



Data Communications and Networking

Fourth Edition

Forouzan

Chapter 2

Network Models

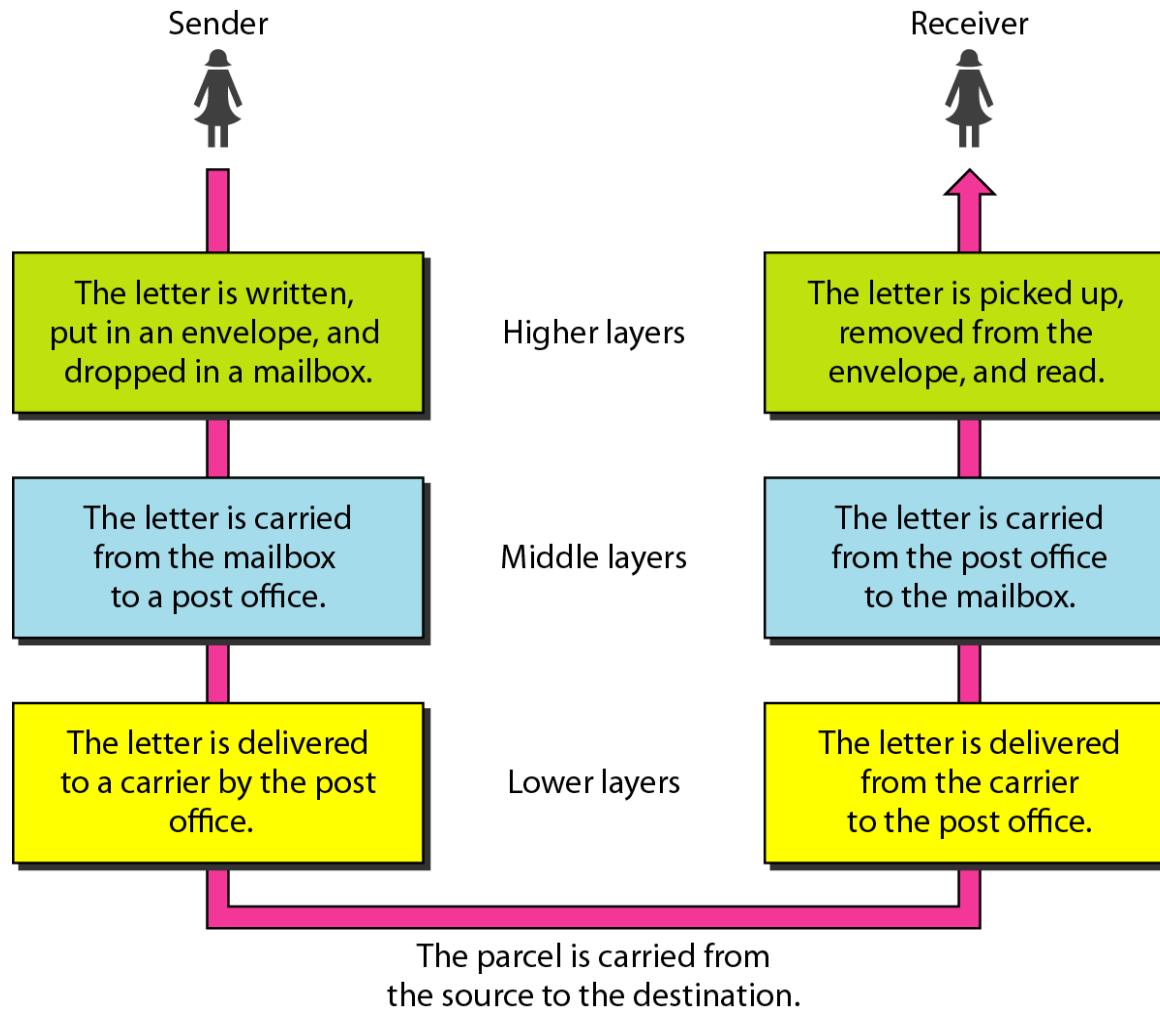
2-1 LAYERED TASKS

*We use the concept of **layers** in our daily life. As an example, let us consider two friends who communicate through postal mail. The process of sending a letter to a friend would be complex if there were no services available from the post office.*

Topics discussed in this section:

**Sender, Receiver, and Carrier
Hierarchy**

Figure 2.1 Tasks involved in sending a letter



2-2 THE OSI MODEL

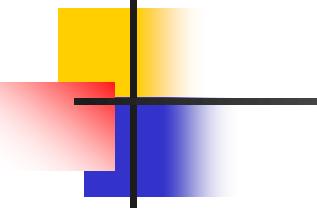
*Established in 1947, the International Standards Organization (**ISO**) is a multinational body dedicated to worldwide agreement on international standards. An ISO standard that covers all aspects of network communications is the Open Systems Interconnection (**OSI**) model. It was first introduced in the late 1970s.*

Topics discussed in this section:

