# Layered Approach to Packet Networks

**Jose L. Muñoz, Juanjo Alins, Oscar Esparza, Jorge Mata**
*Telematics Engineering*
Universitat Politècnica de Catalunya (UPC)

# Outline

# Outline

# Layered model

- Packet networks are complex.
- Different abstraction levels are defined.
- “Divide and conquer”.
- Each layer performs some functions and hides the implementation details to the other layers.
- Interesting features of layered models:
  - Divide problems into easy to use parts.
  - Provide flexibility (modular design).

# A Toy Model I

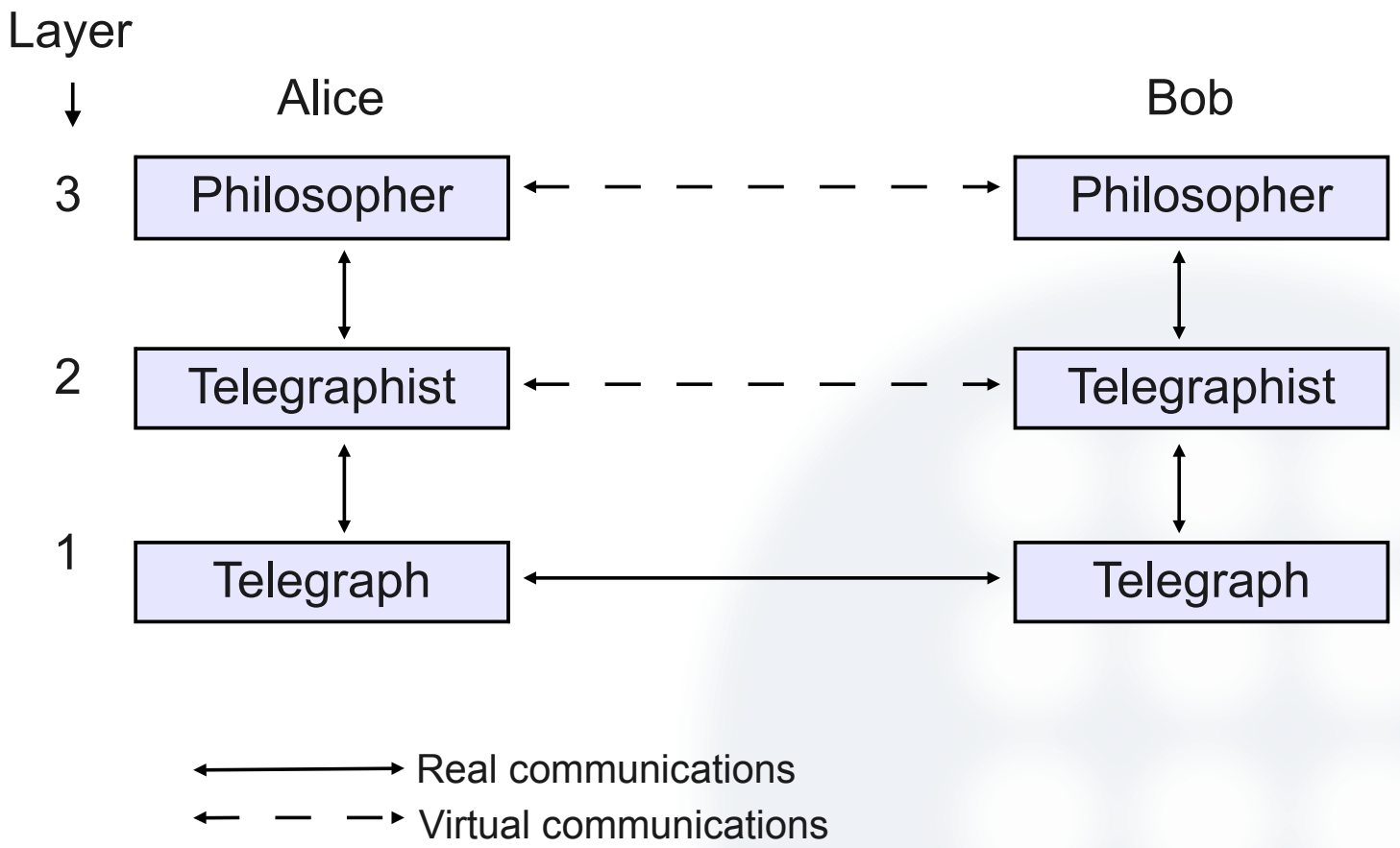
Two philosophers, Alice at USA and Bob at England want to talk about philosophy using the telegraph service.
Monolithic design:

- Alice and Bob must know about philosophy and about how build, connect and use telegraphs.

Layered design with three ”layers“:

- **Philosophers**. Alice and Bob only know about philosophy.
- **Telegraphists**. There are telegraphists that how to use telegraphs.
- **Engineers**. There are engineers that build and connect telegraphs.

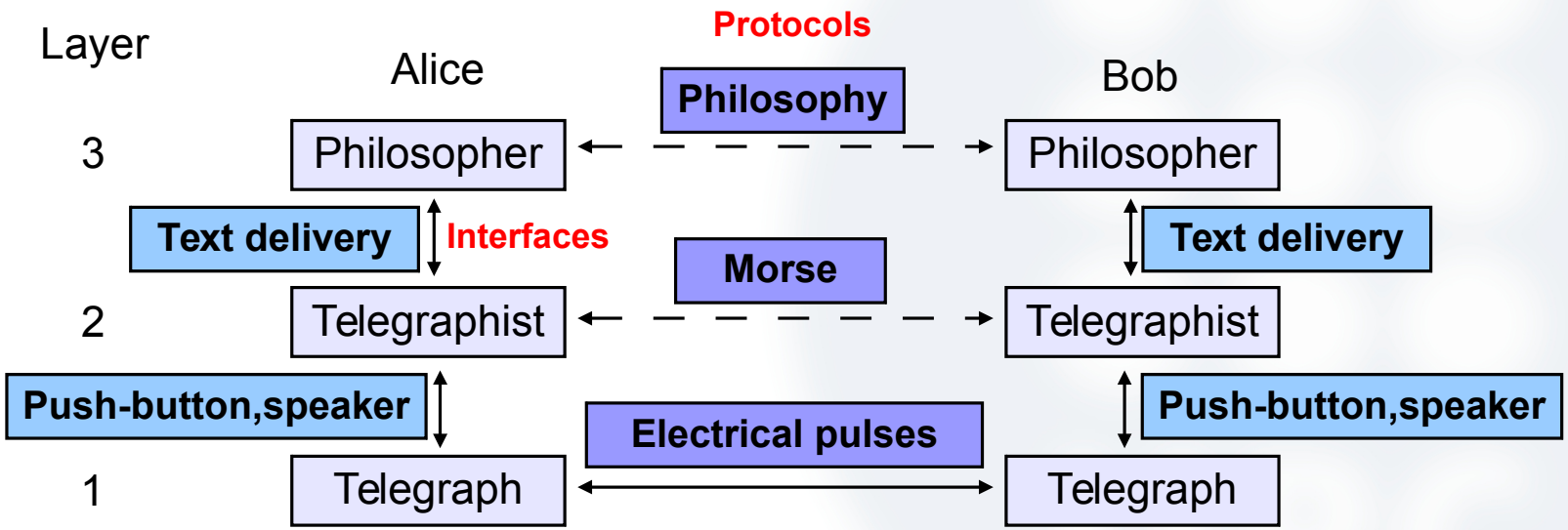
# A Toy Model II




- Layered models are based on the following principles:
  - Layer n provides services **only** to layer n+1.
  - Layer n+1 is the **user** of the services of layer n.

