1. Introduction – Recapitulation of assumed knowledge

PA191: Advanced Computer Networking

Eva Hladká

Slides by: Tomáš Rebok

Faculty of Informatics Masaryk University

Autumn 2024

Course Organization

- attending the lectures is optional
- the knowledge acquired during PB156 course or in book J. Kurose Computer networking is assumed
- course materials will be published on the course webpage (more or less in time ;-))
- assessment methodology:
  - final exam (written form)
  - no priority questions on the exams
- course literature:
  - slides, RFCs, ...
  - literature being announced in relevant course parts

Course Overview

- the course goal:
  - to provide an advanced insight into the area of computer networks and their applications
- discussed topics:
  - advanced IPv6 functionalities
  - advanced routing mechanisms
  - QoS in computer networks
  - ad-hoc/sensor networks
  - peer-to-peer networks/systems
  - mobile services
  - etc.

Recapitulation of assumed knowledge

Recapitulation of assumed knowledge

Lecture overview

1. Course Introduction
2. Lecture overview
3. Introduction
   - Computer Networks in General
   - Network Protocols
   - Standardization
4. Network Models
   - ISO/OSI Model
   - ISO/OSI vs. TCP/IP Model
5. TCP/IP Model
   - L1 – Physical Layer
   - L2 – Data Link Layer
   - L3 – Network Layer
   - L4 – Transport Layer
   - L7 – Application Layer

Lecture overview

Computer Networks

Introduction

- a group of computers and devices interconnected by communications channels that facilitate communications among users and allows users to share resources
- can be used for various purposes:
  - facilitating communications (in various ways – text, speech, video, etc.)
  - sharing hardware
  - sharing files, data, and information
  - sharing software
- fundamental characteristics:
  - *Delivery* – the system must deliver data to the correct destination
  - *Accuracy* – the system must deliver data accurately
  - *Timeliness* – the system must deliver data in a timely manner

Computer Networks

Ideal vs. Real Networks

Ideal Networks

- transparent for users/applications
  - just end-to-end characteristics
- unlimited throughput
- no losses
- no delay/latency and jitter
- keeps packet ordering
- data cannot be corrupted

Real Networks

- have internal structure which influences data transmission
- limited throughput
- (sometimes) data losses
- (sometimes) variable delay/latency and jitter
- (sometimes) do not keep packet ordering
- data can be corrupted

Computer Networks

Required features

- *efficiency* – efficient/maximal use of available throughput
- *fairness* – the same approach to all the data flows (having the same priority)
- *decentralised management*
- *fast convergence when adapting to a new state*
- *multiplexing/demultiplexing*
- *reliability*
- *data flow control* – a protection in order to avoid network's (network devices') and hosts' congestion

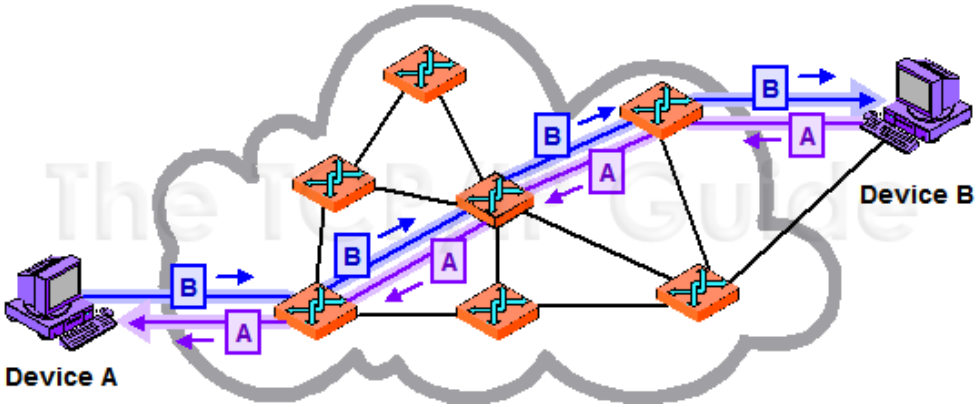
Computer Networks

Basic Approaches I.

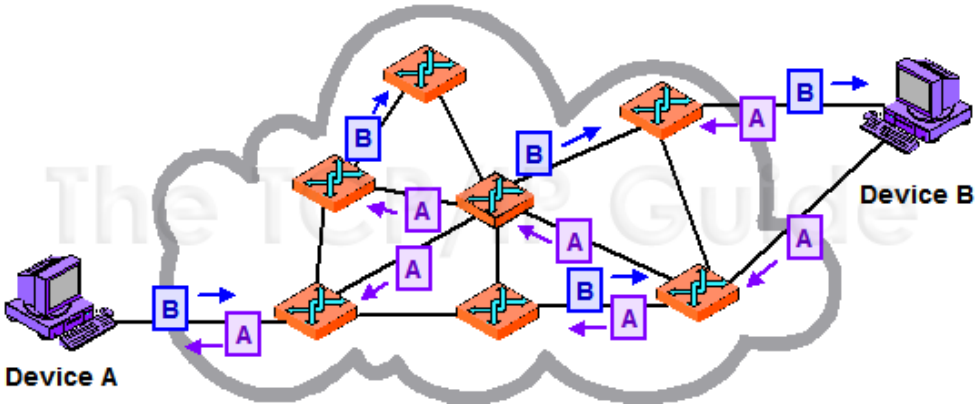
- **connection-oriented networks** (circuit switching networks)
  - a connection (called a *circuit*) is set up between two devices, which is used for the whole communication
  - information about the nature of the circuit is maintained by the network – a *state* has to be maintained
  - circuit may either be a fixed one or created on an "as-needed" basis
  - easy (more or less automatic) implementation of the QoS
  - e.g., the regular telephone system
- **connection-less (state-less) networks** (packet switching networks)
  - no specific path is used for data transfer – the data is chopped up into small pieces (called *packets*) and sent over the network
    - packets can be routed, combined or fragmented
  - on the receiving end the data is read from the packets and re-assembled into the form of the original data
  - no state has to be maintained
  - very hard implementation of the QoS (*best-effort service*)
  - e.g., the Internet

Computer Networks

Basic Approaches II.



connection-oriented network

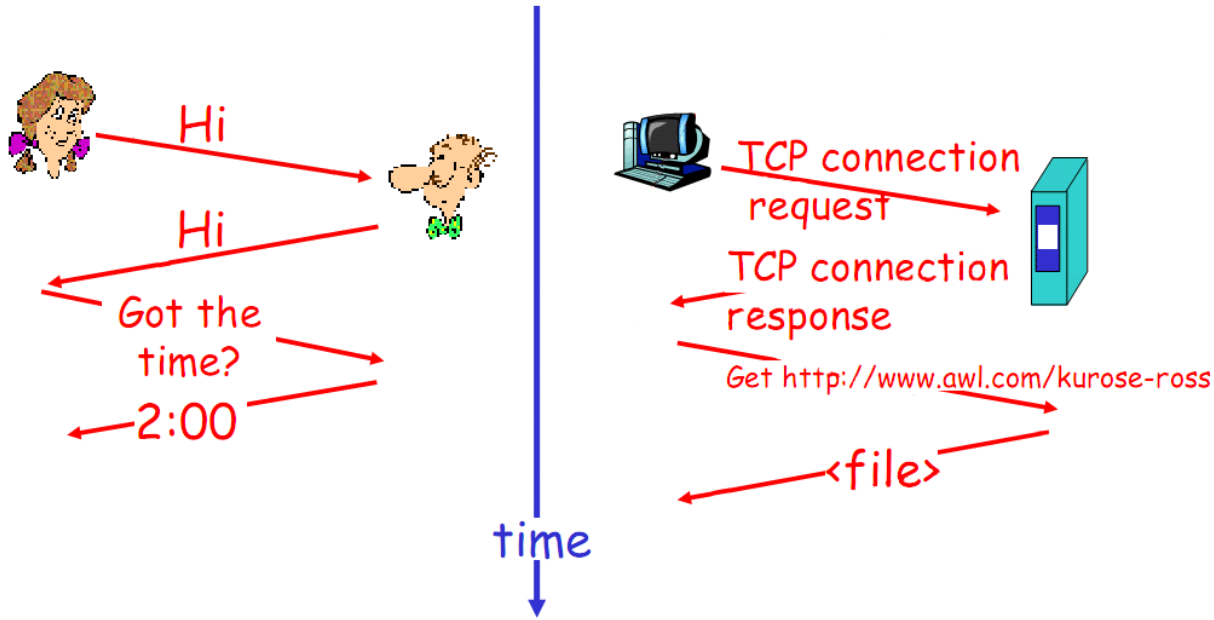


connection-less network

Network (Communication) Protocols I.

- motivated by the need to communicate among several entities (at least two)
  - *entity* = anything capable of sending or receiving information
- the form/method of the communication must be known to all the participating entities
  - they have to **agree on a protocol**

- human analogy:



Network (Communication) Protocols II.

- the **protocol** defines *"What"* the subject of communication is, *"How"* the communication has to behave and *"When"* does it behave
- they define:
  - *syntax* = structure/format of data (the order in which they are presented)
  - *semantics* = refers to the meaning of each section of bits (how should a particular pattern to be interpreted)
  - *timing* = when data should be sent and how fast they can be sent

- examples of network protocols:
  - UDP, TCP, IP, IPv6, SSL, TLS, SNMP, HTTP, FTP, SSH, Aloha, CSMA/CD, ...

Network Protocol

Network Protocol is a set of rules that defines the format and the order of messages exchanged among two or more communicating entities, as well as the actions performed during sending/receiving that messages.

Standardization

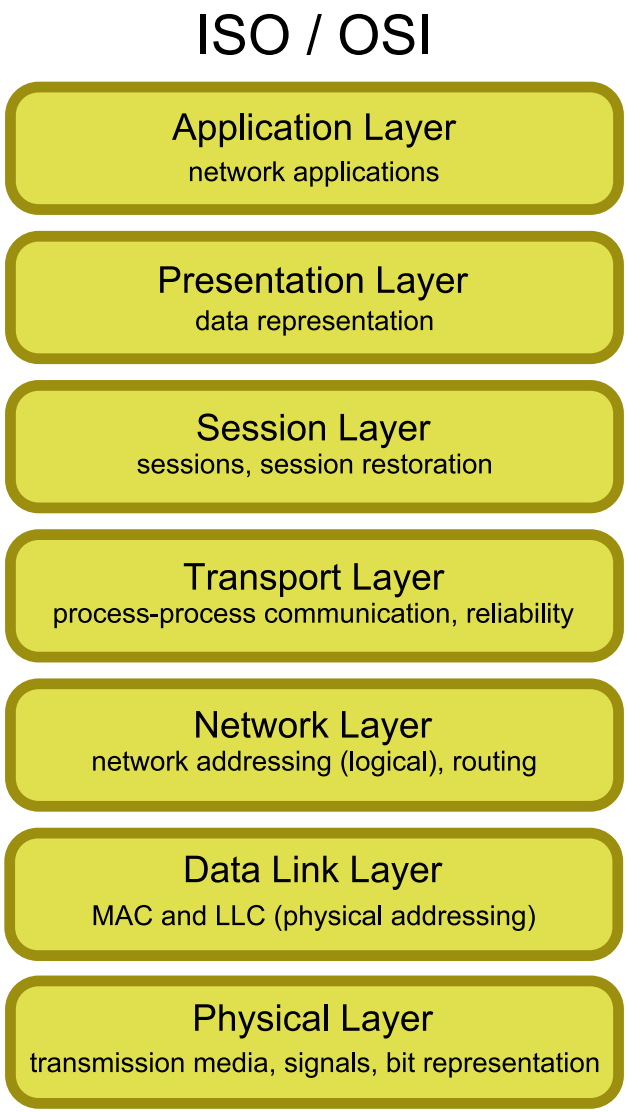
- definition of norms/standards describing various actions, activities, forms/methods of communication, etc. (not only in IT)
- main goals:
  - quality
  - security
  - compatibility
  - interoperability
  - portability
- standards fall into two categories:
  - *de facto* – standards that have not been approved by an organized body but have been adopted as standards through widespread use (they are often established originally by manufacturers)
  - *de jure* – standards legislated by an officially recognized body
- standard IT organizations:
  - ISO, ITU-T, ANSI, IEEE, IETF (*RFCs*), IEC, etc.

