Chapter 1

Introduction to Telecommunication Services, Networks and Signaling

Lorenzo Vangelista

The goal of telecommunication architectures is to provide people (and machines) with *telecommunication services*. In this chapter we give the basic definition of telecommunication services and we provide an introduction to telecommunication networks and signaling as well as to the well-known ISO/OSI model.

1.1 Telecommunication Services

1.1.1 Definition

The definition of telecommunication service is quite broad; basically, it can be considered as the transfer of information. In a telecommunication service at least three actors are usually involved:

1. one or more sources;
2. a carrier;
3. one or more receivers.

Principles of Communications Networks and Systems, First Edition. Edited by Nevio Benvenuto and Michele Zorzi.

The source(s) and the receiver(s) are not necessarily human beings; they can also be computers, for example. Usually, they are also called "customers" or "users", when we do not need to distinguish between the two roles. The carrier is normally a company owning equipment and resources, for example, frequencies, in the case of radio services. The equipment (hardware and software) that allows the information exchange is called *telecommunication network*, and this can be classified as a "local area network" when serving a specific small geographic area or a "wide area network" when serving a large geographic area. As an example the network serving an university campus is a local area network, while a large Internet provider network is a wide-area network.

Telecommunication networks can be categorized in many different ways. The most common classification refers to the services the networks carry. We have then:

- telephony networks carrying voice related services;
- data networks.

Using another criterion, still based on services, we have:

- networks dedicated to one specific service, such as the plain old telephone service (POTS);
- integrated services networks designed to carry different services at the same time, for example, the integrated services digital network (ISDN).

1.1.2 Taxonomies According to Different Criteria

In this section we classify telecommunication services according to different criteria, including symmetry, configuration and initialization. We would like to remark that the roles of source and receiver in a communication are not exclusive, that is, each party involved in the communication clearly can act as

1. a source only;
2. receiver only;
3. both source and receiver, at different time intervals;
4. both source and receiver, at the same time (think of ordinary phone calls).

Regarding the *symmetry* characteristic of telecommunication services, we can distinguish between

- *unidirectional* services, for example, television broadcasting;
- *bidirectional asymmetric* services, for example, Web browsing,[1] where there is a bidirectional exchange of information between the user and the Web server but the amount of

[1] Here we refer to the so-called Web 1.0; with Web 2.0, user-generated content plays a major role.

information flowing from the server to the user is by far larger than that flowing from the user to the Web server;

- *Bidirectional symmetric*, for example, the usual telephone conversation.

If we turn our attention to the *configuration* of services we can make the following distinction:

- *point-to-point* services, where only two users are involved in the telecommunication service, which in turn can be either unidirectional or bidirectional;
- *multipoint* services, involving multiple users, some acting as sources and some as receivers, possibly changing their roles dynamically;
- *broadcast* services, where a single source of information transmits to many receivers.

Telecommunications services can also be categorized based on their *initialization*; there are basically three categories

- *Call-based* services: any user needs to have an *identifier* to activate its service and this can be established at any time. A classical example is a telephone call or conference call.
- *Reservation-based* services: similar to call-based services, except that the service made to the carrier must be scheduled. This is often the case for conference call services nowadays or for bulk-data transfers, usually scheduled overnight.
- *Permanent-mode* services: service is negotiated by a contract between the carrier and the parties involved (two or more) and it is always running from the contract signature on. Examples include data connection between two data centers of a large company or, for a wireless operator without fixed infrastructure, a connection between two of its switches.

From the above examples, we would like to remark that in modern telecommunication networks there must be a mechanism to establish a service as easily as possible; this fundamental mechanism is actually hidden from users. Also, it is a network on its own, with users (usually the real end users) and specific protocols[2] called *signaling protocols*, which will be discussed in detail in section 1.5.

Finally, we consider the classification of telecommunication services according to their *communication mode.* Here we have *interactive* services and *broadcast* services. As far as the interactive services are concerned we can distinguish between:

- *conversational* services, usually bidirectional, where there must be very little delay in delivering the information; consider, for example, a telephone call and how annoying a delay could be – it sometimes happens for internet-based calls or for calls using satellite communication links;

[2] We have not introduced the notion of protocol yet. We can think of a protocol as a set of procedures and rules to exchange data or to let two entities agree on certain actions to be performed.

- *messaging* services, where there are no strict real time constraints but still the delivery time matters. An example is instant messaging, which is very popular nowadays;
- *information retrieval* services, where the interactivity and real time constraints can be mild, as is the case for a database search.

As far as broadcast services are concerned, considering them from the communication mode point of view, we may distinguish between services

- *Without session control* where a user has no control on what is delivered.
- *With session control* where a user can have some control on how the information is presented to her. An example could be the delivery of the same movie on a broadcast channel using different subchannels starting at different times, so that the user can somehow control the start of the movie at her convenience.

1.1.3 Taxonomies of Information Sources

In this section we classify information sources according to their time-varying characteristics and there are two types of sources:

- constant bit rate (CBR): the bit rate[3] emitted by the source is constant over time;
- variable bit rate (VBR): the bit rate may vary over time.

Example 1.1 A The digital encoding of speech according to the ITU G.711 standard (also known as pulse-code modulation or PCM encoding) produces 64 000 bit/s. This values is constant over time regardless of whether the source user actually speaks or not. We can conclude that a voice signal encoded according to the ITU G.711 standard is a constant bit rate source.

Example 1.1 B adaptive multirate (AMR) is a voice-encoding standard employed mainly in cellular networks like universal mobile telecommunications system universal mobile telecommunications system (UMTS) and it is designed to exploit both the speaker's periods of silence and to adapt to the different conditions of the transmission channels over which the bits are transmitted. If the speaker is silent a special sequence of bits is sent and then nothing is emitted until the speaker starts speaking again. On the receiving side when this sequence is received, noise is randomly reproduced, as a comfort feature, just to keep the receiver aware that the link is not broken. The AMR is basically made of eight source encoders with bit rates of 12.2, 10.2, 7.95, 7.40, 6.70, 5.90, 5.15, and 4.75 kbit/s. Indeed, even if the link between the transmitter and the receiver is fed at a constant bit rate, when the transmission channel deteriorates at some point in time, AMR is informed and selects a lower bit rate. Hence, more redundancy bits can be used (resorting to *channel coding* techniques) for protecting the voice encoded bits from channel errors. We can conclude that a voice signal encoded according to the AMR standard is a variable-rate source.

[3] The term "bit rate" has not been defined yet; the reader should rely on an intuitive meaning as the number of binary digits or bits per second (bit/s) sent over the channel.

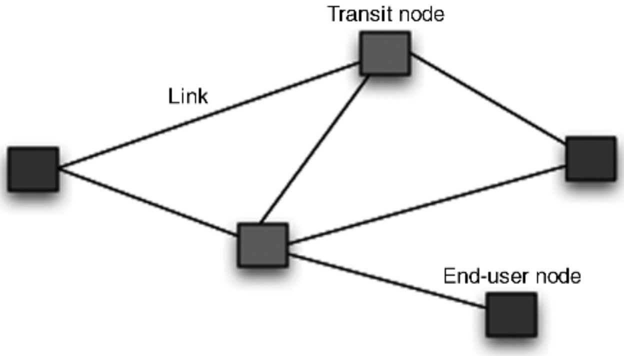

Figure 1.1 A generic network.

1.2 Telecommunication Networks

1.2.1 Introduction

In this section we give a general description of a communication network as an infrastructure for exchanging information.

