MILM / MILKOVIĆ

Chapter 2: Protocols and Reference Models

Communication in computer networks is based on the exchange of messages. If the participants want to communicate with each other, they must be able to understand each other. That is why agreements are made to clarify how communication works. These agreements are called protocols, and they can be found on different levels (layers). This starts with the details of the bit transmission on the bottom layer and continues to the details at the highest layers that specify how the information is displayed.

Because of the requirements that computer networks need to satisfy, communication is subdivided into layers within *reference models*. Each *layer* deals with a particular aspect of communication and provides *interfaces* to the layer above and the layer below. Each interface consists of a set of *operations*, which together define a *service*. In the layers, the data are encapsulated (see *data encapsulation* in section 2.4). Because each layer is self-contained, individual protocols can be modified or replaced without affecting all aspects of communication. This is possible as long as there is no modification to the interface and the behavior for the next upper layer. The three most popular reference models are OSI reference model, TCP/IP reference model and Hybrid reference model.

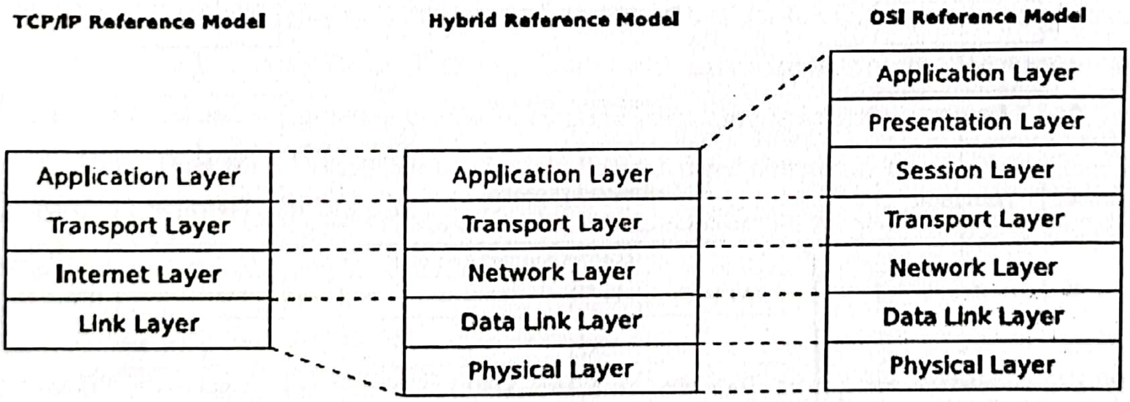

Fig 2.1: Reference Models

2.1 The OSI Reference Model

The OSI model is based on a proposal developed by the International Standards Organization (ISO) as a first step toward international standardization of the protocols used in the various layers.

The model is called the ISO-OSI (Open Systems Interconnection) Reference Model because it deals with connecting open systems—that is, systems that are open for communication with other systems.

The OSI model has seven layers (**Fig. 2.2**). The principles that were applied to arrive at the seven layers can be briefly summarized as follows:

1. A layer should be created where a different abstraction is needed.
2. Each layer should perform a well-defined function.
3. The function of each layer should be chosen with an eye toward defining internationally standardized protocols.
4. The layer boundaries should be chosen to minimize the information flow across the interfaces.
5. The number of layers should be large enough that distinct functions need not be thrown together in the same layer out of necessity and small enough that the architecture does not become unwieldy.