Lecture overview

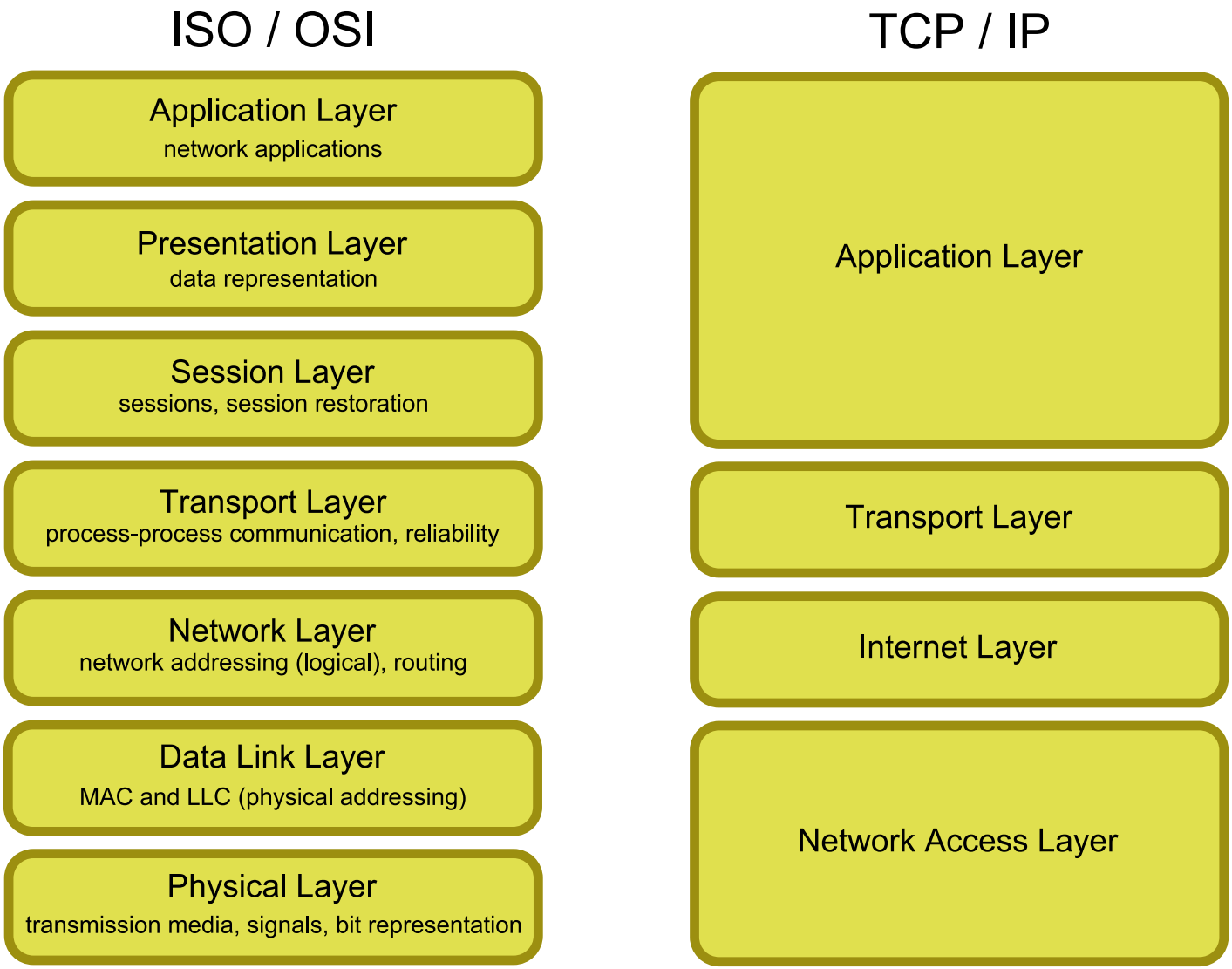
ISO/OSI Model I.

- **7-layer model** proposed by OSI organization in order to ensure compatibility and interoperability of communication systems developed by various vendors
- the purpose of layered architecture:
  - each layer is **responsible for particular functionality**
    - it adds some control information to the data in order to do its job
  - each layer **communicates just with its neighbours**
    - each layer uses the services provided by the lower layer and provides its services to the higher layer
    - the functionality is **isolated** in the particular layer (once a layer changes, just the neighbouring layers have to adapt to such a change)
  - logically, the communication is performed just between peer layers; physically, the communication traverses all the lower layers
  - the layers are just an abstraction – the real implementations are more or less different
- 7 layers not widely accepted $\Rightarrow$ TCP/IP model

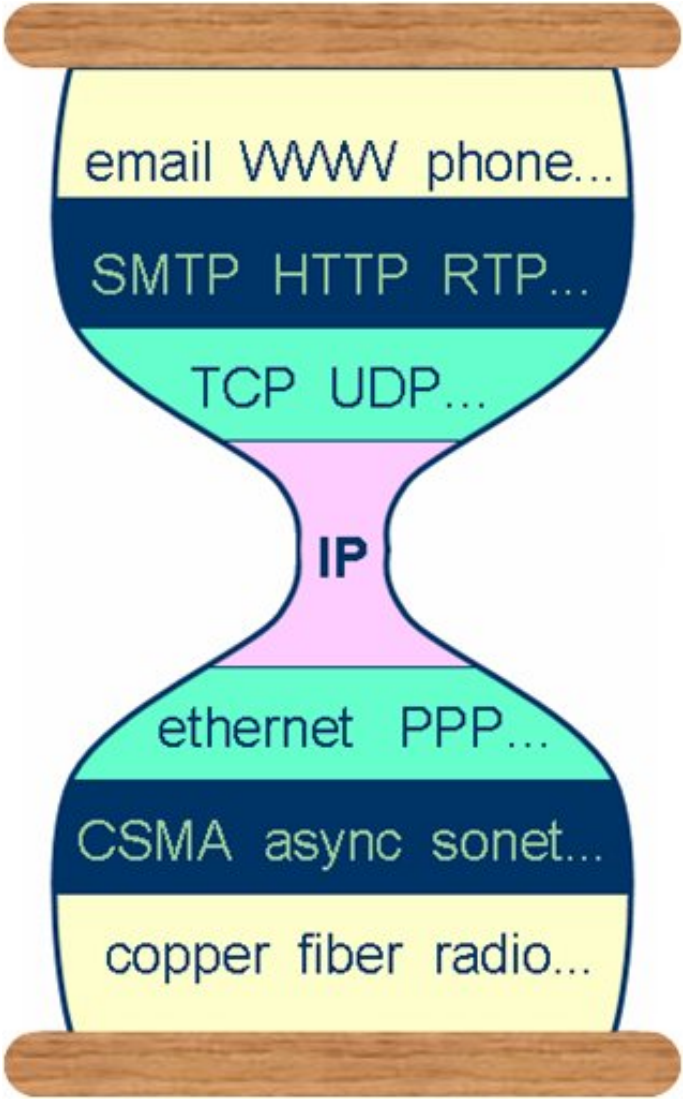
ISO/OSI Model II.



ISO/OSI Model vs. TCP/IP Model



TCP/IP Hourglass Model



Lecture overview

L1 – Physical Layer

Introduction I.

- **Physical Layer:**
  - provides the functionality for an interaction with transmission media
  - provides services for the *Data Link Layer*
    - the Data Link Layer passes/obtains data to/from the Physical Layer in the form of 0s and 1s organized into *frames*
    - the Physical Layer transforms the streams of bits (from frames) into *signals* spread through the transmission media
  - controls the transmission media; for example, decides about:
    - sending/receiving the data (signals)
    - data transformation (coding) into signals
    - the number of logical channels simultaneously transferring data from various sources

L1 – Physical Layer

Introduction II.

- **the main goal:** to ensure a transmission of bits (= the content of passed frames) between sender and receiver
- several standards (RS-232-C, CCITT V.24, CCITT X.21, *IEEE 802.x*) defining electrical, mechanical, functional, and procedural characteristics of interfaces used for connecting various transmission media and devices, e.g.:
  - parameters of the transmitted signals, their meaning and timing
  - mutual relationships of control and state signals
  - connectors' wiring
  - and many many others

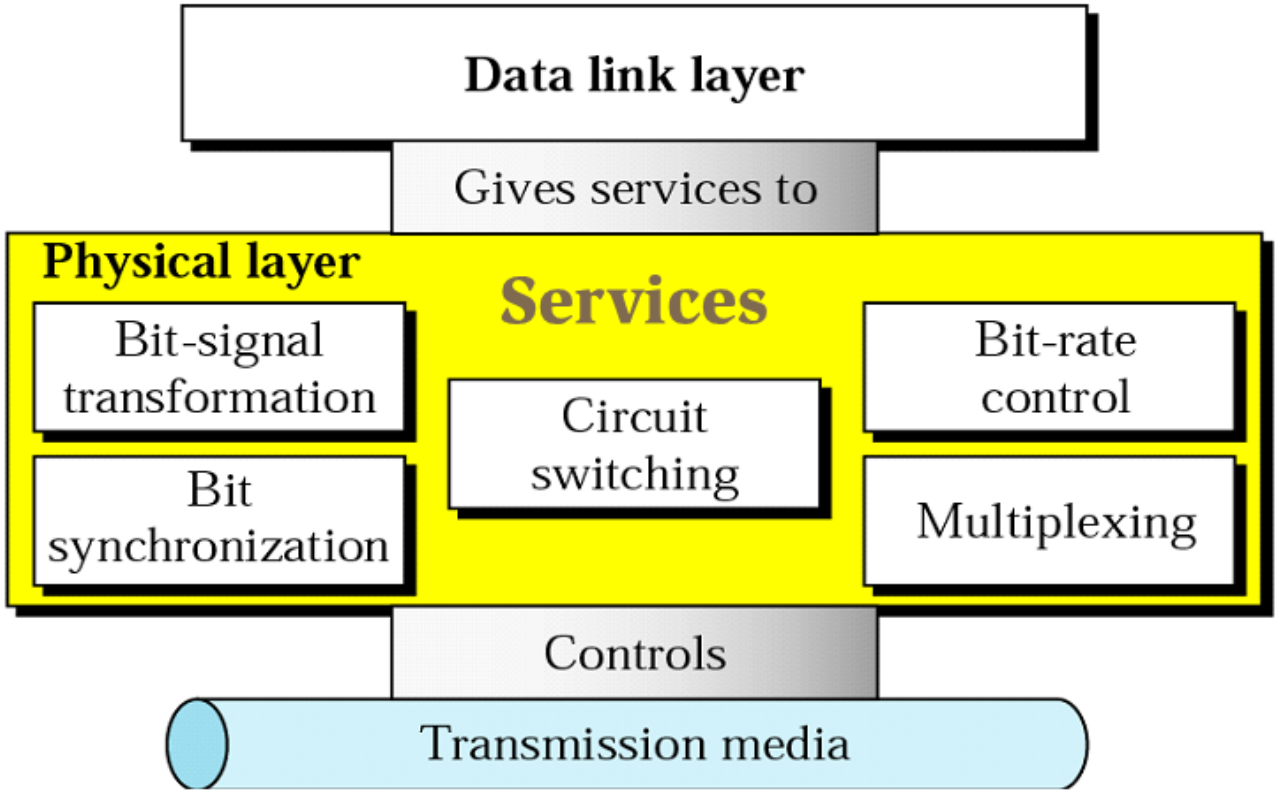


Figure: Position of the Physical Layer.

L1 – Physical Layer

Services

- *Bit-to-Signal Transformation*
  - representing the bits by a signal – electromagnetic energy that can propagate through medium
- *Bit-Rate Control*
  - the number of bits sent per second
- *Bit Synchronization*
  - the timing of the bit transfer (synchronization of the bits by providing clocking mechanisms that control both sender and receiver)
- *Multiplexing*
  - the process of dividing a link (physical medium) into logical channels for better efficiency
- *Circuit Switching*
  - circuit switching is usually a function of the physical layer
  - (packet switching is an issue of the data link layer)

L1 – Physical Layer

Signals

- data is transferred (via transmission media) in the form of (electromagnetic) *signals*
  - the data have to be converted into the signals
- *signal* = a function of time representing changes of physical (electromagnetic) characteristics of the transmission media
- data that have to be transferred (0s and 1s) – *digital* (binary)
- signals spread through the transmission media – *analog* or *digital*
  - some media suitable for both analog and digital transmission – wired media (coaxial cable, twisted pair), optical fibre
  - some media suitable just for analog transmission – ether (air)

L1 – Physical Layer

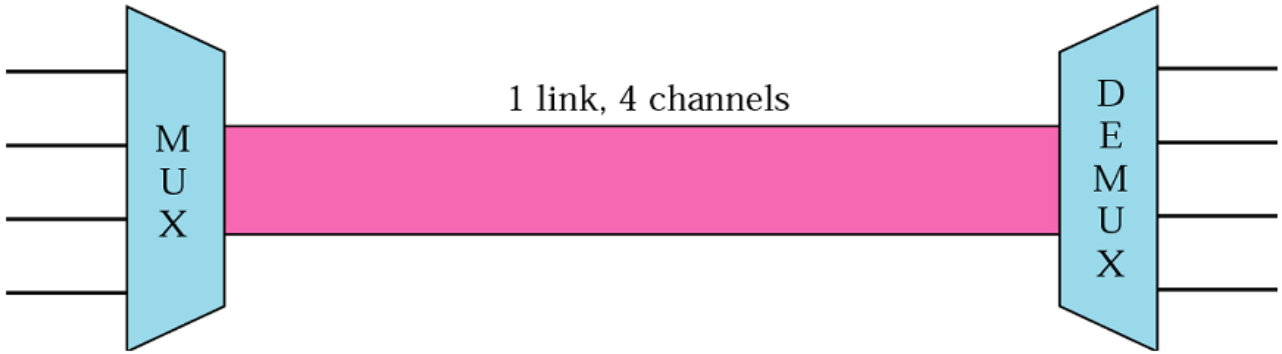
Transmission Media

- provide an environment for the functionality of physical layer
- basic distinction:
  - *guided (wired) media*
    - provide a conduit from one device to another
    - twisted pair (LANs, up to 10 Gbps), coaxial cable, optical fibre (backbones, hundreds of Gbps), etc.
  - *unguided (wire-less) media*
    - transfer an electromagnetic wave without the use of physical conductor
    - the signals are broadcasted (spread) via ether (air, vacuum, water, etc.)
    - radio signals, microwave signals, infrared signals, etc.
- for details see *PV183: Computer Networks Technology*

L1 – Physical Layer

Multiplexing

- *multiplexing* – a technique of sharing an available bandwidth by concurrent communication channels
  - the goal is to maximize the utilization of the media
  - applied especially for optical fibres and non-wired media



- for analog signals:
  - *Frequency-Division Multiplexing (FDM)*
  - *Wave-Division Multiplexing (WDM)*
- for digital signals:
  - *Time-Division Multiplexing (TDM)*

L1 – Physical Layer

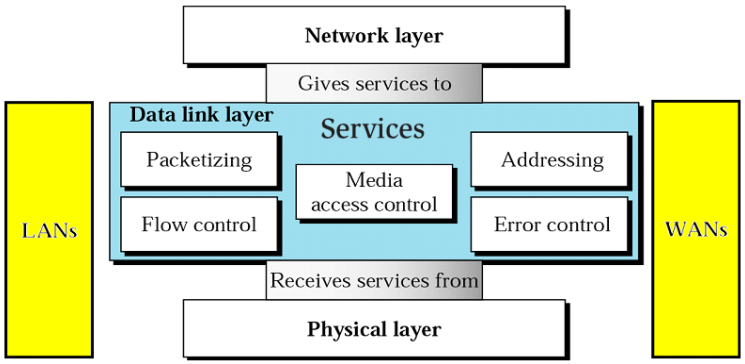
Résumé

- ensures the transmission of particular bits (0s and 1s) between the sender and receiver
- transferred bits are transcoded into the form of signals spread through the transmission media
  - the use of analog signals requires a *modulation*
  - the use of digital signals requires a *transcoding*
    - especially because of synchronization problems
- for the transmission, both wired (twisted pair, optical fibre, etc.) or non-wired (ether) media can be used
  - each of them is suitable for different conditions
  - the technique of sharing a single media by concurrent transmissions is called multiplexing
- *further information:*
  - PB156: Computer Networks (doc. Hladká)
  - PV169: Communication Systems Basics (doc. Staudek)
  - PV183: Computer Networks Technology (dr. Pelikán)