As illustrated in Figure 1.1 a *network* is a collection of *nodes* connected by *links*. Nodes can be either *end-user* or *transit* nodes. Links connect two nodes and indicate that it is possible to exchange telecommunication services between them. When discussing the nature of the links we may distinguish between *physical* and *logical* links; the former represent the presence of an actual resource that connects two nodes and may be used for the communication; the latter are a more general abstraction of the capability of two nodes to communicate. That is to say, there could be many logical links conveyed by a physical link,[4] whereas many physical links could be needed to establish a logical link.

Example 1.2 A In a telephone network the *end-user* nodes are the usual telephones while the *transit* nodes include local exchanges and switches.

Example 1.2 B Let's consider a basic ISDN connection; it usually consists of *physical* link made of a copper twisted pair of cables and three *logical* links: two links (called *B links*) carrying user information and one link (called *D link*) carrying ancillary information used by the network to set up and control the calls. The bit rate of each B link is 64 kbit/s while the D link bit rate is 16 kbit/s.

It is common practice, when dealing with communication networks, to abstract from many details of the implementation, including both physical aspects of the network elements and the policies used to make them communicate with each other. On the one hand, this simply

[4] An example of this case, which will be discussed in the following, is the *bus*, that is, a single communication line which connects several nodes and enable communication among them, but only for one pair at a time.

means that we adopt a layered analysis of the network, as will be discussed later in this chapter. In particular, one layer considers the physical devices and another separate layer deals with the network operations from an abstract point of view. This enables us to look at the problem from a different, more theoretical perspective, founded on the branch of mathematics called *topology*.

A historical background In year 1735 the mathematician Leonhard Euler solved a popular puzzle of the time: the problem of the seven bridges of Königsberg [1].

This was a sort of networking problem, even though of course it did not involve any transmission or data exchange, but rather it was about finding a specific walk through the bridges of a city built over a river. The city was Königsberg, the then-capital of the Province of Prussia, which consisted of four districts (two sides of the river and two small islands in between), connected through seven bridges, as shown in Figure 1.2. More precisely, the problem consisted of finding a walk visiting every bridge, and doing so exactly once. The details of the actual problem can be found in [2]. The full tale of Euler's solution of the problem is very educational and is recommended as further reading.

We are more interested in which instruments to use in order to approach and solve the problem, so that we can use the same instruments for information networks. Euler found a clever way to translate the bridge problem into a much simpler symbolic representation, which could be analyzed from a theoretical perspective.

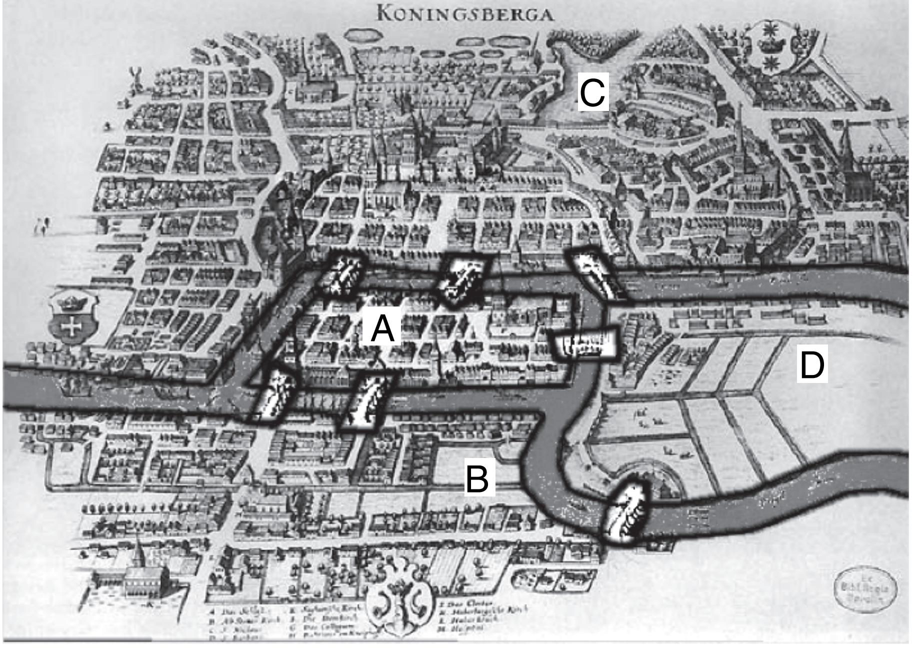

Figure 1.2 A map of Königsberg and its seven bridges. The bridges are highlighted in white with black border. The city districts are labeled A, B, C, D, as done in the original formulation by Euler.

It is clear that an immediate and cheap way to approach the problem is, rather than visiting the city, to use a map; instead of walking through the bridges, one can simply look at the graphical representation. However, Euler realized that a good mathematical model did not need unnecessary details. Thus, instead of a precise map, in which the geometrical proportion of the bridges and the city districts were respected, it was possible to use a simpler scheme. What does matter is just which parts of the city are connected, and how.

Thus, Euler used a simple tabular representation, where he took note of which parts of the city were interconnected and by how many bridges. This is a simple topological representation, where only the strictly necessary details are considered. It is commonly believed that in this way Euler invented the branch of mathematics known as *topology*, and more specifically, the part of it called *graph theory*. However, the original representation he gave of the problem involved a table, not a graph [1]. Yet, later rearrangements made by other scholars proposed a graphical representation, which is equivalent to Euler's table. Such a representation is reported in Figure 1.3.

Through this formulation of the problem, the whole city is modeled as a *graph*, where the districts and the bridges are the *vertices* and the *edges* between vertices, respectively. The problem of traversing the bridges could be analyzed by means of this representation, for example by marking, or specifically *coloring*, the edges corresponding to traversed bridges. It is worthwhile noting that this representation preserves very few characteristics of the original problem. Indeed, the only elements given by the graph are those that are topological invariants of the network – which cannot be altered without substantially changing the problem. In this problem, the topological invariants are the connections between districts along the bridges, whereas it does not matter how large the bridges are or what the physical distance between the districts is.

More in general, to formulate a problem in terms of network topology means to abstract from many aspects and mostly consider just connectivity properties, for example whether two vertices are connected through an edge or not. Once we solve a problem formalized in this manner (or, as is the case for Euler's original problem, we prove it does not admit any solution),

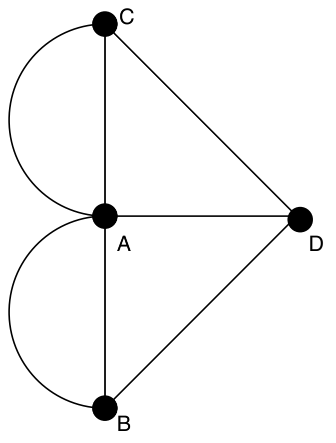

Figure 1.3 The graph resulting from Königsberg's bridge problem.

we do the same for any problem which is *topologically equivalent*, that is, that involves the same questions on a network with identical connectivity properties.

Notation and terminology The topological representations that we will use for communication networks do not significantly differ from that used for Königsberg's bridge problem. Network devices that are able to interconnect with each other are generically called *nodes*; they are represented as the vertices of a graph, which are in turn linked through edges representing connection means, that is, cables, radio links, optical fibers or something similar.

The whole network is thus represented as a graph $\mathcal{G}$, which is often written as $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N}$ is the set of the nodes and $\mathcal{E}$ is the set of the edges. The number of elements in set $\mathcal{N}$ is $|\mathcal{N}|$. According to the context, the elements of $\mathcal{N}$ can be denoted in many ways; it is usual to use integer numbers (in which case the nodes are labeled , for example, as $1, 2, \ldots, |\mathcal{N}|$), or capital Latin letters, which is the notation we will employ in this chapter; we will use instead lower-case Latin letters to denote a generic node. In some cases, the edges are also called *arcs*, *links*, *hops*. We will use all these terms as synonyms.