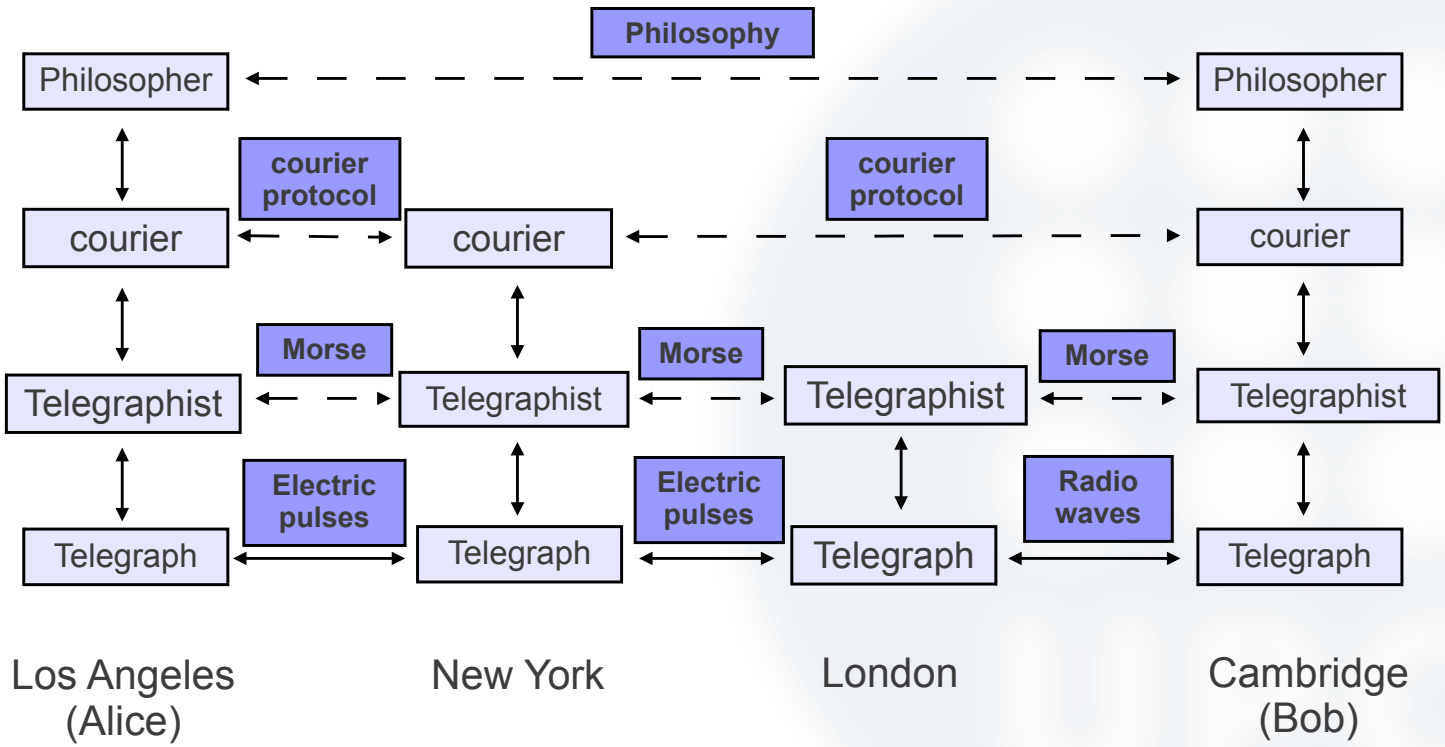
# A Toy Model III

- **Interfaces** are used to communicate layers up and down in the same stack.
- **Protocols** are used to communicate layers of the same level in different systems (n-layer protocol).
- The set of protocols is normally known as “protocol stack“.

