

2 Infrastructure

Infrastructure is a term used for a variety of things. The most common contexts for the term in everyday life are roads, water and wastewater, electricity and telecommunication. The basis for an EKI is the network infrastructure. This starts with laying cables into the floors and/or the walls of organization's buildings or setting up base stations for a wireless network. Today, a medium-sized company building needs several kilometers of cable in order to provide a good network infrastructure for every workplace. Besides the computer network for data communication today there is also still a telephone network for voice communication that forms a separate infrastructure. Although both forms seem to amalgamate in the near future, as current IP telephony and data communication over telephone lines indicate, we will concentrate our considerations on computer networks. Theoretical foundations for computer networks including a layered architecture are presented in section 2.1. Section 2.2 then discusses standard network protocols that are implementations of the layers. Based on the basic network infrastructure, services for data storage, access, messaging and security are offered (section 2.3). More recently, many organizations have also installed an application infrastructure (section 2.3.3) that provides services needed in most enterprise applications (e.g., transaction support).

Overview

On completion of this chapter you should be able to

- describe the most important network standards for PANs, LANs, MANs and WANs,
- use a conceptual model that distinguishes several layers within a network that provide different services and build upon each other,
- identify the most important network protocols for the Internet and categorize them according to layers in the model,
- distinguish several network devices and explain their usage,
- describe the most important technologies for storage, access, messaging and security,
- explain the functionality of application servers and explain how they support enterprise applications.

Learning objectives

2.1 Network Infrastructure

A classification of computer networks helps to understand the features networks can provide. We will first present a list of several classification criteria before we discuss two criteria, topology and geographical expansion, in detail. The section concludes with illustrating the ISO/OSI reference model, a layered architecture for computer networks.

Classification of computer networks

Computer networks can be classified according to a variety of criteria (Table 2-1). The criteria focus either on technical and physical aspects, e.g., the way of transmission, type of media, topology and scale of a network, or on organizational aspects, i.e. who owns and operates the network or who is allowed to access and use the network.

Table 2-1. Criteria for classifying networks

crit er ion	descrip tion	exam ple
transmission	mode of transmission	broadcast, point-to-point
topology	basic structure of the physical network	ring, bus, star
scope	geographical expansion of the network	PAN, LAN, WAN
network owner	institution that operates the network	company network, Internet, Value Added Network
user group	group of users that has access to the network	Internet, Intranet, Extranet
function	purpose of the network	front-end network, back-end server network, backbone
cost	cost for hardware or lease	50 € per month for ADSL
media	type of physical media	fiber optics, copper cable, infrared light, radio
transmission protocol	structure of messages, message passing, etc.	Ethernet, Token Ring, ATM
performance	network bandwidth	low (< 1 MBit/s), medium, high (> 1 GBit/s)

Network owner

The distinction regarding the owner of a network is especially important for wide-area networks since LANs are usually owned by the company that also uses the network. The Internet is an open network that consists of a large number of networks operated by different organizations, mostly large telecommunication provider, but also governmental or scien-

tific organizations. A participant only has to make a contract with one network owner connected to the Internet in order to get access to the whole network. In contrast to that, value-added networks are usually operated by a single organization that connects business partners by providing specialized data exchange services. This is traditionally mostly EDI data, but increasingly more XML data to support business processes.

A more technical distinction can be drawn according to the function of the network (part). A front-end network connects clients to the LAN, while the network that connects the servers is referred to as back-end network. This distinction also effects the required speed of the network. While it's usually sufficient to connect clients with a 100 MB/s connection, the back-end should be faster in order to cope with the traffic resulting from requests from and answers to a large number of clients working with services offered on the servers (e.g., 1 GB/s Ethernet). The term backbone refers to the core network that connects several front- and back-end networks with each other and therefore builds the central component in a network. It is often organized in a ring topology and has the highest demand for fast components so that 10 GB/s Ethernet is desirable.

Cost and performance are usually closely related both in terms of leasing cost for access to public networks and in terms of hardware cost for network components to build a private network. The faster the network should be, the higher the cost. Two examples should point that out. While a 2 MB/s leased line is around 250 € per month a 8 MB/s line is more than four times more expensive. Similar to that, simple 24 port switches that allow to connect 100BaseT Ethernet devices are available starting around 100 €. Enterprise switches with 24 Gigabit ports and high switching performance cost around 2000 €.

Other criteria shown in Table 2-1 will be discussed in conjunction with network standards in section 2.2 and infrastructure services in section 2.3.

Network function

Cost and performance

2.1.1 Topologies

Machines linked in a network communicate by sending messages through a communication medium. In analogy to the traditional mail system these are called packets. The actual content of the message is wrapped in transport data about sender and receiver as well as data to control the transport process. In this section, alternatives to structure communication networks are explained with regard to physical connections between machines. A topology represents the physical communication connections (edges) between machines (nodes) in a network.

The architecture of a network is fundamentally determined by the mode of transmission which can be broadcasting of messages or point-to-point communication. In broadcast networks, all nodes are connected to the same physical medium. Messages sent by one machine are received by all others, although they can either be addressed to one single communication

Broadcast networks

partner (*unicast*) or to all nodes in the network (*broadcast*). Some networks allow addressing of messages to a subset of the network which is termed *multicasting*. A main challenge in broadcast networks is how to utilize the network medium most efficiently which largely depends on management of media access.

Point-to-point networks

In point-to-point networks (sometimes also called *peer-to-peer* networks), only pairs of machines are connected through a physical medium. Distant communication partners thus regularly will not be connected directly to each other. In this case, a message needs to be transmitted via intermediate stations that receive and forward the message to its destination. A challenge in this class of networks is how to find the best out of multiple alternative routes that connect source and destination of the message. In contrast to broadcast networks, point-to-point networks are only capable of *unicasting*.

Topologies of broadcast networks. Bus and ring networks are two basic types of topologies of broadcast networks which are often combined in practice, e.g., a ring network connects multiple bus networks.

Bus network

The left hand side of Figure 2-1 shows a bus network. All nodes are connected to a shared communication medium, e.g., a copper cable, that passively transmits messages in both directions. No routing or forwarding of messages is necessary because every connected station receives all messages.

Ring network

In a ring network, every node is connected to exactly two other nodes. A message is passed only in one direction from one node to the other until its destination is reached. If one station breaks down, the whole communication is blocked. To enhance the reliability of the network, redundant communication channels can be established that allow bypassing defect machines. The nodes usually regenerate the physical network signal with the effect that network size is not limited with respect to the number of participating machines, but by length of the connections between two nodes.

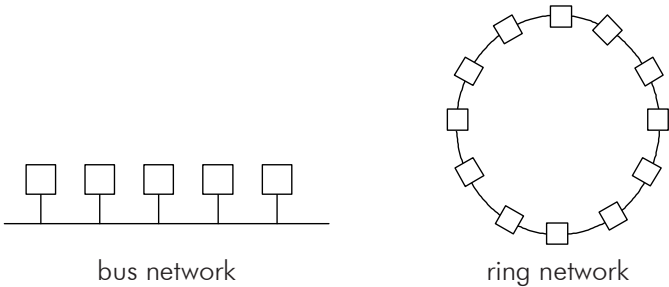


Figure 2-1. Topologies of broadcast networks

Topologies of point-to-point networks. Four basic architectures of point-to-point networks can be distinguished (Figure 2-2):

In a star network, all nodes are physically connected to a central node that handles the entire communication between all nodes. The advantage of this topology is that maintenance and control of the network are simple, because it can concentrate on the central node. A major disadvantage is the dependence of the whole network on performance and availability of the central node.

In a tree network, communication between two nodes always runs over hierarchically superordinated nodes. This architecture thus can be seen as to be composed out of interconnected star networks. Network control is performed by superordinated nodes. Thus, the tree network has the same advantages and disadvantages as the star network: easy administration, but dependence on central nodes.

In a mesh network, every node is connected to two or more other nodes. If all nodes are directly connected to each other, we speak of completely intermeshed networks. Mesh networks are very reliable and can grow without central control. A disadvantage of this architecture is that it can be complex to find the best route between source and destination.

In a loop network, every node is connected to exactly two other nodes. Every single node controls network traffic. Thus, network control is more complex than in the case of centralized network architectures such as star or tree networks. In contrast to a ring network, communication between nodes can happen in both directions and communication media not necessarily have to be of the same type. The nodes in a loop network play a more active role than in a ring network as they forward messages and not just refresh physical signals.

Star network

Tree network

Mesh network

Loop network

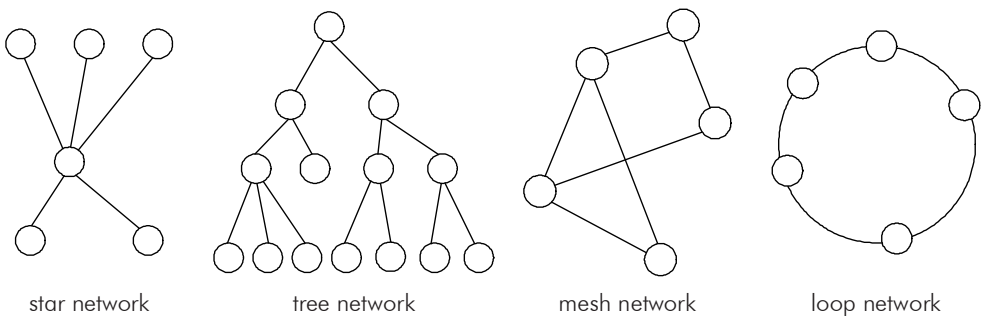


Figure 2-2. Topologies of point-to-point networks

2.1.2 Geographical Expansion

Networks can also be classified according to their scope, i.e. the geographical area they cover. Table 2-2 gives an overview of commonly distinguished classes of networks with respect to their geographical scope.

Table 2-2. Network classification according to scale

interprocessor distance	location examples, network for ...	network class
1 m	workplace	personal area network (PAN)
10 m	conference room	
100 m	company building	local area network (LAN)
1 km	university campus	
10 km	city	metropolitan area network (MAN)
100 km	country	
1000 km	continent	wide area network (WAN)
10.000 km	planet	
		the Internet

Personal area network (PAN)

Personal area networks (PANs) connect devices of a single person. Often, the term implies the use of some form of wireless technology to connect e.g., personal digital assistants (PDA), notebooks and cellular phones, or to connect a PC to a printer and a scanner.

Body area network (BAN)

The term body area network (BAN) is sometimes used to denote very short ranging personal area networks between components of a wearable computer, e.g., computer system that is integrated into a jacket.

Local area network (LAN)

Local area networks (LANs) are used to share hardware and software resources within a workgroup or an organization. They are restricted in size and usually span a single building, site or campus. Although geographic expansion of small LANs and wide reaching PANs may overlap, there is a clear distinction as LANs connect several computers belonging to different people, e.g., in a work group whereas PANs connect several devices belonging to one person.

Metropolitan area network (MAN)

A metropolitan area network (MAN) covers a larger area within a city or region. The term is used in two contexts. One denotation refers to the interconnection o separate LANs of a single organization via leased lines to build a virtual LAN that covers all buildings of the organization on a campus or within a city. The second denotation refers to a network that provides centralized services like Internet access (e.g., with WLAN access points) or cable television to private homes. A recent development is high-speed wireless Internet access via MANs (IEEE 802.16 standard, UMTS).

Wide area network (WAN)

A wide area network (WAN) enables communication with very high bandwidth (e.g., 10 GBit/s) over large distances, e.g., between states,

countries or continents. It basically consists of hosts, e.g., servers, switches, routers, and a communication subnetwork that connects them. The network usually is owned and operated by a telephone company, an Internet service provider or by public authorities. Challenges in WANs are bandwidth management, cost accounting, scalability and high reliability.

Last, but not least the Internet is the network of interconnected networks spanning all continents of our planet (section 2.3.2, 126ff).

Internet

2.1.3 Layered Network Architecture

Computer networks can be seen as hierarchical systems with several layers. Each layer provides certain services to the higher layers which leads to an increasing degree of abstraction for higher layers. The higher layer can therefore be recognized as a service consumer whereas the layer below can be seen as a service provider. The specification of the service is the interface between both layers. There exist a number of layered architectures of which the ISO/OSI reference model is the most important one from a theoretical perspective. The Open Systems Interconnection (OSI) model has been defined by the International Standards Organization (ISO). It specifies seven layers that help to understand network systems (Figure 2-3).

ISO/OSI reference model

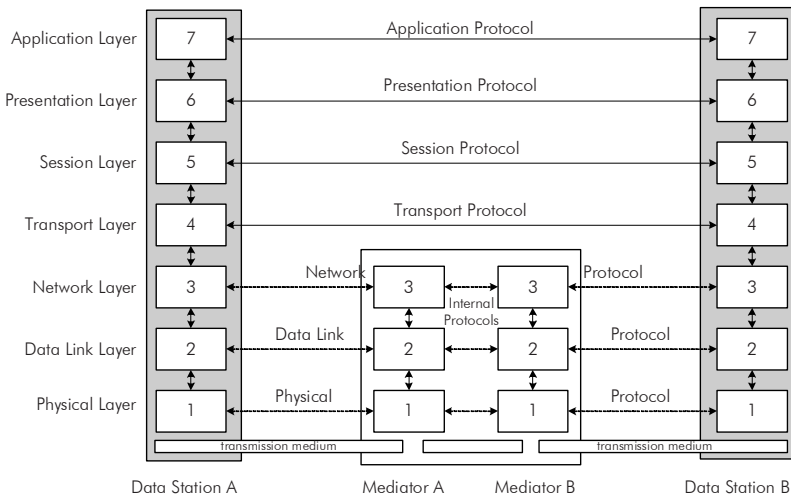


Figure 2-3. ISO/OSI reference model

The physical communication in this model is top-down from the highest to the lowest layer on the sender side, then horizontal over the network media and then bottom-up from the lowest to the highest layer on the receiver side. From a conceptual point of view, each layer on sender side communicates with the layer on the same level on receiver side.

Example

Imagine the owner of a small company in Wellington, New Zealand, who recently met a business man from Munich, Germany, on an industry event in the USA. When she comes home from the event, she wants to send him a message, but she does not know his address details. All she knows is the name of the man and the company he works for. (a) So she gives her secretary the message she wants to send and the instruction to transfer it as soon as possible. (b) The secretary looks up the address details, puts the message into an envelope and puts the address on the envelope. She is not sure about the correct value of stamps she has to put on the envelope, so she gives the letter to the post office of the company. (c) In the post office, they look up the rate and put stamps on the envelope. Then, they hand it over to a logistics provider that handles all the letters and parcels for the company. They do not know on which way the letter will travel to Germany, nor do they care. (d) The logistics provider handles that on its own. The letter goes by truck to the central post office and from there to Auckland airport. The plane with the letter leaves to Singapore, which is a central trade center for passengers as well as goods and flies to Frankfurt/Main airport in Germany. The message moves on to Munich by train until it reaches the post office of the company at its destination. (e) There, the clerk reads the address on the envelope and brings it to the secretary of the receiver (f) who removes the envelope and finally (g) hands it over to the business man.

The company owner back in New Zealand thinks that she has communicated directly with the business man in Germany. In fact, she has only communicated directly with her own secretary.

ISO/OSI layers

The same system applies to the ISO/OSI reference model. Each layer has a distinct purpose that will briefly be described in the following section. The application layer (7) gets data that has to be communicated from the application. The presentation layer (6) is concerned with syntax and semantics of transmitted data. It uses high order data structures (e.g., bank account records). The session layer (5) handles session information and time-outs, so that user-specific information is only saved as long as applications need it. The transport layer (4) provides a virtual channel for communication that can either be connection-oriented or connection-less. The former means data can rely on data passed on earlier. The latter means data is transmitted according to fire-and-forget mechanisms. Connection-oriented transportation can be thought of as having the telephone system as role model. A user picks up the phone, dials a number, communicates with the remote person and hangs up. Connectionless transportation on the other hand can be thought of as having the postal system as role model. Each message/letter carries the full address and is routed through the system independently. The advantage of connectionless transportation is fault tolerance, whereas connection-oriented transportation allows for billing and a guaranteed level of quality (quality of service, QoS). The network

layer (3) is responsible for identification of communication partners which is implemented via unambiguous identification addresses. The data link layer (2) takes care of reliable reconstruction of the transmitted signal by adding checksums to the data. Finally, the physical layer (1) provides access to the transmission medium by modulating the signal.

In between two data stations, there can be several mediators that refresh the signal and route the network packets. They may use different protocols to communicate with each other. Real end-to-end communication is only established on layers four and above.

Mediators

2.2 Network Standards

The following sections discuss concrete implementations for different layers of the ISO/OSI model. These implementations are standardized network protocols. Afterwards, we also present network hardware which can be categorized according to the ISO/OSI layers they are based on.

2.2.1 Physical and Data Link Layer

Starting from the bottom, transmission media which can be wired or wireless have to be examined. For wired media, fibre channel and copper cables can be distinguished with copper cables being further divided into twisted pair, coaxial and power cable. The different types of copper cables vary in cross-section and shielding, which are the main characteristics that influence attenuation and liability to interference. Today, copper is mainly used for end-user connectivity. Most long range cables are fiber optical because of better transmission characteristics. With more and more fiber optical cables produced, costs decrease so that this material becomes affordable and is also used for connecting servers in order to satisfy increasing bandwidth demands. This leads to a further increase in production and decreasing costs so that it seems only a matter of time until fiber optical cables are used to connect end-user computers (classification according to function in Table 2-2 on page 88).

*Media, cable,
fiber, copper*

Wireless media can be divided into radio connections and optical connections that differ not only in wavelength of the signal (1-5 GHz for radio vs. 3-300 THz for optical connections), but also in their diffusion model. Where optical methods usually use point-to-point connections, radio frequency emitting devices usually send in all directions (broadcast). Both types have two important sub-types: terrestrial radio transmission and satellite transmission for radio connections as well as infrared light and directed laser for optical connections.

*Wireless
media, radio,
optical*

Table 2-3 gives an overview of the network standards discussed in the next section with respect to geographical expansion of networks explained in section 2.1.2, 88ff.

Table 2-3. Overview of network standards

	cable-bound	wireless
PAN	USB, Firewire	IrDA, Bluetooth
LAN	Ethernet, Token Ring	WLAN, DECT
MAN and WAN	ATM, FDDI, X.25, FrameRelay, Sonet/SDH	GSM, GPRS, EDGE, HSCSD, UMTS, UWB

USB

Protocols for personal area networks. There are two important standards each for wired and wireless connections between devices in personal area networks. The purpose of these standards is to provide easy connectivity, low implementation costs and small yet robust physical interfaces. For wired connections, USB (universal serial bus) is the dominating standard and can be seen as a successor of the serial RS/232 interface. It is used to connect peripheral devices, e.g., pointing devices and keyboards, as well as digital cameras, scanners and printers to computers. It can also be used to connect peripheral devices with each other (USB2go), although this is still rarely the case. USB version 1.1 allows transmission with up to around 1 MBit per second. Version 2.0 is downwards compatible and specifies transfer speeds up to 480 MBit per second. Version 3.0 further speeds up data transfer rates to 4.8 GB/s by introducing additional connection pins and using an additional optical connection while keeping backwards compatibility. First devices implementing the new standard are expected to be available in late 2008.