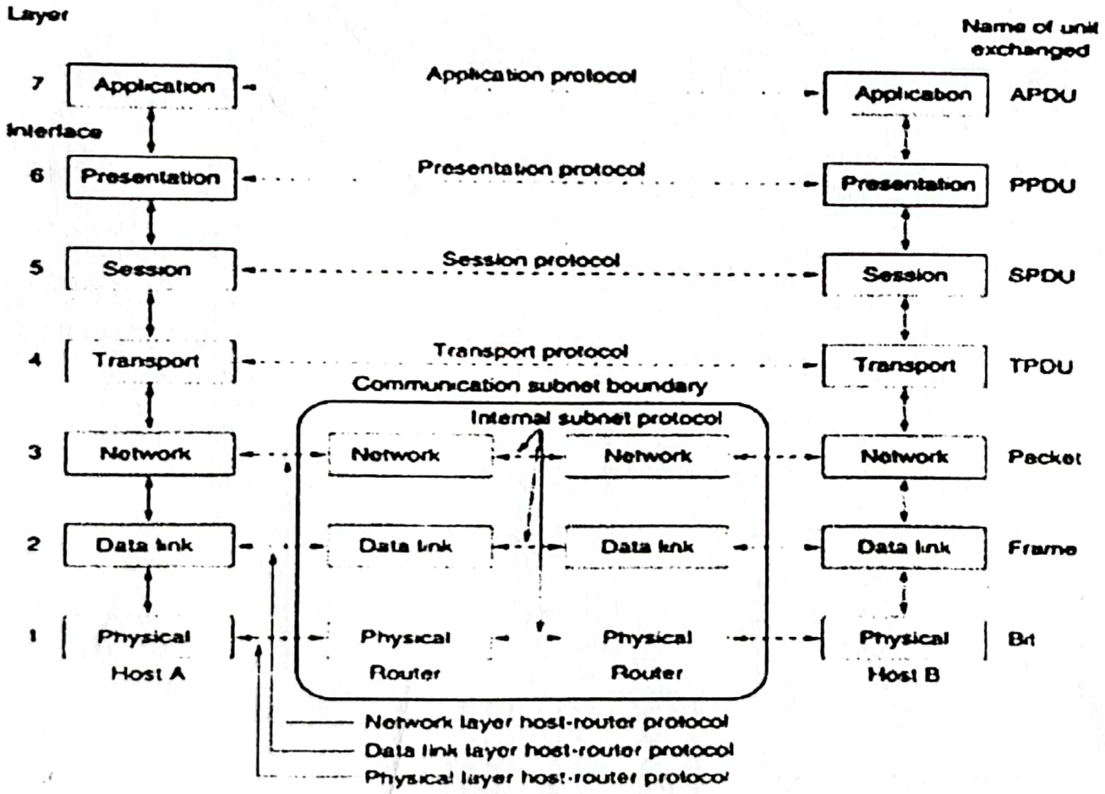

Fig. 2.2: OSI reference Model

2.1.1 The Physical Layer

The Physical Layer deals with the characteristics of the various transmission media. This layer is responsible for the transfer of raw bits (ones and zeros). Here, the physical connection to the medium and the conversion of the data into signals takes place. Protocols of the physical layer define how many bits can be sent per second and whether the transmission can take place simultaneously in both directions.

2.1.2 The Data Link Layer

The main task of the data link layer is to transform a raw transmission facility into a line that appears free of undetected transmission errors to the network layer. If errors occur during the transmission of the bit sequences – this often occurs in practice – a procedure is necessary to detect these errors. This is one of the tasks of the Data Link Layer. Also, the protocols of the Data Link Layer control the access to the transmission medium (for example via CSMA/CD or CSMA/CA).

At the sender, the Data Link Layer transforms the Network Layer packets into *frames* and transmits them with the desired reliability via a physical network from one network device to another. At the receiver, the Data Link Layer identifies the frames in the bit stream of the Physical Layer. The delivery of frames on a physical network requires physical addresses (MAC addresses), whose format defines the Data Link Layer. For error detection, the Data Link Layer protocols attach a checksum to every frame. This way, frames with errors can be detected and thrown away by the receiver. The feature to request discarded frames is not implemented in the Data Link Layer.

In the Data Link Layer, frames can only be exchanged between network devices that are connected to the same physical network. The connection of different physical networks is done by using Bridges and Switches (Multiport Bridges).

2.1.3 The Network Layer

The network layer controls the operation of the subnet. A key design issue is determining how packets are routed from source to destination. Network layer also called Internet layer for this internetworking task requires IP addresses.

At the sender, the Network Layer packs the segments of the Transport Layer into packets. At the receiver, the Network Layer detects the frames of the Data Link Layer and transform into packets.

Routers (which are equipment of network layer) limit logical subnets. Forwarding packets on their way from the sender to the destination, which is called routing. Usually, the connectionless Internet Protocol (IP) is used. Each IP packet is independently *routed* to its destination, and the path is not recorded. This layer is also in charge of congestion control in case too many packets on the network. More generally, the quality of service provided (delay, transit time, jitter, etc.) is also a network layer issue.

