JARL NILSSON
GENUINE SYSTEMS
SKOLGATAN 132
903 32 UMEÅ

Norsk Data

Introduction to DATA COMMUNICATION

ND-60.181.2 EN

Introduction to
DATA COMMUNICATION

ND-60.181.2 EN

This introduction to data communication is written for those of you who need to know some of the important concepts. The manual is general and does not describe a specific product.

Chapter 8 is a short introduction to some important data communication standards. It can be read independently of the rest of the manual.

We assume that you have some basic knowledge about computers and data processing.

CHAPTER 1
PRELIMINARY ENTRY TO THE FIELD

PURPOSE AND APPROACH

The main purpose of data communication is the transport and exchange of data, not the processing of the transported data. A basic requirement is that data transport and exchange proceed in an orderly and controlled manner.

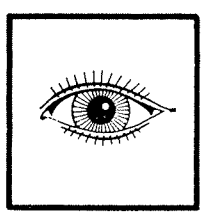
During this data exchange a number of communication problems may arise. It happens, for example, that transmitted data is distorted because of noise on the line. Transmitted data is sometimes lost. Data communication systems are designed to solve these problems or to prevent these problems from occurring.

Organization of the Manual

The general approach chosen in this manual is going from the specific to the general, and then filling more specifics into the general structure. The manual starts with specific information about physical transmission of bits, in chapter 2, and continues with network components and protocols. In a later chapter generalities about communication systems are explained, like the OSI model, before some more details are filled in, like specific information on X.25 and Ethernet. The OSI model defines some general principles of data communication architecture.

The first chapter serves as a preliminary introduction to data communication without going into detail. The details, like how a modem functions, are postponed until chapter 2.

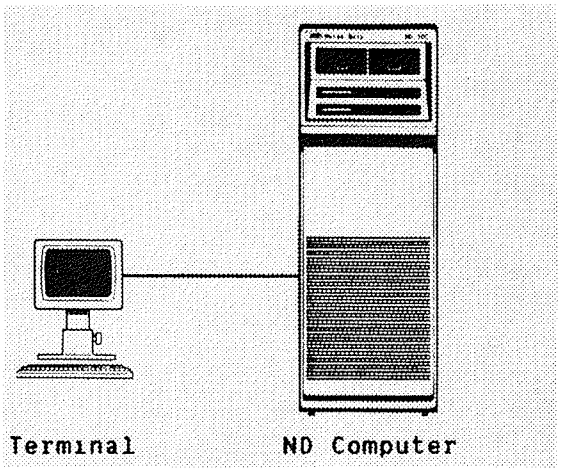
Important note!



A communication system has to be reliable at every possible level. This implies that the physical transmission of bits must function correctly and the same "languages" must be spoken in each end of the communication medium so that no misunderstanding can occur. Therefore it is important to understand the different layers of the Open Systems Interconnection (OSI) model.

CONNECTION BETWEEN A COMPUTER AND A LOCAL TERMINAL

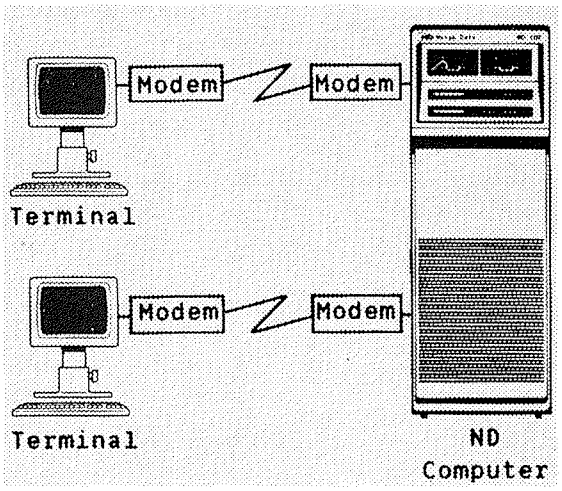
We assume you are familiar with terminals, and you know that a terminal is connected to a computer. If the distance between the terminal and the computer is not too big (say a few hundred meters), it is possible to



connect the terminal by a simple cable (galvanic circuit). The interface between the terminal and the cable is usually a "current loop interface" when dealing with ND computers. This interface will be described later.

The maximum possible distance between the terminal and the computer varies with the transmission speed and the electrical environment around the cable. Ordinarily we will not be able to go beyond 50 meters at a 9600 bits per second (bps) speed, and not beyond 100 meters at 4800 bps. We may, however, go beyond these distances by employing amplifying boxes (often called "local modems") between cable segments.

REMOTE TERMINAL CONNECTIONS

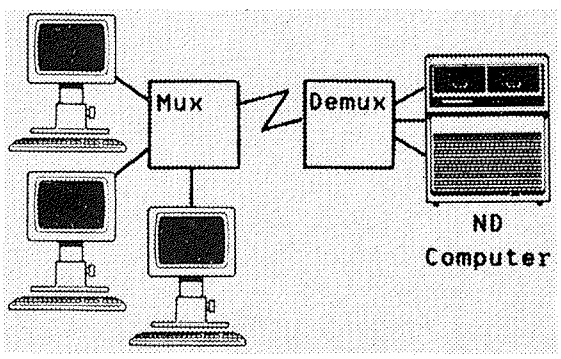


On longer distances special problems with cabling arise:

- The cost of stretching our own cables.
- We may not be allowed to stretch our own cables because we do not own the premises.

The telecommunication companies already have facilities for transport of information, the most common one being a telephone line. Since the telephone line is made for voice transmission, the terminal signals and computer signals must be converted to voice data. This conversion is the task of the modem. Many terminals may be connected to one computer in this way.

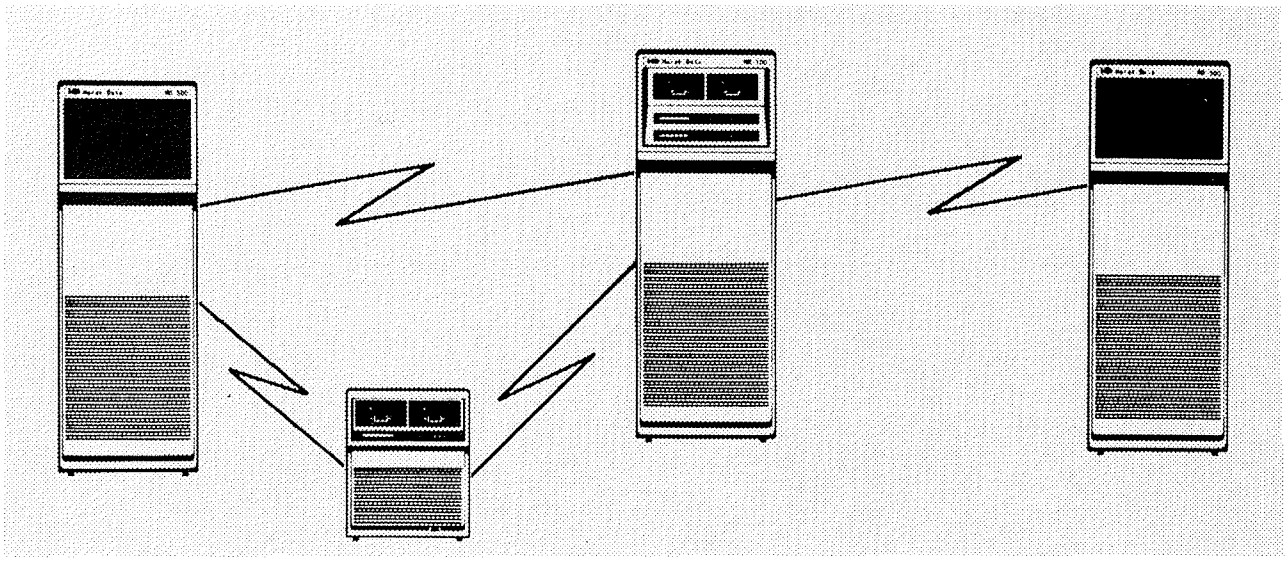
Multiplexing



Those terminals which are located on the same site, may share a telephone line to save transmission costs. This is called multiplexing. The individual data streams are multiplexed onto the same line, and they must be demultiplexed (separated) in the other end.

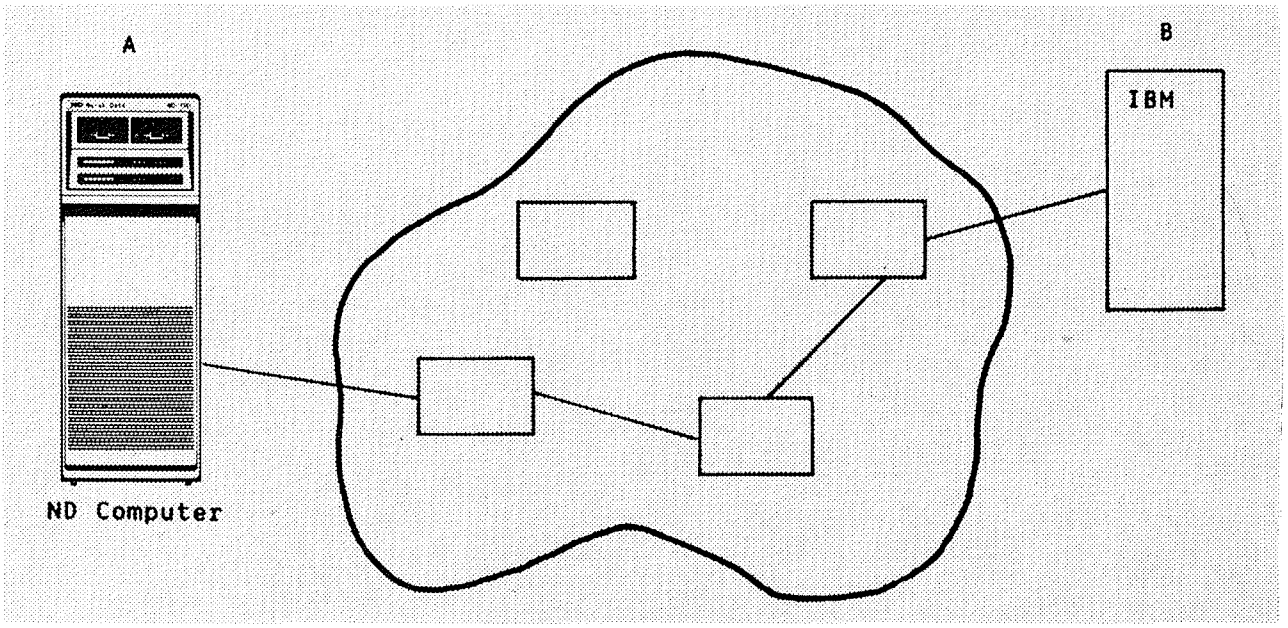
COMPUTER NETWORKS

Computers may also communicate with each other, and be organized in a network. In this example telephone lines are used as transmission media. It is equally possible, provided we deal with local distances, to use ordinary cables instead of telephone lines.



Computer network using telephone lines

Some of the public transmission facilities are computer networks themselves. For example a public data network can handle data transfer between any two subscribers. The computers (nodes) inside the public network are able to route the data correctly from computer A to computer B.



Public data network

CHAPTER 2
TRANSMISSION CONCEPTS

PHYSICAL CHARACTERISTICS OF DATA COMMUNICATION

This chapter explains some fundamental properties of what we call the lowest level in the data communication process. This lowest level has to do with the physical transmission of information across a communication medium (here called channel).

TRANSMISSION MODES

A communication channel transmits and receives information in one of three modes:

Simplex

Under simplex operation, data is transmitted unidirectionally. That means only one station transmits and the other receives.

Half-duplex

Under half-duplex operation, data is transmitted bidirectionally but not simultaneously. That is, both stations can transmit, but only one station transmits at a time.

Full-duplex

Under full-duplex operation, data may be transmitted in both directions simultaneously.

TRANSMISSION FORMATS

The computer has a hardware interface which presents the data to the communication channel as a serial bit stream. This is also true for the terminal. Most computer connections concern the use of character oriented devices such as terminals and printers. The basic unit of information used for organizing the data is therefore the character. The bits sent as electrical signals are consequently packed as characters and the receiver has to be able to recognize the serial stream of bits as characters. The receiver must be able to detect the start of a character and its end.

There are two techniques for recognizing and separating characters from the serial bit stream.

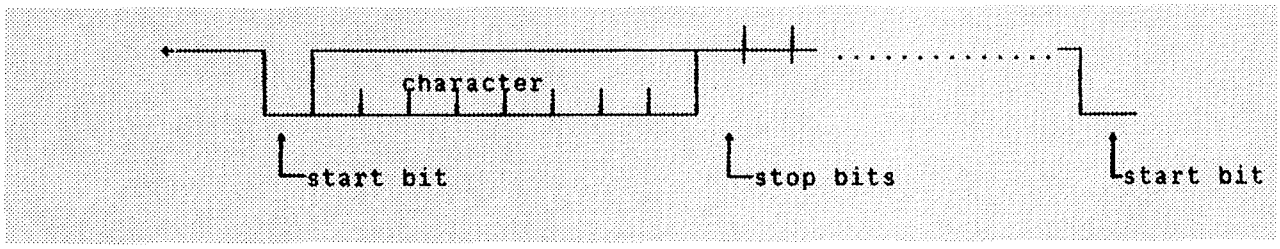
- Asynchronous transmission format
- Synchronous transmission format

ASYNCHRONOUS TRANSMISSION FORMAT

In asynchronous transfer the receiver of the bits detects the start of a character because redundant information is used.

Each character to be transmitted may be preceded by a start bit and terminated by two stop bits. Asynchronous transmission is sometimes referred to as "start/stop transmission". The function of the start bit is telling the receiver where the new character starts. The function of the stop bit is to put the channel back into the initial position, so that the next start bit will be perceived as the start of the next character. The receiver knows the end of the character by counting the number of bits.

Asynchronous character framing is designed for a situation where the characters are transmitted intermittently. This is typical when a person is sitting at a keyboard of a terminal without buffering capacity. Then each character is sent immediately when the corresponding key is pressed.



Asynchronous transmission format

Typical applications

- Computer to nonintelligent terminal communication
- Communication with slow printers

- Communication with unbuffered terminals

Main disadvantage

- Redundant information (the start and stop bits) creates overhead.
- Transmission is slow. In Norway the speed is limited to 1200 bits per second (bps) full duplex and 2400 bps half duplex if the connection is a telephone line.

The two devices (terminal and computer) have to agree upon the number of bits in a character and the bit time. The ASCII character set, which uses 8 bits, is the most widely used. The bit time is set by deciding how many bits per second to transmit.

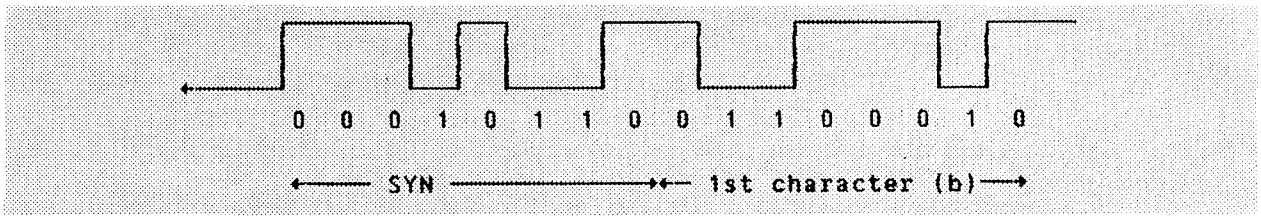
SYNCHRONOUS TRANSMISSION FORMAT

Synchronous transmission uses a block oriented format. No start or stop bits are used.

There are situations where larger volumes of data need to be transferred, such as whole files from intelligent terminals or between computers. To transfer larger volumes of data, higher speeds are used. Synchronous framing is used where a number of characters or blocks of bits are to be transferred.

Synchronous character transfer using serial transmission means that all that appears in the channel is a continuous stream of bits. The receiver has to combine these bits into characters, therefore the first bit of the character has to be found. This problem is solved by using special synchronization characters.

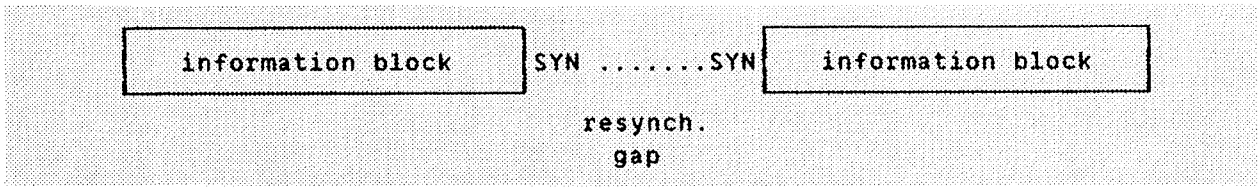
Once the transmitting equipment is switched on, a local clock is started, and a continuous stream of synchronization (SYN) characters is transmitted. A local clock is also started in the receiver end when a SYN character is detected. Once the SYN character is found, the character boundary is determined. This is true for a big class of transmission protocols known as BISYNC. The HDLC protocols also use synchronous transmission, but they do not use SYN characters.



Synchronous character framing

During information transfer the characters are collected by counting the bits which are found by using the local clock. This local clock is synchronized with the local clock at the transmitter, hence the term synchronous transfer.

Over a period of time the clocks tend to become unsynchronized and data may be lost if the bits are not correctly interpreted. This is done by resynchronization at frequent intervals. Resynchronization is accomplished by sending SYN characters.



Synchronous transmission blocks

Typical applications

- Transmission between computer and intelligent terminals
- Communication with buffered devices
- Transmission speeds from 2400 bps to 19200 bps. Lower speeds are possible, but unusual.
- Computer to computer communication

In some cases computer to computer communication uses asynchronous transmission.

The transmission equipment needed for synchronous transmission is in general more expensive than for asynchronous transmission.

NOTE!

Asynchronous and synchronous devices are not directly compatible.

ANALOG TRANSMISSION

Certain transmission channels require analog transmission. A telephone line is one example. Analog transmission of bits is performed by modulating sine waves. We will explain in a later section how the modulation takes place. Now our purpose is to give you three important characteristics of sine waves.

Amplitude

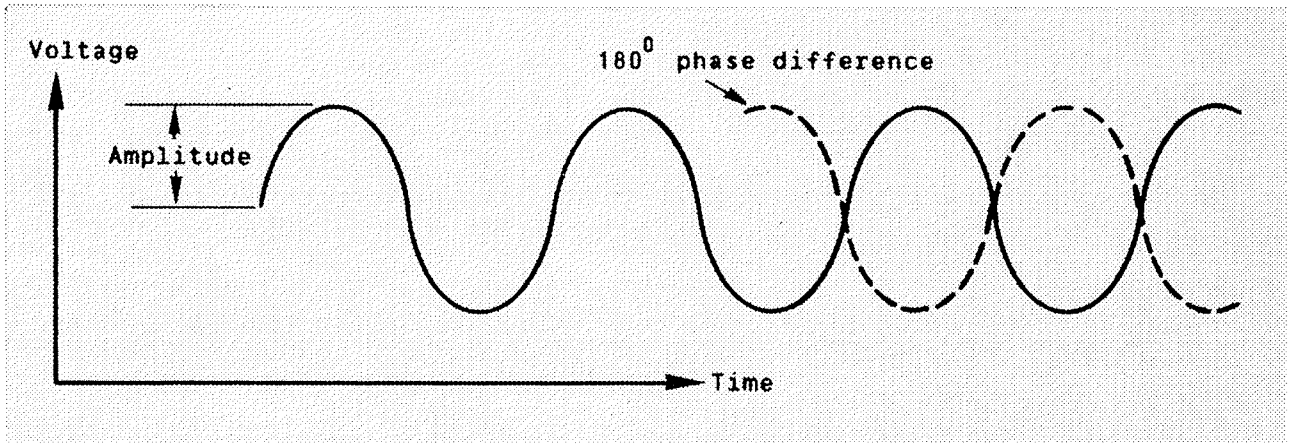
Amplitude is a measure of the height of the wave. For instance a wave in a piece of wire has its amplitude measured in volts.

Frequency

The frequency of a wave is, for our purposes, the number of times a wave shape repeats in a second. We measure it in Hertz (Hz) which means cycles per second.

Phase

The phase is a relative measure: It is the difference in time between two waves. The phase difference is normally given as an angle. In the figure a phase difference of 180 degrees is depicted. You may see that an additional 180 degree displacement of the second wave will cover the original wave. This is a phase difference of 360 degrees (180 + 180) which is indeed equal to 0 degrees, i.e., no displacement.

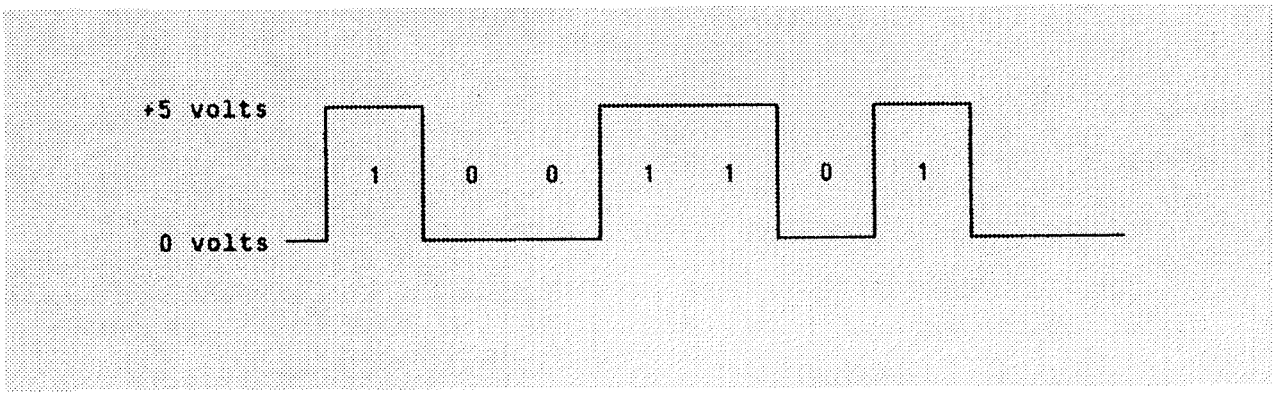


Sine Wave Characteristics

DIGITAL TRANSMISSION

As bits are digital, this is a more direct way of transmitting bits than the analog. The bit patterns can be represented by voltages using only two values, say 0 volts and +5 volts. This can be depicted as a square wave. When we introduced the asynchronous and synchronous transmission modes, we used

square wave coding of the bit patterns. Note that synchronous transmission may use either analog or digital transmission. The same applies to the asynchronous mode.



Square Wave Representation of a Bit Pattern

TRANSMISSION RATES

The transmission rate used in communication spans from 50 bits per second (bps) to 50 Megabits per second (Mbps). A term which is widely used (and misused) is baud. The baud rate is really the signalling rate of the channel which is not necessarily equal to the bit rate. The reason is that it is possible to signal more than one bit at a time.

In this manual we will consistently use bits per second to avoid confusion.

Standard speeds

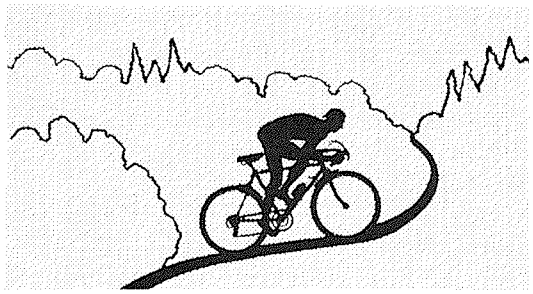
Only a few of the possible bit rates in the interval 50 bps to 50 Mbps are widely used, because the receiving rate must be adapted to the sending rate.

The following is a list of some standard terminal speeds measured in bits per second:

50	110	200	300	600
1200	2400	4800	9600	19200

Some examples of higher standard speeds, used for computer to computer communication, are: 48 Kbps, 64 Kbps and 2.048 Mbps.

Bandwidth

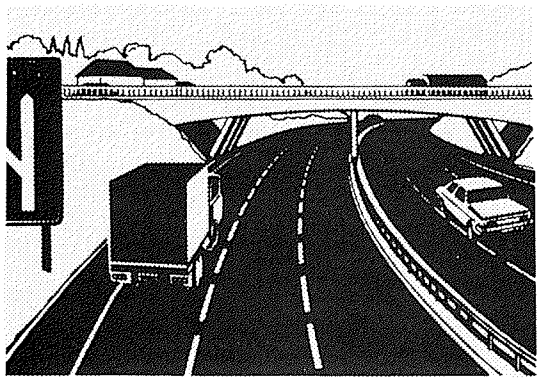


Small bandwidth

The bandwidth defines the capacity of the channel. Strictly speaking the bandwidth is defined as the difference between the highest and the lowest frequencies that can be used on the channel and therefore measured in Hz.

It may be useful at this point to compare a communication channel with a road. The wider the road, the higher capacity the road has for traffic. A super highway has a high bandwidth, whereas a bicycle lane has a small bandwidth.