Firewire

A competing standard for USB 2.0 is the IEEE specification 1394, commonly known as iLink (Sony) or Firewire (other companies). It is mainly used for connecting external high-speed devices like DVD burners or hard disks to computers (especially by Apple Computer, Inc.) and in the video industry for connecting digital video camcorders to computers. The standard can be used for transfer speeds up to 400 MBit per second (1394a). A successor with doubled transfer speed has been specified as 1394b.

IrDA

Wireless connections in PANs are still mainly realized via infrared light signals based on the IrDA standard (Infrared Data Association). Most devices available only support version 1.0 that provides transfer rates of 115 kBit per second. There are also two faster versions that provide 4 MBit per second (version 1.1, also called Fast IrDA) and up to 16 MBit/s (version 1.2, sometimes called Very Fast Infrared, VFIR). However, both are rarely used. All versions have a signal range of 1-2 meters. Mobile phones are the largest group of devices that heavily rely on IrDA for wireless connection to computers and especially notebooks. Optical connec-

tions rely on a direct line of sight between both communication partners. This leads to easy interruptions in the communication, e.g., by shakes that move or rotate the mobile phone just a little so that connection to the computer gets lost.

Bluetooth, a radio standard for PANs, becomes more and more adopted, because radio transmissions are much easier to handle. It is designed to facilitate any kind of wireless connections, e.g., connect keyboards and mice to computers, connect headsets to mobile phones, or connect digital cameras to printers in order to directly print photos. Bluetooth was specified by the Bluetooth Special Interest Group (BSIG) initiated in 1998 by Ericsson, IBM, Intel, Nokia and Toshiba. Version 1 was designed to connect devices within a 10-100 meters range with 1 MBit/s transfer rate and operates at 800 mW transmission power in the 2.4 GHz frequency band. It is a technique for ad-hoc connection of devices where one device declares itself as master and up to seven other devices can connect as slaves and form together a so-called *Piconet*. Devices can also take part in more than one Piconet simultaneously. The resulting overlapping Piconets form a so-called *Scatternet*. To support such different deployment scenarios as described above, every device has to support a number of profiles. A profile specifies a concrete set of protocols that span one or more layers. The profiles are divided into basic and advanced profiles (Table 2-4).

Bluetooth

Table 2-4. Examples for Bluetooth profiles

type	profile	typical purpose (connect ...)
basic	generic access	discovery of remote devices and services
	service discovery	discovery of supported profiles
	dial-up networking	notebook to mobile phone (Web access)
	LAN access	PDA to WLAN access point (PPP)
	generic object exchange	calendar items between PDAs (OBEX)
	synchronization	PDA to PC
advanced	basic imaging	scanner or digital camera to PC
	hands free	headset to mobile phone
	hardcopy cable replacement	PC to printer (formatted text and images)
	human interface device	mouse and keyboard to PC
	local positioning	GPS device to PDA

Despite its increasing dissemination, there are still some interoperability problems between devices of different vendors that are the result of inaccurate specification or sloppy implementation. Therefore, testing

whether two devices interoperate is principally advisable, even if both devices support the same profile.

*Bluetooth 2.0
EDR, Ultra
wideband,
NFC*

Two recent protocol standards are trying to overcome some problems of Bluetooth. The ECMA and ISO standard ultra wide band (UWB) is designed to significantly speed up data transfer rates of Bluetooth to up to 480 MB/s for a distance of 3m by using a large frequency range (3,1-10,6 GHz) and 110 MB/s for 10m. In order to avoid interference with other radio standards in parts of the frequency bands used, the transmitting power is very low (0.6 mW). The transmission method is called orthogonal frequency-division multiplexing (OFDM). This technology will be integrated into Bluetooth 3.0 which will follow up the current Bluetooth 2.0+EDR (enhanced data rates) with a speed of 3 MB/s (gross, 2.2 MB/s net). The other problem of Bluetooth is the robustness of the pairing mechanism of devices. Near-field communication (NFC) has been adopted as Bluetooth 2.1+EDR to overcome this problem and provide a fast and reliable pairing mechanism for two devices within a very short range (<20cm) that reduces pairing time from an average of 6 s to 0.1 s.

*Ethernet, IEEE
802.3*

Protocols for local area networks. The most important and most widely used technology for local area networks is Ethernet (IEEE 802.3). It is a cable-bound transmission standard developed by Xerox PARC in 1970, implements the bus topology (section 2.1.1) and supports different cable types. The Ethernet specification covers layer one and two of the ISO/OSI reference model (physical layer and data link layer). The transfer rate of the first Ethernet specification approved by the IEEE in 1985 is 10 MBit/s (10BaseT), which is not very comfortable for end user PCs any more. Servers that usually handle multiple connections to several end-users simultaneously need even more bandwidth. Thus, two other specifications with 100 MBit/s (100BaseT, Fast Ethernet) and 1 GBit/s (1000BaseX, Gigabit Ethernet) bandwidth have been developed. Recently, a 10 GBit Ethernet standard has been specified for usage in backbones. Modern company Intranets typically run 100BaseT with servers already connected via Gigabit Ethernet. The advantage of Ethernet is that all components necessary to establish a network (like NICs, cable, switches, section 2.2.5, 112ff) are reasonable priced due to their wide-spread use.

CSMA/CD

Ethernet uses a CSMA/CD (carrier sense multiple access with collision detection) method to access the medium, which is a competitive access method and operates as follows. The cable is checked for network traffic. The adapter sends if no traffic can be sensed, otherwise it waits for a random time interval until it tries again. After a signal has been sent, the cable is further checked for collisions with other signals sent simultaneously by other data stations. A strong jamming signal is sent if a collision is detected so that all connected devices are informed about the collision.

*Token Ring,
IEEE 802.5*

Token Ring (IEEE 802.5) is a network standard of theoretic interest due to its different mode, although it is not used that much any more. It has

been developed by IBM and implements a ring topology (section 2.1.1). A token is passed on from one data station in the network to the next one. Only stations that currently have the token are allowed to send. The first computer in the network that goes online generates the token. This is an example for a coordinated access method. The ring topology is only a logical structure and describes the way of the token. Physically, all data stations can also be connected with each other in a star structure. Supported transfer rates are between 4 and 16 MBit per second. Token Ring was popular 20 years ago but is increasingly replaced by Ethernet networks.

Wireless LAN (WLAN, IEEE 802.11 family) is the wireless equivalent to Ethernet. It is radio-based and has a range of approximately 30 meters within buildings and up to 300 meters outside. WLAN is a group of standards that operate in the 2.4 GHz frequency band (802.11b and 802.11g) and in the 5 GHz band (802.11a) respectively. It supports transfer rates of 11 MBit/s (802.11b) and 54 MBit/s (802.11a and g). Still not completely standardized, but already available in some draft implementations is 802.11n which achieves higher transfer rates with multiple antenna (MIMO, multiple input multiple output). The draft specification (draft 4 from May 2008) allows for up to 540 MB/s gross although current implementations reach only about 110 MB/s. Those three standards are supported by a set of extensions, adaptations and corrections (802.11d, e and f) as well as security standards (802.11i).

*WLAN, IEEE
802.11*

With good reason, security is still one of the main concerns for WLAN implementations. In contrast to cable-bound networks, WLAN communication signals can easily be received by anybody in reach of the radio transmission. The first security protocol designed for WLAN that should prevent network intrusion was *WEP* (*wired equivalent privacy*). However, it is considered insecure, since default configuration of most WLAN access points has WEP turned off, WEP passwords have to be manually configured for every single device and default key length is only 56 Bit. (See “WEP, WPA” on page 133).

WLAN security

Another challenge in WLANs is hand over from one access point to the next one when users move. This endeavor is called roaming (not to be confused with roaming in cellular phone networks) and should be as smooth as possible without disturbing the network connection.

Roaming

The only obligatory method for medium access that every WLAN implementation has to have is CSMA/CA (carrier sense multiple access with collision avoidance). As in Ethernet networks, the number of collisions rapidly grows with an increasing number of users on the same access point. Therefore, advanced methods like *RTS/CTS* (request-to-send/clear-to-send) and *PCF* (point coordination function) have been developed to overcome these problems. Since they are not mandatory for compliance to the WLAN specification, only a few manufacturers implement them in their devices.

CSMA/CA

DECT

DECT (digital enhanced cordless telecommunications) is a telephone standard for wireless telephones within company buildings, exhibitions or at home. It must not be confused with cellular phone standards like GSM (see "Protocols for cellular mobile networks" on page 97) that have a much larger range. DECT is designed for personal use, with every household operating its own base station (less than 100 €) whereas GSM base stations are operated by infrastructure providers (a GSM base station is about 10,000 €). DECT uses the frequency band between 1880 and 1990 MHz.

FDDI

Protocols for wide area networks. FDDI (fiber distributed data interface) is a network protocol mainly used for medium distances, e.g., in the backbone of a university spread across a city. As the name suggests, it uses fiber optical cable as medium. It employs a ring topology similar to Token Ring with a second redundant ring to guarantee availability of the network. Compared to data rates that are supported with Ethernet these days, FDDI seems to be relatively slow with data rates of 100 MBit/s for normal FDDI or 200 MBit/s for the full duplex version that uses the second ring simultaneously to the first. A successor called FDDI-2 better supports audio and video transmissions by providing reserved capacity and faster latency times (125 µs). The ring can be up to 100 km long with a maximum distance of 2 km between adjacent network stations.

Frame relay

A connection-oriented network that is widely replacing X.25 is frame relay. It is mainly suited for use in WANs, but can also be used to interconnect LANs, e.g., offices of a company in different cities can be connected in a cost-effective way by providing a secure private IP-based network in contrast to connections over the Internet. Frame relay has no error or flow control and thus is quite simple. It is packet-switched and mainly realized as PVC (permanent virtual circuit), although the specification also supports SVC scenarios (switched virtual circuit). The data rates reach from originally 56 kBit/s (V.34) up to 1.544 MBit/s (T1). Some providers also use special frame relay variants with 34 MBit/s or 45 MBit/s (T3).

ATM

A wide-spread protocol is called ATM (asynchronous transfer mode). It was designed to merge voice, data and cable-television networks and stands out due to its high data rates ranging from 155 MBit/s (OC-3) and 622 MBit/s (OC-12) up to 2.5 GBit/s (OC-48) that are in operation. There are also some 10 GBit/s lines (OC-192) in limited use, spanning up to 40 GBit (e.g., OC-48 with Wave Division Multiplex) in trials. One reason for the possibility of such high speeds is that ATM uses small, fixed size data packages called cells (53 byte) that can be routed in hardware (5 byte header). ATM provides flow control, but no guaranteed delivery due to lacking error control.

SONET/SDH

The commonly used protocol within public switched telephone networks (PSTN) is called SONET (synchronous optical network). There is also a set of CCITT recommendations called SDH (synchronous digital

hierarchy) which is similar to the SONET standard and will not be distinguished here for simplicity reasons. SONET has been designed to unify and replace the former PSTN protocols that were all based on 64 kBit pulse code modulation channels (ISDN) and combines them in different ways to get high speed connections, e.g., T1 equals 24 ISDN channels. A SONET frame is 810 bytes long and is sent 8,000 times per second. Since SONET is synchronous, frames are sent whether there is useful data or not. Sampling frequency of PCM channels in all digital telephone systems is exactly 8,000 Hz, so that there is a perfect match. A basic SONET channel, called STS-1 (synchronous transport signal, OC-1 on optical media) has a data rate of 51.84 MBit/s ($8,000 \text{ frames/s} \times 810 \text{ bytes/frame} \times 8 \text{ bits/byte}$). SONET channels can be combined to yield higher data rates. There are specifications for a number of speeds up to STS-192 (9.953 GBit/s).

Multi protocol label switching is a specification of a connection-oriented and packet-switched network protocol that operates on OSI layers 2 and 3 and is widely replacing dedicated line connection like frame relay in WANs. Its main features are to support multiple service models, traffic management and robust fault recovery. MPLS packets are of variable length and can contain an MPLS header with one or more labels that are used for fast routing of packets.

MPLS

Protocols for cellular mobile networks. Cellular mobile networks initially designed for voice telephony are increasingly used for data transfer.

The most widely used standard for mobile telephony world wide is GSM (*global system for mobile communication*). In February 2004, only 12 years after the launch of the first networks, more than one billion people (almost one sixth of the world's population) were using GSM mobile phones. GSM networks are digital and represent the 2nd generation of mobile networks, with the first generation represented by analogue networks like the A, B and C networks since 1958. GSM is a circuit-switched data (CSD) network and therefore connection-oriented as usual for networks designed for voice telephony. Data rate is 13 kBit/s for voice and 9.6 kBit/s for data connections which sums up to 22.8 kBit/s over all. The frequency range in Europe originally was 890-915 MHz for upload and 935-965 MHz for download known as GSM 900. More recently there is also a frequency range around 1800 MHz in Europe whereas the 1900 MHz band is used mostly in the US. GSM is a cellular network with cell sizes between 300 m in densely populated urban areas and 35 km in rural areas. Adjacent cells use different frequency bands and overlap in their coverage, so that an endeavor from one cell to the other can happen without interference to open connections. Every GSM user needs a *SIM card* (*subscriber identification module*) which is used for identification and to store data. GSM was mainly designed for voice telephony, but today more and more services on mobile phones require data transmissions.

GSM

GPRS

GSM offers insufficient bandwidth for data services and the connection-oriented protocol is not ideal for using IP-based Internet services. Therefore, several new protocols were developed that should supplement GSM to allow data communication with speeds that are equivalent to their wired counterparts. GPRS (general packet radio service) is a packet-switched protocol with higher bandwidth than GSM and the advantage that transferred data is billed instead of connection time. This reproduces a development from time-oriented towards volume-oriented billing methods in the history of wired Internet connections only a few years before. GPRS is combining time slots on demand and offers more efficient coding algorithms than GSM. Both methods together provide data rates of up to 170 kBit/s in theory, although in practice only 53.6 kBit/s are achieved typically. Another shortcoming of GPRS is that typical round trip times (RTT) are over one second, even for small network packets in networks with low load which is high compared to RTTs between 10 and 100 ms for wired connections. RTT is measured from the moment the client sends a request until the response from the server reaches the client. High round trip times lead to a loss in perceived quality and user satisfaction.

EDGE

With EDGE (enhanced data rates for global evolution), telecommunication providers made a last attempt to enhance the wide-spread GSM network before they had to build a completely new, expensive infrastructure for so-called 3rd-generation networks (3G networks). EDGE exists in two versions that enhance either HSCSD (ECSD) or GPRS (EGPRS). It replaces the inefficient GSM modulation method (Gaussian Minimum Shift Keying) with the highly optimized 8-phase shift keying. This leads to a theoretical maximum data rate of 384 kBit/s. However, since 3G networks are built up in many countries (or are already realized, e.g., in Austria), EDGE is not expected to be widely adopted in developed countries. A major drawback of EDGE is that the highest data rates can only be achieved in close proximity to the transmitting station. This drawback is even more obvious than in GPRS, which show the same symptoms.

UMTS

UMTS (universal mobile telecommunications system) is the European part of a family of 3G network standards specified in the IMT-2000 effort (IMT = international mobile telecommunications). It uses the two frequency bands 1920 to 1980 MHz and 2110 to 2170 MHz. Originally, IMT-2000 aimed at a global unification of mobile telephone systems, but it turned out that this was not feasible due to several reasons, e.g., political differences between countries. The new infrastructure necessary to enable full-blown UMTS is called UTRA (universal terrestrial radio access). It envisions two versions called UTRA/FDD with data rates up to 384 kBit/s and UTRA/TDD with data rates up to 2 MBit/s for asymmetric connections and 384 kBit/s for symmetric ones. Besides the infrastructure that depends again on geographic circumstances (rural vs. urban areas), traveling speed of users is another factor that determines the maximum data

rates that can be expected. A minimum data rate of 144 kBit/s is specified for traveling speeds up to 500 km/h whereas the maximum of 2 MBit/s can only be reached with speeds below 10 km/h and a good signal strength.

To further speed up download and upload, enhancements for the standard UMTS were developed that reorganize packet switching in order to use available channels more effectively. According standards are called High Speed Download (Upload) Packet Access (HSDPA / HSUPA) and allow for download rates of 3.6 or even 7.2 MB/s and upload rates of 1.8 and 3.6 MB/s depending on the infrastructure expansion stage. The maximum number of concurrent high speed connection was risen from 5 to 15 participants per cell.

UMTS was not only designed to provide higher data rates. Another main goal is to unify formerly separate standards like DECT, GSM, WLAN and satellite communication standards like Inmarsat. UMTS further specifies four different quality of service (QoS) classes:

- *conversational* for bidirectional services like voice and video telephony,
- *streaming* for unidirectional services like video on demand or radio,
- *interactive* for typical Internet applications with high data integrity demands and low latency,
- *background* for services like SMS or fax that require high data integrity, but do not require low latency.

Figure 2-4 sums up the discussion of cable-bound and wireless network protocols and shows their speed and range.

*HSDPA,
HSUPA*

*Communication
unification*

Summary

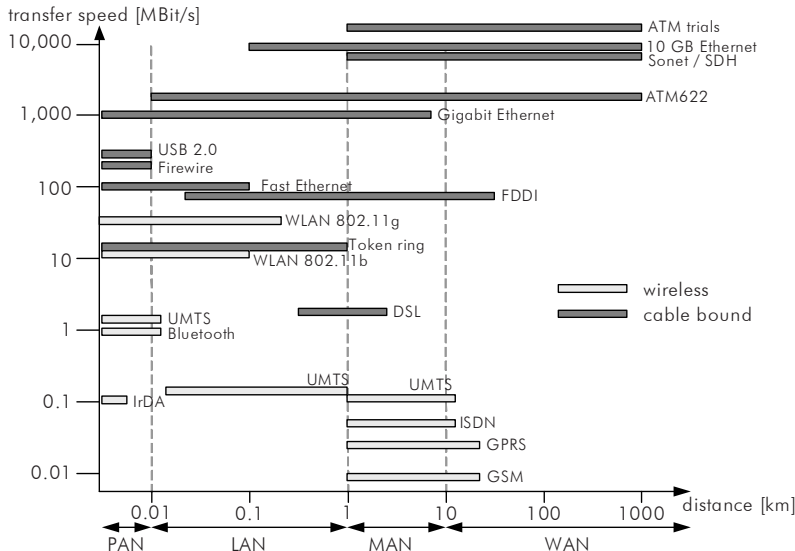


Figure 2-4. Protocols classified according to transfer rate and range

2.2.2 Network and Transport Layer

MAC address The basic mechanisms for identification of network participants and establishing network connections can be found on layers three and four of the ISO/OSI reference model (section 2.1.3, 89ff). The identification on the lowest level uses a so-called MAC address (media access control). This is a 48 bit number usually displayed in hexadecimal system consisting of a 24 bit manufacturer identification number and a 24 bit serial number of the device, e.g., the network adapter. That guarantees a world wide unique number. An example of a MAC address is 08-00-20-AE-FD-7E.