L2 – Data Link Layer

Introduction

- **Data Link Layer:**
  - receives *packets* (being passed from the Network Layer) and transforms them into *frames*
  - in cooperation with the Physical layer ensures the transmission of frames between communicating devices interconnected with a *(shared) transmission media*
    - i.e., just the local (inside a segment) delivery (LAN)
  - ensures the transmission reliability between these devices
  - ensures the flow control in order to avoid receiver congestion
  - controls the access of the devices to shared media (Medium Access Control)



L2 – Data Link Layer

Services

- *Framing*
  - the incoming packets (being passed from the Network Layer) are encapsulated into *frames*
- *Addressing*
  - provides the addresses of physical layer entities – *physical/MAC addresses*
  - frames contain source and destination addresses of communicating entities
- *Error Control*
  - it's not possible to eliminate the errors occurring on the physical layer
  - L2 layer ensures the required level of reliability of the data link (error detection and correction)
- *Flow Control*
  - prevents the receiver congestion
  - *stop-and-wait* mechanism, *sliding-window* mechanism, ...
- *Medium Access Control – MAC*
  - necessary in environments, where the transmission media is shared by several entities
  - eliminates collisions caused by multiple (concurrent) transmissions

L2 – Data Link Layer

Error Control

- a concept of redundancy is used
  - sender adds bits whose value is a function of transmitted data
  - receiver calculates the same function and if the values differ, it detects (tries to repair) an error
  - when using error detection only (or if the error is unrepairable), the receiver requests the sender to repeat the transmission
- *Error Detection, Automatic Request for Retransmission (ARQ)*
  - error detection and transmission repetition ensurance
  - suitable for little-lossy transmission media
  - even/odd parity, *Cyclic Redundancy Check (CRC)*, etc.
- *Forward Error Correction (FEC)*
  - error detection and attempts to data correction (using redundant data)
  - suitable for lossy transmission media (especially with high transmission latency)
  - e.g., *Hamming code*
  - for details see *PV169: Communication Systems Basics*

L2 – Data Link Layer

Medium Access Control (MAC)

- the functionality responsible for coordination of multiple devices' access to shared transmission media
- *The goal:* the elimination of collisions caused by concurrent transmissions (emissions)
  - i.e., concurrent transmissions to a shared transmission environment
- medium access protocols:
  - *random-access protocols* – Aloha, CSMA/CD, CSMA/CA
  - *controlled-access protocols* – based on reservations, polling, tokens, etc.
  - *channelization protocols (multiplex-oriented access)* – FDMA, TDMA, etc.

L2 – Data Link Layer

L2 Networks I.

- local area networks (LANs)
  - a systematic topology for simple networks
    - bus, circle, star, tree, mesh, etc.
  - wider networks are composed by interconnecting simple topologies (local area networks)

- common L2 interconnection devices:
  - bridge
    - transparent network interconnection (all the traffic passes the bridge)
    - separates shared media (collisions do not pass the bridge)
  - switch
    - $\approx$ multi-port bridge

L2 – Data Link Layer

L2 Networks II.

- based on MAC addresses
  - *Backward Learning Algorithm* – the bridge "learns" the locations of network stations (nodes) by listening on the media (observing the source addresses)
  - the frames are switched based on the receiver address
- characteristics:
  - it's possible to create networks with loops (cycles)
    - *Distributed Spanning Tree Algorithm* for the spanning tree calculation is used
  - not suitable for large networks
    - switch tables grow with the number of stations – low convergence

L2 – Data Link Layer

Distributed Spanning Tree Algorithm I.

- **the algorithm goal:** to disable (disuse) some bridges' ports (in order to prevent loops)
- every bridge sends periodical reports
  - <own address, root bridge address, currently known cost of the path to the root bridge>
- once a bridge receives a report from its neighbour, it adapts its idea about the "best" path:
  - it prefers the root with lower address
  - it prefers lower path costs
  - in the case of same paths' costs it prefers lower address
- mechanism:
  - root bridge selection (the lowest address)
  - sequential growth of the tree
  - the "best" paths found define the active bridges' ports
  - the other ports are disabled

L2 – Data Link Layer

Distributed Spanning Tree Algorithm II.

- *root bridge selection phase*
  - once started, all the bridges claim themselves as *Root Bridges* (and report this to the others)
  - each of them sends its report via all its ports
  - based on this information, the root bridge is selected (the lowest address)
- *root ports selection phase*
  - each bridge chooses its *Root Port* – the port with the lowest path cost to the Root Bridge
  - if two ports have the same costs, the one with lower Port ID is selected. The other is disabled (it becomes *non-designated*) in order to prevent loops
- *active/inactive ports selection phase*
  - Root Bridge sets all its ports as active (*Designated*)
  - the bridges communicate via all the links, which do not contain Root Ports, and try to determine the one with the lowest Bridge ID. Once the one is selected, it sets its corresponding port as active; the other disables its port.

- see the animation: `http://frakira.fi.muni.cz/~jeronimo/vyuka/Cisco-spanning_tree.swf`

L2 – Data Link Layer

Résumé

- ensures the transmission of frames between two communicating devices (determined by their MAC addresses) interconnected via shared transmission media
  - ensuring the reliability of the transfer
  - preventing the receiver from the congestion
  - using the medium access control (MAC protocols)
- L2 networks (LANs):
  - (usually) bus, circle, and star topologies
  - the essential devices for building wider area networks are bridges and switches
  - *Backward Learning Algorithm* to determine stations' location (necessary for frames' switching)
  - *Spanning Tree Algorithm* is used for spanning tree determination
- *further information:*
  - PV169: Communication Systems Basics (doc. Staudek)
  - PV183: Computer Networks Technology (dr. Pelikán)
  - graph algorithms – PB165: Graphs and Networks (prof. Matyska, doc. Hladká, doc. Rudová)

L3 – Network Layer

Introduction

- **Network Layer:**
  - provides services for the *Transport Layer*:
    - receives *segments* from the Transport Layer and transforms them into *packets*
    - in cooperation with the Data Link Layer ensures the packets' transmission between communicating nodes *(even between different LANs)*
  - logically joins independent LAN networks
    - the upper layers are provided with an illusion of just a single wide-area network (*WAN*)
  - allows unique identification (addressing) of every host/device on the Internet
  - ensures *routing* of passing packets
  - in cooperation with the Data Link Layer associates the L3-addresses with the L2/MAC-addresses (and vice versa)
  - further services: multicast

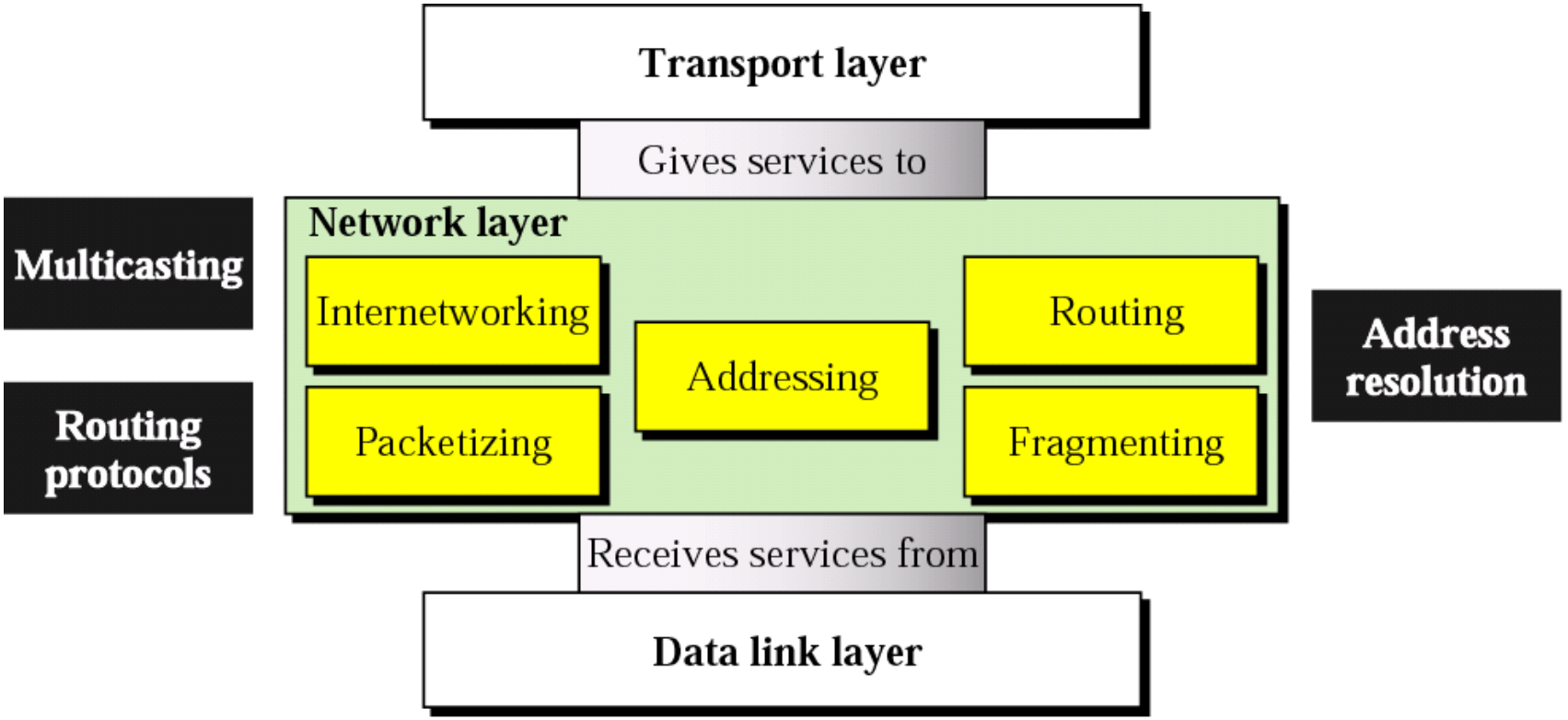


Figure: Position of the Network Layer.

L3 – Network Layer

Services I.

- *Internetworking*
  - logical gluing of heterogeneous physical networks together to look like a single network (from the upper layers' point of view)
    - by such an interconnection, an *internetwork* (shortly *internet*) is created
  - an illusion of a uniform environment provided by a single wide-area network
- *Packetizing*
  - received segments are transformed into packets
- *Fragmenting*
  - a technique to solve the problem of heterogeneous MTUs – when a datagram is larger than the MTU of the network over which it must be sent, it is divided into smaller fragments which are each sent separately
- *Addressing*
  - the entity addresses used on the network layer – so-called *IP addresses*, unique throughout the whole network
  - packets contain source and destination addresses of communicating entities

L3 – Network Layer

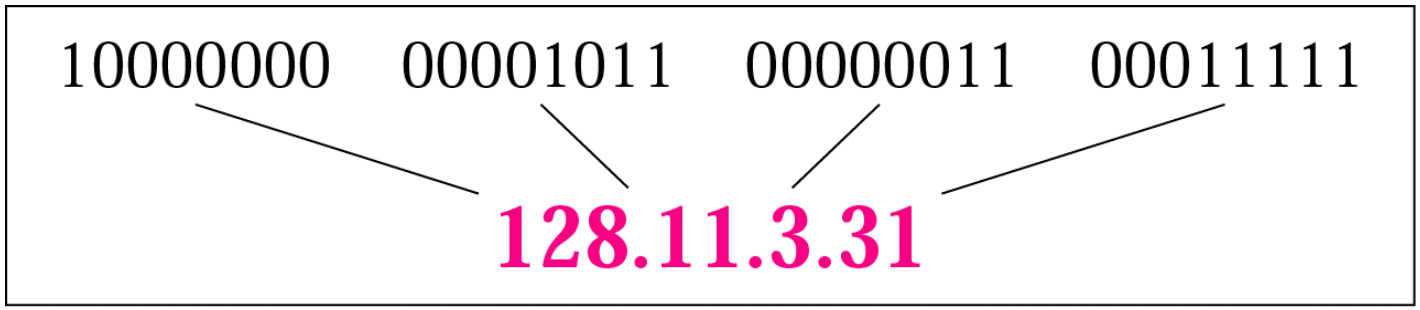
Services II.