Layered Architecture

Peer-to-Peer Processes

Encapsulation



Note

**ISO is the organization.
OSI is the model.**

Figure 2.2 Seven layers of the OSI model

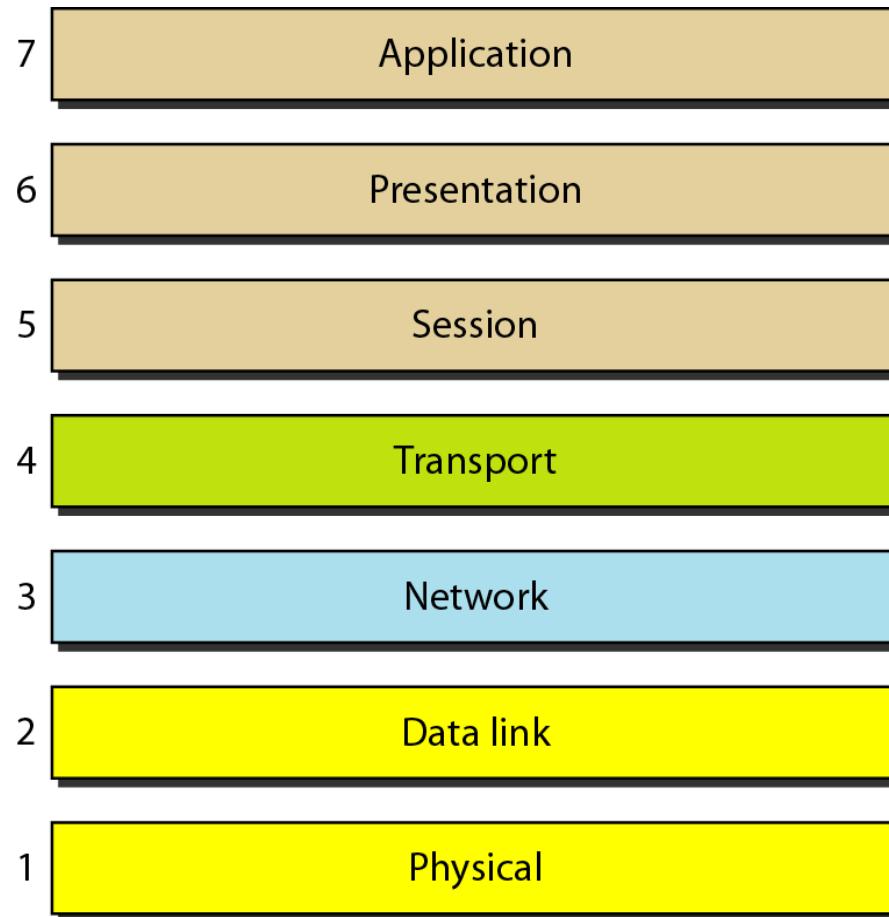


Figure 2.3 *The interaction between layers in the OSI model*

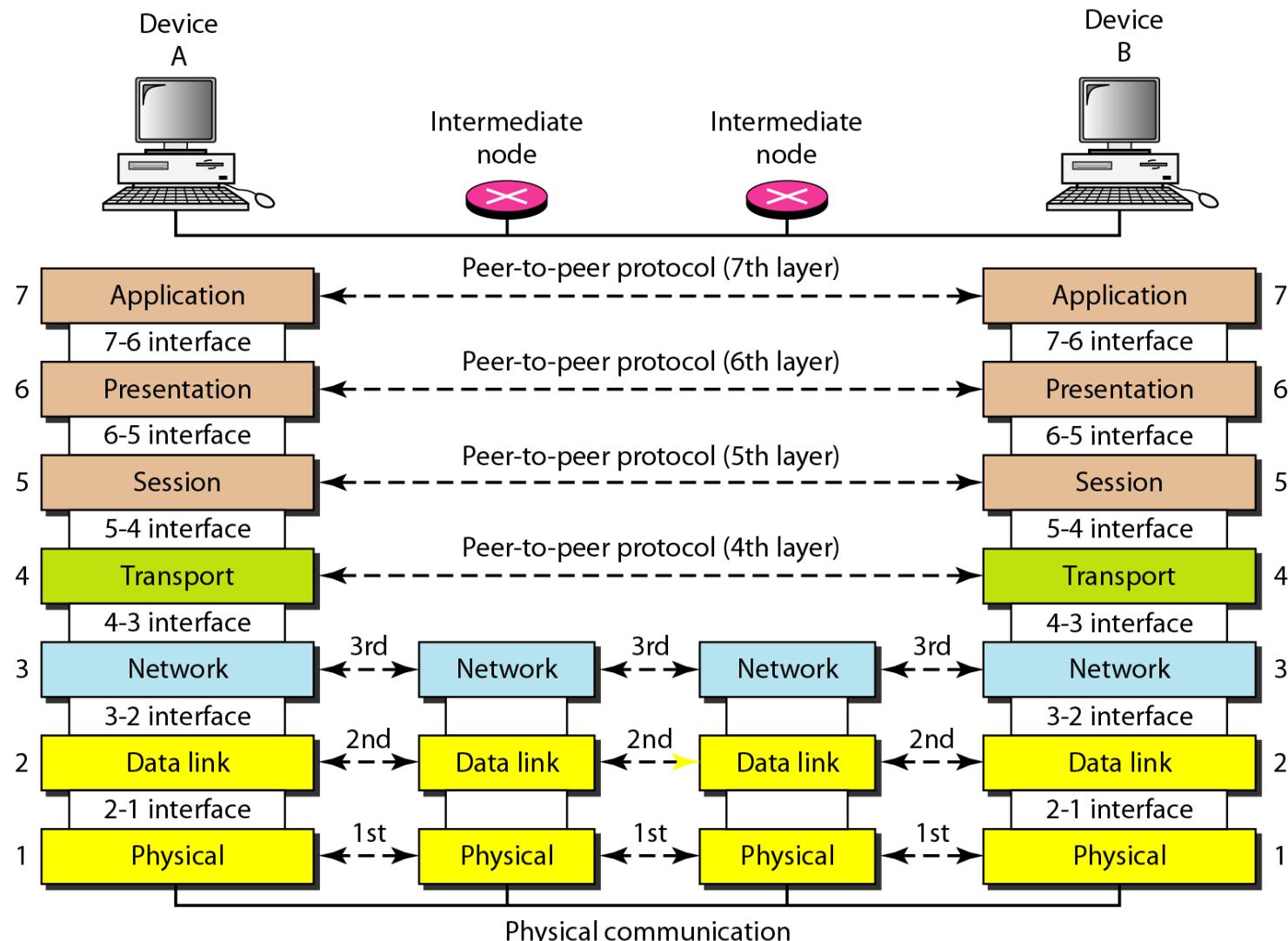
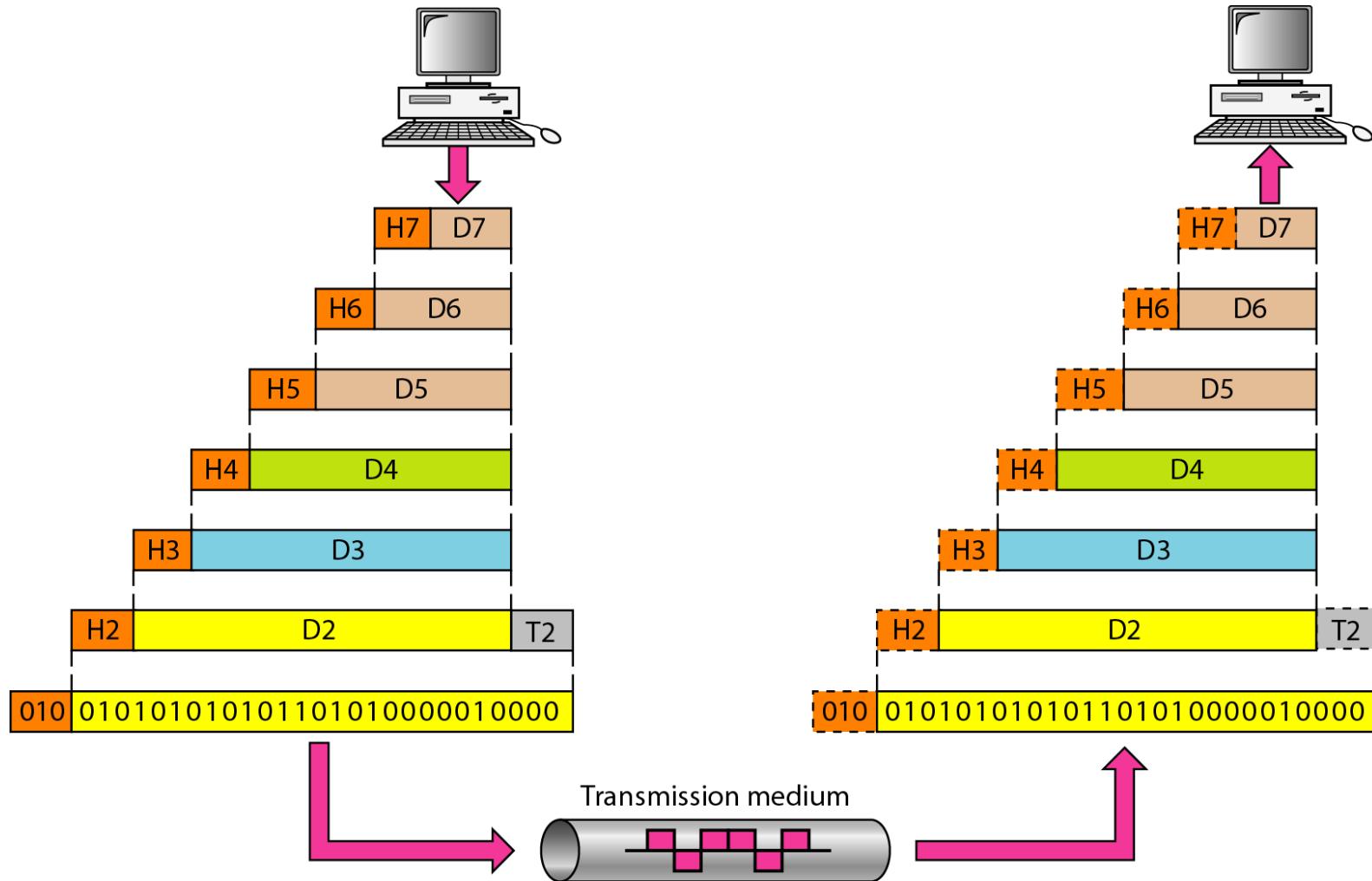


Figure 2.4 An exchange using the OSI model



2-3 LAYERS IN THE OSI MODEL

In this section we briefly describe the functions of each layer in the OSI model.

Topics discussed in this section:

Physical Layer

Data Link Layer

Network Layer

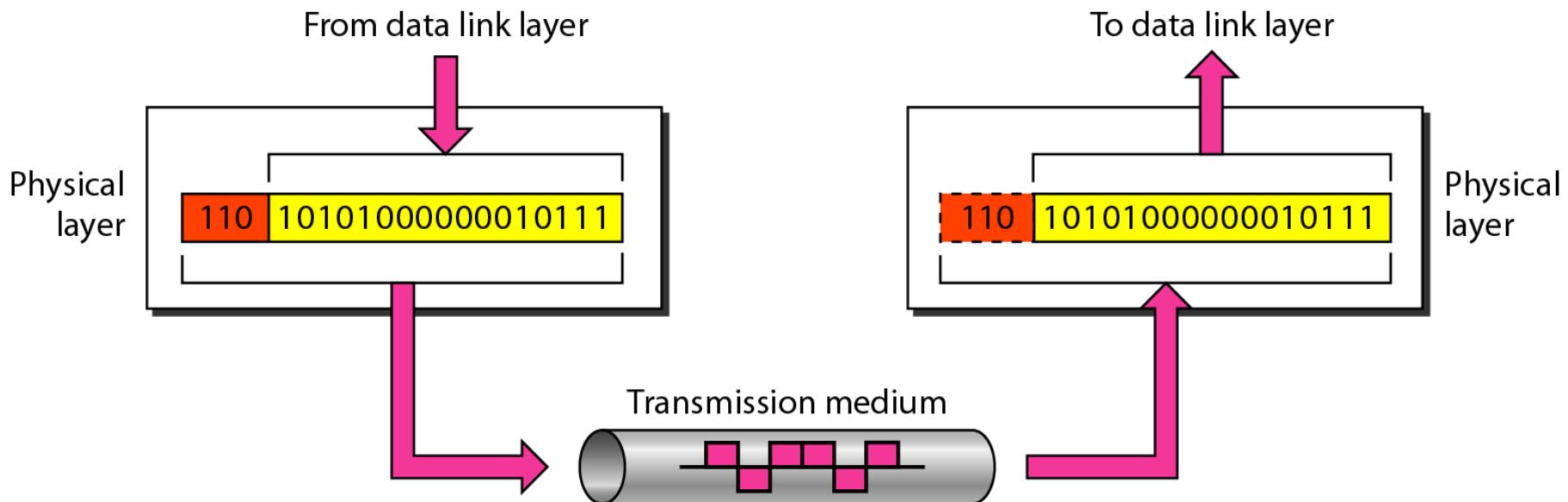
Transport Layer

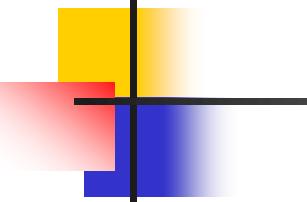
Session Layer

Presentation Layer

Application Layer

Figure 2.5 Physical layer

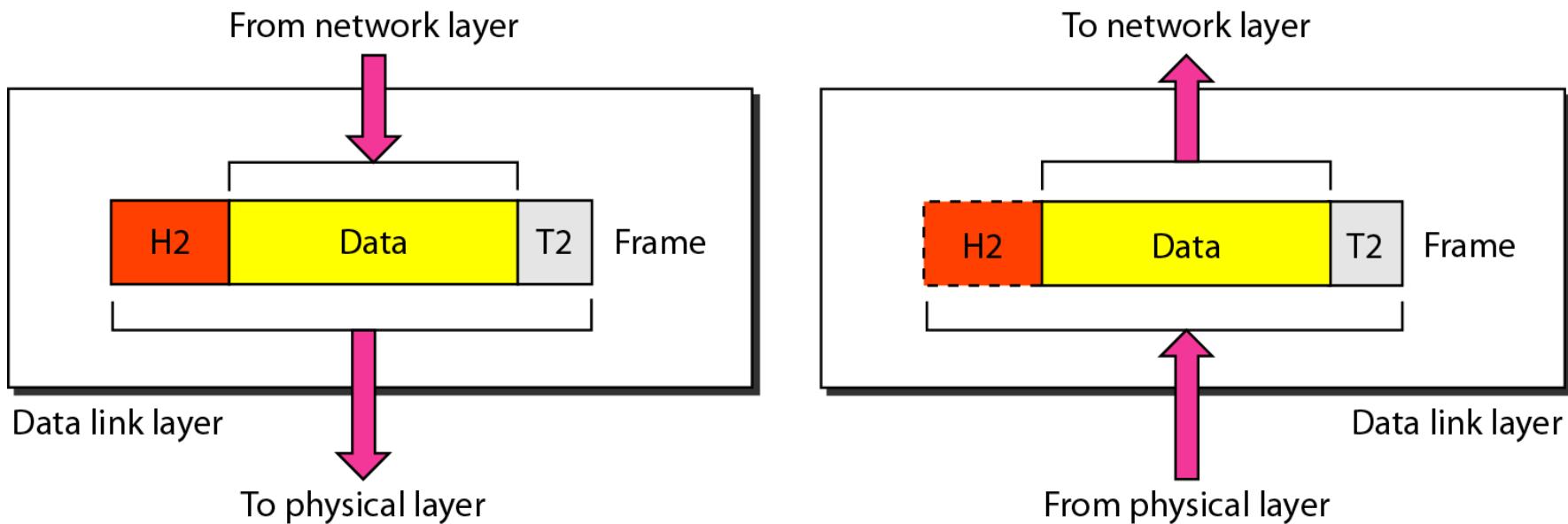


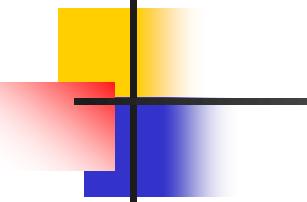


Note

The physical layer is responsible for movements of individual bits from one hop (node) to the next.

Figure 2.6 Data link layer





Note

The data link layer is responsible for moving frames from one hop (node) to the next.

Figure 2.7 Hop-to-hop delivery

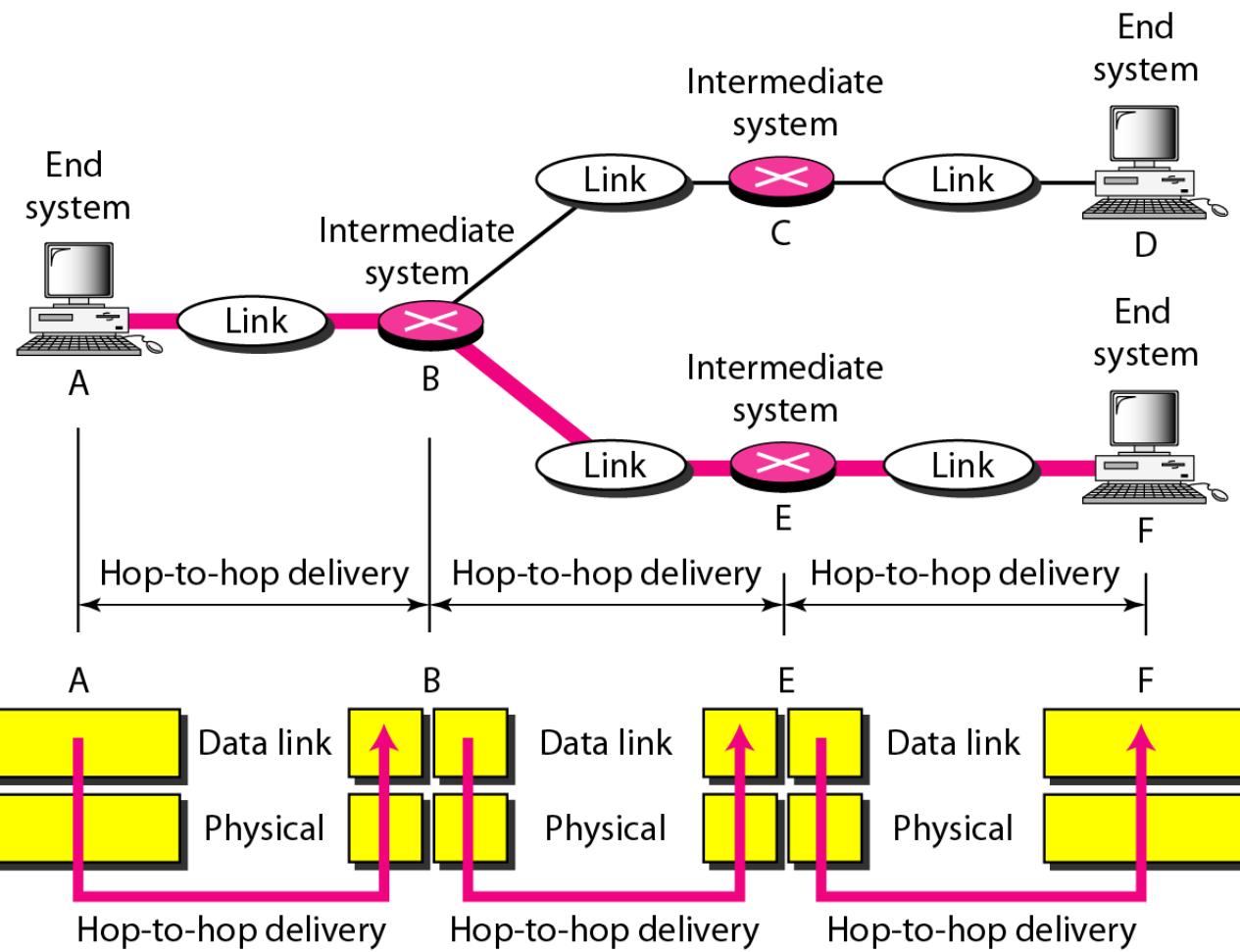
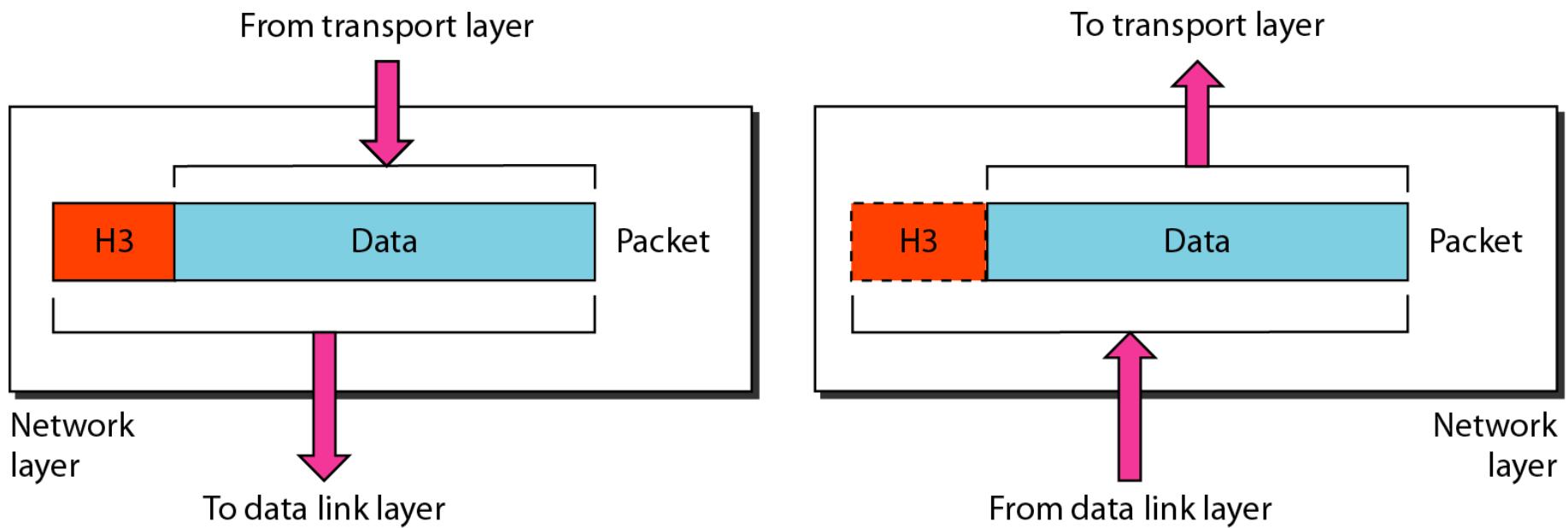
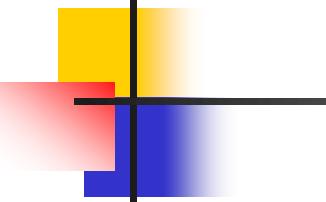


Figure 2.8 Network layer





Note

The network layer is responsible for the delivery of individual packets from the source host to the destination host.