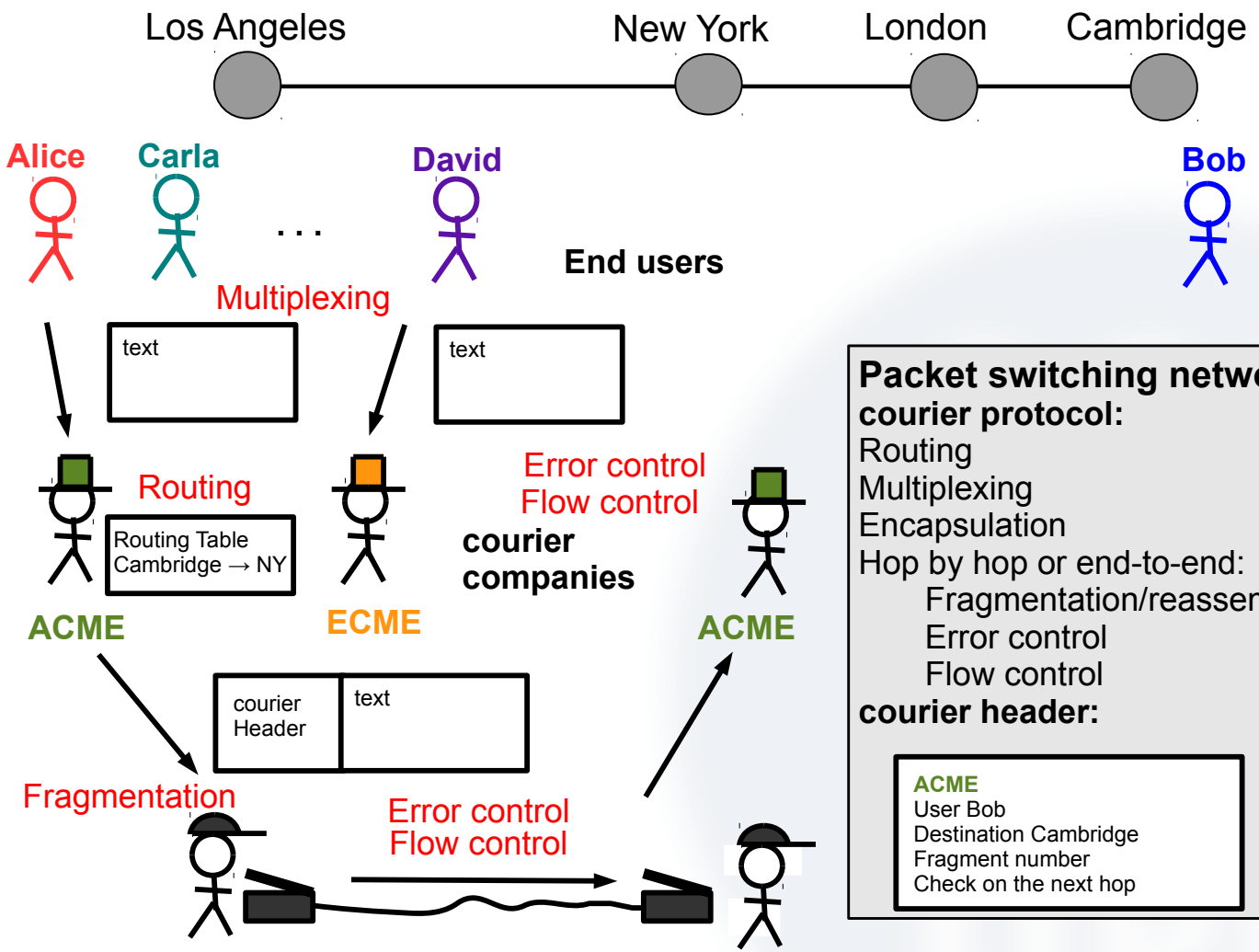


# A Toy Model IV

- Alice is at Los Angeles (USA) and Bob at Cambridge (UK) and they have not a direct telegraphic communication.
- We introduce a new layer, ”courier companies“, to manage some aspects of the communication.



# A Toy Model V

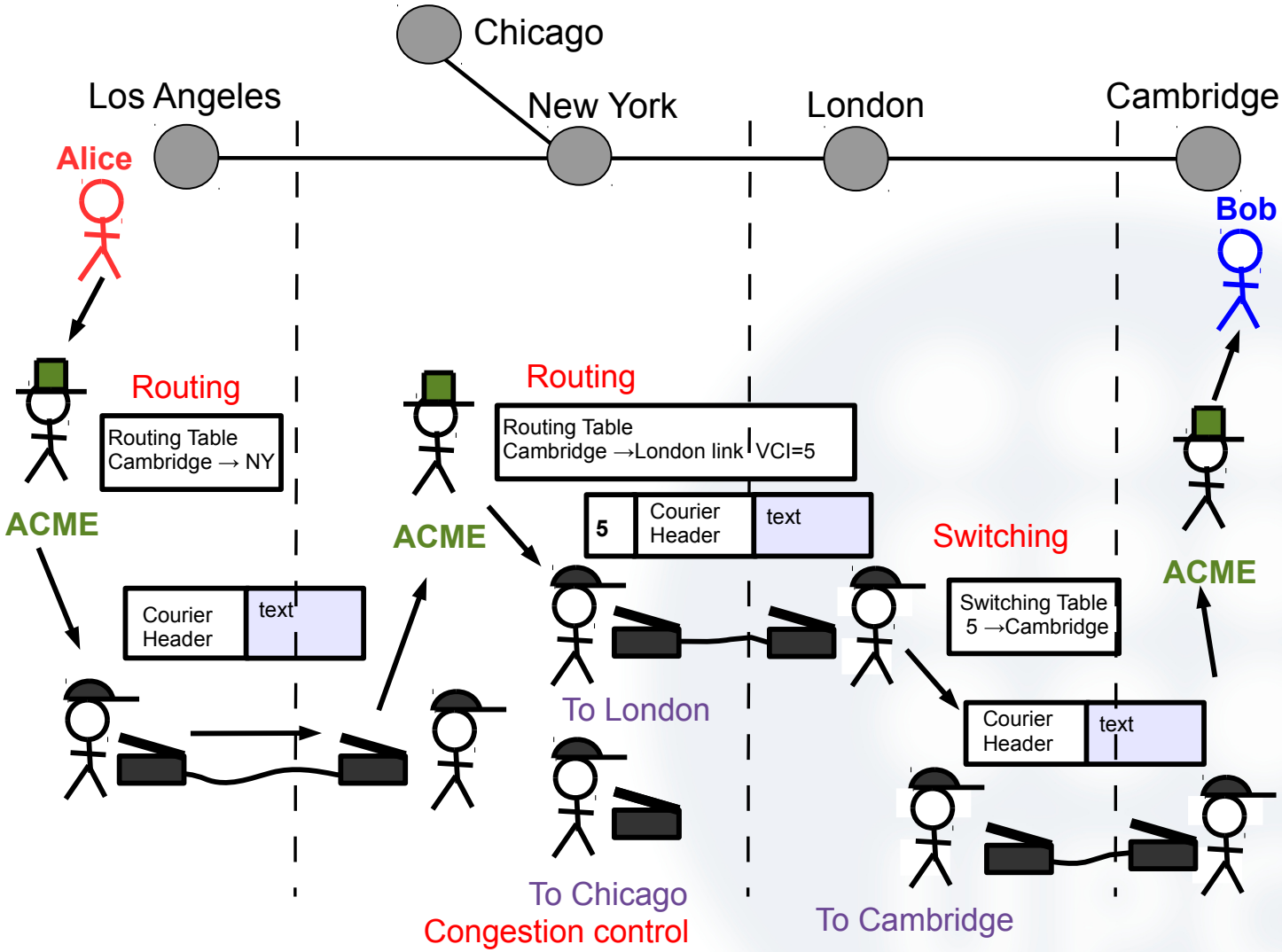



**Packet switching network**
**courier protocol:**
Routing
Multiplexing
Encapsulation
Hop by hop or end-to-end:
- Fragmentation/reassembly
- Error control
- Flow control

