A Appendix: Tutorial on Networks Concepts

A.1 Introduction

This is a short tutorial on network concepts. It explains the basic concepts of physical communication concepts, digital communication concepts, Local and wide area networks, enterprise networks and network architectures, and basic Internet Concepts. The tutorial is intended to introduce basic network concepts that are useful in understanding the wireless networks.

This tutorial material has been extracted from the following book:

> Umar, A., e-Business and Distributed Systems Handbook: Network Module, NGE Solutions, 2003

A very large number of books and magazines cover physical networking topics. The sidebar lists a few.

Additional Sources of Information

Books

Berkowitz, H., “Designing Routing and Switching Architectures for Enterprise Networks”, Macmillan Technical Publishing, 1999.

Clayton,, J., "McGraw-Hill Illustrated Telecom Dictionary (McGraw-Hill Telecommunications)", McGraw Hill, 1998.

Dodd, A, "The Essential Guide to Telecommunications", Prentice Hall, 2nd edition, 1999.

Felt, S., “Wide Area High Speed Networks”, Cisco Systems, 1999.

Marcus, S., “Designing Wide Area Networks and Internetworks: A Practical Guide”, Addison Wesley, 1999.

McCabe, J., “ Practical Computer Network Analysis and Design”, Morgan Kaufmann, 1999.

Muller, N., "Desktop Encyclopedia of Telecommunications (McGraw-Hill Telecommunications)", by McGrawa Hill, 1998.

Muller, N., "Desktop Encyclopedia of Voice and Data Networking", McGraw Hill, 1999.

Kessler, G., Southwick, P., “ISDN Concepts, Facilities, and Services”, McGraw-Hill, 1998.

Newton, H., “Newton's Telecom Dictionary”, 16th Ed, Telecom Books/Miller Freeman, 2000.

Panko, R., "Business Data Networking and Telecommunications", (4th edition), Prentice-Hall, 2003.

Parnell, T., "Building High-Speed Networks", Network Professional's Library, McGraw Hill, 1999.

Perlman, R., “Interconnections: Bridges, Routers, Switches, and Internetworking Protocols”, Second edition,(Addison Wesley Professional Computing Series, 1999.

Stallings, W., " Business Data Communications", 4/E, Prentice Hall, 2001.

Stallings, W., " Local and Metropolitan Area Networks", 6/E , Prentice Hall, 2000.

Stallings, W., " Data & Computer Communications", 6/E, Prentice Hall, 2000.

Stallings, W., " ISDN and Broadband ISDN with Frame Relay and ATM", 4/E, Prentice Hall, 1999.

Tannenbaum, A., "Computing Networks", Prentice Hall, 3/E., 1996.

Ward, A., "Connecting to the Internet: A Practical Guide About LAN-Internet Connectivity", Addison-Wesley, 1999.

Magazines and Journals

For state of the art issues, the following magazines are recommended:

- Computer Networks and ISDN Systems
- Computer Communications
- IEEE Network Magazine
- IEEE Communications Magazine
- IEEE Transactions on Communications

For a state of the market and practice evolution, the following trade journals are recommended:

- Data Communications
- Business Communications Review
- Communications News
- LAN Technology
- LAN Magazine
- Byte Magazine (special issues on LANs and communications)
- PC Magazine (special issues on LANs and communications)

A.2 Physical Communication Characteristics

A.2.1 Overview

A ***communication network*** is a collection of equipment and physical media, viewed as one autonomous whole, that interconnects two or more stations. At the lowest level, a communication network consists of three components (see Figure A-1):

- **Stations (data sources/sinks).** These entities generate and receive the data handled by the communication network. Examples of the data are voice, computer bits, and TV patterns. A station is effectively an end-point (source/sink) in a communication network. Examples of stations are terminals (text and/or graphics), telephones, sensors (temperature, security), TVs, facsimiles, diskless workstations, personal computers, workstations, minicomputers, or mainframes.
- **Data/signal converters**. These devices convert the data to signals for transmission on one end and back to data at the other. Data is propagated from one point to another by means of signals which are electromagnetic representation of data. An example of a converter is a modem which converts data bits to continuous signals which are transmitted across a network. Converters basically translate different formats of data and signals (modems are digital to analog converters and codecs are analog to digital converters). In some networks, called baseband networks, the conversions are bypassed by "pressing" the data directly against the communication wire. In these cases, data and signals are the same.
- **Transmitting facilities**. These facilities deliver (transport) the signals across a network. This transport involves finding a path for the signals, sending the signals over the path, and dealing with signal attenuation and distortion over long transmission paths.The transmission facilities themselves consist of:

- **Links (communication links)** refer to the physical media that is used to interconnect stations in a communication network. Examples of links are telephone lines, coaxial cables, and fiber cables.
- **Intermediate systems**, also sometime referred to as nodes, serve as intermediaries in a communication network. Example of a node is a router which is used to direct traffic from one point to another .

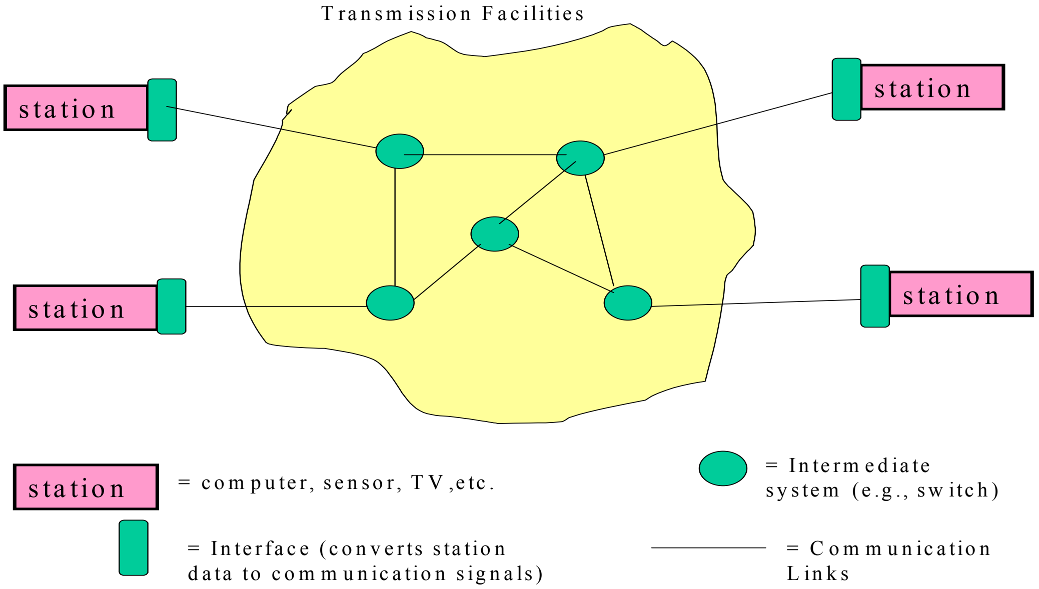

Transmission Facilities
station
station
station
station
station
= computer, sensor, TV,etc.
= Intermediate system (e.g., switch)
= Interface (converts station data to communication signals)
= Communication Links

Figure A-1: Conceptual View of Networks

A communication network may use analog (continuous) or digital (discrete) data for its inputs and outputs. In digital communication systems, all data is digital (it either comes from digital data sources or is converted to digital format). These systems are increasingly being used in most of the current and future distributed applications.

The transmission facilities consist of a wide range of hardware components and software modules. Design of transmission facilities raises questions such as what communication medium to use, how to interconnect the various components (network topology), what methods to use for communication between various components, and what techniques to use for data compression and encryption. .

Network facilities are generally classified into three categories based on the geographical area covered

- **Local area networks (LANs)** which do not use common carrier facilities over short distances with speeds up to 100 Mbps. LANs are commonly used to interconnect computers within the same building and organization.
- **Wide area networks (WANs)** which use common carrier facilities over long distances commonly with speeds in the range of 1.5 Mbps. WANs are used to interconnect remotely located sites and equipment.
- **Metropolitan area networks (MANs)** are essentially large LANs which cover an entire metropolitan area (a city, a suburb, etc.). The evolving metropolitan area networks can be used to interconnect LANs within a metropolitan area.

Communication networks needed for the physical transmission and recognition of data between interconnected devices are moving toward digital communication systems. These systems receive digital data which is regenerated over long distances by eliminating noise. A hierarchy of digital services, known as the T carriers are widely in use. Digital communications are increasingly being used for transmission of data, voice and video in communication systems. ISDN (integrated system digital network) is an example.

A.2.2 Basic Terms and Definitions

As mentioned previously, data conveys meaning to a user and a signal is an encoding of data in some electromagnetic format. Signals are represented as cyclic waves which may be discrete (digital) or continuous (analog). The following properties characterize cyclic waves (see Figure A-2):

- Amplitude which shows the height of the wave
- Frequency which shows the cycles per unit time of the wave
- Phase which shows how far, in degrees, the wave is from its beginning (phase 0).

Data can be digital or analog; signals can also be digital or analog. Figure A-3 shows the techniques used to convert (encode) data into signals. The technique employed depends on the format of data (analog or digital) and the encoded signal (analog or digital). Figure A-3 shows that telephones are used to encode analog data to analog signals and digital transmitters are used to encode digital data to digital signals. These conversion can be simple (i.e., a 0 data bit appears as no voltage and a 1 appears as some voltage) or sophisticated (see, for example [Stallings 1991]). But how are the digital data bits transmitted by using the analog waves? As shown in Figure A-3, modems are used to modulate (convert) digital data to analog signals and then demodulate them back to digital data. We will discuss modems in Section A.2.3. Similarly, a codec (coder-decoder) is used to convert analog data to digital signals. Codecs are described in Section A.3.2.

After conversion, how are the signals transmitted simultaneously over the same medium? In 1874, Baudot, a French scientist, showed that six users could transmit simultaneously on one wire. This was done by sending information at six different frequencies for the six users and by using filters at the receiving end which only received unique frequency (see Figure A-4a). This experiment is the foundation of what is currently known as frequency division multiplexing (FDM) and is used very widely in communication systems of today. The six sessions are referred to as the six channels which operate on the same wire. The filters allow the users to "tune" to a particular channel. Notice the similarity of this example to the modern television sets with multiple channels, where each channel represents a frequency range.

The following communication terms are used widely:

- ***Hertz (Hz)*** = number of cycles per second
- ***Baud*** = number of signal changes per second
- ***Data rate*** = number of bits sent per second (bps). Data rate is equal to the baud rate if one bit is carried per signal. In general, data rate = baud rate x data bits per signal.
- ***Channel*** = a logical communication path
- ***Bandwidth*** = frequency used by a signal, measured in Hz.
- ***Channel capacity*** = number of bits that can be transmitted per second. This is the same as data rate.

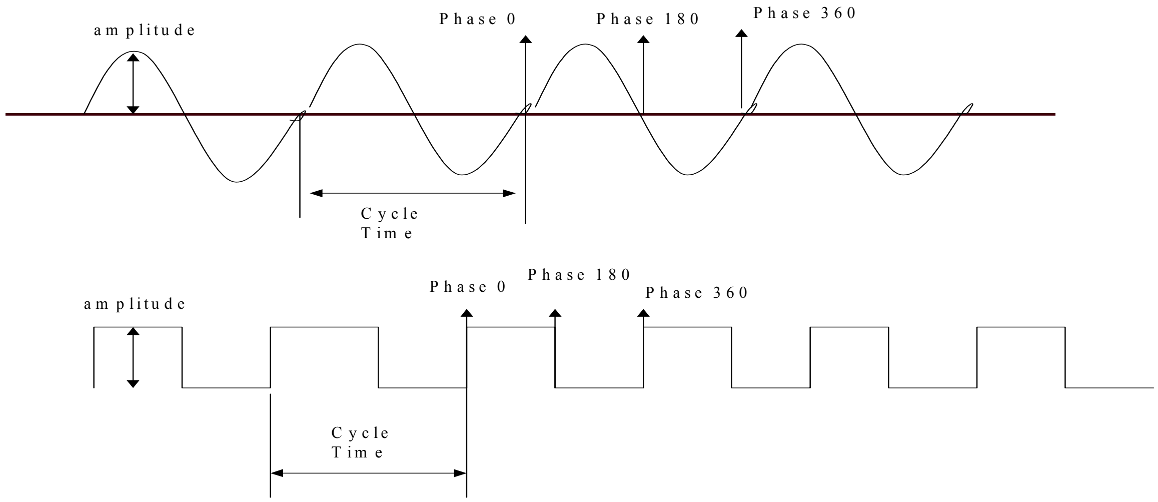


Figure A-2: Cycle Wave Characteristics

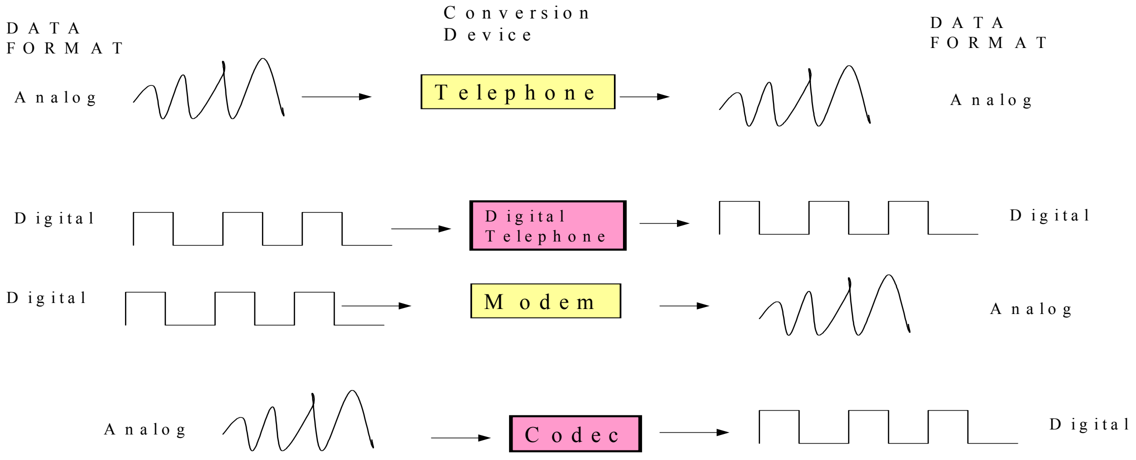
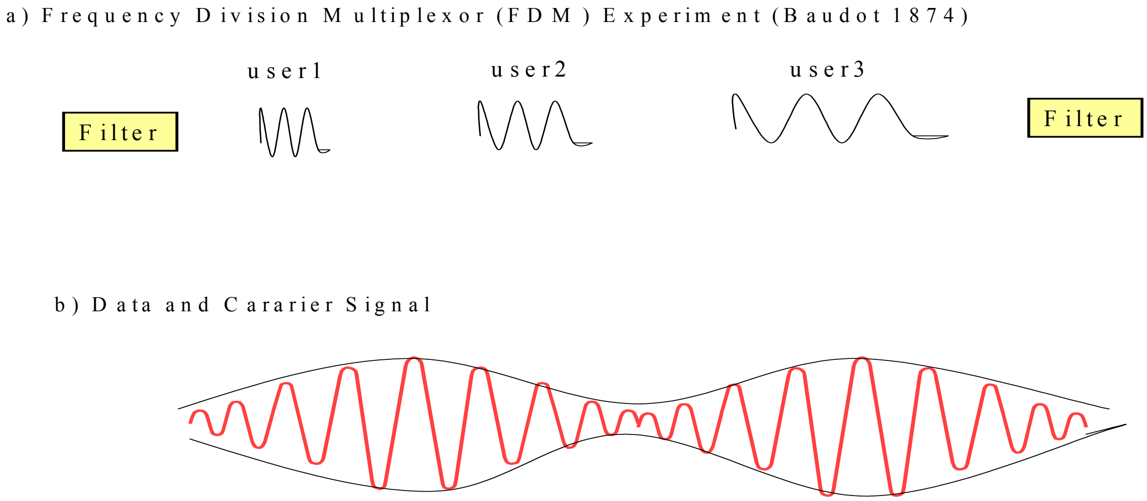


Figure A-3: Examples of Data to Signal Conversions



a) Frequency Division Multiplexor (FDM) Experiment (Baudot 1874)

b) Data and Cararier Signal

Figure A-4: Frequency Division Multiplexing and Data/Carrier Signals

Let us consider an example to illustrate the basic ideas. Human voice uses frequency ranges from 0 to 4000 Hz, but the commonly available voice graded telephone lines (carriers) use the frequency ranges 300 to 3300 Hz to carry a human voice. This bandwidth of 3000 Hz has been found adequate for human conversations. Frequencies lower than 300 and higher than 3300 are not carried well by voice graded lines (this is why the sound of delicate musical instruments is distorted on telephones). If a cable (carrier line) can carry between 30,000 to 42,000 Hz, then it can roughly support the following four voice channels (voice graded lines):

- Channel 1: 30,000 to 33,000 Hz
- Channel 2: 33,000 to 36,000 Hz
- Channel 3: 36,000 to 39,000 Hz
- Channel 4: 39,000 to 42,000 Hz.

At this point we should differentiate between data and carrier signals. A data signal represents the data and a carrier signal represents the signals of a medium (e.g., a telephone line) on which the data is carried. For the example just discussed, the data signals represent the voice and the carrier signals are in the range of 30,000 to 42,000 Hz. Note that the same data signal at various carrier frequencies represents the same information (e.g., data signals at channel 1 and 4 are the same). This is illustrated in Figure A-4b. This principle of dividing up the bandwidth of a carrier into data signal bandwidths, called frequency division multiplexing (FDM), can be used for a rough estimate of the minimum number of users of a cable. For example, if a cable has a BW of 2 MHz then it can support approximately 666 voice channels (telephone lines) by using the following simple formula:

Number of users on a carrier = carrier bandwidth/data bandwidth

Then with a carrier bandwidth of 2 MHz and data bandwidth of 3000 Hz, we obtain 666 users. Some modifications to this formula are needed to allow for "guard bands" (some room left between consecutive users) and for other multiplexing techniques (e.g., the time division multiplexing used in digital communications) which are more efficient than FDM. However, this gives us a gross estimate which may be useful for rough paper and pencil analysis.

We should note that in many cases, data and signals are used as synonyms because a signal is just a representation of data. The distinction is usually necessary for detailed communication engineering design, which is beyond the scope of this book.

A.2.3 Digital to Analog Conversion: Modems and Interfacing Devices

A ***modem*** (modulator/demodulator) is a hardware device which converts digital data to analog signals and vice versa. A modem is also called ***data circuit equipment (DCE)*** or "dataset" in communication systems because it interfaces between voice and data communication systems. Common modulation and demodulation techniques used in modems are (Figure A-5):

- ***Amplitude Modulations (AM)***, where the 0 and 1 bits are represented by the height of an amplitude. For example, 5 volts may be used to represent bit 0 and 10 volts may be used to represent bit 1.
- ***Frequency Modulation (FM),*** where the 0 and 1 bits are represented by two different frequencies, say 1000 cycles per second and 2000 cycles per second, respectively.
- ***Phase Shift Modulation (PSM),*** where a certain phase (e.g., 90) is used to represent bit 0 and a change to bit 1 is indicated whenever the phase of the signal changes.

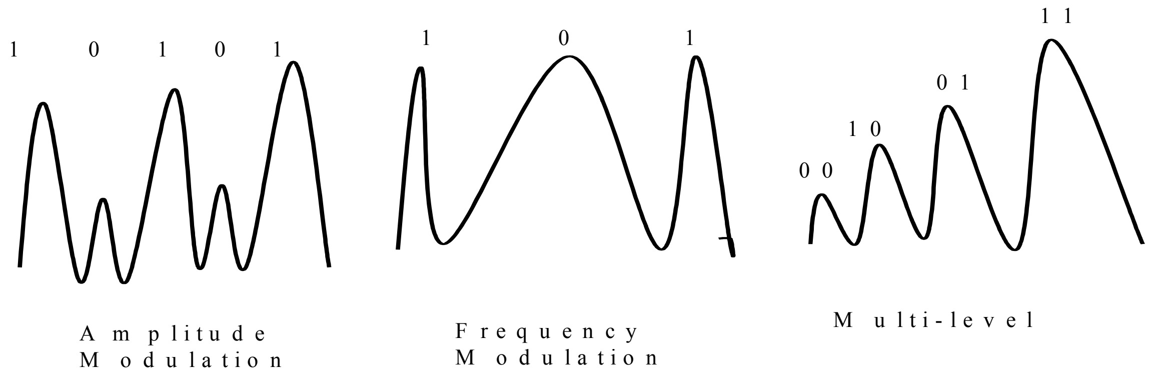


Figure A-5: Signal Modulation/Demodulation

If one signal level carries more than one data bit, as shown in Figure A-5, then the modulation is called ***multilevel modulation***. This technique is used in modems to increase communication speed. For example, many modems have a switch which can be used to increase the data rate from 4800 bps (bits per second) to 9600 bps. This switch essentially starts sending two bits per signal instead of one, thus doubling the bits per second.

An old but still very popular interface standard between modems and terminals (called data terminating equipment - DTE) is the ***EIA RS232-C*** standard. This standard, commonly referred to as the RS232 standard, was developed when computing equipment started using the telephone lines for data transmission. There was a natural need to develop a standard so that the digital computer devices could be connected to the telephone lines through modems. RS232 is officially limited to 20 Kbps for a maximum of 50 feet between the devices. In practice, many installations have used RS232 interfaces at 400 feet and higher distances. RS232-C interface standard is used heavily in the U.S to connect many type of devices (printers, terminals, modems, etc.). It uses 25 pins, out of which a few (6 to 12) are used heavily. Figure A-6 shows the most frequently used pins of RS232.

In most computing devices, RS232 is implemented in a microprocessor chip, known as ***UART (Universal Asynchronous Receiver Transmitter)***, which can be programmed to send/receive data. For example, IBM PCs contain a UART, which is actually an 8250 microprocessor chip. This chip provides transmit and receive registers for data transmission. In addition, it provides control registers for choosing line and modem options (e.g., data rate, stop bits), and status registers to see if anything has been received, etc. The IBM Personal Computer Technical Reference Manual (PN-636453) describes the programming details about how these registers can be accessed. Programs can be written in BASIC, C, Pascal and 8080 assembler to drive the RS232 interfaces. For example, many terminal emulation programs such as Kermit program the 8250 chip to send/receive data on serial communication lines.

In addition to the RS232 interface, many other interface standards have been developed. An example is the EIA RS449 which supports 2 Mbps at a 200 foot distance. It uses 37 pins and is compatible with the International Standards Organization (ISO) standards. RS449 is also a federal government requirement for many equipments and is favored by some network architectures such as the Manufacturing Automation Protocol (MAP). Many adapters between RS232 and RS449 have been developed and are commercially available.

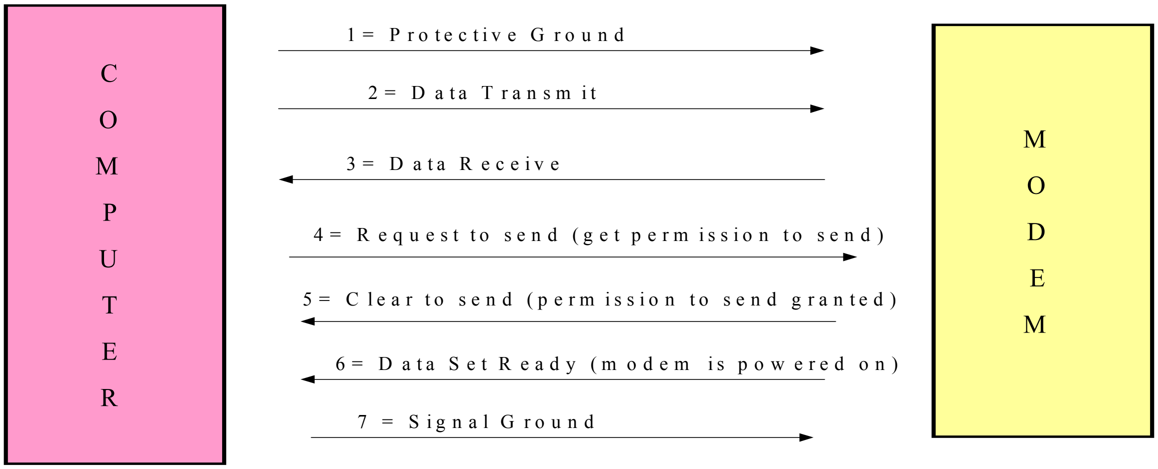

COMPUTER
1 = Protective Ground
2 = Data Transmit
3 = Data Receive
4 = Request to send (get permission to send)
5 = Clear to send (permission to send granted)
6 = Data Set Ready (modem is powered on)
7 = Signal Ground
MODEM

Figure A-6: RS232 Main Pins

A.3 Digital Communication Networks

A.3.1 Analog Versus Digital Communications

Conceptually, a communication network is analog or digital depending on

- Data received/generated by the transmission facility, and
- Techniques used to handle attenuation over long distances.

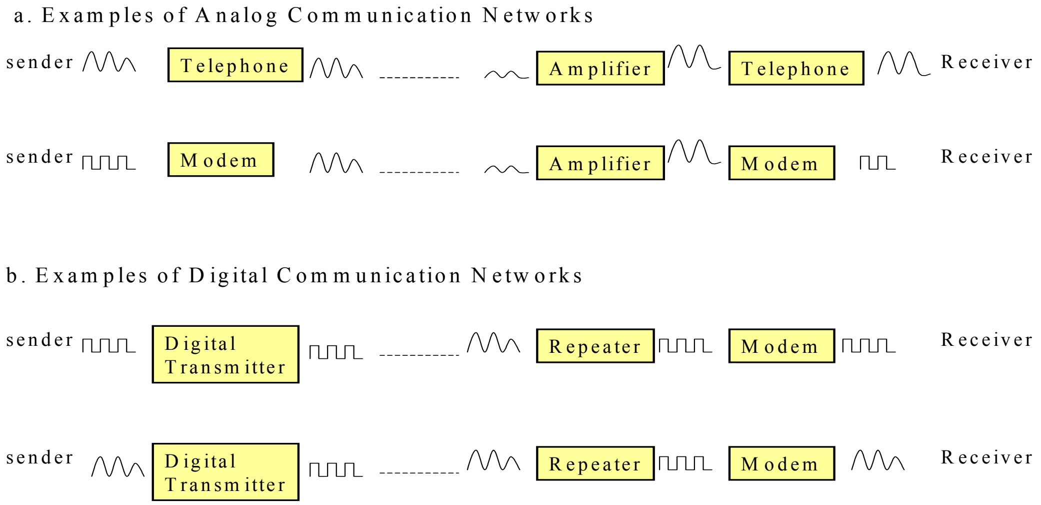


Figure A-7: Digital versus Analog Communications

Analog communication networks receive/generate analog data and use amplifiers to handle attenuation (see Figure A-7a). The analog data is either generated directly by a source (e.g., a human) or is converted to analog by modems. The main problem with analog communication networks, as shown in Figure A-7a, is that the amplifiers do not know the content of the inputs; they amplify whatever is received, including the noise.