Figure 2.9 Source-to-destination delivery

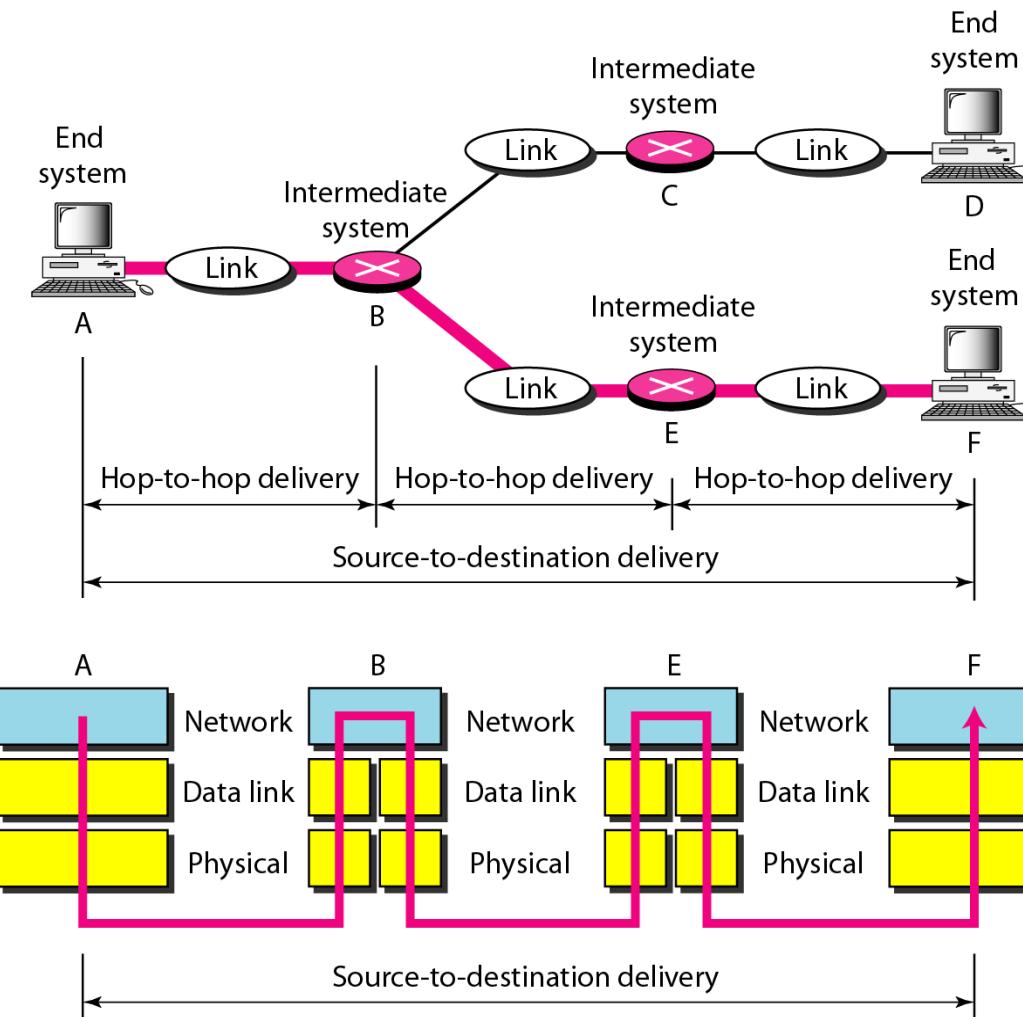
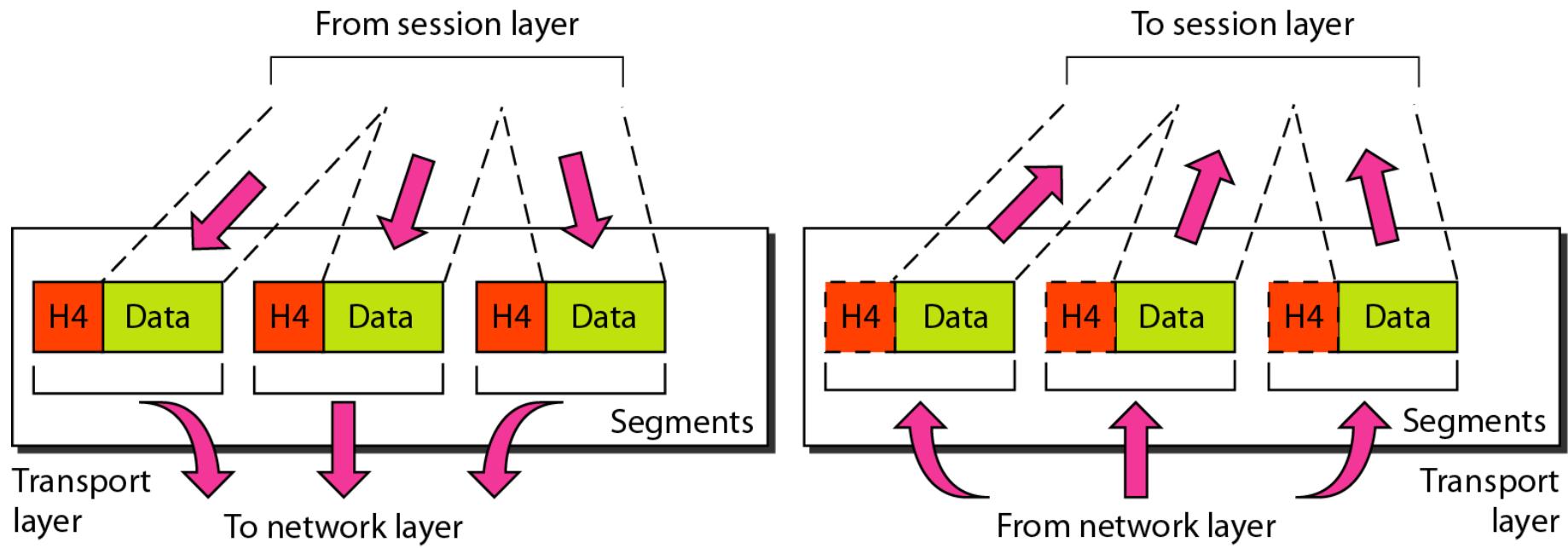
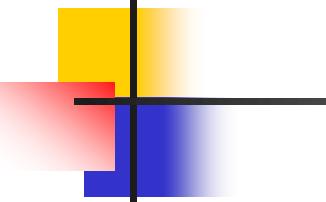