**courier header:**

**ACME**
User Bob
Destination Cambridge
Fragment number
Check on the next hop

# A Toy Model VI



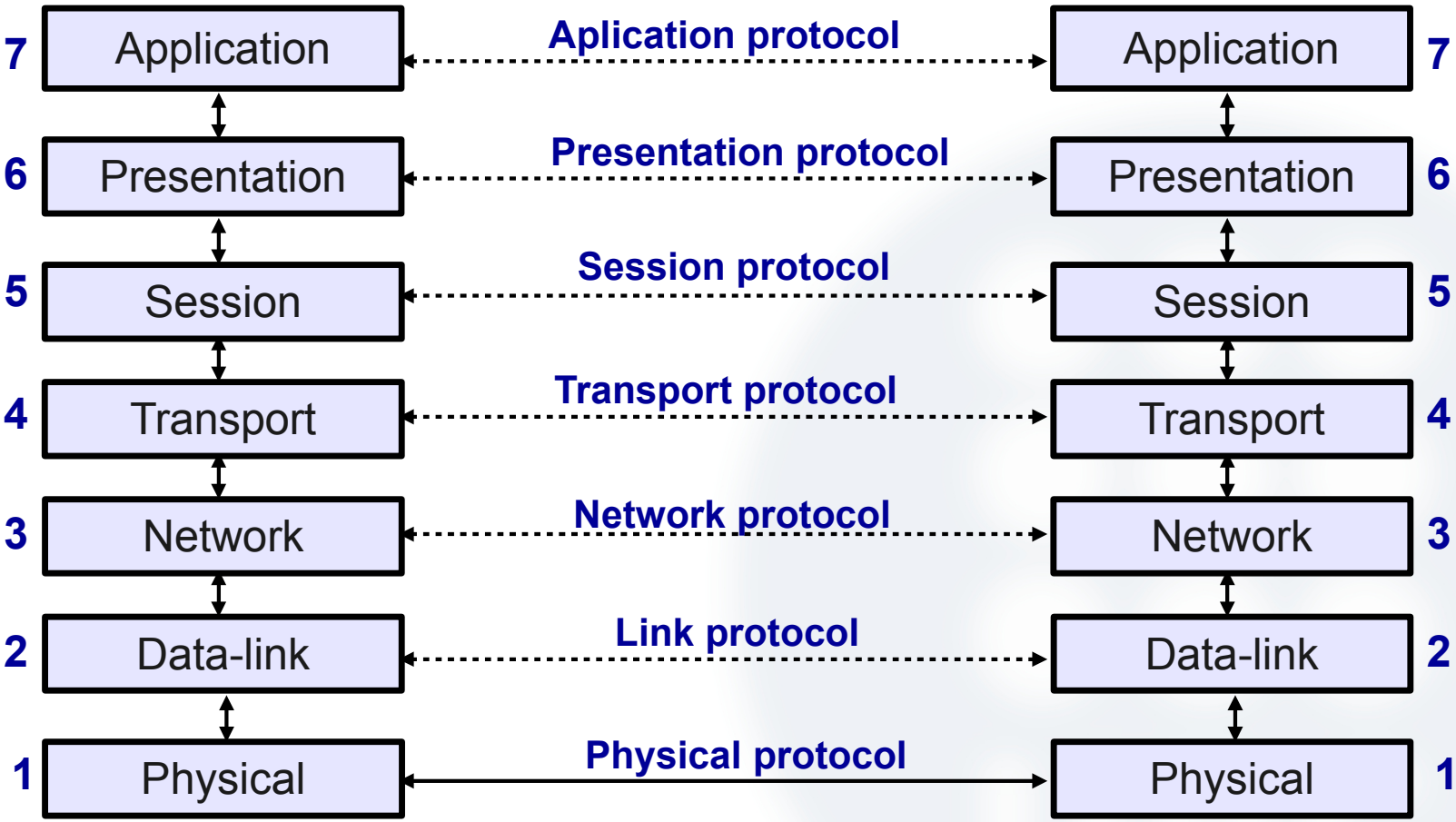
# Introduction to OSI I

- Open Systems Interconnection (OSI) is a standard layered architecture for computer communications.
- Allows direct and indirect communications.
- OSI evolution
  - 60-70s. Proprietary protocols, incompatible and heterogeneous.
  - 70s. Serious concern for standardization.
  - 1972. Another layered architecture: ARPANet Project (Internet).
  - 1984. OSI model adopted as ISO International Standard (ISO / IEC 7498-1).

# Introduction to OSI II

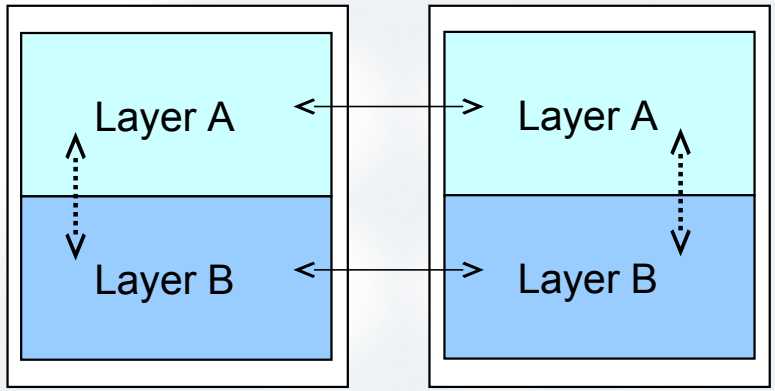
- 1994. The OSI model was adopted as an ITU standard (X.200 standard).
  - ISO. International Standards Organization.
  - ITU. International Telecommunication Union.
- OSI can be implemented but nowadays is barely used.
- The TCP/IP architecture is the most widely used in practice.
- OSI is still used as a reference model.

# OSI layers


7
Application
Aplication protocol
Application
7
6
Presentation
Presentation protocol
Presentation
6
5
Session
Session protocol
Session
5
4
Transport
Transport protocol
Transport
4
3
Network
Network protocol
Network
3
2
Data-link
Link protocol
Data-link
2
1
Physical
Physical protocol
Physical
1