With this last comparison in mind, it is not surprising that the higher the bandwidth, the higher is the possible speed on the channel.



High bandwidth

Let us take the analogy one step further. A road can be curvy and have obstacles like snow or stones on it. Under such conditions we have to drive more slowly. Similarly, a channel which is subject to noise makes the bit stream specially vulnerable at high speeds.

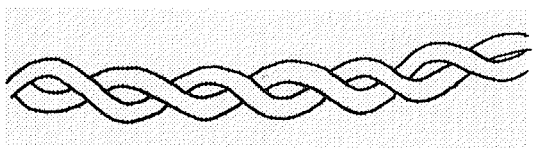
To conclude this section we want to state the fact that the possible speed of a given channel is a function of:

- The bandwidth
- The possible noise on the channel. This again is dependent on the electrical activity in the channel's environment and the properties of the channel itself.

TRANSMISSION MEDIA

In this section we will look at the properties of some specific types of channels.

GALVANIC CIRCUITS



Twisted-pair wire

A galvanic circuit is what we in chapter 1 called a simple cable. For electricity to "travel" through a cable, there must be at least one circuit inside the cable.

When there is one circuit in the cable, we call this a 2-wire circuit, because 2 wires are needed to make a full circuit. A 4-wire circuit consists of two 2-wire circuits.

Full vs. Half duplex

A 4-wire circuit always has full duplex possibilities. It is possible to transmit on one circuit and receive on the other. Similarly, on a 2-wire circuit half duplex operation is always possible. What is less obvious is that a 2-wire circuit may have full duplex possibilities.

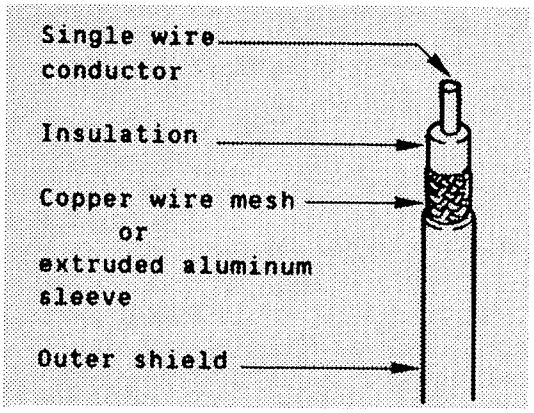
If the bandwidth of a 2-wire circuit is big enough, then it is possible to use one part of the bandwidth for transmitting and the other part for receiving at the same time. This affects the transmission speed, because the speed is a function of the bandwidth.

On the other hand we have a delay problem if we use half duplex, because one end-point has to make sure that the other end is finished transmitting before it can start sending.

Propagation time

The propagation of the signal also causes delay in half duplex operation. If we think about the bit stream as a sequence of signals on the circuit, it takes a certain time for the first signals in the sequence to travel all the way to the destination site. This is called the propagation time and is a property of the channel. After the propagation of the first signal, the rate of reception at the destination site is equal to the transmission speed.

COAXIAL CABLES



Coaxial cable

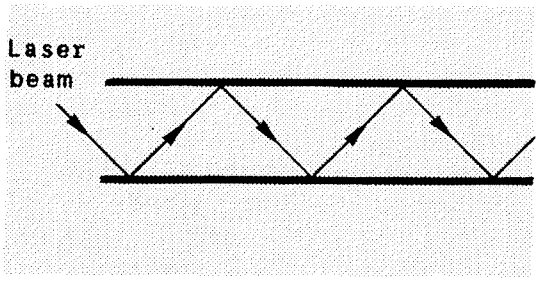
A coaxial cable (coax) consists of a central carrier wire surrounded by fine copper wire mesh (or a different material). PVC or some other material is used to position and insulate the carrier wire from the copper wire mesh. A strong outer shield insulates the whole cable from its environment.

Ethernet cables are examples of coax cables. Another example is cables, used for cable TV. The ways in which these cables are used are described in chapter 7 on Local Area Networks.

Advantages of coax cables compared to galvanic circuits:

- Good insulation (they are almost immune to noise)
- High bandwidth (in the Megahertz range)
- High speed transmission (in the Megabit per second range)

OPTICAL FIBERS



Optical fibre

Optical fibers are made of glass or plastic. Rather than using electrical signals for transmission, optical fibers use light pulses. A modulator translates the electrical signals coming from the computer or the terminal into light pulses.

Optical fibers are almost immune to noise because electro magnetic fields do not interfere with light pulses. Insulating a cable with optical fibers from external light sources is easy, because many materials are impenetrable by light.

Even though the fibers in a cable are extremely thin (something like hair), they have high bandwidths. The light pulse frequencies are high, and the pulse frequency can be varied over a wide range.

Summary of some advantages of optical fibers (compared to metal):

- Better security for the environment and not so easy to tap.
- Very high bandwidth, up to about three Gigaherz. One Gigahertz is 10^9 Hz. New technology may make even higher bandwidths possible.
- The attenuation of signals is minimal.
- Transmission rates may exceed one Gigabit per second (10^9 bps).
- Since fibers are small and light, they save space and weight.

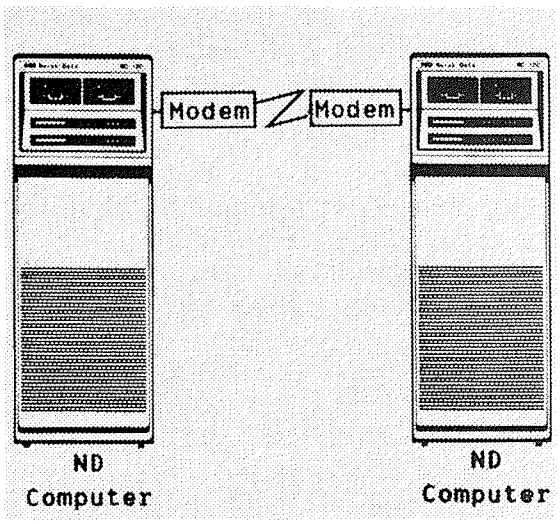
RADIO AND SATELLITE LINKS

Some advantages, compared to solid material, are:

- High bandwidth
- Their independence of solid material saves cabling costs.
- There is great flexibility in changing link locations.
- Satellites cover large portions of the globe.

One big disadvantage is the big propagation delay (see page 16).

MODEMS



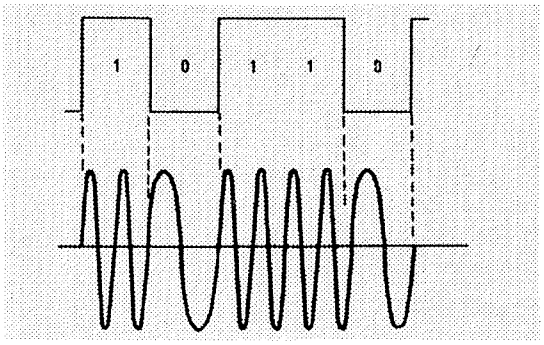
As you remember from page 13 a telephone line may require transmission by an analog wave. Since the bits by nature are digital, they must somehow be converted from digital signals to analog wave type signals. This conversion is called modulation, and is performed by a modem. The acronym MODEM stands for MOdulator DEModulator. The receiver of the message must convert from analog back to digital, and this process is called demodulation.

When the modems are switched on in each end-point, a carrier is established on the line. The carrier is a basic sine wave and does not in this case carry any information in itself. The modulator on the sender side changes the carrier wave when a bit is detected. The type of change done to the carrier depends on the specific modulation technique. A one bit is modulated differently than a zero bit.

MODULATION TECHNIQUES

Base band is the frequency range in which a signal is generated. Base band coding is a term used for the encoding of signals, prior to transmission in a channel, without transforming them by modulation.

FREQUENCY MODULATION (F.M.)



The carrier wave has a constant frequency, because none of its characteristics vary with time. A modulator may change the frequency of the carrier and in that way generate analog signals which correspond to the bits. This is called frequency modulation (f.m.).

Full Duplex Communication

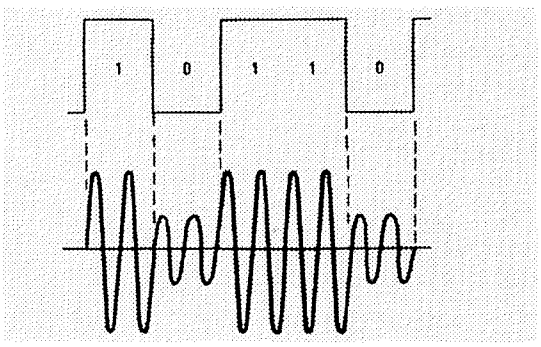
An asynchronous terminal transmits single characters intermittently. In this example it is connected to a computer by a telephone line. A telephone line is always a 2-wire circuit. The terminal operator needs full duplex operation because s/he may want to send a break signal to the computer while s/he receives output.

The f.m. modem divides the bandwidth of the telephone line into two channels. With a transmission speed of 300 bps this is no problem. The specific division of the bandwidth and the encoding of the bits may be as follows:

- Channel 1 uses the area around 1180 Hz as bit zero and 980 Hz as bit one.
- Channel 2 uses the area around 1850 Hz as bit zero and 1650 Hz as bit one.

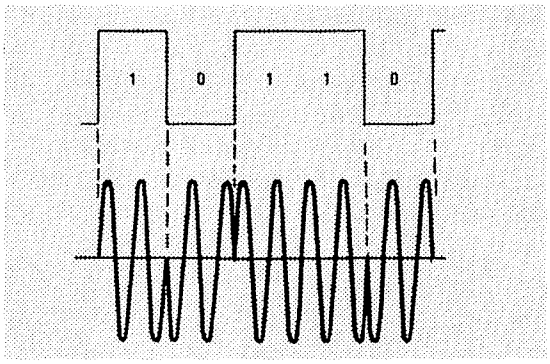
If the terminal uses channel 1 for sending, then the computer uses that channel for receiving. The terminal must use channel 2 for receiving and the computer must use that channel for sending.

OTHER MODULATION TECHNIQUES



Besides frequency modulation there is amplitude modulation (a.m.), phase change modulation and combinations of these.

Amplitude modulation is based on changing the amplitude of the carrier. Bit zero has one amplitude, which is different from the one of the carrier, and bit one is encoded as another amplitude.



Phase change modulation is somewhat more complicated. You may get an idea about it if we say that at a 180 degree phase change of the carrier, the encoded bit stream changes from the previously encoded bit. The level of sophistication increases if we permit more than one angle of phase change. In that case we can encode several bits into the same signal (in which case baud is different from bps).

TELECOMMUNICATION SERVICES

In Europe there are mostly governmentally owned telecommunication companies, typically one agency per country. Quite often the postal services are under the same administration as the telecommunications. We therefore refer to the agency as PTT (Post Telephone and Telegraph).

PTT and PDN

Usually the PTTs offer data transport through the telephone lines. An increasing number of the PTTs offer advanced data communication facilities, the so called public data networks. One type of Public Data Network (PDN) is called circuit switched, and another type is called packet switched.

Data Terminal Equipment

A computer or a terminal can be a subscriber in a public data network. We call this (computer or terminal) a DTE, which stands for Data Terminal Equipment. The reason why a computer may be called a DTE is that the DTE is a termination point for the communication. The term DTE is also used for a computer or a terminal in one end of a telephone line.

Data Communication Equipment

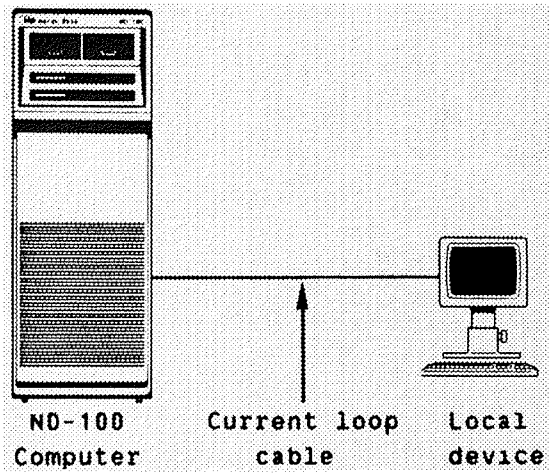
As you remember from page 18 we need modems to convert digital data to analog and back to digital when we communicate via a telephone line. The modem is an example of a DCE, which stands for Data Communication Equipment or Data Circuit terminating Equipment. When we communicate via public data networks, different types of DCEs are used. These will be described later.

INTERFACES FOR TRANSMISSION

In this section we will introduce 3 types of interfaces for transmission of bits:

- Current loop
- V.24
- X.21

CURRENT LOOP

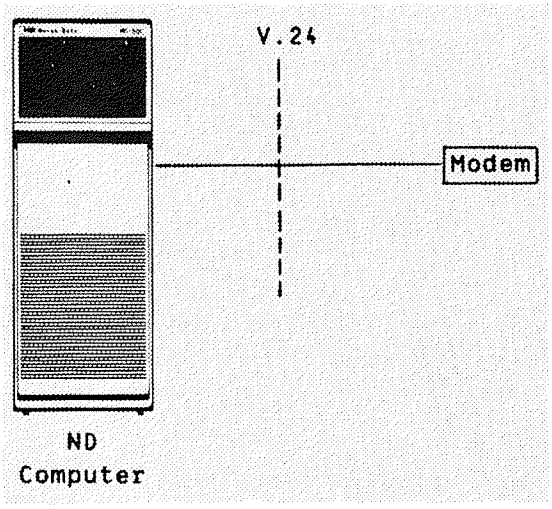


The current loop connection is typically used for local connection of an unintelligent device. This device is typically a terminal, but it may be a printer or some different device.

Both the ND-100 and the local device must have a current loop interface card.

The current loop connection (i.e., the cards plus the cable) consists simply of 2 galvanic circuits. The transmission is digital.

THE V.24 INTERFACE



The V.24 recommendation is an international standard which describes the functional characteristics of the interface between DTE and DCE.

V.24 deals with transmission across analog carriers (like telephone lines). The DCE is therefore in this case a modem.

V.24 contains conventions for:

- Establishing contact between DTE and transmission line (via modem)
- Sending and receiving data as a serial bit stream
- Disconnecting the transmission

Closely related to V.24 is the V.28 recommendation which describes electrical characteristics of the interface, for example the voltage levels.

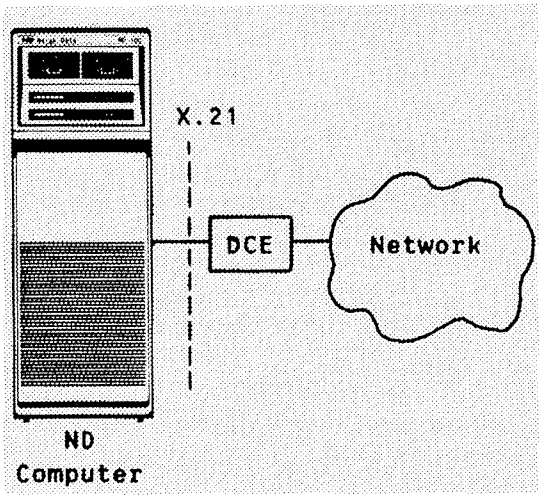
RS232

In North America there is a standard called RS232C. This standard is roughly equivalent to V.24 + V.28. On some devices used in Europe the RS232C standard is specified, which probably means that the interface is compatible with V.24.

Printers may use the V.24 (RS232) interface. This interface is also sometimes used on local connections, where there is no modem.

Various implementations of V.24 (or RS232C) may be different because different subsets of the signals described in the standard are used. This means that two devices following the V.24 standard may not be exactly compatible.

THE X.21 INTERFACE



Like V.24, X.21 also describes conventions for establishing and disconnecting the communication between DTE and DCE. X.21, however, relates to a DCE of a circuit switched public data network. Such a network is based on digital transmission and is designed for transport of data rather than transport of voice.

A circuit switched public data network functions very much like a telephone network from the computer's point of view. DTE A has to "dial" DTE B, using the X.21 conventions.

X.21 bis

If the DTE does not have an interface card obeying the X.21 standard but only a V.24 card, it may still connect to a circuit switched public data network. In this case we need a special DCE which can convert from V.24 signals to X.21 signals. This interface between DTE and DCE is called X.21 bis.

Asychronous Terminals

The data network operates synchronously, which implies that the DTE has to transmit blocks of data in a synchronous way. In case the DTE is an asynchronous terminal sending characters intermittently, some different interface conventions have to be used as well as a different type of DCE. The recommendation is called X.20.

Electrical Characteristics

To define the electrical characteristics of the X.21 signals, X.27 is being used. This X.27 recommendation is the same as the American RS422.

CHAPTER 3
NETWORKS

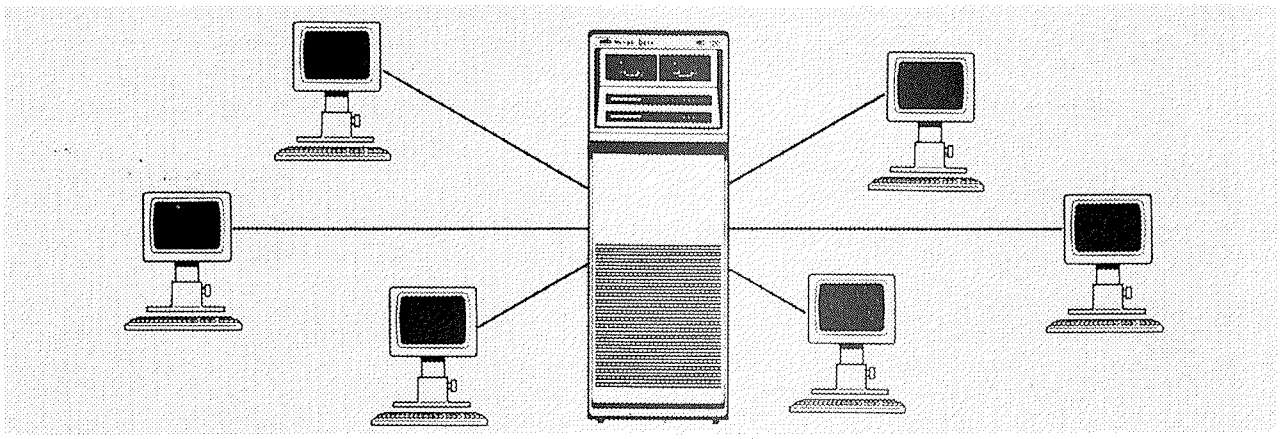
SIMPLE TERMINAL NETWORKS

An unintelligent terminal may be defined as a character oriented device, and each character we key in is sent to the computer and then returned. This return of the character is called echo. The characters are sent intermittently. In other words we have an asynchronous terminal.

We will now look at the network organization used for asynchronous terminals. There are two basic network arrangements used for this group of terminals.

- Star network
- Remote multiplexer network

The star network solution provides each terminal with its own physical connection to the computer.

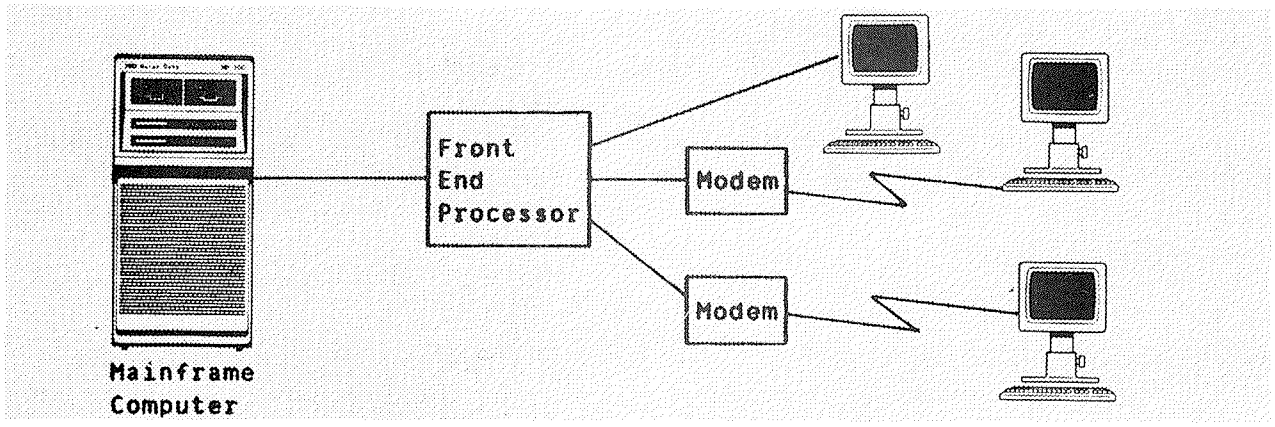


Star network

The star network has many point-to-point circuits between the terminals and the computer. This kind of topology is often used in small networks and for local distances. The main advantages with this configuration is high reliability because only one terminal is affected if a line goes down, and the system is fast because we do not have to wait for free lines.

Front End

To save costs on the computer site, we may employ a Front End Processor (FEP) to take care of all the incoming connections. In this manner the main computer can delegate the communication problems to the FEP. An FEP is often a minicomputer (which is programmable). Small ND computers have been programmed to function as FEPs.

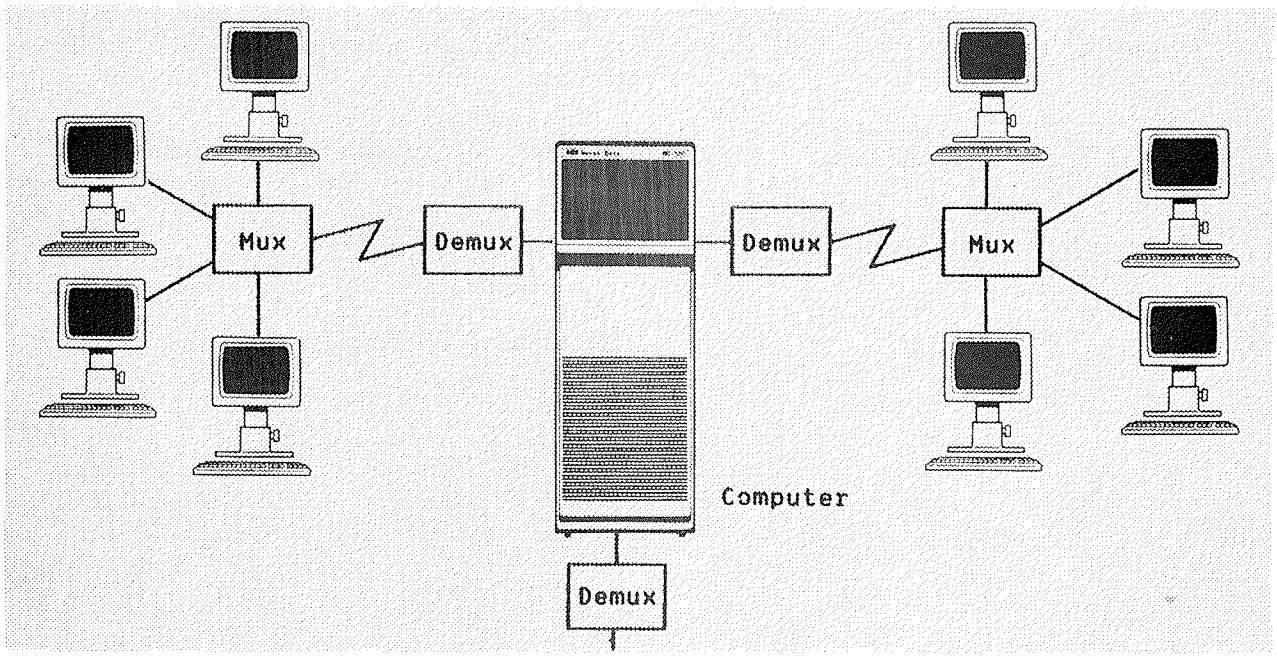


The Front End concept

REMOTE MULTIPLEXER NETWORK

When dealing with Wide Area Networks, the star network topology has a serious drawback because of high costs due to low utlization of lines and communication equipment.

A way to mend that drawback is to employ remote multiplexers, thereby saving line costs.



Remote multiplexer network

In the picture you can see that we need only 2 expensive telephone lines for 8 terminals, provided they are grouped together locally. The connection from a terminal to a multiplexer is short in distance. It may for example be a current loop circuit.

MULTIPLEXING

Multiplexing can be thought of as merging of data streams. If you think about a road network with main roads, secondary roads and junctions, this can be used as an analogy. Cars from different side roads merge with cars on the main road (multiplexing). From the main road cars exit to side roads (demultiplexing).