In a digital communication network, the data received/generated by the transmission facility is digital and repeaters are used in the transmission facility over long distances to recover the patterns of 1's and 0's (see Figure A-7b). The digital data may originate from a digital device such as a computer or it may be digitized before being fed into the transmission facility. Repeaters assume that the input is digital; thus when the signal is attenuated, the repeaters regenerate the original bit patterns (see Figure A-7b). Thus they filter the noise. However, repeaters cannot be used for analog data.

The communications industry is evolving toward the use of digital communication networks. These networks carry digital images of computer bits, voice, video, facsimile, graphics, and many other type of data. Digital communication networks are more attractive for several reasons:

- Digits are more rugged and free of noise because it is easier to detect 1's and 0's even in distorted messages. For example, if amplitudes of 10 volts and 5 volts are used to indicate 1's and 0's, respectively, then a receiver can detect a 1 if the amplitude is greater than 5 and a 0 for 5 volts or less. The analog data, once distorted, cannot be recovered.
- Repeaters along a transmission path can detect a digital signal and retransmit a clean (noise-free) signal. These repeaters prevent accumulation of noise along the transmission path. In contrast, if a distorted analog signal is amplified then the distortion is also amplified.
- Digital communication is especially suitable for computer networks because data bits can be directly fed into a communication medium without any modulation/demodulation (this is the idea behind the baseband local area networks which we will discuss later).
- A single medium (e.g., a cable) can multiplex voice, data and video because they all appear as bits. This allows an organization to develop one backbone network to support all telephone lines, televisions and computers.
- Digital communications are becoming more economical largely due to the availability of chips which can digitize the analog signals efficiently. The theories of digital communications have been around for a number of years but were not economically feasible. It is expected that the costs of digital communications will continue to decrease due to the advances in very large system integration (VLSI).
- Digital communications are more secure than analog communications because digital data streams can be scrambled (encrypted) by using sophisticated computer techniques. The encrypted bits can be deciphered (decrypted) only by equally sophisticated decryption devices/algorithms. The encryption/decryption on analog data (such as human voice) is not sophisticated. This is why "secure" telephone conversations first convert voice to digits before encryption.
- Newer technologies such as optic fibers benefit from digital transmission and advances in voice digitization are reducing the bandwidth requirements for voice-signals. The combined effect is that digital communications is the favored area of investigation and advancement.

Digital communication has been used by common carriers (telephone companies) for many years between telephone branch offices. Integrated Services Digital Network (ISDN) extends the digitization of the network into the customer premises (houses, offices). ISDN uses a "wall to wall" digital communication network to provide the users a universal access to a wide range of communication services. ISDN represents the means to integrate computing and communication technologies into a single framework.

A.3.2 Digitizing Techniques and Codecs

Figure A-8 shows a typical digital network with analog/digital (A/D) converters and interfaces. Digital data can be directly fed into a communication medium without any modulation/demodulation. However, the voice and video signals are analog and need to be digitized through A/D converters known as codecs (coders-encoders). The A/D technique commonly used is ***pulse code modulation (PCM***). The basic principle of digitizing an analog signal is that the signal is sampled at twice the signal bandwidth to faithfully represent the signal. This is based on the following Nyquist formula:

No. of digital samples per second = 2 x bandwidth of analog signal

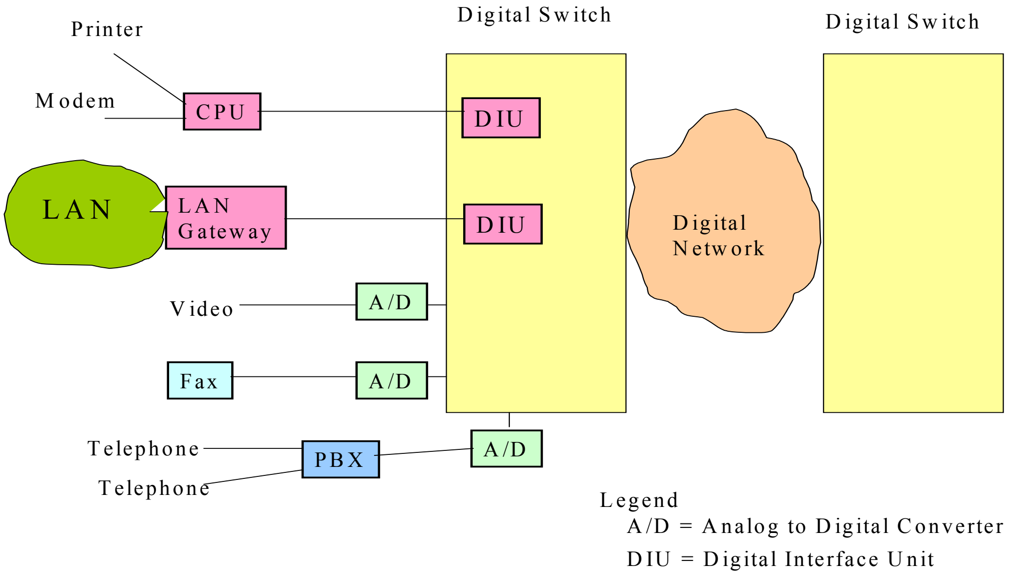


Figure A-8: A Typical Digital Network

To digitize voice, 8000 samples are taken in a second because the maximum bandwidth of human voice is 4000 Hz (the human voice is roughly filtered at 3000 Hz but this sampling rate improves the range). The pulse code modulation (PCM) digitizing technique consists of the following steps:

- Sampling: The analog signal is digitized at 8000 per second.
- Quantizing: Each sampled signal amplitude is converted to a level. PCM allows 128 levels of signal amplitude.
- Encoding: the amplitude is represented by bits. One byte (8 bits) is used to encode 128 levels (7 bits to represent 128 levels and 1 bit for supervisory and control use).

Thus one voice channel is converted to 64,000 bits per second (8000x8 data bits = 64,000 bps of data). This is the main reason why we see most digital transmission facilities provide 64 Kbps channels (each 64 Kbps channel can be used to carry one human telephone conversation). A 64 Kbps channel is termed a DS-0 channel.

Let us illustrate PCM by using another example in which 1 minute of voice signal needs to be stored after digitizing. First, there will be a need for 60x8000 = 480,000 samples. Since each sample occupies 8 bits (1 byte), the total storage required for one minute of digitized voice is 480,000 bytes. This illustrates that digitizing voice has important performance considerations - PCM would generate 8000x8 bits, i.e., 8000 bytes of data for only one second of human voice. Thus transmission of long conversations over communication lines poses large storage requirements especially if the voice needs to be stored at intermediate nodes. Many variants of PCM have been developed to address these problems [Verma90-Chapter 1].

In analog systems, the commonly used multiplexing scheme is FDM (frequency division multiplexing). In digital systems, the commonly used multiplexing scheme is TDM (time division multiplexing). In TDM systems, bits or small clumps of bits from different sources are interleaved on the same communication medium. Framing is used to identify which bits belong to which source so that the signals can be separated at the receiving side. Many variants of TDM have been developed over the years. Digital switches such as the PBXs (private branch exchanges) perform TDM so that one line can be shared by many digital users. In some cases, computing devices perform time division multiplexing so that one cable can be shared by many devices.

A.4 Communication Media Characteristics

Communication media are used in networks to transmit data over short or long distances. Selection and design of communication media play an important role in the cost, reliability and performance of networks. Communication media with high bandwidths and signal/noise ratios are desirable as evidenced by the following fundamental formula, derived by Claude Shannon, in 1948:

$$C = BW \log_2 (1 + S/N)$$

Where C = maximum data rate (in bits per second), BW = bandwidth of a carrier, and S/N shows a signal/noise ratio.

Thus increasing the bandwidth and/or the signal to noise ratio can improve the data rate. Consider, for example, a voice-graded line with signal to noise ratio of 1000 (a common line characteristic):

BW = 3000 Hz for a voice graded telephone line

$\log_2 (1 + 1000) = 10$ (approximately)

maximum data rate = 3000 x 10 = 30,000 bps

To improve the maximum data rate, the bandwidth and/or the signal to noise ratio needs to be improved. High bandwidth media are also desirable because they can support several users. For example, several telephone users can be supported on one high bandwidth cable. Different communication media with different bandwidths, signal/noise ratios, reliability and cost are currently being used in various networks. Examples of the commonly used media are open wire pairs, twisted pair cables, coaxial cables, fiber optic systems, and wireless media.

Open Wire Pairs. This is the oldest, and at present almost obsolete, communication medium. Open wire pairs are low cost bare copper wires which were installed in the early part of this century for telephones and telegraphs. Some of these wires can be still found in some rural areas. Wire pairs are being replaced at present because they are susceptible to damage by weather and suffer from attenuation (signal loss) and crosstalk (interference) problems.

Twisted Pair Cables. These cables are used extensively in telephone circuits in buildings and trunks. Several wires are insulated and then enclosed in a cable. A twisted pair cable may include up to 3,000 wire pairs with a bandwidth up to 250 KHz. These cables do have better performance than open wires, but the signal/noise ratio is low due to crosstalk noise. Twisted pair cables are good for short distance communications.

Coaxial Cables. These cables have been around since the early 1940s and are used extensively in local area networks, long distance toll trunks, urban areas, and cable TV. The technology consists of a single central conductor, surrounded by a circular insulation layer, and a conductive shield. Coaxial cables have high bandwidth (up to 400 MHz) with much higher quality data transmission than the twisted pair cables. For example, a coaxial cable can support over 10,000 voice circuits by using the frequency division multiplexing technique mentioned previously. With different multiplexing techniques, coaxial cables can deliver high data speeds (above 10 Mbps) and support many data, voice, and video channels. This technology is limited due to signal loss at high frequencies.

Optical Fiber. This communication medium is showing more promise and potential for very high data transmission applications. The optical fiber uses light rays instead of electronic pulses for message transmission. A fiber optic cable is very thin, usually resembling a human hair, which uses special cladding so that the light rays cannot escape the cable and thus travel down the cable in a reflective path. The light source used in fiber optic is usually a laser or a light emitting diode (LED). Fiber optics show very high frequency ranges (higher than 20,000 MHz). Because of this high bandwidth, a single fiber optic cable can support over 30,000 telephone lines and can transmit data over 400 Mbps (remember the cable carrying all these telephone lines resembles a single human hair strand!!). Due to their very light weight and high bandwidths, fiber optic use is growing dramatically. Other reasons for the popularity of fiber optics are: a)resilience to fire and gaseous combustion (light waves do not generate electrical sparks), b) very low signal loss and error rates, d) high security characteristics due to the difficulties in tapping fiber optic cables, and e) decreasing costs of fiber optic devices. At the time of this writing, a fiber optic connector is more expensive than a copper cable tap. Thus the largest application of fiber optic cable is in enterprise "backbone networks" which interconnect many networks (see Figure A-9).

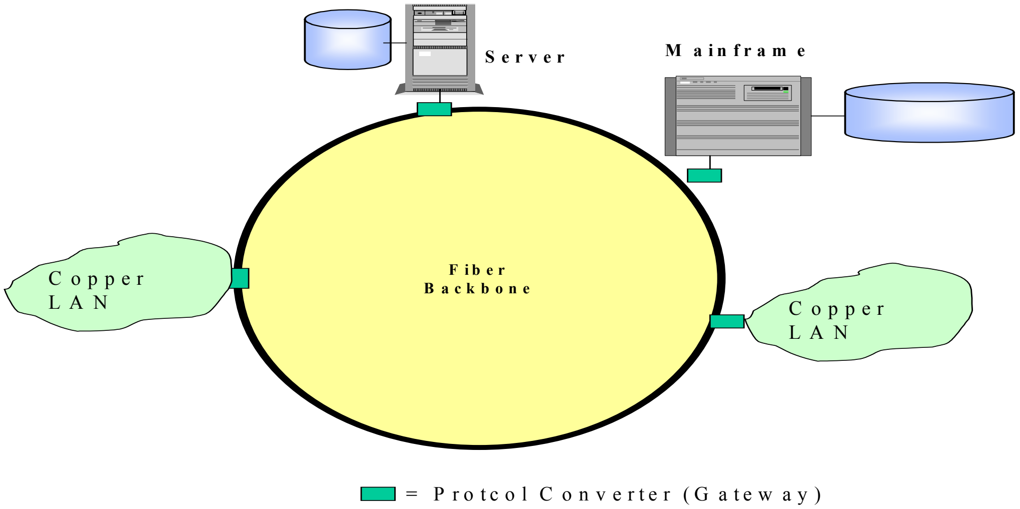


Figure A-9: A Fiber Optic Network

A.5 Topologies, Compression, and Encryption

A.5.1 Network Layout and Topologies

Network topology is concerned with how to interconnect N devices together by using some type of communication medium. The following network topologies are common (see Figure A-10).

The fully connected network, shown in Figure A-10a, connects every device to other devices. For N devices, the number of connections are N(N-1)/2. So, in order to fully connect 10 devices, 45 cables will be needed. This option gives very high reliability of the network because if one cable fails, the message can be routed through another cable. However, it is too expensive for most practical cases and is used rarely.

The tree topology, depicted in Figure A-10b, connects the devices to a hub which passes the messages from one device to the other. As we will see, tree topologies are very common in wide area networks because they are easy to monitor and diagnose/correct. For example, several terminals may be connected to a terminal server, or a terminal controller, which in turn is connected to a computer. Despite their popularity, tree topologies have two main weaknesses. First, if the hub fails then the subnet managed by the hub fails. Second, the number of devices supported depends on the number of communication slots in the hub. For example, if the hub has only 8 slots, then another hub will need to be purchased for the 9th device.

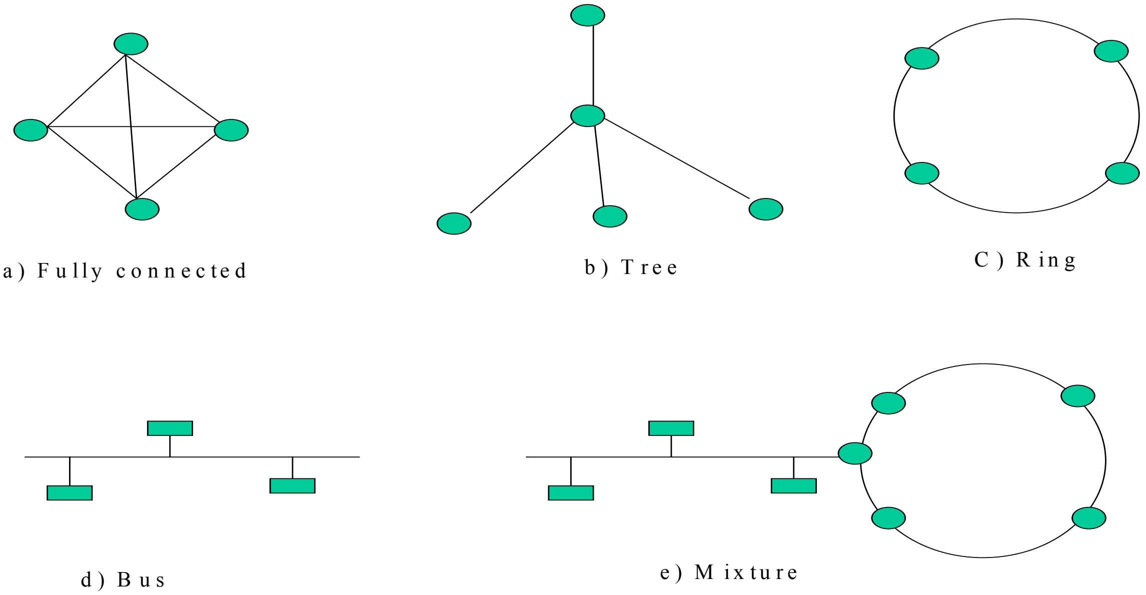


Figure A-10: Network Topologies

A ring connects the devices to form a loop as illustrated in Figure A-10c. In earlier ring systems, each device on the ring served as a relay in which the message was received on one slot and sent out on the other. In these cases, if a device failed then the whole network failed. In modern rings, devices are attached to the ring cable so that if a device fails, the ring continues to work. The main advantage of a ring is that the number of devices connected on the network is not limited by the number of slots in any device (each device may use at the most two slots to be connected to the ring). Many ring oriented LANs are actually wired as trees. We will see that rings are used commonly in local area networks.

The bus topology, shown in Figure A-10d, connects the devices to a cable which is terminated at both ends. All devices on the bus can "listen" to the messages being passed on the bus and insert/retrieve messages whenever needed. The bus topologies are commonly used in local area networks. Perhaps the best example of the bus topology is the Ethernet local area network which will be discussed later.

In addition to these basic network topologies, many mixtures can exist in large networks. For example, a backbone network may be a fiber optic ring to which many devices and local area networks are connected in a tree format, treating the backbone as a hub (see Figure A-10e). In small networks with high data rates, it is possible to think of devices to be fully connected logically even if the physical topology is a tree, bus or a ring. This makes it easier to model a network. Choice of a network topology is nontrivial because it depends on many factors such as the physical location of the work activities, the type of devices that need to be connected, cost considerations, response time requirements, availability requirements, and vendor preferences. See [Sharma 1997, Bertsekas 1992] for a review of the approaches.

A.5.2 Asynchronous Versus Synchronous Transmission and Full/Half Duplex

The information transferred over a communication medium can be asynchronous or synchronous. In ***asynchronous transmission***, also called start-stop transmission, data is transferred one character at a time, at an undeterministic or random time. In order for the start and end of a character to be recognized by the receiver, each character is surrounded by a start and a stop bit, hence the name start-stop (see Figure A-11a). Asynchronous communications were introduced in the old telegraphic days when the start and stop bits physically started and stopped the paper tape mechanical processes on the receiver side. Asynchronous communications are still used very frequently in terminal communications. The sending and receiving logic needed for asynchronous communications is very simple and can be implemented economically in simple devices. The main difficulty with the asynchronous communications is that if a start bit is not recognized then the whole character may be misread. Due to this, asynchronous communications are not used at high speeds because the chances of losing data bits are greater at higher speeds.

In ***synchronous data transmission***, data is transmitted in blocks of many characters. Header and trailer characters are attached to each block so that the receiver can recognize the start or stop of a block (see Figure A-11b). Common examples of synchronous transmission are the transmission of each terminal screen as a single block and file transfers which send and receive many file records as blocks of data. Synchronous communications can operate at much higher speeds than asynchronous because the whole block can be transmitted by using one send command. The logic of generating the block headers and trailer fields is more complicated. In addition, sophisticated error checking fields are generated and verified. The formats of the headers, trailers, and error checking fields depend on the synchronization scheme being employed. Some systems, for example, include the start stop bits in the data block, in addition to the block headers, trailers and error checking fields.

The information can flow over a communication medium in one of the following modes:

- Simplex: one way communications, always. This mode is rarely used today.
- Half Duplex: two way communication, one at a time. This mode is used very commonly in many instances.
- Full Duplex: two way communications, simultaneously. This is the fastest mode of information transmission.

It should be noted that to provide a full duplex path between two end-devices, all intermediate devices and media on the communication path must operate in full duplex mode.

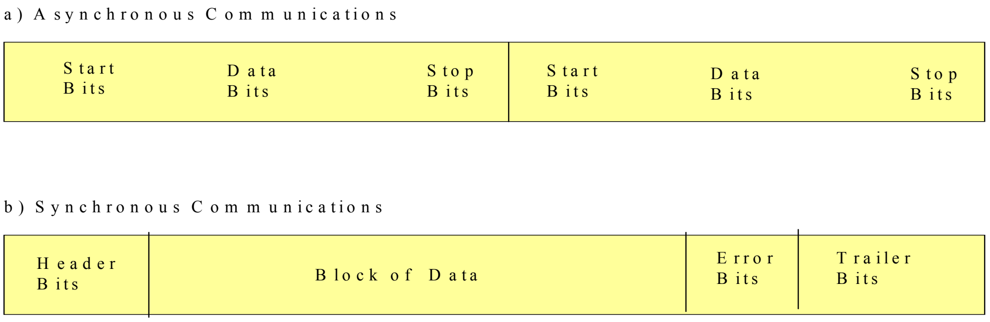


Figure A-11: Asynchronous and Synchronous Communications

A.5.3 Data Encryption and Compression Techniques

Data encryption has been used for a number of years in military applications to mask the military messages so that the hostile interveners could not understand the messages. Due to the increase of sensitive information handled by computer systems (e.g., financial data, confidential records) data encryption/decryption has become a major area of active work. When the data is transmitted over communication channels, it is possible for someone to tap a channel and gain unauthorized access. Thus, data encryption in communication systems is of vital importance. The objectives of data encryption are:

- Privacy
- Authentication
- Data integrity

In the simplest case, data is transformed by a key into an encrypted message. The encrypted message is then transmitted and decrypted on the other side by using the same key. Encryption/decryption can be performed by hardware and/or software. Modern computing systems have the ability to implement very sophisticated encryption/decryption techniques. The same encryption can be used on all data in a system or encryption keys can be more "personalized". For example, instead of using the same encryption/decryption key on all data from all stations in a network, each station or user can use its own encryption/decryption key. A user can have his or her own encryption card which is inserted into a workstation before the user logs on. This card encrypts the data before sending it across the network. The encrypted data can be read only by those users or programs with access to the same encryption key.

A discussion of the encryption/decryption algorithms is beyond the scope of this book. A good overview of these algorithms can be found in [Tannenbaum 1996].

Data compression techniques are used to reduce the size of data that needs to be sent across communication channels. For example, instead of sending 200 consecutive blanks across a network, a data compression technique can send one blank character with a multiplier of 200. Data compression techniques can significantly reduce the transmission time. For example, I used a data compression routine three years ago which reduced the size of data to be sent from 95 million bytes to 7 million bytes, thus cutting the transmission time by 90%. This raised interesting tradeoffs between increasing link speed/cost versus the compression/decompression processing delays and costs. The common data compression techniques are as follows:

- Data string encoding. Commonly used data strings can be encoded for data transmission. For example, customer names can be represented by customer ids for data transmission.

- Word encoding. Some words occur more often than others in natural languages. For example, "the" occurs more frequently in English than many other words. Such words can be encoded by using bit combinations for data transmission.
- Frequency encoding. Some words, or bit strings, are used repeatedly. For example, blank characters appear consecutively in many data files. A blank character followed by a frequency count can be used easily in such systems.

For a detailed discussion of data compression algorithms, see [Tannenbaum 1996].

A.6 LANs, WANs, and MANs

A.6.1 Local Area Networks

Simply stated, a local area network (LAN) is a network of data communication devices within a small area (typically 100 to 1000 meters). The main characteristics of local area networks, also called local networks, are:

- **Private ownership**. The LAN equipment, including the communication media, are privately owned. A legal restriction in the US does not allow individuals to string cables over public property (underground cables are allowed). All above ground communication facilities are provided by the common carriers (AT&T, MCI, Sprint, Regional Telephone Companies, cable TV companies, etc). Thus if you want to communicate between two houses located across a road, you must purchase the common carrier facilities, i.e., you do not privately own the communication medium. This leads to an interesting way to look at a LAN: a LAN is independent of the common carriers. If you happen to own a city, then your LAN may cover the entire city. For the rest of us, our LANs may not go beyond a room.
- **High data rates**. The data rates of LANs are much higher than the common wide area networks. For example, most wide area networks use data rates ranging from 56000 bps (bits per second) to 1.54 Mbps (million bits per second), while most local area networks use data rates between 10 Mbps to 100 Mbps.
- **Low error rates**. The error rates in LANs are much lower than the typical wide area networks. This is mainly because of the short distance and the use of simple communication devices in LANs.
- **Broadcast services**. LANs typically broadcast the messages to receivers in contrast to WANs which usually select a receiver before sending a message. Broadcasts, if misused and disregarded, can cause many administrative problems.

Figure A-12 shows a typical LAN in which many computing devices are connected together through communication media such as twisted pair cables, coaxial cables, or fiber optic cables. At the basic level, a LAN consists of a *LAN segment* (a physical cable) that is connected to various *hosts* such as personal computers, laptops, terminals, printers, TV sets, and/or sensors. A LAN segment may use a bus, ring or tree topology. Larger LANs are formed by interconnecting several LAN segments. A LAN "server" is a computer on the LAN which provides a set of services for sharing common resources. A server may be a personal computer, a workstation, a specialized computer, or a minicomputer. From an end-user point of view, a LAN server allows a user to access remote resources l(e.g., printers, disk drives, files, and databases located on the LAN. Common examples of services provided by the LAN servers are as follows:

- **Printer sharing services**. The computers and terminals on a LAN need to share high quality (e.g., color) printers. A LAN provides access to a common printer (say, LPT2) in addition to a local printer such as LPT1 (see Figure A-12).
- **Disk sharing services**. Many LANs are configured so that one computer (LAN server) has a large disk on which many packages are installed. Many small computers with limited disk storage access the server to retrieve the needed software. The LAN server provides another drive (say drive n) which can be accessed by any computer on the LAN (see Figure A-12).
- **File/database sharing services**. One file/database may be shared by several LAN users. The file/database server coordinates the data access for integrity control. For example, the server may lock the data resource at the file level (if one user is updating the file, then no one can access the file) or at a record level (deny access to the record being updated).