IP address However, a conceptual view of the network that enables logical partitioning into network segments is more suitable for network management. Therefore, a new identification method was established and the IP address (Internet Protocol) was introduced. IP address spaces can be assigned to organizations or organizational units independent of the hardware used. The address range 141.48.0.0 - 141.48.255.255 is assigned to the University of Halle-Wittenberg for example. When a server is replaced by a new one, the IP address stays the same although the MAC address of the server is different from that of its predecessor. IP addresses are 32 bit numbers consisting of four octets that build a kind of hierarchy (e.g., 141.48.204.242).

Network class The four octets are divided into a network address and a computer address. Depending on the number of octets used for network address, the network is called class A (first octet only), class B (first and second octet) or class C (first three octets) network. The number of octets used for computer address decides how many data stations can be part of the network (e.g., two octets equal two bytes, which allows for 2^{16} =65,536 data stations). To further clarify to which class a network belongs, the address space is also divided (Table 2-5).

Table 2-5. Network classes

network class	IP range	network mask	number of hosts
class A	1.0.0.0 - 127.255.255.255	255.0.0.0	16.7 million
class B	128.0.0.0 - 191.255.255.255	255.255.0.0	65,536
class C	192.0.0.0 - 223.255.255.255	255.255.255.0	256

Reserved address ranges Within these address spaces, a part is reserved for private use respectively and may not be used in the public Internet (1.0.0.0 - 10.255.255.255, 172.16.0.0 - 172.31.255.255 and 192.168.0.0 - 192.168.255.255).

Mindful readers may have noticed that there are some addresses left in the IP address space that were not mentioned yet. These addresses are reserved for special purposes. The range from 224.0.0.0 to

239.255.255.255 is reserved for so-called multicasts and the range from 240.0.0.0 to 255.255.255.255 is reserved for research and further extensions of the Internet.

Network participants usually communicate with each other one by one, so that there is one sender and one receiver. This way of communication is called *unicast*. However, in some cases it is useful to send network packets to more than one recipient at the same time which is called *multicasting*. The sender just sends a transmission to the multicast address and does not care how many participants get the message nor who gets it. A server on this address handles distribution of network packets to all data stations that have previously subscribed to the multicast. An application of multicasting is the transmission of a live video stream. As time is an important factor for live video transmissions, the most efficient way is to use multicasting, so that the sender sends the video data only once and does not have to wait for confirmation from every recipient.

Multicast

Besides unicasts and multicasts, there is a third form of network communication called broadcast. A *broadcast* is similar to multicast in that multiple recipients get the transmission of one sender. The difference is that recipients do not have to subscribe to the transmission, but recipients are determined based on the network segment. The last address of every segment is reserved for broadcasts that reach every network participant within that segment. So a broadcast to the address 192.121.17.255 would reach each network device in the 192.121.17.xxx network and a broadcast to 255.255.255.255 would reach each network device in the whole world. As the last case does not make sense due to huge network traffic usually broadcasts are not forwarded to participants outside of the network segment of the sender, so that both examples would lead to the same result. Limiting the range of broadcasts, which is a very popular mechanism, is an additional reason for administrators to further subdivide a network.

Broadcast

This is accomplished by using subnet masks. They have the form of an IP address with 4 octets but use in the simplest form only zeros or 255s for every octet. 255.255.255.0 is an example of a subnet mask that specifies a subnet with three octets for the network address and one octet for the computer address, which would divide a class B network into 255 subnets. The subnet mask has to take the network class into account, e.g., 255.255.0.0 is no valid mask for a class C network as there are already three octets identifying the network.

Subnet mask

Today, classful networks like described above are seldom used. Subdivisions that do not divide networks at the border of an octet but somewhere in between are more commonly used as they use given IP address ranges more efficiently. The term *classless interdomain routing* is used to denote it. The mask 255.255.255.192 (192 decimal = 11000000 binary) uses two additional bits to address the network compared to a class C network, which leaves 6 bits for the computer address. This separates the class C network into four subnets ($2^2=4$) with a maximum of 63 hosts each

CIDR

(00111111 binary = 63 decimal) one of which is the broadcast address. In CIDR notation, the network would be referenced by the starting address and the number of bits used for the network (e.g., 192.168.1.64/26).

Localhost

IP address 127.0.0.1 denotes the so-called localhost and always addresses the machine of the sender. Localhost is even more than a special address. It resides on a separate network interface that is different from the interface connecting to the outside and is called loopback interface. A computer could still reach itself on 127.0.0.1 if there is no network cable connected. In this way, proven mechanisms for communication can be used no matter if the receiving program resides on the same machine or a different one.

Hosts without IP address

The last IP address worth mentioning is 0.0.0.0 that can only be used as a sender address for network participants that do not have a valid IP address yet.

Internet protocol (IP)

The Internet protocol (IP) belongs to layer three of the ISO/OSI reference model (network layer) and its main purpose is identifying network participants and routing network packets from sender to receiver. It specifies addressing of network packets as described above and takes care of other issues like prioritization of packets, connection time-outs and preferred network routes. The version currently used in the Internet is version four (IPv4) which operates with 32 bit addresses ($2^{32} = 4.3$ billion). As the number of available addresses decreases dramatically with the number of Internet users continuously increasing, the need for a larger address space arose. This need will be satisfied with the implementation of IP version six (IPv6) that operates with 128 bit addresses which leads to an enormous amount of addresses (approximately 10^{24} addresses for every square meter on earth).

Routing

Routing means finding a way for packet from sender to receiver. This can involve several data stations that forward the network packet before it reaches its destination. Data stations that function as mediators (section 2.1.3, 89ff) are called *router*. Usually there is at least one router for every network segment border a packet is passing. For wide area connections, packets pass more stations within short distances at the beginning and the end of the travel. In between, there are a few stations with great distances in between them. In our example about the message from New Zealand to Germany it is same: first the message is routed from secretary to post office (short distance), later from Singapore to Frankfurt (large distance) and at the end from the secretary to the business man. Making the way from one data station to the next one is called a hop. Network packets are allowed to make only a limited number of hops before they have to reach their destination. This number is called time-to-live (TTL), can be specified by an administrator and usually is around 30.

ARP

As on lower layers still the MAC address is used for identification, there is a need for translation from IP- to MAC addresses and vice versa. This is accomplished by a translation service called address resolution pro-

tolocol (ARP) for IP to MAC translation, and by reverse ARP (RARP) for MAC to IP translation. Both ARP and RARP are network protocols.

Internet control and manipulation protocol (ICMP) is a protocol on the network layer and the basis for many small but useful tools that are usually part of the operating system. It specifies a number of messages and corresponding responses (e.g., echo). The most well-known application for ICMP is the `ping` command that exists on all operating systems providing network access. With the `ping` command, a user can check whether a host is reachable over the network by sending the ICMP `echo` message to it. If it is reachable, a response (`echo response`) is returned and time between sending and receiving the message is measured. If no response is received within a specified time frame (time-out), the partner is declared to be unreachable, even though it does not automatically mean the partner is really unreachable for any kind of network service (see firewalls in section 2.2.5, 112ff). A second useful application of ICMP is `tracert` (trace route) that helps to find out how many hops and over which routers a packet travels before it reaches its destination.

ICMP, ping

On layer four of the ISO/OSI-reference model, transmission control protocol (TCP) and user datagram protocol (UDP) fulfill the task of transportation. TCP is a connection-oriented protocol that establishes sessions for connecting to remote hosts. Data stored for the duration of the session (session data) is used for authorization and states.

TCP and UDP

Both, TCP and UDP specify 65535 ports. Every connection between two data stations has to specify one port each on sender and receiver side. On receiver side, usually a certain port specified for the service invoked is used. On sender side, any free port can be used. Port numbers from 1 to 1024 are reserved for well-known services on higher layers like HTTP on port 80 or FTP on ports 20 and 21 (section 2.2.3). These port assignments are just a convention for user convenience and no obligation. It is e.g., also possible to communicate over port 8000 or any other port using HTTP, but if the standard port 80 is used, the user does not have to specify the port explicitly as most client applications implicitly use port 80 for http if no port is specified. Port numbers above 1024 are free for use by any application. Despite this, there are some wide-spread applications that use certain default ports and it is good practice not to use one of those when developing an own networking application. Examples are the Oracle database management system that uses port 1521, the Microsoft SQL Server that uses port 1433, HTTP proxies that use port 8080 and many Java application servers (see "Application server" on page 138) that use ports 8000 or 8001.

Ports

Difference between TCP and UDP is that TCP is a connection-oriented protocol whereas UDP is connectionless. IP networks are packet-switched networks, which means that two network messages that are sent by the same sender immediately one after another to the same destination do not necessarily have to take the same route to the destination, nor is there any

*Differences
between TCP
and UDP*

guarantee that they arrive in the same order. For simple network communication, this is no problem as all messages take only one network message and there is no need for authentication or for keeping other session information. Those applications can use UDP as transport protocol that provides no error correction, confirmation for received messages or guaranteed order. TCP on the other hand, invests a substantial amount of protocol overhead¹ into establishing a virtual connection on top of IP in order to maintain data necessary to guarantee correct order of messages and exchange session IDs.

TCP connections

Establishing TCP connections takes place in three phases during which sequence numbers are exchanged to assure the connection. Every single message is confirmed by the receiver. If no confirmation is received within a specified time frame, the message is retransmitted. Another feature of TCP is that sending and receiving data at the same time is possible which is called *full duplex mode*. Additionally, it is possible to handle different higher-level protocols over a single TCP connection in one session which is called *multiplexing*. Finally, *flow control* enables TCP to automatically adjust its speed to different capacities at sender and receiver side.

Encapsulating and unwrapping data

Figure 2-5 shows how data is marked with headers on sender side as it passes down the layers (encapsulation) and how headers are interpreted and removed at receiver side (unwrapping). On the lowest layer there is also a checksum (CRC, cyclic redundancy check) that is used to check the integrity of the frame. The terms used to denote network packets also change from layer to layer. Whereas the term frame or cell is used on lower layers (1+2), we speak of packets on the Internet layer (3) and of messages on the transport layer (4).

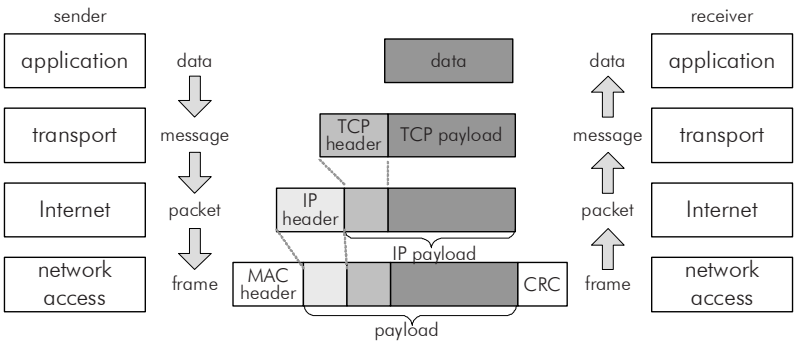


Figure 2-5. Encapsulating and unwrapping data in TCP/IP

¹ Overhead denotes header data to be transmitted together with the payload.

2.2.3 Application Layer

Standard Internet protocols are commonly used for accessing applications, so that the same mechanisms can be used for Intranet and Internet publishing of documents. Figure 2-6 recapitulates the protocols discussed in previous sections and gives an overview of some protocols that will be described in the next sections. Concrete protocols often implement multiple layers defined by the ISO/OSI reference model. In case of TCP/IP, there are only four layers called Internet layers.

ISO/OSI layer	Internet layer	concrete implementations							
		WWW	email	file	directory	host	name	network	IP address
7	Application		SMTP,	transfer	service	sessions	resolution	monitoring	assignment
6		HTTP	POP3,	FTP	LDAP	Telnet	DNS	SNMP	DHCP
5		80	IMAP	20/21	389	23	53	161/162	67/68
4	Transport	TCP					UDP		
3	Internet	IPv4, IPv6							ICMP
2	Network	Ethernet, ...		FDDI, ...		WLAN, ...		IrDA, ...	
1									
		Medium		copper		fiber		radio	

Figure 2-6. Internet protocols and ISO/OSI layers

One of the most important protocols is the *hypertext transfer protocol* (HTTP) that forms the basis of the *world wide Web* (WWW). Version 1.0 was specified in 1991 by the CERN institute in Geneva (current version: 1.1). Purpose of HTTP is delivering content in form of hyperlinked text which means HTML content. HTTP typically runs on port 80 of TCP connections. The basic mechanism is quite easy: a *request* for a resource is sent and the resource is being transferred back to the requestor (*response*).

To identify the resource a mechanism called *uniform resource identifier* (URI) is used, or more precisely a special form of URI called *uniform resource locator* (URL, section 3.1.2, 156ff). The structure of a URL is always the same, although there are some mandatory and some optional parts. An example for a concrete URL is `http://www.heise.de/news/index.html`

First part of a URL is the protocol. Examples are HTTP, FTP and file (file means local files on the requesting machine). If no protocol is specified, Web browsers use HTTP by default. A user name and password can be specified for connections that require authentication, as FTP connections usually do. An @-symbol separates the user name and password part

HTTP

URL

URLs in Web browsers

from the DNS name of the server (tld means top level domain). Any DNS name can be used here, as discussed in section 2.2.4, 110ff. Optionally, a port can be specified that is used on the destination machine to offer the service. If the port is omitted, the default port is being assumed, e.g., port 80 for HTTP or port 21 for FTP. Then a path to a file on the server follows which can contain folders and subfolders and usually ends with `html`. For user convenience, most Web servers support a mechanism called default file, which means that a certain Web page (often `index.html`) is being transmitted if the path includes only a folder, but no file. The complete URL structure as used in Web browsers is

URL structure `protocol://username:password@server.domain.tld/path`

GET, POST HTTP supports two methods for requests, `GET` and `POST`. Originally, `POST` was intended only for the rare cases where users fill in forms and send their content to the server, whereas `GET` was intended to be the default operation. Meanwhile, many Web sites are interfaces to complex applications and nearly every time these dynamic Web sites are refreshed, data is sent to the server (sometimes more, sometimes only a session id). A second form of sending data to the server used with the `GET` operation is called URL encoded sending of data. To accomplish that, the URL as discussed above is extended. A question mark “?” separates the path from parameters that are sent to the server. Parameters are listed in the form `parameter=value` and multiple parameters are separated with ampersands. Two parameters are encoded in the following example URL, `q` for the query term and `sourceid` for used browser.

Example of URL encoding `http://www.google.com/search?q=knowkom&sourceid=mozilla`

A request consists of a *header* specifying the request method, parameters and *payload*. Replies also have a header that contains a *status code* and contents of the requested file as payload. Status codes tell the requestor whether everything went fine (status 200 - OK) or there was an error (e.g., 404 - file not found). A new TCP connection is established for each request. There is also an encrypted version of HTTP called HTTPS (secure HTTP, section 2.3.2, 126ff).

FTP A second protocol important for the Internet is the *file transfer protocol* (FTP). It has been designed to transfer large files of any type, whereas HTTP was designed for (small) HTML files and a few images. Although it is good practice to put larger files that cannot be directly displayed within a Web browser on an FTP server, today a lot of *downloads* (e.g., installers, music files, office documents) are offered via HTTP. FTP uses port 21 to send control commands and list directory contents and port 20 (called `ftp-data`) to transfer files. Basic commands for FTP are `list` (to list the directory contents), `put` (to upload a file from the client to the server) and

get (to download a file from the server to the client). The `help` command gives an overview of all available commands.

Another major Internet protocol is the *network news transfer protocol* (NNTP). It is used for newsgroups where users can post messages to certain topics (section 4.2). NNTP uses TCP-connections on port 119. Similar to FTP, NNTP is partly replaced by HTTP with the advent of HTML-based newsgroups and discussion forums that become increasingly popular. A kind of unification of protocols seems to take place that could be driven by the popularity of Web browsers as central access application. FTP and NNTP both require specialized clients to unfold their full potential and thus their popularity decreases. NNTP

The result of the attempt to bring the Internet to mobile phones is a protocol called *wireless application protocol* (WAP). WAP is a family of protocols that reach from layer three (*wireless control message protocol*, WCMP) to layer seven of the ISO/OSI reference model (*wireless markup language*, WML). Since most modern smart phones can run web browsers that are able to access normal html-pages over http, the importance of WAP diminishes. WAP

Another mechanism that provides access to data over the network is network file access. It can be seen as network extension of file systems and abstracts from the underlying file system. The two most common network file systems are the NFS (*network file system*) used on Unix operating systems and SMB (*server message blocks*) used in Windows environments. Since Windows 2000, the SMB protocol has been enhanced to run over TCP/IP connections instead of requiring the proprietary NetBIOS protocol. This enhancement is called CIFS (*common internet file system*), although it is not commonly used in the Internet. A Unix application called Samba enables Unix servers to act as Windows file servers and most Unix derivatives also support access to Windows file servers. The purpose of both file system extensions is to provide access to files stored on file servers as if they were stored on local hard disks. In order to achieve this, users have to connect to network drives (Windows) or map folders on file servers to an entry point in the local file system (Unix, Windows XP). Other examples for network file protocols are Apple's AFP (Apple file protocol) and Novell's NCP (Netware core protocol). NFS, SMB

WebDAV (*Web distributed authoring and versioning*) is an extension to HTTP to enable read and write access to documents on remote servers, regardless of the operating system used and no matter whether they are simple file servers or more complex systems, e.g., DMS. The standard describes a number of useful functions and mechanisms and is still being extended, e.g., recently with the WebDAV access control protocol, but most implementations support only a small subset of the standard. Especially versioning functions are seldom implemented. Main functions of WebDAV are locking of files that are currently in use by a user, handling WebDAV

of meta-data (properties) and copy or move operations directly on the server. Recent enhancements also allow for versioning and access control.

Messaging. Data should not only be passively provided for access to somebody, but must sometimes be actively sent to a recipient. Communication can happen either synchronously (same time) or asynchronously (different time). Examples for synchronous messaging solutions are *instant messaging* (e.g., ICQ, MSN Messenger, Yahoo Messenger) and *internet relay chat* (IRC). Besides these popular (and somewhat standardized) possibilities there are also numerous proprietary applications that provide similar communication functions (e.g., Groove Virtual Office or IBM Notes Workbench, section 4.3, 285ff). In the following section we will review some of the more established asynchronous communication forms that are more closely related to infrastructure.

