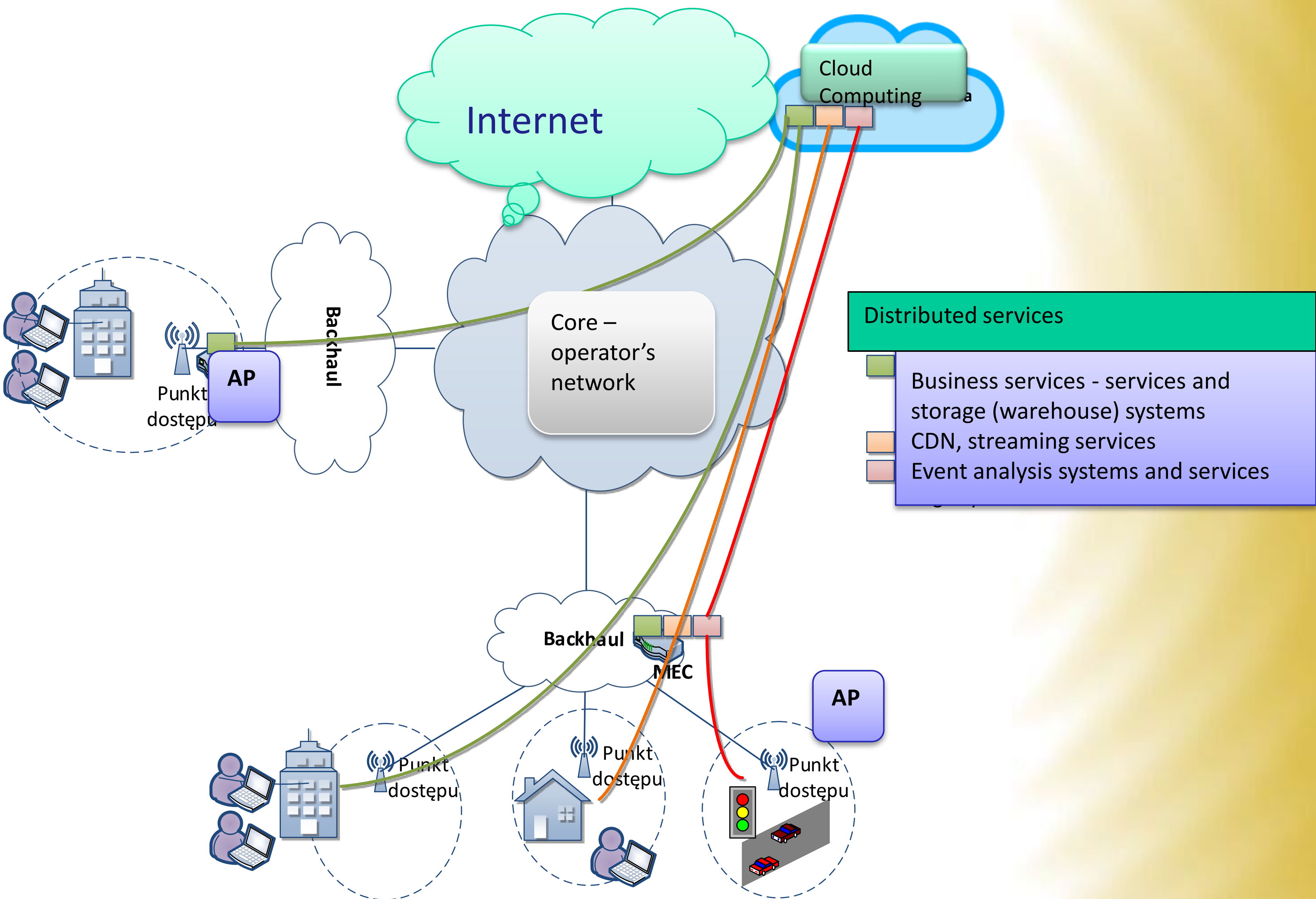


End systems: computers and other devices connected to the network



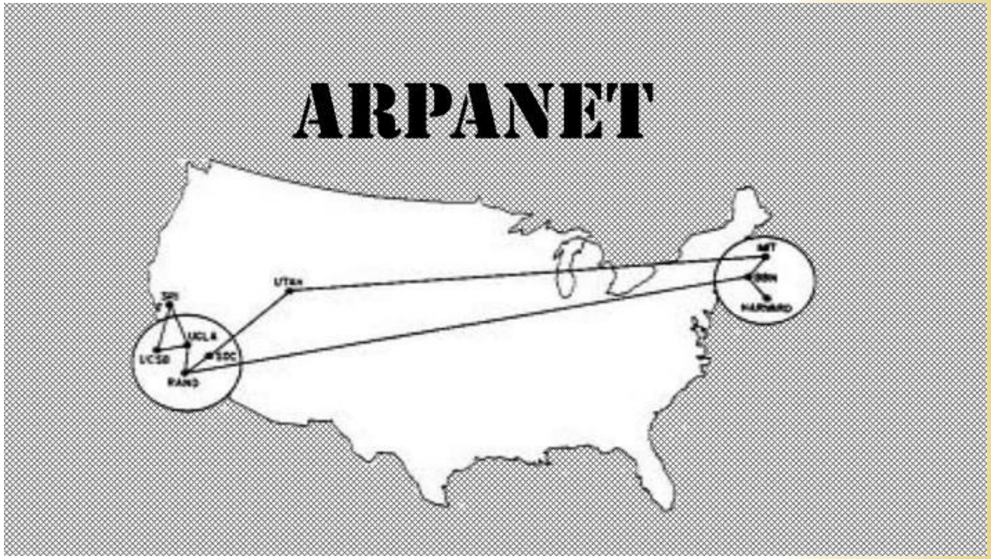
Often called "hosts"

Cloud services

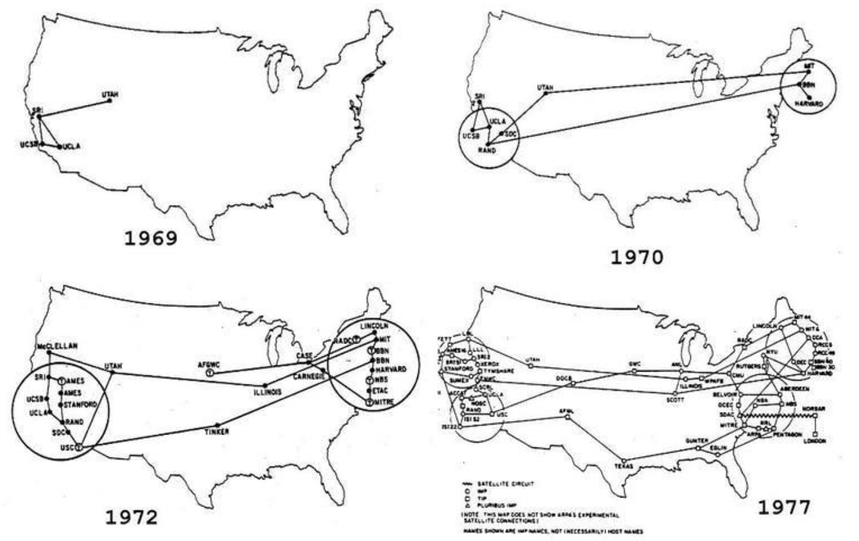


Why are we building networks? - Selected goals

•**The proper distribution of the load between big and expensive computers.** - *Not currently popular classic "load balancing" but time zones in the US were one of the important motivations ... .. of building the ARPA net.*



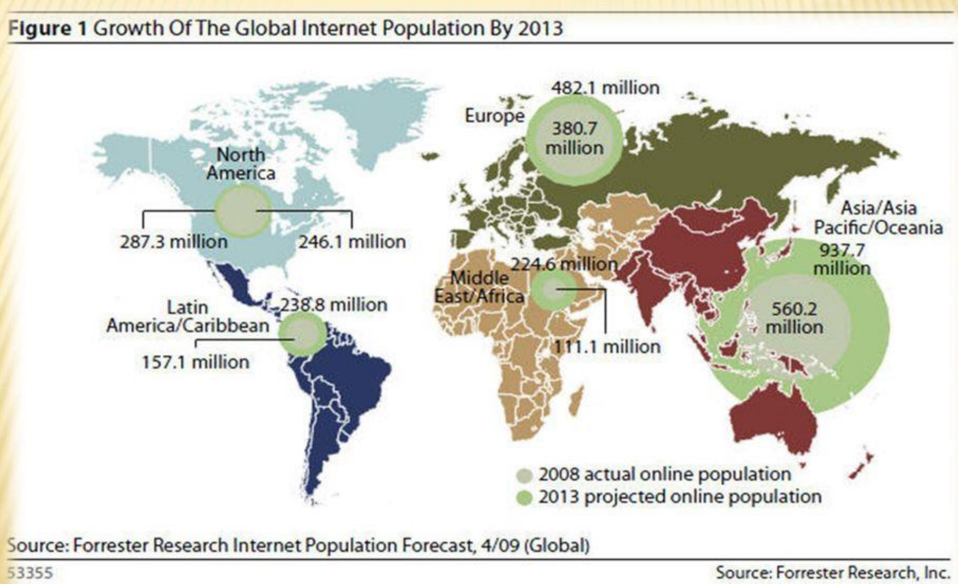
•Ensuring the effective use of hardware (e.g. printers, high-speed computers) and software resources (e.g. shared databases) located in different parts of the network;
•Financial savings in the operation of networks and devices;
•Provision of a wide and distributed communication medium through the network
•Providing higher service reliability through guaranteeing alternative sources of information, computational possibilities and connection routes.



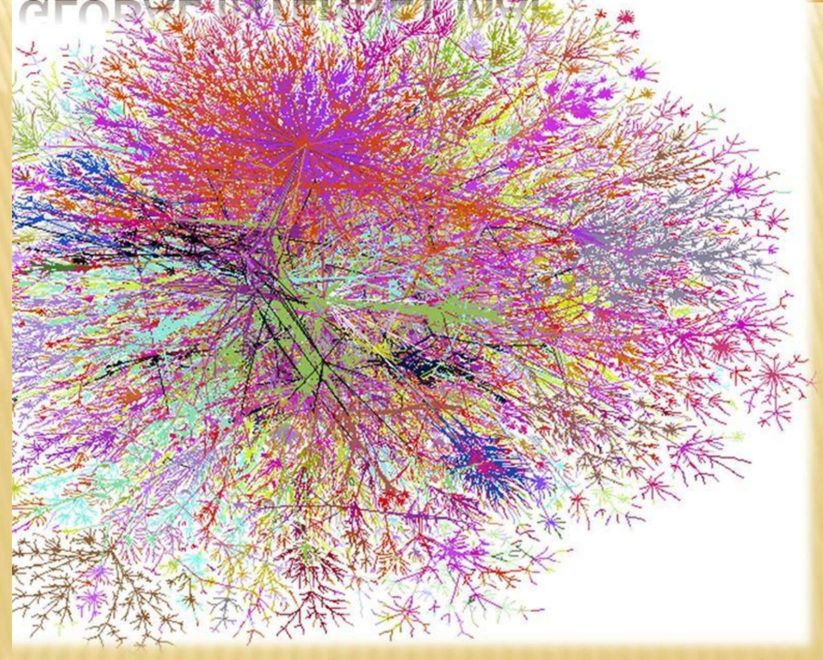
1995 – Birth of Ecommerce

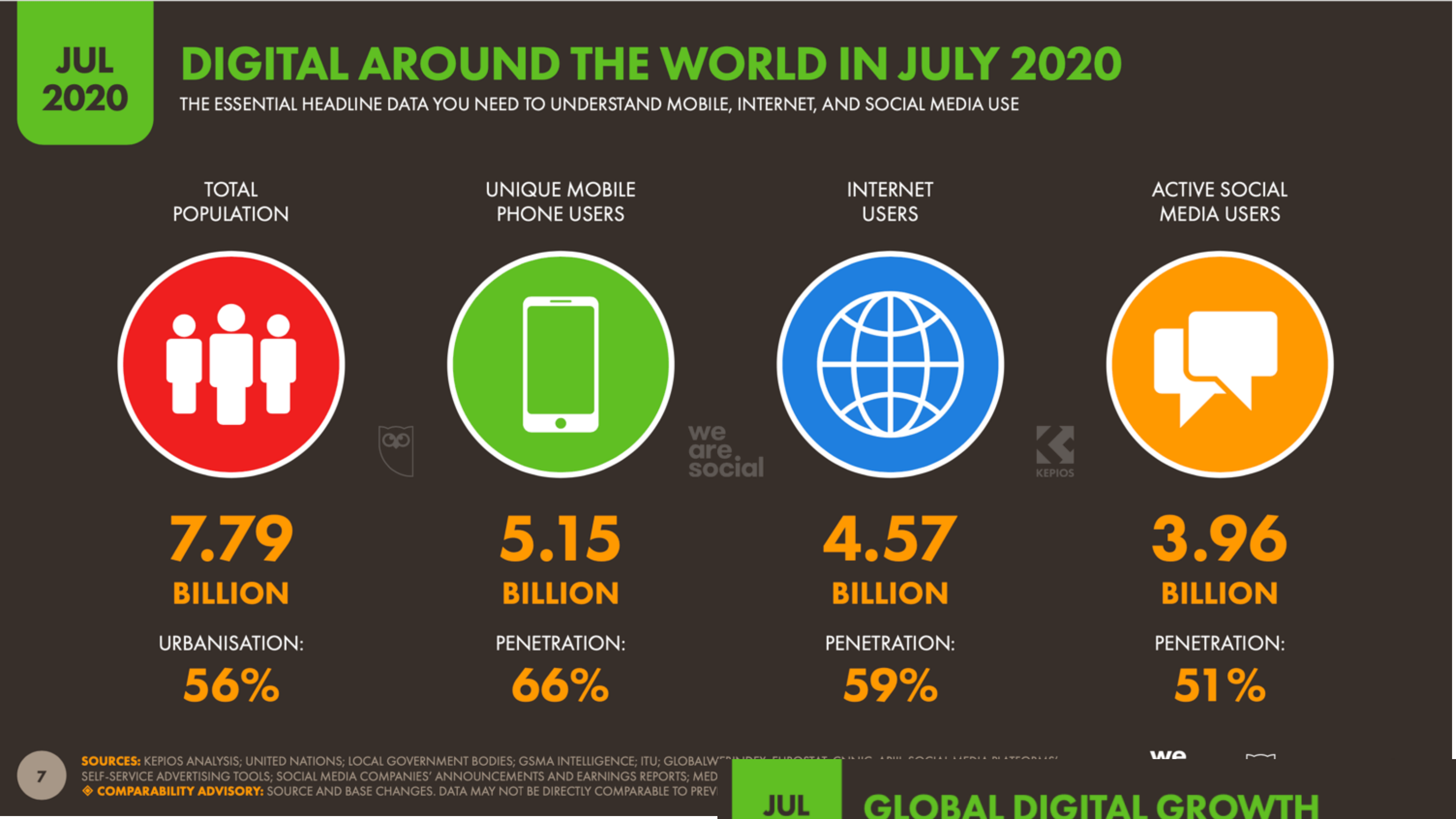
Amazon and Echo Bay have first online sales

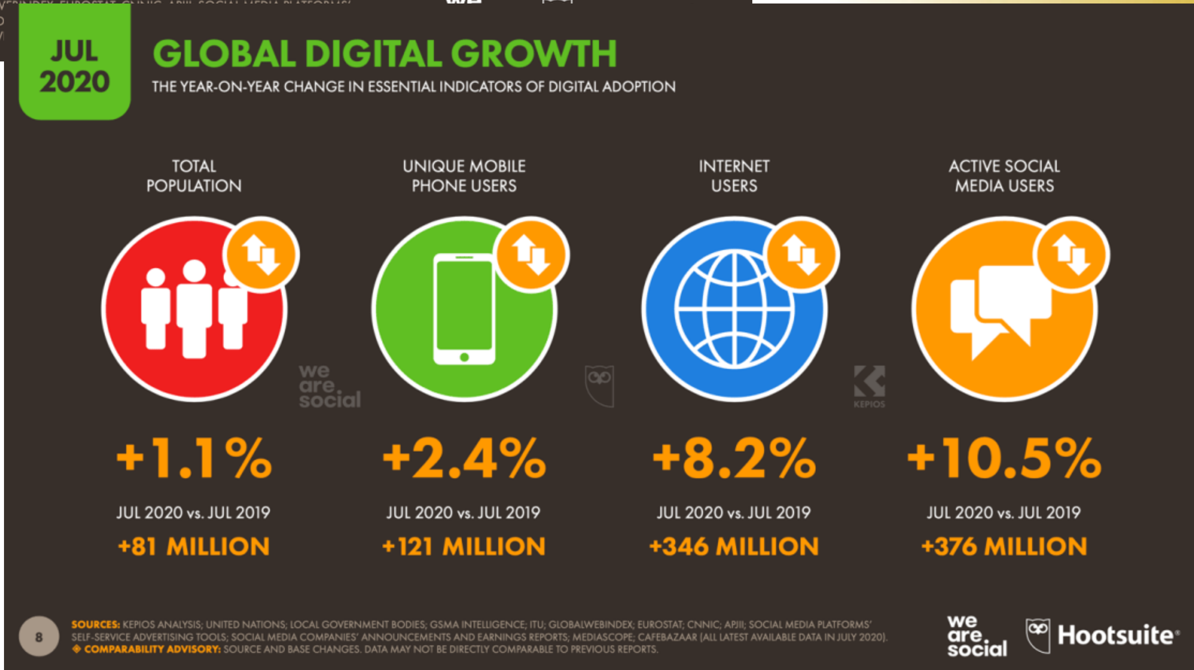
2013 - 2.2 Billion Internet Users



Global Internet Map




JUL 2020
DIGITAL AROUND THE WORLD IN JULY 2020
THE ESSENTIAL HEADLINE DATA YOU NEED TO UNDERSTAND MOBILE, INTERNET, AND SOCIAL MEDIA USE
TOTAL POPULATION
7.79 BILLION
URBANISATION: 56%
UNIQUE MOBILE PHONE USERS
5.15 BILLION
PENETRATION: 66%
INTERNET USERS
4.57 BILLION
PENETRATION: 59%
ACTIVE SOCIAL MEDIA USERS
3.96 BILLION
PENETRATION: 51%
we are social
KEPIOS
7
SOURCES: KEPIOS ANALYSIS; UNITED NATIONS; LOCAL GOVERNMENT BODIES; GSMA INTELLIGENCE; ITU; GLOBALW
SELF-SERVICE ADVERTISING TOOLS; SOCIAL MEDIA COMPANIES' ANNOUNCEMENTS AND EARNINGS REPORTS; MED
COMPARABILITY ADVISORY: SOURCE AND BASE CHANGES. DATA MAY NOT BE DIRECTLY COMPARABLE TO PREV


JUL 2020
GLOBAL DIGITAL GROWTH
THE YEAR-ON-YEAR CHANGE IN ESSENTIAL INDICATORS OF DIGITAL ADOPTION
TOTAL POPULATION
+1.1%
JUL 2020 vs. JUL 2019
+81 MILLION
UNIQUE MOBILE PHONE USERS
+2.4%
JUL 2020 vs. JUL 2019
+121 MILLION
INTERNET USERS
+8.2%
JUL 2020 vs. JUL 2019
+346 MILLION
ACTIVE SOCIAL MEDIA USERS
+10.5%
JUL 2020 vs. JUL 2019
+376 MILLION
we are social
KEPIOS
8
SOURCES: KEPIOS ANALYSIS; UNITED NATIONS; LOCAL GOVERNMENT BODIES; GSMA INTELLIGENCE; ITU; GLOBALWEBINDEX; EUROSTAT; CNNIC; APJII; SOCIAL MEDIA PLATFORMS' SELF-SERVICE ADVERTISING TOOLS; SOCIAL MEDIA COMPANIES' ANNOUNCEMENTS AND EARNINGS REPORTS; MEDIASCOPE; CAFEBAZAAR (ALL LATEST AVAILABLE DATA IN JULY 2020).
COMPARABILITY ADVISORY: SOURCE AND BASE CHANGES. DATA MAY NOT BE DIRECTLY COMPARABLE TO PREVIOUS REPORTS.
we are social
Hootsuite

Classification of computer networks

Due to the territorial scope:

WANs Wide Area Networks
cover countries and even entire continents;
significant propagation delays
susceptibility to damage to transmission links
bandwidth - formerly: from several kb/s to several dozen Mb/s - copper, radio links
- currently: Mb/s to tens/ hundreds of Gb/s - optical fibers
high complexity - complicated maintenance and management procedures;
various arrangements required.