Keep in mind that a "server" is not a hardware device – in fact, server is a function that may be imbedded in a software module or in an ASIC (application specific integrated circuit) chip. In most cases, print server, disk server, and file server are sofware modules that are installed on more powerful computers, thus giving rise to the notion of "server" computers. Typically, a single computer on a LAN houses the print, disk, and file/database servers (e.g., the Novell Netware Server). It is also possible to assign the servers to many computers on a LAN.

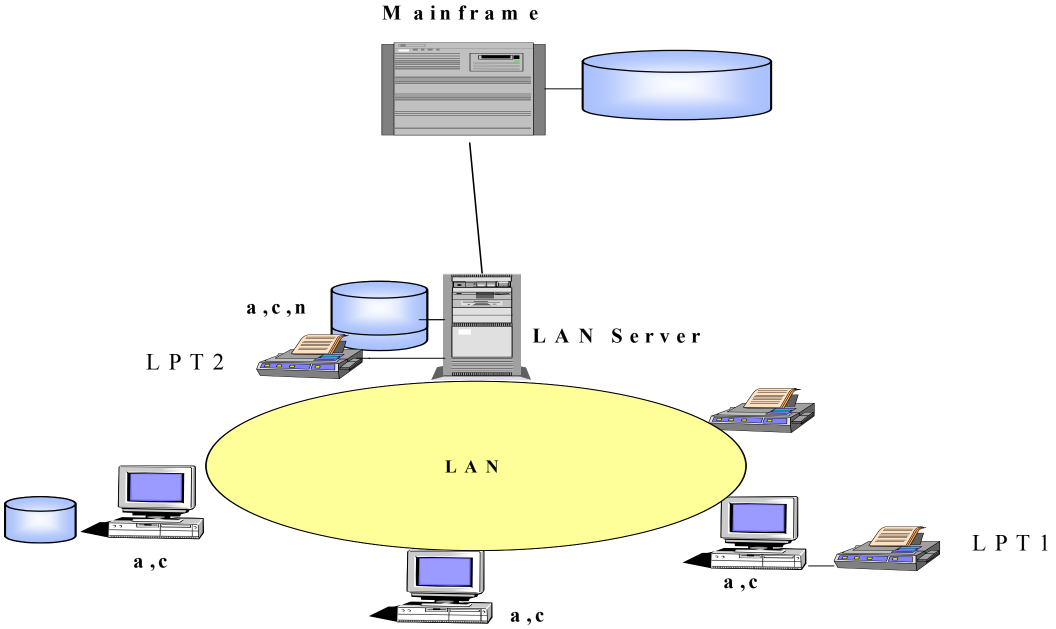


Figure A-12: A Typical Local Area Network

LANs were introduced in the marketplace in the early 1980s. In the early stages, there was no agreement on LAN technologies and many LAN vendors were pushing proprietary solutions. For example, in the early 1980s, more than 50 vendors were marketing LANs on different devices, using different communication media, protocols, and topologies. The IEEE (Institute of Electrical and Electronics Engineers) 802 Committee on LANs was formed in 1980 to develop standards for LANs[1]. Thanks to the work of this committee and market pressures for interoperability and minimization of interfaces, the LAN landscape has simplified considerably with only a few surviving vendors and technologies. We will primarily present a review of the current and future technologies and will not dwell on history, unless of value.

[1] 802 indicates February of 1980, the date when IEEE 802 Committee was formed.

Two techniques for message transmission and recognition are of significance at present (e.g., token passing and CSMA/CD). These techniques, accepted by the IEEE 802 Committee, are called data link control techniques. When a LAN package is purchased, it may need a) cables to interconnect computers; b) adapter cards which connect the cables to the devices; c) LAN server software which is used for sharing the resources (printers, disks, files); and d) station software which is installed at each computer attached to the LAN.

A.6.2 Wide Area Networks

Wide area networks (WANs) are the oldest form of communication networks. WANs use the telecommunication facilities of common carriers (telephone companies) to exchange data between end-devices (computers, telephones, sensors, TVs). We briefly review the telecommunication networks (networks owned by the telephone companies) before describing the WAN components, switching systems, and WAN design issues.

Since the first telephone patent by Alexander Graham Bell in 1876, more than 600 million telephones had been installed throughout the world by the mid 1980s [Datapro 1987]. Since then, this number has jumped dramatically, more than doubled, thanks to the increase in cellular phones. This growth in the telecommunications industry, expected to continue at 6% annually, has led to large telecommunications networks with a variety of terminating equipments (e.g., telephones, computer devices), transmission facilities (communication lines), and switches. The telecommunication networks are designed to minimize the cost of end-to-end connections. For example, to fully connect 100 phones directly with each other, we would need about 5000 lines. Naturally, the number of direct lines between 1 billion telephones would be more than a normal human being would like to imagine.

Figure A-13 shows a typical telecommunications network design. Most of this network is owned by the common carriers (telephone companies). On the customer premise (an office, a house), the user interacts with telephone equipment and computing devices which are connected to the local central office through a subscriber loop. A customer premise may also include other equipment such as local area networks and a private branch exchange (PBX). A PBX, also called a computerized branch exchange (CBX), routes the customer premise calls internally without going to the subscriber loop. The PBX provide the telephone extensions in office buildings (if you call an extension within your office, the PBX in your building routes the call to another phone in the building). A PBX/CBX may be used for all voice and data communications within an office.

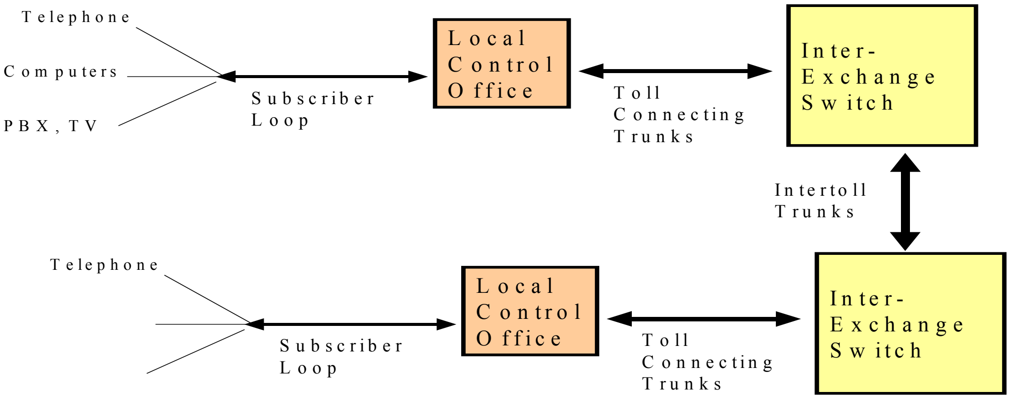


Figure A-13: A Telecommunications Network

The subscriber loop, also known as the local loop, consists of the wires, poles, conduits and other equipment that connects the customer premise equipment to the telephone company's central office. Before deregulation, the telephone companies had a monopoly on the local loop. At present, customers can use cable TV or wireless systems to bypass the local loop.

The local central office, often called an end-office, is the point where a local loop terminates. In a metropolitan area, many local central offices are located based on the population densities. A switch at the end-office routes the call to another end-office either by directly connecting to the end-office (if within the same area code) or through the interexchange switch. The telephone companies divide their service areas into exchanges where an exchange roughly refers to a city or part of a city. The inter-exchange switches route calls between exchanges over trunks. These trunks are called toll trunks because they connect outside the free calling areas. The toll trunks are fast transmission facilities, at present almost all digital.

WANs, as shown in Figure A-14, are quite complex. The computing devices such as terminals (dumb, smart, and/or programmable), printers, sensors, appliances (e.g., TVs, telephones), laptop/desktop computers, minicomputers, and mainframes are the end-points in a WAN and are interconnected through telecommunication lines and an array of intermediate devices such as switches and routers. These lines may be switched (dial up) or leased (permanently connected) between the end points. Packet switching systems that breakup your messages into small packets and then route them over the WAN are much more popular for the reasons we will see in the next chapter.

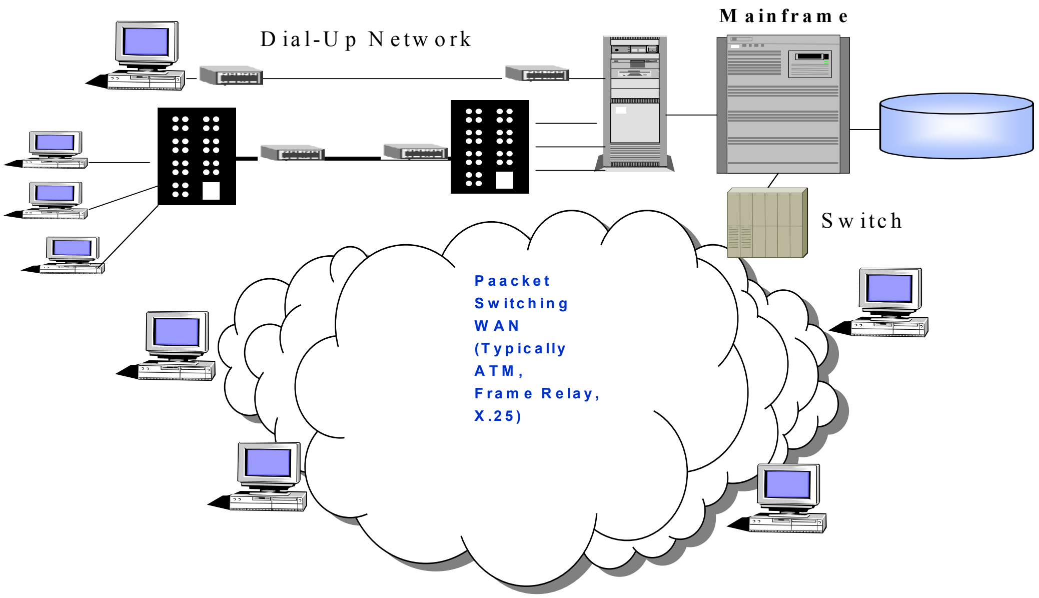


Figure A-14: Common Wide Area Network Configurations

A.6.3 Metropolitan Area Networks

The metropolitan area networks (MANs) extend the scope of local area networks beyond the customer premises to cover a geographical area (e.g., a city or a county). A common definition (there are many slightly varying definitions) is that a MAN is a large LAN under the control of one authority and using a shared transmission medium. The origin of MANs is the cable TV industry that provides connectivity between users within a metropolitan area. The MANs are connected to the WANs for sewnding information to far off places through the Internet. This is the foundation for cable modems – they connect your computer to a MAN that in turn is connected to several WANs. A MAN typically covers 50 Km diameter and operates at data rates above 50 Mbps.

A.7 Enterprise Networks and Network Architecture

A.7.1 Overview

Enterprise networks comprise of many interconnected LANs. Internet, as we will see later, is a large network of networks. ***Network architectures*** needed to specify how these pieces fit together to form a functioning network – they define the components, the functions and the interactions/interfaces (protocols/standards) between the components of a network. A network architecture encompasses hardware, software, standards, data link controls, topologies, and protocols. These components may be very simple or quite complex depending on the size of the network and the nature of devices supported (mainframes, minicomputers, microcomputers, terminals). Network architectures provide a systematic approach to describe the various categories of networks and define exactly what components will be supported and how. A ***protocol*** is a set of precisely defined rules of behavior between two parties. As we will see, protocols in network architectures define the formats and the rules of interaction between peers. Protocols play a key role in integration and interconnectivity in distributed systems.

Figure A-15 presents a simple network architecture model. In this figure, several computers are connected to the network. One or more user applications (e.g., order processing) may run on each computer. Application Support Services "connect" the applications to the network. Network Services provide exchange of messages between the computers.

The functions performed by the application and network services vary widely between the size and complexity of networks and the stations. In the 1970s, these services were viewed as layers where each layer performed a specific function. Different researchers, vendors and standardizing bodies have proposed different layers. For example IBM's SNA (System Network Architecture) uses 7 layers; Department of Defense's Suite, commonly referred to as the TCP/IP Suite, uses 4 layers; and the OSI Reference Model uses 7 layers.

Even though the number of layers differs between vendors, in all cases the lower layers provide low level (closer to the physical network) functions while the high level functions (application interfaces) are performed by the upper layers. For example, the first layer is physical link, the last is the application layer in most network architectures. In these layered systems, the data expands at source as it goes through different layers (additional pieces of information are added as headers in each layer) and shrinks at the sink (headers are removed successively). We will see this in Section A.7.2.

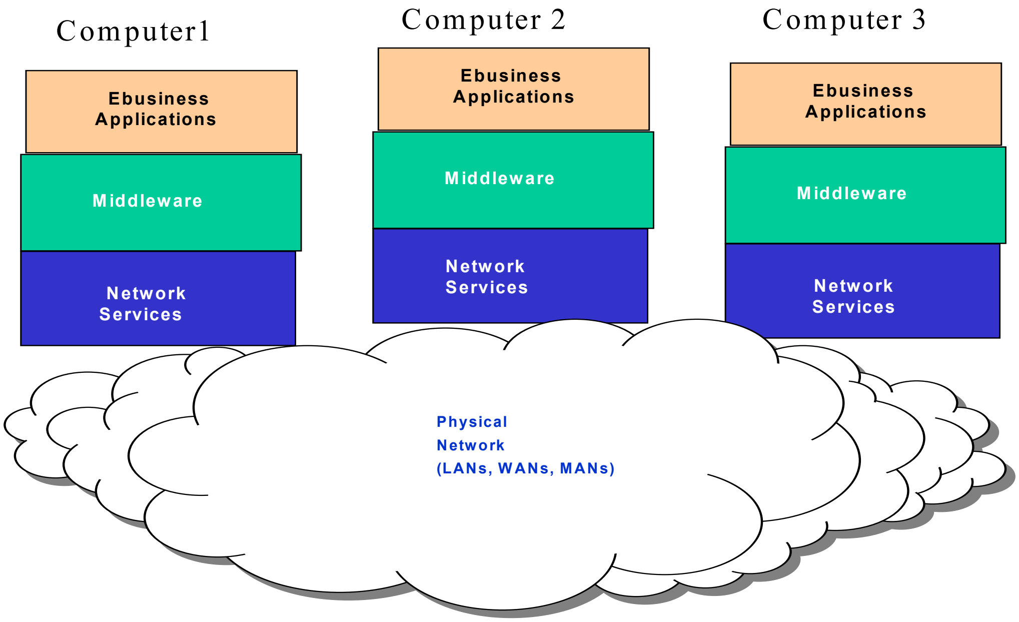


Figure A-15: A Simple View of Network Architecture

A.7.2 The Open System Interconnection (OSI) Reference Model

The Reference Model for Open Systems Interconnection (OSI), commonly referred to as the OSI Model, was proposed by the International Standards Organizations (ISO) Committee 97 in March, 1977. This Model, published as ISO Document 7498, consists of 7 layers shown in **Error! Reference source not found.**. The objective of the ISO subcommittee was to describe a hierarchical or layered set of generic functions that every network must fulfill. Such a generic definition makes it easier to develop interfaces between the growing number of different networks. The ISO subcommittee precisely defined the set of layers and the functions/services performed by each layer. According to the conceptual framework in **Error! Reference source not found.**, the first four layers (Physical, Data Link, Network, and Transport) are lower level layers which are responsible for the delivery of data between applications. These layers perform the following functions (additional information about these layers is given in the next section):

- The Physical Layer is concerned with transmission of bit stream over physical medium and deals with the mechanical, electrical, functional, and procedural characteristics to access the physical medium. For example, the physical interfaces such as RS232 and communication techniques (broadband, baseband) are handled by this layer.
- The Data Link Layer provides for the transfer of information across the physical link by sending blocks of data (frames) with necessary synchronization, error control, and flow control functions. For example, this layer handles the reception, recognition, and transmission of tokens and Ethernet messages.
- The Network Layer provides upper layers with independence from the data transmission, switching and routing technologies needed when the end devices have to cross many networks. For example, the packet switching activities of breaking up and reassembling the messages are performed by this layer.
- The Transport Layer provides transparent transfer of data between end systems and is responsible for end-to-end error recovery and flow control. The Transport Layer only resides in end systems. For example, in a large network with many interconnected LANs and WANs, the transport layer at the end systems will make sure that the messages exchanged between end systems of this network are not lost while fighting their way through various LANs and WANs. A message may be exchanged correctly between two stations of a LAN in a building but it may not be sent properly to a remote computer located in another city.

Whereas the lower four layers are responsible for transport of information between applications, the upper layers support applications (**Error! Reference source not found.**). Specifically:

- The Session Layer controls the communication between applications by establishing, managing, and terminating connections (sessions) between cooperating applications. For example, this layer establishes the full versus half duplex interactions between applications.
- The Presentation Layer provides independence to the application processes from differences in data representation (syntax). The encryption/decryption for security purposes is usually performed in this layer.
- The Application Layer supports the application and user processes which need network services. Examples of the services provided by this layer are terminal emulation, file transfer, electronic mail, distributed database managers, etc.

Figure A-17 illustrates the concept of end systems and intermediate systems in OSI. An ***intermediate system*** only performs functions related to the lowest three layers of the Reference Model (e.g., routing, flow control, and bit transmission). Functions of intermediate systems are implemented in ***relay systems***. An example of a relay system is a router which connects many LANs together. An ***end system*** provides the functions above the Network Layer (Transport, Session, Presentation, Application), in addition to the lowest three layers. Examples of end systems are computers on which applications reside. End system is a synonym of "host", a term commonly used to refer to computers where applications reside. We will discuss these systems in more detail in our discussion of network interconnectivity (Section **Error! Reference source not found.**). The physical aspects of communication networks discussed in Chapter 1 are concerned with layers 1 through 3 of the OSI Model.

Services, Protocols and Standards. The following terms are used frequently in OSI:

- *Service:* The functions performed by layer N for layer N+1.
- *Protocol*: The precise rules of information exchange between two peers.
- *Standard:* An agreed upon formal specification of the protocols and/or the services.

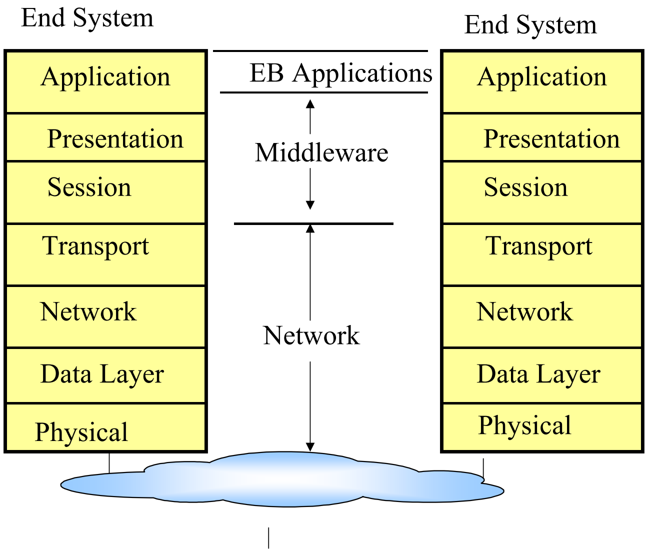


Figure A-16: The OSI Model

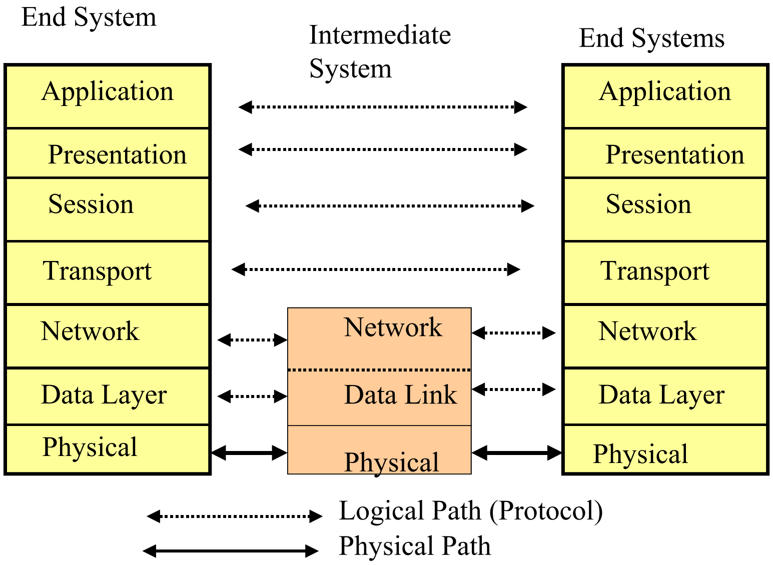


Figure A-17: ISO Model- End Systems, Intermediate Systems and Protocols

Let us discuss these terms a little more. Each OSI layer performs a set of related functions. These functions are viewed as services by the next higher layer. For example, encryption/decryption is a service provided by Layer 6. Layer N hides unnecessary information from layer N+1 (for example, layer 7 does not have to worry about encryption/decryption). In turn, layer N relies on layer N-1 to perform more primitive functions and to conceal the details of those functions. The same set of layered functions must exist in two systems to communicate with each other.

A protocol specifies two things: the message format (e.g., bit pattern) and the rules to interpret and react to the messages. Protocols are the "visible" aspect of OSI because they are the basis for interconnection. Protocols can be tested and verified for conformance. Communication is achieved by having corresponding ('peer') entities in the same layer in two different systems communicate via a protocol. For example, we can think of protocols at each layer of the OSI Model (e.g., physical protocols, data link protocols, network protocols, presentation protocols, application protocols).

Standards can specify high level views as well as detailed procedures. The OSI standards specify the OSI Reference Model, the services to be provided by the different layers of the Model and the protocols for exchange of information between peers. A large number of OSI standards have been published over the years (more than 1000 were published in 1990 [Folts 1990]). These standards define information such as the OSI Reference Model itself, the many services provided by the 7 layers of the Model (different services are defined for different industries and applications) and the detailed protocols between peers at different systems.

Entities. There are one or more entities in each layer of a system. An ***entity*** implements functions of that layer and the protocol for communicating with peer entities in other systems. Examples of an entity are a process implemented on a chip or a software subroutine. Each entity communicates with entities in the layers above and below it across an interface (a port) called a ***service access point***. An entity at layer N requests the services of layer N-1 via invocation of primitives. An example of a primitive is a subroutine call.

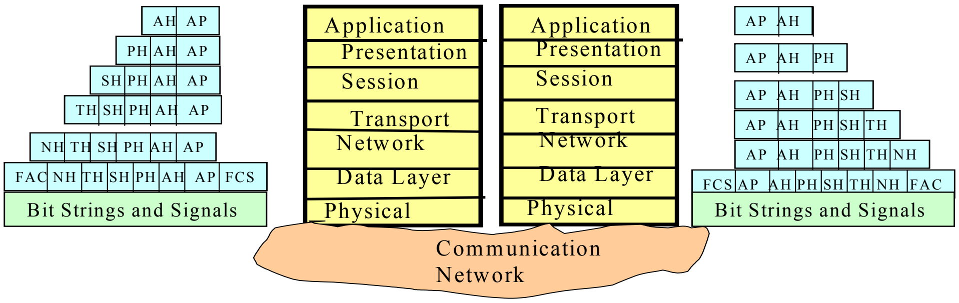
Connection-based and Connectionless Communications. The OSI Model supports connectionless as well as connection-based communications. The basic idea of connection-based service, also called a reliable service, is that before any communication between two entities at layer N takes place, a connection between end systems at N-1 must be established. Connection-based communications involves three phases: connection establishment, data transfer, and connection release. Due to the overhead of connection establishment and connection release, several systems use a "connectionless" service. The notion of connectionless service is almost oximoronic: How can you provide a service when you are not even connected? Basically, a connectionless service means that the communicating parties send and receive self contained data packets without apriori connection (i.e., there is no separate connection establishment and connection release).

An analogy will explain the difference between connectionless versus connection-based services. Connection-based services are similar to the telephone communication while the connectionless services are similar to the postal system. If you use a telephone, then you first dial the number (connection establishment), talk (data transfer), and hang up (connection release). In this case, you establish a connection before data transfer. In contrast, you do not establish a connection before you send a message in a letter, and there is no assurance that the receiver has received the message. However, you can design your own protocol to make sure that the mail was delivered (e.g., put a note in your letter indicating that the receiver must call you immediately to claim one million dollars he has won in a lottery). Due to this analogy, the connection-based service is referred to as a virtual call and the connectionless service is referred to as a datagram service.

Before proceeding with the OSI layer details, we should note a few things. First, it is important to note that OSI is used as a conceptual model -- it is not an implementation model. In other words, it is possible for many layers to be implemented in a single software module or an adapter card. Secondly, the OSI model has not been a major commercial success. In fact very few, if any, commercial implementations of OSI are currently in use. However, OSI is an excellent conceptual framework that provides a common, vendor neutral, terminology that is very valuable in analyzing network architectures and network connectivity.

3.1.1 Information Flow in the OSI Model: An Example

The objective of any data communications network is to exchange data between applications or between users. To do this, the information to be transferred must be formatted, packaged, routed, and delivered. The receiver must then unpackage and possibly reformat this information. These are essentially the functions performed by the seven layers. The information from the application layer in processor 1 moves down through the lower layers in its node until it reaches the physical layer, which physically transmits the data to the physical layer in processor 2. The data then work their way up through the layers in processor 2 until they reach the application layer of that processor. Each layer in the sending processor performs work for or acts on behalf of, its peer layer in the receiving processor. Thus, presentation layers support presentation layers, session layers support other session layers, etc. Between the different layers are interfaces through which the data pass.