Email, SMTP

The most common form of asynchronous electronic communication is electronic mail, or short email. There are different network protocols on the application layer involved in sending and receiving emails. For sending them, the *simple mail transfer protocol* (SMTP) is used, which operates on TCP port 25 to communicate with the server. SMTP is really a simple protocol, so that users that want to try it on their own can directly communicate with the server in a telnet session if they use the following commands. `HELO`, to introduce yourself, `MAIL FROM:` to specify the sender address, `RCPT TO:` to tell the server the recipient's mail address, `DATA:` to start the message body and `SUBJECT:` to specify the subject. A single dot in a new line ends the message body and sends the mail. SMTP is also used to forward the mail from one mail server to the next until it reaches the final server that manages the post box of the receiver.

POP3, IMAP

Receivers can view contents of their post box and download mails with a protocol called *post office protocol version 3* (POP3, port 110). An alternative for POP3 introduced by Microsoft is the *interactive mail access protocol* (IMAP, port 143) which is a bit more flexible than POP3 (e.g., message synchronization between server and multiple clients). Common email servers are postfix and sendmail for Unix-based operating systems and Microsoft Exchange for Windows-based systems.

MIME

Besides the transport and retrieval of email, there also has to be agreement on content formats that should be used. For the text body of emails users can choose between plain ASCII text, rich text and HTML format. The MIME format (*multipurpose internet mail extensions*) is the most commonly used standard for entire messages. It specifies that a message consists of a header, the message body and zero to many attachments. A MIME header defines the encoding of a message (which character set is being used, e.g., ISO-8859-1 also known as Latin 1 or Western Europe encoding) and a content type (or MIME type) for each part of the message (e.g., `text/html`). Non-text parts are transformed from their binary for-

mat into a text format with an encoding known as *base64*. The MIME type is especially important as it is also used with HTTP.

A simplified and limited version of email is the mobile service SMS (*short message system*). Initially considered to be of minor importance by telecommunication providers, it is now one of the largest sources of income for them, especially in Germany with 25 billion SMS sent in 2003. An SMS is up to 160 characters long and is transmitted over free capacities of the signaling channel of mobile communication infrastructures, so that it does not interfere with voice and data connections. The successor of SMS is called EMS (*enhanced message service*) and enables users to transfer larger texts (up to 760 characters), small pictures and ring tones. Since MMS (*multimedia messaging system*) became available shortly afterwards, EMS has never been widely adopted. MMS supports multimedia objects in standard Internet formats instead of using proprietary formats like EMS, e.g., images in GIF and JPEG format audio files in MIDI and MP3 format and even video clips in MPEG format (section 5.2.4).

SMS, EMS,
MMS

A synchronous messaging mechanism is instant messaging. It became famous with Mirabilis ICQ (I seek you) application, which was acquired in 1998 by America Online and relaunched as AOL Instant Messenger (AIM). In 2008, the AIM/ICQ network is still one of the largest based on active users, but the Microsoft Messenger and Yahoo Messenger also have a very large user base. All three major players are based on proprietary protocols that have been frequently changed slightly in the past in order to make it harder for multi-protocol messenger or competing products to keep connected to the global infrastructure. In October 2004, the Extensible Messaging and Presence Protocol (XMPP) was published by the IETF as an open standard (RFC 2779). It is implemented in the open source application Jabber. Since then, XMPP-based applications like Google Talk continue to gain market share and impose pressure on proprietary applications. In 2008, both Microsoft and AOL published the specifications of their IM protocols.

Instant Mes-
saging

Many IM clients provide additional functionality for voice and video communication. Skype is a pioneer in this area and has gained a lot of attention since its foundation in 2003. It was bought by eBay in 2005, reached 100 million registered users in April 2006 and 300 million in Q1 2008 with over 10 million concurrently online. The proprietary Skype protocol is challenged by open standards like the session initiation protocol (SIP) and the established phone standards family H.323. The transportation of voice over IP (VoIP) is a big issue for organizations since it provides cost saving potentials as well as new integration scenarios for unified communications. To benefit from this integration, organizations start to install own servers that provide instant messaging, voice and video telephony. Product examples are Cisco Call Manager, IBM Lotus Sametime and Microsoft Office Communications Server.

Voice and
video telephony

2.2.4 Network Management

Management and supervision of networks must be as simple and efficient as possible. That applies firstly to the administration of logical structures on top of the physical network topology, namely IP addresses and DNS names. Secondly, it includes supervision of hardware components' states and easy remote control of network-related settings on clients.

DHCP

One mechanism to achieve this is the dynamic host configuration protocol (DHCP) that automatically assigns IP addresses to computers. It is quite simple and uses UDP as transport protocol. To exemplify general function of a network protocol, we explain the process of dynamically assigning IP addresses in detail.

DHCP

example

A computer that goes online in a network and has no permanent IP address assigned needs to get a dynamic IP address before it can communicate normally. It thus generates a *DHCP discover* message that could be interpreted as the English sentence: "Can anybody give me an IP address?" It has no own IP address and it does not know which IP address the responsible DHCP server has, nor if there is one at all. Therefore, sending the message as a broadcast from 0.0.0.0 to 255.255.255.255 is the only option. If a DHCP server receives it and has free IP addresses in its pool, it takes one of those and sends it back as a *DHCP offer* message which could be interpreted as: "I can offer you this address! Do you want it?" This message is again sent as a broadcast, as the recipient has no address to send to yet. The computer that needs an IP address receives this message and sends a *DHCP request* message back stating: "Yes, I want this address! Please lease it to me!" The final message sent is from the DHCP server and is called *DHCP acknowledge* which means: "You can take the IP address. I have booked your MAC address on this IP and will give this address to no one else until the lease expires."

Advantages of

DHCP

With four simple messages an IP address is assigned to a computer. Advantages of dynamic address assignment is that administrators do not have to enter addresses manually, IP addresses are better utilized if not all computers are online every day and a central register of all computers on the network together with their IP addresses is created automatically. DHCP servers can be configured so that certain addresses are only assigned to a certain computer (identified by its MAC address).

SNMP

A second protocol easing network management tasks is called simple network management protocol (SNMP, UDP ports 161 and 162). A central application acts as manager and collects status information via SNMP from network devices that act as SNMP agents. Managers can request information about status attributes with the *GET* and *GET-Next* commands that are answered by agents with *GET-Response* messages. Managers can also request a change of state for the network device with a *SET* command. Finally, agents can report critical states that have to get attention immedi-

ately with the `TRAP` message. A network administrator can use the manager to get a current picture of states from the whole network landscape.

The IP addresses are a first step from the physical identification via MAC addresses to a logical identification. Since people are usually better in remembering names than numbers, especially if the names contain information about the services a computer offers, a naming system has been established to identify computers by names instead of numbers. It is called domain name system (DNS) and is a network protocol. Its hierarchical structure is similar to the folder structure in the file system of a computer, but instead of forward '/' or backward slashes '\', dots '.' are used to separate the different levels. An example for a DNS name is:

`www.google.com`

Such a name is called fully qualified DNS name (FQDN) as it includes the name of the computer itself (`www` in our example) and the complete name of the hierarchy it is in. The so-called root domain, which is the top level element of all DNS names is '.', so that the complete FQDN would be "`www.google.com.`". However, the trailing dot is usually omitted.

A DNS name normally contains at least one top level domain, one sub domain (e.g., `google`, `yahoo`, `freenet`, ...) and the host name (often `www` for computers offering World Wide Web services). Top level domains are either purpose-specific (e.g., `com` for commercial use, `tv` for television-related content, `edu` for educational institutions) or country-specific (e.g., `de` for Germany, `uk` for the United Kingdom, `nz` for New Zealand). In addition to DNS names, servers offering DNS name resolution services are also hierarchically structured. Name resolution means conversion from DNS names into IP addresses. If one DNS server cannot resolve an IP address to a DNS name it asks the next DNS server in the hierarchy. For example, the DNS server for `wiwi.uni-halle.de` could ask the DNS server for `uni-halle.de` concerning the DNS name `www.urz.uni-halle.de`. Besides data about the next higher DNS server, every DNS server holds data about the top-level DNS server in the so-called *root hints*. Table 2-6 shows an example for a DNS name with corresponding IP and MAC address.

DNS

FQDN

Parts of a DNS name

Table 2-6. Example for DNS name, IP address and MAC address

address type	value
DNS name	<code>www.wiwi.uni-halle.de</code>
IP address	<code>141.48.204.242</code>
MAC address	<code>00-90-27-7E-16-F8</code>

Every DNS server has an area of responsibility called authority zone. It is the only server that has the authority to give qualified answers to questions concerning naming of computers in this zone. For example, if a client

Authority zone

is asking for the IP address of `www.google.com` only the DNS server responsible for `google.com` can give a *qualified* answer to this question.

Caching

One server would soon be unable to answer all questions for a popular site. Thus, other servers are caching the name, so that the next question concerning the same DNS name can be answered based on the cached data. Such an answer is called *unqualified*, as the IP address could have changed meanwhile without the DNS server outside the authority zone being notified. This is no problem, since IP addresses of publicly accessible servers usually do not change often.

Dynamic DNS

For servers that do change their IP address regularly, it is possible to mark a name as cachable for a short time frame or not at all. This mechanism is called dynamic DNS.

Telnet

Telnet is a network protocol specified by the IEEE and provides a fairly general, bidirectional communications facility with seven-bit character set. It is typically used to provide command-line sessions to administrate hosts on the Internet. The idea behind was to create a virtual terminal or terminal emulation to control a host. The program implementing the client part of the protocol is also called `telnet` and can be used to invoke a Telnet session to a remote host. Telnet operates on the basis of TCP and uses port 23 by default. It provides a connection to interact with the remote operating system, e.g., using a UNIX shell. The program can also be used to connect to other remote services (e.g., SMTP on port 25, section "Messaging" on page 108) and communicate with them using the native commands specified by the respective protocol.

2.2.5 Network Hardware

This section examines network hardware and builds on the explanation of ISO/OSI layers and their implementation in network protocols. Network devices mainly differ with respect to their processing of header data at different layers.

Repeater

One of the simplest devices within the network is a repeater. It is limited to refreshing signals. Refreshing means, that a attenuated electric signal coming in is resent with full strength, so that longer distances can be bridged. Thus, repeaters operate at the physical layer (layer 1).

Hub

A second device type that has to be classified into the physical layer is a hub. It receives signals from one port and forwards it to all other ports. It can be seen as a repeater with more than two ports. A port in this context is a kind of socket where the network cable is plugged in.

Bridge

A network bridge can be seen as a smarter form of a repeater. It forwards network packets only if they are addressed to the respective network segment and therefore avoids collisions being spread into the whole network. It thus has to support physical and data link layer. A bridge is also able to connect networks with different topologies. In a way, a bridge cre-

ates a physical separation between two parts of a network that belong to the same or to different logical segment(s).

Another device class on layers one and two is called switch. A switch is like a hub because it connects multiple computers with each other. In contrast to hubs, it does not just forward a transmission to all other connected devices, but evaluates the target address (MAC address) in the network packet and forwards the transmission only to the port of the recipient or to the uplink (special port to “the rest of the network”) if the recipient is not directly connected to the switch. Therefore, it provides a direct communication between two computers which is significantly reducing collisions.

Switch

Access points manage connected wireless devices and provide an uplink to the wired part of the network. Sophisticated multiplexing methods (e.g., code division, CDM) are needed to directly communication with a single wireless device. Currently, these methods are rarely implemented so that access points can only provide direct communication in uplink direction and forward packets targeted at one of the wireless connected devices to all of them.

Access point

Routers connect network segments. They have one connection to every segment and one IP address for each of them. They inspect packets coming in from one segment and process their headers up to network layer three to identify whether they are targeted to one of the other segments (or a segment not directly connected). They determine to which network interface the packets have to be sent, based on a routing table and IP addresses of network packets. Every computer has a simple form of a routing table built in that is being created based on IP address, network mask and *standard gateway*. Standard gateway in this context is the name of the router of the network segment. The routing table can be displayed with the Windows command `route print` (XP and 2000) or `route -p` on UNIX systems. Routers maintain much more complex routing tables. Since they would be hard to maintain manually, most routers support one ore more of the automatic router configuration protocols, like RIP (router information protocol) or OSPF (open shortest path first).

Router

Devices on higher levels of the ISO/OSI stack are called gateways. We distinguish transport gateways and application gateways. Transport gateways connect two networks that use different connection-oriented transport protocols, e.g., TCP/IP and ATM. As the name suggests, they examine headers of network messages up to the transport layer. Gateways on the application layer are called application gateways and must be able to process application-specific data formats and translate data from one format into another. An email gateway for example could receive email messages with SMTP in the MIME format and forward them to a mobile phone in SMS format (see “Messaging” on page 108).

Gateway

Up to now we have only discussed standalone devices. But computers also need some kind of device or adapter (data transmission equipment) in order to connect to the network. Such an adapter is usually called network

*Network inter-
face card*

interface card (NIC), no matter if it really is an extension card (e.g., PCI or PCMCIA) or a chip on the mainboard of the computer as common today. It is an interface that maintains a permanent connection to the network.

*Modem, DSL
modem, ISDN
adapter*

Devices designed to create temporary connections to the network (dial-up networks) are called modem (modulator/demodulator) based on their initial purpose to translate between digital computer signals and the analogue signals used on phone lines. Nowadays, many telephone lines also operate on a digital basis (ISDN, integrated service digital network), but devices that create network connections over phone lines (e.g., DSL modems, digital subscriber line) are still called modems or sometimes adapter (e.g., ISDN adapter). Those dial-up connections provide much less bandwidth than permanent network connections starting from 14 to 56 kBit/s for analogue modems over 64 kBit/s for ISDN adapters up to 1 to 16 MBit/s for DSL and cable modems. Cable modems are used for Internet connections via TV-cable networks and are widely spread in the United States. For DSL connections, synchronous (SDSL) that provide the same up- and download speeds and asynchronous connections (ADSL) that provide significantly lower up- than download speeds can be distinguished.

2.3 Infrastructure Services

On top of this basic network infrastructure, a set of higher-layer services establishes a more abstract and easy to use platform for applications. Depending on the structure of data, one of several storage services takes care of permanently and securely storing data on a medium. Specialized access protocols most of which are implemented in a Web server today can be used to access stored data from any place in the network. Messaging services are used to actively deliver data to one or more recipients. A set of security services that covers the areas storage, access and messaging supplements the infrastructure at this level.

2.3.1 Storage

Data storage is organized in a layered system similar to the network layers in the ISO/OSI model. It starts with media that store single bits physically e.g., as optical or magnetic marks. A drive is needed to read and write from and to a medium. A controller provides access to the drive for the operating system in close conjunction with a device driver. The file system is a part of the operating system dealing with organization and access to data on a logical level using files and folders. On the application level finally, various application systems provide further abstractions that are

suited either for structured or semi-structured data (section 3.1.1, 154ff). Table 3-1 gives an overview of the layered storage architecture.

Table 2-7. Overview of storage layers

domain	layer	examples	logical unit
information system	business-oriented application	invoice, travel request, conference paper, company presentation, meeting report	document type
	storage-oriented application	document and content management systems, hierarchical storage management systems, database management systems	document, record
operating system	file system	FAT, NTFS, HFS, ext3, vxfs, reiserFS	file, folder
	driver	driver for controller, drive, volume	volume
hardware	controller	IDE, SATA, SCSI	device, drive
	drive	hard disk drive, tape drive, CD drive	block, sector
	medium	magnetic (tape), optical (CD), magneto-optical (MOD), electric (flash)	bit

Media. On the lowest level, all data is stored physically on a medium. Available media differ in terms of cost, capacity, speed, physical space needed and other characteristics. The following classification system helps to clarify the differences.

Availability of media can be separated into online, offline and nearline. Online storage means that the medium is directly available, e.g., main memory (RAM), hard disks. Offline storage denotes media stored separately from the drive in a cabinet that have to be manually put into a drive in order to access them, e.g., CDs, DVDs, tapes. Nearline storage is in between those two forms. Nearline media are actually offline, but can be automatically inserted into a drive by robots or other automated mechanisms, e.g., tape libraries, jukeboxes. A general heuristic is that the faster data has to be accessed, the smaller is the medium that can be used and the higher is the cost per capacity unit.

Online, offline, nearline

Volatile media like RAM loose data after power has been switched off. Non-volatile media like hard disks or CDs keep data regardless of power supply. However, readability of data on non-volatile media is limited. Time varies from 10 to 50 years depending on medium and environmental influences which is still far less than durability of acid-free paper as a medium.

Volatility

Cost of media reaches currently from roughly 25 € per GB for RAM to around 2 € per GB for SCSI hard disks and 0.1 € per GB for DVDs. Tapes are in the same price range as DVDs depending on their type. New media like flash memory in solid state disks is expensive at moment, but gets cheaper quick with production capacities increasing. Table 2-8 shows

Cost

examples for commonly used storage media ordered according to access time. Media on the top of the list are small, fast and costly (SRAM). Towards the bottom of the table, media get slower, larger and cheaper.

Table 2-8. Storage media ordered by access time

medium	typical capacity	typical access time	typical transfer rate	cost (in EUR)
SRAM	0.1 - 16 MB	5 ns	~ 9000 MB/s	~ 70 per MB
DRAM	128 - 8000 MB	25 ns	~ 5000 MB/s	~ 25 per GB
flash memory	1 - 128 GB	100 ns	20-150 MB/s	~ 12 per GB
SAN storage (FC)	500 - 30000 GB	4 ms	100-200 MB/s	~ 10 per GB
hard disk (SCSI)	36 - 300 GB	4 ms	80-110 MB/s	~ 2.0 per GB
hard disk (ATA)	80 - 1000 GB	10 ms	60-80 MB/s	~ 0.2 per GB
UDO disk (MO)	15-30 GB	50 ms	~ 8 MB/s	~ 0.5 per GB
CD	0,7 GB	100 ms	~ 7 MB/s	~ 0.3 per GB
DVD	4.7 or 8.5 GB	100 ms	~ 22 MB/s	~ 0.1 per GB
Blu-ray disk	25 or 50 GB	200 ms	~ 27 MB/s	~ 0.4 per GB
WORM	15 - 300 GB	300 ms	~ 20 MB/s	~ 0.5 per GB
tape	40 - 800 GB	60 s	~ 33 MB/s	~ 0.1 per GB

Access method,
access time

Access method and as a consequence access time are further characteristics of media. Methods are direct access, e.g., hard discs and optical media, and sequential access, e.g., tapes. Sequential access means that access to data saved on parts of the medium far away from each other takes a substantial amount of time, whereas data written directly one after another can be read much faster. For direct access media the difference is much lower. Access time is dependent on that and reaches from less than one millisecond for RAM to about 10 ms for hard disks, 40-100 ms for optical media up to several seconds or even a few minutes for tapes.

Physical characteristics

Media can further be classified according to the way they physically store data. Main types of storage systems are *magnetic* (hard disks, floppy disks and tapes), *optical* (optical disks, e.g., CD and DVD) and *electric* (SRAM, DRAM, flash). There are also some mixtures of the types like *magneto-optical* disks (MOD, e.g., Minidisk).