The set of edges $\mathcal{E}$ is a subset of $\mathcal{N}^2$. Thus, a typical element of $\mathcal{E}$ can be written as (A,B), where A,B $\in \mathcal{N}$. In this case, node A is said to be the source (or the transmitter, or the tail) and node B the destination (or the receiver, or the head) of the edge. However, the inclusion of $\mathcal{E}$ the pairs of the form (i, i), that is, with both elements equal to the same node, is usually avoided. This would correspond to a single-hop looping to and from the same node, which is usually unrealistic.

Notable network topologies Several network topologies are actually possible (see Figure 1.4). Some of them are rather complicated because, as will be discussed in Chapter 10, they can be nested. In other words, a larger network can be formed by connecting two or more networks, so as to create a hierarchical structure. However, there are certain network topologies that are, in a sense, considered to be the most basic and frequently studied. In other words, they are standard cases, and there are proper names for them.

These notable topologies include the following cases:

- *Star* topology: a center node is connected to every other node; there are no links between the other nodes. This case physically represents an interconnection where several end users can communicate only through a transit node (which is the only one to be connected to all of them).
- *Mesh* topology: in this kind of network, every node is connected to every other node; sometimes, this full interconnection is better emphasized by calling it "full mesh."
- *Tree* topology: one way to obtain this topology is to have one node, called *root*, connected with multiple nodes (not connected to each other), called *leaves*; this is actually the same as the star topology defined above, so the star topology is also a tree topology; however, we can also have a tree if we replace any of the leaves with the root of another tree topology, so that the tree is obtained as a hierarchy of nodes; more in general, as will be discussed in Chapter 10, the term "tree" actually applies to all the topologies where there are no loops (as they will be defined in this chapter).
- *Ring* topology: each node is connected to two other nodes only. Moreover, assume we arbitrarily select one node A of the network; then, pick one of the two neighbors of node A,

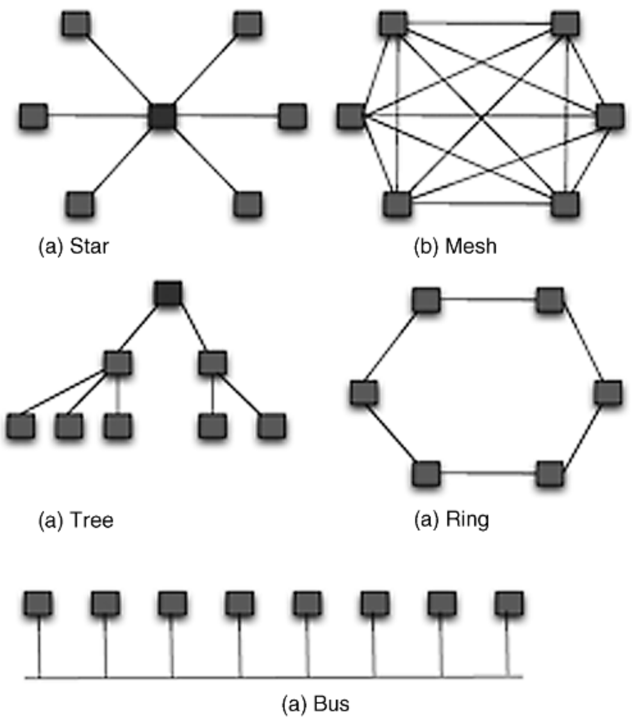

Figure 1.4 Five network topologies.

and call it B. Node B is connected to node A and to the other node (call it C). Now, repeat the procedure indefinitely by replacing (A,B) with (B,C), respectively. Sooner or later, A will be "touched" again by this procedure. This means that the network contains a loop; actually, it is only a single loop.

- *Bus* topology: from the strict topological point of view, this is the same as a mesh topology; that is, every node is connected to every other; however, in a bus topology the connection is achieved with a special communication line that is shared by all nodes. This line, called a *bus* admits multiple ways of interconnecting nodes but it can be used only by a single pair of nodes at a time. The definition of bus topology introduces a further element, beyond the mere placement of nodes and links, because it also requires a characterization in terms of medium access. These differences will become clearer when the medium access control is discussed in Chapter 9.

1.2.2 Access Network and Core Network

The topological representation of the network gives an abstract perspective describing a mathematical structure. From an implementation point of view, most telecommunication networks are further subdivided into two parts:

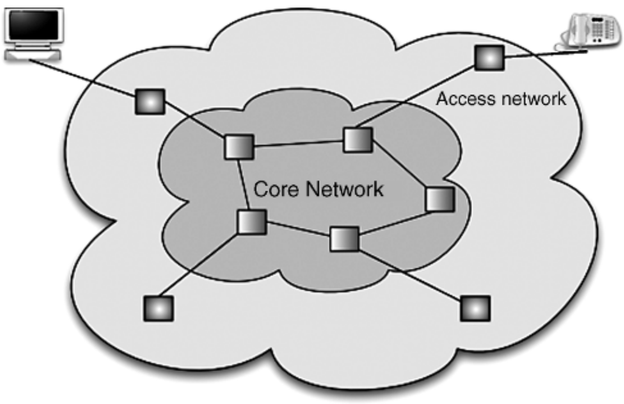

Figure 1.5 Core and access networks.

- The *access network*, which is the part interacting with the end user, collecting and transferring the information to the end user; it is the edge of the network itself and enables users to access the network.
- The *core network*, which is responsible for the transmission of the information over long distances, routing the information from one user to the other, and so on; it is the core of the network itself.

The general architecture is shown in Figure 1.5.

There are networks for which the distinction between access and core networks is not valid: these are the *peer-to-peer* networks where all nodes have the same characteristics and there is no hierarchy. Two remarkable examples of peer to peer networks are:

- the *Skype*™[3] network, where the voice traffic is routed through a series of nodes and there is no "central server" to/from which everything is sent;
- some *sensor networks*, especially *multihop sensor* networks where sensors (for example, measuring the temperature) send their measurements – according to specific protocols – to their neighbors and the information is propagating hop-by-hop to the *information sink*.

Example 1.2 C Consider a very important service nowadays, the asymmetric digital subscriber loop almost surely (ADSL), whose architecture is based on three components (Figure 1.6 provides a generic scheme for this architecture):

- User equipment called an ADSL modem to which the user attaches a PC or other networking equipment like a router to form a local area network. Incidentally, equipment combining modem and router in one piece of hardware modem is widespread.
- A digital subscriber line accesses multiplexer (DSLAM) at the local exchange on the customer side. It intermixes voice traffic and ADSL, that is, it collects data from many modem ports and aggregates their voice and data traffic into one complex composite signal via multiplexing. On the core network

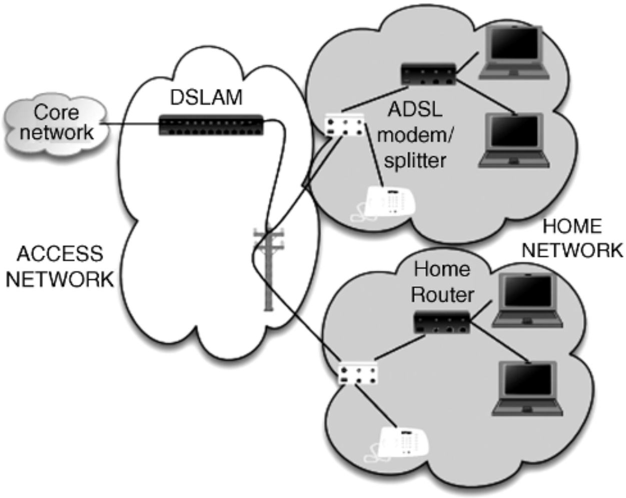

Figure 1.6 Core and access networks for the ADSL service.

side it connects the data signals to the appropriate carrier's core network, and the phone signals to the voice switch. The DSLAM equipment aggregates the ADSL lines over its asynchronous transfer mode (ATM), frame relay, and/or Internet Protocol network port. The aggregated traffic at up to 10 Gbit/s is then directed to a carrier's backbone switch (in the core network) to the Internet backbone.