Figure 2.10 Transport layer





Note

**The transport layer is responsible for the delivery
of a message from one process to another.**

Figure 2.11 *Reliable process-to-process delivery of a message*

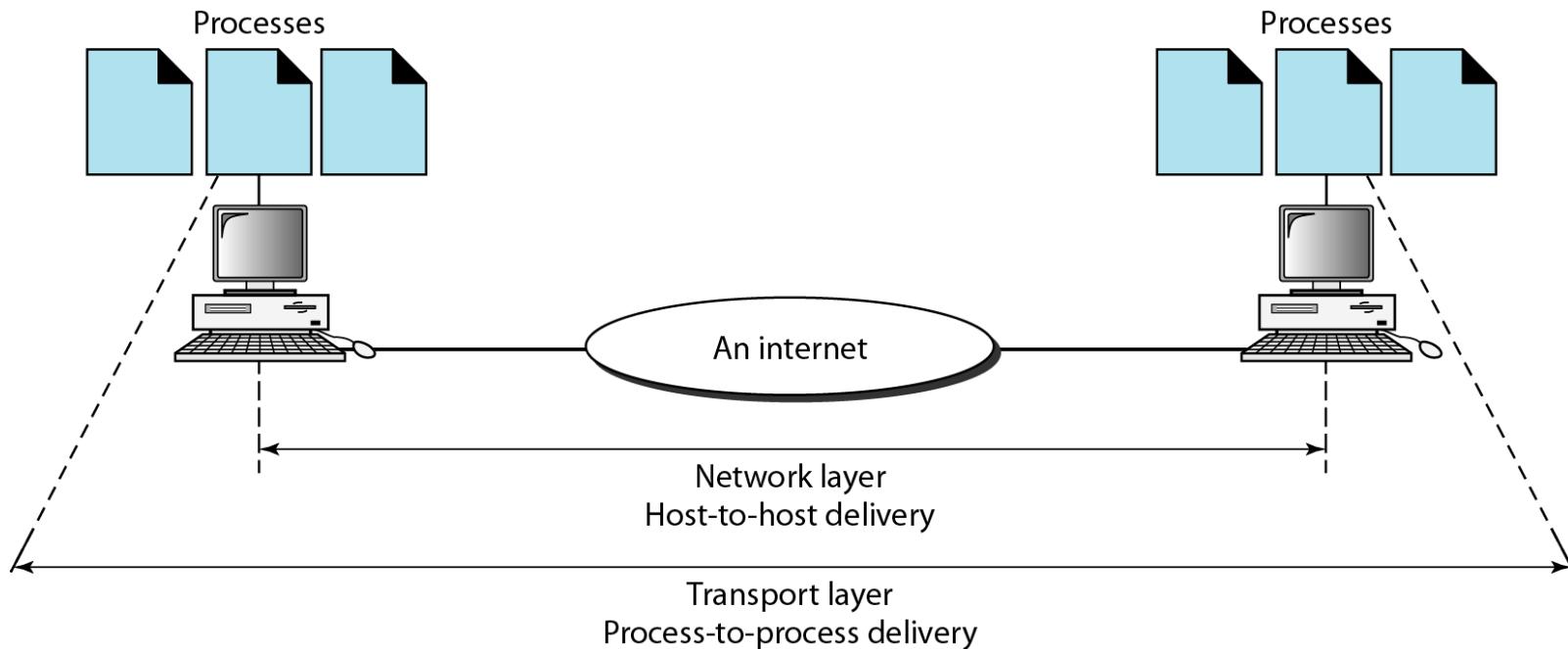
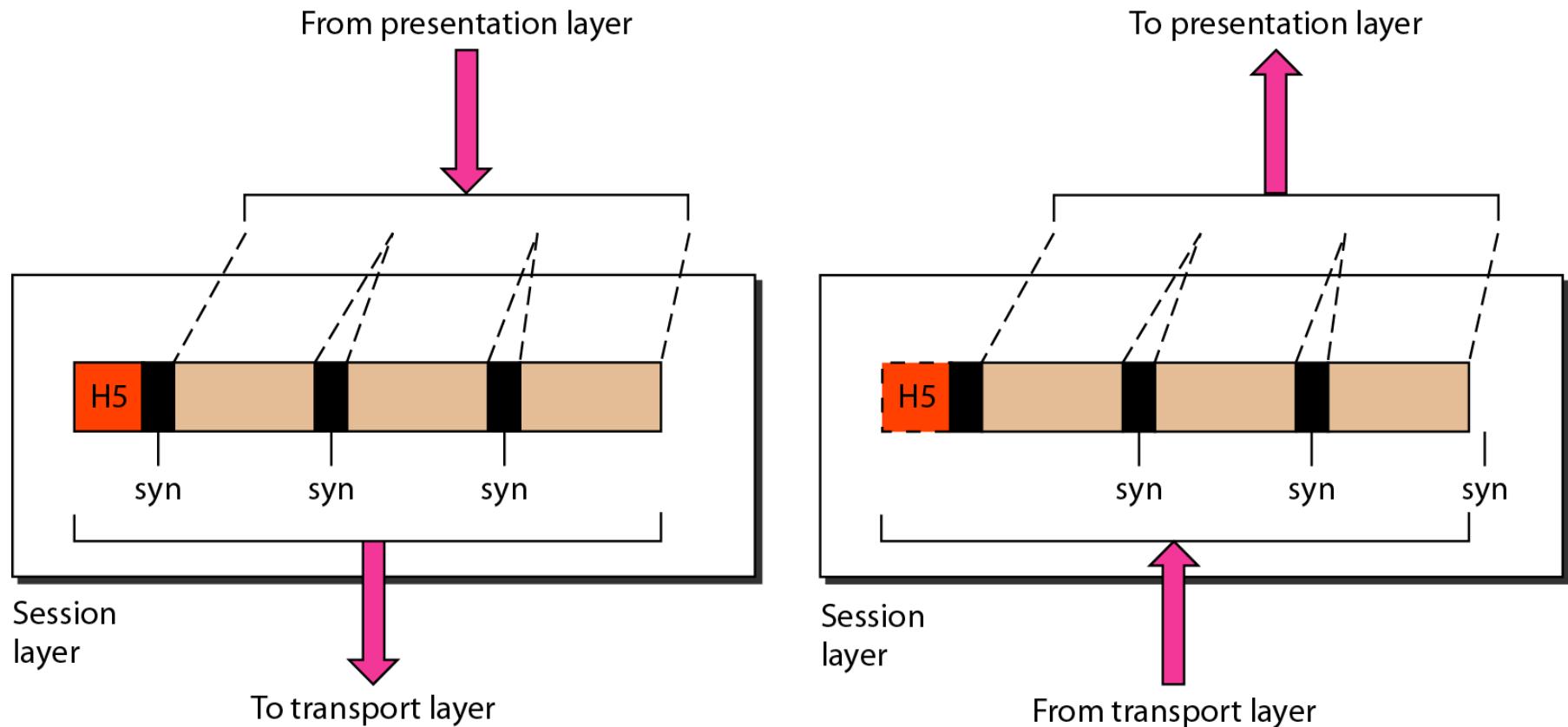


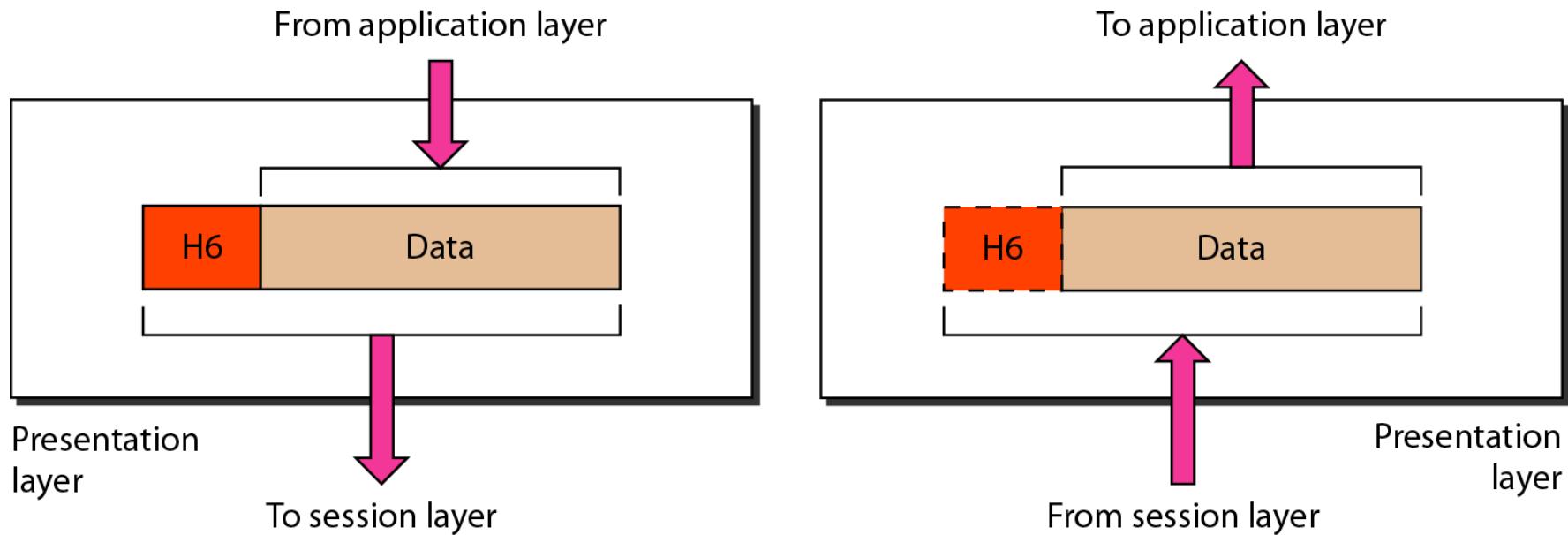
Figure 2.12 Session layer

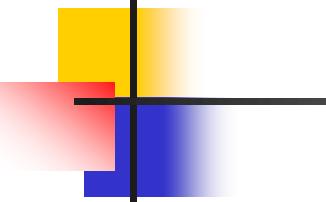


Note

The session layer is responsible for dialog control and synchronization.