Magnetic media

Magnetic media uses very small magnetic particles that are evenly spread over a medium. The read/write head of the drive then uses induction to read and electromagnetic forces to change the magnetic orientation of the particle and therefore make it a binary zero or one.

Optical media

Optical media use refraction characteristics of a material (often silicon) to read. In the case of CDs and DVDs, a laser with a certain wavelength, e.g., 780nm (infrared) for CDs and 650nm (red) for DVDs, about 400nm

(blue) for DVD successor Blue-Ray disc (which won against HD-DVD), sends a light impulse to the disk. The light gets reflected and is detected by a sensor. The zeros and ones are realized with small holes (pits) in the surface (lands), so that light is reflected differently.

Flash memory stores information in an array of transistors, called cells, each of which traditionally stores one bit of information and is usually made of silicon. There are two types of Flash memory: NAND memory is faster (especially for writing large blocks of data) and can be larger (currently up to 16 GB in USB sticks and SD cards and up to 256 GB in solid state discs). NOR memory can be used directly for executing programs and thus is often used in PDAs to store the operating system.

Transfer rate is depending on the density of data on the medium (the higher the density the more bits can be read at once) and the medium's speed of movement. The original CD speed (1x) is 210 rotations per minute (rpm) which equals a transfer rate of 150 kB/s. Current CD drives have 52x speed and therefore rotate with up to 11,000 rpm transferring 7.8 MB/s! DVDs operate with 1,400 rpm resulting in a transfer rate of 1.4 MB/s. Current 16x speed drives offer a transfer rate of 22 MB/s, but that speed is no longer measured against the angular velocity but against velocity at which a track bypasses the head in outer disk areas so that they reach only 8,800 rpm. The same applies to Blu-ray discs which offer 4.5 MB/s at single speed, currently reach 6x speed with 27 MB/s and are specified up to 12x speed which would result in 10,000 rpm and up to 54 MB/s. As a comparison, current hard disks rotate between 5,400 and 15,000 times a minute.

The last characteristic is the ability to rewrite. Some media like hard disks can be rewritten nearly infinite times. Optical disks like CD-RW, DVD-RW or DVD-RAM can be rewritten about a thousand times, whereas data can only be written once to other media like CD-R, DVD-R or DVD+R and WORM (write once read many).

Drive. Media need a drive so they can be read and written. In some cases, drive and medium are inseparable, e.g., hard disk, USB stick, but usually media can be removed from the drive, e.g., tape, CD, SD-card. The drive has to provide mechanisms to read and write data, to position the head to find data, and provides a first logical abstraction of the physical medium.

Mechanisms have been invented that automate insertion and removal procedures and increase available capacity dramatically, as several (usually 8-72) media can be used. Tape libraries deal with tapes and jukeboxes with optical discs. The principle is the same for both systems. Media are put into a special stockroom where a robot arm can pick the selected medium up and insert it into a drive. Each medium is identified by a barcode in order to quickly find the right medium when searching for a file.

Electric media

Transfer rate

Rewrite

*Tape library,
jukebox*

Controller and Driver. Controller and its driver implement the interface between operating system and drives. Program logic is partly stored in a controller's BIOS (basic input output system) and partly in the driver. Over the years, more and more program logic and functionality has been integrated into the controller's BIOS (see "Developments in abstraction from storage media" on page 122). Besides that, controllers differ mainly in the type of interface and the number of drives they can address.

*IDE, SCSI,
SATA*

IDE (integrated drive electronics) was the standard for consumer drives (hard disks as well as CD and DVD drives) for a long time and is able to simultaneously address two drives each on usually two channels. SCSI (small computer system interface) is the standard for drives used in enterprises and can address 7 to 15 drives. Serial ATA (advanced technology attachment) is the most recent standard and tries to fill the gap between IDE (also called ATA or parallel ATA) and SCSI so that it replaced IDE in the consumer market and is also used for low cost devices in the enterprise market. A number of enhancements in the different versions of the interface protocols and partly also the physical interfaces have increased the transfer rate of both SCSI (Ultra SCSI, Ultra Wide SCSI, Ultra160 and Ultra320) and IDE (ATA33, ATA66, ATA150). There is also a serial version of SCSI called serial attached SCSI (SAS) which is more and more replacing other SCSI versions.

*Partition,
volume*

Space on a drive can be separated into one or more partitions that form logical units. More important for enterprise infrastructures are volumes as a logical abstraction of disk drives. In the simplest case, a volume consists of one partition, but it can also span multiple equally big partitions on several disks (see also the section on RAID below). The abstraction layer that handles volumes is called *logical volume manager*. In Storage Area Networks the hierarchy of logical space units is often much more complex with several SAN specific unit types that are aggregated to volumes.

File system. A file system builds on volumes and structures them logically. Folders are used to hierarchically structure files which are the smallest logical unit containing data. A file allocation table (FAT) stores information about positions files begin at and about how long they are.

*Folder hierar-
chy*

UNIX file systems use a root folder ("/") as top-level element whereas Windows systems work with drives representing a volume for the file system and each drive has a drive letter and its own root folder. Since Windows 2000, it is also possible to mount volumes into folders which is common practice in UNIX file systems. Besides that, file systems differ mainly in their capabilities to handle long filenames, to address large volumes, to handle file security (encryption and access control) and to deal with fragmentation and corrupt files (e.g., due to power failure).

*FAT, NTFS,
UFS, ext2*

On Windows systems, FAT was the dominant file system for years, first in version FAT16 that could address volumes with up to 4 GB and later with FAT32. In Windows NT and its successors, NTFS (new technology

file system) is used mainly due to its advantages in file security, i.e. encryption and access control using access control lists (ACLs). Nonetheless, FAT is still relevant as it is used on floppy discs and flash memory. On UNIX systems, the traditional UNIX file system (UFS) has been replaced by several (often OS-specific) newer systems (e.g., Linux ext2, Irix EFS, Sun ZFS).

In the last years, most of them have been replaced by so-called journaling file systems. Journaling introduces the concept of transactions to file systems by recording each write transaction and committing it after success. Thus, corrupt files can easily be identified instead of having to scan the whole disk for any inconsistencies after a crash. In Linux, ext3 adds journaling to ext2. Reiser FS and JFS (Journaling File System) are native journaling file systems and are also used in other UNIX systems.

Journaling file systems

With an increasing number of files being stored, it becomes increasingly difficult to find them using the simple folder hierarchy provided by current file systems. A recent approach to overcome this limitation uses meta-data (section 3.2, 172ff) to dynamically generate views on files or find files according to attributes. It uses the query capabilities of relational database management systems to efficiently access files by querying meta-data. With this approach, queries for e.g., files used last week, that are related to a project or are written by a certain author can easily be answered. Although the topic was widely discussed in 2004 with Microsoft's WinFS and the open source approach DBFS (database file system), in 2008 there is still no wide-spread implementation of the concept.

Database file systems

Apple's HFS+ file system is maybe the only wide-spread implementation of a file system that can handle arbitrary meta-data attributes assigned to files. This enables flexible categorization and description of files so that they can be more easily found using the Spotlight search engine

Meta-data in file systems

Application. On the application layer, data storage is based on files and further abstractions are introduced. *Documents* are logical units for semi-structured data and often are stored in one file, but can also span multiple files, e.g., master and sub-documents in MS Word or files composing an OpenOffice document (for a detailed definition of document see section 4.2, 271ff). *Records* are logical units for structured data and several records are stored within one file.

A record is a set of data treated as a logical unit, also called entity, and consisting of attribute value pairs, also called fields. An attribute describes what kind of data is being stored. It is therefore data about data or meta-data. It has at least a data type, a name and ideally also has a description that helps users to interpret values. Thus, meta-data captures part of the semantic. Data is divided into structured and unstructured data according to the amount of meta-data that is available to describe structure and semantics of the data (section 3.2, 172ff).

Record, field, attribute

File, document, file type, document type

A file is the smallest addressable unit in a file system (section 5.2.4, 382ff) and provides a container to store all kinds of data including executables, dynamic link libraries, configuration and resource files needed to execute an application, as well as text, audio and video files that make up the contents. The association between applications and contents is usually defined by the file type that is specified by the internal file format and the file extension (e.g., doc, html, pdf, zip). In analogy to DBFS, today it is often not sufficient to distinguish the file type. It makes much more sense to concentrate on the document and type, which provide semantic abstractions of the file type. With the wide-spread usage of XML-based file formats an XML file could be a Word document, a configuration file or a Infopath form. A further semantic enhancement would be the handling of document types (section 3.2.4, 191ff) which would allow to distinguish between a travel request and a vacation request although both could be Infopath forms. In addition to that, a vacation request for example could still be found no matter whether it's stored as XML form or PDF form.

Data base management systems

DBMS can be used to store structured data. A database system consists of a collection of structured data and software to describe, store, create, retrieve and manipulate data. They further abstract from file-based data storage mechanisms offered by operating systems. DBMS use relational calculus to store data in tables consisting of typed fields (columns in a table) and data records (rows) where instance values for each field are stored. Tables usually have a primary key that uniquely identifies a single data record in a table. Such a primary key can be referenced in a second table to establish a relation between the two tables and is called foreign key there. Structured query language (SQL, section 3.2.3, 183ff) is used for inserting, retrieving and manipulating data. DBMS additionally provide support for e.g., transactions including rollback of incomplete transactions, fast access to data via indexes and caching, backup and restoring data, verifying integrity, monitoring of performance and user management to control access. A DBMS can handle multiple databases. A database is usually stored in one file for data and a separate file for indexes.

Directory service

A directory service is designed to store structured data and optimized to provide fast read access and simple query capabilities. In contrast to regular databases that store data mostly in relational tables, directories provide hierarchical storage (Figure 2-7). In addition, a directory is designed for fast retrieval instead of data manipulation. Another difference is the lack of sophisticated transaction or roll-back mechanisms. Like databases, directory services principally can hold any kind of structured information. However, they are mainly designed for relatively small portions of data, e.g., for user account management.

Scheme, distinguished name

Each directory has a certain structure defined by a scheme. The scheme can be designed for any purpose like database schemes. Schemes define object classes. Each directory entry is member of one object class. The main object class within a directory is `entry`, which is mostly composed

around real-world concepts such as people or organizations. A unique distinguished name (DN) identifies each entry, similar to fully qualified domain names in DNS. An entry can have one or more attributes, defined by the scheme. Unlike databases, directories typically offer a set of pre-defined object classes and attributes.

```
cn=Ronald Maier,ou=Information Systems,ou=School of
Management,o=University of Innsbruck,c=at
```

Example of distinguished name

The example uses the object classes `country`, `organization`, `organizational unit`, and `person`, which are all subclasses of `entry`. Further relevant object classes are `organizational role`, `organizational person` (a subclass of `person`) and `alias`.

Important attributes for these classes are country name (`c`), street address (`sa`), organization name (`o`), organizational unit name (`ou`), common name (`cn`), surname (`sb`), email address (`mail`), user ID (`uid`), and user password (`userPassword`).

Directory services are commonly used for organization-wide address books and application-spanning login information (section 3.3.2). Most current email clients such as MS Outlook, Apple Mail, Eudora Mail or Mozilla Thunderbird support searching lightweight directory access protocol (LDAP) directories for addresses and many enterprise application support central management of user data in LDAP directories. Examples for directory service implementations are Microsoft Active Directory, Novell eDirectory and the open source system OpenLDAP. Figure 2-7 shows an example for two person objects in a hierarchical organizational structure.

Application areas

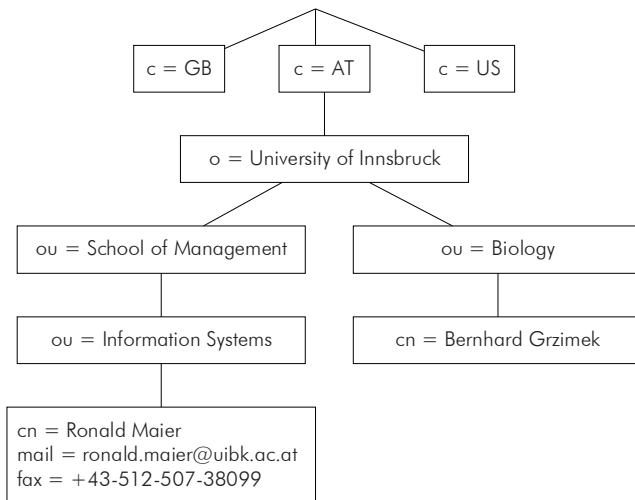


Figure 2-7. Hierarchical structure of an LDAP directory service

LDAP server can be operated redundantly in a master-slave configuration. Entries can only be edited on the master server. A synchronization mechanism called replication updates slave servers when data has changed on the master.

LDAP

LDAP (lightweight directory access protocol) is a client-server protocol for accessing and managing directory data. LDAP was developed at the University of Michigan in 1995 and is part of the X.500 specification. It was originally developed as a lightweight front end for the X.500 directory system, as the original X.500 standard was too resource-intensive for Internet and desktop applications. Whereas the X.500 Directory Access Protocol (DAP) was implemented on a separate network protocol stack, LDAP is based on TCP/IP and uses only string formats for data transport.

*File server,
DMS, CMS*

There are three alternatives for storing semi-structured data. File servers store files of all types in the file system of the operating system and exposes them to network users with a suitable protocol (section 2.2.3). Document management systems (DMS) are designed to store documents together with meta-data and outperform file servers by e.g., providing versioning, advanced locking mechanisms and sophisticated search capabilities (section 4.2.1, 272ff). Content management systems (CMS) are designed to manage Web pages or content for Web publishing, but are also used to manage digital assets of any kind (text, audio, image, video). DMS and CMS features overlap, though, and are more and more integrated in enterprise-wide systems (section 4.4.4, 321ff).

Storage systems. As there are many different media suitable for storage there is a need for unified access. Ideally, users do not need to deal with the question whether data is stored on hard disks, DVDs or tapes. Storage systems manage multiple (different) media, unify access to them, and optimize access times by caching data.

HSM

A concept closely related to that is called hierarchical storage management (HSM) system. Not all data is accessed equally often. Therefore, it is a waste of money to store all data on fast media like hard disks. Data that are rarely used should be stored on cheaper, but slower media. HSM systems integrate access to several different types of storage sub-systems and provide automated, rule-based migration of data from expensive hard disks to cheaper tapes or optical disks. HSM systems are discussed in conjunction with *information life-cycle management* (ILM) which implements more advanced rules related to the data value to decide which data to store on cheaper media.

Developments in abstraction from storage media. Development in the storage area is characterized by increasing abstraction. Programmers do not have to care about how to position the head of the disk drive, but work with records and documents instead. The first abstraction is that logical blocks of data on the disk can be addressed (LBA, logical block address-

ing) instead of having to deal with head movement. Blocks are numbered from 0 to 65535 and can be read and written using their numbers.

In the next step, volume abstraction went down from driver to controller. Controllers no longer provided access to disks, but rather to volumes with a given capacity and certain security and performance characteristics. The mechanism to aggregate multiple hard disk partitions to volumes is called RAID (*redundant array of independent disks*, sometimes the “I” is interpreted as inexpensive). Data stored in a RAID is spread over all disks which is managed by the controller. RAID offers several different operation levels that allow tailoring transfer rate and level of security.

RAID

The most important *RAID levels* in practice are level 0 (striping), level 1 (mirroring) and level 5 (striping with parity). *Striping* spreads data equally on all hard disks, without any redundancy. The result is fast transfer rates and no fault tolerance. If one hard disk fails, then all data is lost. *Mirroring* writes all data redundantly to two disks instead of one. If one disk fails the other one still holds the data. The failed disk can be replaced and data is rebuilt on that disk, so that no data loss occurs. *Striping with parity* can be used with three disks and more. Data is written on two disks and parity information on the third disk. If one disk fails, data can be rebuilt using parity data. There is also a level 10 or 1+0 which is a mixture between striping and mirroring.

RAID levels

A proven strategy in practice is to use two disks with RAID level 1 for the operating system and the remaining disks with RAID level 5 for data. RAID logic can be implemented on the level of the controller, the driver, or even at file system level, but usually it resides in the controller’s BIOS today. Table 2-9 summarizes the RAID levels, indicates their levels of security and speed (data transfer rate) and shows what capacity is actually usable when 6 disks with a capacity of 200 GB each are used.

Table 2-9. Example for different RAID configurations with six 200 GB disks

RAID level (name)	usable capacity	security	speed	description
0 (striping)	1200 GB	--	++	no redundancy
1 (mirroring)	600 GB	++	o	full redundancy
5 (striping with parity)	1000 GB	+	+	1 disk for parity information
10 (striping and mirroring)	600 GB	++	+	full redundancy and striping

Usually, hard disks used for enterprise applications are *hot swappable*, which means that they can be exchanged while the system is running. If a hard disk fails, then it can easily be removed from the system and a new blank hard disk of the same capacity can be inserted instead. Data on the old disk is rebuilt within a few hours so that the system runs without interruptions for RAID levels 1, 5 or 10.

Hot swap hard disks

*Direct attached
storage, RAID
array*

A next step in the development was that the whole storage sub-system has been outsourced from the computer (usually only for servers). The advantage of these RAID arrays that are outside of the computer chassis is that more space for additional disks is available. RAID arrays grew more and more independent, so that a second controller was needed inside the computer chassis to connect to the controller inside the RAID array. Such a storage subsystem is called *direct attached storage* unit (DAS). Existing interfaces (e.g., SCSI) were designed to match transfer speeds of single disks or a hand full at most. New interfaces had to be introduced (e.g., ultra wide SCSI) that are able to deal with the accumulated transfer rate of a dozen and more disks. Fibre channel is the most common interface used today, which is available in two versions allowing transfer rates of 2 GB/s and 4 GB/s.

SAN

The next step was to further separate computer and storage sub system by sharing storage space between multiple servers. To accomplish this, a device similar to a network switch has to be introduced in order to route data from storage unit to servers and vice versa. Such *storage switches* exist for fibre channel. Centralized storage units together with connections to servers are called *storage area network* (SAN). Initially, only entire disks could be joined to RAID volumes and assigned to servers. This lead to inefficient space allocations, especially since standardization of parts forbids to use different disk sizes e.g., 74 GB disks for the operating system and 300 GB disks for data. The goal is to provide disk space as a central service, where capacity, speed and redundancy are the features and parts of the overall disk space with a certain feature set can be dynamically assigned to servers. That includes increasing or decreasing assigned disk space (as long as there is free space on the partition). Several servers can connect to one SAN and every server gets its own space. Alternatively, multiple servers can share disk space (for clusters). Advantages are better utilization of disk space and increased performance, because for RAID level 5 e.g., 20 hard disks can be used for striping instead of 4 or 5 in a single server, which results in theoretically 4 to 5 times more speed and nearly as much in practice. The disadvantage is higher costs, since more intelligence in controllers, SAN switches as well as in the driver and BIOS on servers are required.