- The Internet backbone.

1.3 Circuit-Switched and Packet-Switched Communication Modes

In this section we briefly discuss the two basic modes in which communication can occur in a network, that is, the *circuit-switched* and the *packet-switched* paradigms. Further insight can be found in Chapter 10.

We say that a *circuit-switched* communication mode is used when, in a communication between user A and the user B, there is a uninterrupted physical route over the information that is continuously flowing; the route, traversing the home network, the access network and the core network and again the access network and the home network, is established at the beginning of the communication and not changed along the communication session. This communication mode is also often referred to as "connection orientated."

Conversely, *packet-switched* communication mode is adopted when, in a communication between the user A and the user B, the information (usually digital or digitized) is grouped into packets, each one having embedded the *address* of the user B, and these packets are sent across the networks without the constraint that each and everyone follow the same route. The route that each packet is following may be determined by the nodes it is traversing and can of course vary during the session. This communication mode is often referred to as "connectionless."

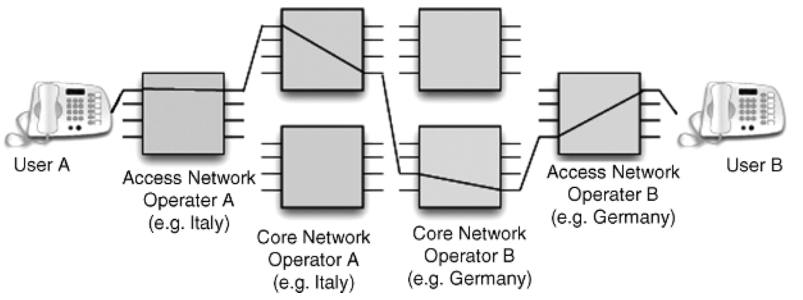

Figure 1.7 Circuit switched network.

The two communication modes have advantages and disadvantages, which will be highlighted later, although the general trend is towards packet-switched communication networks. The following example tries to clarify the differences between the two modes.

Example 1.3 A The plain old telecommunications network (POTS) (that is, the common, basic form of voice-based telephony still in use for residential services nowadays) of Figure 1.7 is probably the most classical example of a circuit-switched network; at the establishment of the phone call a physical deterministic route is established from user A, along the local switches and the transit switches (of the core network) of possibly different operators, to user B. The phone of user B rings only when this route is found and it is available. It could happen that some of the links are completely busy and cannot accommodate another call; in this case, user A gets a busy tone.

Example 1.3 B An example of a packet-switched network is the Internet (see Figure 1.8), which is based on the Internet protocol (IP) where each "user" has an identifier, called an *IP address*. When user A wants to send some digital information to user B, it puts the information in *packets* containing the IP address of user B and presents the packet to the first edge *router* (almost equivalent to the switch of a POTS network) of the network. The rule is that based on already known information, any router forwards the packet to the router attached to it, which is in the best position to get the packet delivered to user B. The choice of the router can change over time, so that two consecutive packets can take different routes and it might as well happen that the latter packet arrives before the former. It is up to the receiver to reorder the packets, check that everything is delivered as expected, and so on.

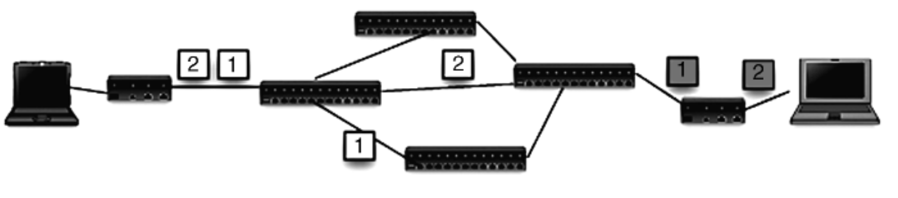

Figure 1.8 Packet switched network – different gray tones of the same box number represent the same packet at different moments in time.

1.4 Introduction to the ISO/OSI Model

In this section we introduce the ISO/OSI model and provide some examples of its application.

1.4.1 The Layered Model

First we introduce the notion of *protocol* as a set of rules defining telecommunication systems interact. To reduce the unavoidable confusion that could be generated by describing protocols without any guideline, a kind of template has been defined to enable

- a consistent definition of different protocols;
- an easier comparison between them; and
- a *translation* of one protocol into another, through a suitable conversion.

This template is the *layered model* and the *ISO/OSI model* is a particular instance of this model.

According to the layered model, a communication, unidirectional or bidirectional, between two systems, A and B, is made of *layers* and *entities* (see Figure 1.9). For each system, each *layer-(N – 1) entity* provides a *service* to a *layer-N entity*. The dialog between the two entities is carried out via an exchange of service messages called *primitives* and data messages called packet data unit (PDU) through the service access point service access point (SAP). In this manner, a layer-*N* entity of system A gets to talk to a peer layer-*N* entity of system B, as

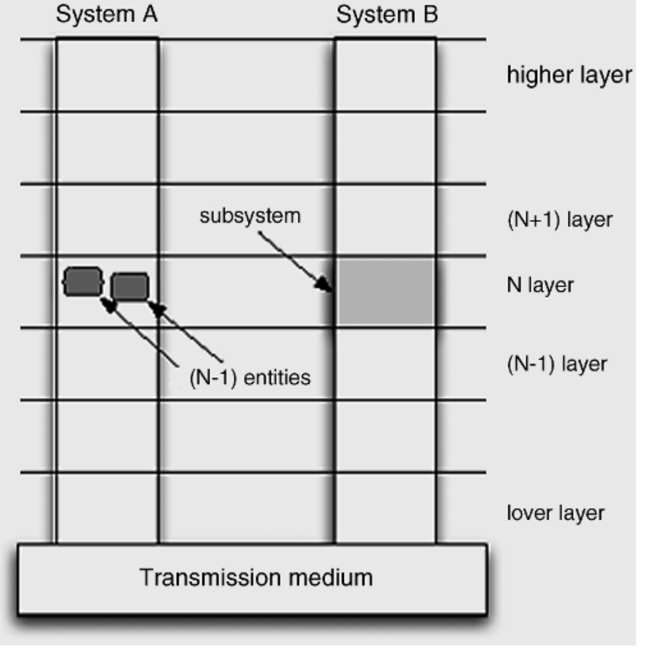

Figure 1.9 Layered architecture.