2.1.4 The Transport Layer

The *Transport Layer* enables the transport of *Segments* between processes on different devices via so-called end-to-end protocols.

At the sender, the Transport Layer splits up the data from the above Layer into segments. At the receiver, the Transport Layer detects and forms the segments from the packets provided by the Network Layer.

While the Data Link Layer address network devices in a physical way, in this layer the running processes are addressed with port numbers. The Transport Layer, therefore, ensures that the data in the Network Layer is delivered correctly.

Transport Layer protocols implement different forms of communication. *Connectionless communication* works analogously to a mailbox. The sender sends messages without prior connection establishment. With connectionless communication, the Transport Layer does not provide a way to validate that a segment arrives at the destination. If validation is required, it must be carried out in the above Layer. The lack of a delivery guarantee is one disadvantage of this form of communication. One benefit is the better data rate because less overhead arises.

One alternative is using *connection-oriented communication*. This works in the same way as the telephone. A connection is established between sender and receiver before data is exchanged. The connection remains active even if no data is transferred. As soon as all data has been exchanged, one of the communication partners terminates the connection.

Connection-oriented communication enables flow control and congestion control, in which the receiver controls the transmission speed of the sender. Depending on the Transport Layer protocol used, the Transport Layer also ensures loss-free delivery of the segments. This is equivalent to a delivery guarantee. The correct order of the segments at the receiver is also guaranteed by this form of communication.

Examples of Transport Layer protocols are the connectionless *User Datagram Protocol* (UDP) and the connection-oriented *Transport Control Protocol* (TCP).

2.1.5 The Session Layer

The tasks of the Session Layer include the establishment, monitoring, and termination of sessions. A session is the basis for a virtual connection between two applications on physically independent computers. Also, the Session Layer is intended to control the dialogues, which means that it determines which communication partner is allowed to send data next. Another task is to provide checkpoints that can be used in longer data transfers for synchronization. If the connection fails, the return to a checkpoint avoids having to start the transmission over again from the beginning.

Protocols that meet the required capabilities of the Session Layer, are among others, Telnet for the remote control of computers and FTP for file transfer. However, these protocols can also be assigned to the Application Layer. The Application Layer contains the protocols that are used by the application programs. Because FTP and Telnet are used directly by the application programs and not by abstract protocols of upper layers, it makes more sense to assign these protocols of the Session Layer to the Application Layer.

2.1.6 Presentation Layer

The Presentation Layer contains rules for the formatting (presentation) of messages. This layer is concerned with the syntax and semantics of the information transmitted. In order to make it possible for computers with different data representations to communicate, the data structures to be exchanged can be defined in an abstract way, along with a standard encoding to be used "on the wire."

2.2 TCP/IP Reference Model

The TCP/IP reference model was developed prior to OSI model. The major design goals of this model were,

1. To connect multiple networks together so that they appear as a single network.
2. To survive after partial subnet hardware failures.
3. To provide a flexible architecture.