IP-SAN, FCoE

Latest development in the storage area is reuse of wide-spread standard technologies in order to lower costs. Fibre channel connections between servers and SANs are replaced by IP connections over Gigabit Ethernet. These technologies are much cheaper due to their universal usage in data and storage networks and resulting volume effects. This type of SAN is referred to as IP-SAN and requires its own IP address. Existing storage protocols have been extended, so that they can run over IP connections. Examples are iSCSI that extends SCSI (small computer systems interface) as well as iFCP (internet fibre channel protocol) and FCIP (fibre channel

over IP) that extend the fibre channel protocol. Recently, FCoE (fibre channel over ethernet) was developed to reduce protocol overhead generated by IP and run fibre channel directly over ethernet connections.

In order to prevent service downtimes resulting from catastrophes like fire or water damage in the whole building, many companies distribute the hardware that runs clustered applications across two buildings that are between a few hundred meters and several kilometers away from each other. This requires, that a storage area network is able to replicate data from one storage device to a second one on a block basis, so that cluster nodes in both buildings have access to the same data although they access different storage devices. This technology is called SAN metro cluster and requires high speed fibre optical connections between the buildings.

Network attached storage (NAS) is directly accessible using standard network protocols such as NFS and SMB (section 2.2.3) and can be seen as a further enhancement of IP-SANs. They can be used for data storage only, since an operating system is needed on servers to connect to a network share. Figure 2-8 gives an overview of the storage sub-systems discussed here and shows how they differ regarding protocols, interfaces, hardware components, and software layers. DAS, SAN and IP-SAN operate with block I/O which enables raw partitions for swapping or database, so that operating systems can be installed on volumes on these devices. NAS operate with file I/O so that they behave like file servers. In fact, many NAS appliances are file servers based on Linux.

Metro cluster

*Network
attached
storage*

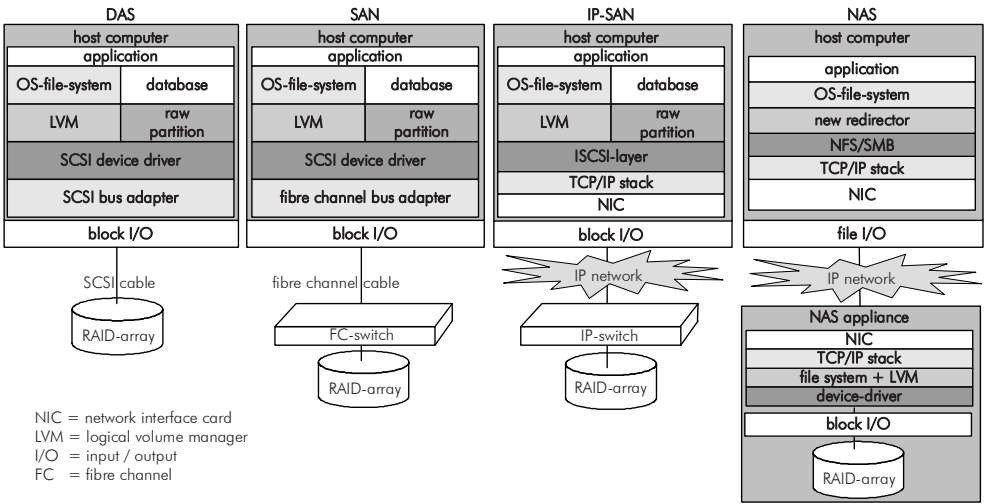


Figure 2-8. Overview of storage subsystems

2.3.2 Security

In order to establish a secure area within company networks that are connected with each other, networks have to be split logically into different security zones.

*Internet,
ARPANET*

Security zones Internet, Intranet, Extranet. Generally speaking, an internetwork or internet is formed when different networks are interconnected. Different means that they are maintained by multiple organizations and that various network technologies (e.g., different protocols or communication media) are connected with each other. The global network commonly known as the Internet is such an internetwork. Its origin dates back to the ARPANET which was developed in the late 1960s and put into operation in 1969 connecting a number of research institutions (e.g., the University of California with the research department of the US Department of Defence) in order to exchange data on joint research projects. The key underlying idea of the Internet is open-architecture networking where choice of individual network technologies should not be dictated by a particular network architecture, but rather could be selected freely by a provider. The second core idea is that the Internet should keep working if some hosts within the network break down, so that in case of an attack the infrastructure could be still used to communicate, although several nodes would be disabled. Connection with other networks is possible through a meta-level *Internetworking Architecture*.

Internet growth

The Internet has grown tremendously from 213 hosts in 1981 over 500 thousand in 1991, 100 million in 2001 and 540 million in 2008 (according to Internet Systems Consortium, www.isc.org). Access to the Internet is unequally distributed with respect to continents and social levels. For example, in North America 73.6% of the population had access to the Internet, whereas in Africa only 5.3% of the population had access².

*Institutions for
network man-
agement*

The Internet is organized decentrally with as little central control as possible. Nevertheless, some central institutions are necessary for network management. The most important ones are:

- *Internet Society* (ISOC), a non-commercial umbrella organization that hosts over 150 Internet-related organizations, e.g., the IETF,
- *Internet Engineering Taskforce* (IETF) that develops new and enhances existing Internet protocols and standards,
- *Internet Architecture Board* that steers development of protocols,
- *Internet Assigned Numbers Authority* (IANA) that manages protocol parameters (e.g., TCP ports) and their assignments to services,
- *InterNIC* (Network Information Center) that is responsible for globally unique addresses,

² Internet world statistics, www.internetworldstats.com, June 2008.

- the independent *Internet Corporation for Assigned Names and Numbers* (ICANN) that is responsible for the allocation of domain names and Internet addresses and will take over tasks of IANA and InterNIC which are controlled by the federal government of the United States.

Internet service providers (ISPs) operate sub-networks with high bandwidths and provide access to the Internet. They are commercial (e.g., EUnet, NTG/XLINK) or non-profit organizations (e.g., DFN in Germany). National telephone companies like the German T-Com or international companies like America Online (AOL) or Microsoft Network (MSN) are commonly known ISPs offering Internet access for private users together with additional services like email services and Web portals.

Internet service providers (ISPs)

All participants of the Internet use the TCP/IP protocol family, a set of specifications that define how machines in the Internet communicate platform-independently. All Internet standards are documented in *requests for comments* (RFCs). TCP and IP specify two protocol layers of the ISO/OSI protocol stack (section 2.2.2, 100ff). Higher-level specifications of Internet services are HTTP for Web sites, SMTP for email, FTP for file transfer and Telnet for remote login (section 2.2.3).

Internet standardization

Even though some exuberant hopes were disappointed after the dot-com hype around the year 2000, the Internet has changed and still changes the business world substantially by offering new potentials for boundless communication and collaboration, enabling new business models and threatening traditional industries like the music and film industry. Development of the Internet is not at its end, many challenges are still not resolved. Examples for current challenges are efficient search and retrieval and especially how contents can be structured and accessed on a semantic level (Semantic Web, section 3.2, enhanced search technologies, section 4.1), security issues (threats from viruses, secure communication, data security), unintended use of Internet resources (e.g., email spamming, excessive file sharing) as well as micro payment for services provided over the Internet.

Changes and challenges

An Intranet is the internal network of an organization implemented with the help of technologies that based on standards of the Internet (e.g., TCP/IP, HTTP, SMTP). Advantages of Internet technologies are low costs for software, availability of skills, easy integration of internal and external systems, vendor-independent standards and their wide-spread usage. An Intranet commonly distributes internal information with the help of HTML pages and is accessible with standard Web browsers. It can deliver internal administrative information and services for employees like information about organizational units, projects and processes, booking of internal resources (e.g., meeting room, video projector, car) or requests for reimbursement of travel expenses which formerly needed paper-based forms or phone calls. Today, Intranets are an important medium for business-to-employee communication. Although the Intranet is usually connected to

Intranet

Extranet

the Internet, it is separated from it by means of a firewall and therefore represents a secure zone, where internal information can be provided. To further enhance security, access to contents usually requires authentication, e.g., with user name and password, so that employees only have access to data that is intended for them.

A portion of an organization's network accessible from the Internet, but only available to selected groups of authenticated users, e.g., customers or partners, is called *Extranet*. It supports the company's business processes at organizational boundaries, e.g., for procurement or order tracking, or enables joint work with the help of shared information spaces and cooperation portals.

Figure 2-9 shows how the three security zones public Internet, Intranet and Extranet are separated. The Web servers that host contents accessible in the Internet and the Extranet are placed in a so-called *demilitarized zone* (DMZ), which controls data flows to and from the public network with a firewall and uses another firewall to further protect the Intranet (see the paragraph on firewalls below).

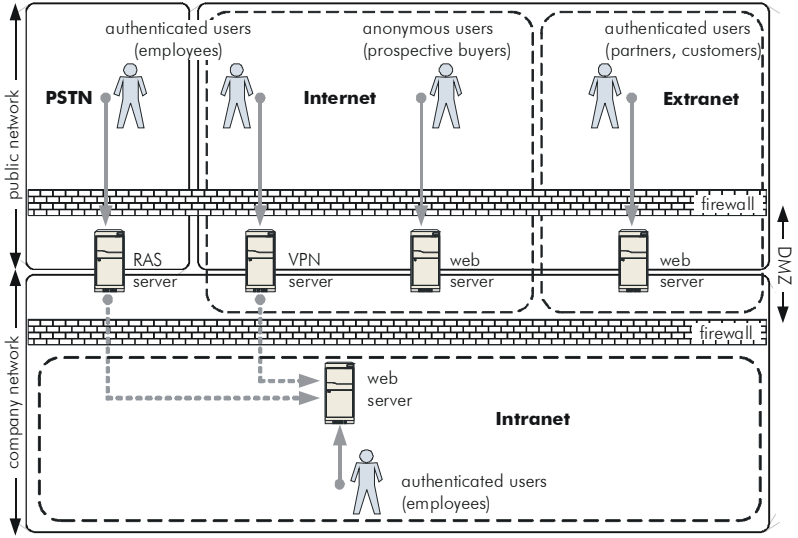


Figure 2-9. Internet, Intranet, Extranet

Employees have access to the Intranet by default, when working in their offices. Additionally, they often need access to the Internet and Extranet as well. Finally, they should be able to access information on the Intranet, when they are out of office. To enable the latter, they can either use dial-up connections directly to a server offering remote access services (RAS) in the company network over the public switched telephone network (PSTN), or they can use an Internet connection to an Internet service provider and

establish a virtual private network over the Internet to securely access the company's Intranet (see the paragraphs on IPsec and VPN below).

Security Threats. All data residing on servers or communicated through the network have to be protected against unauthorized access and other threats. For this purpose, security services must be available within the network and most importantly need to be applied wherever appropriate. The threats a company's infrastructure is facing can be grouped into the following classes:

- *Data loss* happens if important data is deleted, overwritten or lost accidentally, through external attackers or due to hardware failure. Reconstruction of lost data can be time-consuming and costly, if it is possible at all.
- *Manipulation* is the purposeful change of data so that it does not represent the original meaning any more. This can lead to e.g., wrong balance sheets or software code that does not work in the intended way.
- *Unauthorized access* to sensitive data is especially harming organizations if competitors get hold of business secrets. It can also be used to e.g., publish business secrets and influence the company's stock price.
- *Abuse*: company resources can either be abused by employees (e.g., by surfing in the Internet on company cost, often in combination with illegal, or semi-legal contents) or by external attackers that use e.g., hard disk space and Internet connectivity to store and share illegal copies of videos, music files and software.
- *Downtime* means that required infrastructure services are not available or overloaded. For example, if the public Web site of a company is unavailable for a few hours, this is not only damaging the image, but could also mean loss of money due to lost sales over Web shops because customers buy from competitors.

These classes of threats can be detailed to a large number of existing attacks on enterprise infrastructures. We summarize those attacks briefly in Table 2-10 and discuss security measures afterwards.

Encryption of data stored on media and communicated in networks prevents spying and manipulation. Authentication together with restricted access to data is helpful against unauthorized access. Filtering of network packets coming into a company network or leaving it can reduce the danger of any threats and redundancy prevents the loss of data and downtimes. The following section presents implementations for these security measures. They are discussed sorted regarding their applicability to storage, access and messaging.

Concrete attacks

Security measures

Table 2-10. Attacks on enterprise infrastructures

attack	description
Adware	A software application in which advertisements are displayed while the program is running. These applications include additional code that displays the ads in pop-up windows or through a bar that appears on a computer screen. Some adware includes code that tracks a user's personal information and passes it on to third parties, without the user's authorization or knowledge (see spyware).
backdoor, Trojan horse	A method of bypassing authentication and obtaining remote access to a computer, while intended to remain hidden to casual inspection. Backdoors can be unknown programs that simulate useful functions (e.g., Back Orifice) or modifications of well-known programs. The term <i>Trojan horse</i> is derived from the classical myth of the Trojan horse in Homer's <i>Illias</i> .
brute force attack	A method to determine the decryption key of an encrypted message or a password by systematic guessing. This involves generation of a large number of keys either algorithmically or from a predetermined list. The latter is known as dictionary attack.
(D)DoS	A <i>denial-of-service attack</i> (DoS) is an attack on a computer system or network that causes a loss of services to users, typically network connectivity. Such attacks are not designed to gain access to systems. Common forms aim at the consumption of computational resources, such as bandwidth, disk space or CPU time. Distributed denial-of-service attacks (DDoS) use multiple computers (often captured by worms) to overload the target. As the packets come from multiple sender addresses, it is difficult to repel such an attack.
man-in-the- middle	An attack in which an attacker is able to read and modify messages between two parties without either party knowing that the link between them has been compromised. The attacker must be able to observe and intercept messages between the two victims, e.g., by using spoofing. For example, communication to and from the Web site of an online bank could be eavesdropped and account numbers and passwords could be logged without being visible to users that see a simulation of the real Web site.
port scanning	Systematic checking for services presented on open TCP/IP ports, usually as part of an intrusion attempt or a security scan of administrators.
social engi- neering	The practice of gaining people's trust to make them reveal sensitive data e.g., by calling them on the phone and telling them to give their password to the administrator for a routine check.
spamming	Sending identical or nearly identical messages to a large number of recipients without their permission. Addresses of recipients are often harvested from Usenet postings or Web pages, obtained from databases, or simply guessed by using common names and domains. The terms <i>unsolicited commercial email</i> (UCE) and <i>unsolicited bulk email</i> (UBE) are synonyms.
spoofing	Forging the header of an IP packet, so it contains a different address and appears to be sent by a different machine. This is known as <i>IP spoofing</i> . Another form of spoofing called <i>DNS spoofing</i> , fakes DNS-names instead of IP addresses.

Table 2-10. Attacks on enterprise infrastructures

attack	description
Spyware	Computer software that gathers information about a computer user and then transmits it to an external entity without the knowledge or informed consent of the user.
virus	A small program that can replicate and spread itself by means of placing copies of itself in uninfected files. A virus is only spread from one computer to another when an infected file is taken to the uninfected computer, e.g., if a user sends it over a network or carries it on a removable disk. Additionally, viruses can spread to other computers by infecting files on connected network folders.
worm	A self-replicating computer program, similar to a computer virus. A virus attaches itself to, and becomes part of, another executable program. However, a worm is self-contained and does not need to be part of another program to propagate itself. A worm can spread itself autonomously to other computers.

Security measures for storage. Encryption is the process of obscuring information to make it unreadable without special knowledge. A key, e.g., a password, is used to transform clear text data into encrypted data by applying an encryption algorithm. Data encryption denotes the permanent codification of data for secure storage. Communication encryption is encoding data for secure transfer over networks. There are a lot of synonyms for both encryption types like online and wire encryption for communication encryption and offline encryption for data encryption. The length of the key is an indicator for the security level of the encryption since it determines the complexity of breaking the code. Encryption and decryption are computational complex. Thus, key length is limited. For data encryption, usually 512, 1024 or 2048 Bit keys are used whereas communication encryption uses 128 or 56 Bit keys.

Data encryption

Redundancy effects storage in two ways. Firstly, hardware can be designed redundantly (e.g., a controller) so that no downtime occurs. Additionally, controlled data redundancy prevents loss of data. RAID levels 1, 5 and 10 are one way to hold data redundantly (section 2.3.1, 114ff). Another way is backup. Most users tend to forget about personal backups and it is more effective to have a large centralized backup device (like a tape library) instead of giving every user a small backup device (like a CD burner). Therefore, a central service that backs up data regularly (e.g., daily over night or weekly during week ends) over the network is advisable. Examples for backup strategies are *full backup* (all files are written to the backup device) and *incremental backup* (only files changed or added are written to the backup device). A proven strategy is daily incremental backup and weekly full back up, which represents a good balance between consumed storage space, network bandwidth and time for backup and restore. The most common media used for backups are tapes. Their advantages are low cost, long readability of data, relatively low space consump-

Redundancy, RAID, backup

tion per GB of data and an average fast access time if put into a tape library (section 2.3.1, 114ff).

Security measures for access. Encryption and authentication are combined to secure access to valuable resources. Both can occur on several layers of the ISO/OSI reference model. We will discuss encryption first.

*Symmetric
encryption,
DES, AES*

Symmetric and asymmetric encryption methods can be applied to both data and communication encryption. Symmetric encryption uses one key (e.g., a password) to encrypt and the same key is used for decryption. Well-known encryption algorithms for symmetric encryption are Blowfish, DES (data encryption standard), 3DES (triple DES) and AES (advanced encryption standard). For symmetric communication encryption, the main challenge is secure transmission of the key. This requires a secure channel, so the public Internet cannot be used.

*Asymmetric
encryption,
RSA*

Asymmetric encryption uses two keys, a private and a public key (a so-called pair of keys). The private key always has to be kept secure and must not be given to anybody. The public key can be published e.g., via Internet to anybody who wants to communicate with the key owner. For example, the sender of a message (often called Alice in the security domain) takes the public key of the receiver (called Bob) and uses it to encrypt the message. Only the private key of Bob can decrypt the message. The mechanism can also be used for checking the identity of a sender. To do this, Alice takes her private key to encrypt, so everyone who has the corresponding public key can verify that the message is from her.

*Digital signa-
ture*

It is costly in terms of computing power to encrypt the entire message and the mechanism can be used for identification anyway. Therefore, digital signatures are used which are generated using the private key to encrypt checksums of messages (a so-called hash value, named after the algorithm used to generate checksums). The hash value uniquely identifies the message, but cannot be used to reproduce its contents. The encrypted checksum is the digital signature of the message. Everybody who gets the message and possesses the corresponding public key can verify the identity of the sender as well as the integrity of the message, because if the hash value does not match the message any more, the message has been manipulated. Both mechanisms can be combined to digitally sign the message with Alice's private key and afterwards encrypt it with Bob's public key. An example for an asymmetric encryption algorithm is RSA (named after its developers Rivest, Shamir, Adleman).