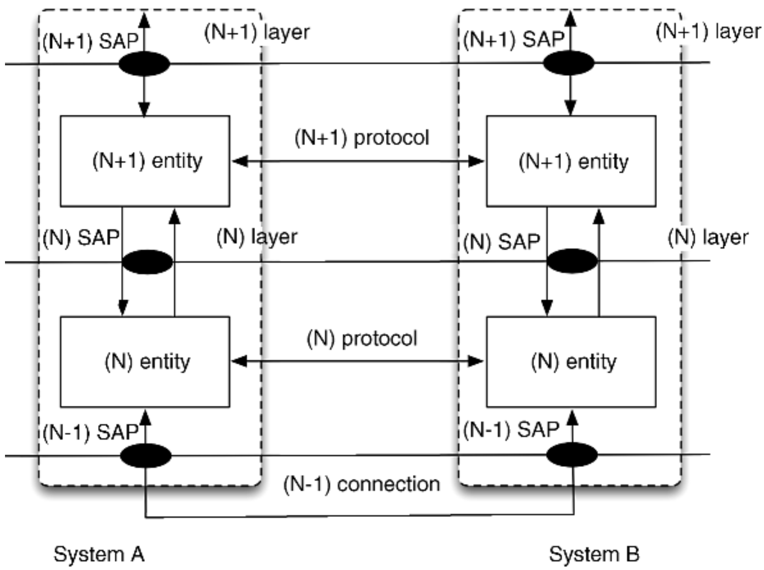

Figure 1.10 SAPs, services and protocols in a layered architecture.

messages go down on system A and then up on system B. The way the two peer entities talk is called *layer-N protocol* and is shown in Figure 1.10.

An insight on how the data message is actually processed is given in Figure 1.11, where we see that through the layer-N SAP a layer-(N+1) PDU is passed down; this PDU contains the actual pieces of information transmitted from one system to the other at a certain layer. The underlying layer, however, does not simply carry the layer-(N+1) PDU to the other side

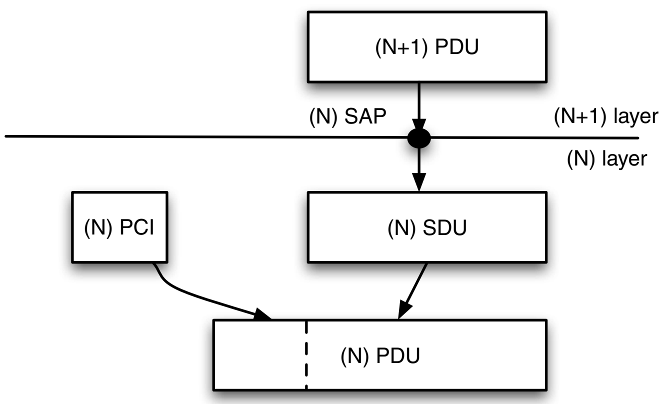

Figure 1.11 PDU, PCI and SDU in a layered architecture.

but clearly needs to add some information needed for the layer-N protocol to work. As a consequence, a protocol control information (PCI) piece of information is appended to the layer-(N+1) PDU (which in the context of layer-N is called *layer-N service data unit service data unit (SDU)* to highlight that it is coming from the SAP) to make the layer-N PDU, as shown in Figure 1.11. Exactly one PDU fitting into one lower layer PDU is a particular case: actually, it is very common that the layer-(N+1) PDU is split into smaller pieces by the layer-N protocol so that the transmission of a layer-(N+1) PDU is accomplished via the transmission of several layer-N PDUs.

As said before, the service interaction through SAPs is carried out through the exchange of primitives, which usually are of four types:

1. *request*
2. *confirm*
3. *indication*
4. *response*

This is shown pictorially in Figure 1.12, where two primitives are shown on one side and two on the other side, grouped as they usually are.

Many protocols do not follow the layered architecture and instead violate it by having, for example, the (N+1)th layer sending primitives to the (N–1)th layer. Furthermore, a recent

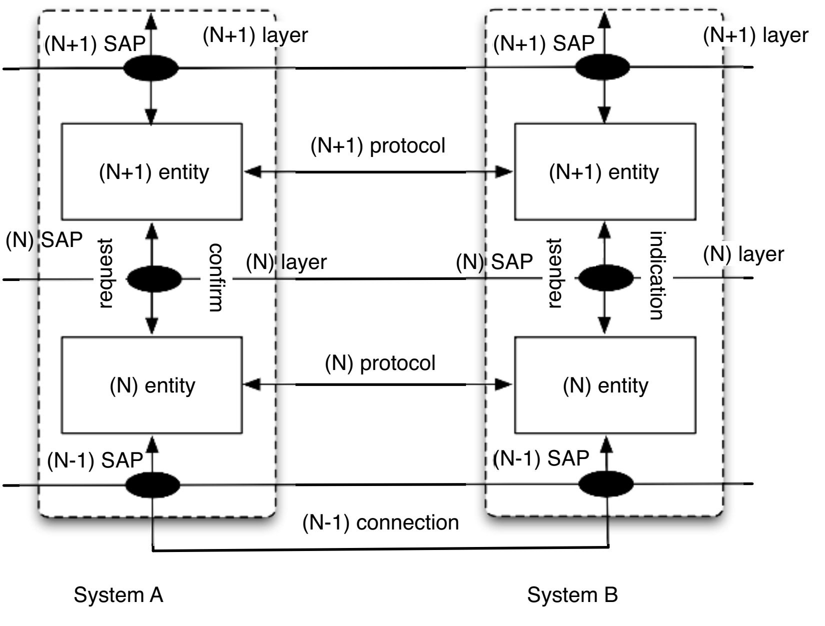

Figure 1.12 Primitives in a layered architecture.

research area called *cross layer* design is deliberately designing protocols to avoid each layer working in an isolated way, as implied by the layered approach.

The interaction between layers described above is reflected in what is called the *encapsulation* of information. In practice, this corresponds to the inclusion of a layer-N PDU as a part of a layer-(N–1) PDU. That is, a piece of information from a higher layer is complemented with some control information, then the resulting data are treated by the lower layer as a whole piece of information. Another control component is appended to it (often as a header, but sometimes as a trailer). This information-hiding process has advantages and disadvantages. It certainly enables modularity of the data, so that each layer only has to manage the control information related to it. However, it may cause a considerable increase in the number of bits sent over the network: in fact, since lower layers are blind to the content of the data exchanged, they are often forced to duplicate control information parts. Finally, there are cases, such as the interaction between TCP/UDP and IP (in particular the pairing of TCP/UDP port and IP address) that do not respect this concept faithfully, as will be discussed in Chapter 10.

1.4.2 The ISO/OSI Model

The ISO/OSI model (Figure 1.13) is a particular instance of the layered protocol architecture specifying:

- the number of layers, set to seven;
- the function and naming of each layer.

In the following we give a short description of each layer.

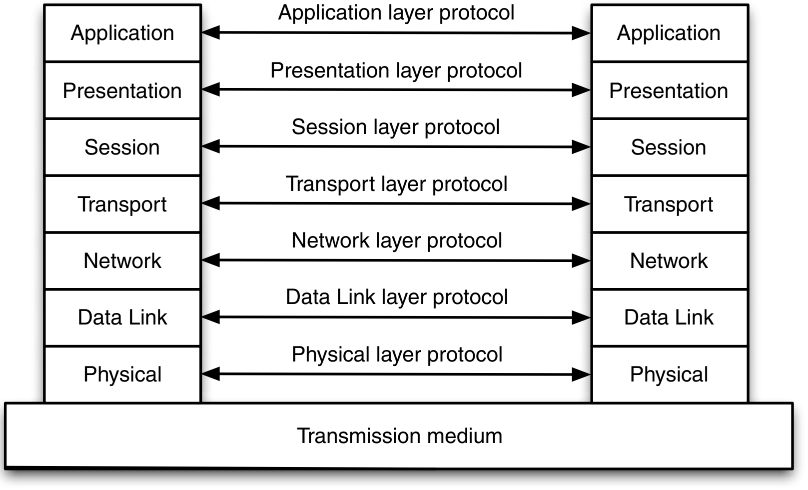

Figure 1.13 The ISO/OSI model.