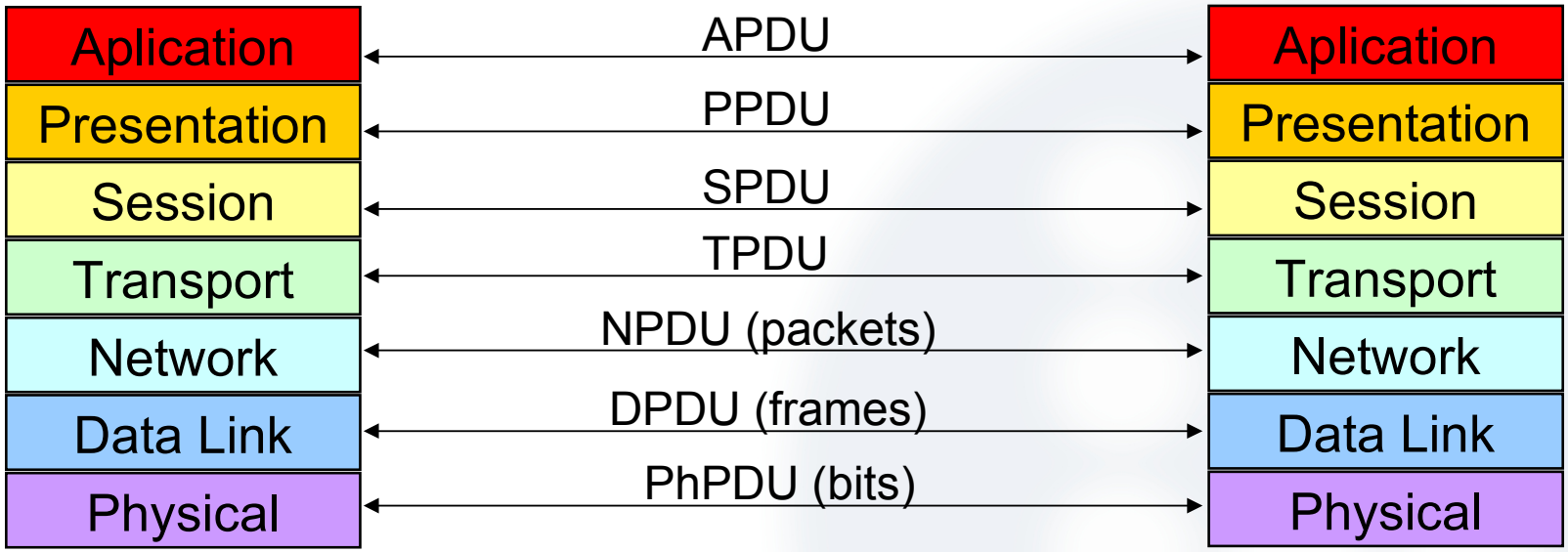
# Interfaces, services, primitives and protocols

- **Services:** layer functionalities.
- **Interfaces**: where we have to invoke services from a higher layer.
- **Primitives:** how to invoke each service on each interface.

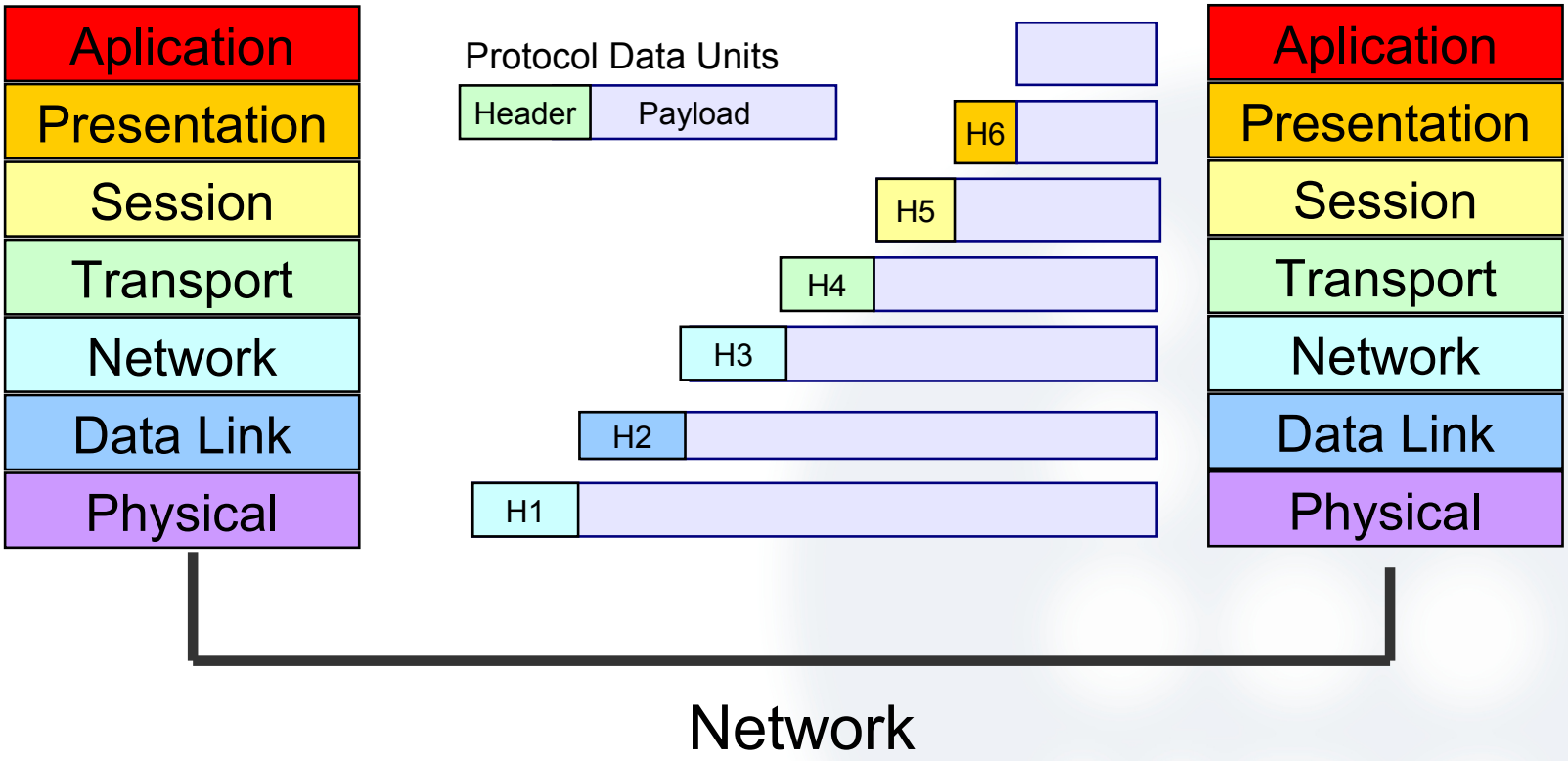



- **Protocols:** set of rules and message format (PDU) necessary to implement the services of the layer.