MANs - city or Metropolitan Area Networks
they connect nodes and stations located in an area with a diameter of up to about 50 km
usually cover the area of the city or housing estates
transmissions speed in such networks range from a few Mb/s to hundreds of Gb/s.
limited complexity

LANs - **Local Area Networks**
include from several to several dozen independent devices arranged in a small area
low construction costs and high reliability of such networks
transmission speeds range from 1 Mb/s to one (or even more) hundred Gb/s.

In case of wireless networks also: PANs (Personal Area Networks) and BANs (Body Are Networks)

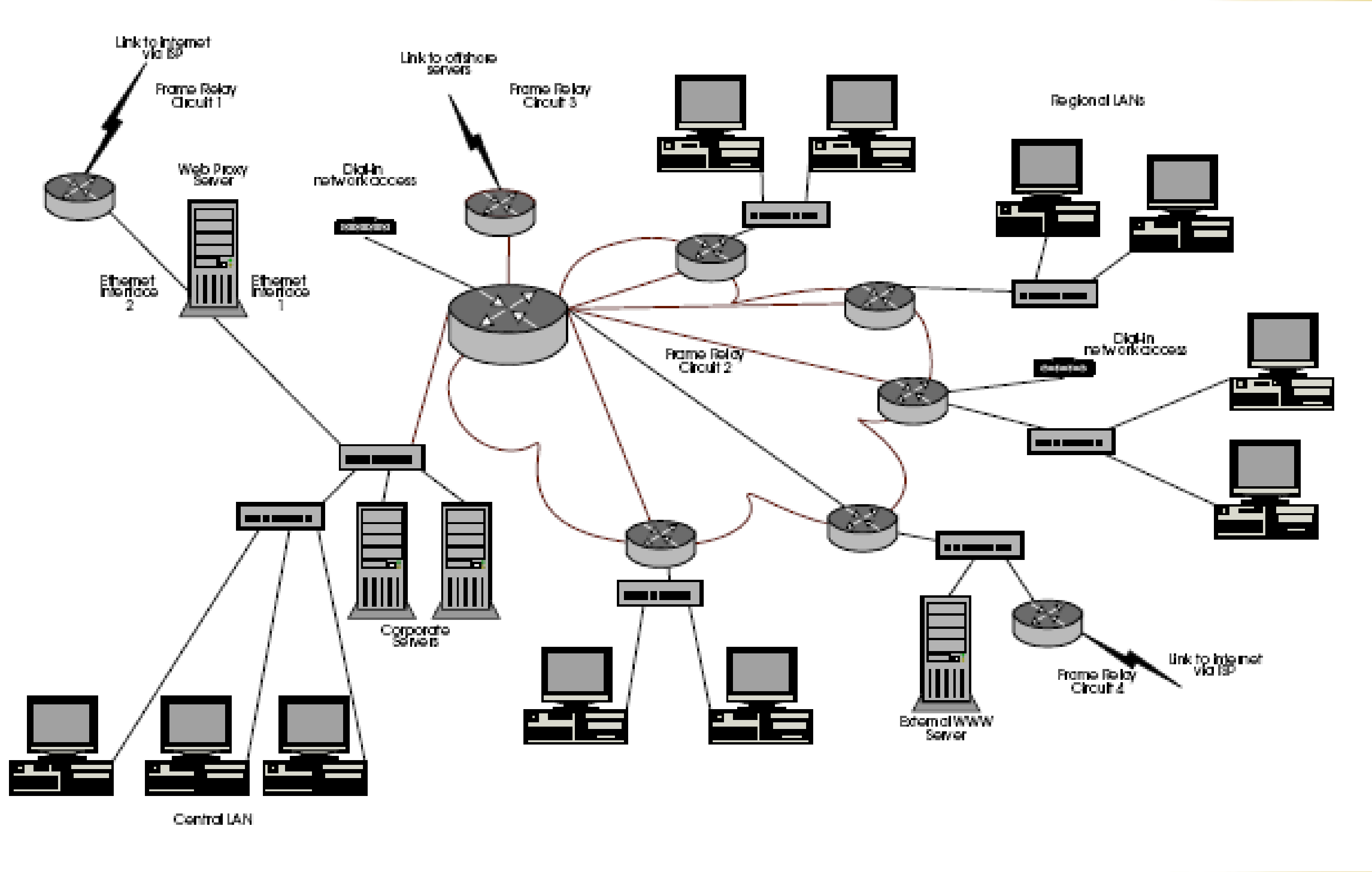
The main components that describe and form the network

- Topologies
- Architectures
- Protocols

POLITECHNIKA GDAŃSKA

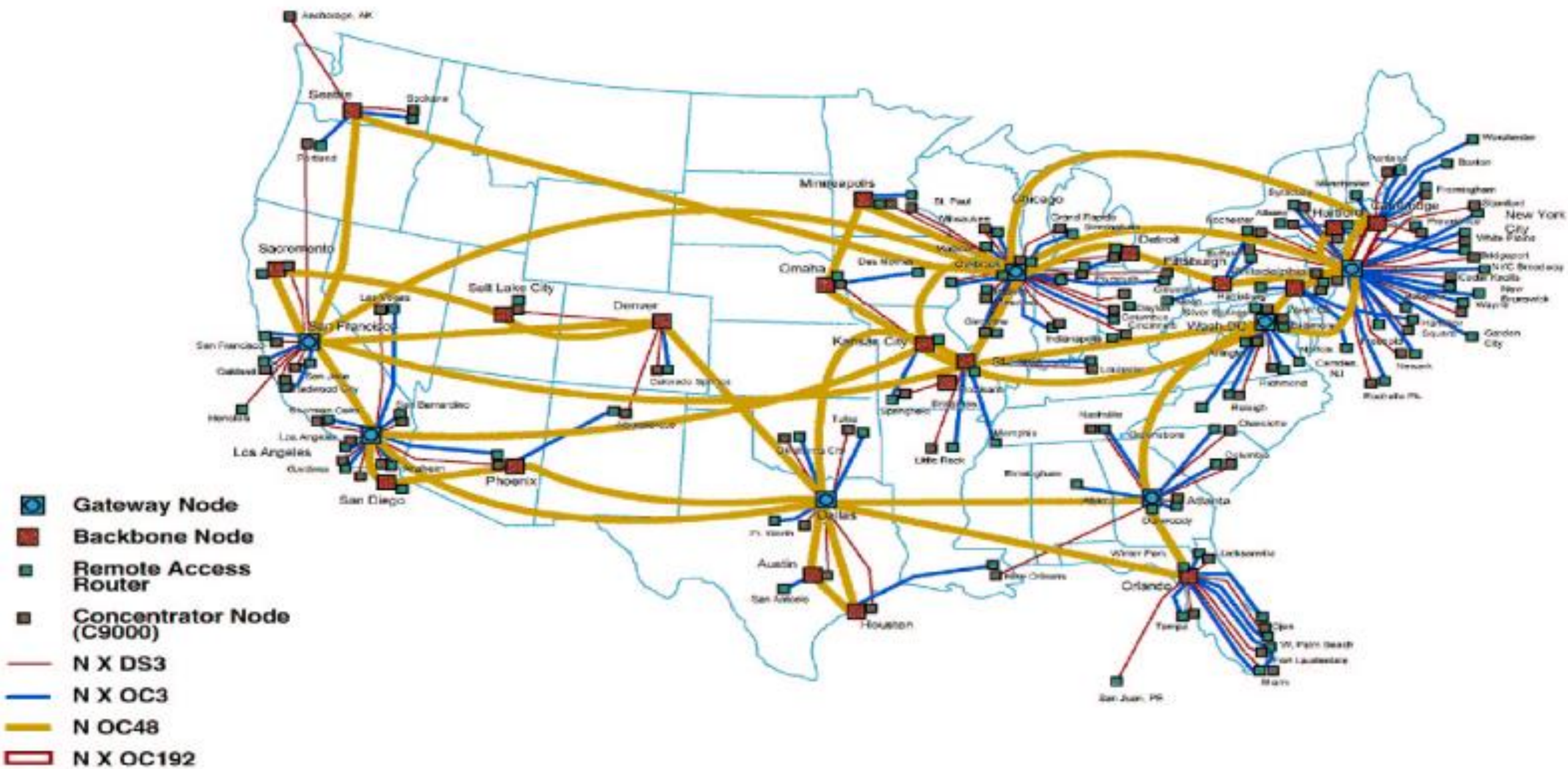
An example of a MAN network

A corporate network



WAN example: a slightly surprising concentration of connections.

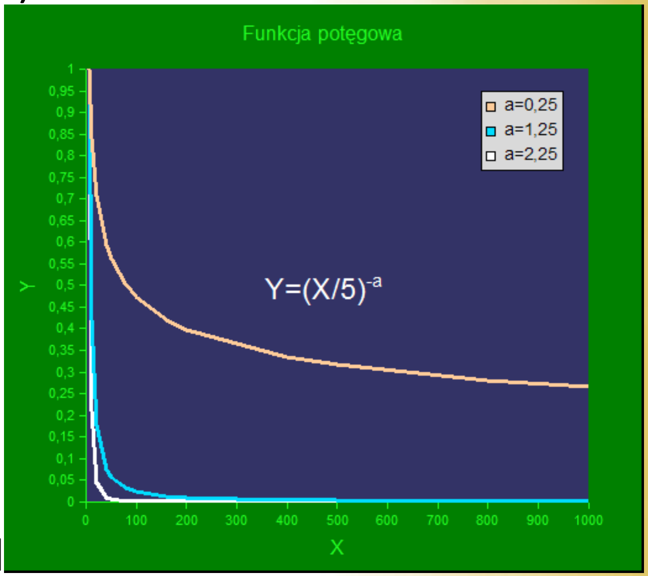
What are the consequences of this?



Scale-free networks >>> e.g. the Internet

In the case of networks, there is a more or less natural tendency to create nodes - network hubs with a large number of connections (**scale-free** networks).

In graph theory, it is a network in which the distribution of the number of connections between nodes complies the power low:



$$P(k) \sim k^{-\gamma}$$

where γ is a parameter appropriate for a given network and usually with values in the range (2,3).

So we have a power distribution!

Various networks, both self-generated and man-made, such as the Internet, social networks, interactions between proteins in living organisms, and others are considered examples of **scale-free** networks. The mechanism of the distribution characteristic of **scale-free** networks is called preferential linkages. **Scale-free** networks grow in some characteristic way. Subsequent nodes are more likely to join those that have a higher rank (more connections).

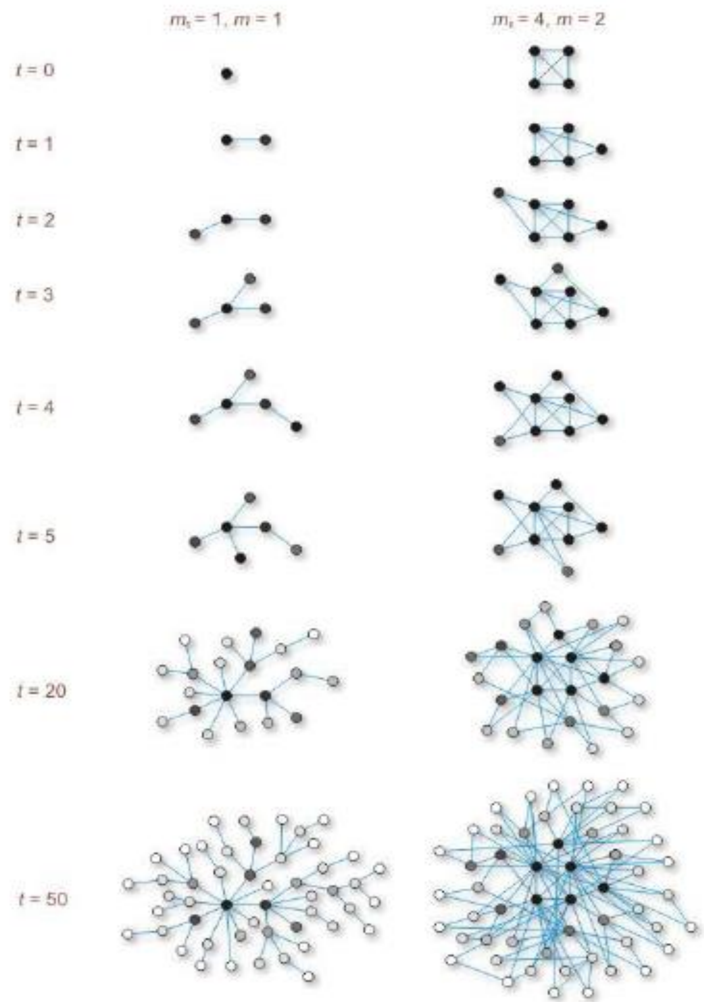
Thus, not necessarily desirable "hubs" (centers) are created – they are potentially more vulnerable to attacks.

We notice a tendency towards a strong concentration of traffic - an increase in the number of connections for selected nodes - scale-free networks

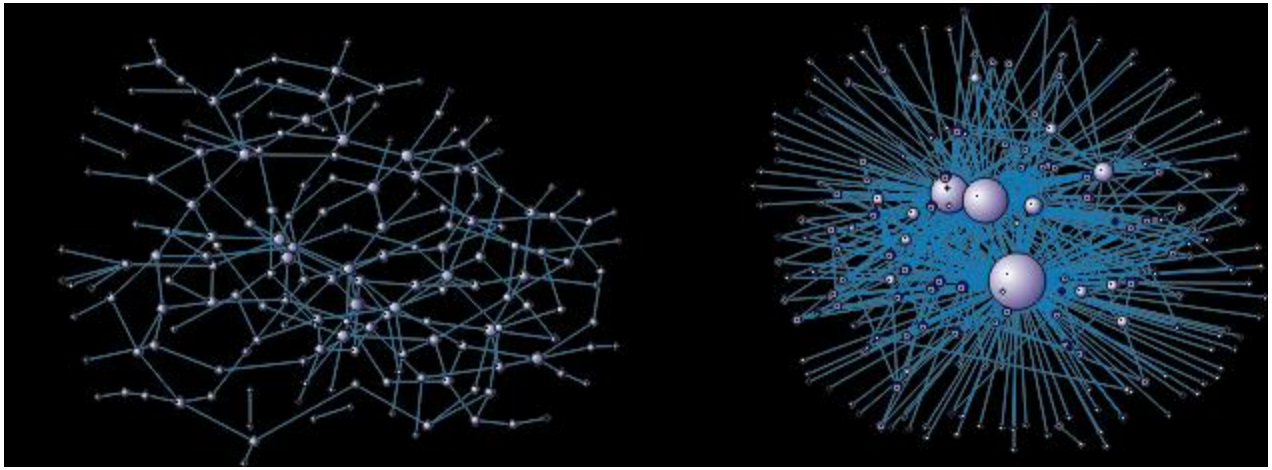
Why scale-free networks?

The probability that a new node (vertex in a graph) connects to one of the older nodes of degree "i" is directly proportional to the degree of that node.

tło żółte



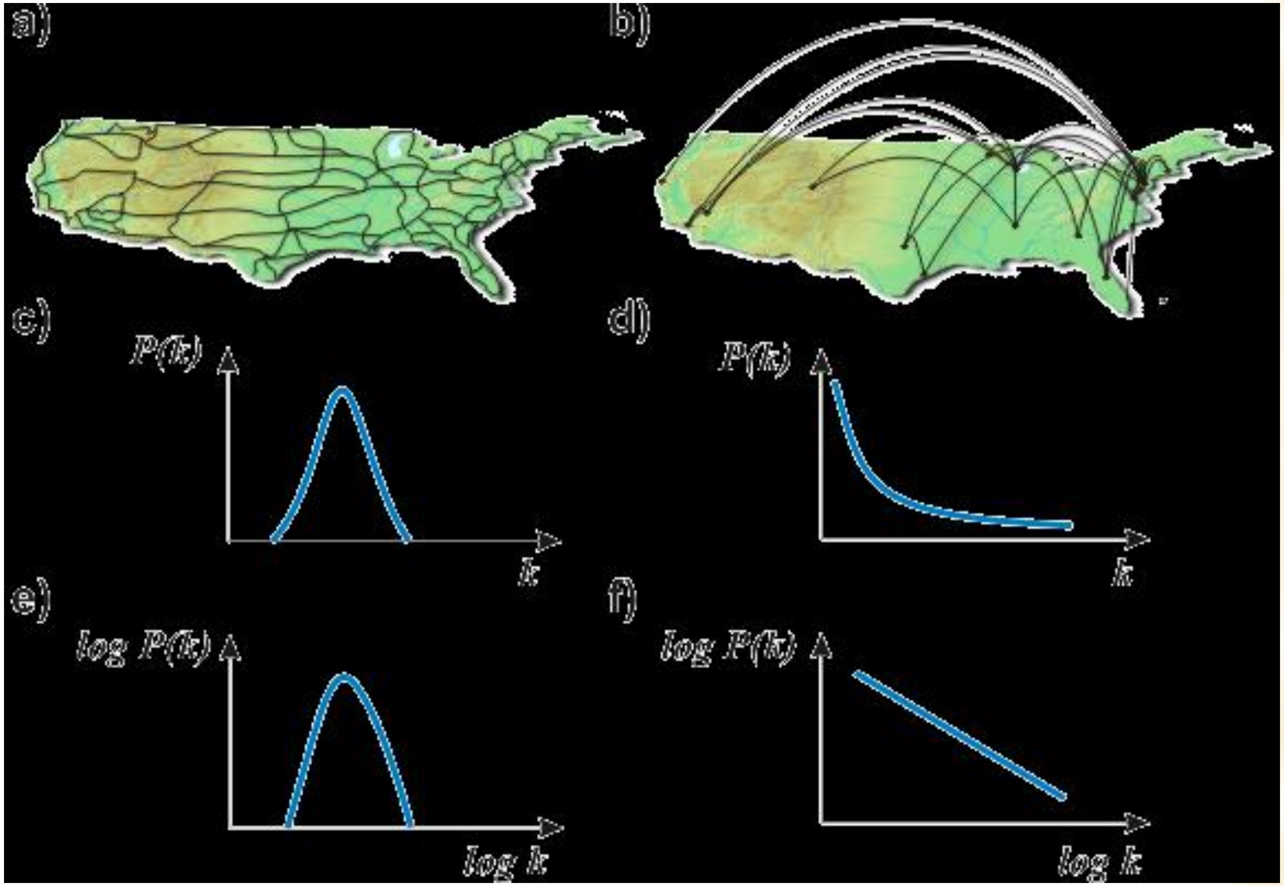
Scale-free networks – power low



We see two networks with the same number of nodes N (vertices) and the same number of E edges (links). The size of the nodes in each of these networks is proportional to their degrees. What distinguishes these network topologies from each other is the distribution P (k) of degrees of the vertices (nodes). This distribution tells us how many nodes there are in a given network with a small and how many with a large number of connections. Generally, P (k) represents the fraction of all nodes in the network that have degree k. We can also say that P (k) describes the probability that a randomly selected node will have degree k.

Although we lose a lot of valuable information by simplifying real systems to a set of nodes and edges (that is, presenting them as a network), it has many advantages. First of all, by doing this, we can apply the same analysis tools to quite different systems. This is how the **Hungarian physicist Albert-Laszlo Barabasi made an astonishing observation at the end of the last century**. He found that different real networks describing completely different systems have a universal property: the distribution of vertex degrees in these networks is of a power character (similar to the network on the right side of the figure).
http://www.if.pw.edu.pl/~agatka/moodle/charakterystyki.html

Not all types of networks show this trend



US highway network (a) and airline network (b). c - f - degree distributions of the vertices of these networks, presented in various scales.