1. The *physical layer* is responsible for the actual delivery of the information over the physical medium; for example, on the transmitter's side, by employing a modulator, a given waveform is sent over the medium when the bit is 0 and a different one is sent when the bit is 1; so, on the receiver's side, a demodulator can reconstruct the 0/1 sequence based on the received waveform.

2. The *data-link layer* is mainly responsible for enabling the local transfer of information. Its intervention is necessary in order to enable the actual exchange of bits through the physical layer. Indeed, there are several functionalities of the data-link layer and they are related to different needs. For this reason, it is common, especially within the IEEE 802 standards, to distinguish between two *sublayers*, which both belong to the data-link layer: (i) the medium access control (MAC), which manages the presence of multiple users on the same medium, and (ii) the logical link control (LLC), which provides the interaction between the upper layers and the MAC as well as performing flow and error control on a local basis. Indeed, the classification of these two entities as two subsets of the second layer, instead of being two separate layers themselves, is more of a conventional matter. Moreover, it is worthwhile noting that their features may not always be required. For example, there might be one single user in the network, in which case multiplexing through MAC could be avoided (or better, simplified, since the availability of the medium still has to be checked). If the network is very simple or the communication medium is guaranteed to be extremely reliable, one may also avoid performing error and flow control on a local basis, thus simplifying the LLC. However, in most cases the operations of the data link layer are quite complex and will be discussed in detail in Chapter 9.

3. The *network layer* is responsible for setting up, maintaining and closing the connection between the network layer entities of different subsystems. It makes the way in which network resources are used invisible to the Upper layers. Referring to Figure 1.14, the network layer entity on system A does not know whether the flow of information is going

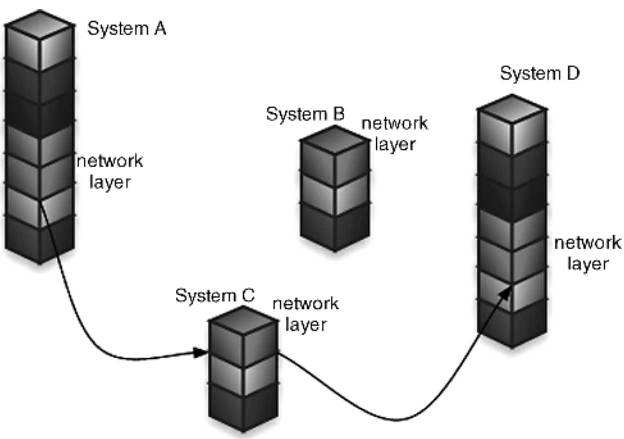

Figure 1.14 The network layer.

through systems B or C while exchanging information with system D. To achieve this goal, the network layer is responsible for *host addressing*, that is, being able to identify all the nodes in the network in a unique manner, and *routing*, that is, identifying the most desirable path for the data to follow through the network. It is worth mentioning that, whereas the data link layer is concerned with local communication exchanges, the network layer has a global view. This is not surprising and is implicit in the layers' names: layer 2 is obviously concerned with single links, while layer 3 considers the whole network.

4. The *transport layer* is responsible for creating a one-to-one *transparent* virtual pipe between the two ends of the communication flow. It does not deal with the network topology; instead it hides the presence of multiple terminals interconnected in a network fashion, which was the key aspect of the third layer. At layer four, communication proceeds as though only two entities existed in the network; when the communication session is established the transport layer sets up this virtual pipe according to the requests of the session layer. It may also include features to deliver an error-free communication, for example, requesting the network layer to retransmit the pieces of information that are in error. Finally, the transport layer can perform *flow control*, that is, it prevents the transport layer on one side from sending information faster than the rate that the receiving transport layer on the other side can handle.
5. The *session layer* The session layer is responsible for establishing, suspending and tearing down in an orderly fashion any communication session the presentation layer might need. In a POTS network, people adopt, as a matter of fact, a *call-control* protocol, which controls the session. In fact, the dialer takes the phone off the hook, waits for the dial tone, dials the number, waits for the ringing tone and subsequently waits for the receiver to answer, talks, then closes the communication by putting the phone on the hook. If during any of these operations a busy tone is heard, the communication is dropped and the dialer is invited to try again later. When using a cellular phone, a slightly different protocol is adopted since the network availability may be known in advance. Furthermore, even a simple POTS call may have multiple concurrent sessions, that is, a call can be put on waiting, meanwhile another call goes on and finally the previous call is resumed.
6. The *presentation layer* is responsible for providing directly to the application layer the session layer services as requested by the applications and for processing the data to make them usable to the applications: for example, the encryption and decryption of the data are usually a responsibility of the presentation layer.
7. The *application layer* is self-explaining: it is the realm of the applications; normally users interact with networks at this level.

1.5 Signaling

In this section we provide an introduction to the concept of signaling, giving some background and indicating why it is essential in every network. A short historic and geographic survey is given, listing different types of signaling system at different times and in different regions.

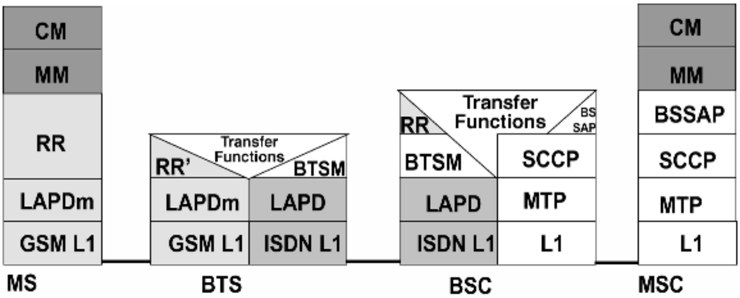

Figure 1.15 The GSM protocol architecture.

1.5.1 Introduction

In almost every system, with some remarkable exceptions such as the Internet, the communication between two end-points involves two *planes*: a *user* plane and a *control* plane. Both of them are structured according to the layered architecture previously described, but usually, we, as users, are exposed only to the *user* plane. Users just send and receive useful data, voice, video; to there is a kind of cooperating network behind the fourth wall which is taking care of setting up the communication links, monitoring them, reconfiguring them if we are moving (as in a cellular network) etc. This is what is called the *control* plane, which is normally implemented according to the layered architecture that is, there is a physical layer (which can be shared somehow with the *user* plane), a data link layer, a network layer and so on. Of course there are SAPs and primitives as well.

To have an idea of how complicated the control plane can be, refer to Figure 1.15, which shows the *control* plane of a GSM (Global System for Mobile communications: originally from Groupe Spècial Mobile) network (for the sake of simplicity the figure only includes the elements related to the voice service).

The *control* flow of information is called *signaling* and the *control* plane protocols are called *signaling protocols*. A coordinated set of signaling protocols and entities is called a *signaling system*.

1.5.2 Channel-Associated and Common-Channel Signaling

There are two types of signaling systems:

- *channel-associated* signaling;
- *common-channel* signaling.

All types of signaling can be divided in two parts: the access network signaling, or *user signaling*, and the core network signaling.

Channel-associated signaling was the first to be used and is divided into *in-band* signaling and *out-of-band* signaling, where it is intended that the *band* is the voice band. There are three *channel-associated* systems still in use:

- the CCITT R1 system, very similar to the Bell System multifrequency signaling used in the USA since after the second World War;
- the CCITT R2 system used in Europe since 1960;
- the CCITT signaling system 5, defined in 1964 and used in intercontinental analog links.