Legend
AP = Application Data
AH = Application header, inserted and removed by application layer
PH = Presentation header, inserted and removed by presentation layer
SH = Session header, inserted and removed by session layer
TH = Transport header, inserted and removed by transport layer
NH = Network header, inserted and removed by network layer
FAC = Flag, address, and control indicators -- inserted and removed by data link layer
FCS = Frame check sequence -- inserted and removed by data link layer

Figure A-18: Information Flow in a Network

Figure A-18 illustrates the information flow in the OSI Model. Communication is between applications in the systems. If application A in computer C1 wishes to send a message to application B in C2, it invokes the application layer (layer 7). Layer 7 at C1 establishes a peer relationship with layer 7 of C2, using a layer 7 protocol. This protocol requires services from layer 6, so the two layer 6 entities use a protocol of their own. Similarly layers 5, 4, 3, 2 and and 1 all have their own protocols. It should be emphasized that the actual bit transfer is only done at the physical layer, which actually passes the bits through a transmission medium. Above the physical layer, each protocol entity sends data down to the next lower layer in order to get the data across to its peer entity.

When application A has a message to send to application B, it transfers the message to an application entity in the application layer. A header is appended to the message that contains the required information for the peer layer 7 protocol. The original message, plus the header, is now passed as a unit to layer 6. Layer 6 treats the whole unit as data, and appends its own header. This process continues down through layer 2, which builds a frame with both a header and a trailer. This frame is then passed by the physical layer onto the transmission medium. When the frame is received by the target system, the reverse process occurs. As the data proceeds to higher layers, each layer strips off the outermost header, acts on the information contained in the header, and passes the remainder up to the next layer. Some layers may fragment the data unit it receives from the next higher layer into several parts. For example, the network layer breaks the application messages into "packets" for packet switching systems. These data units must then be reassembled by the corresponding peer layer before being passed up.

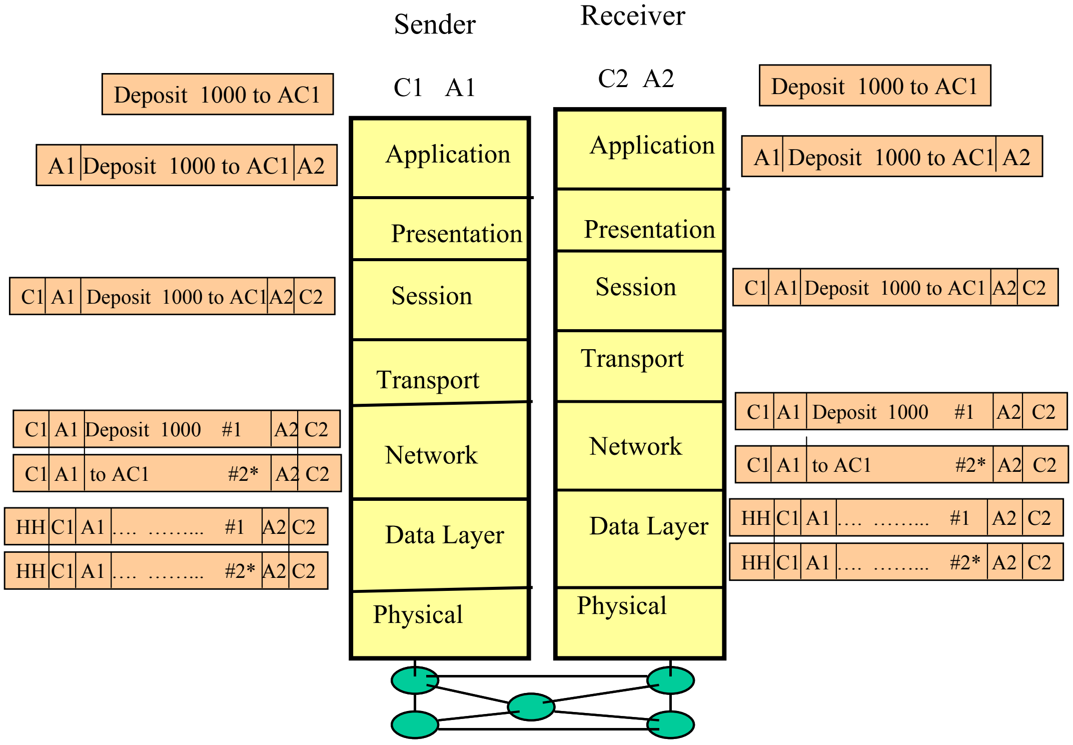

Sender
Receiver
C1 A1
C2 A2
Deposit 1000 to AC1
Deposit 1000 to AC1
A1 Deposit 1000 to AC1 A2
A1 Deposit 1000 to AC1 A2
Application
Application
Presentation
Presentation
Session
Session
C1 A1 Deposit 1000 to AC1 A2 C2
C1 A1 Deposit 1000 to AC1 A2 C2
Transport
Transport
Network
Network
C1 A1 Deposit 1000 #1 A2 C2
C1 A1 to AC1 #2* A2 C2
C1 A1 Deposit 1000 #1 A2 C2
C1 A1 to AC1 #2* A2 C2
Data Layer
Data Layer
HH C1 A1 .... #1 A2 C2
HH C1 A1 .... #2* A2 C2
HH C1 A1 .... #1 A2 C2
HH C1 A1 .... #2* A2 C2
Physical
Physical

Figure A-19: An Example of Information Flow

Figure A-19 illustrates the information flow and the message format in the OSI Model through an example. Suppose a process P1 on computer C1 issues the message: "Deposit 100 to account no. AC1". The user program finds out that process P2 is authorized to handle account credits/debits. Note that the message is now appended with the sender and receiver information (P1 and P2). This data unit is transferred to lower layers where the sender and receiver computers are identified (C1 and C2) and added to the message. The transport layer establishes a path between C1 and C2. Note that if P1 and P2 were on the same machine, then the transport layer services would not be needed (the session layer in C1 can establish a session within C1). The network layer, say, decomposes the message into two packets (#1 and #2). The two packets are now converted into frames which are routed to the network. Note that the network consists of many subnetworks which are connected through intermediate systems C3 and C4. This network provides two alternate paths for messages (one through C3, the other through C4). Let us assume that packet#1 is routed through C3 and packet#2 is routed through C4. These two packets will be reassembled into a single message by the network layer at C2. If, for example, C4 crashes and packet#2 is lost, then the network and transport layers will assure that the lost packet is retransmitted. Note also that the headers are successively removed as the message moves up in the target system. The final message "Deposit 100 to AC1" is eventually received by the process P2 at C2.

A.7.3 Network Interconnectivity

Network interconnectivity is needed in large networks to provide interfaces and transport of messages between remotely located users, applications, databases, and devices. For example, if you access the Paris University Web site from Chicago, then many interconnectivity devices are used to get you from Chicago to Paris. The two principal network interconnectivity devices are:

- **Routers** find a path for a message in larger networks and then send the message over the selected path. Routers use very sophisticated routing algorithms and provide functionality such as "fire walls" for security checking.
- **Gateways** translate one type of protocol to another. In most large networks, protocols of some subnetworks need to be converted to protocols of other subnetworks for end to end communications. A gateway connects two dissimilar network architectures and is essentially a protocol converter. A gateway may be a special purpose computer, a workstation with associated software (e.g., a PC with gateway software), or a software module which runs as a task in a mainframe. An example of gateways for network interconnectivity is the TCP/IP to Novell LANs.

Many routers and gateways are used commonly in enterprise networks and the general Public Internet. For example, if a salesman in Detroit needs to access a customer database in New York, then a series of routers and gateways would be needed to find the path between the two cities. Figure A-20 shows a realistic enterprise network that uses TCP/IP very heavily, except the IBM SNA network (an old network technology) at the mainframe. The routers are used between all TCP/IP network segments and gateways are used to convert the TCP/IP messages to SNA and the Novell protocol. . .

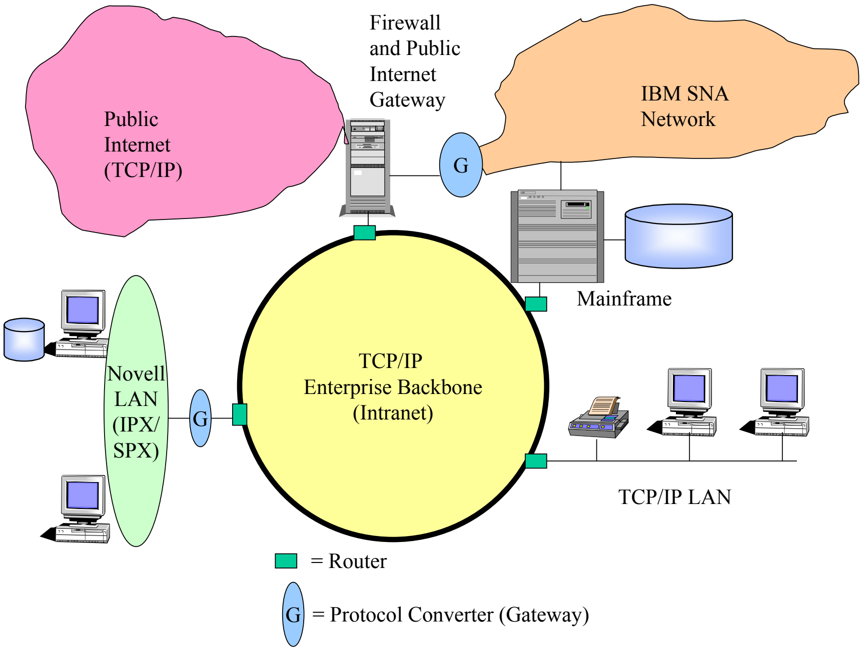


Figure A-20: Network Interconnectivity in an Enterprise Environment

A.7.4 Broadband Networks

The network communication technologies are advancing at a very rapid pace. In particular, we are witnessing growth of high speed, also known as ***broadband***, networks that exceed 100 Mbps data rates. Broadband is a general term that refers to "high speed" data communications, typically1Mbps, or higher. An interest in broadband communications has taken center stage over the past few years because broadband technologies represent the speed at which users can access the Internet. The drivers for the growth of broadband services are:

- The Internet continues to be the primary driver for the communications industry. Internet traffic is doubling every year and is expected to keep growing in the future. The increased traffic is driving the need for networks that can handle the increased bandwidth levels.
- The convergence of data and voice networks onto a single network infrastructure is also a driver of increased bandwidth needs. The converged networks must simultaneously support the highly interactive client /server applications and the voice/video applications (see side bar "Bandwidth Consumers").
- LAN Interconnectivity: Basically, LANs are fast but WANs are not. The demand for high speed LAN-to-LAN interconnection is driving high speed requirements (it is silly to connect high speed 100 Mbps LANs through a 56 Kbps WAN).

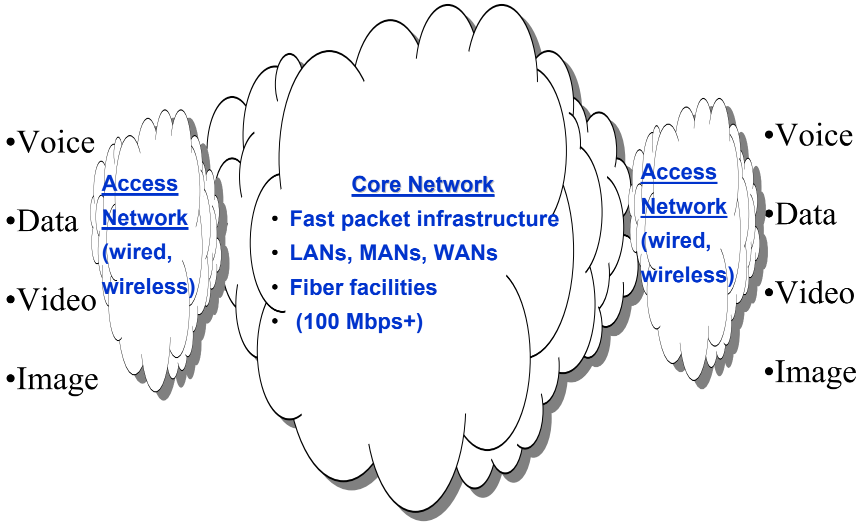


Figure A-21: Conceptual View of "Next Generation Network (NGN)"

Error! Reference source not found. shows a conceptual view of a converged network that supports voice, data, and video over broadband and wireless services. Such networks are being referred to as ***"Next Generation Networks (NGNs)"***. The key players in NGN are:

- ***Core Network*** that provides the high speed short and long-haul capabilities to transfer information between end points. This network uses high speed transmission facilities.
- ***Access network*** that provides access from customer premise to the core network. This network, also known as the ***"last mile***" uses technologies such as DSL, Cable Modem, and other technologies for broadband services. Wireless networks are playing an important role in the last mile through the use of cellular networks, wireless LANs, and wireless local loops.
- **User Services** that support the voice, data, video, and image users. . Examples of these services are ecommerce/ebusiness applications, voice over IP, web-enabled call centers, and unified messaging. In addition, these services can support a variety of appliances

such as IP phone, cellular phone, Fax, building temperature controller, alarm clock, facility sensor, coffee machines and the like.

Currently, most people use modems that connect to the Internet at about 56kbp. Broadband access from home promises data rates around 1Mbps (1 million bits per second). A wide range of choices, such as cable modems and DSL, are available for broadband access from home.

Besides access from home (the last mile), broadband networks are used heavily in the core network. These networks use fast packet switching systems which move millions of packets per second over fiber networks to far off places. Fast packet switching systems, also known as ***fast relays***, have intelligent end-points which can deal with errors in transmission. Thus instead of checking for errors at every switch between New York and Los Angeles, it only checks errors at NY and LA. This considerably increases the throughput (number of bits transferred per second) of the network.

The most commonly used fast packet switching systems used at present are ATM (Asynchronous Transfer Mode) and Frame Relay. ATM is at present being used more commonly. The reasons for this are, as usual, not solely technical.

A.8 Enterprise Networks – The Reality Check

The Open System Interconnection (OSI) Reference Model was introduced to provide a single network architecture standard for enterprise networks. Although the OSI Model itself has not materialized into a widely used commercial technology, it provides an excellent framework for understanding and analyzing the state of the art as well as state of the market developments in networking.

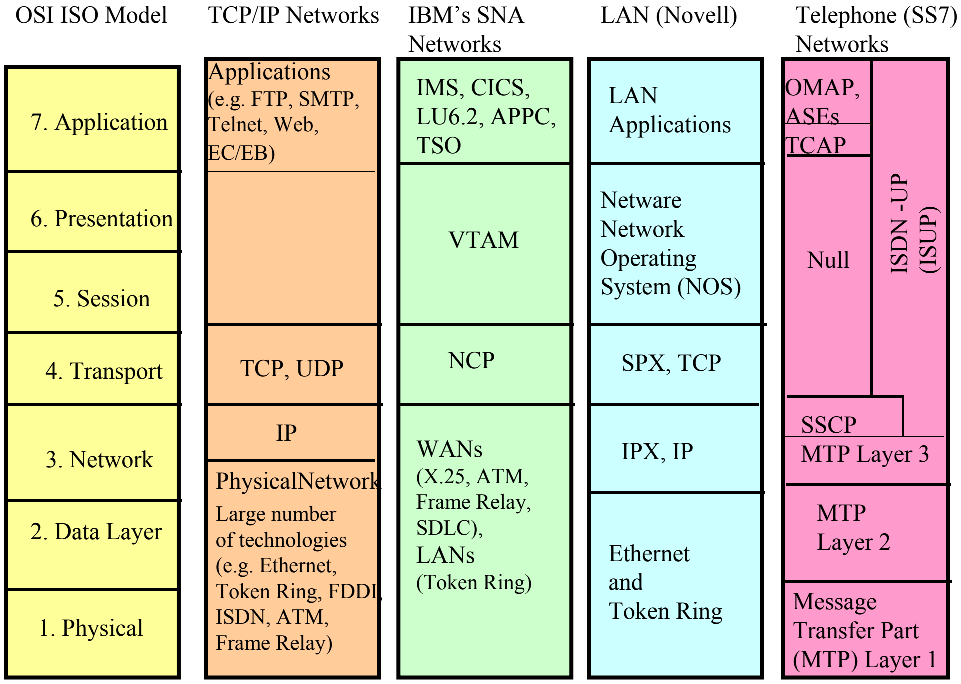

OSI ISO Model
TCP/IP Networks
IBM's SNA Networks
LAN (Novell)
Telephone (SS7) Networks
7. Application
6. Presentation
5. Session
4. Transport
3. Network
2. Data Layer
1. Physical
Applications (e.g. FTP, SMTP, Telnet, Web, EC/EB)
TCP, UDP
IP
PhysicalNetwork Large number of technologies (e.g. Ethernet, Token Ring, FDDI, ISDN, ATM, Frame Relay)
IMS, CICS, LU6.2, APPC, TSO
VTAM
NCP
WANs (X.25, ATM, Frame Relay, SDLC), LANs (Token Ring)
LAN Applications
Netware Network Operating System (NOS)
SPX, TCP
IPX, IP
Ethernet and Token Ring
OMAP, ASEs
TCAP
Null
ISDN -UP (ISUP)
SSCP
MTP Layer 3
MTP Layer 2
Message Transfer Part (MTP) Layer 1

Figure A-22: Comparative View of Network Architecture

Network architectures have been an area of considerable commercial activity since the 1970s. In particular, various "Network Protocol Stacks" have been developed by different industry segments and vendors. We give an overview of the main network stacks and use the OSI Model as a framework for comparing/contrasting network stacks such as Transmission Control Protocol/Internet Protocol (TCP/IP), IBM's System Network Architecture (SNA), LAN network architectures, and Telephony Networks. Figure A-22 shows a comparative view of the various network architectures. Details of this comparative analysis is beyond the scope of this tutorial. The two principle players in the enterprise settings (TCP/IP stack – the core Internet Technologies and the LAN stack are briefly reviewed).

A.8.1 Transmission Control Program/Internet Protocol (TCP/IP) Stack

TCP/IP is the core protocol stack of the Internet. In the late 1960s and early 1970s, the Defense Advanced Research Projects Agency (DARPA) defined a set of network standards and protocols for interconnecting many computers in the ARPANET (Advanced Research Projects Agency Network). Initially referred to as the DOD (Department of Defense) or ARPANET Protocol Suite, these protocols were intended for military networks. For example, the original DOD protocols were issued as military standards. These protocols have dramatically grown in popularity and have become the de facto standards for heterogeneous enterprise networks. At present, TCP/IP is the underlying network architecture used by the Internet.

Although many new protocols have been added to the DOD Suit, Transmission Control Protocol (TCP) and Internet Protocol (IP) are the best known DOD protocols. At present, the entire DOD/ARPANET Protocol Suite is commonly referred to as the TCP/IP Protocol Stack. Figure A-23 shows the architectural layers of TCP/IP. The TCP/IP Suite, as shown in Figure A-23, addresses the layer 3 and above issues. Let us discuss these components briefly.

Internet Protocol (IP) is roughly at layer 3 and can reside on a very large number of physical networks such as Ethernets, token rings, FDDI, ISDN, ATM, Frame Relay, ISDN, X.25, and others. In addition, IP also currently supports several emerging technologies such as cellular networks and gigabit networks. In fact, the diversity of physical networks supported by IP is a major strength of TCP/IP. IP connects hosts across multiple networks and provides a way of moving a block of data from one host machine to another through the network. This block of data is known as a datagram. The delivery of datagrams is made possible by assigning an IP address to every host in the Internet. These addresses are 32 bits in length and are commonly denoted as four decimal numbers separated by periods (e.g., 21.152.214.2). The first part of the address shows which network the host resides on, and the rest of the address shows where within that network the host can be found. IP is an unreliable (connectionless) protocol. This means that datagrams sent from one host to another may not be delivered in the order in which they were sent, may be delivered more than once, or may not be delivered at all. Higher layer protocols are expected to correct this deficiency.

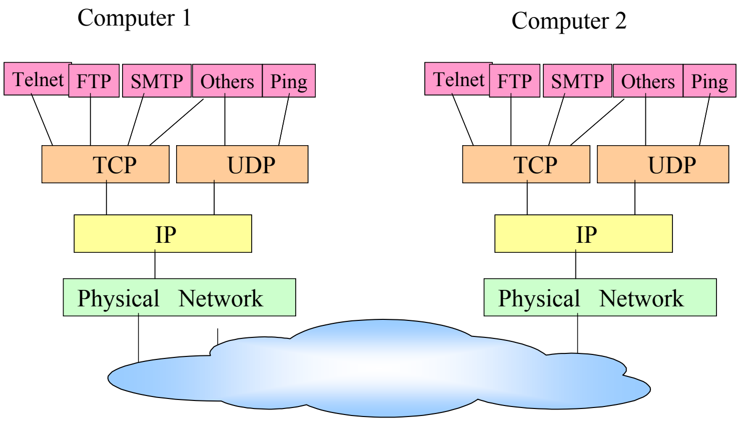


Figure A-23: The TCP/IP Network Stack

Transmission Control Protocol (TCP) resides on top of IP and is responsible for reliable transport between end-systems. TCP connects the application layer processes to IP and provides the functions that are roughly equivalent to layer 4 of the OSI Model. TCP provides a reliable, ordered connection between processes on different hosts. One host may run many processes, so a process to process connection is needed. This means that application processes can establish a TCP connection and expect that data will arrive successfully and in order. A TCP connection is essentially an error-free pipe from one host process to another. This generality allows a variety of higher layer protocols to run on top of TCP.

User Datagram Protocol (UDP) also runs on top of IP and is an alternative to using TCP. Like IP, UDP is an unreliable protocol. In fact, the major function that UDP adds to IP is a way to differentiate more than one stream of data going to or from a host (IP addresses only identify the hosts and not the processes within a host). Due to the unreliability of UDP, it is up to higher layer protocols running on top of UDP to provide reliability if it is needed.

Higher (Application) Layer Protocols run on top of TCP and UDP. The application layer of TCP/IP provides a rich set of file transfer, terminal emulation, network file access, and electronic mail services. New protocols and services for emerging technologies such as World Wide Web are also continually being added to the TCP/IP application layer. It is also possible to define private application protocols as long as both hosts agree on the protocol. The following protocols (the first three belong to the original DOD Suite) are among the best known application protocols defined in the TCP/IP Suite.

- Telnet: This protocol is used to provide terminal access to hosts and runs on top of TCP.
- File Transfer Protocol (FTP): This TCP based protocol provides a way to transfer files between hosts on the Internet.
- Simple Mail Transfer Protocol (SMTP): This TCP based protocol is the Intermet electronic mail exchange mechanism.
- Trivial File Transfer Protocol (TFTP): This UDP based protocol also transfers files between hosts, but with less functionality (e.g., no authorization mechanism). This protocol is used typically for "booting" over the network.
- Network File System (NFS) Protocol: This UDP based protocol has become a de facto standard for use in building distributed file systems through transparent access.
- Xwindow: This is a windowing system that provides uniform user views of several executing programs and processes on bit-mapped displays. Although Xwindow is supposedly network independent, it has been implemented widely on top of TCP.
- SUN Remote Procedure Call (RPC): This protocol allows programs to execute subroutines that are actually at remote sites. RPCs, like Xwindow, are supposedly

network independent but have been implemented widely on top of TCP. SUN RPC is one of the oldest RPCs. Examples of other RPCs are OSF DCE RPC and Netwise RPC.

- Domain Naming Services: This protocol defines hierarchical naming structures which are much easier to remember than the IP addresses. The naming structures define the organization type, organization name, etc.
- SNMP (Simple Network Management Protocol): This is a protocol defined for managing (monitoring and controlling) networks.
- Kerberos: This is a security authentication protocol developed at MIT.
- Time and Daytime Protocol: This provides a machine readable time and day information.

We should mention here the protocols and services being developed for the Internet World Wide Web (WWW). For example, the Web browsers, the Web servers, and the HTTP protocol used in WWW reside in the TCP/IP application layer. As the use of Internet grows, more services and protocols for the TCP/IP application layer will emerge. See Chapter 4 for more details on WWW.

Other frequently used services in TCP/IP are Ping (an echo command), Netstat (command to display the network status of the local host, e.g., active TCP connection and IP routing tables), and Finger (displays information about users of a remote host, e.g., list of all users logged on to the remote host). In addition, the OSI upper layers can be implemented on TCP/IP.

TCP/IP Berkeley Sockets. TCP/IP Berkley Sockets, simply called sockets, allow new applications and protocols to be developed on top of TCP/IP. For example, the application layer protocols developed above use sockets. Sockets are application programming interfaces (APIs) that can be used by C programs residing on two IP hosts to communicate with each other. Let us briefly review TCP/IP sockets.

A socket is an addressed endpoint of communication which conceptually resides above TCP. The addresses associated with the sockets are commonly the IP physical addresses (32 bit host number, 16 bit port number). There are several types of sockets, grouped according to the services they provide. The services include stream sockets, which provide duplex, sequenced flow of data, with no record boundaries; datagram sockets which transfer messages of different sizes in both directions and which are not promised to be reliable and sequenced; and sequenced packet sockets which are similar to stream sockets, with the difference that record boundaries are preserved. Applications in UNIX environments are written by using stream (reliable connection) or datagram mode, by using forking which allows several processes to be initiated by one process and/or by mailboxes, which allow an intermediate file for message transmission. We will discuss socket programming in Chapter 4.

Figure A-24 shows a simplified view of a TCP/IP network. The network consists of an Ethernet TCP/IP LAN, an FDDI LAN, and a frame relay TCP/IP WAN, interconnected through routers (we will discuss routers later). A Mac and a Sun-Unix machine are connected to the FDDI LAN, PCs are connected to the Ethernet LAN, and an MVS mainframe is connected to the WAN (there may be several other devices, but we are showing these just to highlight key points). Each computer ("host") on this network houses a TCP/IP stack and has an IP address, and also has been assigned a domain name (e.g., Dolorese, Pat, Warner, Gomer). This TCP/IP network is very heterogeneous (different computers, different physical networks). However, to the users of this network, it provides a set of uniform TCP/IP services such as email, file transfer, and Web (TCP/IP hides many details). This heterogeneity and support for Web is the main appeal of TCP/IP. We will discuss TCP/IP in detail in the next chapter.