Scale-free networks - a task proposal

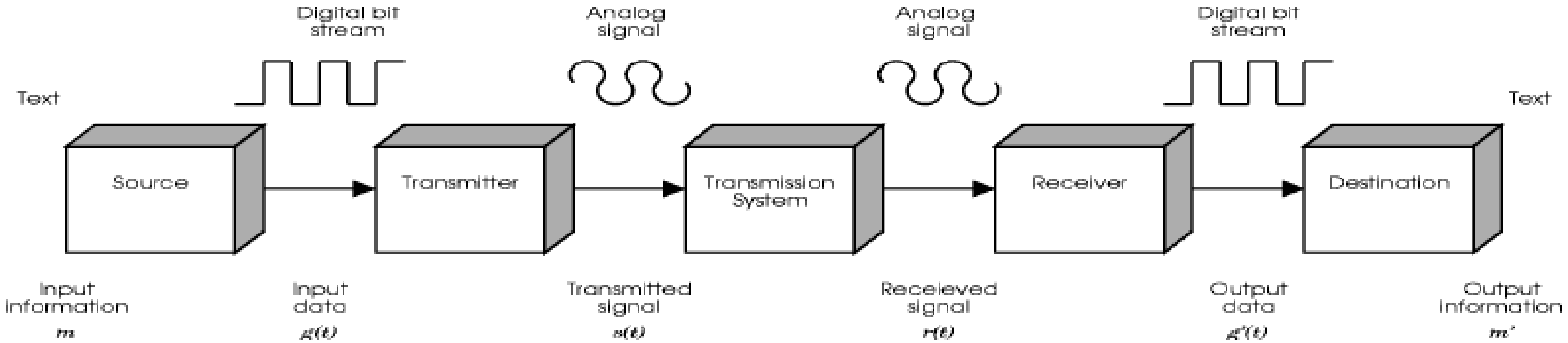
- **Note:** An interesting study (a short description and illustrations in ppt, plus references used): interesting examples of scale-free networks - interesting facts about their exploration and growth rate - generation algorithms - the sooner the better - max. within 1 month - an award for those willing to prepare a report, at the end of the semester (but for no more than 3 people).

Selected communication tasks

- Multiple transmissions - sharing the medium
  Network interfaces
  Signal generation
  Synchronization
  Data formatting
  Segmentation of data
  Exchange management - point-to-point control
  Encoding methods and cryptographic security (identification, authentication, authorization)
  Detection and, if necessary correction of errors
  Return to a proper operation

Addressing schemes
Switching
Routing strategies
Network management
Flow control: node-to-node @ end-to-end control
Data compression
Application identification
Correct implementation of network services and applications, etc.

And many other tasks and functionalities on different levels of abstraction



Problems with creating ICT networks

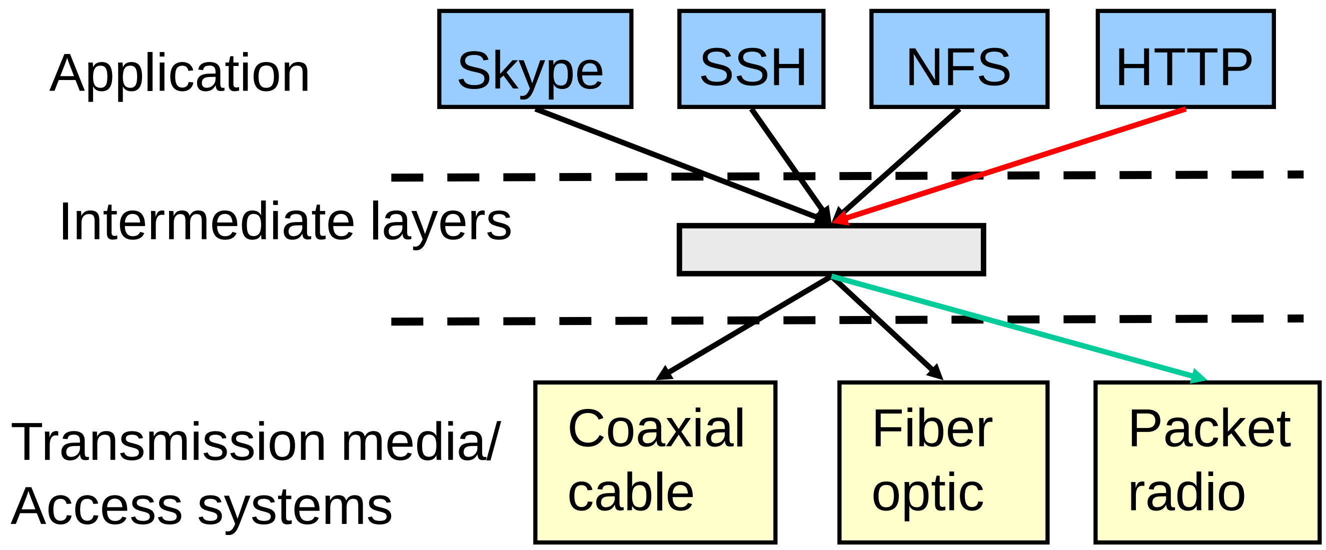
- Many, sometimes completely different network technologies;
  A lot of different applications;
- How to ensure efficient communication, avoiding the enormous complexity of the network and the incompatibility of its components ???

- Should we adapt each application to different technologies?
  - **Definitely not!**

  But how to design the internet avoiding undesirable problems?

Solution: Intermediate layers

- Introduction of intermediate layers providing a set of abstractions for various functionalities and network technologies;
- New app/ media only implemented once;
- Changes and differentiation at "higher intermediate levels".



Proposed solution: a multi-level network protocols architecture

- Architecture is not an implementation in itself; Architecture tells us:

- How to organize/assemble system elements and their implementations;
- What interfaces are to be defined and supported:
- What is the level of abstraction,
- How many layers should be foreseen,
- Where to implement a specific functionality.

- Correctly designed architecture is aimed at ensuring effective (and universal) operation of the network, and at the same time guaranteeing the modularity of network structure.

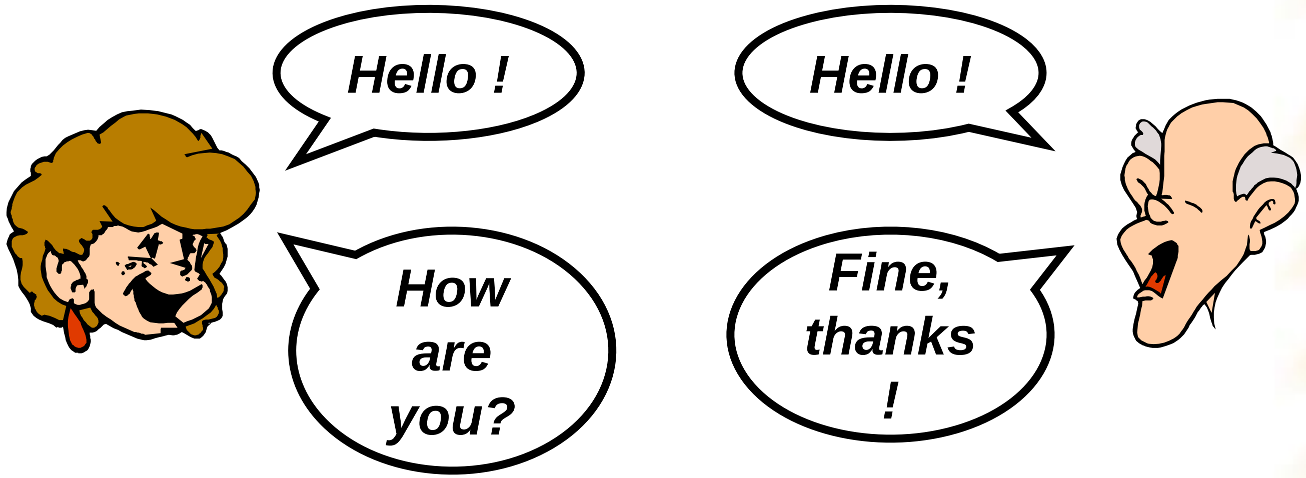
Layers

- Why layers?
  They decompose very complex problems of network communication into a series of better managed fragments;
  They guarantee greater modularity (easier adding new services or modifying the functionality of layers);

- We start building a layered architecture by defining services guaranteed by hardware components, then we describe the sequences of layers that offer transport services to layers located directly above the hardware layer.

Protocols

- They define the message formats - data units as well as rules, including the order in which they are delivered between two communicating network entities.
  They define actions taken when sending and receiving messages or related to other events.
- They allow to perform required network functionalities.
- Example of a layered set of protocols:
  HTTP (for web browsers) uses TCP (for reliable packet transfer), which in turn uses IP (for global node addressing and datagram forwarding). Then we implement local solutions - using Ethernet / WiFi (with MAC addressing).



POLITECHNIKA GDAŃSKA

4 basic protocol features

A set of rules governing communication between two layered entities (**peer entities and peer communication**).

They define formats and order of exchanged messages and the actions taken in sending and receiving them.

Syntax: format> what is the correct structure of the message?

"GET /~hugue/index.html HTTP/1.1\nHOST: www.cs.umd.edu\n\n"

Semantics: what is the meaning of the message?

Get file/~hugue/index.html using the http 1.1 protocol.

Actions: read file/~hugue/index.html from the disk, send it through the socket using the http 1.1 protocol and close the socket

Timing: proper order of messages. Reply follows request

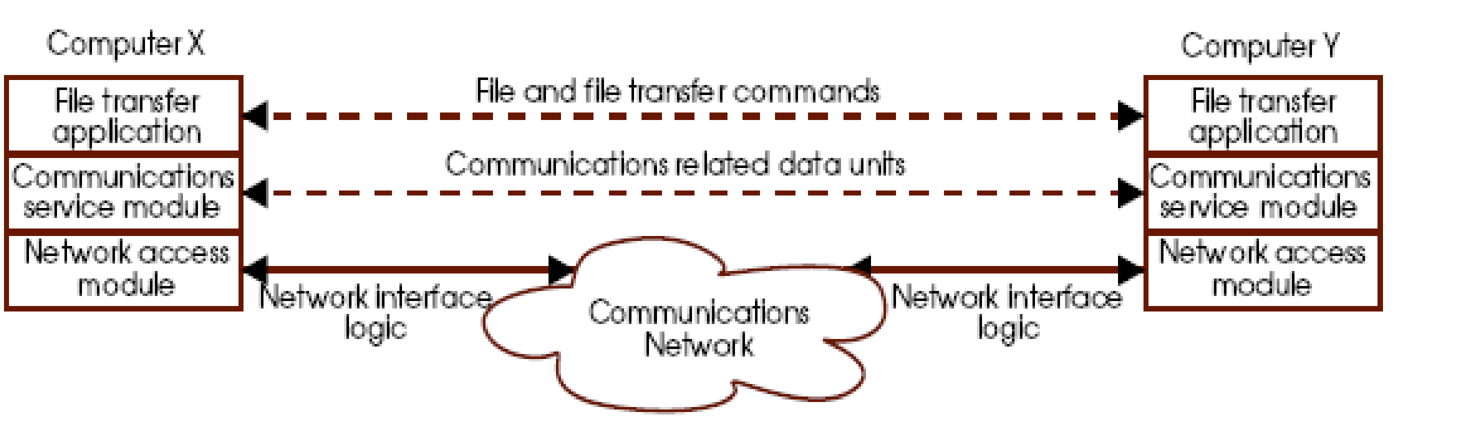
(HTTP - Hypertext Transfer Protocol/ html - HyperText Markup Language)

Realized logical connections: Connection-oriented vs connectionless

- **Connection-oriented transfer mode**:
  The connection is first set up, then data (packets) are sent along a communication path (estabilished in a physical or logical way) – modeled on the telephony solutions.
- **Connectionless transfer mode:**
  Mechanism of forwarding data, based on the snail mail solutions - packets are sent through the network "individually" (datagram-based) without setting up a connection ealier.
- Broadcast vs point-to-point (multicast - anycast)

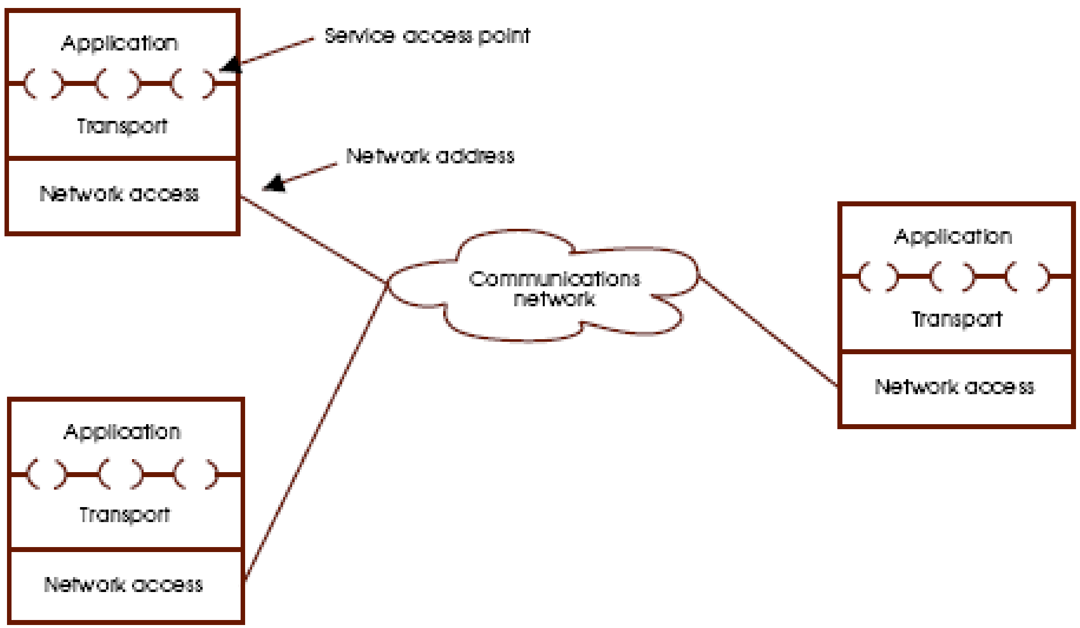
Protocol architectures – a simple example of file transfer

- The protocol architecture provides a defined set of modules for various communication functions:
  File transfer logic:
  Transmission of passwords, commands and messages related to the exchange of files, etc. (unambiguous interpretation and presentation of data)
  Communications logic:
  It ensures: readiness to exchange data between two stations, tracking and supervision of the exchange, guarantees the integrity of transfers, etc. (reliable data transfer)
  Network logic:
  Responsible for physical connections, signaling, etc. (efficient information transfer over the communication subnet)

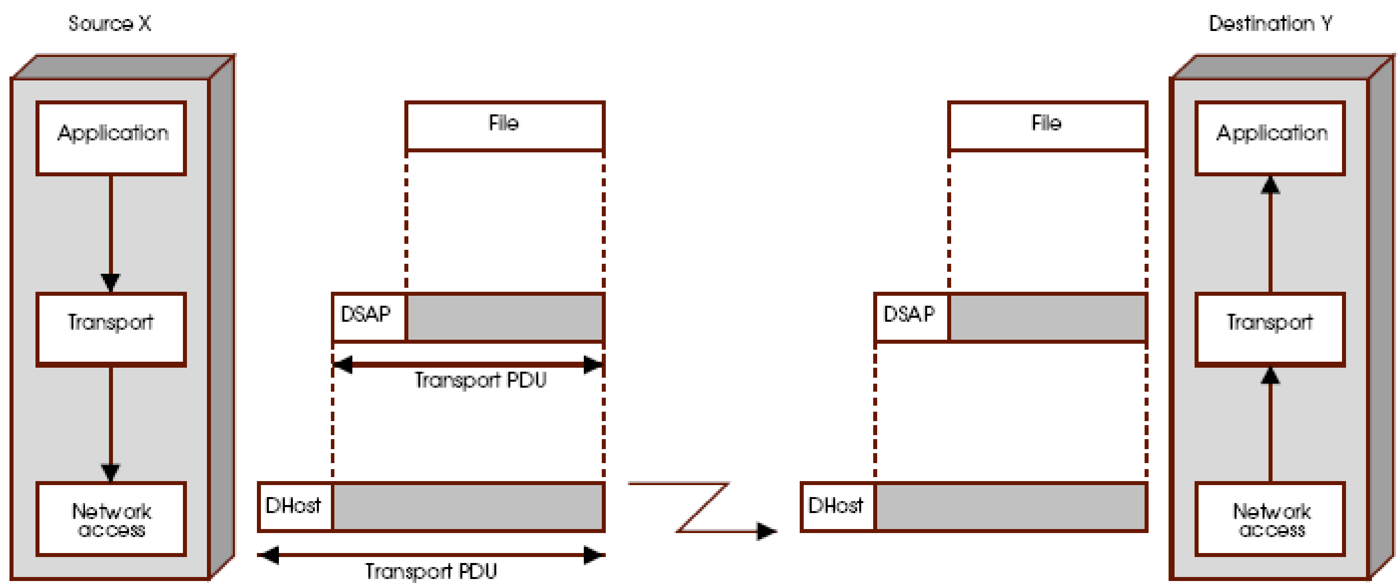


Protocol architectures - simple example of file transfer

- Application layer
- Transport layer
- Network access layer



Protocol architectures: Data encapsulation



Encapsulation is an operation of nesting a given data unit in a new data structure, i.e. wrapping the "old stucture" in a new package.
This is due to the necessity of adding fields specific for a given layer, enabling the implementation of the parameters and functionalities provided (guaranteed) by this layer (and a specific protocol).

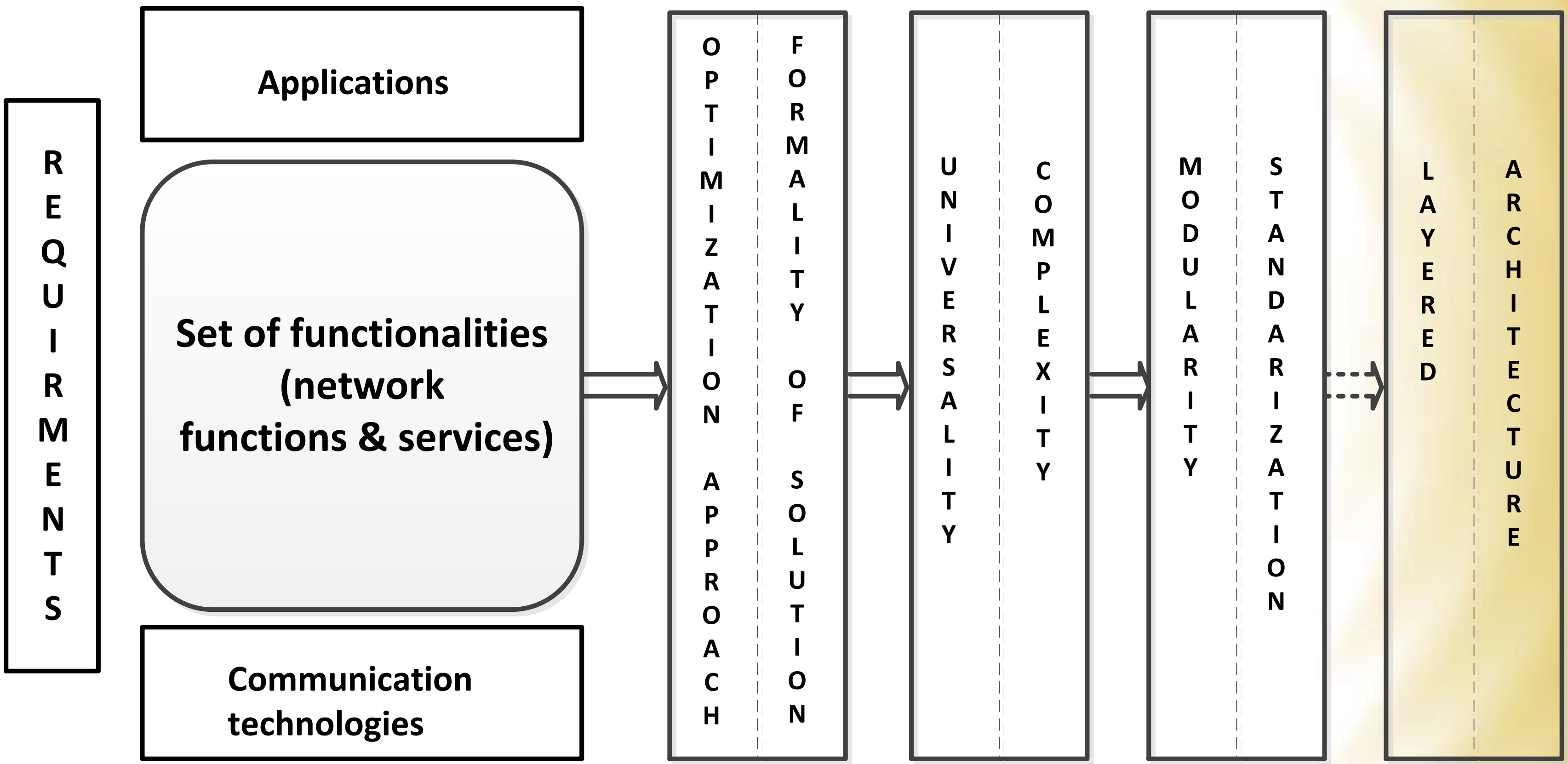
Layered network architectures

Why are network systems designed structurally?