- *Address Resolution*
  - ARP, RARP protocols
- *Routing*
  - the process of selecting paths in a network along which to send network traffic from a source to a particular destination
- *Control Messaging*
  - providing basic information about unavailability to deliver a packet, about a network/host state, etc. – ICMP protocol

L3 – Network Layer

Addressing

- a requirement to *uniquely identify* every host/device connected to the Internet
- a necessity to *systematic address assignment*
  - in order to simplify the routing process
- every device/interface is assigned an *Internet address* (*IP address*)
  - *IPv4 address* (32 bits) vs. *IPv6 address* (128 bits)



L3 – Network Layer

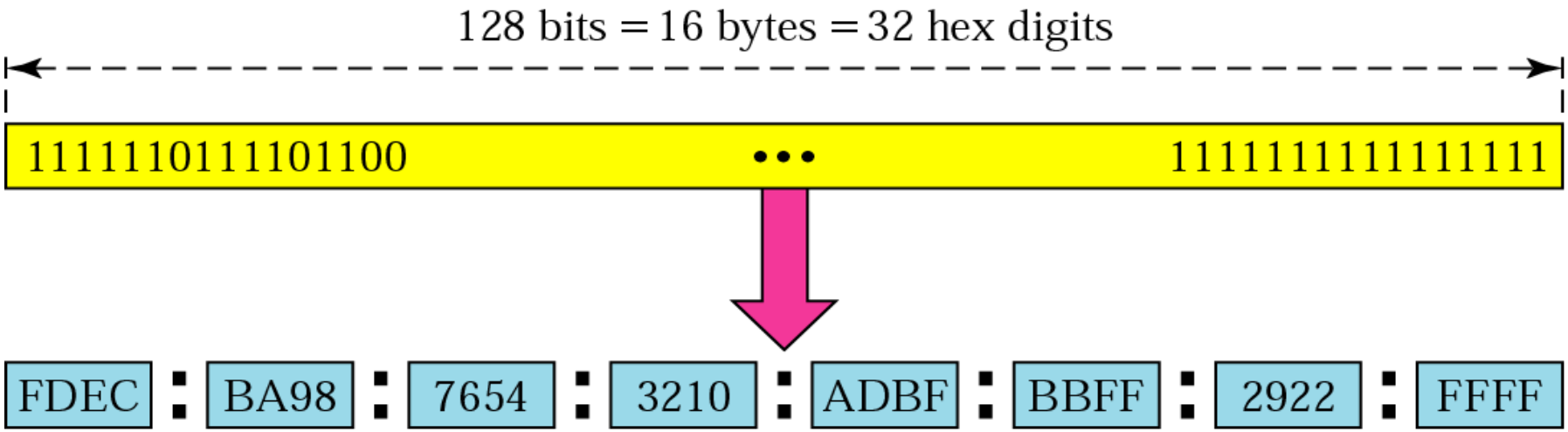
IPv4 Addresses – types

- *Unicast Address* – an identification of a single network interface
  - identification of a single sender/receiver
- *Broadcast Address* – in this case, the data are sent to all the hosts on the particular LAN ("all-hosts broadcast")
  - the source address of such datagrams (sender identification) is unicast address
- *Multicast Address* – used for an identification of a group of receivers (network interfaces) who **applied for the data**
  - routers send such data to all the group members
  - the source address of such datagrams (sender identification) is unicast address

L3 – Network Layer

IPv6 Addresses

- addresses used by the IPv6 protocol (see later)
- (currently) final solution of IP address space shortage
- IPv6 address has 128 bits (= 16 Bytes):
  - 2^{128} of possible addresses ($\approx 3 \times 10^{38}$ addresses $\Rightarrow \approx 5 \times 10^{28}$ addresses for every human on the Earth)
  - a hexadecimal notation instead of decadic notation (in pairs of bytes separated by ":")

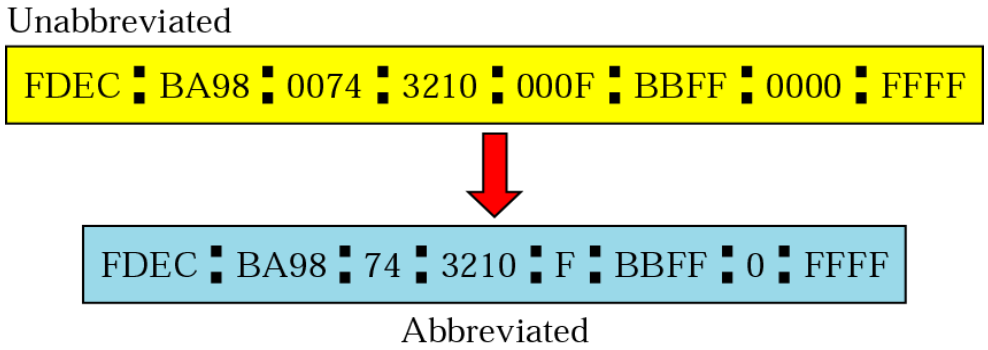


L3 – Network Layer

IPv6 Addresses – address abbreviation

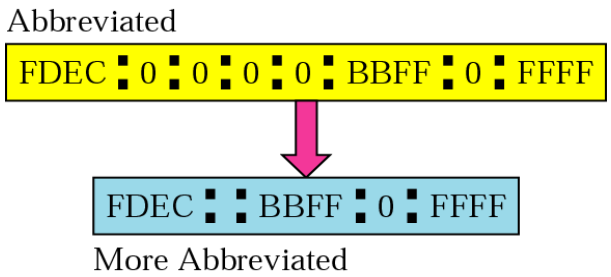
Leading zeros might be omitted in every group:

- 0074 might be written as 74, 000F as F, ...
- 3210 **cannot** be abbreviated!



Consecutive groups of zeros might be omitted:

- and replaced by the "::" symbol
- just **a single** sequence of zero groups might be abbreviated!



L3 – Network Layer

IPv6 Addresses – types

- *Unicast Address* – same as in IPv4 (an identification of a single network interface)
- *Multicast Address* – same as in IPv4 (used for addressing a group of receivers)
  - the data are delivered to all members of the particular groups
  - prefix `ff00::/8`
- *Anycast Address* – a newbie
  - identifies a group of receivers like multicast
  - but the data are delivered just to a single member of such a group (the closest one)

- IPv4 broadcast addresses are not used in IPv6
  - they were substituted by particular multicast groups (e.g., a group of all hosts/routers on the particular LAN)

L3 – Network Layer

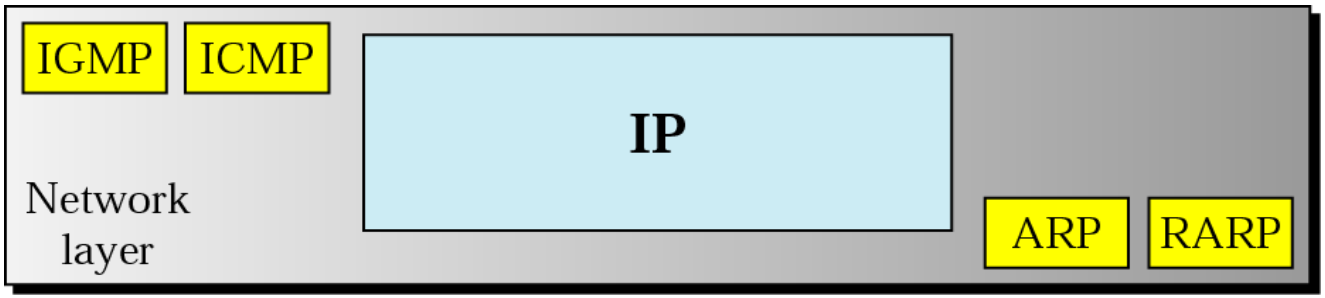
Internet Protocol (IP) I.

- the most widespread network layer protocol
  - ensures data (in pieces called *datagrams*) delivery, even through an intermediate nodes (called *routers*) – *host-to-host delivery*
    - hosts/interfaces are identified by their IP addresses
    - uses *datagram approach* to packet switching, the communication is connectionless
    - $\Rightarrow$ routing
  - provides an unreliable (so-called *best-effort*) service
  - supplemented by a set of supporting protocols (ICMP, ARP, RARP, IGMP)
    - used for nonstandard situations treatment, a distribution of information necessary for correct routing, L2 identification of network interfaces (MAC addresses), etc.
- proposed and standardized in two versions:
  - *Internet Protocol version 4 (IPv4)* – 1981, RFC 791
  - *Internet Protocol version 6 (IPv6)* – 1998, RFC 2460

L3 – Network Layer

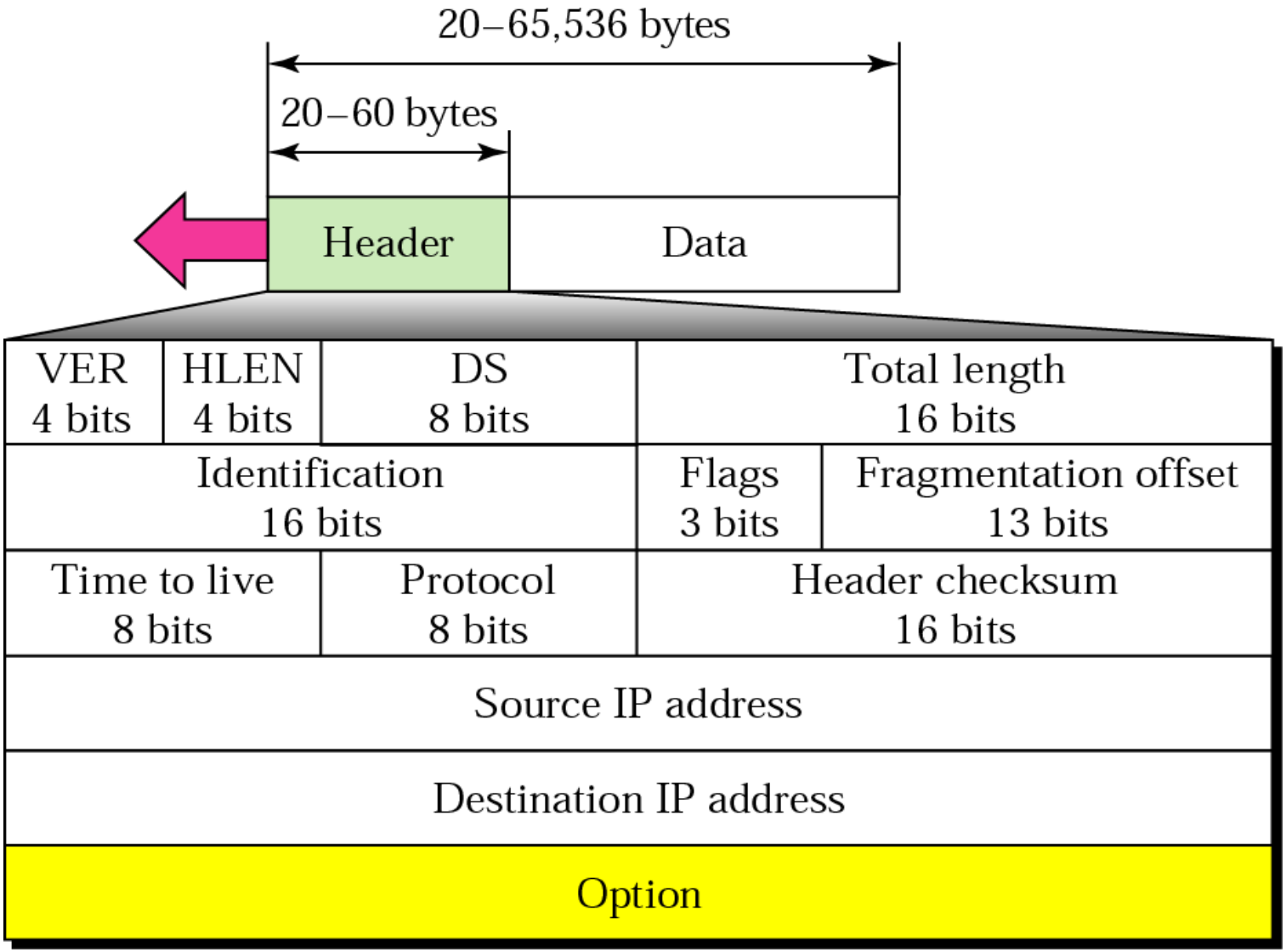
Internet Protocol (IP) II.

Supplementary protocols:



L3 – Network Layer

IPv4 Datagram



L3 – Network Layer

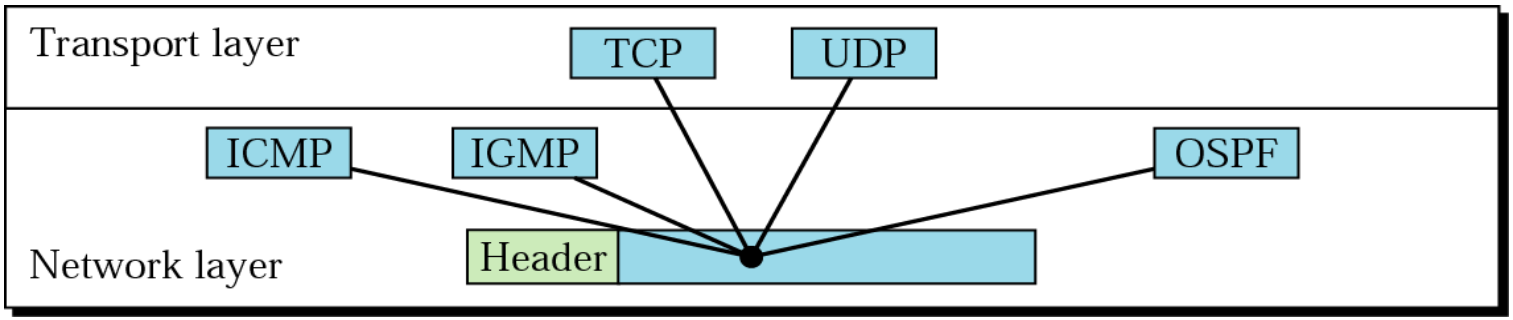
IPv4 Datagram II.

- **Version (VER)** – IP protocol version
- **Header length (HLEN)** – the length of IP datagram header (in 4B words)
  - because of the *Option* field, which makes the length of the header variable
- **Differentiated services (DS)** or **Type of service (TOS)** – defines the class of the datagram for quality-of-service (QoS) purposes
  - necessary for a distinction of "important" (control datagrams, real-time data) and "less important" datagrams
- **Total length** – the length of the whole IP IP datagram (in B)
  - max. $2^{16} - 1 = 65535$ bytes
- **Identification**, **Flags**, **Offset** – fields used for fragmentation
- **Time to live (TTL)** – used to control the maximum number of hops (router) visited by the datagram
  - the sending nodes stores a number in this field ($\approx 2\times$ the biggest number of hops between any two hosts in the network)
  - each router decrements this number by 1
  - if this value equals to zero ($TTL = 0$), the datagram is discarded
  - the purpose is to prevent a datagram from becoming an errant

L3 – Network Layer

IPv4 Datagram III.

- **Protocol** – higher-level protocol identification
  - specifies the final destination protocol to which the IP datagram should be delivered
    - this value helps in multiplexing/demultiplexing process
  - the identifiers are specified by IANA organization
    - e.g., 1 = ICMP, 2 = IGMP, 6 = TCP, 17 = UDP, etc.
    - see `http://www.iana.org/assignments/protocol-numbers`



L3 – Network Layer

IPv4 Datagram IV.