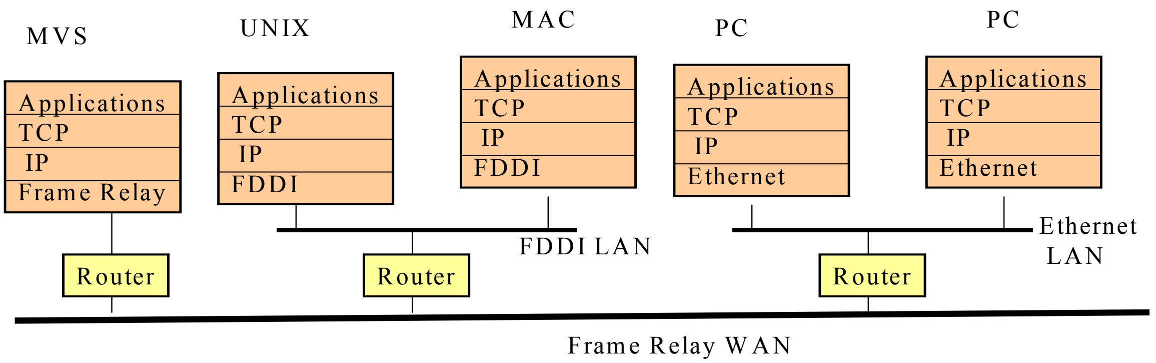


Figure A-24: A Typical TCP/IP Network

A.8.2 LAN Network Stacks and Network Operating Systems

The LAN standards have been developed by the IEEE 802 Committee. This Committee has recognized that some Local Area Networks (LANs) use only layers 1, 2, and 7 of the OSI Model. The IEEE 802 Committee has divided layers 1 and 2 into sublayers (see Figure A-25). In some LANs, layers 3, 4, 5 and 6 are null. Some functions of these layers are simplified and included in the Application, DataLink and Physical Layers of the LAN stack. However, many LANs support all 7 layers (a large number of LANs at present use the TCP/IP protocol Suite). For a discussion of the IEEE 802 standards, see Chapter 1 of this Module. Let us now discuss the higher layer issues.

The application layer of LANs usually includes LAN network operating system (NOS) software that provides services such as print servers, file servers, and user profile managers. In addition, this layer supports the workstation software which routes the workstation requests to the server. Examples of LAN NOSs are Novell NetWare, Microsoft Windows NT, IBM LANA Manager, and Banyan Vines (we will discuss these NOSs later on in this section).

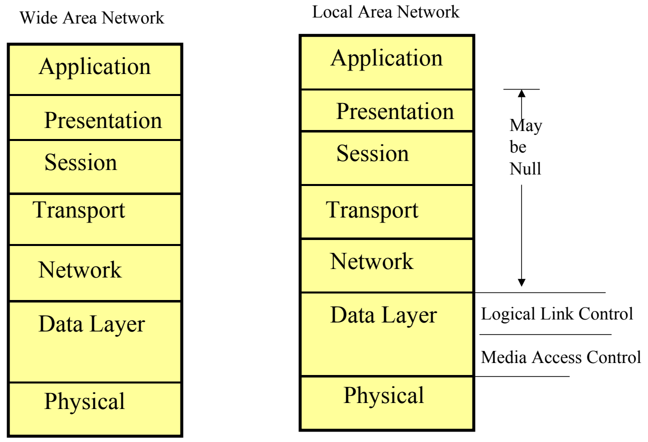


Figure A-25: LAN Layers Defined by IEEE 802

Different LANs use different layer 3 to 6 protocols (in some LANs these layers are null). The following protocols are used commonly at present:

- TCP/IP

- Novell's IPX/SPX
- IBM's NetBIOS

We have already discussed TCP/IP in a previous section and will discuss it more in the next chapter. Let us briefly review the other two.

Internetwork Packet Exchange (IPX) and Sequenced Packet Exchange (SPX) are used in the Novell NetWare LANs. IPX/SPX stack is available on many other LANs for connectivity and interoperability. IPX, a datagram delivery service, operates at the OSI Network Layer level while SPX, a connection-based service, operates at the OSI Session Layer. Technical details about IPX/SPX can be found in the Novell NetWare manuals and tutorials.

NetBIOS (Network Basic Input Output System) is a Session Layer programming interface for IBM PC LANs. NetBIOS is widely used in IBM PC LANS and is currently supported by most major LAN vendors. For example, NetBIOS stack can run in the Novell NetWare LANs. Basically, NetBIOS provides a programming interface at the OSI Session Layer level for sending/receiving data in LAN environments. Technical details about NetBIOS can be found in the IBM Technical Reference Manual, SC30-3383-2.

An area of active standards development is concerned with the wireless LANs. For example, wireless LANs at 2 Mbps are commercially available. A large installed base of wired (copper or fiber optic) LANs currently operates at 10 Mbps. The wireless LANs are also being aimed at this speed, at least, for an equivalent performance. We have discussed Wireless LANs previously in the section on Emerging Technologies.

Figure A-26 shows a conceptual view of a LAN configuration. Each LAN has one or more "LAN Servers" which provide services such as print services, file services, etc. The LAN workstations operate as clients to the LAN servers. The network operating system (NOS) consists of two parts: a NOS server software that is installed on each LAN sever and a NOS client software that is installed on each LAN workstation. The NOS server operating system performs a variety of functions such as managing client sessions, initiating printer operations, managing server shared disk, etc. The NOS client software is in principle much simpler and mainly directs the user requests to the appropriate servers. Some NOS clients are called "redirectors".

Many LAN NOSs have been developed and marketed by different software vendors since the mid 1980s. At present, the main contenders are Novell NetWare, Microsoft's Windows NT, IBM's LAN Manager, and Banyan Vines. Detailed discussion of NOSs is beyond the scope of this tutorial. Many articles in the trade magazines compare and contrast these and other NOSs on a regular basis. An example of such an analysis can be found in [Johnson, J. 1995]. For additional discussion of LANs, see Chapter 1.

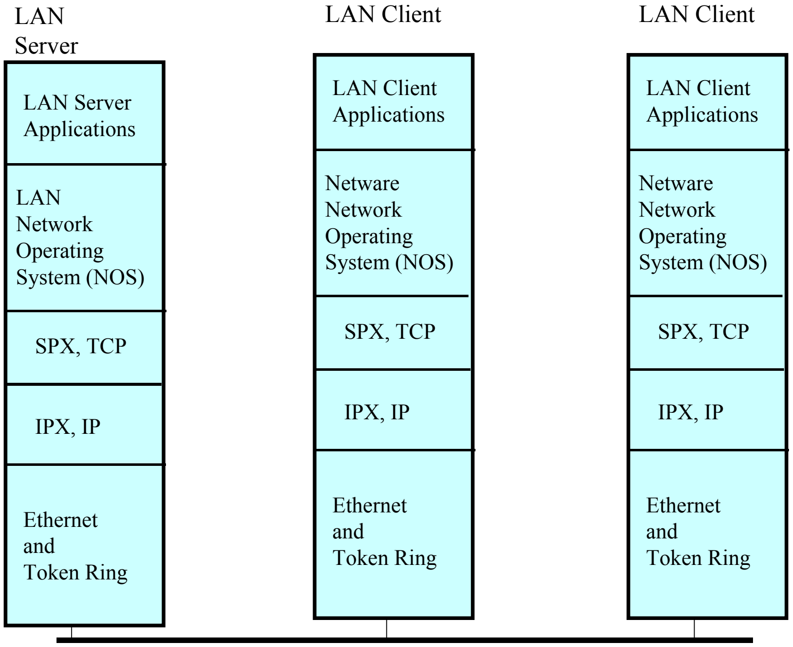


Figure A-26: A Conceptual LAN Configuration

A.9 Internet, Intranet and Extranets

A.9.1 What is Internet?

Technically speaking*, **Internet*** is a network based on the TCP/IP protocol stack. At present, the term Internet is used to refer to a large collection of TCP/IP networks that are tied together through network interconnectivity devices such as routers and gateways. The TCP/IP (Transmission Control Protocol/Internet Protocol), briefly introduced in chapter 3, was developed in the late 1960s and early 1970s by the Defense Advanced Research Projects Agency (DARPA). TCP/IP was developed for interconnecting many computers in the ARPANET (Advanced Research Projects Agency Network). ARPANET initially consisted of five protocols (indicated with * in the following list) that have been augmented with other key protocols (see Figure A-27):

- *Internet Protocol (IP) for interconnecting and routing messages to a large number of physical networks
- *Transmission Control Protocol (TCP) for reliable information transfer
- User Datagram Program (UDP) for fast, but unreliable, information transfer
- *File Transfer Protocol (FTP) for file transfer
- *Simplified Mail Transfer Protocol (SMTP) for email
- *Terminal emulator (Telnet) for terminal emulation
- Hypertext Transfer Protocol (HTTP) for Web applications
- Real Time Protocol (RTP) for audio and video applications

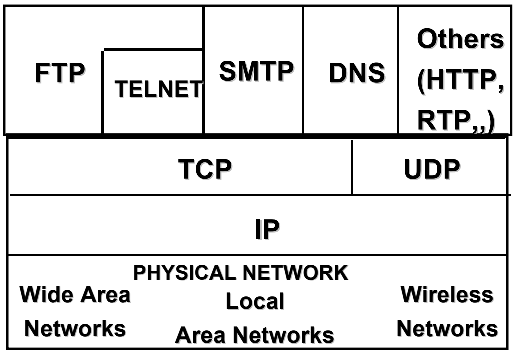


Figure A-27: The IP Stack – Foundation of Internet

Although, the Internet at present uses TCP (i.e., higher level protocols and applications are based on TCP), this may not be true in the future since some future (especially real time) applications may be built directly on IP or newer alternatives to TCP. The main strength of IP is that it runs on top of a very diverse array of physical networks (wide area, local area, wireless). In fact, IP supports almost all current physical network technologies and is expected to support most of the future high speed networks. We thus will use the following simple definition of the Internet:

Definition: Internet is a network of networks that is supported by the Internet Protocol (IP)

What does this mean? Basically it says that you need to have an IP network (or a gateway that translates to IP) to join the Internet. Once you have an IP network, then you can run almost any physical network under it and take advantage of voice, data, or video applications for your ebusiness that run on top of IP (this is also implied in **Error! Reference source not found.**).

On the popular media side, the term cyberspace, first introduced through a science fiction book by [Gibson 1984], has been permanently transferred to our vocabulary. It represents thousands of computers and computer resources around the globe interconnected through the Internet. At present, the term Internet is used to symbolize a Public Internet that is not owned by any single entity -- it consists of many independent IP networks that are tied together loosely.

Initially, the public Internet was used to tie different university networks together. With time, several commercial and private networks have joined the public Internet. The computers on the public Internet have publicly known Internet Protocol (IP) addresses that are used to exchange information over the public Internet (we will discuss IP addresses later). The public Internet at present consists of millions of computers (PCs, Macs, Sun workstations, HP systems, IBM mainframes) that are interconnected through thousands of networks that use different underlying network technologies (ATMs, frame relays, Ethernet LANs, and wireless networks) in different parts of the world. All these computers and networks are tied together through a global IP network (see Figure A-28).

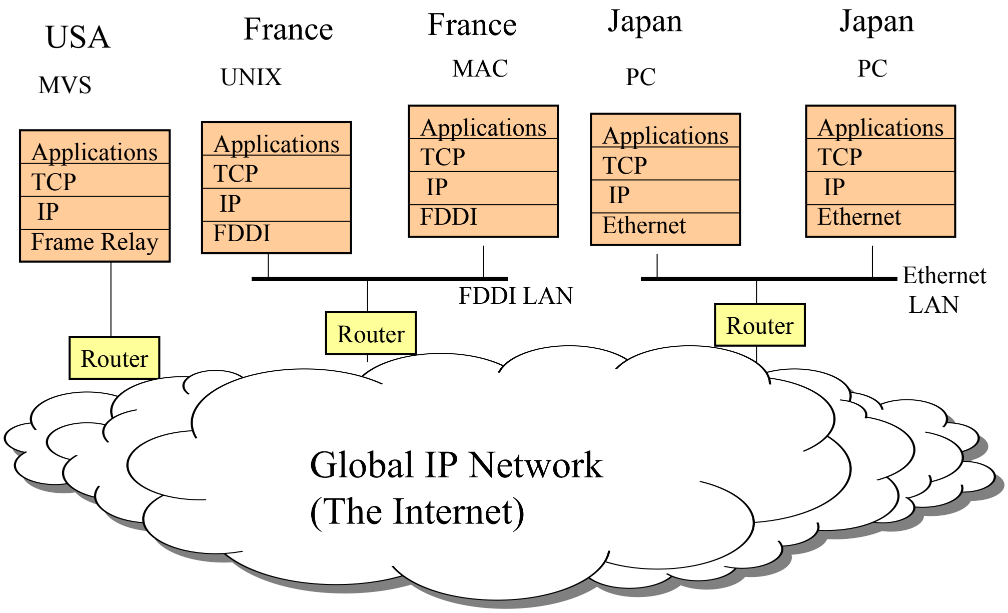


Figure A-28: Public Internet – A Global IP Network

A.9.2 What are Intranets and Extranets?

Intranets, also known as private Internets, are the IP networks that are used by corporations for their own business, especially for exploiting Web technologies. Technically, an Intranet uses the same technology as the public Internet -- it is only smaller and privately owned (the physical network below IP is privately owned) and thus hopefully better controlled and more secure. Thus any applications and services that are available on the public Internet are also available on the Intranets. This is an important point for Web because many companies are using Web technologies on their Intranets for internal applications (e.g., employee information systems).

Extranets, also known as community of interest networks**,** are IP networks that are jointly owned by corporations for conducting secure business processing. These networks use the same Internet technologies, however, the physical network is collectivity owned by corporations to meet the security and reliability requirements imposed by the owners. An example of Extranet is the Automotive Network eXchange (ANX) network formed by manufacturing corporations (GM, Ford, Chrysler, and others). Extranet was formed by the Automotive Industry Action Group (AIAG) to provide a common communication infrastructure among automotive Trading Partners. ANX is intended for North America initially with plan to expand worldwide with over 3000 trading partners. The drivers for ANX are control communications cost with predictable service quality, support of a common set of applications (e.g., EDI, database lookup, web, email, and Computer-Aided-Design [CAD] file transfer), and facilitate introduction of new applications (e.g., videoconferencing, interactive CAD) by using the Internet technology. The architecture of ANX is shown in Figure A-29.

Unless otherwise indicated, the discussion in this chapter is oriented towards the public Internet although most concepts apply to the Intranets and Extranets

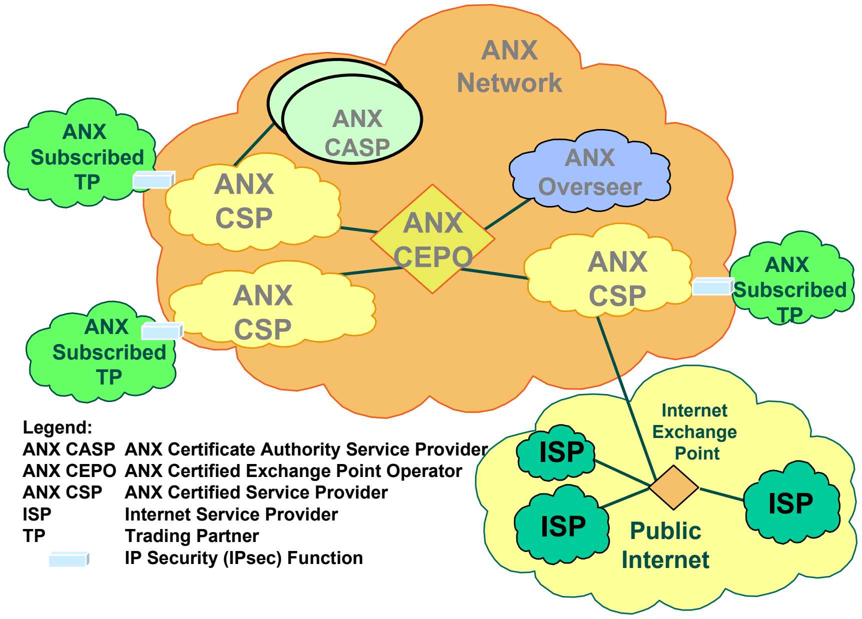

ANX Network
ANX CASP
ANX Subscribed TP
ANX CSP
ANX Overseer
ANX CEPO
ANX CSP
ANX Subscribed TP
ANX CSP
ANX Subscribed TP
Internet Exchange Point
ISP
ISP
ISP
Public Internet
Legend:
ANX CASP ANX Certificate Authority Service Provider
ANX CEPO ANX Certified Exchange Point Operator
ANX CSP ANX Certified Service Provider
ISP Internet Service Provider
TP Trading Partner
▭ IP Security (IPsec) Function

Figure A-29: ANX Extranet Architecture .

A.9.3 User View of Internet

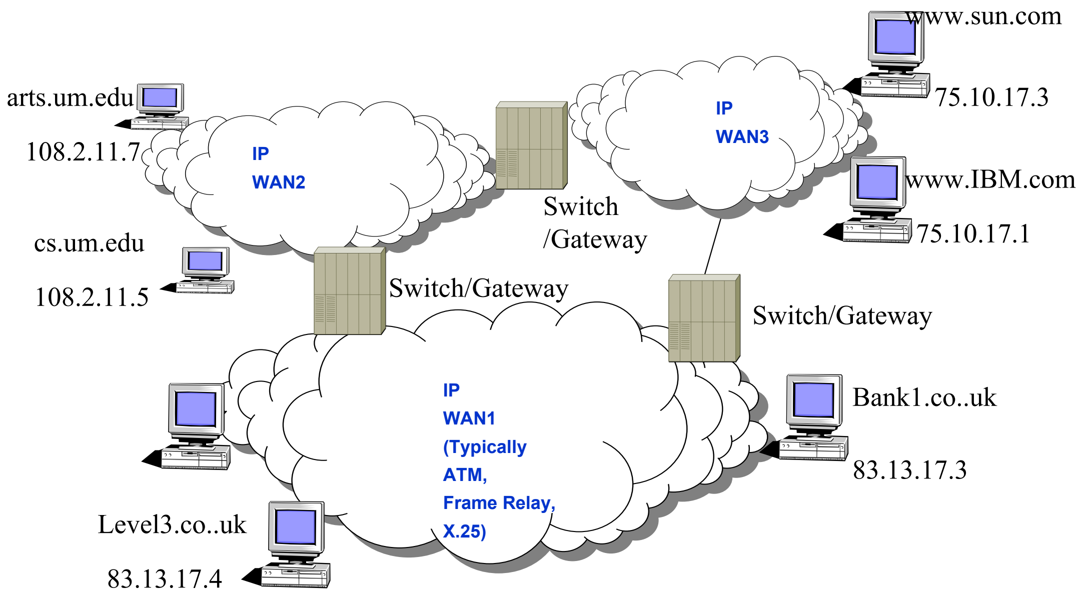
Before getting into the IP networking details, let us briefly review how the Internet is viewed by the users.

From an end user's point of view, Domain Naming Service (DNS) is of key importance because DNS helps the users to locate different resources used in the Internet. DNS defines hierarchical naming structures which are much easier to remember than the IP addresses. For example, the machine with an IP address of 135.25.7.82 may have a domain name of shoeshop.com. A user "mills" may have an email address mills@shoeshop.com. The DNS naming structures define the organization type, organization name, etc. The last word in the domain name identifies an organization type or a country. Consider, for example, the following domain names:

telcordia.com = commercial company Telcordia
ibm.com = commercial company IBM
um.edu = educational institution University of Michigan
omg.org = organization OMG (Object Management Group)
waterloo.ca = waterloo university in canada
lancs.ac.uk = Lancaster University in UK
ansa.co.uk = ANSA consortium in UK
iona.ie = Iona Corporation in Ireland

The Internet uses a large number of domain name servers that translate domain names to IP addresses (the IP routers only understand IP addresses). Domain names are used in the Internet as well as the Web.

Figure A-30 shows a conceptual and partial view of Internet. This Internet shows three networks (a university network with two computers, a commercial company network, and a network in UK). Each computer ("host") on this network has an IP address and also has been assigned a domain name. Internet is very heterogeneous (i.e., different computers, different physical networks.) However, to the users of this network, it provides a set of uniform TCP/IP services (TCP/IP hides many details). We will use this simple Internet to illustrate the key Internet capabilities.



•DNS (Domain Name Services) translates cs.um.edu to 108.2.11.5
•Telnet cs.um.edu = Telnet 108.2.11.5
•FTP cs.um.edu = FTP 108.2.11.5

Figure A-30: Partial View of Internet

Since the Internet is based on TCP/IP, the applications and services provided by TCP are also available on the Internet. From an end-user point of view, the following services have been, and still are, used very heavily on the Internet:

- Email
- Telnet
- FTP

Electronic mail on the Internet is based on the **Simple Mail Transfer Protocol (SMTP)**. This TCP based protocol is the Internet electronic mail exchange mechanism. Email is still one of the most heavily used services in the Internet. Users on the Internet have email addresses such as johnm@cs.um.edu, hevner@sun.com and howard@bank1.co.uk.

Terminal emulation is used to remotely logon to other machines. **Telnet** is used to provide terminal access to hosts and runs on top of TCP. Let us assume that a user "joe" on cs.um.edu needs to remotely logon to the bank1.co.uk machine to run a program "directory". The user would use the following steps (the steps are explained through comments in /* */):

```
> telnet bank1.co.uk              /* invoke Telnet. Could have typed " telnet     85.13.17.3".*/
bank1> enter logon: joe        /* prompt from  bank1 for logon ID. joe is ID */
bank1> password: xxxx           /* prompt from bank1 for password  */
bank1>  directory            /*  run the program "directory"   */
bank1> exit                          /* quit  telnet */
```

File transfer is used for bulk of data transfer over the Internet. The **File Transfer Protocol (FTP)** provides a way to transfer files between hosts on the Internet. Let us assume that a user "garner" on "sun.com" needs to transfer a file from the host arts.um.edu.The following steps would be used (the steps are explained through comments in /* */):

```
> ftp arts.um.edu                 /* invoke FTP. Could have typed " ftp 102.52.10.7"*/
arts> enter logon: garner       /* prompt from arts.um for logon ID. garner is ID */
arts> password: xxxx          /* prompt from arts.um for password  */
arts>  get file1  file2        /*  FTP file transfer command  */
arts> exit (or quit)                   /* quit  FTP  */
```

For many years, Internet had been used mainly by researchers, teachers, scientists, students, and programmers to transfer files and send/receive electronic mail. These users relied on text-based commands to do their job. WWW is a set of services that run on top of the Internet. The two main features of WWW are use of GUI and hypertext to make the life of Internet users easy and fun. We will discuss WWW in the next section in more detail.

We should mention that the users access the Internet either directly or indirectly. ***Direct Internet users*** reside on the machines that have IP addresses while ***indirect Internet users*** remotely logon to the machines with IP addresses. For example, America Online is an Internet Access Provider that actually have machines with IP addresses (direct access). If you subscribe to America Online, then you dial into an America Online machine (i.e., you are indirectly accessing the Internet).

Figure A-31 shows an example of how a student from Drexel university in Philadelphia can exchange email with another student in Izmur (Turkey). It shows the various networks, gateways, and email servers that participate in email exchange.

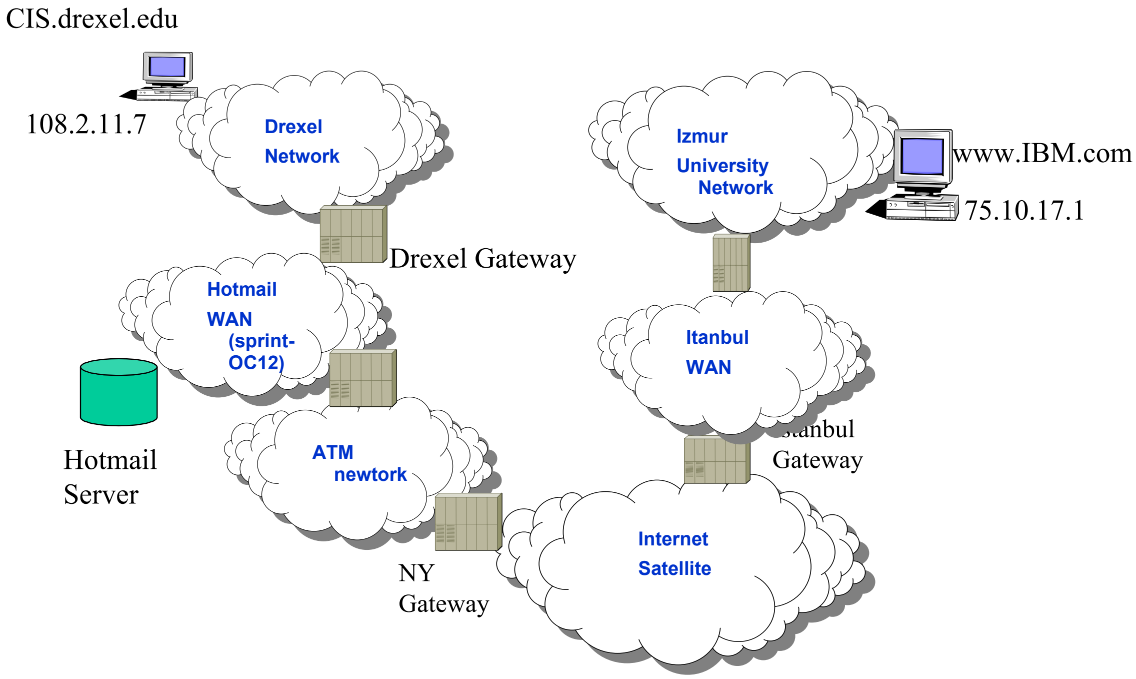


Figure A-31: An example of IP connectivity from Drexel (Philadelphia) to Izmur (Turkey)

Internet Role Players

- Different individuals, groups and organizations play different roles in the Internet. To illustrate these roles, let us envision the Internet as an electronic shopping mall. Then we can discuss the following roles:
- Internet users are the people who visit the shopping mall (i.e., logon to the Internet). The Internet users are essentially the consumers of the services provided by the Internet.
- Content providers are the merchants (individuals, groups or organizations) that provide the products in the shopping mall (i.e., resources available on the Internet). You can think of these content providers as the merchants in the shopping mall.
- Internet access providers (IAPs) are the organizations that facilitate your access to the shopping mall (i.e., give you a communication line and an access port on the Internet). You can think of IAPs as the local authorities that provide you with roads and signs to get you to the shopping malls.
- **Internet** service **providers (ISPs)** are the individuals and organizations that help the content providers set up their shops in the shopping mall (i.e., help in building Web sites). Many small content providers seek the help of ISPs to set up Web servers with appropriate security and backup/recovery.