For reason of space we will not go into details about these systems; they are very likely the ones we interact with on a daily basis in user signaling (e.g., DTMF). For more in-depth information the reader is referred to [4]. Next subsection is instead devoted to the most important *common-channel* signaling system: the CCITT signaling system 7.

1.5.3 SS7

In this section we provide a brief overview of the CCITT signaling system 7. This system is replacing step by step the *channel-associated* signaling systems.

The main difference between this system and the previous ones is that while in the other systems the nodes and the links for the signaling are in common with those carrying the user information, in CCITT signaling system 7 the signaling network (rather ironically) is not in common with the user information network. In other words, the *user plane* is completely separated from the control plane. The reason for using the word "common" when talking about the CCITT signaling system 7 is that a given signaling link is not dedicated to a specific user link: it is used by the signaling network for all the signaling and so is common to all user links. The revolutionary concept brought in by the CCITT signaling system 7 is that there is an overlay network with respect to the one carrying the user information, which is not synchronous with it, perhaps using different physical layer transmission technologies, that controls the user information network. Figure 1.16 illustrates this concept.

The overlay network is called *common channel signaling* (CCS) network. It is made of two different types of nodes:

- *Signaling Point* (SP) nodes, which are at the edge of the network and are either the sources or the destinations of the signaling information; normally they are mapped

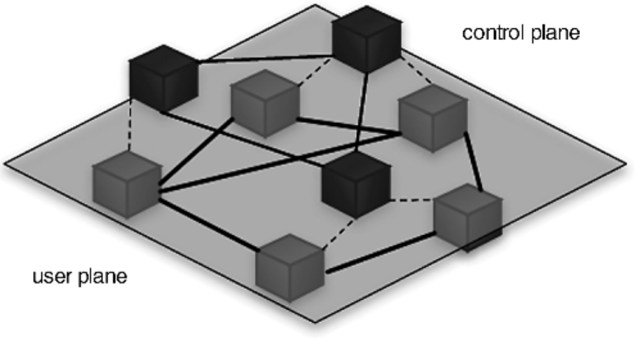

Figure 1.16 The CCITT signaling system 7 concept.

one-to-one to the switches of the user plane network and colocated with them; there is an actual connection between them via suitable interfaces; in the diagrams they are usually represented with circles.

- *Signaling Transfer Point* (STP) nodes, which are the core of the signaling network and their purpose is to switch the signaling messages in the network; in the diagrams they are usually represented with squares.

Some equipment can act as both SP and STP at the same time.

The links connecting the nodes of the CCS network are called *signaling links* (SL) and are commonly made of ISDN (e.g., 64 kbit/s) links, which are always on. The SL are usually represented with dashed lines.

Figure 1.17 shows an example of a network with the CCITT signaling system 7.

Figure 1.18 shows the CCITT signaling system 7 protocol stack. It is made of different components and in its specification divided in four layers, shown in Figure 1.18 and compared again in Figure 1.18 with the ISO/OSI layers definition. According to this latest definition the first three are handled by the STPs, whereas level 4 (i.e., all the rest) is end-to-end and performed by the SPs.

Without going into full details, we now go through the different protocols bottom up. The *Message Transfer Part* is the part of the protocol in charge of delivering and routing the messages from and to the different entities. The SS7 level 4 is basically divided in two parts: one related to a kind of "signaling-in-the signaling" control plane (*control part*) and one related to the "user plane" (*user part*) in the signaling part. The first one is needed to establish, control and tear down the logical connection needed for the signaling and comprises the *Signaling Connection Control Part* (SCCP), the *Intermediate Service Part* (ISP) and the

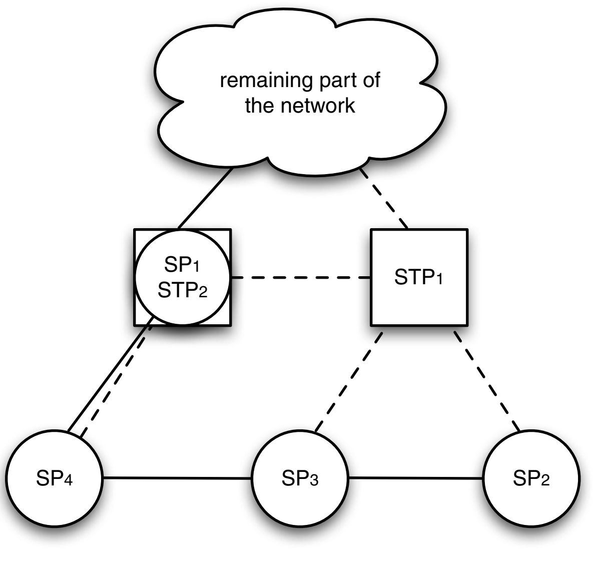

Figure 1.17 Example of a network using the CCITT signaling system 7.

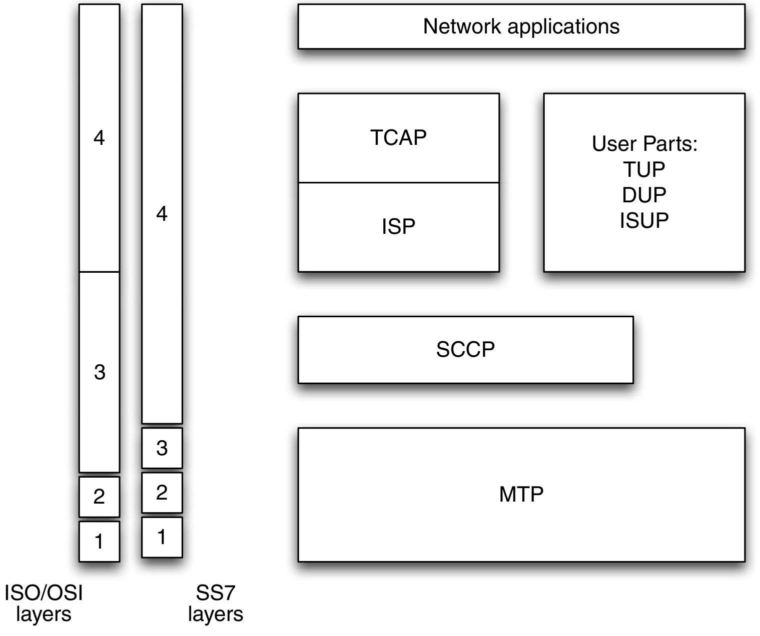

Figure 1.18 CCITT signaling system 7 Protocol Stack.

Transaction Capability Application Part (TCAP). The user plane instead depends on the specific network and can be the plain *Telephone User Part* (TUP), the *Data User Part* (DUP) or finally the *ISDN User Part* (ISUP).

1.5.4 PDH Networks

The plesiochronous digital hierarchy (PDH) is a technology used in telecommunications networks to transport large quantities of data over digital transport equipment such as optic fibers and microwave radio systems. The word "plesiochronous" comes from the Greek, more precisely from $\pi\lambda\eta\sigma\iota o\varsigma$ (near) and $\chi\rho\acute{o}\nu o\varsigma$ (time), so it means "more or less at the same time." In other words, operations are not perfectly synchronous but are not asynchronous either.