*Digital certifi-
cate, PKI*

To guarantee that a public key published somewhere on a Web site is authentic, a system was created that embeds public keys into a kind of digital identity card called certificate. The standard for digital certificates is X.509. These certificates are issued by trustworthy companies or organizations called *certification authorities* (CA) that guarantee with their reputation and a digital signature that the public key is really owned by the person specified in the certificate. Examples for CAs are Verisign and TC

Trustcenter on a global basis and Deutsche Post or Deutsche Telekom in Germany. CAs can issue certificates to persons and also to computers or other CAs. This leads to a hierarchical system with so-called *root CAs* and other subordinated CAs, which is called *public key infrastructure* (PKI). As trust in the integrity of a CA is the basis for the whole system, it is sometimes also called *web of trust*. Principally, everybody can establish a CA and issue certificates. However, those certificates are not trustworthy as they are not issued by a well-known organization. To help users determine which CAs are trustworthy and which are not, every Web browser and most other applications that deal with certificates (e.g., email clients) have a list of well-known CAs. Certificates issued by a CA not included in the list leads to warnings and users can decide to accept them temporarily, permanently or not at all. Users can also decide to put the certificate of the new CA into the list of trustworthy CAs so that every other certificate issued by this CA is also considered trustworthy in the future.

Encryption can be implemented on different layers of the ISO/OSI reference model. Moving through the layers from bottom up, we firstly discuss encryption on the physical layer. Wiretapping is possible for cables, but is much easier for wireless connections as the medium is freely accessible to anybody. Especially wireless networks with medium range (e.g., WLAN) are often accessible from outside of company buildings. The encryption standard for WLANs is WEP (*wired equivalent privacy*), which is often considered unsafe due to small keys used (56 bit) by default and cumbersome manual transmission of the shared access key. In addition, WLAN access points are often badly configured so that recent tests still count over 50% of WLANs being open to wiretapping. The WPA standard (*Wi-Fi protected access*) is designed to overcome these problems with 128 bit encryption and easy configuration. The Wi-Fi Alliance is a consortium of companies that drives the development of WLAN standards. WPA2, the successor of WPA, uses AES for stronger encryption and became part of the IEEE 802.11i specification that deals with enhancements to the WLAN protocol family.

WEP, WPA

Encryption can also be applied at the network layer. One of the most commonly used protocols there is IPsec (*IP security*). As the name suggests, it builds upon the Internet protocol (IP). It is an essential part of IPv6 and an optional addition to IPv4. Asymmetric encryption is used to encrypt IP-payloads (*transport mode*) or entire IP-packets including headers (*tunneling mode*). The latter is used if the final destination of the packet is not using encryption and the end-to-end encryption is established transparently between two machines between the sender and receiver (e.g., to secure the packets' way through the Internet that connects two company Intranets). IPsec is the only way to secure UDP-based communication as all other security protocols build at least partly upon TCP to establish connections that are needed for key exchange. This raises complexity, since

IPsec

VPN

IPsec has to deal with reordering and fragmentation of network packets on its own.

If secure tunnels through the public Internet are used to connect two private networks, we speak of a *virtual private network* (VPN). Servers responsible for the encryption are often central servers also managing Internet access or providing firewall functionality (see below). They encrypt IP packets and wrap them in a new IP envelope addressed to their counterpart in the other private network. The packet is then routed through the Internet, unwrapped and decrypted at the VPN server in the second private network. Afterwards, the unencrypted packets are routed to their final destination. IPsec is often used for dial-up connections to an Internet service provider (ISP) and in cases people want to connect to their organization's Intranet. The tunnel is then being established between the client and the VPN server. Other protocols that can be used for VPNs are PPTP (*point-to-point tunneling protocol*) and L2TP (*layer two tunneling protocol*) which both are emulating a secure layer two infrastructure, so that there is one additional layer compared to IPsec (Figure 2-10).

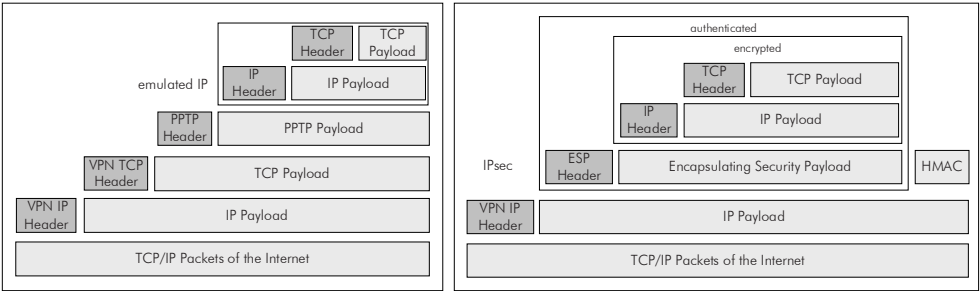


Figure 2-10. PPTP and IPsec packet layers.

SSL, HTTPS

Between transport and application layer, *secure sockets layer* (SSL) can be used to encrypt communication. Like IPsec, SSL uses a public key infrastructure and asymmetric encryption. It is widely used for Internet banking and transactions in online shopping and considered relatively secure when using 128 Bit keys. SSL encrypted HTTP connections are also known as HTTPS and use port 443. The successor of SSL is *transport level security* (TLS) which is also used as part of other protocols.

Authentication

Authentication is the process of uniquely identifying a user in order to get access. Authorization is used to determine whether users are privileged to access the respective system or network (see "Person dimension" on page 191). Three different ways of identification are possible. Users can authenticate using a unique possession (e.g., an identity card), knowledge (e.g., a password) or characteristic (e.g., biometric characteristics like iris or fingerprint). Often a combination of these is used to make authentication more secure, which is called multi-factor authentication (e.g., EC-card

together with a PIN). Authentication is used in different situations, either when users access a certain service in the Intranet, or before a connection to the Intranet is established.

Kerberos is an authentication protocol widely used within corporate Intranets for the first situation (default in Windows 2000 based infrastructures). It works in three phases:

Kerberos

1. Clients talk to the authentication server component and send the user name in clear text to it. The server looks up the corresponding password and uses it to send an encrypted message back to the client consisting of a *session key* and a key for the *ticket-granting server*. The client then uses the password provided by the user to decrypt this message. If the password is correct, the decryption will work and phase two can start. The user name was deleted on the server in the meantime.
2. The client uses (a) the session key to encrypt a timestamp and (b) the ticket key to encrypt the user name and the session key. These two parts are (c) sent to the ticket granting server together with the name of the service it wants to use on the server. The server uses the ticket key to decrypt the session key and sends back (i) the key for the requested service and (ii) a session ticket that depends on the timestamp and is valid only for the requested service and a limited time frame.
3. Finally, the client issues a request to the service it wants to invoke. It exchanges the session ticket using the services key. The service answers by sending back the timestamp increased by one in order to grant its identity and the whole handshake is complete.

The advantage of this system is that the user's password is never communicated over the network. The disadvantage is that the protocol is complex and slow.

An example for the second authentication situation is authentication of users that use dial-up connections to the company, e.g., from a customer site or from their home office. Public phone lines are considered substantially more secure than the public Internet. Therefore, authentication is more important than encryption. The service that provides these dial-up connections for modems or ISDN-adapters is called *remote access service* (RAS). Authentication protocols used here include PAP (password authentication protocol), CHAP (challenge handshake authentication protocol) and EAP (extensible authentication protocol).

RAS

The ISO/OSI reference model gets a bit shaken if you try to apply it to dial-up networks, because the protocols used are partly on higher levels, than the protocols that run on top of them. The reason is that dial-up protocols emulate low level ISO/OSI layers. For example, the *point-to-point protocol* (PPP) resides on layers 3 and 4 and runs on top of ISDN (layers 1 and 2). TCP/IP runs on top of PPP which also covers layer 3 and 4 respectively (see also Figure 2-10 on page 134).

Filtering

Filtering in this context means analyzing network packets, messages or files, checking them for certain characteristics and executing a defined action (e.g., delete, pass through) based on the findings.

Firewall

A service implementing filtering that analyzes network traffic and blocks unwanted communication is called firewall. One has to distinguish between network or enterprise firewalls on the one hand and personal firewalls on the other hand. *Network firewalls* are central devices or computers that reside somewhere in the network and serve as kind of sluice where all the network packets have to pass. Firewalls are often combined with routers as both have to control each network packet (though with a different purpose). *Personal firewalls* are applications that run on a PC and block unwanted traffic to and from this single PC. Firewalls decide upon letting a network packet pass through or blocking it based on rules. Those rules consist of (1) sender IP, (2) destination IP, (3) sender port, (4) destination port, (5) transport protocol, (6) direction, (7) time and (8) a block or pass through instruction. It can be further specified to passively drop the packet or actively deny it. In the first case the sender gets a `time out` message as response, in the second case the sender gets a `service denied` message.

Personal firewalls can additionally inspect the sending or receiving application and are therefore also called *application-level firewalls* in distinction to *packet-level firewalls* that do not have this ability. Sophisticated firewalls support *stateful inspection* which enables them to decide not only based on the current packet, but also based on history of network traffic. For example, a request can be blocked if the same sender address has already sent the same packet ten times before.

Intrusion detection

Intrusion detection systems (IDS) are trained with complex patterns of network packets that indicate attacks to the Intranet. If such a pattern is recognized, the administrator is alerted and if possible the attack is prevented.

Content filter, proxy

Content filtering works on the application layer and is used to prevent employees from accessing Web sites with offending or illegal contents and thus preventing abuse of company resources (hardware and Internet connection). Often, content filters are installed on servers acting as proxies. A proxy has several functions. Outgoing HTTP connections are pooled and can be monitored more easily. Only authenticated and authorized users can access WWW, FTP or other services. Web sites that are accessed can be cached on the proxy so that they are available faster if several employees access the same site which reduces Internet traffic. Internal IP addresses are hidden from the public Internet which saves money as only one public address is needed and also prevents port scanning attacks against clients.

Network address translation

Network address translation (NAT) is used for translation between internal packets (with a client's address as sender) and packets going out to the Internet (with a proxy's address as sender) and can be used independently of proxy servers. Requests to the Intranet get translated to the IP

address of the NAT server and the original sender port is being mapped to a new port on the NAT server. Many clients can be served since there are 65535 ports available and each client usually has only a few open connections at the same time.

Security measures for messaging. Computer viruses are a threat that becomes an increasingly serious problem for companies. Viruses can be harmless and e.g., just display a message at a certain time, but can also be harmful and e.g., delete important files or damage the operating system. The removal of such viruses is costly even for viruses without a harmful part. Antivirus programs detect viruses based on characteristics of its program code that can either be a complete program file (.exe or .com) or can be part of a normally functioning program file. A centrally-hosted anti virus software (together with intelligent patch management to timely close security vulnerabilities of software) has become essential for organizations. Today, attachments of email messages are the most important source for computer viruses and other types of malicious software. Therefore, an antivirus program should be installed on the email server in addition to the one on the proxy server.

Antivirus program

A second filter mechanism that should be installed on an email server is a spam filter. Spam or junk emails are email messages that are unsolicited from the recipient, often advertisements, but also other email messages that are distributed to masses of recipients. Spam filters try to differentiate between wanted and unwanted email messages based on rules that define characteristics of spam emails (e.g., sender address, keywords). One challenge that arises is that wanted email messages should be lost by no means. Therefore, messages that are classified as spam are usually not deleted, but moved to a junk mail folder or appropriately labeled and then transferred to the recipient.

Spam filter

Encryption can also be applied to email messages. PGP (*pretty good privacy*) is a free tool that can be used to encrypt email messages using asymmetric encryption and works together with most common email clients. Due to its wide-spread use with millions of users world wide, it is the de-facto standard. Secure MIME (S/MIME) is a standard developed and maintained by the IETF (Internet engineering task force). Both have in common that they use a public key infrastructure and support encryption as well as digital signing of messages for sender identification and integrity checking.

PGP, S/MIME

2.3.3 Application Infrastructure

Software applications build on basic services for storage, access, messaging and security discussed in previous sections. They support users by reproducing business logic with electronic functions. This helps users to accomplish and coordinate their tasks by e.g., automating workflows as

parts of business processes. Large parts of business logic are different for every industry, business process and maybe even company and is supported by business applications which can be both, standard or proprietary software systems. However, there are some parts of an application that are very similar or identical for almost every application. An application server is a foundation for applications and offers common functionality in terms of services, e.g., user management, transaction handling, object persistency. To understand this, first some principles of component-oriented programming are discussed which is a key for application infrastructures.

History of software development

In history of software development, programming evolved from low-level *coding in machine languages* (e.g., assembler) towards more abstract higher-level programming languages (e.g., Basic) that are closer to human understanding, structuring and solving of problems. Along with that, software got more sophisticated and complex so that mechanisms had to be found helping to deal with that complexity.

A first step was *procedural programming* that allowed better structuring of code with sub-routines, also called functions or procedures (e.g., Pascal or C). The next step was to encapsulate data and functions for manipulating this data in *object-oriented programming* (OOP) so that data is hidden from the object's environment and thus can be manipulated and accessed only in a clearly defined way. Each object was responsible for its own local data (called attributes) and provides functions (called methods) that other objects can invoke in order to retrieve or manipulate this data in a defined way. Every object can be tested on its own which reduces complexity. Examples for OOP languages are Smalltalk, C++ and Java. With the advent of software that is distributed over several computers the need for new concepts arose again. *Component-oriented programming* was introduced with the promise to provide these new concepts.

Definition of software component

Software components are binary units of independent production, acquisition, and deployment that interact to form a functioning system (Szyperski 1997).

Components can be seen as a unit in between an object and a program. They are not able to run on their own. They need a container that executes them and provides additional services.

Application server. This container is called application server. The following paragraphs give an overview of the services an application server provides for components.

Distribution

With OOP, it is possible to invoke methods of objects that run on other computers, e.g., with Java RMI or Microsoft DCOM, section 3.4.1, 219ff. The programmer has to specify exactly where this object is and what name it has. An application server relieves the programmer from this burden by providing a distribution transparent way of invoking remote methods. One

component calls another without having to care whether the other component runs on the same machine or not.

Application servers also take care of optimal distribution of the components by distributing them equally across all application server instances, so that the available machines are efficiently used. This is called *load balancing*. The precondition for this is that multiple instances of one component can be created. Application servers also take care of *multi-threading* and thread coordination. To run multiple server instances on multiple machines that act as one logical server to the environment is called *clustering*. Again, this mechanism should be supported by application servers. The third related mechanism is called *fail-over* and provides for replication of state information and take over of user requests if one server fails.

Load balancing

Enterprise applications frequently run into situations where several tasks have to either run completely successfully or have to be rolled back. This group of tasks is called transaction. An example for a transaction is a credit transfer. If 100€ should be transferred from one account to another and one account is charged 100€, then the money has to be added to the other account. If the latter does not work, withdrawal of money from the first account has to be cancelled, so that no money is lost. A mechanism that guarantees that is called transaction handling. In OOP, programmers have to take care on their own that transactions are handled correctly. With component-oriented programming, programmers only have to specify which functions are involved in a transaction and the application server monitors correct execution of transactions with a final *commit* or a *rollback* of the entire transaction.

Transaction handling

User authentication and authorization, i.e. the identification of a user and assigning of roles, privileges and respective granting of access to functions or data is another central part of any enterprise application (section 3.3). In contrast to OOP, programmers do not have to write their own user management module, but only have to specify which roles a user has been assigned, what privileges are associated with these roles and what privileges are required to access a function. The rest is done by the application server.

Access control

Some data held inside components should be kept permanent. This data is usually stored in (relational) database management systems. In OOP, programmers have to write calls to a database middleware that instantiates the driver for the DBMS, establish a connection to the database, execute SQL statements and close the connection afterwards. This is inefficient and tedious. With an application server, programmers only specify an object-relational mapping from attributes of the component to fields and tables of the database. Loading from and saving to the database is handled by the application server. It guarantees that data is stored persistently and kept current. The application server also performs runtime optimizations so that not every single database operation has to open its own connection

Persistence handling

which is called *connection pooling*. The mechanism of making object data permanent is called persistence handling.

Maintenance

An enterprise application follows a certain life-cycle: additional functions are implemented, errors are corrected in the source code and the hardware that executes the application eventually has to be upgraded or supplemented with increasing numbers of users or transactions. These episodes in the application life-cycle usually lead to unavailability of the system due to maintenance tasks. An application server allows loading of new program code (e.g., extensions, corrections) during runtime. This feature is called *hot deployment*. The class files with the additional or corrected code are copied to a specific folder in the file system (hot folder) and the application server loads it automatically as soon as possible. If an additional server is added to a cluster, or a server is stopped to upgrade the hardware, the system does not have to be stopped. To help administrators to identify potential problems or performance bottlenecks, application servers provide extensive functionality for logging, monitoring and auditing. *Logging* means that the system writes a log entry to a log file or to the system's event log every time an exceptional event occurs (warning or error). *Monitoring* means that the application server provides current status information about resource allocation (disk, memory and network) and runtime state (e.g., number of users logged in). *Auditing* is often based on logging and means in this context the systematic verification of user actions or application transactions.

Runtime optimizations

Application servers relieve programmers to a certain extent from manual optimization of runtime performance. Besides the above-mentioned connection pooling and load balancing, application servers also support caching of database queries, temporary persisting of components that are currently not used and complete management of the life-cycle of a component. Caching is keeping data in memory (or on fast media in general), although they are currently not needed any more in order to have them quickly available when they are needed next time. Components not needed at the moment can be destroyed in order to free resources. If such a component is needed later on, a new one is created (*life-cycle management*). Components holding state information that are important in the future must not be destroyed. Therefore, their state is stored on a disk which is called *dehydration*, before the component is removed from memory. If it is needed again later, its state is restored from disk and the component can be used again.

Differences at a glance

Functions of application servers described above differ from traditional programming only in the way they are invoked. Instead of making explicit calls to specific middleware APIs that do the job, only the need for such a call is specified in component-oriented programming and the application server fulfills the need at runtime. Thus, these mechanisms are also called *implicit middleware*. It can be seen as a higher level of abstraction. To

stress an important part of the discussion above, we can note that OOP supports reuse at build-time whereas component-oriented programming supports reuse at runtime.

An additional trend in developing enterprise applications is the need for an easy way to generate Web front ends for them, due to the fact that Web browsers are becoming the main access tool (section 5.2, 374ff). Application servers address this need by providing HTML derivatives that allow to embed server-side script code that is evaluated at runtime, so that dynamic HTML pages can be realized. Template libraries that help to compose GUI elements from basic HTML tags further support ease of use.

Web front ends

Component Frameworks. The first standard approach to component systems is CORBA (common object request broker architecture) that has been developed by the Object Management Group (OMG). The first widely adopted version 2.0 was specified in 1995 and released in 1997. Current version is 3.0 from 2002. The core of a CORBA system is a central application called *object request broker* (ORB) where all objects have to register, so that other objects can reference them and invoke their methods. The protocol used to communicate with the ORB is called IIOP (Internet inter-ORB protocol, TCP port 684). The name already tells that it can be used not only for communication from objects to ORB but also from one ORB to another. The methods that an object wants to register are specified in a language called IDL (*interface definition language*). IDL provides a set of data types independent of programming languages that can be used for parameters and return values. Each supported programming language has to provide a mapping from IDL data types to language-specific ones. Thus, CORBA provides both language and distribution transparency, but it does not provide a complete application infrastructure according to the functions that were discussed above.