A.9.4 The Evolution of IP Stack

The IP stack, in particular the TCP/IP Protocol Suite, is by far the most popular network architecture at present. It has practically squeezed other popular network stacks such as IBM's SNA almost out of existence). Why and how did this happen? Here are some thoughts:

Initial Phase (1969 to 1990). As stated previously, the TCP/IP Protocol Suite was developed in the late 1960s and early 1970s by the Defense Advanced Research Projects Agency (DARPA). TCP/IP was developed for interconnecting computers in the ARPANET and initially consisted of five protocols (IP, TCP, FTP, TELNET, SMTP). The TCP/IP Suite matured due to the experience gained in the ARPANET project which resulted in the Internet Supernetwork - one of the largest heterogeneous networks in the world. ARPANET itself was discontinued in 1990, but it supported the evolution of different types and numbers of computers over a long time. This capability of TCP/IP to support heterogeneous systems became extremely important because in this time period proprietary network architecture stacks such as IBM's SNA started running into difficulty in supporting widely disparate systems.

Second Phase (1990 to 2000). Since the early 1990s, TCP/IP and Internet have grown dramatically in popularity. The major development, of course, is the introduction of World Wide Web that made TCP/IP indispensable because TCP/IP is the underlying network stack for the Web. The popularity of Web has given a tremendous boost to TCP/IP. In this time period, several other developments are worth noting:

- TCP/IP matured to support the enterprise networks. TCP/IP matured enough for corporate WANs (TCP/IP operates on top of X.25, Frame Relay and ATM communication technologies) as well as corporate LANs (TCP/IP operates on top of Ethernet, Token Ring, and FDDI communication technologies). Many organizations are running enterprise wide networks that are entirely based on TCP/IP. Many others are transitioning to TCP/IP
- The TCP/IP Suite became available on almost all computing systems including microcomputers, minicomputers, and mainframes (it is estimated that there are more than 300 TCP/IP vendors at present). For example, TCP/IP can be used to transfer files between IBM mainframes, PCs, SUN, HP, Macintosh and several other machines. In addition, TCP/IP is closely associated with the UNIX operating system and most UNIX vendors support TCP/IP. For example, the SUN Microsystems UNIX and the Hewlett-Packard HP-UX systems include TCP/IP as part of the basic software package. In addition, Microsoft is shipping TCP/IP with Windows platforms (known as "server pack").
- IP, the lowest protocol in this Suite, started supporting a variety of physical networks such as Ethernets, FDDI based fiber optic LANs, dial up lines, X.25 based packet switching networks or ISDN digital networks. In addition, IP support exists for almost all emerging communications technologies such as cellular networks, ATM, frame relay, and SMDS. The internet technology used by IP allows many computers to communicate across many networks.
- TCP, the layer above IP, started supporting a very wide variety of higher level (application) protocols which allow users to run voice, data and video applications in addition to the email and Web applications. Developments to support voice over IP (VOIP) and video conferencing are worth noting. In addition, distributed middleware such as CORBA are built on top of IP.
- Network management has improved. Simplified Network Management Protocol (SNMP), a TCP/IP based network management protocol, has become widely accepted by vendors and users for network management in heterogeneous networks. Due to the growing importance of network management, SNMP has furthered the popularity of TCP/IP Suite as the glue between disparate networks and devices.

Phase 3 (2001+). By now, TCP/IP has become the de facto standard for interconnecting heterogeneous computer systems. Due to its popularity, the TCP/IP Protocol Suite will continue to evolve. In particular, the convergence of voice, data, and video applications on the same network technology (IP) will be a major area of work. Figure A-32 shows a general view of the Internet that supports data, voice and video applications through a variety of protocols (including H.323 and SIP for voice over IP). This represents an "IP dial tone" which means that an IP address allows you to access data, voice, and/or video applications anywhere in the world. For example, when you pick up a phone you get a dial tone that indicates that you can call anywhere in the world as long as you know the telephone number. Similarly IP dialtone indicates that once you are connected to the Internet, you can communicate with anyone else on the Internet (barring security) through data, voice, or video applications.

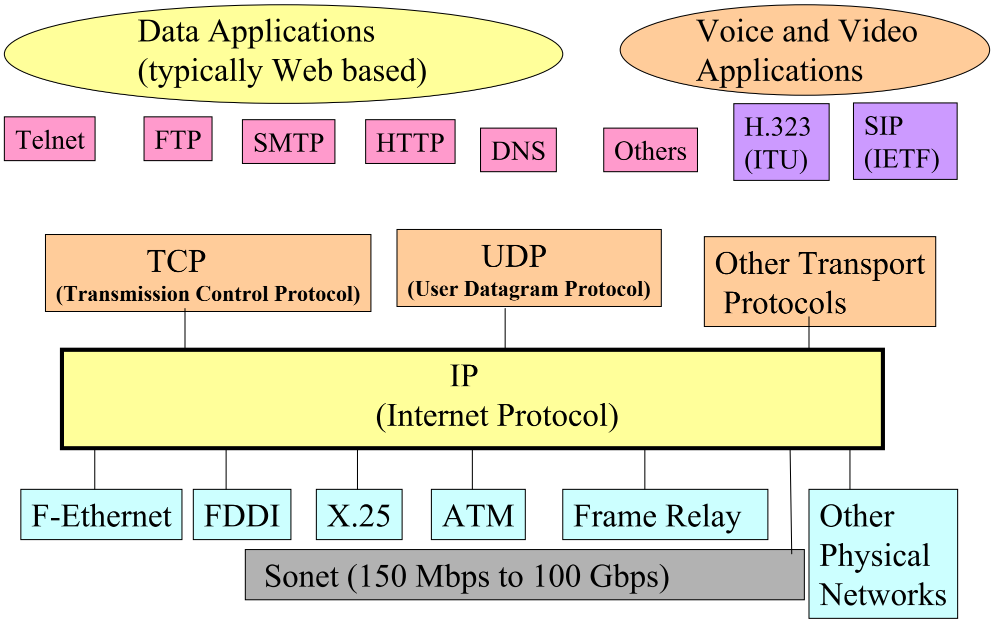


Figure A-32: IP Dial Tone (Running Everything on IP)

However, IP is beginning to show signs of age (remember TCP/IP was introduced around 1969). In particular, IP is running out of addresses. To address this, and other related problems, IPng (IP next generation) is being developed. In addition, the US Government has started a Next Generation Internet (NGI) initiative that will support innovative Internet applications over very fast and highly reliable networks.

A.9.5 Summary of Developments in TCP/IP Stack

Figure A-33 summarizes the developments in the TCP/IP stack since the mid 1990s (after the Internet explosion due to the introduction of Web). The diagram also compares the TCP/IP stack with the ISO/OSI model. The key developments can be discussed in the following areas (going from bottom to top in the diagram):

- Support of IP over emerging networks and network technologies. Examples are:
  - IP support over fast packet switching systems such as ATM and Frame Relay

  - Direct IP support over optic networks such as Sonet and WDM. This work, currently still under development, bypasses the packet switching systems and sends IP packets directly over the raw fiber networks.
  - IP support over wireless networks at wireless WANs (e.g., cellular networks) as well as LANs (e.g., Bluetooth and Wireless Ethernet).
- Developments in the IP layer itself. These include:
  - IPv6
  - Wireless IP - being developed specifically to handle location issues for mobile users (i.e., how to update IP addresses of a user who is roaming around the network).

- Developments in the transport layer. These include:
  - Real Time Protocols (RTP) to support real time processing such as video conferencing
  - RSVP (Reservation Protocols) being developed to support quality of service (QoS) requirement of guaranteed delivery.
- Developments in TCP/IP security. These include:
  - SSL (secure socket layer)
  - IP-sec (IP security)
  - S-HTTP (secure HTTP)
- Developments in middleware and application protocols that run on top of TCP/IP. Examples fall into two broad areas:
  - Protocols for data applications. Major examples are: HTTP for web applications, CORBA-IIOP (Internet Interoperable Protocol) for distributed object applications, DCOM for Microsoft office distributed applications, and ODBC-JDBC drivers for remote database (SQL) access over TCP.
  - Protocols for voice over IP applications. The two key contenders are SIP (session initiation protocol) and H.323.

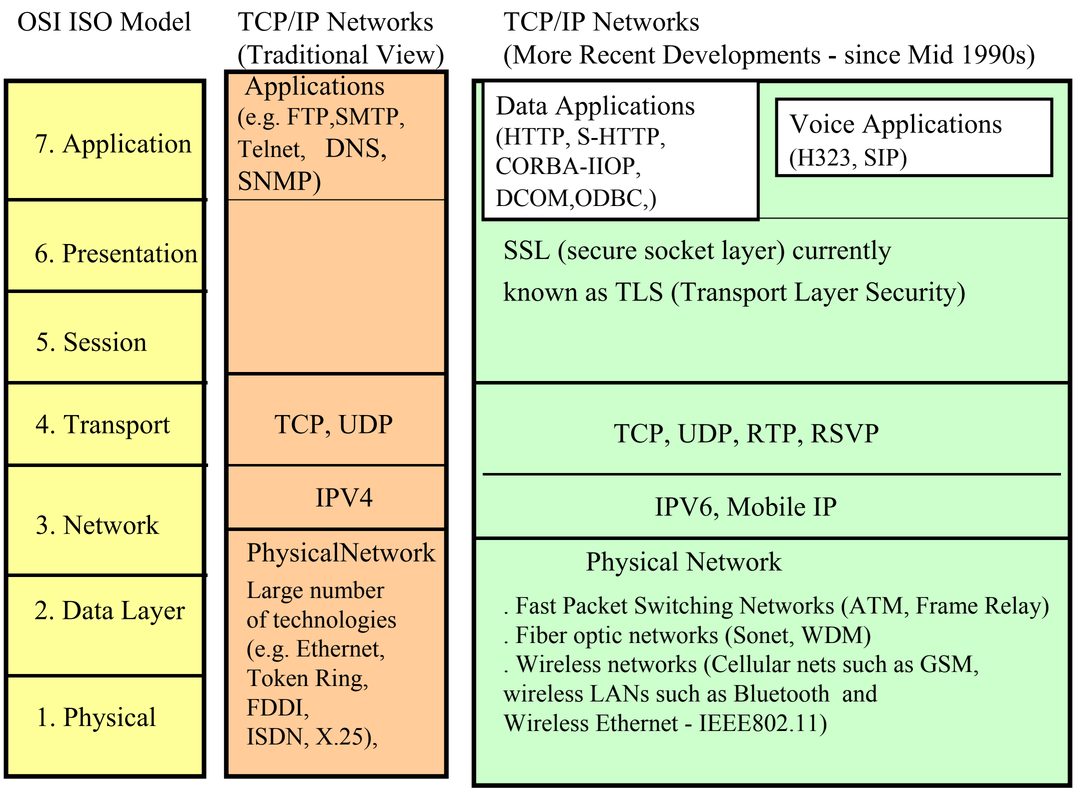

OSI ISO Model
TCP/IP Networks (Traditional View)
TCP/IP Networks (More Recent Developments - since Mid 1990s)
7. Application
6. Presentation
5. Session
4. Transport
3. Network
2. Data Layer
1. Physical
Applications (e.g. FTP,SMTP, Telnet, DNS, SNMP)
TCP, UDP
IPV4
PhysicalNetwork
Large number of technologies (e.g. Ethernet, Token Ring, FDDI, ISDN, X.25),
Data Applications (HTTP, S-HTTP, CORBA-IIOP, DCOM,ODBC,)
Voice Applications (H323, SIP)
SSL (secure socket layer) currently known as TLS (Transport Layer Security)
TCP, UDP, RTP, RSVP
IPV6, Mobile IP
Physical Network
. Fast Packet Switching Networks (ATM, Frame Relay)
. Fiber optic networks (Sonet, WDM)
. Wireless networks (Cellular nets such as GSM, wireless LANs such as Bluetooth and Wireless Ethernet - IEEE802.11)

Figure A-33: Developments in TCP/IP

A.9.6 Internet Standards Bodies: IETF and IAB

The Internet Engineering Task Force (IETF) and the Internet Architecture Board (IAB), formerly known as Internet Activities Board, are the two key players in developing and enforcing Internet standards.

The main custodian of Internet developments is the ***Internet Engineering Task Force (IETF)*** that takes care of the IP stack. IETF is a loosely self-organized group of people who make technical contributions to the engineering and evolution of the Internet and its technologies. It is the principal body devoted to the development of new Internet standard specifications. The IETF is made up of volunteers who meet three times a year to fulfill the IETF mission. Unlike some other standards bodies (e.g., OMG, W3C), there is no membership in the IETF. Anyone may register for and attend any meeting. Serious folks join the IETF or working group mailing lists. The IETF is divided into eight functional areas: Applications, Internet, IP Next Generation, Network Management, Operational Requirements, Routing, Security, and Transport and User Services. Each area has several working groups. A working group is a group of people who work under a charter to achieve a certain goal such as creation of an Informational document, the creation of a protocol specification, or the resolution of problems in the Internet. Most working groups have a finite lifetime (i.e., once a working group has achieved its goal, it disbands). For information about the IETF activities, visit the IETF web site (www.ietf.org).

The ***Internet Architecture Board (IAB)*** provides a framework and focus for most of the research and development of Internet protocols. The IAB is a technical advisory group and is chartered to provide oversight of the architecture of the Internet and its protocols. IAB was originally organized by DARPA to promote R&D. At present, IAB has evolved into an autonomous organization consisting of many task forces with various charters that works closely with IETF. A series of technical reports, called Internet Request for Comments (RFC), describe the protocol proposals and standards. An RFC is a formal document which can become a standard. For example, almost all of the current TCP/IP protocols are specified as RFCs. For details, see (www.iab.org).

To fully understand the interrelationships between IETF and IAB, it is best to understand the overall structure in which the IETF and IAB reside. There are four groups in the structure: the ISOC (Internet Society), the IAB, the IESG (Internet Engineering Steering Group) and the IETF. For a full appreciation of the hierarchy with all the voting procedures, appointments, and appeal processes, visit the IETF site (www.ietf.org) and click on Tao.

In addition to the four Internet bodies, the role of W3C and ITU-T should be mentioned. The ***World Wide Web Consortium (W3C)*** is primarily responsible for development of the Web technologies that reside above the Internet stack. We will discuss W3C in more detail in Chapter 7. Visit (www.w3.org) for details. The ***International Telecom Union (ITU)*** is primarily interested in developing voice applications. ITU is playing a key role in convergence of voice and data applications. Visit the Web site (www.itu.org).

A.10 Scan of the Internet Business: ISPs, NSPs, and VPNs

A.10.1 ISPs and NSPs

Simply stated, a service provider offers you a set of services based on an agreed upon contract. The services can be business services such as physical site security or technical such as Web hosting. Different service provider models, centered around the Internet, are becoming popular to facilitate outsourcing. For example, businesses and consumers can rent services from service providers such as the following (see Figure A-34):

- *Network service providers (NSPs)* that provide the network "pipe" (end to end network communication and routing services) for Ebusiness. Examples of NSPs are the Telecommmunications companies that include a variety of local exchange carriers and long distance carriers.
- *Internet Service Providers (ISPs)* that support Web services and provide access to the public Internet. America Online is a well known example of ISPs.
- *Platform Service Provider (PSP)* that provide the platform services (computing hardware, operating systems, basic middleware) needed to support ecommerce or other applications. PSPs in essence are similar to the old "computing centers" that provided the computing hardware/software for business applications. Due to the emphasis on ecommerce, PSPs are also referred to as CSP (commerce service providers). Examples of PSP/CSPs are Rightworks.com, CommerceOne, and Ariba.net.
- *Application Service Providers (ASPs)* host application components (mostly business aware) that clients use over a wide area network. A very wide range of ASPs have emerged in the recent years with services that range from payroll to inventory control. For example, major software vendors such as SAP, Oracle, and Peoplesoft are becoming ASPs. We will discuss ASPS in more detail later.
- *Business service providers (BSPs)* that provide business services such as mail delivery, customer support, and building security.

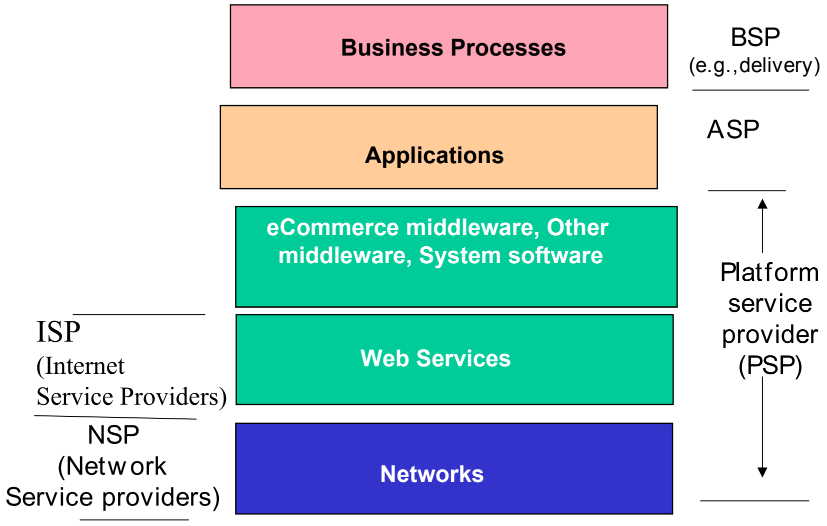

Business Processes
BSP
(e.g.,delivery)
Applications
ASP
eCommerce middleware, Other middleware, System software
Platform service provider (PSP)
ISP
(Internet Service Providers)
Web Services
NSP
(Network Service providers)
Networks

Figure A-34: Service Providers

.

Let us consider Network Service Providers (NSPs) and the Internet Service Providers (ISPs) in some detail.

Network Service Providers (NSPs), also known as Internet Access Providers (IAPs), are the organizations that provide the physical network, i.e., give you a communication line and an access port on the Internet. You can think of IAPs as the local authorities that provide you with roads and signs to get you to the shopping malls. For dialup users, an NSP provides several POPs (points of presence) that the users can dial into as a local call. An example of NSP is UUNET that has POPs around the globe. I am a regular user of UUNET. A GUI shows me the phone number of the nearest POP (I just type in the name and country of the city). When I travel (I have traveled from South Africa to Budapest), I quickly locate a UUNET number and make a local call to reach my office computers.

Internet service providers (ISPs) go beyond the network pipe and offer Internet services such as email, web hosting and web surfing. Many ISPs also provide help in building Web sites. Many small web content providers seek the help of ISPs to set up Web servers with appropriate security and backup/recovery. An ISP may rent NSP, i.e., use an existing POP, or own a network and thus may own POPs. Examples of ISPs are America Online and Asia online. Figure A-35 shows a conceptual view and Figure A-36 shows a physical view of ISPs and NSPs.

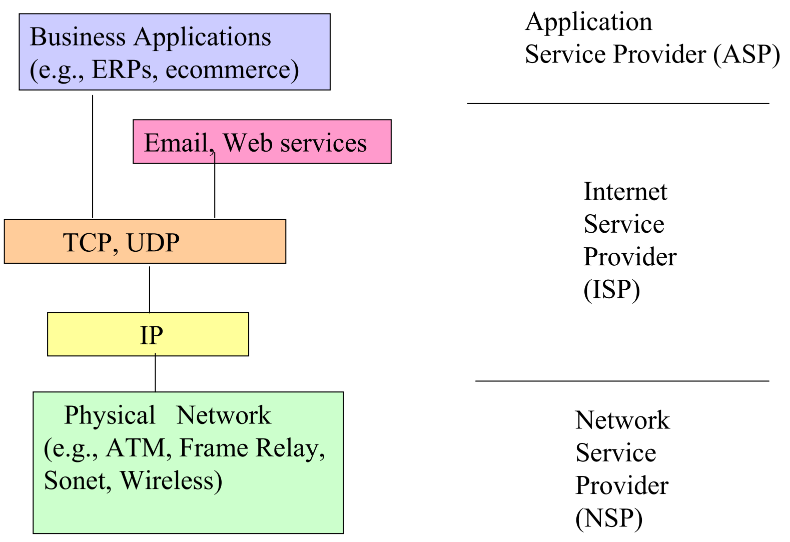

Business Applications
(e.g., ERPs, ecommerce)
Email, Web services
TCP, UDP
IP
Physical Network
(e.g., ATM, Frame Relay, Sonet, Wireless)
Application Service Provider (ASP)
Internet Service Provider (ISP)
Network Service Provider (NSP)

Figure A-35: A Conceptual View of ISPs and NSPs

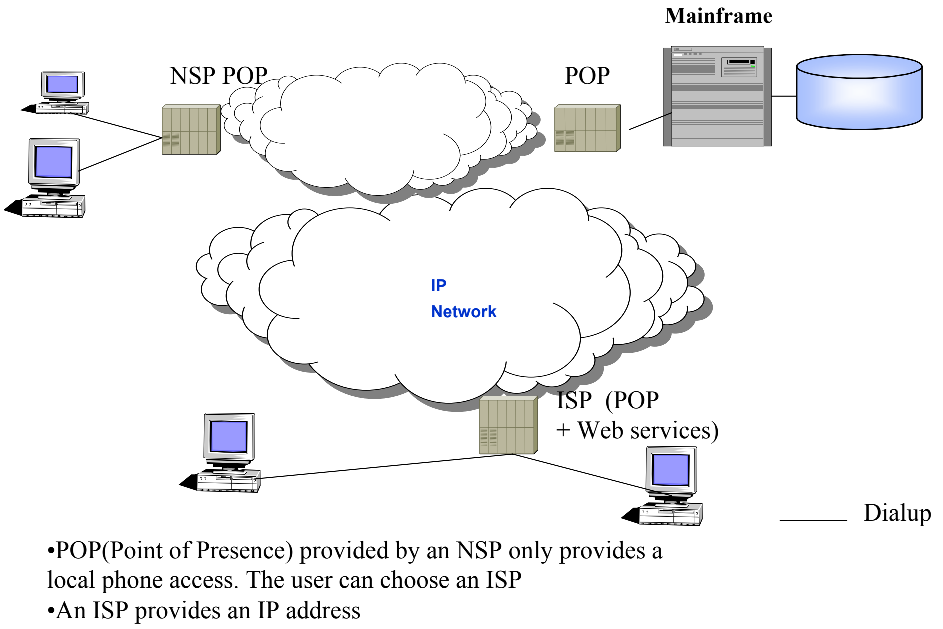

Mainframe
NSP POP
POP
IP
Network
ISP (POP
+ Web services)
Dialup
•POP(Point of Presence) provided by an NSP only provides a local phone access. The user can choose an ISP
•An ISP provides an IP address

Figure A-36: Physical View of ISPs and NSPs

A.10.2 Virtual Private Network (VPN)

Simply stated, a VPN provides dedicated, secure paths, or ***tunnels***, over a network that is shared by other users. VPN networks consist of authenticated and encrypted tunnels over a shared data network (typically, an IP network). The tunnels are set up between a ***network access point (NAP)*** and a tunnel terminating device on the destination network. Shiva Corporation's LanRover Access Switch® is an example of a NAP. A NAP encapsulates packets sent by the mobile user so that the data travels securely over the shared network. A sample VPN is shown in Figure A-37. Thus a corporation can use a shared corporate network for secure remote communications with dedicated leased lines -- a very expensive solution.

NAPs use protocols such as Point-to-Point Tunneling Protocol (PPTP) and Layer Two Forwarding (L2F) to encapsulate the data for Internet travel. PPTP is geared toward ISPs (Internet Service Providers) and has provisions for call origination and flow control, while L2F has less overhead and is suited for managed networks. The best features of both protocols are being combined into a new protocol called Layer Two Tunneling Protocol (L2TP). L2TP has provisions for flow control, call origination and secure tunnels across the Internet.The current protocols such as L2F and PPTP, and future ones such as L2TP, do not preclude the use of a Point-to-Point Protocol (PPP) client from having the tunnel-originating functionality embedded in it directly.