In data communication characters may be multiplexed in different ways. We will talk about 3 different methods:

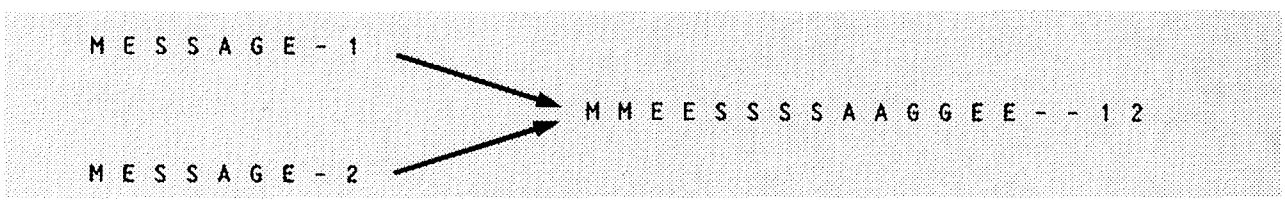
- Time Division Multiplexing (TDM)
- Frequency Division Multiplexing (FDM)
- Statistical Multiplexing

TIME DIVISION MULTIPLEXING (TDM)

As an example let us use three terminals sharing one line. Terminal A sends one character, then terminal B sends one, then terminal C. When C is finished transmitting one character, terminal A gets the next chance.

The general principle is division of the available time into time slots. The terminals are scanned in a round-robin fashion. Every time a terminal has a character to send, this character occupies the current time slot. If the terminal has nothing to send at this time, the time slot allocated to it remains empty.

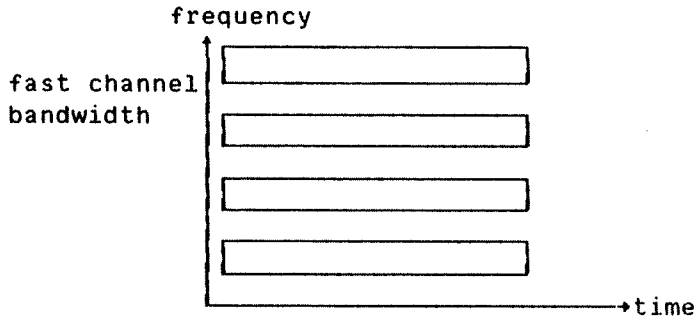
This is a relatively simple technique to implement, and it has an obvious disadvantage. Since the terminal transmits intermittently, there will be many empty slots. If one terminal is inactive, the time slots allocated to the terminal cannot be used by other active terminals.



Example of TDM

FREQUENCY DIVISION MULTIPLEXING

Frequency Division Multiplexing (FDM) is based on dividing the available bandwidth of the line between the channels coming in from the terminals to the multiplexer. This arrangement can be flexible, because slow channels may be allocated a narrower bandwidth than the faster channels. FDM is the most common multiplexing type inside a telecommunication network.



Frequency Division Multiplexing

STATISTICAL MULTIPLEXING

Statistical multiplexing resembles TDM in the sense that the time axis is divided into fixed slots. However, the slots are not allocated on a round-robin basis, but on a first come first serve regime. As soon as the character arrives from a channel it is placed in the next free slot.

If the capacities of the incoming channels (to the MUX) together exceed the capacity of the common line, a problem may arise. If the terminals transmit continuously, congestion will occur. If, on the other hand, we assume that not every terminal sends continuously, we may have solved the problem. On a statistical basis, the traffic pattern evens out.

Concentrator

Statistical multiplexing is sometimes called intelligent TDM. Another term, which is frequently used and means the same, is concentrator.

An example of a concentrator is an ND computer with a special program for multiplexing the characters, from character oriented terminals, before they are sent to a different computer through a telephone line. Norsk Data supplies such programs, for example for Honeywell mainframes.

OTHER TYPES OF TRADITIONAL NETWORKS

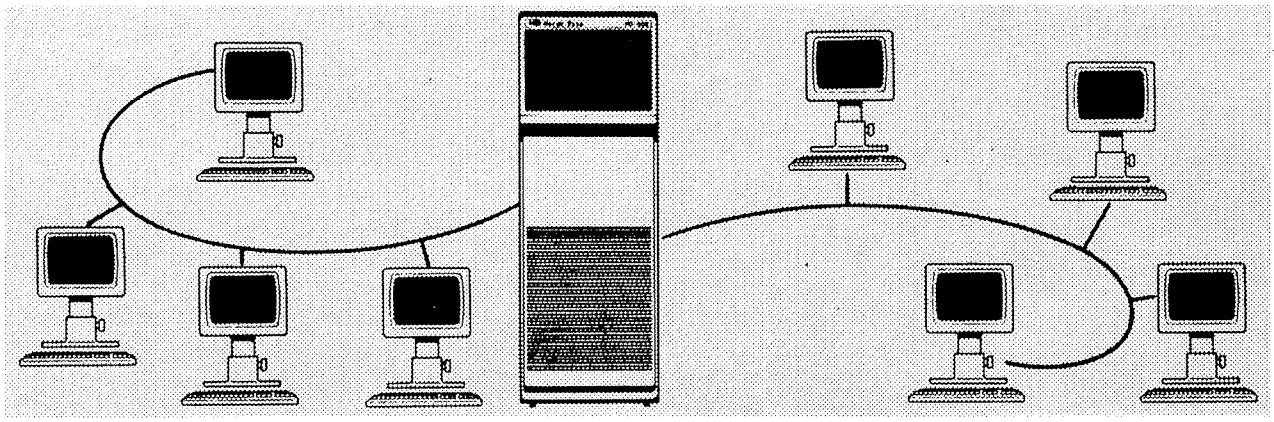
So far we have presented character based networks which are based on unintelligent terminals and centralized computer systems. In the rest of this chapter we will talk about networks with more intelligent terminals, based on synchronous transmission, and networks with several computers.

The star topology can also be used for intelligent terminals, but since these terminals are capable of more complex communication than the asynchronous, more sophisticated techniques may be used.

MULTIPOINT LINES (MULTIDROP)

Multipoint is synonymous with multidrop (at least for our practical purposes). We establish a multipoint line by connecting several terminals to the same circuit.

The multipoint line network aims for optimum use of the transmission medium by having several terminals share the same medium. As only one message can be transferred at a time, strict dicipline is needed to avoid interference. The use of the line is therefore under the control of the computer, and is in the form of a polling regime.



Two multipoint lines connected to a computer

Poll and Select

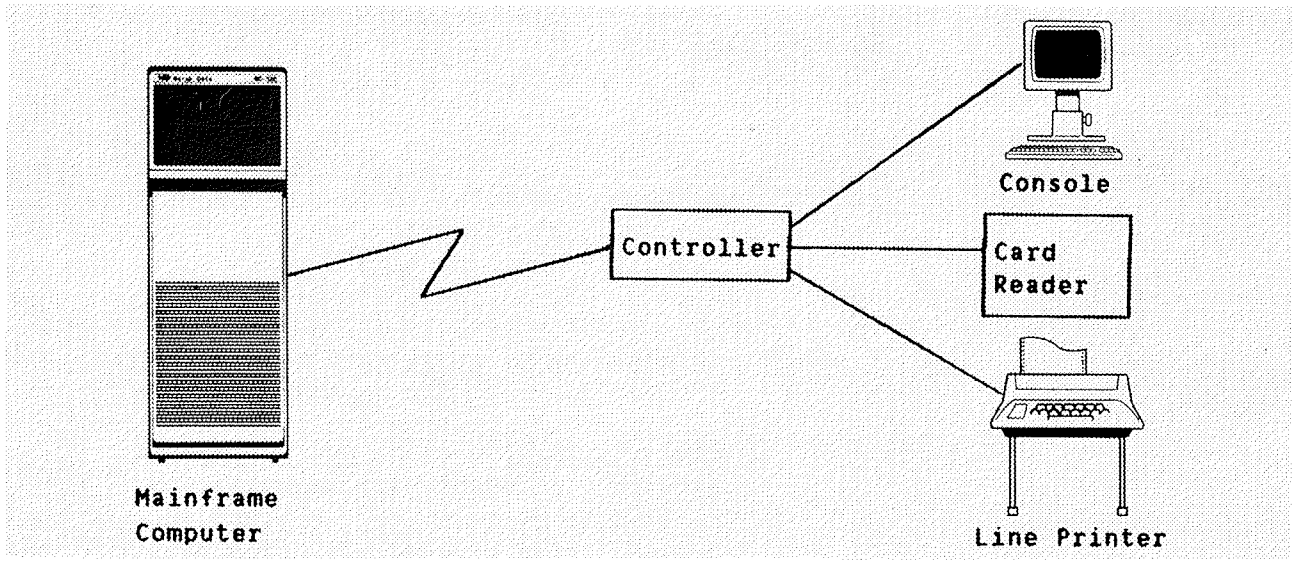
Each terminal sends messages to the computer and receives messages back. One message consists of a block of characters. Before a terminal can send a message to the computer, it has to get an o.k. signal. The poll message from the computer is such a signal. This message reaches all the terminals on the same multipoint line, and terminal A knows it is polled because the address of A is attached to the poll message. The polled terminal sends its already prepared block and gets a positive reply from the computer if the block was received without error. If the terminal has nothing to send, the terminal answers the poll with a control message. Then the computer polls the next terminal. The scheme is repeated in a round-robin fashion.

Sending of a message from the computer to a terminal is done by an addressing method called selecting. We have two types of select: "fast select" where the address is sent in the same block as the message and "normal select" where the terminal first is asked if it is ready to receive a message.

REMOTE JOB ENTRY

Ten to fifteen years ago, computer users often did not have direct access to the computer resources that they needed. If they had card punching equipment themselves, they punched a deck of cards which was transported by car or by the postal service to a remote location. The card deck was read by the computer, together with other jobs, and stored in a queue. When the computer was ready, the job was processed and the results printed (batch processing). The results were sent back by conventional means of transport.

With the development of data communication, the cards could be read on the local site and transferred through a telephone line by a controller to the computer. When the computations were done, the results were transferred back through the telephone line. This is called Remote Job Entry (RJE) or remote batch. The controller (which was pure hardware, made by the same manufacturer as the computer) together with the peripheral equipment is called a Remote Job Entry Terminal.



RJE terminal communicating with a mainframe

EMULATOR PROGRAMS

When the minicomputer was introduced, the behavior of the RJE terminal could be emulated by software. For example the old NORD-20, with only 4K words of memory, was used for this purpose. The mainframe in question was called a host computer.

Todays ND computers can function as RJE terminals and do other functions simultaneously. The software necessary to handle the RJE function is called an emulator program. Such a program makes the computer behave as if it were a terminal manufactured by the vendor selling the host. We will talk more about the type of behavior we have in mind in the next chapter (on protocols).

Card readers are not much used any more, so the card images may be keyed in at a terminal and transferred to disk. The results from the batch run may be transferred to a disk file on the RJE terminal rather than directly to a line printer.

On ND computers a whole range of emulator programs are available: RJE emulators for IBM, Honeywell, CDC, Univac and other mainframes. There is also another type of emulator programs available (for the same hosts) which emulate the multidrop situation. Terminals connected to an ND system are then perceived by the host as mainframe specific terminals connected to a multidrop line.

Access to Several Resources

With emulator programs we have possibilities to access all the functions available in the mainframe system and to use its resources.

COMPUTER NETWORKS

The advantages of organizing computers into a network is similar to the ones mentioned in the previous paragraph. With a network of ND computers, like we have in COSMOS, we can use special facilities installed on the individual computers in the network. COSMOS is the name of Norsk Data's networking system.

There are many ways of organizing the network. In COSMOS, for example, the individual systems are equal partners in the communication process. In other networks one particular system may function as the boss, and each system must have the permission from the boss to send a message to somebody (cfr. poll and select, page 30). A network with a boss is vulnerable, because if the boss goes down, nobody can communicate.

Note

When talking about a computer in a network, we may in the following say **computer**, **system** or **node**.

NETWORK TOPOLOGIES

One aspect of the organization of networks is the network topology. Here are some topologies listed. All these topologies are

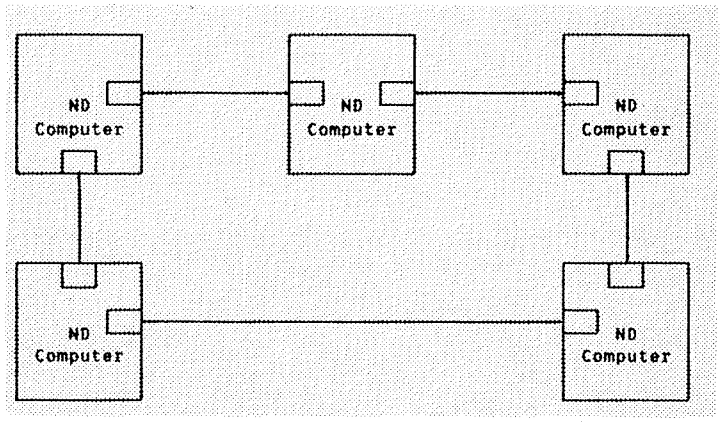
based on point-to-point connections. Note that the individual systems, in any case, may or may not be equal partners.

- Ring
- Star
- Mesh
- Series

Within a COSMOS network not based on the Ethernet option, we need one interface card in each end of a cable to create a channel. We will now introduce the topologies mentioned by making some comparisons.

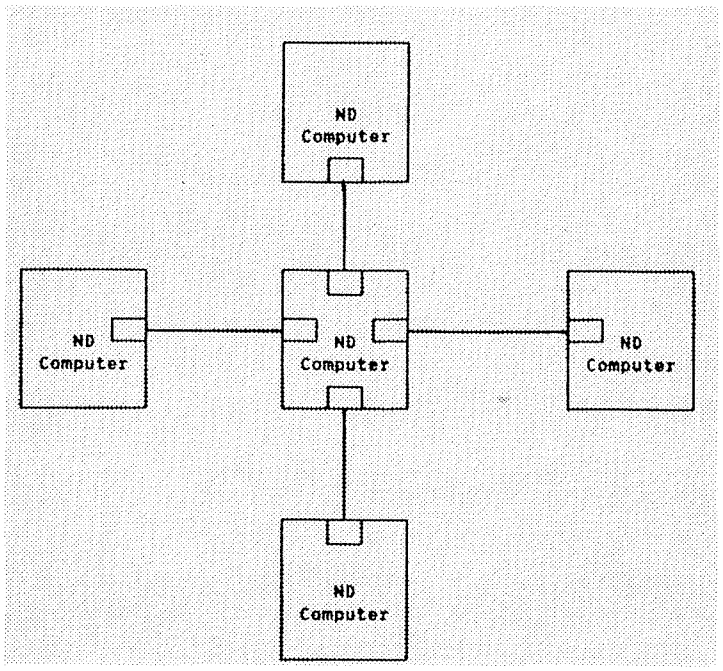
Ring

In this ring configuration, ten interfaces and five cables are required



Star

The star configuration requires less hardware than the ring solution. Eight interfaces and four cables are needed.



Here are some advantages and disadvantages of the star solution compared to the ring solution.

Advantages:

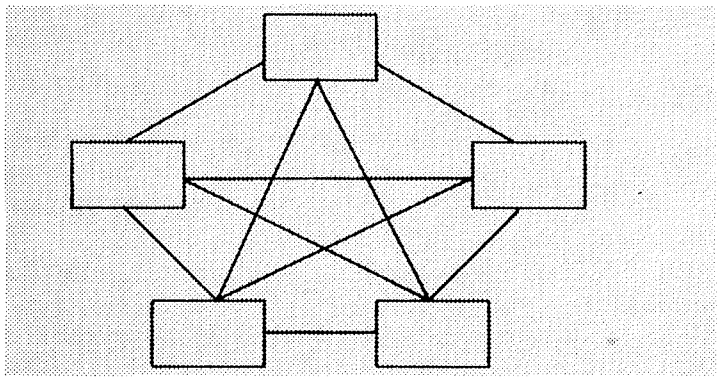
- Going from one node to another, at most one node is a relay node. A relay node is a node you have to go through to reach the destination system.
- A system supervisor on the central node reaches all systems without bothering any relay node.
- A heavy load on a peripheral node does not affect other nodes.
- Malfunctions in peripheral nodes or their communication cables do not affect the other nodes.

Disadvantages:

- If the central node stops, all communication breaks down.
- The central node may become quite busy relaying messages. With a big enough network configuration the central node may not have time for anything else.

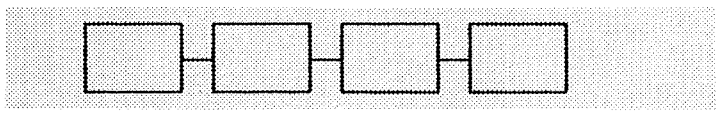
The ring topology may be improved so that any pair of nodes can communicate without going through a relay node. This is called a mesh topology.

Mesh



Series

The series topology may be perceived as a broken ring.



CHAPTER 4
PROTOCOLS

WHAT IS A PROTOCOL?

On page 31 we gave an example of an emulator program which made the computer behave as if it were a terminal manufactured by the vendor selling the host. The set of formal conventions mediating this behavior is called a protocol.

Is V.24 a protocol?

Since a protocol is a set of conventions for exchange of data between two communicating parties, people sometimes call V.24 a protocol. While it is true that parts of V.24 comprise such a set of conventions, there are also other aspects of V.24. The same can be said about some other interface recommendations like X.21 and X.25.

A primitive protocol

If we think of an asynchronous (unintelligent) terminal connected to a computer, the communication software in the computer will usually assume that there is an intelligent human being operating the terminal. This means that the software can omit some of the checking.

Let us say there is an error in the transmission of the @LIST-FILES command. Instead of performing thorough error checking, SINTRAN could just interpret the incoming command as nonsense and give the response NO SUCH FILE. The operator probably has enough fantasy to try again. A typical protocol, on the other hand, would detect a transmission error and ask the terminal for a retransmission of the message.

A typical protocol

A typical protocol may follow this general pattern:

1. A connection is established. This means that the sender makes sure the receiver is ready for a dialog. Sometimes this is called handshake.

2. During the transfer of data, both parties do thorough checking to make sure that transmission errors are not accepted and that no data is lost.

3. The connection is terminated. This means that the receiver is told there is no more data. The dialog is terminated.

Link protocols

The task of a link protocol is to make sure that the bits are securely transmitted through the channel (here called link) between two computers. Some of the main functions performed by a link protocol are:

- Assembling the data to be transmitted into a data block with transmission beginning and ending markers. This is called framing and the transmission block is called a frame.
- Achieving data transparency. This allows a link to treat any bit pattern, including normally restricted control characters, just as a pure data stream.
- Controlling the flow of data across the link. It is essential not to transmit bits faster than they can be received at the other end. Otherwise the receiver overflows and data is overrun, or all buffering capacity is used up.
- Controlling errors. This involves detection of errors using some kind of redundancy check. It also involves acknowledgement of correctly received messages and requests for retransmission of faulty messages.

We shall now present two classes of link protocols. The two protocol classes are:

- Binary Synchronous (BISYNC) protocols
- High level Data Link Control (HDLC) protocols

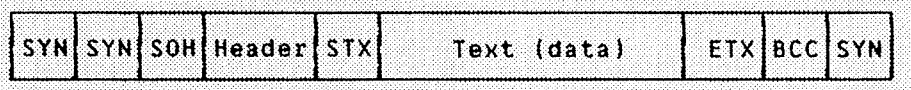
BINARY SYNCHRONOUS PROTOCOLS

Some data communication experts have predicted that these protocols soon will be obsolete. Whether this is true or not, the fact is that BISYNC is still the type of link protocols which is most used. We will briefly describe some important features.

As the name indicates, BISYNC is made for synchronous transmission. Besides it can handle half duplex but not full duplex transmission.

BISYNC is based on character control. This means that some of the characters in a BISYNC frame are defined as communication control characters. Some of the control characters are used as transmission block beginning and end markers, while others are used in the information field for control purposes. If a data bit pattern coincides with the value of a control character, the receiver must avoid interpreting this bit sequence as a control character. This is called data transparency. Usually a lot of software is introduced to handle the numerous control characters.

This is an illustration of the BISYNC frame format:



←——transfer direction

The syn, stx, etx in the BISYNC message format shown in the figure are control characters. These are used to delimit the various parts of the message.

- SYN is the synchronization character, refer to page 11.
- SOH stands for Start Of Header. This means that a header is included in the message. All the following characters, up to the next control character, comprise the header. The format of the header may vary.
- STX stands for Start of TeXt. This character means that the following characters comprise the message transported across the link. The data field (message field) must be an exact number of characters.
- ETX stands for End of TeXt. This marks the end of a message. If there is too much data in a message to fit into a single transmission block, this character means that this is the last transmission block in the message transfer. If more data follows in another block, the character ETB (End of Text Block) is used.

- BCC means Block Character Check. This is a bit sequence calculated as the longitudinal parities of all the characters between STX and ETX. The receiver performs the same calculation and compares the received BCC with its own calculated value. If the value of the received BCC is different from the calculated value, an error is detected. A retransmission of the frame is then requested.

Longitudinal parity

This is an example of generating a BCC based on even parity:

```
0 1 1 0 1 0 0 1  Char. 1
1 1 1 0 0 1 1 1  Char. 2
0 0 1 1 0 0 0 0  Char. 3
1 0 0 1 1 0 0 1  Char. 4
-----------------------
0 0 1 0 0 1 1 1  BCC
```

We have simplified the situation, so the frame has 4 arbitrary characters. Each row of bits represents one character. Each 4-bit column has either an odd or an even number of one bits. The bits of the BCC are calculated so that each 5-bit column has an even number of one bits.

Protocol information

From the format shown in the illustration of a BISYNC frame, we can see that the text field is only a part of the whole block of characters transferred. The header and BCC contain the information used by the protocol to transfer the data. The extra characters used by the protocol are redundant to the information being transferred. This redundancy reduces the efficiency of the data channel. All protocols introduce redundant information, the more complex the protocol the more redundant information is added.

Establishing a connection in BISYNC

As we saw on page 37, a protocol usually specifies that the communicating parties shake hands before they start talking. In BISYNC this is done by first sending a frame consisting of only three characters: SYN,SYN,ENQ. ENQ stands for ENQuiry which means that the sender wants to start a dialog. If the receiver is willing to have a dialog, it answers with an ACKO frame consisting of the four characters: SYN,SYN,DLE,0. DLE stands for Data Link Escape and has different functions. This is the first step in establishing a connection.

In the next step the enquirer has to identify himself and the receiver has to respond with an acknowledgement if she is still willing to communicate.

What is BISYNC used for?

BISYNC is a class of protocols and different varieties exist both for RJE and poll and select (on a multidrop). Norsk Data has RJE emulator programs which follow the IBM 2780 and 3780 (both are BISYNC protocols). Other computer manufacturers than IBM have made their special versions of BISYNC, but calling the protocols by different names. An example is Honeywell GRTS-115. The IBM 3270 is probably the best known for poll and select. ND has emulators for the last two mentioned as well as for many others.

Disadvantages with BISYNC

We will list some of the main disadvantages with BISYNC:

- Since full duplex transmission is not possible, utilization of links with long propagation delays is especially poor. An example is a satellite link.
- The solution to the data transparency problem is awkward. If a control character is detected inside the STX,ETX bracket, the bracket is enclosed between two DLEs (Data Link Escape). Still it is possible that a DLE is detected within the bracket, and this problem is solved by prefixing the DLE itself with another DLE.
- The error check based on the BCC character is not as good as it could be. Some bits could be altered during transmission and the message still be accepted.

HDLC PROTOCOLS

HDLC stands for High level Data Link Control. This is a class of protocols, invented later than BISYNC. HDLC is bit oriented rather than character oriented. HDLC is likely to become the most widely used link protocols for distributed processing and computer networks.

Objectives

Some main objectives of the various HDLC protocols are:

- The data field can be any bit pattern and any number of bits.
- The error detection algorithm is reliable.
- The protocol is able to operate efficiently on links with long propagation delays.
- The same protocol is able to handle point-to-point and multidrop links.
- The same protocol can handle both half and full duplex transmission.

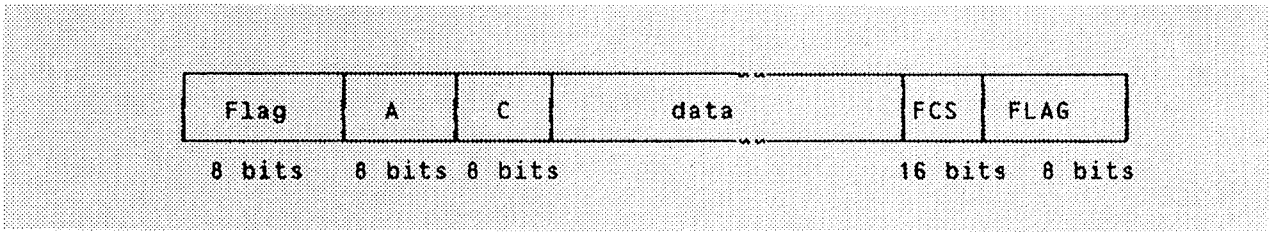
Varieties of HDLC

The following are different protocols within the HDLC class:

- Link Access Protocol (LAP)
- Link Access Protocol Balanced (LAPB). Both LAP and LAPB are CCITT recommendations.
- Synchronous Data Link Control (SDLC) is IBM's version of HDLC.