Figure 2.13 *Presentation layer*

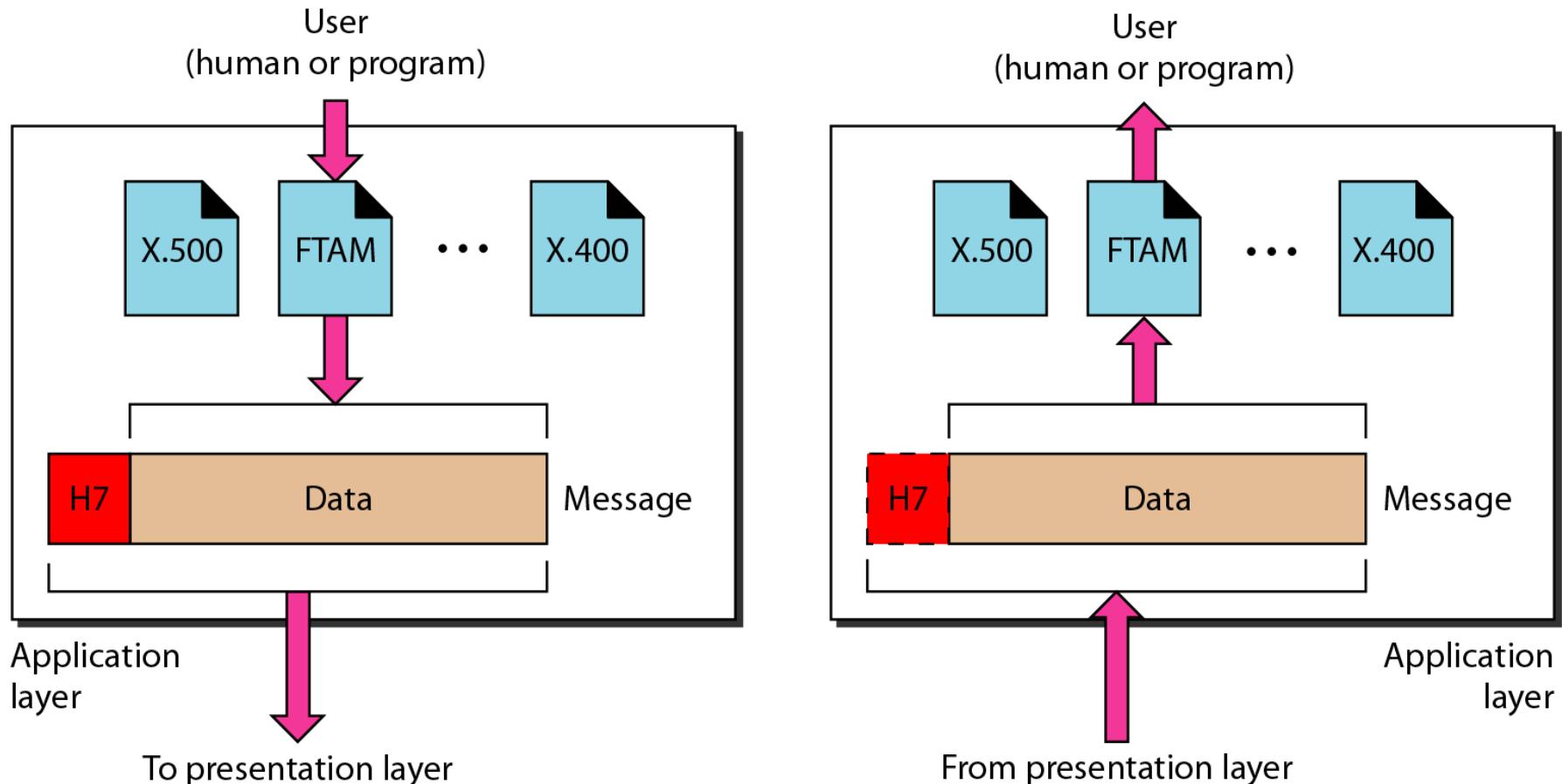


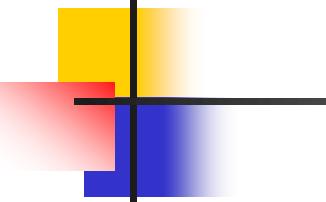


Note

**The presentation layer is responsible for translation,
compression, and encryption.**

Figure 2.14 Application layer

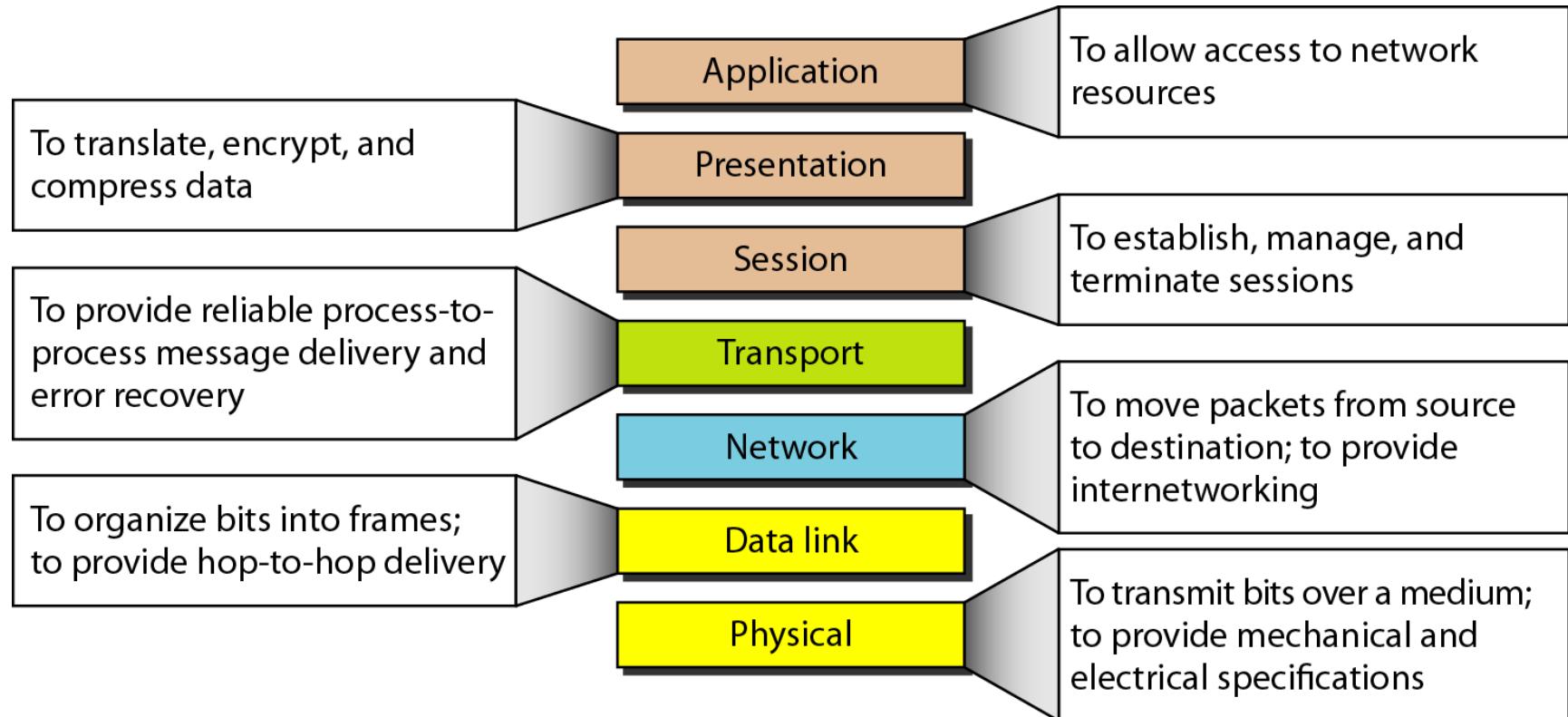




Note

The application layer is responsible for providing services to the user.

Figure 2.15 Summary of layers



2-4 TCP/IP PROTOCOL SUITE

*The layers in the **TCP/IP protocol suite** do not exactly match those in the **OSI model**. The original **TCP/IP protocol suite** was defined as having four layers: **host-to-network**, **internet**, **transport**, and **application**. However, when **TCP/IP** is compared to **OSI**, we can say that the **TCP/IP protocol suite** is made of five layers: **physical**, **data link**, **network**, **transport**, and **application**.*

Topics discussed in this section:

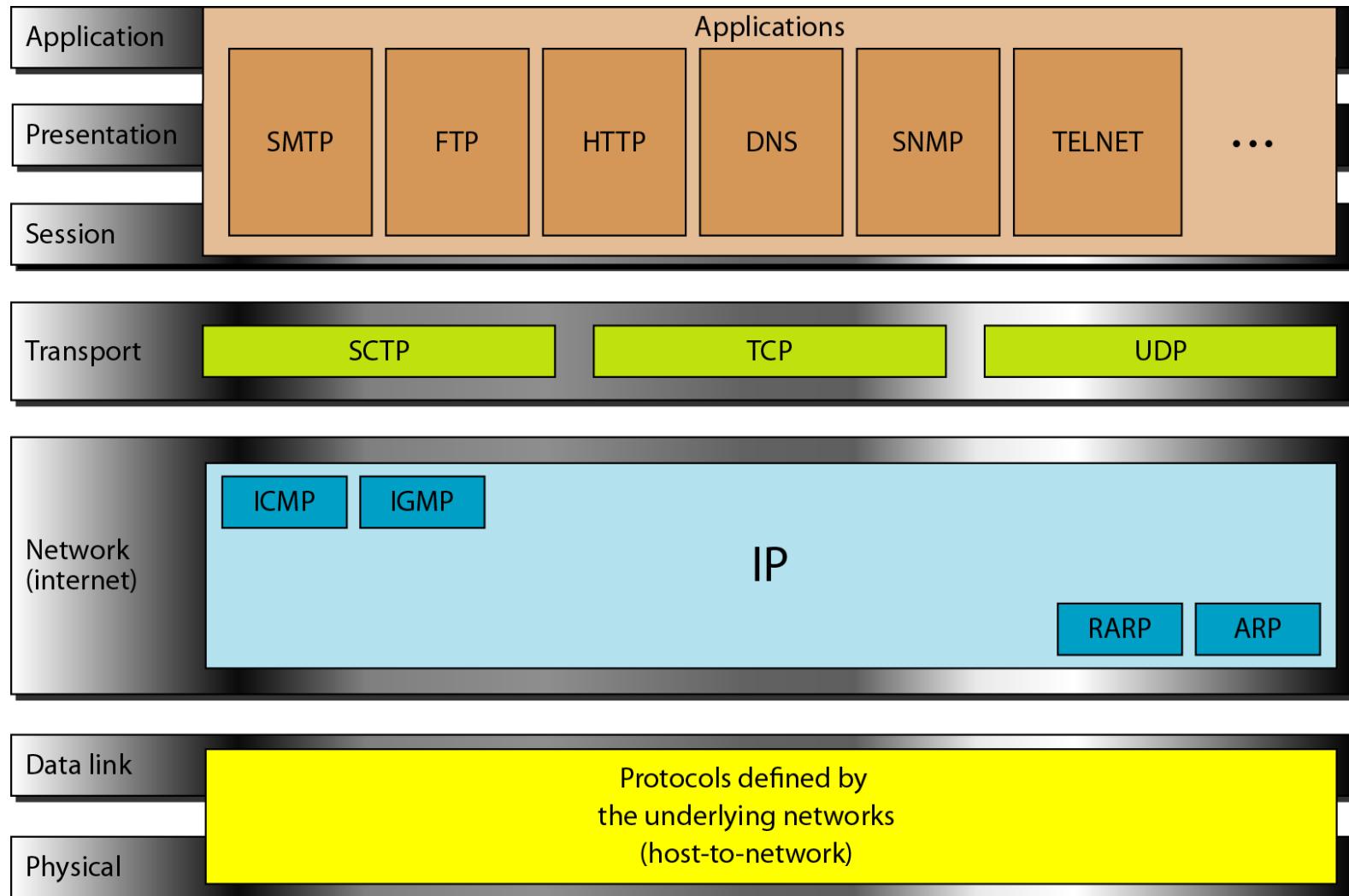
Physical and Data Link Layers

Network Layer

Transport Layer

Application Layer

Figure 2.16 TCP/IP and OSI model



2-5 ADDRESSING

*Four levels of addresses are used in an internet employing the TCP/IP protocols: **physical, logical, port, and specific.***