- **Header checksum** – the checksum of the IP *header*
  - data are *not* included
    - data checksums are provided by the L4-level (Transport Layer)
  - the main reason for distinction:
    - the checksums have to be recomputed on every router the datagram visits because of header changes (e.g., TTL field)
    - $\Rightarrow$ computing the checksum from the header only is simpler = the processing is faster
- **Source IP address**, **Destination IP address** – 32-bit IPv4 address identifying sender/receiver
- **Options** – optional part of IP datagrams, used especially for network testing and debugging
- **Data** – the data being transferred

L3 – Network Layer

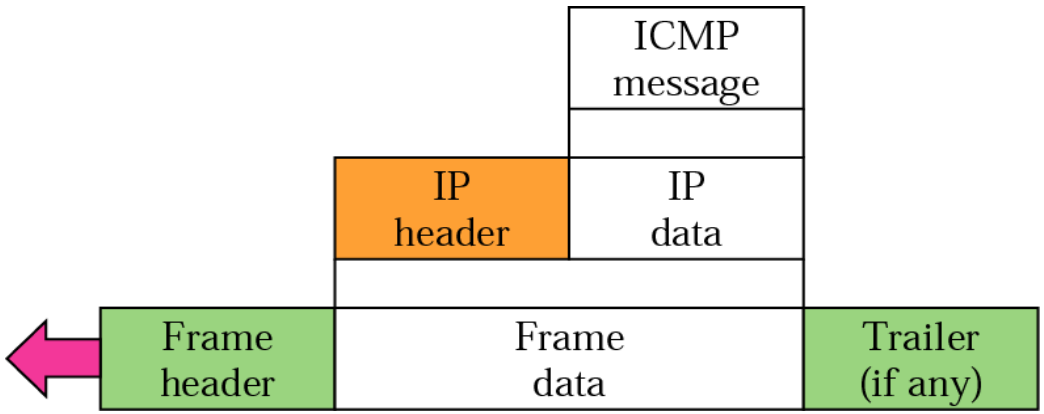
Internet Control Message Protocol (ICMP) I.

- IP protocol provides unreliable (best-effort) service
  - without any mechanism to inform the sender about errors arisen during data delivery
  - without any mechanism for network state testing
- *Internet Control Message Protocol (ICMP)*
  - RFC 792
  - a supplementary protocol for IP protocol
  - provides information about errors arsed during data delivery
  - provides basic information about the network state
- message examples:
  - *Destination unreachable* – "Destination" is a protocol, port, host, or network
  - *Time exceeded* – an information about TTL expiration or when all fragments that make up a message do not arrive at the destination host within a certain time limit
  - *Echo request/reply* – a request for reply

L3 – Network Layer

Internet Control Message Protocol (ICMP) II.

ICMP Encapsulation:



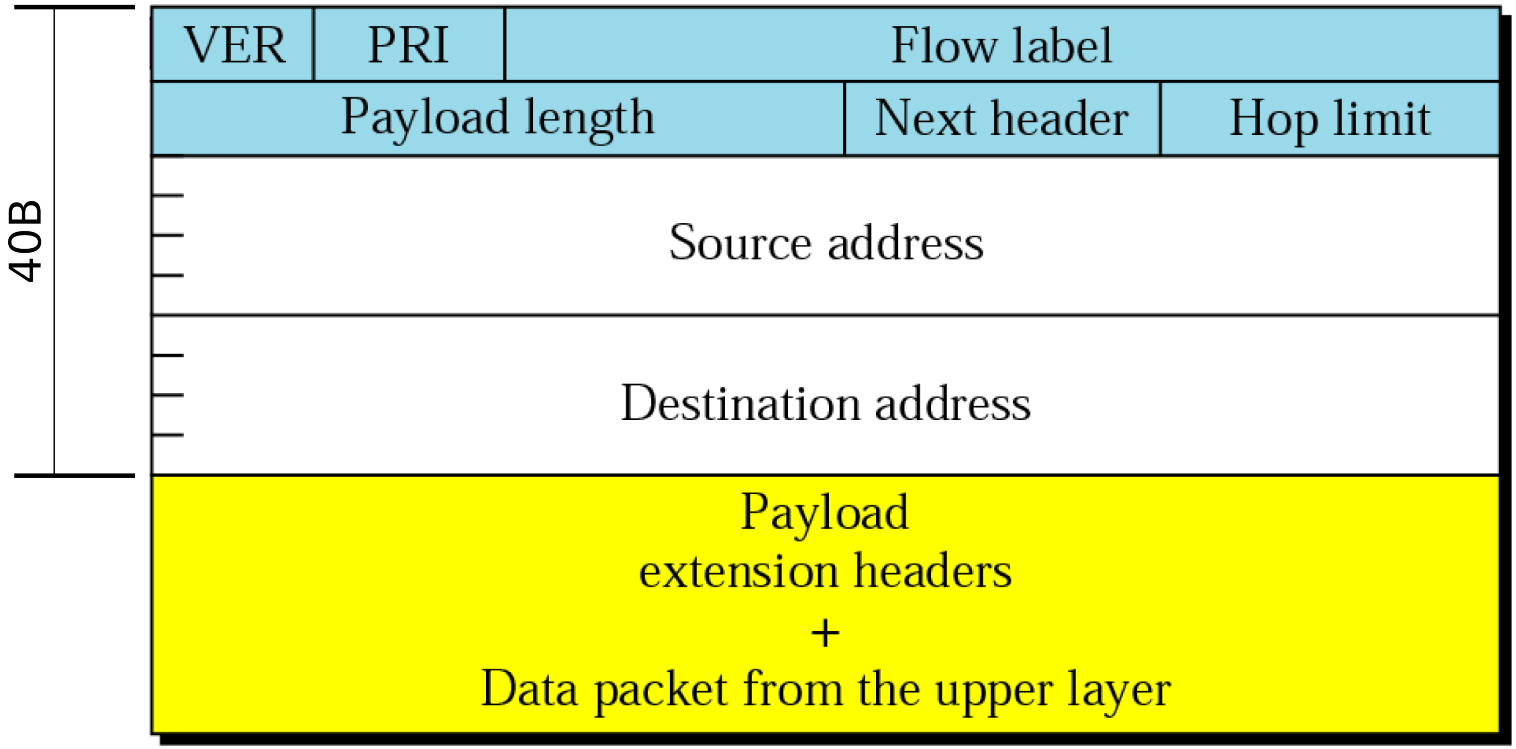
L3 – Network Layer

IP Protocol version 6 (IPv6) – main features

- *larger address space* – 128-bit IPv6 address, 2^{128} of unique addresses
- *better (simpler) header format* – basic 40B header containing just the most important information
- *allowance for extensions* – via so-called *extension headers*
- *Support for real-time transfers* – flows' tagging, flows' priorities
- *Support for more security* – data authentication, encryption, and integrity support
- *Mobility support* – via so-called *home agents*
- *Device autoconfiguration support* – statefull and stateless autoconfiguration

L3 – Network Layer

IPv6 Datagram – basic header I.



- fixed (40B) header length
- checksum, options, and fragmenting information are not included in basic header any more
  - options and fragmenting information available via extension headers
  - checksum removed without any compensation (ensured on L2 and L4)

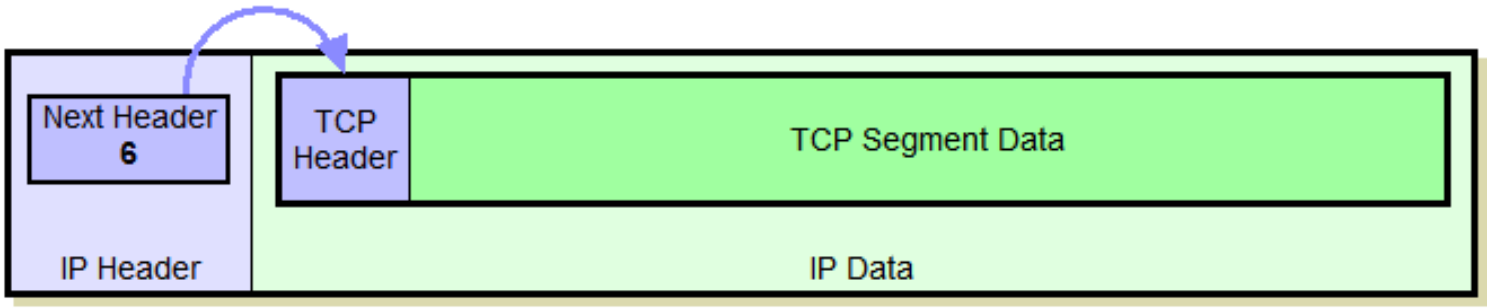
L3 – Network Layer

IPv6 Datagram – basic header II.

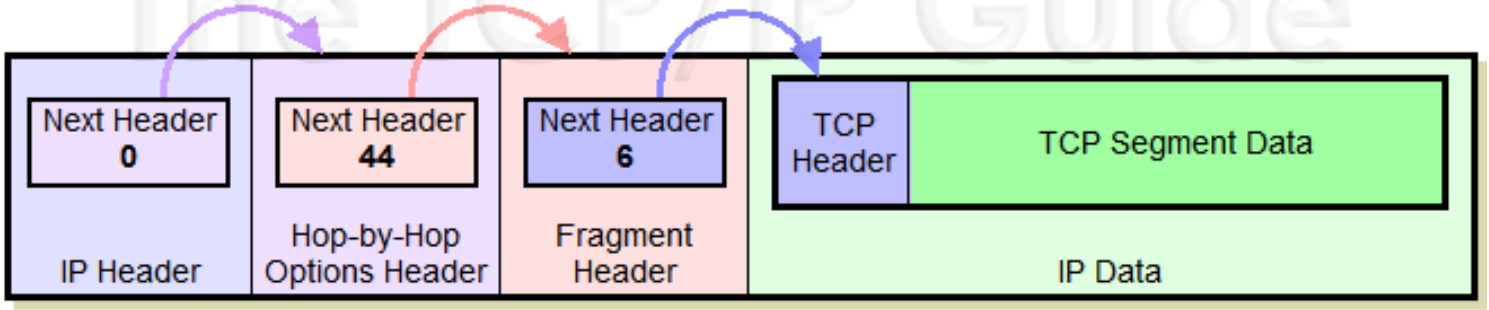
- **Version (VER)** – version number of the IP (currently 6)
- **Priority (PRI)**, also *Traffic Class* – datagram priority with respect to traffic congestion
- **Flow label** – designed to provide special handling for a particular flow of data
  - not widely used yet
- **Payload length** – the total length of the IP datagram excluding the base header
- **Next header** – defines the header that follows the base header in the datagram (extension header or transport header)
- **Hop limit** – $\approx$ TTL in IPv4
- **Source/Destination address** – IPv6 address of source/destination node

L3 – Network Layer

IPv6 Datagram – extension headers



IPv6 Datagram With No Extension Headers Carrying TCP Segment



IPv6 Datagram With Two Extension Headers Carrying TCP Segment

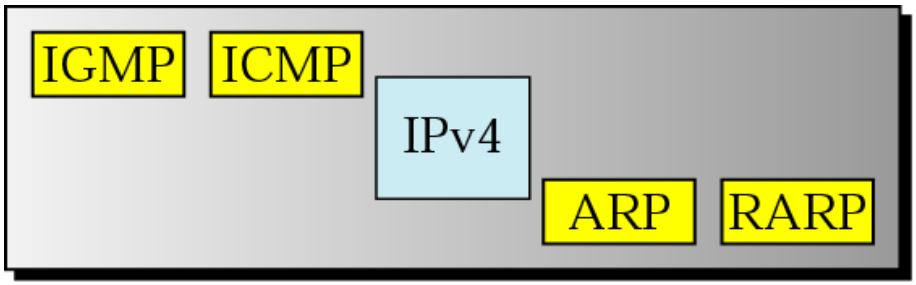
several extension headers have been defined

- e.g., Hop-By-Hop Options, Routing, Fragment, Encapsulating Security Payload, Authentication Header, etc.

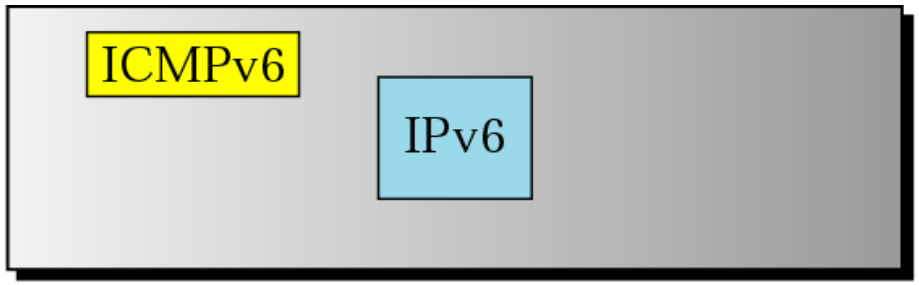
L3 – Network Layer

ICMPv6

- ICMP protocol version 6 (ICMPv6)
  - based on same mechanisms as the ICMPv4
  - moreover, includes the functionality of ARP and IGMP protocols
    - using so-called *Neighbour Discovery* protocol working in cooperation with ICMPv6



Network layer in version 4



Network layer in version 6

L3 – Network Layer

Routing

- **Routing** = the process of finding a path in the network between two communicating nodes
  - the route/path has to satisfy certain constraints
  - influenced by several factors:
    - *static ones:* network topology
    - *dynamic ones:* network load

L3 – Network Layer

The Global View Problem

- the global knowledge of network topology is problematic
  - it's very difficult to acquire it
  - if yet acquired, it's not actual any more
  - it has to be locally relevant
- a local view of network topology represents a *routing table*
- the difference between local and global knowledge can lead to:
  - cycles/loops (i.e., black holes)
  - oscillation (load adaptability)

L3 – Network Layer

Routing – the goal

- the main goal of routing is:
  - to find optimal paths
    - the optimality criterion is a *metric* – a cost assigned for passing through a network
  - to deliver a data packet to its receiver
- the routing *usually* does not deal with the whole packet path
  - the router deals with just a single step – to whom should be the particular packet forwarded
    - somebody "closer" to the recipient
    - so-called *hop-by-hop* principle
  - the next router then decides, what to further do with the received packet