Unlike OSI reference model, TCP/IP reference model has only 4 layers. They are,

1. Host-to-Network Layer also called Link Layer
2. Internet Layer

3. Transport Layer
4. Application Layer

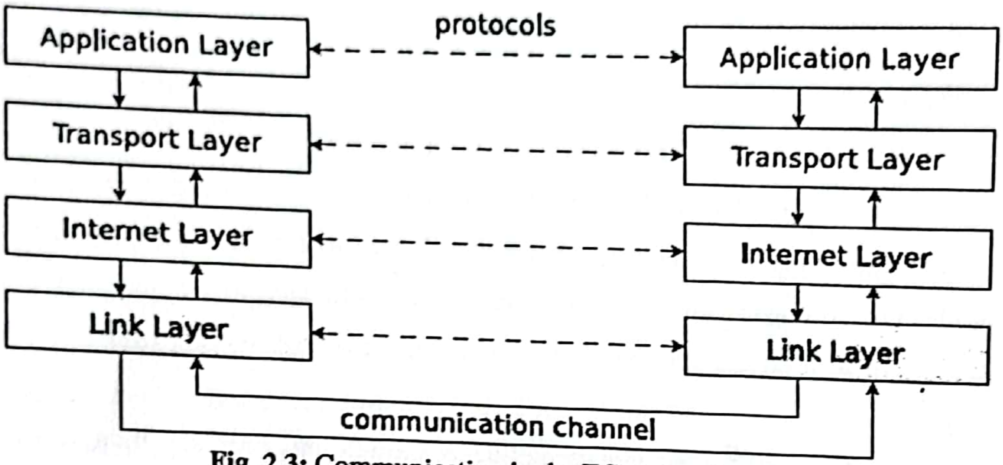

Fig. 2.3: Communication in the TCP/IP Reference Model

This reference model was developed in the context of the Arpanet and is considered as the basis of the Internet. For each layer, the functionality is specified. Communication protocols must implement these requirements. The concrete implementation is not specified and can be different. Therefore, for each one of the four layers, multiple protocols do exist.

Table 2.1: Layers in the TCP/IP Reference Model

Layer	Name	Protocols (examples)
4	Application Layer	HTTP, FTP, SMTP, POP3, DNS, SSH, Telnet
3	Transport Layer	TCP, UDP
2	Internet Layer	IP (IPv4, IPv6), ICMP, IPsec, IPX
1	Link Layer	Ethernet, WLAN, ATM, FDDI, PPP, Token Ring

Each layer adds additional *header* information to the message (see Figure 2.4). Some protocols (e.g., Ethernet) add not only a header in the Link Layer but also a *trailer* at the end of the message. The receiver analyzes the header (and trailer) on the same layer.

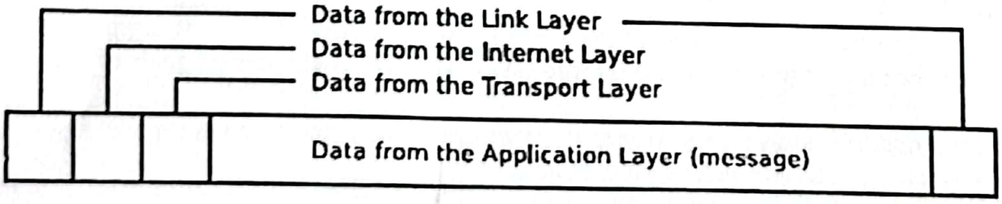
Data from the Link Layer
Data from the Internet Layer
Data from the Transport Layer
Data from the Application Layer (message)

Fig. 2.4: Each Layer adds additional Data to the Message

The tasks of the individual layers are discussed using the OSI reference model in the previous section.

2.3 Hybrid Reference Model

The names of the top two layers and the tasks of the top three layers are identical to the layers of the TCP/IP reference model (see Figure 2.5). The Internet Layer in the TCP/IP reference model and the Network Layer in the hybrid reference model differ only in the name.

The tasks of the individual layers are discussed using the OSI reference model in the previous section.