The European *E-carrier system* is described below. The basic data-transfer rate is a data stream of 2048 kbit/s (which approximately corresponds to 2 Mbit/s if a megabit is considered as 1024 kilobits). For speech transmission, this is broken down into thirty 64 kbit/s channels plus two 64 kbit/s channels used for signaling and synchronization. Alternatively, the whole 2 Mbit/s may be used for nonspeech purposes, for example, data transmission. The exact data rate of the 2 Mbit/s data stream is controlled by a clock in the equipment generating the data. The exact rate is allowed to vary by some small amount ($\pm 5 \cdot 10^{-5}$) on either side of an exact 2.048 Mbit/s. This means that multiple 2 Mbit/s data streams can be running at slightly different

Table 1.1 PDH hierarchies: in brackets the rate in number of basic user channels at 64 kbit/s, DS stands for Digital Signal and E for Europe.

	North America	**Japan**	**Europe**
Level zero	64 kbit/s (DS0)	64 kbit/s	64 kbit/s
First level	1.544 Mbit/s (DS1) (24) (T1)	1.544 Mbit/s (24)	2.048 Mbit/s (32)
Intermediate level	3.152 Mbit/s (DS1C) (48)	–	–
Second level	6.312 Mbit/s (DS2) (96)	6.312 Mbit/s (96), or 7.786 Mbit/s (120)	8.448 Mbit/s (128) (E2)
Third level	44.736 Mbit/s (DS3) (672) (T3)	32.064 Mbit/s (480)	34.368 Mbit/s (512) (E3)
Fourth level	274.176 Mbit/s (DS4) (4032)	97.728 Mbit/s (1440)	139.264 Mbit/s (2048) (E4)
Fifth level	400.352 Mbit/s (DS5) (5760)	565.148 Mbit/s (8192)	565.148 Mbit/s (8192) (E5)

rates. In order to move several 2 Mbit/s data streams from one place to another, they are combined that is, multiplexed in groups of four. This is done by taking one bit from stream number 1, followed by one bit from stream number 2, then number 3, then number 4. The transmitting multiplexer also adds additional bits in order to allow the far-end receiving demultiplexer to distinguish which bits belong to which 2 Mbit/s data stream, and so correctly reconstruct the original data streams. These additional bits are called *justification* or *stuffing* bits. Because each of the four 2 Mbit/s data streams is not necessarily running at the same rate, some compensation has to be made. The transmitting multiplexer combines the four data streams assuming that they are running at their maximum allowed rate. This means that occasionally (unless the 2 Mbit/s is really running at the maximum rate) the multiplexer will look for the next bit but this will not yet have arrived. In this case, the multiplexer signals to the receiving multiplexer that a bit is missing. This allows the receiving multiplexer to correctly reconstruct the original data for each of the four 2 Mbit/s data streams, and at the correct, different, plesiochronous rates. The resulting data stream from the above process runs at 8448 kbit/s (about 8 Mbit/s). Similar techniques are used to combine four 8 Mbit/s plus bit stuffing, giving 34 Mbit/s.[5] By taking four 34 Mbit/s flows and adding bit stuffing, a 140 Mbit/s flow is achieved. Four 140 Mbit/s flows are further combined (again, plus bit stuffing) into a 565 Mbit/s flow. This is the rate typically used to transmit data over an optic fiber system for long-distance transport. Recently, many telecommunication companies have been replacing their PDH equipment with synchronous digital hierarchy (SDH) equipment capable of much higher transmission rates.

The Japanese and American versions of the PDH system differ slightly in minor details, but the principles are the same. Table 1.1 shows the details of these hierarchies.

[5] In this and in the following computations, it is implicit that bit stuffing increases the amount of bits exchanged. For example, 34 Mbit/s is strictly larger than what is coming from the four 8 Mbit/s flows.

1.5.5 SDH Networks

Synchronous networking differs from PDH in that the exact rates that are used to transport the data are tightly synchronized across the entire network. This synchronization system allows entire intercountry networks to operate synchronously, greatly reducing the amount of buffering required between elements in the network. Both SDH and synchronous optical networking (SONET) (the North American counterpart of SDH, which is mainly a European technology) can be used to encapsulate earlier digital transmission standards, such as the PDH standard, or directly employed to support either ATM or so-called Packet over SONET/SDH (POS) networking. As such, it is inaccurate to think of SDH or SONET as communications protocols in and of themselves, but rather they should be thought of as generic and all-purpose transport containers for moving both voice and data. The basic format of a SDH signal allows it to carry many different services in its Virtual Container (VC) because it is bandwidth-flexible.

The basic unit of framing in SDH is a STM-1 (Synchronous Transport Module level – 1), which operates at 155.52 Mbit/s. SONET refers to this basic unit as a STS-3c (Synchronous Transport Signal – 3, concatenated), but its high-level functionality, frame size, and bit-rate are the same as STM-1. SONET offers an additional basic unit of transmission, the STS-1 (Synchronous Transport Signal – 1), operating at 51.84 Mbit/s – exactly one-third of a STM-1/STS-3c. Some manufacturers also support the SDH equivalent STM-0, but this is not part of the standard.

SONET and SDH are both using a frame-based transmission. The frame is organized by grouping eight bits at a time (octet-based) and is represented as a bidimensional matrix, organized in rows and columns, where some parts are devoted to overhead and the rest is left for actual payload. This bidimensional payload is then scanned by a pointer, which selects the octet to be transmitted.

In Table 1.2 the details of the SONET and SDH hierarchies are given. In Figure 1.19 a typical architecture for an SDH network is shown: the main equipment is the *Add-Drop Multiplexer* (ADM) which is capable of extracting from an higher rate stream a slower rate one and, at the same time, inserting a slower rate stream into a higher rate one.

Table 1.2 SDH and SONET hierarchies.

SONET optical carrier level	SONET frame format	SDH level and frame format	Payload kbit/s	Line rate kbit/s
OC-1	STS-1	*STM-0*	50 112	51 840
OC-3	STS-3	STM-1	150 336	155 520
OC-12	STS-12	STM-4	601 344	622 080
OC-24	STS-24	–	1 202 688	1 244 160
OC-48	STS-48	STM-16	2 405 376	2 488 320
OC-192	STS-192	STM-64	9 621 504	9 953 280
OC-768	STS-768	STM-256	38 486 016	39 813 120
OC-3072	STS-3072	STM-1024	153 944 064	159 252 240

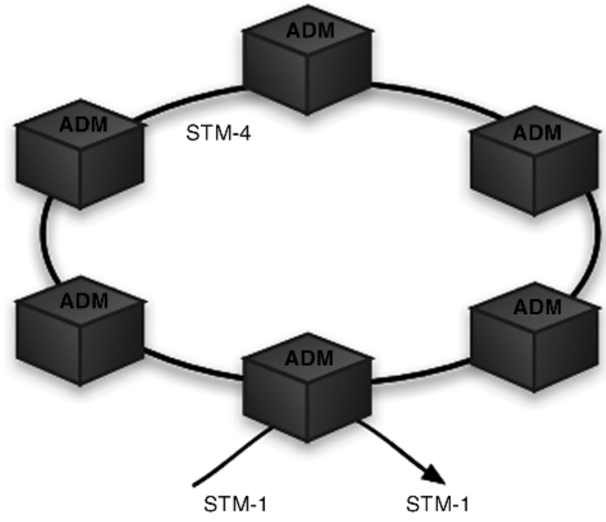

Figure 1.19 SDH ring with ADMs.

References

1. Hopkins, B. and Wilson, R. J. (2004) The Truth about Königsberg, *College Mathematics Journal*, **35**, 198–207.
2. Alexanderson, G. L. (2006) Euler and Königsberg's bridges: A historical view, *Bulletin (New Series) of the American Mathematical Society*, **43**, 567–573.
3. Skype, `www.skype.com.`
4. van Bosse, J. G. (1998) *Signaling in Telecommunication Networks*. John Wiley & Sons, Inc., New York, NY.