# Protocol Data Units



# Encapsulation

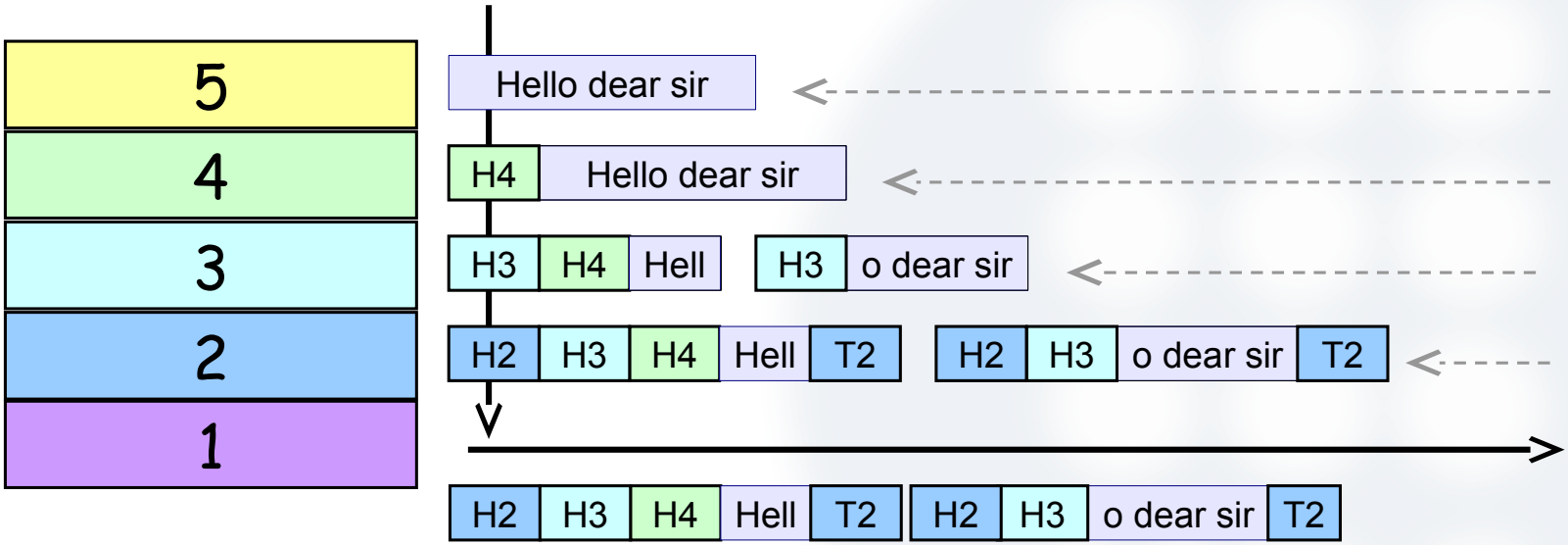


# Multiplexing

- In general, when a PDU arrives at a node:
  - There can be different protocols for each layer, so we need to identify the correct protocol for each layer.
  - The services of a layer may be used simultaneously by multiple users at the upper layer.
  - Thus, we have to identify the correct user at each upper layer.
- As a result, we can say that **we need identifiers or multiplexing keys (mux_keys) to deliver the payload to the correct user of the correct layer**.
- The mux_keys are included as part of the overhead (typically, in the header of the PDU).

# Fragmentation and Reassembling I

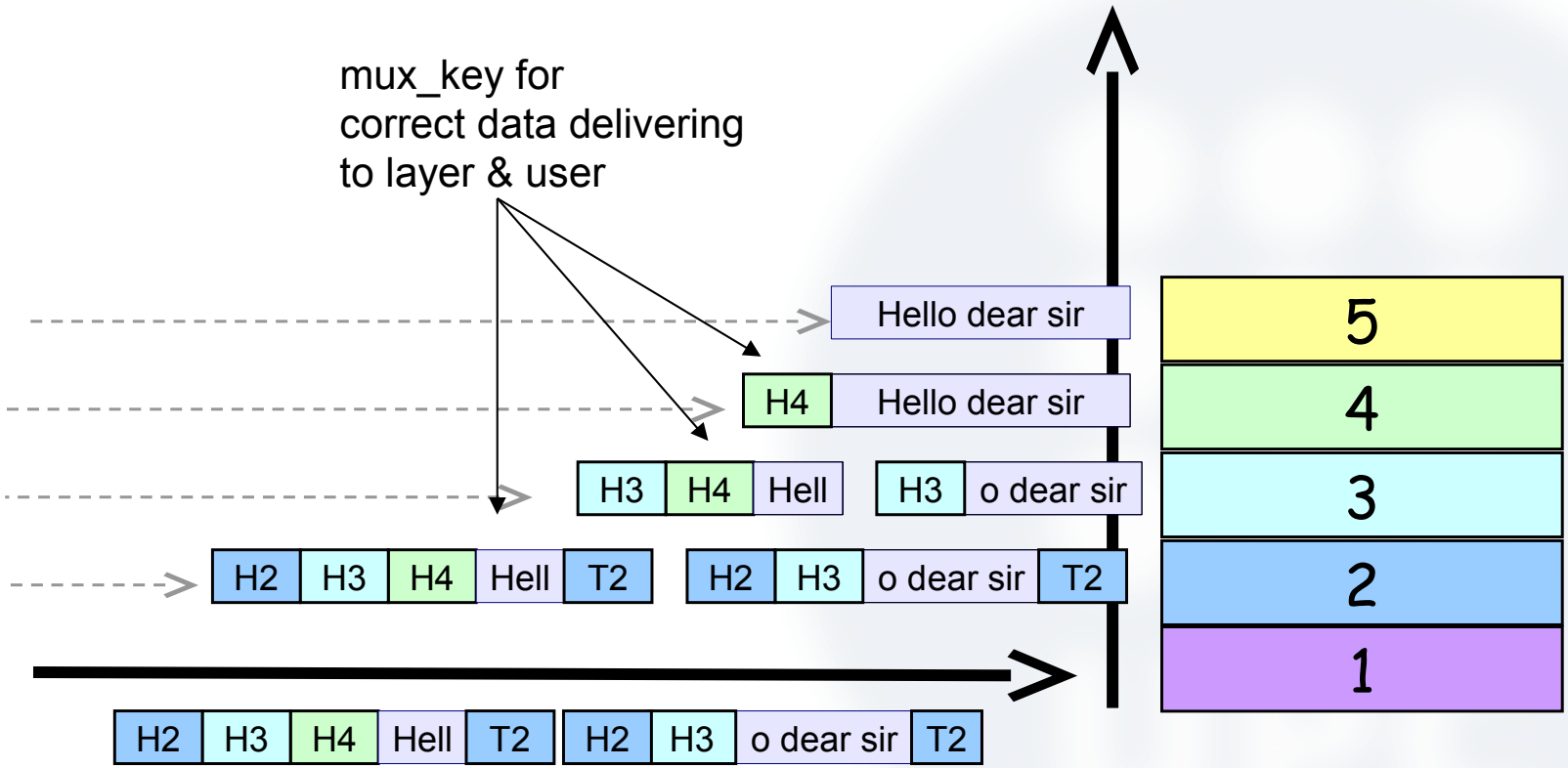
- Fragmentation is related with fitting data of an upper layer in the available payload size, breaking these data into smaller pieces if necessary:




- In the previous protocol stack fragmentation is performed by the network layer.