L3 – Network Layer

Routing – basic approaches

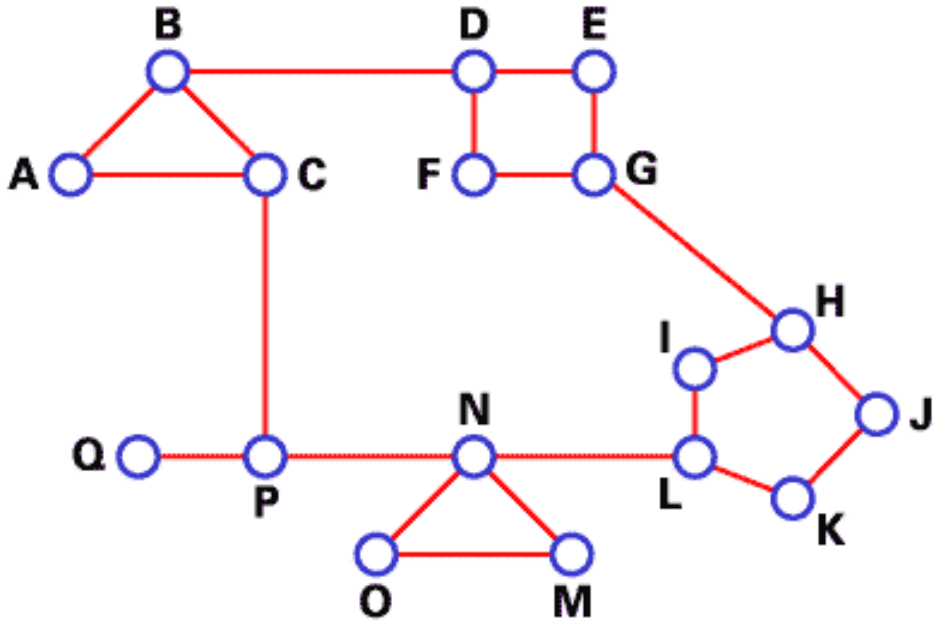
The basic approaches divide based on the routing table creation/maintenance:

- *static (non-adaptive)*
  - manually (by hand) edited records
  - suitable for a static topology and smaller networks
- *dynamic (adaptive)* – these respond to network changes
  - complex (usually distributed) algorithms
  - e.g.:
    - *centralized* – a centre controls the whole routing
    - *isolated* – every node on its own
    - *distributed* – nodes' cooperation

L3 – Network Layer

Routing – mathematical view

- the routing can be seen as a problem of graph theory
- a network can be represented by a graph, where:
  - nodes represent routers (identified by their IP addresses)
  - edges represent routers' interconnection (a data link)
  - edges' value = the communication cost
  - *the goal:* to find paths having minimal costs between any two nodes in the network



L3 – Network Layer

Routing – routing algorithms' required features

Required features of any routing algorithm:

- accuracy
- simplicity
- effectivity and scalability
  - to minimize an amount of control information ($\approx 5\%$ of the whole traffic!)
  - to minimize routing tables' sizes
- robustness and stability
  - a distributed algorithm is necessary
- fairness
- optimality
  - *"What should be treated as the best path?"*

L3 – Network Layer

Routing – basic approaches to distributed routing

Basic approaches to distributed routing:

- *Distance Vector (DV)* – Bellman-Ford algorithm
  - the neighboring routers periodically (or when the topology changes) exchange complete copies of their routing tables
  - based on the content of received updates, a router updates its information and increments its *distance vector number*
    - a metric indicating the number of hops in the network
  - i.e., *"all pieces of information about the network just to my neighbors"*
- *Link State (LS)*
  - the routers periodically exchange information about states of the links, to which they are directly connected
  - they maintain complete information about the network topology – every router is aware of all the other routers in the network
  - once acquired, the Dijkstra algorithm is used for shortest paths computation
  - i.e., *"information about just my neighbors to everyone"*

L3 – Network Layer

Distance Vector – RIP protocol

- the principal actor of DV routing
  - RIPv1 (RFC 1058)
  - RIPv2 (RFC 1723) – adds several features (e.g., an authentication of routing information)
- the networks are identified using the CIDR mechanism
- the number of hops is used as a metric
  - transfer of a packet between two neighboring routers = 1 hop
  - infinity = 16
    - $\Rightarrow$ the RIP cannot be used for networks with minimal amount of hops between any two routers > 15
- the routers send the information periodically every 30 seconds
  - triggered updates when a state of a link changes
  - timeout 180s (detection of connection errors)
- usage:
  - suitable for small networks and stable links
  - not advisable for redundant networks

L3 – Network Layer

Link State – OSPF protocol

- *Open Shortest Path First*
- currently the mostly used LS protocol
- metric: *cost*
  - a number (in the range between 1 and 65535) assigned to each router's network interface
  - the lower the number is, the better the link/path is (i.e., will be preferred)
  - by default, every interface is automatically assigned a cost derived from the link's throughput
    - $cost = 100000000/bandwidth$ (bw in bps)
    - might be manually edited
- extensions:
  - message authentication
  - routing areas – next layer of hierarchy
  - load-balancing – more links/paths with the same cost

L3 – Network Layer

Routing – Link State vs. Distance Vector

Link State

- *Complexity:*
  - every node has to know the cost of every link in the network $\Rightarrow$ $O(nE)$ messages
  - once a link state changes, the change has to be propagated to *every* node
- *Speed of convergence:*
  - $O(n^2)$ alg., sends $O(nE)$ messages
  - sustains from oscillations
- *Robustness:*
  - wrongly functional/compromised router spreads wrong information just about the links it is directly connected to
  - every router computes routing tables on its own $\Rightarrow$ separated from routing information propagation $\Rightarrow$ a form of robustness
- *Usage:*
  - suitable for large networks

Distance Vector

- *Complexity:*
  - once a link state changes, the change has to be propagated just to the *closest neighbors*; it is further propagated just in cases, when the changed state leads to a change in the current shortest paths tree
- *Speed of convergence:*
  - may converge more slowly than LS
  - problems with routing loops/cycles, *count-to-infinity* problem
- *Robustness:*
  - bad computation is spread through the network $\Rightarrow$ may lead to a "confusion" of other routers (bad routing tables)
- *Usage:*
  - suitable just for smaller networks

L3 – Network Layer

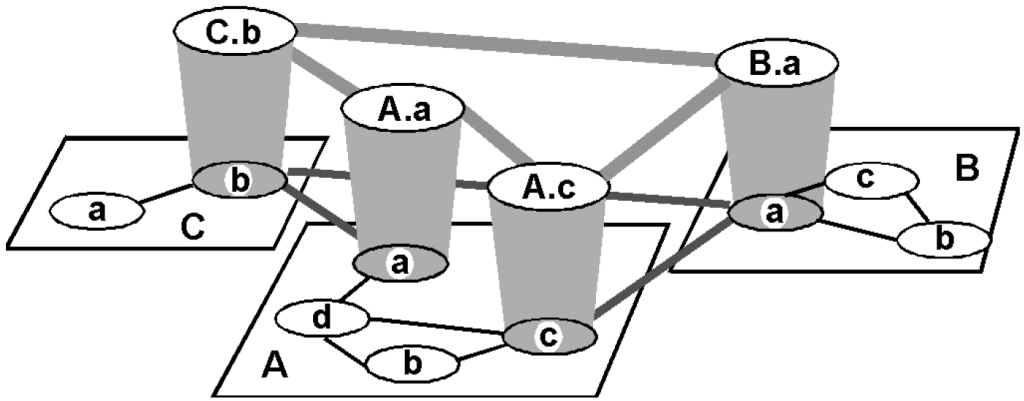
Autonomous Systems

- the goal of Internet's division into *Autonomous Systems* is
  - a reduction of routing overhead
    - simpler routing tables, a reduction of exchanged information, etc.
  - a simplification of the whole network management
    - particular internets are managed by various institutions/organizations
- autonomous systems = domains
  - a 16bit identifier is assigned to every AS/domain
    - *Autonomous System Number (ASN)* – RFC 1930
    - assigned by *ICANN (Internet Corporation For Assigned Names and Numbers)*
  - correspond to administrative domains
    - networks and routers inside a single AS are managed by a single organization/institution
    - e.g., CESNET, PASNET, ...
  - a distinction according to the way an AS is connected to the Internet:
    - *Stub AS*
    - *Multihomed AS*
    - *Transit AS*

L3 – Network Layer

Autonomous Systems – routing

- separated routing because of scalability reasons:
  - *interior routing*
    - routing inside an AS
    - under the full control of AS's administrator(s)
    - the primary goal is the performance
    - so-called *Interior Gateway Protocols (IGP)* (e.g., RIP, OSPF)
  - *exterior routing*
    - routing among ASs
    - the primary goal is the support of defined policies and scalability
    - so-called *Exterior Gateway Protocols (EGP)* (e.g., EGP, BGP-4)
  - a cooperation of interior and exterior routing protocols is necessary



L3 – Network Layer

Autonomous Systems – exterior routing (BGP)

- *Border Gateway Protocol*
  - currently version 4 (BGP-4)
- proposed due to Internet's grow and demands on complex topologies support
  - supports redundant topologies, deals with loops/cycles
- employs so-called *Path Vector* routing
  - not only paths' costs, but the full descriptions of the whole paths are exchanged
- allows a definition of routing rules (policies)
- makes use of the fully reliable TCP protocol
- uses CIDR for paths' aggregation

L3 – Network Layer

IP Multicast

A classical solution of group communication in the network:

- Just a single data copy goes every network link
- A feature of the network (hop-by-hop service, no end-to-end service)
- Non-reliable delivery (best effort, UDP, group address)
- Spread wideness restricted by TTL (Time To Live) field of packets

How to identify a group?

- $\Rightarrow$ multicast IP address
  - *IPv4:* class D (`224.0.0.0` – `239.255.255.255`)
  - *IPv6:* prefix `ff00::/8`

Two basic approaches to multicast routing:

- *Source Based Tree*
- *Shared Tree (Core Based Tree)*

L3 – Network Layer

IP Multicast – Source Based Tree vs. Core Based Tree

Source Based Tree

- Top-down activity (from the constituent)
- Periodic broadcast
- Cutting the subtrees with no clients
- Wideness restriction – TTL
- Suitable for closely located groups
- Drawbacks: overhead, flooding by broadcasts
- Protocols: DVMRP (RIP), MOSPF (OSPF), PIM–DM

Core Based Tree

- A core is established – ensured by meeting points (MPs)
- A client contacts a MP
- Down-top activity (from the receiver)
- Reduces broadcast $\longrightarrow$ better scalability
- Drawback: a dependence on the core availability
- Protocols: CBT, PIM–SM

L4 – Transport Layer

Introduction

Transport Layer:

- provides its services to the *Application Layer*:
  - obtains data coming from sending application and transforms them into *segments*
  - received segments delivers to the destination application
- in cooperation with the network layer ensures data (segments) delivery between communicating *applications/processes*
  - providing transmission reliability, if required
  - provides them with a logical communication channel
    - an illusion of direct physical interconnection
  - so-called *process-to-process delivery*
- the lowest layer providing so-called *end-to-end* services
  - the headers generated on the sender's side are interpreted "only" on the receiver's side
  - the transport layer data are seen by routers as a payload of transmitted packets

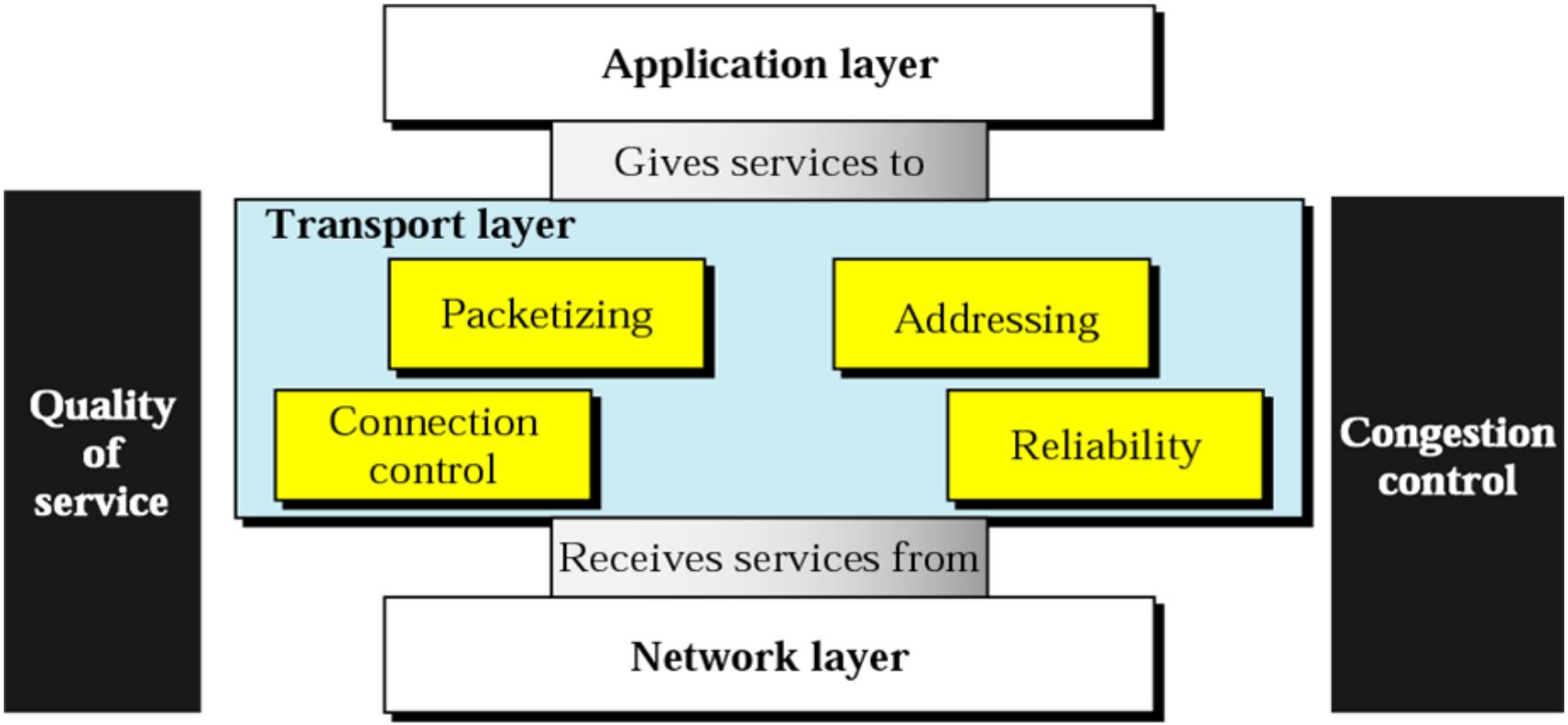


Figure: Position of the Transport Layer.