- the need to ensure mutual multifaceted cooperation of computer equipment,
- to enable replacement/ modification of individual hardware or software modules,
- to reduce network complexity.

Additionally:
- we decompose complex tasks and network functionalities into selected groups in terms of the level of abstraction and communication requirements,
- we organize different functions, responsible for the implementation of the effective and reliable transfer of information between end stations, in a form of independent (and often isolated) layers.

All network software, together with hardware components, create the so-called layered architecture.


REQUIRMENTS
Applications
Set of functionalities (network functions & services)
Communication technologies
OPTIMIZATION APPROACH
FORMALITY OF SOLUTION
UNIVERSALITY
COMPLEXITY
MODULARITY
STANDARIZATION
LAYERED
ARCHITECTURE

Layered network architectures

Major layered architectures:

ISO-OSI (International Standards Organization - Open Systems Interconection),
TCP / IP (Transmission Control Protocol/ Internet Protocol),
SNA (Systems Network Architecture) by IBM,
DNA (Digital Network Architecture) by DEC.

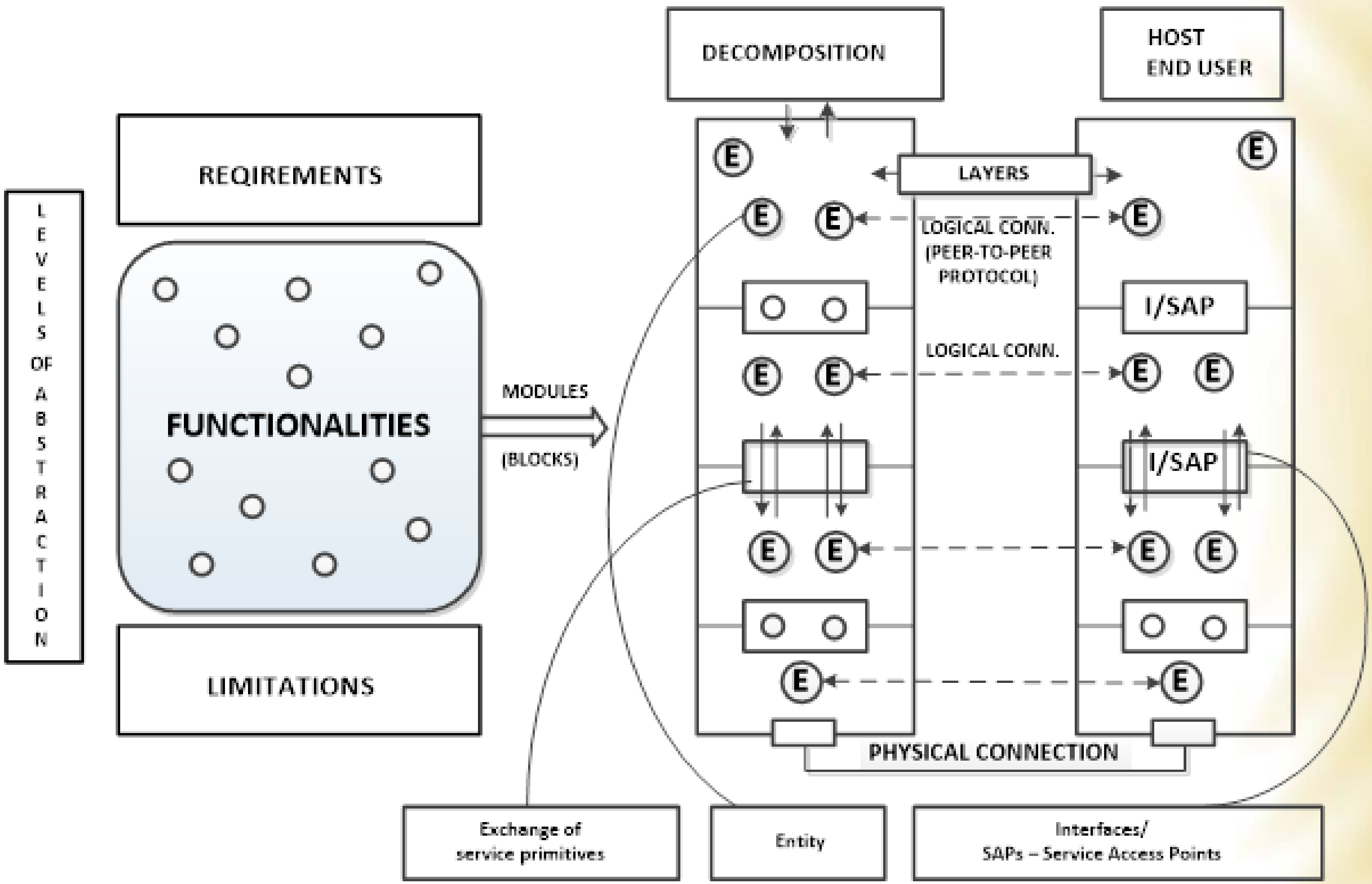
Main differences:

number of layers,
the way of their implementation,
rules for establishing connections between stations.

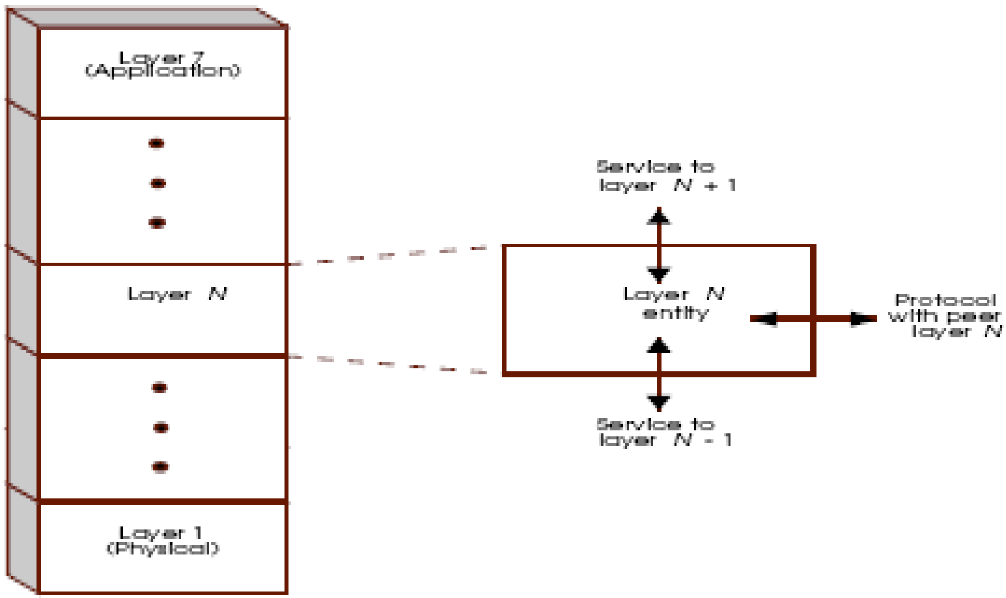
Layered network architectures

Principles of functioning of the layer mechanisms

The task of the N-th layer is to offer transport services for the upper layer, i.e. N + 1;
A set of rules and conventions used in information exchange between N-th layer processes in communicating stations (network nodes) is conventionally called the N-th layer protocol;
Objects (representing protocols) of the same layer, on different computers, execute layered communication, establishing with each other a logical connection.


LEVELS OF ABSTRACTION
REQIREMENTS
FUNCTIONALITIES
LIMITATIONS
MODULES
(BLOCKS)
DECOMPOSITION
HOST
END USER
E
LAYERS
LOGICAL CONN.
(PEER-TO-PEER
PROTOCOL)
I/SAP
LOGICAL CONN.
PHYSICAL CONNECTION
Exchange of
service primitives
Entity
Interfaces/
SAPs – Service Access Points

Layered network architectures:
Their components

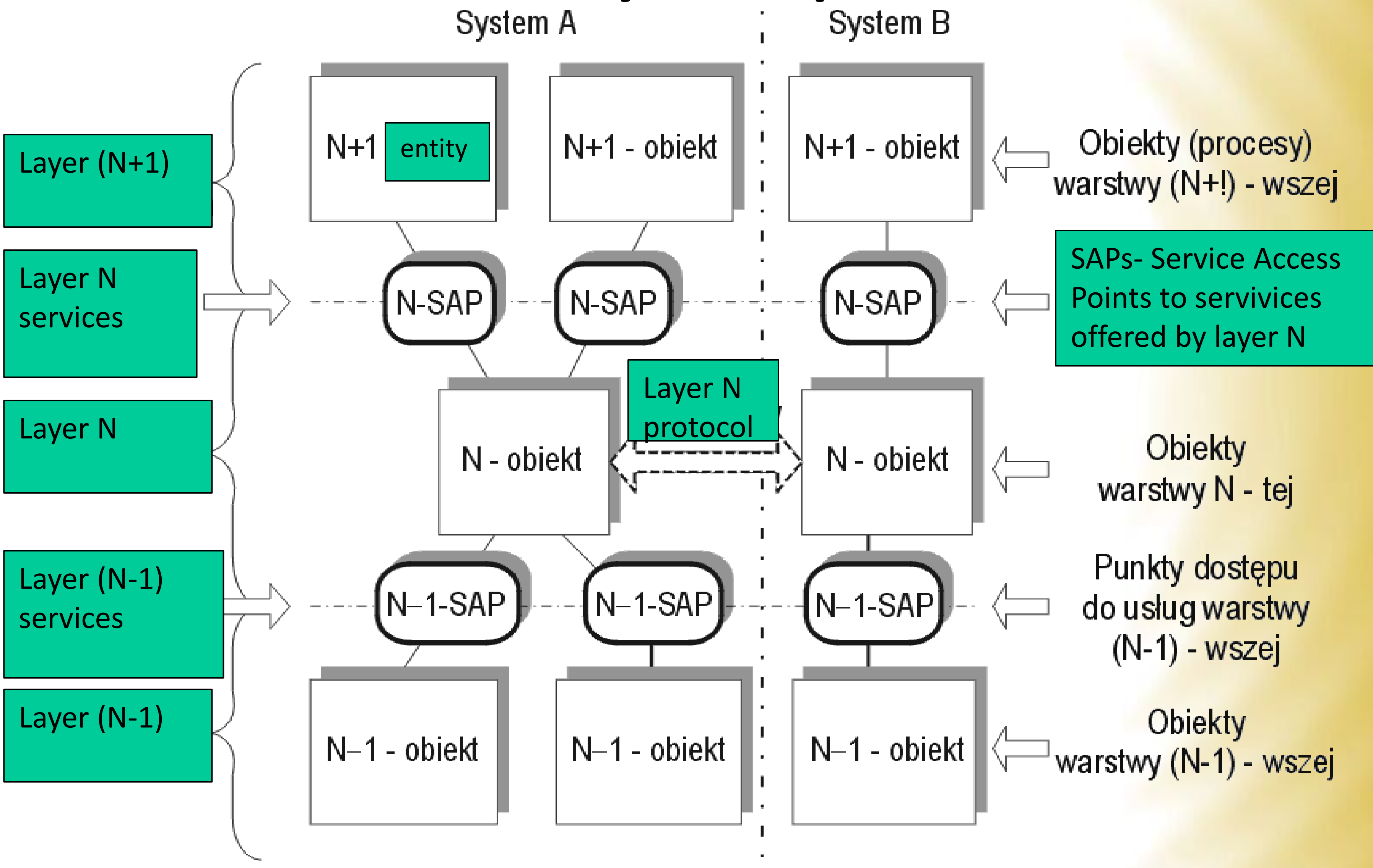


A modular design:

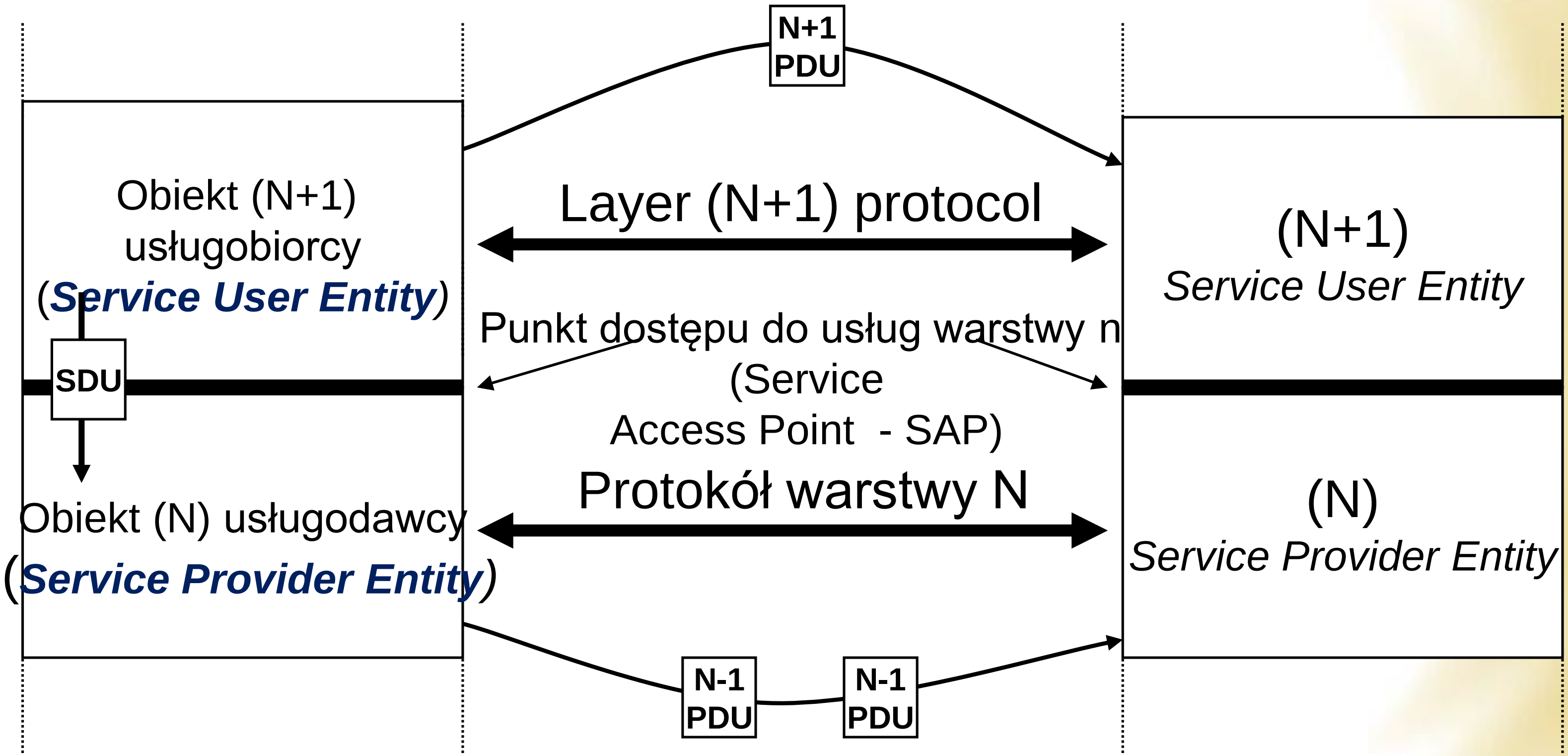
- simple interfaces,
- information hiding is employed,
- lower layers have greater level of detail
- upper layers are independent of such detail.

Layered network architectures

Inter-layer cooperation



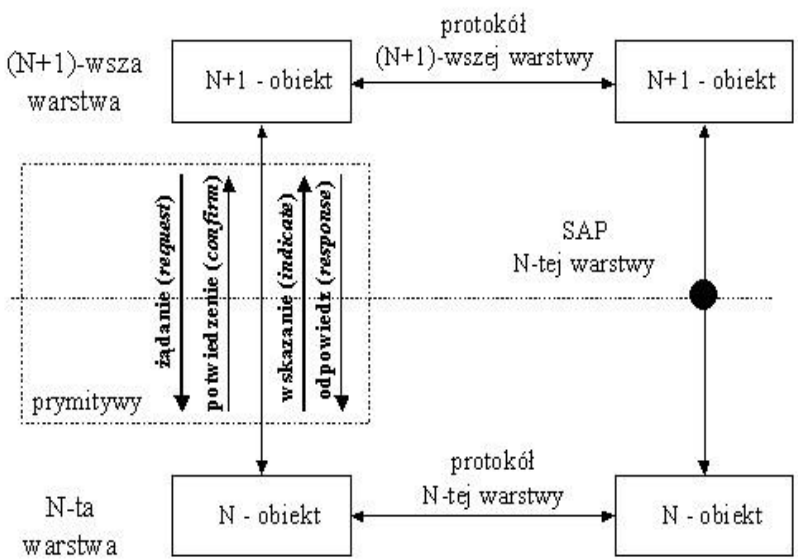
Layered network architectures
Layers collaboration