HDLC frame format

This is an illustration of the HDLC frame format:



The control fields described below are divided into 8-bit units called octets:

- The flag octet marks the beginning and end of the frame. It is also used for synchronization and to achieve data transparency.
- The A-octet is used, among other things, for addressing on multidrop lines. There may for example be a series of work stations on the computer in one end of a link. These work stations may reside on a multidrop line. If the computer in the

other end of the link sends to a particular work station, the address of that station must be stored in the A-octet.

- The C-octet is called the control octet. It has some different functions which will soon be explained.

- FCS stands for Frame Check Sequence. In HDLC this is a redundant bit pattern based on polynomial coefficients and is called Cyclic Redundancy Check (CRC). Error detection based on this method is more effective than parity check. We will not explain the mathematical details behind the algorithm.

There are 3 types of HDLC frames:

- Information frames (I-frames)
- Supervisory frames (S-frames)
- Unnumbered frames (U-frames)

The flag octet

The use of the flag as beginning and ending marker of the frame presents a transparency problem. The flag pattern must not occur anywhere within the data in the frame. This means that the part of the frame between the flags must be transparent to the mechanism searching for flags on a received frame. To ensure that the flag pattern does not appear within the frame the HDLC protocols use a technique called bit stuffing.

Bit stuffing

The sender inspects the frame before transmission. When a sequence of five ones occurs, a zero is inserted as illustrated below:

```
..01111111... becomes ....011111101...
..01111110.... becomes ....01111100...
                              ↑
                           Inserted
```

The zero is inserted regardless of the value of the sixth bit. On ND computers this is done by the hardware. It only increases the length of the frame by one bit for every sequence of five one bits. As the receiving hardware scans the incoming frame, a zero

following a sequence of five ones is removed. If the sequence of five ones is followed by another one, then the flag may have been found. The seventh bit is then inspected and if it is zero, then we have the flag (01111110). If the seventh bit is another one bit (01111111), then there is an error, in which case the frame is aborted.

Flow control

It is important not to transmit bits faster than they can be received at the other end. Otherwise the receiver overflows and data is overrun, or all buffering capacity is exhausted. Several schemes for flow control are available. The HDLC protocols use the N(R) and N(S) fields in the C-octet for flow control. Further details are explained in the next section.

HDLC under COSMOS

On a COSMOS network the software part of the transport medium is called XMSG. XMSG uses its own version of LAPB as the link protocol.

FURTHER DETAILS ON HDLC

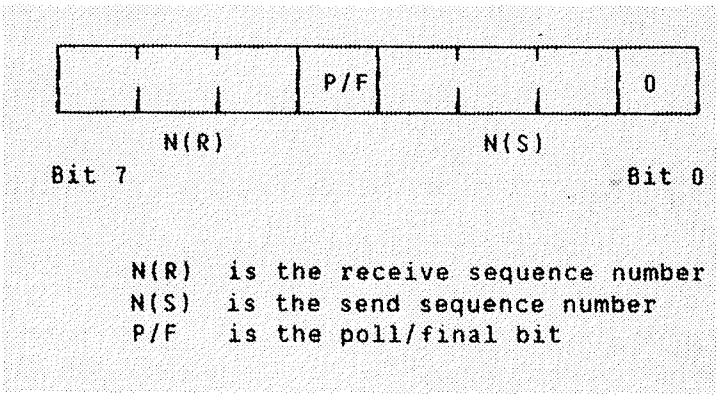
This section explains the HDLC protocols in more detail. If it is too detailed for your purpose, you may skip it.

The C-octet

This eight-bit pattern determines the type of frame. It is also used for sequence numbering, acknowledgement, and flow control. The command field has a different meaning for the three types of frame (information, supervisory and unnumbered). A zero in bit 0 means that this is an information frame.

I-frames

The information frame is the only type that carries data. The C-octet of the I-frame looks like this:



The sequence numbers are used for error correction and flow control. Notice that the numbering is modulo 8, i.e., the numbers go from 0 to 7 and then back to 0. N(S) is the sequence number of the frame sent. N(R) is the sequence number of the next frame which is expected from the other party.

Suppose that A sends 3 frames to B with sequence numbers 0 through 2. If B receives these free of error, B then sends a frame to A with N(R) equal 3. In this way B tells A that it wants frame number 3 next time. This also means that B received frames including frame 2 free of error. B has in other words acknowledged all frames from A including frame 2.

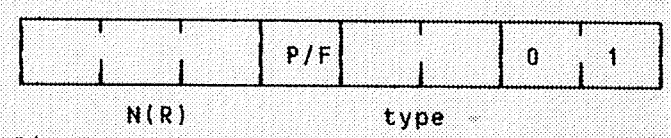
This type of acknowledgement is called piggybacked acknowledgement, because the acknowledgement is put "on the back of" an I-frame. The alternative to piggybacking is using S-frames for acknowledgements.

Suppose that A sends 3 frames to B with sequence numbers 0 through 2. If B finds an error in frame number 1, B will send a frame to A with N(R) equal 1.

S-frames

When bit 0 is one and bit 1 is zero in the C-octet, this means that the frame is a supervisory frame. Bits 2 and 3 tell which type of supervisory frame. Supervisory frames are not given sequence numbers as they do not contain any data. They may be used to acknowledge information frames therefore bits 5, 6 and 7 contain the N(R) for acknowledgement. In general the S-frames are used for control purposes.

The C-octet of an S-frame looks like this:



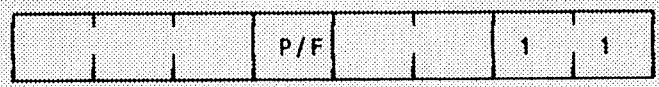
Possible types are:

00 RR - Receive Ready
01 REJ - Reject
10 RNR - Receive Not Ready
11 SREJ - Selective Reject

The RR frame is used as a positive acknowledgement, i.e., the received frames are free of errors. RNR tells the sender to stop sending since the receiver is short of buffers. REJ tells the sender that a frame is missing and instructs the sender to retransmit from frame number N(R) onwards. SREJ is sometimes used instead of REJ, for instance on a satellite link. Under certain circumstances it is very inefficient to retransmit all the frames following the erroneous one. It may also be worth mentioning that the C-octet can be extended to 16 bits (in which case it is no longer an octet). This means that the sequence numbers can go up to modulo 128. For satellite links, which have long propagation times, sending many frames before acknowledgement gives a good utilization of the link.

U-frames

Unnumbered frames are control frames without sequence number. They are used for instance to establish connections and to terminate dialogs. This is the format of the C-octet:



C/R C/R

Bit 7 Bit 0

C/R stands for Command/Response. These fields define the types of U-frames, for example:

SABM	-	Set Asynchronous Balanced Mode
UA	-	Unnumbered Acknowledgement
DM	-	Disconnect Mode
DISC	-	Disconnect

SABM is used only with LAPB. The party that takes the initiative to communicate sends SABM to initiate a handshake. The other party answers with UA if he agrees to talk. Otherwise he rejects the request by responding with DM. DISC is used for terminating a dialog which has actually taken place.

The window mechanism

As we have seen, the N(R) and N(S) fields are used for flow control purposes. The sender and receiver maintain a "window" of frame numbers with a certain width that slides across the sequence number range. The window has a lower edge: the last acknowledged frame. The upper edge of the window is the

highest frame number that the sender may send at this time. In other words, the window defines the sequence numbers of possibly outstanding messages. That they are "outstanding" means that they are sent, but not yet acknowledged. When the sender receives a positive acknowledgement, the window is updated (it is slid forward). The window size must be agreed upon by both parties. Seven is one standard size.

HIGHER LEVEL PROTOCOLS

Referring to the I-frame in HDLC, there is a field of data. Ordinarily a part of this data is really control information. In a network there is for example a need for address fields for the receiver of the message as well as for the sender of the message. Please do not confuse this with the address field in HDLC (the A-octet) which is used for distinguishing between stations on the same link (that is if we have multidrop). A network consists of many links, and the routing problems within the network are problems at a different level than the secure transmission of bits across one single link.

Link and network

Bit stuffing, handshake between two endpoints of a link, detection of errors due to transmission across a link etc., are all problems belonging to the link level. The HDLC protocols are link level protocols. Routing problems, on the other hand, are problems belonging to the network level. Network level protocols are designed to take care of such problems.

Meaningful data

Problems at still higher levels may occur. Consider for instance the communication between two computers in a network where one file is transferred from one computer to the other. Suppose that the file and record formats are different on the two. How does the receiving computer know in what way to interpret and store the incoming data?

In the next chapter we will introduce a general model, called the OSI model, for data communication system architecture. This architecture is based on the kind of levels (also called layers) indicated in this last section.

CHAPTER 5
DATA COMMUNICATION SYSTEM ARCHITECTURE

LAYERED ARCHITECTURE

As we saw in the previous chapter, page 47, different levels of protocols exist. In its early days, a data communication system used only one protocol and thereby lumped several functions together. One disadvantage to that was that every time a new hardware device was implemented, we had to make changes in the protocol software.

Advantages

The advantages of a layered approach are many. Here are some of them:

- A complex problem can be broken down into several smaller problems.
- By assigning functions to layers, we obtain a clear structure.
- Changes at one level do not necessarily affect other levels.
- With a good architecture we can solve complicated problems. For instance it is possible to build in translation mechanisms so that incompatible products can communicate.

THE OSI REFERENCE MODEL

Several years ago the International Standards Organization (ISO) recognized the need for a basic framework in which computer networking standards could evolve. A subcommittee was appointed to develop a model for Open Systems Interconnection (OSI). The idea behind "open systems" is that different types of computer systems can communicate within the same network. The latest document from the subcommittee is Draft Proposal ISO/DP 7498. We will refer to it as DIS 7498. DIS stands for Draft International Standard.

Framework

The OSI model is both a framework for development of specific standard protocols and a reference model for data communication architecture.

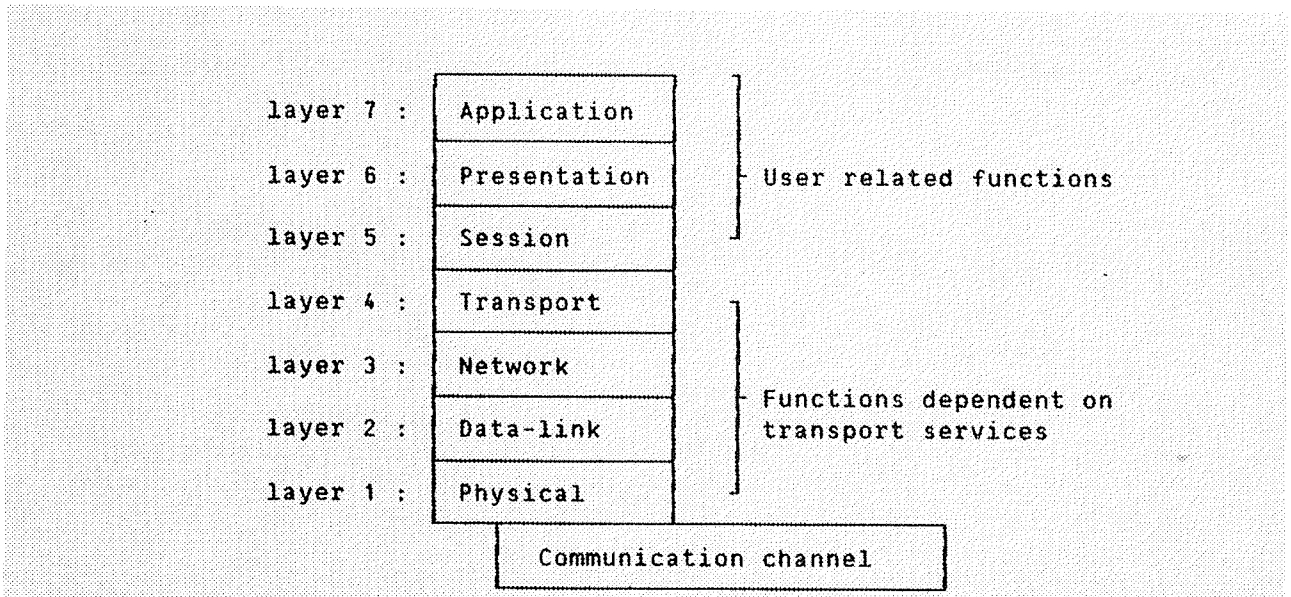
The reference model defines a number of logical functions that are needed in modern data communication systems. Furthermore, it divides those functions into a number of separated layers where each layer fulfills certain functions.

Features

Some of the general features are:

- Each layer depends on the services of the next layer below it. However, the layer depends on the services only, not on the way they are implemented or the protocol or hardware involved. Therefore, we can exchange the contents of one layer without affecting other layers.
- The top three layers are independent of the transport service used. The bottom 4 layers are dependent on the transport service, but are independent of the user related functions.
- The reference model defines a vocabulary of terms so that different people or standards describing the same things will use the same terms.
- The reference model does not prescribe how to implement communication systems.

The OSI Reference Model identifies 7 functional layers, called layers 1 to 7. Layer 1 is closest to the communication channel, layer 7 is the interface to the user application programs.



Note that this figure depicts only one node. A different node must exist in the other end of the transmission medium for communication to take place. That node has the same seven layers.

Terminology

The terminology in the written OSI recommendation is rather abstract and in many cases difficult to understand. We will anyway present some of it before we explain the contents of each layer in the next sections.

The OSI terminology defines an N-Layer (N=1,2,...,7) as a subdivision of the OSI architecture.

Entity

An N-entity is an active element within an N-layer. In other words, an entity is some (addressable) function within an OSI layer. For example a modem is a physical layer entity. The routing mechanism is a network layer entity. Notice that we did not say how this routing mechanism is implemented. We did not say that it is an RT program or a part of one (or perhaps several) program. As far as the OSI model is concerned, an entity is the abstract notion of an active element.

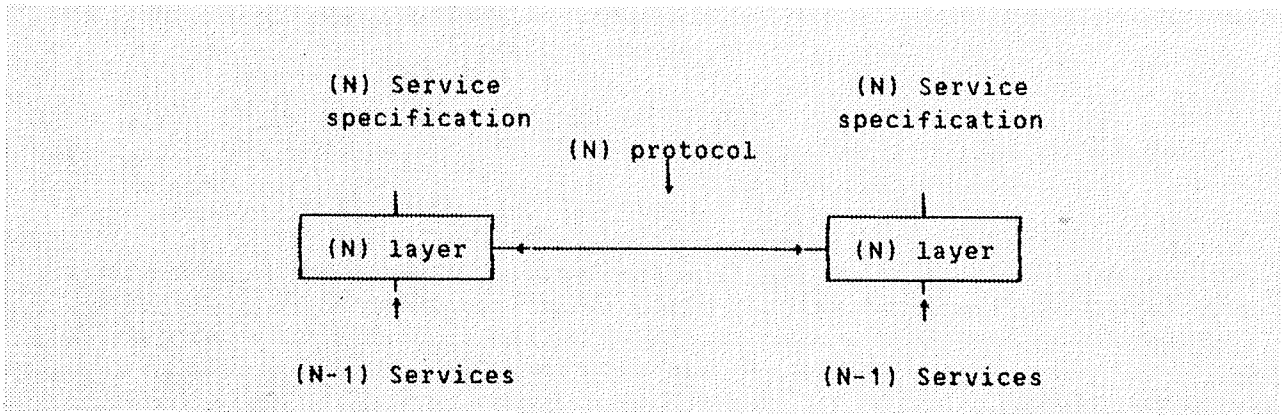
Service

Each layer provides a defined service to the layer above. The service is defined by the N-service specification. To provide the service, the N-layer uses the services of the N-1 layer augmented by new functions.

Specifically how the service is performed (by for example using monitor calls) is installation dependent. The OSI model says nothing about how to do this.

Protocol

The new functions are implemented by means of an N-protocol between two N-layers. The N-protocol is a set of conventions for exchange of data between the two corresponding layers.



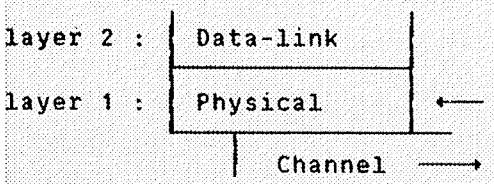
Being this abstract and sometimes rather vague, there may be many differences between OSI based systems. Using a metaphor, an OSI based system is like a car. A car is

something that has 4 wheels (usually), one steering wheel, a motor etc. Two cars of different make are not totally compatible. There are so many standardized functions, however, that a person has no major difficulty adjusting to driving a new car.

In the next sections we will present each layer in the OSI model.

THE PHYSICAL LAYER

Some features are already described in chapter 2 on transmission concepts. In this section we look at transmission from an OSI point of view.



According to DIS 7498:

"The physical layer provides mechanical, electrical, functional and procedural means to activate, maintain and deactivate physical connections for bit transmission between data-link entities."

In other words, the physical layer transmits and receives bit streams over a transmission medium (channel). The access to the channel involves mechanical matters, for instance connector size, pins, pin-diameter, male-female, etc.; electrical matters, for instance voltage levels, polarities, impedances.

Some physical links are permanent, others need to be established before data can be sent/received. This is what is meant by "activate, maintain and deactivate physical connections, etc."

Services

Some of the services provided by the physical layer to the data-link layer are:

- Identifiable physical connections
- Transport of data units. A data unit at the physical layer (physical-service-data-unit) is one bit when using serial transmission.
- Fault notification, for example if the carrier goes down

- Sequencing, meaning that the physical layer will deliver data units (bits) in the same order as they were transmitted.

Entities

Examples of physical layer entities are:

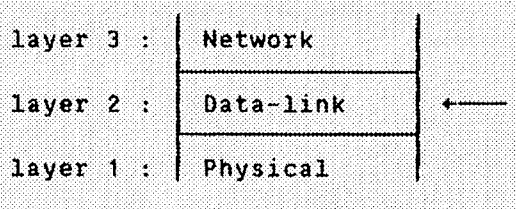
- Modems using telephone lines and frequency modulation to indicate 0 or 1 bits.
- A baseband coaxial medium and a transceiver to access the medium used by an Ethernet physical layer.

Example of physical layer service definition:

- V.24 or X.21

THE DATA-LINK LAYER

Many of the data-link layer features are described in chapter 4 on protocols. Most of that chapter deals with data-link protocols, although higher level protocols are mentioned. This section describes the data-link layer in OSI terms.



According to DIS 7498:

"The Data-link provides functional and procedural means to establish, maintain and release data-link connections between network entities and to transfer data-link-service -data-units. The data-link connection is built on one or more physical connections."

Services

Some of the services provided by the data-link layer to the network layer are:

a. Transfer of data in "data-link-service-data-units". The data-link layer accepts blocks of data (service-data-units) from the next higher layer and delivers these to the other end.

b. Detection and possibly correction of errors introduced by the physical layer.

c. Flow control on the data-link. Data-blocks (data-link-service-data-units) are transmitted across a data-link connection in the same order as they were presented to the data-link layer. This is called sequencing.

d. Setup and clearing of data-link connections.

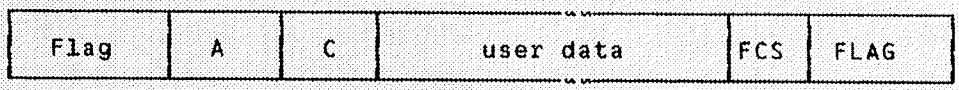
The data-link layer has improved the quality of service from the physical layer. The simple bit stream on the physical level can contain errors, and the data-link service detects transmission errors. To be able to offer this improvement, the data-link layer entities use a protocol between them. Examples of such protocols are BISYNC, and HDLC protocols.

Protocol control info.

Generally, protocols are implemented by adding Protocol Control Information (PCI) to the user data. At the receiving end, the PCI is used to follow the protocol. The PCI is removed and the user data is passed to the next layer.

It is like putting user data in an envelope and writing the PCI on the outside. The envelope can then be handled by the postal service. The postal service does not need to understand the user data inside. In the end, the envelope is opened and the user data is read by the receiver.

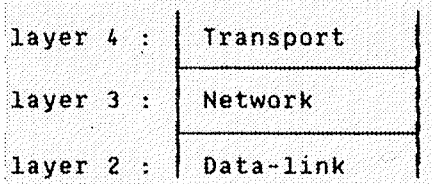
An example is the HDLC I-frame format:



The flag, A, C, and FCS fields comprise the PCI.

A number of data-link layer connections can be connected to build a data network.

THE NETWORK LAYER

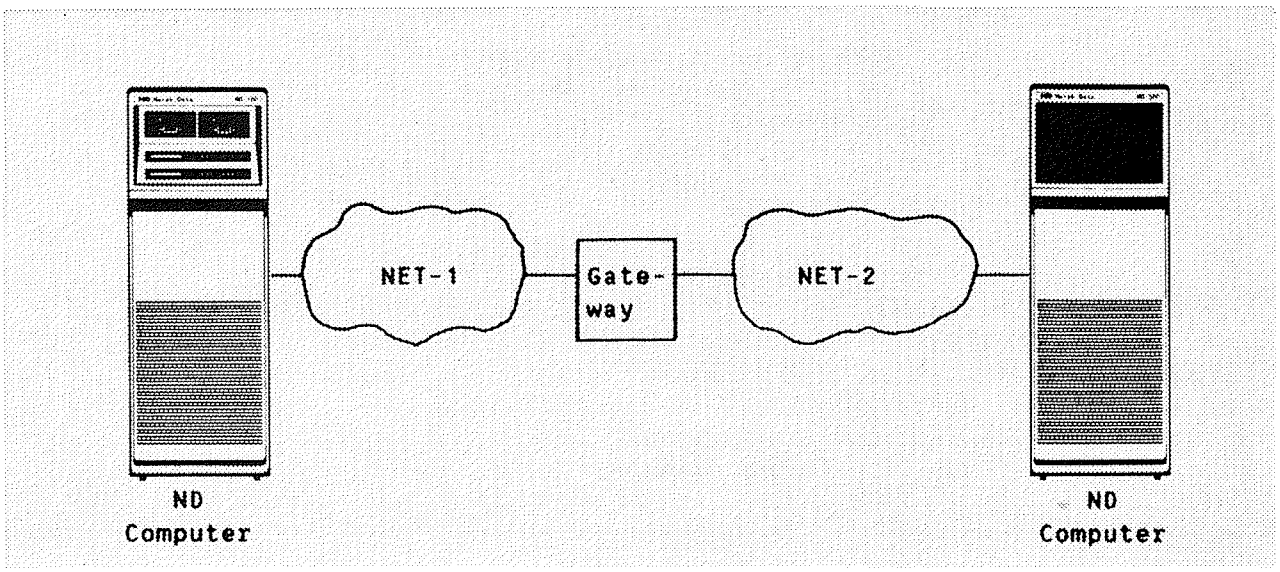


According to DIS 7498:

"The network layer provides the means to establish, maintain and terminate network connections between systems containing application entities and the means to exchange network-service-data-units between transport entities over network connections."

The main service provided by the network layer to the transport layer is routing and network address administration. The transport layer does not know or concern itself with how the network layer routes data through networks, except with the response times if relevant.

The network layer provides network connections across networks, even across several networks in tandem or in parallel.



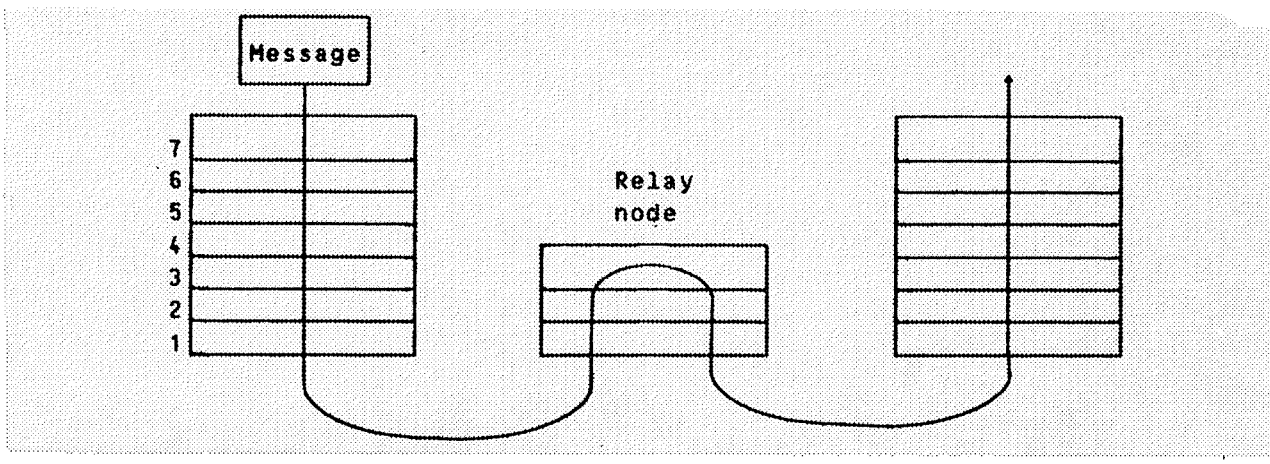
Networks in tandem.