CORBA

Java offers a different approach and can be seen as a reference for component frameworks. Services discussed above are provided by a central application called Java application server (JAS). There is an overwhelming number of technologies and standards connected to JAS subsumed under the headline *Java 2 enterprise edition* (J2EE, in contrast to Java 2 standard edition J2SE and the Java 2 micro edition J2ME). J2EE is under control of Sun Microsystems and is a definition for a large set of interfaces on the one hand and an implementation of these definitions (Java class files) on the other hand. Besides the free implementation from Sun, there are several commercial and open source implementations of the J2EE specifications. IBM, BEA (now Oracle) and others provide own commercial implementations with partly significantly increased performance. JBoss (now Redhat) is an example for an open source implementation that receives increasing interest from business customers.

J2EE

Java is an interpreted language which means that every Java program needs the *Java runtime environment* (JRE, also called *Java virtual*

JRE, JVM

machine, JVM) in order to be executed. The Java compiler does not create directly executable machine code, but Java bytecode which is interpreted by a JVM. Therefore, Java programs run on any platform that has a JVM so that Java is platform-independent. The disadvantage is less performance because of interpretation overhead, which can be partly compensated with *just-in-time compilation* (JIT), the compilation of byte code into machine code at runtime.

EJB

An application server can be seen as an extension of a JVM that provides additional services. The central concept in J2EE are *Enterprise Java Beans* (EJB) which must not be mixed up with Java Beans, a client-side technology. EJBs are components running in an application server. There are different types of EJBs, namely entity beans, session beans and message-driven beans. *Entity beans* are briefly spoken wrappers around data or business data encapsulated in objects. They usually load and save their data from and to a DBMS and are therefore persistent. Application servers provide a mechanism for automatically handling database operations which is called *container managed persistence* (CMP). *Session beans* hold business logic to manipulate entity beans and can be seen as a software implementation of a workflow action or an entire workflow. They are usually not persistent and their lifetime is (despite the name) not necessarily as long as a user session lasts (although it can be). Session beans come in two flavors, *stateful* and *stateless*, meaning that they preserve an internal state over time and various method calls or not. *Message-driven beans* are messaging objects that are similar to session beans as they represent actions, but in contrast to them process requests asynchronously.

Supporting
technologies

In the following, technologies are described that supplement EJBs. The *Java messaging service* (JMS) is used by message-driven beans and acts as a vendor-independent interface to message-oriented middleware (see "Messaging" on page 108). Therefore, it is similar to the *Java database connectivity* (JDBC) which provides vendor-independent interfaces to communicate with a DBMS. It is not surprising that there is also an interface to directory services called *Java naming and directory interface* (JNDI). JNDI plays an important role in J2EE as it is used to register and locate components. Without JNDI, there is no communication between enterprise beans which does not mean that an LDAP directory service is required for a small J2EE application. A simple implementation for JNDI (often without persistence) is delivered with every application server. Other APIs worth mentioning in this context are *Java mail* for email messaging, *Java transaction API* (JTA) and *Java transaction service* (JTS) for secure transaction handling and *Java authentication and authorization service* (JAAS) for securing access to applications.

The following specifications are not directly related to the features described in section "Application server" on page 138, although they are essential for modern applications. The *Java API for XML parsing* (JAXP)

provides interfaces for XML parser (section 3.1.3, 157ff), the *J2EE connector architecture* (JCA) provides a defined way of accessing third-party enterprise systems (e.g., on IBM mainframes or ERP systems like SAP R/3). *Java servlets* and *Java server pages* (JSP) provide functionality for Web presentation of the application. Altogether, Java application servers provide all functions of application servers as described above and offer even more.

Application infrastructure provided by Microsoft is tightly integrated into the Windows operating system which makes it harder to locate them since no central application exists. Component technologies of Microsoft are subsumed under the headline .NET (spelled “dot net”). Starting with the similarities to Java, .NET is a platform for interpreted programming languages. The runtime environment is called *common language runtime* (CLR) and is interpreting code in Microsoft *intermediate language* (IL). There is also a JIT compiler that can be used to translate IL code into native machine code at runtime. The runtime environment is currently available for Windows platforms from Microsoft. An open source implementation of the .NET framework called *Mono* which is currently sponsored by Novell brings at least the core APIs of .NET to Linux and some other UNIX platforms.

The corresponding technology to EJBs is called *.NET managed components*, often referred to as *COM+ components*. COM+ is an enhancement of COM (*component object model*) and DCOM (distributed COM) and does not necessarily have to be a .NET managed component, so it is not quite right to mix those two up. .NET managed components are a subset of COM+ components. To make the confusion complete, sometimes the term *EnterpriseService* is used in that context which is the name of the packet that includes the root class (*ServicedComponent*) all components must inherit. COM+ components are equivalent to session beans.

Message-driven COM+ components can be realized with the help of the *Microsoft Message Queue* (MSMQ), but there is no real equivalent for entity beans. Automatic persistence could only be managed manually or with the help of third-party add-ons like NHibernate until recently Microsoft introduced .NET 3.5 that allows to generate object-relational mappings using the tool *SQLMetal* and query as well as update data using language integrated queries (LINQ). LINQ uses a SQL-like syntax to access and manipulate data sources no matter whether they originate from SQL databases, XML files or in-memory object collections. Prior to LINQ, database access worked with the ADO.NET API (ADO = *ActiveX database objects*). The *System.DirectoryServices* namespace provides classes for communication with directory services. In contrast to J2EE, directory services are of minor importance in .NET. Transaction support is provided by the *Microsoft transaction service* (MTS) and the *distributed transaction coordinator* (DTC). The *System.Web.Mail*

.NET

COM+

MSMQ, LINQ
ADO.NET,
MTS, DTC

namespace bundles classes for email messaging. Authentication and authorization is not packaged within a special part of the .NET framework, but spread across multiple namespaces. The most important ones are `System.Web.Security` and `System.Net`. Not surprisingly, the recommended authentication modes are Windows-integrated authentication (either with local users of the server or users in a Windows domain) and role-based authentication using group-membership information from Active Directory. However, it is also possible to use forms authentication (user name and password are entered in an HTML form) or Microsoft passport authentication as well as storing data about roles in SQL Server or Access databases.

ASP.NET

Corresponding technologies for most other Java APIs described above exist as well. Classes for XML handling are in the `System.XML` namespace, Web front-ends are created using the ASP.NET API. Only a JCA equivalent is missing. The only product that could be seen as a single connector in a JCA sense is the MS *host integration server* (HIS), which can be used to connect to IBM mainframe systems.

Differences to Java

Main differences between .NET and J2EE are that .NET is programming language independent (C#, Visual Basic, C++ and some other languages can be used to generate IL code) whereas Java is platform-independent. The infrastructure for J2EE is bundled mainly in a single program called application server whereas .NET uses a number of services from different Microsoft products, e.g., Windows, MSMQ and the Internet Information Server (IIS). Another difference is that there is no .NET support for container-managed persistence.

Recent developments

Summed up, both Java and .NET provide a solid basis for modern applications and provide a number of APIs to support most routine programming tasks. A lot of programming tasks can be solved with a few lines of code. Despite that, complexity for programmers was not reduced that much but mainly shifted from solving problems with a lot of code and basic APIs to learning the substantial complexity of component frameworks with all dependencies and possibility, in order to find the best solution for a given problem. Complexity of the frameworks still rises and it even seems that it rises quicker than ever. The introduction of additional APIs like Enterprise Java Beans 3.0 that now supports annotation like .NET is just one of the many new features in J2EE 5 and Java SE 6. Windows Workflow Foundation is an example for a new feature in .NET 3.0. It introduces a complete workflow framework and a basic workflow engine that makes it easy to write custom workflows and integrate them into own .NET applications or Microsoft Sharepoint. LINQ changes the way to write applications with database backend fundamentally and both Microsoft and Sun strive to keep up with the bunch of new specifications that come up in the Web Service area (e.g., WS-Addressing, WS-Policy, Semantic Annotations for WSDL).

Enterprise service bus (ESB). While component frameworks and application server provide a reliable basis for a single enterprise application, in EKI the integration and data exchange between different applications is a key requirement. To achieve that, standards-based middleware products have been developed under the headline enterprise service bus.

An enterprise service bus can be defined as a middleware platform that provides data messaging, transformation, routing and monitoring based on Web service technologies for loosely coupled software components.

The basic functionality of an ESB is to broker data between different software components. Instead of creating one-to-one relationships between every pair of components that need to exchange data, the ESB is introduced as a central hub for communication. That leads to a reduction in complexity for data integration (section 3.1, 153ff).

*Brokered data
exchange*

ESBs typically include a repository used to resolve service addresses at run time so that the source component only has to label the message with a target ID. The ESB then is responsible for translating the ID into a valid target address that can be contacted using one of the available transport mechanisms. Often, they are also capable of routing messages based on a predefined set of criteria.

*Address indi-
rection*

Besides the messaging protocols discussed in section 2.2.3, ESBs usually support more reliable communication methods. A form of asynchronous transport for messages that is only used for communication between servers is *message-oriented middleware* (MOM, also called *message queues*). In contrast to email, where delivery is not guaranteed, message queues provide a reliable way of sending messages where senders can be sure that messages are delivered. Message queues use proprietary protocols on different TCP ports (e.g., 1881 for IBM or 1801 for Microsoft). Some products additionally support SSL encrypted delivery. Common products in that system class are BEA message queue, IBM MQseries, Microsoft MSMQ and SonicMQ.

*Messaging,
MOM*

Besides MOM, ESBs need to support Web service technologies like HTTP, XML, XSD, SOAP and WSDL (section 3.4.1, 219ff). The principle for message exchange is always the same. Data is received using one of the supported receiving methods, e.g., HTTP, SMTP, FTP or the file system (drop folders that are frequently checked for incoming data). Depending on the receiving channel and the contents of the message or its envelope, the sender and message type are identified. If the data is not already in XML format, it is converted into XML and compared to a defined schema to make sure it conforms to the specification associated with the sender and message type. Afterwards the target for the message is determined. If the message is not yet in the correct format that the target expects, the ESB transforms the message using XSLT (section 3.1.5, 165ff).

*Routing and
transformation*

or proprietary technologies. After it was made sure, that the message conforms to the target schema, it can be sent to the target system using a standard sending adapter, e.g., MOM, or a target system specific adapter, if the target system is not yet supporting standard technologies, e.g., legacy or mainframe systems.

*Product exam-
ples*

All major manufacturers of software infrastructures offer products that provide ESB functionality either in a single product or as a combination of multiple products. Examples are IBM WebSphere Enterprise Service Bus, Microsoft Biztalk Server, Oracle Service Bus, SAP Netweaver Process Integration or Sun's Open ESB.

Case example

Global Industry Group, a large company with 10,000 office workers world-wide in a hundred locations has to manage around 250 switches. Its network is structured in a star topology with four core switches that build the company backbone in the headquarters and are arranged in a double ring to provide maximum fault tolerance. Those four switches are connected using 10 GB/s fibre optical cable and cost over a hundred thousand Euros. Around 800 servers provide application services in the company. Most of them are located in the branch offices serving as domain controller, file server and offering basic networking services like DHCP. The 50 plants have additional database servers and specialized application servers. Company-wide services such as email, Internet presence, document management and ERP systems are hosted in the headquarters on only a few dozen servers, connected via Gigabit Ethernet. Most central servers are connected to one of the three SANs that provide central storage with a net capacity of 15, 20 and 30 TBs mirrored between the two data centers that are located in the same city as the headquarters. The central anti-spam system receives some 2.2 million emails a day, although only around 25 thousand emails are serious business mail which means nearly 99% of the whole mail traffic is spam.

The Intranet consists of the company's LANs that are connected mainly by MPLS lines with transfer rates between 1 and 8 MBit/s. Only a few large administration offices with several hundred workers have high speed lines with 20, 50 and 100 MBit/s transfer rates. In some developing countries, only bundled ISDN lines with 128 or 256 kB/s combined transfer rate are available and must suffice for a dozen or more employees. ISDN lines also serve as fallback in case that the MPLS lines fail. A few locations also have satellite backup lines with 256 kB/s but a very large latency of 500 ms and more.

The company currently moves parts of the documents that are stored on decentral file servers to central DMS, so that several locations need bandwidth upgrades in order to work continuously with the documents. Another current project has the goal to increase availability of WLAN in all administrative buildings. This requires foremost solid planning of positions of access points and direction of antennas so that they achieve maximum coverage and still can easily be integrated into the LAN. In order to enable WLAN access to the Internet for guests in the company, a separate virtual LAN area has to be created that only gives way to the Internet, but has no routes to the corporate Intranet. Authentication for WLAN access of employees uses X.509 certificates on notebooks together with domain membership of machines in the corporate Active Directory. This mechanism is also used for home offices of 400 employees with part or full time workplaces at home.

Questions and Exercises

1. Make yourself familiar with the commands `ipconfig`, `ping` and `tracert` (on Windows operating systems, for Unix, Linux and Mac OS the command is `ifconfig`). Use the option `"/?"` or `"-?"` to get a help message that explains the usage of the respective command line tool. Note the IP address of your own computer and determine to which network class your computer belongs.
2. Use the `tracert` command to find out how many hops a network packet has to make to reach a server in New Zealand (e.g., `www.nzx.com`). The hostnames of routers often contain information about the location of the router and the provider that owns the network (e.g., `sl-bb22-chi-15-0.sprintlink.net` is located in Chicago and owned by Sprintlink). Which route does the ICMP echo packet travel? Identify major network providers. Use `tracert` several times with different targets which reside in different countries (e.g., University Web servers). Identify major network hubs.
3. Send an email message from a fictive address to your regular email account. All you need is an SMTP server where you have an account and a telnet session on port 25 to this server. First of all you should say hello to the server and tell him the name of the sending machine. The corresponding command to do that is `"HELO name"`. Then you tell the server the sender address with the `FROM:sender@domain.com` command. Then you say who should receive the mail with `RCPT TO:receiver@domain.com`. Each command is followed by a `<return>`. Then you can start the message with `DATA:` followed by a return. Within the message body, you can use the `SUBJECT:here` comes the subject command to specify the subject and type a short message in the next line. Conclude the message with a dot in a new line and press return again to send the mail. The server should state something like `message accepted for delivery`.
4. What is LDAP? What is it used for in enterprise knowledge infrastructures? Write the complete LDAP path for Bernhard Grzimek in Figure 2-7 on page 121.
5. Try to receive a HTML page from a Web server using telnet. Connect to `www.uibk.ac.at` on port 80. Type `GET` and press return to get the start page of the server. Why is the connection lost after the HTML code was transferred? Use a browser to connect to the Web server and have a look at the page source (right click). Try to retrieve the Web page `iwi/ronaldmaier.html` from the same Web server. Use `GET iwi/ronaldmaier.html` in the telnet session to port 80. Why is the content different from the page displayed in the browser?

6. Use the command line version of an FTP client (comes with Windows 2000 and XP and is also available on most other operating systems) to download files from an FTP server. Open a command window and type `ftp`. Open a connection with `o server_DNS_name`. List the contents of the directory with `list` or `ls`. Change to other directories on the server with `cd directory_name`. You can do most usual file operations on the FTP server, like `cd` (change directory), `md` (make directory) and `rm` (remove file) if you have permissions to do so. You can do the same operations from within you FTP client on your local system if you put an exclamation mark “!” in front of the command, so `!ls` lists the contents of a directory on your local machine. If you have identified a file on the server you want to download use `get filename` to initiate the transfer. With `close` you can end the session and leave the client with `quit`.
7. Install a freely available personal firewall on your computer that allows you to create rules regarding ports, addresses and protocols (e.g., AtGuard, iptables, Kerio, Tiny). Create rules that allow outgoing ping commands, DNS resolution and access to the Web over HTTP.
8. Advanced readers can install a network sniffer, e.g., Wireshark. Start to capture the network traffic, then do a `ping` and visit a Web site. Then stop capturing the network traffic. Why are there UDP packets going to port 53?
9. Use the “component services” in the “administration” program group (Windows 2000 or XP) to inspect the COM+ components that are installed on your system.
10. Which ones of the following statements about protocols on ISO/OSI layers are correct:
 - (a) HTTP, TCP and FTP are application layer protocols.
 - (b) Ethernet and IP are network layer protocols.
 - (c) SMTP, HTTP and FTP are application layer protocols.
 - (d) TCP and UDP are transport layer protocols.
 - (e) HTTP and HTTPS are WWW layer protocols.
11. Explain in detail what happens and which protocols are affected when you send an email to your friend in a different country. Sketch the network as a graph and explain what the protocols are needed for.
12. What is changed if communication needs to be secure? Which alternatives are there for establishing secure communication between two subsidiaries of a company located in different cities?
13. Sketch the differences between Internet, Intranet and Extranet. Explain how VPN works and sketch a network in which VPN is used as a graph. Write the IP address of the packet sent between computers on the lines connecting nodes in the graph.

Further Reading

Since many networking technologies like TCP, IP and HTTP are well established and have been quite stable for a decade there are several good textbooks and excellent Web resources that deal with network basics. For application infrastructure, currently only books that deal with either EJB or .NET are available on the market.

Löwy, J. (2005): *Programming .NET Components*. 2nd edition, O'Reilly, USA

The book provides a complete introduction of the .NET framework at a detailed technical level together with a focus on component-oriented programming. The author shows how to build large-scale enterprise applications from architectural aspects up to concrete technical solutions with Microsoft technologies.

Peterson, L. L., Davie, B. S. (2007): *Computer Networks. A System Approach*. 4th edition, Morgan Kaufmann, San Francisco

Peterson and Davie also give a comprehensive account of network technologies and also cover all ISO/OSI layers. Due to the recency of the latest edition, it is current in its coverage at the time of publication of this book.

Roman, E., Ambler, S., Jewell, T. (2005): *Mastering Enterprise Java Beans*. 3rd edition, Wiley Publishing, New York

The Java 2 Enterprise Edition with Enterprise Java Beans as their main concept can be seen as the mother of modern application servers. This book describes the challenges for application developers, how to solve them with application servers and discusses the details of related Java technologies.

Tanenbaum, A. S. (2003): *Computer Networks*, 4th edition, Prentice Hall, Upper Saddle River, New Jersey

Tanenbaum covers every aspect of computer networks in great detail and can be seen as the reference in that area. Despite its depth and technical style, it is still very readable.

Enterprise Knowledge Infrastructures

Maeder, M.; Hädrich, Th.; Peinl, R.

2009, XII, 445 p. 82 illus., Softcover

ISBN: 978-3-540-89767-5