L4 – Transport Layer

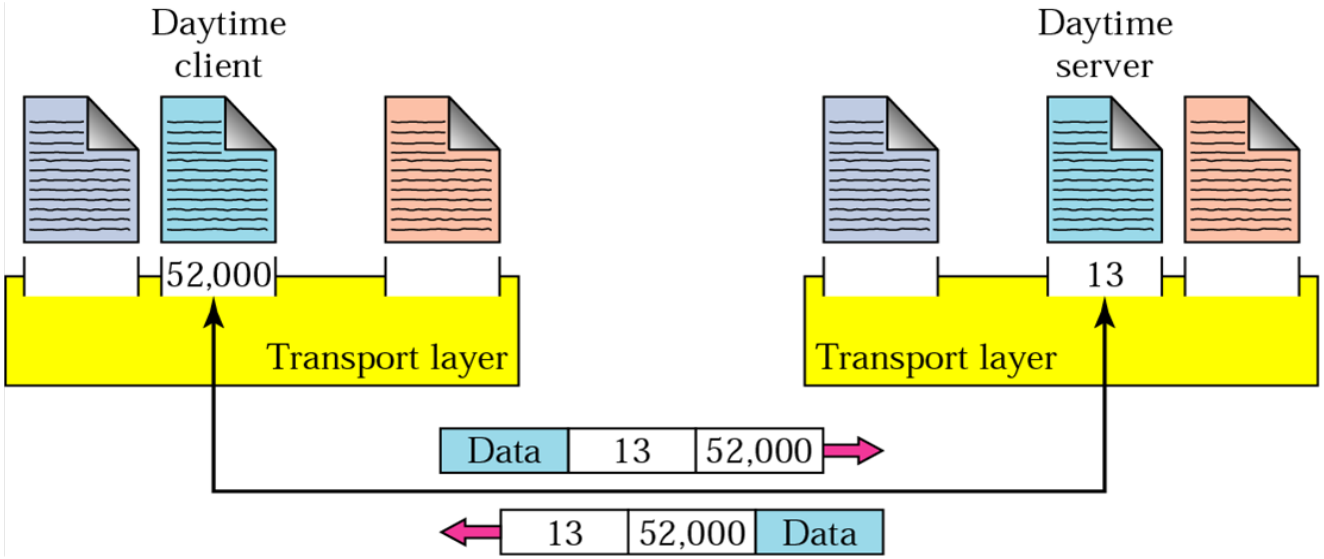
Services

- *Packetizing*
  - the data provided by an application are transformed into packets (having a transport header added)
- *Connection Control*
  - *connection-oriented* and *connectionless* services
- *Addressing*
  - the addresses of transport layer entities (= network applications/services) – so-called *ports*
  - the packets contain source and destination ports (an identification of source and destination application)
    - an application is uniquely identified in the network by the pair *IP_address:port*
- *Connection Reliability*
  - *Flow Control* and *Error Control*
    - provided on the node-to-node principle by lower layers, L4 provides it on the *end-to-end* principle
  - ensures a reliability over *best-effort* service (IP)
- *Congestion Control and Quality of Service (QoS) ensurance*

L4 – Transport Layer

Addressing – ports

- addresses on L4 – *port numbers (ports)*
  - ≈ addresses of services
  - identify a sending application on the sender node (identified by its IP address)
  - identify a receiving application on the receiver node (identified by its IP address)
- ports are identified by *16-bit number*
  - range 0 – 65535



L4 – Transport Layer

Connection-oriented vs. Connection-less Services

Connection-oriented services

- prior to the transmission, a connection is established (and maintained during the whole transmission)
- packets are numbered
  - their delivery/undelivery is explicitly acknowledged

Connection-less services

- packets are sent to the destination application without any connection being established
- packets are not numbered ($\Rightarrow$ they aren't acknowledged)
  - might be lost, delayed, delivered out-of-order, etc.

L4 – Transport Layer

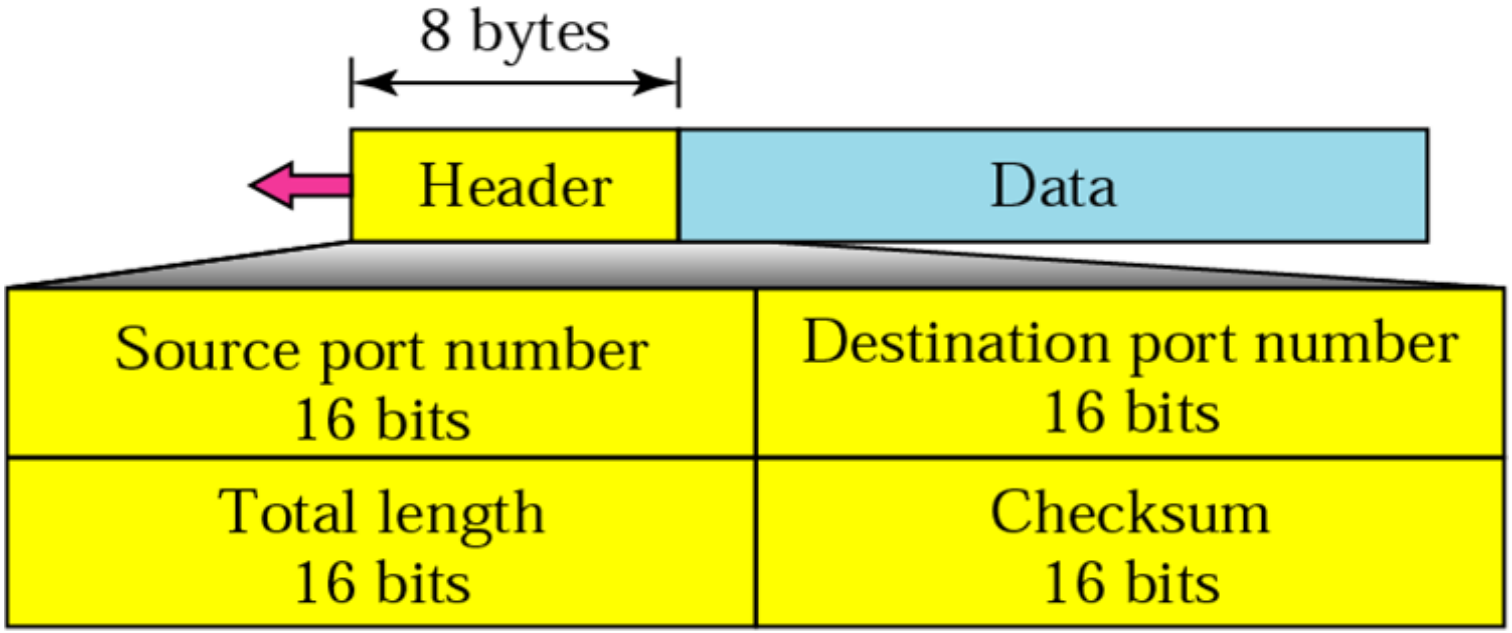
User Datagram Protocol (UDP)

User Datagram Protocol (UDP)

- the simplest transport protocol providing a **connection-less** and **unreliable** service
  - provides *best-effort* service
  - enriches the IP layer services just by process-to-process communication and simple error control
  - if a reliability has to be ensured, it must be provided by the application
- *main features:* simplicity, minimal overhead
  - no connection establishment/maintenance necessity (brings a delay in the beginning of the transmission)
  - no necessity to maintain state information by the communicating nodes
  - small/simple header
- selected applications:
  - processes requiring just a simple "request – reply" communication (e.g., the DNS (Domain Name Service))
  - processes/protocols with internal flow and error control (e.g., TFTP (Trivial File Transport Protocol))
  - real-time transfers
  - multicast transfers

L4 – Transport Layer

UDP header



- **source port** – the identification of sending service/application
- **destination port** – the identification of receiving service/application
- **length** – the total length of the UDP packet (header + data)
- **checksum** – the UDP packet checksum (header + data)

L4 – Transport Layer

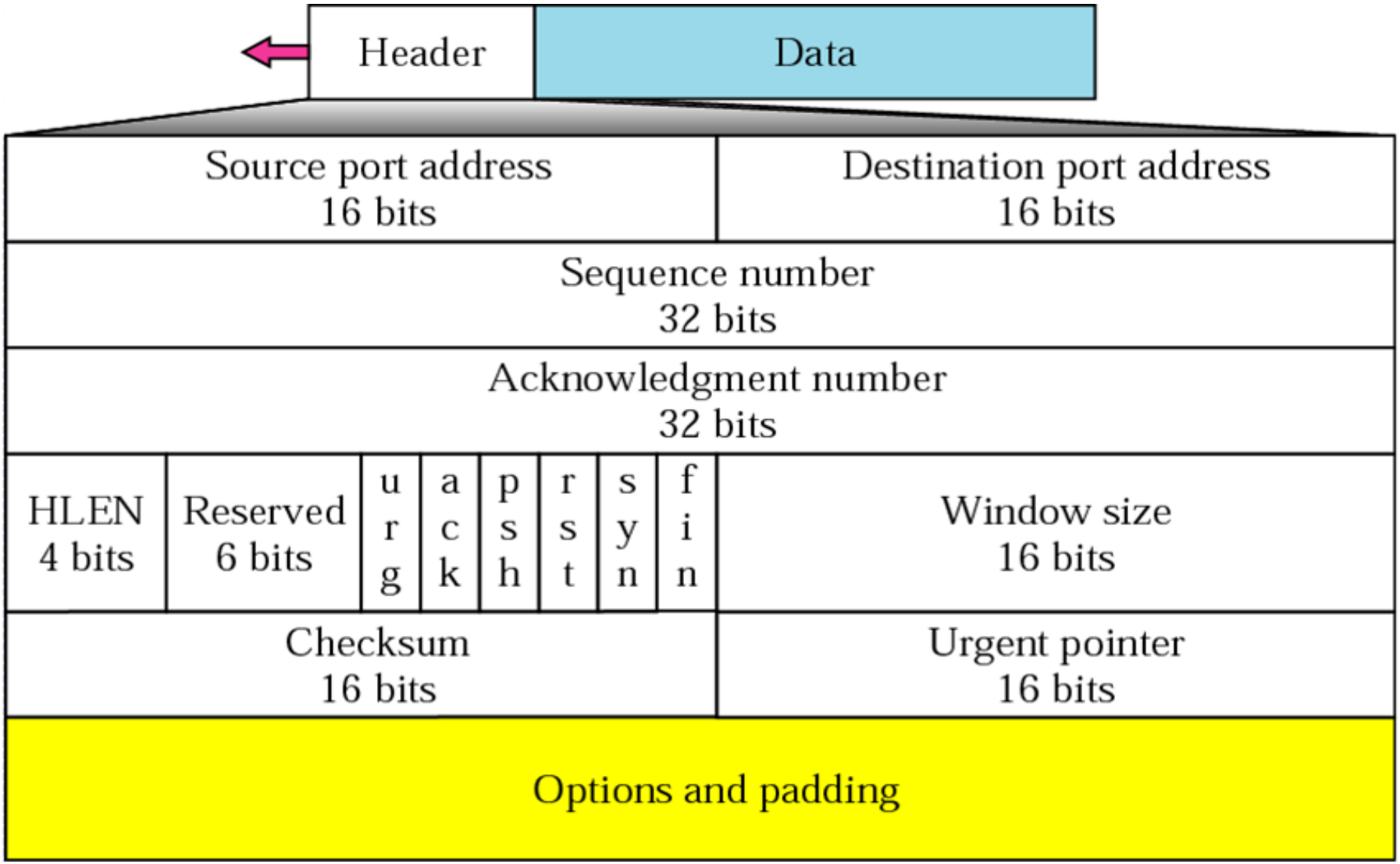
Transmission Control Protocol (TCP)

Transmission Control Protocol (TCP)

- transport protocol providing **connection-oriented** and fully **reliable** service
  - if possible, the data sent by the sender will be received by the receiver – complete and in the right order
  - in comparison with the UDP protocol, the TCP is byte-stream oriented (UDP works with blocks of data)
- prior to a communication, a *connection* has to be established between sender and receiver
  - so-called *three-way handshake* taking place prior to the communication ensures the exchange of all necessary information
  - the connection is distinguishable just on the end nodes (end-to-end service)
    - the routers are not aware about the connections
  - an established connection might be used for fully duplex communication
    - the control data are enclosed in the backward data (so-called *piggybacking*)
  - just **point-to-point** connections are supported
    - the communication among more peers (a-la multicast) is not supported
- multiplexing/demultiplexing and error control same as in the UDP

L4 – Transport Layer

TCP header I.



L4 – Transport Layer

TCP header II.

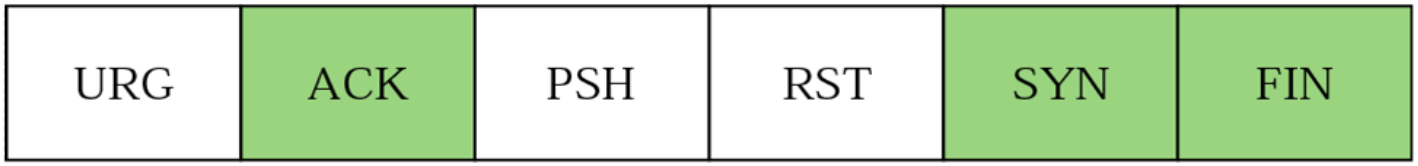
- **source port** – the identification of sending service/application
- **destination port** – the identification of receiving service/application
- **sequence number** – the number assigned to the first byte of data contained in the segment
- **acknowledgement number**
  - the byte number that the receiver is expecting to receive in the next segment
  - *piggybacking*
- **header length** – the total length of the TCP header (in 4B words)
- **reserved**

L4 – Transport Layer

TCP header III.