# Fragmentation and Reassembling II

- Reassembling:



# Fragmentation and Reassembling III

- Where to fragment:
  - **Fragmentation at the source.** The source node creates PDUs that can be transmitted over all the links of the network between the source and the destination without further fragmentation.
  - Typically, to do so, the smallest payload that can be transmitted over the network has to be discovered.
  - **Fragmentation at intermediate nodes.** Intermediate nodes need to fragment because the payload of incoming PDUs is bigger than the payload of outgoing PDUs.
- Where to reassemble:
  - **Reassembling at an intermediate node.** Some intermediate node reassembles a previously fragmented PDU.
  - **Reassemble at destination.** The destination reassembles the fragments of a PDU.
- We can use padding for fragment generation.

# Layer 1: Physical layer I

## Definition

The main objective of this layer is to define the functions that convert **bit streams** into physical events capable of being transmitted.

- The physical layer defines characteristics such as:
  - The specification of the connector and/or cables and the functions of each part of these elements. For example, use pin ”x“ to transmit and pin ”y“ to receive.
  - Properties of the electromagnetic signals used. For example, we will use two levels of electric signal -5 volts and 5 volts.
  - Sequence of events that makes it possible to exchange a bit stream. For example, to transmit a ”0“, change the polarity of the signal from -5 volts to 5 volts.

# Layer 2: Data link layer I

## Definition

The data link layer defines how data is encapsulated in **frames** to be transmitted between adjacent nodes (or nodes in the same local area network). This layer can provide also the means to detect and possibly correct errors that may occur in the physical layer.

- Data link layers are divided into two sublayers:
  - MAC sublayer (Medium Access Control):
    - Frame format.
    - Media Access Control.
  - LLC Sublayer (Logical Link Control):
    - Flow and error control between adjacent nodes.

# Layer 3: Network layer

## Definition

The network layer is responsible for **packet** forwarding including communications through intermediate nodes. This layer must allow the interconnection of heterogeneous data links and data link networks.

- Important aspects about this layer include the following:
  - Defining a network addressing plan.
  - Defining the routing capabilities. Intermediate nodes capable of routing are called ”routers“.
  - Packet fragmentation and reassembly.
  - Congestion control.

# Layer 4: Transport layer

## Definition

This layer resides on endpoints and it ensures the delivery of data between processes that run in different nodes.

- Important aspects about this layer include the following:
  - Flow segmentation and reassembly.
  - End-to-end error and flow control.
  - Provides various types of services: connection-oriented, not connection oriented, reliable, etc.

# Layer 5: Session layer

## Definition

The main purpose of this layer is to define how to initiate, coordinate and complete ”conversations“ between processes. These conversations are called sessions and can be composed of several transport connections (flows).

- Important aspects about this layer include the following:
  - The sessions offer several services such as:
    - Dialog control: who must transmit at a given time (full-duplex, half duplex, simplex).
    - Synchronization: establishing checkpoints that enable to restore a fallen communication.

# Layer 6: Presentation layer

## Definition

This layer defines data representation to ensure that the information sent by the application layer at end-point is understood by the application layer of the other endpoint.

- This layer typically includes services such as:
  - Character code translation (eg. ASCII to UTF-8).
  - Data compression.
  - Encryption/decryption (although this can be done also in other layers).

# Layer 7: Application layer

## Definition

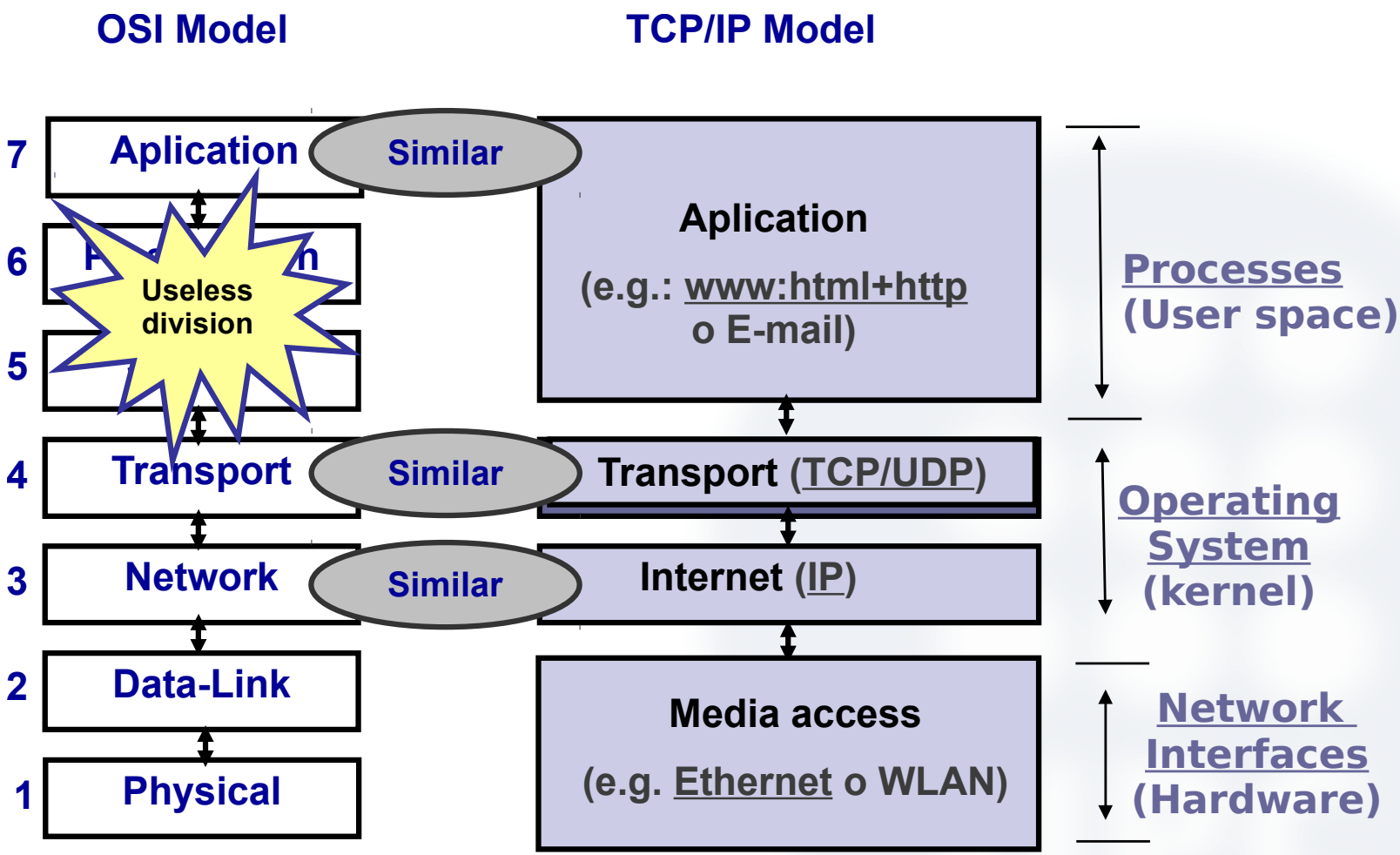
This is the last layer (it does not offer services to other layers of the OSI model). Applications use the lower layers to implement some service for the end user.