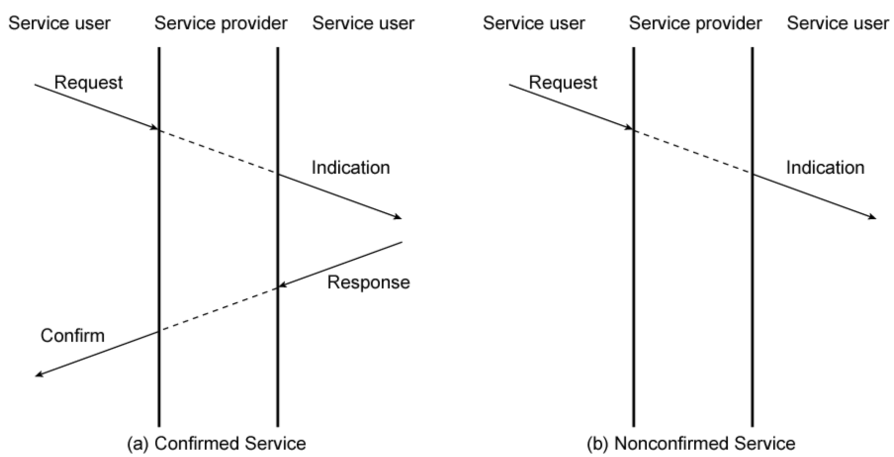


PDU - Protocol Data Unit
SDU - Service Data Unit

Layered network architectures

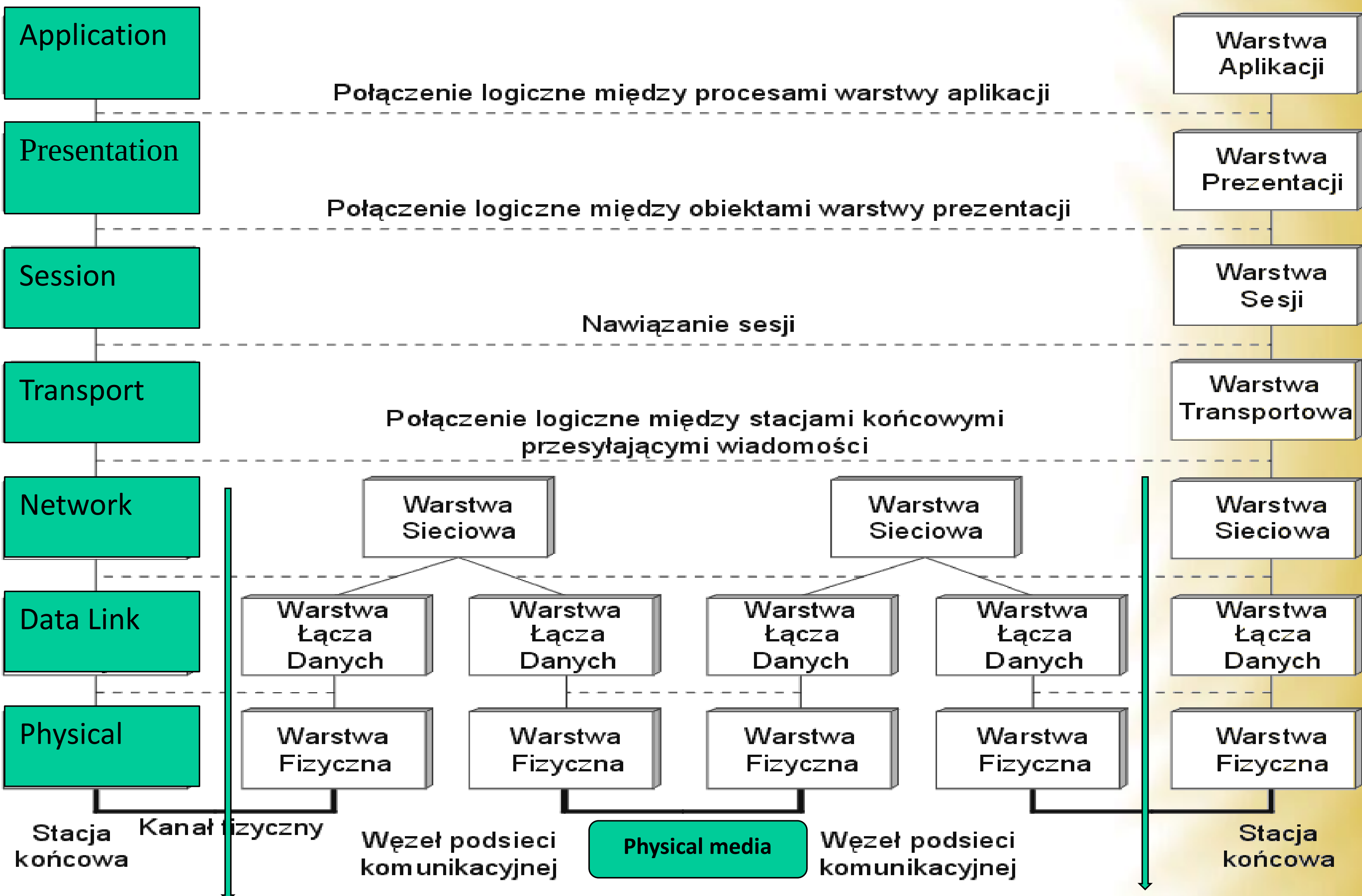
Inter-layer cooperation

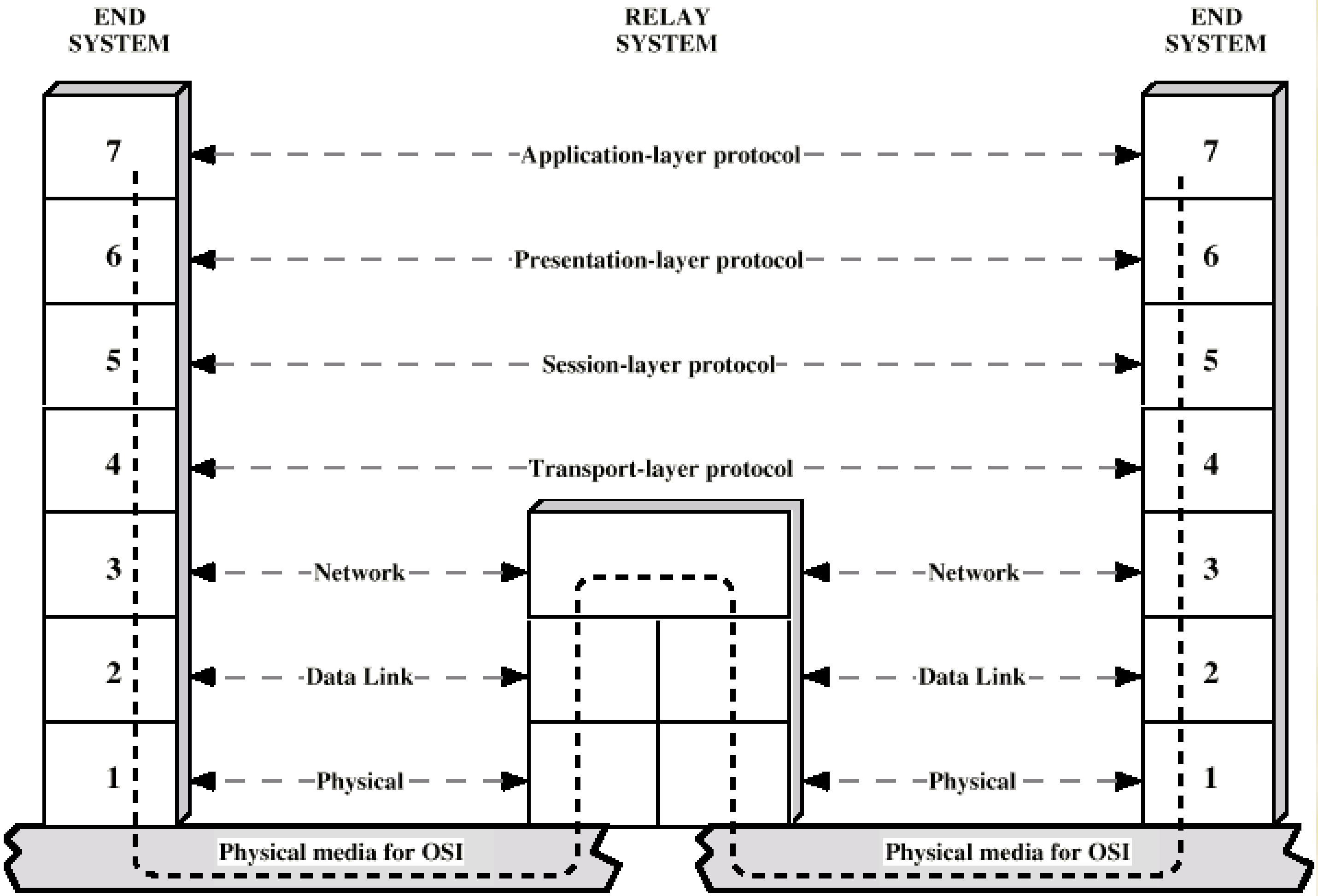




REQUEST	A primitive issued by a service user to invoke some service and to pass the parameters needed to specify fully the requested service.
INDICATION	A primitive issued by a service provider either to: indicate that a procedure has been invoked by the peer service user on the connection and to provide the associated parameters, or notify the service user of a provider-initiated action.
RESPONSE	A primitive issued by a service user to acknowledge or complete some procedure previously invoked by an indication to that user.
CONFIRM	A primitive issued by a service provider to acknowledge or complete some procedure previously invoked by a request by the service user.

ISO OSI Architecture




END SYSTEM
RELAY SYSTEM
END SYSTEM
7
6
5
4
3
2
1
Application-layer protocol
Presentation-layer protocol
Session-layer protocol
Transport-layer protocol
Network
Data Link
Physical
Network
Data Link
Physical
7
6
5
4
3
2
1
Physical media for OSI
Physical media for OSI

ISO OSI Architecture

Physical layer

Defines the rules by which bits flow between communicating stations

Takes into account the need of synchronization;

Adapts the form of signals to the properties of the medium;

Defines a hardware interface … described by four characteristics: *mechanical, electrical, functional and procedural*;

An example of a physical layer solution is the RS 232 C interface.

Data link layer

Creates data frames;

Ensures reliable transmission of frames over a noisy digital channel;

Offers the point-to-point frame flow control;

Controls access to the communication medium;

Examples of this layer protocols are : HDLC (ISO) and IEEE 802.x standards

ISO OSI Architecture

Network layer
Responsible for route selection between source and destination station;
Offers network protection against overloads (and deadlocks);
Ensures "transparent" information transfer between networks (it does not interfere with the content of the transmitted data);
Provides segmentation and re-segmentation of transmitted packets.

Transport layer
Ensures reliable and "transparent" data transfer between end stations (end-to-end control);
Offers detection of erroneously transmitted/retransmitted or lost data blocks;
Provides end-to-end flow control with the use of a sliding window mechanism as well as congestion control.

ISO OSI Architecture

Session layer

Responsible for:

Establishing and termination a session and managing the connection (session) between two processes;

Dialog control between processes - determining which from communicating processes, when and for how long may send information.

Presentation layer

Convertion of user data to a standard form used in the network;

Encryption of transmitted information;

Provision of data compression;

Convertion of formats and data types.

ISO OSI Architecture

Application layer

Provides users with services offered by the OSI environment, enabling, in particular:

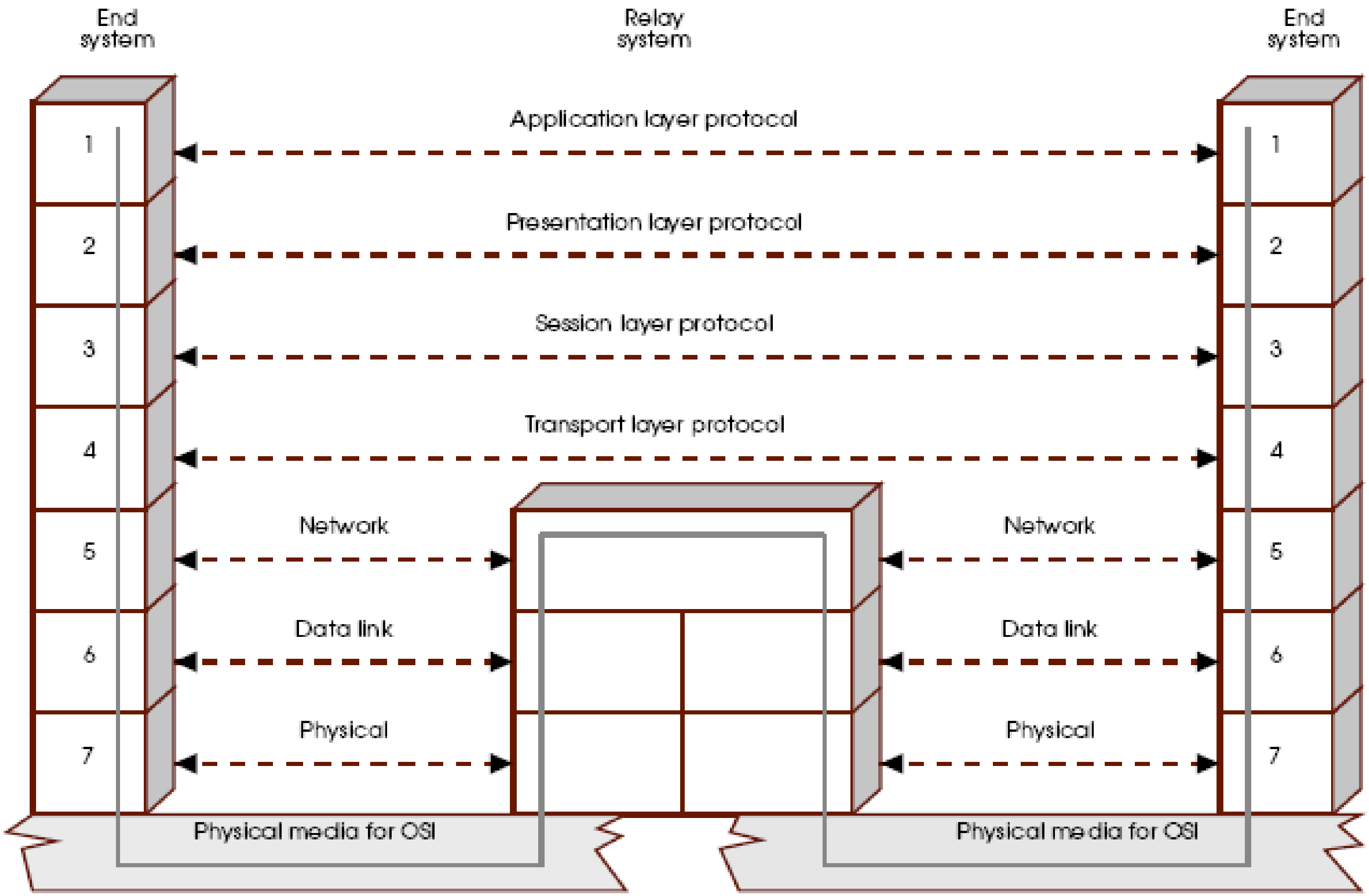
file transfer and operation on remote files,

access and operation on remote databases,

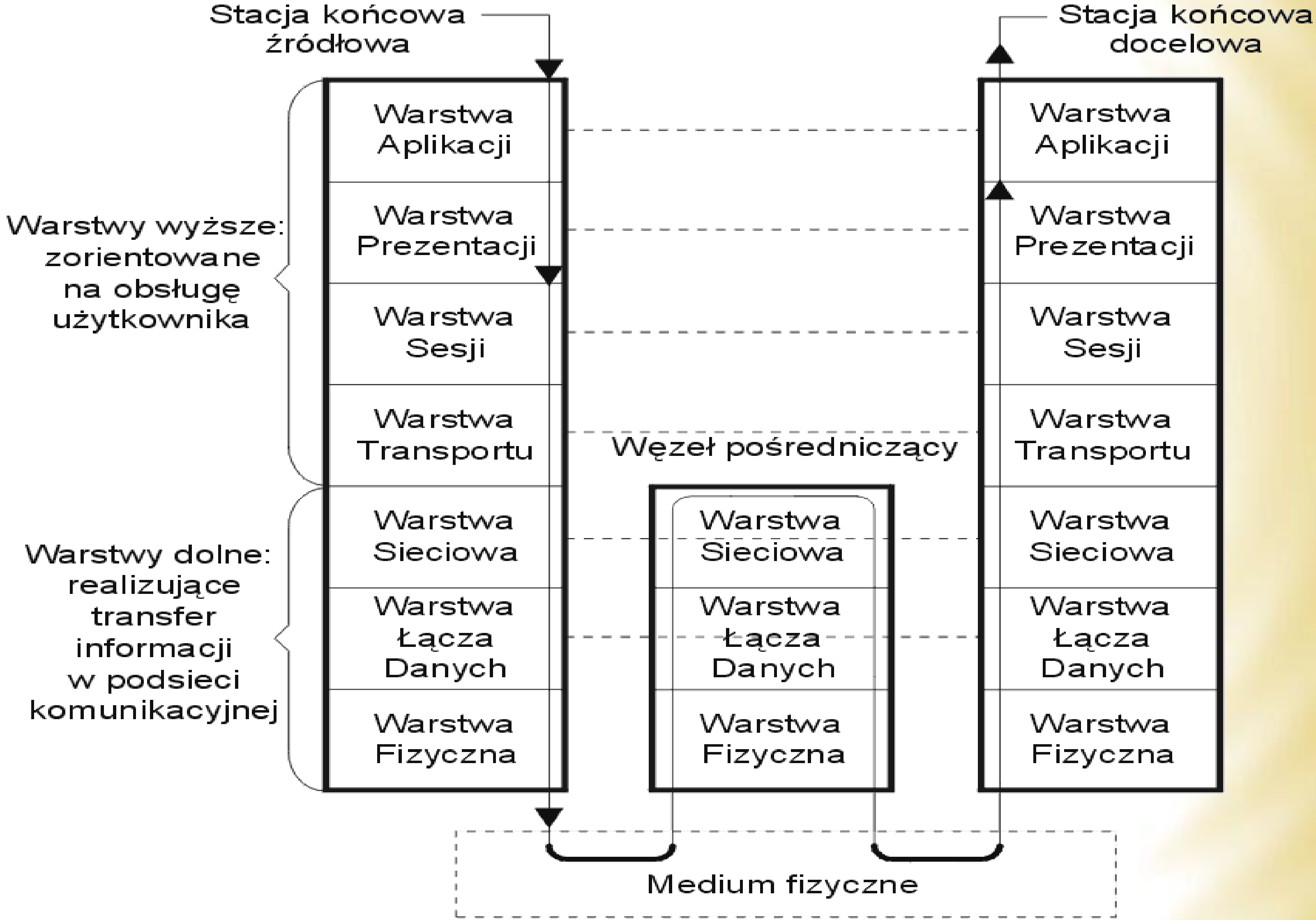
execution of remote computing tasks,

sending and receiving electronic mail.