Services summarized

Here are some of the services that the network layer provides:

- Network addresses for use by the transport layer to identify transport entities
- Establishing and releasing network connections
- Routing
- Transfer of network-service-data-units
- Flow control
- Sequencing
- Error notification

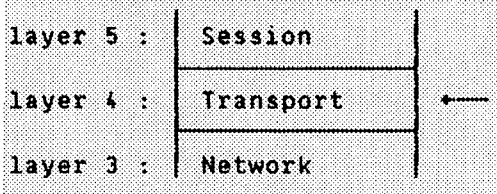
One well known network layer implementation is the network layer in X.25.



Routing. (Two systems with a relay node between)

THE TRANSPORT LAYER

The transport layer is the dividing line between the network dependent and the network independent parts of the OSI architecture. The layers below the transport layer are network dependent and the layers above deal with user related functions (such as transport of meaningful data).



According to DIS 7498:

"The transport service provides transparent data transfer between session entities and relieves them from any concern with the detailed way in which reliable and cost effective transfer of data is achieved."

Services

Here are some of the services provided by the transport layer:

- Transport addresses to identify session entities
- Transfer of data between transport addresses
- Establishment and release of transport connections
- Possibly choice of network based on cost/efficiency requirements from the session layer

The services provided by the layers underneath the transport layer, are transparent to the session layer. All the session layer is interested in is the transport service. This service is offered by the transport layer, but actually provided by all the 4 bottom layers. These 4 layers are sometimes called the transport subsystem, since they are all somehow involved with data transport. The transport layer is the interface to the user related functions (in layers 5 to 7) of the communication system.

The Transport Service Definition and the Transport Protocol have now reached the stage of being a Draft Proposal (DP) to become ISO standards (ISO/DP 8072 and DP 8073).

Protocol

The ISO transport protocol is a further development of the transport protocol developed by the European Computer Manufacturers Association, ECMA72.

The transport service definition specifies a certain level of service that is network independent. To offer this service,

the transport protocol must bridge the difference between the service offered by the network layer and the required service level.

The transport protocol can be very simple when the level of service from the network layer gives almost what we want. If the network layer service is far from what we want, a more complex transport protocol is needed to make up the difference.

This is reflected in the transport protocol specification that allows 5 classes of protocols, each providing more service:

```
Class 0 : Simple Terminal, Teletex, no error recovery.
          No multiplexing.

Class 1 : Basic Class, recovery on top of X.25 networks.

Class 2 : Multiplex, allows multiplexing on top of network
          connection.

Class 3 : Combination of class 1 & 2.

Class 4 : Error Detection & Recovery Class, extensive error
          detection and correction.
```

Teletex

Class 0 is already in use. It is CCITT's recommendation for TELETEX, which is an advanced "telex" service for document transfer.

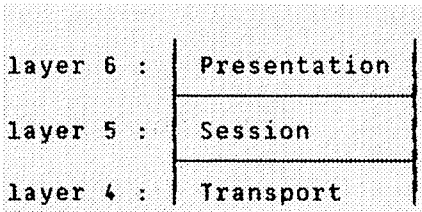
Classes 1 - 3 rely on reliable networks that have an acceptable notified error rate, like X.25 networks.

Class 4 does not rely on the network but checks for errors like duplicates, missing or damaged packets, or packets out of sequence. Further, class 4 is suited to run on top of a datagram network, for instance Ethernet.

Note

When we talk about multiplexing at a level higher than the physical, we mean transport of data blocks from different data streams on the same virtual connection between two N-layers.

THE SESSION LAYER



According to DIS 7498:

"The purpose of the session layer is to provide the means necessary for cooperating presentation entities to organise and synchronise their dialogue and to manage their data exchange."

Remote log on

One important responsibility of this level is that of remote log on, in which a user on one node logs on to a second. It appears to the second node as if the user is local, because the session layer masks the fact that the user is remote.

Dialog control

The session layer provides means for dialog control, for instance, your turn to send or my turn to send, etc. In addition, session layer connections can survive transport connection breakdowns and hide these from the presentation entities.

Bracketing

Another session layer activity is bracketing, also called quarantine data service. Bracketing prevents execution of a critical database update until all information needed for that update is received. This eliminates the chance of a partial update when transmission is interrupted.

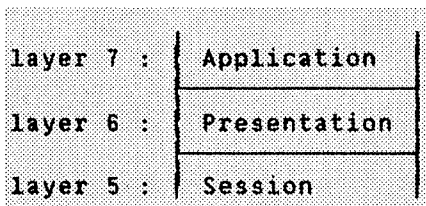
Services summarized

Some of the services provided by the session layer summarized:

- Session connection, establishment and release.
- Normal data exchange.
- Quarantine data service.
- Interaction management. (Your turn, my turn).
  - Two-Way-Simultaneously (TWS)
  - Two-Way-Alternate (TWA)
  - One-Way

The proposals for session service specification and session protocol are still being developed. ECMA has a Session protocol standard, ECMA75. ISO session working papers meet this standard.

THE PRESENTATION LAYER



According to DIS 7498:

"The Presentation Layer provides for the representation of information that application entities either communicate or refer to in their dialogue."

Services

Some examples of presentation layer services/entities are:

- Data conversion, for example between EBCDIC and ASCII
- Encryption and decryption
- Compression and compaction
- File translation
- Virtual terminal protocols

Encryption

Encryption is important to prevent people from using data stolen during transport. Especially on public data networks it is easy to tap into a communication link. Radio and satellite links are more vulnerable than cables. Both communicating parties must know the encryption algorithm. The data is encrypted by the sender and must be decrypted by the receiver.

Compression and compaction

Data compression eliminates repeated characters, such as blanks at the end of a line. Compaction removes data redundancy by encoding the data in an alternate form. Both compression and compaction reduces the length of the message. This improves the efficiency of the communication system.

Implementation

An interesting thing about encryption and compression is that they can easily be implemented in hardware. Since the OSI model says nothing about how to implement services, there is no prohibition against using hardware entities on high levels.

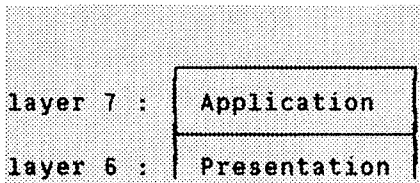
Virtual file

Since different types of computers use different file formats, an important function is translation from one format to another. This is a presentation layer task, since a format is used to present information.

Virtual terminal

A virtual-terminal protocol defines a hypothetical terminal. Mappings are established between a virtual terminal's defined functions and the real terminal's functions. If the programmer follows the rules established by the virtual-terminal protocols, her program will run on all of the supported network terminals.

THE APPLICATION LAYER



According to DIS 7498:

"As the highest layer in the reference model of Open Systems Interconnection, the Application Layer provides a means for the application-process to access the OSI environment."

A number functions are recognized, but developing standards and protocols for the application layer has hardly begun.

Some of the recognized functions are:

- Identification of communication partners (name, address)
- Establishment of authority to communicate
- Agreement on privacy means
- Authentication of communication partners
- Agreement on procedures for data integrity, for example checkpoints and rollback

Distributed functions

As distributed data processing develops we will most likely see distributed data bases and distributed operating systems in

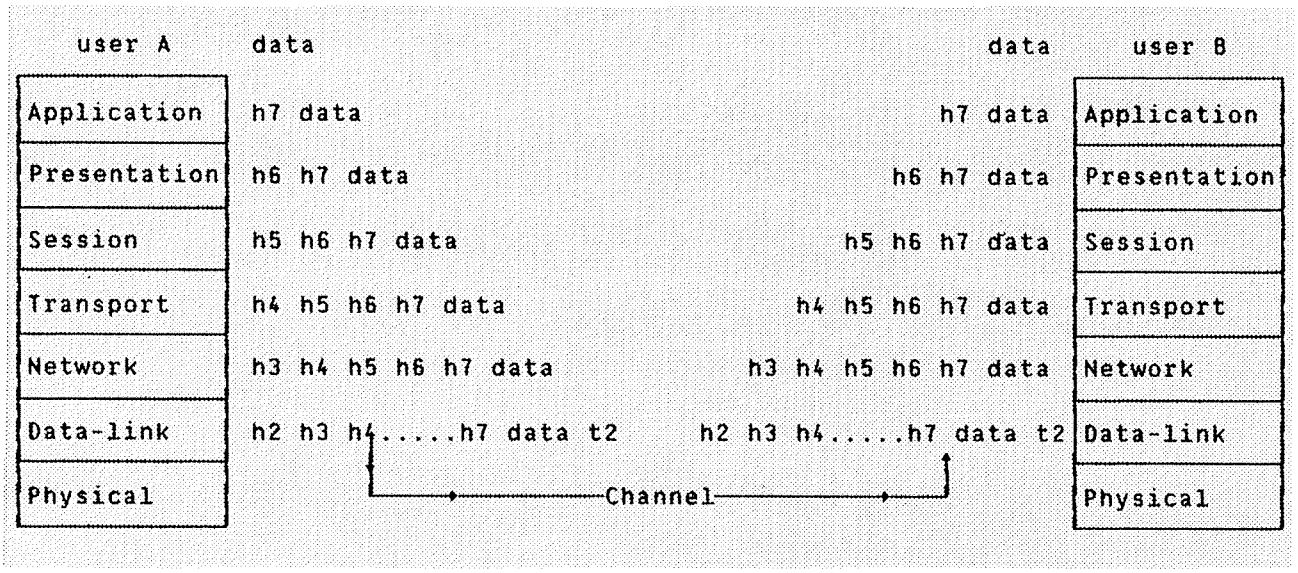
operation. The application layer will probably play an important role for this kind of distributing.

In a distributed operating system environment we will behave as if the whole operating system resides in the node we are connected to. It will be the task of the application layer to find out which node in the network can process our command, and communicate with that node. Also with truly distributed data bases we need not know where in the network the part of the database which we want to access resides. That can be left to the application layer.

COMMUNICATION BETWEEN USERS

What interest the users of a communication system is the exchange of messages between applications. This means that the way communication is performed by the communication system is transparent to the user. The users perceive direct communication between applications.

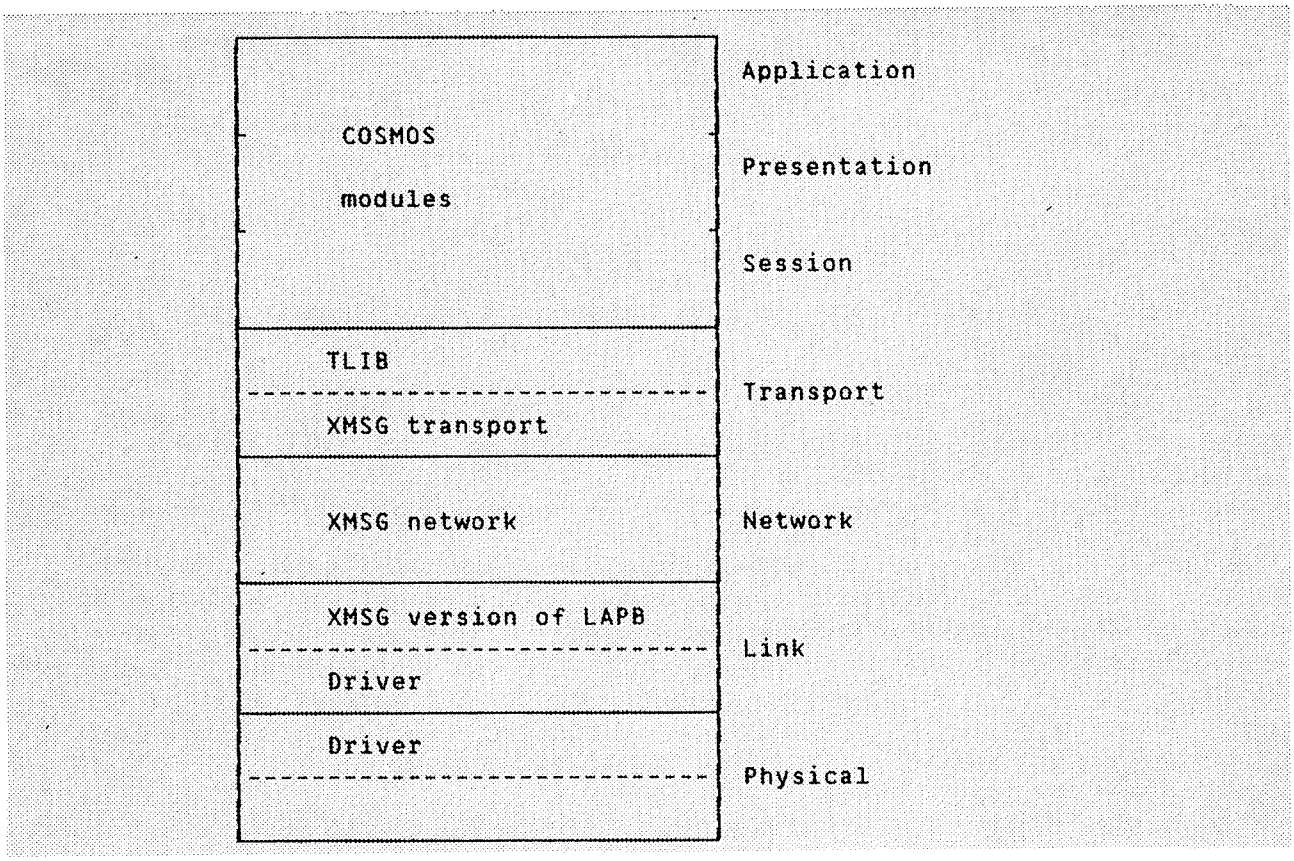
Knowing something about the layered architecture proposed by ISO, we can perhaps visualize a message as it travels from an application in one system to an application in another system.



The message from user A goes to the application layer which adds a header (H7). This is like wrapping the message in an envelope. The envelope goes to the next layer which wraps it into another envelope. This goes on until the whole frame is transmitted by the physical layer.

In the destination system the envelopes are removed one by one until the original message (with its format possibly translated by the presentation layer) is delivered to user B.

THE COSMOS IMPLEMENTATION



The figure presents in a very simplified way how some of the COSMOS modules fit together with the OSI model. Only the software modules are depicted. There is hardware both on the physical level and at the link level. Note that the proportions in the picture are not correct. The driver part is relatively smaller in size than it looks in the picture.

Driver

The driver is divided between layer 1 and 2. On the physical level we can have an HDLC card or a megalink card. The driver is a piece of software which reads data from such a card into a memory buffer, or writes from the memory buffer into a register on the card.

XMSG

XMSG is a system for generalized program to program communication within or between computers. XMSG is divided between layer 2, 3, and 4. It does not cover all of layer 2, and it does not cover all of layer 4. At the link level a modified version of LAPB is

implemented as a part of XMSG. The bit stuffing and FCS check are implemented in hardware.

XMSG is built up of modules distinguishing between XMSG link level, XMSG network level, and XMSG transport level.

TLIB

TLIB stands for Transport LIBrary. Since XMSG does not cover all of the transport layer, TLIB is implemented on top of it. TLIB provides an interface to the transport subsystem which conforms to the services defined in the OSI reference model.

For the layers above layer 4 (transport), we do not follow the OSI reference model. One reason for this is that the recommendation is rather unclear at the higher layers. The protocols defined by ISO are not followed at this time.

Options

Within COSMOS there are many options. For example, if we want to use a public data network (X.21 or X.25 based), we install servers within our transport subsystem. An example of a server is presented in the next section.

AN EXAMPLE OF SESSION TO SESSION COMMUNICATION

In this section we will present skeletons of two communicating programs based on the TLIB implementation at the transport level.

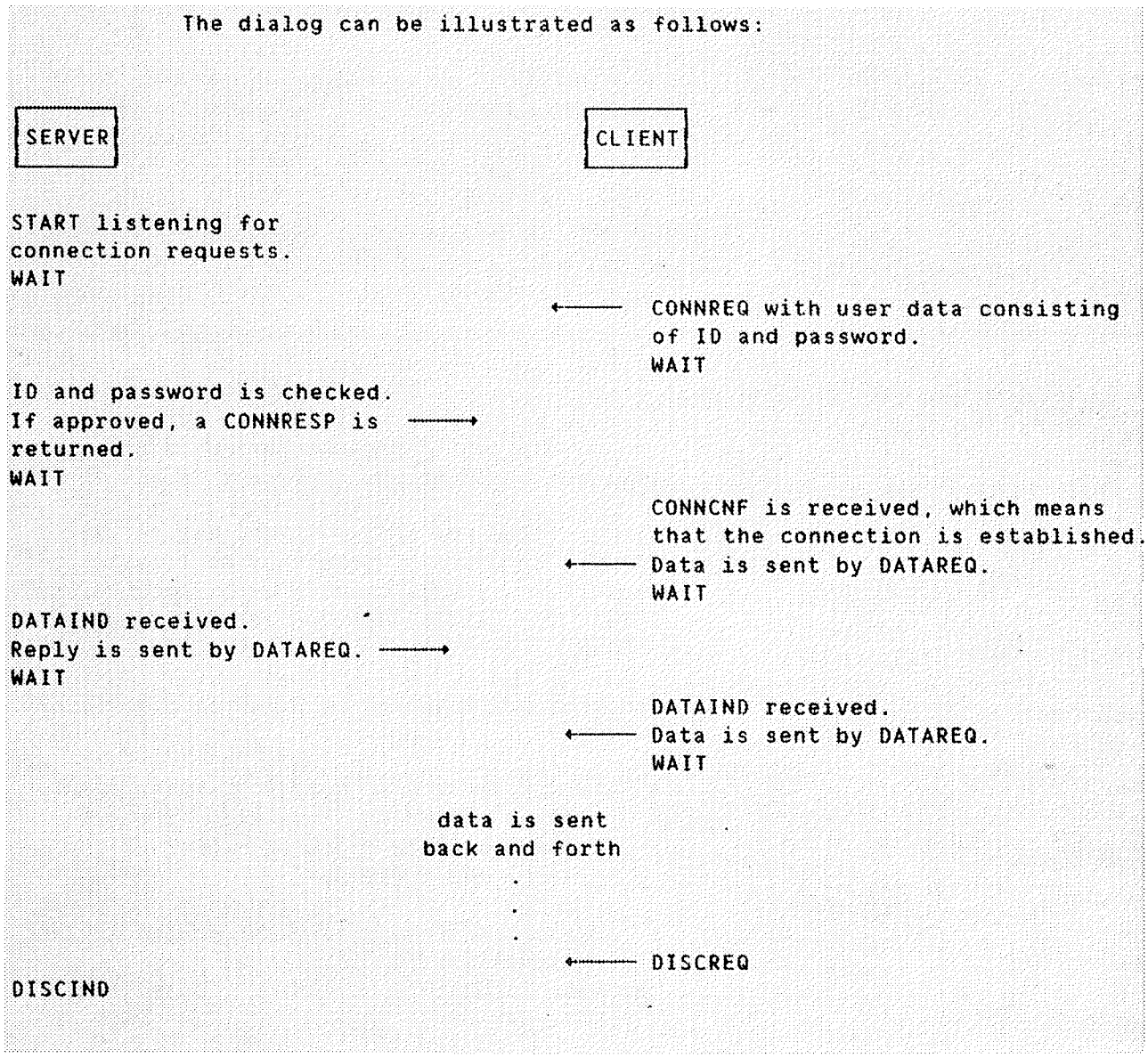
The programs are called SERVER and CLIENT. The SERVER performs some kind of service (specifically what service is of no interest here). The CLIENT gets the service from the SERVER.

The session presented is based on the following:

- Establishing the session is initiated by sending a CONNREQ (connection request) message with ID (identification of the client) and a password.
- The server perceives the incoming CONNREQ message as a CONNIND (connection indication). It checks the ID and password.

- Assuming that the server approves the request, it sends a CONNRESP (connection response) back to the client.
- The client perceives the incoming CONNRESP as a CONNCNF (connection confirmation). The connection is now established.
- CLIENT and SERVER now exchange data, based on a two-way-alternate scheme.
- When finished with the data exchange, either party sends a DISCREQ (disconnect request). This is perceived as a DISCIND (disconnect indication in the other end). The dialog is finished.

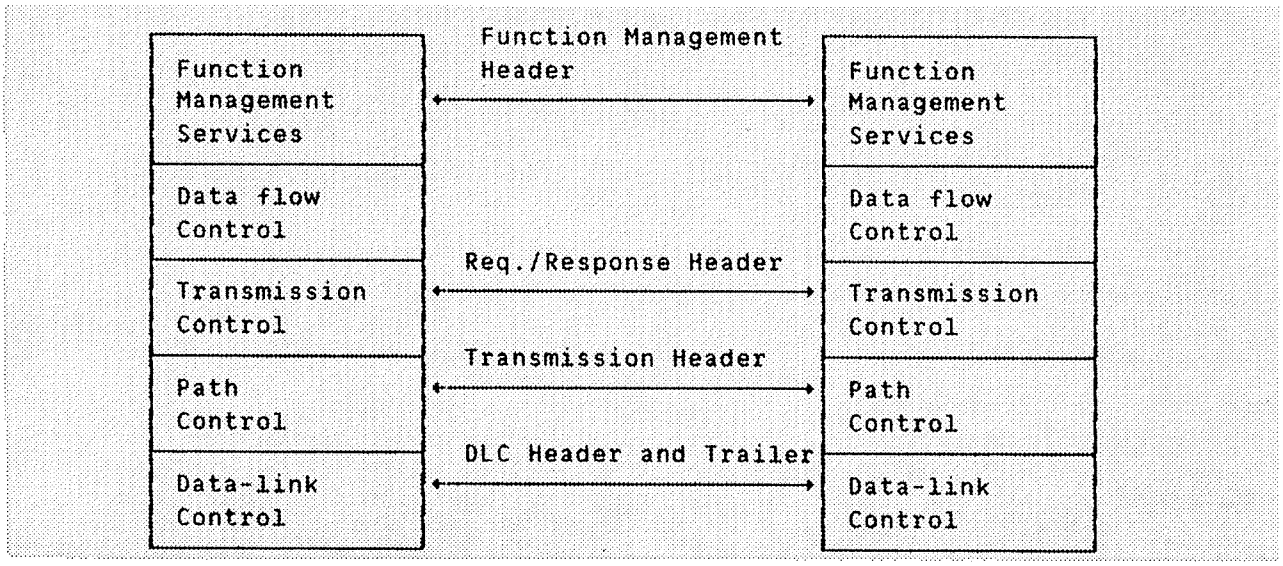
The dialog can be illustrated as follows:



SYSTEM NETWORK ARCHITECTURE

IBM has its own special networking concept called SNA, which stands for System Network Architecture. The architecture is layered and resembles OSI.

The communication between the corresponding layers, called peer-layers, can be depicted as follows:



On the Data-Link Control level IBM's SDLC protocol is used. SDLC stands for Synchronous Data-Link Control, and is IBM's version of HDLC. This layer also takes care of the physical transmission.

The Path Control layer is similar to the network layer in OSI.

If we compare the Transmission Control layer with OSI, we can say roughly that it consists partly of session layer functions and partly of transport layer functions.

The two highest layers deal with some user related functions and they are difficult to compare with OSI. As for OSI the application resides above the top layer.

SNA with COSMOS

Norsk Data has made software emulating RJE and interactive 3270 terminals in an SNA network. This means that you for example can communicate from your ND terminal, via the ND computer, with host computers in some IBM SNA network.

CHAPTER 6
WIDE AREA NETWORKS

TYPES OF WANS

As we mentioned on page 5, some public transmission facilities are themselves computer networks. The public data network depicted on page 5 is a public packet switched data network. We also mentioned that a wide area network can be built around telephone lines.

Telephone lines

The use of telephone lines involves several disadvantages.

- The transmission technology was developed for speech rather than binary digits.
- If we use dial-up lines, the transmission is slow.
- Dedicated connections between every computer in the network are very expensive. If there are n computers in the network, then (1/2)n(n-1) lines are needed and have to be paid for.

Digital, synchronous networks

Except for networks based on telephone lines, there are two main types of Wide Area Networks (WANs):

- Public packet switched data networks (based on X.25)
- Public circuit switched data networks (based on X.21)

A computer may subscribe to a service within such a network and be able to communicate with any other (compatible) subscriber in the same network. The rest of this chapter is dedicated to circuit switched and packet switched networks.

CIRCUIT SWITCHED NETWORKS

Circuit switching is based on the principles used in telephone and telex communications: A connection is established between two points. Once the connnection has been set up, two users have exclusive access to it. A circuit switched connection may be compared to a pipeline connecting two points. What is sent in the pipeline, and the intensity of use, is a private affair involving nobody but the two users.

Equipment compatibility

The equipment used at the end points must be closely matched. For instance, it must operate at the same transfer rate and use the same coding system.

Circuit switching takes place at the physical level. The connection seems transparent to the layers above. The line remains open during breaks in transmission. A disadvantage is the fact that the line utilization may be low, which can lead to high costs.

Connection establishment

One big advantage is the very fast connection establishment. Disconnect is comparably efficient. If the price is low on connection establishment as compared to keeping the line open when it is idle, a disconnect with a subsequent reconnect when there is data to send will pay off (provided the idle time is long enough).