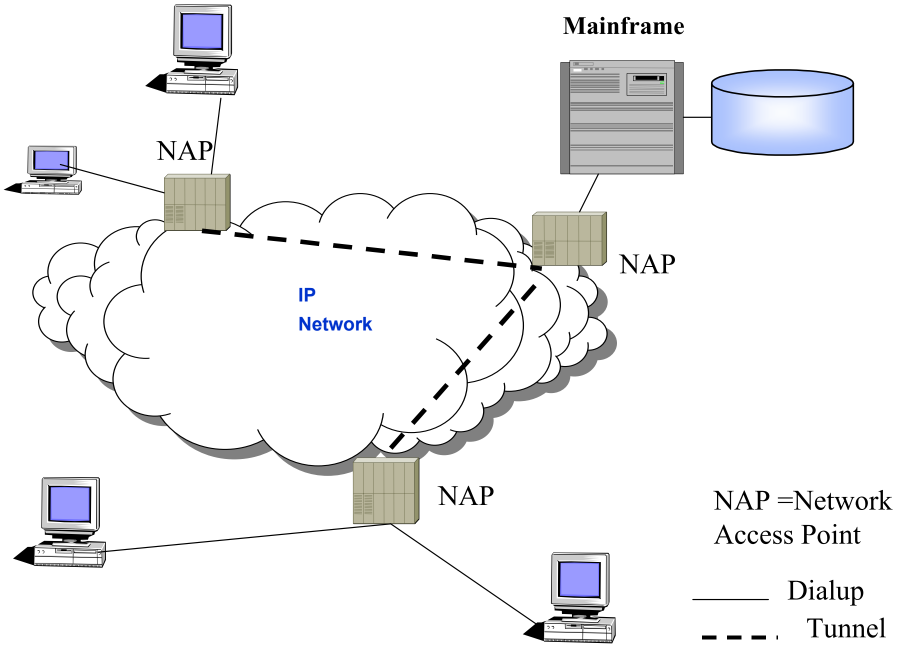


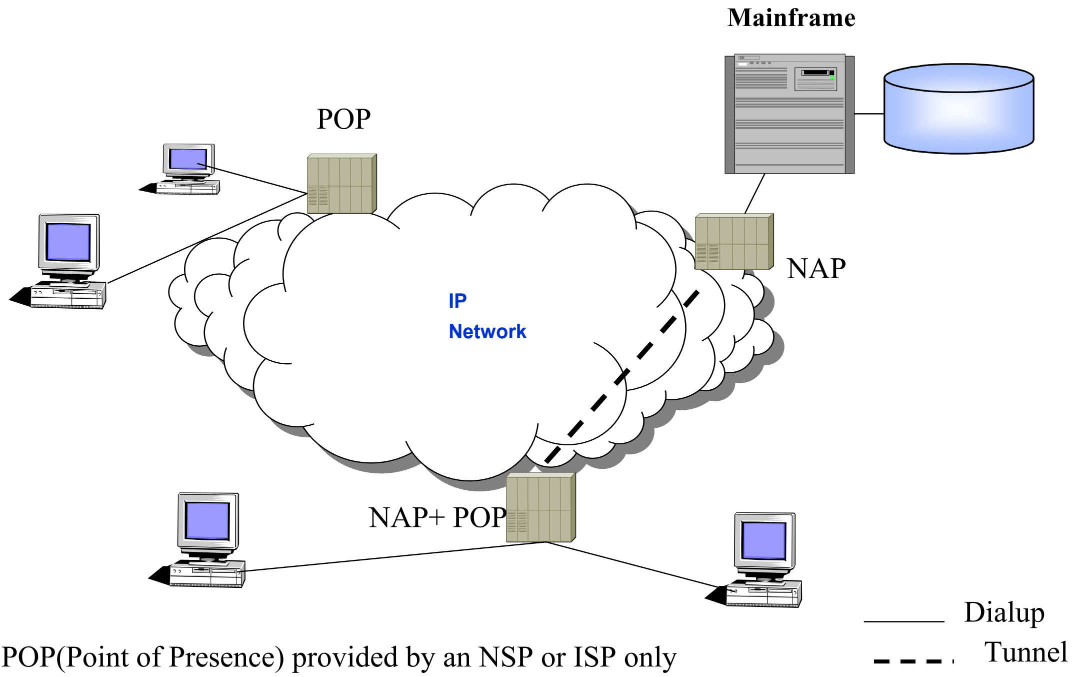
Figure A-37: A VPN

Early attempts to provide VPN remote access involved simply encrypting every packet. They employed encryption hardware that encrypted and compressed data before it traveled on a shared data network. Current typical VPN configurations establish a secure tunnel between the NAP server and a tunnel-terminating device on the local network. The NAP server resides at the ***local point of presence (POP)*** that allows you to make a local call. An ISP or a network service provider may own a POP and a NAP (an ISP only needs to have a NAP if it wants to support VPN). A user initiates a dial-up session to a local POP, where a NAP server authenticates the user and then establishes a tunnel through its Internet "cloud," which terminates at the edge of the user's corporate network. The IP packets are encapsulated in a tunneling protocol such as PPTP or L2F, and these packets are, in turn, packaged by an IP packet containing the address of the corporate network, the packet's ultimate destination. Note that in this case the NAP server at the POP assigns the user an IP address. The encapsulated packets can be encrypted end-to-end using IP-Sec or an equivalent protocol. All packaging/unwrapping and encryption/decryption is transparent to the end user.

VPN users have basically two choices: install VPN software at their machine site or use VPN capabilities of an ISP. With a VPN-enabled client, the users install software on their laptops and basically develop an end-to-end tunnel. The advantage of this *Internet service provider-*

independent configuration is that mobile users can dial into any traditional POP to establish a VPN tunnel to a corporate network, independent of their contracted service provider. If the software is not embedded in the client, an *ISP-dependent model, the* participating ISPs are required to support VPN technology in the NAP server. The choice between the service provider-dependent and -independent models depends on port availability, backbone performance and client deployment. These considerations are beyond the scope of this book. Visit the VPN Consortium website (www.vpnc.org) for a detailed discussion of the tradeoffs.

Currently, a large number of companies offer VPN services. Examples are Shiva, telecommunication companies (e.g., Southwestern Bell and Nortel), and network service providers such as UUNET. Additional information about VPN can be found at the VPN Consortium website (www.vpnc.org). Figure A-38 gives a view of how VPN can be offered by an NSP.



•POP(Point of Presence) provided by an NSP or ISP only provides a local phone access

•NAP (Network Access Point) provides a secure tunnel over a shared network to support a VPN. An ISP may own a NAP also.

Figure A-38: A VPN as a service from an NSP

Time to Take a Break

✓ • Internet, Intranet, and Extranets
• The IP Stack
• Next Generation IP and Internet

Suggested Review Questions Before Proceeding

- What exactly is the Internet and what are Intranets and extranets?
- How has the TCP/IP stack evolved into such a powerful tool? Give some highlights.
- What are the Internet standards bodies and what do they do?
- What are ISPs, NSPs, and VPNs? Explain though examples.

.

A.11 The Internet Protocol (IP) Suite – A Quick Tour

A.11.1 Overview

Transmission Control Protocol (TCP) and Internet Protocol (IP) are the best known protocols in the IP stack (they operate roughly at layers 3 and 4 of OSI). Over the years, the entire ARPANET Protocol Suite has become known as the ***TCP/IP Protocol Suite***. As stated previously, TCP/IP has dramatically grown in popularity in the last 20 years and is supported very widely. For example, TCP/IP can be used to transfer files between IBM, SUN, HP, IBM PCs and several other machines. IP, the lowest protocol in this Suite, can reside on a very wide variety of physical networks such as Ethernets, FDDI based fiber optic LANs, dial up lines, X.25-based packet switching networks, wireless networks, or ISDN digital networks. TCP, the layer above IP, supports a very wide variety of higher level (application) protocols which allow users to emulate terminals, transfer files, and send/receive mail between different computers (see Figure A-39).

Due to its popularity, the TCP/IP Protocol Suite continues to evolve. The TCP/IP Suite originally consisted of five basic protocols: IP, TCP, FTP, Telnet and SMTP. Later, Domain Naming Services (DNS) and Simple Network Management Protocol (SNMP) were added to the TCP/IP basic protocols. In addition, many other protocols and user applications have been developed around TCP/IP. The TCP/IP Suite addresses the layer 3 and above issues. The Application Layer of this network architecture provides a rich set of file transfer, terminal emulation, network file access and electronic mail services. It is important to note that a user application may choose to use any of the TCP/IP layers or may directly communicate with the physical network. Let us review the key protocols.

A.11.2 The Internet Protocol (IP)

The Internet Protocol (IP) is the lowest layer protocol defined in the IP Suite. It runs on top of whatever protocols are in use in the physical network (Ethernet, X.25, token ring, serial, ATM, Frame Relay, wireless networks, etc.). IP connects hosts across multiple networks (Internet) and provides a way of moving a block of data from one host machine to another through the Internet. This block of data is known as a ***datagram.***

The delivery of datagrams is made possible by assigning an IP address to every host in the Internet. These addresses are 32 bits in length and are commonly denoted as four decimal numbers separated by periods (e.g., 21.152.214.2). The first part of the address shows which network the host resides on, and the rest of the address shows where within that network the host can be found.

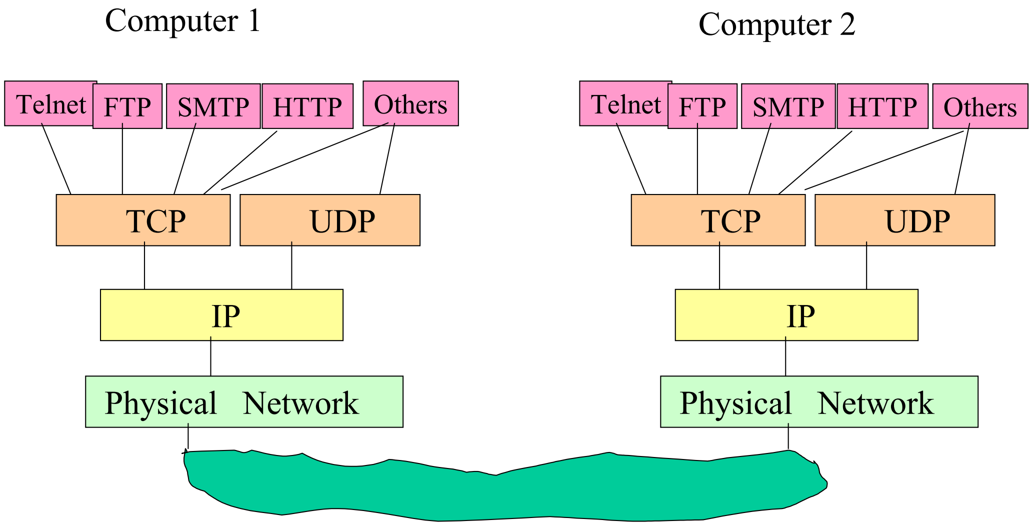


Figure A-39: TCP/IP Network --- The Initial Stack

IP is an unreliable (connectionless) protocol. This means that datagrams sent from one host to another may not be delivered in the order in which they were sent, may be delivered more than once, or may not be delivered at all. Higher layer protocols are expected to correct this deficiency. Unreliable protocols are much simpler and cleaner to implement and facilitate dynamic routing (route around problems) of datagrams within the networks.

The current IP (technically known as IPv4) is more than 20 years old. IPv4 has been remarkably resilient in spite of its age, but it is beginning to have problems. Most importantly, there is a growing shortage of IPv4 addresses, which are needed by all new machines added to the Internet. The ***next generation of the Internet Protocol (IPv6)*** is intended to support Internet traffic for many years into the future by providing enhancements over the capabilities of the existing IPv4 service. IPv6 corrects a number of problems in IPv4, such as the limited number of available IPv4 addresses. It also adds many improvements to IPv4 in areas such as routing and network autoconfiguration. IPv6 is expected to gradually replace IPv4, with the two coexisting for a number of years during a transition period. IPv6 is an area of intense activity at the IETF (Internet Engineering Task Force).

A.11.3 The Transmission Control Protocol (TCP)

The Transmission Control Protocol (TCP) runs on top of IP and is by far the most heavily used transport protocol in the Internet. It provides a reliable, ordered connection between processes on different hosts. One host may run many processes, so a process to process connection is needed. This means that application processes can establish a TCP connection and expect that data will arrive successfully and in order.

A TCP connection is essentially an error-free pipe from one host process to another. There is no inherent meaning at this layer to the data sent over this pipe. This generality allows a variety of higher layer protocols to run on top of TCP. The fact that this protocol is connection based means that some overhead is incurred at connection setup and disconnect. This is appropriate for applications that need to send large amounts of data to one place, but it might not be so appropriate for quick, short exchanges.

A.11.4 The User Datagram Protocol (UDP)

The User Datagram Protocol (UDP) also runs on top of IP and is an alternative to using TCP. Like IP, UDP is an unreliable protocol. In fact, the major function that UDP adds to IP is a way to differentiate more than one stream of data going to or from a host (IP addresses only identify the hosts and not the processes within a host). Due to the unreliability of UDP, it is up to higher layer protocols running on top of UDP to provide reliability if it is needed.

UDP is appropriate for applications which exchange a small amount of information, such as a single request and a reply to it. In such applications, the overhead of establishing a connection is unnecessary. UDP may also be appropriate in applications requiring the exchange of data with more than one host. Since every UDP datagram is individually addressed, a host can talk to as many other hosts as necessary without having to establish a TCP connection to every one of them.

A.11.5 Additional Transport Layer Protocol - SCTP (Simple Control Transmission Protocol).

SCTP was designed originally as the transport layer for telephone signaling (ISUP, Q.931, TCAP, etc.) over IP networks, since there are intrinsic limitations for TCP for this job. But as the work progresses, the IETF Transport Area Working Group is paying more and more attention to it. It has long been recognized by the Internet community that TCP, which serves most IP applications well until recently, is limited in certain aspects for real time applications and a "Next-Generation TCP" will be needed. IETF is looking at SCTP as a candidate.

A.11.6 Sample Higher (Application) Layer Protocols

Many higher layer protocols run on top of TCP and UDP. It is also possible to define private application protocols as long as both hosts agree on the protocol. The following protocols (the first three belong to the original DOD Suite) are among the best known application protocols defined in the IP Suite.

- **Telnet:** This protocol is used to provide terminal access to hosts and runs on top of TCP**.**
- **File Transfer Protocol (FTP):** This TCP based protocol provides a way to transfer files between hosts on the Internet.
- **Simple Mail Transfer Protocol (SMTP):** This TCP based protocol is the Intermet electronic mail exchange mechanism.
- **Trivial File Transfer Protocol (TFTP):** This UDP based protocol also transfers files between hosts, but with less functionality (e.g., no authorization mechanism). This protocol is used typically for "booting" over the network.
- **Network File System (NFS)** Protocol: This UDP based protocol has become a de facto standard for use in building distributed file systems through transparent access.
- **Xwindow**: This is a windowing system that provides uniform user views of several executing programs and processes on bit-mapped displays. Although Xwindow is supposedly network independent, it has been implemented widely on top of TCP**.**

- **SUN Remote Procedure Call (RPC):** This protocol allows programs to execute subroutines that are actually at remote sites. RPCs, like Xwindow, are supposedly network independent but have been implemented widely on top of TCP. SUN RPC is one of the oldest RPCs. Examples of other RPCs are Netwise RPC and OSF RPC. RPCs are described in detail in chapter 5.
- **Domain Naming Services**: This protocol defines hierarchical naming structures which are much easier to remember than the IP addresses. The naming structures define the organization type, organization name, etc.
- **Time and Daytime Protocol**: This provides machine readable time and day information.
- **SNMP (Simple Network Management Protocol**): This is a protocol defined for managing (monitoring and controlling) networks.
- **Kerberos:** This is a security authentication protocol developed at MIT.
- **RTP (Real Time Protocol).** A protocol for voice and video applications.
- **HTTP (Hypertext Transfer Protocol).** Used to transfer files in Web.
- **H323.** Protocol to support voice over IP (proposed by ITU-T)
- **SIP (Session Initiation Protocol).** Protocol for voice over IP (proposed by IETF)
- **N3270.** IP-based terminal emulator for 3270

Other frequently used services in IP are Ping (an echo command), Netstat (command to display the network status of the local host, e.g., active TCP connection and IP routing tables), and Finger (displays information about users of a remote host, e.g., list of all users logged on to the remote host). In addition, the OSI upper layers can be implemented on TCP/IP as specified in the RFC1006.

A.11.7 Common TCP/IP Implementations

TCP/IP has been implemented on most machines under the Unix operating system. In many Unix systems such as SUN, TCP/IP is shipped as part of the operating system. TCP/IP is also becoming available on many other operating systems. An interesting example is the availability of TCP/IP on IBM mainframes. The IBM mainframes use two operating systems: MVS and VM. TCP/IP is currently available on MVS as well as VM. These implementations provide NFS and Xwindow support in addition to the common Internet Application Layer protocols. MVS-NFS is especially interesting because it allows MVS files to be accessed transparently from Unix workstations. This is a major step toward file transparency in most corporations.

A.12 Internet Protocol (IPv4) – A Closer Look

This section takes a closer look at IP with emphasis on the widely used IPv4. We will discuss IPv6 in a later section and point out the differences between IPv4 and IPv6 at that point.

A.12.1 Internet Architecture and Routers

The Internet, as stated previously, consists of several IP-based physical networks (network of networks) that are interconnected. The physical networks, also called local networks or subnets, may be LANs, WANs or MANs. An example of Internet (IP network) is shown in Figure A-40. This IP network connects many disparate physical networks into a coordinated unit and hides the details of the physical network hardware; it allows computers to communicate independent of their physical network connections.

Internet ***routers***, sometime also referred to as internet gateways, are used between physical networks of an IP network and perform the functions of a relay that directs traffic to its destination. Routers are software programs, typically residing in dedicated computers, that shuffle messages between physical networks. For example, the routers shown in Figure A-40 pass the messages between the four networks. In general, an outgoing message from a host first checks to see if its destination is on the same physical network. If its destination is not on the physical network, then it goes to a router for routing.

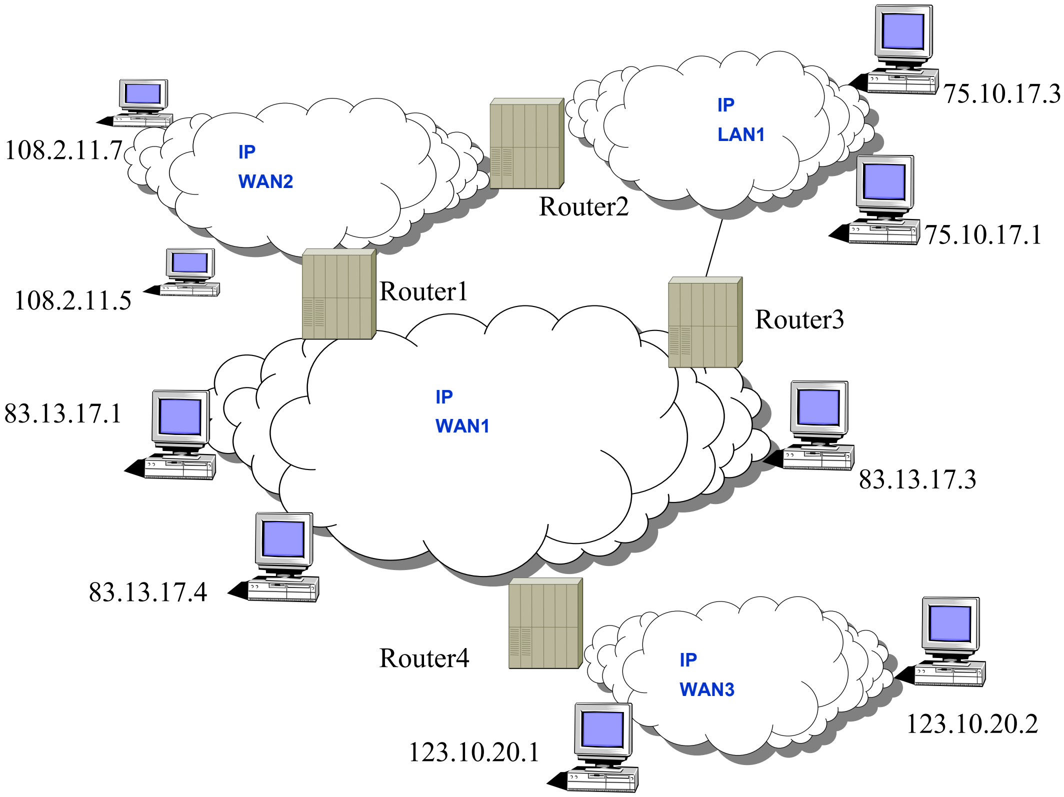


Figure A-40: A Sample IP Network

The role of a router becomes more complex as the complexity of the IP network grows. In a complex network, the routers must understand the network topology and must know how to get to the next router. For example, Router4 in Figure A-40 must know how to pass messages from WAN3 to LAN1 through the intermediate routers. In all cases, routers are responsible for routing messages to a destination network and not to a destination host. In most cases, routers are dedicated computers which house the routing tables. The size of the routing table depends on the number of physical networks and not on the number of computers in an Internet. We will consider the Internet routing in Section A.12.3.

A router may be a core router which is maintained by a recognized authority. An example of a core router is the router that connects the “backbone” Internet, maintained by IAB, to local nets. Other routers can be introduced by owners of a private Internet or other groups of the Internet Community (IC). The IC spans the United States and extends to most foreign countries as well. At present, many networks using TCP/IP, including most university networks, are part of the IC. The IC connection allows the exchange of files and mail as well as terminal connections for all IC users. There are many ways to establish a connection between a local area network and the IC, and these communications options are expanding rapidly. For example, IC links for Windows, Unix, and IBM mainframe networks are currently available.

Several algorithms for routing between gateways have been proposed and are currently operational. Some of these algorithms are used in the core routers while others are used by private gateways. Discussion of these algorithms is beyond the scope of this book. The interested reader can find more information on this topic in several IETF RFCs. A good overview is presented in [Comer 2000, Tannenbaum 1996].

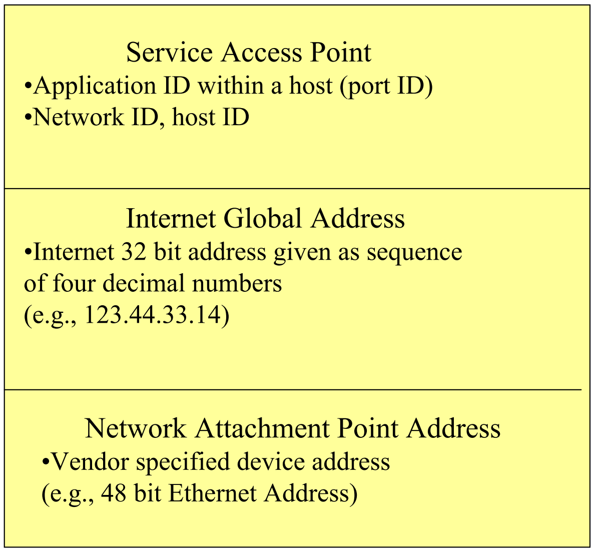
A.12.2 Internet Addressing in IPv4

Every host and router in the Internet has an IP address, which encodes its network number and host number. The Internet address is a 32 bit address commonly denoted as four decimal numbers separated by periods, e.g., 125.102.112.5. The IP address is unique: no two machines have the same IP address. If an enterprise cannot present globally unique addresses to the Internet, it may be forced to deploy private, isolated address space that is not visible to the Internet. Users in private address spaces with non-unique addresses typically require gateways, and possibly ***Network Address Translators (NATs)***, to manage their connectivity to the outside world. In such situations, some services are simply not available. A NAT is meant to allow an enterprise to have whatever internal address structure it desires, without concern for integrating internal addresses with the global Internet. This is seen as particularly convenient in the existing IPv4 world, with its more cumbersome address space management. The NAT device sits on the border between the enterprise and the Internet, converting private internal addresses to a smaller pool of globally unique addresses that are passed to the backbone and vice versa.

How do the IP addresses relate to application (higher level) and physical device addresses (lower level). The IP addressing scheme allows different networks, hosts and applications to identify the sources and destination of messages in a network. The addressing scheme in the Internet has three levels (see Figure A-41):

- **The network attachment point address**, which identifies the physical location (e.g., an Ethernet controller address) where the host is connected to a physical network. This address is unique within a physical network and conforms to the addressing scheme of the physical network (e.g., 48 bit for Ethernet, 16 or 48 bit for a token ring, and 10 decimal digits in a packet switching network).
- **The Internet address,** also called the global network address, which shows the (network ID, host ID) pair for identifying hosts attached to different physical networks in an Internet. The Internet address, as stated previously, is a 32 bit address commonly denoted as four decimal numbers separated by periods, e.g., 25.102.142.5. This address identifies the location of the host within an Internet: the first part of the address tells which network the host resides on, and the rest of the address shows where within that network the host can be found. A registration authority, Network Information Center, is responsible for assigning the IP addresses on the Internet.
- **The service access point (SAP**), which identifies an application process within a host connected to the Internet. An SAP allows a complete application to application communication through a three level address (network ID, host ID, application ID). Thus an application A1 on host H1 connected to NET1 can send a message to an application A2 on host H2 connected to NET2. Many applications at H1 can simultaneously communicate with applications at H2. In practice, an SAP appears as a port number in TCP (e.g., TCP port for FTP and Telnet) or UDP.

Schemes are needed to map the higher level application addresses to lower level device addresses. The mapping of an Internet address to a network attachment address is commonly done by an ***address resolution protocol (ARP).*** A common method used by ARP includes tables at each machine which contain pairs of Internet and network attachment addresses. An address table is created by the installer of IP software at a host. Other methods encode physical addresses in the Internet address. Details about ARP can be found in RFC826. The translation from SAP to Internet address is performed by the TCP software. TCP pre-assigns some port numbers for commonly used applications (e.g., FTP, Telnet) and allows other applications to use "free" ports for communications.



Service Access Point
•Application ID within a host (port ID)
•Network ID, host ID

Internet Global Address
•Internet 32 bit address given as sequence of four decimal numbers (e.g., 123.44.33.14)

Network Attachment Point Address
•Vendor specified device address (e.g., 48 bit Ethernet Address)

Figure A-41: Address Levels in IPv4

A.12.3 Internet Routing in IPv4

The Internet uses a connectionless (unreliable) delivery protocol. This protocol is called unreliable because delivery is not guaranteed. The packets may be lost, duplicated or delivered out of order but the Internet will not detect such conditions. The protocol is called connectionless because it does not rely on an already established connection. It is the responsibility of higher layer protocols to establish connections and to verify the correct delivery of messages.

The IP sends each message as a ***datagram*** which is its basic unit of information transfer. Each datagram contains a header and data areas. The datagram header contains the IP addresses of the sender and receiver. Figure A-42 shows a typical datagram format. Some of the main fields in the datagram are as follows:

- Two different fields, version and length fields, show the IP version and the length of the datagram header. The total length field gives the total size of the datagram in bytes.
- The service type fields are used to choose the quality of routing service needed: low delay, high throughput and high reliability. Setting these bits does not guarantee this type of service, it just tells the routing algorithm what your preferences are.
- The fragmentation control fields are used to show if you, or a gateway along the path to the destination, wish to break your datagram into fragments (packets), which can be

assembled on the other side. You can choose to avoid fragmentation altogether. Fragmentation allows you to send long datagrams (longer than the maximum transmission unit of any single physical network along the path).

- The time field shows the maximum time a datagram can survive in the network. This field, usually represented as the maximum number of gateway hops, forces datagrams to be deleted from the Internet so that they do not roam around forever (a datagram may be roaming around due to problems with a routing protocol).
- The checksum is used to ensure the integrity of header values.
- The source and destination IP addresses obviously show the sender and receiver of the datagram. These addresses use the 32 bit Internet addresses which show the network and host IDs.