ISO OSI Architecture
Data flow over the network

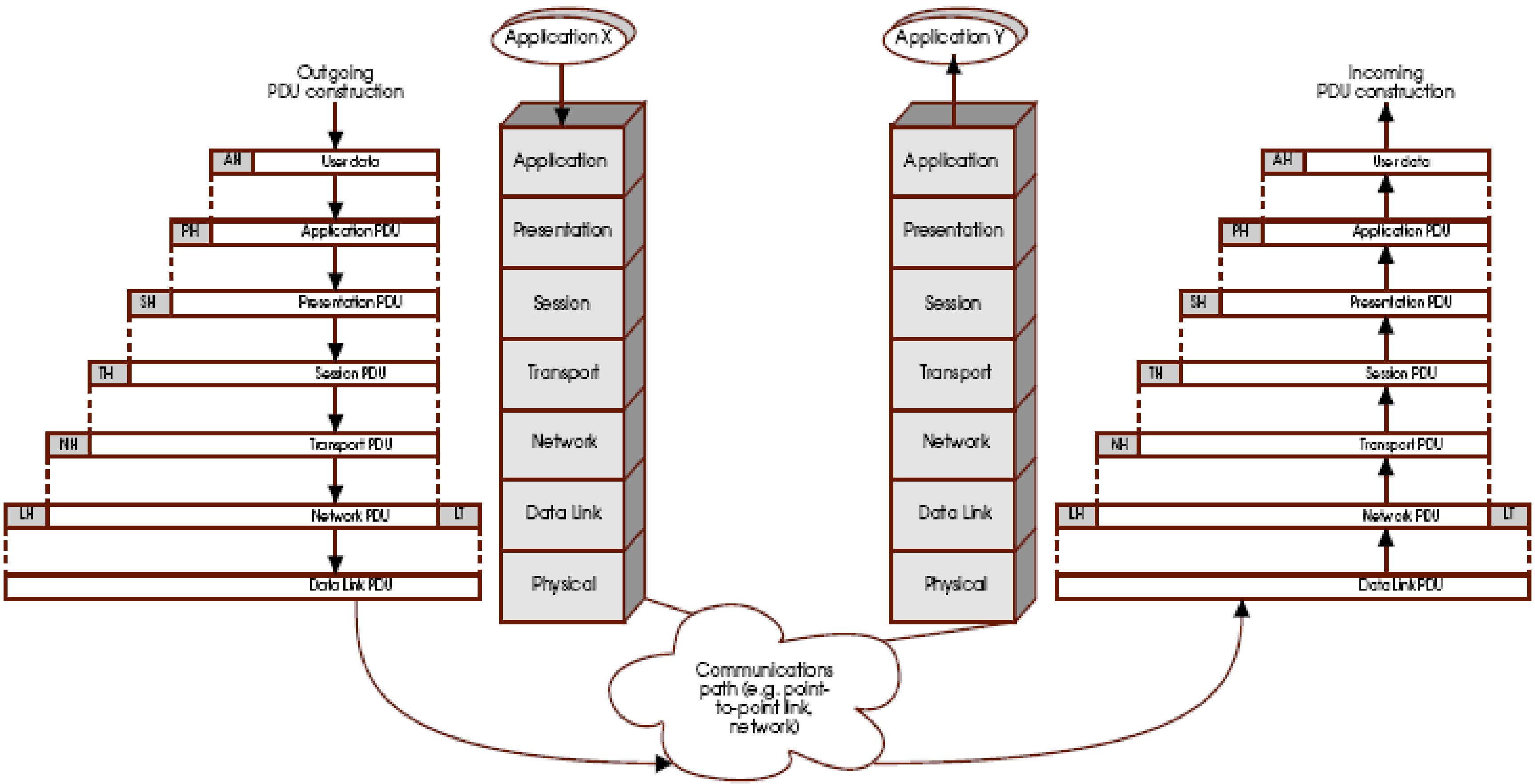


ISO OSI Architecture
Data flow over the network

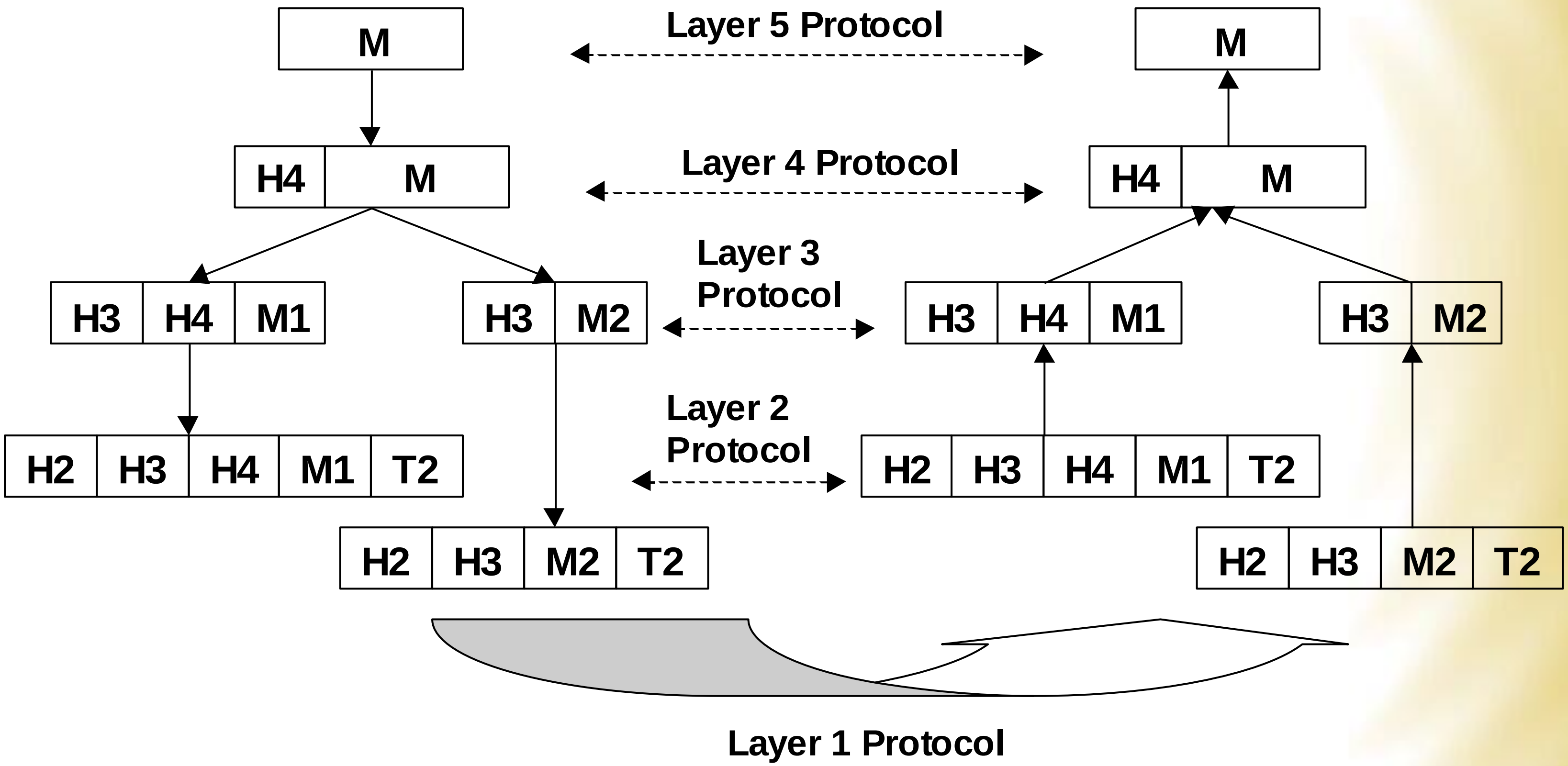


ISO OSI architecture

Encapsulation



Encapsulation



OSI Architecture
Basic functionalities

End System

- **Application**
- **Presentation**
- **Session**
- **Transport**
- **Network**
- **Data link**
- **Physical**

Application-specific exchange of messages

Reliable delivery, error recovery, e2e flow control, congestion control

Routing, segmentation and reassembly, network-wide addressing

Addressing (local), medium access, p2p flow control, error control

Voltage swing, bit duration, connector type, synchronization

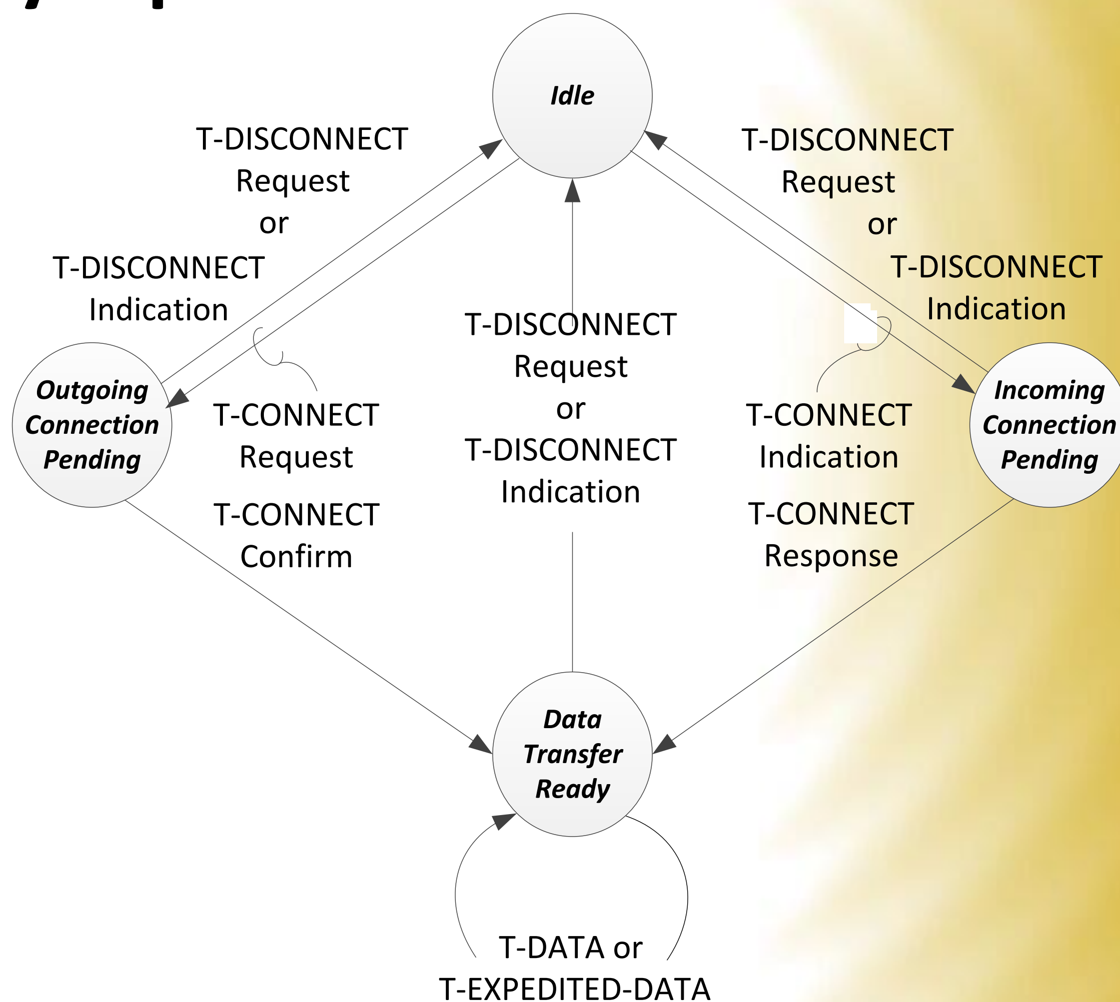
ISO OSI architecture
Protocols - ISO specification

Service	Primitive	Parameters
Connection Estabilishment	T-CONNECT request and indication	Called address Calling address Expedited data Quality of service User data
	T-CONNECT response and confirmation	Responding address Expedited data Quality of service User data
Data transfer	T-DATA request and indication	User data
Expedited data transfer (optional)	T-DATA request and indication	User data
Connection release	T-DISCONNECT request	User data
	T-DISCONNECT indication	Disconnect reason User data

ISO-OSI Architecture
Formal description of states in ISO T4 transport layer protocol

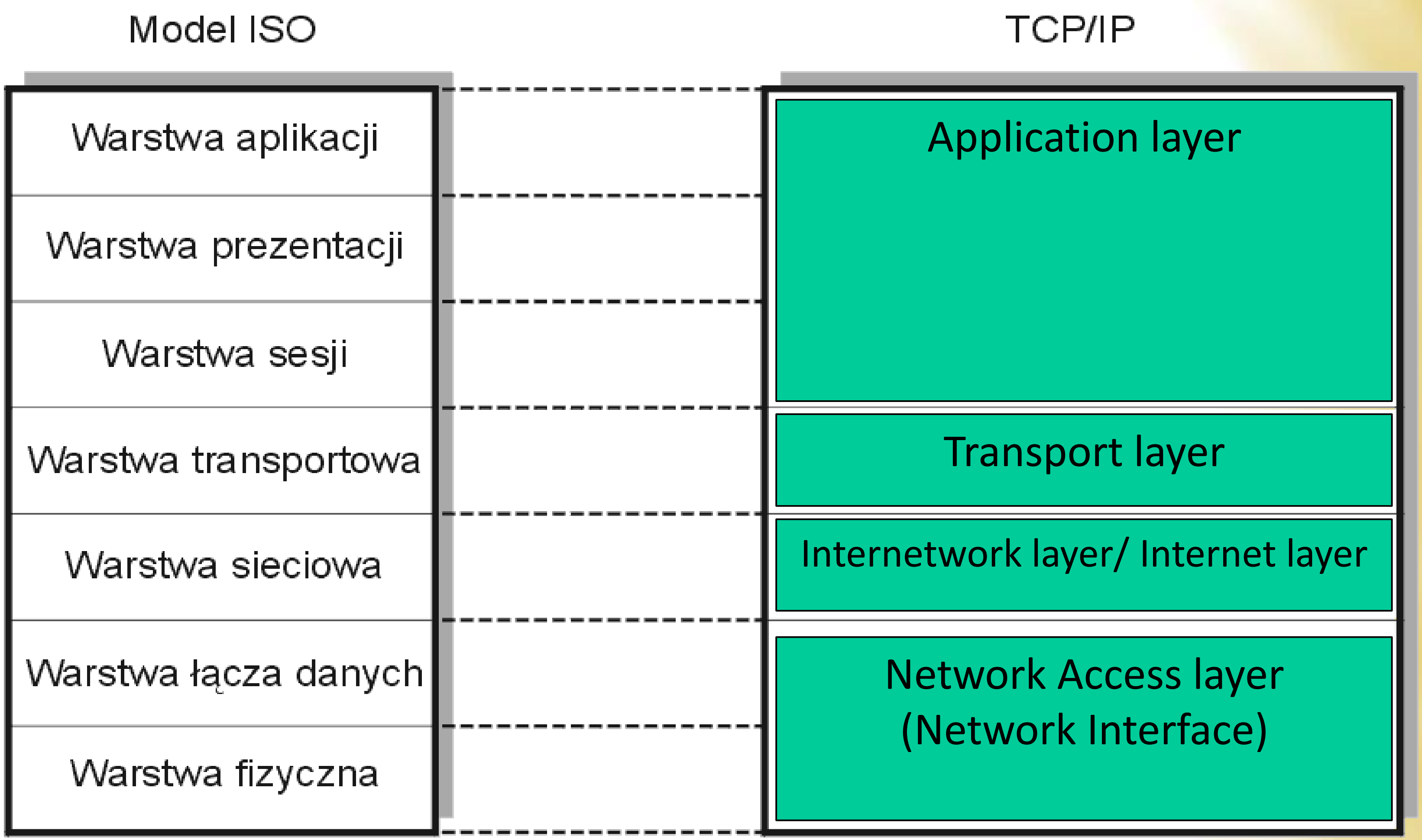
Description method :
Finite state machine
States
State transitions
(& Primitives)

Goal:
Formal description
Verification of the protocol correctness
Validation of functional features