Features

The network cannot assist with error detection or correction. This has to be done at higher levels.

Some advantages:

- Very fast connection establishment
- Guaranteed throughput
- Fixed delay
- No imposed protocol in the data phase

Some disadvantages:

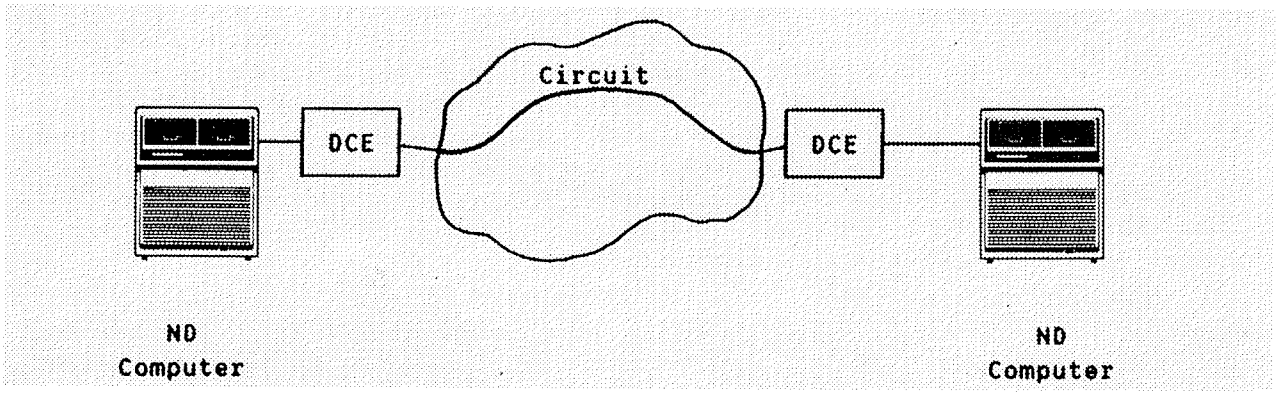
- No buffering in the network
- No speed conversion
- No code conversion

Implementations

The telecommunication authorities of the Nordic countries have cooperated on the specification, X.21, and the procurement of a data network based on circuit switching. The network is called Nordic Public Data Network (NPDN). Japan and Germany have also installed public circuit switched networks.

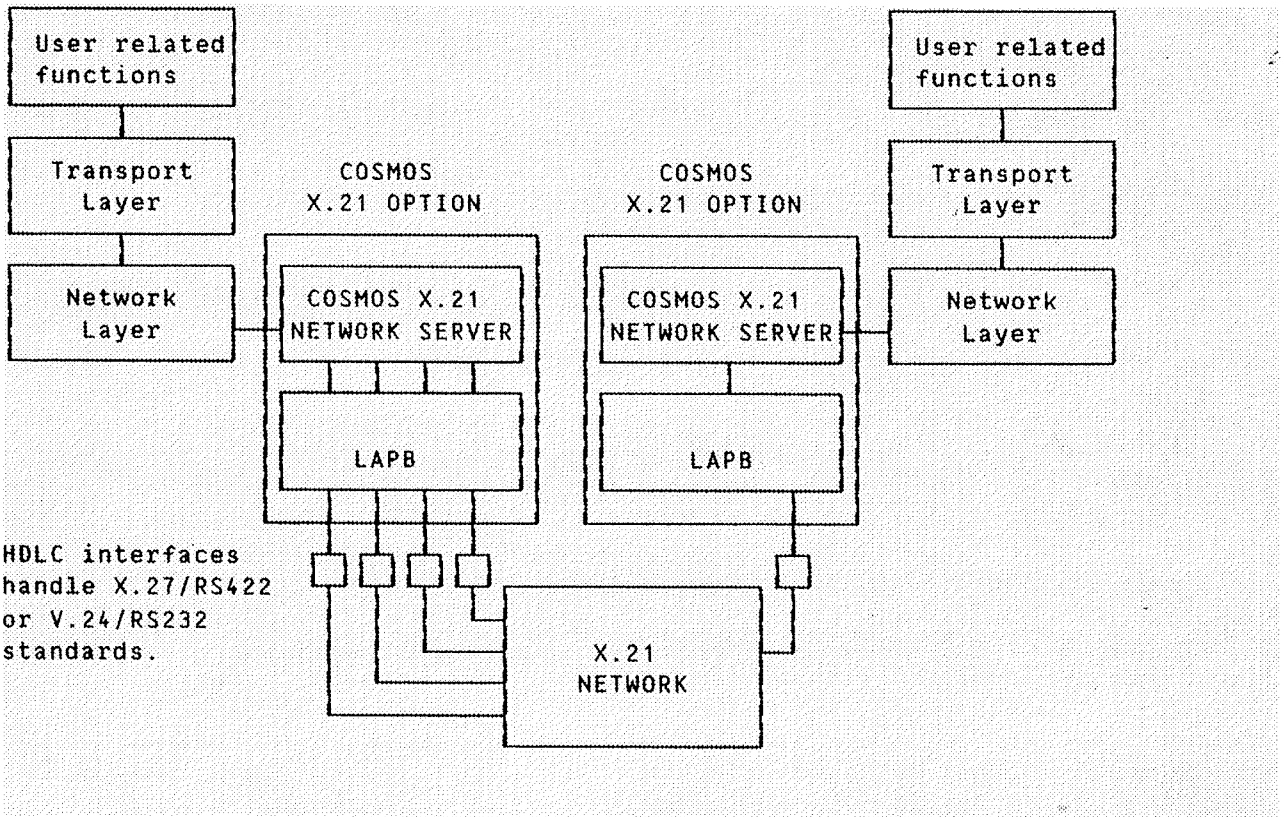
The NPDN has a very low connect charge, making it feasible to disconnect the idle line.

For a description of the X.21 and related interfaces, please see page 22.



A circuit switched data network

THE COSMOS X.21 OPTION



COSMOS X.21 OPTION

The COSMOS X.21 option allows use of an X.21 based circuit switching network as a transmission medium under COSMOS. The X.21 Network Server functions as a bridge between two XMSG entities in the COSMOS Network Layer.

PACKET SWITCHED NETWORKS

If data networks based on circuit switching are like telephone communications, then packet switching might be compared to transmission of telegrams. The telegram has a delivery address attached to it, and someone routes the telegram to that destination. Before we proceed with the explanation of packet switching, we will discuss the topic of addressing.

Addressing

Before data can be transported, someone must know where to transport it. The destination of the data is indicated by an address in some form.

Different systems often have different addressing forms and conversion is then needed. Addresses used in one system are usually totally incomprehensible to another system. Conversion has to take place at the connection points between the two systems.

Adressing structures may be flat or hierarchial. In hierarchial addresses, the addresses are constructed from fields that each define a part of the address. For instance:

Address = Country/Network/Computer/Socket

In hierarchial systems, gateways only need to use parts of addresses to execute the routing function; the unused parts are passed on for use at lower levels.

In flat address spaces, the location of all addresses must be looked up in a table since the address itself has no structure.

Addresses, names and routes are often confused:

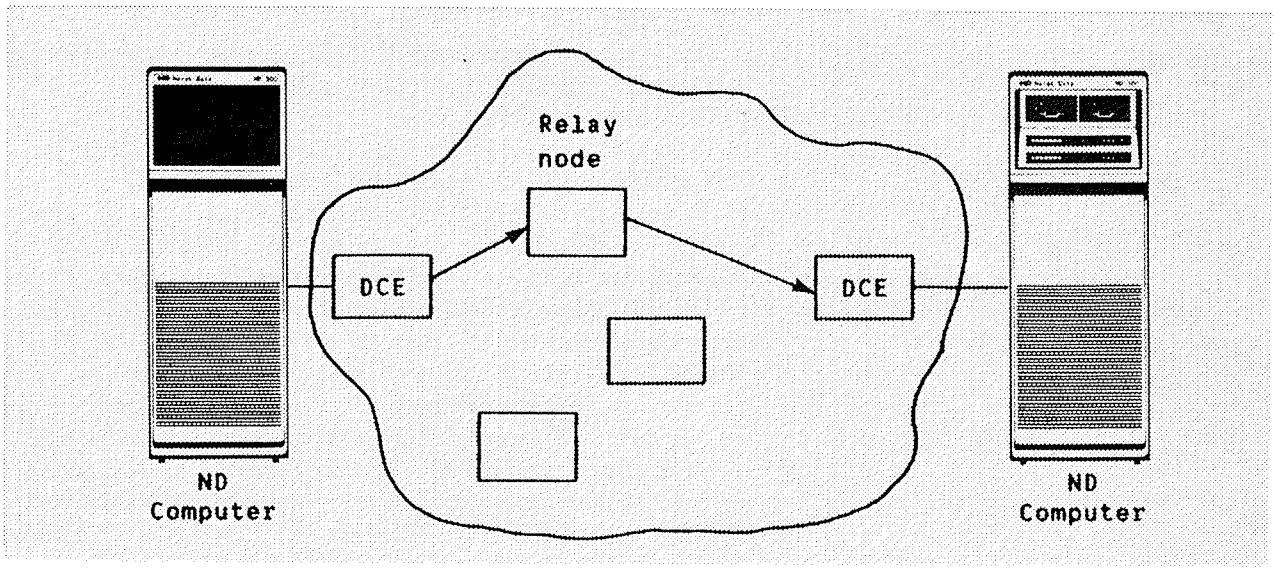
```
Name    : what you call it
Address : where it is
Route   : how you get there
```

The packet concept

In packet switched networks all data is structured into blocks called packets. Each packet carries the destination address. The network stores these packets and routes them to the destination address with a "store and forward" technique.

Store and forward

This means that a packet is sent from one node in the network to the next where it is stored temporarily. The node looks at the destination address of the packet and finds out which is the next node along the route of transmission. When the node is ready and a link to the next node is available, the packet is forwarded. Note that packets with different source and destination addresses may be logically multiplexed on the same link.



Packet switching.

There are basically two different modes of packet switching:

- Connection oriented (virtual circuit)
- Datagram oriented (connectionless)

Virtual circuit

With connection oriented data communication, an association is established between two points before data is exchanged.

In connection oriented packet switching, the packets do not need to carry a full address. Instead, a short address (indicating the connection) is used to save transmission capacity.

The relation between the short address (the channel no. in X.25) and the real address must be remembered by all the switches along the way. This relation is deleted when the connection is released.

The data exchange can be controlled connectionwise. For example flow control can be applied for each connection. The connection provider usually guarantees that no packets are lost or duplicated or delivered out of sequence.

Connection oriented communication is typically used when larger amounts of data are exchanged between the connection endpoints. The overhead in establishing the connection is then very small compared to the data exchange phase.

Datagrams

Connectionless communication does not require a connection between endpoints.

In a packet switching system all packets must then carry the full address since no connection (or short address) exists.

A typical use of connectionless communication is in transaction systems where the data exchange between two points consists of a sequence of one question and one answer. Keeping a connection through the whole sequence of questions and answers is an expensive solution. On the other hand, the overhead of setting up a new connection for each question and answer is large compared to the data exchanged.

Connectionless systems are often called datagram systems. From the network's point of view a datagram is a data packet that travels from source to destination without any relation to previous or later packets. Within the network the datagram has no history.

In a datagram network packets may be lost, duplicated, or delivered out of sequence.

Some packet switched networks

Packet switching was first successfully demonstrated in large scale by the ARPANET in USA. Since then a large number of countries have or will have public data networks in operation based on the X.25 recommendation from the International Consulative Committee for Telegraphs and Telephones (CCITT).

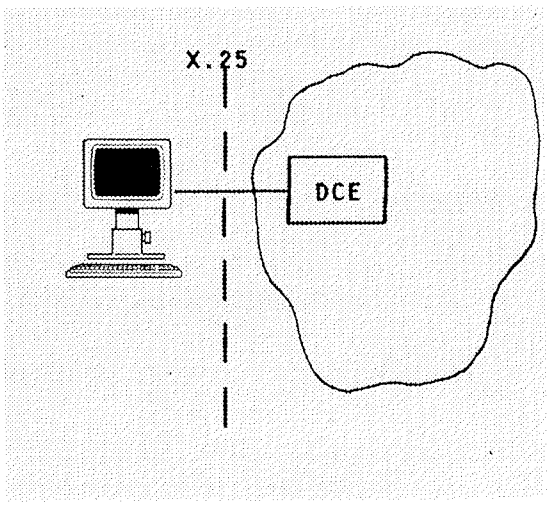
In most countries outside Scandinavia, packet switching is the most commonly used technique in public data networks. Here is a list of countries and the name of their public networks based on the X.25 recommendation:

USA	: Telenet Tymnet	England	: PSS / IPSS
Canada	: Datapac	EEC	: Euronet
Germany	: Datex-P	Netherlands	: DN-1
Austria	: Datex-P	Switzerland	: Telepac
France	: Transpac	Norway	: Datapak
Belgium	: DCS	Sweden	: Telepak
Finland	: Finnpak	Spain	: NID/Iberpac
Portugal	: Datacess	Denmark	: Datapak
Greece	: Helpac	Luxembourg	: Luxpac
		Italy	: Itapac

From Norway it is possible to connect to most European countries and to USA.

The first version of X.25 was published in 1976, and later amended in 1978, 1980, and 1984. X.25 defines the interworkings between a packet-mode DTE (DTE = Data Terminal Equipment, user equipment) and a DCE (Data Circuit Terminating Equipment). The DCE is part of the public network.

THE X.25 RECOMMENDATION



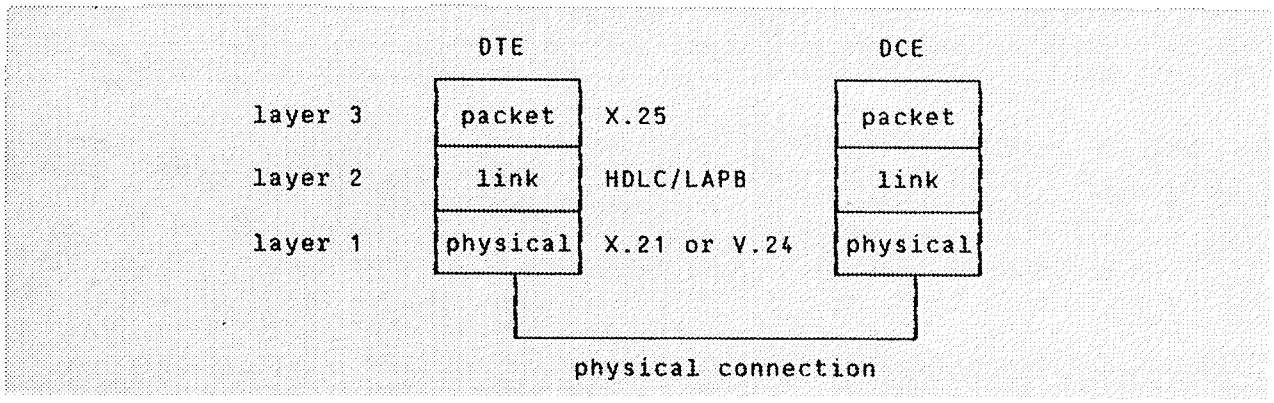
X.25 is an interface, recommended by CCITT, between a DTE and a DCE in a packet switched public data network. X.25 does not relate to the physical transmission of bits, but describes a set of rules for the exchange of data packets between DTE and DCE. Some of the rules have to do with the packet formats, other rules describe how certain control packets can be used for flow control, congestion control, error detection, recovery etc.

Note that the X.25 recommendation does not say anything about how packets are transported within the public data network, so there are many ways of implementing a public packet switched network based on X.25.

A packet switched network is based on the transport of packets which include destination addresses, by sending the packet one step at a time from station to station. Such a station which resides entirely within the public data network is called a node and is dedicated to data transport. The node is a computer which will analyze the packet header

when a packet is received, and find out which route to take through the network, i.e., which node to forward the packet to.

The X.25 recommendation specifies 3 levels: Packet layer, link layer and physical layer. Note that these layers are not exactly as in OSI. The relationship between the layers of X.25 and OSI will be explained later.



The illustration shows how the X.25 interface has to do with a logical (on level 3) rather than a physical connection between DTE and DCE. The DCE is a node in the network.

Link level

The X.25 recommendation says that the protocol at the link level may be either LAP (Link Access Protocol) or LAPB (Link Access Protocol Balanced), but LAPB is preferred. Both of these link protocols are varieties of HDLC (High level Data Link Control), please refer to page 41.

Physical level

The X.25 recommendation says that on the physical level (layer 1) we may choose between a V.24 or an X.21 interface. Please see pages 21 and 22 for a description of these.

THE X.25 PACKET LEVEL

From the DTE, we can establish conversations with many other DTE's at the same time. This means that several users of the same computer system and/or several application programs can conduct their conversations simultaneously over the one link to the network. In other words, the data streams are logically multiplexed.

The data belonging to each conversation is packed into packets that contain a data part and a header part that contains control information. Some packets, called control packets, only contain the header part. The control information indicates whether the packet is a data packet or a control packet.

Fast select

X.25 is based on connection oriented communication. In the 1984 version there are no datagram possibilities in X.25 based networks, but there is something called "fast select" which resembles datagrams. In fast select we typically send the user data in the same packet that establishes the connection. The answer we receive as user data in the disconnect packet. In the rest of this section we will talk about truly connection based communication where we have a connect phase, a data transfer phase, and a disconnect phase.

Connection oriented

Part of the control information in the packet header is a "logical channel number" that is unique for each conversation from a DTE. The network and the DTE can demultiplex data streams from the link by looking at the logical channel number.

Virtual circuit

A "Virtual Circuit" is an association between the logical channel number A at the DTE-a and the logical channel number B at DTE-b. A packet sent by DTE-a on logical channel A is delivered to DTE-b on logical channel B.

PVC and SVC

Virtual circuits may be permanent (PVC = Permanent Virtual Circuit), always established by the network. Or they may be Switched Virtual Circuits (SVC), established by the users of the network.

A number of operations can be applied to virtual circuits:

- Open a switched virtual circuit from the user to DTExxx. (CALL)
- Close a certain switched virtual circuit. (DISCONNECT)
- RESET a virtual circuit (PVC and SVC), to indicate lost data or errors in the network.

- Send/Receive data packets over a virtual circuit.

The switched virtual circuits can be in several different states:

- SVC Setup or Clearing state
- Information Transfer state

For each virtual circuit a number of facilites can be chosen during the circuit setup. Here are some of these optional user facilities:

- Extended packet numbering (modulo 128)
- Other packet window size (default = 2)
- Reverse charge, meaning that the called DTE pays for this call
- Incoming/outgoing calls barred
- Fast select
- Window size negotiation
- And many more

FURTHER DETAILS ON THE X.25 PACKET LEVEL

This section explains the X.25 packet level in more detail. If it is too detailed for your purpose, you may skip it.

SVC setup

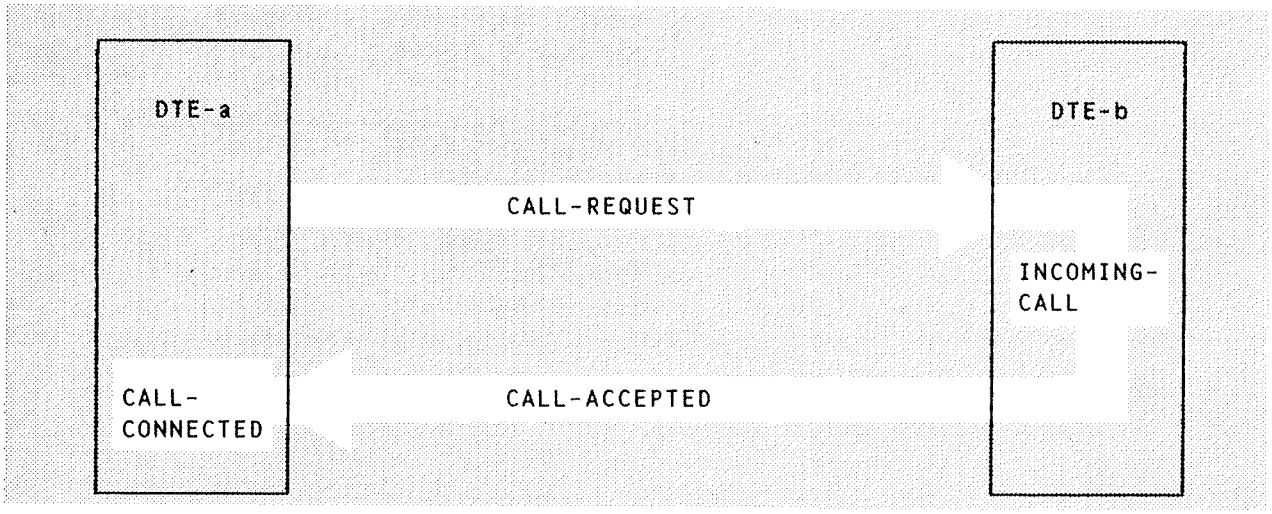
The following steps have to be performed to establish a switched virtual circuit:

- The calling DTE-a (= the one who wants to establish the SVC) sends a "CALL-REQUEST" packet to the network. The CALL-REQUEST packet contains the number of the DTE-b to which the SVC is to be established.
- The network delivers this CALL-REQUEST packet to DTE-b. DTE-b perceives the packet as an INCOMING-CALL packet.
- DTE-b answers with a CALL-ACCEPTED packet to accept the incoming call.

- The network now delivers the CALL-ACCEPTED packet to DTE-a. DTE-a perceives the packet as a CALL-CONNECTED packet. The SVC can be used for data transfer.

- If DTE-b does not want to accept the call from DTE-a, it can answer instead by a CLEAR packet, thus refusing the call.

This diagram illustrates the successful setup of a call:



Data transfer

The data transfer state allows transport of data packets across a virtual circuit. The flow for each virtual circuit is controlled separately. One circuit may be sending happily, while another is blocked.

Flow control

The flow control uses the same window principle as we have seen before in the HDLC link procedures (page 41). All data packets carry a send-sequence number P(S), all acknowledgements carry a receive-sequence number P(R). When P(S) - P(R) = window size for a certain virtual channel, no more data can be sent until a packet-level acknowledgement is received on that channel.

Note that the flow across a link is fundamentally different from the packet-level flow. Several virtual circuits are in general multiplexed on one link. The flow on one or more links may function perfectly, where as the higher level gets congested. This could happen if only one link does not function well. You probably understand now why acknowledgements and flow control exist both on the link and the packet levels.

End to end control

Since X.25 is not an End-end protocol but rather a network interface definition, a number of questions remain. One of these is whether packet acknowledgements come from the network or from the remote DTE. This could be very important for applications using X.25.

Another question is whether small packets can be combined to one larger packet by the network, or whether a large data packet can be broken into several smaller ones.

To give the user control over those matters, there is a bit called the D-bit in the packet header (D = Delivery confirmation). If the D-bit is set, the acknowledgement for that packet will come from the remote DTE, rather from the local DCE.

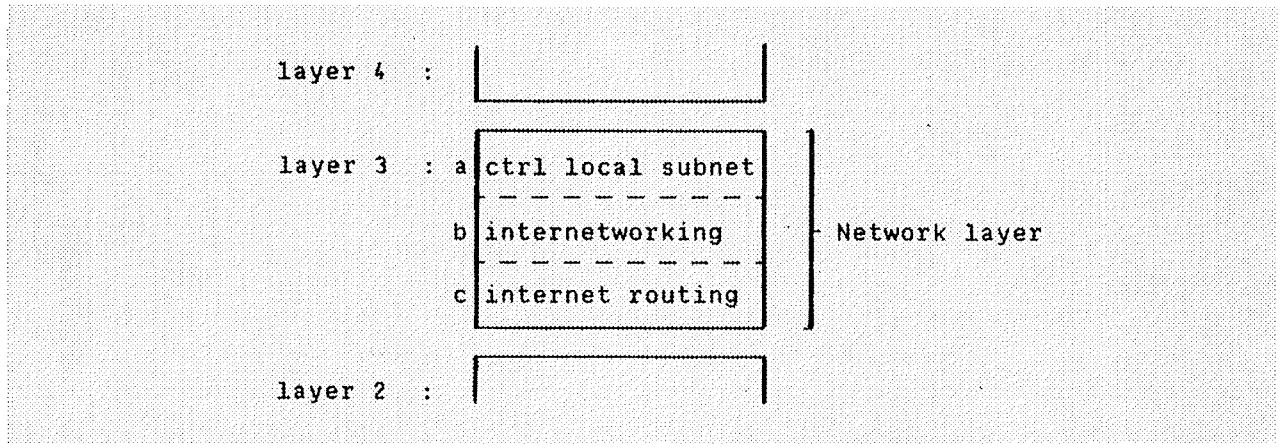
Another bit in the header is the M-bit (M = More Data Mark), telling the network or the receiving DTE whether a sequence of packets belong together and should be combined.