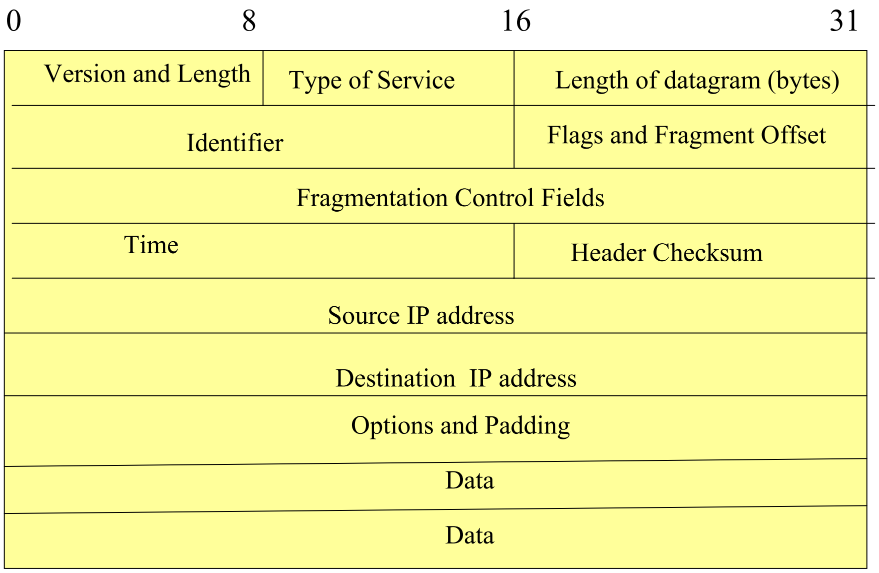


Figure A-42: Format of an Internet Datagram (in bits) for IPv4

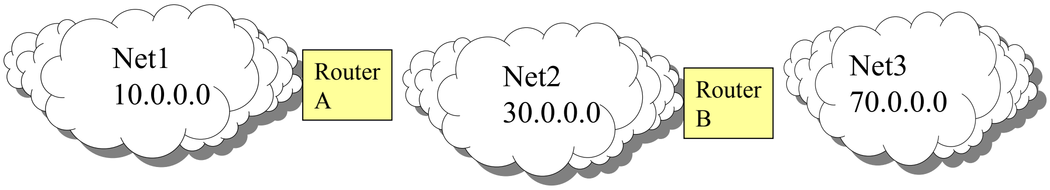
IP supports several service options for security, performance and fault management. For example, IP allows security labeling (classification level, handling restrictions, etc.) for security control, route recording (gateways encountered) for debugging, and timestamping (time trace of a datagram) for performance monitoring. These choices are specified in the options field of the datagram.

The datagrams are routed through the Intenet by using a variety of routing algorithms. The choice of the routing algorithm depends on the nature and complexity of the Internet. The IP routing occurs at a higher level (global) than the physical network routing in a subnet. The IP routing is responsible for transferring messages between the physical networks of an Internet. The physical networks (subnets) are interconnected through gateways. The routing of a message within a physical network (e.g., an Ethernet) is the responsibility of network routing (the data link protocol) mechanism of the network.

The route can be direct (within a physical network) or indirect (between networks through gateways). Internet routing algorithms usually employ routing tables which show possible destinations. An example of a routing table for a simple Internet is given in Figure A-43. A typical routing algorithm used in IP is as follows:

- Extract the destination address (DA) from the datagram.

- Find the route for DA from the routing table.
- If DA is a direct path (within this subnet), send the message directly.
- If DA is an indirect path, send the message to the proper subnet or gateway.
- If none, then give a routing error.



a) A Simple IP Network

Destination Network	Route Information
10.0.0.0	Direct
30.0.0.0	Direct
70.0.0.0	Indirect (route to gateway B, address 30.0.0.0)

b) Routing Table for Router A

Figure A-43: Example of a Routing Table

Briefly, the Internet datagrams travel from router to router until they reach a router that can deliver them directly to the destination host. If a router cannot route or deliver a datagram due to an addressing problem or congestion, it needs to instruct the host to take action. The mechanism commonly employed to communicate the errors is the ***Internet Control Messages Protocol (ICMP).*** ICMP messages travel across the Internet in the data portion of the datagram just like all other traffic. The destination of an ICMP message is the IP software on the destination machine and not the application process. Basically, ICMP provides a communication mechanism between the IP software at various machines (hosts and/or gateways) in the networks. ICMP is considered a required part of IP. More details about ICMP can be found in the DARPA standard RFC 792.

Routing in the Global Internet. How does Internet routing work in the large global Internet with millions of computers? The current IPv4 system uses an address hierarchy, similar to telephone area codes, to sort traffic towards networks attached to the Internet backbone. Without an address hierarchy, backbone routers would be forced to store route table information on the reachability of every network in the world. Given the current number of IP subnets in the world and the growth of the Internet, it is not feasible to manage route tables and updates for so many routes. With a hierarchy, backbone routers can use IP address prefixes to determine how traffic should be routed through the backbone. Hierarchical address systems work in much the same way as telephony country codes or area codes, which allow long-haul phone switches to route calls efficiently to the correct country or region using only a portion of the full phone number.

IPv4 does not explicitly support hierarchical addresses like the telephone system. IPv4 was initially designed with class A, class B, and class C addresses, which divided address bits between network and host but did not create a hierarchy that would allow a single high-level address to represent many lower-level addresses. (see the sidebar “IPv4 Address Classes”). To simulate hierarchy, IPv4 started to use a technique called ***Classless Inter Domain Routing (CIDR)*** which uses bit masks to allocate a variable portion of the 32-bit IPv4 address to a network, subnet, or host. CIDR permits "route aggregation" at various levels of the Internet hierarchy, whereby backbone routers can store a single route table entry that provides reachability to many lower- level networks.

IPv4 Address Classes

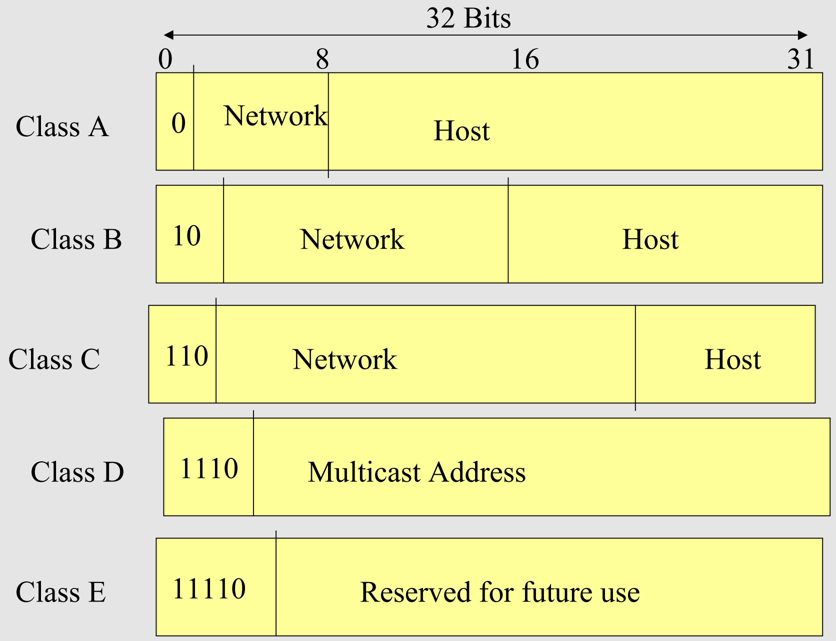
Every host and router in the Internet has a 32 bit IP address, which encodes its network number and host number. The address is a 32 bit address commonly denoted as four decimal numbers separated by periods, e.g., 113.102.142.5. The lowest address is 0.0.0.0 and the highest is 255.255.255.255.

IPv4 was initially   designed with class A, class B, and class C addresses, which divided address bits between network and host.   These address formats are shown below. Notice the tradeoff between network address space versus the host address space.

Class D addresses were introduced to support ***multicasting.*** In multicasting, you send the information to a large number of receivers simultaneously. Each class D address identifies a group of hosts. When a process sends a packet to a class D address, the message is delivered to the entire group.

Class E addresses are reserved for future considerations.

Network numbers are assigned by the ***NIC (Network Information Center)*** to avoid confusion.



A.13 Transmission Control Protocol (TCP) – A Closer Look

TCP is a reliable and connection-based protocol that resides above IP. It is most widely used (all Web applications use TCP because HTTP is built on top of TCP). Although TCP is usually discussed as a protocol on top of IP, TCP is a general purpose protocol that can be used on other delivery systems. For example, TCP can directly run on top of an Ethernet LAN, dial up telephone lines, a high speed fiber optic network, a packet switching system or a slow speed serial connection. The main strength of TCP is the large number of delivery systems it can run on (this, however, requires global addressing). It should be emphasized that TCP is a protocol and not a software package. Basically, TCP defines a set of rules and message formats for reliable service which have been implemented in the TCP software packages.

TCP allows many applications on one host to communicate concurrently with other applications on another host. The concept of a ***port***: is introduced to specify an endpoint in a host. Each port is a unique address within a host. Thus the TCP traffic sender and receiver addresses are given by the host Internet address plus the port number within the host. TCP has many ports reserved for specific services while other ports are assigned dynamically to the applications. Table A-1 shows some common port numbers in TCP. It can be seen that FTP, SMTP, Telnet, and several other applications have pre-assigned TCP port numbers.

Table A-1: Examples of Assigned TCP port Numbers

Port No	Keyword	Description
0		Reserved
1-4		Not assigned
5	RJE	Remote job entry
7	Echo	Echo port
11	Users	No. of active users
13	Daytime	Daytime
15	Netstat	Network Status
20	FTP Data	FTP data send/receive
21	FTP	FTP session management
23	Telnet	Terminal logon/logoff
25	SMTP	Simple Mail Transport Protocol
53	Domain	Domain name server
80	HTTP	HTTP default port for Web

The two endpoints (ports) are essential in TCP because TCP is a connection oriented protocol (UDP also has ports). The applications on both sides must establish a connection between the two ports before TCP traffic can be initiated. Communication between two applications using TCP commonly involves the following steps:

- Application A at host H1 performs a "passive open" by indicating that it will accept an incoming connection. This request is sent to the operating system at H1 which assigns a port number at H1. The passive open can be specific (listen for a specific user) or unspecific (listen to any user such as print or file user).
- Application B at host H2 performs an "active open" to establish a connection with A at H1.
- The TCP software modules at H1 and H2 establish and verify the connection.
- The application A can now send and/or receive data from B.

TCP divides the application data into segments, where each segment has a sequence number and travels as an Internet datagram. To ensure reliable communication, TCP uses a positive acknowledgement with timeout algorithm. This algorithm consists of the following steps:

- The sender sends a message.
- The receiver receives a message and sends an acknowledgement.
- The sender receives the acknowledgement and sends the next message.
- If the sender does not get an acknowledgement within a specified time period, it times out and resends the message.

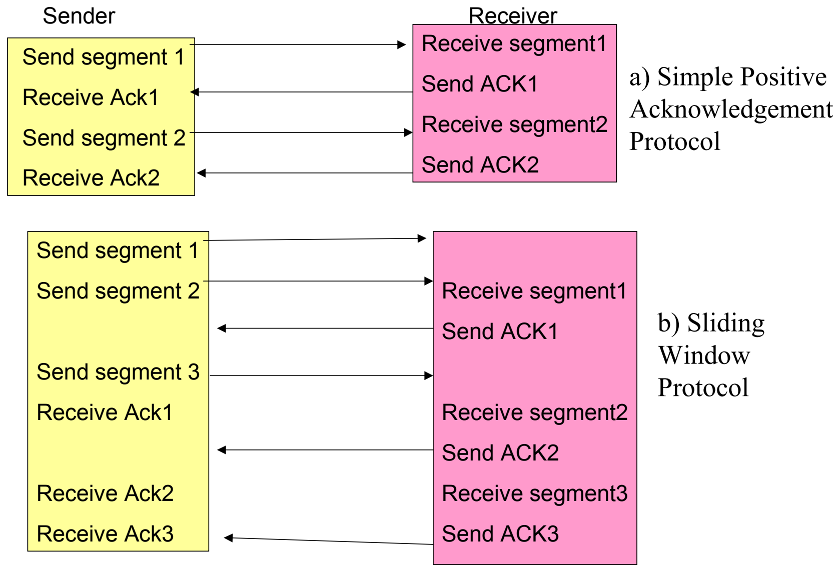


Figure A-44: Sliding Window Versus Simple Acknowledgement

To improve efficiency of network communication, TCP uses a sliding window protocol. Simply stated, a sliding window allows the sender to keep on sending n data units (n is the sliding window size) before waiting for an acknowledgement. Figure A-44 illustrates the difference between a positive acknowledgement (stop-and-wait) and sliding window protocol. This is a much more efficient protocol because the sender does not have to wait for acknowledgement of a message before sending the next message. In the limiting case when the window size is 1, the sliding window protocol becomes a simple positive acknowledgement algorithm.TCP uses a variable size sliding window for flow control (see appendix A). For example, if the receiver buffer is getting full, it may give the sender a small window size to minimize incoming traffic. In the limiting case, a sliding window size of zero stops incoming traffic. Adequate flow control is essential in an Internet environment because in such an environment several machines with different speeds and capacities are interconnected.

As stated previously, TCP uses segments as the basic unit of information transfer between TCP modules. Segments are exchanged to establish connection, to send data, to send acknowledgements, to send window size, and to close a connection. The format of a TCP segment is shown in Figure A-45. The main fields of interest in the TCP segment are as follows:

- The Source Port and Destination Port show the sender and receiver application programs.
- The Sequence Number shows the starting position of data in the sender's segment and the Acknowledgement Number shows the position of the last byte received (next byte expected). These two fields are used to synchronize the positions of data sent and received.
- The Flags are used to show the type of data being sent in the segment (information data, acknowledgement, command, etc.).
- The Window parameter is used for flow control. This parameter sets the sliding window size and is used to tell the sender how much data to send.
- The Urgent indicator is used to tell TCP to deliver this segment before others.

TCP does leave several issues for implementers of TCP software. For example, an implementer can choose to use piggybacking of acknowledgements with data, use timers to retransmit unacknowledged data and send one byte at a time instead of maximum segment size, etc. Additional technical details about TCP are given in the DARPA RFC 793 by Postel. For detailed coverage, the books [Comer 2000, Tannenbaum 1996] are recommended.

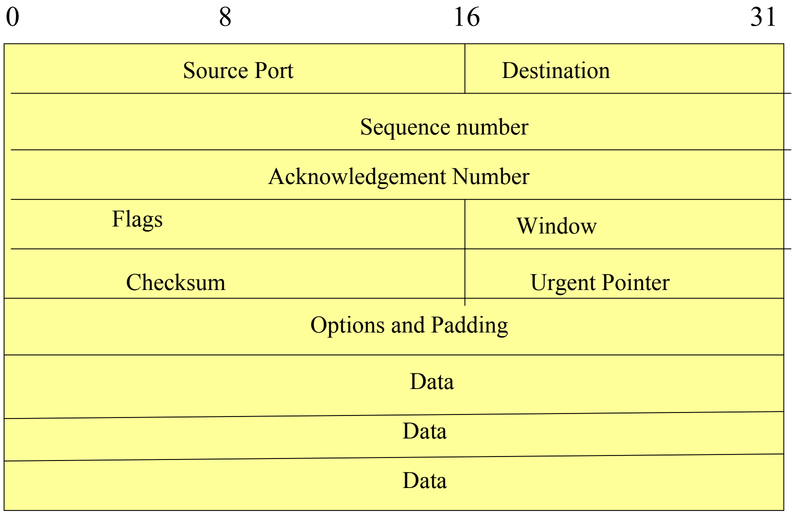


Figure A-45: Format of a TCP Segment

A.14 Application Layer Protocols in TCP/IP

A.14.1 Overview

A rich set of higher-level (application) protocols run on top of TCP or UDP. These include, as indicated previously, the older basic protocols such as FTP and Telnet plus the newer more popular protocols such as HTTP. All higher-level protocols have some common characteristics:

- They can be standardized and shipped with the TCP/IP product. For example, the TCP/IP Suite includes application protocols such as Telnet, FTP and SMTP. These are the most widely implemented application protocols, but many others exist. Each particular TCP/IP implementation includes a set of application protocols.
- They use UDP or TCP as a transport mechanism. Recall that UDP is unreliable and offers no flow-control; so in this case, the application has to provide its own error recovery and flow-control routines. It is easier to build applications on top of TCP, a reliable, connection-oriented protocol. Most application protocols use TCP, but many applications are built on UDP especially for higher performance of connectionless services.
- Most use the client-server model of interaction in which one host acts as a client and the other as a server. The client hosts send a request over the Internet that is received and processed by the server. The tasks performed by a server can be simple or complex. For example, a time-of-day server simply returns the current time whenever it receives a client request; a file server receives requests to perform file reads/writes and returns the results.

Section **Error! Reference source not found.** (Attachment A) takes a closer look at the traditional Internet application protocols such as FTP, Telnet, and NFS. The following section highlights the end-user aspects through an example.

A.14.2 An Example of Traditional Internet Applications

Figure A-46 shows a simplified view of an IP network. The network consists of an Ethernet IP LAN and an ATM IP WAN, interconnected through a router. A Unix minicomputer and a PC-Windows desktop are connected to the LAN and an MVS machine is connected to the WAN (there may be several other devices, but we are showing these three just to highlight key points). Each computer ("host") on this network has an IP address and also has been assigned a domain name (e.g., joe.college1.edu, sam.college1.edu, Tom.bank3.com). This IP network is very heterogeneous (different computers, different physical networks). However, to the users of this network, it provides a set of uniform TCP/IP services (TCP/IP hides many details). Let us illustrate the use of this network.

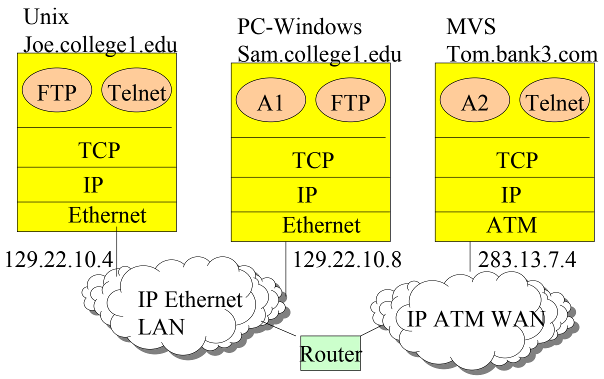


Figure A-46: A Sample Internet Configuration

Let us assume that the Unix user needs to transfer a file from the PC ("sam.college1.edu"). The user would use the following steps (the steps are explained through comments in /* */):

```
joe> ftp sam.college1.edu       /* invoke FTP. Could have typed " ftp 129.22.10.8"*/
sam> enter logon: umar         /* prompt from sam for logon ID. umar is ID */
sam> password: xxxx           /* prompt from sam for password  */
sam>  get file1  file2          /*  FTP file transfer command  */
sam> quit                         /* quit  FTP  */
```

Now let us assume that the Unix user needs to remotely logon to the machine "tom" to run a program "account". The user would use the following steps (the steps are explained through comments in /* */):

```
> telnet tom                      /* invoke Telnet. Could have typed " telnet 283.13.7.4".*/
tom> enter logon: umar         /* prompt from tom for logon ID. umar is ID */
tom> password: xxxx           /* prompt from tom for password  */
tomr>  account            /*  run the program "account"   */
tom> quit                         /* quit  telnet */
```

A client/server application runs between the PC and the MVS machine (A1 is the client and A2 is the server). A1 and A2 use TCP/IP sockets to exchange information over the TCP/IP network. To use this application, the following steps would be needed:

start A2 on tom

start A1 on sam

use A1 (this usage will automatically send the requests to A2)

Notice that the user does not know anything about the underlying network technologies to remotely logon, transfer files and invoke client/server applications. The user only needs to know the address of the host to perform interactions.

Time to Take a Break

✓• Internet, Intranet, and Extranets

✓• The IP Stack

• Next Generation IP and Internet

Suggested Review Questions Before Proceeding

- What are the key participants of the TCP/IP stack and what do they do?
- Why is IP such an important player in the Internet?
- What is the basic difference between TCP and UDP?
- List some of the applications that have been developed on top of TCP as well as UDP
- What are the limitations of IPv4?
- How does the IP routing really work when you send an email (SMTP) from your home in New York to your friend in Thailand?

A.15 References

Berkowitz, H., "Designing Routing and Switching Architectures for Enterprise Networks", Macmillan Technical Publishing, 1999.

Bertsekas and Gallager, "Data Networks", second edition, Prentice Hall, 1992.

Bernstein, L., and Yuhas, C.M., "Network Architectures for the 21st Century", IEEE Communications, Jan. 1996, pp. 24-30.

Bird, D., "Token Ring Network Design", Addison Wesley, 1994.

Bisdikioan, C., "A Performance Analysis of the IEEE 802.6 (DQDB) Subnetwork with the Bandwidth Balancing Mechanism", Computer Networks and ISDN Systems, Vo. 24, No. 5, 1992, pp. 367-386.

Black, U., "Emerging Communications Technologies", Prentice Hall, 1997.

Black, U. D., "Data Communications, Networks and Distributed Processing", 2nd ed., Reston Publishing Co., 1987.

Black, U., "Advanced Internet Technologies", Prentice Hall, 1998.

Carlo, J., et al "Understanding Token Ring Protocols and Standards", Artech, 1998.

Charles, G., "LAN Blueprint: Engineering It Right", McGraw Hill, 1997.

Christiansen, P., "Networking with Novell NetWare ", Computer Science Press, 1991.

Clayton, J., "McGraw-Hill Illustrated Telecom Dictionary" McGraw Hill, 1998.

Drefler, F., "PC Magazine Guide to Connectivity", 3rd edition, Ziff-Davis, 1995.

Dodd, A. “The Essential Guide to Telecommunications”, Prentice Hall, 2nd edition, 1999.

Duran, J. and Visser, J., "International Standards for Intelligent networks", IEEE Communications Magazine, Feb. 1992, pp. 34-43.

Felt, S., “Wide Area High Speed Networks”, Cisco Systems, 1999.

Gareiss, R., "X.25 Packet Switching Goes Mobile", Data Communications Magazine, Sept. 1994, pp. 41-42.

Hoffert, E. and Gretson, G., "The Digital News System at EDUCOM: A Convergence of Interactive Computing, Newspapers, Television and High-Speed Networks", Comm. of the ACM, April 1991, pp.113-116.

Hawley, G., "Historical Perspectives on the U.S. Telephone Loop", IEEE Communications, March 1991, pp. 24-30.

Johnson, J., "Enterprise NOSs: Now 's the Time", Data Communications, May 21, 1995, pp. 40-50.

Joeseph, C. and Muralidhar, K., "Network Management: A Manager's Perspective", Enterprise Network Event Conf. Proceedings, Baltimore, May 1988, pp. 5.163-5-174.

Kessler, G., Southwick, P., “ISDN Concepts, Facilities, and Services”, McGraw-Hill, 1998

King, S., "Switched Virtual Networks", Data Communications magazine, September 1994, pp. 66-84.

Marcus, S., “Designing Wide Area Networks and Internetworks: A Practical Guide”, Addison Wesley, 1999.

Martin, J., "Telcommunications and the Computer", Prentice Hall, 5th edition, 1992.

Mccabe, J, “ Practical Computer Network Analysis and Design”, Morgan Kaufmann, 1999.

Muller, J., “Desktop Encyclopedia of Telecommunications”, McGraw Hill, 1998.

Muller, N., “Desktop Encyclopedia of Voice and Data Networking”, McGraw Hill, 1999.

Nassar, D., “Token Ring Troubleshooting Guide”, ToExcel, 1998.

Newton, H., “Newton's Telecom Dictionary”, 15 Ed, Telecom Books/Miller Freeman, 1999.

Panko, R., "Business Data Networking and Telecommunications", (4th edition), Prentice-Hall, 2003.

Parnell, T., “Building High-Speed Networks”, McGraw Hill, 1999.

Ramos, E., Schroeder, Al and Simpson, L., "Data Communications and Networking Fundamentals Using Novell NetWare ", MacMillan, 1992.

Ranum, M., "A Network Firewall", Proc. World Conference on System Administration and Security, Washington, D.C, July 1992.

Reingold, L. and Lisowski, B., "Sprint's Evolution to Broadband ISDN", IEEE Communications, August 1992, pp. 28-31.

Rhodes, P., “Building a Network : How to Specify, Design, Procure, and Install a Corporate LAN”, McGraw Hill, 1995.

Riley, S., Breyer, R., "Switched, Fast, and Gigabit Ethernet," Mtp Network Engineering Series, 1999.

Sandesara, N., Ritchie, G., and Engel-Smith, B., "Plans and Considerations for SONET Deployment", IEEE Communications Magazine, August 1990, pp. 26-34.

Saunders, S., "Traffic Jam at the LAN Switch", Data Communications Magazine, Nov. 1994, pp. 53-58.

Sharma, R., “Network Design Using Econets”, International Thomson Publishing, 1997.

Stallings, W., " Business Data Communications", 4/E, Prentice Hall, 2001.

Stallings, W., " Local and Metropolitan Area Networks", 6/E , Prentice Hall, 2000.

Stallings, W., " Data & Computer Communications", 6/E, Prentice Hall, 2000.

Stallings, W., " ISDN and Broadband ISDN with Frame Relay and ATM", 4/E, Prentice Hall, 1999.

Tannenbaum, A., "Computing Networks", Prentice Hall, 3rd., 1996.

Taylor, S., "Making the Switch to High-Speed WAN Services", Data Communications Magazine, July 1994, pp. 87-94.

Townsend, C., "Networking with the IBM Token-Ring", TAB Books, 1987.

Umar, A., "Distributed Computing and Client/Server Systems", Prentice Hall, Revised, 1993.

Van Norman, H., "WAN Design Tools: The New Generation", Data Communications, Oct. 1990, pp.129-138.

Ward, A., "Connecting to the Internet: A Practical Guide About LAN-Internet Connectivity", Addison-Wesley, 1999.