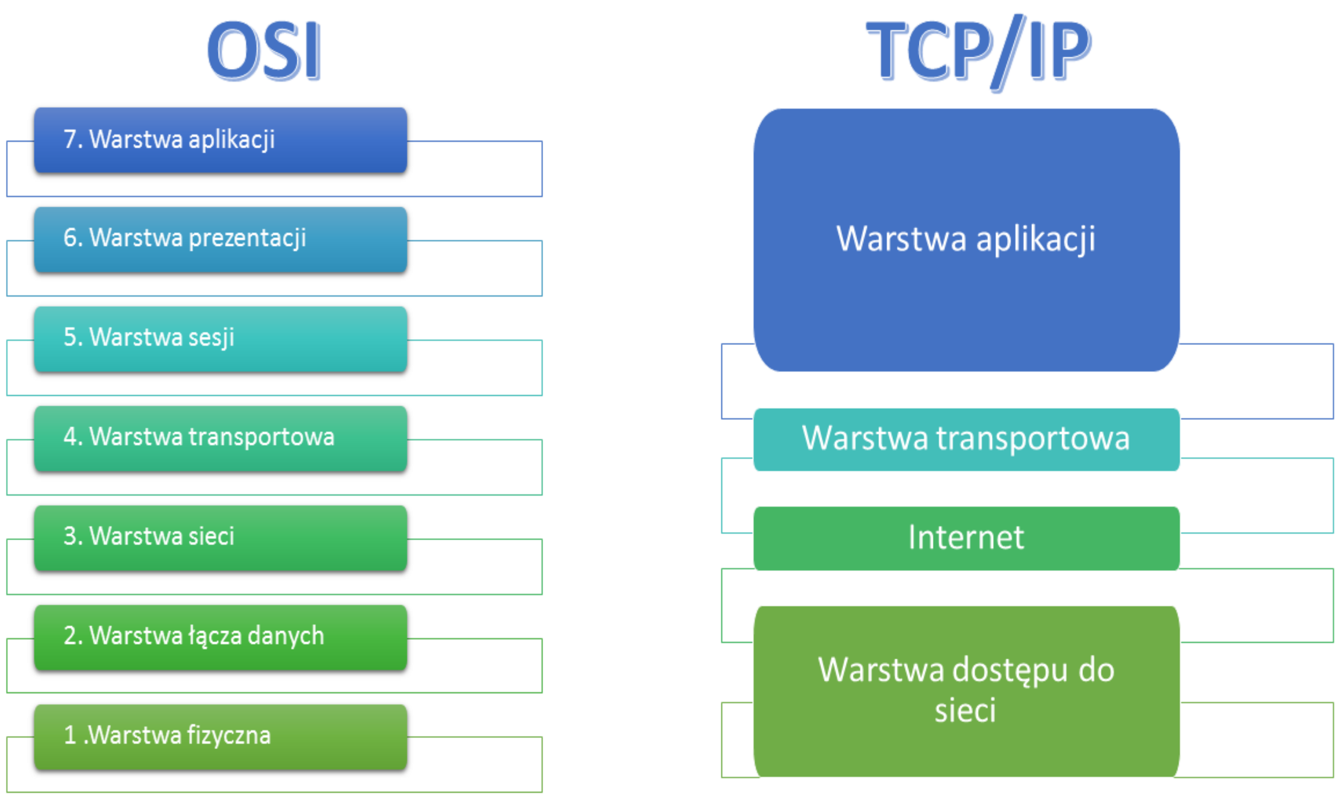


ISO/OSI and TCP/IP architectures

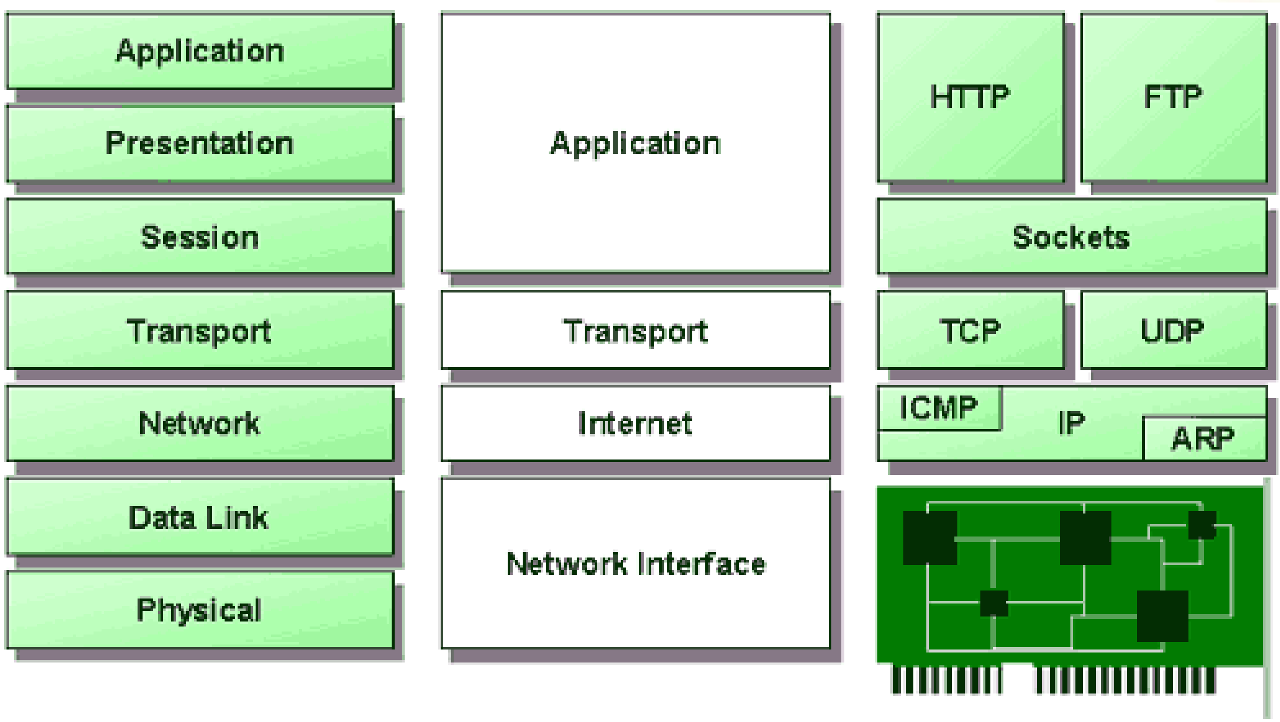
Simple comparison



ISO/OSI & TCP/IP Architectures



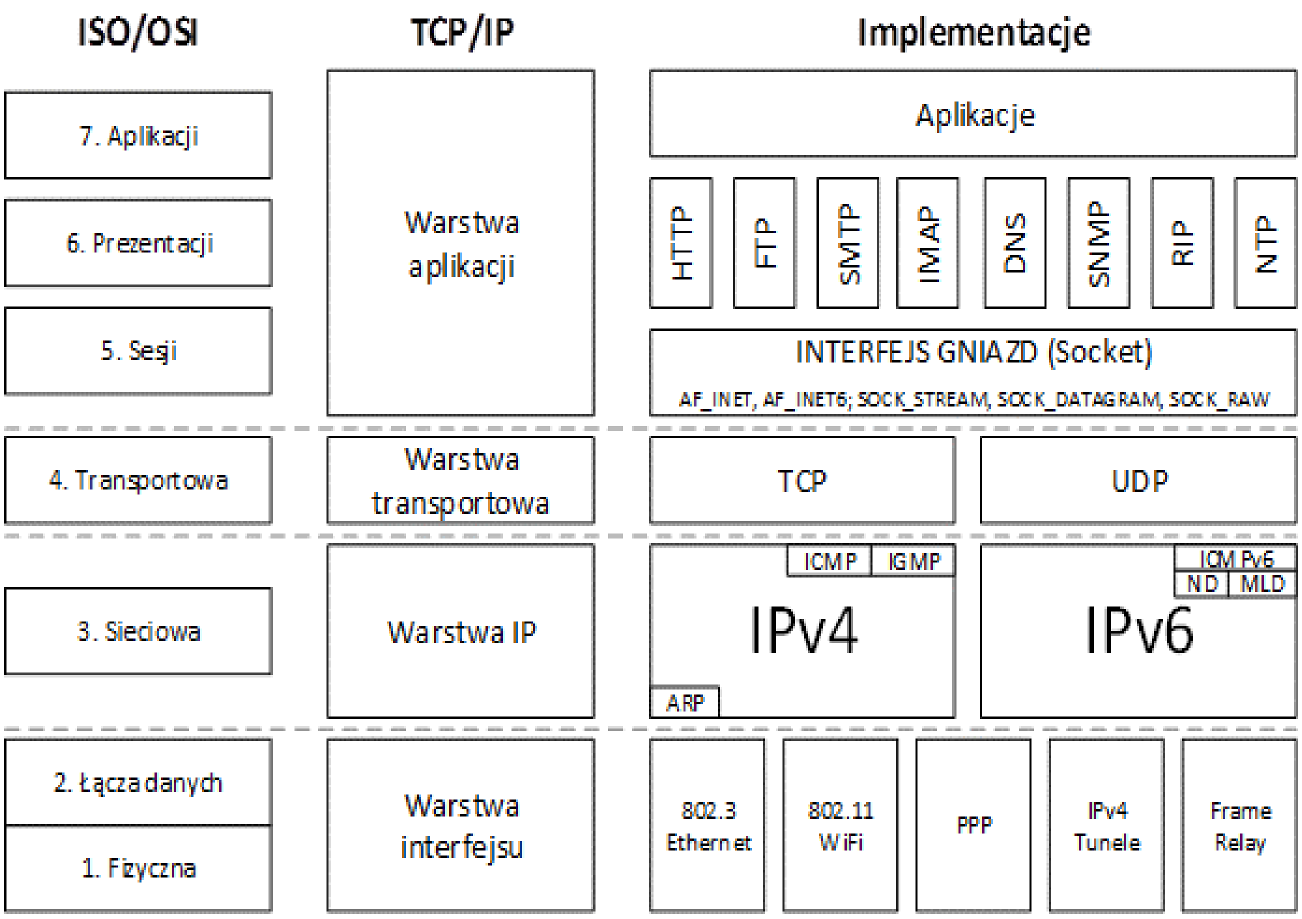
ISO/OSI & TCP/IP Architectures



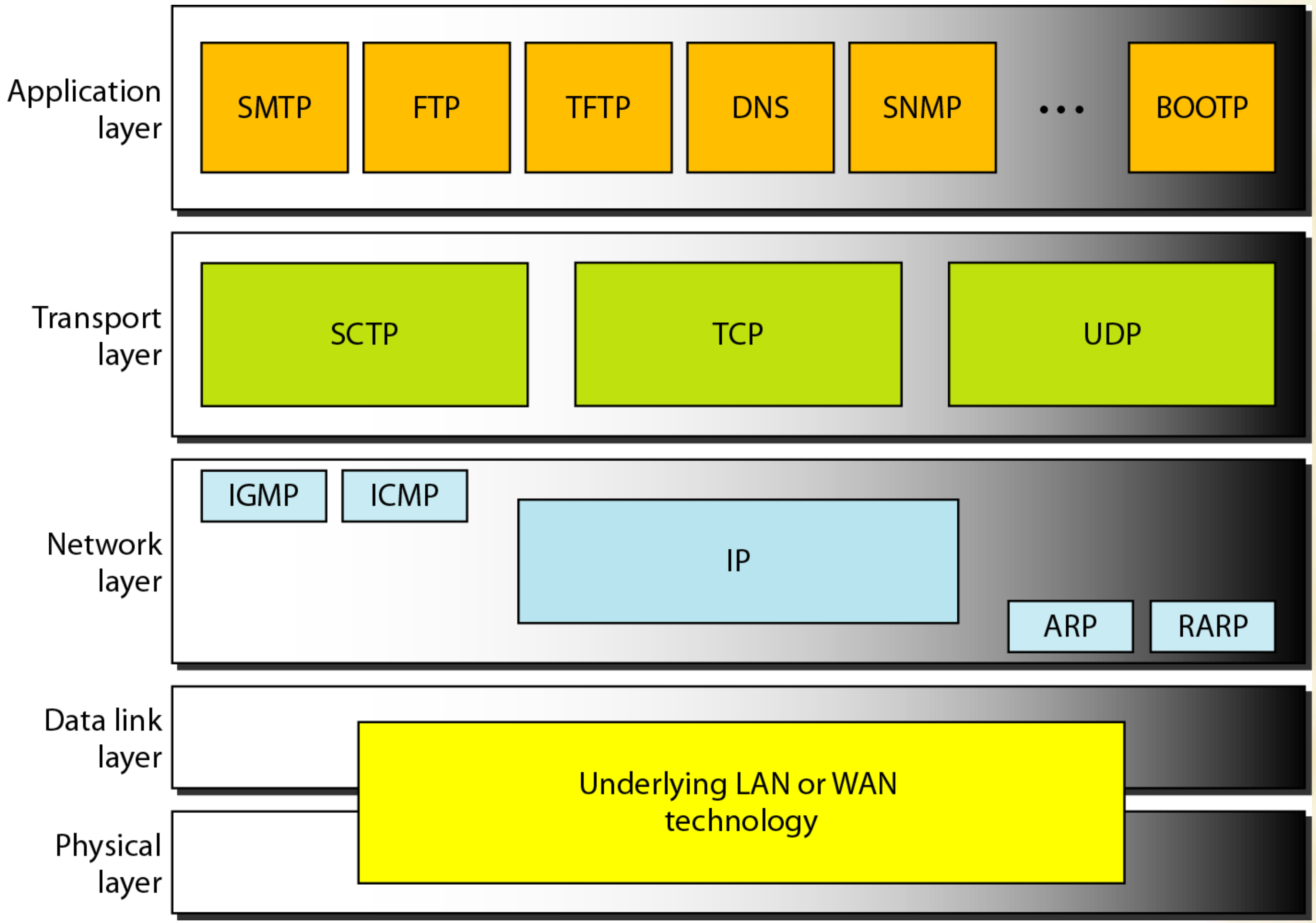
OSI and TCP/IP

Source: "Introducing TCP/IP," by FindTutorials.com

ISO/OSI & TCP/IP Architectures



TCP/IP Architecture



TCP/IP Architecture

The purpose of developing a set of TCP / IP network protocols was to enable communication between systems from many different vendors.

Changes in the TCP/IP architecture are closely related with the development of the Internet - controlled by, among others, Internet Engineering Task Force - IETF.
The principles of Internet operation are issued by IETF and described in RFC (Requests for Comments) documents.

Internet protocols are supported by most network operating systems.
At the beginning it was the UNIX environment and Ethernet type local area networks.

TCP/IP Architecture

TCP / IP was created earlier than the ISO-OSI reference model;
Due to this, the layers are not so clearly and unambiguously defined, and their presence (the existence of the layered architecture itself) results more from the analysis of functions performed by the protocols rather than as assumed by the network designers.

TCP/IP Architecture

Layer 1 and Layer 2 protocols depend on the technology of the Internet component networks:
- in the case of LAN networks, it can be: Ethernet, WiFi or Token-Ring
- for WANs: PPP, SLIP, X.25 and Frame Relay protocols.

A connectionless IP protocol is used in the interconnection layer.

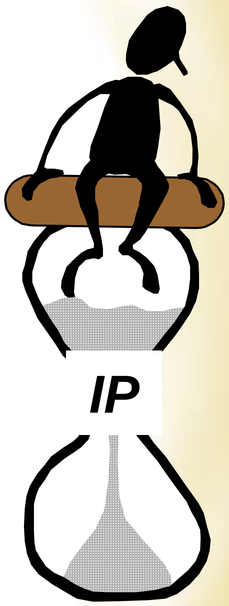
Routing protocols are responsible for selecting a datagram route through the network. Internet routing is hierarchical.

TCP is responsible for the integral data transfer between end systems (Hosts).

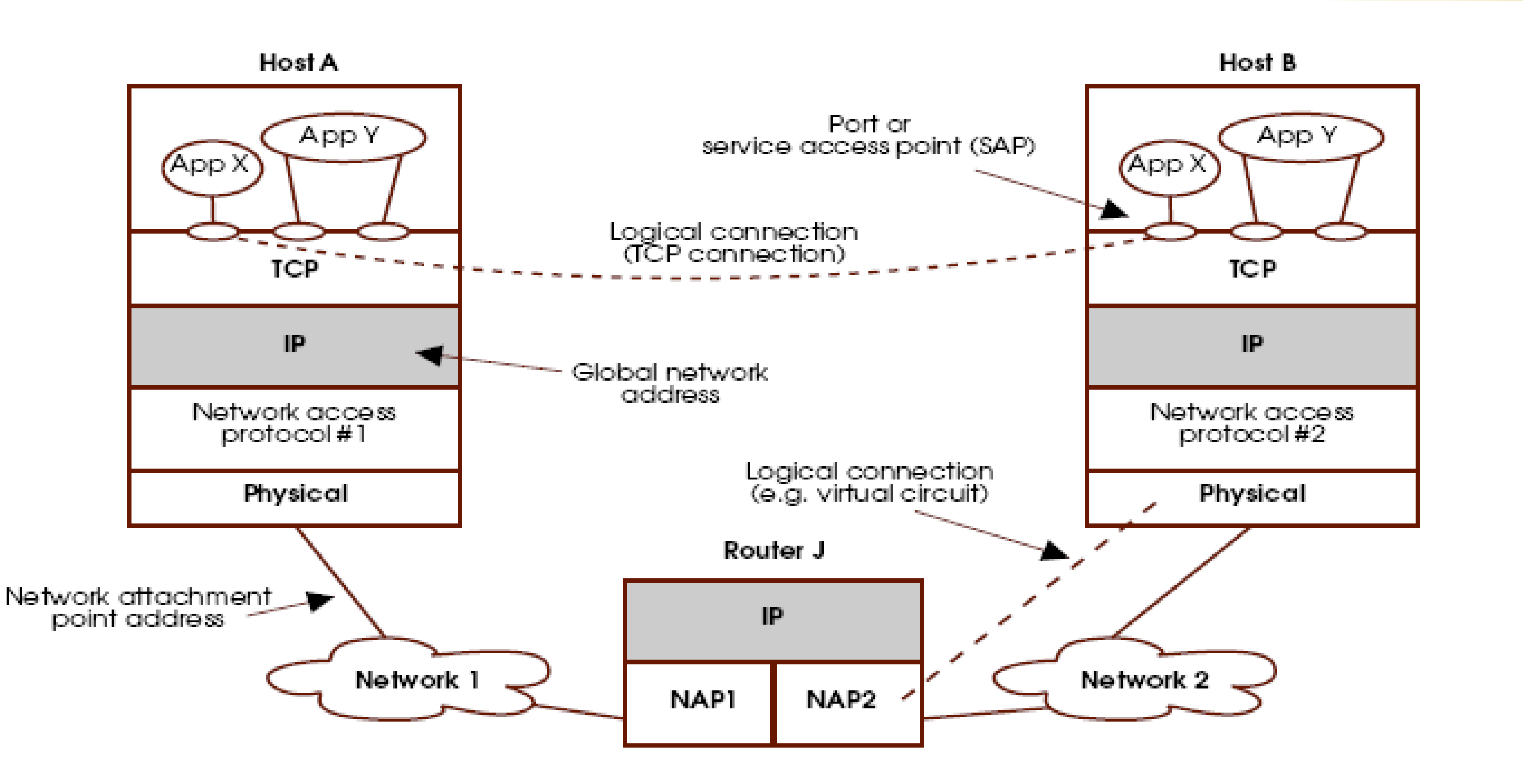
POLITECHNIKA
GDAŃSKA

TCP/IP architecture: Why is IP so popular?

- Hourglass-like shape of the TCP/IP protocols set:
- Many protocols work "over" IP;
- IP works "over everything" .
- **Architecture principles:**
  Minimalism, autonomy
  Best effort service
  Stateless routers
  Decentralized control

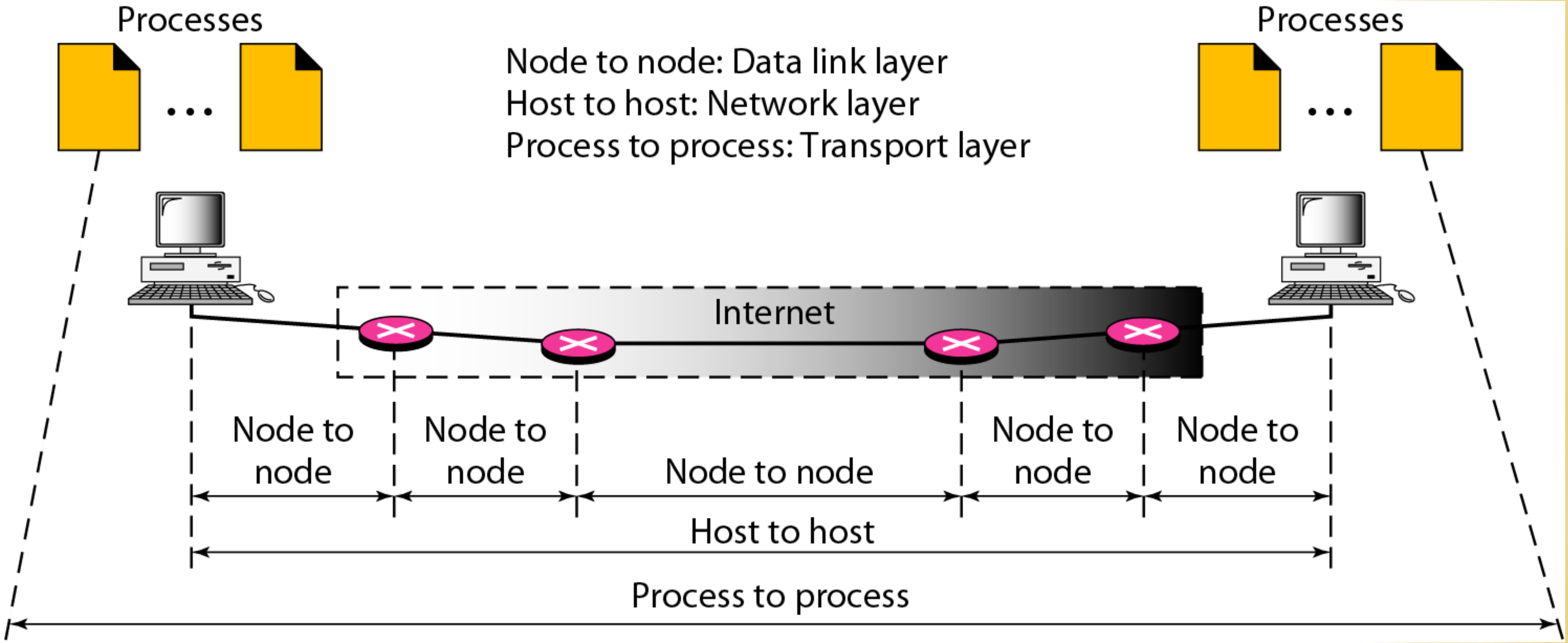


TCP/IP Architecture



- TCP stack not necessary for router
- Multiple network access protocols are possible at the same time

TCP/IP Architecture

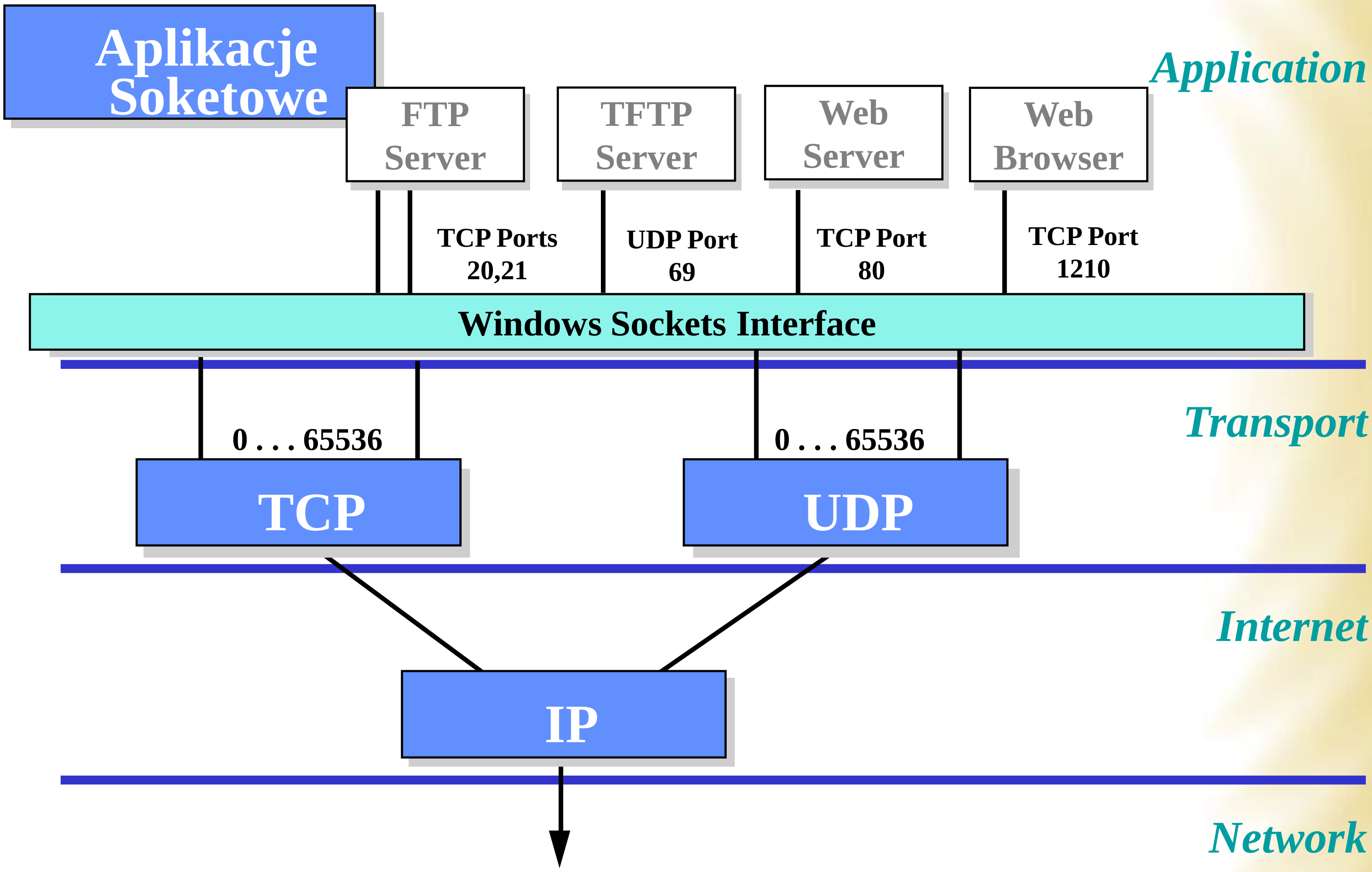


TCP/IP architecture

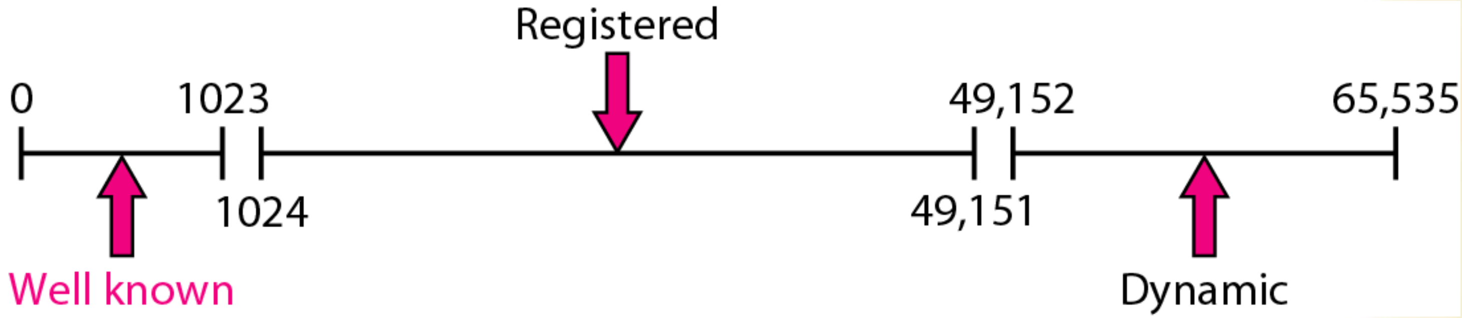
Hierarchy of TCP/IP addresses

- **Ports** are destinations on end computers - on host devices. **Ports** define the binding of an application to the transport layer. **Transport layer addresses** consist of a network IP address and a port number.
- In the Internet, each host is assigned a unique number – a part of the **IP address** is used for routing.
- In LAN Ethernet networks (as well as in other LAN/ WLAN networks) each host (a network interface card) is responsible for recognizing the message directed to it using also a unique **MAC address** (so-called ***physical address***).