Topics discussed in this section:

Physical Addresses

Logical Addresses

Port Addresses

Specific Addresses

Figure 2.17 *Addresses in TCP/IP*

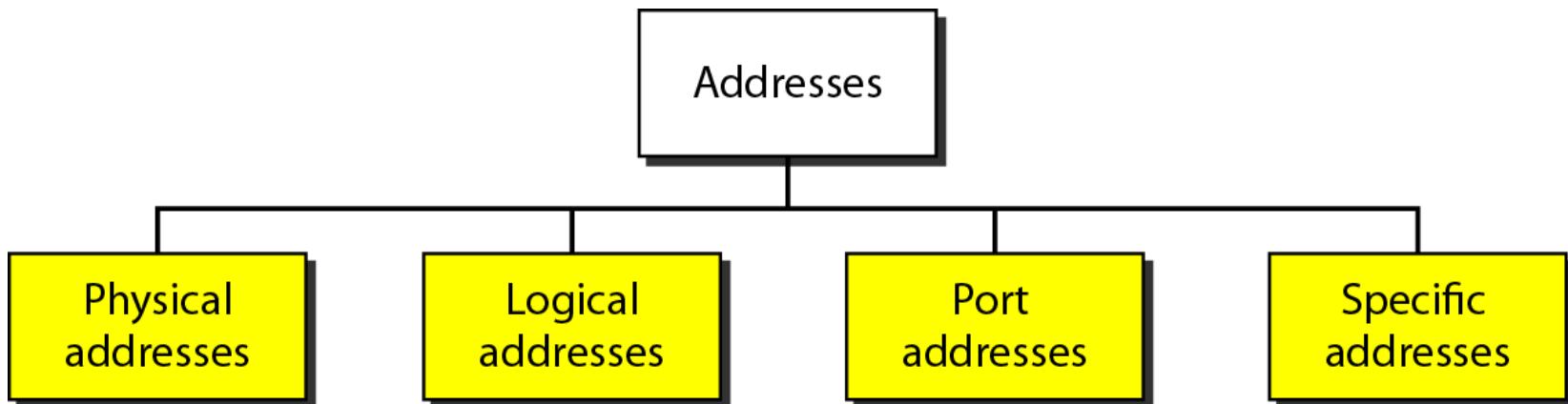
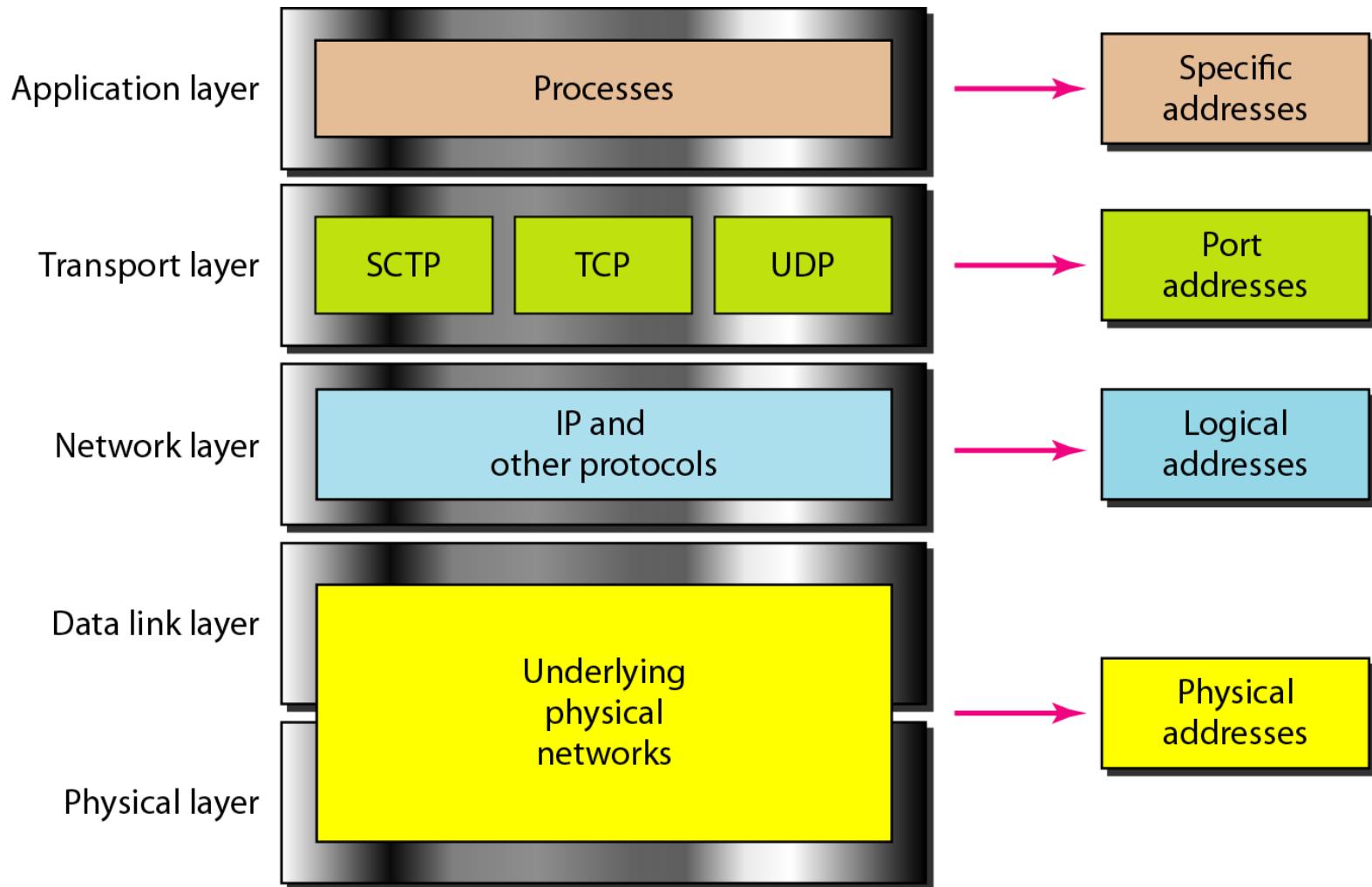
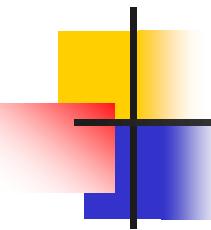


Figure 2.18 Relationship of layers and addresses in TCP/IP

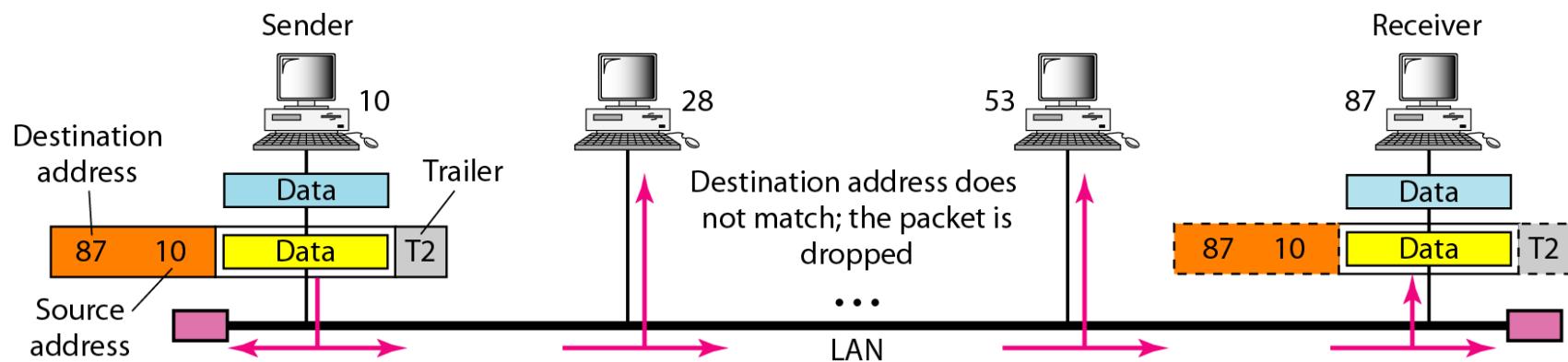


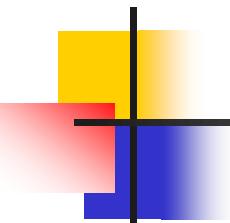


Example 2.1

*In Figure 2.19 a node with physical address 10 sends a frame to a node with physical address 87. The two nodes are connected by a link (bus topology LAN). As the figure shows, the computer with physical address **10** is the sender, and the computer with physical address **87** is the receiver.*

Figure 2.19 Physical addresses



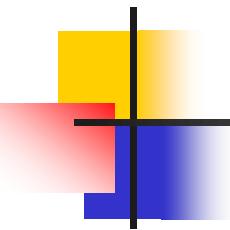


Example 2.2

*As we will see in Chapter 13, most local-area networks use a **48-bit** (6-byte) physical address written as 12 hexadecimal digits; every byte (2 hexadecimal digits) is separated by a colon, as shown below:*