GATEWAYS

Packets may be sent across several networks. This is for example the case if a DTE in Norway communicates with a DTE in USA. Then there must be a mechanism which adapts the packets to the other network. This mechanism is called a gateway. A gateway may be a node in a network. It may also, for example, be a piece of software doing conversion. Conversion is necessary if the maximum packet size is different in the two networks. In that case the gateway has to divide the packets that are too big into smaller ones.

X.25 AND OSI

X.25 is not concerned with how the packets are transported within the network, nor is it concerned with how packets are transported between networks. If we relate this to the OSI model, these are clearly network level functions. We can divide the OSI network layer in three parts where layer 3c does internet routing, layer 3b does the internetworking and layer 3a controls the local subnet.



One standard for internetworking and internet routing is X.75. This standard is also used for routing within an X.25 based network.

PACKET ASSEMBLY DISASSEMBLY (PAD)

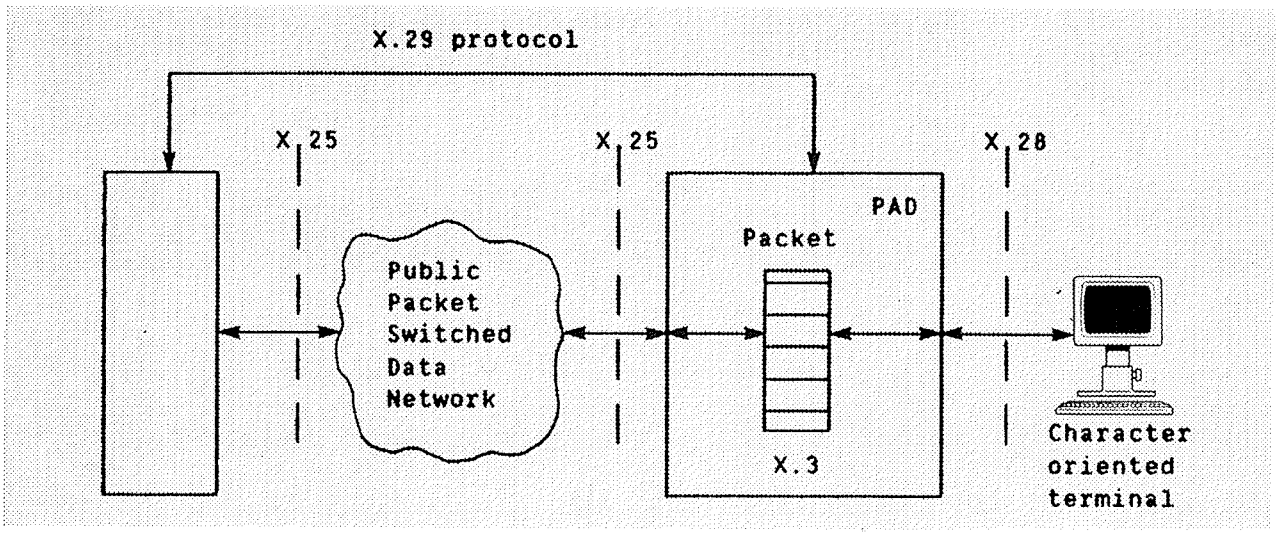
X.25 based networks use synchronous transmission. It is therefore easy to connect computers or devices having synchronous interfaces and which otherwise obey the X.25 standard.

If we want to connect an asynchronous unintelligent terminal to an X.25 network, a PAD (Packet Assembly Disassembly) is needed to pack the characters so the format is acceptable to the network. The PAD assembles the characters from the terminal and disassembles the characters from the network to the terminal.

The PAD may be a hardware box or it may be a piece of software residing in a computer which the asynchronous terminal is directly connected to.

Triple X

X.3 is the name of the standard for assembly and disassembly of packets. In addition to this we need an interface between the PAD and the asynchronous terminal, the X.28 recommendation. We also need a protocol between the PAD and the packet oriented DTE which the asynchronous terminal is ultimately communicating with. This last protocol standard is called X.29.



The relationship between X.25 and triple X.

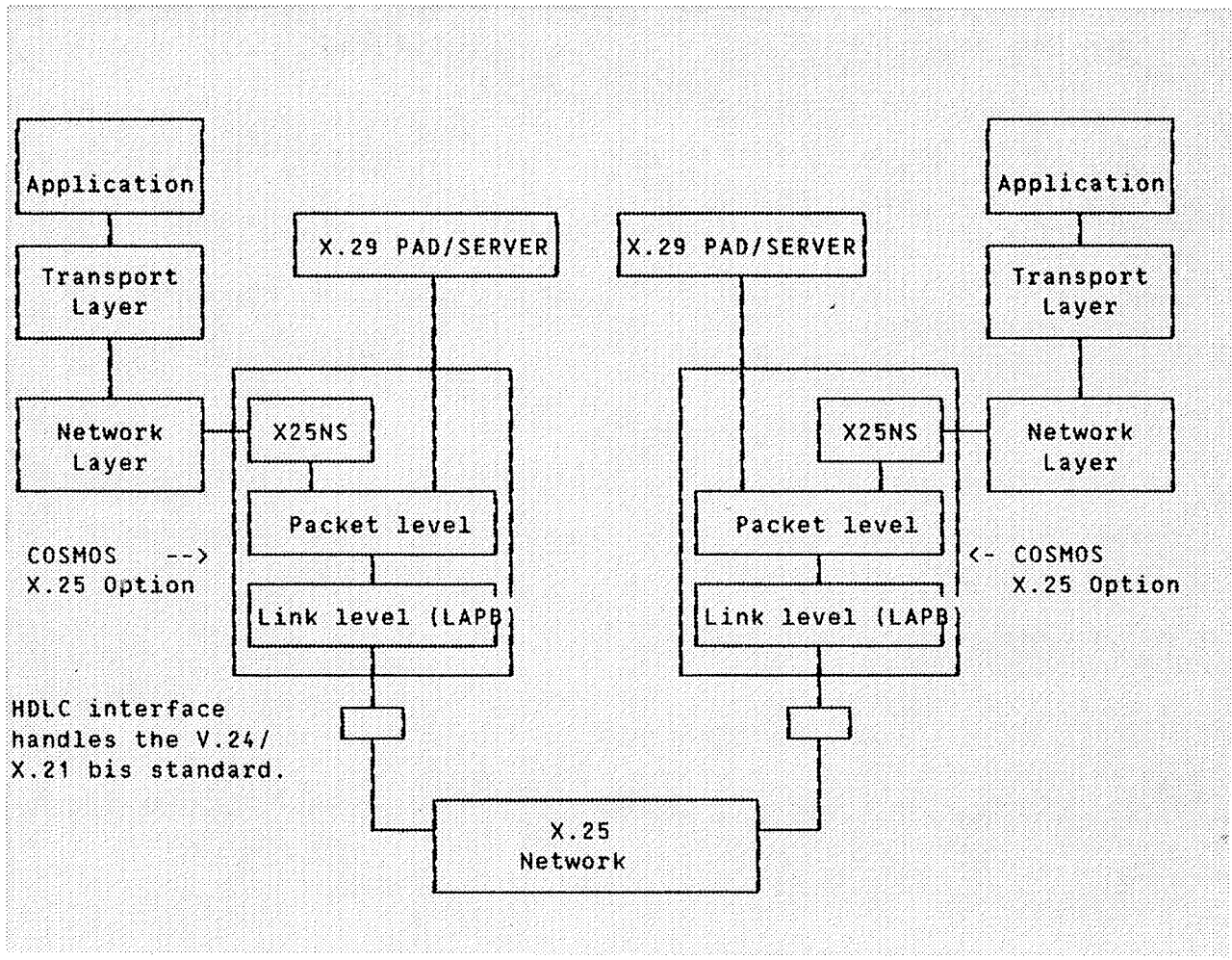
COSMOS WITH PUBLIC PACKET SWITCHED DATA NETWORKS

The COSMOS X.25 option is a subsystem used to interconnect COSMOS local area networks through an X.25 packet switched network.

The COSMOS PAD program, together with the COSMOS X.25 option, is designed to make it possible for any user terminal connected to an ND system to communicate with remote hosts over an X.25 network.

The remote host may be of any type, Norsk Data hosts or hosts manufactured elsewhere, provided that it supports terminal access via an X.25 network based on X.3, X.28 and X.29 (triple x).

The COSMOS X.29 SERVER, gives 'host-service'. With this subsystem terminals connected to a PTT PAD service or terminals on computers with a similar PAD can acess the ND computer as if they were directly connected. The computers with the PAD service need not be ND computers.



Communication between ND computers using X.25.
X.25NS stands for X.25 Network Server.

CHAPTER 7
LOCAL AREA NETWORKS

PURPOSE AND CHARACTERISTICS OF LOCAL AREA NETWORKS

If we look at the information flow in an organization, we will probably find that most of this information remains within the organization. Some studies conclude that only about 20 % or less of the information leaves the organization. In other words: 80 % or more of the information flow is local.

The interest in moving data quickly has grown enormously during the last years. This is caused by the increase in distributed processing. Since the cost of computing is falling, more distributed processing will become available in the form of work stations, personal computers and minicomputers.

Technology for fast data transport has been developed especially for local distances. The multiaccess cable provides the basis for many local networks. This is true to such an extent that many people talk about LANs as if they necessarily are based on multiaccess cables. An explanation of "multiaccess cable" will be provided soon.

We will now list what characterizes a local area network. It is based on information given out by IEEE (Institute of Electrical and Electronics Engineers, USA).

Characteristics

Characteristics of a local area network:

- Short distances within the network, which typically serves a set of offices, a campus, or a factory, etc.
- High speeds, typically in the megabit range (one megabit is one million bits)
- Low error rate, typically less than one gigabit. This means that less than one bit out of 10^9 becomes distorted during the transmission.

TYPES OF LAN

In the previous section we mentioned the multiaccess cable. On such a cable we can plug work stations and computers in such a way that each of these units can reach any other unit without sending the message through a relay node. This way we can reduce overhead with routing and we avoid some cable spaghetti. If you wonder about the meaning of "cable spaghetti", please look at all the cords in back of a computer which has many terminals hooked up.

There are also other types of arrangements. For example, COSMOS may be used with point-to-point connections. Norsk Data also has the Ethernet option which is based on multiaccess.

Topologies

There are three network topologies that are specially interesting:

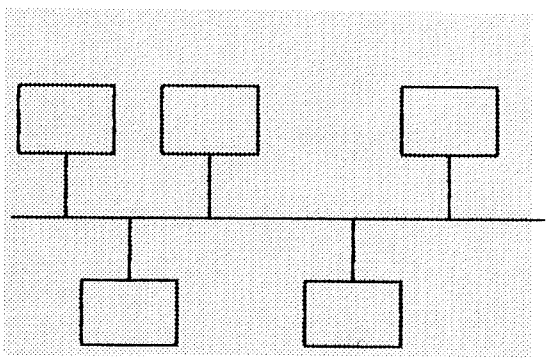
- Bus networks
- CBX based networks
- Ring networks

BUS NETWORKS

In a bus network the computers and the work stations are connected by a single line. This line is shared by all the units. The bus makes multiaccess possible.

Work station

Since the term "work station" is widely used, it may be appropriate to explain what it means. There is one difficulty: The term is used in several different ways. What seems to be common between the different usages is that a work station is a one person system designed to do some kind of work. Very often (but not always) a work station is a component in a local area network.



Bus network

Bus networks are easy to configure and expand. This makes them well suited in an expanding office environment. The bus can be made of any type of solid material, e.g., twisted pairs, coaxial cable, or optical fiber.

If you remember what is written about multidrop lines on page 29, you may see similarities. There are some important differences:

- Multidrop is most often used in wide area networks, a bus is never used as a link in a WAN.
- Multidrop is based on poll and select which is a centralized procedure. The communication is slow.

Access methods

The last point leads us to access methods. How does a single station access the bus? First of all there is the question of centralized control versus full distribution. With centralized control one station is defined as the boss which decides who can send when. In a fully distributed network each station can send whenever the link is ready to transport that station's message. Centralized control creates overhead. It is not typically used on a LAN bus.

CSMA/CD

One popular access method is called Carrier Sense Multiple Access with Collision Detection (CSMA/CD). This is described in the following paragraphs:

All stations can hear all other stations. In other words, a transmission is not physically directed. A station that wishes to transmit, monitors the cable to see if any of the other stations are sending.

If another station is sending, the station defers. Collisions can still occur if two stations simultaneously find the cable free and start transmitting at the same time.

To limit the time wasted during collisions, each sending station monitors the cable to detect interference from other stations (collisions). If a collision is detected, the sending station aborts its transmission and tries again at a later time.

Collisions can only occur during the 'collision window', which is the time it takes to transmit the first part of a frame. The size of the collision window is decided by the time it takes the signals to reach the most remote station from any given station that starts sending.

For reasons of network throughput, the collision window must be small compared to the average block transmission time.

With CSMA/CD we can have a fully distributed system. There is no need for central control.

ETHERNET

The Ethernet standard was published in the fall of 1980 in an attempt to establish an industry standard before everyone developed their own versions. This standard is a further development of the original Ethernet developed by Dec, Intel, and Xerox starting in 1974.

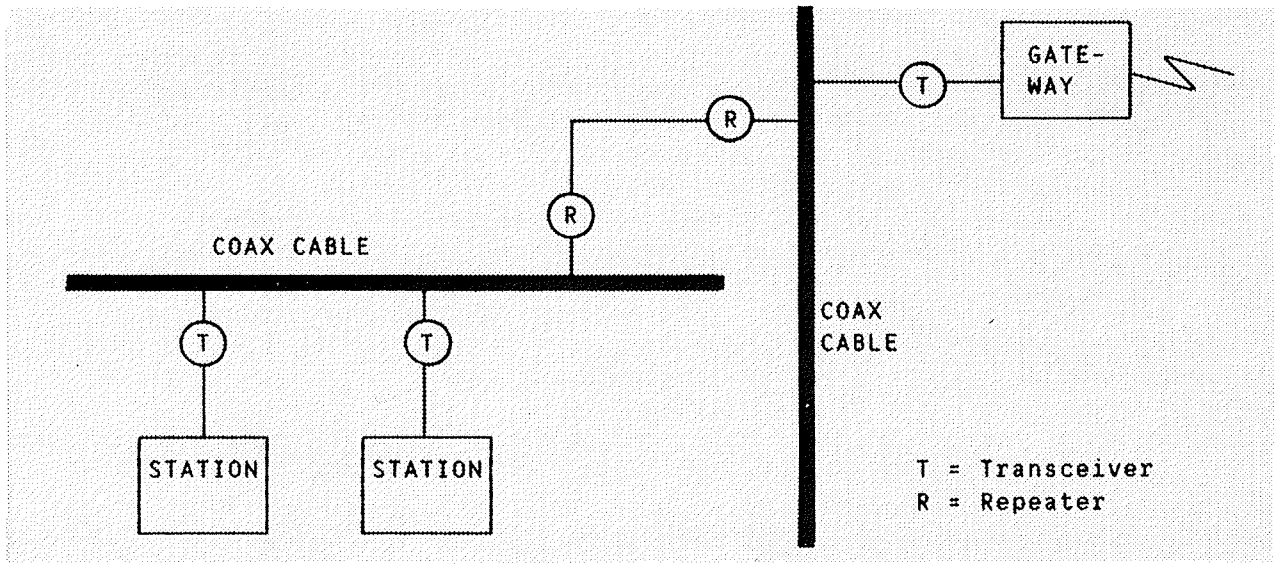
The standard is very rigid, and does not allow any choices or options. This was done on purpose to increase the possibility that products from different manufacturers would work together.

Very close to the Ethernet standard is IEEE 802.3 (the same as ISO DIS 8802.3) for CSMA/CD including mechanical and electrical specifications of the physical transmission. In fact, this standard is somtimes referred to as the "Ethernet standard". For further explanation on IEEE 802, please see page 112.

Ethernet is a bus LAN, and it consists of a number of different components. A coaxial cable is usually the transmission medium, but optical fiber may be used. Stations are pieces of equipment, connected to the cable, that send and receive data.

Components

The stations are normally connected to the cable by means of transceivers, but stations and transceivers may be integrated. Several coaxial cables may be linked together. For this we need repeaters to reduce attenuation of the signals. Gateways may be used to communicate outside of the LAN, via wide area networks.



Ethernet bus elements.

Cable access

The word "transceiver" is a synthesis of transmitter and receiver. In an Ethernet a CSMA/CD access method is used. Both carrier sense and collision detection is performed by the transceiver. The transceiver senses the carrier in the cable, which means that somebody is transmitting. The transceiver detects collisions by monitoring the electrical signal level in the cable.

Limitations

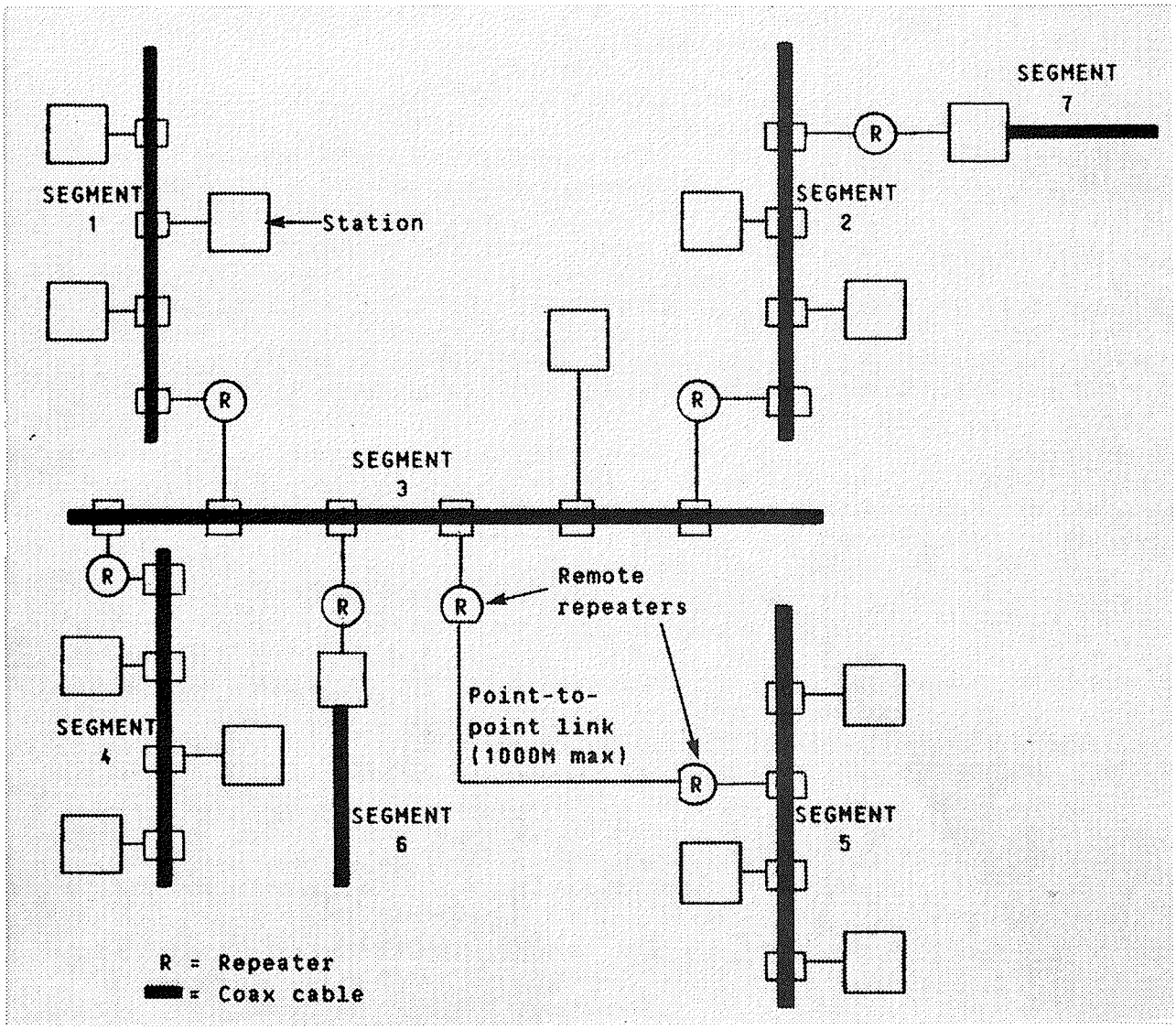
Here are some limitations of an Ethernet:

- Speed is 10 megabits per second. This is what the cable can do.
- There is a maximum of 5 cable segments between any stations in the network.
- Each cable segment can be maximum 500 meters.
- There is a maximum of 1024 stations on one net.
- The cable from the computer to the segment can be maximum 50 meters.

The maximum diameter of an Ethernet is 2.5 kilometers (5 cable segments of 500 m each). The reason for this is the CSMA/CD accessing scheme. You may remember that collisions can only occur during the collision window, and that the collision window is the time needed for the first transmitted signal to propagate to all the stations in the network. The collision window must be kept reasonably

small, because we do not want to waste too much time on collisions. Therefore a maximum distance between any two stations (the diameter) must be set.

It is also possible to include one local link of 1000 meters, provided the maximum diameter of 2.5 km. is not exceeded. This normally uses optical fiber as the transmission medium.



A typical large configuration

The Ethernet frame

A frame consists of a preamble that indicates the start of the frame, followed by two 48-bit addresses: the destination station address and the sending station address. Then comes the data part of between 48 and 1502 bytes of data, followed by a 32-bit cyclic checksum.

There is an interesting feature of the 48-bit address field. With 48 bits we can generate more addresses than there are people in this world today! So the potential is there that every person can have at least one work station, with a unique global address, hooked up to some Ethernet, and communicate with anyone else in the world through gateways.

ND'S ETHERNET BASIC SOFTWARE

This software package includes the Ethernet driver and the link Layer (Layer 2 according to the OSI-model). The product requires that you make your own applications, directly interfacing with the link level, or that you obtain this from a vendor.

This basic software is also interesting if you want to connect simpler equipment like laser printers, intelligent terminals or PCs directly to one segment.

COSMOS ETHERNET

Services provided by COSMOS can be utilized via Ethernet. Norsk Data's implementation of Ethernet follows the IEEE 802.3 (ISO DIS 8802.3) standard. The standard describes the CSMA/CD protocol.

Features

We would like to highlight the following Ethernet features:

- In Ethernet, both the length of a segment and the number of segments can be varied. This, along with the possibility of building up multi Ethernet configurations makes Ethernet quite flexible concerning the area covered. It is possible to have only one segment.

- By choosing an Ethernet solution you avoid the "cable spaghetti" which often occurs with point-to-point solutions.

- Ethernet gives very reliable communication. If one computer/node in the network goes down, no other computer connected to the network will be affected. The network as a whole will still function satisfactorily. A point-to-point network would not if a central node/computer went down. It is in fact possible to connect one computer physically to one segment of the Ethernet while traffic on the segment is running.

- Ethernet has advantages when the traffic consists of bursts of data, for example interactive processing. This is because a station may get control of the cable immediately upon demand. Types of LAN based on token passing or a slotted ring solution do not give this opportunity because every station has to wait for its turn. The types of LAN last mentioned are described in the next section.

- New connections may easily be added to an existing COSMOS Ethernet network without any changes in COSMOS. A large number (maximum 1024) may be added.

- Networks can be split into subnetworks of closely related computers/functions, providing you with so-called "departmental networks". These can handle a high rate of traffic without affecting other departments' networks. They are tied together via gateways to allow high speed communication between the different departments.

- The physical interface in the ND-computers attached to the Ethernet cable system, has a 68000 microprocessor. The interface currently has 128 Kbytes memory accessible from the main ND-100/500 and the 68000 microprocessor. This minimizes the load on the main CPU, because the 68000 microprocessor handles the different Ethernet protocols.

- Computer systems from different vendors may be connected to the same physical network, provided they support Ethernet communication. Until protocols are standardized above level 2 (we are

referring to the OSI model), they may co-exist, but real communication cannot take place unless they have common higher level protocols. As standardization breaks through at the higher levels, communication between systems from different vendors will increase.