TCP/IP architecture
Ports and sockets



TCP / IP architecture
Port number ranges in IANA



TCP / IP architecture
internet sockets

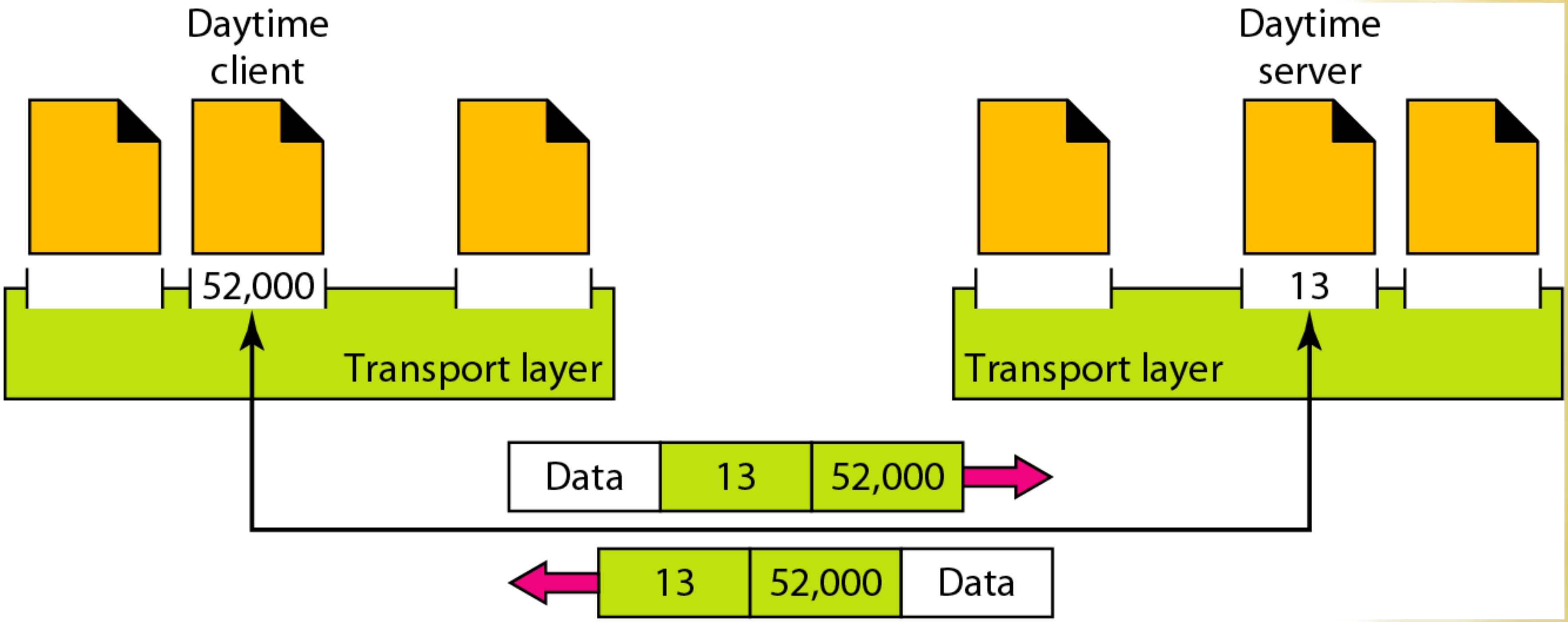
The combination of the host IP address, number, and type of port over which communication takes place is called **the Internet socket**.

The Internet socket is:
IP address, port number, port type.

Notation:
IP address 192.254.60.10, port number 23, TCP socket type >>> 192.254.60.10: 23 TCP

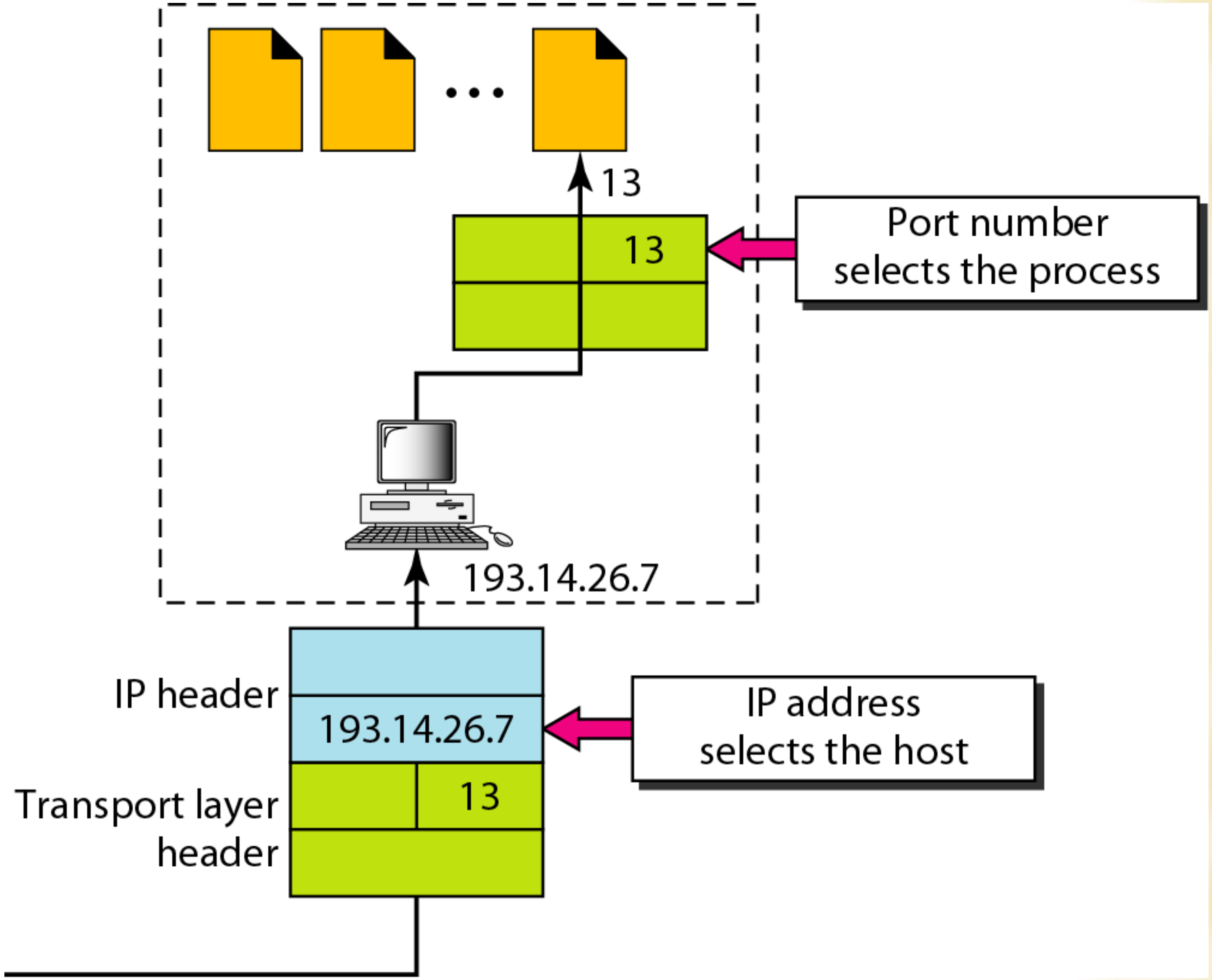
TCP/IP architecture

Port numbers

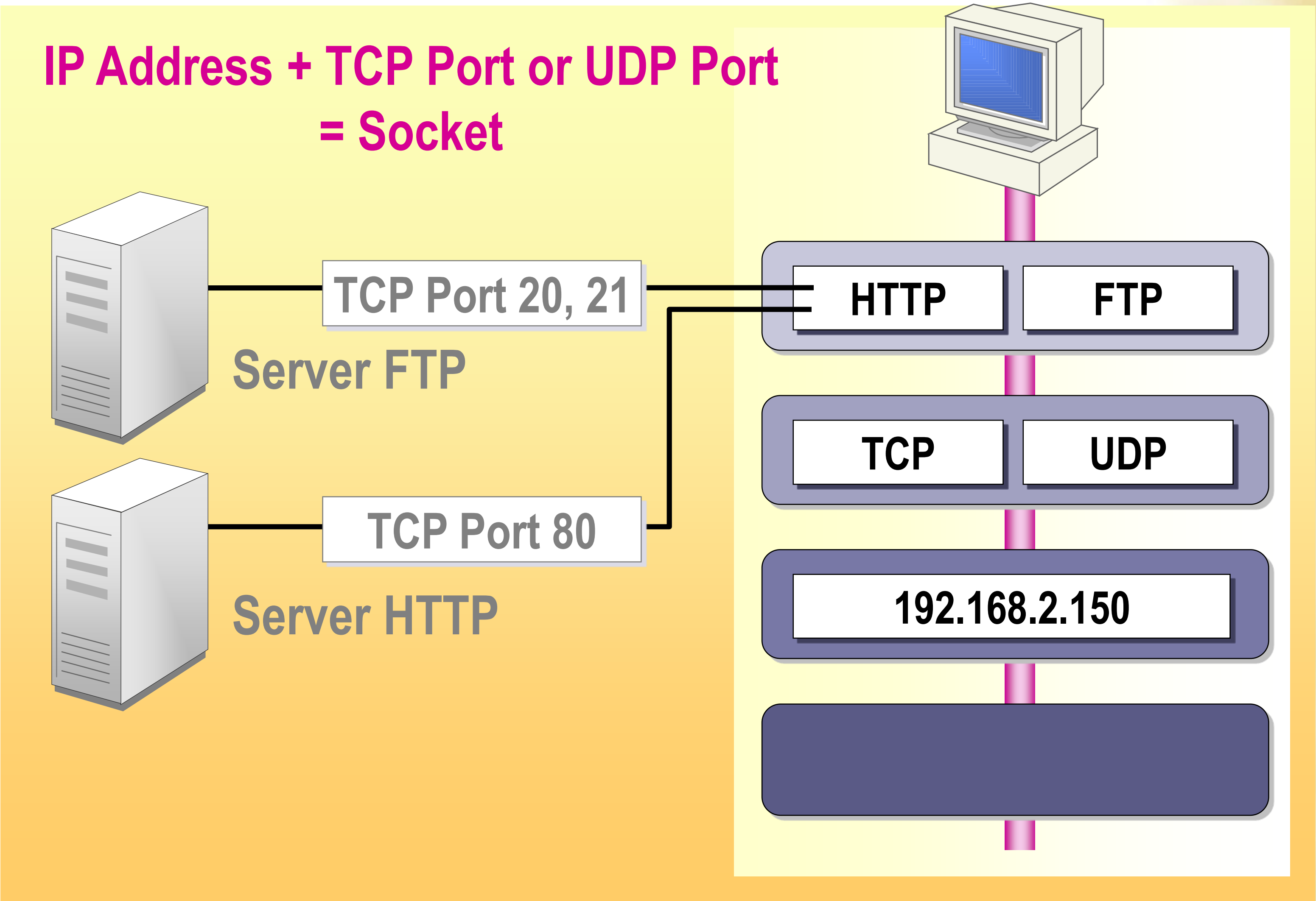


TCP/IP architecture

IP addresses vs port numbers



TCP / IP architecture
Application identification



TCP/IP architecture
Application layer

- How does the browser get data from the web server?
- We use one of the application protocols: HyperText Transfer Protocol (HTTP).
- Users activate applications that "chat" using the application protocol.
- The applications utilize a transport protocol to transmit and receive data.
- Application examples: FTP, SMTP, DNS, SMB, DHCP, ...

Hypertext Transfer Protocol (HTTP)

World Wide Web (World Wide Web) protocol. HTTP is defined by RFC 2616. It is the HTTP protocol that is used to send requests to share web documents.

The job of websites is to publish information - and the HTTP protocol does just that.

HTTP is so useful because it provides a standardized way for computers to communicate with each other. It specifies the form of the client's data requests and the form of the server's response to these requests.

It is included in the stateless protocols, due to the fact that it does not retain any information about previous transactions with the client, after the transaction is completed, everything "is lost" - this is why cookies have become so popular.

HTTP uses port number 80.

Application layer example

- HTTP - sketch:
  - GET /~hugue/index.html HTTP/1.1
  - Host: www.cs.umd.edu

GET /~hugue/index.html HTTP/1.1
Host: www.cs.umd.edu

HTTP Message

Transport layer

- Provides end-to-end communication between applications.
- Transport Protocol: Transmission Control Protocol (TCP) a reliable, connection-oriented transport protocol.
- Divides the application message stream into segments.
- It works with the Internet Layer (interconnect) to send and receive.
- In general, a transport protocol can be
  - unreliable or reliable,
  - connection or connectionless,

  and the data flow may be regulated or not….
- Others: UDP, STCP (Stream Control Transmission Protocol - message-oriented protocol)

Transport Layer Example

- TCP - sketch:
  - Source Port: 1081
  - Destination Port: 80
  - Checksum: 0xa858

Src: 1081 Dst: 80 Chksum: 0xa858	GET /~hugue/index.html HTTP/1.1 Host: www.cs.umd.edu
TCP header	HTTP Message

Internet layer

- Responsible for routing between hosts - communicating devices.
- Accepts requests to send packet (s) to the destination address.
- Internet Protocol (IP) wraps TCP segments into IP datagrams with an IP header and uses a routing algorithm to decide whether to send them directly or indirectly.
- It also handles incoming IP datagrams.
  If they are addressed to the local host then it removes the IP datagram header and passes the segment to TL.

Internet Layer Example

- IP - sketch:
  - Time to live: 128
  - Header checksum: 0x57d1
  - Source: my home pc (69.140.128.222)
  - Destination: www.cs.umd.edu (128.8.10.143)

IP datagram header	TCP header	HTTP Message
Src: 69.140.128.222 Dst: 128.8.10.143 TTL: 128	Src: 1081 Dst: 80 Chksum: 0xa858	GET /~hugue/index.html HTTP/1.1 Host: www.cs.umd.edu

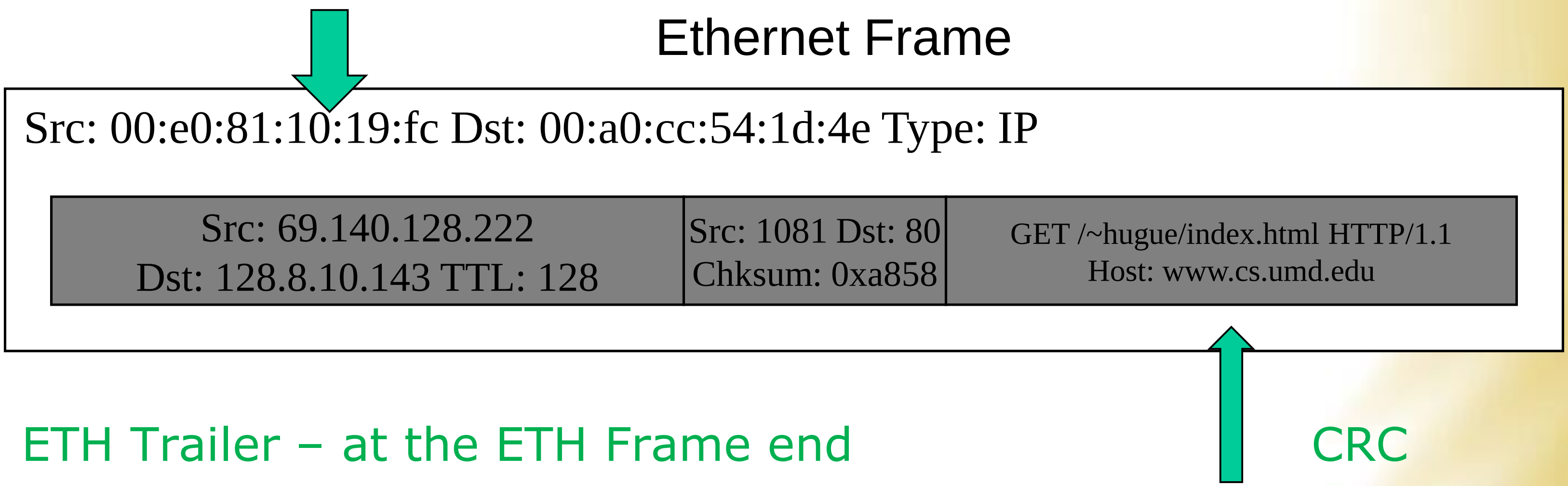
Network interface layer

- Receives IP datagrams and after appropriate conversion transmits them to the appropriate network.
- It can be a relatively simple network controller (e.g. a controller for Ethernet networks) or a more complex subsystem with successive DLL protocols.

Network Interface Layer Example

- Ethernet - sketch:
  - Destination: 00:a0:cc:54:1d:4e
  - Source: 00:e0:81:10:19:fc
  - Type: IP
  - ETH Header

Ethernet Frame



Domain Name System (DNS)

- It is a system of dedicated servers and a communication protocol that ensures the conversion of addresses known to Internet users (descriptive - mnemonic) into addresses understandable to devices - making up a computer network. Thanks to the use of DNS, a mnemonic name, e.g. pl.wikipedia.org, can be changed to the corresponding IP address, i.e. 145.97.39.135.

- DNS addresses consist of Internet domains separated by dots. For example, in Wikipedia> org means a functional domain of an organization,> wikipedia - a domain belonging to the Wikimedia foundation, and> pl - a Polish domain in the institution's network. This way, it is possible to build a hierarchy of names that order the Internet.

- DNS is a complex computer and legal system. On the one hand, it ensures the registration of Internet domain names and their association with IP numbers. On the other hand, it provides day-to-day service for computers finding IP addresses corresponding to individual (descriptive)names.

- The basics of the DNS protocol were described in 1982 in the document IETF - RFC 819

Standards/ Standardization Organizations

- Why standards - advantages:
  They stimulate the development of the market, strengthen competition;
  They reduce costs;
  They guarantee cooperation and mutual compatibility/ even of different architectures.
  Standards disadvantages:
  They tend to freeze technology;
  They introduce redundancy.

- Standard types:
  Informal (developed as de facto standards);
  Formal;
  Accepting the use of informal standards.

Major standardization organizations

- ISO (International Standards Organization)
  An international organization whose members are representatives of national organizations.
- ANSI is the American ISO member.

- ITU-T (International Telecommunications Union - formerly CCITT) - brings together administrators and network operators….

-

Major standardization organizations

- ETSI (European Telecommunications Standards Institute)

- IEEE (Institute of Electrical and Electronics Engineers)
  An international organization (with American roots) whose members are representatives of various companies and institutions.
  A major player in the market of LAN and MAN standards.

Major standardization organizations

- The Internet Society
  Responsible for the publication and development of Internet standards;
- Three basic organizations:
  Internet Architecture Board
  Internet Engineering Task Force
  Internet Engineering Steering Group
- Standardization process:
  Internet draft;
  Request for Comments (RFCs);
  Internet standard.

Major standardization organizations

IETF (Internet Engineering Task Force)

RFC (at least two meetings annually):

Acceptance policy (procedure):

The proposal must be technically competent and well justified;

It must have a number of independent implementations, showing at the same time the principles and possibilities of cooperation;

It should have fairly broad support and be recognized as useful.

Applicable RFCs include, but are not limited to:

IP (RFC791)

TCP (RFC793)

HTML (RFC1866)

The data is available at: http://www.ietf.org/rfc.html