07:01:02:01:2C:4B

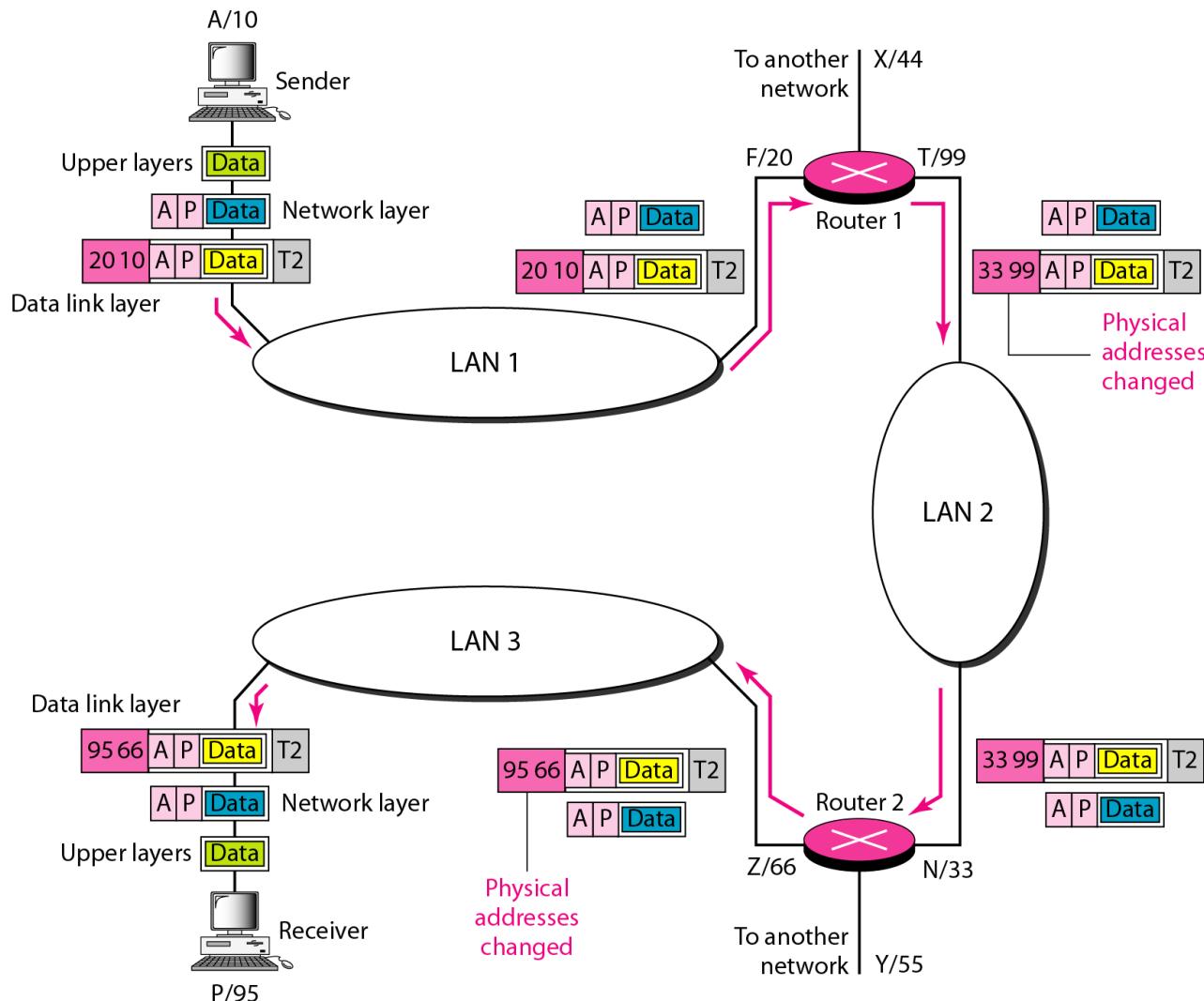
A 6-byte (12 hexadecimal digits) physical address.

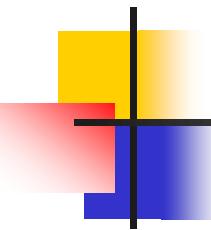


Example 2.3

Figure 2.20 shows a part of an internet with two routers connecting three LANs. Each device (computer or router) has a pair of addresses (logical and physical) for each connection. In this case, each computer is connected to only one link and therefore has only one pair of addresses. Each router, however, is connected to three networks (only two are shown in the figure). So each router has three pairs of addresses, one for each connection.

Figure 2.20 IP addresses

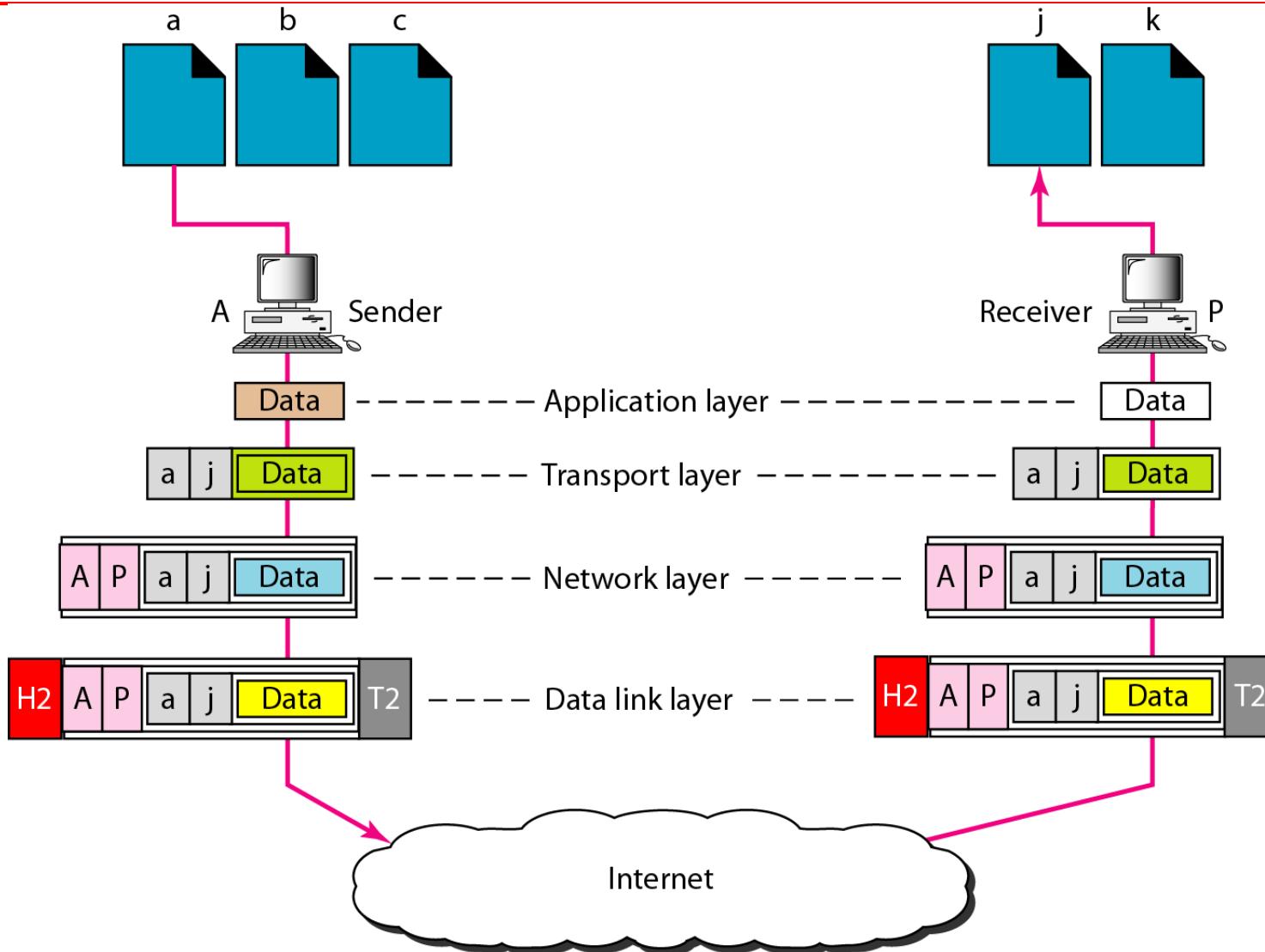


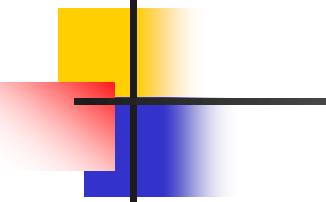


Example 2.4

Figure 2.21 shows two computers communicating via the Internet. The sending computer is running three processes at this time with port addresses a, b, and c. The receiving computer is running two processes at this time with port addresses j and k. Process a in the sending computer needs to communicate with process j in the receiving computer. Note that although physical addresses change from hop to hop, logical and port addresses remain the same from the source to destination.

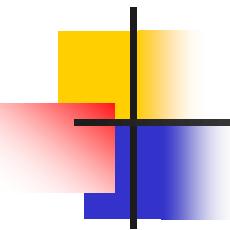
Figure 2.21 Port addresses





Note

**The physical addresses will change from hop to hop,
but the logical addresses usually remain the same.**

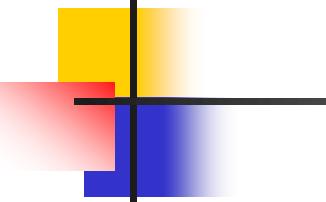


Example 2.5

As we will see in Chapter 23, a port address is a 16-bit address represented by one decimal number as shown.

753

**A 16-bit port address represented
as one single number.**



Note

**The physical addresses change from hop to hop,
but the logical and port addresses usually remain the same.**