- An example of an application that uses the network is ”web browsing (www)“.

# Drawbacks of the OSI model

- The development of the OSI model took too much time.
- The architecture was rather complex to implement.
- Not very good mapping between functions and layers. For example, there are many cases of two layers with the same functionality.
- OSI was poorly accepted in USA.
- TCP/IP was successfully implemented in the UNIX operating system.

# OSI vs. TCP/IP


OSI Model
TCP/IP Model
7
Aplication
Similar
6
Useless division
5
4
Transport
Similar
3
Network
Similar
2
Data-Link
1
Physical
Aplication
(e.g.: www:html+http o E-mail)
Transport (TCP/UDP)
Internet (IP)
Media access
(e.g. Ethernet o WLAN)
Processes (User space)
Operating System (kernel)
Network Interfaces (Hardware)

# Interconnection Devices

By now, we will consider only the three lower levels:

- **L1. Repeaters** and **Hubs** interconnect physical medium to transmit **bit streams** (physical layer signals). These devices also amplify or restore the signal.
- **L2. Switches** and **bridges** manage **frames** (data link layer PDUs).
- **L3. Routers** manage network **packets** (network layer PDUs).

# Outline

# Circuits vs. packets

| Circuits | Datagrams | Virtual Circuits |
|---|---|---|
| **Deterministic multiplexing** | **Statistical multiplexing** | **Statistical multiplexing** |
| Multiplexing **channels** of constant bandwidth. We can send a data flow through the circuit. | Multiplexing **packets** with overhead, queues and processing by intermediate nodes. | Multiplexing **packets** with overhead, queues and processing by intermediate nodes (less processing per packet than datagrams) . |
| Simplified network model: **wire** | Simplified network model: set of **queues** and **CPUs**. **Layered approach**. | Simplified network model: set of **queues** and **CPUs**. **Layered approach**. |
| **Blocking** probability in circuit establishment, initial delay and signaling network required. | Packet transmission with variable delay, possible losses and disorders due to **congestion**. | Packet transmission with variable delay and possible losses due to **congestion** (rarely disorders). |
| **Phases** (circuit establishment, transfer and release). | **Without phases** (independent datagrams). | **Phases** (virtual circuit establishment, transfer and release). |
| **Signaling**, **numbering trees** and **switching tables**. | **Routing tables**. | **Signaling**, **routing tables** and **switching tables**. |
| **No routing decisions** during data transmission. | Routing decisions **per packet**. | **No routing decisions** during data transmission. |
| **Good for Real time services** with **constant bit rate**. Example: telephonic conversation. | **Good for no real time services**. Example: file transfer (e-mail, recorded voice, etc.) | **Good for no real time services**. Example: file transfer. |

# Review of Concepts

- Up to now you should be able to recognize:
  - Nodes, links and direct/indirect communications.
  - Routing and switching.
  - End-to-end versus hop-by-hop mechanisms.
  - Error control (retransmission or forward).
  - Flow control (between two nodes).
  - Congestion Control (intermediate node).
  - Layers, protocols and interfaces.
  - Protocol stack.
  - Protocol Data Units (PDUs), bits (L1), frames (L2) and packets (L3).
  - Overhead, headers and trailers.
  - Encapsulation (sending data top down).
  - Multiplexing (at different layers).
  - Fragmentation (fit user data in the available payload size).

# Outline

# Standardization I

- Proprietary protocols: each manufacturer uses its own protocols:
  - SNA (IBM)
  - DECNET (Digital)
  - AppleTalk (Apple)
  - IPX (Novell)
- Interoperability standards are needed
- They can be:
  - De facto, also sometimes called industry standards. Eg IBM PC.
  - De jure, for example. OSI protocols, X.25, ATM, A4 size paper.

# Standardization II

- Normalization or standardization is the drafting and adoption of standards that are established to ensure:
  - Using devices of different providers.
  - Quality of products.
  - Security of operation.
- Normalization is the process of development, implementation and improvement of the standards that apply to various scientific, industrial or economic activities in order to manage and enhance them.

# International standardization organizations I

- ITU (International Telecommunications Union).
- Created in 1934.
- Three sectors:
  - ITU-T: Telecommunications Sector.
    - Old CCITT: Comité Consultatif International Telegraphique et Telephonique
  - ITU-R: Radiocommunications Sector
    - Old CCIR: Comité Consultatif International des Radiocommunications
  - ITU-D: Development Sector Telecommunications
- Technical recommendations on telecommunications, rates, interfaces, etc.

# International standardization organizations II

- Formed by:
  - Government representatives (with voting rights).
  - Operators.
  - Manufacturers.
  - Scientific organizations.

# International standardization organizations III

- ISO (International Standard Organization)
  - ITU-T: International nongovernmental organization that produces industrial and commercial international standards.
  - Purpose:
    - Coordination of national standards.
    - Increase international commercial transactions.
    - Development and transference of standard technologies.
  - Standards in many areas: quality, telecommunications, measures.
  - Its members are the National Standards Bodies:
    - ANSI (USA), AENOR (Spain), DIN (Germany), etc.

# International standardization organizations IV

- IEEE (Institute of Electrical and Electronics Engineers)
  - www.ieee.org
  - Promote creativity, development and integration, sharing and applying advances in information technology, electronics and general science for the benefit of humanity and the professionals themselves.

# International standardization organizations V

- **ISOC (Internet Society)**
  - An international organization that aims to promote the Internet use and access.
  - Ensure the development, evolution and use of open Internet.
  - Individual members and corporations, organizations, governments, universities and others.
  - The **Internet Engineering Task Force (IETF)** is an organized activity of ISOC.
  - The goal of the IETF is to make the Internet work better.
  - The IETF produces standards for the Internet called **Requests For Comments (RFCs)**.

# More standardization organizations

- ANSI (American National Standards Institute):
  - Manufacturers and carrier service providers.
  - Eg FDDI.
- ETSI (European Telecommunication Standard Institute):
  - Telecommunications standardization organization in Europe.
  - Eg Hyperlan, GSM, DECT.
- CMT (Comisión del mercado de las telecomunicaciones)