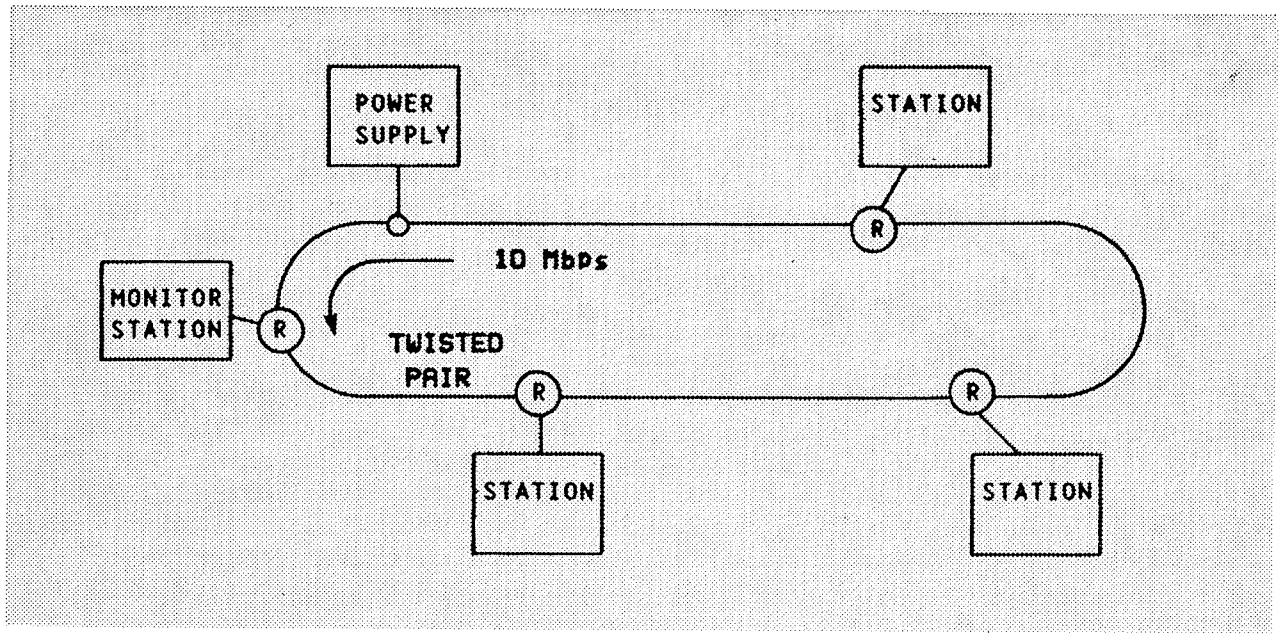
RING NETWORKS

If all the nodes in a network are connected so they build a ring, we call it a ring network. Each node receives data in one end and sends data out in the other end. This means that every node is a relay node. There also has to be a certain degree of centralized control in a ring network. For example someone has to decide which node has the right to transmit in case there is a conflict. Another control function has to do with taking a packet out of circulation if it goes around the ring forever and nobody perceives the packet as belonging to him.

Cambridge ring

We will use the Cambridge ring, which is a type of slotted ring, as an example of a ring network. In this type of network one and only one minipacket of fixed size circulates constantly. A special station called a monitor is used to generate and maintain the minipacket.

To improve reliability a "repeater" is isolated from the rest of the computer interface. This means that one computer may go down and the network still be functioning. The message on the ring is passed from repeater to repeater until it reaches the destination node.



Cambridge ring

The minipacket, which circulates on the ring, is marked as either empty or full. If the minipacket is empty, the receiving computer can fill it with a message, mark it as full, and fill in the source and destination addresses. The reserved minipacket then travels round to the other stations on the ring. The message is picked up by the destination node.

There are other ways of organizing a ring. One way is to use a token. The ring is then simply called a token ring. An explanation of the token concept is given in the next section.

TOKEN BUS

There are alternative methods to CSMA/CD concerning the regulation of access to a common bus. We will mention one of them, called token passing.

In a token passing system, control passes from one station to another in a controlled way. The sending station sends a token to another station when it gives away its right to transmit.

A 'logical ring' is constructed on top of a physical bus. In this way we can compute the

maximum waiting time for each station under all loads, if the order of transmission and the amount each station transmits is known.

The problem with token passing is token maintenance. The system works only as long as there is one token. To make sure that this is always the case, some kind of central control is required.

COMPUTERIZED BRANCH EXCHANGE

In the Computerized Branch Exchange (CBX) type LAN we use the local area telephone lines both for voice and for data. This gives us a star network with the CBX as the central node.

The CBX appears under some different names, for example:

- PABX : Private Automatic Branch eXchange
- PBX : Private Branch eXchange
- DPABX : Digital Private Automatic Branch eXchange

In general these systems offer lower data rates (64 Kbps). The advantage lies in saving cable costs by using existing telephone cables.

In Europe the public telephone companies control the use of local cables if the local lines are connected to the public telephone network. This control will considerably slow down the integration of voice and data in Europe.

BROADBAND LAN

The broadband LANs use a common broadcast medium that is divided into transmission bands each reserved for a special purpose. Broadband cables are cheaper than baseband and can cover greater distances. The most obvious bands are the 'voice', 'data' and 'image' bands.

Voice band

On the voice band, the local CBX can multiplex voice conversations either by frequency division or by using digital switching in the case of digital branch exchanges.

Data band

On the data band, data is sent as data packets in the same way as in the Ethernet type of networks. To achieve broadcast facilities on broadband networks, the stations generally transmit on one frequency band to the 'head end' of the cable.

At the 'head end', the signals are shifted to another frequency band and transmitted to all stations. The 'head end' functions as a radio transmitter.

In this way, two way amplifiers on the cable can be used. Also broadband LAN needs to find a solution to the contention problem, i.e., which station sends when.

Image band

The image band can be used for CATV (Community Antenna TeleVision), closed circuit television, or for cable television distribution.

The truly integrated networks are still very much in the planning stage and require an all digital approach for voice, data and image.

Broadband is the most common LAN type in USA, and baseband is the most common in Europe.

SUMMARY

The field of data communication has experienced a tremendous development during the last decade. The 'lower layer' functions are getting clearer, and practical and proven standards are becoming available.

General advances in computer technology have created large increases in information processing speeds and in storage capacities. This leads to greater demands for the movement of information as well. The people working on data communication face a challenge in the next decade to design and implement data communication systems that fulfill user needs at an acceptable price.

The general user needs communication facilities that are inexpensive, that can communicate anywhere and that are reliable.

CHAPTER 8
SHORT INTRODUCTION TO SOME IMPORTANT STANDARDS

OVERVIEW

The purpose of this chapter is to give you a brief overview of the most important data communication standards. It can be read independently of the rest of this manual. Many of the standards are described in more detail in previous chapters.

Many of the recommendations (e.g., OSI and X.25) are not officially called "standards", but they are actually treated as such both by vendors and in this document.

Organizations

Many organizations are concerned with standardization. The ones we will deal with now are:

- EIA (Electronic Industries Association, USA): We will mention the following standards: RS232C and RS422. They both have their international counterparts in CCITT recommendations (V- and X-series), but they are anyway often referred to in Europe.
- CCITT (International Consulative Committee for Telegraphs and Telephones). We are interested in two series of recommendations, the V-series and the X-series. The V-series relates to data transmission over analog networks, and the X-series deals with public digital data networks.
- ISO (International Standards Organization) has made the OSI (Open System Interconnection) reference model, which is a 7 layer model for data communications architecture. ISO is also involved in local area network (LAN) standards.
- ECMA (European Computer Manufacturers Association) is making various standards for local area networks and also protocol standards for some OSI layers.
- IEEE (Institute of Electrical and Electronics Engineers, USA) is the most active organization making LAN standards. Documents coming from the project IEEE 802 are often referred to as authoritative sources.

THE V-SERIES

The most well known and probably the most important standards in the V-series that a nonexpert should know about are V.24 and V.28.

V.24

V.24 describes the functional characteristics of the interface between DTE (Data Terminal Equipment) and DCE (Data Communications Equipment) at the physical level. The DTE is either a computer or a terminal. Since the V-series deals with transmission across analog carriers (like telephone lines), the DCE is a modem.

The function of the modem is conversion between digital and analog signals. Voice data, which is the only data type travelling across a telephone line is coded as analog signals.

V.24 contains conventions for:

- Establishing contact between the DTE and the transmission line (via modem)
- Sending and receiving data as a serial bit stream
- Disconnecting the transmission

V.24 may therefore be considered a communications protocol at the physical level (bit level).

RS232C

The American RS232C is roughly equivalent to V.24 + V.28.

V.28

V.28 describes the electrical characteristics (e.g., voltage levels) of the V.24 signals.

THE X.21 RECOMMENDATION

Like V.24, X.21 also describes conventions for establishing and disconnecting the communication between DTE and DCE at the physical level. X.21, however, relates to a DCE pertaining to a circuit switched public data network. Such a network is based on digital transmission and is designed for

transport of data rather than transport of voice. The network is usually operated by some public telecommunications organization.

A circuit switched public data network functions very much like a telephone network from the computer's point of view. DTE A has to "dial" DTE B, using the X.21 conventions.

RECOMMENDATIONS RELATED TO X.21

If the DTE does not have an interface card obeying the X.21 standard but only a V.24 card, it may still connect to a circuit switched public data network. In this case we need a special DCE which can convert from V.24 signals to X.21 signals. This interface between DTE and DCE is called X.21 bis.

X.21 bis

X.20

The above mentioned data network operates synchronously, which means that the DTE has to transmit blocks of data in a synchronous way. In case the DTE is an asynchronous terminal sending characters intermittently, some different interface conventions have to be used as well as a different type of DCE. The recommendation is called X.20.

X.27 = RS422

To define the electrical characteristics of the X.21 signals, X.27 is used. This X.27 recommendation is the same as the American RS422.

THE X.25 RECOMMENDATION

X.25 is an interface between a DTE and a DCE in a packet switched public data network. Unlike the previously mentioned standards X.25 does not relate to the physical transmission of bits. X.25 describes a set of rules for exchange of data packets between DTE and DCE.

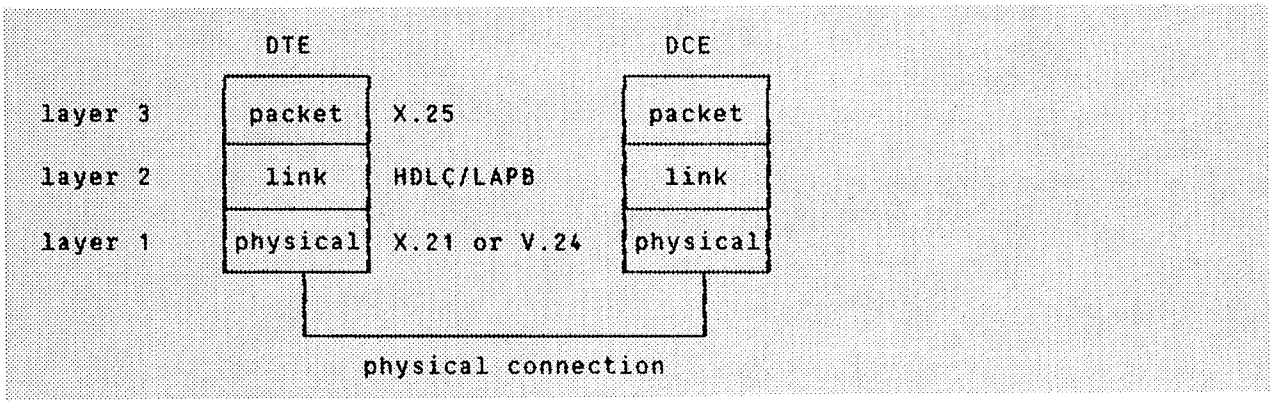
Some of the rules have to do with the packet formats, other rules describe how certain control packets can be used for flow control, congestion control, error detection, recovery etc.

Packet switched network

A packet switched network is based on the transport of packets which include destination addresses, by sending the packet one step at a time from station to station. Such a station, which resides entirely within the public data network, is called a node and

is dedicated to data transport. The node is a computer which analyzes the packet header when a packet is received and finds out which route to take through the network, i.e., which node to forward the packet to.

The X.25 recommendation specifies 3 levels: Packet layer, link layer and physical layer.



The illustration shows how the X.25 interface has to do with a logical (on level 3) rather than a physical connection between DTE and DCE. The DCE is a node in the network.

The link level

The X.25 recommendation says that the protocol at the link level may be either LAP (Link Access Protocol) or LAPB (Link Access Protocol Balanced), but LAPB is preferred. Both of these link protocols are varieties of HDLC (High level Data Link Control). One of the purposes of the link protocol is to hide the weaknesses of the physical connection from the layer above. This is done for instance by automatic error detection (errors in the physical transmission may occur due to noise on the transmission medium) and retransmission of bad frames.

The physical level

The X.25 recommendation says that on the physical level (layer 1) we may choose between a V.24 or an X.21 interface.

RECOMMENDATIONS RELATED TO X.25

As the X.25 recommendation relates to synchronous networks, other interfaces have been made in order to accomodate asynchronous terminals.

PAD

The characters transmitted intermittently from the asynchronous terminal must be assembled into data packets by a PAD (Packet Assembly Disassembly) before it can go to the packet network DCE. The PAD may be a hardware box or it may be a piece of software residing in a computer which the asynchronous terminal is directly connected to.

X.3/X.28/X.29

X.3 is the name of the standard for assembly and disassembly of packets, but in addition to this we need an interface between the PAD and the asynchronous terminal (X.28) and a protocol between the PAD and the packet oriented DTE which the asynchronous terminal is ultimately communicating with (this last protocol standard is called X.29).

X.3, X.28, and X.29 comprise what is called "triple x".

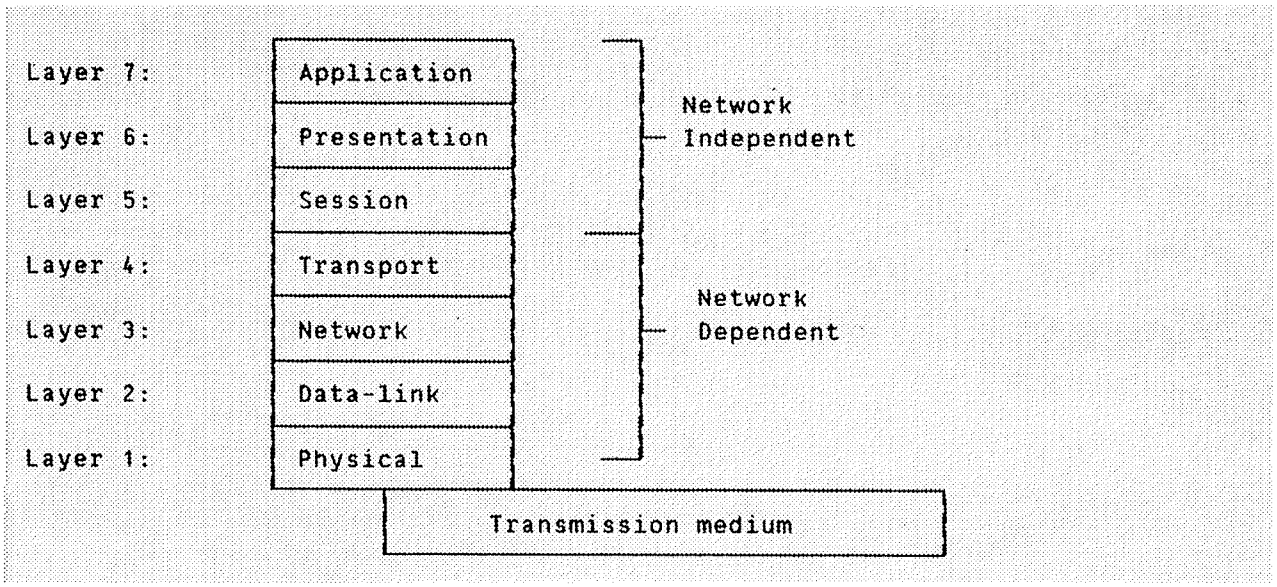
THE OSI REFERENCE MODEL

Computers from different manufacturers are very often compatible at the physical level. They may for example use hardware interfaces conforming to the V.24 recommendation. If we are lucky they may even use the same link protocol, for instance LAPB, and even follow the X.25 recommendation on level 3. We will in all probability still be faced with compatibility problems as the record formats may be entirely different, which is quite fatal during a file transfer. The alphabets may be different, the computers may have different ideas about who is going to "talk" when (simultaneously or alternately, who may initiate, who may terminate and under which circumstances) etc.

The seven layer reference model OSI provides a tool for achieving compatibility in architecture all the way up to the application. This does not mean that we have two fully compatible systems if they only follow OSI. There is in fact quite a bit of room for divergent implementations within the

OSI framework, but standardizing the framework is in itself a step in the right direction.

The layers of OSI are illustrated in the following picture:



Most of the OSI recommendation is described in rather abstract terms. Although the authors have had X.25, HDLC, X.21 and V.24 in mind, the OSI does not specify anything concrete which means that the above mentioned standards do not necessarily have to be adhered to in order to satisfy the OSI reference model. As we move upwards in the hierarchy, the contents of the layers become increasingly vague. Most implementations so far deal with the bottommost layers.

Assuming that you understand the general idea behind layers 1 and 2 from the previous sections, we will now describe each of the other layers starting with number 3.

Layer 3

The network layer has to do with the routing of messages and network address administration. As we compare the network layer with layer 3 under X.25, we see that in the latter case they talk about a "packet layer".

This packet layer would be a permissible implementation of the OSI network layer if we add something to the packet layer. Since X.25 is only concerned with the interface between DTE and DCE it says nothing about routing internally in the public data network.

Neither does it mention connections across networks. To access a database in California from a terminal in Oslo, we may have to go through both the Norwegian DATAPAK and the American TYMNET packet switched public networks. The X.75 recommendation could be used for implementing intra- and inter networking at OSI level 3.

Layer 4

The task of the transport layer is to complete the end-to-end transport of messages. If the underlying network layer gives thorough service, the transport layer has to do less than if the network layer gives less thorough service.

The transport layer completes the transport medium, delivering messages or message fragments on the "doormat" of the layer above, namely the session layer. A transport protocol has been developed by ECMA and is called ECMA72.

Layer 5

The purpose of the session layer is to provide a session connection between the two communicating parties, transfer messages by using the services of the transport layer, delivering complete data units (which may have been received as message fragments from the transport layer) to the presentation layer, and managing the interaction between the communicating systems. ECMA has a session protocol standard, ECMA75.

Layer 6

The presentation layer provides format conversion, code conversion (e.g., EBCDIC to ASCII) and other things related to how the messages are presented (on a screen, in a file, etc.).

Layer 7

The application layer provides the actual application with an interface to the communications part of the system. This means that the application, whether that is a user program or functions performed by a terminal user, is separate from layer 7 and logcially resides on top of it.

What exactly layer 7 may perform is not yet clear, but one thing that has been suggested is the distribution of operating system functions. If we perceive a network of computers as one system with distributed functions, it may be feasible to have different parts of the operating system to run in different computers rather than having all the parts duplicated in every computer.

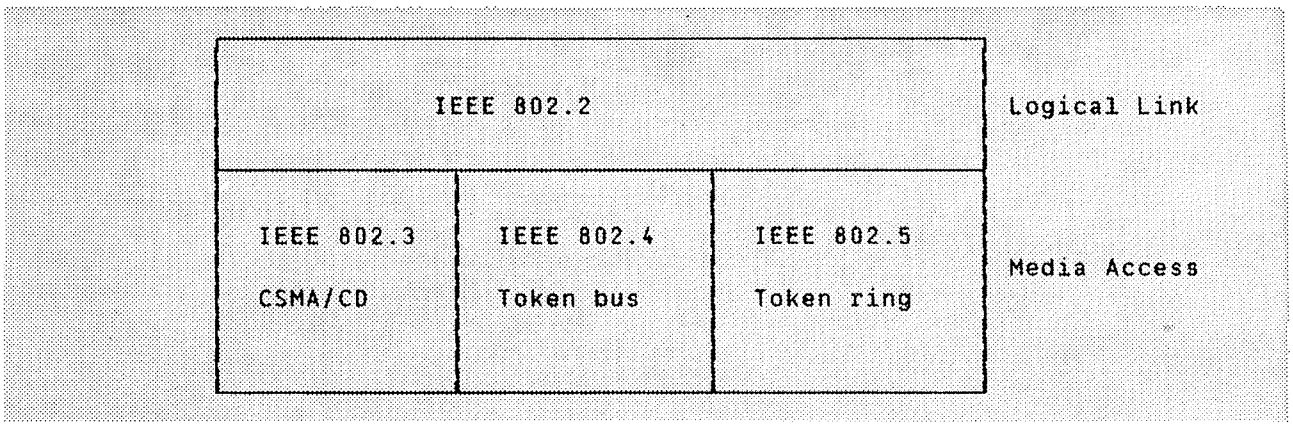
LOCAL AREA NETWORKS

Standardization work within this field is relatively new. One of the first efforts was some work done by the manufacturers DEC, Intel and Xerox to standardize Ethernet. The physical transmission channel in an Ethernet is basically one coax cable which all the work stations tap into (a multiaccess cable).

The Ethernet standard, which is more or less followed by several LAN vendors, covers layer 1 and most of layer 2 (referring to the OSI model). ECMA has proposed standards close to the Ethernet "standard" for layer 1 and 2. The ECMA72 is proposed as a LAN standard for layer 4. IEEE is the most active organization making LAN standards, and is influencing the work of ECMA and ISO.

IEEE has made standards which cover OSI layers 1 and 2. These IEEE 802 standards include CSMA/CD (often referred to as "Ethernet"), token bus, and token ring.

Without following the OSI division, IEEE 802 uses two sublayers: logical link and media access. The same logical link sublayer is used by any of the media access implementations. If somebody wants to change from CSMA/CD to token bus, they only have to change the media access sublayer from IEEE 802.3 to IEEE 802.4.



INDEX

Index

PRINTING RECORD	
PRINTING	NOTES
	Version 1 not published
08.85	Version 2

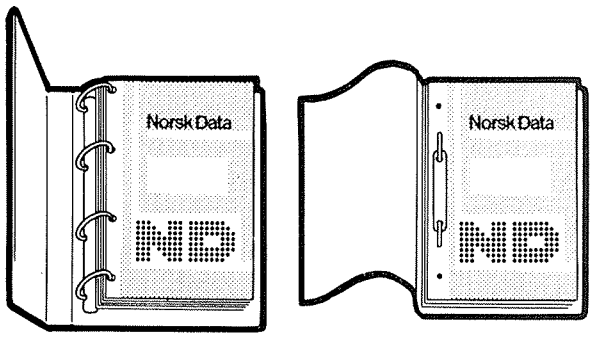
Manual Name: Introduction to DATA COMMUNICATION
Manual No.: ND-60.181.2 EN
Date: 08.85

UPDATING

Manuals can be updated in two ways, new versions and revisions. New versions consist of a completely new manual which replaces the old one, and incorporate all revisions since the previous version. Revisions consist of one or more single pages to be merged into the manual by the user, each revised page being listed on the new printing record sent out with the revision. The old printing record should be replaced by the new one.

New versions and revisions are announced in the ND Customer Support Information and can be ordered from the address below.

The reader's comments form at the back of this manual can be used both to report errors in the manual and to give an evaluation of the manual. Both detailed and general comments are welcome.



RING BINDER OR PLASTIC COVER

The manual can be placed in a ring binder for greater protection and convenience of use. Ring binders may be ordered in two widths, 30 mm and 40 mm.

The manual may also be placed in a plastic cover. This cover is more suitable for manuals of less than 100 pages than for larger manuals.

Please send your order, as well as all types of inquiries and requensts for documentation to the local ND office, or (in Norway) to:

Norsk Data A.S
Graphic Center
P.O. Box 25 BOGERUD
N - 0621 OSLO 6 - Norway

I would like to order

. . . . . Ring Binders, 30 mm, at NOK 20.- per binder

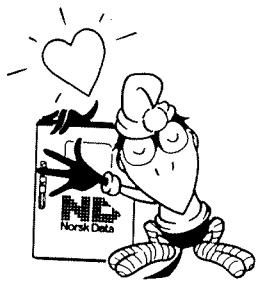
. . . . . Ring Binders, 40 mm, at NOK 25.- per binder

. . . . . Plastic Covers, at NOK 10.- per cover

Name: . . . . . .

Company: . . . . . .

Address: . . . . . .



SEND US YOUR COMMENTS!

Are you frustrated because of unclear information in our manuals? Do you have trouble finding things? Why don't you join the Reader's Club and send us a note? You will receive a membership card – and an answer to your comments.

Please let us know if you:
– find errors
– cannot understand information
– cannot find information
– find needless information.

Do you think we could improve our manuals by rearranging the contents? You could also tell us if you like the manual.

Send to:
Norsk Data A.S
Documentation Department
P.O. Box 25 BOGERUD
N - 0621 OSLO 6 - Norway

NOTE!

This form is primarily for documentation errors. Software and system errors should be reported on Customer System Reports.

Manual Name: Introduction to DATA COMMUNICATION Manual number: ND-60.181.2 EN

Which version of the product are you using? _____

What problems do you have? (use extra pages if needed) _____

_____

_____

_____

_____

_____

Do you have suggestions for improving this manual? _____

_____

_____

_____

_____

_____

_____

Your name: _____ Date: _____

Company: _____ Position: _____

Address: _____

_____

What are you using this manual for? _____

_____

Norsk Data's answer will be found on the reverse side.

Answer from Norsk Data: _____

Answered by: _____ Date: _____

Norsk Data A.S
Documentation Department
P.O. Box 25 BOGERUD
N - 0621 OSLO 6 - Norway

Systems that put people first

NORSK DATA A.S OLAF HELSETS VEI 5 P.O. BOX 25 BOGERUD 0621 OSLO 6 NORWAY
TEL.: 02 - 29 54 00 - TELEX: 18284 NDN