- **control** – 6 control bits identifying various control information

URG: Urgent pointer is valid	RST: Reset the connection
ACK: Acknowledgment is valid	SYN: Synchronize sequence numbers
PSH: Request for push	FIN: Terminate the connection



- **window size** – the size of the window that the other party must maintain
  - used for the *Flow Control* service (see the next slide)
- **checksum** – the checksum of the TCP segment (header + data)
- **urgent pointer** – used when the segment contains urgent data (out-of-order delivery)
- **options**

L4 – Transport Layer

Flow Control vs. Congestion Control I.

TCP controls the amount of sent data in such a way, that:

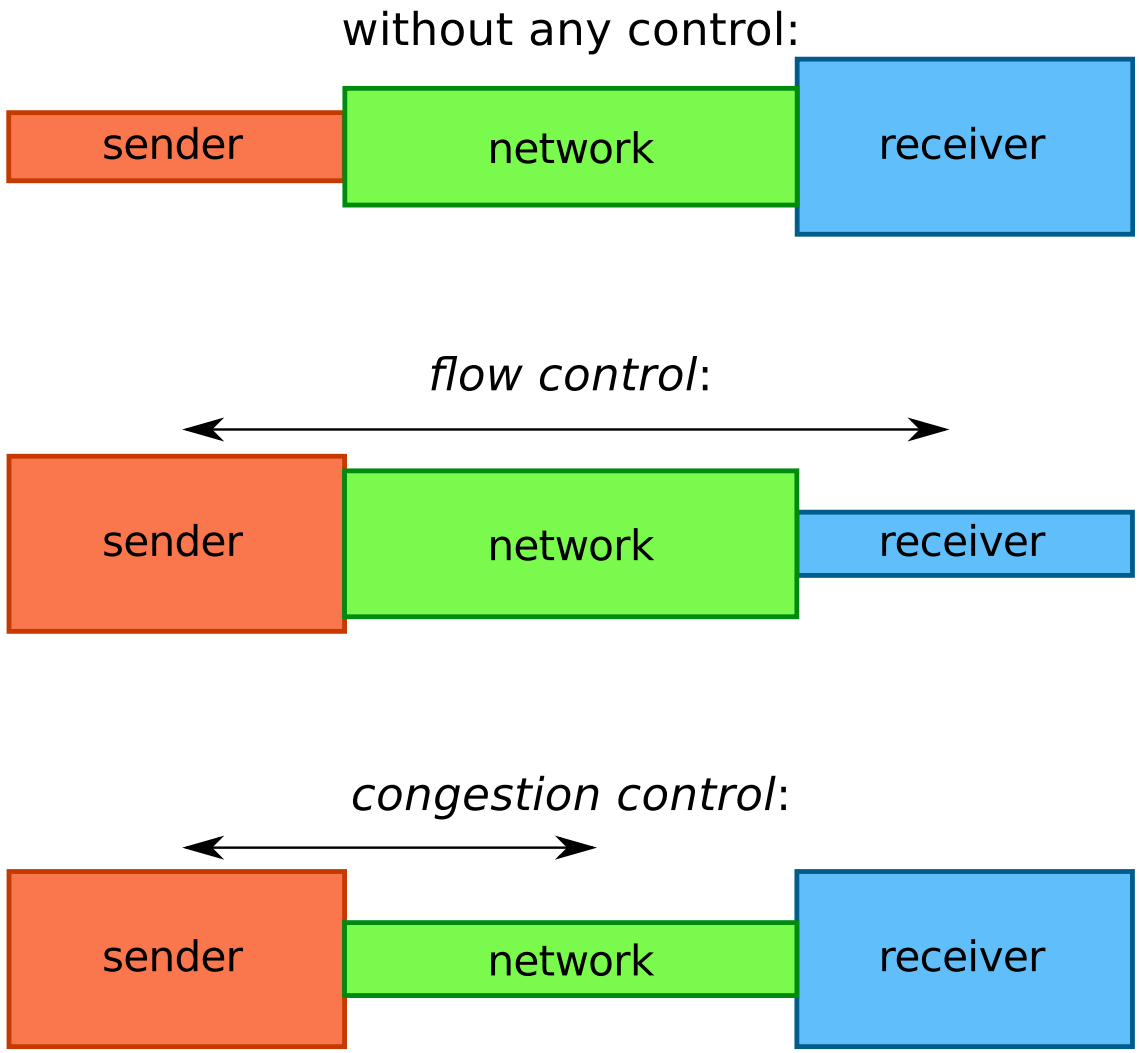
- *protects the receiver from being congested* = **Flow Control**
- *protects the network from being congested* = **Congestion Control**

The amount of data allowed to be sent to the network is defined by:

- the receiver's window size (flow control)
- by the size of so-called *congestion window* (congestion control)
  - maintained on the sender side
- the amount of data allowed to be sent to the network – limited by the **lower value** of both parameters

L4 – Transport Layer

Flow Control vs. Congestion Control II.



L4 – Transport Layer

Résumé

- ensures the communication of particular applications
- providing an optional reliability ensurance
  - UDP protocol for fast, but non-reliable packet transmission
    - just the error control (using checksums) is provided
  - TCP protocol for fully-reliable byte-stream transmission
    - the transmission reliability ensured by repeated sending (ARQ mechanisms)
    - provides a mechanism for flow control (receiver protection from a congestion) – explicit information provided by the receiver
    - provides a mechanism for congestion control (network protection from a congestion) – an estimation of available throughput (AIMD mechanism)
- *further information:*
  - PB156: Computer Networks (doc. Hladká)
  - PV183: Computer Networks Technology (dr. Pelikán)

L7 – Application Layer

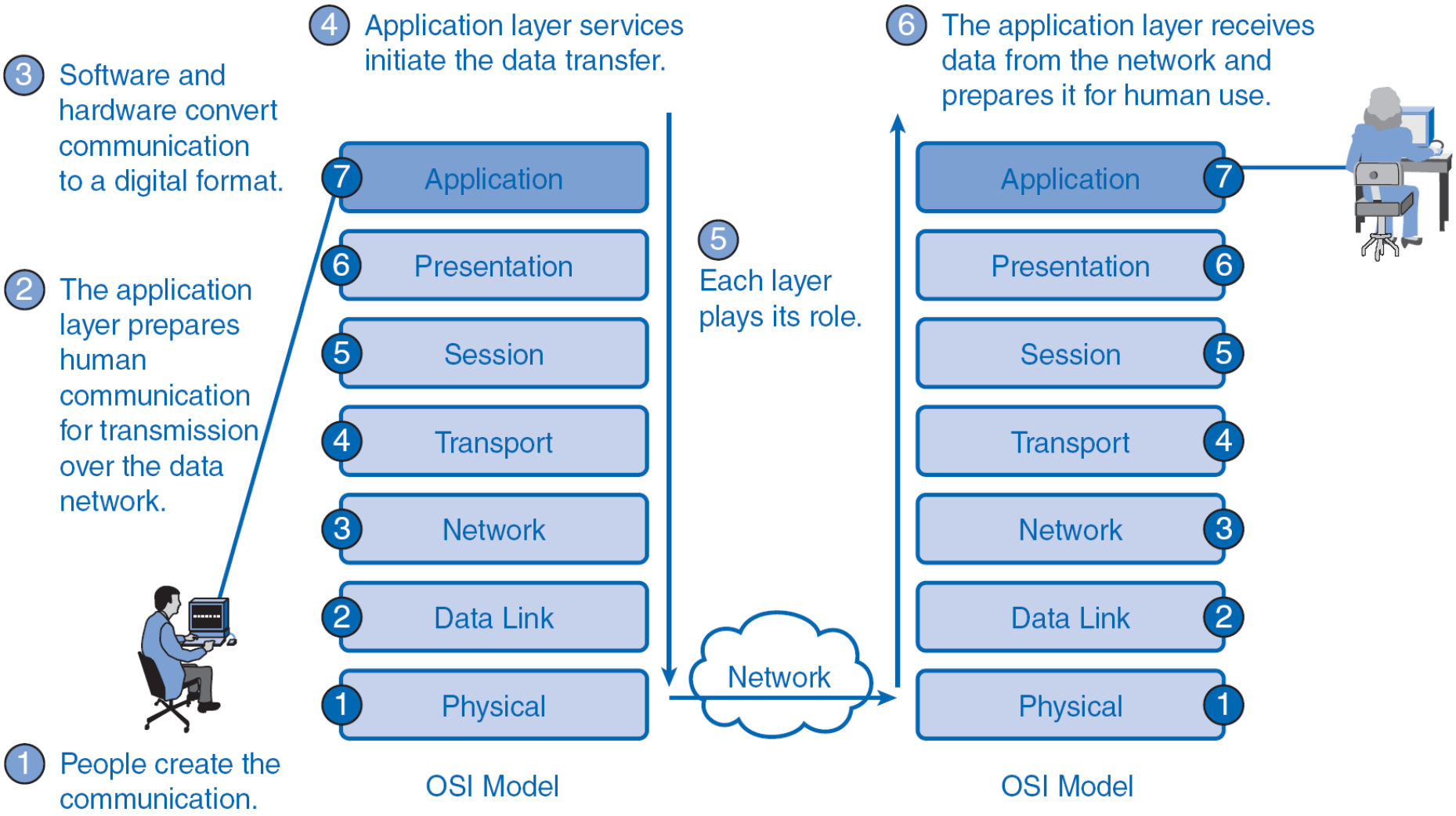
Introduction I.

Application Layer:

- provides services to *users*:
  - application programs specific for a particular purpose
    - e.g., electronic mail, WWW, DNS, etc. etc.
  - applications = the main reason for computer networks existence
- comprises *network applications/programs* and *application protocols*
  - application protocols (HTTP, SMTP, etc.) are **parts of** network applications (web, email)
    - they are not applications on their own
    - the protocols define a form of communication between communicating applications
  - application protocols define:
    - types of messages, which the applications exchange (*request/response*)
    - messages' syntax
    - messages' semantics (a semantics of particular fields)
    - rules, when and how the messages are exchanged

L7 – Application Layer

Introduction II.



L7 – Application Layer

Basic Application Classification/Distinction

According to employed communication model:

- Client-Server model
  - Thin vs. Fat clients
- Peer-to-peer model

According to the way of accessing the information:

- pull model – the data transfer is initiated by a client
- push model – the data transfer is initiated by a server

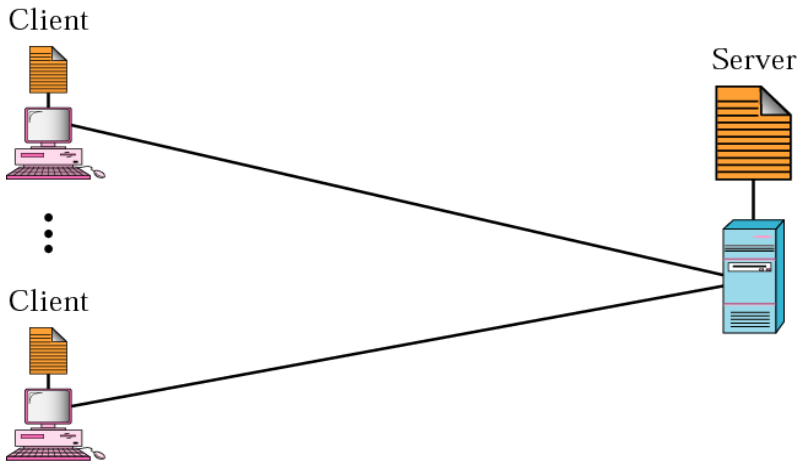
According to the demands on the computer network:

- applications with low demands on the computer network
- applications with high demands on the computer network

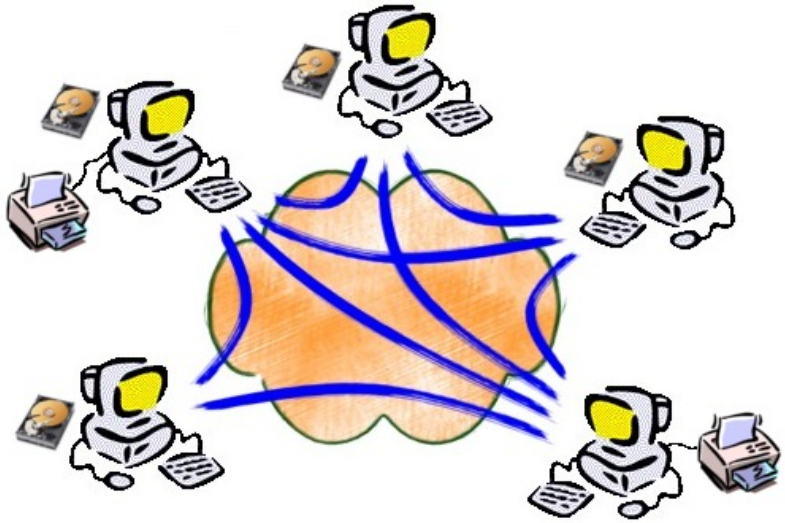
L7 – Application Layer

Client-Server vs. Peer-to-peer

Client-Server



Peer-to-peer



L7 – Application Layer

Résumé

- provides services to users
  - acts as an interface between users and computer network
- the applications can be distincted according to various criteria
  - client/server vs. peer-to-peer, pull vs. push model, demands on the computer network, etc.
- examples of Internet's fundamental applications and application protocols:
  - name service (DNS)
  - World-Wide-Web (HTTP)
  - electronic email (SMTP)
  - file transfer (FTP)
  - multimedia transmissions (RTP/RTCP)
- *further information:*
  - PB156: Computer Networks (doc. Hladká)
  - PV160: Net-centric computing II. (prof. Matyska)
  - PV188: Principles of Multimedia Processing and Transport (doc. Hladká, dr. Liška, Ing. Šiler)