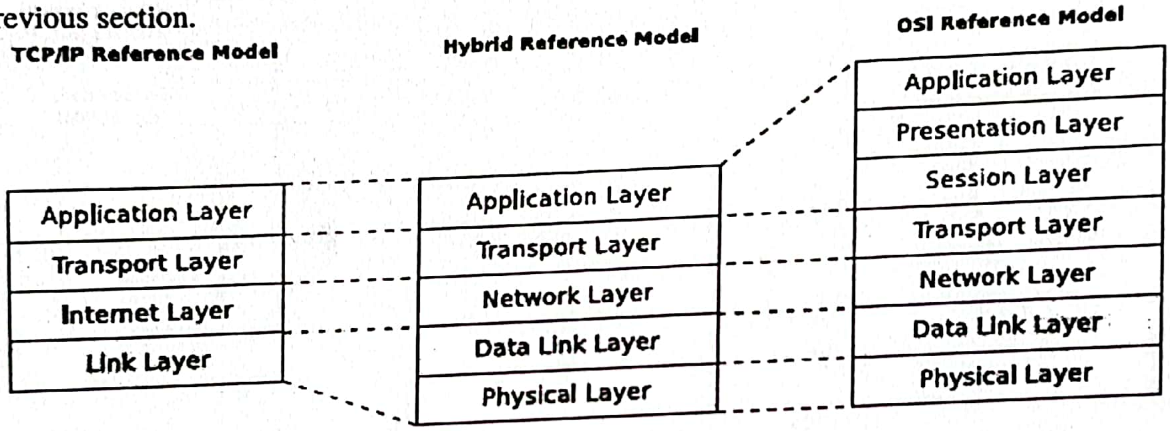

Fig.2.5: Comparison of the Reference Models

2.4 How Communication works

The *communication flow* is demonstrated by using the hybrid reference model (see Figure 2.6). *Vertical communication* describes the process in which the data passes through the layers of the reference model used. A message is packed layer by layer from the top layer to the bottom layer and extracted by the receiver in the reverse order from the bottom layer to the top layer. The sender adds a header to the data in each layer and a trailer in the Data Link Layer. These additional headers and trailers are analyzed and removed in the respective layers by the receiver. These operations are called *encapsulation* and *de-encapsulation.*

In *horizontal communication*, sender and receiver each use the same protocol functions on the same layers.

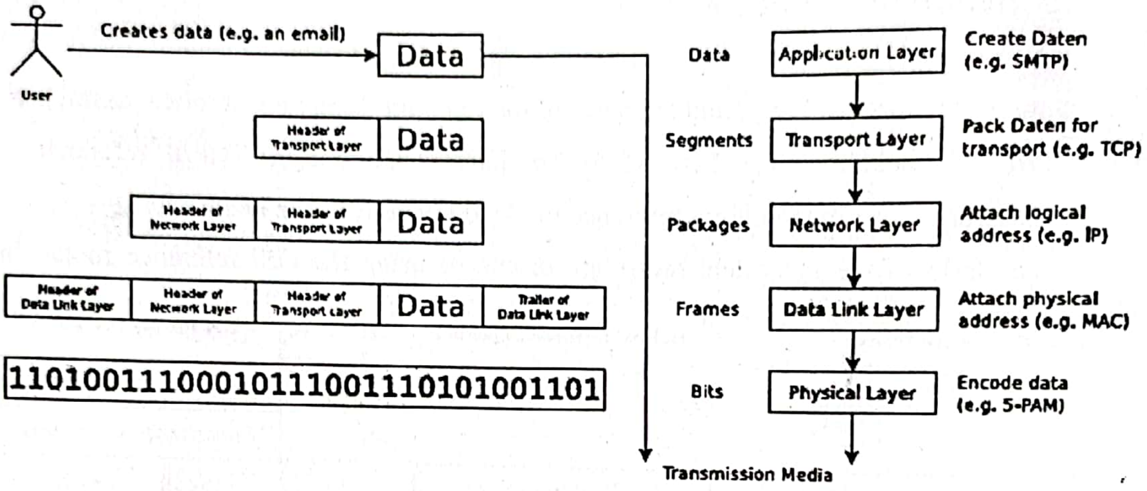

Fig. 2.6: Data Encapsulation in vertical Communication

2.5 Conclusion on the Reference Models

The TCP/IP reference model is considered the basis of the Internet. It specifies for each layer which functionalities need to be implemented by it, but not how this is done.

The OSI reference model is very similar to the TCP/IP reference model. The models only differ in two aspects. In the OSI reference model, the tasks of the Network Layer are divided between the Physical Layer and the Data Link Layer. The tasks of the Application Layer in the OSI reference model are divided between the Session Layer, Presentation Layer, and Application Layer.

The hybrid reference model is helpful because the TCP/IP reference model does not distinguish between the Physical Layer and the Data Link Layer, while their task areas are entirely different. However, subdividing the Application Layer into three layers has not proven to be meaningful and does not take place in practice, because the functionalities, which are intended for the Session Layer and the Presentation Layer, are implemented nowadays by the Application Layer protocols.

The hybrid reference model illustrates the functioning of computer networks in a realistic way because it distinguishes the Physical Layer and the Data Link Layer and at the same time does not subdivide the Application Layer. It combines the benefits of the TCP/IP reference model and the OSI reference model without taking over their drawbacks.